DGK Ausschuss Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 944

Randa Natraš

Machine Learning for Modeling and Forecasting the Ionosphere Vertical Total Electron Content, Including Space Weather Effects and Uncertainty Quantification

München 2025

Verlag der Bayerischen Akademie der Wissenschaften, München

ISSN 0065-5325 ISBN 978-3-7696-5356-4

DGK Ausschuss Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 944

Machine Learning for Modeling and Forecasting the Ionosphere Vertical Total Electron Content, Including Space Weather Effects and Uncertainty Quantification

Vollständiger Abdruck
der TUM School of Engineering and Design
der Technischen Universität München
zur Erlangung des akademischen Grades einer
Doktor-Ingenieurin (Dr.-Ing.)
genehmigten Dissertation

von

Randa Natraš

München 2025

Verlag der Bayerischen Akademie der Wissenschaften, München

ISSN 0065-5325 ISBN 978-3-7696-5356-4

Adresse des Ausschusses Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften:

Ausschuss Geodäsie (DGK) der Bayerischen Akademie der Wissenschaften
Alfons-Goppel-Straße 11 ● D – 80 539 München
Telefon +49 – 89 – 23 031 1113 ● Telefax +49 – 89 – 23 031 - 1283 / - 1100
e-mail post@dgk.badw.de ● http://www.dgk.badw.de

Prüfungskommission

Vorsitzender:	Prof. Dr. Roland Pail
Prüfende der Dissertation:	1. apl. Prof. Dr.-Ing. Michael Schmidt
	2. Prof. Dr. phil. nat. Urs Hugentobler
	3. Prof. Dr. Benedikt Soja

Die Dissertation wurde am 10.08.2023 bei der Technischen Universität München eingereicht
und durch die TUM School of Engineering and Design am 23.11.2023 angenommen.

Diese Dissertation ist auf dem Server des Ausschusses Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften unter <http://dgk.badw.de/>
sowie auf dem Server der Technischen Universität München unter
<https://mediatum.ub.tum.de/1717358> elektronisch publiziert

ISSN 0065-5325

ISBN 978-3-7696-5356-4

Acknowledgments

Completing this dissertation would not have been possible without the support of many people. Therefore, I would like to thank all those who supported and accompanied me on my doctoral journey.

First and foremost, I would like to express my deepest appreciation to my supervisor, Prof. Michael Schmidt, for supporting my research proposal and my application for the DAAD research grant so that I could write this dissertation in the first place. I am deeply grateful for your guidance, constant support and encouragement, dedicated involvement, and many discussions and advice.

Without the German Academic Exchange Service (DAAD), which supported me with the research grant for my entire doctorate, I would not have been able to embark on this journey.

I am grateful to Prof. Benedikt Soja for the discussions and collaboration, for reviewing my dissertation, and for providing me with valuable suggestions for my work. I would also like to thank Prof. Urs Hugentobler for reviewing my dissertation and sharing his knowledge and expertise.

I would like to thank Dr. Marie Dominique and the PROBA2 Guest Investigator Program for the opportunity of two research visits to the Royal Observatory of Belgium (ROB) in 2022. I sincerely appreciate her help and support in collecting and processing solar data, sharing her knowledge of solar physics, introducing me to space weather monitoring at the ROB, and answering all my questions.

Many thanks to Prof. Manuel Hernandez Pajarez for the opportunity of a 2-month research stay at the UPC Barcelona in 2022, funded by the TUM Graduate School. I am incredibly grateful for our discussions on the ionosphere and his valuable suggestions for my work.

Special thanks to my mentor, Dr. Tatjana Gerzen, for our discussions and her guidance in dealing with challenges during this process.

I also would like to acknowledge Prof. Robert Weber, who supervised me during my 9-month research stay at TU Wien in 2017/2018, and part of my research on regional ionosphere modeling there has been incorporated into this dissertation. I am deeply grateful for our discussions, for sharing his knowledge, and for supporting my application for the DAAD research grant.

A big thank you goes to Prof. Medzida Mulic for introducing me to the exciting world of satellite geodesy and ionospheric research with GNSS. Thank you for sharing your knowledge with me, constantly encouraging me, and supporting my application for the DAAD research grant.

Many thanks to my colleagues, the co-authors of my papers, Dr. Andreas Goss, Ms. Dzana Halilovic, and Dr. Nina Magnet, for their valuable contributions and fruitful collaboration.

My thanks go to all former and current colleagues at DGFI-TUM with whom I had collaborated and exchanged ideas during my Ph.D. Thank you for the pleasant working atmosphere and the interdisciplinary discussions during the numerous events at the institute. I would like to thank Prof. Florian Seitz for the opportunity to write my dissertation at DGFI-TUM.

I would also like to thank the colleagues with whom I had discussions and scientific exchanges at conferences and during my research stays. They provided valuable impulses for this dissertation.

Lastly, I would like to mention my family, especially my mother and aunt. Thank you for always believing in me and supporting me unconditionally in all my endeavors, which kept my spirit and motivation high during this process.

Abstract

Space weather describes the varying conditions between the Sun and the Earth, affecting in particular the magnetosphere, the ionosphere, and the thermosphere. It can degrade Global Navigation Satellite Systems (GNSS) operations, primarily through its effects on the ionosphere. The ionospheric GNSS signal delay can affect the accuracy of position determination, especially for single-frequency observations. The ionosphere models broadcasted by the satellite systems (e.g., Klobuchar, NeQuick-G) do not contain enough details to correct single-frequency observations with sufficient accuracy. During solar storms, dual-frequency positioning may also be affected. Therefore, a reliable ionospheric model is required to correct ionospheric effects in GNSS applications in a timely and accurate manner. Dual-frequency observations can be used to calculate the integral of electron density along the GNSS signal in the ionosphere, which can be mapped to the Vertical Total Electron Content (VTEC) to correct for ionospheric effects in GNSS applications.

Traditional models for forecasting ionospheric parameters, such as the Auto Regressive Integrated Moving Average (ARIMA), the Fourier series expansion, or the Principal Component Analysis (PCA), have the major drawback of modeling only linear relationships and failing to describe nonlinear relationships and abrupt changes in the ionosphere. Recently, Machine Learning (ML) methods have attracted considerable interest in space weather and ionosphere research, with a focus on modeling nonlinear relationships. Previous ML applications for the ionosphere have involved Deep Learning (DL) methods, while other methods have rarely been discussed. DL methods can suffer from high complexity and excessive parametrization and be difficult to understand and interpret. One of the main issues of previous work is considering ML-based ionosphere models as a black box and the lack of transparency and interpretability since there is no indication of when the results cannot be trusted. Despite the widespread use of ML, there has been minimal discussion of quantifying the uncertainty, thus lacking an indication of how confident and reliable the results of an ML system are.

This dissertation presents an ML approach for ionosphere modeling and forecasting, including modeling the space weather component and quantifying the uncertainties of the results to obtain 95% confidence intervals. Thus, this work aims to develop accurate and reliable ionosphere models that incorporate space weather information from various satellites and observatories, and apply ML techniques to model and forecast VTEC to meet the needs of GNSS users for effective ionosphere correction. Nonlinear relationships are modeled from data describing solar activity, the solar wind, interplanetary and Earth's magnetic fields, and the ionosphere. We discuss the ML-based VTEC model development workflow and explain the learning algorithms applied, the data preparation procedure, and input feature engineering, as well as the methods and approaches developed for model building, uncertainty quantification, model training, and evaluation. Using ML, the VTEC function is approximated from the input features to develop three ionosphere products of this dissertation: (1) Regional Ionosphere Map (RIM) with ML-based spatiotemporal modeling, (2) ML-based ensemble forecasting, and (3) ML-based probabilistic forecasting with uncertainty quantification.

In this dissertation and in **P-I**, RIMs are developed in several countries of the western Balkans based on dense Continuous Operating Reference Stations (CORS) observations. Subsequently, a RIM for the western Balkans is created using an Artificial Neural Network (ANN) that combines regional ionosphere parameters estimated from the CORS data with spatiotemporal (latitude, longitude, hour of day), solar (F10.7 index) and geomagnetic (Kp and Dst indices) input features. The RIMs are tested during the solar maximum in March 2014, a geomagnetic storm in March 2015, and the solar

minimum in March 2018. The developed RIMs mimic the integrated electron density much more effectively than the Klobuchar model. In addition, the RIMs significantly reduce the ionospheric effects on single-frequency positioning, demonstrating their necessity for use in positioning applications.

VTEC forecast is performed for selected ionospheric grid points at low, mid and high latitudes along the same longitude. The ML-based VTEC models are developed using learning algorithms of Decision Tree and ensemble learning of Random Forest, Adaptive Boosting, and Gradient Boosting in **P-II**. Furthermore, ensemble learning models are combined into an ensemble meta-estimator to compensate for their individual weaknesses and emphasize their strengths to produce a model with improved accuracy and generalization. It also allows quantifying uncertainties propagated through an ensemble, as is done in **P-III** and **CP-II**.

Unlike the standard K-fold cross-validation used in previous studies, the forecast models developed in this dissertation are trained, optimized, and validated by modifying the K-fold cross-validation to make it applicable to time series data. This is done because of the time dependency of the VTEC and space weather observations, which must be considered for robust and accurate model training and validation, which was not the case with the previously used method. Moreover, the relative importance of input features to VTEC forecast is estimated using the decision tree approach to provide insight into what the model has learned and to what extent our understanding of important predictors has increased. Also, the impact of different learning algorithms and data on the accuracy of the ML-based VTEC model is analyzed and discussed.

Several approaches to quantify uncertainties in ML-based VTEC models and provide 95% confidence intervals are applied in **P-III**: (1) Ensemble of ML-based models, (2) Gradient Tree Boosting with quantile loss function (QGB), (3) Bayesian neural network (BNN), and (4) BNN with estimated data uncertainties. Techniques that consider only model parameter uncertainties in (1) and (3) estimate narrow confidence intervals and over-optimistic results, whereas accounting for both model parameter and data uncertainties with the BNN approach in (4) leads to wider confidence intervals and more realistic quantification of VTEC forecast uncertainties. However, the latter approach suffers from a high computational cost, while the QGB approach is the most computationally efficient solution with still realistic uncertainties. As the feature importance analysis shows, the QGB confidence intervals are largely determined by space weather indices. In addition, the QGB approach is extended to include high-resolution VTEC, solar, and geomagnetic activity data to develop high-resolution VTEC models for various forecast horizons of 15 minutes, 1 hour, 3 hours, 6 hours, and 24 hours ahead and to forecast the effects of solar flares on the ionosphere.

The results demonstrate the feasibility of the ML-based VTEC forecast during quiet and storm periods with reliable uncertainty quantification. Based on the investigations and results of this dissertation, recommendations for future work are discussed.

Kurzfassung

Das Weltraumwetter beschreibt die wechselnden Bedingungen zwischen der Sonne und der Erde, die insbesondere die Magnetosphäre, die Ionosphäre und die Thermosphäre betreffen. Es kann den Betrieb globaler Satellitennavigationssysteme (GNSS) beeinträchtigen, vor allem durch seine Auswirkungen auf die Ionosphäre. Die ionosphärische GNSS-Signalverzögerung kann die Genauigkeit der Positionsbestimmung beeinträchtigen, insbesondere bei Beobachtungen auf einer Frequenz. Die von den Satellitensystemen ausgestrahlten Ionosphärenmodelle (z. B. Klobuchar, NeQuick-G) enthalten nicht genügend Details, um Einfrequenzbeobachtungen mit ausreichender Genauigkeit zu korrigieren. Bei Sonnenstürmen kann auch die Zweifrequenz-Positionierung beeinträchtigt werden. Daher ist ein zuverlässiges Ionosphärenmodell erforderlich, um ionosphärische Effekte in GNSS-Anwendungen zeitnah und genau zu korrigieren. Zweifrequenzbeobachtungen können zur Berechnung des Integrals der Elektronendichte entlang des GNSS-Signals in der Ionosphäre verwendet werden, das auf den absoluten vertikalen Elektroneninhalt (VTEC) abgebildet werden kann, um ionosphärische Effekte bei GNSS-Anwendungen zu korrigieren.

Traditionelle Modelle für die Vorhersage von Ionosphärenparametern, wie z. B. Auto Regressive Integrated Moving Average (ARIMA), die Fourier-Reihenentwicklung oder die Hauptkomponentenanalyse (PCA), haben den großen Nachteil, dass sie nur lineare Beziehungen modellieren und abrupte Veränderungen in der Ionosphäre nicht erfassen sowie nichtlineare Beziehungen nicht beschreiben können. In letzter Zeit sind Methoden des maschinellen Lernens (ML) in der Weltraumwetter- und Ionosphärenforschung auf großes Interesse gestoßen, wobei der Schwerpunkt auf der Modellierung nichtlinearer Beziehungen liegt. Bei früheren ML-Anwendungen für die Ionosphäre wurden Deep Learning (DL)-Methoden eingesetzt, während andere Methoden kaum diskutiert wurden. DL-Methoden können leiden unter hoher Komplexität und übermäßiger Parametrisierung, und können schwer zu verstehen und zu interpretieren sein. Eines der Hauptprobleme früherer Arbeiten besteht darin, dass ML-basierte Ionosphärenmodelle als Blackbox betrachtet werden und dass es an Transparenz und Interpretierbarkeit mangelt, da es keine Hinweise darauf gibt, wann den Ergebnissen nicht vertraut werden kann. Trotz des weit verbreiteten Einsatzes von ML wurde die Quantifizierung der Unsicherheiten bisher kaum diskutiert, so dass es keinen Hinweis darauf gibt, wie sicher und zuverlässig die Ergebnisse eines ML-Systems sind.

In dieser Dissertation wird ein ML-Ansatz für die Modellierung und Vorhersage der Ionosphäre vorgestellt, einschließlich der Modellierung der Weltraumwetterkomponente und der Quantifizierung der Unsicherheiten der Ergebnisse, um 95%iger Konfidenzintervalle zu erhalten. Ziel dieser Arbeit ist es also, genaue und zuverlässige Ionosphärenmodelle zu entwickeln, die Weltraumwetterinformationen von verschiedenen Satelliten und Observatorien einbeziehen, und ML-Techniken zur Modellierung und Vorhersage von VTEC anzuwenden, um den Bedürfnissen der GNSS-Nutzer nach einer effektiven Ionosphärenkorrektur gerecht zu werden. Nichtlineare Beziehungen werden aus Daten modelliert, die die Sonnenaktivität, den Sonnenwind, interplanetare und geomagnetische Felder sowie die Ionosphäre beschreiben. Wir diskutieren den ML-basierten Arbeitsablauf zur Entwicklung des VTEC-Modells und erläutern die angewandten Lernalgorithmen, das Verfahren zur Datenaufbereitung und das Input-Feature-Engineering sowie die Methoden und Ansätze, die für die Modellbildung, die Quantifizierung der Unsicherheit, das Modelltraining und die Evaluierung entwickelt wurden. Mithilfe von ML wird die VTEC-Abbildungsfunktion aus den Eingangsmerkmalen approximiert, um drei Ionosphärenprodukte dieser Dissertation zu entwickeln: (1) Regionale Ionosphärenkarte (RIM)

mit ML-basierter räumlich-zeitlicher Modellierung, (2) ML-basierte Ensemble-Vorhersage und (3) ML-basierte probabilistische Vorhersage mit Unsicherheitsquantifizierung.

In dieser Dissertation und in **P-I** werden RIMs in mehreren Ländern des westlichen Balkans auf der Grundlage dichter Beobachtungen von kontinuierlich arbeitenden Referenzstationen (CORS) entwickelt. Anschließend wird ein RIM für den westlichen Balkan mit Artificial Neural Network (ANN) erstellt, das regionale Ionosphärenparameter, die aus den CORS-Daten geschätzt werden, mit raum-zeitlichen (Breitengrad, Längengrad, Tageszeit), solaren (F10.7-Index) und geomagnetischen (Kp- und Dst-Index) Eingangsmerkmalen kombiniert. Die RIMs werden während des Sonnenmaximums im März 2014, eines geomagnetischen Sturms im März 2015 und des Sonnenminimums im März 2018 getestet. Die entwickelten RIMs ahmen die integrierte Elektronendichte viel effektiver nach als das Klobuchar-Modell. Darüber hinaus verringern die RIMs die ionosphärischen Auswirkungen auf die Einzelfrequenzpositionierung erheblich, was ihre Notwendigkeit für den Einsatz in Positionierungsanwendungen unter Beweis stellt.

Die VTEC-Vorhersage wird für ausgewählte ionosphärische Gitterpunkte in niedrigen, mittleren und hohen Breitengraden entlang desselben Längengrads durchgeführt. Die ML-basierten VTEC-Modelle werden unter Verwendung der Lernalgorithmen Entscheidungsbaum und Ensemble-Lernen von Random Forest, Adaptive Boosting und Gradient Boosting in **P-II** entwickelt. Darüber hinaus werden Ensemble-Lernmodelle in einem Ensemble-Meta-Schätzer kombiniert, um ihre individuellen Schwächen zu kompensieren und ihre Stärken zu betonen, um ein Modell mit verbesserter Genauigkeit und Generalisierung zu erzeugen. Es ermöglicht auch die Quantifizierung von Unsicherheiten, die sich durch ein Ensemble ausbreiten, wie in **P-III** und **CP-II**.

Im Gegensatz zur standardmäßigen K-fachen Kreuzvalidierung, die in früheren Studien verwendet wurde, werden die in dieser Dissertation entwickelten Prognosemodelle trainiert, optimiert und validiert, indem die K-fache Kreuzvalidierung modifiziert wird, um sie auf Zeitreihendaten anwendbar zu machen. Dies geschieht aufgrund der Zeitabhängigkeit der VTEC- und Weltraumwetterbeobachtungen, die für eine robuste und genaue Modellschulung und -validierung berücksichtigt werden müssen, was bei der zuvor verwendeten Methode nicht der Fall war. Darüber hinaus wird die relative Bedeutung der Eingangsmerkmale für die VTEC-Vorhersage mit Hilfe des Entscheidungsbaum-Ansatzes geschätzt, um einen Einblick zu erhalten, was das Modell gelernt hat und inwieweit sich unser Verständnis der wichtigen Prädiktoren verbessert hat. Auch die Auswirkungen verschiedener Lernalgorithmen und Daten auf die Genauigkeit des ML-basierten VTEC-Modells werden analysiert und diskutiert.

In **P-III** werden mehrere Ansätze zur Quantifizierung von Unsicherheiten in ML-basierten VTEC-Modellen und zur Bereitstellung von 95%-Konfidenzintervallen angewendet: (1) Ensemble von ML-basierten Modellen, (2) Gradient Tree Boosting with Quantile loss function (QGB), (3) Bayesian neural network (BNN) und (4) BNN mit geschätzten Datenunsicherheiten. Techniken, die nur Modellparameter-Unsicherheiten in (1) und (3) berücksichtigen, schätzen enge Konfidenzintervalle und zu optimistische Ergebnisse, während die Berücksichtigung sowohl von Modellparameter- als auch von Datenunsicherheiten mit dem BNN-Ansatz in (4) zu breiteren Konfidenzintervallen und einer realistischeren Quantifizierung der VTEC-Prognoseunsicherheiten führt. Der letztgenannte Ansatz leidet jedoch unter einem hohen Rechenaufwand, während der QGB-Ansatz die rechnerisch effizienteste Lösung mit immer noch realistischen Unsicherheiten darstellt. Wie die Analyse der Merkmalsbedeutung zeigt, werden die QGB-Konfidenzintervalle weitgehend durch Weltraumwetterindizes bestimmt. Darüber hinaus wird der QGB-Ansatz um hochaufgelöste VTEC-, Sonnen- und geomagnetische Aktivitätsdaten erweitert, um hochaufgelöste VTEC-Modelle für verschiedene Vorhersagehorizonte von 15 Minuten, 1 Stunde, 3 Stunden, 6 Stunden und 24 Stunden im Voraus zu entwickeln und um die Auswirkungen von Sonneneruptionen auf die Ionosphäre vorherzusagen.

Die Ergebnisse zeigen die Machbarkeit der ML-basierten VTEC-Vorhersage in ruhigen und stürmischen Zeiten mit zuverlässiger Unsicherheitsquantifizierung. Auf der Grundlage der Untersuchungen und der Ergebnisse dieser Dissertation werden Empfehlungen für die zukünftige Forschung diskutiert.

Preface

This cumulative dissertation consists of the following three first-author peer-reviewed journal papers:

P-I Randa Natras, Andreas Goss, Dzana Halilovic, Nina Magnet, Medzida Mulic, Michael Schmidt, Robert Weber (2023): **Regional Ionosphere Delay Models based on CORS data and Machine Learning** In: NAVIGATION: Journal of the Institute of Navigation, 70 (3) navi.577, https://doi.org/10.33012/navi.577

P-II Randa Natras, Benedikt Soja, Michael Schmidt (2022): **Ensemble Machine Learning of Random Forest, AdaBoost and XGBoost for Vertical Total Electron Content Forecasting** In: Remote Sensing, 14 (15), https://doi.org/10.3390/rs14153547

P-III Randa Natras, Benedikt Soja, Michael Schmidt (2023): **Uncertainty Quantification for Machine Learning-based Ionosphere and Space Weather Forecasting: Ensemble, Bayesian Neural Network and Quantile Gradient Boosting** In: Space Weather, 21, e2023SW003483, https://doi.org/10.1029/2023SW003483

In the Appendix, the three original publications mentioned above are included with a statement about each author's contribution.

In addition to journal publications, the first-author peer-reviewed conference papers complement the dissertation:

CP-I Randa Natras, Michael Schmidt (2021): **Machine Learning Model Development for Space Weather Forecasting in the Ionosphere** In: CEUR Workshop Proceedings of the CIKM 2021 Workshops co-located with 30th ACM International Conference on Information and Knowledge Management (CIKM 2021), Vol. 3052, pp. 1-7, https://ceur-ws.org/Vol-3052/short10.pdf

CP-II Randa Natras, Benedikt Soja, Michael Schmidt (2022): **Machine Learning Ensemble Approach for Ionosphere and Space Weather Forecasting with Uncertainty Quantification** In: IEEE Xplore Proceedings of 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC), Gran Canaria, Spain, 2022, pp. 1-4, https://doi.org/10.23919/AT-AP-RASC54737.2022.9814334

CP-III Randa Natras, Dzana Halilovic, Medzida Mulic, Michael Schmidt (2023): **Mid-latitude Ionosphere Variability (2013–2016), and Space Weather Impact on VTEC and Precise Point Positioning** In: Ademović, N., Mujčić, E., Mulić, M., Kevrić, J., Akšamija, Z. (eds) Advanced Technologies, Systems, and Applications VII. IAT 2022. Lecture Notes in Networks and Systems, vol 539. Springer, Cham., pp. 471–491, https://doi.org/10.1007/978-3-031-17697-5_37

A summary of the conference papers is provided in the Appendix.

Within this dissertation, the findings from the first-author publications are placed into context. The publications within this dissertation are cited, and each is identified with the letter P, CP (P for journal publication, CP for conference publication), and the corresponding Roman number. Furthermore, parts of the publications are cited in direct form, with numbering for equations, figures, and tables adapted to the scope of this dissertation. These direct quotes are written indented in the main body of the dissertation.

In addition, the findings of the following co-author peer-reviewed publication are incorporated into the dissertation:

C-I Veronika Barta, Randa Natras, Vladimir Srećković, David Koronczay, Michael Schmidt, Desanka Šulic (2006): **Multi-instrumental investigation of the solar flares impact on the ionosphere on 05–06 December 2006** In: Frontiers in Environmental Science, 904335, https://doi.org/10.3389/fenvs.2022.904335

A summary of the co-author publication is provided in the Appendix.

Moreover, the studies conducted during the author's research stays at Royal Observatory of Belgium, Brussels (ROB) on forecasting the solar flare impact on the ionosphere and at Universitat Politècnica de Catalunya; Technical University of Catalonia, Spain (UPC) on high-temporal probabilistic ionosphere forecast with uncertainty quantification using Quantile Gradient Boosting are incorporated into the dissertation.

The main body of this dissertation begins with the motivation, research objectives, and questions, followed by an introduction to the subject of the ionosphere and space weather, geodetic observational techniques, and Vertical Total Electron Content (VTEC) modeling. The explanation and examples of ionosphere variabilities and space weather effects are mainly based on the publication **CP-III** Natras et al. (2023b). Thereafter, an introduction to machine learning for VTEC is provided, the fundamental theory of learning from data and learning algorithms adapted to VTEC problem, based on publications **P-I** Natras et al. (2023a), **P-II** Natras et al. (2022a), **CP-I** Natras & Schmidt (2021), including the developed approaches to quantify uncertainties in VTEC forecasting, based on publications **P-III** Natras et al. (2023d), **CP-II** Natras et al. (2022b). Subsequently, the applied procedures for developing machine learning-based VTEC models, such as data selection (time periods and features preparation), training, cross-validation and testing phases, and model architecture selection, are explained. The whole chain of machine learning-based VTEC model development is demonstrated, mainly based on publication **P-II**. The machine learning approach to spatio-temporal VTEC modeling for regional ionosphere estimation and its application in positioning is afterward discussed, based on publication **P-I**. The forecasting results by ensembling the models are analyzed, based on **P-II**, including the estimation of 95% confidence intervals from the ensemble, based on **CP-II**, **P-III**. Then, the uncertainty quantification results from different approaches based on **P-III** are compared and analyzed, followed by the development of the high-resolution probabilistic VTEC model during the author's research stay at UPC. In addition, a first attempt to forecast the solar flare impact on the ionosphere is presented, including a description of the processing and integration of solar observations covering the EUV and X-ray spectra of solar irradiance to allow short-term forecast of the ionosphere response to strong solar flares, which was conducted during the author's research stay at ROB. VTEC modeling and forecasting results are discussed for selected quiet and disturbed test periods regarding solar and geomagnetic activity. Finally, conclusions and an outlook on future work are provided.

Contents

Acronyms

AdaBoost Adaptive Boosting. 40, 44–46, 52, 76, 77, 85, 86, 110, 112

AI Artificial Intelligence. 3, 4, 7, 34, 106, 109

ANFIS Adaptive Neuro-Fuzzy Inference System. 10

ANN Artificial Neural Network. 4, 8–10, 12, 38, 40, 41, 46–51, 54, 66–68, 72, 75, 76, 82, 97, 101–104, 110, 112

ARIMA Autoregressive Integrated Moving Average. 9, 10

ARMA Autoregressive Moving Average. 10

BH Bosnia-Herzegovina. 82

BIMF Barcelona Ionospheric Mapping Function. 23

BNN Bayesian Neural Network. 54, 76, 77, 86, 103, 104

BNN+D Bayesian Neural Network including Data Uncertainty. 55, 86–90, 104, 105, 107, 112

CAS Chinese Academy of Sciences. 32

CDDIS Crustal Dynamics Data Information System. 62

cGAN conditional Generative Adversial Network. 10

CHAMP CHAllenging Minisatellite Payload. 21

CI confidence interval. 87, 89, 93, 94, 96, 112

CME Coronal Mass Ejection. 1, 18, 20, 29, 30, 105, 109

CNN Convolutional Neural Network. 10

CODE Center for Orbit Determination in Europe, University of Bern, Switzerland. 32, 33, 62, 82, 83, 111

CORS Continuous Operating Reference Stations. 22, 62, 63, 82, 107, 111

DGFI-TUM Deutsches Geodätisches Forschungsinstitut der Technischen Universität München. 3, 33, 82, 109

DL Deep Learning. 4, 5, 8–12, 103, 109

DORIS Doppler Orbitography and Radiopositioning Integrated by Satellite. 21

DoY Day of Year. 38, 39, 42, 57, 58, 66–68, 90, 110, 111

Dst Disturbance storm time. 3, 20, 29, 38, 39, 64, 77

DT Decision Tree. 76, 77

EMA Exponential Moving Average. 38, 66, 67, 90, 98, 99, 102, 103, 112

EPN EUREF Permanent Network. 22, 25–27, 29, 31, 62, 63, 67, 82, 107, 109–111, 230

ESA European Space Agency, Germany. 3, 32, 109

EUV Extreme Ultraviolet. 19, 27, 64

FTP File Transfer Protocol. 63

GAIM Global Assimilation of Ionospheric Measurements. 32

GBDT Gradient Boosting Decision Tree. 10, 13

GBoost Gradient Boosting. 40, 44–46, 52, 54, 104, 110

GIM Global Ionosphere Map. 20, 24, 26, 32–34, 62, 82, 83, 108, 109, 111

GITM Global Ionosphere-Thermosphere Model. 32

GLONASS GLObal'naya NAvigatsionnaya Sputnikovaya Sistem. 21, 25, 63

GMF GeoMagnetic Field. 20, 39, 64, 68, 111

GNSS Global Navigation Satellite System. 2, 4, 20–27, 29–35, 62, 63, 101, 109, 110

GOES Geostationary Operational Environmental Satellite. 39, 64

GPS Global Positioning System. 1, 20, 21, 25, 32, 62, 63, 82

GSFC Goddard Space Flight Center, USA. 64

GT ground truth. 89, 93, 94, 112

HoD Hour of Day. 39, 42, 56–58, 66, 67, 90, 97, 110

HOI Higher Order Ionosphere terms. 23, 30, 35

IAAC Ionosphere Associated Analysis Centers. 32

IGS International GNSS Service. 22, 24, 32

IMF Interplanetary Magnetic Field. 39, 64, 68, 111

IONOWB IONOsphere model for the Western Balkans. 34, 62, 63, 66, 82, 84, 85, 111, 112

IONOWB_AI IONOsphere model for the Western Balkans with Artificial Intelligence. 66, 82–85, 98, 111, 112

IPP Ionospheric Pierce Point. 23, 24, 34, 107

IRI International Reference Ionosphere. 32

JPL Jet Propulsion Laboratory, USA. 26, 32, 109

KL Kullback-Leibler. 54

LSTM Long Short-Term Memory. 10, 103

LT Local Time. 25, 88

Ly-α Lyman-alpha. 39, 64

ML Machine Learning. 4–12, 14, 15, 31, 34, 35, 37–41, 46, 52, 53, 61, 62, 64, 66, 67, 72, 74–77, 85, 86, 101–104, 107–111, 114

MLP MultiLayer Perceptron. 47, 50, 76, 82, 103

MSE Mean Squared Error. 40, 55, 56

NASA National Aeronautics and Space Administration, USA. 2, 26, 62, 64, 109

NAVSTAR NAVigational Satellite Timing And Ranging. 21

NLL Negative Log-Likelihood. 56

NOAA National Oceanic and Atmospheric Administration, USA. 19, 27, 109

NRCan Geodetic Survey of Natural Resources Canada. 32

OPTIMAP Operational Tool for Ionosphere Mapping and Prediction. 127, 133

PCA Principal Component Analysis. 9, 10

PPP Precise Point Positioning. 20, 23, 25, 29, 35, 81, 82, 84, 85, 112, 230

PROBA PRoject for Onboard Autonomy. 3, 39, 64, 65, 111

QGB Quantile Gradient Boosting. 54, 75–77, 86, 90, 104–106

ReLU Rectified Linear Unit. 49, 50, 110

RF Random Forest. 76, 77, 97, 102, 112

RIM Regional Ionosphere Map. 13, 32–34, 62, 63, 66, 82, 83, 106, 107, 111

RMS Root Mean Square. 12, 29–31, 70, 71, 75, 82, 85, 86, 91, 98, 99, 110–112

RNN Recurrent Neural Network. 10

RO Radio occultation. 21

ROB Royal Observatory of Belgium, Brussels. viii, 64, 81

RTK Real-Time Kinematic. 23

S$_n$ Sunspot number. 18, 19, 39, 67, 68, 109, 111

SE Super-Ensemble. 46, 52, 53, 76, 77, 86, 110

sfu solar flux unit. 27, 68, 82, 85, 88, 109, 111

SGD Stochastic Gradient Descent. 50, 55

SLM Single Layer Model. 23, 24, 109

SSC Sudden Storm Commencement. 30, 109

STEC Slant Total Electron Content. 23

STEREO Solar TErrestrial RElations Observatory. 3

SVM Support Vector Machine. 10

SW Space Weather. 67, 69, 77

SWS Solar Wind Speed. 39

TECU Total Electron Content Unit. 23, 29, 42, 51, 56, 68, 82, 92, 99, 110, 111

TOPEX TOPography EXperimentTOPography EXperiment. 21, 32

UPC Universitat Politècnica de Catalunya; Technical University of Catalonia, Spain. viii, 32, 62, 81

UQ Uncertainty Quantification. 8, 11, 14

UTC Coordinated Universal Time. 20, 27, 29, 31, 64, 81–83, 95, 97, 110–112

UV Ultraviolet. 17–19, 27, 64, 65, 95

VR Voting Regressor. 76, 77, 85, 99, 112

VTEC Vertical Total Electron Content. 23–34, 37, 52, 53, 55–58, 63, 67–70, 72–77, 86, 92, 101, 104, 108–112, 114

WHU Wuhan University, China. 32

XGB eXtreme Gradient Boosting. 10, 45, 76

XGBoost eXtreme Gradient Boosting. 13, 45, 75–77, 85, 86, 98, 99, 112

1 Introduction

1.1 Motivation

Imagine a scenario in which the electric power suddenly goes out, navigation services, mobile phones, and the internet stop working, and public transport systems and airports shut down. And no one knows how long it will last - a few hours, a week, a month, a year, or even longer. It sounds more like a story from a science fiction movie. But it can suddenly become our reality. The leading cause of such a catastrophe would be our star, the Sun, which can create time-varying conditions in near-Earth space produced by space weather. The term space weather refers to dynamic changes in interplanetary and interstellar space, which are caused by the Sun and affect in particular the magnetosphere, the ionosphere, and the thermosphere. As space weather events, we mainly distinguish between solar flares and Coronal Mass Ejection (CME); for details, see Section 2.2. Space weather can disable the power supply, communications, and navigation systems across the country or around the world. Similar examples have already happened in the past.

In March 1989, the Sun released a cloud of gas from a solar flare equivalent to the energy of thousands of nuclear bombs exploding simultaneously. It caused a 12-hour power outage in Quebec, Canada, with millions of people suddenly finding themselves in dark buildings, underground pedestrian tunnels, and stuck elevators; schools, stores, businesses, and the airport closed; and the Montreal Metro stopped running(1). Yet this storm was only about one-third the size of the Carrington Event, the largest recorded solar event, which hit the Earth in 1859 and caused the failure of telegraph systems across America and Europe. The telegraph is now a thing of the past, but many new technologies we increasingly rely on are equally vulnerable to space weather bursts. Another solar storm, in 2003, known as the Halloween storm, caused the radio and Global Positioning System (GPS) blackouts, substantial financial losses for airlines that had to reroute hundreds of flights around the world, the loss of track of low Earth orbit satellites for days, and left people in Malmö, Sweden, without electricity (Pulkkinen et al., 2005). An extreme solar storm as strong as the 1859 Carrington event occurred in July 2012 and fortunately missed the Earth(2). If it had come just a week earlier, it would have hit the Earth, and the consequences would have been catastrophic.

Given modern technology and the fact that today's world depends on electric power, communications, and navigation, a solar storm of the same magnitude as the Carrington event or even larger would cause widespread disruptions. Such an event would be a serious threat to our highly vulnerable, technology-dependent society and our modern way of life. It would damage satellites or make them inoperable for some time, drag them out of orbit, damage power grids, and disrupt satellite communications, navigation and surveillance systems, timing, mobile phone networks, and internet connections. Banking systems, aviation, trains, and autonomous vehicles would also be affected. According to experts, it could even take several years up to decades before all information, communication, and navigation systems are fully functional again, electronic systems are repaired, and everything is back online. The total economic damage could run into the trillions of dollars.

Numerous impacts and significant economic losses due to space weather events have already been observed in space-based and ground-based technical systems such as satellites, power grids, aviation,

(1)Source: https://www.nasa.gov/topics/earth/features/sun_darkness.html

(2)Source: https://science.nasa.gov/science-news/science-at-nasa/2014/23jul_superstorm communications, and navigation systems (see, e.g., Lanzerotti (2017); Baker et al. (2004); Eastwood et al. (2017)), as illustrated in Figure 1.1. Consequently, space weather is identified as the most significant risk to technology and society, such as in the 2019 US National Space Weather Strategy and Action Plan (Knipp & Gannon, 2019).

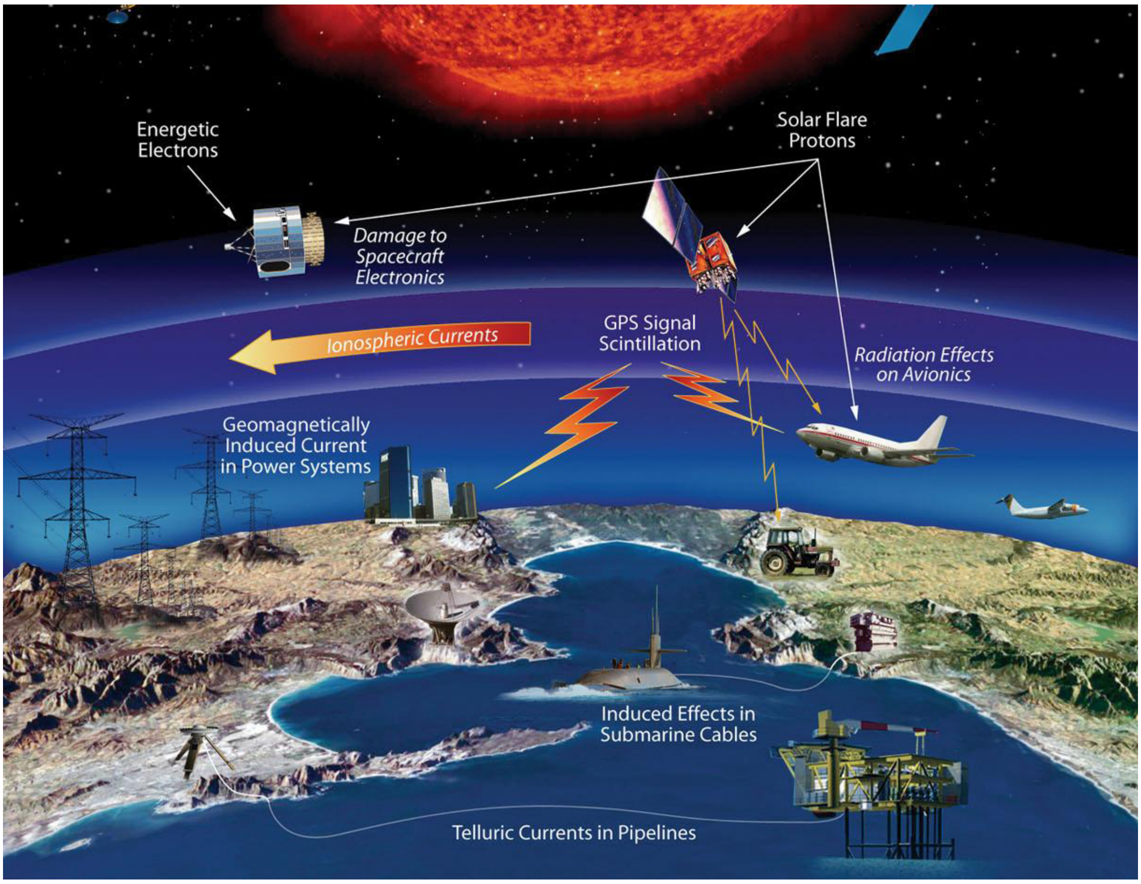

Energetic Electrons
Solar Flare Protons
Damage to Spacecraft Electronics
Ionospheric Currents
GPS Signal Scintillation
Radiation Effects on Avionics
Geomagnetically Induced Current in Power Systems
Induced Effects in Submarine Cables
Telluric Currents in Pipelines

Figure 1.1: Solar storms have many noticeable impacts on and near Earth, including communications and navigation disruptions, radiation hazards, spacecraft damage, power system outages, and pipeline corrosion. Illustration source: National Aeronautics and Space Administration, USA (NASA).

Of particular concern is the disruption of the Global Navigation Satellite System (GNSS), which is ubiquitous in mobile phones, airplanes, and automobiles. Numerous industries and the military rely on satellite-based positioning and navigation for accurate navigation, orientation, tracking, and guidance. The ionosphere, the upper part of the Earth's atmosphere, plays a critical role in the propagation of electromagnetic waves in satellite-based navigation systems. It causes a signal delay and bending, including signal scintillation, which is more pronounced during space weather events when irregular and strong fluctuations can occur. These effects are challenging to model and adequately correct in positioning solutions using conventional mathematical approaches, degrading the positioning and navigation performances (see, e.g., Natras et al. (2019a, 2023b); Poniatowski & Nykiel (2020); Zakharenkova & Cherniak (2021); Luo et al. (2018a)). Therefore, the space weather impact on the ionosphere must be considered and estimated in GNSS applications.

Since our society is highly dependent on GNSS applications that require accurate positioning, navigation, and timing, it is urgently necessary to develop advanced forecasting methods of space weather events to mitigate the catastrophic consequences of this hazard and the failure of space- and ground-based technological systems.

However, we still do not know much about how solar storms form to predict them well. To make more reliable space weather forecasts and assess potential impacts, a much better understanding of the likelihood of space weather disturbances and their effects is needed. It involves a complex chain of physical and dynamic processes between the Sun, interplanetary space, the Earth's magnetic field, and the ionosphere, which must be considered when modeling and predicting space weather impacts (Figure 1.2). However, due to our limited understanding of these coupled processes, we do not have functional relations that can accurately describe them.

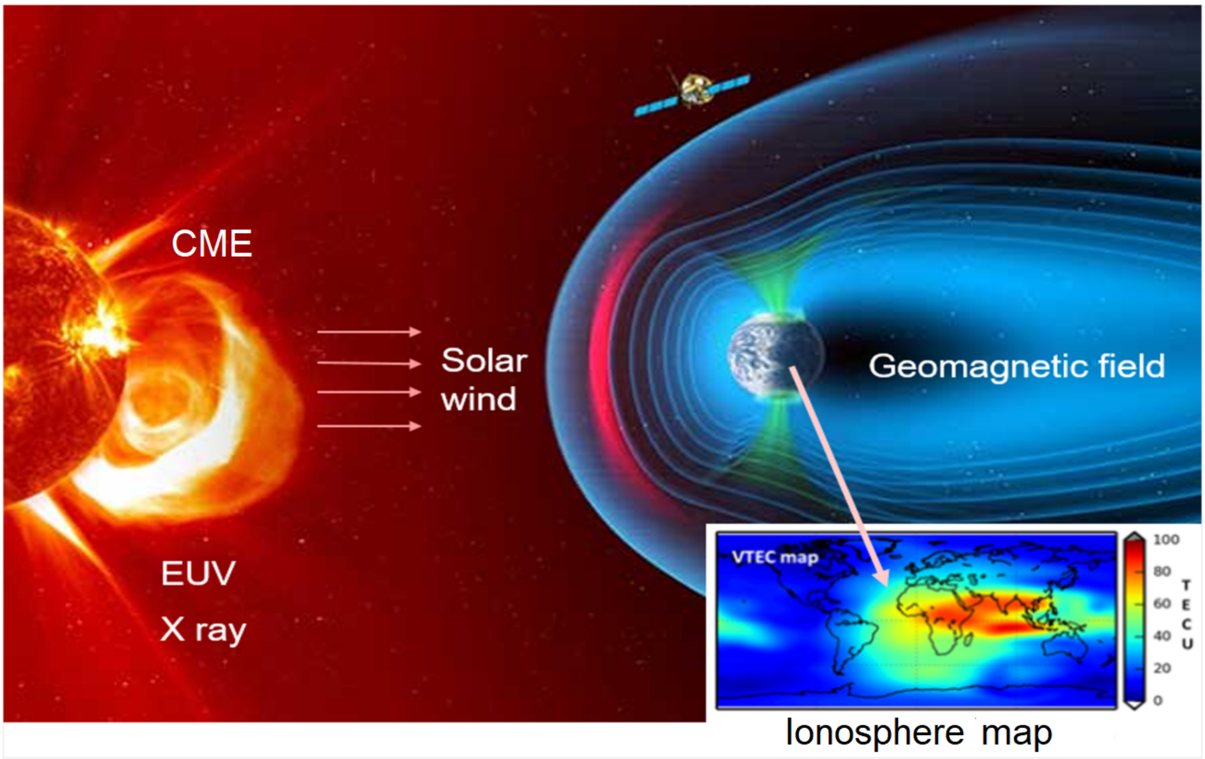


Figure 1.2: Illustration of the chain of space weather processes. (Source: European Space Agency, Germany (ESA) (background image) and Deutsches Geodätisches Forschungsinstitut der Technischen Universität München (DGFI-TUM) (ionosphere map)).

In recent decades, more and more satellites have been launched to study the Sun and the space weather processes from the Sun to the Earth, such as Solar Orbiter (Müller et al., 2013), Solar TErrestrial RElations Observatory (STEREO) (Kaiser & Adams, 2007) and PRoject for Onboard Autonomy (PROBA) (Dominique et al., 2013). In addition, solar and geomagnetic observatories are developing various indices to quantify the intensity and effects of solar storms, such as the F10.7 index (Tapping, 1987), the Kp index (Matzka et al., 2021) and the Disturbance storm time (Dst) index (Wanliss & Showalter, 2006a). With growing computational capabilities and new computational techniques such as Artificial Intelligence (AI), these data can be used in ways that overcome shortcomings in traditional modeling and forecasting solar events and their effects on Earth.

The last few years have witnessed a massive growth in the use of AI to predict space weather phenomena, from conditions on the Sun to their effects on Earth, including the ionosphere. AI offers a new possibility to "learn" these complex, dynamical relationships directly from data and to find functions that describe nonlinear space weather processes. The state of the art of AI application to the ionosphere is described in Section 1.5. Previous studies' challenges and research gaps are discussed in Section 1.6 and addressed with the research objectives outlined in Section 1.7.

The dissertation focuses on the problem of ionosphere modeling and forecasting using AI with special emphasis on modeling the space weather impact on the ionosphere. This task is defined by modeling and forecasting the electron content of the ionosphere, which can be estimated from dual- or multi-frequency GNSS observations and used as external correction information in positioning and navigation.

Considering the problems outlined above, the major motivations for this dissertation are

- **Development of ionosphere models considering nonlinear relationships and the space weather effects on the ionosphere, with particular emphasis on the forecasting of space weather-induced ionosphere perturbations,**
- **Quantifying the uncertainties of the developed models,**
- **Using such solutions as an early-warning system for space weather disturbances in the ionosphere and correcting ionospheric delays in positioning and navigation.**

1.2 Machine Learning: Definition, Types and Phases

Machine Learning (ML) is a subfield of AI and one of the fastest growing fields today. It offers the possibility to learn functions and relationships from historical data, unlike traditional modeling approaches that require explicitly defined functions to describe relationships and patterns in data (Natras & Schmidt, 2021). The first mathematical models of AI/ML emerged in the 1940s. However, the significant rise of ML did not begin until the 21st century, when the availability and affordability of computing power increased tremendously, enabling exponential growth of ML (Figure 1.3). The new term Deep Learning (DL) was introduced in the ML community in the last few decades. It represents a subfield of ML that includes a particular type of learning algorithm called Artificial Neural Network (ANN) with numerous intermediate layers between the input and output layers. In recent years, ML

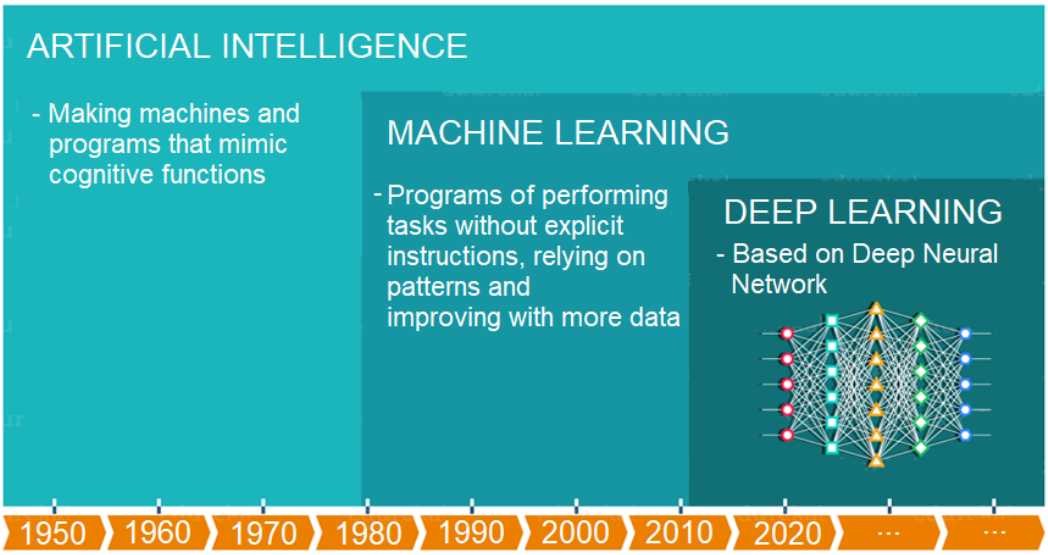

ARTIFICIAL INTELLIGENCE
- Making machines and programs that mimic cognitive functions
MACHINE LEARNING
- Programs of performing tasks without explicit instructions, relying on patterns and improving with more data
DEEP LEARNING
- Based on Deep Neural Network
1950
1960
1970
1980
1990
2000
2010
2020
...
...

Figure 1.3: Timeline of AI, ML and DL history.

and DL have attracted considerable interest in space weather research (Camporeale et al., 2018), with a focus on finding nonlinear relationships that can describe the underlying physical behavior of the Sun-Earth system.

Machine learning consists of three components:

(1) **Learning algorithm**: It is a set of procedures. An algorithm learns to recognize patterns after being trained on a large set of examples in the training data.

(2) **Training data**: They are used to train a learning algorithm.

(3) **Model**: Once a learning algorithm has been trained, the result is an ML-based model representing an ML or DL system. It has a determined architecture and parameters and specifies a function that maps the input to the output.

The main types of ML implementations, based on whether an output variable is available to the learning system, can be divided into supervised and unsupervised learning (Figure 1.4). In supervised learning, the goal is to learn a mapping from input to output, given a set of input-output pairs in the learning phase, and it can be considered as a function approximation task. It is called "supervised" because the outcome variable is given and guides the learning process. In an unsupervised learning problem, we only observe the features and have no outcome measurements. Here, the task is to analyze how the data are organized or clustered, to find patterns in the data, or to reduce the dimensionality of the data. This is sometimes called a knowledge discovery technique. A brief explanation of the supervised learning technique and associated terminology is provided in the following.

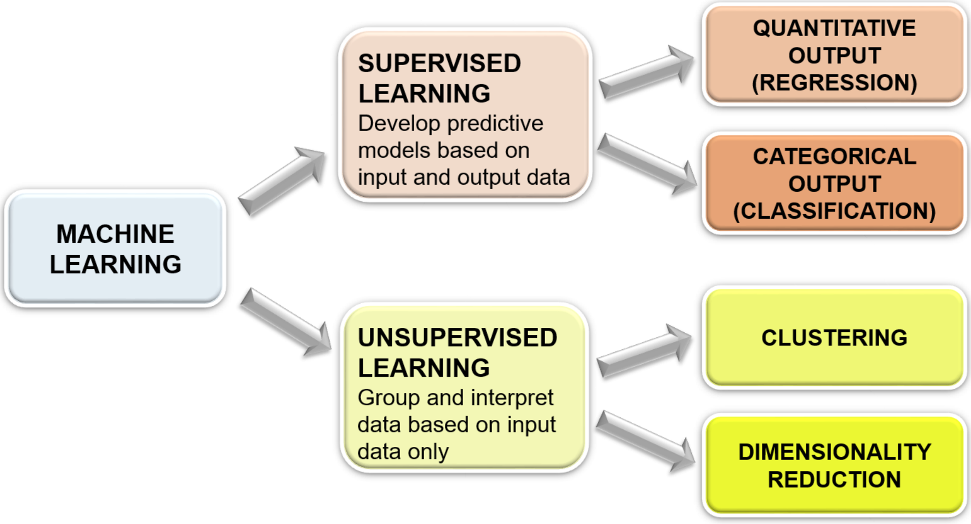

MACHINE LEARNING
SUPERVISED LEARNING
Develop predictive models based on input and output data
QUANTITATIVE OUTPUT (REGRESSION)
CATEGORICAL OUTPUT (CLASSIFICATION)
UNSUPERVISED LEARNING
Group and interpret data based on input data only
CLUSTERING
DIMENSIONALITY REDUCTION

Figure 1.4: The main types of ML implementations depend on the type of learning or whether the output is available to a learning system: supervised learning (output is given, i.e., the data is labeled) and unsupervised learning (no labeled data is available to a system).

The output variable in supervised learning can be: (1) quantitative, such as the ionospheric quantity of electron density, which is called the regression problem; (2) categorical, such as binary categories "ionospheric storm" and "no ionospheric storm", which is called the classification problem. Regression methods are used to model relationships between the input and the output variables, which can be described using linear and/or nonlinear methods. In contrast to linear regression, nonlinear regression offers flexible curve fitting capabilities, whereas linear regression may be inadequate. Both regression and categorical methods used in supervised learning are mainly nonlinear methods for estimating parameters, where the results are expressed as nonlinear functions of the unknown parameters, and the models can be univariate or multivariate.

As shown in Figure 1.5, three phases can be distinguished in the ML model development:

1. Training, learning or parameter estimation,
2. Hyperparameter tuning or model selection,
3. Prediction, forecast, point estimation or inference.

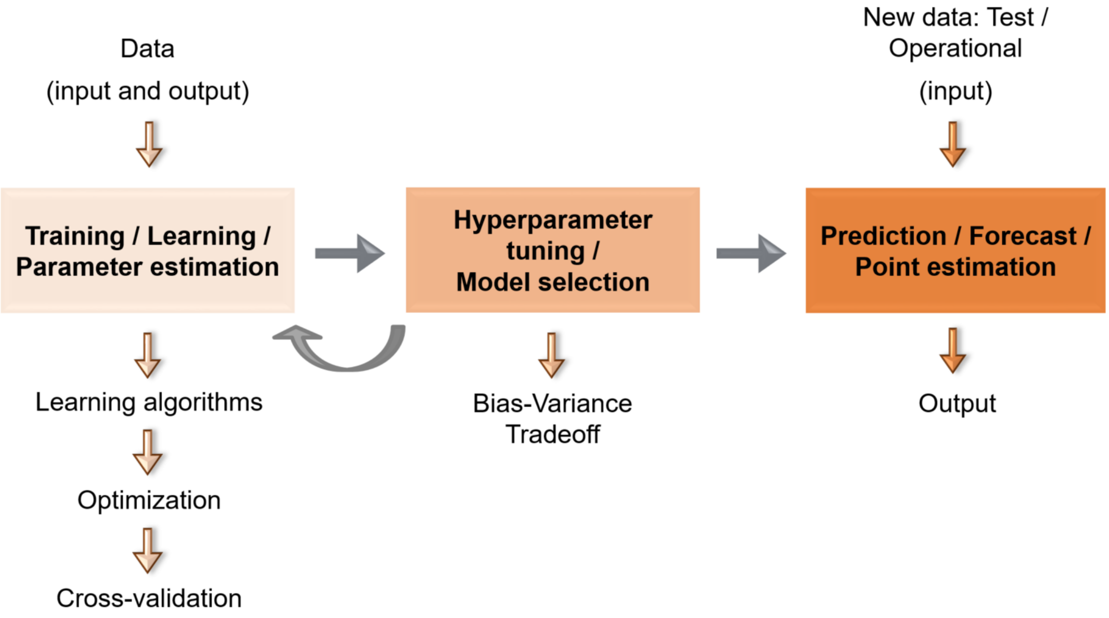

Data
(input and output)
New data: Test / Operational
(input)
Training / Learning / Parameter estimation
Hyperparameter tuning / Model selection
Prediction / Forecast / Point estimation
Learning algorithms
Optimization
Cross-validation
Bias-Variance Tradeoff
Output

Figure 1.5: Three main phases in the development of an ML model: training/learning, tuning of hyperparameters/model selection, and prediction/forecast.

The learning process aims to find a model, i.e., a mapping function between input variables, called features or covariates, and output data, called the response variable, with optimized parameters so that the resulting model performs well on new data. In the training or parameter estimation phase, the model is fitted using the available data. First, the input data and the corresponding output, i.e., ground truth, are observed from historical data, referred to as training data. The term "ground truth" refers to the "real" information or "correct" answer to a particular problem or question determined by observations. The training data are used to build a model during the learning process, which leads to a function that maps input to output. Given training data, model parameters are adjusted based on some quality measures. This is done by an objective or cost minimization, which provides an optimization problem for searching optimal parameters. The ultimate goal is to optimize the model to perform well on new, future data. Therefore, it is not enough for the model to fit the training data well; it must also perform well on unseen data. The cross-validation method simulates the model's behavior on future data.

To perform well on unseen data, we need to strike a balance[3] between fitting the data and finding a general function to describe the problem. This includes deciding on the structure of the model, such as the number of model components or free parameters, known as hyperparameters. The separation between model parameters and hyperparameters is usually determined by distinguishing between

[3] This is called a bias-variance tradeoff in ML. Further information can be found in Section 4.3.1.

what can be optimized numerically versus what must be determined using search techniques. More precisely, parameters result from fitting or an adjustment during training, while hyperparameters have to be set manually and tuned. Another way to distinguish is to consider the parameters as the explicit internal parameters of a model and hyperparameters as external configuration variables. The choice of hyperparameters significantly affects the performance of the model. The problem of choosing a model from a set of models obtained with different values of hyperparameters and learning algorithms is called model selection.

Estimating the output from new input observations using a function obtained as a result of the learning process is usually referred to as prediction. Thus, observations are predicted for times or locations where no measurements have been made. In this case, a trained model is applied to new input data, such as test or operational data. These data are not seen during the learning process, i.e., for which the output is unknown or assumed to be unknown. In other words: At this stage, the parameters and choice of the model are already fixed, and the model is applied to new vectors representing new input data points. When a future value is estimated based on the past and most recent data records and their trends, this can be called a forecast. Forecasting is a subdiscipline of prediction in which future outcomes are estimated based on time series data. Thus, the main difference between forecasting and prediction is the consideration of the time dimension.

1.3 Machine Learning: Applications and Issues

ML is increasingly used across various disciplines and application domains. The use of ML is especially necessary or helpful in the following cases:

- For modeling nonlinear relationships.
- For predictions and forecasts.
- For processing multidimensional and multivariate data.
- For detecting patterns and anomalies and gaining insights into complex problems (e.g., data clustering, data analysis, data segmentation, etc.).
- Complex problems for which there is no satisfactory solution using a traditional approach and a solution can be found using ML methods (e.g., speech, face and handwriting recognition, autonomous driving, image processing, computer vision, etc.).
- For the automated processing of large amounts of data (e.g., large databases from the growth of automation, continuous acquisition of data from satellites, etc.).
- For developing self-customizing programs (e.g., personalized product recommendations).

However, there are also challenges and concerns with the application of ML that need to be mentioned and considered when developing ML-based models and using their results for decision making:

- Challenges in obtaining a comprehensive data set for training that is inclusive, unbiased, and of good quality.
- The performance of models derived from the training data depends on the quality of that data.
- Noisy, low-quality, biased, imprecise, unreliable, or problematic data given to an ML system or problematic instructions given in training a model, lead to problematic results[4] and jeopardize the validity of the resulting model.

[4] The classic adage for this concept in the AI and ML community is "Garbage in, garbage out", meaning that nonsense/garbage input data and/or nonsense instructions to an AI system produce nonsense output.

- Treating the models as a black box[5], whose results are not questioned and taken as given. Although ML/DL models can be undoubtedly complex, they should not be treated as black boxes if the algorithms are known, if the training data and training procedures are transparent, and if the model is available to look inside it and examine what its components do.
- In the absence of external, human-given information about the data relationship, ML may make incorrect assumptions.
- Challenges in accurately interpreting and explaining the results.
- Ethics, fairness, accountability and transparency in ML-based model development.

To address the problem of using ML as a "black box" and the challenges of ethics, fairness, accountability, and transparency, new types of interactions between humans and ML algorithms are defined, commonly referred to as human-in-the-loop machine learning (Mosqueira-Rey et al., 2023). This involves close interaction between users and the learning system so that a human domain expert can be involved in and have control over the learning process. Explainable AI techniques, physics-informed ML, and the Uncertainty Quantification (UQ) can improve the interpretation and explanation of ML results, as well as reduce or prevent incorrect assumptions from an ML system. In this dissertation, explainable AI techniques are applied to analyze the contributions of input variables to the results. To improve the explanation and interpretability, the uncertainties of the results from the data and model are quantified and compared using different approaches. To ensure transparency and reproducibility, all ML and UQ methods and approaches used, modified, and developed in this dissertation are explained in Chapter 3 with the corresponding mathematical formulations. To address the data issue, the curation and labeling of the data sets need to be examined and transparently reported, e.g., how the data were cleaned and preprocessed and how reliable and reproducible the data are. Therefore, data acquisition, processing, feature extraction, data partitioning, model training, and validation, including model parameter selection, are reported in Chapter 4.

1.4 Machine Learning vs. Geodetic Estimation

The following presents the main similarities and differences between approaches commonly used in geodetic parameter estimation and ML. However, a detailed elaboration is beyond the scope of this dissertation.

One of the optimization methods for estimating unknown parameters in geodesy is the least squares method, which is used to estimate linear model parameters by minimizing the sum of squares of the measurement errors. Implementations of the traditional approaches for solving nonlinear estimation problems in geodesy are associated with a significant increase in computational costs and can, therefore, be effective only for a small number of data, according to Sholokhov et al. (2020). On the other hand, ML methods are generally applied as nonlinear methods, i.e., for modeling nonlinear relationships on large data sets. For parameter optimization, numerical methods such as gradient descent are often used in ANN to approximately construct the solution to a problem. Gradient descent is used to find the minimum of a differentiable loss function and can be applied to any method, not limited to ML. For instance, it is also applied in geodesy, such as in ionosphere electron density modeling using the constraint optimization approach (Lalgudi Gopalakrishnan & Schmidt, 2022).

Supervised learning in ML can be somewhat equated with a point estimation method in geodesy, where the vector of unknown parameters is computed from measurements. The set of parameter vectors spans a vector space containing all possible values for the parameters, called the parameter space

[5] The term black box refers to a system where we can only observe the input and output variables, but not the internal processes. The opposite of a black box is sometimes called a glass box. An AI glass box would be a system with algorithms, training data, and models available for anyone to see.

(Koch, 1999). Similarly, we can establish equivalence and say that we have both the parameter space and the hyperparameter spaces in ML. The latter contains all possible values of the hyperparameters. The weights in ANN and the value of the split point in the decision tree are examples of parameters, while the number of hidden layers and hidden neurons in ANN or the maximum depth of the decision tree are examples of hyperparameters. The hyperparameter space is searched using methods such as grid search or random search, while the parameters are estimated numerically from the given data using optimization techniques.

The unknown parameters can be estimated by an estimator. An estimator in geodetic terminology refers to the function of the observations (Koch, 1999), while an estimator in ML is commonly referred to as a learning algorithm (Hastie et al., 2009). When the interval in which the values of the parameters lie with a given probability is estimated, we speak of interval estimation, where the limits of the intervals are called confidence limits, and the interval is called a confidence interval in both geodetic (Koch, 1999) and ML (Murphy, 2012) terminology.

Bayesian statistics has been used in geodesy (Koch, 2018) as well as in ML (Neal, 2012). It allows the estimation of uncertainty expressed in terms of probability by applying Bayes's theorem. It can be applied to both linear and nonlinear stochastic systems. The latter is commonly used in the ML approach, where in most cases Bayesian inference is combined with ANN (Neal, 2012). In Bayesian statistics, a confidence interval can be derived in which the unknown parameters lie with a certain probability. The unknown parameters here are random variables, whereas in traditional statistics, they are treated as fixed quantities. Bayesian statistics is often used in geodesy together with the Monte Carlo method, which generates random variables from given distributions (Koch, 2018). Furthermore, Monte Carlo simulation allows the propagation of uncertainties and has been used to estimate nonlinear errors in geodetic computations (Wyszkowska, 2017; Zou et al., 2022). Monte Carlo can also propagate uncertainties through ANN, e.g., by dropping neurons in the neural network's hidden layers, approximating Bayesian inference (Gal & Ghahramani, 2016). Moreover, the Markov Chain Monte Carlo method is applied to estimate the distribution function of deterministic and stochastic parameters for geodetic time series (Olivares-Pulido et al., 2020). In ML, the Markov Chain Monte Carlo approximates the posterior probability distribution of the parameters in the Bayesian ANN approach. Another faster method is variational inference, which is applied in this dissertation in Section 3.3.4.

The maximum likelihood method is also used for parameter estimation in geodesy, where the density function of the observations must be specified (Koch, 1999). This method estimates unknown parameters of probability distributions of random parameters or weights by minimizing the negative log-likelihood and is applied in geodesy as well as in ML.

1.5 State of the Art

Before the use of ML for ionosphere modeling, traditional or conventional models to forecast ionospheric parameters have been used, e.g., the Autoregressive Integrated Moving Average (ARIMA) (Krankowski et al., 2005), the Fourier series expansion (Badeke et al., 2018), or the method of empirical orthogonal functions (Chen et al., 2020) also known as Principal Component Analysis (PCA), to name a few. However, the major drawback of the traditional linear models is that they do not capture abrupt changes in the ionosphere and cannot describe nonlinear relationships. The development of a more accurate and reliable ionosphere forecasting model remains an essential challenge in the field of ionospheric and GNSS research.

Recently, ML methods have attracted considerable interest in space weather and ionosphere research, emphasizing modeling nonlinear relations. Previous ML applications for the ionosphere mostly involve DL methods, i.e., the ANN. Various ANN architectures have been proposed, such as the feed-forward

ANN (e.g., Adolfs & Hoque (2021); Tebabal et al. (2019)), Recurrent Neural Network (RNN) as the Long Short-Term Memory (LSTM) and RNN combined with Convolutional Neural Network (CNN), LSTM-CNN (e.g., Liu et al. (2020b); Tang et al. (2020a); Kaselimi et al. (2022)), Encoder-Decoder LSTM Extended (ED-LSTME) (e.g., Xiong et al. (2021)), conditional Generative Adversial Network (cGAN) (e.g., Lee et al. (2020)), as well as Adaptive Neuro-Fuzzy Inference System (ANFIS) (e.g., Ghaffari Razin & Voosoghi (2020)). Only a few of the previous ionosphere forecasting studies have applied ML methods outside of DL, such as Gradient Boosting Decision Tree (GBDT) (Han et al., 2022), eXtreme Gradient Boosting (XGB) (Zhukov et al., 2021), Support Vector Machine (SVM) (Xia et al., 2021) and nearest neighbor (Monte-Moreno et al., 2022).

The results of the previous research demonstrate that ML-based ionosphere models can find nonlinear patterns in the data and outperform traditional linear methods, such as the PCA (Uwamahoro & Habarulema, 2015) and the ARIMA models (Tang et al., 2020a). According to Kaselimi et al. (2022), feedforward neural networks can approximate nonlinear Autoregressive Moving Average (ARMA) relationships and improve ionosphere modeling. Researchers using ML techniques other than ANN reported improved accuracy compared to DL-based ionosphere models, e.g., the XGB-based global ionosphere model (Zhukov et al., 2021) and the GBDT-based local ionosphere model (Han et al., 2022) outperformed the commonly used ANN and LSTM approaches. The largest errors in both the ML-based and traditional linear ionosphere models were observed in the equatorial anomaly region and for space weather events such as geomagnetic storms. However, the ML models outperform the traditional linear models (e.g., PCA (Uwamahoro & Habarulema, 2015) and ARIMA (Tang et al., 2020b)).

1.6 Research Gaps

A review of previous work reveals that most of the proposed ML-based ionosphere approaches utilize different types of ANN, i.e., DL methods. However, many other ML methods have either not yet been investigated for ionosphere modeling or have been limitedly discussed. Moreover, this limited number of studies is restricted to only a few ML methods. The probable reason for this gap is that DL methods are widely known as breakthrough methods for highly complex problems, such as automotive driving, face recognition, and speech recognition. However, it can be argued that other ML methods can be applied to small data sets, as opposed to DL methods that require large datasets to reach their full potential. In this context, results from Han et al. (2022) and Zhukov et al. (2021) confirm improved performance of other ML methods for ionosphere forecasting than ANN. This is likely due to a limited training data set in Han et al. (2022) and the fact that DL models are often overparametrized, which leads to overfitting of the data (Hastie et al., 2009). As a result, a model may correspond too close to the training data to the extent that it negatively affects the model's performance on new data by reducing its ability to generalize. As the amount of data increases, DL methods for ionosphere forecasting improve their performance, as reported in Srivani et al. (2019) and Ruwali et al. (2021). In this context, dense data, e.g. GNSS observations collected over a region or a globe over a long period of time, can lead to a large amount of data for ionosphere modeling suitable for DL approaches.

How the data is partitioned and the model is validated can introduce additional biases by affecting ML model architecture and the choice of parameters and hyperparameters. In the modeling and forecasting ionosphere quantities, a time dependence between observations must be preserved when partitioning the data. Previous studies have implemented simple hold-out or classic K-fold validation techniques (Figure 1.6). In the hold-out validation procedure, the data are divided into subsets of fixed data points: training data, which is mainly 60% to 70% of the data, while the remaining part of the data is divided equally into validation and test data sets. In K-fold cross-validation, on the other hand, the data is randomly partitioned into k equally sized folds containing different training and validation data points at each iteration. K-fold cross-validation has been shown to be more accurate than the

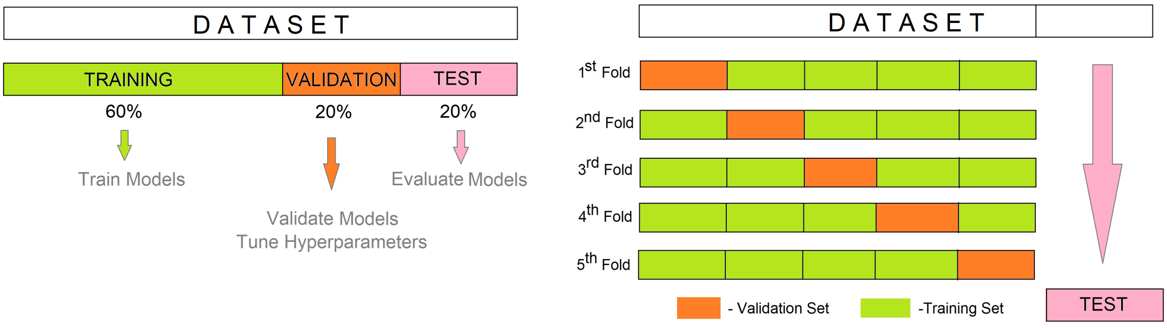


Figure 1.6: Data partitioning. Left: hold-out validation, right: K-fold validation.

simple hold-out validation because it can reduce the variance and thus the overfitting problem (Blum et al., 1999). However, simple K-fold cross-validation can be problematic when the observations are time-dependent, as it randomly splits the data. When applied to time series data, a model can be trained using the subsequent observations and predicted using the previous observations. It makes no sense to train the model using observations from the future and forecast values in the past. Also, we want to avoid looking into the future when training the model, which can lead to model bias. The aforementioned validation approaches can lead to more optimistic forecasting results and cause bias in model selection and architecture due to the abovementioned problems. For these reasons, this dissertation does not pursue the previous approaches. Instead, a new method is proposed in Section 4.2.2, K-fold time series cross-validation, shown in Figure 4.7.

The choice of input data for an ML model significantly impacts the model performance. Previous ML-based ionosphere models incorporated some observations or indices of solar and geomagnetic activity, using a similar set of input variables for most studies, e.g., geomagnetic indices Kp and Dst at the 3-hour and 1-hour resolution, respectively, and solar radio flux F10.7 at daily resolution. However, there are also models based solely on ionosphere input variables, such as in Monte-Moreno et al. (2022). The temporal resolution of earlier ML-based ionosphere models is 2 hours or 1 hour.

One of the main issues of previous work is the consideration of ML-based ionosphere models as a black box and the lack of transparency and interpretability, as there is no indication when the results cannot be trusted, which can lead to scientific skepticism toward ML and DL solutions. Despite the widespread use of ML and DL methods, there has been very little or no discussion on probabilistic ML/DL and UQ in the space weather and ionosphere domain. Previous ML-based ionosphere studies have focused only on a single prediction for each input, while probabilistic assessment has not been addressed, resulting in a lack of information on how confident and reliable the models and results are.

1.7 Research Objectives and Questions

This dissertation has three main goals:

- **Adapt and develop new ML-based ionosphere modeling and forecasting approaches to improve accuracy, explainability, and computational efficiency.**
- **Model and forecast the effects of space weather on the ionosphere, i.e., ionosphere perturbations caused by strong geomagnetic storms and powerful solar flares.**
- **Develop high-resolution ML-based probabilistic ionosphere solutions for different forecast horizons that integrate quantification of uncertainties and provide confidence intervals.**

These general goals lead to the following research objectives:

1. Not to treat and view ML model as a black box, but rather to understand the mathematical and statistical theories and concepts of the model development. In addition to the transparency and understanding of the learning algorithms, the transparency of the model training, the selection of hyperparameters, and the data on which it is trained are also essential here.
2. Contribute to closing the gap between DL and other ML methods[6] for ionosphere and space weather modeling by implementing less or not at all examined learning algorithms for this task.
3. Develop simpler, computationally efficient, and more interpretable ML approaches for ionosphere in terms of complexity, data requirements, input features contribution analysis, and parametrization.
4. Quantify the uncertainties in ML-based ionosphere models, provide information on how certain the results are and where most of the uncertainty comes from to improve the explainability and interpretability of the models and solutions.
5. Forecast the space weather effects on the ionosphere, such as geomagnetic storms and solar flares.
6. Increase the temporal resolution of ML-based ionosphere models by integrating high-resolution data.

Research questions are defined below to demonstrate how the corresponding objectives are addressed.

Q-1 What is the impact of the data on ML-based ionosphere modeling and forecasting in terms of the input features, data length, and observations resolution? On the other hand, what is the impact of using different learning algorithms?

This question is analyzed in **P-I** through **P-III** as well as in **CP-II** and Chapter 5. The data set preparation in terms of input observations, features, length, and period is discussed in Section 4.1. The computational efficiency of the ML-based ionosphere models developed with different learning algorithms is compared in Section 4.5.

Q-2 Can reduction of regular ionosphere variations and background ionosphere information in the data through daily differencing improve the learning and generalization of the ML-based ionosphere model?

In **P-II**, models are trained solely on daily differenced data and compared to models trained on the original, i.e., the non-differenced, data set. The idea behind using daily differencing is to reduce or eliminate the dominant regular daily ionosphere fluctuations and to model just the remaining signal in which other sources of fluctuations, such as space weather effects, can be more easily extracted. The Root Mean Square (RMS) analysis of ML-based ionosphere models trained with differenced or non-differenced data and their combination is presented in Section 4.1.3.

Q-3 How efficient are ML methods other than ANN, such as decision trees or ensemble learning techniques, in modeling and forecasting the ionosphere?

This question is addressed in **P-I**, **P-II** and **P-III**, where different ML methods from the decision tree, ensemble learning of decision trees to ANN are used and the results are analyzed and compared. Chapter 3 discusses the ML algorithms and modeling approaches in this dissertation, while their results are summarized in Chapter 5.

Q-4 How can data and model uncertainties be modeled, and how do uncertainties change when the ionosphere is perturbed by a geomagnetic storm?

[6] Here, DL refers to methods based solely on ANN algorithms, while other ML methods include algorithms other than ANN. Since DL is a subset of ML, DL methods are also considered as ML methods.

P-III examines this question in detail by developing different approaches and comparing them for a quiet period and a space weather period, while **CP-II** provides results of one of the developed approaches for a space weather period. Section 3.3 presents the uncertainty estimation approaches developed and applied in this dissertation, while Chapter 5 analyzes and summarizes the results.

Q-5 How do forecast accuracy and uncertainties change as the forecast horizon expands and concerning quiet and storm ionosphere conditions?

This question is analyzed in Chapter 5, which provides a detailed analysis of different forecast horizons.

Q-6 Is it possible to forecast the ionosphere response to a strong solar flare by incorporating high-resolution ionosphere, solar, and geomagnetic activity data?

An analysis is presented in Chapter 5 to answer this question.

Q-7 What is the contribution of the different input features to the model result?

P-I, **P-II**, **P-III**, and **CP-II** contain analyses of the feature importance. The method for computing feature importance is explained in Section 3.4, while the results are summarized in Section 4.6.

The answers to the above research questions are provided in Chapter 6.

1.8 Outline

This dissertation discusses various aspects and steps of ionosphere modeling and forecasting using ML techniques. A flowchart of the ML-based ionosphere model development with an overview of the processing chain is presented in Figure 1.7, with the corresponding dissertation sections discussing the respective part of the processing chain. The first-author journal and conference publications cover all levels of the processing chains, except that **P-III** and **CP-II** also include the quantification of uncertainty.

Following the processing chain, five processing levels can be identified in which the developments within the scope of this dissertation intervene:

(1) **Data level**: it includes data acquisition, selection of input observations and time periods, data preprocessing, and input feature preparation to form a suitable data set.

(2) **Modeling level**: it involves the selection of learning algorithms for developing the ML-based ionosphere model, training and cross-validating on a prepared data set, and the hyperparameters tuning. Techniques to estimate uncertainties are also incorporated.

(3) **Product level**: it represents an ML-based ionosphere model resulting from the modeling level. There are three main ionosphere products of this dissertation: (a) Regional Ionosphere Map (RIM) with ML-based spatiotemporal modeling, (b) grid point ML-based ensemble forecasting, and (c) grid point ML-based probabilistic forecasting.

(4) **Evaluation level**: a final evaluation or model test performed on the data that was not used at the modeling level.

(5) **Interpretation level**: a feature importance analysis is performed to provide information on the important input features of the developed model.

Chapter 2 provides an introduction to the ionosphere and space weather, ionosphere variability, observation and modeling techniques, as well as the necessity for ionosphere models. Afterward, this dissertation introduces new ML methods for ionosphere forecasting, such as Random Forest in Section 3.2.2, Adaptive Boosting in Section 3.2.3, Quantile Gradient Boosting in Section 3.3.3 and Bayesian Neural Network in Section 3.3.4, while GBDT and eXtreme Gradient Boosting (XGBoost) in Section

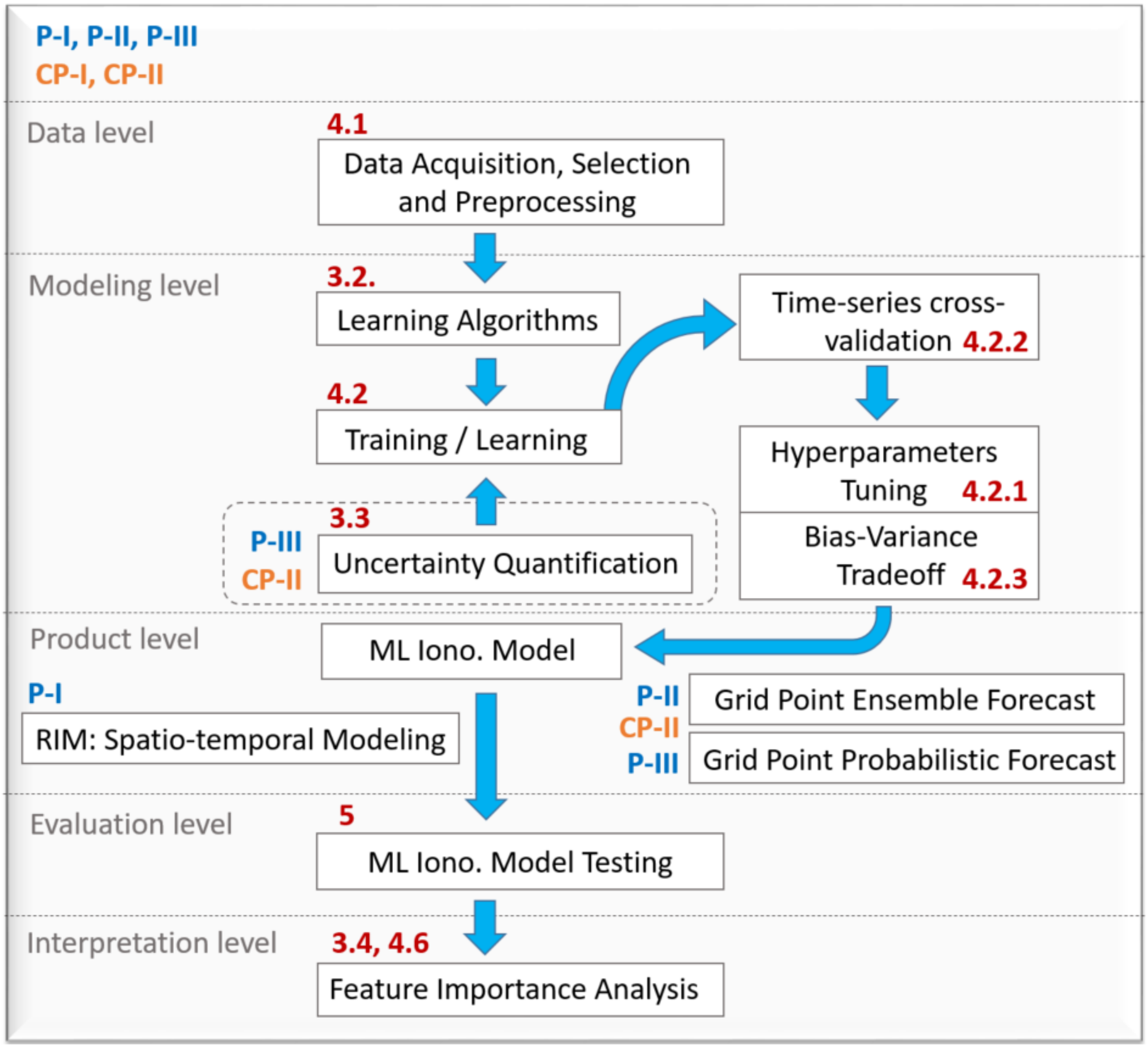

P-I, P-II, P-III
CP-I, CP-II
Data level
4.1
Data Acquisition, Selection and Preprocessing
Modeling level
3.2.
Learning Algorithms
4.2
Training / Learning
Time-series cross-validation 4.2.2
Hyperparameters Tuning 4.2.1
Bias-Variance Tradeoff 4.2.3
3.3
P-III
CP-II
Uncertainty Quantification
Product level
ML Iono. Model
P-I
RIM: Spatio-temporal Modeling
P-II
CP-II
P-III
Grid Point Ensemble Forecast
Grid Point Probabilistic Forecast
Evaluation level
5
ML Iono. Model Testing
Interpretation level
3.4, 4.6
Feature Importance Analysis

Figure 1.7: Flowchart of the processing chain for the ML-based ionosphere models development with reference to the first-author journal and conference publications in blue and orange, respectively, and the corresponding dissertation sections in dark red.

3.2.3 have previously only been reported in one paper each (Han et al., 2022; Zhukov et al., 2021). In addition, several models are combined into a super-ensemble to create a model with higher accuracy and improved generalization in Section 3.2.4. The dissertation also quantifies uncertainties in ML-based ionosphere models for the first time in Section 3.3. Thus, several UQ techniques are developed for the probabilistic forecast of ionosphere and space weather effects on the ionosphere, specifically, for forecasting the impacts of geomagnetic storms and strong solar flares on the ionosphere in this dissertation. Their performance and effectiveness in estimating uncertainties and 95% confidence intervals under quiet and extreme space weather conditions are tested and presented in Sections 5.3 and 5.4. Since the accuracy of an ML model is highly dependent on the data, this dissertation pays special attention to selecting and deriving suitable input features that can accurately describe complex ionosphere variations. Therefore, new input variables are introduced, such as solar wind plasma speed, interplanetary magnetic field index, derived features of first and second derivatives and moving averages, daily differenced data, high-resolution solar X-ray and EUV observations, a high-resolution geomagnetic activity index, and high-resolution ionosphere data (Table 3.1). The related feature engineering is presented in 4.1.3, the exploratory data analysis in 4.1.4, and the impact of data and feature selection on the developed ML-based ionosphere models are analyzed in 5.6. Section 4.1 addresses the systematic analysis, selection, and preparation of input data, data partitioning, and the choice of data time frames in a way to enhance model performance, particularly in learning signatures of space weather events. The modified cross-validation method for time series data in Section 4.2.2 further contributes to reliable training, hyperparameters tuning, and validation in this dissertation.

As mentioned earlier, one of the aims of this dissertation is not to treat and view ML as a black-box model, as is usually emphasized in previous applications of ML. Instead, the aim is to understand the mathematical and statistical theories and concepts of how the ML-based ionosphere models are developed in this dissertation, how modeling functions are "learned" from the data, to understand the results obtained, to provide information on the most important input variables in the developed models, and how and why they are selected. Furthermore, by providing information on how uncertain the results are and where most of the uncertainty comes from, the explainability of the ML-based ionosphere models and solutions in this dissertation is further increased.

2 Ionosphere Variability and Models

2.1 Ionosphere Formation and Layers

Depending on the temperature, the atmosphere is divided into the following layers: troposphere, stratosphere, mesosphere, thermosphere, and exosphere (Figure 2.1, left). The region of the atmosphere above ∼50 to 1000 km or more is also called the ionosphere and depends on ionization. It represents the ionized region of the upper atmosphere that contains free electrons and positive ions. The source of its ionization is solar radiation during the day, e.g., X-ray and Ultraviolet (UV) emissions from the Sun, and cosmic radiation at night. The ionosphere layers are defined by the electron density profile (Figure 2.1, right), and can be divided into:

- D layer: between ∼50 km and ∼90 km above the Earth;
- E layer: between ∼90 km and ∼130 km;
- F1 layer: between ∼130 km and ∼250km;
- F2 layer: above ∼250 km.

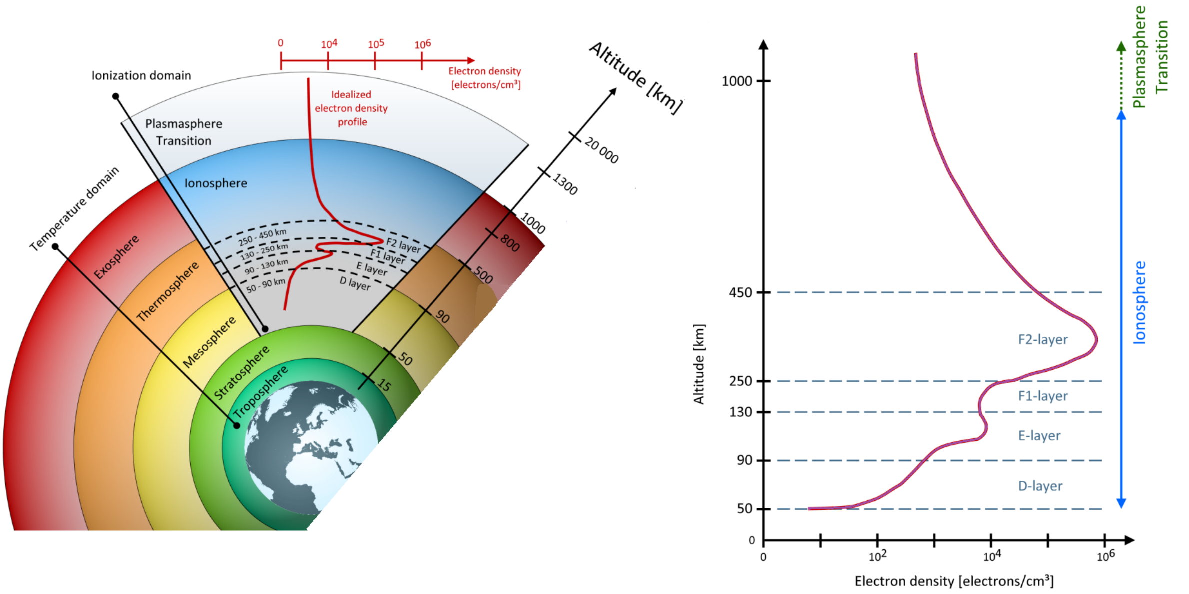


Figure 2.1: Left: Representation of atmosphere layers based on the temperature and ionization variations, including an electron density distribution profile (red). Right: Idealized electron density distribution in the Earth's ionosphere and plasmasphere, as well as the characteristic ionosphere layers during daytime. (Source: Limberger (2015)).

The most significant electron density values are given in the upper F layer, divided into the F1 and F2 layers during the daytime. The F1 layer is much weaker than F2 and merges into the higher F2 layer at night, while the D layer disappears and the E layer weakens or vanishes altogether. Within the E layer, a sporadic layer of higher electron density can occur at any time of day, preferentially in the late morning and early evening, especially in the summer months and briefly in winter, revealing strong seasonal and diurnal patterns. Unlike all the other layers, the F2 layer is present during both day and night.

The distinctive separation of layers in the ionosphere is the result of the energy of the solar spectrum at different altitudes and its absorption in the atmosphere, as well as recombination processes, depending on the atmospheric density and the composition of the upper atmosphere itself, varying with altitude (Cander, 2019). The degree of ionization is subject to substantial variations that correlate with solar activity and Earth's rotation. In addition, geomagnetic activity and the perturbative solar-terrestrial interaction also play an essential role and are discussed in the next section.

2.2 Space Weather

Space weather originates from active regions on the Sun, where many sunspots are located. Although there is no universally accepted definition, it can be described as dynamic conditions on the Sun that affect various parts of the spheres, such as the magnetosphere, ionosphere, and thermosphere, all of which form a coupled system.

The level of solar activity and the solar cycle are usually presented by solar indices such as the Sunspot number (S_n) and the solar radio flux of the solar emission at 10.7 cm wavelength (F10.7 index). S_n as defined by Rudolf Wolf (Friedli, 2016) measures the number and groups of sunspots on the Sun. The F10.7 index represents the integrated emission from the solar disc at 2800 MHz, i.e., 10.7 cm wavelength (Covington, 1969), which is closely related to the S_n, UV, and visible solar irradiance records (Tapping, 1987, 2013). The number of sunspots on the Sun increases and decreases in solar cycles of approximately 11 years (Figure 2.2). The solar minimum refers to the years when the number of sunspots is the lowest, while the solar maximum is in the years when sunspots are the most numerous. The last solar maximum occurred in 2014, with a maximum 13-month smoothed S_n of 116.4 in April 2014, representing the peak of solar activity in solar cycle 24. Thereafter, there was a gradual decline until the solar minimum in 2019. The magnitude of cycle 24 was about half the size of cycle 23. In 2020, a new solar cycle began, and we are in the ascending phase approaching the next solar maximum. Cycle 25 is predicted to reach its maximum in 2025.

Two main types of solar activity are significantly important: solar flare and CME. A solar flare is a sudden and violent explosion on the Sun that releases energy from its inner region. It can be classified based on the peak flux of X-rays, as given in Table 2.1[(1)]. The frequency of solar flares coincides

Table 2.1: Classification of solar flares from the strongest to the weakest.

Class	Intensity I $\left(\frac{Watt}{m^2}\right)$
X	$10^{-4} \leq$ I
M	$10^{-5} \leq$ I < 10^{-4}
C	$10^{-6} \leq$ I < 10^{-5}
B	$10^{-7} \leq$ I < 10^{-6}
A	I < 10^{-7}

[(1)] Within each class, there is a linear scale from 1 to 9; for instance, an X2 solar flare is twice as strong as an X1 solar flare and four times as strong as an M5 solar flare.

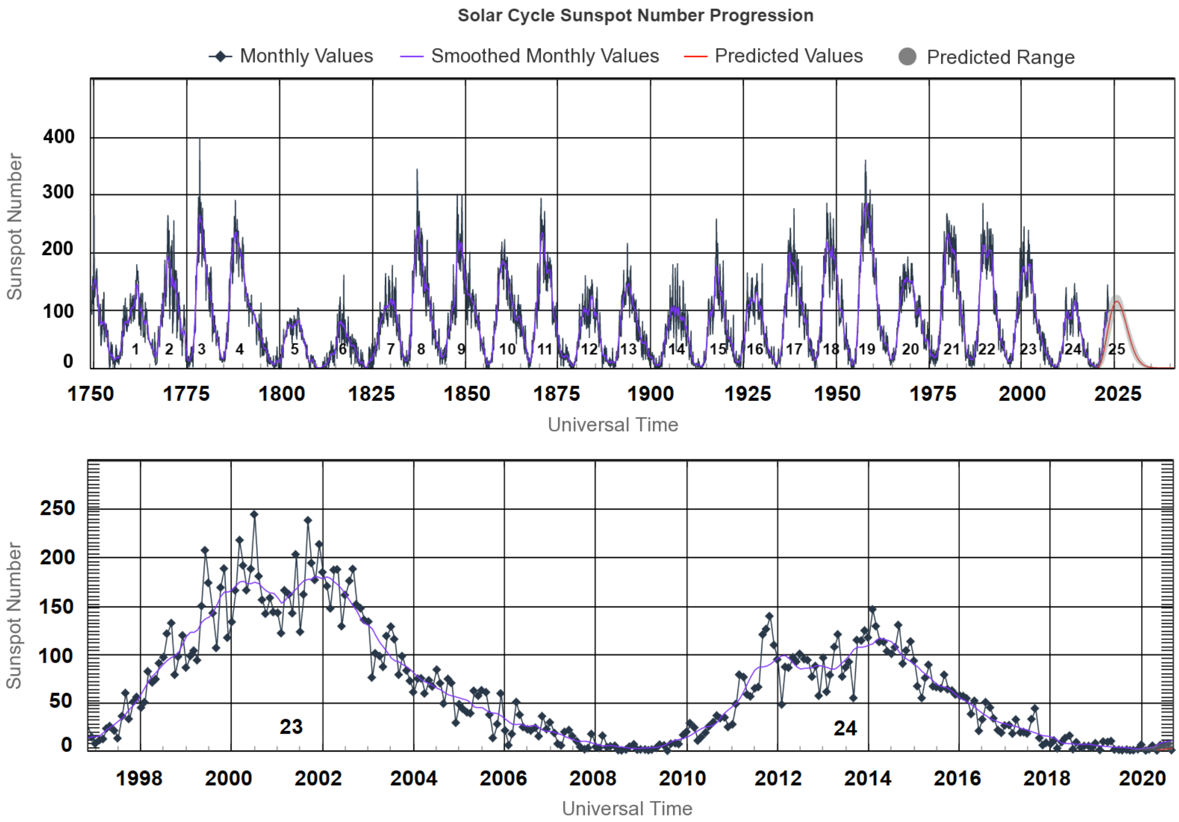


Figure 2.2: Top: the observed solar cycle S_n progression from 1750 to the present and the forecast until December 2040. Bottom: The last two solar cycles, 23 and 24. The black line with the rectangular data points represents the monthly averaged data overlaid by a 13-month weighted smoothed line of monthly averaged data. The mean forecast for the current solar cycle 25 is given by the red line with the gray-shaded region of expected uncertainties. (Source: National Oceanic and Atmospheric Administration, USA (NOAA), Boulder, USA.)

with the 11-year solar cycle and increases as the solar maximum approaches. When analyzing the impact of a solar flare on the Earth, i.e., its geo-effectiveness, two essential factors must be considered: the intensity of a flare (Table 2.1) and the location on the solar disk. The Extreme Ultraviolet (EUV) enhancement is lower for flares occurring near the solar limb, resulting in lower geo-effectiveness, i.e., lower ionosphere response, than for flares occurring in the center of the solar disk[(2)] (Qian et al., 2010; Hernández-Pajares et al., 2012a). The reason is that the solar atmosphere absorbs the UV radiation, and its absorption is larger at the limb of the solar disk due to the longer path lengths (Hernández-Pajares et al., 2012a). However, the X-ray flux remains unaffected. This means that EUV observations are affected by the location on the solar disk, while X-ray observations reflect only the intensity of a flare (Table 2.1), regardless of its position. In other words, a strong solar flare occurring at the solar limb will have a very poor geo-effectiveness, as in the case of the X28.0 class flare on November 4, 2003 (see Hernández-Pajares et al. (2012a)). The expected geo-effectiveness for a limb flare is 2 to 3 times weaker than that at the center of the solar disk (Qian et al., 2010). In addition, the radiation photons directed towards the Earth from a solar flare cause a sudden increase in ionization in the daylight ionosphere, depending on the solar zenith angle (Hernández-Pajares et al., 2012a).

On September 6, 2017, the Sun emitted an X9.3 flare, the strongest in solar cycle 24. The X9.3 flare was

[(2)] The EUV irradiance depends strongly on the distance of the solar flare location from the center of the solar disk.

accompanied by solar radio emission and caused an increase in the dayside ionosphere (Berdermann et al., 2018; Yasyukevich et al., 2018). Furthermore, the Precise Point Positioning (PPP) error increased by three times, as reported in Yasyukevich et al. (2018), impacting both single and dual frequency PPP, thus severely affecting GNSS positioning performance and navigation services in Europe (Berdermann et al., 2018). These solar flares led to severe geomagnetic storms on September 8. The main findings from previous studies of the ionosphere response to a solar flare can be summarized as follows:

- Ionization suddenly increased during the X9.3 flare occurrence at 12 Coordinated Universal Time (UTC) on September 6, 2017 and lasted until 17 UTC (Yasyukevich et al., 2018). The increase was much higher, at least about three times, in the ionosphere information derived directly from the GPS observations than in the Global Ionosphere Map (GIM) (Yasyukevich et al., 2018).

- The dependence between the increased ionization during a solar flare in the daylight ionosphere and the solar zenith angle is reported in Barta et al. (2022); Hernández-Pajares et al. (2012b); Yasyukevich et al. (2018).

- The most pronounced solar flare-induced effects on the ionosphere are observed around local noon when the solar zenith angle is close to zero (Liu et al., 2004).

- **C-I** reported a latitude-dependent increase in ionization of 10% to 20% during a solar flare, with the duration of the high ionisation increasing with decreasing latitude.

- A pure solar flare impact is expected to occur in a mid-latitude ionosphere region (Berdermann et al., 2018) because there are no overlapping disturbances as in the polar or equatorial ionosphere regions.

- The geo-effectiveness of solar flares depends on the flare distance from the center of the solar disk (Hernández-Pajares et al., 2012b; Qian et al., 2010), being 2-3 times stronger for a disk-center flare than for a limb flare (Qian et al., 2010).

Solar flares can be accompanied by CMEs, which are ejections of magnetized solar plasma that usually reach the Earth within 1 to 3 days and interact with the Earth's magnetosphere (Figure 1.2). A large cloud of plasma particles and the magnetic field, erupting from the Sun, plows through the constantly outflowing ionized plasma of the upper solar atmosphere, the so-called solar wind. When the cloud reaches the Earth, it can induce increased energy into the GeoMagnetic Field (GMF), leading to temporary disturbances in known as geomagnetic storms. These storms can be quantified by indices such as the Kp index, derived from geomagnetic observatories between 44°and 60°N or S of geomagnetic latitude (Bartels, 1963), and the Dst index derived from a network of geomagnetic observatories near the equator (Sugiura, 1964). The effect of a storm visually manifests at high latitudes in the form of a bright aurora around local midnight, representing remarkable manifestations of solar-terrestrial interaction.

As mentioned earlier, space weather events, including associated geomagnetic storms and ionosphere disturbances, become more frequent as the solar maximum approaches. The probability of such events during solar maximum is much larger because of the high number of sunspots, but their occurrence during solar minimum is not excluded. The strongest space weather events in solar cycle 24 were the St. Patrick's storm in 2015, occurring near solar maximum, as well as a chain of intense solar flares and CMEs for several consecutive days in September 2017 during the declining phase of the solar cycle. However, this is not the only example of severe space weather events in the low phase of a solar cycle. The most giant recorded solar storm to date, the Carrington event, occurred during solar minimum in September 1859 (Hudson, 2021; Tsurutani et al., 2012).

2.3 Observation Techniques

Electromagnetic measurements used in geodetic applications such as satellite-based positioning and navigation are influenced by free electrons in the Earth's ionosphere. With the modernization of satellite systems and the use of multi-constellation and multi-frequency measurements, the ionosphere effects can be reduced, and the ionosphere can continuously be monitored using globally distributed sensor stations. The following geodetic satellite observation techniques are used to model the ionosphere:

- GNSS : US NAVigational Satellite Timing And Ranging (NAVSTAR) GPS, the Russian GLObal'naya NAvigatsionnaya Sputnikovaya Sistem (GLONASS), the European Galileo, and the Chinese BeiDou;
- Satellite altimetry(3): e.g. TOPEX-Poseidon, Jason-2 and Jason-3;
- Doppler Orbitography and Radiopositioning Integrated by Satellite (DORIS)(4): e.g. Jason-2 and Jason-3, Cryosat-2;
- Radio occultation (RO) measurements(5): e.g. Formosat-3/COSMIC, CHAllenging Minisatellite Payload (CHAMP).

The different space-geodetic observation techniques can be combined for more accurate ionosphere modeling by exploiting the advantage of their different spatial and temporal distributions and the different observation characteristics of each technique, e.g., Dettmering et al. (2011). In this dissertation, ionosphere modeling and forecasting are performed using only ionosphere information derived from GNSS observations.

GNSS refer to a constellation of satellites from all existing global navigation satellite systems that provide worldwide coverage and transmit signals to GNSS receivers for positioning, navigation, and timing. They currently include GPS, GLONASS, Galileo, and the BeiDou navigation satellite systems. GNSS consist of three segments:

1. Space segment: it comprises a constellation of satellites in different orbital planes that transmit the navigation message required for user positioning and navigation, which contains information on satellite orbits and clocks, satellite health information, and ionosphere parameters of the Klobuchar model(6).
2. Ground or control segment: it is responsible for monitoring the satellites and their signal transmissions, performing analyses, estimating satellite orbits and clock errors, and generating the navigation message sent to the space segment.
3. User segment: it includes the GNSS receivers that can be used anywhere around the world. The user only receives the signals from the satellites and processes them without communicating with the satellite.

GPS was originally developed for military purposes, but today GNSS signals are mostly used for civilian purposes such as surveying, precision agriculture, and navigation in cars, aircraft, ships, smartphones, etc. Although the primary application of GNSS is positioning and timing, GNSS is today also used for remote sensing of the atmosphere by providing permanent monitoring and global coverage that can be used for weather forecasting and climate modeling in both the troposphere and

(3) For more information on the concepts of satellite altimetry and the TOPography EXperimentTOPography EXperiment (TOPEX)/POSEIDON mission, see Chelton et al. (2001), and on the combination of satellite altimetry and GNSS for ionosphere modeling, see Todorova et al. (2008).

(4) For more information on the system, see Jayles et al. (2010); Auriol & Tourain (2010), and on its use for ionosphere modeling, see Dettmering et al. (2014).

(5) For more information on the RO atmosphere sounding, see Wickert et al. (2001), and its use for ionosphere studies, see Jakowski et al. (2002); Arras et al. (2008).

(6) For more details on the broadcasted Klobuchar model, see Section 2.6.

ionosphere. Apart from affecting GNSS accuracy and performance, the information obtained from the GNSS signals provides the opportunity to monitor and study the ionosphere and space weather processes to understand and model their variations and mechanisms.

It is relevant to mention permanent ground-based networks of continuously operating GNSS reference stations. The International GNSS Service (IGS) operates a global network of more than 500 permanent quality stations. Regional GNSS station networks focus on specific regions, such as the EUREF Permanent Network (EPN), with stations distributed exclusively in Europe. These two networks provide freely and openly accessible high-precision GNSS data, products, and services in support of the terrestrial reference frame, Earth observation and research, monitoring of tectonic deformations and sea-level variations, positioning, navigation and timing, and other applications that benefit the scientific community and society. Another permanent ground-based GNSS network type is the Continuous Operating Reference Stations (CORS), which provides continuous real-time positioning data for a given area. Government agencies, universities, or other organizations typically install and maintain them. The GNSS data collected by the CORS stations are processed and transmitted to users via the internet, usually through a web-based interface, allowing them to access high-precision positioning data. The CORS networks consist of dense stations that provide more comprehensive GNSS coverage and higher accuracy and reliability. However, not all CORS providers make their data available for free, so subscription-based pricing options are available depending on service need and length of use. CORS are used for surveying, mapping, and geodetic research, among other applications, as well as to support autonomous vehicles and provide high-precision positioning data to other GNSS receivers in the area.

2.4 Impact on GNSS Signals

Free electrons in the ionosphere interfere with the propagation of microwave signals generated by communication and navigation systems such as GNSS, causing signal delay and bending. From Table 2.2, it can be seen that the ionospheric error can be 4 m (Table 2.2), making it the dominant GNSS error source.

Table 2.2: GNSS ranging error sources (Parkinson, 1996).

Source	Error range (m)
Ephemeris data	2.1
Satellite clocks	2.1
Ionosphere	4.0
Troposphere	0.7
Multipath	1.4
Receiver measurement	0.5

Multi-frequency observations allow the reduction of the ionospheric range error by forming an ionosphere-free linear combination L_3 as

$$L_3 = \frac{1}{f_1^2 - f_2^2}\left(f_1^2 L_1 - f_2^2 L_2\right), \tag{2.1}$$

where f_1, f_2 are the frequencies associated with the carriers L_1 and L_2. The L_3 combination can eliminate the first-order ionospheric term, which accounts for more than 99.9% of the total ionospheric delay associated with phase and code GNSS measurements (Hernández-Pajares et al., 2011). However, after applying the L_3 combinations, the Higher Order Ionosphere terms (HOI) remain, which can cause GNSS errors. Their magnitude depends mainly on the degree of solar activity and the geomagnetic and ionosphere conditions (Hoque & Jakowski, 2008). They usually can be neglected, but in extreme space weather conditions, they can reach tens of centimeters, especially at low altitude angles. So even precise techniques such as the dual-frequency PPP and Real-Time Kinematic (RTK) can experience a decrease in performance under extreme space weather conditions and ionosphere disturbances (see, e.g., Jacobsen & Andalsvik (2016); Luo et al. (2018b); Bergeot et al. (2011)). On the other hand, mass-market GNSS receivers typically operate at a single frequency, meaning that the ionospheric range error must be corrected or mitigated by an external correction.

Moreover, space weather events can cause irregularities in electron density, such as ionospheric scintillation. It leads to diffraction and scattering of GNSS signals, which can exhibit severe amplitude fading and random carrier phase fluctuations, increasing range measurement errors, the probability of signal lock loss, and signal acquisition failure (Linty et al., 2018). GNSS scintillation occurs most frequently near the magnetic equator during solar maximum but can also occur anywhere on Earth during any phase of the solar cycle (Kintner et al., 2007). In addition, the dispersive nature of the ionosphere slows down a higher-frequency signal less than a lower-frequency signal. For example, the f_1 frequency is less affected by the ionosphere scintillation than the f_2 frequency, which in turn is less affected than the f_5 frequency.

2.4.1 Total Electron Content

The ionosphere's effect on GNSS signal propagation can be characterized by the Slant Total Electron Content (STEC), which represents the integrated electron density along the signal ray path in the ionosphere. It is defined as

$$STEC = \int_r^p N_e \, ds \tag{2.2}$$

where N_e is the electron density along the signal ray path between the satellite p and the receiver r. STEC is measured in Total Electron Content Unit (TECU), where 1 TECU= 10^{16} electrons/m^2.

The ionosphere models derived from GNSS observations usually assume that the ionosphere layer is a shell of infinitesimal thickness in which all free electrons are concentrated (Schaer, 1999). This approach is referred to as Single Layer Model (SLM) and is shown in Figure 2.3. The intersection point of the line of sight between the satellite and the receiver with the ionosphere layer is called the Ionospheric Pierce Point (IPP). When modeling the ionosphere, STEC can be mapped to the Vertical Total Electron Content (VTEC) as follows

$$F(z) = \frac{STEC}{VTEC} = \frac{1}{\cos z'} \quad \text{with} \quad \sin z' = \frac{R}{R+H} \sin z \tag{2.3}$$

where

$F(z)$ is the SLM mapping function that depends on the elevation,

H is the mean altitude of the ionosphere shell, i.e., the SLM height,

z' is the zenith angle of the signal path at altitude H,

R is the mean radius of the Earth.

It is important to note that there are also many other mapping functions $F(z)$, such as the modified SLM mapping function (Schaer, 1999), Barcelona Ionospheric Mapping Function (BIMF) (Lyu et al., 2018), and mapping function with ionosphere varying height (Xiang & Gao, 2019).

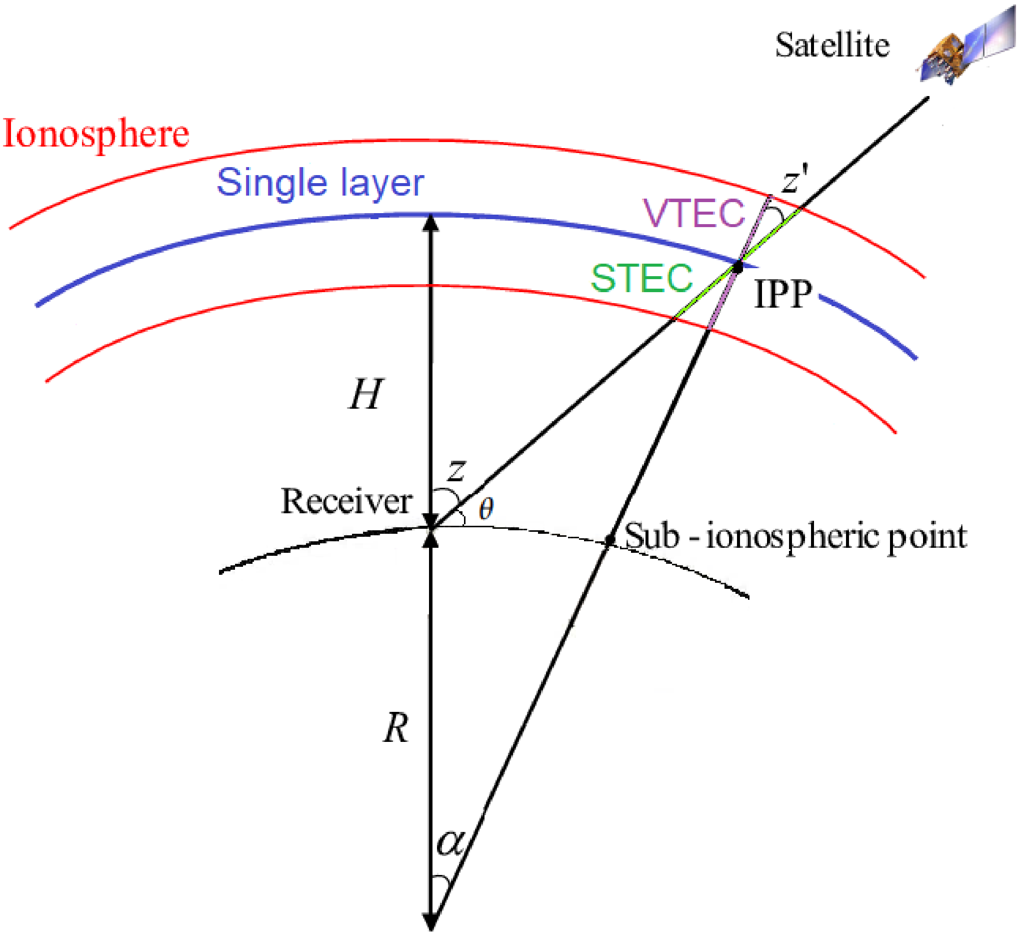


Figure 2.3: Geometry of the SLM for the ionosphere modeling.

The SLM height is usually set to the expected height of the maximum electron density. It is typically between 350 km to 450 km (Schaer, 1999; Jiang et al., 2017; Mannucci et al., 1998), while the IGS analysis centers have adopted a fixed height of 450 km for the GIM (Feltens, 2003). From Equation (2.3) it follows

$$VTEC = STEC \cos \left[\sin^{-1} \left(\frac{R \cos \theta}{R + H} \right) \right], \tag{2.4}$$

where θ is the elevation angle at the receiver location. VTEC has been widely used to model, study and monitor the ionosphere and correct GNSS observations for ionosphere effects in positioning.

2.4.2 Geometry-free Linear Combination L_4

Dual-frequency GNSS observations facilitate the estimation of the ionosphere range delay by subtracting observations at different frequencies, i.e., by forming the geometry-free linear combination L_4. This means that all frequency-independent effects, such as the satellite-receiver geometrical range, clock errors, and tropospheric delay, among others, are canceled, while the ionospheric delay is preserved (Ciraolo et al., 2007). The L_4 combination allows the estimation of VTEC as a function of geographic or geomagnetic latitude ϕ_{IPP} and Sun-fixed longitude s_{IPP} at IPP. The observation equation can be outlined as

$$L_4 = L_1 - L_2 = -\alpha \cdot \left(\frac{1}{f_1^2} - \frac{1}{f_2^2} \right) \cdot F(z) \cdot VTEC\left(\phi_{IPP}, s_{IPP}\right) - \beta \cdot \phi + B_4, \tag{2.5}$$

where

α is a constant with $\alpha = 4.03 \cdot 10^{17} \frac{m}{s^2 TECU}$,

f_1, f_1 are the frequencies associated with the carriers L_1 and L_2,

$\beta \cdot \phi$ represents the wind-up term associated with the right-handed polarized GPS signal, typically a centimeter-scale term,

B_4 is a constant phase bias with $B_4 = \lambda_1 \cdot B_1 - \lambda_2 \cdot B_2$ caused by the initial phase ambiguities B_1 and B_2 with their corresponding wavelengths λ_1 and λ_2, i.e., an unknown integer number of cycles. The initial phase ambiguities B_1 and B_2 are estimated as real-value parameters in the initial least-squares adjustment using the L_1 and L_2 observations at both frequencies.

The carrier phase ambiguity term can be estimated by the carrier phase leveling (Mannucci et al., 1998). In this method, the carrier phase measurements are "leveled" to the geometry-free combination of code pseudoranges by averaging the difference between them for each continuous arc of the samples (see, e.g., Wang et al. (2020); Goss (2021)).

2.5 VTEC variability

VTEC is a highly variable quantity concerning local time, season, receiver location, phase of the solar cycle, solar activity, space weather processes, geomagnetic conditions, etc. To analyze the regular and irregular VTEC variations in **CP-III**, VTEC values were derived[7] from GNSS (GPS+GLONASS) observations of the EPN station SRJV, located in Sarajevo $(43.87°, 18.41°)$ with a time sampling of 30 seconds. In addition, the SRJV station coordinates were estimated using dual-frequency PPP to analyze the impact of the ionosphere on coordinate estimation during severe space weather.

Section 2.5.1 gives an overview of the geographical classification of the ionosphere regions. This is followed by Sections 2.5.2 - 2.5.5 that examine daily, monthly, seasonal, and solar cycle VTEC variations for the period from January 2013 to December 2016, as well as the space weather effects on the ionosphere and positioning performance during the severe geomagnetic storm in 2015.

2.5.1 Geographical VTEC Variability

The ionosphere exhibits considerable geographical variations depending on the geomagnetic latitudes. The generally accepted division of the ionosphere is into high-latitude/auroral region, mid-latitude region, and low-latitude/equatorial region (Figure 2.4). There are also transitional regions between the mid-latitude and auroral regions. The high-latitude ionosphere extends over the 60°of geomagnetic latitude. However, during geomagnetic storms, it can expand toward the equator and reduce the width of the mid-latitude ionosphere region. The low-latitude ionosphere lies within 20° of geomagnetic latitude and includes the equatorial anomaly. An equatorial anomaly or equatorial plasma bubble occurs at the magnetic equator up to about ±15° latitude. It includes the higher-density ionosphere irregularities associated mainly with the equatorial spread of the F layer, which develops shortly after sunset at equator ∼19:00 Local Time (LT), rises vertically, extends poleward, and reaches the anomalies a few hours later.

[7] For more details, see Section 4.1.1 in this dissertation or Section 2.1 in **CP-III** Natras et al. (2023b).

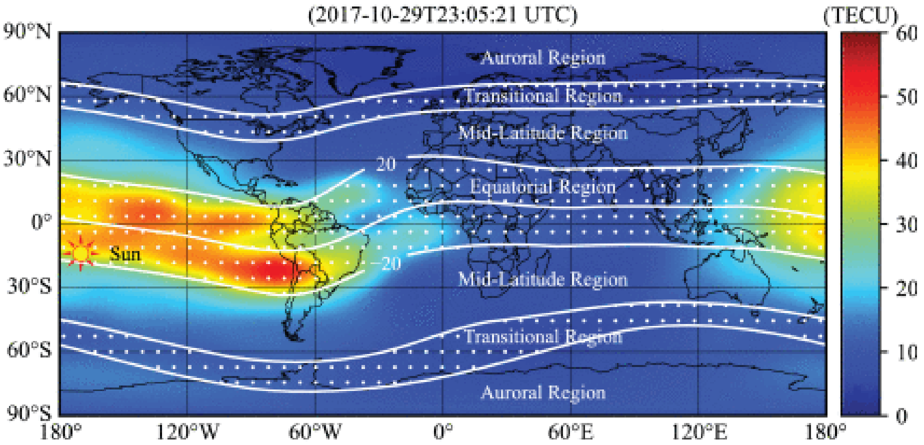


Figure 2.4: Ionosphere regions according to their geomagnetic latitudes overlaid on a GIM from the NASA Jet Propulsion Laboratory, USA (JPL). (Source: https://eartharxiv.org/repository/view/1092/).

2.5.2 Diurnal VTEC Variability

The diurnal VTEC values within each month with the average VTEC of each month and the standard deviations are analyzed for the solar maximum year in Figure 2.5. The diurnal VTEC variations are

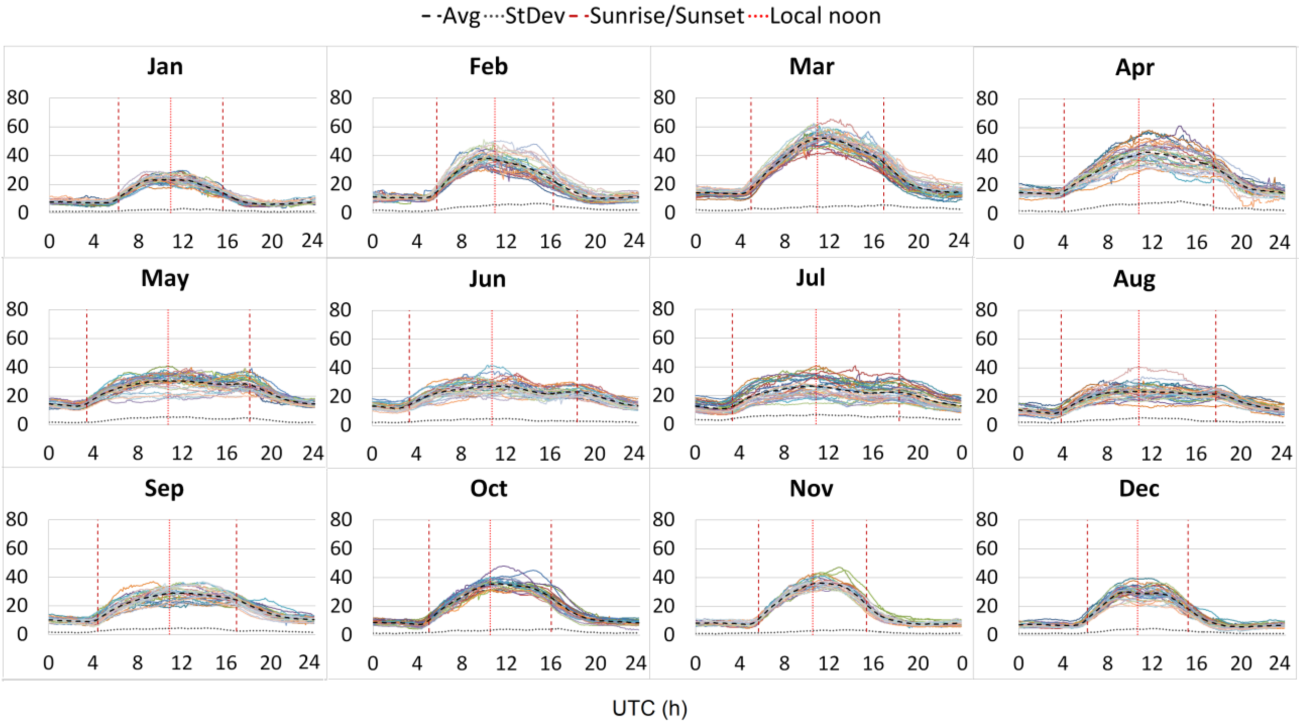


Figure 2.5: VTEC from the GNSS EPN SRJV from January to December 2014. The 24-hour VTEC values for each day are presented with solid lines in different colors. Dashed black line: the average VTEC of each month. Dotted black line: VTEC standard deviation.

characterized by lower values at night and higher values during the day. More specifically, VTEC reaches its lowest value before sunrise and its highest around local noon, from 12:00 up to 14:00 UTC. In between, there is a gradual rise and fall from sunrise to noon and from noon to sunset, respectively. The longer the days, the wider the 24-hour VTEC curve, and vice versa. This effect is directly related to the solar irradiance and zenith angle, as the daily variations in the ionosphere are mainly controlled by fluctuations in the solar EUV/UV irradiance (Vaishnav et al., 2021).

The magnitude of peak VTEC values varies monthly. In the first half of the year, January - April 2014, there is an evident increase in the peak VTEC values, while the VTEC curve flattens from May to August, only to rise again from September - October. The VTEC variations in the different months are analyzed in more detail in the next section.

2.5.3 Solar Cycle and Monthly VTEC Variability

The largest monthly VTEC values and standard deviations within the period from January 2013 to December 2016 are recorded in 2014, i.e., in the solar maximum year (Figure 2.6). Moderate VTEC values and standard deviations can be observed in the previous and the following years, i.e., 2013 and 2015. The months closer to the solar maximum have higher VTEC values, such as August to

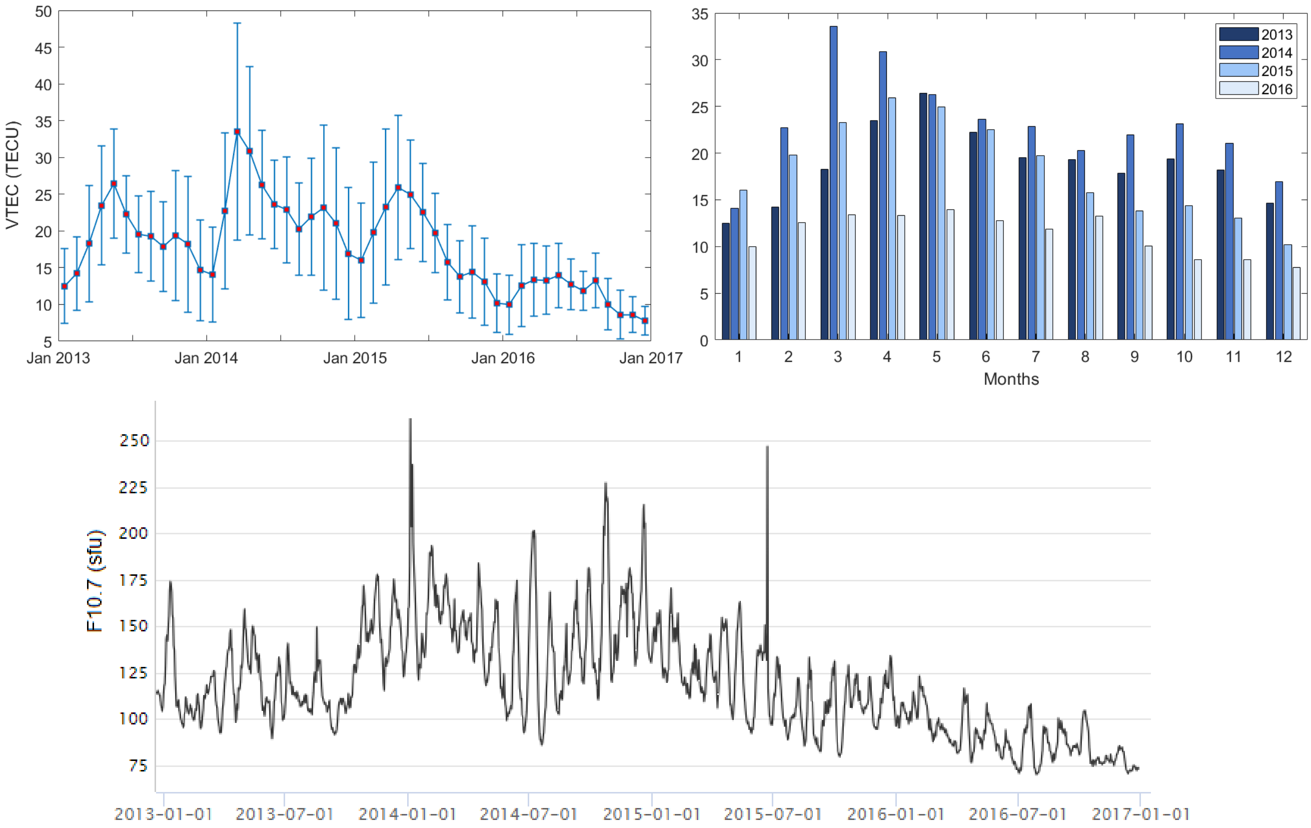


Figure 2.6: Top left: monthly VTEC values of the GNSS EPN SRJV with standard deviations from January 2013 to December 2016. Top right: bar chart of monthly VTEC for EPN GNSS SRJV for the years 2013, 2014, 2015, and 2016. Bottom: NOAA observed F10.7 solar radio in solar flux unit (sfu) from January 2013 to December 2016.

December in 2013 and January to July in 2015. These months are characterized by an increased solar activity, reflected in the F10.7 solar radio flux. A significant decrease in monthly VTEC values and their standard deviations are observed in 2016. They are at least twice as low as in the year of solar maximum. This is because the year 2016 represents the descending phase of the solar cycle, characterized by lower solar activity, as shown by the F10.7 index, leading to a decrease in VTEC values. These results show the apparent effect of the solar cycle, as the VTEC values and their standard deviations increase and decrease significantly as the solar cycle progresses.

The months with the highest VTEC values within a year are March, April, and May, while the lowest values are in January and December. A VTEC increase can be observed from January to spring. Thereafter, VTEC values decrease from June until September, when they slightly increase again until they decrease in December. These trends result from seasonal VTEC variability.

2.5.4 Seasonal VTEC Variability

To analyze seasonal VTEC variations, the months are divided into four groups (Figure 2.7) as

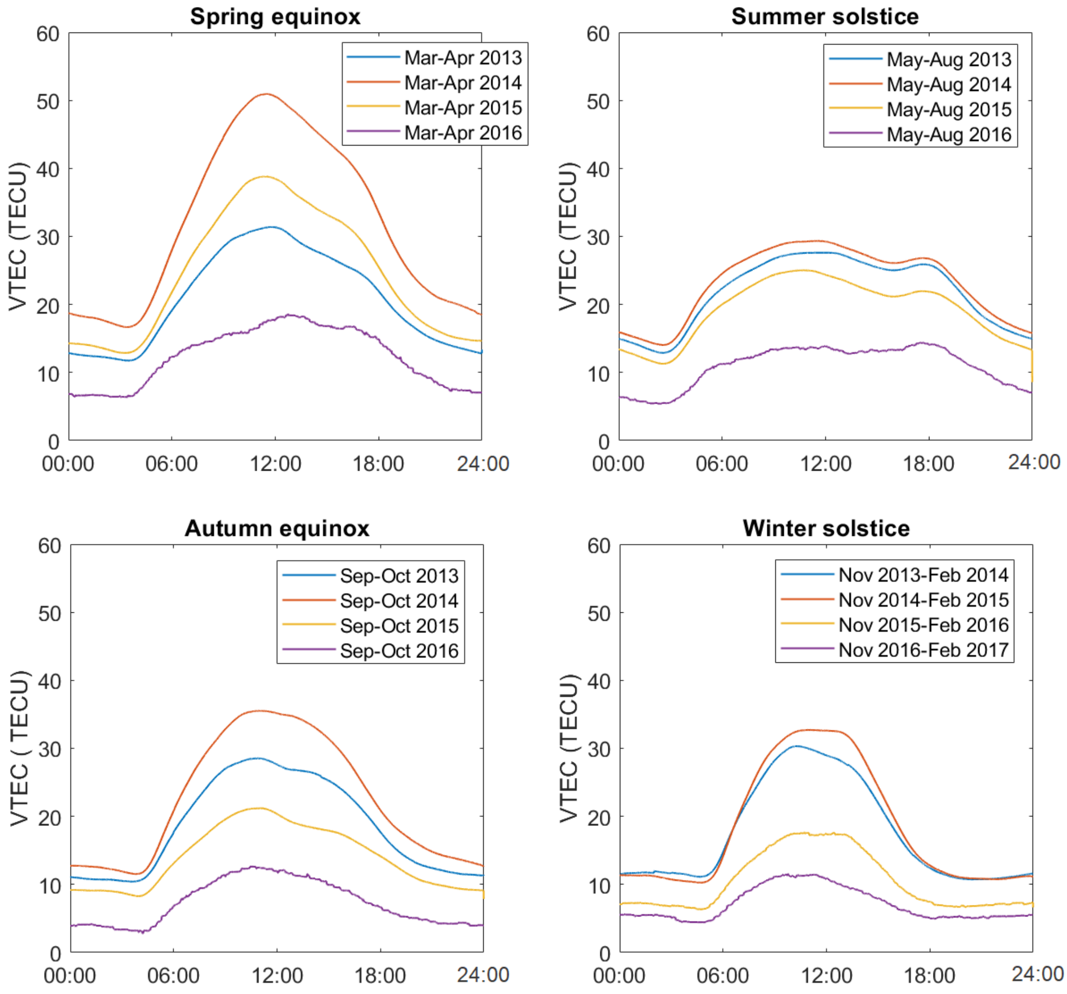


Figure 2.7: Seasonal VTEC variability with diurnal variations: spring equinox (top left), summer solstice (top right), autumn equinox (bottom left), winter solstice (bottom right).

- Spring equinox: March and April;
- Summer solstice: May, June, July and August;
- Autumn equinox: September and October;
- Winter solstice: November, December, January and February;

and VTEC is averaged in each group to obtain the seasonal VTEC values as in Cander (2019).

The highest seasonal VTEC values are in 2014, followed by 2013, 2015, and 2016, from the highest to the lowest seasonal VTEC values. The exception is the spring equinox, where the VTEC values of 2015 are higher than those of 2013. In 2014, the spring equinox VTEC values increase to over 50 TECU around local noon, which corresponds to a total daily increase of around 30 TECU. In contrast, VTEC in 2016 is below 20 TECU during the day, with the total daily increase being twice as low.

Comparing the different seasons, the VTEC values are the highest during the spring equinox and show an enormous change from the lowest to the highest values within 24 hours, while they are much lower during the other seasons. The summer solstice VTEC is higher than the winter solstice VTEC in 2015 and 2016, but in 2013 and 2014, the situation is reversed. This effect is known as the winter anomaly (Pandit et al., 2021), which is usually observed during higher solar activity. It occurs due to the Earth's shorter distance from the Sun (Shimeis et al., 2014), the seasonal variations in thermospheric O/N_2 concentration[8] (Zhang et al., 2005), and the transport of ionization from the summer to the winter hemisphere (Pandit et al., 2021), resulting in a change in the composition of ionosphere constituents and a lower electron recombination rate in winter. Consequently, VTEC is higher at the winter solstice than at the summer solstice during high solar activity.

2.5.5 Geomagnetic Storm and Impact on PPP

VTEC variations during a geomagnetic storm, including the effects on the PPP, are presented for the case of one of the strongest geomagnetic storms of solar cycle 24, known as the St. Patrick's storm. The storm was characterized by a main phase on March 17, 2015 during which the Kp index reached a value of 8 and the Dst index was less than -200 nT. This was followed by a recovery phase on March 18, which lasted several days (see Figure 8 in **P-I** and Figure 8 in **CP-III**). The Earth remained under the high-speed solar wind flow until March 27, resulting in unsettled to sporadically active geomagnetic conditions.

After the arrival of the CME in the Earth's magnetic field and the onset of the geomagnetic storm, the VTEC fluctuates considerably (Figure 2.8). During the main storm phase on March 17, an increase in VTEC can be observed with two significant peaks: more than 50% shortly after local noon and 150% in the evening, i.e., shortly before 18:00 UTC. The following recovery days are characterized by VTEC decrease of 50% around local noon and 60% to 80% at night. Such anomalous VTEC variations during a space weather event are the result of physical processes such as the change in thermospheric composition O/N_2, ionospheric disturbance dynamo, prompt penetration of electric fields, thermospheric winds, or their combination (Astafyeva et al., 2015; Nava et al., 2016).

The positioning errors and their daily RMS values in the east, north, and up components of the EPN station SRJV are estimated from the 300-sec PPP solutions processed in the Bernese GNSS software and the weekly combined EPN solutions according to Equations (9) and (10) in **CP-III**, and shown in Figure 2.9. In March 2015, positioning errors are mostly within 4 cm in the east component, within 2 cm in the north component, and below 10 cm in the up component. They increase suddenly from March 5 to 7 and March 18 to 23. The first period of increase, March 5 - 7, coincides with heavy

[8] The thermospheric O/N_2 (density ratio of atomic oxygen number to molecular nitrogen number) is an important indicator for determining the prevailing composition and is used to study the perturbations during geomagnetic storms.

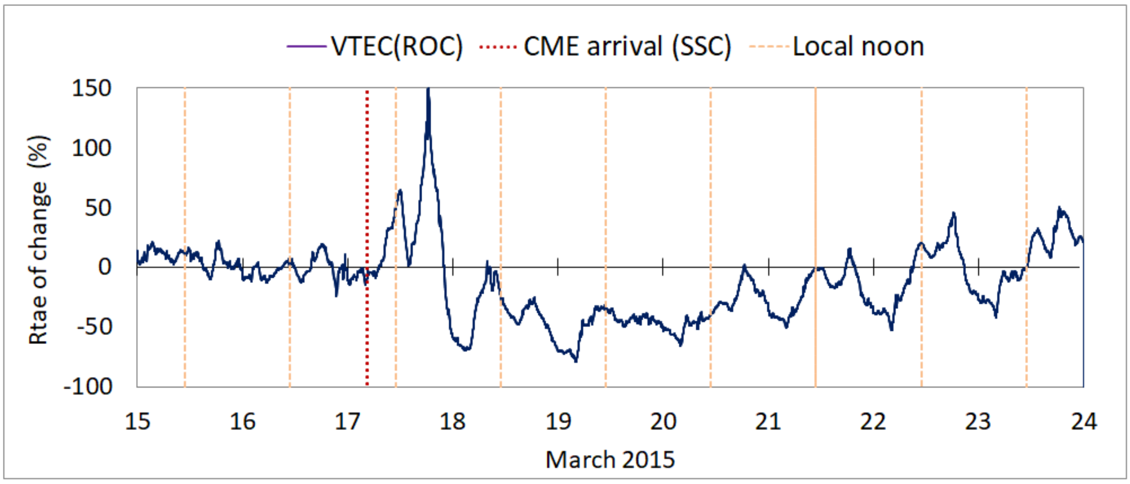


Figure 2.8: VTEC rate of change (ROC) between the observed and regular VTEC for March 15-23, 2015, calculated from Equation (7) in **CP-III**. The regular VTEC is the average over the five days of the quietest geomagnetic activity in March 2015. The orange dashed line indicates the local noon time, while the red dotted line represents the arrival of the CME on Earth, signifying the Sudden Storm Commencement (SSC).

snowfall in Sarajevo, where precipitation and accumulation of snow on and around the GNSS antenna contributed to the amplification of signal scattering and degradation of positioning performance (Natras et al., 2019b). Thus, the cause of coordinate deviations there is not the ionosphere but the meteorological conditions in the lower atmosphere, and further discussion of these effects is beyond the scope of this dissertation. The largest errors after the CME arrival are 6 cm, 7 cm, and 20 cm for the east, north, and up components, respectively. This represents an increase of 1.5 times for the east component, three times for the north component, and more than two times for the up component in terms of positioning errors for the entire March 2015. Consequently, the daily RMS values increase by up to 5 cm in the up component and about 2 cm in the east and north components.

2.5.6 Summary

The analyses in Sections 2.5.2 to 2.5.5 show diurnal, monthly, seasonal, and solar cycle phase-dependent VTEC variations, as well as sudden and intense irregular VTEC changes during a geomagnetic storm. The VTEC variability follows the solar cycle progression, which is most significant during solar maximum and gradually decreases towards the solar minimum. As for the seasonal dependence, an increasing VTEC trend is observed in the equinox months, with the largest VTEC values and standard deviations during the spring equinox, while the decrease occurs mainly in the solstice months. In addition, an ionospheric winter anomaly is detected during solar maximum. Space weather caused severe perturbations in the ionosphere, leading to significant deviations of 50% to 150% from regular ionosphere conditions and affecting the accuracy of the dual-frequency mid-latitude GNSS positioning, resulting in increased coordinate deviations. These deviations can be attributed to HOI terms that the L_3 linear combination cannot remove. Therefore, the space weather effects cannot be ignored in the study of the ionosphere as well as in the GNSS applications.

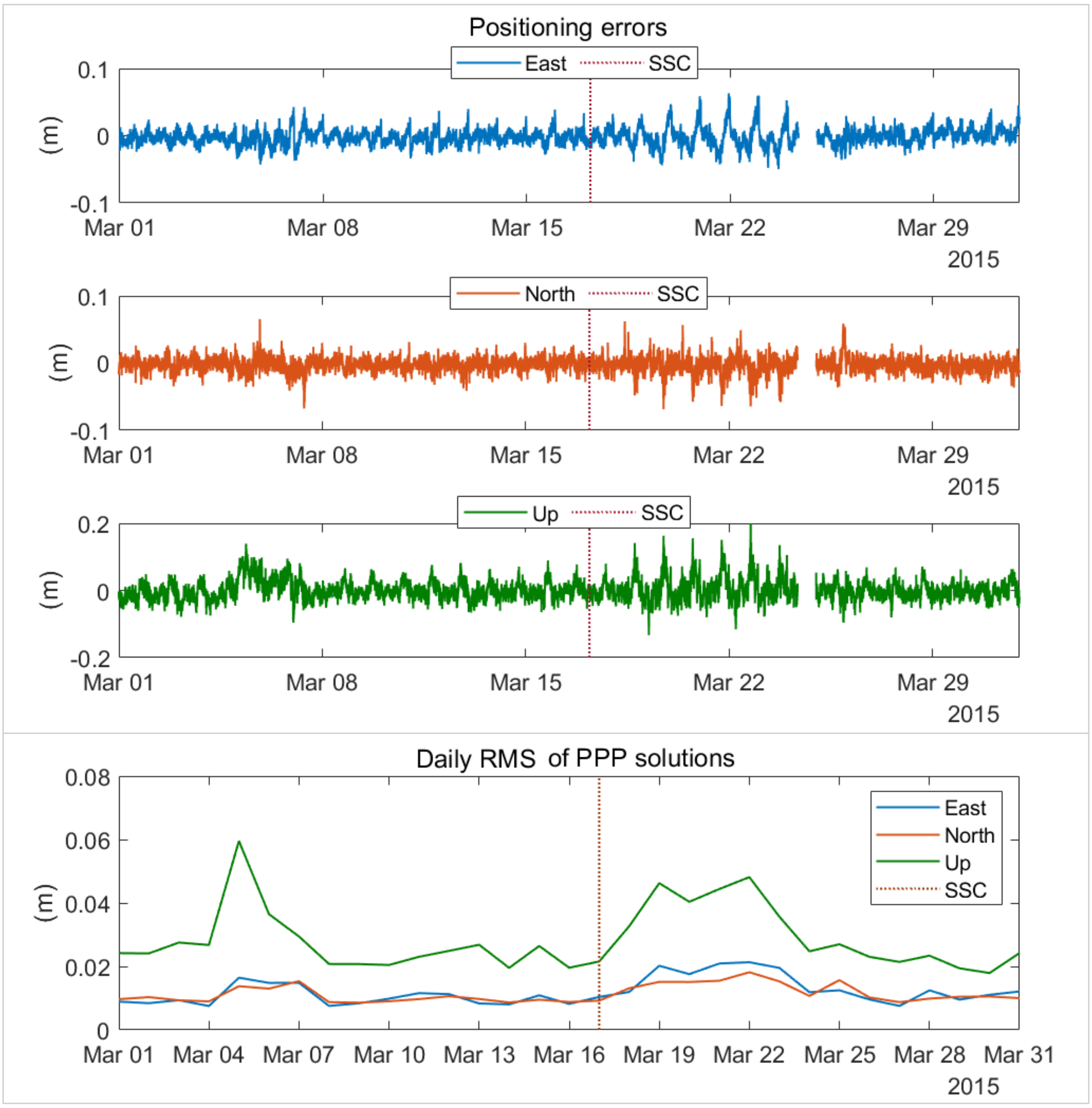


Figure 2.9: From top to bottom: positioning errors of the GNSS pseudo-kinematic processing results for the east, north, and up components, and daily RMS values of positioning errors. The results refer to the EPN SRJV station and March 2015. GNSS observations for March 24, 2015 from 08:50 to 23:59:30 UTC are missing.

These results demonstrate the necessity for an accurate and precise ionosphere model to correct ionospheric effects in positioning solutions. Therefore, the work in this dissertation aims to develop ionosphere models that incorporate space weather information from various satellites and observatories and apply ML techniques to model and forecast VTEC in order to meet the needs of GNSS users for effective ionosphere correction.

In the following Section 2.6, the main categories of existing ionosphere models are briefly discussed, followed by the main mathematical approaches that have been used to date for GNSS-based regional and global ionosphere modeling in the context of the dissertation.

2.6 Ionosphere Models

Existing ionosphere models can be categorized as physical, empirical, or mathematical (Schmidt et al., 2015; Farzaneh & Forootan, 2018) as shown in Table 2.3. Physical ionosphere models are based on physical and chemical processes in the ionosphere, requiring complicated numerical procedures with high computation cost, such as the Global Assimilation of Ionospheric Measurements (GAIM) model (Schunk et al., 2004) and the Global Ionosphere-Thermosphere Model (GITM) (Ridley et al., 2006). In contrast, empirical models describe the ionosphere using mathematical functions derived from historical observational data and statistics (Radicella & Nava, 2020). These models represent average conditions and regular ionosphere variations, i.e., its "climate". Examples of such climatological models include the International Reference Ionosphere (IRI) (Bilitza, 2018) and NeQuick (Nava et al., 2008). To correct the ionospheric delay in single-frequency observations, navigation satellite systems broadcast coefficients in the navigation message based on the empirical approaches. For instance, the well-known Klobuchar model (Klobuchar, 1987), which has been adopted in GPS, or a particular version of the NeQuick model, called NeQuick-G, implemented in Galileo (Orus Perez et al., 2018).

Table 2.3: Overview of different ionosphere models and their main characteristics.

Model	Category	Characteristics
GAIM **GITM**	Physical Physical	Represent physical laws, complicate numerical calculations
IRI **NeQuick**	Empirical Empirical	Represent average conditions: ionosphere "climate"
Klobuchar	Broadcasted Empirical	Simple, corrects ∼50% of the ionosphere error in GPS
NeQuick-G	Broadcasted Empirical	Simple, corrects ∼70% of the ionosphere error in Galileo
GIM	Mathematical	GNSS-VTEC, e.g., spherical harmonics, B-splines
RIM	Mathematical	GNSS-VTEC, e.g., Taylor series, B-splines
AI/ML-based	"Learning" from data	GNSS + space weather data, nonlinear modeling

The ionospheric delay can be more accurately modeled using GNSS observations, where VTEC is modeled using a variety of mathematical approaches and thus can be classified as mathematical models. GNSS-derived ionosphere maps are recognized as useful external sources that can provide information to single-frequency GNSS users to mitigate the first-order ionospheric delay and consequently reduce the ionospheric range error (Petit & Luzum, 2010), where we can distinguish between GIM and RIM.

GIMs are routinely generated by the Ionosphere Associated Analysis Centers (IAAC) of the IGS that include the Center for Orbit Determination in Europe, University of Bern, Switzerland (CODE), ESA, JPL, Universitat Politècnica de Catalunya; Technical University of Catalonia, Spain (UPC), Geodetic Survey of Natural Resources Canada (NRCan), Wuhan University, China (WHU) and Chinese Academy of Sciences (CAS). The GIM products of the IAAC are used to generate a combined weighted solution of the IGS with a spatial sampling of 2.5°× 5°in latitude and longitude, respectively, and a temporal resolution of two hours (Hernández-Pajares et al., 2009). It generally has a global relative error of 10% to 20% compared to the VTEC estimated from observations of the TOPEX satellite (Orús et al., 2002). The final GIM products typically have a latency of up to 3 weeks, while the rapid GIMs are generated with a latency of 1 to 2 days (Li et al., 2020; Liu et al., 2021), which may limit their use in real-time positioning applications.

The global VTEC distribution can be mathematically represented by spherical harmonics as

$$VTEC(\theta, S, t) = \sum_{n=0}^{n_{max}} \sum_{m=0}^{n} \tilde{P}_{nm}(\sin\theta)\,(a_{nm}(t)\cos(m \cdot S) + b_{nm}(t)\sin(m \cdot S))\,, \tag{2.6}$$

where

θ, S, t are the geographic/geomagnetic latitude, the Sun-fixed longitude, and time, respectively,

n_{max} is the maximum degree of the spherical harmonic expansion,

$\tilde{P}_{nm}$ are the normalized Legendre functions of degree n and order m,

a_{nm}, b_{nm} are the unknown coefficients of the spherical harmonics, i.e., the global ionosphere model parameters to be estimated.

CODE applies, for instance, a spherical harmonics expansion up to degree and order of 15 to produce GIMs.

Another approach to global VTEC modeling is based on localizing basis functions, for instance, B-spline functions, such as in the DGFI-TUM GIMs. They are modeled as a series expansion in trigonometric and polynomial B-spline functions. Since B-splines are localizing functions, this approach copes with data of heterogeneous density, i.e., unevenly distributed observations and data gaps (Schmidt et al., 2011). It can also be used to generate a multi-scale representation (Schmidt, 2012; Goss et al., 2019). The global VTEC model with B-spline functions reads

$$VTEC_{glob}(\phi, \lambda, t) = \sum_{k_1=0}^{K_{J_1}-1} \sum_{k_2=0}^{K_{J_2}-1} d_{k_1,k_2}^{J_1,J_2}(t) N_{k_1}^{J_1}(\phi) T_{k_2}^{J_2}(\lambda), \tag{2.7}$$

where

ϕ, λ, t denote the geomagnetic latitude and longitude, and time, respectively,

$N_{k_1}^{J_1}(\phi)$ are polynomial B-splines as a function of the geomagnetic latitude,

$T_{k_2}^{J_2}(\lambda)$ are trigonometric B-splines as a function of the geomagnetic longitude,

$d_{k_1,k_2}^{J_1,J_2}$ are initially unknown time-dependent series coefficients,

J_1, J_2 are the resolution levels.

The total numbers of B-spline functions are estimated as $K_{J_1} = 2^{J_1} + 2$ and $K_{J_2} = 3 \cdot 2^{J_2}$, where the numerical values for two levels J_1 and J_2 are determined by the average sampling interval of the data and spectral content of the VTEC signal (see Schmidt et al. (2015); Goss et al. (2019)). By simultaneously estimating coefficients of the B-spline representation and the biases of the carrier phases, and integrating them into an adaptive Kalman filter, (near) real-time VTEC maps can be generated (Erdogan et al., 2017, 2020, 2021)).

In contrast to GIMs, RIMs can be more accurate because they have higher spatial and, in some cases, higher temporal resolution as they incorporate observations from dense GNSS networks. A regional VTEC B-spline model can be set up as the sum of the global model (Equation 2.7) and the regional correction part as

$$VTEC_{reg}(\phi, \lambda, t) = VTEC_{glob}(\phi, \lambda, t) + \Delta VTEC_{reg}(\phi, \lambda, t) \tag{2.8}$$

with

$$\Delta VTEC_{reg}(\phi, \lambda, t) = \sum_{k_3=0}^{K_{J_3}-1} \sum_{k_4=0}^{K_{J_4}-1} d_{k_3,k_4}^{J_3,J_4}(t) N_{k_3}^{J_3}(\phi) N_{k_4}^{J_4}(\lambda) \tag{2.9}$$

where

J_3, J_4 denote the resolution levels,

k_3, k_4 are shift parameters applied to geographic latitude and longitude, respectively,

$N_{k_3}^{J_3}(\phi)$, $N_{k_4}^{J_4}(\lambda)$ are polynomial B-splines as a function of geographic latitude and longitude, respectively, defined within the study region.

Similar to the global VTEC definition in (2.7), the values for the regional levels J_3 and J_4 must be determined, see Goss et al. (2020).

Another mathematical approach for regional modeling is the Taylor series expansion. It is used to develop the regional VTEC model named RIM IONOsphere model for the Western Balkans (IONOWB) in **P-I**, which is defined as follows

$$VTEC(\theta_{IPP}, S_{IPP}, t) = \sum_{n=0}^{n_{max}} \sum_{m=0}^{m_{max}} c_{nm}(t)(\theta_{IPP} - \theta_0)^n (S_{IPP} - S_0)^m \tag{2.10}$$

where

θ_{IPP}, S_{IPP}, t are the geographic latitude, the Sun-fixed longitude at the IPP, and time, respectively[9],

θ_0, S_0 denote the geographic latitude and the Sun-fixed longitude of the origin of the Taylor series expansion,

n, m are the degree values of the Taylor series expansion in geographic latitude and Sun-fixed longitude, respectively,

n_{max}, m_{max} are the maximum degree values of the Taylor series expansion in geographic latitude and Sun-fixed longitude, respectively[10],

c_{nm} stands for the unknown coefficients of the Taylor series expansion, i.e., the regional ionosphere model parameters to be estimated (Dach et al., 2015; Natras et al., 2023a).

The choice of the maximum degree of Taylor series expansion for RIM development depends on the distribution of the available data and the local rates of VTEC change. In practice, it is usually performed at very low degree values (Magnet, 2019).

In recent years, state-of-the-art AI methods, in particular ML, have been used to identify and model nonlinear relationships between space weather and ionosphere quantities to improve modeling and forecasting of space weather processes, leading to a new generation of ionosphere models: ML-based data-driven models. The techniques and procedures for developing ML-based ionosphere models in this dissertation are discussed in Chapters 3 and 4.

2.7 Necessity of the Models

The ionosphere models in the form of GIMs, RIMs, or location/station-specific models can help in GNSS data processing in the following cases:

1. If only single-frequency data are available.
2. When single-frequency positioning is applied using the broadcasted ionosphere model, which can deviate from the actual ionosphere conditions.
3. When large ionosphere gradients occur, which can strongly affect differential positioning (see, e.g., Abdullah et al. (2007)), and ground-based augmentation systems (see, e.g., Caamano et al. (2021); Supriadi et al. (2022);

[9] For the calculation of θ_{IPP} and S_{IPP}, see Equations (6) and (7) in **P-I**.

[10] n_{max} is set to 1 and m_{max} to 2; see Table 2 in **P-I**.

4. To shorten the convergence time of carrier phase ambiguities, i.e., to enable fast integer ambiguity resolution in PPP (see, e.g., Psychas et al. (2018); Zhang et al. (2022c)) and improve ambiguity resolution for medium-length and long baselines in a network processing (see, e.g., Wanninger (1995); Odijk (2000); Zhang et al. (2022a)).

5. For a small-scale and high-precision network with a maximum extent of about 10 km, it is recommended to use only L_1 observations[11] in combination with an ionosphere model, even if dual-frequency data are available, in order to obtain precise solutions and eliminate or significantly reduce the ionosphere-induced scale bias[12] under homogeneous and moderate ionosphere conditions (see, e.g., Dach et al. (2015); Schaer et al. (1996)).

6. To account for HOI corrections by providing an external ionosphere model. This is also relevant when applying the ionosphere-free linear combination[13] (Dach et al., 2015).

7. During ionospheric scintillations, the f_1 frequency experiences less scintillation effect than the f_2 frequency, and tracking the L_2 signal becomes difficult. In this situation, combining the L_1 signal with an ionosphere model may provide better results than dual-frequency positioning (Kintner et al., 2007).

In addition to the need for GNSS applications, ionosphere models can contribute to the understanding of the processes in Earth's upper atmosphere and are a critical component in the study of space weather processes and their effects on Earth since the ionosphere is a final link in the space weather chain (Figure 1.2).

Chapter 3 details the learning algorithms and approaches developed, modified, and applied in this dissertation for the ML-based regional and forecasting VTEC models, including the uncertainty estimation. It concludes by presenting how the relative feature importance is calculated to provide an insight into what the model has learned based on the important input variables.

[11] For very short baselines, 1-10 km, L_1/L_2-based solutions perform significantly better than L_3 linear combination, although using only L_1 and ignoring L_2 often leads to even better results (Dach et al., 2015).

[12] Using only single-frequency observations without an ionosphere model leads to ionosphere-induced scale bias resulting in apparent contraction of the network, i.e., baseline shrinkage.

[13] After applying the L_3 combinations, the HOI terms remain, which depend mainly on the degree of solar activity, geomagnetic and ionosphere conditions, and can reach tens of centimeters during extreme space weather events (Hoque & Jakowski, 2008).

3 Machine Learning for VTEC

3.1 Learning from VTEC-related Data

In this dissertation, supervised learning is used for ionosphere modeling and forecasting, where the model "learns" from historical data, i.e., from given examples of input and output data that can be understood as past experiences. The term learning here refers to finding an approximation function that maps input variables to the output of VTEC (Figure 3.1). Thus, supervised learning can be viewed as a function estimation problem (Friedman, 2001). The learning is performed by optimizing the performance of a learning algorithm for the task of VTEC modeling and forecasting.

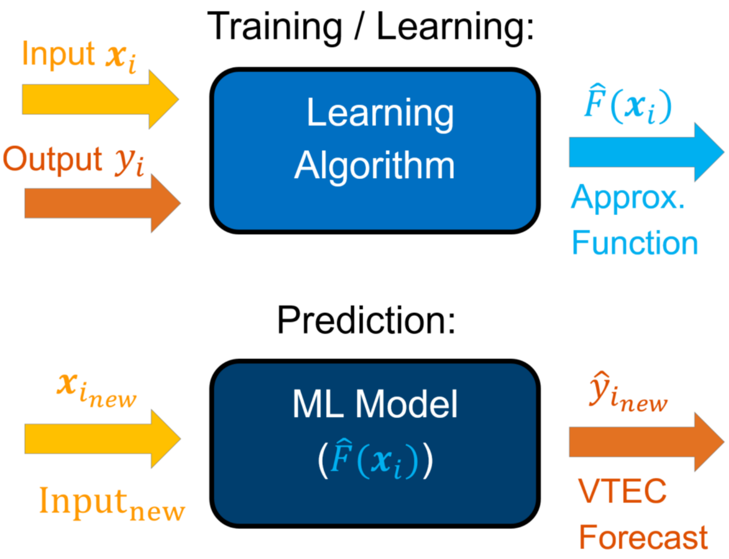


Figure 3.1: Supervised learning scheme for VTEC forecast.

A well-posed ML problem can be defined as by Mitchell (1997): "A computer program is said to learn from experience E with respect to task T and a performance measure P, if its performance on T, as measured by P improves with experience E". In the context of this work, the ML problem is identified with the task of VTEC modeling and forecasting, where the experience is provided in the form of training data, and an objective cost function defines a performance measure.

Let us define a training sample for the VTEC forecasting problem, which includes p input features in a vector $\mathbf{x}_i$ and an output $y_i = F(\mathbf{x}_i)$ for each of the N observations at time stamp i with $i = 1, 2, \ldots, N$ as in Equation (3.1). The vectors $\mathbf{x}_i$ can be interpreted as the rows of the $N \times P$ predictor matrix $\mathbf{X} = (\mathbf{x}_i^T)$, whereas the columns represent the input features $\widetilde{\mathbf{x}}_p$ with $p = \{0, 1, 2, \ldots, P-1\}$. The components $x_{i,p}$

of the $N \times 1$ column vector $\widetilde{\mathbf{x}}_p = [x_{1,p}, x_{2,p}, \ldots, x_{N,p}]^T$ represent a time series of the p^{th} input features.

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \cdots \\ \mathbf{x}_N^T \end{bmatrix} = \begin{bmatrix} x_{0,1}, x_{1,1}, x_{2,1}, \ldots, x_{P-1,1} \\ x_{0,2}, x_{1,2}, x_{2,2}, \ldots, x_{P-1,2} \\ \cdots \\ x_{0,N}, x_{1,N}, x_{2,N}, \ldots, x_{P-1,N} \end{bmatrix} = \begin{bmatrix} DOY_1, HOD_1, R_1, \ldots, VTEC_1'' \\ DOY_2, HOD_2, R_2, \ldots, VTEC_2'' \\ \cdots \\ DOY_N, HOD_N, R_N, \ldots, VTEC_N'' \end{bmatrix}, \tag{3.1}$$
$$= \left[\widetilde{\mathbf{x}}_0, \widetilde{\mathbf{x}}_1, \widetilde{\mathbf{x}}_2, \ldots, \widetilde{\mathbf{x}}_{P-1}\right],$$
$$\mathbf{y} = \mathbf{VTEC}(i+t) = \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_N \end{bmatrix} = \begin{bmatrix} VTEC_{1+t} \\ VTEC_{2+t} \\ \cdots \\ VTEC_{N+t} \end{bmatrix},$$

where t stands for values of a forecast horizon, with $t = 1$ for the 1-hour forecast in **P-II**, $t = 24$ for the 24-hour forecast in **P-II, P-III, CP-II**, and t also includes $\{1/4, 3, 6\}$ for the 15-minute, 3-hour and 6-hour forecasts, respectively, in Section 5.4. For abbreviations in (3.1), see Table 3.1. Note that y_i refers to the VTEC for a single forecast horizon $i + t$. However, y_i can be modified to consider multiple forecast horizons simultaneously, which was not done in this work. Exponential Moving Average (EMA) in Table 3.1 gives more weight and, therefore, more importance to the most recent data points, so that the most recent data have more influence than the oldest, and can be formulated as in Equation (4.1). d stands for days and takes the values $d = \{1, 4, 7\}$.

The learning task for VTEC forecasting can be formulated as follows: given the values of an input vector $\mathbf{x}_i$ the goal is to find an approximation $\hat{F}(\mathbf{x}_i)$ of the function $F(\mathbf{x}_i)$ that maps the input $\mathbf{x}_i$ to the output y_i, and estimates $\hat{y}_i$ as

$$\begin{aligned} & y_i + e_i = \widehat{y}_i = \widehat{VTEC}_{(i+t)} = \widehat{F}(\mathbf{x}_i) \\ & \mathbf{x}_i = [DoY_i, HOD_i, R_i, F10.7_i, SW_i, Bz_i, Dst_i, Kp_i, AE_i, \\ & \qquad VTEC_i, VTEC_{EMA(30)}i, VTEC_{EMA(4)}i, VTEC_i', VTEC_i'']^T, \end{aligned} \tag{3.2}$$

where e_i is an error. $\widehat{F}(\cdot)$ refers to the approximation function of the nonlinear relationship between the VTEC output and the input vector consisting of solar, interplanetary and geomagnetic indices, as well as the previous VTEC values, defined in **P-II**. This function is unknown and is therefore approximated by optimizing learning algorithms for the task of VTEC forecasting.

A general formulation of the approximation function can be expressed using a series expansion as

$$\hat{y}_i = \hat{F}(\mathbf{x}_i; \{\beta_m, \boldsymbol{\gamma}_m\}_1^M) = \sum_{m=1}^{M} \beta_m h(\mathbf{x}_i; \boldsymbol{\gamma}_m), \tag{3.3}$$

where β_m with $m = \{1, 2, ..., M\}$ are the expansion coefficients and $h(\mathbf{x}_i; \boldsymbol{\gamma}_m)$ are a suitable set of basis functions of the input vector $\mathbf{x}_i$ parametrized by a set of coefficients $\boldsymbol{\gamma}$. They allow a more flexible representation for $F(\mathbf{x})$. Traditional examples are polynomial and trigonometric expansions. An example of nonlinear expansions is the sigmoid function, often used in ANN. For example, for the single hidden layer neural network, it is $h(\mathbf{x}_i; \boldsymbol{\gamma}) = \sigma(\gamma_0 + \gamma_1^T \mathbf{x}_i)$, where $\sigma(.)$ is the sigmoid function formulated in (3.25) and $\boldsymbol{\gamma}$ parametrizes the linear combination of the input variables. In a decision tree algorithm, $h(\mathbf{x}_i; \boldsymbol{\gamma}_m)$ represents a small tree, where $\boldsymbol{\gamma}_m$ parameterizes the splitting variables, the splitting point and the results at the terminal nodes. The function $h(\mathbf{x}_i; \boldsymbol{\gamma}_m)$ is also called the "base learner" (Friedman, 2001), and the Equation (3.3) is the basis of many ML methods.

Using prepared training samples of input features and output as defined in (3.1), an approximation $\hat{F}(\mathbf{x}_i)$ is obtained by minimizing the objective cost function

$$Cost = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{F}(\mathbf{x}_i)), \tag{3.4}$$

Table 3.1: Overview of input and output data for ML-based VTEC models.

Paper / Diss. Section	**Input**	**Output**
P-I	(i)	(i)
1-hour data sampling	Regional ionosphere coefficients Latitude, Longitude Hour of Day (HoD) Solar radio flux F10.7 index GMF Dst index GMF Kp·10 index	VTEC
P-II	(i)	$(i$ + 1 h, i + 24 h)
1-hour data sampling	Day of Year (DoY) HoD Sunspot number (S$_n$/R) F10.7 index Solar Wind Speed (SWS) Interplanetary Magnetic Field (IMF) Bz index GMF Dst index GMF Kp·10 index AE index VTEC VTEC EMA over previous 30 days $\left(VTEC_{EMA(30)}\right)$ VTEC EMA over previous 4 days $\left(VTEC_{EMA(4)}\right)$ First VTEC derivative $(VTEC')$ Second VTEC derivative $(VTEC'')$	VTEC
P-III	(i)	$(i$ + 24 h)
1-hour data sampling	as in **P-II**	VTEC σ / Confidence interval
Sec. 5.4	$\{i$ - $d \cdot 24$ h$, ..., i - 30$ min$, i - 15$ min$, i\}$	(i + 15 min, i + 1 h i + 3 h, i + 6 h, i + 24 h)
15-minute data sampling	GMF SYM/H index VTEC	VTEC 95% confidence interval
	(i)	(i + 15 min, i + 1 h i + 3 h, i + 6 h, i + 24 h)
	GMF SYM/H index VTEC VTEC$_{EMA(30)}$ VTEC$_{EMA(4)}$ VTEC$'$ VTEC$''$ PROBA2 LYRA solar data Geostationary Operational Environmental Satellite (GOES) Lyman-alpha (Ly-$\boldsymbol{\alpha}$) solar data	VTEC 95% confidence interval

which represents the loss function $\mathcal{L}$ averaged over the training data. The loss function used in **P-I, P-II**, and partly in **P-III** is the squared error ${e_i}^2$ [(1)]

$$\mathcal{L}(y_i, \hat{F}(\mathbf{x}_i)) = e_i^2 = (y_i - \hat{y}_i)^2. \tag{3.5}$$

Substituting (3.5) into (3.4), the resulting objective cost is the Mean Squared Error (MSE). In this way, the function that describes the input-output relationship is modified as a response to differences between the original y_i and generated output $\hat{y}_i$. This represents learning by examples commonly referred to as *learning* or *training*. The ability of a model to make predictions for new data not seen during training is called *generalization*. By learning from ionosphere and space weather data, we aim to build a model that approximates VTEC close enough to real values, can generalize, and is applicable to data likely to be encountered in practice.

3.2 VTEC-Adapted Learning Algorithms

Learning approaches for VTEC modeling and forecasting developed in this dissertation and discussed in the publications are summarized in Table 3.2. The learning algorithms are ordered by complexity, i.e., from a simple decision tree (item 1), moderately complex ensemble learning algorithms (items 2 - 5), and more complex ensemble modeling (items 6 - 7), to highly complex neural networks (items 8 - 10). They are explained in a similar order in the dissertation by first discussing ML solutions that provide a single VTEC output for each input observation (Sections 3.2.1 to 3.2.5), and then ML solutions for probabilistic VTEC output (Sections 3.3).

Decision tree-based algorithms are conceptually simple but are powerful learning techniques that can be easily adapted to solve both linear and nonlinear modeling problems in either classification or regression tasks. In this thesis, several tree-based learning algorithms have been applied, namely the Regression Decision Tree and ensemble learning such as Random Forest, Gradient Boosting (GBoost), and Adaptive Boosting (AdaBoost). Decision trees can handle mixed discrete and continuous input variables, perform automatic selection of input variables, are relatively robust to outliers, and can work with both small and large data sets. A single tree is simple and easy to interpret but has lower accuracy. However, by combining a large number of trees into an ensemble, the accuracy can be improved significantly, but this comes with a loss of some interpretation ability (Hastie et al., 2009). Thus, ensemble learning aims to combine the predictions of multiple simple models or base learners, such as a decision tree, to improve generalization and robustness.

ANN is a state-of-the-art technique with widespread applications in many fields today. Inspired by networks of biological neurons, the central idea of ANNs is to extract linear combinations of the input variables as derived features and then model the target output as a nonlinear function of these features (Hastie et al., 2009). ANNs can handle extensive and high-dimensional data and solve highly complex tasks but are generally overparameterized.

3.2.1 Decision Tree

Decision trees can be classified according to the type of output variables as classification trees, i.e., trees with categorical output, and regression trees, i.e., trees with numerical output. Within this study, the regression tree was grown on the training data using recursive binary splitting. A small regression tree with a depth of 3 is shown in Figure 3.2 for ease of illustration. There, the tree for VTEC nowcasting was grown using time information, solar and geomagnetic indices as input, and VTEC as output.

[(1)] Other loss functions used in this work are the quantile loss (Section 3.3.3) and the negative-log likelihood loss (Section 3.3.4), which are used to quantify uncertainties and provide probabilistic VTEC forecasts.

Table 3.2: ML approaches for VTEC modeling and forecasting developed in this dissertation, with their abbreviations, use cases, the output of the model based on each ML approach, the publication in which the results were presented, and a section of the dissertation explaining these methods. CI stands for confidence intervals, μ and σ stand for mean and standard deviation, respectively.

Item	**Learning algorithm**	**Abbreviation**	**Use case**	**Model output**	**Paper**	**Section**
1	Decision Tree	DT	Forecasting	VTEC	**P-II**	3.2.1
2	Random Forest	RF	Forecasting	VTEC	**P-II**	3.2.2
3	eXtreme Gradient Boosting	XGBoost/GBoost/GB	Forecasting	VTEC	**P-II**	3.2.3
4	Quantile Gradient Boosting	QGB	Probabilistic forecasting	VTEC quantiles	**P-III**	3.3.3
5	Adaptive Boosting	AdaBoost/AB	Forecasting	VTEC	**P-II**	3.2.3
6	Voting Regressor	VR	Forecasting	VTEC	**P-II**	3.2.4
7	Super-Ensemble	SE	Forecasting	VTEC	**P-III**, **CP-II**	3.2.4, 3.3.2
8	Artificial Neural Network	ANN	Spatio-temporal modeling	VTEC	**P-I**	3.2.5
9	Bayesian Neural Network	BNN	Probabilistic forecasting	VTEC	**P-III**	3.3.4
10	BNN including data uncertainty	BNN-D	Probabilistic forecasting	μ, σ of VTEC	**P-III**	3.3.4.1

Each regression tree model can be formally expressed as

$$T(\mathbf{X};\Theta) = \sum_{j=1}^{J} \gamma_j(\mathbf{X} \in R_j) \tag{3.6}$$

with a set of parameters $\Theta = \{\gamma_j, R_j\}_{j=1}^{J}$. $\{R_j\}_1^J$ are disjoint regions that collectively cover the space of all joint values of the input variables $\mathbf{X}$. $j = 1, 2, ..., J$ stands for the number of distinct and non-overlapping regions $\{R_1, R_2, \ldots, R_J\}$. The regions represent the nodes in Figure 3.2. The parameters of a single tree are the coefficients $\{\gamma_j\}_1^J$ and the quantities that define the boundaries of the regions $\{R_j\}_1^J$: the split variables $\widehat{\mathbf{x}}_t$ with $t \in \{0, 1, 2, ..., J-1\}$, and the values of the split variables, called split points, s, that split the nodes of the tree. Since the regions are disjoint, Equation (3.6) is equal to

$$T(\mathbf{X}) = \gamma_j. \tag{3.7}$$

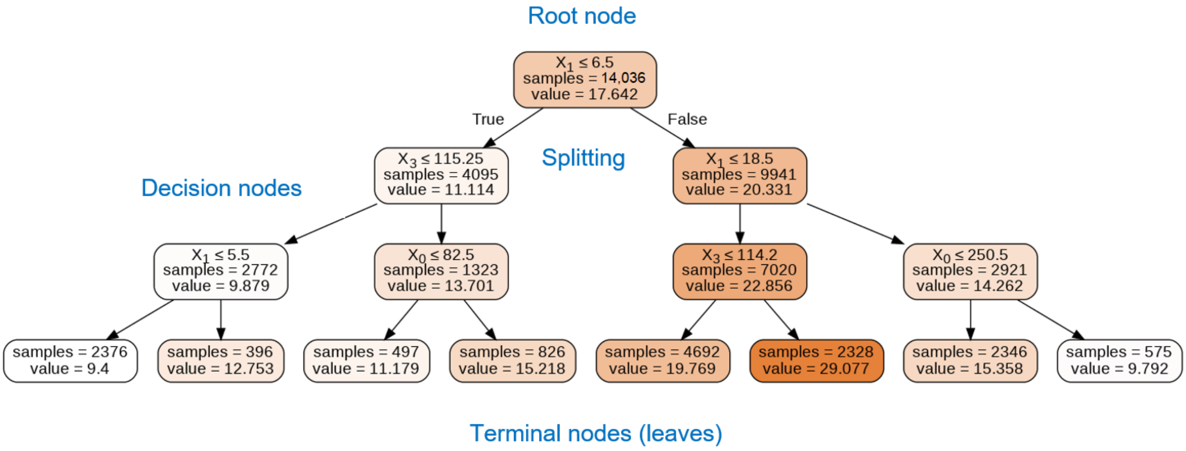


Figure 3.2: A small decision tree with a maximum depth of three for VTEC nowcast at 10°E 10°N. The input features $\widetilde{\mathbf{x}}_p$ are denoted by indices 0, 1, and 3 corresponding to DoY, HoD and F10.7, respectively. Since the tree is small, only input features with the largest impact on VTEC are considered. A larger tree would increase the number of input features considered and the complexity of the interactions modeled. The color shading of the nodes corresponds to the magnitude of the VTEC output. Nodes with a larger VTEC are colored darker, while nodes with a smaller VTEC are colored brighter. (Taken from **P-II**)

The approach begins at the top of the tree, called the root node, as presented in Figure 3.2. At this point, all observations belong to a single region R. The root node contains 14,036 observation samples. The mean VTEC value γ of all observations within the region R is 17.642 TECU. The decision splitting in the root node is given as $\widetilde{\mathbf{x}}_1 \leq 6.5$, which represents the split point s, while input variable $\widetilde{\mathbf{x}}_1$, representing HoD, is the split variable of the region R. The input space is then divided into two distinct and non-overlapping regions R_1, where the condition is True, i.e., $\widetilde{\mathbf{x}}_1 \leq 6.5$, and R_2, where the condition is False, i.e., $\widetilde{\mathbf{x}}_1 > 6.5$. Therefore, considering a split variable $\widehat{\mathbf{x}}_t = \widetilde{\mathbf{x}}_p$ with $p \in \{1, 2, \ldots, P-1\}$ and split point s, two regions can be defined, based on a decision splitting, as

$$R_1(p,s) = \{\mathbf{X} \mid \widetilde{\mathbf{x}}_p \leq s\},\ R_2(p,s) = \{\mathbf{X} \mid \widetilde{\mathbf{x}}_p > s\} \tag{3.8}$$

(Hastie et al., 2009). The split variable $\widetilde{\mathbf{x}}_p$ and split point s are found in a way to solve

$$\min_{p,s}[\min_{\gamma_1} \sum_{\mathbf{x}_{p,i} \in R_1(p,s)} (y_i - \gamma_i)^2 + \min_{\gamma_2} \sum_{\mathbf{x}_{p,i} \in R_2(p,s)} (y_i - \gamma_i)^2], \tag{3.9}$$

and the inner minimization is solved by

$$\gamma_1 = \frac{1}{N} \sum_{\mathbf{x}_i \in R_1(p,s)} y_i, \ \gamma_2 = \frac{1}{N} \sum_{\mathbf{x}_i \in R_2(p,s)} y_i. \tag{3.10}$$

The procedure continues further down the tree so that the input space, which covers all the joint values of the predictor variables, is divided into J distinct and non-overlapping regions $\{R_1,R_2,\ldots,R_J\}$. This means that the space of the input variables is successively split, i.e., a node is partitioned into two subnodes or regions further down the tree. A sub-node that is subdivided into further sub-nodes is called a decision node. The values in each rectangle in Figure 3.2 represent the mean VTEC output γ_j of the y_i falling in the region R_j as in Equation (3.10). A tree stops growing when a node has fewer than the minimum number of observations required for the split or when it reaches the maximum depth of the tree, if specified. This node represents the terminal node or leaf. As can be seen, a decision tree is a simple and interpretable method, easily visualized by a two-dimensional graph, and is an example of a white-box model.

3.2.2 Random Forest

Random Forest (Breiman, 2001) represents a modification of the so-called bagging or bootstrap aggregation technique, where a large collection of decorrelated trees is created and averaged (Figure 3.3). When building each tree, a random sample of v input variables is considered as split candidates from a complete set of P input features. As the VTEC time series forecasting is performed, each new training set is drawn from the original training set without replacement, as opposed to the standard Random Forest procedure with replacement. Thus, a single regression tree T_b for $b = \{1, 2, \ldots, B\}$ for VTEC forecasting is grown by recursively repeating the following steps for each tree node until the minimum node size is reached:

1. Select a random sample of v input variables from the full set of P input variables;
2. Find the best splitting variable and split point among the v input variables;
3. Split the node into two subnodes.

The procedure is applied to all B trees. The function can be expressed as an average of all B trees

$$\widehat{VTEC}_{(i+t)} = \widehat{F}(\mathbf{x}_i) = \frac{1}{B} \sum_{b=1}^{B} T(\mathbf{x}_i; \Theta_b), \tag{3.11}$$

where Θ_b characterizes the b^{th} tree in terms of split variables, split points at each splitting node, and terminal node values of VTEC. Breiman (2001) demonstrated that randomness and diversity in tree construction lead to a lower generalization error and a model with reduced variance.

3.2.3 Adaptive and Gradient Boosting

The motivation for developing boosting was to combine the outputs of various base learners such as an J-node tree to produce a powerful "committee" (Breiman et al., 1984). In the boosting method, the trees are grown sequentially using the information from previously grown trees with modified versions of the training data (Figure 3.3).

Each boosted tree can be expressed as a sum of M trees

$$F_M(\mathbf{x}_i) = \sum_{m=1}^{M} T(\mathbf{x}_i; \Theta_m), \tag{3.12}$$

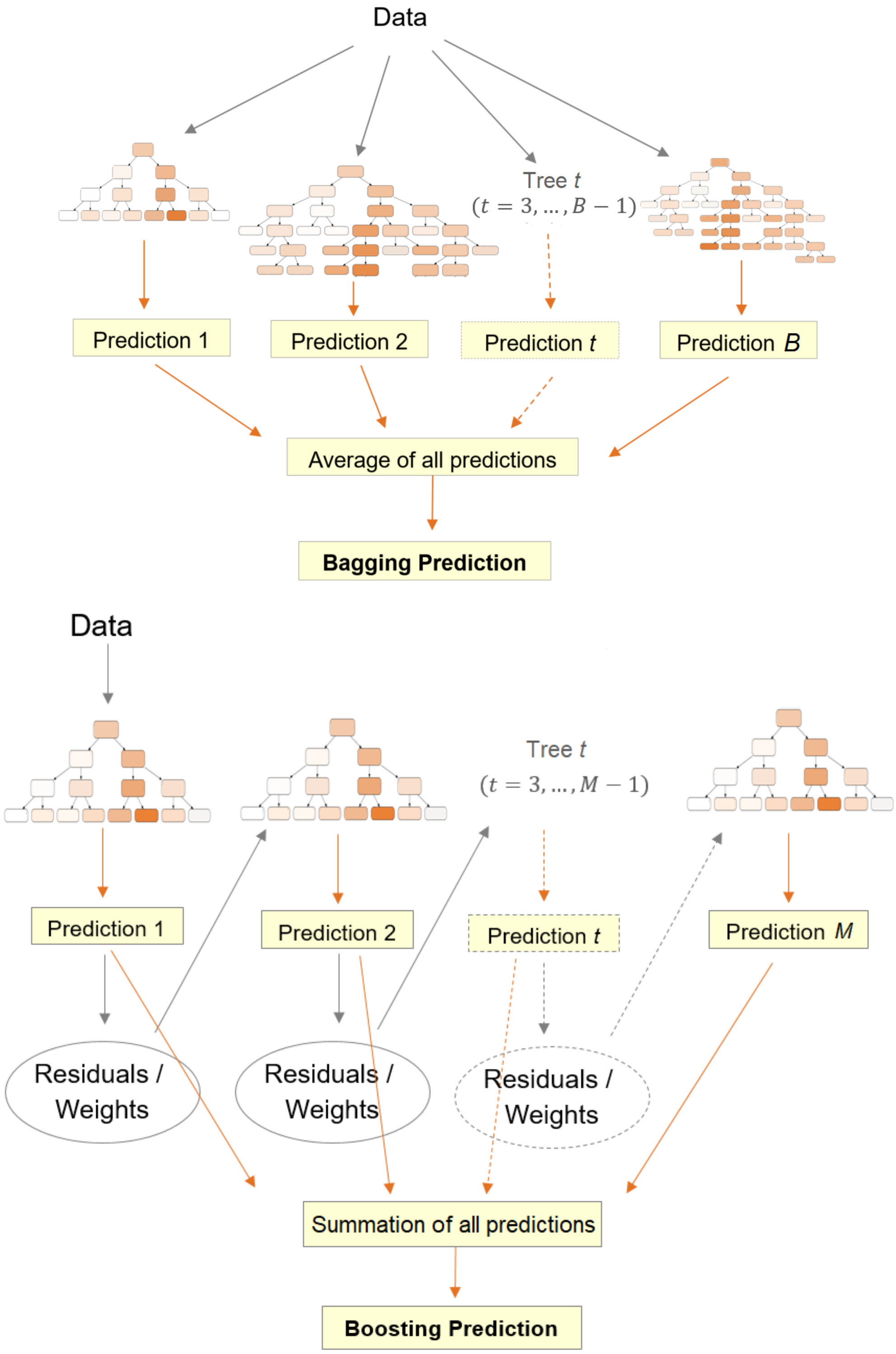


Figure 3.3: Diagrams of bagging (Random Forest) and boosting (AdaBoost and GBoost) methods. (Source: **P-II**)

where $\Theta_m = \{\gamma_{j_m}, R_{j_m}\}_{j_m=1}^{J_m}$ according to Equation (3.6). The final model can be expressed as a sum of such trees

$$\widehat{VTEC}_{(i+t)} = \widehat{F}(\mathbf{x}_i) = F_M(\mathbf{x}_i) = F_{M-1}(\mathbf{x}_i) + \gamma_{j_M}(\mathbf{x}_i \in R_{j_M}), \tag{3.13}$$

where $F_{M-1}(\mathbf{x}_i)$ represents the model consisting of $M-1$ trees, while the right-hand side of Equation (3.13) represents the current M^{th} tree.

At each step, the following should be solved

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} \mathcal{L}(y_i, F_{m-1}(\mathbf{x}_i) + T(\mathbf{x}_i; \Theta_m)), \tag{3.14}$$

where $\mathcal{L}$ is the loss function. Given the region R_{j_m}, the optimal constants γ_{j_m} in each region can be found as

$$\hat{\gamma}_{j_m} = \arg\min_{\gamma_{j_m}} \sum_{\mathbf{x}_i \in R_{j_m}} \mathcal{L}(y_i, F_{m-1}(\mathbf{x}_i) + \gamma_{j_m}). \tag{3.15}$$

In Adaptive Boosting (AdaBoost) (Freund & Schapire, 1997), the data are modified by applying weights $\{w_1, w_2, \ldots, w_N\}$ to each of the training example/pairs $(\mathbf{x}_i, y_i)$. In the first step, $m = 1$, all weights are initialized to $w_i = \frac{1}{N}$, i.e., the data are equally weighted. For each successive step, $m \in \{2, 3, \ldots, M\}$, the weights are modified individually, and the training is repeated using the weighted observations. More specifically, at steps $m \in \{2, 3, \ldots, M\}$, the weights increase for the wrongly predicted observations in the previous step, while the weights for correctly predicted observations decrease. Therefore, observations that are difficult to predict receive an increasing attention during training as the iterations proceed. In the end, the weighted predictions from all trees, i.e., steps, are combined to produce the final model output, the VTEC forecast, as in Equation (3.13).

Gradient boosting (GBoost) offers a generalization of boosting to an arbitrary differentiable objective function in Equation (3.5). A tree is trained on the original training data in the first step. Then the negative gradient is computed as

$$-g_{i_m} = -\left[\frac{\partial \mathcal{L}}{\partial F(\mathbf{x}_i)}\right]_{F=F_{m-1}}. \tag{3.16}$$

For the squared error loss, the negative gradient represents the residual between the original and the estimated output $-g_{i_m} = y_i - F_{m-1}(\mathbf{x}_i)$. For each successive iteration $m \in \{2, \ldots, M\}$, a regression tree is fitted to the residuals g_{i_m} from the previous iteration within terminal regions R_{j_m}. Afterwards, the function is updated as in Equation (3.13). XGBoost[(2)] (Chen & Guestrin, 2016) is an optimized GBoost algorithm that applies the shrinkage technique as a regularization strategy to avoid overfitting a model. This is implemented by scaling the contribution of each tree by a factor $0 \leq \nu < 1$ in (3.12) as

$$F_m(\mathbf{x}_i) = F_{m-1}(\mathbf{x}_i) + \nu \cdot \sum_{j_m=1}^{J_m} \gamma_{jm}(\mathbf{x}_i \in R_{j_m}), \tag{3.17}$$

where the parameter ν represents the learning rate of the boosting procedure.

Boosting approaches have proven to work also on small data sets with structured input data (Duan et al., 2020) and to be a powerful winning method in many data science competitions (Chen & Guestrin, 2016). The main reason is that it can be seen as a form of regularization that helps prevent overfitting by eliminating irrelevant features.

[(2)] In **P-II** it is referred to as XGB, while in **P-III** it is referred to as GB and GBoost. All GBoost VTEC models developed in this thesis use the shrinkage technique.

3.2.4 Ensemble Meta-Estimator

The idea behind ensemble modeling in this work is to combine multiple VTEC models that use either different learning algorithms or different training data sets, or both. An ensemble meta-estimator then aggregates the results of each ensemble member and combines them into a final output. This method is helpful for a set of various models to compensate for their individual weaknesses and create a single model with improved generalization ability.

Voting Regressor (VR) in **P-II** stands for the VTEC ensemble meta-estimator that incorporates multiple ML-based VTEC models with different ensemble learning, namely Random Forest, AdaBoost, and GBoost form the VR1 model as shown in Figure 3.4, while Random Forest and GBoost form the VR2 model (see Table 3 in **P-II**). Subsequently, the individual VTEC results across all models are averaged to produce a final VTEC forecast. In **P-III**, the ensemble meta-estimator is referred to as a Super-Ensemble (SE), which combines various ML-based VTEC models that use both different ensemble learning algorithms and different data sets. Therefore, the main difference between VR and SE approaches developed in this dissertation is that VR combines VTEC models trained on the same data set, while SE combines VTEC models trained on altered data set and is further used to quantify the uncertainty propagated through an ensemble. For a detailed description of ensemble modeling to quantify uncertainty, see Section 3.3.2.

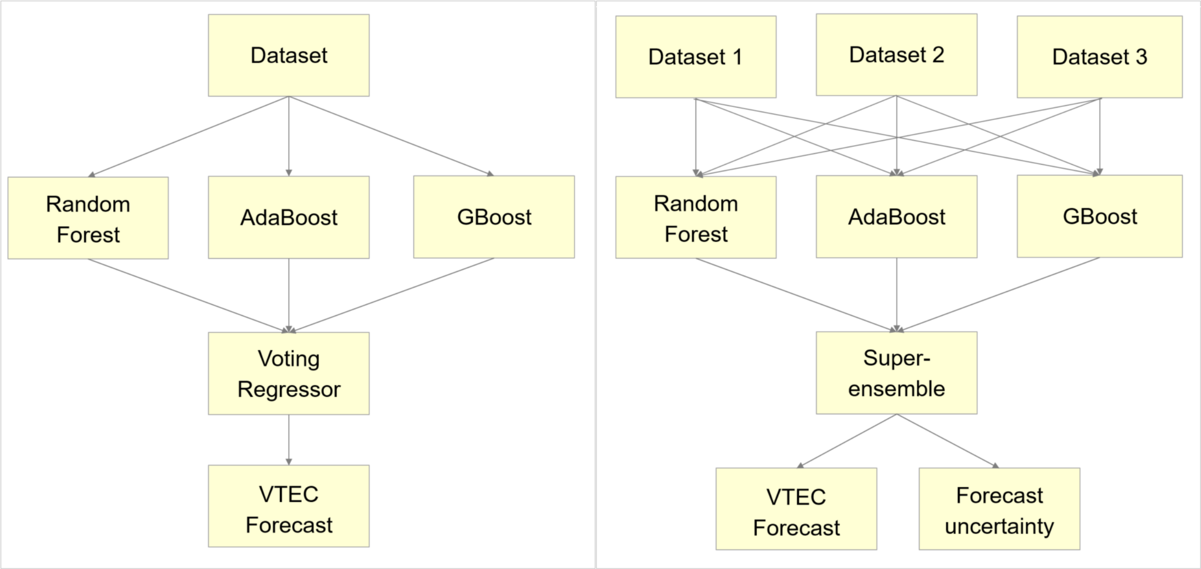


Figure 3.4: Overview of the development of two ensemble meta-estimators: VR1 (left) and SE (right).

3.2.5 Artificial Neural Network

The main principle of the ANN learning method is to extract linear combinations of the input variables as derived features and then model the target output as a nonlinear function of these features. The ANN usually consists of an input layer with input variables, an output layer with an output value, and one or more layers in between. These layers in the mid of the network are called hidden layers because they are not directly observed. They contain hidden neurons that model nonlinear combinations of the original input variables by computing the derived hidden features. These derived features can be viewed as a series expansion of the original input **X**, with the parameters of the basis functions learned from the data. The parameter or weight matrices can be defined as follows

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}^{(1)} \\ \mathbf{W}^{(2)} \\ \cdots \\ \mathbf{W}^{(K-1)} \end{bmatrix} \quad \text{and} \quad \mathbf{W}^{(k-1)} = \begin{bmatrix} \mathbf{w}_1^{(k-1)^T} \\ \mathbf{w}_2^{(k-1)^T} \\ \cdots \\ \mathbf{w}_{SL_k}^{(k-1)^T} \end{bmatrix} = \begin{bmatrix} w_{1,0}^{(k-1)}, w_{1,1}^{(k-1)}, \ldots, w_{1,PL_{k-1}}^{(k-1)} \\ w_{2,0}^{(k-1)}, w_{2,1}^{(k-1)}, \ldots, w_{2,PL_{k-1}}^{(k-1)} \\ \cdots \\ w_{SL_k,0}^{(k-1)}, w_{SL_k,1}^{(k-1)}, \ldots, w_{SL_k,PL_{k-1}}^{(k-1)} \end{bmatrix}, \tag{3.18}$$

where the matrix $\mathbf{W}$ contains all weights in the network. $\mathbf{w}_0$ is the bias. $\mathbf{W}^{(k-1)}$ is the weight matrix assigned to neurons from the previous layer $(k-1)$ containing PL_{k-1} neurons to estimate the values of SL_k neurons in the next layer (k) with $k \in \{2, 3, \ldots, K\}$, $sl_k \in \{1, 2, \ldots, SL_k\}$ and $pl_{k-1} \in \{0, 1, \ldots, PL_{k-1}\}$.

The linear model that consists of an input layer and an output layer given a vector of input $\mathbf{x}_i$ and weights $\mathbf{w}^{(1)}_{sl_k, pl_{k-1}}$ with $SL_k = 1$ for the single output neuron can be expressed as

$$y_i + e_i = \widehat{y}_i = \widehat{F}(\mathbf{x}_i) = \sum_{pl_{k-1}=0}^{PL_{k-1}} w^{(1)}_{1,pl_{k-1}} \cdot x_{pl_{k-1},i} = q(\mathbf{w}_1^{(1)^T} \cdot \mathbf{x}_i), \tag{3.19}$$

where q is the activation function, which in this case is linear. In the input layer, the neurons pl_{k-1} are equal to the input features p in Equation (3.1). In the matrix $\mathbf{X}$ in (3.1), an element $x_{0,i} = 1$ is added in each row. This is convenient to include because of the bias term $w_{sl_k,0}$ to obtain the above equation in vector form as a scalar product. The weights are determined in the training phase using the least squares method to minimize the sum of the squares of the errors e_i.

In a fully feed-forward connected network, all neurons in one layer are connected to all neurons in the next layer, also known as a MultiLayer Perceptron (MLP) network. A feed-forward network means that neurons are connected in one direction from the input layer to the output layer so that information is propagated forward from one layer to the next, as shown in Figure 3.5. MLPs are universal approximators, i.e., they can approximate any continuous function if there are sufficient many hidden neurons and layers to any desired level of accuracy (Hornik, 1991).

The number of neurons in the first layer equals the length of the vector $\mathbf{x}_p$. The neurons in each layer can be denoted as activation neuron vector $\mathbf{a}_i^{(k)}$ with $k = \{1, \ldots, K\}$. The input layer can be expressed as $\mathbf{a}_i^{(1)} = \mathbf{x}_i$. Hidden layers propagate the derived features from $\mathbf{a}_i^{(2)}$ to $\mathbf{a}_i^{(K-1)}$ as shown in Figure 3.5. They create a nonlinear mapping between the input variables and the output of the network using a nonlinear activation function as depicted in Figure 3.6. The last layer provides the output $\hat{y}_i = \mathbf{a}_i^{(K)}$. The activation neurons in the hidden layer can be calculated as

$$\mathbf{a}_i^{(k)} = q(\mathbf{W}^{(k-1)^T} \cdot \mathbf{a}_i^{(k-1)}), \tag{3.20}$$

The network output $\hat{y}_i$ is calculated as

$$\hat{y}_i = \mathbf{a}_i^{(K)} = q(\mathbf{W}^{(K-1)^T} \cdot \mathbf{a}_i^{(K-1)}). \tag{3.21}$$

In the hidden layer, a nonlinear activation function is used. In the final layer, a linear activation function is common for quantitative output. Afterward, backpropagation is applied to backpropagate the error from the output layer to the input layer and to adjust the weights of the ANN using an optimization algorithm to minimize the loss in (3.5).

Applying Equation (3.20) to the example shown in Figure 3.5, the first and last neuron in the first

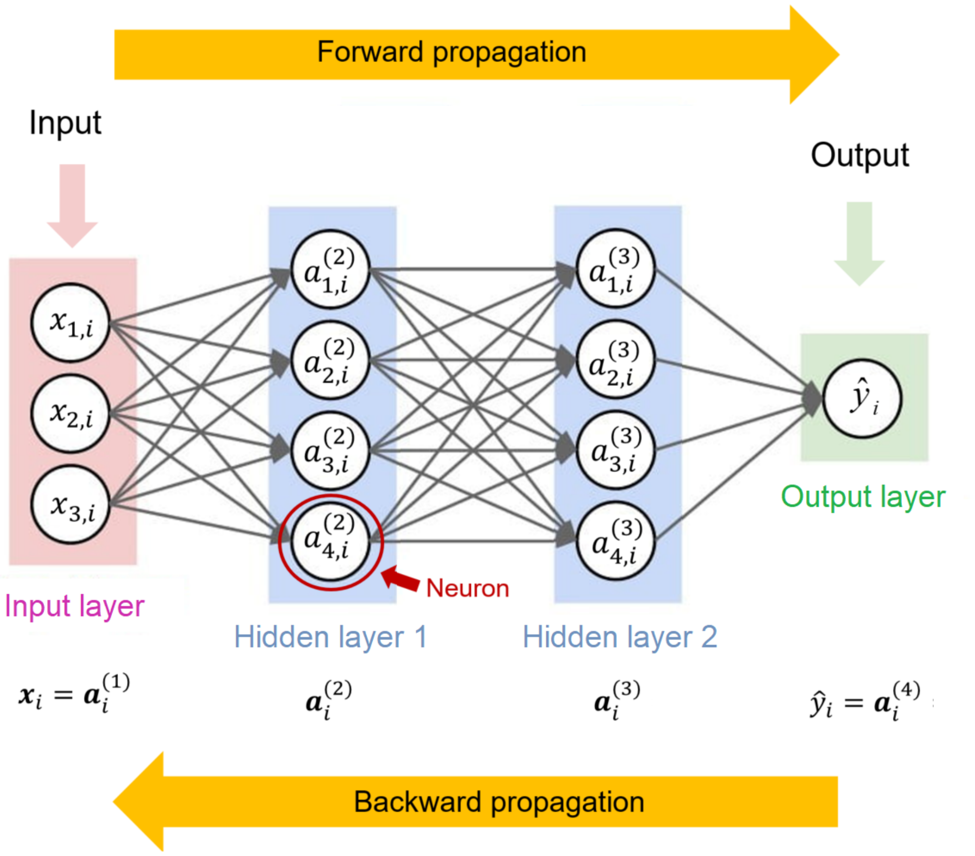


Figure 3.5: Illustration of an ANN architecture with an input layer with three neurons, two hidden layers with four neurons each, and an output layer with one neuron. Note: The bias neuron in the input and hidden layers is not shown in the diagram but is included in the computation.

hidden layer can be estimated as follows

$$a^{(2)}_{1,i} = q\left(w^{(1)}_{1,0} \cdot x_{0,i} + w^{(1)}_{1,1} \cdot x_{1,i} + w^{(1)}_{1,2} \cdot x_{2,i} + w^{(1)}_{1,3} \cdot x_{3,i}\right) = q\left(\sum_{pl_1=0}^{3} w^{(1)}_{1,pl_1} \cdot x_{pl_1,i}\right) = q\left(\mathbf{w}_1^{(1)^T} \cdot \mathbf{x}_i\right) = q\left(\mathbf{w}_1^{(1)^T} \cdot \mathbf{a}_i^{(1)}\right)$$

$$a^{(2)}_{4,i} = q\left(w^{(1)}_{4,0} \cdot x_{0,i} + w^{(1)}_{4,1} \cdot x_{1,i} + w^{(1)}_{4,2} \cdot x_{2,i} + w^{(1)}_{4,3} \cdot x_{3,i}\right) = q\left(\sum_{pl_1=0}^{3} w^{(1)}_{4,pl_1} \cdot x_{pl_1,i}\right) = q\left(\mathbf{w}_4^{(1)^T} \cdot \mathbf{x}_i\right) = q\left(\mathbf{w}_4^{(1)^T} \cdot \mathbf{a}_i^{(1)}\right),$$

and in the second hidden layer as follows

$$a^{(3)}_{1,i} = q\left(w^{(2)}_{1,0} \cdot a^{(2)}_{0,i} + w^{(2)}_{1,1} \cdot a^{(2)}_{1,i} + w^{(2)}_{1,2} \cdot a^{(2)}_{2,i} + w^{(2)}_{1,3} \cdot a^{(2)}_{3,i} + w^{(2)}_{1,4} \cdot a^{(2)}_{4,i}\right) = q\left(\sum_{pl_2=0}^{4} w^{(2)}_{1,pl_2} \cdot a^{(2)}_{pl_2,i}\right) = q\left(\mathbf{w}_1^{(2)^T} \cdot \mathbf{a}_i^{(2)}\right)$$

$$a^{(3)}_{4,i} = q\left(w^{(2)}_{4,0} \cdot a^{(2)}_{0,i} + w^{(2)}_{4,1} \cdot a^{(2)}_{1,i} + w^{(2)}_{4,2} \cdot a^{(2)}_{2,i} + w^{(2)}_{4,3} \cdot a^{(2)}_{3,i} + w^{(2)}_{4,4} \cdot a^{(2)}_{4,i}\right) = q\left(\sum_{pl_2=0}^{4} w^{(2)}_{4,pl_2} \cdot a^{(2)}_{pl_2,i}\right) = q\left(\mathbf{w}_4^{(2)^T} \cdot \mathbf{a}_i^{(2)}\right).$$

where PL_{k-1} is the number of neurons in the previous layer, including one bias neuron. A general formulation for each neuron in the hidden or last layer can be defined as

$$a^{(k)}_{sl_k,i} = q\left(\sum_{pl_{k-1}=0}^{PL_{k-1}} w^{(k-1)}_{sl_k,pl_{k-1}} \cdot a^{(k-1)}_{pl_{k-1-1},i}\right) = q\left(\mathbf{w}_{sl_k}^{(k-1)^T} \cdot \mathbf{a}_i^{(k-1)}\right). \tag{3.22}$$

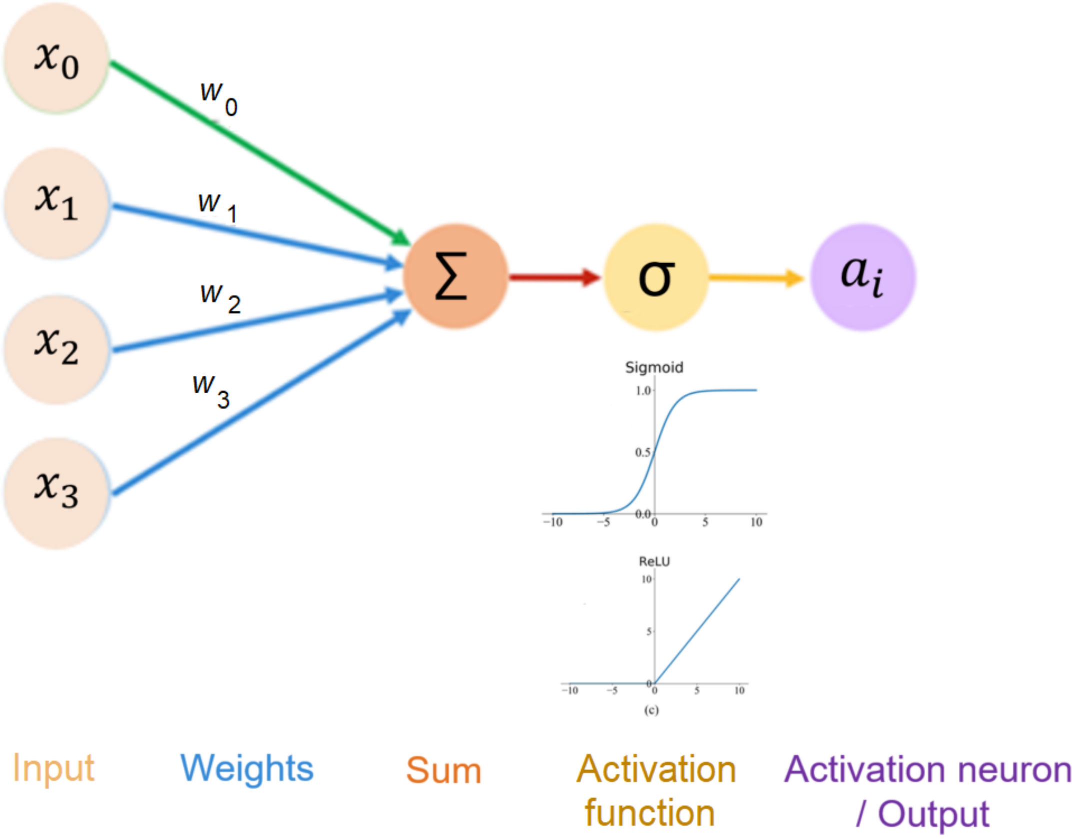


Figure 3.6: Simple representation of ANN procedure for VTEC modeling and forecasting, starting from the input layer with three input neurons plus a bias neuron and multiplying by the associated weights, summarizing in the second step, applying the activation function in the third step, and estimating the value of an activation neuron in the next (hidden) layer, or the final output in the output layer in the last step. Two activation functions are illustrated: Sigmoid (top) and Rectified Linear Unit (ReLU) (below).

If we assume the sigmoid function for the activation function q as in **P-III**, the basis functions above are then sigmoid basis functions. Then the hidden neurons can be considered as a basis expansion of the input $\mathbf{x}_i$, where the parameters of the basis functions are learned from the data. For the last layer, the output of an ANN can be defined as the weighted sum of basis functions of all previous layers as

$$\begin{aligned}
\hat{y}_i = a_i^{(4)} \quad &= q\left(\sum_{pl_3=0}^{4} w_{1,pl_3}^{(3)} \cdot a_{pl_3,i}^{(3)}\right) \\
&= q\left(\sum_{pl_3=0}^{4} w_{1,pl_3}^{(3)} \cdot q\left(\sum_{pl_2=0}^{4} w_{1,pl_2}^{(2)} \cdot a_{pl_2,i}^{(2)}\right)\right) \\
&= q\left(\sum_{pl_3=0}^{4} w_{1,pl_3}^{(3)} \cdot q\left(\sum_{pl_2=0}^{4} w_{1,pl_2}^{(2)} \cdot q\left(\sum_{pl_1=0}^{3} w_{1,pl_1}^{(1)} \cdot a_{pl_1,i}^{(1)}\right)\right)\right) \\
&= q\left(\sum_{pl_3=0}^{4} w_{1,pl_3}^{(3)} \cdot q\left(\sum_{pl_2=0}^{4} w_{1,pl_2}^{(2)} \cdot q\left(\sum_{pl_1=0}^{3} w_{1,pl_1}^{(1)} \cdot x_{pl_1,i}\right)\right)\right). \qquad (3.23)
\end{aligned}$$

If a linear activation function is used in the last layer as in **P-I**, then the output of the ANN becomes

$$\hat{y}_i = \sum_{pl_3=0}^{4} w_{1,pl_3}^{(3)} \cdot q\left(\sum_{pl_2=0}^{4} w_{1,pl_2}^{(2)} \cdot q\left(\sum_{pl_1=0}^{3} w_{1,pl_1}^{(1)} \cdot x_{pl_1,i}\right)\right). \tag{3.24}$$

The equations (3.22) to (3.24) show that each hidden layer and the output layer are modeled by means of basis functions of the neurons of the previous layer. A single-layer feedforward neural network with linear activation in the last layer can be viewed as an adaptive basis function (Hastie et al., 2009). Therefore, basis functions are building blocks for complex functions to approximate the mapping between input and output in ANN.

To implement the backpropagation, an activation function must be differentiable to compute the derivatives of the loss function with respect to the network weights and optimize the weights using an optimization technique to reduce the network errors. In a smaller ANN, the sigmoid function is usually used for the activation function, defined as

$$q(z) = \sigma(z) = \frac{1}{1+e^{-z}} \tag{3.25}$$

while in large/deep neural networks, the ReLU function is more common, formulated as

$$q(z) = \mathrm{ReLU}(z) = \begin{cases} z, & z > 0 \\ 0, & \text{otherwise} \end{cases}. \tag{3.26}$$

In this work, the ReLU function is used in **P-I** for an ANN-MLP model with 32 input neurons, three hidden layers with ten neurons each, and an output layer with one neuron. The sigmoid function is used in **P-III** for the Bayesian ANN model with 14 input neurons, one hidden layer with 32 neurons, and one output layer with one output neuron. ReLU is a computationally much simpler and faster function that significantly reduces both training and evaluation time, which can be helpful for large networks with many neurons. When using the ReLU function, not all neurons are activated at the same time, i.e., they are deactivated when the output of the linear transformation is zero. On the other hand, sigmoid activation requires an exponent to be calculated, which takes more time. Moreover, in very deep networks, a saturation of the activation function can be a problem known as the vanishing gradient problem. The ReLU function is less easy to saturate than the sigmoid activation function (Sharma et al., 2020). It is also observed that the models trained with ReLU converge faster.

The most commonly used optimization algorithm in ANN is the Stochastic Gradient Descent (SGD) (Bottou, 1991, 2004, 2012). Unlike standard gradient descent, which is applied to all training data to update the weights once, SGD updates the weights after each data subset in each iteration, also called a batch, according to the gradient of the cost function for the current example. The batch size controls the number of training samples used for a single gradient update, i.e., an update of the model weights. The number of epochs corresponds to the complete training data set passing through the network, i.e., when an entire data set is passed forward and backward through the ANN. The SGD algorithm is shown to be faster and more reliable in reaching minima than the standard gradient descent and converges even when the loss function is not differentiable everywhere (Bottou, 1991). The basic principle of SGD is to update the weights stepwise, controlled by the learning rate along a preferred direction that is a function of the previous gradient evaluations. The weights are updated as

$$\mathbf{W}_{new}^{(k)} = \mathbf{W}_{old}^{(k)} - \alpha \cdot \frac{\partial Cost}{\partial \mathbf{W}^{(k)}}, \tag{3.27}$$

where α is the learning rate, $Cost$ is the objective cost function defined in (3.4), *old* represents the values initialized in the first forward run of the ANN or the values from the previous backpropagation iteration, while *new* are the values updated during the current ANN backpropagation. The learning rate controls the size of the step that the gradient descent takes towards the minimum (Figure 3.7).

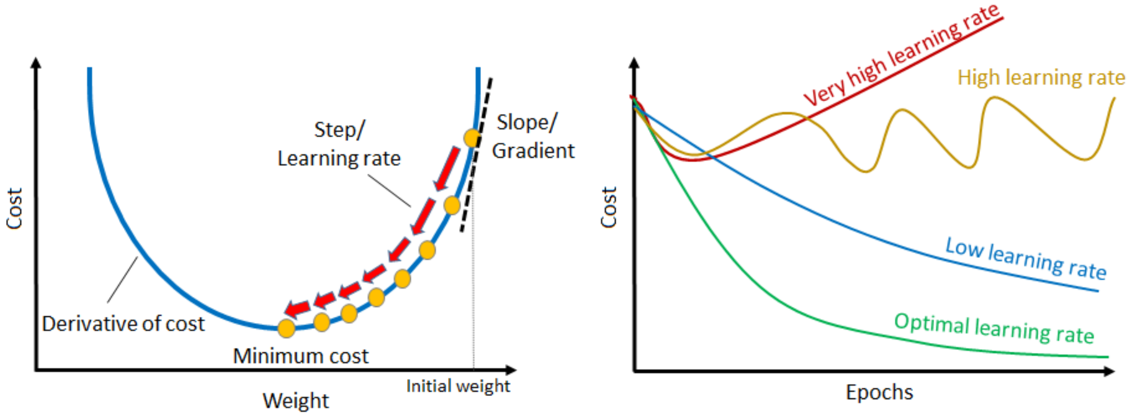


Figure 3.7: Left: Convergence to the minimum cost if an appropriate size of learning rate is used. Right: The effects of different learning rates versus ANN epochs. When the learning rate is low, the cost decays slowly and linearly. At high learning rates, the cost looks more exponential and can even oscillate between convergence and divergence. With an optimal learning rate, the cost converges at a reasonable speed and reaches the minimum.

Consequently, it determines how much the weights will be adjusted with respect to the objective cost gradient, i.e., how fast they will move toward the optimal ones. When α is too small, the gradient descent converges slowly. On the other hand, if α is too large, the gradient descent can overshoot the minimum. So it may fail to converge or even diverge. If the learning rate is low enough, the algorithm converges to the minimum. α is usually kept constant. Convergence will also occur with a fixed learning rate because the gradient descent automatically takes smaller steps as we approach the minimum, i.e., the slope is less steep as the derivatives get closer to zero. The learning rate for the ANN-based VTEC models in **P-I** and **P-III** is found by computing the objective cost function for a range of learning rate values and choosing the one for which the minimum cost is achieved (Figure 3.8, left). The convergence of the objective cost with the chosen learning rate is then checked with training and cross-validation data sets while increasing the number of epochs (Figure 3.8, right). The training cost converges uniformly with increasing epoch number, while cross-validation cost converges to a specific epoch value, here up to 300 epochs. The slight divergence thereafter is a consequence of the variance increase as the model is better fitted to the data (see Section 4.3.1).

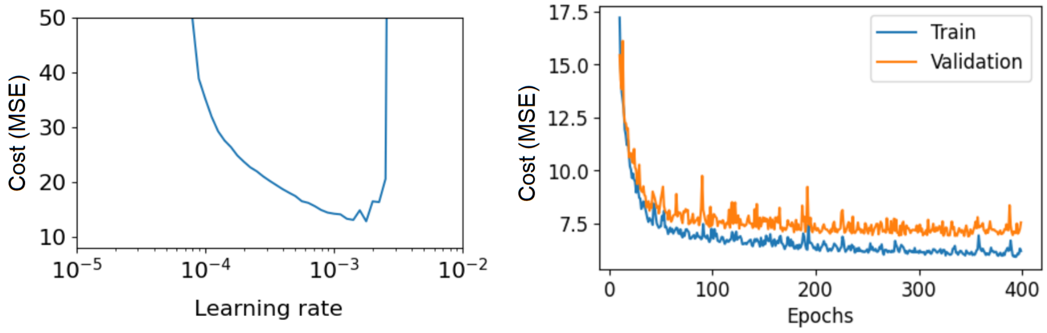


Figure 3.8: Left: The cost with respect to different learning rate sizes for the ANN-based VTEC model in **P-I**. Right: The cost over time for training and cross-validation data for the same model. The cross-validation cost follows the training cost. The gap between the training and cross-validation costs indicates the extent of overfitting. The costs are reported in TECU.

3.3 Probabilistic VTEC Forecasting

This section introduces methods for representing uncertainty and developing probabilistic VTEC forecasting. The main goal of producing a VTEC forecast in a probabilistic framework is to provide information on how reliable the model results are and to quantify the corresponding uncertainties.

3.3.1 Classification of Uncertainty

The uncertainty can be classified into three categories (**P-III**):

- Model parameter uncertainty occurs due to incomplete knowledge, which can be due to a lack of training data, or poor information in training data. This is the deterministic or systematic part of uncertainty, which can be reduced with more knowledge about the system, e.g., by adding more information-rich data. This uncertainty is often referred to as epistemic in the ML literature.
- Data uncertainty: it is related to uncertainty in measurements due to the noise inherent in the data or the stochastic nature of the process generating the data. This is the stochastic or random part of uncertainty and, therefore irreducible. In the ML literature, it is also known as aleatoric uncertainty.
- Uncertainty associated with the limitation of the learning models, i.e., the approximation of the target function. For example, model selection involves a certain choice of free parameters/hyperparameters, and it is not possible to fully explore the hyperparameter space. Thus, it comes down to a trade-off between the complexity of the model and its ability to generalize to unseen data. Ultimately, the ML process consists of various steps of learning and approximating an unknown mapping function from input to output, and the errors and uncertainties associated with these steps may contribute to the uncertainty of the output.

This work focuses on estimating and evaluating model and data uncertainties, while the third type of uncertainty is difficult to quantify. In the following Sections 3.3.2 to 3.3.4.1, various methods applied in this dissertation for the probabilistic VTEC forecast with a 95% confidence interval are presented.

3.3.2 Ensemble Modeling

Ensemble modeling combines multiple diverse models to estimate an outcome using different algorithms, data sets, or both. In this dissertation, the ensemble VTEC model aggregates the mean output across all the base models, i.e., ensemble members, to produce a final VTEC forecast with reduced error on previously unseen data. The model is referred to as SE as described by **CP-II** and **P-III**. The developed SE model improves VTEC forecast compared to the single model within the ensemble by averaging the results over a set of functions of well-performing ML models for VTEC forecast.

The developed SE model combines VTEC models based on learning algorithms of Random Forest (Section 3.2.2), AdaBoost and GBoost (Section 3.2.3), each trained using three data sets with different versions of input features and output. They are prepared from **P-III** data set in Table 3.1, here called D-**P-III**, as:

1. Data set D-**P-III**;
2. Daily differences[3] for both input features and output;

[3] The idea behind computing the daily differences is to remove the dominant, regular daily VTEC variations so that a learning algorithm is trained only on the remaining VTEC variations. In addition, differencing reduces temporal dependencies and trends and stabilizes the mean of the data set, which can improve modeling.

3. Input data combining the input features from D-**P-III** in item 1. and their daily differences from item 2., while the output corresponds to the output from D-**P-III** in item 1.

As a result, nine ensemble member models are created within the SE for each selected VTEC grid point. The randomness in the nine models is introduced by the learning algorithms and the data, i.e., by training the three algorithms mentioned above of the three data subsets individually.

The loss function used in all $M = 9$ ensemble member models is the squared error

$$\mathcal{L}_m = e_{m_i}^2 = (y_i - \widehat{y}_{m_i})^2, \tag{3.28}$$

where $\widehat{y}_{m_i}$ is the VTEC forecast for each of the N observations at time i with $i \in \{1, 2, ..., N\}$ from the m^{th} ensemble member with $m \in \{1, 2, ..., M\}$.

The ensemble approach can be viewed as an approximation of a distribution, and thus, its diversity can be used as an indicator of the model parameter uncertainty (Hüllermeier & Waegeman, 2021). In this case, the results of M independently trained base models are averaged, forming a joined distribution $p(\mathbf{y}|\mathbf{X})$ as

$$p(\mathbf{y}|\mathbf{X}) = \frac{1}{M} \sum_{m=1}^{M} p(\mathbf{y}|\mathbf{X}, \theta_m), \tag{3.29}$$

where θ represents a set of model parameters. The ensemble output $\widehat{y}_i$ is estimated as the ensemble mean μ_i

$$\widehat{y}_i = \mu_i = \frac{1}{M} \sum_{m=1}^{M} \widehat{y}_{m_i}, \tag{3.30}$$

and the standard deviation of the ensemble members with respect to the ensemble mean is defined as

$$\sigma_i = \sqrt{\frac{1}{M} \sum_{m=1}^{M} (\widehat{y}_{m_i} - \widehat{y}_i)^2}. \tag{3.31}$$

The ensemble spread, i.e., the confidence interval, is represented in a probabilistic framework in terms of lower bounds LB and upper bounds UB with 95% confidence as

$$UB = \widehat{y}_i + 2\sigma_i, \quad LB = \widehat{y}_i - 2\sigma_i. \tag{3.32}$$

3.3.3 Quantile Loss and Quantile Gradient Boosting

Quantile methods estimate the median and/or other quantiles of the response variable, in contrast to the least squares method, which estimates the mean of the response variable. Quantile estimation is integrated into the ML approach in this dissertation and in **P-III** by substituting the squared error loss function from (3.5) with the error loss multiplied by selected quantile values as defined in (3.33). This loss produces a parametric distribution shift, allowing a learning algorithm to learn a particular quantile instead of the mean. More precisely, the quantile loss is obtained by multiplying quantile values β by positive and negative residuals e_i between the ground truth of the output variable y_i and the estimated output $\widehat{y}_i = \widehat{F}(\mathbf{x}_i)$ as

$$\mathcal{L}(y_i, \widehat{F}(\mathbf{x}_i)) = \begin{cases} \beta \cdot e_i & \text{if} \quad e_i \geq 0, \\ (\beta - 1) \cdot e_i & \text{if} \quad e_i < 0 \end{cases} \qquad e_i = y_i - \widehat{y}_i \tag{3.33}$$

The quantile values of β are set to 0.025 and 0.975 for estimating the lower and upper confidence bounds, respectively, to obtain a confidence interval of 95%. The mean quantile $\beta = 0.50$ provides the median of the probabilistic VTEC forecast.

The main properties of the quantile loss function according to the results in Chan (2021); Amell et al. (2022); Tagasovska & Lopez-Paz (2019) can be summarized as follows

- It models heterogeneous variations in the objective distribution.
- It avoids the distribution assumption, which means that it can be used when the error distribution is not Gaussian.
- It can be easily combined with any learning algorithm by simply substituting its loss function.
- It can model data uncertainty.

P-III combines the quantile loss with the GBoost learning algorithm from Section 3.2.3, resulting in a Quantile Gradient Boosting (QGB) VTEC model.

3.3.4 Bayesian Neural Network

The Bayesian Neural Network (BNN) represents a modification of an ANN in which probability distributions of the network weights replace the deterministic/single value weights. The probability distributions are used to model the uncertainty in the weights and consequently can be used to estimate the uncertainty due to the model parameter uncertainty based on Bayes' theorem. The posterior parameters θ to be trained are the mean μ and the standard deviation σ of the posterior weight distribution. The idea is to learn a distribution of the weights approximating the Bayesian posterior distribution. The parameters θ can be learned by variational Bayesian inference facilitated by a standard neural network backpropagation technique during the training process (Blundell et al., 2015). This technique is called Bayes by Backprop.

Given a training data set $D = (x_i, y_i)$ with $i = 1, 2, ...N$, the likelihood function $p(D|w)$ can be constructed, which is a function of the weights w (Blundell et al., 2015). Maximizing the likelihood function yields the maximum likelihood estimate of w. The usual optimization objective in ML is to minimize the negative log-likelihood. Multiplying the likelihood $p(D|w)$ by a prior distribution $p(w)$ is proportional to the posterior distribution $p(w|D) \propto p(D|w)p(w)$ according to Bayes' theorem (Koch, 2018). An analytical solution for the posterior $p(w|D)$ in neural networks is not feasible. We can approximate the true posterior with a variational distribution $q(w|\theta)$ of the function whose parameters we want to estimate. This can be done by minimizing the Kullback-Leibler (KL) divergence between $q(w|\theta)$ and the true posterior $p(w|D)$.

KL divergence measures the dissimilarity of the variational probability distribution of the weights $q(w|\theta)$ with the posterior probability distribution of the weights $p(w|D)$. It is also called relative entropy in probability and information theory (Murphy, 2012). For this problem, it can be defined as

$$KL[p(w|D)\|q(w|\theta)] = p(w|D) \cdot (\log p(w|D) - \log q(w|\theta)) = p(w|D) \cdot \log \frac{p(w|D)}{q(w|\theta)}. \tag{3.34}$$

Normally, the reverse KL divergence is used in the variational inference (Murphy, 2012), which is also done here, and it is

$$\begin{aligned} KL(q(w|\theta)\|p(w|D)) \quad &= q(w|\theta) \cdot \log \frac{q(w|\theta)}{p(w|D)} \\ &= q(w|\theta) \cdot \log \frac{q(w|\theta)}{p(w)p(D|w)} \\ &= q(w|\theta) \cdot \log \frac{q(w|\theta)}{p(w)} - \log p(D|w) \\ &= KL[q(w|\theta)\|p(w)] - \log p(D|w). \end{aligned} \tag{3.35}$$

The idea behind variational inference is to choose an approximation $q(w|\theta)$ to the distribution and then try to make this approximation as close as possible to the true posterior $p(w|D)$. This reduces the variational inference to an optimization problem, and from (3.35), the optimization objective function can be defined as

$$Cost = -\log p(D|w) + KL[q(w|\theta)\|p(w)], \tag{3.36}$$

which can be split into two parts: the left term on the right side corresponds to the negative log-likelihood, and the right term on the right side is the KL divergence between the variational distribution $q(w|\theta)$ and the prior $p(w)$, which can also be seen as the regularization term (Blundell et al., 2015). Here we assume a Gaussian distribution as in (3.39) and fixed data noise in (3.40), then the left term in (3.36) corresponds to the standard MSE loss as in (3.5).

The prior weight distribution is defined as a Gaussian distribution with a mean $\mu = 0$ and a diagonal covariance with a standard deviation $\sigma = 1$. A sample of the weights w is obtained by randomly sampling ϵ from $\mathcal{N}(0,1)$, then scaling it by a standard deviation σ, and shifting it by a mean μ as

$$w = \mu + \sigma \cdot \epsilon. \tag{3.37}$$

For numerical stability, the network is parametrized with ρ instead of σ. ρ is transformed with the so-called softplus activation function as

$$\sigma = \log(1 + \exp(\rho)) \tag{3.38}$$

to ensure that σ is always non-negative (Blundell et al., 2015). The algorithm proceeds by sampling from the variational posterior distribution, computing a forward pass through a network, and then backpropagating through the model parameters to update them. The gradients are calculated with respect to the mean and the standard deviation to update the previous distribution parameters using the SGD optimization algorithm, similarly as in (3.27), except that here we use distributional parameters σ and μ instead of deterministic weights.

Each time the model is run with the same input variables, a new set of parameters is sampled from the distribution, and a result is produced. Running the model multiple times makes it possible to examine how this affects the model results. For instance, running the model with 100 iterations result in 100 ensemble members. If the results are consistent, it means that the model is confident. In this dissertation and in **P-III**, the VTEC forecast is estimated as the mean of an ensemble of results from 100 iterations, while the 95% confidence interval is calculated as in (3.32).

The described BNN implementation is deterministic, i.e., it produces a single VTEC forecast for each run, and the uncertainty is calculated from an ensemble of many iterations. The Bayesian approach to neural network modeling aims to capture the model parameter uncertainty due to limited training data. The BNN can be extended to a probabilistic implementation by enabling the model to output a distribution described in the following section.

3.3.4.1 BNN including Data Uncertainty

To implement BNN in a probabilistic way, the normal distribution parameters, μ and σ, are modeled as output. Usually, the observation noise is assumed to be fixed, as done in the BNN implementation above. In this step, the noise will be data-dependent and learned as a function of the data. Therefore, the Bayesian Neural Network including Data Uncertainty (BNN+D) will be able to capture also data uncertainty due to irreducible noise in the data or due to the stochastic nature of the process of generating the data.

As mentioned above, we assumed the Gaussian likelihood, parameterized by the mean and standard deviation as

$$p(D|w) = l(y|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2}. \tag{3.39}$$

The Negative Log-Likelihood (NLL) loss reads

$$\begin{aligned}
\mathcal{L} &= -\log l\left(y_i|\mu,\sigma\right) \\
&= -\left[\log\left[\frac{1}{\sigma\sqrt{2\pi}}\right] + \log\left[e^{-\frac{1}{2}\left(\frac{y_i-\mu}{\sigma}\right)^2}\right]\right] \\
&= -\left[\log\frac{1}{\sigma} + \log\frac{1}{\sqrt{2\pi}} + \left[-\frac{1}{2}\left(\frac{y_i-\mu}{\sigma}\right)^2 \cdot \log(e)\right]\right] \\
&= -\left[\log(1) - \log(\sigma) + \log(1) - \log\sqrt{2\pi} + \left[-\frac{1}{2}\frac{(y_i-\mu)^2}{\sigma^2}\right]\right] \\
&= -\left[-\log(\sigma) - \log\sqrt{2\pi} - \frac{1}{2}\frac{(y_i-\mu)^2}{\sigma^2}\right] \\
&= -\left[-\frac{1}{2}\cdot\log(\sigma^2) - \frac{1}{2}\cdot\log(2\pi) - \frac{1}{2}\frac{(y_i-\mu)^2}{\sigma^2}\right] \\
&= \frac{1}{2}\left[\log(\sigma^2) + \log(2\pi) + \frac{(y_i-\mu)^2}{\sigma^2}\right] \\
&= \frac{1}{2}\left[\log(\sigma^2) + C + \frac{(y_i-\mu)^2}{\sigma^2}\right] \\
&\approx \frac{1}{2}\left[\log(\sigma^2) + \frac{(y_i-\mu)^2}{\sigma^2}\right],
\end{aligned} \tag{3.40}$$

where μ is the predicted mean and σ is the standard deviation of the output. C is a constant equal to $\log(2\pi)$, which can be neglected. In this case, instead of the MSE loss in (3.36), the NLL loss in (3.40), which accounts for the observation noise, is used to compute how likely the ground truth are to deviate from the estimated distribution produced by the model. Also, the model can provide a probability distribution as an output, i.e., μ and σ, instead of a single point estimate. To provide μ and σ as the output of the VTEC probabilistic model, an output layer is created with two neurons: one for the mean forecast output and one for the standard deviation output. The 95% confidence interval is computed from the predictive standard deviation according to Equation (3.32).

3.4 VTEC Relative Feature Importance

It is often useful to provide information about the underlying relationships between the input variables and the output of the model to improve the interpretation and understanding of what the model has learned. In particular, this includes understanding the most influential input variables to the approximation function $\hat{F}(\mathbf{X})$.

A decision tree-based method is used in this study to estimate the relative importance or contribution of each input variable to the VTEC forecast. Let us define a decision tree with J terminal nodes and $J-1$ internal (non-terminal) nodes, e.g., the decision tree in Figure 3.9. The nodes of the tree are partitioned during training by a particular split variable $\widehat{\mathbf{x}}_t$ with $t \in \{0, 1, ..., J-1\}$, and the value of that variable s. Specifically, at each node t, one of the input variables $\widetilde{\mathbf{x}}_p$ is used as the split variable $\widehat{\mathbf{x}}_t$ to split the node into two subnodes, as described in 3.2.1.

The approach starts at the root node when all observations belong to a single region R, i.e., 17,488 observation samples in Figure 3.9. The mean VTEC of all observations within the region R is 10.34 TECU, and the squared error is 14 TECU. The decision split in the root node is given as $\widetilde{\mathbf{x}}_1 \leq 6.5$, representing the split point, while the input variable $\widetilde{\mathbf{x}}_1$, representing HoD, is the split variable of the region R. The input space is then divided into two subregions: R_1 (on the left, where the condition

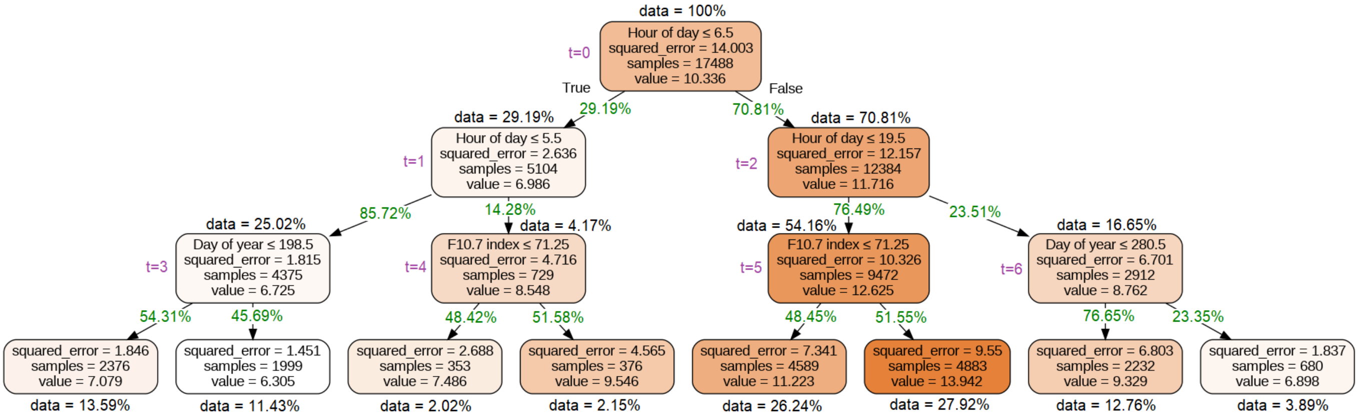


Figure 3.9: A decision tree of depth 3 for VTEC nowcast at 10°E 40°N. The input features used as split variables consist of DoY, HoD, and F10.7 index. t enumerates the splitting nodes. Data quantifies the percentage of data contained in each node relative to the total data set in the root node. The percentage of the data in green is calculated relative to the data in the "parent" node. A "parent" node is a node that is split into two "child" nodes, i.e., the right and left subnodes. The color shading of the nodes from lighter to darker corresponds to the size of the VTEC output from smaller to larger.

is true, i.e., $\widetilde{\mathbf{x}}_1 \leq 6.5$), and R_2 (on the right, where the condition is False, i.e., $\widetilde{\mathbf{x}}_1 > 6.5$). Therefore, considering a splitting variable $\widehat{\mathbf{x}}_t = \widetilde{\mathbf{x}}_p$ and a split point s, two splitting regions are defined: R_1 and R_2. The squared error loss is used to measure the improvement after the split.

The improvement after splitting a single region with the splitting variable $\widetilde{\mathbf{x}}_p$ can be calculated as the importance of the splitting variable for that specific node as

$$imp_t(\widehat{\mathbf{x}}_t = \widetilde{\mathbf{x}}_p) = \frac{d(t) \cdot SqE - (d_1(t) \cdot SqE_1 + d_2(t) \cdot SqE_2)}{100\%}, \tag{3.41}$$

where $d(t)$ represents 100% of the samples from the region R corresponding to node t, while $d_1(t)$ and $d_2(t)$ are the percentages of the samples from $d(t)$ that go into the left and right subnodes, respectively, forming subregions R_1 and R_2. These percentages are shown in green in Figure 3.9. The term SqE is the squared error in region R, SqE_1 and SqE_2 are squared errors in the subregions R_1 and R_2, respectively. The term $imp_t(\widehat{\mathbf{x}}_t = \widetilde{\mathbf{x}}_p)$ represents the empirical improvement of the objective function when $\widetilde{\mathbf{x}}_p$ is used as the splitting variable of node t in the tree, and it is calculated for each splitting node in the tree.

The importance of the feature $\widetilde{\mathbf{x}}_p$ in the tree can be calculated as the summation of the importance in each node where it is used as a splitting variable, weighted by the percentage of the data from the entire data set as

$$Imp_T(\widetilde{\mathbf{x}}_p) = \frac{\sum_{t=0}^{J-1} imp_t(\mathbf{x}_t = \widetilde{\mathbf{x}}_p) \cdot D(T)_t}{100\%}, \tag{3.42}$$

where $D(T)_t$ is the percentage of data in node t relative to the entire training data set in the root node, noted as data in Figure 3.9. The summation is performed over t internal nodes of the tree. $Imp_T(\widetilde{\mathbf{x}}_p)$ defines the importance of the input variable $\widetilde{\mathbf{x}}_p$ in a decision tree as the sum of the improvements in minimizing the objective function as a result of using that variable for spliting a node multiplied by the probability of using that node for new data sample.

The relative importance is calculated by dividing the importance of each feature in the tree by the sum of the importance of the features estimated for each node in Equation (3.41), expressed as

$$RelImp_T(\widetilde{\mathbf{x}}_p) = \frac{Imp_T(\widetilde{\mathbf{x}}_p)}{\sum_{p=0}^{P-1}\sum_{t=0}^{J-1} imp_t(\widehat{\mathbf{x}}_t = \widetilde{\mathbf{x}}_p)}. \tag{3.43}$$

For a collection of decision trees $\{T_m\}_1^M$, Equation (3.43) can be generalized as the relative importance averaged over all trees

$$RelImp(\widetilde{\mathbf{x}}_p) = \frac{1}{M}\sum_{m=1}^{M} RelImp_{T_m}(\widetilde{\mathbf{x}}_p). \tag{3.44}$$

In summary, the main components for calculating feature importance for a decision tree are: the improvement of the objective cost function when $\widetilde{\mathbf{x}}_p$ is used as the splitting variable, the summation over all nodes of the tree that use feature $\widetilde{\mathbf{x}}_p$ as the splitting variable, and the probability of using that node for a sample data point, expressed as a percentage of the data in that node.

In the next part, the calculation of the relative feature importance for VTEC is demonstrated using the example from Figure 3.9. Let us start from the root node. Firstly, we compute the improvement in the loss after splitting the root node ($t = 0$) by the feature $\widetilde{\mathbf{x}}_1$, i.e., HoD into two subnodes. Subsequently, the improvement can also be calculated for other nodes ($t = 1$ and $t = 2$), where the feature $\widetilde{\mathbf{x}}_1$ is again used as a splitting variable. Starting from Equation (3.41), the improvements for each node t where $\widetilde{\mathbf{x}}_1$ is used as the splitting point can be calculated as

$$\begin{aligned}
imp_0(\widehat{\mathbf{x}}_0 = \widetilde{\mathbf{x}}_1) &= \frac{100\% \cdot 14.003 - (29.19\% \cdot 2.636 + 70.81\% \cdot 12.157)}{100\%} = 4.625\\
imp_1(\widehat{\mathbf{x}}_1 = \widetilde{\mathbf{x}}_1) &= \frac{100\% \cdot 2.636 - (85.72\% \cdot 1.815 + 14.28\% \cdot 4.716)}{100\%} = 0.407\\
imp_2(\widehat{\mathbf{x}}_2 = \widetilde{\mathbf{x}}_1) &= \frac{100\% \cdot 12.157 - (76.49\% \cdot 10.326 + 23.51\% \cdot 6.701)}{100\%} = 2.683
\end{aligned}$$

Similarly, the calculations can be performed for other nodes where the splitting variable is the feature $\widetilde{\mathbf{x}}_0$, i.e., DoY and the feature $\widetilde{\mathbf{x}}_2$, i.e., $F10.7$, as

$$\begin{aligned}
imp_3(\widehat{\mathbf{x}}_3 = \widetilde{\mathbf{x}}_0) &= \frac{100\% \cdot 1.815 - (54.31\% \cdot 1.846 + 45.69\% \cdot 1.451)}{100\%} = 0.150\\
imp_6(\widehat{\mathbf{x}}_6 = \widetilde{\mathbf{x}}_0) &= \frac{100\% \cdot 6.701 - (76.65\% \cdot 6.803 + 23.35\% \cdot 1.837)}{100\%} = 1.058\\
imp_4(\widehat{\mathbf{x}}_4 = \widetilde{\mathbf{x}}_2) &= \frac{100\% \cdot 4.716 - (48.42\% \cdot 2.688 + 54.58\% \cdot 4.565)}{100\%} = 0.923\\
imp_5(\widehat{\mathbf{x}}_5 = \widetilde{\mathbf{x}}_2) &= \frac{100\% \cdot 10.326 - (48.45\% \cdot 7.341 + 51.55\% \cdot 9.55)}{100\%} = 1.846
\end{aligned}$$

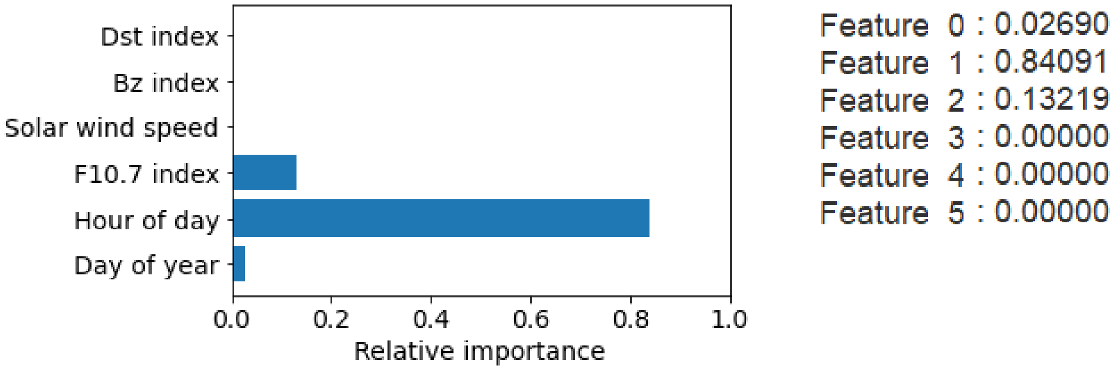
The importance Imp_T of each input feature for the tree T in Figure 3.9 can be calculated according to Equation (3.42) as

$$\begin{aligned}
Imp_T(\widetilde{\mathbf{x}}_0) &= \frac{25.02\% \cdot 0.150 + 16.65\% \cdot 1.058}{100\%} = 0.214\\
Imp_T(\widetilde{\mathbf{x}}_1) &= \frac{100\% \cdot 4.625 + 29.19\% \cdot 0.407 + 70.81\% \cdot 2.683}{100\%} = 6.644\\
Imp_T(\widetilde{\mathbf{x}}_2) &= \frac{4.17\% \cdot 0.923 + 54.16\% \cdot 1.846}{100\%} = 1.038
\end{aligned}$$

Finally, the relative importance of the features is computed, following Equation (3.43), as

$$\begin{aligned}
RelImp_T(\tilde{\mathbf{x}}_0) = \frac{0.214}{0.150+1.058+4.625+0.407+2.683+0.923+1.846} &= \frac{0.214}{7.896} = 0.027 \\
RelImp_T(\tilde{\mathbf{x}}_1) &= \frac{6.644}{7.896} = 0.841 \\
RelImp_T(\tilde{\mathbf{x}}_2) &= \frac{1.038}{7.896} = 0.132
\end{aligned}$$

Figure 3.10 shows the numerical and graphical results of the feature importance calculation using the Python scikit-learn library. The results are consistent with the calculation above.



```
Feature  0 : 0.02690
Feature  1 : 0.84091
Feature  2 : 0.13219
Feature  3 : 0.00000
Feature  4 : 0.00000
Feature  5 : 0.00000
```

Figure 3.10: Relative feature importance estimated with the library function *feature_importances_* from the library *sklearn.tree.DecisionTreeRegressor*. Since the tree is small, only three of the six input features were considered for the growth of the tree.

4 ML-VTEC Model Development

The ML-based VTEC model development is an iterative process. The workflow can be summarized into four main phases, as shown in Figure 4.1:

1. Problem formulation and data acquisition;
2. Data exploration and preparation including feature engineering;
3. Model training and optimization;
4. Model evaluation and deployment.

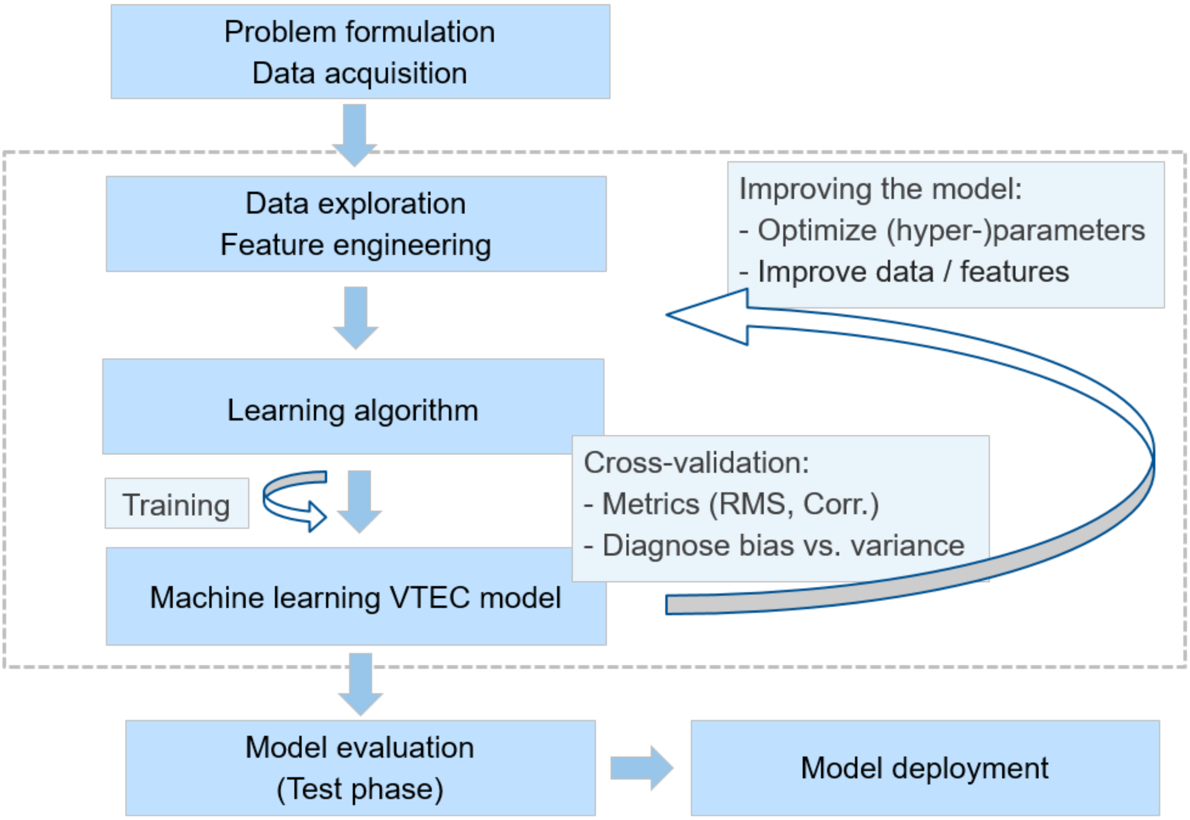


Figure 4.1: The ML-based VTEC model development workflow begins with initial data selection, feature engineering, and learning algorithm selection. The data and hyperparameters are iteratively improved based on the cross-validation analysis. Once the process is complete, the model is evaluated on new data and can be deployed. (adapted from **CP-I**).

The problem in this dissertation is defined and formulated in such a way that it can be solved using ML and supervised learning techniques. The data needed for the problem is also defined and collected. Other vital questions related to the data are: Is the data representative of the task, of high quality, and of sufficient size? The data should accurately reflect the real-world situation and be representative of the ultimate goals of the modeling and forecasting. This dissertation addresses these questions as much as possible before training a model.

The goal of data exploration and preparation in this dissertation is to analyze the data using visualization and statistical methods, clean the data by finding missing values and outliers, remove duplicate or irrelevant data, and prepare the data set for a learning algorithm. Feature engineering involves transforming the raw data into suitable features that better represent the underlying problem, with two main goals:

1. Preparation of an appropriate input data set compatible with the requirements of the modeling problem and the learning algorithm, as discussed in Section 4.1.
2. Improve the performance of the ML models by properly selecting the input features, as discussed in Sections 4.1.3, 4.1.4, and 5.6.

Data preparation is often the most time-consuming step in ML model development. In this dissertation, the first attempt at model development uses initially cleaned data, and depending on the performance of the model and problems encountered, the data are iteratively and systematically improved.

When building an ML model, it is necessary to decide which algorithms and model architecture to use. In this dissertation, an initial model is trained, and the error analysis is used to improve and optimize the model by tuning the hyperparameters, as explained in Section 4.3, and improving the data set in terms of length, resolution, input feature selection, and so on. This process requires many iterations.

The developed models in this dissertation are evaluated at the end with data not used to build the models to independently assess their accuracy. In the model implementation or deployment phase, when the model is applied in real-time, the essential components are monitoring and maintaining the model by tracking various metrics.

4.1 Data Acquisition and Preparation

The performance of an ML model is highly dependent on the data, and therefore the data should be prepared and processed in a way that improves learning. To develop an ML-based VTEC model, it is crucial to determine a combination of input and output data and perform data preprocessing.

4.1.1 VTEC Data

The VTEC values in **P-II**, **P-III**, **CP-I** and **CP-II** are extracted from the GIMs with hourly resolution provided by CODE available in the Crustal Dynamics Data Information System (CDDIS) directory of NASA's archive of Space Geodesy data[(1)]. VTEC values are prepared for three grid points at 10°E 70°N, 10°E 40°N, and 10°E 10°N, reflecting high-, mid- and low-latitude ionosphere regions.

For the high-resolution ML-based VTEC models in Sections 5.4 and 5.5, the VTEC data are obtained from the GIM UQRG generated at UPC-IonSAT at 15-minute resolution by combining tomography based on a multilayer voxel model with kriging interpolation (Hernández-Pajares et al., 2017; Roma-Dollase et al., 2018). VTEC values are extracted at 10° longitude, and 40°, 45°, 50°, 55°, 60° latitude.

In **P-I**, VTEC data are estimated from observations of permanent GNSS stations belonging to the CORS and the EPN (Figure 4.2). The RIM IONOWB for the region of the Western Balkans is developed using GPS observations from the following CORS networks: the Albanian GNSS Permanent Stations

[(1)] https://cddis.nasa.gov/archive/gnss/products/ionex

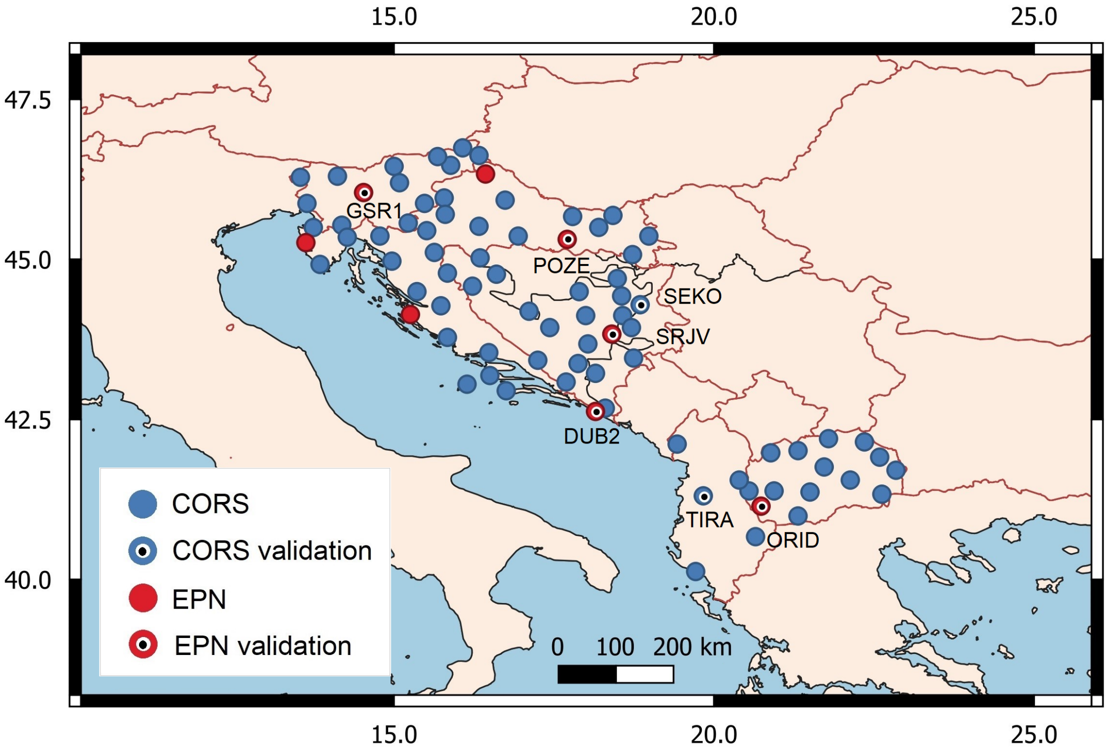


Figure 4.2: The locations of the CORS (blue dots) and the EPN (red dots) dual-frequency stations whose observations are used for regional VTEC modeling. The names of the stations used for the RIMs validation are indicated on the map. (adapted from **P-I**)

(AlbGNSS), the Bosnia and Herzegovina Positioning Service (BIHPOS), the Croatian positioning system (CROPOS), the Macedonian positioning system (MAKPOS), and the Slovenia Geodesy Navigation Location (SIGNAL). In addition, observations from eight EPN stations in this region are used. Observations from the CORS networks in Serbia, Kosovo, and Montenegro were unavailable for the study. The network selected for the RIM IONOWB includes about 80 CORS and EPN stations ranging from about 40°N to 47°N and 13°E to 23°E. The GNSS data are processed in the Bernese GNSS software version v.5.2 (Dach et al., 2015)[(2)]. The ionosphere mapping is performed at the undifferenced/zero-difference level by analyzing the geometry-free linear combination L_4 of phase observations[(3)] formed by subtracting observational data at different frequencies as in Equation (2.5). This means that all frequency-independent effects, such as the satellite-receiver geometrical range, clock errors, and tropospheric delay, are eliminated, while the ionospheric range delay remains. Thus, the L_4 linear combination is used to estimate VTEC multiplied by the mapping function, and the modeling is done by Taylor series expansion in (2.10).

In **CP-III**, VTEC values are estimated with a time sampling of 30 seconds from GNSS (GPS + GLONASS) observations of the EPN station SRJV at 43.87°, 18.41°, and downloaded from the EPN File Transfer Protocol (FTP) server: `https://\gls{epn}cb.eu/ftp/obs/`. The calibration method of Ciraolo et al. (2007) is applied to the carrier phase measurements.

[(2)] Details of the processing steps in the Bernese GNSS Software are described in Section 2.2 of **P-I**.

[(3)] The carrier phases are not fitted to code observations, but only phase observations are utilized in **P-I**.

4.1.2 Space Weather Data

To model solar-terrestrial processes and the effects of solar events and geomagnetic storms on the ionosphere, solar and geomagnetic activity data are also entered as input variables into ML-based VTEC models.

Data of solar activity, solar wind, the IMF, and the GMF fields are part of the NASA/Goddard Space Flight Center, USA (GSFC) OMNI data set and can be obtained from OMNI-Web. The low-resolution OMNI data used in **P-II**, **P-III**, **CP-I** and **CP-II** are available at `https://omniweb.gsfc.nasa.gov/form/dx1.ht\gls{ml}`. The high-resolution OMNI data used in Section 5.4 and 5.5 are available at `https://omniweb.gsfc.nasa.gov/form/omnimin.ht\gls{ml}`. The low-resolution OMNI-Web data used consist of sunspot number R (daily resolution), F10.7 solar radio flux (daily resolution), solar wind plasma speed (1-hour resolution), the IMF Bz index (1-hour resolution), the GMF Dst index (1-hour resolution), the GMF Kp index (3-hour resolution), the AE index (1-hour resolution), as shown in Table 3.1. From the high-resolution OmniWeb data, the GMF SYM/H index is used in this work. It measures the intensity of the storm-time ring current similar to the Dst index, but with the advantage of 1-minute resolution (Wanliss & Showalter, 2006b).

In **P-II**, **P-III**, **CP-I**, and **CP-II**, the data are prepared as 1-hour time samples. A few missing values are replaced by the average of a previous and subsequent value. Data with a lower time resolution, such as the F10.7 index, R, and Kp index, are interpolated with the previous values, as this technique is also implemented in the OMNI-Web to create 1-hour samples. More specifically, the value of the Kp index at a given time is used for the next two hours until the next value of Kp is available.

To model the solar activity with high-resolution data, two types of solar irradiance observations are tested. One represents observations from the LYRA instrument on board the PROBA2 (PRoject for Onboard Autonomy) satellite, which measures solar soft Xray + EUV wavelengths. The other is Ly-$\boldsymbol{\alpha}$ from the GOES (Geostationary Operational Environmental Satellite) satellite, which measures solar UV irradiance. Using high-resolution solar data, this dissertation develops for the first time a forecast model of the ionosphere response to a solar flare using an ML approach.

The chromospheric Ly-α emissions at 121.6 nm can represent a large fraction of the photon energy of the solar flare reaching the Earth and describe the ionosphere enhancements caused by solar flares (Milligan et al., 2020). In addition, Ly-α emissions are significantly dependent on the position of the flare on the Sun, i.e., they are lower when the flare occurs near the limb of the Sun (Milligan et al., 2020). This center-to-limb variation is consistent with the geo-effectiveness of solar flares depending on the distance of the flare from the solar disk center, as explained in Section 2.2. Calibrated 1-min Ly-α data are obtained from GOES-15[(4)] for the period January 2013 to December 2016, and GOES-16[(5)] for the period February to December 2017. During preprocessing, the data are cleaned of flags such as Earth eclipse. However, there are systematic drops in the Ly-α data from around 7:30 to 11:30 UTC for GOES-15 and from around 2:30 to 8:30 UTC for GOES-16 data. These drops occur due to geocoronal absorption by the Earth's upper atmosphere and are removed.

The PROBA2 satellite with LYRA instrument orbits the Earth in a polar, dawn–dusk Sun-synchronous orbit at an altitude of about 720 km. Based on the level of degradation and data availability, observations from the Zirconium channel of the nominal unit 2 with a bandwidth 6 – 20 nm + <2 nm and purity of 92.2% denoted as level 3 products are selected in this dissertation (Table 1 in Dominique et al. (2013)). They are available from the PROBA2 Science Center, Royal Observatory of Belgium, Brussels (ROB) at `https://proba2.sidc.be/lyra/data/bsd/`. The data are calibrated and averaged over 1 min. However, the level 3 data contain non-solar anomalies not corrected during calibration but must be filtered out in data preprocessing (Figure 4.3). Thus, the following anomalies are filtered out during

[(4)] Calibrated GOES-15 data for the Ly-α channel are obtained from `https://www.ncei.noaa.gov/data/goes-space-environment-monitor/access/euvs/GOES_v4/G15/`

[(5)] Calibrated GOES-16 data for the Ly-α channel are downloaded from `https://data.ngdc.noaa.gov/platforms/solar-space-observing-satellites/goes/goes16/l2/data/euvs-l2-avg1mscience/`

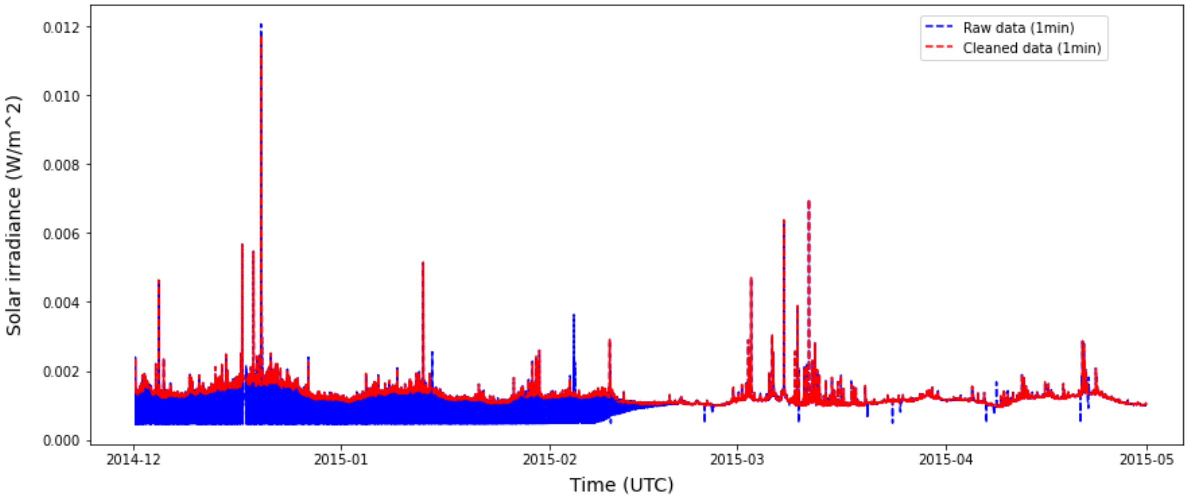


Figure 4.3: PROBA2 LYRA 1-min data: raw (blue) and cleaned (red) data for the period December 2014 - May 2015.

data preprocessing:

- Offpoint of PROBA2 during calibration or scientific campaigns, resulting in signal fluctuations.
- Calibration campaigns, on average once every two weeks.
- Wide-angle rotations of the spacecraft, which are systematic effects that occur four times per orbit (one orbit is 100 minutes long) when the spacecraft rotates 90° around the axis facing the Sun to avoid shadowing by the Earth.
- Winter UV occultations when PROBA2 transits the Earth's shadow from November to February. They last a maximum of 25 minutes per orbit. Because of their regularity, they are easy to spot.
- Slow stabilization or recovery of the detectors after they have been exposed to light during calibration. It can take several hours. In this dissertation, 8 hours of data were cleaned after the calibration phase.
- South Atlantic Anomaly (SAA) disturbances occur when electrons produced by high-energy protons hit the detectors during the transit of the SAA, making the LYRA signal noisier.
- Moon in LYRA, i.e., solar eclipse.
- Spacecraft anomaly, which leads to anomalous and missing data.

To obtain 15-minute samples for the high-resolution ML-based VTEC forecasting in Sections 5.4 and 5.5, the SYM/H, LYRA, and Ly-α data are averaged over the preceding 15-minute period.

4.1.3 Feature Engineering

Feature engineering is the concept of deriving features from raw historical data to establish valid relationships between the input features and the output. Calculating a derived feature from one or more available variables and selecting the right ones can enhance learning performance. In this dissertation, new features are derived from time information, lags, rolling windows, differentiation, and scaling, which are explained in more detail below.

Two datetime features are created from the timestamp value of each observation: the hour of the day (HoD) and day of the year (DoY). By extracting these features, an ML-based VTEC model can capture temporal patterns, such as diurnal and seasonal patterns prevalent in VTEC variability.

Lag input features are values from prior timesteps created under the assumption that events in the past may influence the future or contain some inherent information about the future. For example, the VTEC feature of the previous day at a particular time, e.g., 1:00 p.m., can be created to forecast a similar VTEC value at 1:00 p.m. the next day, considering regular daily VTEC variations. Therefore, the lagged VTEC from the previous hour is used for the 1-hour VTEC forecast, while the lagged VTEC of the prior day is used for the 1-day VTEC forecast in this dissertation.

The main objective of using rolling window statistics in this dissertation is to create VTEC features from a given data sample by defining a range, i.e., a rolling window of observations, that includes the sample itself and some specified number of prior samples. One of the most popular rolling statistics is the moving average, which uses a rolling time window to calculate the average of that window as the current value. The EMA is a type of weighted moving average that gives more weighting or importance to recent data as it is considered more relevant than older data. Since new data has a larger weight, the EMA reacts more quickly to data changes than the simple moving average. The EMA of the VTEC can be defined by assigning an exponentially increasing weight to each term in the moving average window as

$$EMA(VTEC)_t = \frac{VTEC_t + (1-w)VTEC_{t-1} + (1-w)^2 VTEC_{t-2} + ... + (1-w)^t VTEC_0}{1 + (1-w) + (1-w)^2 + ... + (1-w)^t}, \tag{4.1}$$

where t is the window size and $w = \frac{2}{t+1}$ is a smoothing factor. The moving averages of the lagged VTEC features are used to detect the overall trend of the VTEC time series data. Thus, the data are shifted by 4 days and 30 days, and the EMAs of these lags are obtained.

In the high-resolution VTEC models in Sections 5.4 and 5.5, lag features of VTEC and the SYM/H index from the previous seven days are used as input features instead of aggregated values. Adding lag features can be referred to as a sliding window method. This dissertation uses a sliding window method with a fixed window of 7 days of the previous timesteps.

The first and second VTEC derivatives are calculated as new features in **P-II**, **P-III**, **CP-I**, and **CP-II**. The first derivative represents the rate of change of a variable, i.e., the velocity, while the second derivative indicates how the rate of change/velocity of the variable changes, i.e., the acceleration.

In **P-I**, the regional ionosphere Taylor series coefficients of the RIM IONOWB are used as input features for the ANN-based model called RIM IONOsphere model for the Western Balkans with Artificial Intelligence (IONOWB_AI), along latitude and longitude, HoD, F10.7, Kp and Dst features; Table 3.1.

Feature engineering also includes the process of feature scaling, such as standardization, if required. The input data for an ANN in **P-I** and **P-III** (Sections 3.2.5 and 3.3.4) are standardized to obtain data with a zero mean and a standard deviation equal to one. That is a standard procedure in learning algorithms sensitive to the scaling of input features, such as ANN. Feature scaling ensures that all input variables are treated equally, even with different scales. It also allows faster convergence of the gradient descent method. Feature standardization is defined as

$$xs_{p,i} = \frac{x_{p,i} - \bar{x}_p}{\sigma_p} \tag{4.2}$$

where $x_{p,i}$ is the input value at timestep i of the p^{th} feature with $p = 1, 2, ..., P$, $\bar{x}_p$ and σ_p are the mean value and the standard deviation of the p^{th} feature, respectively, $xs_{p,i}$ is the resulting standardized input value at timestep i of the p^{th} feature. Scaling is performed independently for each feature. The mean values and standard deviations calculated for the training data are stored to apply them to validation and testing data to perform the consistent transformation. On the other hand, the learning algorithms based on decision trees, presented in Sections 3.2.1 to 3.2.3, do not require data normalization because they are not sensitive to the scaling of the input data due to the completely different learning methodology in which the decision tree is built on the partitioning of the data according to the most influential features, see Section 3.2.1. So the data are not standardized in these cases. Furthermore, sine and cosine transformations of time information, such as HoD and DoY, are performed in ANN to preserve their cyclic meaning, as

$$HOD_{sin} = \sin\left(\frac{2\pi \cdot HOD}{24}\right), \quad HOD_{cos} = \cos\left(\frac{2\pi \cdot HOD}{24}\right)$$

$$DOY_{sin} = \sin\left(\frac{2\pi \cdot DOY}{365.25}\right), \quad DOY_{cos} = \cos\left(\frac{2\pi \cdot DOY}{365.25}\right). \tag{4.3}$$

In **P-II**, **P-III**, and **CP-II**, two main approaches are undertaken to prepare the data sets:

1. After preprocessing, the data $(\mathbf{x}_i, y_i)$ for $i = 1, 2, \ldots, N$ are used in the learning algorithm. This data set is referred to as non-differenced data.
2. The data, except HoD and DoY, are time-differenced $\Delta\mathbf{x}_i, \Delta y_i$ by calculating the difference between an observation at time step $t_i + 24$ and observation at time step t_i, i.e., $\Delta\mathbf{x}_i = \mathbf{x}_{i+24} - \mathbf{x}_i$ and $\Delta y_i = y_{i+24} - y_i$. The EMA and time derivatives are then calculated from the differenced VTEC values. In the end, the forecasted VTEC differences are reconstructed by adding up the VTEC values of the previous day. This data set is referred to as non-differenced data.

The idea behind creating daily differences is to remove the dominant daily VTEC variations, leaving the learning algorithm with only the remaining signatures associated with other sources of VTEC fluctuations to learn from. Also, differencing reduces time dependence and trends and stabilizes the mean of the data set.

4.1.4 Exploratory Data Analysis

Exploratory data analysis is performed to identify significant relationships, patterns, and correlated data and summarize their characteristics to support the selection of input data for the ML-based VTEC model. It is crucial to prepare an appropriate data set for the task of VTEC modeling and forecasting. The goal of the exploratory data analysis is to create training data with enough relevant input features, not too many irrelevant and not too many correlated input features.

These properties are checked with the correlation matrix between the input features and the output in Figure 4 in **P-I** and Figure 6 in **P-II**. The correlation matrix for strong and severe geomagnetic storms with Kp≥ 7 show a weak to moderate relationship between VTEC and all input observations, i.e., the relationship with the solar wind speed, Bz, Kp, AE data is significantly enhanced. These relationships are not visible in the correlation matrices over the entire period of January 2015 - December 2016 because these events are rare and unrepresented. However, the correlations become apparent during a Space Weather (SW) event. A scatter plot between two variables is used to analyze linear and nonlinear relationships, as shown in Figures 4.4 and 4.5. The histogram is also used to visually represent the data distribution, as in Figure 4.5. The correlations between the daily VTEC of the EPN SRJV station $(43.87°, 18.41°)$ and the solar indices show a moderate positive correlation (Figure 4.5), which is slightly higher for F10.7 than for S$_n$.

The so-called Forward Feature Selection method is another method for selecting input features. This is an iterative method used in **P-I**, where we started with basic features of regional ionospheric

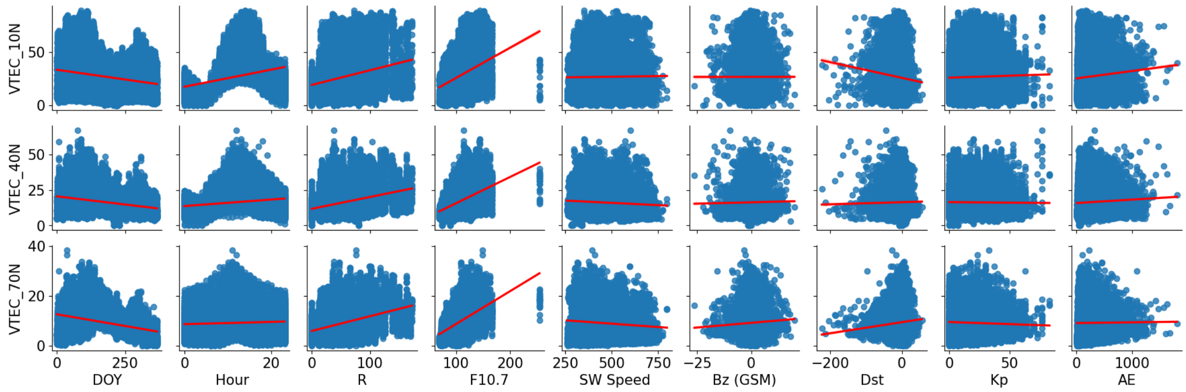


Figure 4.4: Scatter plots between the 1-hour VTEC and the observations of solar activity (daily data samples), IMF and GMF activity (1-hour and 3-hour data samples), as well as, datetime features of Hour and DoY for the period January 2015 - December 2016. The red line represents the linear regression line.

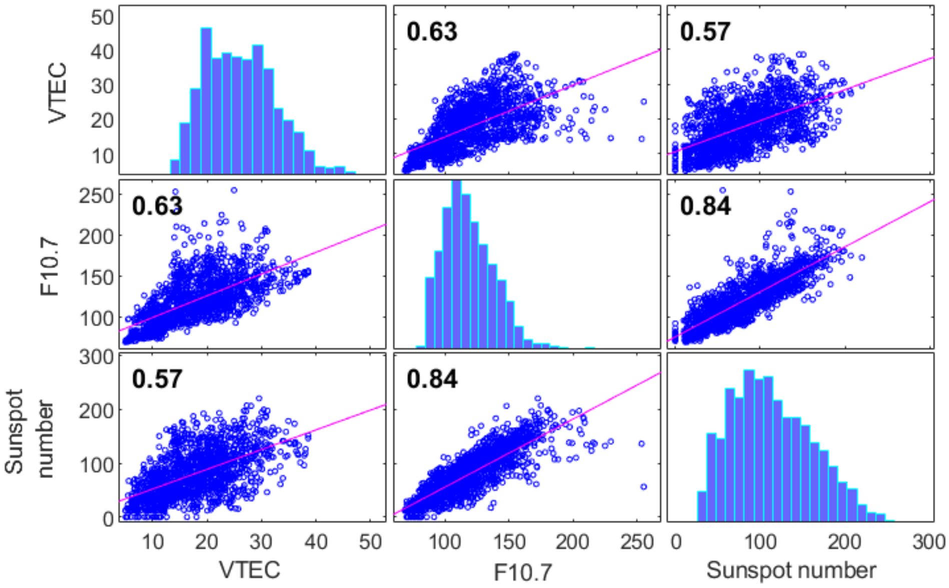


Figure 4.5: Scatter plots and histograms of daily VTEC in TECU, F10.7 solar flux in sfu, and sunspot number S_n from January 2013 to December 2016. The pink line represents the linear regression line, (taken from **CP-III**).

coefficients, and added a new feature at each iteration. The new feature is kept only if it improves the model's accuracy. The results of this method are presented in Figure 5.12.

Another method for selecting input features could be the feature importance analysis explained in Section 3.4. This procedure can estimate essential features for ANN-based models. However, it is not mandatory for Decision Tree-based models, as automatic feature selection already occurs when growing a decision tree. More precisely, it is done internally when training a decision tree. Therefore, feature importance analysis is primarily used at the interpretation level in this dissertation to estimate the contribution of each input feature to the model results.

4.2 Data Partitioning

To develop VTEC models, the data is divided into three sets:

1. Training set: for training the model and optimizing its parameters;
2. Cross-validation set: for measuring the model performance and optimizing its hyperparameters;
3. Test set: holdout set used at the end to estimate the performance error on new data, called the generalization error.

The model is fitted to the training set, and its performance is evaluated against the validation set to select the model complexity. The cross-validation set is used repeatedly to tune the hyperparameters. A popular solution is a K-fold cross-validation in Figure 1.6, where the training data is split into k folds, with part of the folds used for training and part for cross-validation. In this dissertation, K-fold cross-validation is implemented on a rolling basis to be suitable for time series data, as explained in Section 4.2.2. The final performance of the model is evaluated on the test set, which was not used in either training or validation of the model. Calculating the error on a test set approximates the expected value for future or operational data. The test set is, therefore, used only once to evaluate the final performance of the model and not to train or tune the model.

4.2.1 Selection of Time Periods

Figure 4.6 (top left) shows that 85% of the 3-hour Kp data from 2009 to 2019 indicate quiet conditions in the geomagnetic field, while only 2% of the Kp data reached an index of 5 or higher. This means that geomagnetic storms are largely underrepresented and can be considered as minority examples since SW events occur relatively rarely compared to quiet periods, leading to a data imbalance. On the other hand, the SW examples are of particular interest because they contain useful knowledge, e.g., about irregular ionosphere variations, and are essential for forecasting purposes in the case of an early warning system. However, imbalanced data can lead to a model biased towards most cases (Krawczyk, 2016). Ensemble learning is a recognized method that can yield significant improvements in the case of a skewed distribution (Krawczyk, 2016), and boosting learning algorithms have been shown to be suitable for problems with imbalanced data (Esposito, 2020).

The number of geomagnetic storms ($Kp \geq 5$) is the highest in the years after the solar maximum, reached in April 2014, i.e., from 2015 to 2017, and in 2012, before the solar maximum (Figure 4.6, top right). The years 2015 and 2016 have more of these events than other years (Figure 4.6, bottom). They are also close to the solar maximum. Therefore, 2015 and 2016 are selected for training and cross-validation to have more examples of storm events and conditions near the solar maximum. The following year, 2017, is chosen for testing because the strongest storm of solar cycle 24 occurred in September 2017. The training data set has 99 days with reported $Kp \geq 5$, including 56 days in 2015 and 43 days in 2016. In the test data set, $Kp \geq 5$ applies to 37 days.

4.2.2 K-fold Time Series Cross-Validation

Since the observations are time-dependent, this dissertation proposes the technique of time series cross-validation on a rolling basis for VTEC forecasting in order to preserve the time dependence. Hence, the VTEC model is trained and cross-validated using many folds of data. For reliable performance

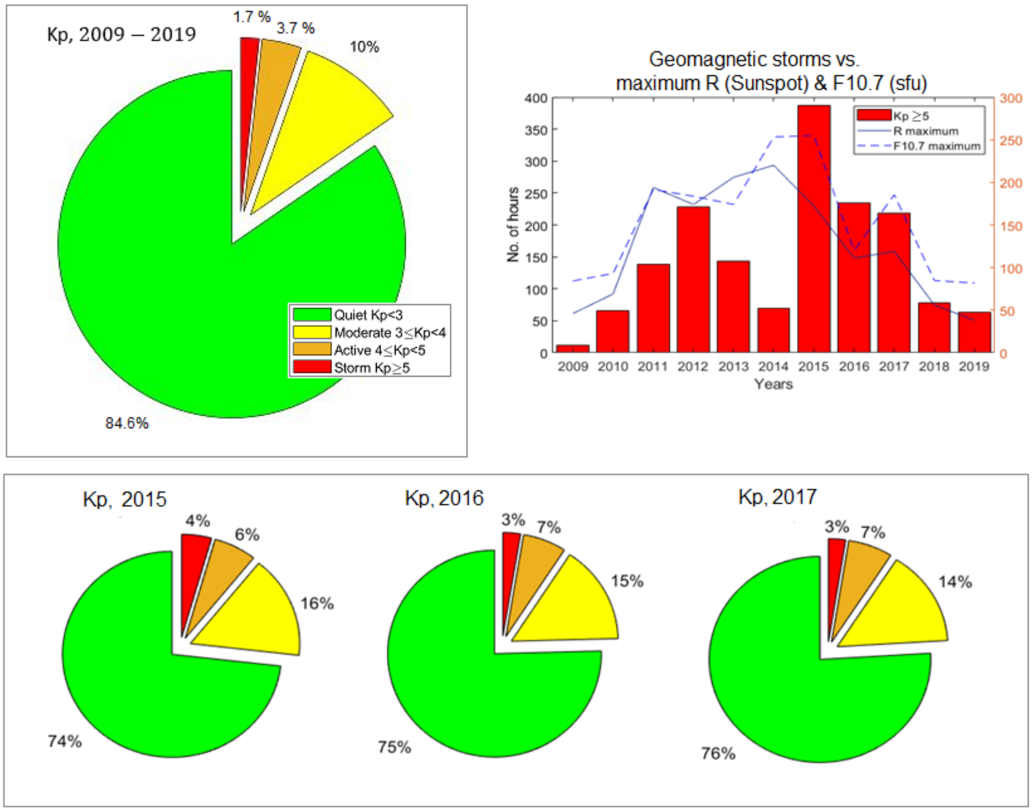


Figure 4.6: Percentage of Kp data with values $Kp < 3$, $3 \leq Kp < 4$, $4 \leq Kp < 5$ and $Kp \geq 5$ denoting quiet, moderate, active and storm conditions in the geomagnetic field, respectively, over solar cycle 24 (2009-2019) (top left) and for years 2015, 2016 and 2017 (bottom). Top right: Number of hours of Kp data with values $Kp \geq 5$ vs. maximum values of sunspot number R and solar flux F10.7 (both referenced to the right y-axis) from 2009 to 2019. (from **P-II**)

evaluation, it is recommended to adopt a large number of data folds (Wong & Yeh, 2020). The data is divided into two sets at each iteration: one set represents the training data, and the other the cross-validation data, as shown in Figure 4.7, both of which are defined in Section 4.2. The model is trained using the training set, i.e., the parameters of the model are optimized, while the hyperparameters are tuned using the cross-validation data. The training set in each fold consists only of observations made prior to the observations that form the cross-validation set. The cross-validation set from the previous iteration is included in the following training set, and subsequent data points are forecasted.

The effect of the k-fold size on the RMS of VTEC forecasting is analyzed for the Decision Tree and the Random Forest using the varying k-fold sizes: $k \in (6, 10, 20, 30, 40, 50)$, for the cross-validation and test sets. The results are presented in Figure 8 in **P-II**. The RMS for low-latitude VTEC forecast is lowest for $k = 20$, while the RMS for high and mid-latitude VTEC is similar for all values of k-folds. Therefore, the $k = 20$ folds is selected for a two-year cross-validation period, January 2015 - December 2016, in **P-II**, **P-III**, **CP-I** and **CP-II**. For a detailed analysis of the RMS on training and cross-validation data sets for each k-fold within 20 folds, see Figures 8 and 9 in **P-II**. Figure 4.8 illustrates the mid-latitude VTEC data with training and cross-validation sets for the first and last fold.

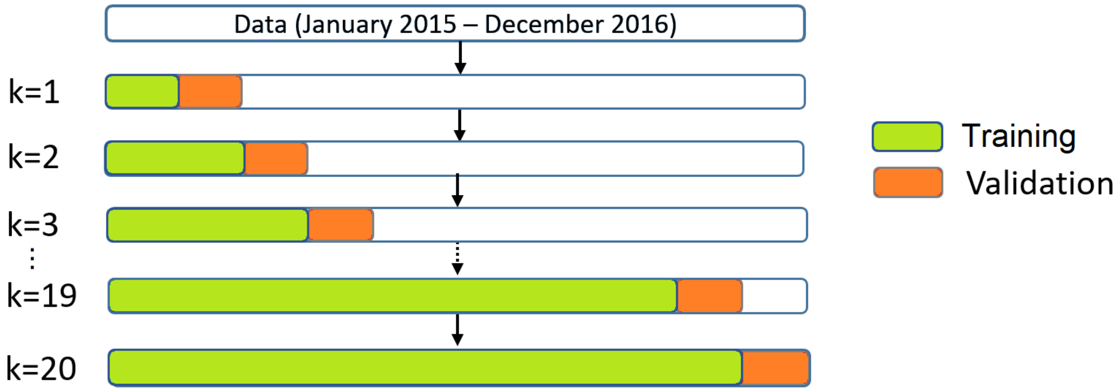


Figure 4.7: Model performance evaluation by time series cross-validation with $k = 20$ folds to prevent overfitting and evaluate model performance. The final metric is calculated as the average RMS of all $k = 20$ folds (taken from **P-II**).

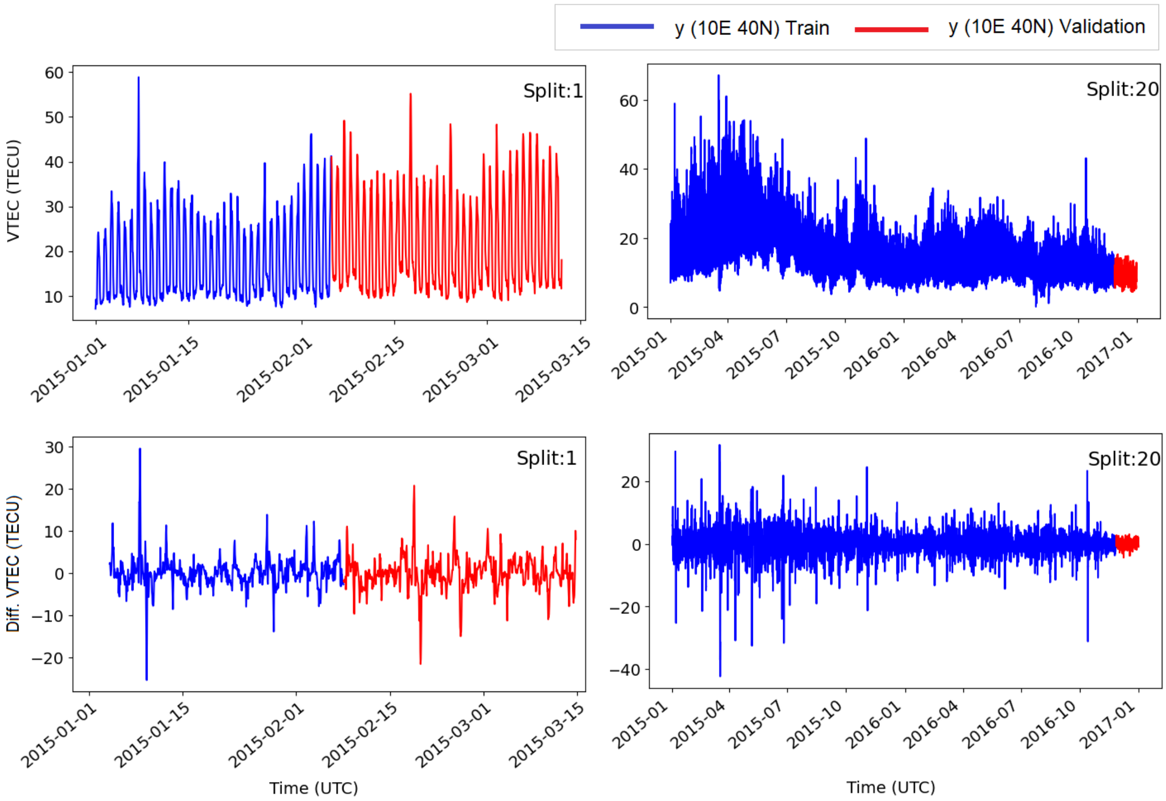


Figure 4.8: Time series cross-validation of VTEC at 10° 40° with the training data set in blue and the cross-validation data set in red for the 1^{st} fold and the 20^{th} fold. Top: non-differenced VTEC, bottom: daily differenced VTEC.

4.3 Model Hyperparameters

Parameters, such as the splitting variable and the value of the splitting point in a decision tree or the weights in an ANN, are estimated from the training data using an optimization procedure, as already explained in Sections 3.2.1, 3.2.3 and 3.2.5. However, every learning algorithm has certain parameters, known as hyperparameters, which cannot be estimated from the data but must be tuned for a particular modeling problem, here VTEC estimation and forecasting. The hyperparameters determine the model architecture and control the model complexity, see Table 4.1. Their values depend on the data and the problem. However, the space of hyperparameters cannot be fully explored. Usually, the values of hyperparameters are found by trying different combinations within a specific range of values and evaluating the performance of each model in a way to balance bias and variance, as described in Sections 4.3.1 and 4.3.2. However, the residual sum of squares of the training data cannot be used to determine their values, as this would reduce the ability of a model to generalize to new data. Therefore, the cross-validation data set is introduced in Section 4.2, and K-fold time series cross-validation in Section 4.2.2.

Table 4.1: Hyperparameters tuned

	Hyperparameters
ANN	Number of hidden layers, Number of hidden neurons, Learning rate
Decision Tree	Maximum tree depth, Minimum number of samples in split and leaf nodes
Random Forest	Maximum tree depth, Maximum number of features, Number of trees
AdaBoost	Maximum tree depth, Number of trees
XGBoost/GBoost	Maximum tree depth, Number of trees, Learning rate

4.3.1 Bias-Variance Tradeoff

The generalization error can be decomposed into a bias and variance component. The bias term is the squared difference between the average of the estimate and the true mean, while the variance is the expected squared deviation of an average estimate around its mean. Model selection involves a particular choice of hyperparameters, which have an enormous impact on the accuracy, complexity, and computational cost of the model. Thus, it comes down to a trade-off between the complexity of the model to capture higher-order nonlinear functions and its ability to generalize to previously unseen data, which is called a bias-variance trade-off.

As can be seen in Figure 4.9, the training, the validation, and the test errors decrease with increasing model complexity up to a certain point. However, in many cases, the training error continues to decrease beyond that point, but the validation and test error increases. This is the point at which the model begins to overfit the training data due to its excessive complexity and high variance. Overfitting is a common problem in ML when a model matches a particular data set too closely or exactly to the extent that negatively affects the model performance on new data and reduces its generalization ability. A similar situation can occur for a fixed model complexity when training time is increased, especially in the case of an ANN that can iterate over the training data many times. If a model is trained longer with the same data, the learning algorithm will adapt better to the data (see Figure 3.8). In this case, it is advisable to stop training before overfitting becomes a problem, i.e., before the validation error increases, resulting in a model with an appropriate complexity. In the case of a high bias, the model is not complex enough with its approximation function and, therefore, underfits the data.

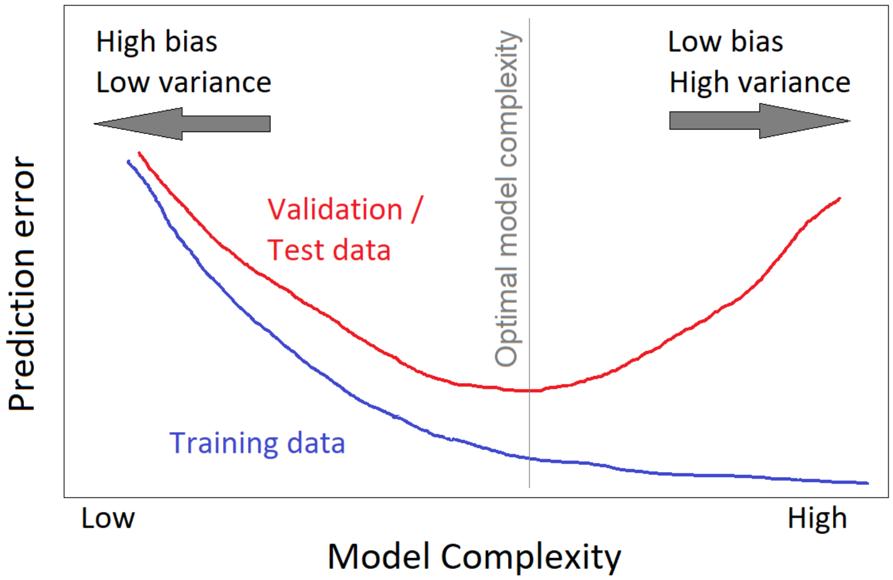


Figure 4.9: Training and validation/test errors as a function of model complexity. An optimal model complexity yields a balanced model with neither a large bias, underfitting the data, nor a large variance, overfitting the data (taken from **P-II**).

Figure 4.10 depicts a detailed flowchart of the ML-based VTEC model development in this dissertation. Poor performance on the training and cross-validation data results from high bias, while poor performance on the cross-validation data only is often the result of high variance. These issues are addressed by increasing or decreasing the model complexity by adjusting the hyperparameters to select the appropriate model architecture.

To overcome the problem of overfitting, neural networks often require large amounts of data to train them properly and to find the parameters. Another method to solve the problem of overfitting is to add some regularization or early-stopping procedure to stop training as soon as the model performance in a validation set no longer improves (applied in **P-I** and **P-III**). On the other hand, the randomness used in Random Forest tree construction in **P-II**, such as randomly selecting input features or combinations of input features at each node to grow a tree, as explained in Section 3.2.2, reduces the problem of overfitting. The high bias in decision tree-based is addressed in this dissertation by adding new input features, increasing the size of a tree by increasing the maximum depth of the tree and decreasing the minimum number of samples required to split a node in the tree, increasing the number of features considered in the search for the best node split and increasing the number of trees. The high variance in decision tree-based VTEC models is addressed by increasing the size of the data set, decreasing the tree size, reducing the number of features to be considered when splitting a node in a tree, decreasing the number of trees and decreasing the value of the learning rate.

The training data size also plays a vital role in model performance. The more data available for training, the better the generalization ability of a model. If we use a small training data set, the model tends to have a low training error. However, the test error will be very high because it has not learned much about the variations in the data distribution to make predictions for cross-validation or test instances. On the other hand, as more training data are used, the training error increases. This is because the variance in the training data increases as new data examples are added, while the model complexity remains the same. However, the test error decreases because the model can now generalize better.

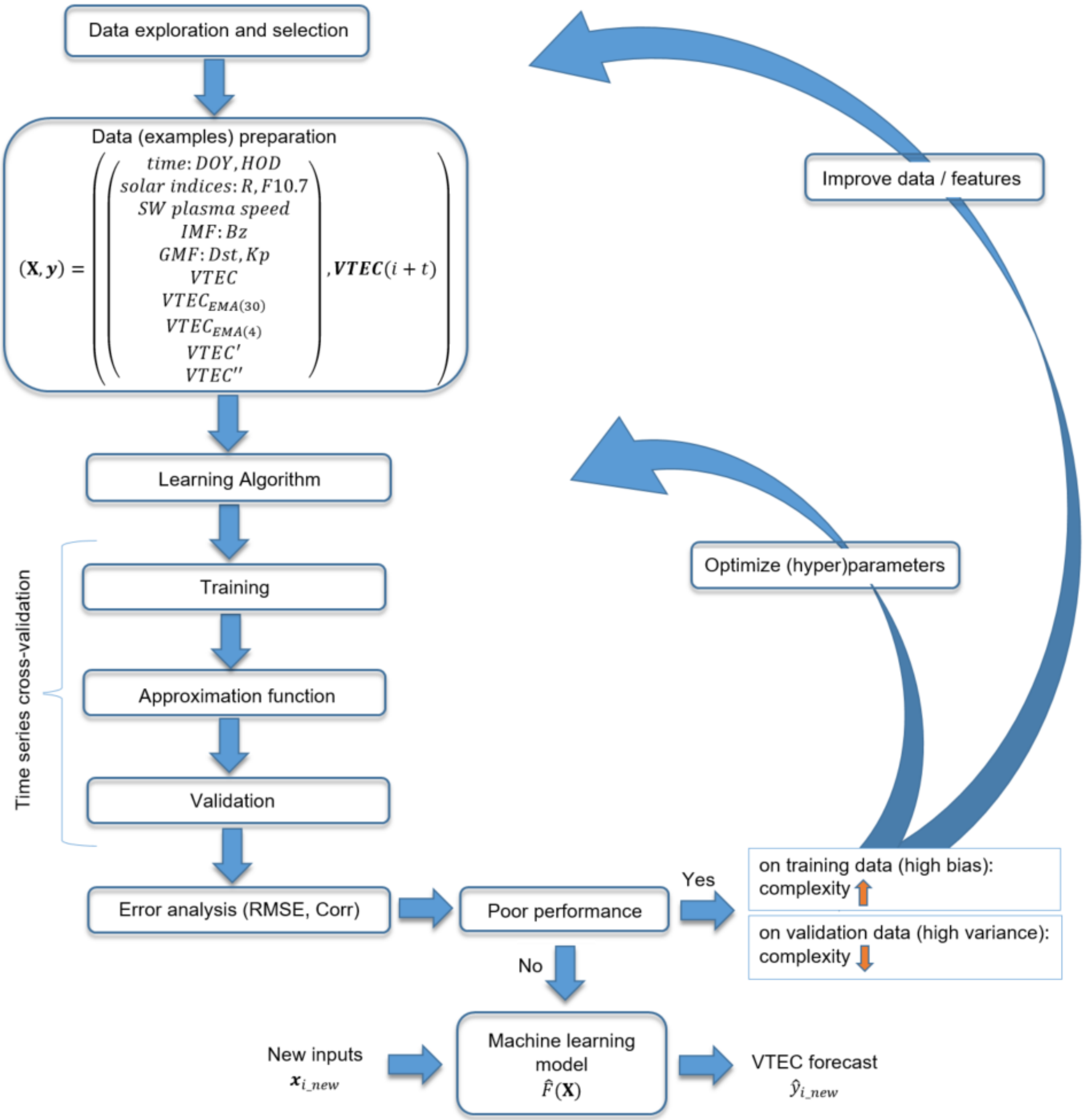

Data exploration and selection
Data (examples) preparation
$(\mathbf{X}, \boldsymbol{y}) = \left(\left(\begin{array}{c} time: DOY, HOD \\ solar\ indices: R, F10.7 \\ SW\ plasma\ speed \\ IMF: Bz \\ GMF: Dst, Kp \\ VTEC \\ VTEC_{EMA(30)} \\ VTEC_{EMA(4)} \\ VTEC' \\ VTEC'' \end{array}\right), \boldsymbol{VTEC}(i+t)\right)$
Learning Algorithm
Time series cross-validation
Training
Approximation function
Validation
Error analysis (RMSE, Corr)
Poor performance
Yes
No
on training data (high bias): complexity ↑
on validation data (high variance): complexity ↓
Improve data / features
Optimize (hyper)parameters
New inputs
$\boldsymbol{x}_{i\_new}$
Machine learning model
$\hat{F}(\mathbf{X})$
VTEC forecast
$\hat{y}_{i\_new}$

Figure 4.10: Flowchart of ML-based VTEC model development from data exploration, selection, and preparation through training and cross-validation to the resulting model with the targeted approximation function. The model is optimized with respect to its performance. The optimized VTEC model is used to forecast the VTEC for new input data, (taken from **P-II**).

The task is to find a balanced model that neither learns from the noise, i.e., overfits the data, nor makes too poor assumptions about the data, i.e., underfits the data. The final model complexity is chosen in a way to trade off the bias with the variance in order to minimize the cross-validation error and, consequently, the test error.

4.3.2 Model Architecture Selection

As already mentioned, the choice of hyperparameters is a special case of a more general problem known as model architecture selection, where we have to choose between models with different levels of complexity. For example, the size of a tree controls the complexity of the decision tree-based model, while the number of hidden layers and hidden neurons controls the complexity of the ANN-based model. The model complexity must be selected to balance bias with variance, as explained in the previous subsection 4.3.1.

To reduce the problem of overfitting and improve accuracy, when building an ML-based VTEC model, randomness is considered in the tree construction. So the size of the maximum number $v = 6$ of features of the random subsets of input variables is considered when splitting a node of the tree. The smaller the value of v, the correlations between any pair of trees in the Random Forest are reduced, and also the overfitting. However, if there are only a few relevant input features among many, v should be set to a high value to allow the algorithm to find the relevant variables. For XGBoost, a smaller value of the learning rate ν results in a lower test error but requires a more significant number m of iterations (Friedman, 2001). In addition, the data for each boosting procedure in this dissertation is subsampled for each tree to further prevent overfitting. In an ANN, having more hidden units than too few is better. With too few hidden units, the model may not be flexible enough to capture the nonlinearities in the data. On the other hand, if there are many hidden units, the excess weights can be shrunk towards zero with a suitable regularization technique. The use of multiple hidden layers in ANN in **P-I** allows the construction of hierarchical hidden features at different levels of resolution. Algorithms based on decision trees are relatively robust to the settings of the hyperparameters, so small changes generally lead to similar results, and the search for the appropriate parameters and hyperparameters does not take as much time as with an ANN.

To select the hyperparameters, the performance on the training data is plotted together with the performance on the cross-validation data as a function of model complexity, as shown in Figure 4.11 for the developed QGB VTEC model. The performance on the validation set serves as a proxy for the performance of the test data. The training of ML-based VTEC models is stopped at the right point of complexity, which is indicated by the purple dotted line and is referred to as "selected". The line represents the hyperparameters selected that minimize the RMS on the cross-validation data. Selected hyperparameters and the range of values used to search values for hyperparameters are provided in Table 3 of **P-I**, Table 2 of **P-II**, Table 2 and Table S1 of **P-III**.

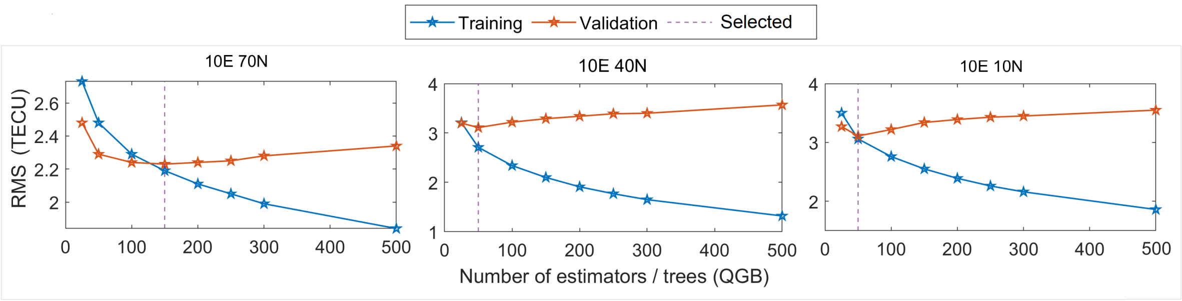


Figure 4.11: Learning curves: training and cross-validation RMS values against different values of the hyperparameter number of trees of the QGB VTEC model for 10°E 70°N (left), 10°E 40°N (mid), and 10°E 10°N (right).

4.4 Developed Models

The developed ML-based models for VTEC within the scope of this dissertation are:

1. ANN-MLP Model for Regional VTEC Modeling (**P-I**)
2. Decision Tree (DT) Model for VTEC Forecasting (**P-II**, **CP-I**)
3. Random Forest (RF) Model for VTEC Forecasting (**P-II**, **CP-I**)
4. XGBoost Model for VTEC Forecasting (**P-II**, **CP-I**)
5. Model for VTEC Forecasting (**P-II**, **CP-I**)
6. Voting Regressor (VR) Model for VTEC Forecasting (**P-II**, **CP-I**)
7. Super-Ensemble (SE) Model for Probabilistic VTEC Forecasting (**P-III**, **CP-II**)
8. QGB Model for Probabilistic VTEC Forecasting (**P-III**)
9. BNN Model for Probabilistic VTEC Forecasting (**P-III**)
10. BNN Model with Data Uncertainties for Probabilistic VTEC Forecasting (**P-III**)

4.5 Computational Efficiency

The training and validation time with 20 folds in the K-fold cross-validation for the single Decision Tree model amounts to less than 5 seconds, while for the ensemble learning models, it increases from 30 seconds for XGB to about 5 minutes for Random Forest (Table 4.2). The VR approach consists of the VR1 (Random Forest, AdaBoost, and XGB) and VR2 (Random Forest and XGB) models, see Section 3.2.4 and Table 3 in **P-II**, with a training time less than 5 minutes. The testing time for each model is less than one second, demonstrating the proposed models computational efficiency in **P-II**.

On the other hand, the two BNN-based approaches are the most computationally intensive. A single training iteration with two years of data takes about an hour or more, which can be considered a drawback of the BNN method. The most computationally efficient model in **P-III** is QGB, which takes only 1 minute for a single training iteration with two years of data. When the models are trained and optimized, the execution using test data is fast and takes 1 second for one year of data.

Since the high-resolution VTEC models in Section 5.4 and 5.5 contain a larger amount of training data, i.e., 15 minutes of data sampling instead of 1 hour and 4 years of data instead of 2 years, the computational cost of training a model consequently increases. Therefore, the QGB approach was selected, which is fast to train and tune as it has low computational costs. QGB VTEC models trained with data from January 2013 to December 2016 are marked by "II", while the remaining models were trained with data from January 2015 to December 2016. "-7d" means that the input features include the lagged history of input observations over seven days prior to timestamp *i* (inclusive), as explained in Section 4.1.3. When the seven-day history of input observations is taken as input features, the training is extended by about 2 to 3 hours compared to other QGB VTEC models with input features at time *i*. When the training data is extended from two to four years, the training time increases by 100 minutes for the "-7d" models. However, execution on test data is only extended by 1 sec.

4.6 Model Interpretation

Relative feature importance analysis based on Decision Tree, explained in Section 3.4, is used to estimate the contribution of each input feature to the ML-based VTEC model result in order to

Table 4.2: Computational cost for a single training and a test run in seconds for a single VTEC grid point on the NVIDIA Tesla P100 GPU with 16 GB memory.

Machine learning-based VTEC model	Training and Validation (sec)	Testing (sec)	Publication / Dissertation Section
DT	2 - 4	<0.01	**P-II**, **CP-I**, 5.2
RF	300 - 330	~ 0.30	**P-II**, **CP-I**, 5.2
AdaBoost	65 - 85	~ 0.10	**P-II**, **CP-I**, 5.2
XGBoost	30 - 40	~ 0.05	**P-II**, 5.2
VR1	~ 250	~ 0.35	**P-II**, **CP-I**, 5.2
VR2	~ 200	~ 0.25	**P-II**, **CP-I**, 5.2
SE	~1280 (~20 min)	1.35	**P-III**, **CP-II**, 5.3
QGB	~50-80 (~1 min)	<0.1	**P-III**, 5.3
BNN	~1920-3650 (~30-60 min)	1.37	**P-III**, 5.3
BNN-2	~5900 (~100 min)	1.40	**P-III**, 5.3
QGB,	~ 50	0.10	5.4
QGB, II	~ 300 (~5 min)	0.16	5.4
QGB, -7d	~ 5730 (~96 min)	1.30	5.4
QGB, -7d II	~ 12340 (~205 min/~3,4 h)	1.23	5.4

interpret what the ML-based VTEC model has learned and which input features have been selected as relevant[6]. This information helps to understand the underlying relationship between the input features and the model output. More detailed analysis can be done by, for example, examining specific events such as SW events and geomagnetic storms. Because the data reflect geophysical processes and relationships, such findings can reveal the underlying physical processes and show whether the results are consistent with our physical understanding, and reveal new relationships and patterns. The feature importance for data samples of geomagnetic storms $Kp \geq 5$ are explored in Figure 11 in **P-II**. The feature importance has also been assessed for the whole data set in Figure 6 in **P-I**, Figure 10 in **P-II**, Figure 8 in **P-III**, and Figure 5 **CP-II**.

During geomagnetic storms, the relative importance of the input features describing the solar activity, solar wind, and magnetic activity generally increases, while the contribution of the input features describing previous VTEC values decreases (see Figures 10 and 11 of **P-II**). On the other hand, the relative importance of the input features of solar activity, solar wind, and magnetic activity is mostly higher for differenced data sets, proving that removing the daily regular VTEC variations and training on the remaining ionospheric background information improves the learning of SW-related features (see Figures 10 and 11 in **P-II**). Of particular interest is the higher importance of the AE index for high-latitude VTEC, the Kp index for mid-latitude VTEC, and the Dst index for low-latitude VTEC in the differenced data set, considering that these indices are measured in these latitudinal regions.

The relative importance of the input features for the probabilistic 24-hour VTEC forecast by the QGB model is estimated for the upper confidence bound (top), median VTEC (mid), and lower confidence bound (bottom) in Figure 4.12. For the median VTEC, the most important input feature is the lagged VTEC at time step t_i for forecasting VTEC at time step t_{i+24h}. The other input features have much smaller contributions. This is due to the prevailing diurnal VTEC variations, where day-to-day VTEC

[6] The feature importance analysis can also be used in exploratory data analysis in Section 4.1.4 to select the important features before training. However, this is not necessary for a Decision Tree, as the significant features are selected automatically, as explained in Section 3.2.1.

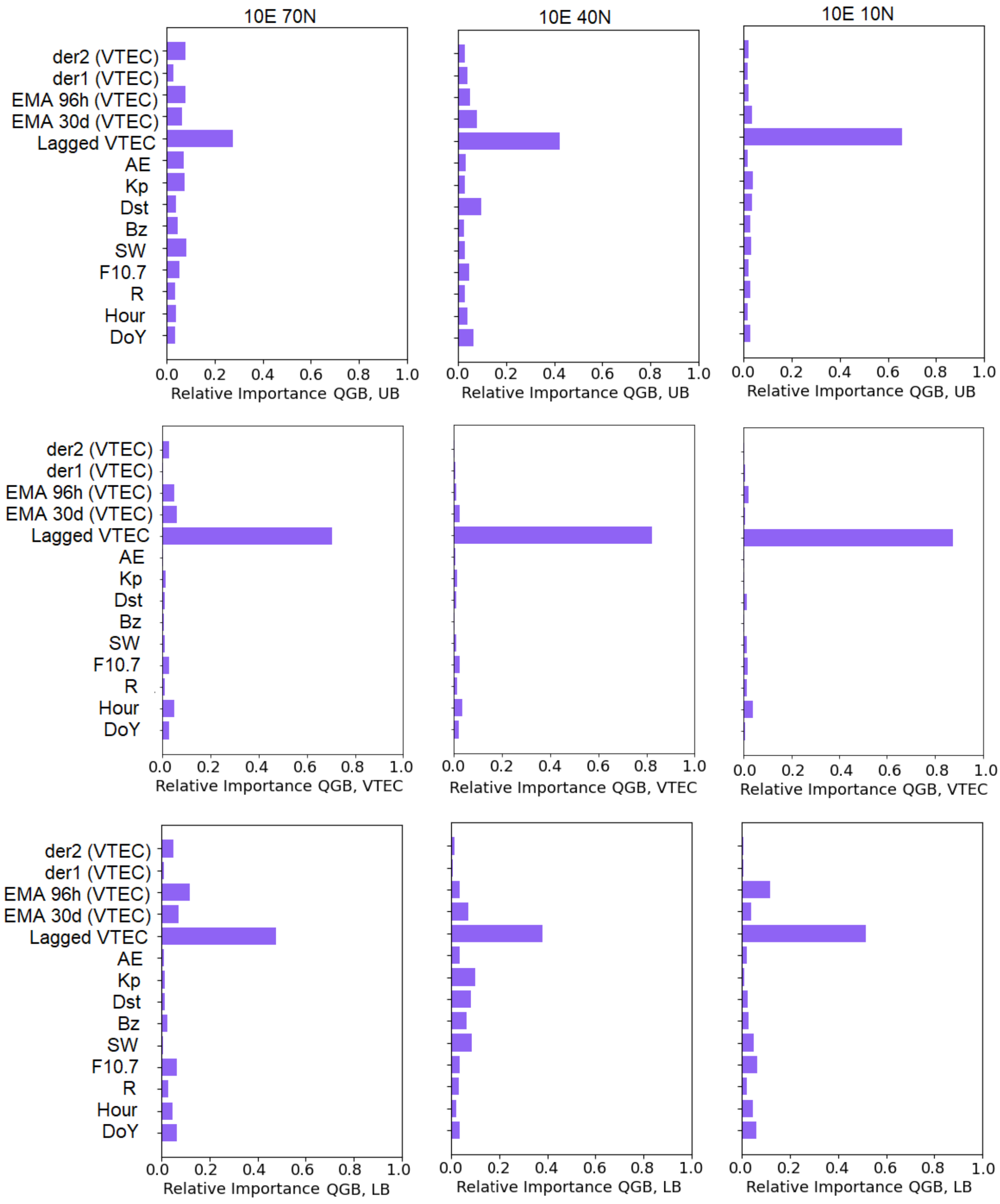


Figure 4.12: Relative importance of input features for probabilistic 24-hour QGB VTEC forecast. Top: upper bound (UB), mid: median VTEC forecast, bottom: lower bound (LB). Left column: 10°70°, mid column: 10°40°, right column: 10°10°.

usually does not change much during quiet conditions. On the other hand, other input features have much larger importance in estimating the lower and upper confidence bounds, such as the AE, Kp, Dst, and solar wind (SW) indices. Here, the objective function minimizes the positive and negative residuals between the ground truth and the model output for the upper and lower bounds, respectively; see Equation 3.33. These residuals are more strongly influenced by solar and geomagnetic activity than the median VTEC. Thus, the lagged VTEC contributes 20% to 50% less to the confidence bounds estimate than to the median VTEC estimate, while the space weather input features increase their contribution. These results suggest that the confidence intervals are determined by the space weather features in addition to the VTEC-related features.

5 Results and Discussion

This chapter presents the following results of this dissertation and related discussion:

1. The main results of the ML-based regional ionosphere model of **P-I** and its application in single-frequency PPP in Section 5.1.
2. The analysis of the ensemble VTEC forecasting models from **P-II** concerning different levels of geomagnetic and solar activity, not included in **P-II**, in Section 5.2.
3. The main summary results of the probabilistic VTEC forecasting from **P-III** in Section 5.3.
4. High-resolution probabilistic forecasting for different forecast horizons in Section 5.4.
5. Forecast of the effects of an intense solar flare on the ionosphere in Section 5.5.
6. The influence of input features selection, data preparation, data resolution, data length, and learning algorithms on the results of the developed ML-based VTEC models in Section 5.6.

To the best of the author's knowledge, this dissertation is the first study to develop approaches to quantify uncertainties in ML-based ionosphere modeling and forecasting to provide reliable probabilistic results and confidence intervals. Therefore, this chapter includes a large portion of the uncertainty quantification results, including those not yet been published in journal papers **P-I**, **P-II**, **P-III** and conference papers **CP-I**, **CP-II**, **CP-III**. In this context, we include not only the main results of the various probabilistic approaches from **P-III**, but also the results that have not yet been published. For example, the studies listed above in items 4 and 5. They represent an improvement of the study **P-III** in terms of high-resolution data, high-resolution developed models, an extension to multiple forecast horizons, and a forecast of the effects of a solar flare on the ionosphere, produced during the author's research stays in ROB, Belgium, and UPC, Spain in 2022.

The analysis for the study in the first item above is conducted for the periods March 20–26, 2014, and March 15–20, 2015. The first period covers the solar maximum, peaked in April 2014, with increased solar activity but dominantly quiet geomagnetic conditions (see Figure 8 in **P-I**). The second period includes the St. Patrick's Day Storm, which occurred on March 17, 2015, with a Kp index value of 8. This event was followed by a recovery phase that began on March 18, 2015, and lasted several days until the geomagnetic field returned to normal conditions. The number of sunspots and the solar radio flux F10.7 were significantly lower in the second study period than in the first.

The analyses for the studies in items 3 through 6 are performed for the period January 1 - December 31, 2017, the period of space weather events from September 6 to 10, 2017, and the quiet period in terms of solar and geomagnetic activity from April 25 to 29, 2017. September 6 to 10, 2017, is one of the most intense solar activity periods, with the strongest solar flare of class X9.3, which peaked at 12:02 UTC on September 6. Earthward-directed CMEs were also emitted from the Sun on September 4 and 6. The first CME arrived at 23:43 UTC on September 6 and caused moderate geomagnetic conditions on September 7, while the second CME, from the solar flare X9.3, triggered a sudden storm at around 23 UT on September 7. This led to severe geomagnetic storms on September 8 with a maximum Kp of 8. The main phase of the storm was characterized by the two pronounced minima of the Dst and SYM/H indices at around 1 UTC and 14 UTC on September 8 (Figures 5.5, bottom right, and 5.9, bottom). Afterward, the recovery phase began, lasting about three days, i.e., until September 11.

5.1 Regional VTEC Modeling

The paper **P-I** deals with regional ionosphere modeling. In this section, the main results are presented and discussed, focusing on two developed RIMs: IONOWB and IONOWB_AI. The RIM IONOWB is based on the two-dimensional Taylor series expansion (2.10). The ionosphere information is estimated from the CORS and EPN observations as described in Section 4.1.1. The RIM IONOWB is the basis for the RIM IONOWB_AI model, which is developed using the ANN MLP architecture from Section 3.2.5. More specifically, the regional ionosphere Taylor series coefficients of the RIM IONOWB are input features of the RIM IONOWB_AI, see Table 3.1. The results are presented as regional VTEC maps. VTEC values from the RIM IONOWB_AI, the RIM OTHR[(1)], the GIM CODE, and the Klobuchar model are estimated on the $1° \times 1°$ grid in latitude and longitude from 40°N to 47°N and 13°E to 23°E. The CORS and EPN stations used to estimate RIM IONOWB are shown in Figure 4.2.

The VTEC maps for March 21, 2014, at 12 UTC, a day of high solar activity in the solar maximum phase with F10.7 = 153 sfu and quiet geomagnetic conditions with Kp = 2, are shown in Figure 5.1 in the upper block. The VTEC maps of the RIM IONOWB_AI, the RIM OTHR, and the GIM CODE show the lowest ionization from 46°N to 47°N and the highest ionization from 40°N to 42°N. The RIMs IONOWB_AI and OTHR differ by less than 3 TECU with the mean difference of 0.9 TECU. Their largest absolute difference of approximately 2 to 3 TECU is in the region from 43°N 21°E to 44°N 23°E. Their smallest differences, less than 1 TECU, are in the areas covering Slovenia, Croatia, Bosnia-Herzegovina (BH) (except around 44° latitude), and Montenegro. On the other hand, the absolute differences between the IONOWB_AI and the GIM CODE are up to 4 TECU, with a mean absolute difference of 1.5 TECU. The most significant differences are in the southern part of the map, where only a few IGS and EPN stations are used to determine the GIM. In contrast, in the areas of the western Balkans where no GNSS stations are used to estimate the RIM IONOWB_AI, Serbia, Kosovo and Montenegro, there are minor differences with a mean of 1 TECU and 1.1 TECU from the RIM OTHR and the GIM CODE, respectively.

The VTEC maps for March 17, 2015, at 12 UTC, the day of the severe geomagnetic storm with F10.7 = 139 sfu and Kp = 8, are shown in the lower block of Figure 5.1. During the main phase of the storm, the ionization increased over the western Balkans. The VTEC differences of the RIM IONOWB_AI compared to the RIM OTHR and the GIM CODE are up to 4.4 TECU in the northern part, especially in the areas at the northern and northeast borders of the study region, and mainly outside the western Balkans region. The mean differences for the areas of the western Balkans where no GNSS observations are used in the RIM IONOWB_AI are about 0.8 TECU and 1 TECU with respect to the RIM OTHR and the GIM CODE, respectively. The VTEC values for Slovenia, Croatia and BH, whose GNSS observations are used, also agree better with the RIM OTHR, with their mean difference of 0.5 TECU.

The GPS broadcasted Klobuchar model underestimates VTEC by more than 20 TECU for both days and cannot approximate the sudden VTEC increase during the storm on March 17, as shown in Figures 12 and 13 in **P-I**. These results reveal that the Klobuchar model deviates significantly from the newly developed RIM IONOWB_AI. On the other hand, the RIM IONOWB_AI correspond much more closely to the GIM CODE, and much better to the high-resolution RIM OTHR. These results suggest that the RIM IONOWB_AI could replace the Klobuchar model in single-frequency positioning.

To assess the RIM IONOWB and the RIM IONOWB_AI in single-frequency positioning, vertical and horizontal RMS position errors from 24-hour single-frequency PPP solutions are estimated according to the Equations (20a) and (20b) in **P-I** for selected stations highlighted in Figure 4.2 for March 2014 and March 2015, and are shown in Figure 5.2. Using the L1 frequency without ionospheric delay correction, the vertical position errors are about three to four times higher than the horizontal position errors. The vertical RMS position errors are about 5.5 m and 3.5 m in March 2014 and 2015, respectively.

[(1)] European RIM based on polynomial B-spline functions developed at DGFI-TUM as a two-step VTEC model; see (2.8), (2.9).

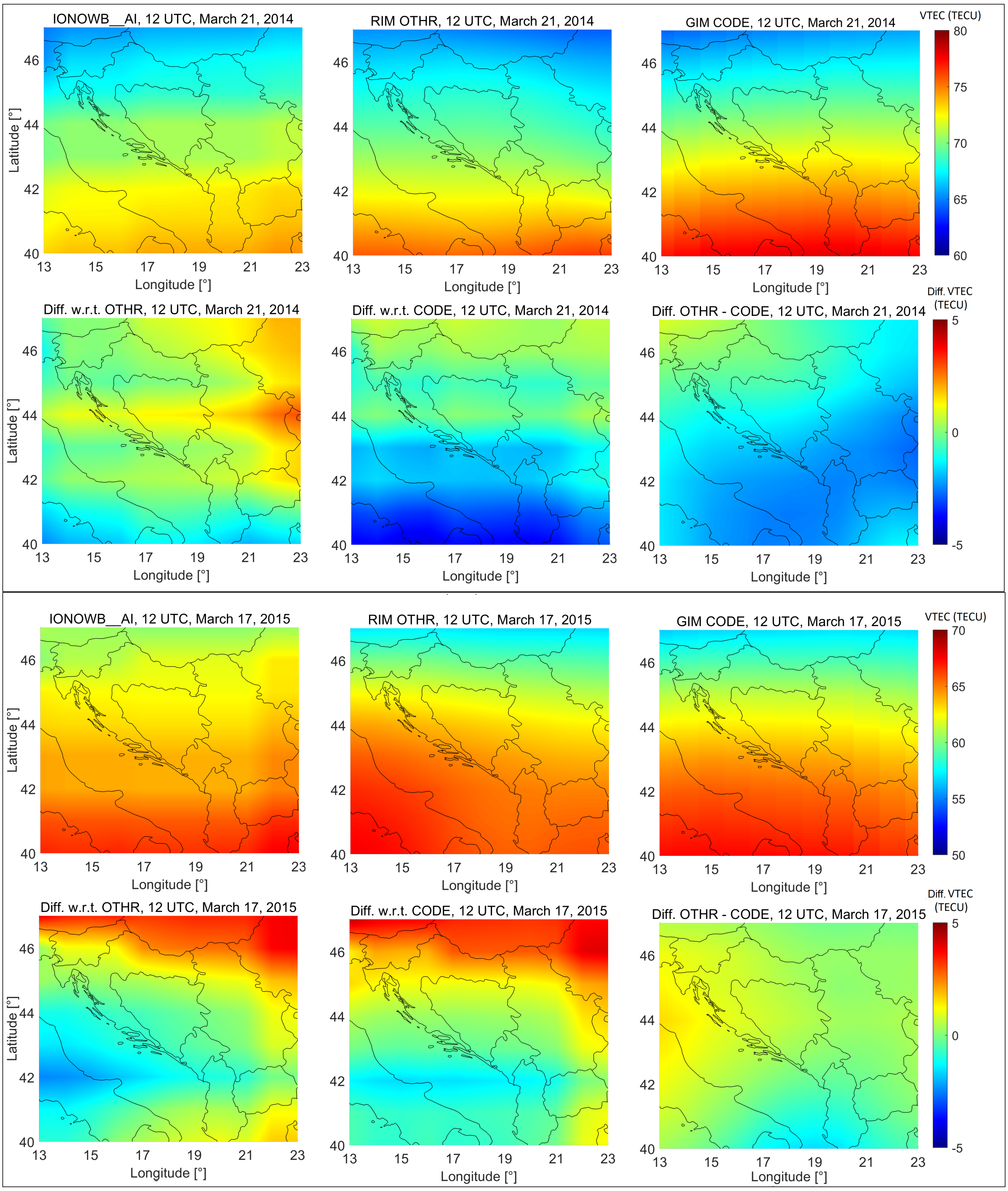

IONOWB__AI, 12 UTC, March 21, 2014
RIM OTHR, 12 UTC, March 21, 2014
GIM CODE, 12 UTC, March 21, 2014
VTEC (TECU)
80
75
70
65
60
Diff. w.r.t. OTHR, 12 UTC, March 21, 2014
Diff. w.r.t. CODE, 12 UTC, March 21, 2014
Diff. OTHR - CODE, 12 UTC, March 21, 2014
Diff. VTEC (TECU)
5
0
-5
IONOWB__AI, 12 UTC, March 17, 2015
RIM OTHR, 12 UTC, March 17, 2015
GIM CODE, 12 UTC, March 17, 2015
VTEC (TECU)
70
65
60
55
50
Diff. w.r.t. OTHR, 12 UTC, March 17, 2015
Diff. w.r.t. CODE, 12 UTC, March 17, 2015
Diff. OTHR - CODE, 12 UTC, March 17, 2015
Diff. VTEC (TECU)
5
0
-5
Latitude [°]
40
42
44
46
Longitude [°]
13
15
17
19
21
23

Figure 5.1: Top: RIM IONOWB_AI, RIM OTHR and GIM CODE. Bottom: $VTEC_{IONOWB\_AI} - VTEC_{OTHR}$, $VTEC_{IONOWB\_AI} - VTEC_{CODE}$, $VTEC_{OTHR} - VTEC_{CODE}$. All from left to right. Upper block: 12 UTC, March 21, 2014; lower block: 12 UTC, March 17, 2015, (adapted from **P-I**).

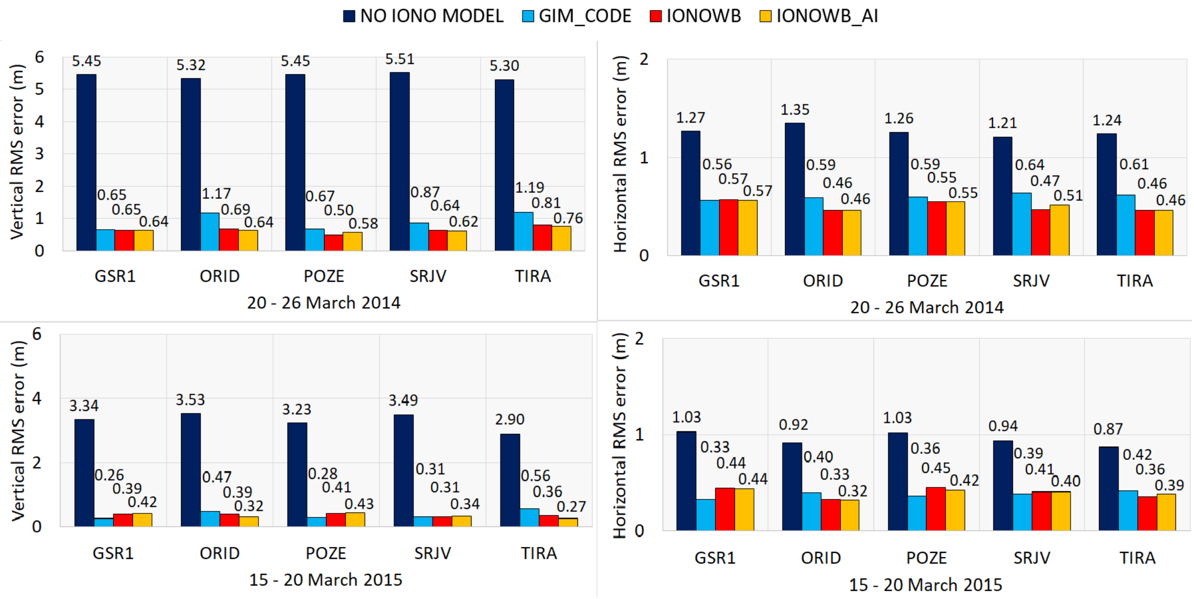


Figure 5.2: RMS errors of single-frequency positioning solutions without ionosphere corrections, with GIM CODE, RIM IONOWB and RIM IONOWB_AI. Shown are vertical position RMS errors on March 20–26, 2014 (top left), horizontal position RMS errors on March 20–26, 2014 (top right), vertical position RMS errors on March 15–20, 2015 (bottom left), and horizontal position RMS errors on March 15–20, 2015 (bottom right), (taken from **P-I**).

After applying the ionospheric delay corrections from the GIM CODE and the developed RIMs, the vertical accuracy is improved by 80% to 90% and the horizontal accuracy is improved by 50% to 60%.

In March 2014, after applying the IONOWB and IONOWB_AI ionospheric delay corrections, the position errors are at the same level as for the GIM CODE for station GSR1 and even better for all the other stations. A significant improvement in position accuracy with the RIMs IONOWB and IONOWB_AI is observed for the stations SRJV, ORID, and TIRA. Also, in March 2015, both vertical and horizontal position accuracy is improved for ORID and TIRA stations after applying the RIMs IONOWB and IONOWB_AI. These two stations are located in the lower part of the study region where more significant VTEC differences between IONOWB_AI and GIM CODE are observed in Figure 5.1. These differences may be attributed to the fact that information from very few stations is used to estimate the GIM CODE in this area, but this information is incorporated into the newly developed RIMs. Therefore, the RIMs IONOWB and IONOWB_AI outperform the GIM CODE for ORID and TIRA stations. However, better positioning results are obtained with the GIM CODE for GSR1 and POZE stations in March 2015, located in the upper part of the western Balkans. There, more considerable VTEC differences are observed between the RIM IONOWB_AI and the GIM CODE during the main storm phase, which is attributed to different VTEC perturbations within and outside the study region. Since a storm can cause significant deviations in the ionosphere, the RIM for Slovenia would most likely benefit from the inclusion of stations from other neighboring countries to improve positioning accuracy during a storm.

Figure 5.3 depicts the percent decrease in vertical and 3D RMS position errors, calculated from Equations (20a) and (20c) in **P-I** and averaged over all stations evaluated, for single-frequency PPP using ionospheric delay corrections from the developed RIMs and the GIM CODE, compared to the single-frequency PPP without ionospheric delay corrections. The newly developed RIMs decrease the vertical and 3D position error by about 5% compared to the GIM CODE during the solar maximum,

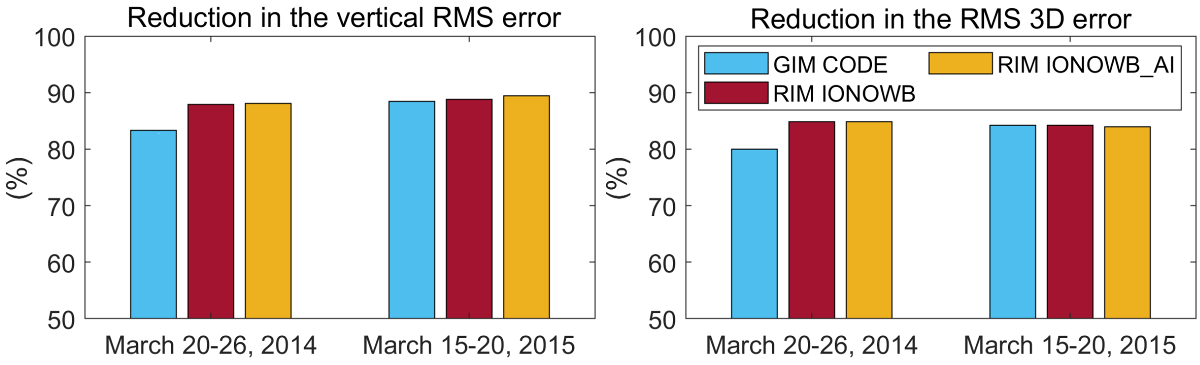


Figure 5.3: Percentage reduction of the vertical RMS (left) and the 3D RMS (right) errors of the single-frequency PPP with the GIM CODE, the RIM IONOWB and the RIM IONOWB_AI compared to no ionosphere correction, for March 20–26, 2014 and March 15–20, 2015.

i.e., in March 2014. More specifically, the vertical and 3D position accuracy improvements are 88% and 85% for the new RIMs and 83%, and 80% for the GIM CODE, respectively. During the geomagnetic storm in March 2015, the new RIMs improve the 3D position accuracy by 84%, which is similar to that resulting from the GIM CODE. For both study periods, the new RIMs decrease the vertical RMS position error more effectively than the GIM CODE.

5.2 Ensemble Forecasting vs. Kp and F10.7

The ML-based ensemble VTEC forecasting is analyzed in detail in **P-II**. The following presents the analysis of the various developed ionosphere models for the 24-hour forecast such as single Decision Tree and different ML-based ensemble models, such as Random Forest, AdaBoost, XGBoost, and Voting Regressor[(2)], regarding the varying geomagnetic and solar activity, represented by the Kp index and the F10.7 index, shown in Figure 5.4. For the analysis, the Kp values are grouped into $\{1, 2, 3, ..., 9\}$ and the F10.7 index into $\{70, 80, 90, ..., 140\}$. The RMS is calculated within each group so that Kp of 1 includes $0 < \text{Kp} \leq 1$, Kp of 2 includes $1 < \text{Kp} \leq 2$, F10.7 of 70 includes $60 < \text{F10.7} \leq 70$, the F10.7 of 80 includes $70 < \text{F10.7} \leq 80$, and so on.

The Decision Tree VTEC model provides mostly the highest RMS values compared to the ensemble VTEC models in both geomagnetic and solar activity analyses. On the other hand, the ensemble VTEC models perform similarly, with no clear winner. For all ML methods tested, an increase in RMS values is observed as the Kp index increases. This is particularly noticeable for Kp index values of 7 to 9. The results of ML-based VTEC models trained on differenced and non-differenced data are similar for high-latitude VTEC for both Kp and F10.7 analyses. For mid-latitude VTEC, the models trained on differenced data have two times lower RMS than other models for the highest Kp value of 9, i.e., during severe storms. Similarly, for $\text{F10.7} = 120$ sfu, the models trained on differenced data have twice lower RMS than others. For low-latitude VTEC, the RMS values are similar between the models trained on differenced and non-differenced data, with slightly higher RMS values for Kp of 7 and 8 for the models trained on differenced data. Similar is the case for $\text{F10.7} = 120$ sfu. Regarding all models, the maximum increase in RMS is about 2 to 4 times for $\text{Kp} > 7$ compared with $\text{Kp} = 1$. For F10.7 > 110 sfu, the maximum RMS increase is about 1.5 to 2 times compared with $\text{F10.7} = 70$ sfu.

[(2)] The VR VTEC model is developed in this dissertation as an ensemble of ensembles or an ensemble meta-estimator by averaging different ML-based ensemble VTEC models. More specifically, VR1 is an ensemble of Random Forest, AdaBoost and XGBoost, while VR2 is an ensemble of Random Forest and XGBoost, as explained in Section 3.2.4.

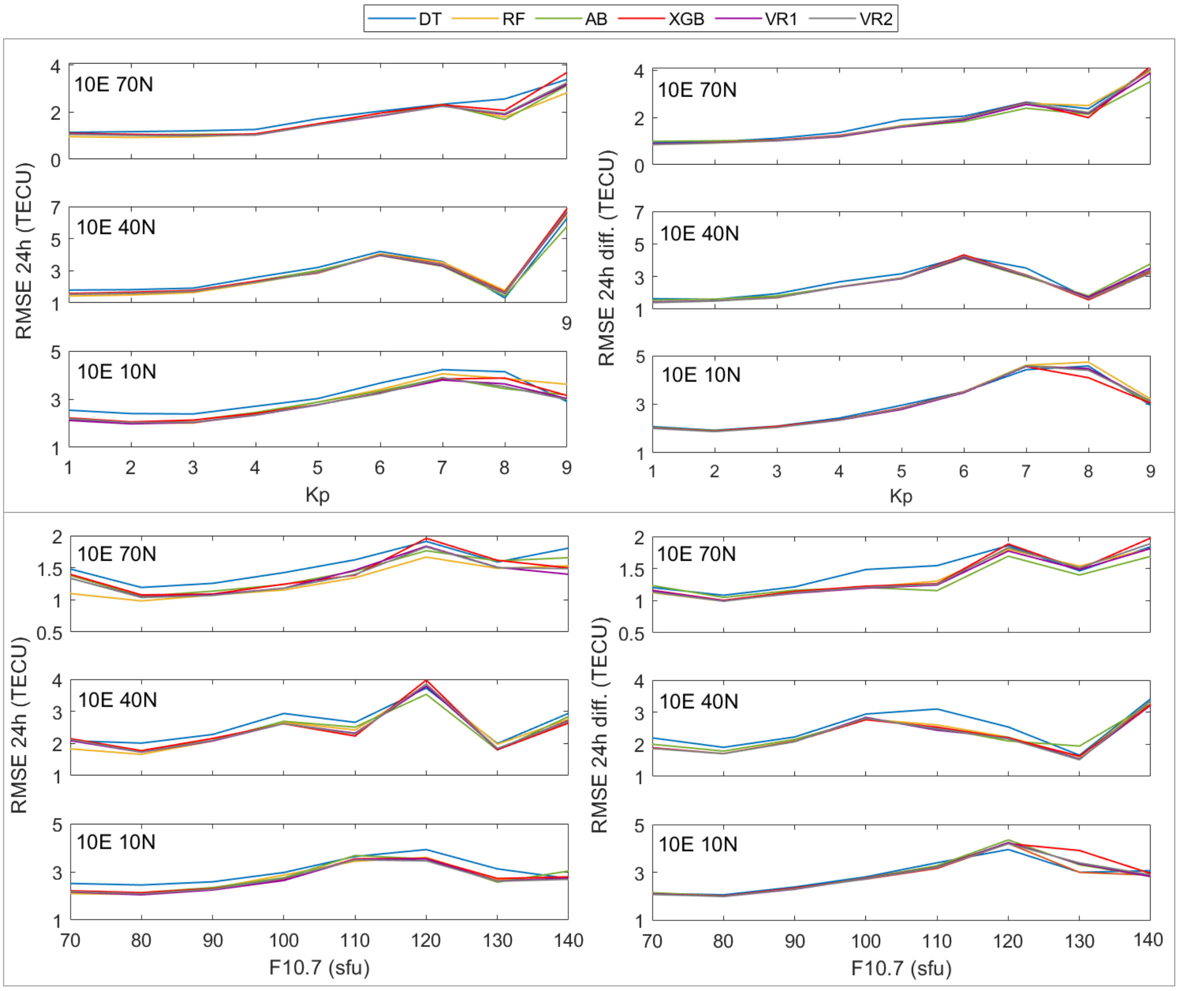


Figure 5.4: RMS of the developed VTEC forecast models: Decision Tree (DT), Random Forest (RF), AdaBoost (AB), XGBoost (XGB), and Voting Regressor (VR) for the 24-hour forecast in relation to the Kp index in the top panel and F10.7 index in the bottom panel. Left: models trained on non-differenced data. Right: models trained on daily differenced data.

5.3 Probabilistic VTEC Forecast

In this dissertation, ML-based probabilistic VTEC models estimate 95% confidence intervals by accounting for uncertainties in the model parameters and/or data, described in Section 3.3, to provide information on how confident and reliable the results are, based on the following approaches

1. Super-Ensemble (SE) of multiple ensemble models and different data sets in 3.3.2.
2. Quantile Gradient Boosting (QGB) with estimated quantiles for confidence interval in 3.3.3.
3. Bayesian Neural Network (BNN) with the probability distributions of the parameters in 3.3.4.
4. BNN+D, which includes also the data uncertainty in 3.3.4.1.

The upper and lower bounds of the 95% confidence interval estimated by applying the above approaches are visualized in Figure 5.5 by setting the vertical axis to zero. In the quiet period, the QGB and BNN+D confidence intervals for the mid-latitude VTEC are similar in size, while the QGB

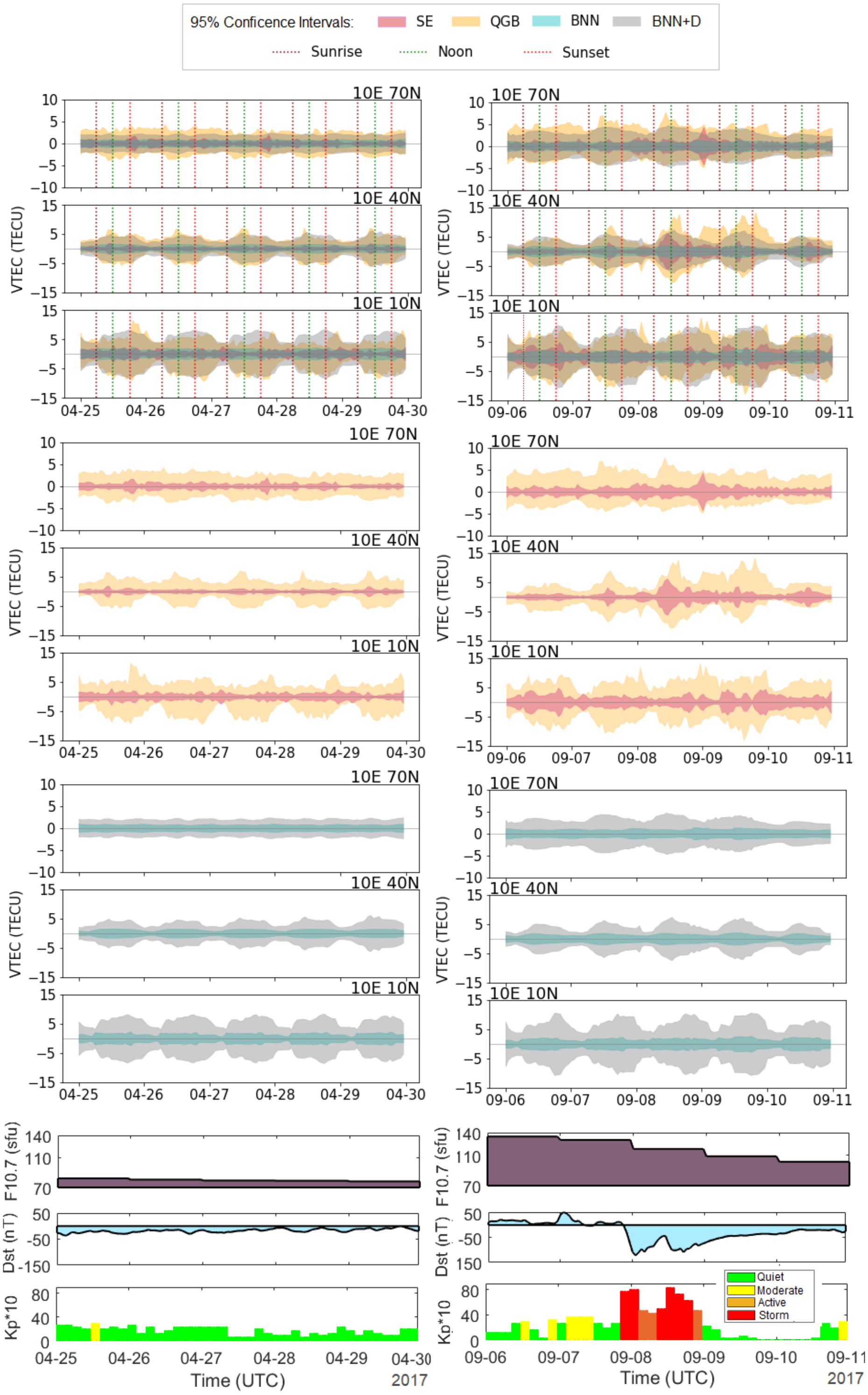


Figure 5.5: 1^{st} panel: the 95% confidence interval (CI) of probabilistic VTEC models, 2^{nd} panel: SE and QGB, 3^{rd} panel: BNN and BNN+D, 4^{th} panel: F10.7, Dst, and Kp·10. Left: April 25 - 29, 2017; right: September 6 - 10, 2017, (taken from **P-III**).

confidence interval for the high-latitude VTEC is wider and the BNN+D confidence upper bound for the low-latitude VTEC is slightly larger. The SE and BNN confidence intervals are the narrowest and similar in size. During the storm, the confidence intervals become wider as the changes in the geomagnetic field occur. For the SE and QGB models, they are around two times wider and more variable on the day of the geomagnetic storm maximum, September 8, and the following day of the recovery phase, September 9, than during the quiet period, while they increase slightly for the BNN and BNN+D models. The largest upper confidence bound for high and mid-latitude VTEC in this period comes from the QGB approach. For the low-latitude VTEC, the upper bounds of the QGB and BNN+D models are similar in size.

For both study cases, quiet and storm, it can be seen that the QGB and SE confidence intervals are more variable and often have peaks, while for BNN and BNN+D, they are smoother and more uniform from day to day. As for the size of the confidence intervals, they are narrower for SE and BNN, while for QGB and BNN+D, they are at least 3 times wider. The confidence intervals of all approaches are wider around local noon for the mid-latitude VTEC, while for the low-latitude VTEC, there is an additional increase in the upper bound after sunset that lasts for several hours.

The post-sunset rise in the upper low-latitude VTEC bounds of the QGB and BNN+D models are visible for September 6 - 9, geomagnetic storm period with F10.7 > 110 sfu, and on April 25 - 29 with low geomagnetic activity, and both periods being close to the equinox (Figure 5.6). The effect is more pronounced in the QGB VTEC model. The post-sunset VTEC enhancement was detected at low latitudes within the equatorial ionization with actual VTEC observations in Dashora et al. (2019); Kutiev et al. (2007); Kumar et al. (2022); Liu et al. (2020a), to name a few. It develops during 2 to 3 hours after sunset, with a peak around 19:00 - 20:00 LT (Kutiev et al., 2007; Kumar et al., 2022), and occurs during prolonged periods of low geomagnetic activity (Kutiev et al., 2007), as well as during geomagnetic storms (Dashora et al., 2019), with stronger intensity around equinoxes (Liu et al., 2020a), and when the F10.7 index exceeds 110 sfu (Kumar et al., 2022). Therefore, the patterns of increase in the upper low-latitude VTEC confidence bound after sunset are consistent with the observations of the post-sunset VTEC enhancement at low latitudes reported in the previous studies.

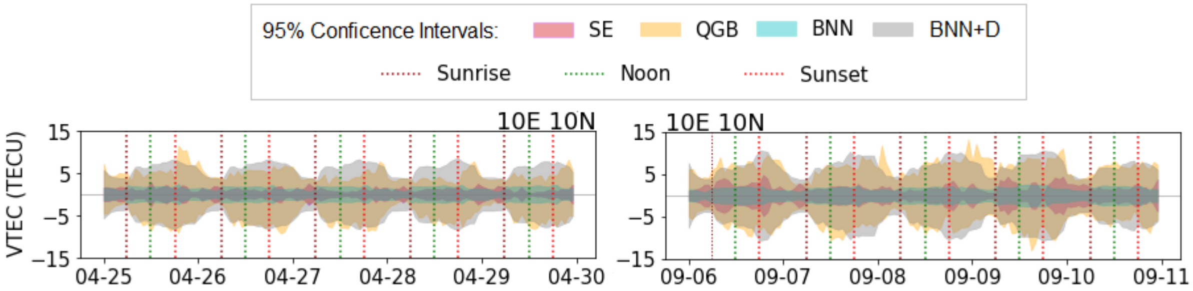


Figure 5.6: Patterns of low-latitude VTEC post-sunset enhancement visible in the upper bounds of the QGB and SE confidence intervals.

Figure 5.7 shows the 24-hour VTEC forecast with the 95% confidence intervals of the developed QGB and BNN+D models for the quiet period (left) and the storm period (right) in 2017. The QGB and BNN+D approaches provide 3 to 4 times wider confidence intervals than the other two approaches, containing more than 95% of ground truth in 2017 and even 100% during the quiet period (see Table 3 in **P-III**). The most significant amount of ground truth outside the confidence intervals is on September 7 and during the first and second Dst minima, representing the maximum intensity of the geomagnetic storm. The magnitude of ground truth outside the confidence interval is up to 4 to 5 TECU during the September 2017 space weather events. During the quiet period, it is less than 2 TECU and occurs less

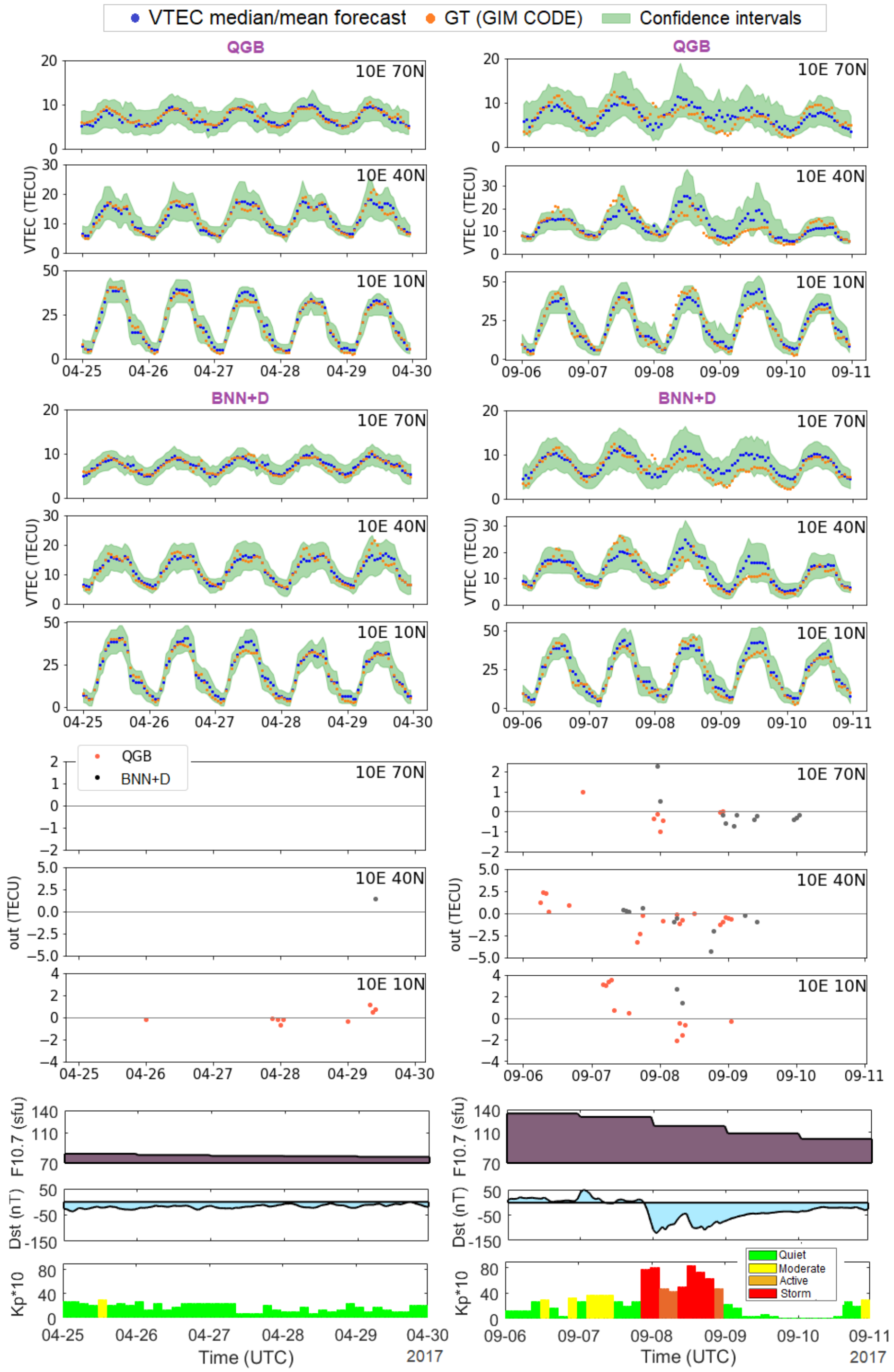


Figure 5.7: Median VTEC forecast of QGB (1^{st} panel) and mean VTEC forecast of BNN+D (2^{nd} panel), both with 95% CI. 3^{rd} panel: ground truth (GT) outside the CI; positive values: GT is above the upper CI limit; negative values: GT is below the lower CI limit. 4^{th} panel: F10.7, Dst, and Kp·10. Left: April 25 - 29, 2017; right: September 6 - 10, 2017, (taken from **P-III**).

frequently. For the BNN+D approach, only one ground truth value is outside the confidence intervals during the quiet period in April 2017.

The advantages and disadvantages of the developed models for probabilistic VTEC forecast are listed in Table 5.1. The QGB and BNN+D approaches provide wider confidence intervals that include around 95% of ground truth and are thus more realistic and reliable than the other two approaches. In addition, the QGB approach avoids the distributional assumption and is the fastest (Table 4.2).

Table 5.1: Advantages and disadvantages of the developed approaches for probabilistic VTEC forecast

	SE	**QGB**	**BNN**	**BNN+D**
PROS	Improved mean VTEC No distribution assumption	Fast to train CI $\sim$ 95% GT No distribution assumption	Higher Corr. to GT	CI $>$ 95% GT
CONS	Many models to train Uncertainty too small	Estimate each quantile	Slow to train Gaussian distribution Uncertainty too small	Slow to train Gaussian distribution

5.4 High-Resolution Probabilistic VTEC

Based on the results from **P-III**, the QGB approach was selected for high-resolution VTEC probabilistic forecasting, which was developed using high-resolution data as reported in Section 4.1. Table 5.2 shows the statistics of the developed high-resolution QGB VTEC models with different input data sets as explained in Section 4.5. Forecasts are made for 15 minutes, 1 hour, 3 hours, 6 hours, and 24 hours in the future. Different models are developed based on the input features:

1. The input features include VTEC, HoD, DoY and SYM/H at time moment i, while the output is VTEC at the time moment $i + t$, where $t \in \{15\text{min}, 1\text{h}, 3\text{h}, 6\text{h}, 24\text{h}\}$ for 15 minutes, 1 hour, 3 hours, 6 hours and 24 hours, respectively, depending on the forecast horizon, denoted as "VTEC SYM/H";

2. The input features include those from item 1 and additionally derived VTEC features such as EMA over the previous 30 and 4 days, and first and second derivatives, denoted as "VTEC SYM/H+" and "M1" in Section 5.5;

3. The input features include delayed/lagged VTEC values for the previous 24 hours, 4 days, and 7 days prior timestamp i (inclusive), denoted as "1d VTEC", "4d VTEC", and "7d VTEC", respectively;

4. The input features include those from item 3 plus additionally delayed SYM/H values for the previous 24 hours, 4 days, and 7 days prior timestamp i (inclusive), denoted as "1d VTEC SYM/H", "4d VTEC SYM/H", and "7d VTEC SYM/H", respectively;

5. The models trained with data from January 2015 to December 2016 are referred to as "I", while the models trained with data from January 2013 to December 2016 are referred to as "II".

Not every data set combination was tested for every forecast horizon, but all combinations were tested for the 24-hour forecast. The selected best models are highlighted in blue and are used for further analysis. Two naive forecasts are chosen for the baseline models. The first naive forecast assumes

Table 5.2: Statistics of the test data for the QGB VTEC forecast at 10° 40°. RMS and correlation coefficients (Corr.) are calculated between the median QGB VTEC and ground truth. CI_{avg} represents the average lower and upper bounds distance from the forecast median QGB VTEC. In (%) represents the percentage of ground truth within the 95% confidence interval. The best and worst results are highlighted in green and red, respectively. When all developed high-resolution probabilistic QGB models have the same correlation coefficients, no values are highlighted, i.e., all are black.

Forecast horizon	**Data set / QGB model**	**January 1 - December 31, 2017** RMS, Corr., CI_{avg}, In (%)	**September 6-10, 2017** RMS, Corr., CI_{avg}, In (%)	**April 25-29, 2017** RMS, Corr., CI_{avg}, In (%)
1 day	1d VTEC I	1.71, 0.91, 3.07, 93.38,	3.47, 0.74, 3.70, 68.48	1.26, 0.96, 3.45, 99.79
	4d VTEC I	1.66, 0.91, 2.96, 93.19	3.13, 0.79, 3.46, 68.68	1.22, 0.96, 3.40, 98.75
	7d VTEC I	1.66, 0.91, 3.02, 93.15	3.18, 0.78, 3.46, 69.52	1.20, 0.96, 3.46, 98.75
	1d VTEC SYM/H I	1.68, 0.91, 3.04, 91.92	3.18, 0.78, 4.07, 77.24	1.28, 0.96, 3.24, 99.58
	4d VTEC SYM/H I	1.63, 0.91, 2.91, 92.21	2.81, 0.84, 3.55, 74.74	1.26, 0.96, 3.21, 98.54
	7d VTEC SYM/H I	1.63, 0.91, 2.98, 91.96	2.68, 0.86, 3.64, 75.99	1.21, 0.96, 3.25, 99.37
	VTEC SYM/H I	1.95, 0.89, 3.04, 89.64	3.28, 0.78, 4.04, 77.87	1.37, 0.96, 3.72, 97.08
	VTEC SYM/H+ I	1.81, 0.90 2.84, 90.25	3.05, 0.81, 3.79, 76.20	1.24, 0.96, 3.37, 97.70
	7d VTEC SYM/H II	1.64, 0.91, 2.90, 91.64	2.72, 0.85, 3.72, 78.71	1.26, 0.96, 3.20, 99.37
	Naive: Frozen ionosphere	2.02, 0.87, / , /	3.70, 0.73, / , /	1.30, 0.96, / , /
	QGB from **P-III**	1.91, 0.89, 3.51, 93.63	3.55, 0.78, 4.73, 81.67	1.26, 0.96, 3.72, 100.0
15 min	1d VTEC SYM/H I	0.37, 1, 0.88, 94.74	0.52, 1, 1.08, 89.77	0.32, 1, 0.85, 97.50
	4d VTEC SYM/H I	0.39, 1, 0.98, 94.86	0.54, 1, 1.22, 89.56	0.33, 1, 0.96, 97.50
	VTEC SYM/H I	0.38, 1, 0.77, 93.97	0.44, 1, 0.78, 88.31	0.37, 1, 0.80, 96.87
	VTEC SYM/H+ I	0.35, 1, 0.70, 93.50	0.38, 1, 0.80, 88.54	0.33, 1, 0.76, 97.29
	VTEC SYM/H+ II	0.36, 1, 0.71, 93.81	0.36, 1, 0.78, 90.00	0.32, 1, 0.75, 96.25
	Naive: Frozen ionosphere	0.42, 0.99, / , /	0.46, 1, / , /	0.45, 1, / , /
1h	1d VTEC SYM/H I	0.86, 0.98, 1.89, 93.42,	1.18, 0.98, 2.46, 88.31	0.76, 0.98, 2.01, 97.29
	7d VTEC SYM/H I	0.88, 0.98, 2.08, 93.22	1.22, 0.98, 2.55, 81.25	0.81, 0.98, 2.30, 97.08
	VTEC SYM/H+ I	0.82, 0.98, 1.53, 92.14	0.95, 0.98, 1.74, 87.71	0.79, 0.98, 1.67, 95.42
	VTEC SYM/H+ II	0.88, 0.98, 1.82, 92.37	0.93, 0.98, 1.75, 88.52	0.83,0.98, 1.80, 95.41
	Naive: 25°40°	0.92, 0.98, / , /	1.13, 0.98, / , /	0.82, 0.98, / , /
3h	1d VTEC SYM/H I	1.54, 0.92, 2.80, 91.47	2.36, 0.89, 4.20, 87.06	1.21, 0.96, 3.10, 97.50
	7d VTEC SYM/H I	1.43, 0.93, 2.56, 91.06	2.15, 0.93, 3.43, 82.26	1.19, 0.96, 2.94, 95.82
	VTEC SYM/H+ I	1.71, 0.90, 2.71, 88.90	2.04, 0.92, 3.14, 88.90	1.51, 0.94, 3.54, 94.38
	7d VTEC SYM/H II	1.46, 0.93, 2.63, 91.47	2.30, 0.90, 3.56, 85.80	1.19, 0.96, 2.90, 96.24
	Naive: 55°40°	1.78, 0.90, / , /	2.51, 0.90, / , /	1.80, 0.91, / , /
6h	1d VTEC SYM/H I	1.73, 0.90, 2.98, 94.74	2.67, 0.85, 4.58, 88.31	1.25, 0.96, 0.85, 98.54
	7d VTEC SYM/H I	1.56, 0.92, 2.79, 95.82	2.46, 0.90, 3.67, 82.26	1.15, 0.97, 3.28, 97.70
	VTEC SYM/H+ I	1.95, 0.88, 3.10, 88.21	2.67, 0.87, 3.52, 81.25	1.39, 0.96, 3.90, 97.08
	7d VTEC SYM/H II	1.59, 0.92, 2.81, 91.14	2.60, 0.87, 3.80, 84.55	1.23, 0.97, 3.08, 97.91
	Naive: 100°40°	2.06, 0.87, / , /	2.81, 0.89, / , /	1.96, 0.91, / , /

frozen ionosphere conditions, where VTEC $(i+t)$ equals VTEC (i) for 15-minute and 24-hour forecasts. However, the frozen ionosphere assumption is not reasonable for 3-hour and 6-hour forecasts due to diurnal VTEC variability. Therefore, another naive forecast is introduced where we used VTEC values from the longitude of 25° for the 1-hour forecast, 55° for the 3-hour forecast, and 100° for the 6-hour forecast along the same latitude of 40°. Here, we consider that the ionosphere structures move from east to west relative to the solar zenith angle and assume similar ionosphere conditions at different longitudes along the same latitude at the same local time. Zhang et al. (2022b) proposed a similar approach where the GIM is predicted by a previous period in a geomagnetic latitude and local time coordinate system.

Figures 5.8 and 5.9 present the median VTEC forecast (blue) and the 95% confidence interval (green) of the developed high-resolution QGB ionosphere models with two years of training data from January 2015 to December 2016 (left) and four years of training data from January 2013 to December 2016 (right). These results correspond to the models highlighted in blue in Table 5.1. Figure 5.8 represents a period of the quiet ionosphere from April 25 to 29, 2017, and Figure 5.9 a period of the disturbed ionosphere during space weather events from September 6 to 10, 2017.

The 15-minute forecast has the narrowest confidence interval, deviating less than 1 TECU from the median VTEC forecast, making them the most confident. This is due to the fact that the ionosphere does not change significantly within 15 minutes. Moreover, the model uses VTEC information from the current step together with the previous first and second VTEC derivatives. Therefore, it is not difficult to forecast VTEC for the next 15 minutes. Its confidence interval includes around 94% to more than 97% of the ground true VTEC in 2017, and around 90% during the space weather event.

When the forecast horizon is increased to 1 hour, the confidence interval becomes about twice as large. It includes ground truth VTEC 95% of the time in 2017 and 88% during the space weather events. For the further forecast horizons of 3 hours, 6 hours, and 24 hours, the confidence intervals extend 3 to 4 TECU from the median VTEC forecast. Their intervals include 92% of ground truth VTEC in 2017, 96% to even 99% during the quiet period, and around 80% during the space weather period.

The confidence intervals tend to be larger and more variable for days with strong VTEC fluctuations (Figure 5.9) than for quiet days (Figure 5.8). During the first SYM/H minimum on September 8, a small part of the ground truth, up to 0.5 TECU, is outside the confidence interval. During the second SYM/H minimum in the second half of September 8, a larger proportion of the ground truth lies outside the lower confidence bound. This trend continues during the recovery phase on September 9.

The models trained on four years of data provide confidence intervals that include 3.5% more ground truth VTEC during sudden disturbances from severe space weather than the models trained on two years of data, as shown in Table 5.1. Furthermore, the magnitude of ground truth VTEC falling outside the predicted confidence intervals is up to 2 TECU smaller during the intense VTEC increase on September 7, and up to 1 TECU smaller for the quiet period for the models trained on 4-year data than for the models trained on 2-year data.

5.5 Forecasting the Solar Flare Impact

This section analyzes the applicability of high-resolution solar irradiance data for the ML-based forecast of the ionosphere response to a strong solar flare. Thus, Ly-α and LYRA observations are added separately to the QGB VTEC model, denoted as the "VTEC SYM/H+ II" model in Table 5.2, for the 1-hour forecast trained with data from January 2013 to December 2016. The QGB VTEC models with added Ly-α and LYRA observations are referred to as the M2 and M3 models, respectively, while the M1 model is "VTEC SYM/H+ II" model without Ly-α and LYRA observations.

The period from September 6 to 10, 2017, represents one of the most flare-productive periods of the solar cycle 24, with the strongest solar flare X9.3 emitted on September 6, 2017, peaking at 12:02

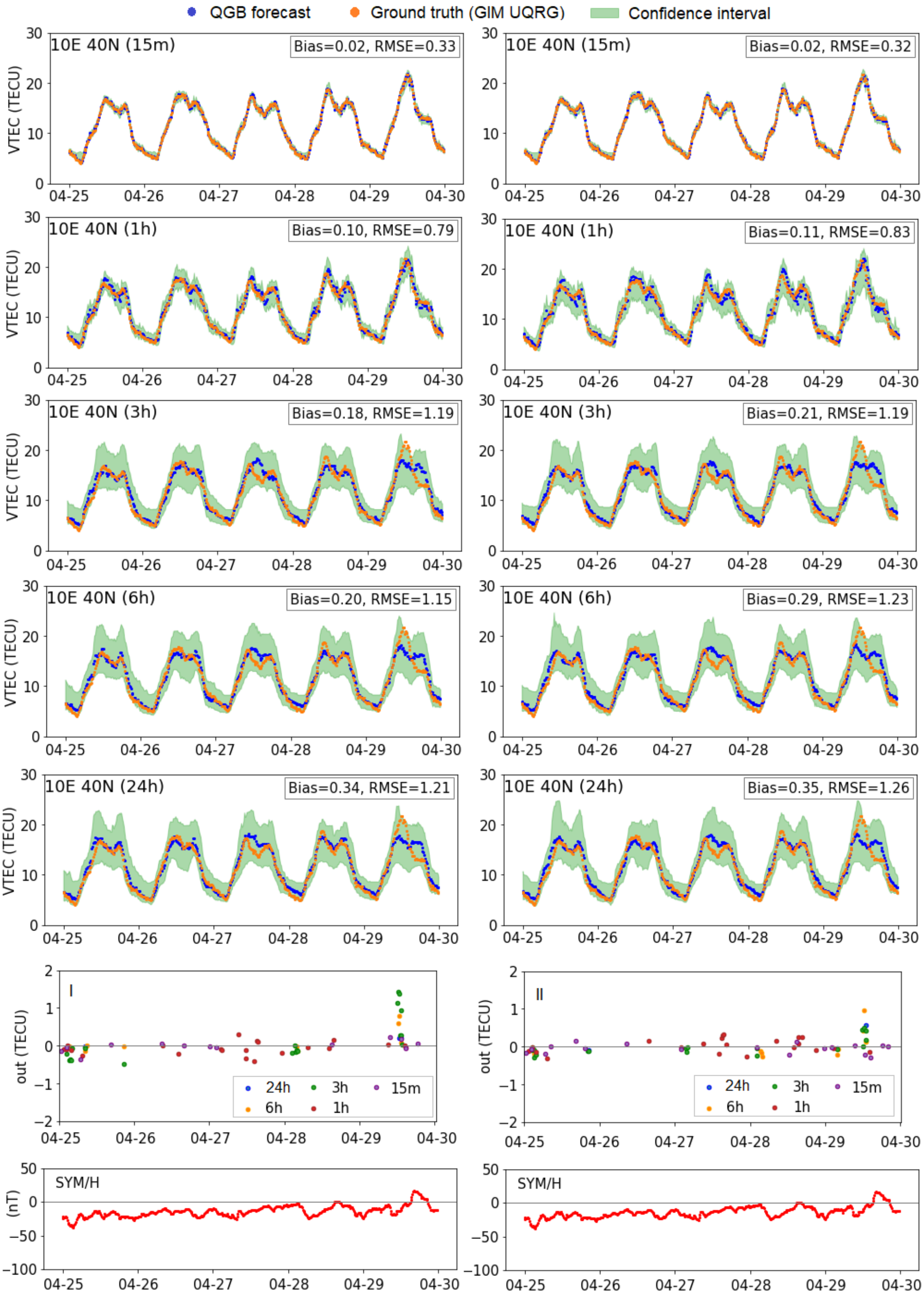


Figure 5.8: QGB VTEC median at 10° 40° with the 95% CI using 2-year (left) and 4-year training data (right), for different forecast horizons for April 25-29, 2017. 1st row: 15-min, 2nd row: 1-h, 3th row: 3-h, 4th row: 6-h, and 5th row: 24-h forecasts. 6th row: GT outside the CI; positive values: GT is above the upper CI limit; negative values: GT is below the lower CI limit. 7nd row: SYM/H index.

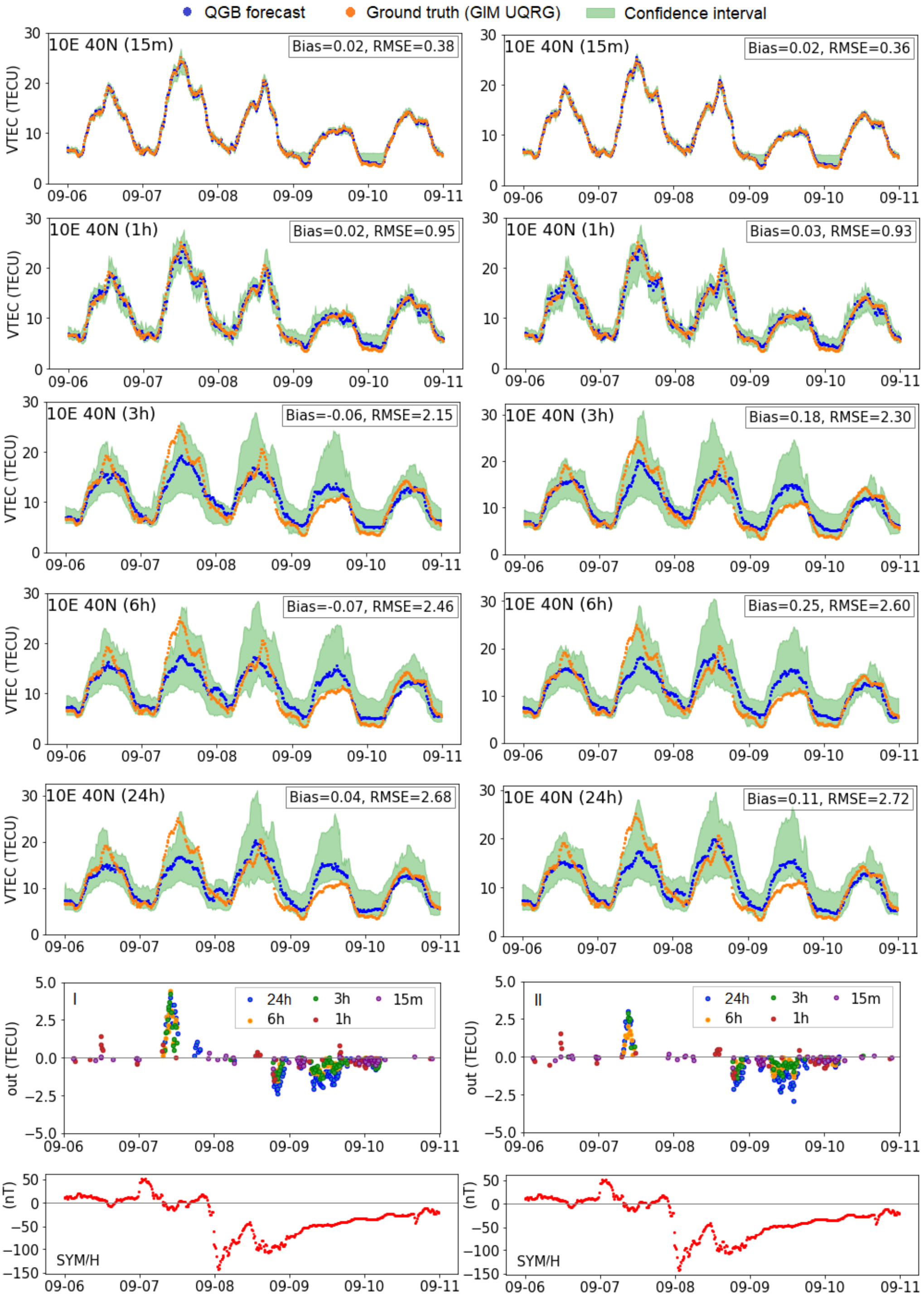


Figure 5.9: QGB VTEC median at 10° 40° with the 95% CI using 2-year (left) and 4-year training data (right), for different forecast horizons for September 6-10, 2017. 1st row: 15-min, 2nd row: 1-h, 3th row: 3-h, 4th row: 6-h, and 5th row: 24-h forecasts. 6th row: GT outside the CI; positive values: the amount by which GT is above the upper CI limit; negative values: the amount by which GT is below the lower CI limit. 7nd row: SYM/H index.

UTC (Table 5.3). The solar flare location was in the southwest of the solar disk. On September 6 and the day before, there was a quiet geomagnetic activity with a maximum Kp value of 4. This means that the ionosphere was not under the influence of geomagnetic disturbances and thus provided an opportunity to analyze the pure solar flare effect. Also, the intensity of the solar flare, its location on the solar disk, and occurrence around local noon at 10° of longitude make the X9.3 solar flare an excellent example for this study. As in Section 2.2 mentioned, the solar flare effects on the ionosphere depend on the latitude and solar zenith angle. In this context, the strongest impact on the ionosphere is expected in the mid-latitude, around local noon and from a flare occurred near the center of the solar disk, which is the case for this study example.

Table 5.3: Information on solar flare X9.3 of September 6, 2017, and maximum Kp value for that day and the previous day.

Start	**Peak**	**End**	**Intensity**	**Location**	**Kp$_{\max}$**	**Kp$_{\max}$**
(UTC)	(UTC)	(UTC)			Sept 6, 2017	Sept 5, 2017
11:53:00	12:02:00	12:10:00	X9.3	S08W33	3+	4

The UV emission of Ly-α increases with the occurrence of the X9.3 solar flare, soon peaks, and then begins to decrease a bit, as shown in Figure 5.10. Soon it increases again, followed by a slow decrease during the next 3 hours, implying that the UV emission remained higher for a longer time. Similar trends can be observed in the LYRA data. However, its second peak is much smaller than the first one.

The increase in ground truth VTEC compared to the 15-day VTEC median is 3 to 5 TECU from high- to mid-latitudes along the 10° of longitude. Comparing the VTEC forecast without solar input data (Figure 5.10, middle) and with the Ly-α data (Figure 5.10, right), the differences become visible during the increase in UV emissions, indicated by a brown rectangle. The model using L-α data as input, denoted M2, forecasts VTEC values that are more consistent with ground truth than the model without Ly-α observations, denoted M1. The model using LYRA data instead of Ly-α data, denoted M3, forecasts an increase in VTEC at 40° of latitude one step or 15 minutes later, i.e., at 13 UTC. The differences from ground truth are smaller for the M2 and M3 models than for the M1 model. The differences between the M2 and M1 models are up to 2 TECU at 40° of latitude and up to 0.5 TECU at 60° of latitude within the brown rectangle. These differences reveal latitudinal effects during high UV emissions, which are the largest for mid-latitude VTEC and the smallest for high-latitude VTEC. Such an effect is not visible outside the brown rectangle. These differences between the M2 and M1 models are mainly positive within the brown rectangle due to the increased ionization during the high UV emissions released by the solar flare and captured by the M2 model but not by the M1 model.

The correlation matrices in Figure 5.11 show positive correlations. The forecasted VTEC of the M2 model correlates by 0.2 to 0.3 more strongly with ground truth than the M1 and M3 models. The correlation of ground truth and the M2 model to Ly-α is strongest at the mid-latitude of 40°, where the most substantial solar flare impact on the ionosphere is observed in Figure 5.10. The correlation of VTEC to Ly-α decreases with increasing latitude. The M1 model has the lowest correlation to Ly-α at mid-latitudes of 40° and 45°, while the situation reverses at high latitudes of 50° and above.

It is important to note that it is difficult to observe the pure solar flare effect in the high-latitude ionosphere due to the overlapping disturbances from the polar ionosphere region and that the pure solar flare effect can only be seen in the mid-latitude ionosphere (Berdermann et al., 2018). Thus, in the results for VTEC at high latitudes, 50° and above, it is challenging to distinguish the effects of the solar flare. On the other hand, the results for mid-latitude VTEC at 40° and 45° reveal the influence of the solar flare in the ground truth VTEC data, which are best captured by the M2 model, i.e., QGB VTEC model with added Ly-α observations. The results show that the M2 model better forecasts the increased ionization in the mid-latitude ionosphere due to the solar flare than the models without

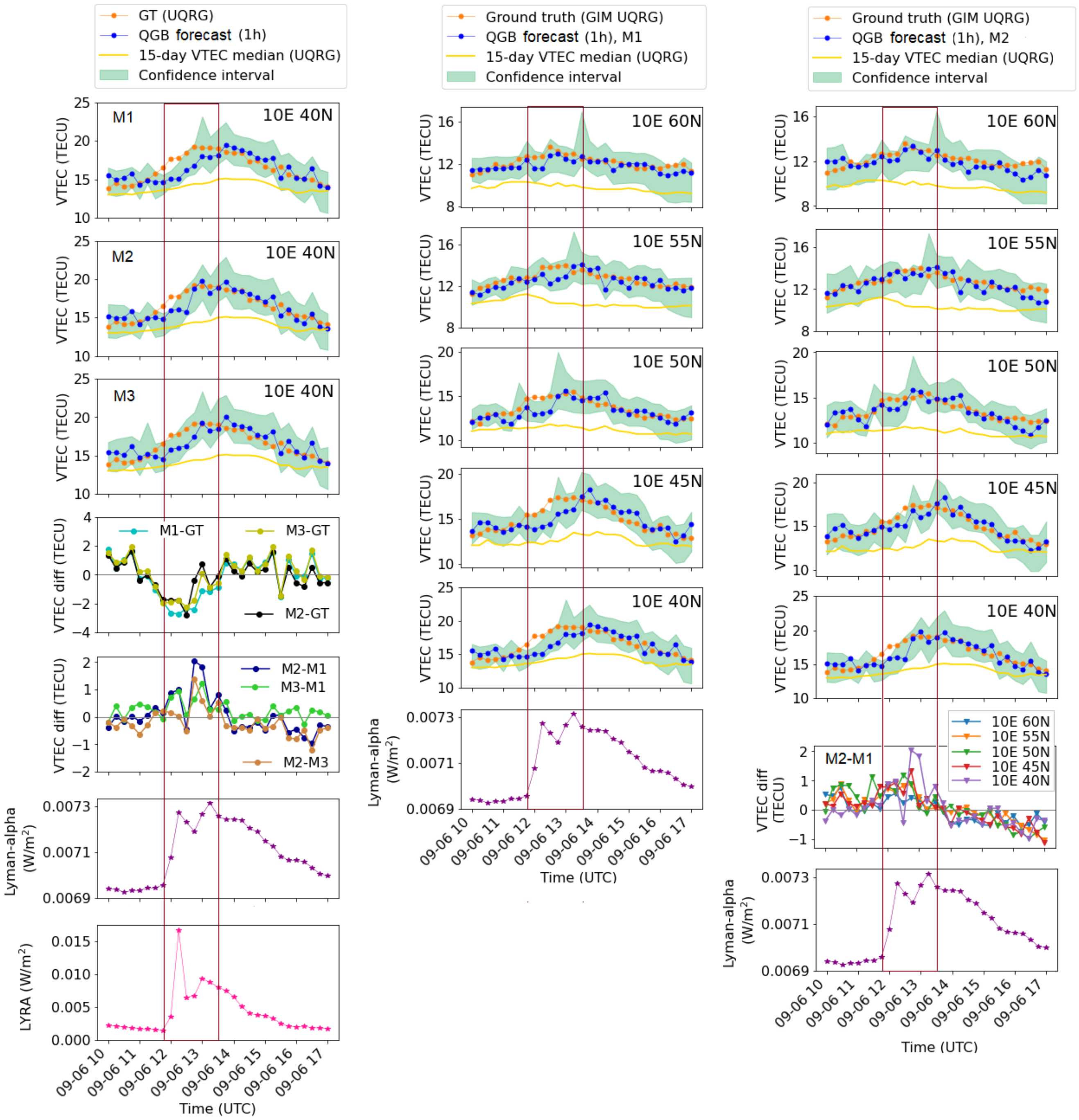


Figure 5.10: Left: Median VTEC forecast of the QGB models M1 (1st row), M2 (2nd row) and M3 (3rd row) with CI and GT; Differences between median VTEC forecasts of the QGB models and GT (4th row); differences between median VTEC forecasts of the QGB models (5th row); Ly-α (6th row, left and mid; 7th row, right); LYRA (7th row). Median VTEC forecast for 60° (1st row), 55° (2nd row), 50° (3rd row), 45° (4th row), 40° (5th row) of latitude along the 10° longitude of the M1 (mid) and M2 (right). Differences between median VTEC forecasts of the M1 and M2 for different latitudes (6th row, right).

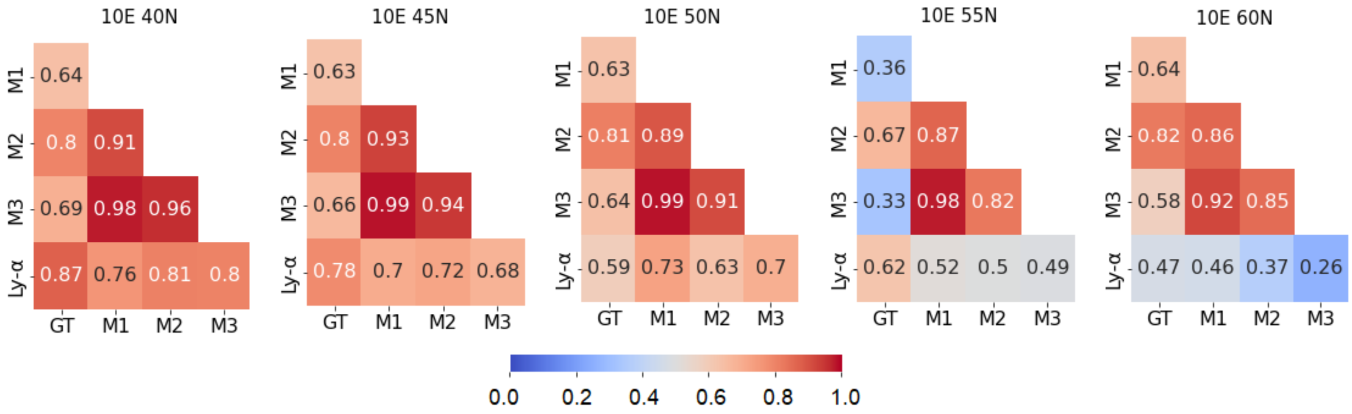


Figure 5.11: Lower half of correlation matrices for September 6, 2017, during increased Ly-α emissions from 11:00 to 16:00 UTC.

Ly-α or with LYRA data. Moreover, the correlation with the ground truth VTEC during the effects of the X9.3 solar flare on the ionosphere and increased UV emissions is highest with the VTEC of the M2 model of about 0.8, while at the same time, other models provide correlations of 0.6 and even lower. These results imply that adding the Ly-α observations allows modeling of the solar flare impact on VTEC and provides forecasts consistent with ground truth.

5.6 Data and Algorithm Selection Effect

The RMS and correlation coefficients for the ANN and RF regional VTEC models of **P-I** with different input features are shown in Figure 5.12. Table 3 of **P-I** lists the developed models and associated input features. The ANN1 model is linear, with no hidden layer, and its input features consist of regional ionosphere coefficients, latitude, and longitude. All other models are nonlinear. The ANN2 model has the same input features as the ANN1 model. The ANN3 model adds a feature of HoD, the ANN4 model adds a feature of the F10.7 index, the ANN5 model adds features of Kp and Dst indices, while the ANN6 model removes the input features of regional ionosphere coefficients. The input features of

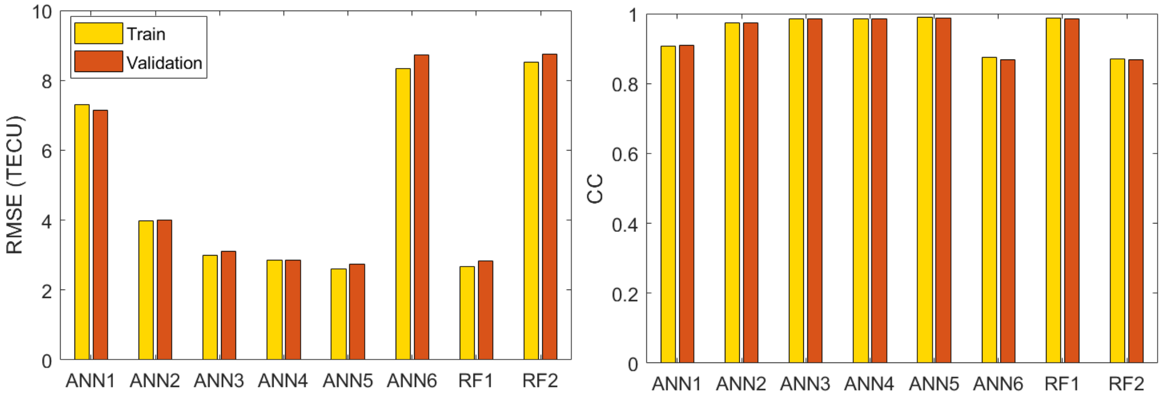


Figure 5.12: The RMS (left) and correlation coefficients (right) for training and validation of ANN and RF models, (taken from **P-I**).

the ANN5 model are used for the RF1 model and those of the ANN6 model for the RF2 model. The ANN5 model is selected as the optimal for the RIM IONOWB_AI.

Applying the nonlinear modeling reduces the RMS by around 3 TECU. The stepwise addition of the temporal, solar, and geomagnetic activity input features further reduces the RMS of the ANN-based model accordingly. These input variables are useful to derive temporal, solar, and geomagnetic-related VTEC features and relationships. Furthermore, removing the regional ionosphere coefficients from the ANN6 and RF2 models increases the RMS by a factor of three and decreases the correlation coefficients by 0.1. These results suggest that modeling VTEC with only the spatiotemporal, geomagnetic, and solar parameters is insufficient. These data alone cannot provide an accurate description of VTEC. However, the modeling can be significantly improved when combined with more influential, descriptive parameters of the ionosphere, such as regional ionosphere coefficients. When comparing different learning algorithms, the RMS between the ANN and RF models are very similar, with differences up to 0.1 TECU. These results show that using different input features has a much larger impact than using different learning algorithms in **P-I**.

Two data sets, non-differenced and daily differenced, are prepared in **P-II**, **P-III** and **CP-II**, as described in Section 4.1.3. Differencing was used in this dissertation to reduce time dependence and trends and stabilize the mean of the data set. Furthermore, daily differencing allows the regular daily VTEC variations and, thus the background ionosphere to be removed, leaving only the remaining signatures associated with other sources of VTEC fluctuations to be modeled. In addition, for the analysis in Figure 5.13, a third data set is prepared where differenced and non-differenced data are combined with the EMAs and the time derivatives calculated from the non-differenced VTEC to forecast non-

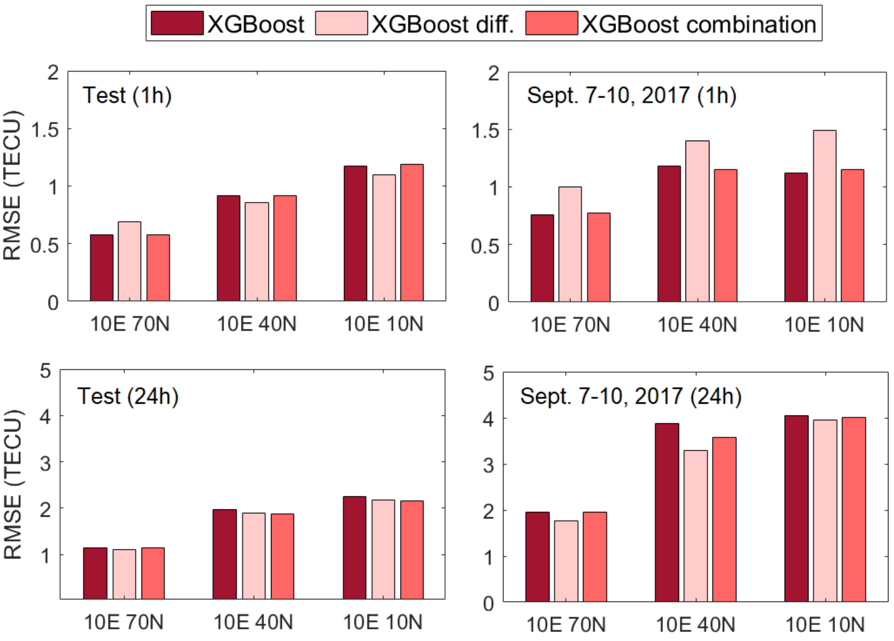


Figure 5.13: The RMS values for January-December 2017 (right) and the September 7-10, 2017 geomagnetic storm (left) for 1-hour (top) and 24-hour (bottom) forecasts for the XGBoost VTEC model. The models trained with differenced data are labeled "diff", and those with non-differenced data are not labeled. The XGBoost combination includes both differenced and non-differenced data as input, and the output is non-differenced VTEC, (taken from textbfP-II)

differenced VTEC, which is referred to as the XGBoost combination. The analysis is performed for the XGBoost VTEC model because it is fast compared to other models (Table 4.2). Moreover, the boosting methods, especially XGBoost, provide the lowest RMS during the space weather event, see Table 5 in **P-II**.

The RMS values for the 1-hour VTEC forecast are similar for the models with the non-differenced data and the data combination. They are higher than the RMS for the differenced data for mid and low VTEC in 2017, but smaller during the geomagnetic storm. For the 24-hour forecast, the RMS for mid- and low-latitude VTEC with the data combination is lower than for the non-differenced data and more similar to the RMS for differenced data. The differenced data result in the lowest RMS in the 24-hour forecast compared to the other two data sets. The enormous improvement with differenced data is for the mid-latitude VTEC during the geomagnetic storm when the RMS is reduced by 0.6 TECU compared to the non-differenced data.

The improvement of the VR1 VTEC model in **P-II** by incorporating new VTEC features is shown in Figure 5.14. First, the models are trained with the first ten input variables in Table 3.1, i.e., without

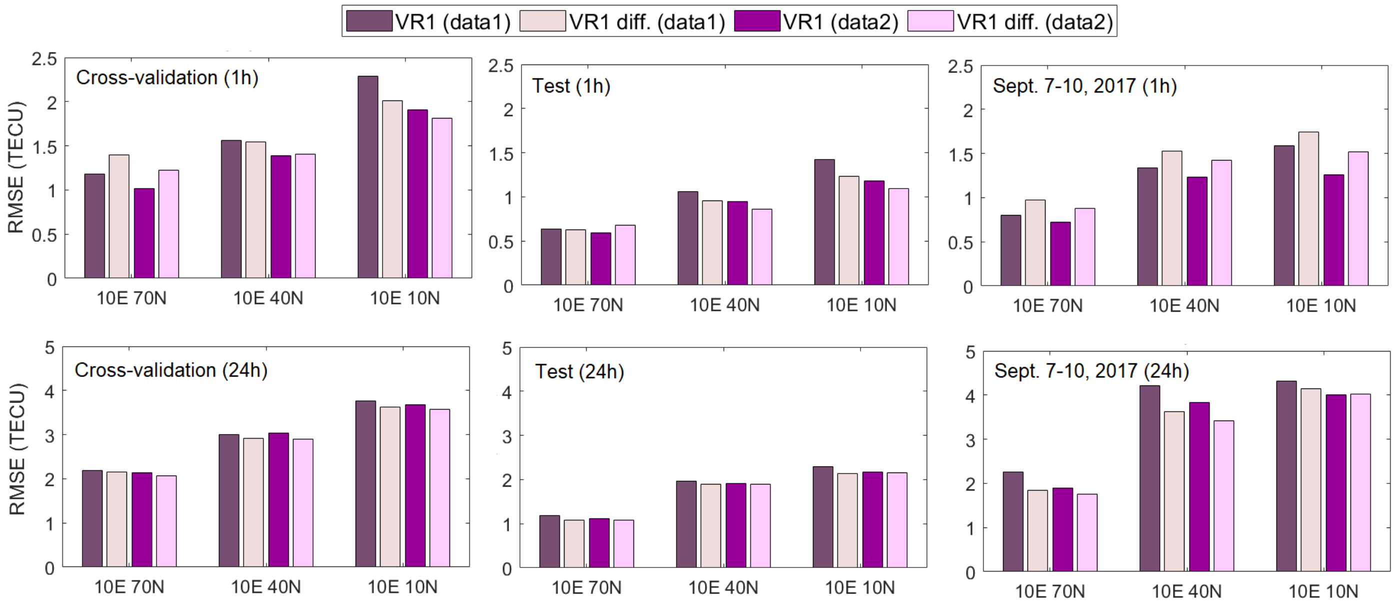


Figure 5.14: The RMS for the cross-validation data (left), the January-December 2017 test data (mid), and the September 7-10, 2017 geomagnetic storm (right) for 1-hour (top) and 24-hour (bottom) forecasts for the VR1 model. The models trained with differenced data are labeled "diff". Data2 refers to the data in Table 3.1, while Data1 contains Data2 without the input features of EMA, VTEC′ and VTEC″, (taken from **P-II**)

EMAs and time derivatives, referred to as Data1. Later, EMAs and time derivatives of VTEC are added, referred to as Data2. For both non-differenced and differenced data, the RMS values decrease by 0.2 - 0.5 TECU when these new VTEC features are added.

Figure 5.15 represents the percent RMS improvement, i.e., a percent reduction in RMS values, between QGB VTEC models with different input features that produced the highest and lowest RMS values in Table 5.2 shown under (a); in terms of naive forecast shown under (b); between QGB VTEC models with 15-minute and 1-hour resolutions shown under (c). Also, an analysis is performed for VTEC models trained with different learning algorithms for 24-hour forecast, based on Table 3 in **P-III** and shown under (d).

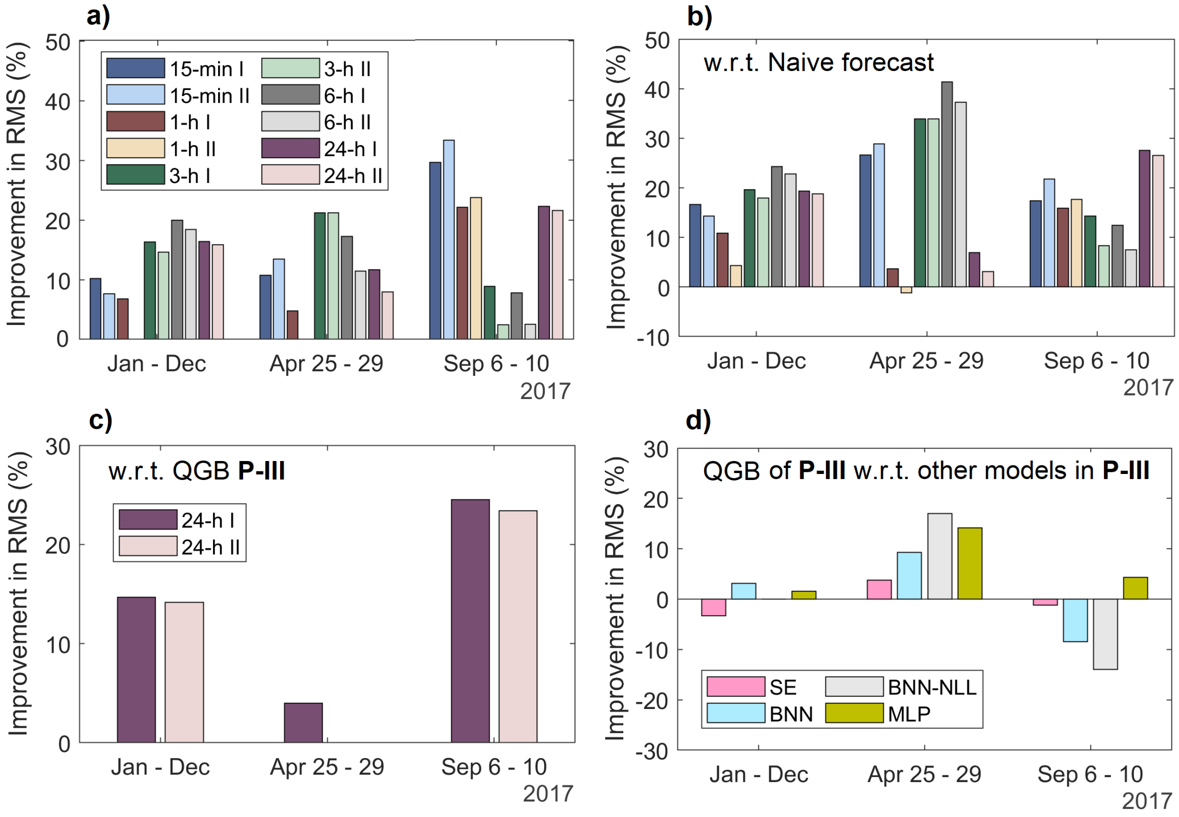


Figure 5.15: Percentage improvement in RMS for 15-minute resolution QGB VTEC models highlighted in blue in Table 5.2 compared to: a) the models with RMS values highlighted in red in Table 5.2; b) the naive forecast in Table 5.2; c) the 1-hour resolution QGB VTEC model from **P-III**; for 2-year training data (I) and 4-year training data (II). d) RMS improvement of the 1-hour resolution QGB VTEC model compared to the other models from **P-III**.

The more suitable input features lead to an RMS improvement of up to 10% to 20% in the year 2017 and April 2017, and around 20% and more than 30% during the September 2017 space weather events. The improvements over the naive forecast are more prominent for the quiet period than the space weather period, except for the 1-hour and 24-hour forecasts. Comparing the RMS values of the QGB VTEC models with 15-minute and 1-hour resolutions, the RMS improvements of 15% and 25% can be observed in 2017 and the space weather period, respectively, while they can be neglected in the quiet period. Considering that the RMS is improved by 16% in 2017 and 20% in the space weather period by selecting suitable input features in (a), it can be concluded that a large part of the improvements in (c) is due to the improved choice of input features and a smaller part due to the increased temporal resolution. Comparing the different models in **P-III** concerning the 1-hour resolution QGB VTEC model with the same input features and training data length for the 24-hour forecast, their RMS differences from January to December 2017 are around 3% and lower and can therefore be neglected. In the quiet period of April 2017, the improvements are to 15%, while in the space weather period, the other ML-based VTEC models, except MLP, perform better by 1% to 14%. This is, at most, half the improvement compared to the improvement of the QGB VTEC model with a better selection of input features.

6 Summary and Conclusion

In geodesy, space weather is identified as a natural hazard to GNSS infrastructure and positioning and navigation applications. The impact of space weather on GNSS occurs primarily through the ionosphere and the plasmasphere. This upper part of the atmosphere significantly influences the propagation of GNSS signals and can degrade the positioning accuracy. As the demand for GNSS accuracy increases and our society becomes more dependent on applications that require high-precision positioning, navigation, and timing, it is essential to develop advanced modeling and forecasting methods for the space weather effects on GNSS and the ionosphere in order to correct for these effects in an accurate and timely manner. These methods must rely on modeling nonlinear solar-terrestrial coupling processes to develop a system limiting the detrimental effects of solar storms.

In this dissertation, novel ionosphere models are developed using state-of-the-art ML techniques. In that regard, various learning algorithms and methods are exploited to approximate nonlinear space weather processes in order to model and forecast VTEC including also probabilistic forecast. The models are data-driven, meaning they extract patterns and find modeling or mapping functions from data describing the solar activity, solar wind speed, interplanetary and the Earth's magnetic field, and the ionosphere. In addition, a K-fold time series cross-validation method is implemented to robustly train and validate the models using many data folds and considering time dependencies. The work has also investigated the performance of ML models for different forecast horizons and data, including data transformation, input feature selection, data length, temporal data resolution, etc. In addition, this dissertation analyses the performance of bagging and boosting ML as well as ANN methods. Moreover, the relative importance of input features to VTEC forecast is estimated. It is important to emphasize that this dissertation, for the first time, develops ML-based probabilistic VTEC forecast models with 95% confidence. Furthermore, a thorough investigation of different approaches for uncertainty quantification is conducted. This means that the models not only forecast a VTEC value but also forecast 95% confidence intervals that provide information on how confident and reliable the results are by accounting for the uncertainties in the model parameters and/or data.

6.1 Novel contributions

To summarize the results of this dissertation, the main novelties are listed below. This dissertation makes contributions and innovations in the field of ionosphere modeling and forecasting, including the space weather component, in the following aspects:

1. Development of simpler, computationally efficient, and more interpretable ML approaches for VTEC forecasting. The newly developed models based on decision trees are fast to optimize, computationally efficient, and applicable to limited datasets.

2. Modification of the standard K-fold cross-validation method for VTEC forecasting into a K-fold time series cross-validation by evaluating a model on a rolling basis and preserving time dependencies between observations.

3. Combining ensemble learning models into a super-ensemble to improve forecasting results, especially during a space weather event, and to quantify uncertainties.

4. Providing information about the most important input features and how reliable the results are with respect to a given confidence interval to increase explainability.
5. Introduction of input observations of solar wind plasma speed, interplanetary magnetic field, high-resolution solar X-ray and EUV, derived features of first and second derivatives, moving averages, and daily differenced data into an ML-based ionosphere model.
6. The model and data uncertainties in ML-based ionosphere models are quantified with different approaches developed and implemented to provide information on how confident the results are and where most of the uncertainty comes from, providing probabilistic ionosphere models and improving the interpretability of the models and solutions.
7. Increased temporal resolution of ML-based ionosphere models to 15 minutes by integrating high-resolution solar, geomagnetic, and ionosphere data.
8. Development of ML-based forecast model of the solar flare effects on the ionosphere.

It is also important to emphasize that in this dissertation, the developed ML-based VTEC models are not treated as black boxes whose procedures are not questioned. A black box would be a system where we only observe the input and output variables but not the internal processes. Although ML/DL models are undoubtedly complex, it can be argued that they should not be treated as black boxes if the algorithms are known, if the training data and training procedures are transparent, and if the model is available to examine what it does (e.g., its parameters, its hyperparameter-driven architecture, etc.). In this context, the black box system can be viewed from the perspective of users who treat and use ML as such because they are unaware of some of the components described above. The opposite of a black box is sometimes called a glass box. To make an ML system a glass box, algorithms, training data, cross-validation data, and models must be known and made available. Therefore, the mathematical procedures behind the applied algorithms, the training data as well as the training and cross-validation procedures are presented transparently in this dissertation. In addition, software, data, and trained models are made available for inspection (see Chapter Software 6.3).

6.2 Answers to the Research Questions

The research objectives listed in Section 1.7 are accomplished. The research questions are addressed in various parts of this dissertation and the first authors' publications. The answers to the questions and the discussion of the dissertation results are again placed in the context below. Corresponding references are provided in blue-colored responses, and relevant results are summarized with key findings from the first authors' publications and the dissertation sections that answer a particular research question.

Q-1 What is the impact of the data on ML-based ionosphere modeling and forecasting in terms of the input features, data length, and observations resolution? On the other hand, what is the impact of using different learning algorithms?

Ref.: Sections 5.4 and 5.6, **P-I**, **P-II**, **P-III**

The results in Section 5.6 show that an ML-based ionosphere model is more affected by the selection of input features than by the selection of learning algorithms.

The RMS reduction after selecting appropriate input features in **P-I** is up to 1.5 TECU (Figure 5.12). On the other hand, ANN or RF RIMs result in a difference of only 0.1 TECU on the same data set in **P-I**.

Including input features such as the EMA and time derivatives of VTEC in **P-II** reduces the RMS by up to 0.5 TECU (Figure 5.14). Using daily differenced observations instead of the original ones reduces the RMS by about 0.6 TECU for the mid-latitude VTEC during the geomagnetic storm (Figure 5.13).

Different input feature combinations are analyzed in Table 5.2, Section 5.4 for high-resolution VTEC forecasting. The results show that input features of current VTEC and SYM/H values together with EMA, first and second derivatives of VTEC are suitable for 15-minute and 1-hour forecasts. In contrast, for the 3-hour, 6-hour, and 24-hour forecast horizons, the history of the previous 7-day VTEC and SYH/H values is better suited as input features. Including the SYM/H geomagnetic activity index improves the 24-hour forecast by about 16% during the September 2017 space weather events.

The appropriate input features result in an RMS decrease of up to 10% to 20% in 2017, and more than 30% during the storm period for the high-resolution ML-based VTEC models (Figure 5.15). This represents a much larger fraction of the RMS reduction than that achieved by increasing the temporal resolution from 1 hour to 15 minutes. Using different learning algorithms in **P-III** results in an RMS reduction that is half that obtained by improved input feature selection.

The models trained on 4-year data provide confidence intervals that include 3.5% more ground truth VTEC during the storm than those trained on 2-year data. Also, the magnitude of ground truth falling outside the confidence intervals is 1 to 2 TECU lower for the models trained on 4-year data.

Q-2 Can reduction of regular ionosphere variations and background ionosphere information in the data through daily differencing improve the learning and generalization of the ML-based ionosphere model?

Ref.: Sections 5.2 and 5.6, **P-II**, **CP-II**

The performance of the developed models is slightly distinct for non-differenced and differenced data (Figure 5.13). For space weather events, improvements with differenced data are observed for the longer forecast horizon, i.e., the 24-hour forecast. Using the differenced data, where the prevailing regular diurnal ionosphere variability is reduced or eliminated, results in an RMS decrease of more than 0.5 TECU for the 24-hour forecast during the severe storm (Figure 5.13). The improvements are also observed for the 1-hour and 24-hour forecasts in the year 2017. Only for the 1-hour VTEC forecast during the storm, the models perform better with the non-differenced data, i.e., they have smaller RMS values. It can be concluded that daily differenced data enhance the performance of the ML-based VTEC model for 24-hour forecasting, including a geomagnetic storm.

When using differenced data, the RMS values between different models for different Kp values, especially for the geomagnetic storm with $Kp \geq 5$, are largely consistent between the different ensemble models and show smaller differences from the single Decision Tree (Figure 5.4). This suggests that daily differencing facilitates the learning of structural patterns in the data, even when a less complex model, such as a single Decision Tree, is used.

Q-3 How efficient are ML methods other than ANN, such as decision trees or ensemble learning techniques, in modeling and forecasting the ionosphere?

Ref.: Chapters 3 and 5, **P-I**, **P-II**, **P-III**, **CP-I**, **CP-II**

This dissertation applies learning algorithms as the single Decision Tree and an ensemble of trees as Bagging: Random Forest, Boosting: Gradient, and Adaptive Boosting to estimate an approximation function for VTEC forecasting. In addition, BNN is applied to VTEC forecasting for the first time. Before this, the most commonly used ML approaches for ionosphere modeling, and forecasting were based on MLP and LSTM, which belong to the ML subfield of DL, as explained in Section 1.6.

Analyzing the different developed ML-based VTEC models, we conclude that combining a large number of trees in an ensemble of Random Forest and Boosting improves the accuracy and outperforms a single Decision Tree-based solution. Ensemble learning improves the VTEC forecast over the single tree from about 10% to 30% (Table 5 in **P-II**). Further improvements in accuracy and generalization are achieved by combining ensemble learning models in a meta-ensemble model, such as Voting Regressor in **P-II** and Super-Ensemble in **P-III** and **CP-II**. Meta-ensemble models, i.e., the ensemble of ensembles, provide the lowest RMS and the highest correlation coefficients to ground truth compared to a single ensemble member in **P-II** and **CP-II**.

The results in Chapter 5 and in **P-II**, **P-III**, **CP-I** and **CP-II** show that the developed ML-based VTEC models can capture variations in VTEC, which are consistent with ground truth for the forecast horizons from 15 minutes to 24 hours. These results demonstrate that the ML approaches other than ANN can model and forecast VTEC effectively in different ionosphere regions during quiet and storm periods.

Moreover, the developed ML-based VTEC models not based on the ANN architecture are computationally much more efficient than, for example, the BNN model, as can be seen in Table 4.2. The Decision Tree VTEC model is the fastest but not the best solution since it usually yields the highest RMS values. The GBoost and QGB VTEC models, both based on Gradient Boosting, turn out to be better solutions in terms of accuracy and computational efficiency. In addition, QGB VTEC models have advantages in estimating uncertainties and providing reliable confidence intervals (see the answer to the research question **Q-4**). Regarding computational efficiency, the QGB VTEC model is 30 to 60 times faster than the BNN VTEC model during the training and cross-validation process. These results further demonstrate the benefit of applying the ML methods of ensemble learning for ionosphere modeling and forecasting. Ultimately, the choice of ML method depends on the complexity and availability of the data, the interpretability requirements, and the available computing resources.

Q-4 How can data and model uncertainties be modeled, and how do uncertainties change when the ionosphere is perturbed by a geomagnetic storm?

Ref.: Sections 3.3, 5.3 and 5.4, **CP-II**, **P-III**

The findings on the development of probabilistic VTEC models with 95% confidence and the quantification of uncertainties in VTEC forecast from **P-III** and **CP-II** can be summarized as follows:

1. The SE and BNN approaches provide the lowest uncertainties and, thus, overconfident results. The ground truth VTEC is outside the confidence intervals about $\sim$ 50% of the time.

2. The approaches that capture data uncertainties, QGB and BNN+D, provide wider confidence intervals that contain the ground truth VTEC $\sim$ 95% of the time and are thus more realistic.

3. As for forecasting the mean or median VTEC, the SE approach often yields the lowest RMS value, demonstrating the power of an ensemble to improve the accuracy of an estimate. On the other hand, BNN tends to provide the highest correlations to ground truth, especially during a storm.

4. Confidence intervals, especially of the QGB VTEC model, exhibit variations depending on the daytime, solar irradiance, geomagnetic activity, and post-sunset low-latitude ionosphere enhancement. They are narrower at night, wider around local noon at mid-latitudes, broader and more variable with the change in the geomagnetic field, and increase after sunset at low latitudes depending on geomagnetic and solar conditions.

5. The relative importance of input features shows that the confidence intervals of the QGB VTEC model are determined by space weather indices in addition to VTEC-related input features, with lagged VTEC dominating (see Figure 8 in P-III).

6. The most computationally intensive method is BNN+D, while QGB is the fastest.

During the space weather period, the confidence intervals become wider as the changes occur in the geomagnetic field. The SE and QGB confidence intervals are about twice wider and more variable on the day of the geomagnetic storm maximum and the following day of the recovery than during the quiet period (Figure 5.5). On the other hand, the BNN and BNN+D confidence intervals increase slightly during the space weather event.

The high-resolution probabilistic QGB VTEC forecast analysis in Section 5.4 shows that the size and shape of the confidence intervals follow sudden, intense ionosphere perturbations during space weather. They are larger and more variable from day to day for the September 6-11, 2017 period than for the quiet period April 25-29, 2017, when the confidence intervals are more similar from one day to another. However, they are not entirely constant in the quiet period and quickly adjust to changes in VTEC. The confidence intervals for the space weather period are largest for September 7 and 8, implying that the forecasts are less confident. Both days are after the arrival of the CMEs, resulting in moderate geomagnetic conditions on September 7 and severe geomagnetic conditions on September 8. Another day with a wider confidence interval is September 9, when recovery from the storm began. One day later, on September 10, the forecasts become much more confident as the disturbances in the ionosphere calm down and the SYM/H index recovers. These results demonstrate the dependence of the QGB probabilistic VTEC forecasts on conditions in the ionosphere and geomagnetic field, as well as the sensitivity to perturbations in the ionosphere by the geomagnetic field.

Q-5 How do the forecast accuracy and uncertainties change as the forecast horizon expands, and concerning quiet and storm ionosphere conditions?

Ref.: Section 5.4

The impact of different forecast horizons on model accuracy and uncertainty quantification is discussed for the high-resolution QGB VTEC model based on the results in Section 5.4. The 15-minute forecast has the smallest confidence intervals, deviating less than 1 TECU from the median VTEC forecast. When the forecast horizon is increased to 1 hour, the confidence intervals and, thus, the uncertainties become about twice as large. For the further forecast horizons of 3 hours, 6 hours, and 24 hours, the confidence intervals are about 3 to 4 TECU wide from the median VTEC. It can be concluded that the 15-minute and 1-hour forecasts are the most confident, while the 3-hour, 6-hour, and 24-hour forecasts have a wider and similar range of confidence intervals (Figures 5.8 and 5.9, and Table 5.2).

The confidence intervals tend to be larger and more variable for storm days, where strong VTEC fluctuations occur, than for quiet days. The QGB VTEC models highlighted in blue in Table 5.2 provide confidence intervals that contain the ground truth VTEC more than 90% of the time in 2017 and at least 96% during the quiet period, while the percentage decreases from 90% to about near 80% during severe space weather, depending on the forecast horizon.

The RMS values increase with the increase of the forecast horizons. They are the lowest for the 15-minute forecast and about twice as large for the 1-hour forecast. For the 3-hour, 6-hour, and 24-hour forecasts, the RMS values are about 2 to 2.5 times and 1.5 times larger than for the 1-hour forecast during the space weather and quiet periods, respectively. The RMS values between the 3-hour, 6-hour, and 24-hour forecasts differ by only 0.2 to 0.5 TECU.

Q-6 Is it possible to forecast the ionosphere response to a strong solar flare by incorporating high-resolution ionosphere, solar, and geomagnetic activity data?

Ref.: Section 5.5

High-resolution solar observations, LYRA from the PROBA2 satellite, and Ly-α from GOES-15 and GOES-16 are added separately to the QGB VTEC model with the high-resolution ionosphere and geomagnetic activity data to model the solar flare effects on the ionosphere in Section 5.5. Analyzed for the 1-hour forecast, the results show that integrating the Ly-α observations contributes more than the LYRA data. The reason is that the Ly-α emissions depend significantly on the flare location on the Sun, which is consistent with the geo-effectiveness of a solar flare. On the other hand, the LYRA data are not sensitive to the solar flare location, which contradicts the location-dependent geo-effectiveness of a solar flare. The QGB VTEC with Ly-α data increases 1 hour earlier than the QGB VTEC without solar observations. Their difference is up to 2 TECU for the VTEC grid point at 10° 40°. The increase in VTEC from the model with Ly-α data is about 3 TECU in only 15 minutes and corresponds closely to the increased ionization in the ground truth VTEC.

Since we used the GIM VTEC, the solar flare impact is much smaller in the data than in reality. However, if VTEC were calculated directly from GNSS observations, the effect would be much more significant, as reported in Yasyukevich et al. (2018). In this case, adding solar observations of Ly-α would be more meaningful and provide a more substantial improvement. The properties of a solar flare effect on VTEC found in previous studies of Barta et al. (2022); Berdermann et al. (2018); Yasyukevich et al. (2018) are consistent with the results in this dissertation. More specifically, the QGB VTEC model with Ly-α approximates patterns such as the latitudinal dependence of solar flare effects and the strongest impact at mid-latitudes, as well as the long duration of increased VTEC at mid-latitudes.

The limitations of introducing these high-resolution solar data into VTEC forecasting must also be mentioned. Both the Ly-α and the LYRA data need to be cleaned of various effects and artifacts, resulting in missing data. When comparing the two data sets, we had to remove much more data from Ly-α than from the LYRA data. The reasons are the frequent effects of Earth eclipse and geocoronal absorption in Ly-α due to the geostationary location of the satellites. Interpolation of the removed data is challenging because it can be difficult to represent a realistic situation. Providing incorrect data, such as assuming no significant solar activity during a solar flare, can negatively impact the ionosphere model. On the other hand, if data cannot be interpolated, a backup solution is needed when Ly-α is not available, e.g., to use the model with LYRA observations since they are more complete or if both data sets are not available to switch to a model that does not require solar observations. Therefore, a reliable solution can only be achieved by combining observations from different solar missions that complement each other or using different VTEC models as backups.

Q-7 What is the contribution of the different input features to the model result?

Ref.: Section 4.6, **P-I**, **P-II**, **P-III**, **CP-II**

Knowing the underlying relationships between an ML model's input and output is useful for interpreting what the model has learned and which input features have been selected as relevant. The computation is performed as described in Section 3.4.

The relative importance of the input features shows that the VTEC forecast with non-differenced data is mainly based on previous VTEC values, temporal information, and the solar radio flux F10.7, especially for mid- and low-latitude VTEC in **P-II**. Similar results are obtained by RIM in **P-I**. The contribution of other variables such as solar wind speed, Bz, AE, Kp, and Dst indices becomes more significant with differenced data in **P-II**. As a result, the models trained on differenced data use almost all of the input features to forecast VTEC, especially during storm periods. The relative importance of the input features in **P-III** shows that the confidence intervals for the QGB model are determined by space weather indices in addition to VTEC-related input features, with lagged VTEC dominating.

6.3 Outlook

Early warning and forecasting systems are essential to limit the detrimental effects of solar storms. Further rapid development and adaptation of the emerging AI tools for operational ionosphere and space weather forecasting systems can be expected in the next decade. The results of this dissertation demonstrate the feasibility of nonlinear modeling using AI techniques and the performance of the developed ML-based ionosphere models under quiet and storm conditions. Also, this dissertation proposes methods for reliable and computationally effective VTEC forecasting, including uncertainty quantification. However, the work can be further improved. Therefore, recommendations for future work are provided below.

□ Future work related to the **P-I** study should include the addition of dense CORS observations from other countries within the western Balkans to improve the accuracy of the developed RIMs and reduce the observation gaps. Also, observations from the EPN stations near and further away from the study region borders can be introduced to fill in the observation gaps between the IPPs. The GNSS data should be processed over a more extended period to re-train the developed ANN-based RIM to improve its accuracy and applicability. The ML approach can be further improved by automatically extracting features in the spatial domain using a convolutional neural network and in the temporal domain using a recurrent neural network. To facilitate their application in national positioning services, the RIMs should be implemented in near-real time with a latency of up to 3 hours.

□ Include information on the uncertainties of the input observations and the observations used as the ground truth for model training. This dissertation uses the input variables and ground truth from the GIM CODE for the 1-h resolution models and the GIM UQRG for the 15-min resolution models. However, the GIM products are not error-free. The standard deviations of VTEC at 10° 40° from GIM UQRG are around 2 TECU in 2017, regardless of whether it is a calm or a storm period. Thus, they are in a similar range as the confidence interval for the 1-hour VTEC forecast. In further work, information on the standard deviation of the GIM should be considered in training and cross-validation of the models or their evaluation and estimation of the corresponding statistics. Such information can be included as an input feature of an ML-based model, as in Kiani Shahvandi & Soja (2022).

□ Based on the findings in **P-III** and Section 5.3, a probabilistic VTEC forecast that only considers the model parameter uncertainties is insufficient. The ML-based VTEC models trained with different learning algorithms using the same or similar data set within an ensemble show similar performances. The reason is that an approximation function is estimated from similar data, which leads to smaller discrepancies between the solutions of the ensemble members. The ensemble approach for uncertainty quantification can be improved by training the ensemble members on different subsets of data covering different study cases and time periods, which can increase diversity and randomness among ensemble members.

□ A probabilistic VTEC forecast that accounts for both model and data uncertainties would be the optimal solution, as demonstrated by the BNN+D approach. Due to its computational complexity, modification may be required to obtain a computationally efficient and accurate model. In this context, the advantage of fast gradient boosting computation on decision trees can be exploited. The QGB VTEC model could be improved by adding the model uncertainties, e.g., via an ensemble of multiple diverse QGB VTEC models or via virtual ensembles (Malinin et al., 2021) using a single gradient boosting model. Instead of estimating multiple quantile functions separately, the method can be improved to estimate them simultaneously (e.g., Han et al. (2021); Liu & Wu (2011)). Moreover, adding information about the uncertainty of the observations directly into a model may improve probabilistic estimation and provide a more realistic representation of uncertainties (e.g., Kiani Shahvandi & Soja (2022)).

□ The developed VTEC forecast models can be used to support positioning applications. For a global or regional application, the models should be spatially extended. For operational purposes, the models need to use VTEC input from the rapid or real-time GIMs or estimate it directly from GNSS observations.

□ The results of the developed models should be validated for the latest period as a new solar cycle is underway. Also, modeling and forecasting of the space weather effects on the ionosphere can be implemented using VTEC values estimated and calibrated directly from GNSS observations rather than from the GIMs. Based on previous findings, the space weather effects should be much more pronounced there, as they are significantly reduced in GIMs due to the modeling and smoothing procedures.

□ The results support the idea of data importance, which is the core of ML and one of the main factors for ML performance. The results also show that the uncertainty arising from the data is much larger than that of the model parameters, which means that the input data of an ML-based ionosphere model are much more important to be considered for future improvements. Therefore, the input data should be further explored with new observations and extraction of input features suitable for modeling and forecasting VTEC over longer horizons and during space weather events. Incorporating input features that can characterize the space weather impact in a helpful way for learning may be beneficial. Including data with higher temporal resolution and minimizing the need to interpolate values may also reduce uncertainties. Further studies also need to investigate how high-resolution space weather data can be leveraged to be applicable to ML, as they are often characterized by data gaps and artifacts.

□ Based on the analysis in Section 4.2.1, the modeling and forecasting of space weather impacts suffer from few examples of space weather events because they occur relatively infrequently compared to quiet periods, resulting in an uneven data representation or imbalanced data. If the training data contains balanced cases, model performance for space weather events is expected to improve and the uncertainties to reduce. The selection of a learning algorithm that can handle imbalanced data sets is also essential in this context. For example, ensemble learning is recognized as a method that can provide significant improvements in the case of a skewed distribution (Krawczyk, 2016), and boosting learning algorithms are shown to be suitable for applications with imbalanced data (Esposito, 2020), both of which were applied in this dissertation. However, there is room for improvement in handling imbalanced data sets. Recommended solutions to be explored include improving the input features for learning rare space weather-related VTEC signatures, training on the balanced data set with oversampling or undersampling, or developing a cost-sensitive solution that can adjust the penalty for the degree of importance assigned to the minority case.

□ Combining physical laws and equations with machine learning to develop a physics-informed ML-based ionosphere model. Physics-informed ML can embed the knowledge of physical laws that govern a given data set into the learning process and can be described by partial differential equations (PDEs). It is often implemented using deep learning methods (see Raissi et al. (2019), Karniadakis et al. (2021)). Adding prior knowledge of physical laws will act as regularization during training to narrow the space of acceptable solutions and increase the correctness of the function approximation. In this way, embedding the prior information will improve the information content of the data and make it easier for the learning algorithm to capture the solution even with a small number of training examples. Physics-informed ML-based ionosphere models can also provide physically interpretable information, as researchers often criticize ML methods as being "black boxes".

List of Figures

List of Tables

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. https://www.tensorflow.org/. Software available from tensorflow.org

Abdullah, M., Strangeways, H. J., & Walsh, D. M. (2007). Effects of ionospheric horizontal gradients on differential gps. *Acta Geophysica*, 55, 509–523.

Adolfs, M. & Hoque, M. M. (2021). A neural network-based tec model capable of reproducing nighttime winter anomaly. *Remote Sensing*, 13(22). https://doi.org/https://doi.org/10.3390/rs13224559

Amell, A., Eriksson, P., & Pfreundschuh, S. (2022). Ice water path retrievals from meteosat-9 using quantile regression neural networks. *Atmospheric Measurement Techniques*, 15(19), 5701–5717. https://doi.org/10.5194/amt-15-5701-2022

Arras, C., Wickert, J., Beyerle, G., Heise, S., Schmidt, T., & Jacobi, C. (2008). A global climatology of ionospheric irregularities derived from gps radio occultation. *Geophysical Research Letters*, 35(14). https://doi.org/https://doi.org/10.1029/2008GL034158

Astafyeva, E., Zakharenkova, I., & Förster, M. (2015). Ionospheric response to the 2015 st. patrick's day storm: A global multi-instrumental overview. *Journal of Geophysical Research: Space Physics*, 120(10), 9023–9037. https://doi.org/https://doi.org/10.1002/2015JA021629

Auriol, A. & Tourain, C. (2010). Doris system: The new age. *Advances in Space Research*, 46(12), 1484–1496. https://doi.org/https://doi.org/10.1016/j.asr.2010.05.015. DORIS: Precise Orbit Determination and Applications to Earth Sciences

Badeke, R., Borries, C., Hoque, M. M., & Minkwitz, D. (2018). Empirical forecast of quiet time ionospheric total electron content maps over europe. *Advances in Space Research*, 61(12), 2881–2890. https://doi.org/https://doi.org/10.1016/j.asr.2018.04.010

Baker, D. N., Daly, E., Daglis, I., Kappenman, J. G., & Panasyuk, M. (2004). Effects of space weather on technology infrastructure. *Space Weather*, 2(2). https://doi.org/10.1029/2003SW000044

Barta, V., Natras, R., Srećković, V., Koronczay, D., Schmidt, M., & Šulic, D. (2022). Multi-instrumental investigation of the solar flares impact on the ionosphere on 05–06 december 2006. *Frontiers in Environmental Science*, 10. https://doi.org/10.3389/fenvs.2022.904335

Bartels, J. (1963). Discussion of time-variations of geomagnetic activity, indices kp and ap, 1932-1961. *Annales de Geophysique*, volume 19, 1.

Berdermann, J., Kriegel, M., Banyś, D., Heymann, F., Hoque, M. M., Wilken, V., Borries, C., Heßelbarth, A., & Jakowski, N. (2018). Ionospheric response to the x9.3 flare on 6 september 2017 and its implication for navigation services over europe. *Space Weather*, 16(10), 1604–1615. https://doi.org/https://doi.org/10.1029/2018SW001933

Bergeot, N., Bruyninx, C., Defraigne, P., Pireaux, S., Legrand, J., Pottiaux, E., & Baire, Q. (2011). Impact of the halloween 2003 ionospheric storm on kinematic gps positioning in europe. *GPS Solutions*, 15(2), 171–180. https://doi.org/10.1007/s10291-010-0181-9

Bilitza, D. (2018). Iri the international standard for the ionosphere. *Advances in Radio Science*, 16, 1–11. https://doi.org/10.5194/ars-16-1-2018

Blum, A., Kalai, A., & Langford, J. (1999). Beating the hold-out: Bounds for k-fold and progressive cross-validation. *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, COLT '99, 203–208. https://doi.org/10.1145/307400.307439

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, 1613–1622.

Bottou, L. (1991). Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, volume 91, 12.

Bottou, L. (2004). *Stochastic Learning*, 146–168. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-28650-9_7

Bottou, L. (2012). *Stochastic Gradient Descent Tricks*, 421–436. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-35289-8_25

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Breiman, L., Friedman, J., Stone, C., & Olshen, R. (1984). *Classification and Regression Trees*. Taylor & Francis.

Caamano, M., Juan, J. M., Felux, M., Gerbeth, D., González-Casado, G., & Sanz, J. (2021). Network-based ionospheric gradient monitoring to support gbas. *NAVIGATION: Journal of the Institute of Navigation*, 68(1), 135–156. https://doi.org/10.1002/navi.411

Camporeale, E., Wing, S., & Johnson, J. (2018). *Machine learning techniques for space weather*. Elsevier.

Cander, L. R. (2019). Ionospheric variability. *Ionospheric space weather*, 59–93. Springer. https://doi.org/10.1007/978-3-319-99331-7_4

Chan, J. S. K. (2021). Predicting loss reserves using quantile regression running title: Quantile regression loss reserve models. *Journal of Data Science*, 13(1), 127–156. https://doi.org/10.6339/JDS.201501_13(1).0008

Chelton, D. B., Ries, J. C., Haines, B. J., Fu, L.-L., & Callahan, P. S. (2001). Chapter 1 satellite altimetry. *Satellite Altimetry and Earth Sciences*, volume 69 of *International Geophysics*, 1–ii. Academic Press. https://doi.org/https://doi.org/10.1016/S0074-6142(01)80146-7

Chen, P., Liu, H., & Ma, Y. (2020). Empirical orthogonal function analysis and modeling of global ionospheric spherical harmonic coefficients. *GPS Solutions*, 24, 1–17.

Chen, T. & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 785–794.

Ciraolo, L., Azpilicueta, F., Brunini, C., Meza, A., & Radicella, S. M. (2007). Calibration errors on experimental slant total electron content (tec) determined with gps. *Journal of geodesy*, 81, 111–120.

Covington, A. E. (1969). Solar radio emission at 10.7 cm, 1947-1968. *Journal of the Royal Astronomical Society of Canada, Vol. 63, p. 125*, 63, 125.

Dach, R., Lutz, S., Walser, P., & Fridez, P. (2015). *Bernese gnss software version 5.2. user manual. astronomical institute, university of bern, bern*.

Dashora, N., Suresh, S., & Niranjan, K. (2019). Interhemispheric asymmetry in response of low-latitude ionosphere to perturbation electric fields in the main phase of geomagnetic storms. *Journal of Geophysical Research: Space Physics*, 124(8), 7256–7282. https://doi.org/https://doi.org/10.1029/2019JA026671

Dettmering, D., Limberger, M., & Schmidt, M. (2014). Using doris measurements for modeling the vertical total electron content of the earth's ionosphere. *Journal of Geodesy*, 88, 1131–1143.

Dettmering, D., Schmidt, M., Heinkelmann, R., & Seitz, M. (2011). Combination of different space-geodetic observations for regional ionosphere modeling. *Journal of geodesy*, 85, 989–998.

Dominique, M., Hochedez, J. F., Schmutz, W., Dammasch, I. E., Shapiro, A. I., Kretzschmar, M., Zhukov, A. N., Gillotay, D., Stockman, Y., & BenMoussa, A. (2013). The lyra instrument onboard proba2: Description and in-flight performance. *Solar Physics*, 286(1), 21–42. https://doi.org/10.1007/s11207-013-0252-5

Duan, T., Anand, A., Ding, D. Y., Thai, K. K., Basu, S., Ng, A., & Schuler, A. (2020). Ngboost: Natural gradient boosting for probabilistic prediction. *International Conference on Machine Learning*, 2690–2700.

Eastwood, J. P., Biffis, E., Hapgood, M. A., Green, L., Bisi, M. M., Bentley, R. D., Wicks, R., McKinnell, L.-A., Gibbs, M., & Burnett, C. (2017). The economic impact of space weather: Where do we stand? *Risk Analysis*, 37(2), 206–218. https://doi.org/10.1111/risa.12765

Erdogan, E., Schmidt, M., Goss, A., Görres, B., & Seitz, F. (2020). Adaptive modeling of the global ionosphere vertical total electron content. *Remote Sensing*, 12(11). https://doi.org/10.3390/rs12111822

Erdogan, E., Schmidt, M., Goss, A., Görres, B., & Seitz, F. (2021). Real-time monitoring of ionosphere vtec using multi-gnss carrier-phase observations and b-splines. *Space Weather*, 19(10), e2021SW002858. https://doi.org/https://doi.org/10.1029/2021SW002858

Erdogan, E., Schmidt, M., Seitz, F., & Durmaz, M. (2017). Near real-time estimation of ionosphere vertical total electron content from gnss satellites using b-splines in a kalman filter. *Annales Geophysicae*, 35(2), 263–277. https://doi.org/10.5194/angeo-35-263-2017

Esposito, D. (2020). *Introducing Machine Learning* (1st edition ed.). Microsoft Press; Safari.

Farzaneh, S. & Forootan, E. (2018). Reconstructing regional ionospheric electron density: A combined spherical slepian function and empirical orthogonal function approach. *Surveys in Geophysics*, 39(2), 289–309. https://doi.org/10.1007/s10712-017-9446-y

Feltens, J. (2003). The international gps service (igs) ionosphere working group. *Advances in Space Research*, 31(3), 635–644. https://doi.org/https://doi.org/10.1016/S0273-1177(03)00029-2. Description of the Low Latitude and Equatorial Ionosphere in the International Reference Ionosphere

Freund, Y. & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Friedli, T. K. (2016). Sunspot observations of rudolf wolf from 1849–1893. *Solar Physics*, 291(9), 2505–2517. https://doi.org/10.1007/s11207-016-0907-0

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5), 1189 – 1232. https://doi.org/10.1214/aos/1013203451

Gal, Y. & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of the 33rd International Conference on Machine Learning*, 1050–1059.

Ghaffari Razin, M. R. & Voosoghi, B. (2020). Ionosphere time series modeling using adaptive neuro-fuzzy inference system and principal component analysis. *GPS Solutions*, 24(2), 1–13. https://doi.org/https://doi.org/10.1007/s10291-020-0964-6

Goss, A., Schmidt, M., Erdogan, E., Görres, B., & Seitz, F. (2019). High-resolution vertical total electron content maps based on multi-scale b-spline representations. *Annales Geophysicae*, 37(4), 699–717. https://doi.org/10.5194/angeo-37-699-2019

Goss, A., Schmidt, M., Erdogan, E., & Seitz, F. (2020). Global and regional high-resolution vtec modelling using a two-step b-spline approach. *Remote Sensing*, 12(7). https://doi.org/10.3390/rs12071198

Goss, A. M. J. (2021). *Generation of high-resolution global and regional multi-scale B-spline models of the vertical total electron content based on low-latency GNSS data*. Technische Universität München.

Han, X., Dasgupta, S., & Ghosh, J. (2021). Simultaneously reconciled quantile forecasting of hierarchically related time series. *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, 190–198. https://proceedings.mlr.press/v130/han21a.html

Han, Y., Wang, L., Fu, W., Zhou, H., Li, T., & Chen, R. (2022). Machine learning-based short-term gps tec forecasting during high solar activity and magnetic storm periods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 115–126. https://doi.org/https://doi.org/10.1109/JSTARS.2021.3132049

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction* (2 ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7

Hernández-Pajares, M., García-Rigo, A., Juan, J. M., Sanz, J., Monte, E., & Aragón-Àngel, A. (2012a). Gnss measurement of euv photons flux rate during strong and mid solar flares. *Space Weather*, 10(12). https://doi.org/https://doi.org/10.1029/2012SW000826

Hernández-Pajares, M., García-Rigo, A., Juan, J. M., Sanz, J., Monte, E., & Aragón-Àngel, A. (2012b). Gnss measurement of euv photons flux rate during strong and mid solar flares. *Space Weather*, 10(12). https://doi.org/https://doi.org/10.1029/2012SW000826

Hernández-Pajares, M., Juan, J. M., Sanz, J., Aragón-Àngel, , García-Rigo, A., Salazar, D., & Escudero, M. (2011). The ionosphere: effects, gps modeling and the benefits for space geodetic techniques. *Journal of Geodesy*, 85(12), 887–907. https://doi.org/10.1007/s00190-011-0508-5

Hernández-Pajares, M., Juan, J. M., Sanz, J., Orus, R., Garcia-Rigo, A., Feltens, J., Komjathy, A., Schaer, S. C., & Krankowski, A. (2009). The igs vtec maps: a reliable source of ionospheric information since 1998. *Journal of Geodesy*, 83(3), 263–275. https://doi.org/10.1007/s00190-008-0266-1

Hernández-Pajares, M., Roma-Dollase, D., Krankowski, A., García-Rigo, A., & Orús-Pérez, R. (2017). Methodology and consistency of slant and vertical assessments for ionospheric electron content models. *Journal of Geodesy*, 91(12), 1405–1414. https://doi.org/10.1007/s00190-017-1032-z

Hoque, M. M. & Jakowski, N. (2008). Estimate of higher order ionospheric errors in gnss positioning. *Radio Science*, 43(5). https://doi.org/https://doi.org/10.1029/2007RS003817

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4, 251–257.

Hudson, H. S. (2021). Carrington events. *Annual Review of Astronomy and Astrophysics*, 59, 445–477.

Hüllermeier, E. & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3), 457–506. https://doi.org/https://doi.org/10.1007/s10994-021-05946-3

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Jacobsen, K. S. & Andalsvik, Y. L. (2016). Overview of the 2015 st. patrick´s day storm and its consequences for rtk and ppp positioning in norway. *J. Space Weather Space Clim.*, 6, A9. https://doi.org/10.1051/swsc/2016004

Jakowski, N., Wehrenpfennig, A., Heise, S., Reigber, C., Lühr, H., Grunwaldt, L., & Meehan, T. K. (2002). Gps radio occultation measurements of the ionosphere from champ: Early results. *Geophysical Research Letters*, 29(10), 95–1–95–4. https://doi.org/https://doi.org/10.1029/2001GL014364

Jayles, C., Chauveau, J., & Rozo, F. (2010). Doris/jason-2: Better than 10cm on-board orbits available for near-real-time altimetry. *Advances in Space Research*, 46(12), 1497–1512. https://doi.org/https://doi.org/10.1016/j.asr.2010.04.030. DORIS: Precise Orbit Determination and Applications to Earth Sciences

Jiang, H., Wang, Z., An, J., Liu, J., Wang, N., & Li, H. (2017). Influence of spatial gradients on ionospheric mapping using thin layer models. *GPS Solutions*, 22(1), 2. https://doi.org/10.1007/s10291-017-0671-0

Kaiser, M. L. & Adams, W. J. (2007). Stereo mission overview. *2007 IEEE Aerospace Conference*, 1–8. https://doi.org/10.1109/AERO.2007.352745

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, 3(6), 422–440.

Kaselimi, M., Voulodimos, A., Doulamis, N., Doulamis, A., & Delikaraoglou, D. (2022). Deep recurrent neural networks for ionospheric variations estimation using gnss measurements. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1–15. https://doi.org/https://doi.org/10.1109/TGRS.2021.3090856

Kiani Shahvandi, M. & Soja, B. (2022). Inclusion of data uncertainty in machine learning and its application in geodetic data science, with case studies for the prediction of earth orientation parameters and gnss station coordinate time series. *Advances in Space Research*, 70(3), 563–575. https://doi.org/https://doi.org/10.1016/j.asr.2022.05.042

Kintner, P. M., Ledvina, B. M., & de Paula, E. R. (2007). Gps and ionospheric scintillations. *Space Weather*, 5(9). https://doi.org/https://doi.org/10.1029/2006SW000260

Klobuchar, J. A. (1987). Ionospheric time-delay algorithm for single-frequency gps users. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(3), 325–331. https://doi.org/https://doi.org/10.1109/TAES.1987.310829

Knipp, D. J. & Gannon, J. L. (2019). The 2019 national space weather strategy and action plan and beyond. *Space Weather*, 17(6), 794–795. https://doi.org/10.1029/2019SW002254

Koch, K.-R. (1999). *Parameter Estimation in Linear Models*, 149–269. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-03976-2_4

Koch, K. R. (2018). Bayesian statistics and monte carlo methods. *Journal of Geodetic Science*, 8(1), 18–29. https://doi.org/doi:10.1515/jogs-2018-0003

Krankowski, A., Kosek, W., Baran, L., & Popinski, W. (2005). Wavelet analysis and forecasting of vtec obtained with gps observations over european latitudes. *Journal of Atmospheric and Solar-Terrestrial Physics*, 67(12), 1147–1156. https://doi.org/https://doi.org/10.1016/j.jastp.2005.03.004. Space weather and RF communications: monitoring and modelling

Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4), 221–232. https://doi.org/10.1007/s13748-016-0094-0

Kumar, A., Chakrabarty, D., Pandey, K., & Yadav, A. K. (2022). Solar flux dependence of post-sunset enhancement in vertical total electron content over the crest region of equatorial ionization anomaly. *Journal of Geophysical Research: Space Physics*, 127(5), e2021JA030156. https://doi.org/https://doi.org/10.1029/2021JA030156

Kutiev, I., Otsuka, Y., Saito, A., & Tsugawa, T. (2007). Low-latitude total electron content enhancement at low geomagnetic activity observed over japan. *Journal of Geophysical Research: Space Physics*, 112(A7). https://doi.org/https://doi.org/10.1029/2007JA012385

Lalgudi Gopalakrishnan, G. & Schmidt, M. (2022). Ionospheric electron density modelling using b-splines and constraint optimization. *Earth, Planets and Space*, 74(1), 1–23.

Lanzerotti, L. J. (2017). Space weather: Historical and contemporary perspectives. *Space Science Reviews*, 212(3), 1253–1270. https://doi.org/10.1007/s11214-017-0408-y

Lee, S., Ji, E.-Y., Moon, Y.-J., & Park, E. (2020). One-day forecasting of global tec using a novel deep learning model. *Space Weather*, 19(1), 2020SW002600. https://doi.org/10.1029/2020SW002600

Li, Z., Wang, N., Hernández-Pajares, M., Yuan, Y., Krankowski, A., Liu, A., Zha, J., García-Rigo, A., Roma-Dollase, D., Yang, H., Laurichesse, D., & Blot, A. (2020). Igs real-time service for global ionospheric total electron content modeling. *Journal of Geodesy*, 94(3), 32. https://doi.org/10.1007/s00190-020-01360-0

Limberger, M. (2015). *Ionosphere modeling from GPS radio occultations and complementary data based on B-splines*. Technische Universität München.

Linty, N., Minetto, A., Dovis, F., & Spogli, L. (2018). Effects of phase scintillation on the gnss positioning error during the september 2017 storm at svalbard. *Space Weather*, 16(9), 1317–1329. https://doi.org/https://doi.org/10.1029/2018SW001940

Liu, J., Zhang, D., Mo, X., Xiong, C., Hao, Y., & Xiao, Z. (2020a). Morphological differences of the northern equatorial ionization anomaly between the eastern asian and american sectors. *Journal of Geophysical Research: Space Physics*, 125(3), e2019JA027506. https://doi.org/https://doi.org/10.1029/2019JA027506. e2019JA027506 2019JA027506

Liu, J. Y., Lin, C. H., Tsai, H. F., & Liou, Y. A. (2004). Ionospheric solar flare effects monitored by the ground-based gps receivers: Theory and observation. *Journal of Geophysical Research: Space Physics*, 109(A1). https://doi.org/https://doi.org/10.1029/2003JA009931

Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020b). Forecasting global ionospheric tec using deep learning approach. *Space Weather*, 18(11), e2020SW002501. https://doi.org/https://doi.org/10.1029/2020SW002501

Liu, Q., Hernández-Pajares, M., Lyu, H., & Goss, A. (2021). Influence of temporal resolution on the performance of global ionospheric maps. *Journal of Geodesy*, 95(3), 34. https://doi.org/10.1007/s00190-021-01483-y

Liu, Y. & Wu, Y. (2011). Simultaneous multiple non-crossing quantile regression estimation using kernel constraints. *Journal of Nonparametric Statistics*, 23(2), 415–437. https://doi.org/10.1080/10485252.2010.537

Luo, X., Gu, S., Lou, Y., Xiong, C., Chen, B., & Jin, X. (2018a). Assessing the performance of gps precise point positioning under different geomagnetic storm conditions during solar cycle 24. *Sensors*, 18(6). https://doi.org/10.3390/s18061784

Luo, X., Gu, S., Lou, Y., Xiong, C., Chen, B., & Jin, X. (2018b). Assessing the performance of gps precise point positioning under different geomagnetic storm conditions during solar cycle 24. *Sensors*, 18(6). https://doi.org/10.3390/s18061784

Lyu, H., Hernández-Pajares, M., Nohutcu, M., García-Rigo, A., Zhang, H., & Liu, J. (2018). The barcelona ionospheric mapping function (bimf) and its application to northern mid-latitudes. *Gps solutions*, 22(3), 67.

Magnet, N. (2019). *Giomo: A robust modeling approach of ionospheric delays for GNSS real-time positioning applications*. Wien.

Malinin, A., Prokhorenkova, L., & Ustimenko, A. (2021). Uncertainty in gradient boosting via ensembles. *International Conference on Learning Representations*. https://openreview.net/forum?id=1Jv6b0Zq3qi

Mannucci, A. J., Wilson, B. D., Yuan, D. N., Ho, C. H., Lindqwister, U. J., & Runge, T. F. (1998). A global mapping technique for gps-derived ionospheric total electron content measurements. *Radio Science*, 33(3), 565–582. https://doi.org/https://doi.org/10.1029/97RS02707

MATLAB (2020). *version 2020a*. The MathWorks Inc.

Matzka, J., Stolle, C., Yamazaki, Y., Bronkalla, O., & Morschhauser, A. (2021). The geomagnetic kp index and derived indices of geomagnetic activity. *Space Weather*, 19(5), e2020SW002641. https://doi.org/https://doi.org/10.1029/2020SW002641

Milligan, R. O., Hudson, H. S., Chamberlin, P. C., Hannah, I. G., & Hayes, L. A. (2020). Lyman-alpha variability during solar flares over solar cycle 24 using goes-15/euvs-e. *Space Weather*, 18(7), e2019SW002331. https://doi.org/https://doi.org/10.1029/2019SW002331. e2019SW002331 10.1029/2019SW002331

Mitchell, T. (1997). Machine learning.

Monte-Moreno, E., Yang, H., & Hernández-Pajares, M. (2022). Forecast of the global tec by nearest neighbour technique. *Remote Sensing*, 14(6). https://doi.org/https://doi.org/10.3390/rs14061361

Mosqueira-Rey, E., Hernández-Pereira, E., Alonso-Ríos, D., Bobes-Bascarán, J., & Fernández-Leal, Á. (2023). Human-in-the-loop machine learning: A state of the art. *Artificial Intelligence Review*, 56(4), 3005–3054.

Murphy, K. P. (2012). *Machine learning : a probabilistic perspective*. MIT Press.

Müller, D., Marsden, R. G., St. Cyr, O. C., Gilbert, H. R., & The Solar Orbiter, T. (2013). Solar orbiter. *Solar Physics*, 285(1), 25–70. https://doi.org/10.1007/s11207-012-0085-7

Natras, R. (2023a). *Randa-lab/Bayesian_Neural_Network_Probabilistic_Io nosphere_VTEC: Bayesian_Neural_Network_Probabilistic_Ionosphere*. https://doi.org/10.5281/zenodo.7858906

Natras, R. (2023b). *Randa-lab/Quantile_Gradient_Boosting_for_Probabili stic_VTEC: Quantile_Gradient_Boosting_Probabilisti c_Ionosphere_Evaluation*. https://doi.org/10.5281/zenodo.7858661

Natras, R., Goss, A., Halilovic, D., Magnet, N., Mulic, M., Schmidt, M., & Weber, R. (2023a). Regional ionosphere delay models based on cors data and machine learning. *NAVIGATION: Journal of the Institute of Navigation*, 70(3). https://doi.org/10.33012/navi.577

Natras, R., Halilovic, D., Mulić, M., Schmidt, Michael", e. N., Mujčić, E., Mulić, M., Kevrić, J., & Akšamija, Z. (2023b). Mid-latitude ionosphere variability (2013–2016), and space weather impact on vtec and precise point positioning. *Advanced Technologies, Systems, and Applications VII*, 471–491. https://doi.org/https://doi.org/10.1007/978-3-031-17697-5_37

Natras, R., Horozovic, D., & Mulic, M. (2019a). Strong solar flare detection and its impact on ionospheric layers and on coordinates accuracy in the western balkans in october 2014. *SN Applied Sciences*, 1(49), 1–14. https://doi.org/https://doi.org/10.1007/s42452-018-0040-9

Natras, R., Krdzalic, D., Horozovic, D., Tabakovic, A., & Mulic, M. (2019b). Gnss ionospheric tec and positioning accuracy during intense space and terrestrial weather events in b&h. *Geodetski vestnik*, (1). https://doi.org/10.15292/geodetski-vestnik.2019.01.73-91

Natras, R. & Schmidt, M. (2021). Machine learning model development for space weather forecasting in the ionosphere. *CEUR Workshop Proceedings of the CIKM 2021 Workshops co-located with 30th ACM International Conference on Information and Knowledge Management (CIKM 2021)*, volume 3052. http://ceur-ws.org/Vol-3052/short10.pdf

Natras, R., Soja, B., & Schmidt, M. (2022a). Ensemble machine learning of random forest, adaboost and xgboost for vertical total electron content forecasting. *Remote Sensing*, 14(15). https://doi.org/https://doi.org/10.3390/rs14153547

Natras, R., Soja, B., & Schmidt, M. (2022b). Machine learning ensemble approach for ionosphere and space weather forecasting with uncertainty quantification. *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, 1–4. https://doi.org/https://doi.org/10.23919/AT-AP-RASC54737.2022.9814334

Natras, R., Soja, B., & Schmidt, M. (2023c). *Dataset of Machine Learning forecasted VTEC from paper: Uncertainty Quantification for Machine Learning-based Ionosphere and Space Weather Forecasting*. https://doi.org/10.5281/zenodo.7741342

Natras, R., Soja, B., & Schmidt, M. (2023d). Uncertainty quantification for machine learning-based ionosphere and space weather forecasting: Ensemble, bayesian neural network, and quantile gradient boosting. *Space Weather*, 21(10), e2023SW003483. https://doi.org/https://doi.org/10.1029/2023SW003483. e2023SW003483 2023SW003483

Nava, B., Coïsson, P., & Radicella, S. (2008). A new version of the nequick ionosphere electron density model. *Journal of Atmospheric and Solar-Terrestrial Physics*, 70(15), 1856–1862. https://doi.org/https://doi.org/10.1016/j.jastp.2008.01.015. Ionospheric Effects and Telecommunications

Nava, B., Rodríguez-Zuluaga, J., Alazo-Cuartas, K., Kashcheyev, A., Migoya-Orué, Y., Radicella, S., Amory-Mazaudier, C., & Fleury, R. (2016). Middle- and low-latitude ionosphere response to 2015st. patrick's day geomagnetic storm. *Journal of Geophysical Research: Space Physics*, 121(4), 3421–3438. https://doi.org/https://doi.org/10.1002/2015JA022299

Neal, R. M. (2012). *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media.

Odijk, D. (2000). Improving ambiguity resolution by applying ionosphere corrections from a permanent gps array. *Earth, planets and space*, 52, 675–680.

Olivares-Pulido, G., Teferle, F. N., & Hunegnaw, A. (2020). *Markov Chain Monte Carlo and the Application to Geodetic Time Series Analysis*, 53–138. Springer International Publishing. https://doi.org/10.1007/978-3-030-21718-1_3

Orus Perez, R., Parro-Jimenez, J. M., & Prieto-Cerdeira, R. (2018). Status of nequick g after the solar maximum of cycle 24. *Radio Science*, 53(3), 257–268. https://doi.org/https://doi.org/10.1002/2017RS006373

Orús, R., Hernández-Pajares, M., Juan, J., Sanz, J., & Garcıa-Fernández, M. (2002). Performance of different tec models to provide gps ionospheric corrections. *Journal of Atmospheric and Solar-Terrestrial Physics*, 64(18), 2055–2062. https://doi.org/https://doi.org/10.1016/S1364-6826(02)00224-9

Pandit, D., Ghimire, B., Amory-Mazaudier, C., Fleury, R., Chapagain, N. P., & Adhikari, B. (2021). Climatology of ionosphere over nepal based on gps total electron content data from 2008 to 2018. *Annales Geophysicae*, 39(4), 743–758. https://doi.org/10.5194/angeo-39-743-2021

Parkinson, Bradford W., S. J. J. J. A. P. E. P. (1996). *Global Positioning System, Volume 1 - Theory and Applications*. American Institute of Aeronautics and Astronautics (AIAA). https://app.knovel.com/hotlink/toc/id:kpGPSVTA03/global-positioning-system/global-positioning-system

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Petit, G. & Luzum, B. (2010). Iers conventions (2010). Report, Bureau International des Poids et mesures sevres (france).

Poniatowski, M. & Nykiel, G. (2020). Degradation of kinematic ppp of gnss stations in central europe caused by medium-scale traveling ionospheric disturbances during the st. patrick's day 2015 geomagnetic storm. *Remote Sensing*, 12(21). https://doi.org/10.3390/rs12213582

Psychas, D., Verhagen, S., Liu, X., Memarzadeh, Y., & Visser, H. (2018). Assessment of ionospheric corrections for ppp-rtk using regional ionosphere modelling. *Measurement Science and Technology*, 30(1), 014001. https://doi.org/10.1088/1361-6501/aaefe5

Pulkkinen, A., Lindahl, S., Viljanen, A., & Pirjola, R. (2005). Geomagnetic storm of 29–31 october 2003: Geomagnetically induced currents and their relation to problems in the swedish high-voltage power transmission system. *Space Weather*, 3(8). https://doi.org/https://doi.org/10.1029/2004SW000123

Qian, L., Burns, A. G., Chamberlin, P. C., & Solomon, S. C. (2010). Flare location on the solar disk: Modeling the thermosphere and ionosphere response. *Journal of Geophysical Research: Space Physics*, 115(A9). https://doi.org/https://doi.org/10.1029/2009JA015225

Radicella, S. M. & Nava, B. (2020). Chapter 6 - empirical ionospheric models. *The Dynamical Ionosphere*, 39–53. Elsevier. https://doi.org/https://doi.org/10.1016/B978-0-12-814782-5.00006-6

Raissi, M., Perdikaris, P., & Karniadakis, G. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378, 686–707. https://doi.org/https://doi.org/10.1016/j.jcp.2018.10.045

Ridley, A., Deng, Y., & Tóth, G. (2006). The global ionosphere–thermosphere model. *Journal of Atmospheric and Solar-Terrestrial Physics*, 68(8), 839–864. https://doi.org/https://doi.org/10.1016/j.jastp.2006.01.008

Roma-Dollase, D., Hernández-Pajares, M., Krankowski, A., Kotulak, K., Ghoddousi-Fard, R., Yuan, Y., Li, Z., Zhang, H., Shi, C., Wang, C., Feltens, J., Vergados, P., Komjathy, A., Schaer, S., García-Rigo, A., & Gómez-Cama, J. M. (2018). Consistency of seven different gnss global ionospheric mapping techniques during one solar cycle. *Journal of Geodesy*, 92(6), 691–706. https://doi.org/10.1007/s00190-017-1088-9

Ruwali, A., Kumar, A. J. S., Prakash, K. B., Sivavaraprasad, G., & Ratnam, D. V. (2021). Implementation of hybrid deep learning model (lstm-cnn) for ionospheric tec forecasting using gps data. *IEEE Geoscience and Remote Sensing Letters*, 18(6), 1004–1008. https://doi.org/https://doi.org/10.1109/LGRS.2020.2992633

Schaer, S. (1999). *Mapping and predicting the Earth's ionosphere using the Global Positioning System*, volume 59. Institut für Geodäsie und Photogrammetrie, Eidg. Technische Hochschule . . . .

Schaer, S., Beutler, G., Mervart, L., Rothacher, M., & Wild, U. (1996). Global and regional ionosphere models using the gps double difference phase observable. *Proceedings of the IGS Workshop*.

Schmidt, M. (2012). Towards a multi-scale representation of multi-dimensional signals. *VII Hotine-Marussi Symposium on Mathematical Geodesy*, 119–127.

Schmidt, M., Dettmering, D., Mößmer, M., Wang, Y., & Zhang, J. (2011). Comparison of spherical harmonic and b spline models for the vertical total electron content. *Radio Science*, 46(6). https://doi.org/https://doi.org/10.1029/2010RS004609

Schmidt, M., Dettmering, D., & Seitz, F. (2015). *Using B-Spline Expansions for Ionosphere Modeling*, 939–983. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-54551-1_80

Schunk, R. W., Scherliess, L., Sojka, J. J., Thompson, D. C., Anderson, D. N., Codrescu, M., Minter, C., Fuller-Rowell, T. J., Heelis, R. A., Hairston, M., & Howe, B. M. (2004). Global assimilation of ionospheric measurements (gaim). *Radio Science*, 39(1). https://doi.org/https://doi.org/10.1029/2002RS002794

Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12), 310–316.

Shimeis, A., Amory-Mazaudier, C., Fleury, R., Mahrous, A., & Hassan, A. (2014). Transient variations of vertical total electron content over some african stations from 2002 to 2012. *Advances in Space Research*, 54(11), 2159–2171. https://doi.org/https://doi.org/10.1016/j.asr.2014.07.038

Sholokhov, A., Berkovich, S., & Kotov, N. (2020). Nonlinear estimation of navigation and geodetic parameter on the basis of the point-mass method taking into account the statistical relationship of node weights. *2020 27th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, 1–4. https://doi.org/10.23919/ICINS43215.2020.9133936

Srivani, I., Siva Vara Prasad, G., & Venkata Ratnam, D. (2019). A deep learning-based approach to forecast ionospheric delays for gps signals. *IEEE Geoscience and Remote Sensing Letters*, 16(8), 1180–1184. https://doi.org/https://doi.org/10.1109/LGRS.2019.2895112

Sugiura, M. (1964). Hourly values of equatorial dst for the igy. *Ann. Int. Geophys. Yr.*, Vol: 35. https://www.osti.gov/biblio/4554034

Supriadi, S., Abidin, H. Z., Wijaya, D. D., Abadi, P., Saito, S., & Prabowo, D. U. (2022). Construction of nominal ionospheric gradient using satellite pair based on gnss cors observation in indonesia. *Earth, Planets and Space*, 74(1), 71.

Tagasovska, N. & Lopez-Paz, D. (2019). Single-model uncertainties for deep learning. *Advances in Neural Information Processing Systems*, volume 32. https://proceedings.neurips.cc/paper/2019/file/73c03186765e199c116224b68adc5fa0-Paper.pdf

Tang, R., Zeng, F., Chen, Z., Wang, J.-S., Huang, C.-M., & Wu, Z. (2020a). The comparison of predicting storm-time ionospheric tec by three methods: Arima, lstm, and seq2seq. *Atmosphere*, 11(4). https://doi.org/10.3390/atmos11040316

Tang, R., Zeng, F., Chen, Z., Wang, J.-S., Huang, C.-M., & Wu, Z. (2020b). The comparison of predicting storm-time ionospheric tec by three methods: Arima, lstm, and seq2seq. *Atmosphere*, 11(4). https://doi.org/10.3390/atmos11040316

Tapping, K. F. (1987). Recent solar radio astronomy at centimeter wavelengths: The temporal variability of the 10.7-cm flux. *Journal of Geophysical Research: Atmospheres*, 92(D1), 829–838. https://doi.org/https://doi.org/10.1029/JD092iD01p00829

Tapping, K. F. (2013). The 10.7cm solar radio flux (f10.7). *Space Weather*, 11(7), 394–406. https://doi.org/https://doi.org/10.1002/swe.20064

Tebabal, A., Radicella, S., Damtie, B., Migoya-Orue', Y., Nigussie, M., & Nava, B. (2019). Feed forward neural network based ionospheric model for the east african region. *Journal of Atmospheric and Solar-Terrestrial Physics*, 191, 105052. https://doi.org/https://doi.org/10.1016/j.jastp.2019.05.016

Todorova, S., Hobiger, T., & Schuh, H. (2008). Using the global navigation satellite system and satellite altimetry for combined global ionosphere maps. *Advances in Space Research*, 42(4), 727–736. https://doi.org/https://doi.org/10.1016/j.asr.2007.08.024

Tsurutani, B. T., Verkhoglyadova, O. P., Mannucci, A. J., Lakhina, G. S., & Huba, J. D. (2012). Extreme changes in the dayside ionosphere during a carrington-type magnetic storm. *J. Space Weather Space Clim.*, 2, A05. https://doi.org/10.1051/swsc/2012004

Uwamahoro, J. C. & Habarulema, J. B. (2015). Modelling total electron content during geomagnetic storm conditions using empirical orthogonal functions and neural networks. *Journal of Geophysical Research: Space Physics*, 120(12), 11,000–11,012. https://doi.org/10.1002/2015JA021961

Vaishnav, R., Schmölter, E., Jacobi, C., Berdermann, J., & Codrescu, M. (2021). Ionospheric response to solar extreme ultraviolet radiation variations: comparison based on ctipe model simulations and satellite measurements. *Annales Geophysicae*, 39(2), 341–355. https://doi.org/10.5194/angeo-39-341-2021

Wang, J., Huang, G., Zhou, P., Yang, Y., Zhang, Q., & Gao, Y. (2020). Advantages of uncombined precise point positioning with fixed ambiguity resolution for slant total electron content (stec) and differential code bias (dcb) estimation. *Remote Sensing*, 12(2). https://doi.org/10.3390/rs12020304

Wanliss, J. A. & Showalter, K. M. (2006a). High-resolution global storm index: Dst versus sym-h. *Journal of Geophysical Research: Space Physics*, 111(A2). https://doi.org/https://doi.org/10.1029/2005JA011034

Wanliss, J. A. & Showalter, K. M. (2006b). High-resolution global storm index: Dst versus sym-h. *Journal of Geophysical Research: Space Physics*, 111(A2). https://doi.org/https://doi.org/10.1029/2005JA011034

Wanninger, L. (1995). Improved ambiguity resolution by regional differential modelling of the ionosphere. *Proceedings of the 8th international technical meeting of the satellite division of the institute of navigation (ION GPS 1995)*, 55–62.

Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60), 3021. https://doi.org/10.21105/joss.03021

Wickert, J., Reigber, C., Beyerle, G., König, R., Marquardt, C., Schmidt, T., Grunwaldt, L., Galas, R., Meehan, T. K., Melbourne, W. G., & Hocke, K. (2001). Atmosphere sounding by gps radio occultation: First results from champ. *Geophysical Research Letters*, 28(17), 3263–3266. https://doi.org/https://doi.org/10.1029/2001GL013117

Wong, T.-T. & Yeh, P.-Y. (2020). Reliable accuracy estimates from k-fold cross validation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8), 1586–1594. https://doi.org/10.1109/TKDE.2019.2912815

Wyszkowska, P. (2017). Propagation of uncertainty by monte carlo simulations in case of basic geodetic computations. *Geodesy and Cartography*, vol. 66(No 2). https://doi.org/10.1515/geocart-2017-0022

Xia, G., Liu, Y., Wei, T., Wang, Z., Huang, W., Du, Z., Zhang, Z., Wang, X., & Zhou, C. (2021). Ionospheric tec forecast model based on support vector machine with gpu acceleration in the china region. *Advances in Space Research*, 68(3), 1377–1389. https://doi.org/https://doi.org/10.1016/j.asr.2021.03.021

Xiang, Y. & Gao, Y. (2019). An enhanced mapping function with ionospheric varying height. *Remote Sensing*, 11(12). https://doi.org/10.3390/rs11121497

Xiong, P., Zhai, D., Long, C., Zhou, H., Zhang, X., & Shen, X. (2021). Long short-term memory neural network for ionospheric total electron content forecasting over china. *Space Weather*, 19(4), e2020SW002706. https://doi.org/https://doi.org/10.1029/2020SW002706

Yasyukevich, Y., Astafyeva, E., Padokhin, A., Ivanova, V., Syrovatskii, S., & Podlesnyi, A. (2018). The 6 september 2017 x-class solar flares and their impacts on the ionosphere, gnss, and hf radio wave propagation. *Space Weather*, 16(8), 1013–1027. https://doi.org/https://doi.org/10.1029/2018SW001932

Zakharenkova, I. & Cherniak, I. (2021). Effects of storm-induced equatorial plasma bubbles on gps-based kinematic positioning at equatorial and middle latitudes during the september 7–8, 2017, geomagnetic storm. *GPS Solutions*, 25(4). https://doi.org/10.1007/s10291-021-01166-3

Zhang, R., Gao, C., Wang, Z., Zhao, Q., Shang, R., Peng, Z., & Liu, Q. (2022a). Ambiguity resolution for long baseline in a network with bds-3 quad-frequency ionosphere-weighted model. *Remote Sensing*, 14(7). https://doi.org/10.3390/rs14071654

Zhang, S., Wu, X., & Hu, X. (2022b). The variation characteristics and prediction performance of tec in the geomagnetic latitude and local time coordinate. *Radio Science*, 57(12), e2022RS007544. https://doi.org/https://doi.org/10.1029/2022RS007544

Zhang, S.-R., Holt, J. M., van Eyken, A. P., McCready, M., Amory-Mazaudier, C., Fukao, S., & Sulzer, M. (2005). Ionospheric local model and climatology from long-term databases of multiple incoherent scatter radars. *Geophysical Research Letters*, 32(20). https://doi.org/https://doi.org/10.1029/2005GL023603

Zhang, X., Ren, X., Chen, J., Zuo, X., Mei, D., & Liu, W. (2022c). Investigating gnss ppp–rtk with external ionospheric constraints. *Satellite Navigation*, 3(1), 1–13.

Zhukov, A. V., Yasyukevich, Y. V., & Bykov, A. E. (2021). Correction to: Gimli: Global ionospheric total electron content model based on machine learning. *GPS Solutions*, 25(1), 21. https://doi.org/10.1007/s10291-020-01063-1

Zou, C., Ding, H., & Wang, L. (2022). A scr method for uncertainty estimation in geodesy non-linear error propagation: Comparisons and applications. *Geodesy and Geodynamics*, 13(4), 311–320. https://doi.org/https://doi.org/10.1016/j.geog.2021.11.003

Software

Within the scope of this dissertation, several software packages and tools were developed in MATLAB and Python. Special attention was given to the consideration of open-source software. The exception is that for data processing in **P-I**, commercial Bernese GNSS Software and Operational Tool for Ionosphere Mapping and Prediction (OPTIMAP) software were used. Below is a list of the software used in the first-author journal papers **P-I**, **P-II**, and **P-III**, with corresponding references and links.

- **Bernese GNSS Software** (Dach et al., 2015): scientific, high-precision, multi-GNSS data processing software.
  Source: `http://www.bernese.unibe.ch/`
- **QGIS**: Quantum Geographic Information System, free and open source geographic information system.
  Source: `https://qgis.org/en/site/`
- **Scikit-learn** (Pedregosa et al., 2011): free software machine learning library for the Python programming language.
  Source: `https://scikit-learn.org/`
- **TensorFlow** (Abadi et al., 2015): free and open-source software library for machine learning and artificial intelligence for the Python programming language.
  Source: `https://www.tensorflow.org/`
- **Seaborn** (Waskom, 2021): Statistical data visualization library for the Python programming language.
  Source: `https://seaborn.pydata.org/`
- **Matplotlib** (Hunter, 2007): plotting library for the Python programming language.
  Source: `https://matplotlib.org/`
- **Matlab** (MATLAB, 2020): programming and numeric computing platform.
  Source: `https://uk.mathworks.com/products/matlab.html`
- **NumPy**: Numerical Python for scientific computing.
  Source: `https://numpy.org/`
- **Pandas**: Python Data Analysis Library.
  Source: `https://pandas.pydata.org/`

The software developed in **P-III** is openly available through Zenodo and GitHub. The probabilistic VTEC forecast results for the year 2017 based on the four uncertainty quantification approaches from **P-III** are available under the Creative Commons Attribution 4.0 International license on Zenodo (Natras et al., 2023c). The software used to develop, i.e., train, cross-validate, and test the BNN VTEC models, as well as the training, the cross-validation, and the test data sets can be found in Natras (2023a). The developed QGB VTEC models and software for loading and evaluating them against the test data set are provided in Natras (2023b).

Appendix

In the following, a classification of the contributions by R. Natras in percent for each first-author journal publication **P-I**, **P-II**, and **P-III** is given in Table 1.

Table 1: Contribution of Randa Natras to the first-author journal publications

	Weight	Estimated contribution		
Paper		**P-I**	**P-II**	**P-III**
Idea and Conceptual Design	30%	80%	85%	90%
Computation and Realization	10%	70%	85%	85%
Analysis, Interpretation and Conclusions	30%	80%	85%	85%
Figure and Table Compilation	10%	85%	95%	95%
Manuscript Writing	20%	80%	85%	90%
Estimated Total Contribution		80%	86%	89%

In order to estimate the own contribution to each paper, percentage estimates are given with respect to five contribution criteria. It should be noted that the percentages are only approximations. A percentage rating of the total own contribution to each publication was estimated as a weighted average of the five contribution criteria, taking into account the significance of the contribution.

Statement from the doctoral candidate concerning his*her own contribution for publications submitted within the framework of a publication-based dissertation, as well as Statement of Confirmation from the co-authors according to Article 7, section 3, clause 2 TUM Doctoral Regulations (PromO)

Name of doctoral candidate:

Randa Natras

Publication 1 (title of publication; complete list of authors; journal incl. year and edition number; page numbers):

Natras, R., Goss, A., Halilovic, D., Magnet, N., Mulic, M., Schmidt, M., and Weber, R. (2023a). **Regional Ionosphere Delay Models Based on CORS Data and Machine Learning**. NAVIGATION: Journal of the Institute of Navigation, 70 (3) navi.577. https://doi.org/10.33012/navi.577

	Overview of own contribution with percentages
Idea and Conceptual Design	80%
Computation and Realization	70%
Analysis, Interpretation and Conclusions	80%
Figure and Table Compilation	85%
Manuscript Writing	80%
Estimated Total Contribution	**80%**

Publication 2 (title of publication; complete list of authors; journal incl. year and edition number; page numbers):

Natras, R., Soja, B., and Schmidt, M. (2022a). **Ensemble Machine Learning of Random Forest, AdaBoost and XGBoost for Vertical Total Electron Content Forecasting**. Remote Sensing, 14(15): 3547. https://doi.org/10.3390/rs14153547

	Overview of own contribution with percentages
Idea and Conceptual Design	85%
Computation and Realization	85%
Analysis, Interpretation and Conclusions	85%
Figure and Table Compilation	95%
Manuscript Writing	85%
Estimated Total Contribution	**86%**

Publication 3 (title of publication; complete list of authors; journal incl. year and edition number; page numbers):

Natras, R., Soja, B., and Schmidt, M. (2023). **Uncertainty Quantification for Machine Learning-based Ionosphere and Space Weather Forecasting**. Space Weather, AGU (under review)

	Overview of own contribution with percentages
Idea and Conceptual Design	90%
Computation and Realization	85%
Analysis, Interpretation and Conclusions	85%
Figure and Table Compilation	95%
Manuscript Writing	90%
Estimated Total Contribution	**89%**

Munich, 20.07.2023. [redacted]

Place, date, signature of doctoral candidates

As a co-author, I herewith confirm that the above Statement of Own Contribution is correct. I agree to the usage of the publication for the above-mentioned dissertation project within the framework of a publication-based dissertation.

Name co-author	Date and signature
Andreas Goss	21.07.2023 [redacted]
Dzana Halilovic	25.07.2023 [redacted]
Nina Magnet	24.07.2023 [redacted]
Medzida Mulic	26.07.2023. [redacted]
Michael Schmidt	20.7.2023 [redacted]
Benedikt Soja	20.07.2023 [redacted]
Robert Weber	21.7.2023 [redacted]

Please note: As an alternative to the original signed version, a scanned copy of the signed statement is also acceptable.

A1 Publication I

Reference

Natras, R., Goss, A., Halilovic, D., Magnet, N., Mulic, M., Schmidt, M., and Weber, R. (2023a). **Regional Ionosphere Delay Models Based on CORS Data and Machine Learning**. NAVIGATION: Journal of the Institute of Navigation, 70 (3) navi.577. https://doi.org/10.33012/navi.577

Copyright

This paper was published in the open access journal NAVIGATION: Journal of the Institute of Navigation. It is available under the Creative Commons license at https://navi.ion.org/. The copyrights remain with the authors.

Declaration of own contribution

The concept of the paper was proposed by R. Natras and discussed with all co-authors. R. Natras processed the GNSS data in Bernese GNSS software version v.5.2 (Dach et al., 2015) with the valuable support of the co-authors. The MATLAB software for regional ionosphere modeling using Taylor series expansions was partially developed by R. Natras. The OTHR model data used for the validation were generated by A. Goss in the OPTIMAP processing software. R. Natras proposed the machine learning methodology. The machine learning-based regional ionosphere models were developed by R. Natras, and their computations are based on the Python programming language and the TensorFlow/Keras open-source software platform, machine learning library (Abadi et al., 2015). The numerical investigations and validations were performed by R. Natras together with co-authors. R. Natras compiled the figures and tables with contributions and suggestions from co-authors. The figures were created in MATLAB (MATLAB, 2020). The manuscript was written by R. Natras and reviewed and edited by all co-authors.
The overall contribution of R. Natras to P-I is estimated to be 80%.

Received: 22 February 2021 | Revised: 31 October 2022 | Accepted: 10 December 2022

DOI: 10.33012/navi.577

ORIGINAL ARTICLE

Regional Ionosphere Delay Models Based on CORS Data and Machine Learning

Randa Natras[1] | **Andreas Goss**[1] | **Dzana Halilovic**[3] | **Nina Magnet**[2] | **Medzida Mulic**[4] | **Michael Schmidt**[1] | **Robert Weber**[3]

[1] Deutsches Geodätisches Forschungsinstitut der Technischen Universität München (DGFI-TUM), Department of Aerospace and Geodesy, Technical University of Munich, Munich, 80333, Germany

[2] OHB Digital Solutions GmbH, Graz, 8044, Austria

[3] Department of Geodesy and Geoinformation, Vienna University of Technology, Vienna, 1040, Austria

[4] Department of Geodesy and Geoinformation, University of Sarajevo, Sarajevo, 71000, Bosnia-Herzegovina

Correspondence
Randa Natras
Deutsches Geodätisches Forschungsinstitut der Technischen Universität München (DGFI-TUM)
Department of Aerospace and Geodesy
Technical University of Munich
Munich, 80333
Germany,
Email: randa.natras@tum.de

Abstract

The ionospheric refraction of GNSS signals can have an impact on positioning accuracy, especially in cases of single-frequency observations. Ionosphere models that are broadcasted by the satellite systems (e.g., Klobuchar, NeQuick-G) do not include enough details to permit them to correct single-frequency observations with sufficient accuracy. To address this issue, regional ionosphere models (RIMs) have been developed in several countries in the western Balkans based on dense Continuous Operating Reference Stations (CORS) observations. Subsequently, a RIM for the western Balkans was built using an artificial neural network that combined regional ionosphere parameters estimated from the CORS data with spatiotemporal (latitude, longitude, hour of day), solar (F10.7) and geomagnetic (Kp, Dst) parameters. The RIMs were tested at the solar maximum (March 2014), a geomagnetic storm (March 2015), and the solar minimum (March 2018). The new RIMs mimic the integrated electron density much more effectively than the Klobuchar model. Furthermore, RIMs significantly reduce the ionospheric effects on single-frequency positioning, indicating their necessity for use in positioning applications.

Keywords

artificial neural network, ionosphere delay modeling, machine learning, regional ionosphere model, single-frequency positioning, vertical total electron content

1 | INTRODUCTION

The ionized upper part of the Earth's atmosphere known as the ionosphere affects the propagation of radio waves that are generated by communication and navigation systems. Consequently, ionospheric refraction can affect the accuracy and reliability of positioning applications that rely on global navigation satellite systems (GNSS). Dual-frequency GNSS observations facilitate the estimation of ionospheric effects by forming the geometry-free linear combination (L_4) as well as reductions of most of the ionospheric range error via an ionosphere-free linear combination (L_3) (Hofmann-Wellenhof et al., 2001). However, the ionosphere remains one of the major sources of error in single-frequency positioning, where the first-order ionospheric term accounts for more than 99.9% of the total ionospheric delay associated with phase and code GNSS measurements (Hernández-Pajares et al., 2011). Because mass-market GNSS receivers commonly operate on a single frequency, the ionospheric range error needs to be corrected or at least mitigated by deploying models that minimize the ionospheric effects.

Due to the complex nature of these processes as well as the solar-terrestrial coupling system, different approaches for modeling the ionosphere have been developed. Current models of the ionosphere can be categorized as physical, empirical, or mathematical (Farzaneh & Forootan, 2018). State-of-the-art methods utilize artificial intelligence, specifically machine learning techniques, to identify nonlinear relationships among the variables to improve forecasting, especially factors related to space-weather processes (Camporeale et al., 2018; Natras & Schmidt, 2021). Physical ionosphere models are based on physical and chemical processes in the ionosphere, as shown by the Global Assimilation of Ionospheric Measurements (GAIM) model (Schunk et al., 2004) and the Global Ionosphere-Thermosphere Model (GITM) (Ridley et al., 2006). These models rely on observations and mathematical representations of physical laws and can thus be quite complicated, as they require formidable numerical procedures with high computational costs. By contrast, empirical models describe the ionosphere with mathematical functions derived from historical observational data and statistics (Radicella & Nava, 2020). These models represent average conditions and regular variations of the ionosphere (i.e., its “climate”). Examples of such climatological empirical models are the International Reference Ionosphere (IRI) (Bilitza, 2018) and NeQuick (Nava et al., 2008).

To correct the ionospheric delay in single-frequency observations, navigation satellite systems broadcast coefficients within the navigation message that are based on these empirical approaches. These models have been widely applied largely due to their simplicity. For instance, the well-known Klobuchar model (Klobuchar, 1987) that was adopted in the global positioning system (GPS) eliminates at least 50% of the ionospheric range delay error. Similarly, a special version of the NeQuick model denoted as NeQuick-G that has been implemented in Galileo can correct approximately 70% of the ionospheric code delay (Orus Perez et al., 2018).

The ionospheric refraction of the single-frequency observations can be modeled more precisely using GNSS observations to estimate the total electron content (TEC) along the signal path within the ionosphere; this value is proportional to the ionospheric refraction range. These models are typically based on estimations of the vertical TEC (VTEC) on a global scale using different mathematical approaches. For example, global ionosphere maps (GIMs) are routinely generated by Ionosphere Associated Analysis Centers (IAACs) of the International GNSS Service (IGS) including CODE (Center for Orbit Determination in Europe, Astronomical Institute, University of Bern, Switzerland), ESOC/ESA (European Space Operations Center from European Space Agency, Darmstadt, Germany), JPL (Jet Propulsion Laboratory, Pasadena, California, USA), UPC (Universitat Politècnica de Catalunya; Technical University of Catalonia, Spain), NRCan (Canadian Geodetic Survey of Natural Resources Canada), WHU (Wuhan University, China), CAS (Chinese Academy of Sciences, China) and the OPTIMAP-Group, DGFI-TUM (Deutsches Geodätisches Forschungsinstitut der Technischen Universität München; German Geodetic Research Institute of the Technical University of Munich, Germany). The global VTEC distribution can be represented mathematically in the CODE maps by spherical harmonics (Schaer, 1999) with a spatial sampling of 2.5° × 5° in latitude and longitude, respectively, and a temporal resolution of two hours until the year 2015 and one hour thereafter. The GIM products of the IAACs are used to generate a combined solution of the IGS (Hernández-Pajares et al., 2009), which generally has global relative errors of 10% to 20% compared to the VTEC estimated from topography experiment (TOPEX) satellite altimeter missions observations (Orús et al., 2002). The GNSS-derived ionosphere VTEC models and maps are useful external sources that provide information for single-frequency GNSS users seeking to mitigate the first-order ionospheric delay and directly reduce the ionospheric range error (IERS conventions, 2010). The final GIM products commonly feature a time delay of 1–2 weeks, while the rapid GIMs are generated with a latency of 1–2 days (Li et al., 2020; Liu et al., 2021). The latency of GIM products can limit their use in real-time positioning applications.

In contrast to these models, regional ionosphere VTEC models (RIMs) may be more accurate as they have higher spatial and in some cases, also higher temporal resolution because they can incorporate observations from dense GNSS networks. The RIMs are based on a different set of mathematical approaches for VTEC representation. For example, in Europe, the Royal Observatory of Belgium (ROB) generates RIMs in near real-time by applying thin plate spline interpolation (Bergeot et al., 2014). Other approaches for regional VTEC mapping in Europe include series expansions in tensor products of polynomial B-spline functions (Goss et al., 2020), weighting functions that take into account the location and the magnitude of the global electron maximum (Magnet, 2019), approximations of the ionosphere single layer model with Taylor expansions of degree two (Boisits et al., 2020), among others. It is critical to recognize that most of the RIMs were developed for large regions, such as the European continent.

Regarding the temporal resolution of the ionospheric products, use of the slant TEC (dSTEC) RMS can result in improvements of approximately 13% and 20% when the temporal sampling is increased from 2 hrs to 1 hr for low and high-resolution global ionosphere B-spline models, respectively, (Goss et al., 2019). However, increasing the temporal sampling from 1h to 10 min resulted in minimal improvements of only 3% to 4% for low and high-resolution global ionosphere B-spline products, respectively. Further investigation of global VTEC products in different latitudinal ranges (Liu et al., 2021) revealed no differences in the standard deviations of the GIMs at 1 hr, 45 min, 30 min, and 15 min temporal resolutions in the mid-latitudinal range (30° N to 50° N) with regard to VTEC from the Joint Altimetry Satellite Oceanography Network (JASON) satellite altimeter mission. By increasing the temporal resolution from 1 h to 15 min, the RMS of the dSTEC assessment decreased by less than 0.01 TECU in the mid-latitudinal range (30° N to 50° N). By contrast, an improvement of approximately 0.2 TECU in standard deviation and of 0.5 TECU in the RMS was obtained when the temporal resolution was increased from two hrs to one hr.

Recently, there has been increasing interest in state-of-the-art machine learning applications for global and regional ionosphere modeling. Among these studies, a feed-forward neural network was applied for regional VTEC modeling over Nigeria (Okoh et al., 2016) that was based on inputs of geomagnetic latitude and longitude, year, day of the year, the hour of the day, Dst index, sunspot number, and critical plasma frequency (foF2) from the International Reference Ionosphere (IRI) model. Other applications include regional VTEC modeling over Brazil using inputs of latitude and longitude (Leandro & Santos, 2007), global VTEC modeling with inputs of latitude and longitude, day of the year, F10.7 and Kp indices (Orus Perez, 2019), regional VTEC modeling over South Africa using the day of the year, the hour of the day, a four-month running mean of sunspot number and the running mean of the previous eight hours of the Ap index (Habarulema et al., 2009), as well as others. Zhao et al. (2021) used spherical harmonics with a neural network based on an extreme machine-learning technique for real-time modeling of the ionospheric delay. Similarly, Zhang et al. (2019) successfully combined the machine learning algorithm of a support vector machine with the regional VTEC polynomial model based on the CORS observations. Liu et al. (2020) utilized a type of recurrent neural network known as long short-term memory (LSTM) to forecast global VTEC maps; the LSTM model processes sequences of past observations of spherical harmonic coefficients, extreme ultraviolet (EUV) flux, and Dst index, among others to understand temporal-dependent relationships. Kaselimi et al. (2020) also applied an LSTM model with inputs that included satellite position coordinates, the azimuth and the elevation angles of the satellite, and the output of the VTEC. The wavelet neural network for VTEC modeling (El-Diasty, 2017) combines the inputs of the day of the year, the hour of the day, latitude, and longitude of the ionosphere pierce point (IPP) with the estimated Klobuchar model at the same IPP point and VTEC output data for the model training estimated from the CORS stations observations. Another application of support vector machines extrapolates VTEC in regions where GNSS observations are not available (Kim & Kim, 2019). The inputs include the hour of the day, the day of the year, F10.7 and Kp indices, sunspot number, and VTEC from areas where GNSS observations were available to provide estimates for VTEC for regions devoid of GNSS observations (Kim & Kim, 2019). Also, an ensemble of tree-based meta-estimators has been developed by combining random forest, adaptive boosting (AdaBoost), and extreme gradient boosting (XGBoost) methods for VTEC forecasting during both calm and stormy geomagnetic conditions (Natras et al., 2022a) and estimating forecast uncertainties as an ensemble spread (Natras et al., 2022b). Various machine-learning approaches and applications have been proposed. The input data can include spatial information (latitude and longitude of the station or the IPP or the satellite position, among others), temporal information (year, day of year, hour of day), solar activity (sunspot number, F10.7, EUV flux, and others), geomagnetic activity (Kp and Dst) as well as ionosphere information (IRI, Klobuchar, GNSS-derived VTEC, VTEC polynomial model, and spherical harmonic coefficients). Most previous studies employed a fully-connected feed-forward neural network. Regarding the size of the training dataset, different lengths of time were used in previous studies, including one day (Zhang et al., 2019; Zhao et al., 2021), four days (El-Diasty, 2017; Kaselimi et al., 2020), ten days (Leandro & Santos, 2007), one to two years (Kim & Kim, 2019; Liu et al., 2020; Natras et al., 2022a), and four years (Habarulema et al., 2009; Okoh et al., 2016; Orus Perez, 2019) among others. Results from previous studies demonstrated the feasibility of machine learning for VTEC estimation, with a focus on artificial neural networks and also when applied to a limited dataset of several days. While large datasets are typically perceived as essential for training neural networks, it is possible to train these models with relatively small amounts of data (Motamedi et al., 2021; Rajpurkar et al., 2020). The lack of large datasets will need to be compensated by high-quality data. This represents the core of a data-centric approach in machine learning (Motamedi et al., 2021). In these cases, data quality is more important than the dataset size, i.e., shifting from big data to good data. This facilitates effective training of neural networks with smaller datasets.

Most of the aforementioned ionosphere models rely on GPS or GPS + GLONASS observations from the IGS, the European EUREF Permanent Network (EPN), and/or the CORS network as a source of VTEC information. The yellow area in Figure 1 shows the approximate location of the Balkan Peninsula (36° N to 48° N, 13° E to 30° E). The distribution of IGS and EPN tracking ground stations is poorer in this region compared to the rest of Europe. However, most countries in this region operate dense CORS networks. This is also a case in the countries in the western part of the Balkan Peninsula, whose

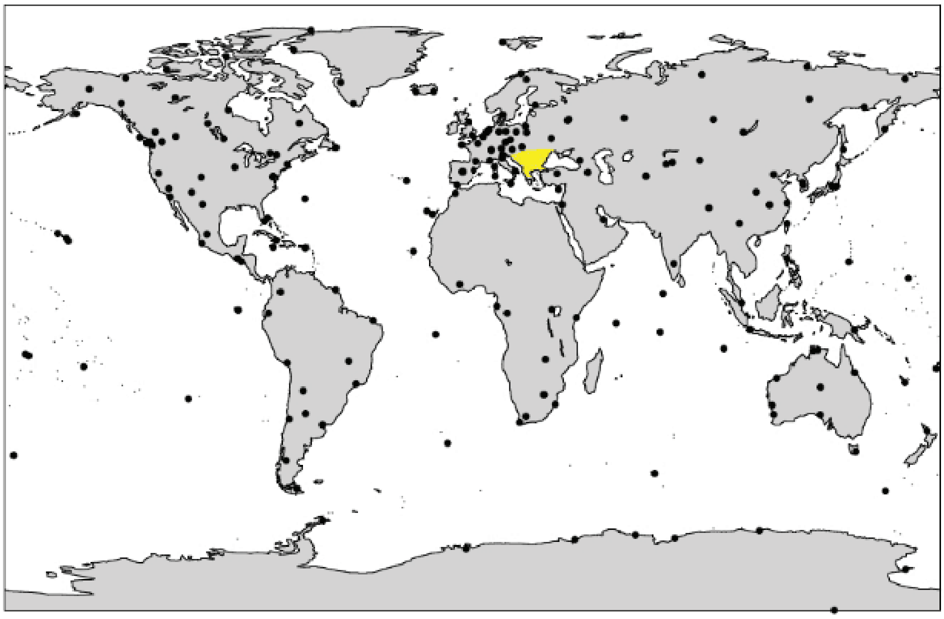


FIGURE 1 Locations of GNSS tracking ground stations used by CODE to produce GIMs (obtained from Jee et al. (2010)).

observations have not been previously used to generate ionosphere VTEC models. These networks provide denser GNSS observation coverage for VTEC modeling in the western Balkans.

This research aims to extend current knowledge in the field of RIM development based on GNSS observations by generating ionosphere models for a region that is much smaller than most continents using information available from dense CORS networks. With this in mind, this paper describes the development, validation, and applicability of regional GNSS-based ionosphere models while accounting for ionosphere effects in positioning applications. We will determine how to use data from CORS networks for regional/national VTEC modeling and identify the advantages that VTEC models developed for small regions can bring to positioning applications, using the western Balkans as a test case example. Therefore, the aim of the study is not to describe the physical and/or chemical processes associated with the ionosphere constituents, but instead to generate models that take into account and mitigate the ionospheric effects that hinder GNSS/GPS positioning utilizing available CORS data. Until now, no regional ionosphere model based on the national GNSS infrastructure has been established or developed in the countries in the western Balkans region. To address this knowledge gap, this paper presents three new regional ionosphere models based primarily on the observations from the CORS networks: (i) a first model developed for a small region inside one country; (ii) a second model developed for several countries within the western Balkans; and (iii) a third model that covers the entire western region of the Balkan Peninsula. As part of the third model, we propose an approach based on state-of-the-art machine learning techniques including the use of an artificial neural network. These models will be evaluated in single-frequency precise point positioning. To the best of our knowledge, this is one of the first studies to use observations from the CORS networks located in the western part of the Balkan Peninsula to generate GNSS-based ionosphere models and to evaluate them in positioning applications to mitigate the effects of ionospheric refraction.

2 | METHODOLOGY

The impact of the ionosphere on the propagation of radio waves propagation can be described by the STEC in Equation (1):

$$STEC = \int_{k}^{i} N_e(s)ds \tag{1}$$

where $N_e(s)$ is the electron density along the signal ray path between the satellite i and the receiver k. STEC is measured in TEC units (TECU), where 1 TECU= 10^{16} electrons/m^2. The vertical TEC (VTEC) can be expressed as shown in Equation (2):

$$VTEC = STEC \cdot \cos z' \tag{2}$$

where z' is the zenith angle of the signal path in a mean altitude H of the ionospheric shell. The ionosphere is approximated as the single-layer model (SLM), which assumes that all free electrons are concentrated within a shell of infinitesimal thickness. The SLM mapping function $F(z)$ can be written as shown in Equation (3):

$$F(z) = \frac{STEC}{VTEC} = \frac{1}{\cos z'} \qquad \text{with} \qquad \sin z' = \frac{R}{R+H} \cdot \sin z \tag{3}$$

where z is the zenith angle at the height of the receiver, R is the mean Earth radius and H is the aforementioned height of the SLM above the Earth's surface (Schaer, 1999). The SLM height typically ranges between 350 km to 450 km (Jiang et al., 2017; Mannucci et al., 1998; Schaer, 1999). IGS analysis centers have adopted a thin layer height of 450 km for the ionosphere products (Feltens, 2003). In this study, we also adopted the SLM height of 450 km above the Earth's surface.

2.1 | Selection of Study Region and Data

Figure 2 shows the locations of stations that belong to the CORS (blue dots) and the EPN (red dots) networks whose dual-frequency GPS observations were used in this study to generate VTEC models. Two VTEC models were developed based on the regions covered, namely, RIM IONO_BH and RIM IONO_WB.

To estimate the RIM IONO_BH, the station network was selected to include nine Bosnia and Herzegovina Positioning Service (BIHPOS) stations each located approximately 80 km from the central EPN station in Sarajevo. An orange ellipse on the map marks this area. The RIM IONO_WB was derived from GPS observations obtained from the following CORS networks: the Albanian GNSS Permanent Stations (AlbGNSS), BIHPOS, the Croatian Positioning System (CROPOS), the Macedonian Positioning System (MAKPOS), and the Slovenia-Geodesy-Navigation-Location (SIGNAL). We also used observations from eight EPN stations within this region. Observations from the CORS networks located in Serbia and Kosovo were not available from the network providers and are not freely available to the public; thus this information was not used in this study. Also, historical data from the CORS network in Montenegro were not preserved for these study periods and thus not available for use in this study. Consequently, CORS observations from these countries were not included in the RIM IONO_WB. The network selected for the RIM IONO_WB included approximately 80 CORS and EPN stations between approximately 40° N to 47° N and 13° E to 23° E. RIM IONO_WB was developed as an independent

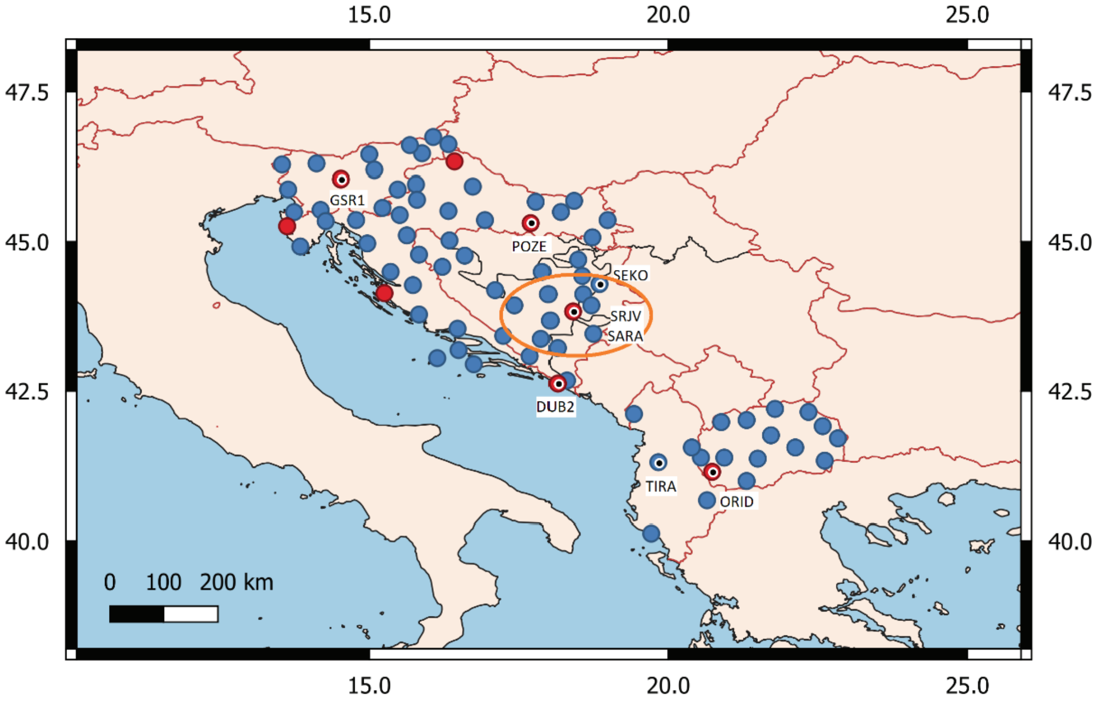


FIGURE 2 Map documenting the locations of the dual-frequency stations of the CORS (blue dots) and the EPN (red dots) networks whose observations were used for VTEC modeling. Stations used for the RIM validation are highlighted with an inner black dot with a white rim; their names are indicated on the map.

model that would be applicable for each of the participating countries mentioned above. Therefore, the RIM IONO_WB includes files with the extension "ION" that contain separate sets of ionospheric corrections for each country. The reference point of the Taylor series expansion was set approximately in the mid-range of each country included in the study. Permanent stations that were used to validate the newly-developed RIMs are highlighted on the map with inner black and white circles (Figure 2). These stations were not used for estimating the regional models. The following stations shown on the map were used to validate the RIM IONO_BH: EPN SRJV and BIHPOS SARA (which are very close to one another), BIHPOS SEKO, and, additionally, two stations outside the modeled region, EPN POZE and EPN DUB2. Stations used for the RIM IONO_WB and the RIM IONO_WB_AI validation included EPN GSR1, EPN POZE, EPN SRJV, IGEWE TIRA, and EPN ORID.

2.2 | Regional Ionosphere Modeling with Bernese GNSS Software

Bernese GNSS Software version v.5.2 (Dach et al., 2015) was used to process GNSS data, estimate ionosphere models and perform positioning. Before generating ionosphere models, several pre-processing steps must be carried out. Table 1 summarizes routines and the processing steps according to the order of their execution. Additional details on each routine and processing step in the Bernese GNSS software are described by Dach et al. (2015). The ionosphere models were estimated within the IONEST routine using the Bernese-recommended values (Dach et al., 2015) listed in Table 2. This routine works only on the assumption of GPS zero-difference observations.

TABLE 1
Processing Steps in the Bernese GNSS Software

Steps	Bernese software routine
Cutting 24-hour RINEX observations into 1-hour files	CCRINEXO
Orbit and Earth's orientation information preparation	POLUPD, PRETAB, ORBGEN
Satellite clock correction files preparation	RNXCLK, CCRNXC
Import of RINEX observation data into the Bernese format	RXOBV3
Receiver clock synchronization	CODSPP
Ionosphere model estimation	IONEST

TABLE 2
The Specific Options for the VTEC Modeling (IONEST Routine)

Pre-processing and processing options	
Linear combination for break detection (data cleaning)	L_4
A priori sigma of a single observation	0.01 m
Elevation cut-off angle	15°
Height of the single layer	450 km
Degree of Taylor series expansion in latitude (n_{max})	1
Degree of Taylor series expansion in hour angle (m_{max})	2

Ionosphere mapping was performed on the undifferenced (zero-difference) level by analyzing the so-called geometry-free linear combination (L_4) of phase observations which are formed by subtracting observables at different frequencies. This means that all frequency-independent effects such as the satellite-receiver geometrical range, clock errors, and tropospheric delay, among others, were canceled out, while the ionospheric effect remained (Ciraolo et al., 2007). Thus, the geometry-free linear combination includes the ionospheric delay and can be used to estimate the ionosphere model. With this approach, the carrier phases are not fitted to code (pseudo-range) observations; this means that no code-phase leveling was applied.

The observation equation for zero-difference phase observations can be outlined as shown in Equation (4):

$$L_4 = L_1 - L_2 = -\alpha \cdot \left(\frac{1}{f_1^2} - \frac{1}{f_2^2}\right) \cdot F(z) \cdot VTEC(\phi_{IPP}, s_{IPP}) - \beta \cdot \phi + B_4 \tag{4}$$

where L_4 is the geometry-free phase observable, $\alpha = 4.03 \cdot 10^{17}\,\mathrm{ms^{-2}TECU^{-1}}$ is a constant, f_1, f_2 are the frequencies associated with the carriers L_1 and L_2, $\beta \cdot \phi$ represents the wind-up term associated with the right-handed polarized GPS signal (typically a centimeter-level term), $B_4 = \lambda_1 \cdot B_1 - \lambda_2 \cdot B_2$ is a constant phase bias caused by the initial phase ambiguities (i.e, unknown integer number of cycles) B_1 and B_2 with their corresponding wavelengths λ_1 and λ_2. The initial phase ambiguities B_1 and B_2 are estimated as real-value parameters within the initial least-squares adjustment using the phase observations L_1 and L_2 on both frequencies. The initial phase ambiguity has the same value provided that no loss of signal lock occurs. At least one parameter (B_4) has to be solved for each receiver and satellite pass since it contains phase ambiguities. For cycle slip detection, differences between two satellites for the same epoch are formed from the measurements in the observation files. If cycle slips are detected, the geometry–free linear combination L_4 is checked in order to determine their size in both frequencies. A new ambiguity is set up at the corresponding epoch of the detected cycle slip. Pre-processing options were set in the IONEST routine to define a new ambiguity parameter B_4 for each detected cycle slip. The resulting geometry-free linear combination in Equation (4) allows us to estimate VTEC as a function of geographic latitude ϕ_{IPP} and the Sun-fixed longitude s_{IPP} at the intersection point of the line-of-sight between the satellite and the receiver with the ionospheric layer. This intersection point is known as the ionospheric pierce point (IPP).

The regional VTEC model is based on the two-dimensional (2-D) Taylor series expansion shown in Equation (5):

$$VTEC(\phi_{IPP}, s_{IPP}) = \sum_{n=0}^{n_{\max}} \sum_{m=0}^{m_{\max}} c_{nm} (\phi_{IPP} - \phi_0)^n \; (s_{IPP} - s_o)^m \tag{5}$$

where n, m are the degree values of the 2-D Taylor series expansion in geographic latitude and Sun-fixed longitude, $n_{\max}$, $m_{\max}$ are the maximum degree values of the Taylor series expansion in geographic latitude and Sun-fixed longitude (Table 2), ϕ_0, s_0 denote the coordinates of the origin of the Taylor expansion and c_{nm} stands for the unknown coefficients of the Taylor series expansion, i.e., the regional ionosphere model parameters to be estimated (Dach et al., 2015; Wild, 1994).

The Sun-fixed longitude s is related to the local solar time (LT) as shown in Equation (6):

$$s_{IPP} = LT - \pi \approx UT + \lambda_{IPP} - \pi \tag{6}$$

where UT is an abbreviation for the Universal Time, LT is local time, and λ_{IPP} denotes the geographical longitude of the IPP; all values are in radians.

The latitude ϕ_{IPP} and the longitude λ_{IPP} as shown in Equation (7) and Equation (8):

$$\phi_{IPP} = \sin^{-1}(\sin\phi_k \cos\psi + \cos\phi_k \sin\psi \cos A_z) \tag{7}$$

$$\lambda_{IPP} = \lambda_k + \sin^{-1}\left(\frac{\sin\psi \sin A_z}{\cos\phi_{IPP}}\right) \tag{8}$$

where ϕ_k, λ_k are the latitude and longitude of the receiver k, A_z stands for the azimuth from the receiver to the satellite, and ψ is the angle between the lines joining the center of the Earth with the IPP and receiver location. It can be calculated as shown in Equation (9):

$$\psi = \frac{\pi}{2} - E_l - \sin^{-1}\left(\frac{R}{R+H}\cos E_l\right) \tag{9}$$

where E_l stands for the elevation angle.

Ionosphere models were derived with a temporal sampling of one hour with the Bernese default extension ION and later combined into a 24-hour file for each day, namely the RIM IONO_BH and RIM IONO_WB, as described in Section 2.1.

When applying the ionosphere model presented in Equation (5), the ionospheric range corrections (in meters) for the zero-difference GPS observations of the i-th frequency can be computed as shown in Equation (10):

$$\Delta_i(\phi_{IPP}, s_{IPP}, z) = \pm\frac{\alpha}{f_i^2} F(z) VTEC(\phi_{IPP}, s_{IPP}) \qquad \text{with} \quad i = 1, 2 \tag{10}$$

where the negative sign is used for phase observations and the positive sign for code observations.

2.3 | Regional Ionosphere Modeling with Machine Learning

Machine learning represents a branch of artificial intelligence (AI) that can approximate the function between inputs and outputs based on rules/relationships that an AI system has "learned" from the data during the learning/training phase. This characteristic distinguishes machine learning from traditional modeling/programming approaches which require an extensive list of rules describing relationships between inputs and outputs to be explicitly specified (Natras & Schmidt, 2021). More specifically, machine learning addresses the problem of finding an approximation function that maps inputs (called predictors, features, or independent variables) to one or more outputs (called responses or dependent variables). This study focuses on the estimation of a single output (VTEC). Therefore, the output is presented below as a single-column vector. To start, a dataset was prepared that contained the measurements of the input feature $\ddot{\boldsymbol{x}}_j$ $(j=1, 2, \ldots, p)$ and the output $\boldsymbol{y}$ for a set of observations i $(i=1, 2, \ldots, n)$ as indicated in Equation (11):

$$\mathbf{X}=\begin{bmatrix}\boldsymbol{x}_1^T\\ \boldsymbol{x}_2^T\\ \vdots\\ \boldsymbol{x}_n^T\end{bmatrix}=\begin{bmatrix}x_{1,0} & \cdots & x_{1,p}\\ x_{2,0} & \cdots & x_{2,p}\\ \vdots & \ddots & \vdots\\ x_{n,0} & \cdots & x_{n,p}\end{bmatrix}\quad \ddot{\boldsymbol{x}}_j=\begin{bmatrix}x_{1,j}\\ x_{2,j}\\ \vdots\\ x_{n,j}\end{bmatrix}\quad \boldsymbol{y}=\begin{bmatrix}y_1\\ y_2\\ \vdots\\ y_n\end{bmatrix} \tag{11}$$

Artificial neural networks (ANNs) represent a state-of-the-art technique with widespread applications in many fields. The ANN usually consists of an input layer with input variables, an output layer with output variables, and one or more hidden layers between them. Hidden layers allow the model to create a complex nonlinear mapping function between the network inputs and outputs by introducing a nonlinear activation function (Sharma et al., 2020). To estimate the values of the neurons in a current layer, parameters (weights) need to be added to the values of the neurons from the previous layers. The weight matrices can be defined as shown in Equation (12):

$$\mathbf{W}=\begin{bmatrix}\mathbf{W}^{(1)}\\ \mathbf{W}^{(2)}\\ \vdots\\ \mathbf{W}^{(K-1)}\end{bmatrix}\qquad \mathbf{W}^{(k-1)}=\begin{bmatrix}\boldsymbol{w}_1^T\\ \boldsymbol{w}_2^T\\ \vdots\\ \boldsymbol{w}_L^T\end{bmatrix}=\begin{bmatrix}w_{1,0} & \cdots & w_{1,p}\\ w_{2,0} & \cdots & w_{2,p}\\ \vdots & \ddots & \vdots\\ w_{L,0} & \cdots & w_{L,p}\end{bmatrix} \tag{12}$$

Where the matrix $\mathbf{W}$ contains all weights in the neural network. $\mathbf{W}^{(k-1)}$ are weight matrices assigned to p neurons from the previous layer *(k–1)* in order to estimate values of L neuron(s) in the current hidden or output layer (k) with $k=(2, 3, \ldots, K)$, where $k = 1$ corresponds to the input layer. $w_{l,o}$ is the bias term $(l=1, 2, \ldots, L)$. In the matrix $\mathbf{X}$ we added an element $x_{i,o}=1$ to each row. It is convenient to include this element, because of the bias term $\boldsymbol{w}_o$ as this will permit us to present Equation (13) in vector form as a scalar product. Thus, the linear model will predict the output y_i given a vector of input features $\boldsymbol{x}_i$ and weights $\boldsymbol{w}_l$ with $l=1$, i.e., the model consists of an input and an output layer, which can be expressed as shown in Equation (13):

$$y_i + e_i = \hat{y}_i = f(\boldsymbol{x}_i, \boldsymbol{w}_1) = \sum\nolimits_{j=0}^{p} w_{1,j} x_{i,j} = \boldsymbol{w}_1^T \boldsymbol{x}_i \tag{13}$$

where e_i is an error. The weights in $\mathbf{W}$ are determined during the training phase using the method of least squares to minimize the sum of the squares of the error e_i and as shown in Equation (14):

$$R(\mathbf{W}) = \sum\nolimits_{i=1}^{n} e_i^2 = \sum\nolimits_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{14}$$

In a fully connected network, all neurons in one layer are connected to all neurons in the next layer; this is also known as a multilayer perceptron network (Ramchoun et al., 2016). A feed-forward network indicates that the connections between the neurons are all in one direction (from input to output) and hence the information is fed-forward from one layer to the next. Afterward, the backpropagation is used to adjust the weights of the ANN using a stochastic gradient descent method (Bottou, 1991) to minimize $R(\mathbf{W})$, as shown in Equation (14). In this study, we tested different setups of an ANN architecture and inputs. These specifications are presented in Table 3. In total, six ANN and two Random Forest (RF) models have been developed. One ANN model does not contain a hidden layer (ANN1); all other ANN models have hidden layers. The ANN1 model is an example of the linear model, where the outputs are estimated as described by Equation (13). The architecture of the ANN5 model is shown in Figure 3. ANN5 is a fully-connected feed-forward network with an input layer, three hidden layers, and an output layer. The number of neurons in the first layer corresponds to the length of the vector $\boldsymbol{x}_j$. Neurons in each layer can be referred to as activation neuron vector $\boldsymbol{a}_i^{(k)}$ ($k = 1, \ldots, K$). The input layer can be expressed as $\boldsymbol{a}_i^{(1)} = \boldsymbol{x}_i$. Hidden layers compute the derived features from $\boldsymbol{a}_i^{(2)}$ to $\boldsymbol{a}_i^{(K-1)}$. The last layer provides the output $\widehat{y}_i = \boldsymbol{a}_i^{(K)}$. Activation neurons in the hidden layer can be calculated as described in Equation (15):

$$\boldsymbol{a}_i^{(k)} = q(\boldsymbol{z}) = q\left(\boldsymbol{W}^{(k-1)^T} \boldsymbol{a}_i^{(k-1)}\right) \tag{15}$$

where q is the activation function (Hastie et al., 2009). The activation function used in hidden layers is known as a ReLu (Rectified Linear unit) function (Sharma et al., 2020). ReLu is a non-linear activation function that is widely used in deep learning models because of its simplicity and effectiveness. It is defined in Equation (16):

$$q(\boldsymbol{z}) = \max(0, \boldsymbol{z}) \tag{16}$$

meaning that the function will generate a value of zero for any negative value of $\boldsymbol{z}$; for any positive value, it will return that value. The network output $\widehat{y}_i$ can be defined as shown in Equation (17):

$$\widehat{y}_i = \boldsymbol{a}_i^{(K)} = q\left(\boldsymbol{W}^{(K-1)^T} \boldsymbol{a}_i^{(K-1)}\right). \tag{17}$$

The optimal hyper-parameters (number of neurons, number of hidden layers, learning rate for the stochastic gradient descent, and so on) were selected by testing different values within a certain range and observing the changes in $R(\mathbf{W})$. Hyperparameters were searched within the following intervals: number of hidden units = [5, 10, 20, 30, 40, 60], number of hidden layers: [1, 2, 3, 4], epochs = [100,

TABLE 3

AI Models with Input data, Architecture, and Training Quantities. A bias unit is added to the input and the hidden layers in the ANN. ReLu, Rectified Linear Unit, SGD, Stochastic Gradient Descent.

AI models	Input data	Architecture	Training
ANN1	Regional ionosphere coefficients Latitude, Longitude	Input layer: 27 neurons Output layer: 1 neuron Linear mapping	Optimizer: SGD Learning rate: 2e-1 Momentum: 0.9 Epochs: 400 Batch size: 300
ANN2	Regional ionosphere coefficients Latitude, Longitude	Input layer: 27 neurons Hidden layer 1: 10 neurons Hidden layer 2: 10 neurons Hidden layer 3: 10 neurons Output layer: 1 neuron Activation function: ReLU	Optimizer: SGD Learning rate: 1e-3 Momentum: 0.9 Epochs: 400 Batch size: 300
ANN3	Regional ionosphere coefficients Latitude, Longitude HoD_{sin}, HoD_{cos}	Input layer: 29 neurons Hidden layer 1: 10 neurons Hidden layer 2: 10 neurons Hidden layer 3: 10 neurons Output layer: 1 neuron Activation function: ReLU	Optimizer: SGD Learning rate: 1e-3 Momentum: 0.9 Epochs: 400 Batch size: 300
ANN4	Regional ionosphere coefficients Latitude, Longitude HoD_{sin}, HoD_{cos} F10.7	Input layer: 30 neurons Hidden layer 1: 10 neurons Hidden layer 2: 10 neurons Hidden layer 3: 10 neurons Output layer: 1 neuron Activation function: ReLU	Optimizer: SGD Learning rate: 1e-3 Momentum: 0.9 Epochs: 400 Batch size: 300
ANN5 IONOWB_ AI	Regional ionosphere coefficients Latitude, Longitude HoD_{sin}, HoD_{cos} F10.7 Kp, Dst	Input layer: 32 neurons Hidden layer 1: 10 neurons Hidden layer 2: 10 neurons Hidden layer 3: 10 neurons Output layer: 1 neuron Activation function: ReLU	Optimizer: SGD Learning rate: 1e-3 Momentum: 0.9 Epochs: 400 Batch size: 300
ANN6	Latitude, Longitude HoD_{sin}, HoD_{cos} F10.7 Kp, Dst	Input layer: 7 neurons Hidden layer 1: 10 neurons Hidden layer 2: 10 neurons Hidden layer 3: 10 neurons Output layer: 1 neuron Activation function: ReLU	Optimizer: SGD Learning rate: 1e-4 Momentum: 0.9 Epochs: 200 Batch size: 50

RF1	Regional ionosphere coefficients
	Latitude, Longitude
	HoD
	F10.7, Kp, Dst
		Number of trees= 300
		Min_samples_split=5
		Min_samples_leaf=3
			Criterion: Mean squared error

RF2	Latitude, Longitude
	HoD
	F10.7, Kp, Dst
		Number of trees= 300
		Min_samples_split=5
		Min_samples_leaf=5
			Criterion: Mean squared error

200, 300, 400, 500], batch size = [50, 100, 200, 300, 400, 500]. Batch size refers to the number of training examples propagated through the network in one iteration (forward/backward pass). The number of epochs represents the number of complete passes through all the training examples. Hyperparameters that minimize $R(\mathbf{W})$ were selected. The optimal number of hidden units was identified as 10 with

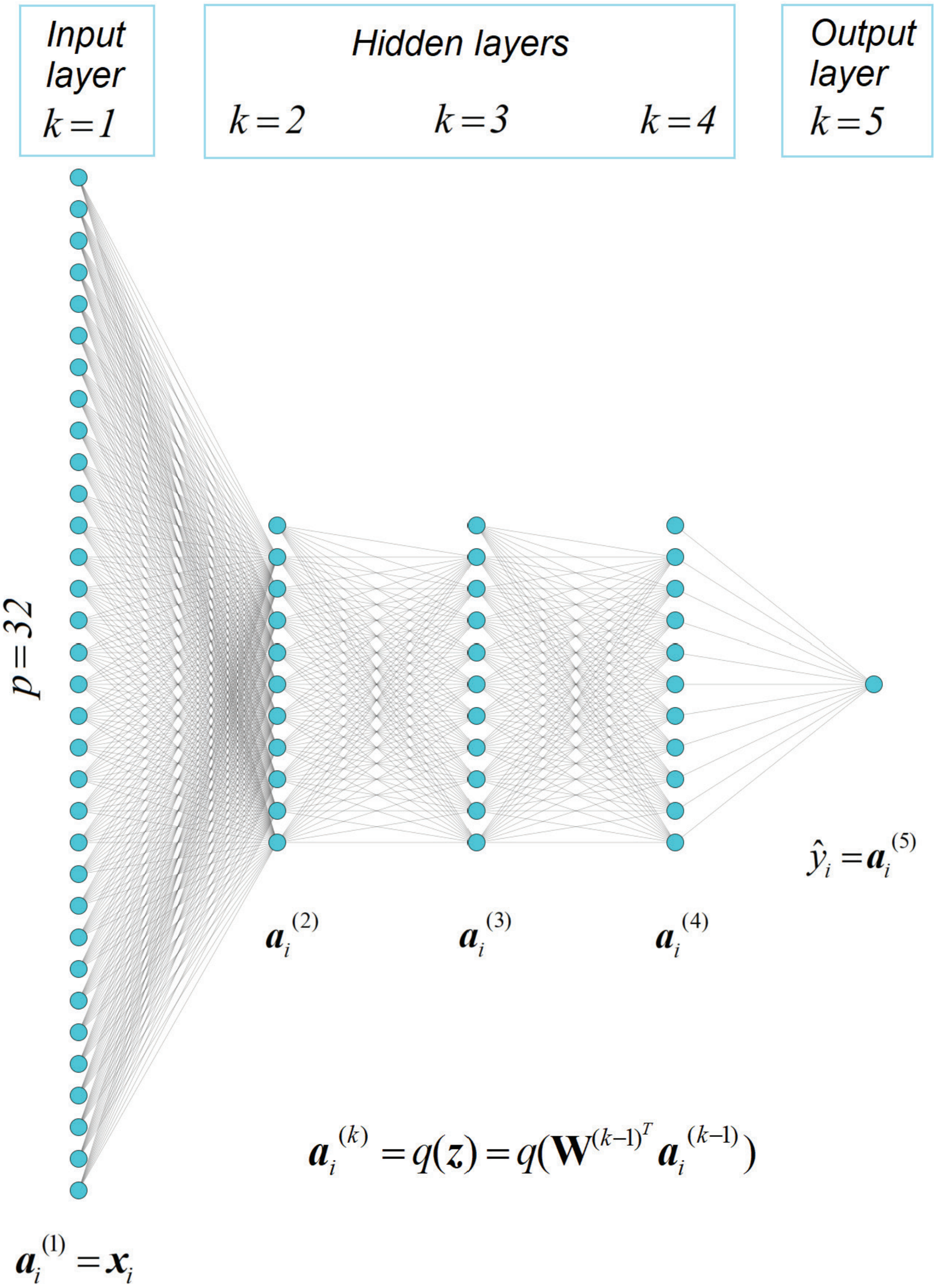


FIGURE 3 Representation of the architecture of the ANN5 model with one input, three hidden, and one output layer (right). An additional bias unit was added to the input and the hidden layers. ANN architectures were drawn using the web-based tool NN-SVG (LeNail, 2019).

three hidden layers. This architecture was kept constant in all ANN models from Table 3 tested; additional fine-tuning resulted in no significant differences. The addition of more neurons and more layers to the ANN increases the complexity of the interactions between layers and neurons. While this can be perceived as a positive, too much complexity can lead to overfitting the training data as the machine learns from its noise and is thus not capable of generalizing information from new examples. Choosing the appropriate complexity of an ANN requires careful attention when optimizing the model. Additionally, the Random Forest (RF) algorithm (Breiman, 2001), which builds an ensemble of decision trees, can be used to compare results and determine which input features are relevant. The RF algorithm provides the possibility of easily estimating the relative importance or contribution of each input feature (Breiman, 2001). A detailed explanation of how a decision tree and RF are built to address the VTEC problem can be found in Natras et al. (2022a). The RF model includes 300 decision trees; the quality of splits within the trees is measured with the mean squared error.

Most of the input data for these models consists of the coefficients c_{nm}, i.e. regional ionosphere parameters from Equation (5), which is estimated for each hour in each country within the study region (Section 2.1) by processing the GNSS observations in the Bernese GNSS Software. During the training phase, the geographical coordinates belonging to the origin of the coefficient expansion were provided with the output data of the VTEC at these positions. Information on latitude and longitude was then added to extract spatial interactions. Additional input data were introduced to each new model. The information regarding time was added to extract temporal dependencies and features. For the ANN models, the sine and cosine components are calculated to preserve their cyclic significance, as shown in Equation (18):

$$HoD_{\sin} = \sin\left(\frac{2\pi\ HoD}{24}\right) \qquad HoD_{\cos} = \cos\left(\frac{2\pi\ HoD}{24}\right) \tag{18}$$

where *HoD* is the hour of the day. Decision tree-based algorithms, such as RF, do not require time information to be split into sine and cosine components (Boussard et al., 2017). Also, RF considers just one input feature at a time as the splitting variable when building a tree (Breiman, 2001). Therefore, it will fail to process sine and cosine components simultaneously, although they are expected to be considered as one system. Data on solar and geomagnetic activity such as the solar flux F10.7 and the geomagnetic indices of Kp and Dst were also introduced gradually. Eventually, the models (ANN6 and RF2) were trained solely on information regarding time, location, and solar and geomagnetic activity. The data were shuffled and randomly divided into training (80%) and validation datasets (20%). Using the training data, we built models that can approximate the function between inputs and outputs and estimate the outcome for new inputs. Using validation data, values for the hyperparameters were selected. All input features were scaled (i.e. standardized) to obtain data with a mean value of zero and a standard deviation of one. This is standard procedure in many learning algorithms that are sensitive to the scale of the input features (Zheng & Casari, 2018). This ensures that all inputs are treated equally, even if the variables have different scales. Also, gradient descent can converge faster, meaning that the optimal parameters for each neuron can be located more quickly. Feature standardization is defined in Equation (19):

$$\tilde{x}_{i,j} = \frac{x_{i,j} - \bar{x}_j}{\sigma_j} \tag{19}$$

where $x_{i,j}$ is the data point *i* from *j*-th input feature, $\bar{x}_j$ is the mean of the *j*-th input feature (over all data points from test and validation datasets together), σ_j is the standard deviation of the *j*-th input feature, $\tilde{x}_{i,j}$ is the resulting standardized data point *i* from *j*-th input feature. Scaling is performed on each input feature independently. Mean and standard deviation values are then stored for later data to perform the consistent transformation.

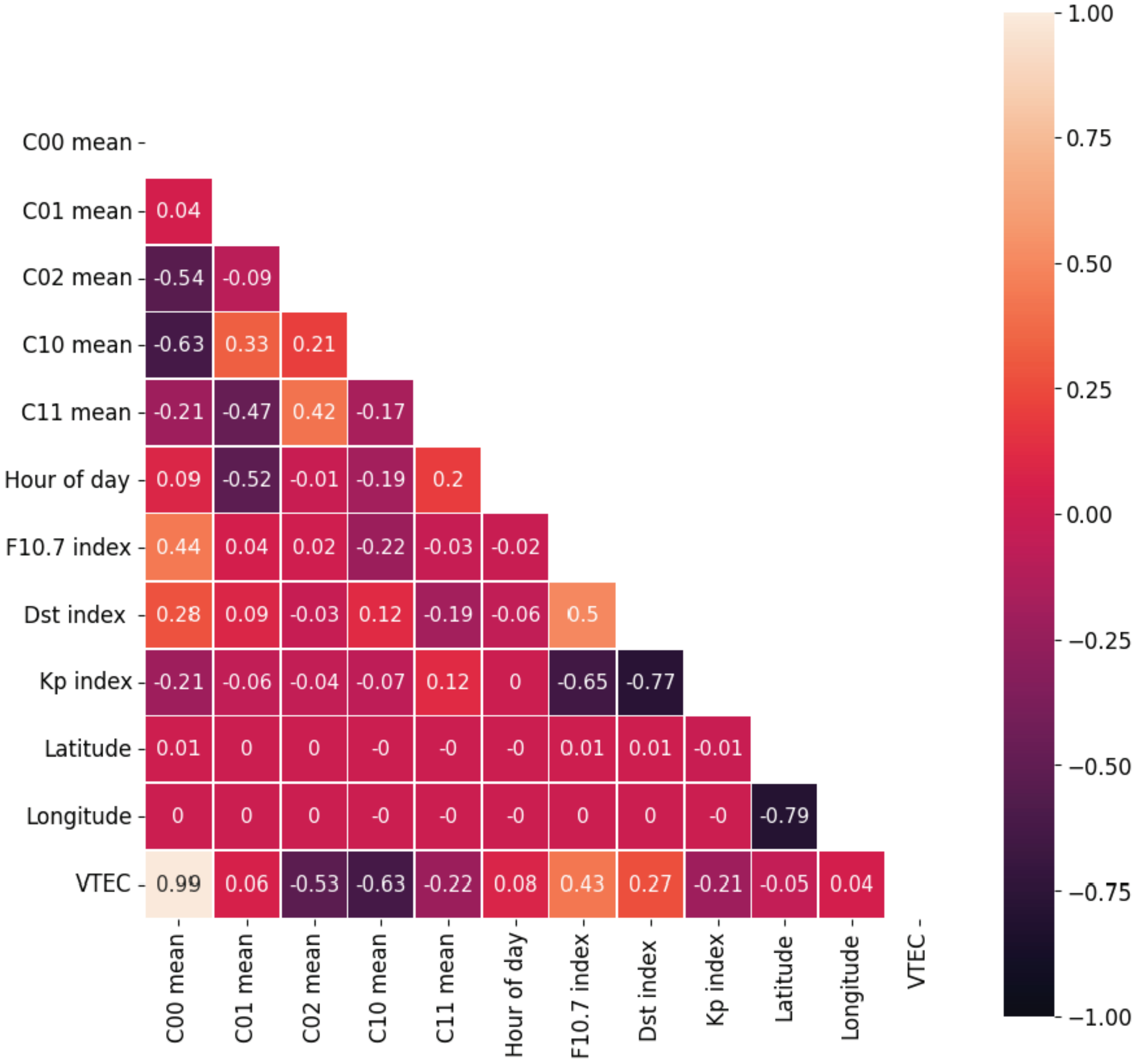


FIGURE 4 Correlation heatmap of the input features and the output of the machine learning model.

A correlation heatmap is created from a 2D correlation matrix of the input features and the output (Figure 4). Coefficients $\boldsymbol{c}_{nm}$ of the regional ionosphere are presented as Cnm in the heatmap. Five coefficients were estimated for each country, as explained in Section 2.1. Mean correlations for coefficients are presented. The coefficient $\boldsymbol{c}_{00}$ has the strongest positive correlation to VTEC, followed by $\boldsymbol{c}_{10}$ and $\boldsymbol{c}_{02}$, both of which have a moderate negative relationship to VTEC. The F10.7 index has a positive moderate relationship, while the Dst and Kp indices have a weaker relationship with VTEC. The lowest correlations involve HoD, and latitude and longitude, which were added to extract temporal and spatial features.

Figure 5 and Table 4 show that the temporal, solar, and geomagnetic activity information increase the ANN model accuracy, respectively, in terms of the RMSE and the correlation coefficients (CCs) between the VTEC model output and VTEC from Equation (5). These input data are useful for deriving temporal, solar, and geomagnetic features and relationships. Moreover, removing the regional ionosphere coefficients from the AI models (ANN6 and RF2) increases

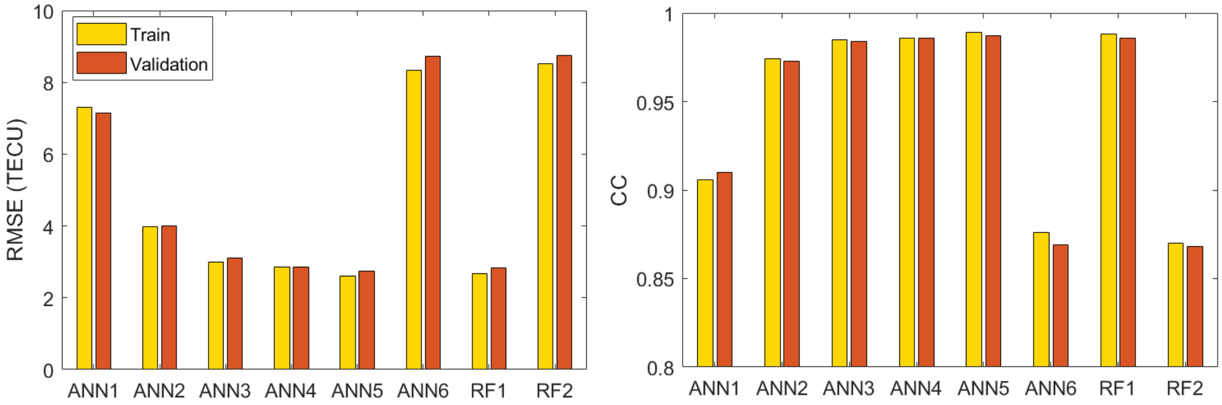


FIGURE 5 The RMSE (left) and correlation coefficients (CCs, right) for training and validation of ANN and RF models.

TABLE 4
RMSE and CCs for Training and Validation Datasets for Different AI Models.

	Dataset	ANN1	ANN2	ANN3	ANN4	ANN5	ANN6	RF1	RF2
RMSE (TECU)	Train	7.31	3.99	3.00	2.86	2.60	8.34	2.67	8.51
	Validation	7.15	4.01	3.11	2.86	2.73	8.73	2.83	8.75
CC	Train	0.906	0.974	0.985	0.986	0.989	0.876	0.988	0.870
	Validation	0.910	0.973	0.984	0.986	0.987	0.869	0.986	0.868

the RMSE approximately three times and decreases the CCs by approximately 0.1. The ANN6 and RF2 models are not as accurate as the linear model with the regional ionosphere coefficients (ANN1). These results indicate that estimating the VTEC using only the spatiotemporal (latitude, longitude, and HoD), geomagnetic, and solar parameters (F10.7, Kp, and Dst) will not be sufficient. These data alone cannot provide an accurate description of VTEC. However, in conjunction with more influential, descriptive parameters of VTEC variations, such as regional ionosphere coefficients, one can significantly improve the model. Based on these results, the ANN5 model was selected as optimal and henceforth referred to as the RIM IONOWB_AI model.

Knowing the underlying relationships between the inputs and the output of the AI model is useful for interpreting what the AI model has learned and which input features have been selected as relevant. The relative importance of the input features was calculated as a root square of the sum of the squared improvements over the nodes in a tree generated by RF when a specific input feature $\boldsymbol{x}_j$ was selected (Breiman, 2001). The importance of the input features was estimated using the RF1 and RF2 models (Figure 6). The relative importance of each coefficient is calculated as the mean importance of the regional ionosphere coefficients estimated for five countries. For the RF1 model, the first coefficient $\boldsymbol{c}_{00}$ contributes most to the VTEC output, followed by HoD and F10.7 index. Contributions from other coefficients were at similar levels. Smaller contributions were provided by the Dst index, longitude, latitude, and Kp index. These parameters extract spatial dependence and provide additional geomagnetic-dependent VTEC variations and relationships, but do not have a dominant influence in defining the VTEC variations with respect to the other inputs. Because the coefficients were calculated from the CORS and EPN observations, they already contain regional ionosphere information related to the origin of the Taylor series expansion and therefore, they have a large impact on the model output. When training a model with no regional ionosphere parameters, the highest contribution (nearly 70%) comes from HoD. The other most influential inputs are the F10.7, Dst, and Kp indices. The contributions of these inputs are higher in the RF2 model than in the RF1 model. This is because the RF2 model does not include input parameters that describe the regional ionosphere. Regional ionosphere coefficients correlate much more closely with the VTEC than any of the other input features. The least critical contributions come from latitude and longitude. This may be because the F10.7, Kp, and Dst

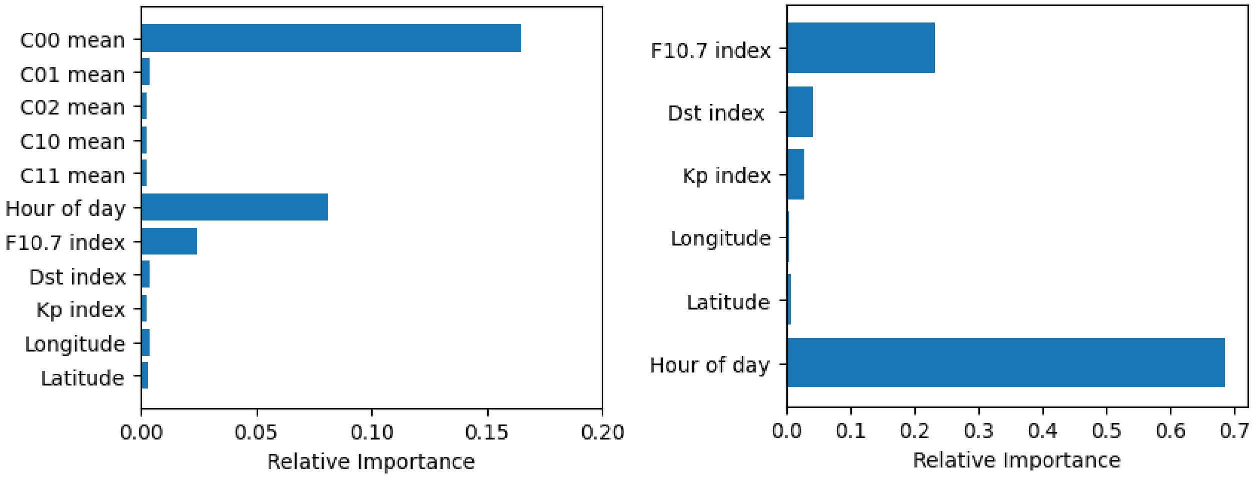


FIGURE 6 Relative importance of input variables when training AI models with (left) and without coefficients (right) estimated using Random Forest (RF).

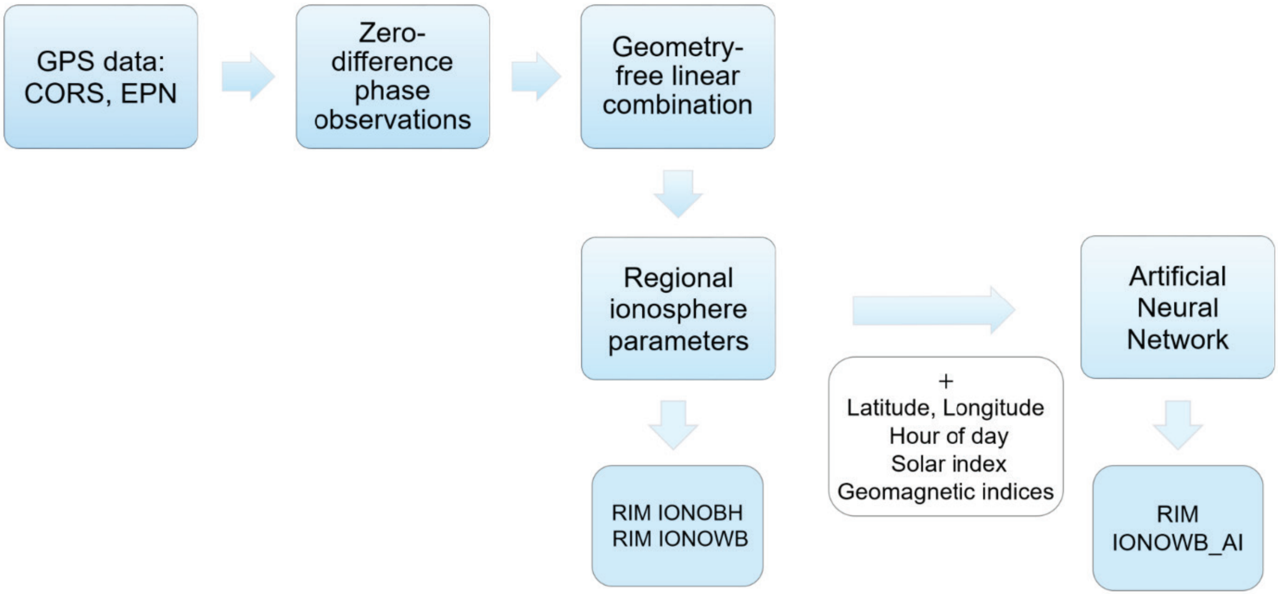

GPS data: CORS, EPN
Zero-difference phase observations
Geometry-free linear combination
Regional ionosphere parameters
RIM IONOBH
RIM IONOWB
+
Latitude, Longitude
Hour of day
Solar index
Geomagnetic indices
Artificial Neural Network
RIM IONOWB_AI

FIGURE 7 Flowchart leading to the development of RIMs. First, GPS data from local CORS and EPN observations were processed in the Bernese GNSS software to form geometry-free linear combinations of phase observations which were then used to estimate coefficients representing the regional ionosphere parameters that form the basis of the RIM IONOBH and the RIM IONOWB. Regional ionosphere parameters were then fed into the ANN along with the spatial and temporal parameters as well as solar and geomagnetic indices, resulting in the RIM IONOWB_AI model.

indices are not spatially dependent. These results provide us with some intuition of how the AI models learn from these datasets.

Regional ionosphere parameters estimated from the CORS and EPN observations for each country included in the development of RIM IONO_WB were combined with spatiotemporal (latitude, longitude, and HoD), solar (F10.7 index), and geomagnetic (Kp and Dst indices) parameters via an AI technique using the ANN algorithm; this has resulted in the RIM IONOWB_AI model (Figure 7). This model can estimate VTEC for any location in the western Balkans.

2.4 | Selection of the Study Period

Three time periods were selected for the study:

- March 20–26, 2014,
- March 15–20, 2015, and
- March 20–26, 2018.

These periods were chosen because they included different phases of the solar cycle and varying ionosphere activity. Specifically, the solar maximum occurred in 2014, the declining phase of the solar cycle was in 2018, and a severe geomagnetic storm occurred in March 2015. There were large ionosphere disturbances over the study region that resulted in VTEC changes of 60% to 150% on the day of the storm in March 2015 and between 50% and 80% on the following days compared to its regular variability (Natras et al., 2023). Furthermore, all periods evaluated were around the spring equinox, when one can expect to find the highest number of electrons within the ionosphere over the study region (Natras et al., 2023).

The RIM IONOBH model was estimated for the first and third study periods, while the RIM IONOWB was generated for the first and second study periods. Therefore, while both models were established for the period of the solar maximum in 2014, different time frames were chosen for the second study period, including the solar minimum period (March 2018) for the RIM IONOBH and the geomagnetic storm period (March 2015) for the RIM IONOWB. This was because the models were developed independently of each other as part of two different projects implemented at different research institutes with different objectives, i.e., to study RIMs in the period of solar maximum (2014) and minimum (2018), and to analyze RIMs during a severe geomagnetic storm (2015). Furthermore, data were not available from all CORS networks for all periods examined; this resulted in different second study periods.

2.4.1 | Overview of Solar and Geomagnetic Activity

Solar and geomagnetic indices are presented in Figure 8 for periods March 20–26, 2014 (left), March 15–20, 2015 (middle), and March 20–26, 2018 (right). These indices include the sunspot number R, radio flux F10.7 of the Sun's emission at the 10.7 cm wavelength (Covington, 1969), disturbance storm time Dst (Sugiura, 1964), and Kp (Chapman & Bartels, 1962) obtained from the NASA/GSFC OMNI data set via OMNIWeb (https://omniweb.gsfc.nasa.gov/form/dx1.html). The first study period represents a solar maximum that reached its peak in April 2014. Therefore, the number of sunspots was increased throughout,

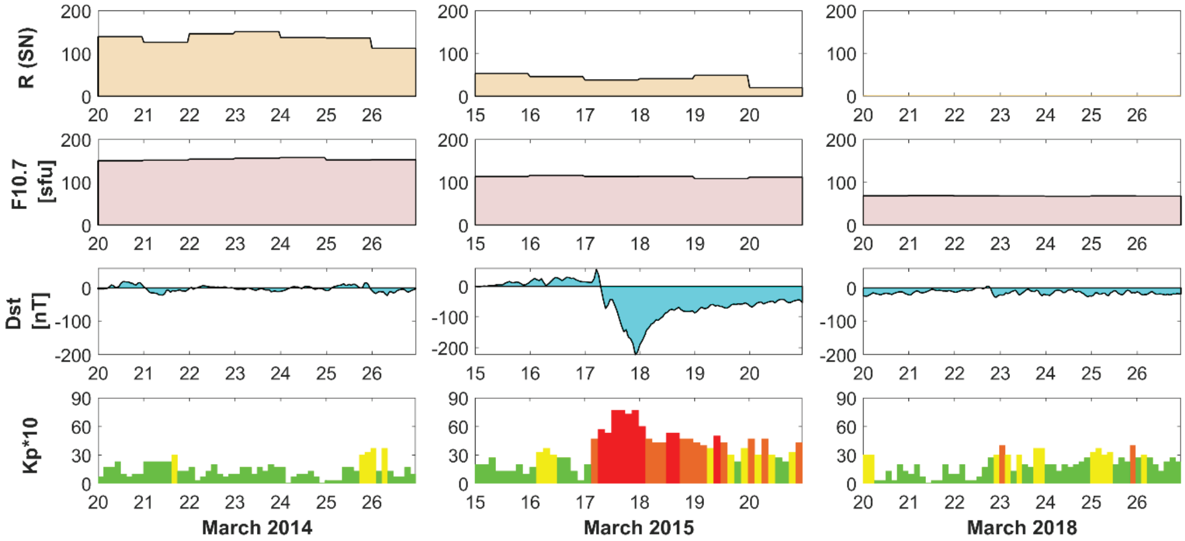


FIGURE 8 Overview of solar and geomagnetic indices for the three study periods. Left panel: March 20–26, 2014; middle panel: March 15–20, 2015; right panel: March 20–26, 2018. From top to bottom: R sunspot number (SN), solar radio flux F10.7 in sfu (solar flux units), Dst in nT, Kp (Quiet $kp \cdot 10 < 30$, Moderate $30 \leq kp \cdot 10 < 40$, Active $40 \leq kp \cdot 10 < 50$, Storm $kp \cdot 10 \geq 50$).

ranging from 112 to 151, and F10.7 had values larger than 150 sfu. By contrast, the geomagnetic conditions were mostly quiet with Kp values below 4 with small fluctuations in the Dst index at approximately zero. The second investigation period included the strongest geomagnetic storm of the solar cycle 24, known as the St. Patrick's Day Storm, that occurred on March 17, 2015. This severe storm was characterized by a main phase in which the Kp index reached a value of 8 and the Dst index was less than −200 nT. This was followed by a recovery phase that began on March 18, 2015, and lasted for several days until the geomagnetic field returned to normal conditions. It is worth mentioning that both the number of sunspots (from 20 to 53) and the solar radio flux (about 110 sfu) were significantly lower compared to those measured during the first period. The third study period featured low levels of solar activity during the decline phase of the solar cycle toward its minimum. This period was characterized by no sunspots and an F10.7 less than 70 sfu. The geomagnetic activity was calm to active, with no geomagnetic storms.

2.5 | Validation of the Regional Ionosphere Models

The RIMs IONOBH and IONOWB were validated against other ionosphere models, including:

- GIMs final products from CODE and IGS (from https://cddis.nasa.gov/archive/gnss/products/ionex/)
- European RIM from the GIOMO model, based on weighting functions and developed at the Department of Geodesy and Geoinformation, Vienna University of Technology (Magnet, 2019)
- European RIM based on polynomial B-spline functions developed at DGFI-TUM as a two-step VTEC model (TSM-Product 2c) named as OTHR model (Goss et al., 2020).

- NeQuick2 (computed via https://t-ict4d.ictp.it/nequick2/nequick-2-web-model)
- GPS broadcasted model Klobuchar. VTEC values were calculated from broadcast ephemeris data (https://cddis.nasa.gov/archive/gnss/data/daily/).

The climatological empirical model NeQuick2 (Nava et al., 2008) is the latest version of the NeQuick ionosphere electron density model developed for the computation of slant electron density profiles and TEC in the ionosphere at a given height, geocentric latitude, and geocentric longitude. Information on solar activity is provided to the model by the daily solar radio flux F10.7. The NeQuick2 model takes into account the contribution of the plasmasphere to VTEC up to 40,000 km. Although the electron density in the plasmasphere is much lower than in the ionosphere, under some conditions, such as at night and during periods of low solar activity, the contribution of the plasmasphere may represent a larger proportion of the VTEC. The relative global contribution of the plasmasphere to the VTEC depends on latitude and solar activity, with a minimum contribution of about 10% during daytime hours and a maximum of up to 60% at night.

The VTEC data of the RIM GIOMO model were provided for the study periods in March 2014 and March 2018 during the project that led to the development of the IONOBH. By contrast, VTEC data from the RIM OTHR model were available for all study periods.

The ionosphere models were also evaluated for precise point positioning (PPP) by processing single-frequency (L_1) observations for selected GNSS stations using the Bernese GNSS Software. The following cases of L_1 PPP solutions were carried out:

- L_1 positioning solution without ionospheric corrections,
- L_1 positioning solution with ionospheric corrections from the final GIM CODE, and
- L_1 positioning solution with ionospheric corrections from the RIMs IONOBH and IONOWB.

However, there is no interface in the Bernese software that can be used to process broadcasted ionospheric delay models. Therefore, we could not apply corrections from the Klobuchar model. The following processing steps were implemented for the PPP method: data preprocessing (RNXSMT), import of data into the Bernese format (RXOBV3), preparing orbit and Earth's orientation information (POLUPD, PRETAB, ORBGEN), data preprocessing 2 (CODSPP, GPSEST, RESRMS, SATMRK), performing a solution for epoch parameters, and/or creation of normal equations NEQ (GPSEST) and an NEQ-based final session solution (ADDNEQ2). More details on each step are included in Dach et al. (2015). Positioning errors were estimated as differences between "true" positions and the single-frequency positioning results expressed as north, east, and up components. The weekly combined EPN solutions (https://www.epncb.oma.be/ftp/product/combin/) were used as "true" positions for the EPN stations. For the CORS stations, "true" positions were estimated with the dual-frequency PPP method in the Bernese GNSS software. For the analysis, errors were expressed as 1-D RMS vertical and 2-D RMS horizontal position errors, as well as, 3-D RMS position errors as shown in Equation (20):

$$RMS\_vertical\_error = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Delta U_i^2} \tag{20a}$$

$$2D\_RMS\_horizontal\_error = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\Delta E_i^2 + \Delta N_i^2)} \tag{20b}$$

$$3D\_RMS\_error = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\Delta E_i^2 + \Delta N_i^2 + \Delta U_i^2)} \tag{20c}$$

where ΔE_i, ΔN_i, ΔU_i are errors in the east, north, and up (vertical) components, respectively, of the i-th position estimate sample and n is the total number of position estimate samples.

3 | RESULTS

Section 3.1. compares the new RIMs with other ionosphere models as described in Section 2.5. The developed RIMs were then applied to correct the ionospheric range error in single-frequency positioning. The results of this procedure are presented in Section 3.2.

3.1 | VTEC Results

The results shown in Figure 9 include the VTEC time series for the EPN station SRJV estimated from the RIM IONOBH for March 2014 (first panel) and March 2018 (third panel); differences between the RIM IONOBH and other ionosphere models are also shown (second and fourth panels). The IONOBH VTEC variability was at least five times higher in March 2014 (solar maximum) than in March 2018 (solar minimum). The VTEC values from the GIMs are mostly higher than the VTEC values from the RIM IONOBH. The largest differences between the GIMs and the RIM IONOBH occur during the night, with differences of up to 10 TECU in March 2014 and up to approximately 5 TECU in March 2018. During the day, the differences between the RIM IONOBH and the GIMs are reduced by a factor of 2 (i.e., mostly below 5 TECU in March 2014 and 2 TECU in March 2018). Higher differences were observed with respect to the GIOMO model, up to 20 TECU in 2014 and primarily up to 5 TECU in 2018. Interestingly, the GIOMO corresponds more effectively to the RIM IONOBH in 2018 and at night. Most of the results from RIM OTHR are in better agreement with the RIM IONOBH with differences smaller than those observed from the GIM CODE in March 2014 and March 2018. The models NeQuick2 and Klobuchar underestimate the VTEC in March 2014, while the Klobuchar model overestimates VTEC in March 2018. Differences from the Klobuchar model are up to 20 TECU. In March 2018, the results from the RIM IONOBH were in better agreement with those from the NeQuick2 model with differences up to 5 TECU during the day and mostly below 2 TECU at night. These results correspond to results reported by Wang et al. (2017) and Shi et al. (2019). In their studies, they show that climatological models, such as NeQuick2 and IRI2016, provide extremely underestimated values for VTEC during periods of high solar activity; however, these models are consistent with the GIM IGS in mid-latitudes and during periods of low solar activity (Shi et al., 2019). Differences between the RIMs IONOBH and IONOWB_AI are below 5 TECU for March 2014, with higher differences observed primarily at night.

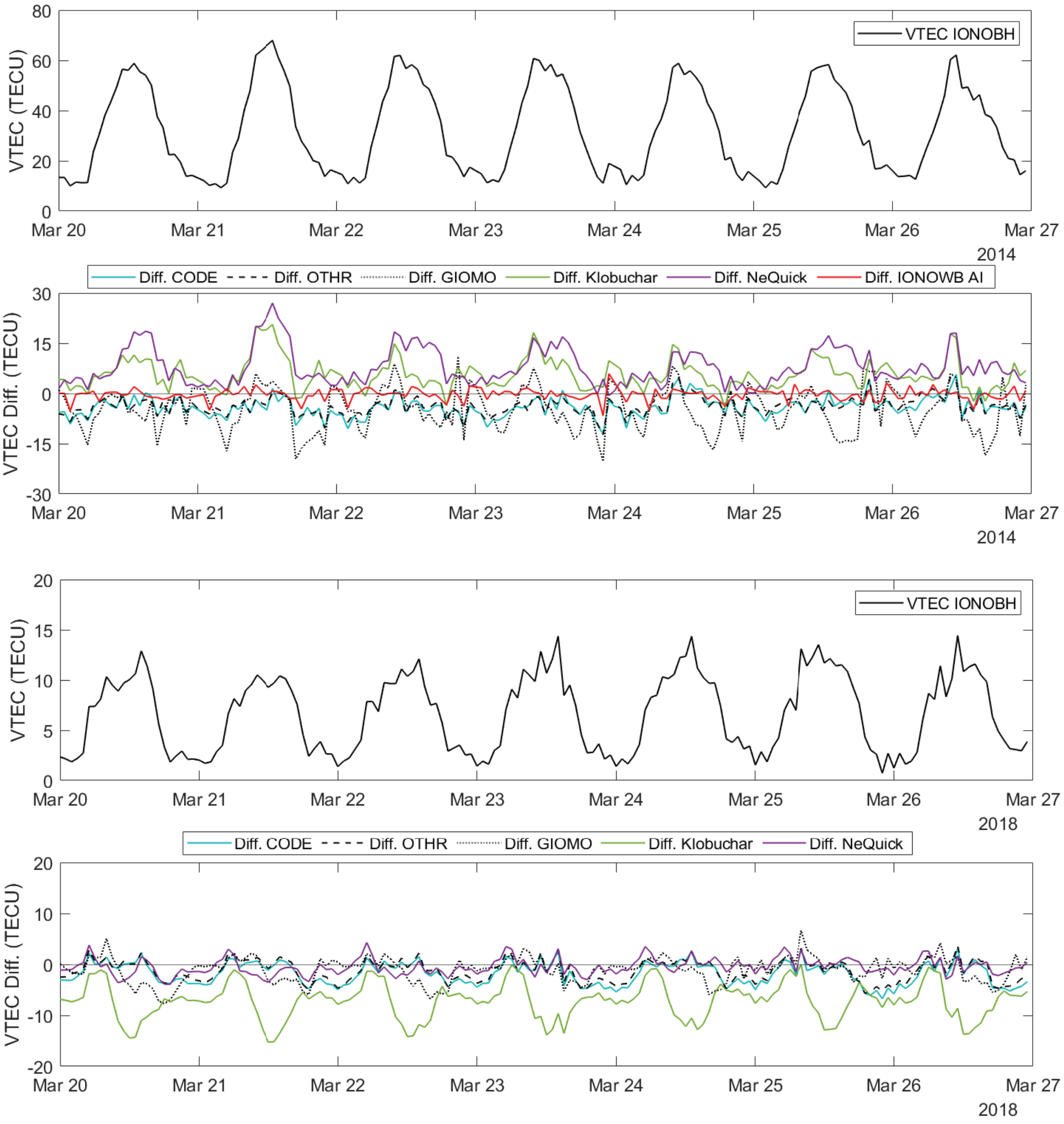


FIGURE 9 VTEC time series and differences between VTEC from the RIM IONOBH and other models that were estimated based on the location of the EPN SRJV station. Top two panels: March 20–26, 2014 (solar maximum); bottom two panels: March 20–26, 2018 (solar minimum). Note the different scaling of the y-axes due to the effect of the solar cycle on VTEC.

The results shown in Figure 10 illustrate the differences between VTEC of the RIM IONOWB and VTEC of the other models, as well as VTEC time series from the RIMs IONOWB for the location of the EPN station SRJV for March 2014 (top two panels) and March 2015 (bottom two panels). In March 2014, the VTEC differences between the RIM IONOWB and other models were comparable to the results obtained with the RIM IONOBH (Figure 9, top).

In March 2015, the regular maximum VTEC daily values during March 15–16 (i.e., without disturbances from a geomagnetic storm) were about 20 TECU lower than in March 2014. During these days, the differences in VTEC from the RIM IONOWB with respect to the GIMs and the OTHR model were mostly less than 5 TECU; by contrast, differences from the NeQuick2 model were as high as 10 TECU during the daytime and below 3 TECU at night. On the day of the St.

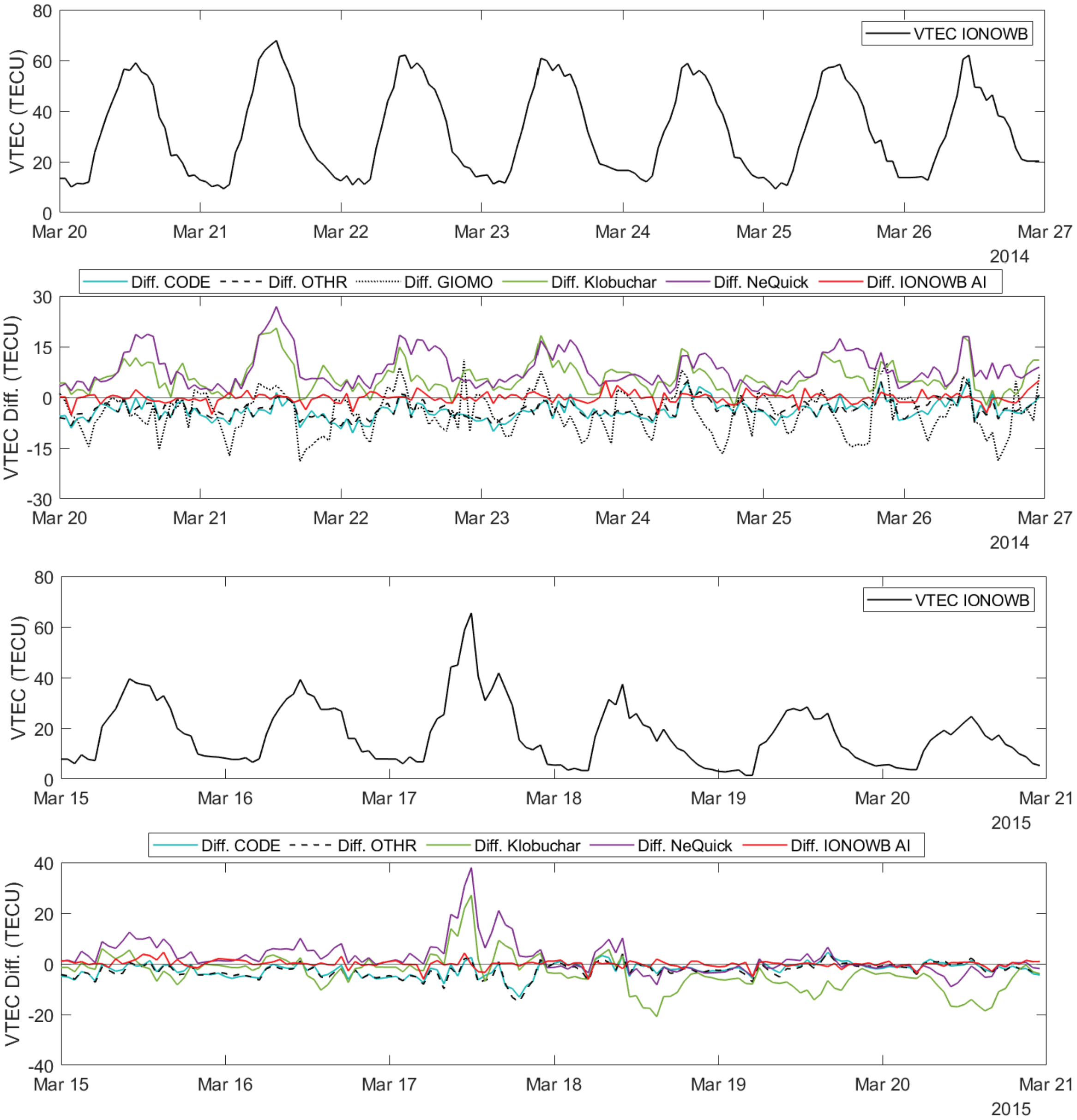


FIGURE 10 VTEC time series and differences between VTECs from the RIM IONOWB and other ionosphere models estimated based on the location of the EPN SRJV station. Top two panels: March 20–26, 2014 (solar maximum); bottom two panels: March 15–20, 2015 (includes a severe geomagnetic storm). Note the different scaling of the y-axes due to the solar cycle and the effects of the geomagnetic storm on VTEC.

Patrick's geomagnetic storm (March 17), the RIM IONOWB showed two VTEC peaks, including one around local noon (greater than 60 TECU) and another in the evening (greater than 40 TECU). The maximum differences detected on this day were 15 TECU. During the daytime, the differences from GIMs were mostly below 5 TECU, and almost zero during occurrences of the two aforementioned peaks. The higher VTEC differences between the RIM IONOWB and the other models occurred later in the evening. By contrast, the largest differences (up to 40 TECU) were those associated with the climatological model during the main storm phase; these findings correspond to results reported by Wang et al. (2017). The VTEC decreased during the recovery phase of the storm (March 18–21). Differences in the VTEC were mostly below 4 TECU compared to those of the GIMs and the

OTHR model. With respect to the NeQuick2 model, the differences were below 10 TECU during the daytime and below 3 TECU at night. A comparison of the RIM IONOWB and IONOWB_AI reveals differences of up to 5 TECU, mainly occurring at night. During the daytime of the main phase of the storm (March 17), the differences were mostly around 0 with maximum differences of less than 4 TECU. Differences between the RIM IONOWB and the Klobuchar model were as high as 10 TECU before the storm, near 30 TECU during the main storm phase, and up to 20 TECU during the recovery.

A comparison between the newly-developed RIMs IONOBH and IONOWB and the other models shows that the smallest differences during daytime are between the new RIMs and the final GIM CODE. At night, the new RIMs agree most frequently with the NeQuick2 model, as it takes into account the contributions of the plasmasphere electron content to the ionospheric VTEC.

Figure 11 presents the RMSE between VTECs of the newly-developed RIMs and those from GIM CODE, RIM OTHR, RIM GIOMO, and the Klobuchar (Klob.) model for the study periods March 2014, March 2015, and March 2018. The RMSE is provided for the entire day from 00:00 to 23:00 UTC (upper left) and during daytime hours from 6:00 to 16:00 UTC (upper right). Correlation coefficients are provided for the entire day (bottom) together with the mean RMSE and the mean correlation coefficients.

An analysis of the RIMs IONOWB and IONOWB_AI was performed for the stations EPN GSR1, EPN POZE, EPN SRJV, IGEWE TIRA, and EPN ORID. An analysis of RIM IONOBH was done for the stations EPN SRJV, BIHPOS SEKO, EPN POZE, and EPN DUB2. The RMSE values were lower in the daytime than at night, with values mostly below 4 TECU, except for the Klobuchar model. The

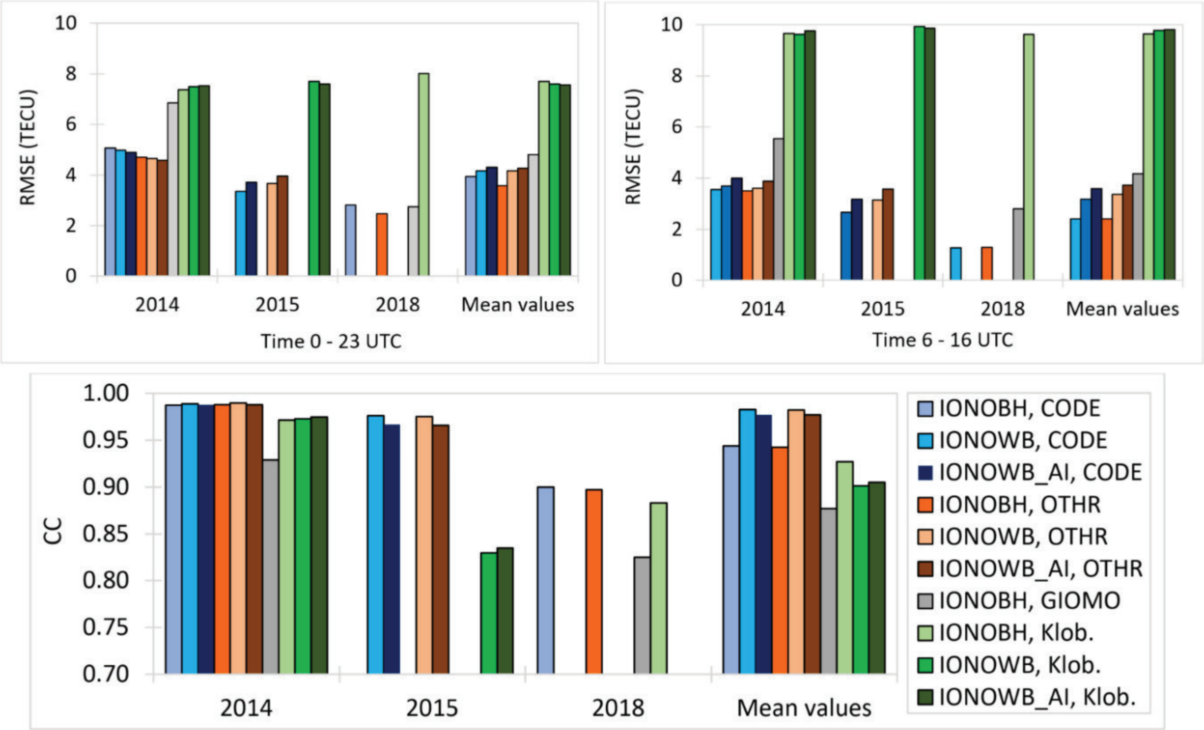


FIGURE 11 Top: RMSE values for study periods March 2014, March 2015, and March 2018 for the entire day (0:00 to 23:00 UTC, upper left) and daytime hours only (6:00 to 16:00 UTC, upper right). Bottom: Correlation coefficients for the entire day (00:00 to 23:00 UTC). Note that correlations from 0.7 to 1.0 are shown.

highest RMSE values were compared to the Klobuchar model, while the lowest were compared to the GIM CODE and the RIM OTHR. The RMSE values with respect to Klobuchar were about 7.5 TECU for all three study periods; the highest discrepancies were during daytime when the RMSE approached 10 TECU. A comparison of the RMSE values between the new RIMs and GIOMO revealed values of approximately 7 TECU in March 2014, including values below 5.5 TECU during the daytime. In March 2018, the RMS errors were reduced by a factor of 2, i.e., to below 3 TECU. In 2014, RMSE values compared to those from the GIM CODE and RIM OTHR were about 5 TECU for the entire day, and about 3.5 to 4 TECU during the daytime alone. In 2015, the RMSE values were below 4 TECU, while in 2018, the RMSE values were below 1.5 TECU during daytime and below 3 TECU at night when compared to the GIM CODE and RIM OTHR. In March 2014, the RMSE with respect to the RIM OTHR were approximately 0.3 TECU lower than

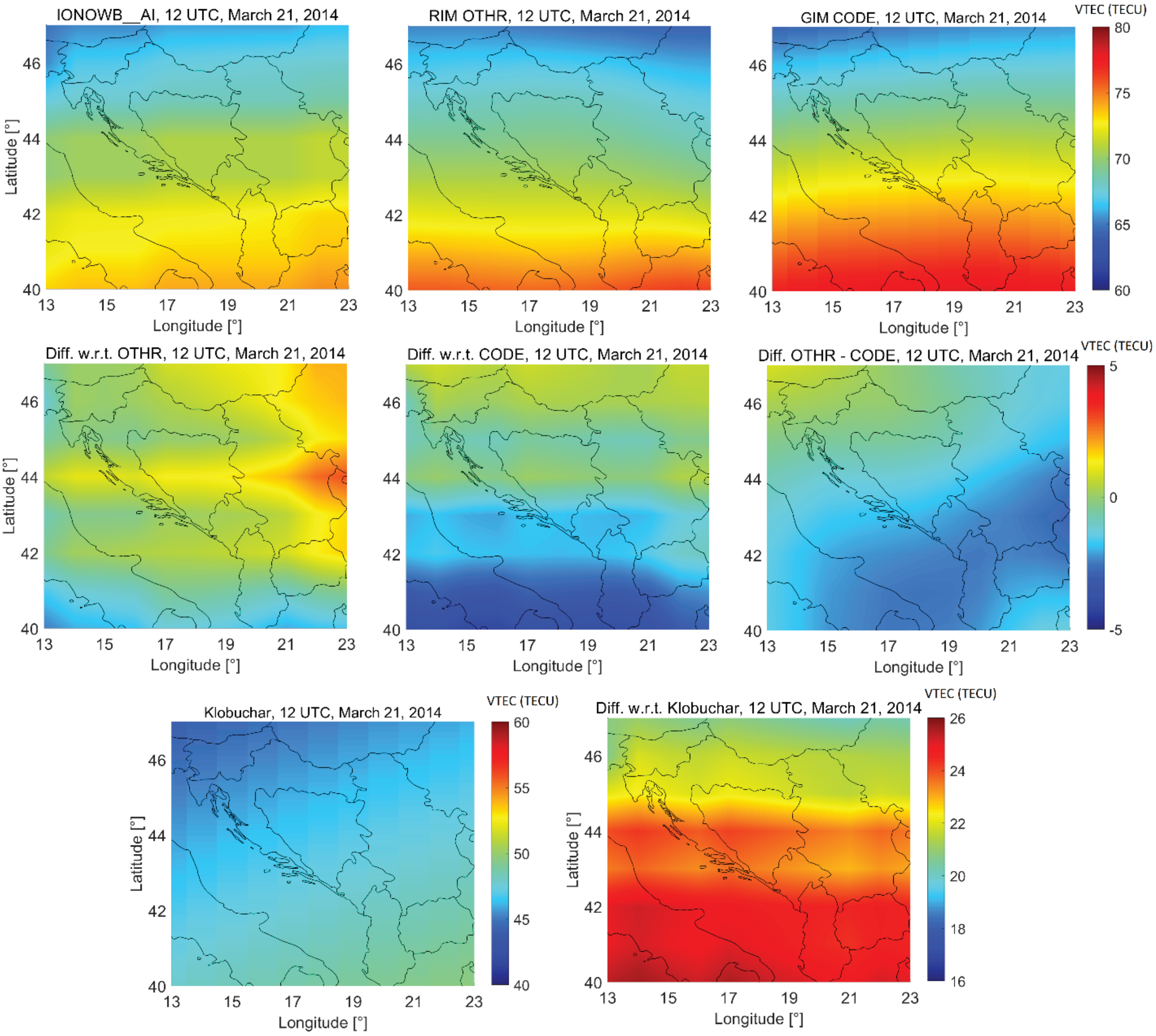

IONOWB__AI, 12 UTC, March 21, 2014
RIM OTHR, 12 UTC, March 21, 2014
GIM CODE, 12 UTC, March 21, 2014
VTEC (TECU)
80
75
70
65
60
Diff. w.r.t. OTHR, 12 UTC, March 21, 2014
Diff. w.r.t. CODE, 12 UTC, March 21, 2014
Diff. OTHR - CODE, 12 UTC, March 21, 2014
VTEC (TECU)
5
0
-5
Klobuchar, 12 UTC, March 21, 2014
VTEC (TECU)
60
55
50
45
40
Diff. w.r.t. Klobuchar, 12 UTC, March 21, 2014
VTEC (TECU)
26
24
22
20
18
16
Latitude [°]
46
44
42
40
Longitude [°]
13
15
17
19
21
23

FIGURE 12 VTEC maps (from left to right) for RIM IONOWB_AI, RIM OTHR, and GIM CODE (top). Middle: VTEC differences (from left to right): $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{OTHR}}$, $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{CODE}}$, and $\text{VTEC}_{\text{OTHR}} - \text{VTEC}_{\text{CODE}}$. Bottom: VTEC map for the Klobuchar model (left) and VTEC difference $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{Klob}}$ (right). All maps were from 12 UTC on March 21, 2014.

RMSE for the GIM CODE; during the daytime, the RMSE values differed by less than 0.1 TECU. In March 2015, values from the newly-developed RIMs corresponded with those from the GIM CODE at 0.3 TECU level during the entire day, and 0.5 TECU better during the daytime than those provided by the RIM OTHR. In March 2018, the RMSE from the RIM OTHR were about 0.35 TECU lower than those from the GIM CODE, where their differences during the daytime can be neglected (i.e., 0.03 TECU). Correlations were the highest between new RIMs, and the GIM CODE and RIM OTHR.

Figure 12 presents VTEC maps for March 21, 2014, which was a day with high solar activity (F10.7 = 153 sfu) and quiet geomagnetic conditions (Kp = 2) at 12 UTC. The VTEC maps for March 17, 2015, the day of the severe geomagnetic storm at 12 UTC (F10.7 = 139 sfu, Kp = 8) and for March 18, 2015, the day after the main storm phase at 12 UTC (F10.7 = 114 sfu and Kp = 5) are shown in Figure 13

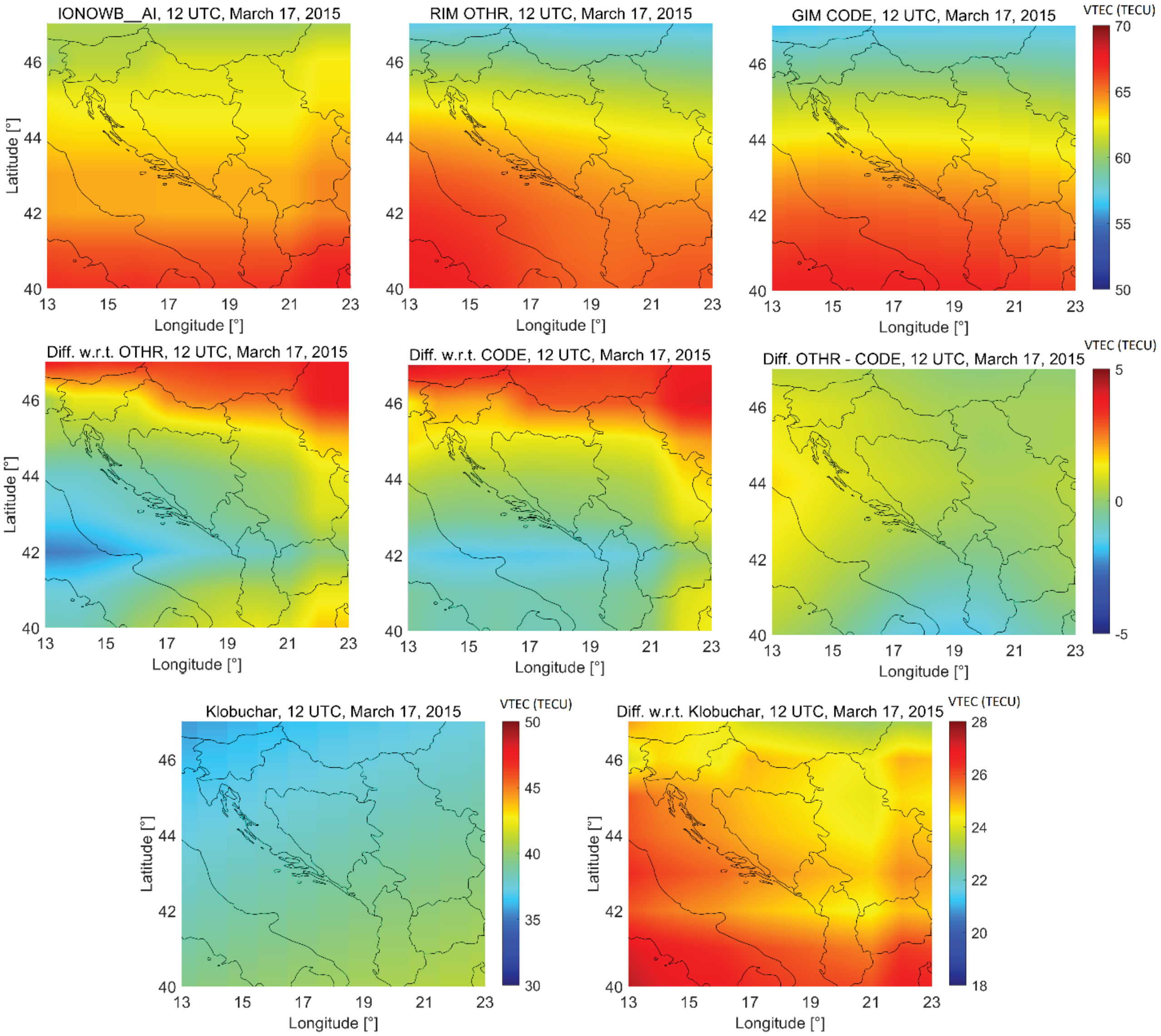

IONOWB__AI, 12 UTC, March 17, 2015
RIM OTHR, 12 UTC, March 17, 2015
GIM CODE, 12 UTC, March 17, 2015
VTEC (TECU)
70
65
60
55
50
Diff. w.r.t. OTHR, 12 UTC, March 17, 2015
Diff. w.r.t. CODE, 12 UTC, March 17, 2015
Diff. OTHR - CODE, 12 UTC, March 17, 2015
VTEC (TECU)
5
0
-5
Klobuchar, 12 UTC, March 17, 2015
VTEC (TECU)
50
45
40
35
30
Diff. w.r.t. Klobuchar, 12 UTC, March 17, 2015
VTEC (TECU)
28
26
24
22
20
18
Latitude [°]
46
44
42
40
Longitude [°]
13
15
17
19
21
23

FIGURE 13 VTEC maps (from left to right) for RIM IONOWB_AI, RIM OTHR, and GIM CODE (top). Mid: VTEC differences (from left to right): $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{OTHR}}$, $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{CODE}}$, $\text{VTEC}_{\text{OTHR}} - \text{VTEC}_{\text{CODE}}$. Bottom: VTEC map for Klobuchar (left) and the VTEC difference $\text{VTEC}_{\text{IONOWB\_AI}} - \text{VTEC}_{\text{Klob}}$ (right). All maps relate to 12 UTC on March 17, 2015.

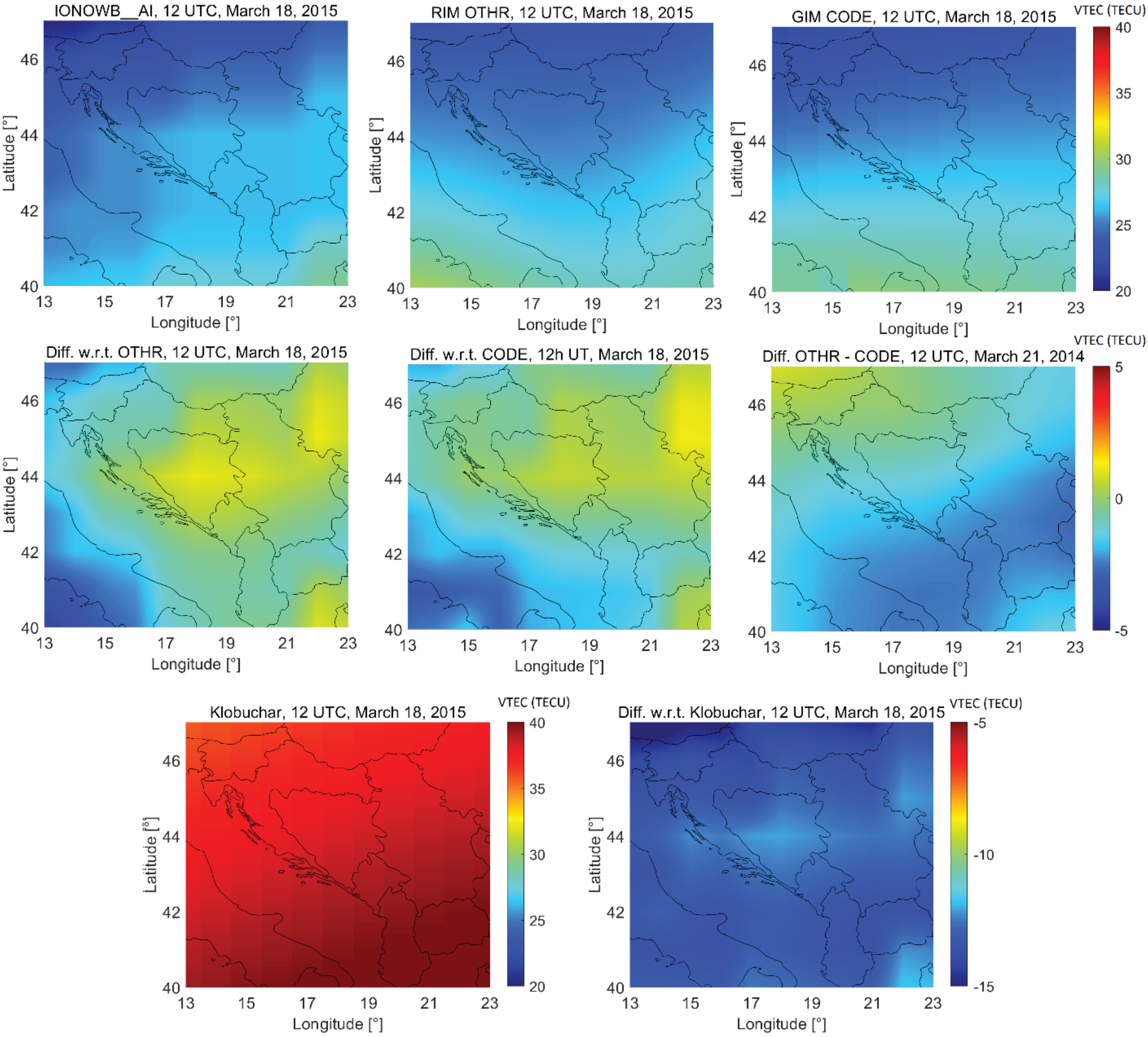

IONOWB__AI, 12 UTC, March 18, 2015
RIM OTHR, 12 UTC, March 18, 2015
GIM CODE, 12 UTC, March 18, 2015
VTEC (TECU)
40
35
30
25
20
Diff. w.r.t. OTHR, 12 UTC, March 18, 2015
Diff. w.r.t. CODE, 12h UT, March 18, 2015
Diff. OTHR - CODE, 12 UTC, March 21, 2014
VTEC (TECU)
5
0
-5
Klobuchar, 12 UTC, March 18, 2015
VTEC (TECU)
40
35
30
25
20
Diff. w.r.t. Klobuchar, 12 UTC, March 18, 2015
VTEC (TECU)
-5
-10
-15
Latitude [°]
46
44
42
40
Longitude [°]
13
15
17
19
21
23

FIGURE 14 VTEC maps (from left to right) for RIM IONOWB_AI, RIM OTHR, and GIM CODE (top). Middle: VTEC differences (from left to right): $\mathrm{VTEC}_{\mathrm{IONOWB\_AI}} - \mathrm{VTEC}_{\mathrm{OTHR}}$, $\mathrm{VTEC}_{\mathrm{IONOWB\_AI}} - \mathrm{VTEC}_{\mathrm{CODE}}$, and $\mathrm{VTEC}_{\mathrm{OTHR}} - \mathrm{VTEC}_{\mathrm{CODE.}}$ Bottom: VTEC map for the Klobuchar model (left) and the VTEC difference $\mathrm{VTEC}_{\mathrm{IONOWB\_AI}} - \mathrm{VTEC}_{\mathrm{Klob}}$ (right). All maps relate to 12 UTC on March 18, 2015.

TABLE 5
Overview of the Mean Differences for the Solar Maximum (March 21, 2014) and the Geomagnetic Storm (March 17–18, 2015) for the Region from 13°E to 23°E Longitude and from 40°N to 47°N Latitude. Mean values from both periods (final column) were calculated based on the mean differences from March 21, 2014, and the mean differences averaged over March 17–18, 2015.

Differences between models	Absolute mean differences (TECU)			
	March 21, 2014	March 17, 2015	March 18, 2015	Mean values
IONOWB_AI – OTHR	0.94	1.36	1.03	1.07
IONOWB_AI – CODE	1.48	1.40	1.05	1.35
OTHR – CODE	1.51	0.55	0.49	1.02
IONOWB_AI – Klob.	23.24	25.16	13.24	21.22

and Figure 14, respectively. VTECs from the RIM IONOWB_AI, RIM OTHR, GIM CODE, and the Klobuchar model were estimated on the 1° × 1° grid in latitude and longitude in the range from 40°N to 47°N and 13°E to 23°E. VTEC maps of RIM IONOWB_AI, RIM OTHR, and GIM CODE show the lowest daily ionization, mostly from 46°N to 47°N (upper parts of the maps), and the highest ionization from 40°N to 41°N (lower parts of the maps).

During solar maximum (Figure 12), RIMs IONOWB_AI and OTHR differed by less than 3 TECU with a mean difference of 0.9 TECU (Table 5). The largest differences between the RIMs were from 43°N 21°E to 44°N 23°E (3 TECU) and along 40°N (about 2 TECU). The smallest differences were in the areas covering Slovenia, Croatia, Bosnia-Herzegovina (BH; except around 44° latitude), and Montenegro with differences of less than 1 TECU and a mean difference of 0.4 TECU. Differences between the IONOWB_AI and the GIM CODE are up to approximately 4 TECU with a mean difference of 1.5 TECU. The largest differences were observed in the southern part of the map, where only a few IGS/EPN stations were used to estimate the GIM for the study area. By contrast, regions in which no GNSS stations were used to estimate the RIM IONOWB showed smaller differences. For example, the regions of Serbia and Kosovo (19°E–23°E, 42°N–46°N) showed a mean difference of 1.2 TECU compared to the RIM OTHR and 0.9 TECU compared to the GIM CODE. The average VTEC differences for the region of Montenegro (18°E–20°E and 42°N–44°N) were 0.8 TECU compared to RIM OTHR and 1.3 TECU compared to the GIM CODE. The VTEC difference maps between the RIM IONOWB_AI, RIM OTHR, and GIM CODE reveal unique VTEC features from the IONOWB_AI model with nonlinear spatial variations. The Klobuchar model underestimated VTEC by more than 20 TECU.

During the main phase of the storm (Figure 13), the ionization increased across the western Balkans. VTEC differences of RIM IONOWB_AI compared to RIM OTHR and GIM CODE show higher values in the northern part of the maps (up to 4.4 TECU). No GNSS observations from this region nor any of the more northern countries were used to generate the RIMs in this study; this led to the larger differences observed during the storm. The mean differences for the western Balkans regions, where GNSS observations were not used, were up to 1.2 TECU (Serbia, Kosovo) and 0.5 TECU (Montenegro) with respect to the RIM OTHR, and 1.4 TECU (Serbia, Kosovo) and 0.6 TECU (Montenegro) with respect to the GIM CODE. The VTEC values of the countries Slovenia, Croatia, and BH correspond better to the RIM OTHR with a mean difference of 0.5 TECU. The VTEC values in the southern part of this region correspond more closely with those from the GIM CODE with a mean difference of 0.7 TECU. The Klobuchar model underestimated VTEC by more than 24 TECU and failed to approximate the sudden VTEC increase during the main storm phase.

On the day after the main storm phase, the ionization was twice as low as on the previous day (Figure 14). The highest differences between the RIM IONOWB_AI, and the RIM OTHR and GIM CODE were identified in a region that included Italy, which was not part of this study. The average differences in the western Balkans countries were 0.6 TECU and 0.7 TECU between IONOWB_AI and the RIM OTHR and the GIM CODE, respectively. The mean differences over the western Balkans regions, with GNSS observations that were not used in this study, were 0.5 TECU (Serbia, Kosovo) and 0.6 TECU (Montenegro) compared to the RIM OTHR, and 0.6 TECU (Serbia, Kosovo) and 0.7 TECU (Montenegro) compared to the GIM CODE. The largest differences with regard to the RIM OTHR were along 44°N latitude (up to 1 TECU), while differences up to 0.6 TECU were detected for the same location compared to the GIM CODE. By contrast, the RIMs IONOWB_AI and OTHR showed better agreement in the area 18°E–23°E and 42°N–40°N with a mean difference of 0.6 TECU; the mean difference was twice as high. at 1.2 TECU compared to the GIM CODE. The Klobuchar model overestimated VTEC by more than 10 TECU, and therefore, cannot be used to approximate the decline in VTEC during the recovery phase of the storm.

Our results (Figure 9–Figure 14) reveal that results from the Klobuchar model deviate significantly from those generated by the newly-developed RIMs. Findings generated by the new RIMs correspond much more closely to the GIM CODE, which is considered the most accurate and precise source of VTEC information that is currently available. These results suggest that the new RIMs might replace the broadcasted Klobuchar model for applications involving single-frequency positioning.

3.2 | Single-Frequency Positioning Solutions

To assess the RIM IONOBH, vertical and horizontal RMS position errors were calculated from 24-hour solutions for March 2014 and March 2018 for selected stations (Figure 15).

Positioning solutions without ionospheric corrections revealed RMS vertical position errors of approximately 5.5 m and 1.3 m in March 2014 and March 2018, respectively. As expected, single-frequency positioning solutions with ionospheric corrections represented a significant improvement, especially with respect to the vertical component. When applying the final GIM CODE, the vertical RMS errors were between 0.7 and 1 m in March 2014 and between 0.1 and 0.3 m in March 2018. The application of the new RIM IONOBH led to a higher vertical position accuracy for all stations with an RMS error between 0.4 and 0.7 m in March 2014 and 0.1 and 0.2 m in March 2018. The horizontal RMS errors between the GIM CODE and

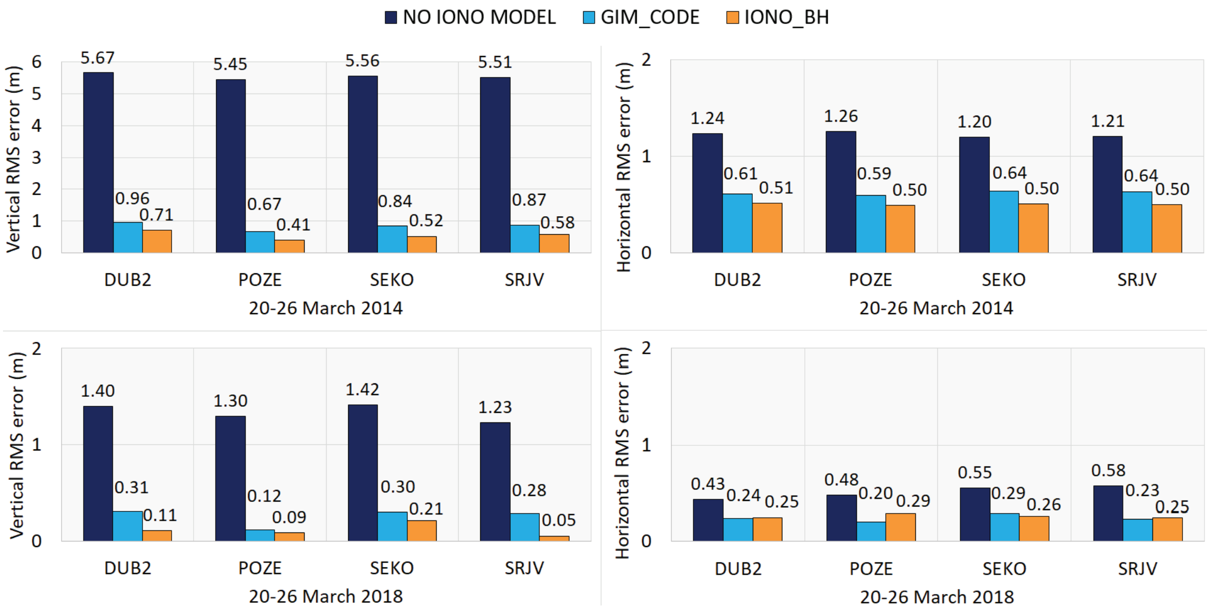


FIGURE 15 RMS errors of single-frequency positioning solutions without ionospheric corrections compared to the GIM CODE and the RIM IONO_BH. Shown are vertical position errors on March 20–26, 2014 (top left), horizontal position errors on March 20–26, 2014 (top right), vertical position errors on March 20–26, 2018 (bottom left), horizontal position errors on March 20–26, 2018 (bottom right).

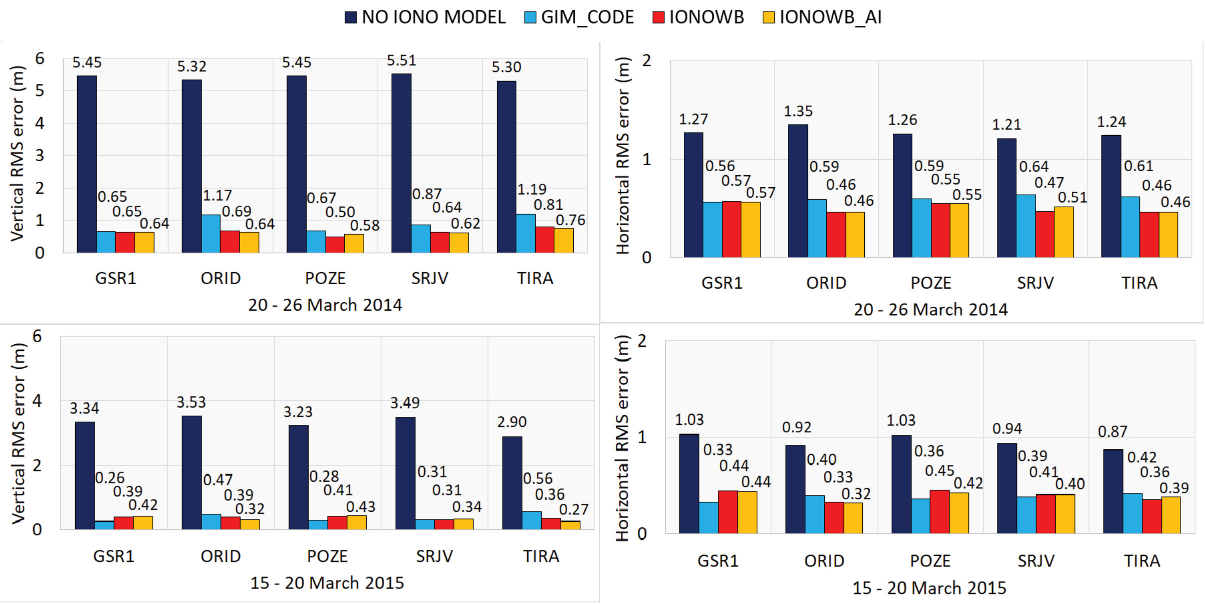


FIGURE 16 RMS errors of single-frequency positioning solutions without ionospheric corrections compared to the GIM CODE and the RIM IONO_WB. Shown are vertical position errors on March 20–26, 2014 (top left), horizontal position errors on March 20–26, 2014 (top right), vertical position errors on March 15–20, 2015 (bottom left), and horizontal position errors on March 15–20, 2015 (bottom right).

the RIM IONOBH were similar, at approximately 0.5 m for the RIM IONOBH and 0.6 m for the GIM CODE in March 2014 and 0.2 to 0.3 m in March 2018.

Figure 16 presents both vertical and horizontal RMS errors determined in March 2014 and March 2015 for selected stations that were used to assess the RIMs IONOWB and IONOWB_AI. Positioning solutions without ionospheric corrections had RMS vertical position errors of approximately 3.5 m in March 2015. Once the final GIM CODE was applied, the vertical RMS errors were between 0.6 and 1.2 m in March 2014 and 0.3 and 0.6 m in March 2015. After applying the new RIMs IONOWB and IONOWB_AI, the vertical RMS errors were between 0.5 and 0.8 m in March 2014 and 0.3 and 0.4 m in March 2015. The horizontal RMS errors were approximately 0.6 m for the CODE GIM and from 0.5 to 0.6 m for the RIMs IONOWB and IONOWB_AI in March 2014. In March 2015, the horizontal RMS errors were between 0.3 and 0.4 m for the CODE GIM, the RIMs IONO_WB, and IONOWB_AI. In March 2014, positioning errors after applying the IONOWB and IONOWB_AI corrections were at the same level as the GIM CODE (station GSR1) or better (for all the other stations). A significant improvement in position accuracy when using the RIMs IONOWB and IONOWB_AI was observed for stations SRJV, ORID, and TIRA in March 2014. In March 2015, both the vertical and horizontal position accuracy for the stations ORID and TIRA was improved by applying the RIMs IONOWB and IONOWB_AI. These stations are located in the lower part of the study region where larger VTEC differences between IONOWB_AI and GIM CODE were observed (Figure 9 and Figure 11). These differences may be attributed to the fact that information from very few stations was used to estimate the GIM CODE in this area; information from these stations was used in the newly-developed RIMs. Therefore, RIMs IONOWB and IONOWB_AI outperform the GIM CODE in this region. However, better positioning results were obtained with the GIM CODE in March 2015 stations located in the upper parts of the study region (i.e., GSR1 and POZE). There, larger VTEC differences between the RIM IONOWB_AI and GIM CODE were observed during the main storm phase

TABLE 6

RMS errors for Vertical (1D), Horizontal (2D), and 3D Position Solutions from Static 24-hour Positioning Data. Data were averaged over all stations examined. Improvement of the RMS 3D error was observed compared to the solutions generated without ionosphere corrections.

Study periods		RMS Vertical error	RMS Horizontal error	RMS 3D error	Improvement In 3D error
March 2014	**NO IONO**	5.46	1.25	5.49	
	GIM CODE	0.91	0.60	1.10	79.96%
	IONOBH	0.56	0.50	0.75	86.34%
	IONOWB	0.66	0.50	0.83	84.88%
	IONOWB_AI	0.65	0.51	0.83	84.88%
March 2015	**NO IONO**	3.30	0.96	3.43	
	GIM CODE	0.38	0.38	0.54	84.26%
	IONOWB	0.37	0.40	0.54	84.26%
	IONOWB_AI	0.35	0.39	0.55	83.97%
March 2018	**NO IONO**	1.34	0.51	1.45	
	GIM CODE	0.25	0.24	0.38	73.79%
	IONOBH	0.11	0.26	0.36	75.17%

(Figure 10) due to the larger VTEC differences in the neighboring countries, which are located outside the region examined in this study. Since a storm can induce significant variability in the ionosphere, the RIM for Slovenia would most likely benefit from including stations from other neighboring countries to improve position accuracy during a storm.

When using the L_1 frequency without ionospheric range delay corrections, the vertical position errors are approximately three to four times higher than the horizontal position errors (Figure 12 and Figure 13). After applying ionospheric corrections from the GIM CODE and the RIMs, the vertical accuracy was improved by 80 to 90% and the horizontal accuracy was improved by 50 to 60%. This is consistent with findings reported by Wang et al. (2013) who showed that ionospheric corrections reduce ionospheric vertical delay errors more effectively than horizontal delay errors. Also, the ionosphere models can produce biases with a common sign in the line-of-sight slant observations that may be gathered from negative elevations. The ionosphere model projection bias in the north-south and the east-west directions can be partially compensated by one another. In this study, the elevation angle was set to 15° to gather more observations covering wider regions while reducing observations from the negative elevations to avoid positioning error bias.

Table 6 depicts RMS vertical, 2D horizontal, and 3D positional errors averaged over all stations evaluated; the improvement in RMS 3D error is compared to the solutions without ionosphere corrections. The newly-developed RIMs improved the 3D positional accuracy by 85% or more during the solar maximum (March 2014); by contrast, the improvement observed with GIM CODE was approximately 80%. During the geomagnetic storm, the new RIMs improved the 3D positional accuracy by approximately 84%, which was similar to that resulting from the GIM CODE. Interestingly, the new RIMs improve the vertical positional accuracy more effectively than the GIM CODE for all study periods. At times of low solar activity (March 2018), improvements of approximately 74% and 75% were observed for the GIM CODE and the new RIMs, respectively.

Since the mean differences and RMS errors between the new RIMs and the Klobuchar model are at least two to three times higher than those between the new RIMs and the GIM CODE (Section 3.1), the expected positioning error using the Klobuchar model is at least twice as large. For example, the RMS position error is expected to be more than 2 m during a solar maximum. In March 2014, the VTEC of the Klobuchar model was on average approximately 15 TECU lower than the VTEC of the new RIMs. Considering that 1 TECU is equivalent to 0.162 m of L1 signal delay, a difference of 15 TECU results in a remaining L1 signal delay of approximately 2.4 m. This is about two or three times higher than the 3D position solutions obtained using the GIM CODE or the new RIMs, respectively. Even larger position errors could be expected during the March 2015 geomagnetic storm.

4 | DISCUSSION AND CONCLUSIONS

Regional VTEC models for BH (IONOBH), for specific countries in the western part of the Balkan peninsula (IONOWB), and the western Balkans region (IONOWB_AI) have been developed using the GNSS observations that were available from local CORS and the EPN networks in these countries. The IONOBH and IONOWB models are based on regional ionosphere coefficients of a Taylor series expansion estimated from the CORS observations processed in the Bernese GNSS Software. Machine learning techniques were also utilized to develop the IONOWB_AI model. The IONOWB_AI model includes coefficients of Taylor series expansion, spatial (latitude, longitude), temporal (time of day), solar (F10.7), and geomagnetic (Kp and Dst) parameters to estimate the VTEC at any specified position and time by applying a feed-forward neural network with backpropagation to minimize errors. These newly-developed RIMs were validated against GIMs, European RIMs, the NeQuick2 climatological empirical model, and the broadcasted Klobuchar model for different study periods including those featuring high and low solar cycle phases and quiet and disturbed geomagnetic periods. The spring equinox, when the highest VTEC values during a year can be expected in this study region, has also been covered. The new RIMs were applied in single-frequency positioning to evaluate their ability to mitigate the ionospheric refraction effect on precise positioning applications.

During the solar maximum, the differences between all examined models were higher than at the solar minimum, as expected. Differences between the VTEC values from the RIMs IONOBH and IONOWB and the GIMs were minor during the daytime and higher at night in March 2014. GIMs usually deliver higher VTEC values than the RIMs IONOBH and IONOWB. The mean difference in the solar maximum (March 2014) was approximately 5 TECU, while the mean difference at the solar minimum (March 2018) was approximately 2 TECU (i.e., reduced by more than a factor of 2). The largest differences were observed at night when the plasmaspheric contributions to VTEC prevail; agreement during the daytime is at least two times better. During the days before the storm (March 15–16, 2015), the mean differences were approximately 3 TECU. On the day of the severe storm (March 17, 2015), the mean differences were approximately 5 TECU, while during the recovery phase, the mean differences were below 2 TECU. Thus, we conclude that the differences between VTEC values from the RIMs IONOBH and IONOWB and those from the GIMs are dependent on the solar cycle phase and geomagnetic activity.

The analysis with respect to the European RIMs shows better agreements with the OTHR model than with the GIOMO model, especially during the maximum of the solar cycle phase (March 2014). Differences between the new RIMs and the OTHR are often smaller than differences between GIM CODE and the RIMs. However, differences with respect to the OTHR and the GIM CODE are quite similar to one another with similar trends, indicating small differences between these two models.

The largest discrepancy was between the newly-developed RIMs and the Klobuchar model. The Klobuchar model underestimated VTEC by more than 20 TECU during the daytime during the solar maximum and the main phase of the geomagnetic storm when compared to results from the new RIMs. In the recovery phase of the geomagnetic storm and at the solar minimum, the Klobuchar model overestimated VTEC by up to 20 TECU. The new RIMs correspond at least 4 to 10 times better to the GIM CODE than to the Klobuchar model, with mean differences from 2 to 5 TECU.

The NeQuick2 climatological model underestimated VTEC most of the time. During the solar minimum (March 2018), the NeQuick2 model was in much better agreement with the new RIMs than during the solar maximum (March 2014). During the main storm phase, differences with respect to the NeQuick2 model were the highest. These differences can be explained by the fact that the NeQuick2 provides a climatological description of the electron density in the ionosphere and its performance depends significantly on both solar and geomagnetic activity. Our results are consistent with those reported by Shi et al. (2019) and Wang et al. (2017).

The RIM IONOWB_AI, which is based on an ANN and the IONOWB regional ionosphere, spatiotemporal, solar, and geomagnetic parameters, can estimate VTEC for any location in the western part of the Balkan peninsula. In areas where no GNSS observations were used to estimate regional ionosphere parameters, the RIM IONOWB_AI provides VTEC with sufficient accuracy compared to the RIM OTHR and GIM CODE. Moreover, VTEC maps reveal non-linear VTEC signatures and regional ionosphere variations not seen in the GIM CODE. During the daily VTEC peaks (at 12 UTC) the mean differences in the study region are mostly below 1.50 TECU compared to findings generated by the RIM OTHR and GIM CODE. During the main storm phase, when the largest VTEC values were detected, the RIM IONOWB_AI estimated VTEC with differences that were primarily below 2 TECU (at 12 UTC) with respect to the RIM OTHR and the GIM CODE. During the recovery phase, the observed differences were even lower. By contrast, the Klobuchar model failed to describe the VTEC variations associated with geomagnetic activity. Collectively, our results demonstrate that the VTEC estimated by RIM IONOWB_AI provides a much better estimate of the integrated electron density than can be achieved using the Klobuchar model during the solar maximum or a geomagnetic storm.

Single-frequency positioning solutions show a similar or in some cases better positional accuracy when applying ionosphere corrections from the new RIMs (i.e., IONOBH, IONOWB, and IONOWB_AI) rather than from the final GIM CODE. Vertical positional accuracy can be achieved with these newly-developed RIMs at the decimeter level. Compared to the final GIM CODE, improvements in positioning with the new models are particularly effective when observed during the solar maximum. Also, the improvements are more significant for all study periods for stations located in the southern part of the region. This may be because of the higher ionization observed over those stations, and because no GNSS stations are used in this area to produce the GIM CODE. As a result, the RIMs show better performance in these regions. During the geomagnetic storm (March 2015), the positional accuracy achieved when applying RIMs IONOBH, IONOWB, and IONOWB_AI were within a similar range as that obtained with the final GIM CODE. Slightly better performance with the GIM CODE was achieved during March 2015 for stations located in the northern part of the study region (GSR1 and POZE). This may be because the GNSS stations from neighboring countries in this region were not used to generate the new RIMs. VTEC maps show larger differences when comparing outcomes from the RIM IONOWB_AI and GIM CODE in this northern area as a result of drastic changes in conditions in the ionosphere during the main phase of the severe geomagnetic storm. Given the fact that this was the strongest storm of the previous 11-year solar cycle and that modeling such intense ionospheric variations remains a significant challenge, the results obtained using the new RIMs are satisfactory. As the results indicate, positional accuracy depends strongly on solar and geomagnetic activity. Moreover, the newly developed IONOBH, IONOWB, and IONOWB_AI models facilitate the correction of the ionospheric refraction in a way that leads to increased positioning accuracy; improvements in 3D positioning of greater than 80% were observed during periods of solar maximum and severe geomagnetic storm.

The results presented here demonstrate that VTEC models generated for small regions from dense observations from the CORS networks can provide much more accurate estimates of VTEC than can be achieved using the Klobuchar model, which is currently used as the standard for single-frequency positioning. Additionally, a machine learning approach enabled us to generate spatiotemporal RIMs that can estimate VTEC for areas in which no GNSS observations were available. The results reveal that the new RIMs are in much better agreement with the GIM CODE than with the Klobuchar model. Taking into account that the GIM CODE is considered the most accurate and precise source of VTEC information currently available, the new RIMs might surpass this standard and significantly improve ionosphere modeling compared to the Klobuchar model. The new RIMs correct single-frequency range errors with an accuracy that is similar to that of the final GIM products and in some cases even better. Furthermore, the final GIMs are produced with a latency of several days or even weeks, and thus they are not well suited for real (or even near-real) time positioning applications. The available real-time Klobuchar model fails to provide an accurate description of the ionosphere and, consequently, it cannot precisely reduce the ionospheric signal refraction. The RIMs developed as part of this study can be generated from GNSS observations from the CORS network in near-real time (within approximately three hours) and corrections can be provided to the single-frequency GNSS users. The findings suggest that these new models may properly address the needs of single-frequency GNSS users and provide an effective ionospheric correction.

Recommendations and plans for future work include:

- Models with a higher temporal resolution may be beneficial for use in capturing sudden VTEC variations due to geomagnetic storms, traveling ionosphere disturbances, and other features.
- The addition of dense CORS observations from other countries within the study region can improve the accuracy of the RIMs and fill and/or reduce the observation gaps.
- Observations from the EPN stations near and further away from the border of the study region can be introduced to fill in the observation gaps between the IPPs if additional CORS observations are not available.
- GNSS data should be processed over a longer period to retrain the ANN model and thus improve its accuracy and applicability.
- The machine learning approach can be improved by automatically extracting features in the spatial domain using a convolutional neural network and those in the temporal domain using a recurrent neural network. It would be interesting to compare this combination with the approach presented in this paper.
- It may also be interesting to test the RIMs over a longer period.
- To facilitate their application in national positioning services, models should be implemented in near-real time.

FUNDING

This research was funded by OeAD-GmbH (ICM-2017-06548) within the framework of Ernst Mach Grant Worldwide; a Short-Term Research Grant from Deutscher Akademischer Austauschdienst (DAAD); and Technical University of Munich (TUM) within the framework of Open Access Publishing Program within the DEAL agreement.

CONFLICT OF INTEREST / COMPETING INTERESTS

The authors declare no conflict of interest.

AUTHORS' CONTRIBUTION

Conceptualization: R.N.; data curation: R.N., N.M, A.G., M.M.; methodology: R.N.; formal analysis: R.N., software: R.N., N.M.; validation: R.N., Dz.H.; investigation: R.N.; visualization: R.N.; writing – original draft: R.N.; writing - review and editing: all authors; supervision: M.S., R.W.; funding acquisition: R.N.

ACKNOWLEDGMENTS

The authors are grateful to the providers of CORS National permanent GNSS networks, including the Republic Administration for Geodetic and Property Law Affairs and Federal Administration for Geodetic and Property Affairs of Bosnia-Herzegovina for BIHPOS GNSS data; State Geodetic Administration of the Republic of Croatia for CROPOS GNSS data; Institute of Geosciences, Energy, Water, and Environment Tirana and Prof. B. Nurce for AlbGNSS GNSS data; Agency for the Real Estate Cadastre Republic of North Macedonia for MAKPOS GNSS data; and the Geodetic Institute of Slovenia for SIGNAL GNSS data. We also thank the EUREF network for EPN GNSS data, CODE for the GIM data, the IRI Working Group for the IRI-online computation service; NASA/GSFC's Space Physics Data Facility OMNIWeb service and CDDIS, which is NASA's Archive of Space Geodesy data. We also extend our thanks to N. Hanna for estimating VTEC from the Klobuchar model and we acknowledge the Open Source Geographic Information System (QGIS) for generating one of the figures. Finally, we thank the reviewers whose valuable comments and suggestions helped to clarify our thinking and improve the quality of this manuscript.

REFERENCES

Bergeot, N., Chevalier, J.-M., Bruyninx, C., Pottiaux, E., Aerts, W., Baire, Q., Legrand, J., Defraigne, P., & Huang, W. (2014). Near real-time ionospheric monitoring over Europe at the Royal Observatory of Belgium using GNSS data. *Journal of Space Weather and Space Climate*, *4*, A31. https://doi.org/10.1051/swsc/2014028

Bilitza, D. (2018). IRI the international standard for the ionosphere. *Advances in Radio Science*, *16*, 1–11. https://doi.org/10.5194/ars-16-1-2018

Boisits, J., Glaner, M., & Weber, R. (2020). Regiomontan: a regional high precision ionosphere delay model and its application in precise point positioning. *Sensors*, *20*(10), 2845. https://www.mdpi.com/1424-8220/20/10/2845

Bottou, L. (1991). Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nımes*, *91*(8), 12.

Boussard, M., Mars, C., Dès, R., & Chopinaud, C. (2017). Periodic split method: learning more readable decision trees for human activities. Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle,

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Camporeale, E., Wing, S., & Johnson, J. R. (2018). *Machine learning techniques for space weather*. Elsevier. https://doi.org/10.1016/B978-0-12-811788-0.09994-7

Chapman, S., & Bartels, J. (1962). *Geomagnetism.* (*v.2*) Clarendon Press.

Ciraolo, L., Azpilicueta, F., Brunini, C., Meza, A., & Radicella, S. M. (2007). Calibration errors on experimental slant total electron content (TEC) determined with GPS. *Journal of Geodesy*, *81*(2), 111–120. https://doi.org/10.1007/s00190-006-0093-1

Covington, A. E. (1969). Solar Radio Emission at 10.7 cm, 1947–1968. *Journal of the Royal Astronomical Society of Canada*, *63*, 125–132.

Dach, R., Lutz, S., Walser, P., & Fridez, P. (2015). *Bernese GNSS Software Version 5.2*. University of Bern, Bern Open Publishing. https://doi.org/10.7892/boris.72297

El-Diasty, M. (2017). Regional ionospheric modeling using wavelet network model. *The Journal of Global Positioning Systems*, *15*(1), 2. https://doi.org/10.1186/s41445-017-0007-y

Farzaneh, S., & Forootan, E. (2018). Reconstructing regional ionospheric electron density: a combined spherical Slepian function and empirical orthogonal function approach. *Surveys in Geophysics*, *39*(2), 289–309. https://doi.org/10.1007/s10712-017-9446-y

Feltens, J. (2003). The international GPS service (IGS) ionosphere working group. *Advances in Space Research*, *31*(3), 635–644. https://doi.org/10.1016/S0273-1177(03)00029-2

Goss, A., Schmidt, M., Erdogan, E., Görres, B., & Seitz, F. (2019). High-resolution vertical total electron content maps based on multi-scale B-spline representations. *Annals of Geophysics*, *37*(4), 699–717. https://doi.org/10.5194/angeo-37-699-2019

Goss, A., Schmidt, M., Erdogan, E., & Seitz, F. (2020). Global and regional high-resolution VTEC modelling using a two-step B-spline approach. *Remote Sensing*, *12*(7), 1198. https://www.mdpi.com/2072-4292/12/7/1198

Habarulema, J. B., McKinnell, L.-A., Cilliers, P. J., & Opperman, B. D. L. (2009). Application of neural networks to South African GPS TEC modelling. *Advances in Space Research*, *43*(11), 1711–1720. https://doi.org/10.1016/j.asr.2008.08.020

Hastie, T., Tibshirani, R., Friedman, J., & SpringerLink (Online service). (2009). *The Elements of Statistical Learning Data Mining, Inference, and Prediction* (2nd edition). Springer-Verlag. http://iclibezp1.cc.ic.ac.uk/login?url=https://doi.org/10.1007/978-0-387-84858-7

Hernández-Pajares, M., Juan, J. M., Sanz, J., Aragón-Àngel, À., García-Rigo, A., Salazar, D., & Escudero, M. (2011). The ionosphere: effects, GPS modeling and the benefits for space geodetic techniques. *Journal of Geodesy*, *85*(12), 887–907. https://doi.org/10.1007/s00190-011-0508-5

Hernández-Pajares, M., Juan, J. M., Sanz, J., Orus, R., Garcia-Rigo, A., Feltens, J., Komjathy, A., Schaer, S. C., & Krankowski, A. (2009). The IGS VTEC maps: a reliable source of ionospheric information since 1998. *Journal of Geodesy*, *83*(3), 263–275. https://doi.org/10.1007/s00190-008-0266-1

Hofmann-Wellenhof, B., Lichtenegger, H., & Collins, J. (2001). *Global positioning system theory and practice* (5th revised edition). Springer Wien.

IERS conventions. (2010). (IERS Technical Note, Issue 36). Verlag des Bundesamts für Kartographie und Geodäsie.

Jee, G., Lee, H.-B., Kim, Y. H., Chung, J.-K., and Cho, J. (2010), Assessment of GPS global ionosphere maps (GIM) by comparison between CODE GIM and TOPEX/Jason TEC data: Ionospheric perspective, J. Geophys. Res., 115, A10319, https://doi.org/10.1029/2010JA015432.

Jiang, H., Wang, Z., An, J., Liu, J., Wang, N., & Li, H. (2017). Influence of spatial gradients on ionospheric mapping using thin layer models. *GPS Solutions*, *22*(1), 2. https://doi.org/10.1007/s10291-017-0671-0

Kaselimi, M., Voulodimos, A., Doulamis, N., Doulamis, A., & Delikaraoglou, D. (2020). A causal long short-term memory sequence to sequence model for TEC prediction using GNSS observations. *Remote Sensing*, *12*(9), 1354. https://www.mdpi.com/2072-4292/12/9/1354

Kim, M., & Kim, J. (2019). Extending the coverage area of regional ionosphere maps using a support vector machine algorithm. *Annals of Geophysics*, *37*(1), 77–87. https://doi.org/10.5194/angeo-37-77-2019

Klobuchar, J. A. (1987). Ionospheric time-delay algorithm for single-frequency GPS users. *IEEE Transactions on Aerospace and Electronic Systems*, *AES-23*(3), 325–331. https://doi.org/10.1109/TAES.1987.310829

Leandro, R. F., & Santos, M. C. (2007). A neural network approach for regional vertical total electron content modelling. *Studia Geophysica et Geodaetica*, *51*(2), 279–292. https://doi.org/10.1007/s11200-007-0015-6

LeNail, A. (2019). Nn-svg: publication-ready neural network architecture schematics. *Journal of Open Source Software*, *4*(33), 747. https://doi.org/10.21105/joss.00747

Li, Z., Wang, N., Hernández-Pajares, M., Yuan, Y., Krankowski, A., Liu, A., Zha, J., García-Rigo, A., Roma-Dollase, D., Yang, H., Laurichesse, D., & Blot, A. (2020). IGS real-time service for global ionospheric total electron content modeling. *Journal of Geodesy*, *94*(3), 32. https://doi.org/10.1007/s00190-020-01360-0

Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020). Forecasting global ionospheric TEC using deep learning approach. *Space Weather*, *18*(11), e2020SW002501. https://doi.org/10.1029/2020SW002501

Liu, Q., Hernández-Pajares, M., Lyu, H., & Goss, A. (2021). Influence of temporal resolution on the performance of global ionospheric maps. *Journal of Geodesy*, *95*(3), 34. https://doi.org/10.1007/s00190-021-01483-y

Magnet, N. (2019). *Giomo: A robust modelling approach of ionospheric delays for GNSS real-time positioning applications*. [Dissertation, Vienna University of Technology]. https://doi.org/10.34726/hss.2019.21396

Mannucci, A. J., Wilson, B. D., Yuan, D. N., Ho, C. H., Lindqwister, U. J., & Runge, T. F. (1998). A global mapping technique for GPS-derived ionospheric total electron content measurements. *Radio Science*, *33*(3), 565–582. https://doi.org/10.1029/97RS02707

Motamedi, M., Sakharnykh, N., & Kaldewey, T. (2021). A data-centric approach for training deep neural networks with less data. *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*. https://doi.org/10.48550/arXiv.2110.03613

Natras, R., Halilovic, D., Mulić, M., & Schmidt, M. (2023). Mid-latitude ionosphere variability (2013–2016), and space weather impact on VTEC and precise point positioning. In: Ademović, N., Mujčić, E., Mulić, M., Kevrić, J., Akšamija, Z. (eds.) *Advanced Technologies, Systems, and Applications VII*. Lecture Notes in Networks and Systems (vol. 539). Springer. https://doi.org/10.1007/978-3-031-17697-5_37

Natras, R., & Schmidt, M. (2021). Machine learning model development for space weather forecasting in the ionosphere. In G. Cong & M. Ramanath, *CIKM 2021 Workshops* CEUR Workshop Proceedings, RWTH Aachen. http://ceur-ws.org/Vol-3052/short10.pdf.

Natras, R., Soja, B., & Schmidt, M. (2022). Ensemble machine learning of Random Forest, AdaBoost and XGBoost for vertical total electron content forecasting. *Remote Sensing*, *14*(15), 3547. https://doi.org/10.3390/rs14153547

Natras, R., Soja, B., & Schmidt, M. (2022). Machine learning ensemble approach for ionosphere and space weather forecasting with uncertainty quantification. 2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC). https://doi.org/10.23919/AT-AP-RASC54737.2022.9814334

Nava, B., Coïsson, P., & Radicella, S. M. (2008). A new version of the NeQuick ionosphere electron density model. *Journal of Atmospheric and Solar-Terrestrial Physics*, *70*(15), 1856–1862. https://doi.org/10.1016/j.jastp.2008.01.015

Okoh, D., Owolabi, O., Ekechukwu, C., Folarin, O., Arhiwo, G., Agbo, J., Bolaji, S., & Rabiu, B. (2016). A regional GNSS-VTEC model over Nigeria using neural networks: a novel approach. *Geodesy and Geodynamics*, *7*(1), 19–31. https://doi.org/10.1016/j.geog.2016.03.003

Orus Perez, R. (2019). Using TensorFlow-based neural network to estimate GNSS single frequency ionospheric delay (IONONet). *Advances in Space Research*, *63*(5), 1607–1618. https://doi.org/10.1016/j.asr.2018.11.011

Orus Perez, R., Parro-Jimenez, J. M., & Prieto-Cerdeira, R. (2018). Status of NeQuick G after the solar maximum of cycle 24. *Radio Science*, *53*(3), 257–268. https://doi.org/10.1002/2017RS006373

Orús, R., Hernández-Pajares, M., Juan, J. M., Sanz, J., & García-Fernández, M. (2002). Performance of different TEC models to provide GPS ionospheric corrections. *Journal of Atmospheric and Solar-Terrestrial Physics*, *64*(18), 2055–2062. https://doi.org/10.1016/S1364-6826(02)00224-9

Radicella, S. M., & Nava, B. (2020). Chapter 6: Empirical ionospheric models. In Materassi, M., Forte, B., Coster, A. J. & Skone, S (Eds.), *The Dynamical Ionosphere*. Elsevier. pp. 39–53 https://doi.org/10.1016/B978-0-12-814782-5.00006-6

Rajpurkar, P., Park, A., Irvin, J., Chute, C., Bereket, M., Mastrodicasa, D., Langlotz, C. P., Lungren, M. P., Ng, A. Y., & Patel, B. N. (2020). AppendiXNet: deep learning for diagnosis of appendicitis from a small dataset of CT exams using video pretraining. *Scientific Reports*, *10*(1), 3958. https://doi.org/10.1038/s41598-020-61055-6

Ramchoun, H., Idrissi, M. A. J., Ghanou, Y., & Ettaouil, M. (2016). Multilayer perceptron: architecture optimization and training. *International Journal of Interactive Multimedia and Artificial Intelligence*, *4*(1), 26–30. https://doi.org/10.9781/ijimai.2016.415

Ridley, A. J., Deng, Y., & Tóth, G. (2006). The global ionosphere–thermosphere model. *Journal of Atmospheric and Solar-Terrestrial Physics*, *68*(8), 839–864. https://doi.org/10.1016/j.jastp.2006.01.008

Schaer, S. (1999). *Mapping and predicting the Earth's ionosphere using the global positioning system.* [Dissertation, The University of Bern].

Schunk, R. W., Scherliess, L., Sojka, J. J., Thompson, D. C., Anderson, D. N., Codrescu, M., Minter, C., Fuller-Rowell, T. J., Heelis, R. A., Hairston, M., & Howe, B. M. (2004). Global assimilation of ionospheric measurements (GAIM). *Radio Science*, *39*(1), RS1S02. https://doi.org/10.1029/2002RS002794

Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, *4*(12), 310–316. https://ijeast.com/papers/310-316,Tesma412,IJEAST.pdf

Shi, C., Zhang, T., Wang, C., Wang, Z., & Fan, L. (2019). Comparison of IRI-2016 model with IGS VTEC maps during low and high solar activity period. *Results in Physics*, *12*, 555–561. https://doi.org/10.1016/j.rinp.2018.12.022

Sugiura, M. (1964). Hourly values of equatorial Dst for the IGY. *Annals of the International Geophysical Year*, *35*, 9–45.

Wang, N., Yuan, Y., Li, Z., & Huo, X. (2013). Impact of ionospheric correction on single-frequency GNSS positioning. In Sun, J., Jiao, W., Wu, H. & Shi, C. *China Satellite Navigation Conference (CSNC) 2013 Proceedings* Berlin, Heidelberg.

Wang, N., Yuan, Y., Li, Z., Li, Y., Huo, X., & Li, M. (2017). An examination of the Galileo NeQuick model: comparison with GPS and JASON TEC. *GPS Solutions*, *21*(2), 605–615. https://doi.org/10.1007/s10291-016-0553-x

Wild, U. (1994). *Ionosphere and geodetic satellite systems permanent GPS tracking data for modelling and monitoring*. [Dissertation, The University of Bern].

Zhang, Z., Pan, S., Gao, C., Zhao, T., & Gao, W. (2019). Support vector machine for regional ionospheric delay modeling. *Sensors*, *19*(13), 2947. https://doi.org/10.3390/s19132947.

Zhao, T., Pan, S., Gao, W., Qing, Z., Yang, X., & Wang, J. (2021). Extreme learning machine-based spherical harmonic for fast ionospheric delay modeling. *Journal of Atmospheric and Solar-Terrestrial Physics*, *216*, 105590. https://doi.org/10.1016/j.jastp.2021.105590

Zheng, A., & Casari, A. (2018). *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. O'Reilly Media, Inc.

How to cite this article: Natras, R., Goss, A., Halilovic, D., Magnet, N., Mulic, M., Schmidt, M., & Weber, R. (2023). Regional ionosphere delay models based on CORS data and machine learning. *NAVIGATION, 70*(3). https://doi.org/10.33012/navi.577

A2 Publication II

Reference

Natras, R., Soja, B., and Schmidt, M. (2022a). **Ensemble Machine Learning of Random Forest, AdaBoost and XGBoost for Vertical Total Electron Content Forecasting**. Remote Sensing, 14(15): 3547. https://doi.org/10.3390/rs14153547

Copyright

This paper was published in the open access journal Remote Sensing, MDPI. It is available under the Creative Commons license at https://www.mdpi.com/journal/remotesensing. The copyrights remain with the authors.

Declaration of own contribution

The concept and outline of the paper were suggested by R. Natras and discussed with all co-authors. R. Natras proposed the machine learning methodology. The machine learning-based ionosphere models were developed by R. Natras and implemented in the Python programming language and the ScikitLearn open-source software machine learning library (Pedregosa et al., 2011). The computations as well as the analyses and evaluations were performed by R. Natras with the valuable input of the co-authors. All co-authors supported the progress by discussing the results and suggesting validation and analysis approaches. All figures and tables included in the paper were compiled by R. Natras with suggestions from co-authors. The figures were created in Python using Seaborn (Waskom, 2021) and Matplotlib (Hunter, 2007) and in MATLAB (MATLAB, 2020). The manuscript was written by R. Natras, and reviewed and edited by the co-authors.
The overall contribution of R. Natras to P-II is estimated to be 86%.

remote sensing

MDPI

Article

Ensemble Machine Learning of Random Forest, AdaBoost and XGBoost for Vertical Total Electron Content Forecasting

Randa Natras [1,*], Benedikt Soja [2] and Michael Schmidt [1]

[1] Deutsches Geodätisches Forschungsinstitut (DGFI-TUM), TUM School of Engineering and Design, Technical University of Munich, 80333 Munich, Germany; mg.schmidt@tum.de

[2] Institute of Geodesy and Photogrammetry, ETH Zurich, 8093 Zurich, Switzerland; soja@ethz.ch

[*] Correspondence: randa.natras@tum.de

Abstract: Space weather describes varying conditions between the Sun and Earth that can degrade Global Navigation Satellite Systems (GNSS) operations. Thus, these effects should be precisely and timely corrected for accurate and reliable GNSS applications. That can be modeled with the Vertical Total Electron Content (VTEC) in the Earth's ionosphere. This study investigates different learning algorithms to approximate nonlinear space weather processes and forecast VTEC for 1 h and 24 h in the future for low-, mid- and high-latitude ionospheric grid points along the same longitude. VTEC models are developed using learning algorithms of Decision Tree and ensemble learning of Random Forest, Adaptive Boosting (AdaBoost), and eXtreme Gradient Boosting (XGBoost). Furthermore, ensemble models are combined into a single meta-model Voting Regressor. Models were trained, optimized, and validated with the time series cross-validation technique. Moreover, the relative importance of input variables to the VTEC forecast is estimated. The results show that the developed models perform well in both quiet and storm conditions, where multi-tree ensemble learning outperforms the single Decision Tree. In particular, the meta-estimator Voting Regressor provides mostly the lowest RMSE and the highest correlation coefficients as it averages predictions from different well-performing models. Furthermore, expanding the input dataset with time derivatives, moving averages, and daily differences, as well as modifying data, such as differencing, enhances the learning of space weather features, especially over a longer forecast horizon.

Keywords: machine learning; ensemble learning; ionosphere; Vertical Total Electron Content (VTEC) forecasting; space weather

check for updates

Citation: Natras, R.; Soja, B.; Schmidt, M. Ensemble Machine Learning of Random Forest, AdaBoost and XGBoost for Vertical Total Electron Content Forecasting. *Remote Sens.* **2022**, *14*, 3547. https://doi.org/10.3390/rs14153547

Academic Editor: José Fernández

Received: 21 June 2022
Accepted: 16 July 2022
Published: 24 July 2022

1. Introduction

Space weather is recognized as the greatest risk to the Global Navigation Satellite System (GNSS) [1]. As our society is heavily dependent on GNSS applications that require high-precision positioning, navigation, and timing, it is urgently necessary to develop advanced forecasting methods of the space weather impact on GNSS in order to mitigate the catastrophic consequences of this hazard. The impact of space weather and ionosphere on GNSS signals can be estimated from GNSS observations in the form of the Slant Total Electron Content (STEC) [2–4], which is proportional to the relative ionospheric delay of GNSS signals. STEC is usually mapped to the vertical TEC (VTEC) by approximating the ionosphere as a single layer model, assuming that all free electrons are concentrated within a shell of infinitesimal thickness. VTEC exhibits latitudinal and longitudinal variations, diurnal, seasonal, semi-annual, and sunspot cycle variations, as well as coupling effects [5–7]. Furthermore, space weather can produce intense, irregular ionosphere variabilities, which can be difficult to model with traditional mathematical approaches and to properly minimize in positioning solutions, degrading positioning and navigation performances [8–11]. A complex chain of physical and dynamical space weather processes between the Sun, the interplanetary space, the Earth's magnetic field, and the ionosphere must be taken into account when modeling and forecasting these disturbances in the ionosphere. However, we have a limited understanding of these coupled processes and often do not have defined functions that can describe them precisely. On the other hand, artificial intelligence and machine learning offer a new possibility to learn these relationships directly from data, discover the hidden relationships and find functions that describe space weather processes.

Machine learning is today one of the most rapidly growing areas [12]. It is suitable for problems that are too complex or vast for traditional approaches, or for which there is no known solution at all, by offering a new possibility of learning directly from data, as opposed to traditional modeling approaches of explicitly defining rules/functions to describe relationships and patterns in data [13]. Recently, machine learning methods have been attracting considerable interest in many technical and scientific fields, including space weather research [14] with a focus on modeling the nonlinear relations that can describe the underlying physical behavior of the system. The previous machine learning applications to the VTEC mostly include deep learning with artificial neural network (ANN). Different ANN architectures were proposed such as feed-forward ANN [15–17], recurrent ANN such as popular Long Short-Term Memory (LSTM) [18–20] and combined with convolution (LSTM-CNN) [21,22], Encoder-Decoder LSTM Extended (ED-LSTME) [23], neural network autoregressive with external input (NARX) [24,25], conditional Generative Adversial Network (cGAN) [26], as well as Adaptive Neuro-Fuzzy Inference System (ANFIS) [27,28]. Only a few of the earlier work applied machine learning methods outside of deep learning, such as Gradient Boosting Decision Tree (GBDT) [27], eXtreme Gradient Boosting (XGBoost) [29], Support Vector Machine (SVM) [30] and nearest neighbour [31].

To represent the impact of solar activity on VTEC, the solar radio flux F10.7 is usually used as an input to a machine learning model, while less often sunspot number or EUV index. Some studies also introduced the solar zenith angle alongside the F10.7 index. Geomagnetic activity is usually represented with Kp index and/or Ap index, Dst index and auroral electrojet indices, or time-weighted Ap, Kp and Dst indices. Diurnal VTEC variations are commonly modeled with hour of day, while day of year is used for extracting seasonal VTEC variations. Additionally, input data include information on previous VTEC values. Furthermore, geographic coordinates and geomagnetic latitude are also incorporated when developing a single model for different VTEC grids/regions. However, some studies have different approaches, for instance, using spherical harmonics [18], Taylor series expansion [16] or principal components of VTEC as input [28]. Moreover, there are models developed solely on VTEC input and output data, such as [31]. VTEC data were usually extracted from Global Ionosphere Maps (GIM) such as CODE [15,30], IGS [24,32], UPC-IonSAT [31] or calculated directly from raw GNSS observations, such as from the CORS (Continuous Operating Reference Stations) observations [16]. Previous studies demonstrate that the input data selection for a machine learning model significantly influences the model prediction, and consequently its accuracy. The temporal resolution of VTEC machine learning models is often 1 h.

Current state of research is mostly related to short-time forecasting from 1 h forecasting, such as in [15], to 1-day, such as in [24,33], and 2 day forecasting [31].

Regarding the spatial extent of the studies, forecasting was mostly performed for a single or few GNSS stations or grids. In addition, some studies have been done for regional, such as [16,17], and global modeling, such as [31]. For the training, various data lengths were used, from less than 1 year, several years until covering one or two solar cycles. However, most of the discussed studies used 2 to 3 years of data length. Ruwali et al. [22] and Srivani et al. [19] observed that deep learning VTEC models increase their accuracy significantly with increasing training dataset length.

Regarding the VTEC model performance, the RMSE for 1 h VTEC forecast in low-latitudes ranges from 2 to 5 TECU with different learning algorithms and different levels of solar activity [21,25,27,30]. For the mid-latitude 1 h VTEC forecast, RMSE is about 1.5 TECU in 2018 [30], while 1 day VTEC forecast has an RMSE of 4 TECU in high solar activity and 2 TECU in low solar activity [32]. Accuracy of the 1 day VTEC forecast globally is about 3 to 5 TECU depending on the models and level of solar activity [24,26,31]. The machine learning VTEC models outperform the traditional linear methods, such as the empirical orthogonal function (EOF) [34] and the autoregressive integrated moving average (ARIMA) method [20]. Moreover, the global XGBoost VTEC model provides a lower RMSE of around 1 TECU than the ANN model in 2017 [29]. The GBDT VTEC model outperforms ANN and LSTM VTEC models by about 6% during high solar activity [27]. The largest errors in machine learning VTEC models have been observed for the equatorial anomaly region and space weather events. Overall, the results of previous studies demonstrate that machine learning can find nonlinear patterns in the data and outperforms traditional linear modeling methods.

Previous Research Gaps and Our Contribution

A review of previous work reveals that most machine learning VTEC approaches have been proposed for different types of ANN, i.e., deep learning methods. However, there are many other methods in the field of machine learning which has either not been investigated or have been limited discussed. The probable reason for this gap is that deep learning methods have been widely known as the ground-breaking methods today in various fields such as automotive driving, speech-recognition. However, as a matter of fact, it can be claimed that other machine learning methods have significantly good performance in the analysis of small datasets, whereas deep learning is often incapable of performing this task and tend to easily overfit the data [35]. The capabilities of different types of machine learning methods have not been so far explored nor used widely for VTEC modeling. Previous results [27,29] confirm remarkably well performance of such machine learning methods for VTEC forecast, much better than neural networks. This is probably due to the issue of the limited training dataset, where other machine learning methods outperform deep learning methods. These machine learning methods have been less commonly discussed in the overall previous VTEC studies, which are predominantly based on deep learning. In addition, we realized that this limited number of studies was restricted to a few machine learning methods. On the other hand, deep learning methods are already studied in detail in numerous papers, and they also require "big data" in order to exploit their full potential, as already reported [19,22], as they are often overparametrized, which tends to overfit the data [35]. As a result, a model can correspond too closely to the training data to the extent that it negatively impacts the model performance on new data, i.e., reducing its ability to generalize. Moreover, the complexity of using deep learning methods was another motivation for selecting a more simple approach to the problem of VTEC forecasting.

Therefore, we want to bridge this gap and perform studies in direction of exploring new learning algorithms for VTEC forecast. In this context, we introduce new methods for VTEC forecast such as Random Forest and Adaptive Boosting (AdaBoost) (Section 2.3.2), while GBDT and XGBoost have been so far reported in only one paper each [27,29]. Moreover, we combine different models into a meta-ensemble via Voting Regressor to produce a model of higher accuracy with improved generalization (Table 1).

The way the data are partitioned and the model is validated can introduce additional bias into the machine learning model as it influences its architecture and parameter selection. Previous studies implemented simple hold-out validation or the classic k-fold validation technique. In the hold-out procedure, the data are divided into subsets of fixed data points: training data (mainly comprising 70% to 90% of the dataset), while the remaining part of the data is equally divided into validation and test datasets. On the other hand, in the k-fold cross-validation data are randomly partitioned into k equally sized folds containing different training and validation data points in each iteration. K-fold cross-validation is shown to be more accurate than the simple holdout method, because it can reduce variance and hence, decrease the overfitting problem [36]. However, if observations are temporally dependent, the simple k-fold cross-validation can be problematic and should be modified [37], since the training and validation samples are no longer independent. Ghaffari Razin and Voosoghi [28], however, used different testing method of splitting

24 VTEC values each day in 12 values for training $(1, 3, 5, \ldots, 23$ UT) and 12 for testing $(2, 4, 6, \ldots, 24$ UT). This approach is also problematic for temporally dependent VTEC, leading to simple interpolation of VTEC values, and furthermore, dataset contamination, when training data are not carefully distinguished from validation or testing data. Because of these reasons, we are not following previous approaches.

We propose a more appropriate approach for time-series forecasting by modifying classic k-fold cross-validation into time series cross-validation. In the modified version, we apply rolling cross-validation to VTEC forecasting, which is more suitable for a time-series problem, by following [38]. Contrary to the standard k-fold cross-validation, here the VTEC model is not trained on subsequent observations and forecasted on previous (past) observations. This would result in past data being predicted using the model that is trained on future (i.e., subsequent) data. It makes no sense to use the values from the future to forecast values in the past. In addition, we want to avoid looking into the future when training the model. Furthermore, when using the classic k-fold method for VTEC forecasting, the models are trained on observations prior to and after specific time periods and then forecasting is performed for time periods in between. This represents also the interpolation of datapoints between the time frames for which the model was trained. It can lead to more optimistic results and introduce bias in model selection and architecture, as already mentioned. In VTEC forecasting, there is a temporal dependency between observations, and this relationship needs to be preserved during validation/testing. Time series cross-validation preserves a temporal dependency, where a model is evaluated on a rolling basis using many data folds (Section 2.4.1).

Furthermore, a new set of input data is introduced. Reviewing previous work we noticed that a similar set of input features has been mostly used. In this study, the input data are expanded with new observations, such as solar wind plasma speed, index of the interplanetary magnetic field, as well as derived features of first and second derivatives and moving averages (Table 1). Since the machine learning model accuracy is highly dependent on the data, we gave special attention and consideration to the selection and derivation of appropriate input features that can precisely describe complex VTEC variations. In addition, systematic analysis, selection and preparation of input data, and selection of data timeframes in a way to enhance machine learning model performance are addressed and pointed out in the paper, especially for learning rare space weather events (Section 3.1). Furthermore, daily differences of input and output data are estimated and machine learning models for VTEC forecast have been trained for the first time on differences, besides the original data (Section 2.1). Machine learning performance on differences (de-trended data) is discussed compared to the original dataset (Section 3). This study also discusses the contributions of input predictors to the VTEC forecast.

To sum up, this paper presents a novel approach for forecasting VTEC and space weather impact, with the following main contributions and innovations:

1. Machine learning methods of bagging and boosting are introduced for the VTEC forecasting problem.
2. Tree-based learning algorithms are applied to overcome the deficiencies of the commonly used deep learning approaches to VTEC forecasting in terms of complexity, "big data" requirements, and highly parameterized model (prone to overfitting the data). Here, we adopted learning algorithms for VTEC forecasting that are simple, fast to optimize, computationally efficient, and usable on a limited dataset.
3. Moreover, we introduce an ensemble meta-model that combines predictions from multiple well-performing VTEC models to produce a final VTEC forecast with improved accuracy and generalization than each individual model.
4. Time series cross-validation method is proposed for VTEC model development to preserve a temporal dependency.
5. Additional VTEC-related features are added, such as first and second derivatives, and moving averages. Special attention is also paid to time period selection and relations within the data to have more space weather examples and near-solar maximum

conditions, as well as to enable learning and forecasting of complex VTEC variations, including space weather-related ones.

6. Machine learning models are trained and optimized solely using daily differences (de-trended data) along the models with original data.
7. The relative contribution of the input data to the VTEC forecast is analyzed to provide an insight into what the model has learned, and to what extent our physical understanding of important predictors has increased.

Table 1. Overview of input and output data for machine learning models.

Input Data (Time Moment: *i*)	Output Data (*i* + 1 h, *i* + 24 h)
Day of year (DOY) Hour of day (HOD) Sunspot number (R) Solar radio flux (F10.7) Solar wind (SW) plasma speed Interplanetary magnetic field (IMF) Bz index Geomagnetic field (GMF) Dst index GMF Kp index·10 AE index VTEC (10°70°, 10°40°, 10°10°) EMA of VTEC over previous 30 days EMA of VTEC over previous 4 days (96 h) First VTEC derivative (VTEC$'$) Second VTEC derivative (VTEC$''$)	VTEC (10°70°, 10°40°, 10°10°)

During this study, the following questions were raised:

1. Can other, simpler learning algorithms than ANN capture diverse VTEC variations for 1 h and 24 h VTEC forecasts?
2. Can ensemble meta-model achieve better performance than a single ensemble member?
3. How can VTEC models be improved in terms of data and input features? Also, does the new input dataset bring new information to the VTEC model?
4. Can data modification, such as differencing, enhance the VTEC model learning and generalization?

2. Methodology

The VTEC model based on machine learning "learns" directly from the historical data or given examples, which can be understood as past experiences. Learning is achieved by optimizing the performance of a machine learning algorithm for a task of VTEC prediction, which presents the prediction of a continuous variable, commonly referred to as regression in machine learning.

2.1. Data Selection and Preparation

Machine learning is based on data. Therefore, they are impacting the performance of machine learning algorithms to a big extent. Thus, when developing a machine learning model, it is essential to select and prepare data in a way that enables model learning. In addition, the data have to be representative of new cases that may arise in practice in order to generalize well.

In this paper we use supervised learning, where the set of measurements of both input and output data need to be clearly specified and prepared, known as training data, in order to construct the prediction function. Let us define a training sample of vector $\mathbf{x}_i$ and an output $y_i = F(\mathbf{x}_i)$ at time stamp i with $i = 1, 2, \ldots, N$ as in Equation (2). Whereas, the vectors $\mathbf{x}_i$ can be interpreted as the rows of the $N \times P$ predictor matrix $\mathbf{X}$, the columns represents the input variables $\widetilde{\mathbf{x}}_p$ with $p = 0, 1, 2, \ldots, P-1$. The components $x_{i,p}$ of the $N \times 1$ column vector $\widetilde{\mathbf{x}}_p = [x_{1,p}, x_{2,p}, \ldots, x_{N,p}]^T$ represent a time series of the pth input variable. A series of N observations $(\mathbf{x}_i, y_i)$ was prepared for the training and the cross-validation, from January 2015 to December 2016, while the test dataset covers the period from January 2017 to December 2017. The VTEC values for three grid points at high-latitude (10°, 70°), mid-latitude (10°, 40°), and low-latitude (10°, 10°) were extracted from the Global Ionosphere Maps (GIM) of CODE (Center for Orbit Determination in Europe), while data of solar and magnetic activity were obtained from NASA/GSFC's OMNIWeb [39]. Therefore, VTEC from GIM CODE was assumed to be the ground truth in this study. Three grid points for VTEC values were selected along the same longitude (10°) in order to represent latitudinal VTEC variations corresponding to different ionosphere regions (low- mid- and high-latitude), alongside other VTEC variabilities. In addition, the hour of the day (HOD) and the day of the year (DOY) were added as input to model the VTEC temporal dependencies. In addition, new input quantities were calculated, such as the exponential moving average (EMA), and first and second time derivatives of VTEC, denoted as VTEC′ and VTEC″. Forecasting is performed for 1 h and 24 h in the future (Equation (2)). Table 1 provides an overview of the data. Separate models were developed for each grid point and each forecast horizon. The dataset for training and cross-validation (January 2015–December 2016) comprises of totally 17,544 examples, while the test dataset (January–December 2017) contains 8760 examples. Datasets were prepared with 1 h temporal resolution.

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \dots \\ \mathbf{x}_N^T \end{bmatrix} = \begin{bmatrix} x_{0,1}, x_{1,1}, \dots, x_{P-1,1} \\ x_{0,2}, x_{1,2}, \dots, x_{P-1,2} \\ \dots \\ x_{0,N}, x_{1,N}, \dots, x_{P-1,N} \end{bmatrix} = \begin{bmatrix} DOY_1, HOD_1, \dots, VTEC_1'' \\ DOY_2, HOD_2, \dots, VTEC_2'' \\ \dots \\ DOY_N, HOD_N, \dots, VTEC_N'' \end{bmatrix}, \tag{1}$$

$$\mathbf{X} = \left[\widetilde{\mathbf{x}}_0, \widetilde{\mathbf{x}}_1, \dots, \widetilde{\mathbf{x}}_{P-1}\right],$$

$$\mathbf{y} = \mathbf{VTEC}(i+t) = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_N \end{bmatrix} = \begin{bmatrix} VTEC_{1+t} \\ VTEC_{2+t} \\ \dots \\ VTEC_{N+t} \end{bmatrix},$$

where $t = 1$ for the 1 h forecasting and $t = 24$ for the 24 h forecasting, for abbreviations see Table 1.

Firstly, the data were preprocessed and prepared with an 1 h time sampling. A few missing values encountered are replaced with the average value of a previous and subsequent value. Some data were not provided as 1 h samples, such as F10.7 and R (24 h samples) and Kp (3 h samples). There, values were interpolated with the previous one, as it is done at the OMNIWeb. Afterwards, two approaches followed:

1. After preprocessing, the data $(\mathbf{x}_i, y_i)$ for $i = 1, 2, \dots, N$ are used for the machine learning algorithm. In this paper this dataset is referred as non-differenced data.
2. Data (except HOD, DOY, EMA and the time derivatives) are time-differenced $(\Delta\mathbf{x}_i, \Delta y_i)$ by calculating the difference between an observation at time $t + 24$ h and an observation at time step i, i.e., $\Delta\mathbf{x}_i = \mathbf{x}_{i+24} - \mathbf{x}_i$ and $\Delta y_i = y_{i+24} - y_i$. Differencing was used to reduce temporal dependence and trends, as well as, stabilize mean of the dataset [38], by reducing daily variations. In this paper this dataset is referred as differenced data. Values of EMA and time derivatives were calculated from differenced VTEC. At the end, predicted VTEC differences were reconstructed by adding up the VTEC values from the previous day.

2.2. Supervised Learning

Supervised learning can be seen as the function estimation or predictive learning problem. The learning task can be stated as follows: given the values of an input vector $\mathbf{x}_i$ (predictor or the independent variable) the aim is to find an approximation $\widehat{F}(\mathbf{x}_i)$ of the function $F(\mathbf{x}_i)$ which maps the input $\mathbf{x}_i$ to the output y_i (response or the dependent variable) and provides a prediction denoted by $\widehat{y}_i$ as

$$\begin{aligned} y_i + e_i = \widehat{y}_i = \widehat{VTEC}_{(i+t)} = \widehat{F}(\mathbf{x}_i) \\ \mathbf{x}_i = [DOY_i, HOD_i, R_i, F10.7_i, SW_i, Bz_i, Dst_i, Kp_i, AE_i, \\ VTEC_i, VTEC_{EMA(30)}i, VTEC_{EMA(4)}i, VTEC_i', VTEC_i'']^T \end{aligned} \tag{2}$$

where e_i is an error. $\widehat{F}(\cdot)$ refers to the approximation function of the nonlinear relationship between the output value of the VTEC forecast and the input vector considering solar, interplanetary and geomagnetic indices, as well as the previous VTEC values. This function is unknown, and is therefore, approximated by optimizing learning algorithms for the task of VTEC forecasting. Using prepared training samples of input and an output in Equation (2), an approximation $\widehat{F}(\mathbf{x}_i)$ of the function $F(\mathbf{x}_i)$ is estimated by minimizing the value of objective (loss) function L. Employed objective function in this study is the squared error

$$L = \frac{1}{N}\sum_{i=1}^{N} e_i^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2. \tag{3}$$

In this way, the function that describes the input/output relationship is modified as a response to differences between the real VTEC value y_i and generated VTEC prediction $\widehat{y}_i$. This represents learning by examples commonly referred as learning or training phase [35].

2.3. Tree-Based Machine Learning Algorithms

Tree-based algorithms are conceptually simple, but powerful machine learning methods that can perform well on both small and large datasets to solve linear and nonlinear modeling problems. Several tree-based machine learning algorithms have been applied in this study, namely Regression tree and ensemble learning such as Random Forest, eXtreme Gradient Boosting (XGBoost), and Adaptive Boosting (AdaBoost).

2.3.1. Regression Trees

Decision trees can be classified based on the type of output variable as classification (categorical output) and regression (numerical output) trees. Within this study, the regression tree was grown on the training data using recursive binary splitting. A small regression tree with a depth of 3 is shown in Figure 1 for ease of illustration. There, the tree for VTEC nowcasting was grown using time information, solar and geomagnetic indices as input, and VTEC as output.

Each regression tree model can be formally expressed as

$$T(\mathbf{X};\Theta) = \sum_{j=1}^{J} \gamma_j (\mathbf{X} \in R_j) \tag{4}$$

with a set of parameters $\Theta = \{\gamma_j, R_j\}_{j=1}^{J}$. $\{R_j\}_1^J$ are disjoint regions that collectively cover the space of all joint values of the input variables $\mathbf{X}$ from Equation (4). The regions represent nodes in Figure 1. The parameters of a single tree are the coefficients $\{\gamma_j\}_1^J$ and the quantities that define the boundaries of the regions $\{R_j\}_1^J$ such as the splitting variables $\mathbf{x}_l$ and the values of those variables (split points) s that splits the nodes of the tree. Since the regions are disjoint, Equation (4) is equal to

$$T(\mathbf{X}) = \gamma_j. \tag{5}$$

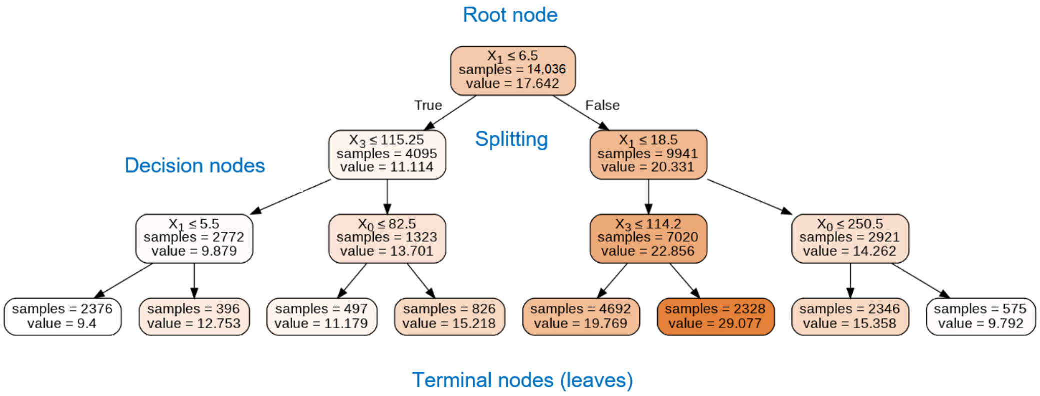


Figure 1. A small decision tree (maximum depth is 3) for VTEC nowcasting (10°10°) based on input data of temporal information, solar and magnetic activity, and output of VTEC. Inputs $\widetilde{\mathbf{x}}_p$ are denoted by the indices 0, 1 and 3, which correspond to DOY, HOD and F10.7 index, respectively. Since the tree is small, it takes into account only inputs that have the highest impact on VTEC. Growing a larger tree will result in more nodes that can consider other inputs as well.

The approach begins at the top of the tree, called the root node, as presented in Figure 1. At this point all observations belong to a single region R. The root node in Figure 1 contains 14,036 observation samples. The mean VTEC value γ of all observations within the region R is 17.642 TECU. The decision splitting in the rote node is given as $\widetilde{\mathbf{x}}_1 \leq 6.5$, which represents the split point, while input variable $\widetilde{\mathbf{x}}_1$, representing HOD, is splitting variable of the region R. The input space is then divided into two distinct and non-overlapping regions R_1 (where the condition is True, i.e., $\widetilde{\mathbf{x}}_1 \leq 6.5$) and R_2 (where condition is False, i.e., $\widetilde{\mathbf{x}}_1 > 6.5$). Therefore, considering a splitting variable $\widetilde{\mathbf{x}}_p$ with $p = 1, 2, \ldots, P-1$ and split point s, two splitting regions can be defined, based on a decision splitting, as [35]

$$R_1(p,s) = \{\mathbf{X} \mid \widetilde{\mathbf{x}}_p \leq s\},\ R_2(p,s) = \{\mathbf{X} \mid \widetilde{\mathbf{x}}_p > s\}. \tag{6}$$

The splitting variable $(\widetilde{\mathbf{x}}_p)$ and split point s are found in a way to solve [35]

$$\min_{l,s}[\min_{\gamma_1} \sum_{\mathbf{x}_{p,i} \in R_1(p,s)} (y_i - \gamma_i)^2 + \min_{\gamma_2} \sum_{\mathbf{x}_{p,i} \in R_2(p,s)} (y_i - \gamma_i)^2] \tag{7}$$

for any choice $\mathbf{x}_l$ and s, the inner minimization is solved by

$$\gamma_1 = \frac{1}{N} \sum_{\mathbf{x}_i \in R_1(p,s)} y_i,\ \gamma_2 = \frac{1}{N} \sum_{\mathbf{x}_i \in R_2(p,s)} y_i. \tag{8}$$

The procedure continues further down on the tree, so that the input space, which covers all joint values of the predictor variable $\mathbf{X}$, is divided into J distinct and non-overlapping regions R_1,R_2,…,R_J. This means that the space of the input variables is successively split, i.e., a node is divided into two sub-nodes or regions further down on the tree. A sub-node that is divided into further sub-nodes is called a decision node. The values in each of rectangle (Figure 1) represent the mean VTEC output γ_j of the y_i falling into region R_j as in Equation (8). A tree stops growing when a node has fewer than a minimum number of observations needed for splitting. This node represents the terminal node or leaf. As can be seen, a decision tree is a simple and highly interpretable method, easily visualized by a two-dimensional graphic, representing an example of a white-box model.

2.3.2. Ensemble Learning

The goal of ensemble learning is to combine predictions of several simple models or base learners, such as an J-node regression tree, to improve generalizability and robustness over a single model. Popular ensemble methods include bagging and boosting.

Random Forest [40] represents a modification of the so-called bagging or bootstrap aggregation technique, which builds a large collection of de-correlated trees and then averages them (Figure 2). When building each tree, a random sample of v input variables is considered as split candidates from a full set of p inputs. Since the forecasting of time series is performed, each new training set is drawn without replacement from the original training set. Thus, a single regression tree T_b (for $b = 1, 2, \dots, B$) is grown by recursively repeating the following steps for each tree node until the minimum node size is reached:

1. Select random sample of v input variables from the full set of p variables;
2. Find the best splitting variable and split point among the v input variables;
3. Split the node into two sub-nodes.

This procedure is applied to all B trees. The function can be expressed as an average of all B trees

$$\hat{F}(\mathbf{x}_i) = \frac{1}{B}\sum_{b=1}^{B} T(\mathbf{x}_i; \Theta_b), \tag{9}$$

where Θ_b characterizes the bth tree in terms of splitting variables, cutpoints at each splitting node and terminal node values. Breiman [40] demonstrated that randomness and diversity in trees construction lead to lower generalization error and an overall better model with reduced variance.

In the boosting method, the trees are grown sequentially using the information from previously grown trees with modified version of the training data (Figure 2). Each boosted tree can be expressed as

$$F_m(\mathbf{x}_i) = F_{m-1}(\mathbf{x}_i) + \sum_{y=1}^{J_m} \gamma_{jm}(\mathbf{x}_i \in R_{jm}), \tag{10}$$

while the final model can be represented as a sum of such trees

$$\hat{F}(\mathbf{x}_i) = F_M(\mathbf{x}_i) = \sum_{m=1}^{M} T(\mathbf{x}_i; \Theta_m) = F_{M-1}(\mathbf{x}_i) + \sum_{y=1}^{J_M} \gamma_{jM}(\mathbf{x}_i \in R_{jM}) \tag{11}$$

for the set of regions and constants $\Theta_m = \{R_{jm}, \gamma_{jm}\}_1^{J_m}$. $F_{m-1}(\mathbf{x}_i)$ represents the previous model, while left side of Equation (11) represents the current tree.

In the AdaBoost [41], the data are modified by applying weights $w_1, w_2, \dots, w_N$ to each of the training examples $(\mathbf{x}_i, y_i)$ (Figure 2). In the first step, all weights are initialized to $w_i = \frac{1}{N}$, i.e., the data are trained in the usual manner. For each successive step $m = 2, 3, \dots, M$, weights are modified individually and the training is repeated using the weighted observations. More specifically, at step m, the weights increase for the wrongly predicted observations in the previous step, while the weights for correctly predicted observations decrease. Therefore, observations that are difficult to predict receive increasing attention as iterations proceed. In the end, weighted predictions from all trees, i.e., steps, are combined to produce the final prediction as in Equation (11).

Gradient boosting offers a generalization of boosting to an arbitrary differentiable objective function in Equation (3). In the first step, a tree is trained on the original training data. Then for $i = 1, 2, \dots, N$ the gradient is computed as [35]

$$-g_{im} = -[\frac{\partial L}{\partial F(\mathbf{x}_i)}]_{F=F_{m-1}}. \tag{12}$$

For the squared error loss, the negative gradient represents the residual between the original and the estimated output $-g_{im} = y_i - F_{m-1}(\mathbf{x}_i)$. For each successive iteration ($m = 2, \dots, M$), a regression tree is fitted to the residuals g_{im} (from the previous iteration) within terminal regions R_{jm} ($j = 1, 2, \dots, J_m$) (Figure 2). Afterwards, the function is updated as in Equation (10). XGBoost [42] is an optimized gradient boosting algorithm that applies shrinkage technique as the regularization strategy to avoid overfitting. This is implemented by scaling the contribution of each tree by a factor $0 \leq \nu < 1$ in Equation (10) as

$$F_m(\mathbf{x}_i) = F_{m-1}(\mathbf{x}_i) + \nu \cdot \sum_{y=1}^{J_m} \gamma_{jm}(\mathbf{x}_i \in R_{jm}), \tag{13}$$

where the parameter ν represents the learning rate of the boosting procedure.

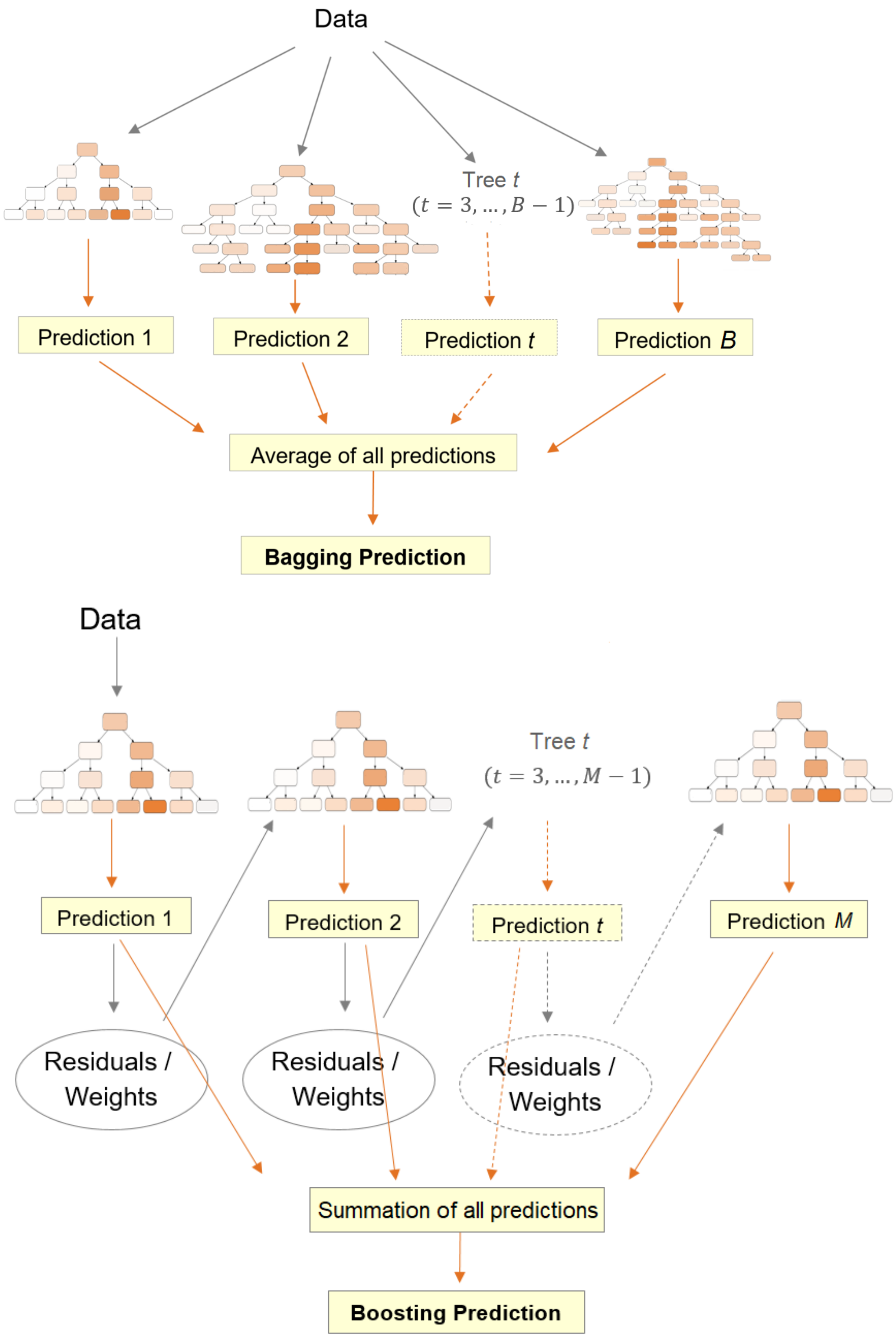


Figure 2. Diagrams of bagging and boosting methods.

Voting Regressor is an ensemble meta-estimator that comprises several machine learning models and averages their individual predictions across all models to form a final prediction. This method is useful for a set of well performing models to compensate for their individual weaknesses in order to build a single model that can better generalize.

It is often useful to provide information about the underlying relationships between the inputs and the outputs of the model to improve understanding of what the model has learned. Using tree-based methods, it is possible to easily estimate the relative importance or contribution of each input variable to forecasted VTEC. Proposed by [43], it is calculated as the improvement in minimization of the objective function as a result of using input variable $\mathbf{x}_l$ to split the node within a tree. The relative importance of a variable $\mathbf{x}_l$ is then calculated as the sum of such improvements overall all internal nodes, for which it was chosen as the splitting variable. For a collection of decision trees $\{T_m\}_1^M$, the relative importance is averaged over all trees.

2.4. Model Selection and Validation

Parameters, such as the splitting variable and the value of the splitting point, are estimated from data during the learning phase using an optimization algorithm, as already discussed. However, every learning algorithm has certain parameters, known as hyperparameters, that cannot be estimated from data, but need to be tuned for a given modeling problem. Hyperparameters determine the model architecture and control the model complexity. Their optimal values depend on the data and the problem. They are typically found by trying different combinations and evaluating the performance of each model. However, the residual sum of squares on the training data cannot be used to determine their values, since that would reduce the ability of a model to generalize future data. Therefore, we used three sets of data, namely the training set (to train the model), validation (to measure the model performance and optimize its parameters/hyperparameters), and test set (used only at the end to estimate the generalization error). In this way, we selected the most optimal (hyper)parameters and provided a measure of the overall reliability and accuracy of the proposed machine learning models.

2.4.1. Time Series Cross-Validation

Since observations are temporally dependent, we applied the time series cross-validation technique to preserve a temporal dependency, where a model is trained, optimized, and evaluated on a rolling basis using many data folds. For reliable performance evaluation, a large number of folds should be adopted [44]. The data are divided into two folds at each iteration: a training set and cross-validation set (Figure 3). The model is trained on the training set, while the (hyper)parameters, that minimize the RMSE, are found using the cross-validation set. The training set consists only of observations that occurred prior to observations that form the cross-validation set. The cross-validation data from the previous iteration are included as part of the next training data set and subsequent data points are forecasted. The final metric is calculated as the average of the RMSE obtained in each cross-validation iteration.

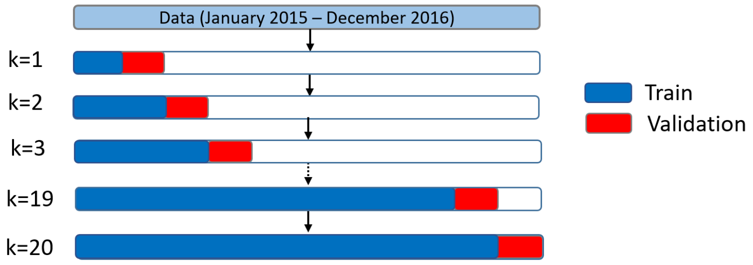


Figure 3. Evaluation of model performance using time series cross-validation with 20 folds to prevent overfitting and evaluate model performance in a robust way. The final metric is calculated as the average RMSE of every cross-validation iteration.

2.4.2. Model Architecture

Hyperparameters of tree-based models, that commonly need to be optimized, are the maximum depth of the tree (max_depth), number of trees in ensemble learning (n_estimators), the value of learning rate (in boosting), etc. The size of a tree controls the complexity of the model. Too large tree results in a more complex model that can overfit the training data, and consequently may not generalize well. The size $v = 6$ (max_features) of the random subsets of input variables is considered when splitting a node to introduce randomness in tree construction in order to improve the accuracy and reduce the overfitting problem. The lower size of v reduces the correlation between any pairs of trees in Random Forest and hence, reduces overfitting. However, if there are only a few relevant variables out of many, v should be set to a higher value, so that the algorithm can find the relevant variables. For XGBoost, smaller values of learning rate ν (more shrinkage) result in a lower test error but require a larger number of iterations m [45]. Moreover, data are subsampled for every tree to further prevent overfitting. Optimal hyperparameters and the range of values used to search optimal values for hyperparameters are provided in Table 2, where min_samples_split and min_samples_leaf are the minimum number of samples required in an internal node and leaf node, respectively. Similar values for hyperparameters were found for the global XGBoost VTEC model in [29], namely 100 trees, a maximum tree depth of 6, and a learning rate of 0.1.

Table 2. Hyperparameters of developed machine learning models.

Model	Selected Hyperparameters	Range of Search

Decision Tree	max_depth = 5–8
	min_samples_split = 10–20
	min_samples_leaf = 10
		[4, 5, 6, 7, 8, 9, 10, 15, 20]
		[2, 5, 10, 15, 20]
		[2, 5, 10, 15, 20]

Random Forest	max_features = 6
	max_depth = 8–10
	min_samples_split = 10
	min_samples_leaf = 5
	n_estimators = 300
		[4, 5, 6, 7, 8]
		[4, 6, 8, 10, 12, 15, 20]
		[2, 5, 10, 15, 20]
		[2, 5, 10, 15, 20]
		[50–500] interval of 50

AdaBoost	max_depth = 6–8
	n_estimators = 50
		[3, 4, 5, 6, 7, 8, 9, 10, 15]
		[50, 100, 150, 200, 300]

XGBoost	max_depth = 4–6
	n_estimators = 100
	learning_rate = 0.1
	subsample = 0.5
		[3, 4, 5, 6, 7, 8, 9, 10, 15]
		[50, 100, 150, 200, 300]
		[0.01, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3]
		[0.3, 0.5, 0.7, 1]

Individual models were trained for each grid point and each forecast window, i.e., six models were developed for each machine learning method in Table 3. A total of 72 models were developed: 36 for non-differenced data and 36 for differenced data. Learning algorithms were implemented in the Python programming language using Scikit-learn library [46].

Table 3. Overview of developed VTEC machine learning models.

Abbreviation	Machine Learning Model	Approach
DT	Decision (Regression) Tree	Single tree
RF	Random Forest	Bagging ensemble
AB	AdaBoost	Boosting ensemble
XGB	XGBoost	Boosting ensemble
VR1	Random Forest, AdaBoost & XGBoost	Meta-ensemble
VR2	Random Forest & XGBoost	Meta-ensemble

Figure 4 depicts a flowchart of the VTEC machine learning model development. Based on its performance on training and validation data, the model was optimized in terms of hyperparameters and data. In the case of high bias, the model with an approximation function is not complex enough, and therefore, it underfits the data. As the model complexity increases, the variance tends to increase, while the bias tends to decrease, which results in a decrease in training error (Figure 5). However, too much complexity leads to an increase in the validation error and consequently to a large test error due to overfitting (high variance). The aim is to find a balanced model that neither learns from the noise (data overfitting) nor makes poor assumptions about the data (data underfitting). The final model complexity is chosen in a way to trade off bias with variance, i.e., balance bias with variance to minimize the validation error and, consequently, the test (generalization) error. High bias was fixed by adding new input features and increasing the values for max_depth, max_features, n_estimators. The high variance was addressed by decreasing the values for max_depth, max_features, n_estimators, learning_rate, subsample, as well as increasing the values for min_samples_split and min_samples_leaf (Table 2).

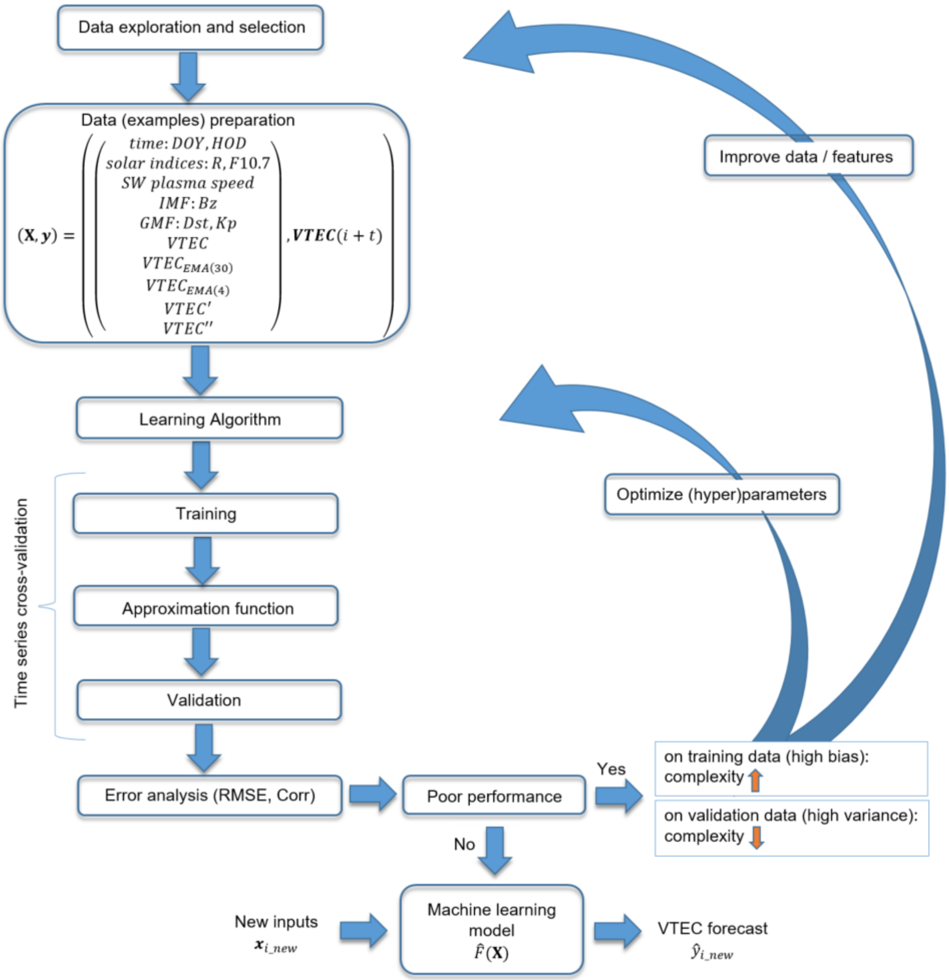

Data exploration and selection
Data (examples) preparation
$(\mathbf{X},\mathbf{y}) = \left(\begin{pmatrix} time: DOY, HOD \\ solar\ indices: R, F10.7 \\ SW\ plasma\ speed \\ IMF: Bz \\ GMF: Dst, Kp \\ VTEC \\ VTEC_{EMA(30)} \\ VTEC_{EMA(4)} \\ VTEC' \\ VTEC'' \end{pmatrix}, \boldsymbol{VTEC}(i+t)\right)$
Learning Algorithm
Training
Approximation function
Validation
Time series cross-validation
Error analysis (RMSE, Corr)
Poor performance
Yes
No
on training data (high bias): complexity ↑
on validation data (high variance): complexity ↓
Improve data / features
Optimize (hyper)parameters
New inputs
$\boldsymbol{x}_{i\_new}$
Machine learning model
$\hat{F}(\mathbf{X})$
VTEC forecast
$\hat{y}_{i\_new}$

Figure 4. Flowchart of VTEC machine learning model development from data exploration, selection, and preparation to training and cross-validation until the final machine learning model with the target approximation function is not found. The model is optimized in terms of its performance. Poor performance on training and validation data is the result of high bias, while poor performance on validation data is the result of high variance. They can be solved by increasing or decreasing the model complexity, respectively. The final machine learning model can be used to forecast VTEC on new input data.

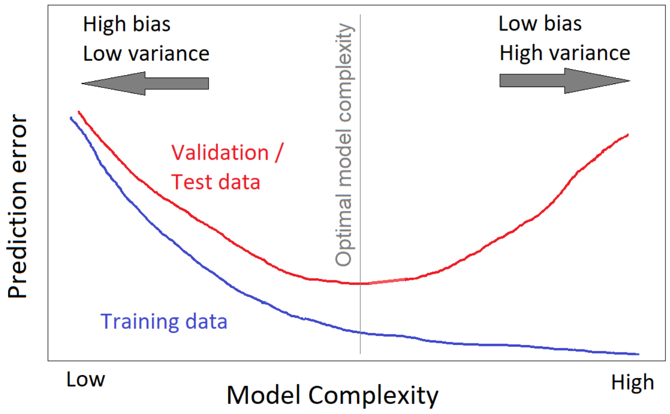


Figure 5. Training and validation/test errors as a function of model complexity. Optimal model complexity yields a balanced model that has neither large bias (data underfitting) nor large variance (data overfitting).

The training and validation time with 20-folds for the single Decision Tree model is under 5 s, while it increases for the ensemble learning models from 30 s for XGBoost to about 5 min for Random Forest (Table 4). Overall, their execution time is under 5 min, while the testing time is less than one second for each of the models, demonstrating the computational efficiency of the proposed models.

Table 4. Execution time in seconds of the VTEC models using NVIDIA Tesla P100 GPU with 16 GB memory.

Machine Learning Model	Training and Validation (s)	Testing (s)
DT	2–4	<0.01
RF	300–330	∼0.30
AB	65–85	∼0.10
XGB	30–40	∼0.05
VR1	∼250	∼0.35
VR2	∼200	∼0.25

3. Results

3.1. Exploratory Data Analysis

Exploratory data analysis is performed to identify significant patterns and correlated data, as well as to summarize their properties to support the selection of input data for the machine learning model. It is important to prepare a suitable dataset for learning algorithms to enable the learning of important features for VTEC forecasting. Our goal was to create training data with enough relevant and not too many irrelevant inputs and also not too much correlated input data. These properties can be verified with the correlation matrix between the input and the output data (Figure 6).

Using the non-differenced data (Figure 6, top left), a weak positive and negative linear relationship between VTEC and the time information, hour and DOY, respectively, can be noticed. A weak to a moderate positive relationship can be seen between VTEC and solar indices (R and F10.7). The relationship to the solar wind and magnetic activity data (Bz, Dst, AE) indicates a very weak to no relationship at all. On the other hand, the relationship between differenced VTEC and the time information disappears and there is a very weak relationship with solar indices (Figure 6, bottom left). However, the relationship between differenced VTEC and differenced data of solar wind and magnetic activity increased. The relationship between VTEC(t) with VTEC(t + 1 h) and VTEC(t + 24 h) is very high positive for non-differenced data, while for differenced data is high positive for VTEC(t + 1 h) and low positive for VTEC(t + 24 h). The heatmap for periods of strong and severe geomagnetic storms (Kp ≥ 7) reveals a weak to moderate relationship between VTEC and all input data (Figure 6, right). The relationship to data of solar wind speed, Bz, Kp, and AE is significantly increased. These relationships are not visible in the heatmap over the entire training period (Figure 6, left) as these events are rare and unrepresented in the dataset (Figure 7). However, during the space weather event, it becomes apparent that these data are relevant and should be taken into account. The VTEC prediction during space weather events is clearly a case of unrepresented classes, i.e., data imbalance, where space weather events are in minority compared to the quiet period.

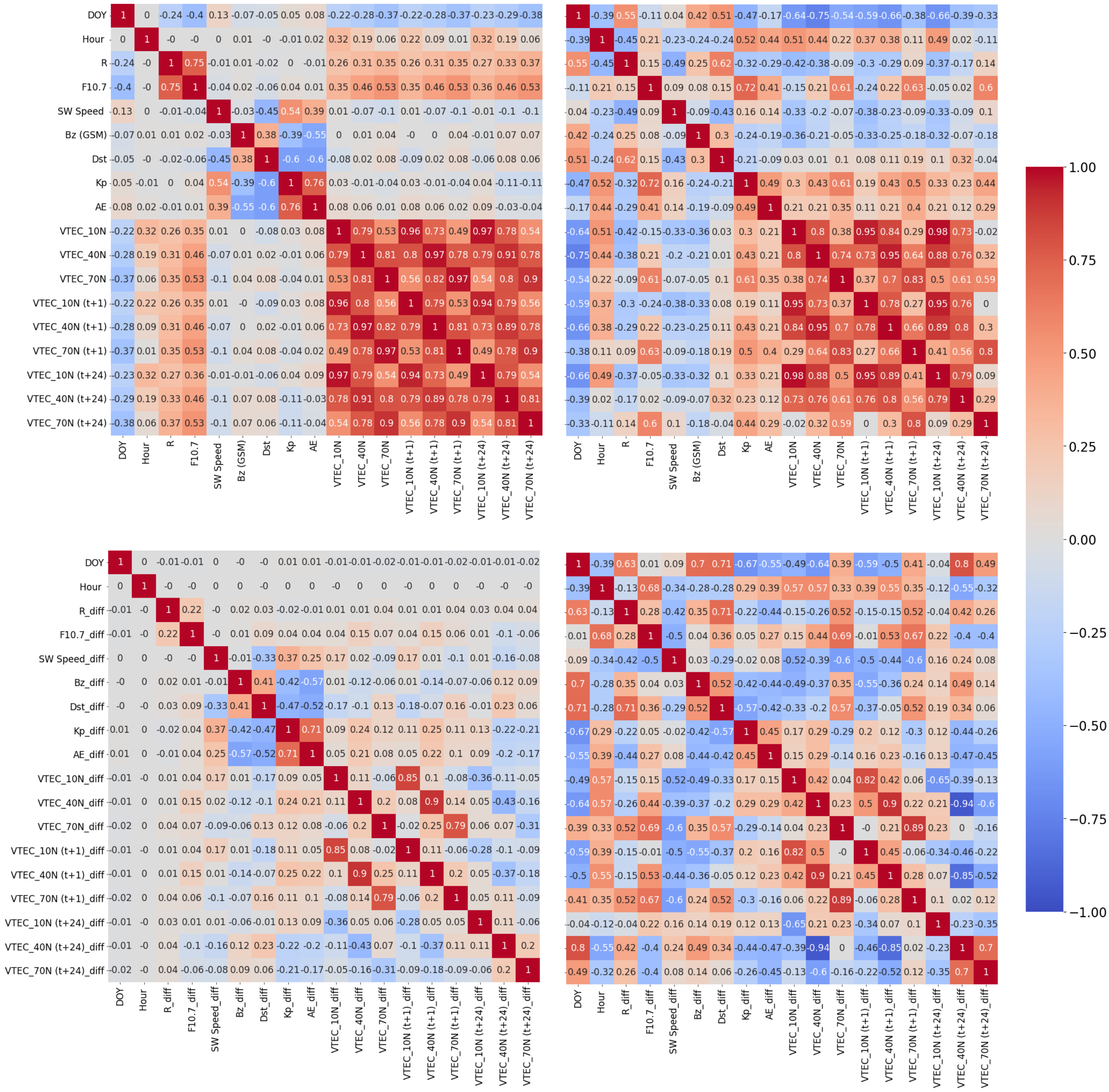


Figure 6. Correlation matrix between the model input and ground-truth VTEC. Top: non-differenced data, bottom: differenced data. left: training data (2015–2016), right: training data for Kp ≥ 7.

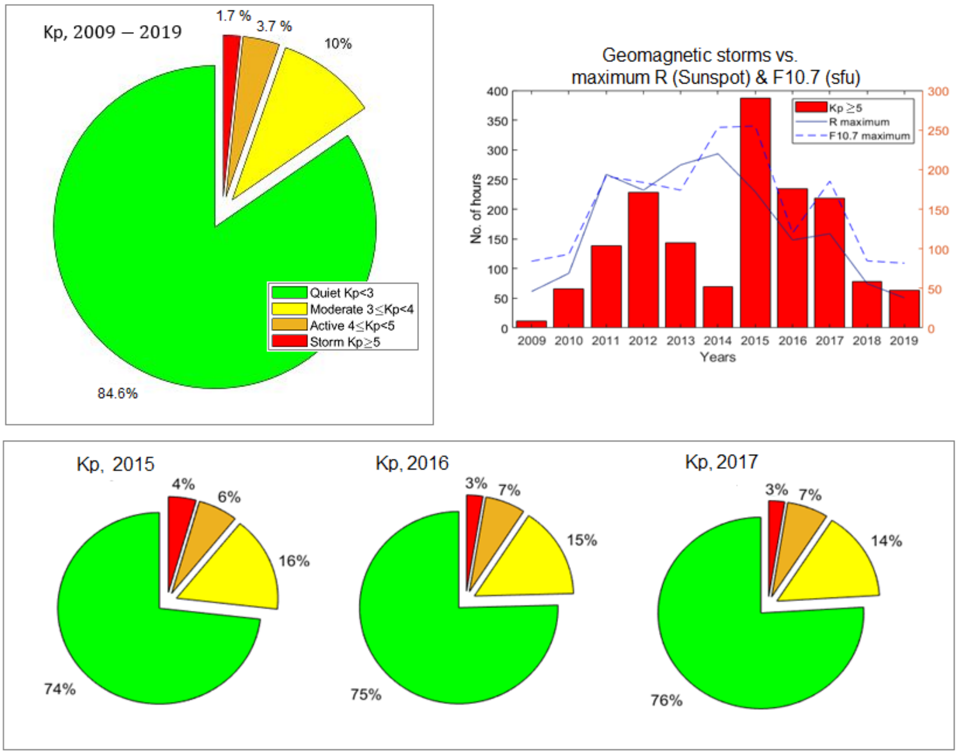


Figure 7. Percentage of Kp data with the values Kp < 3, $3 \leq$ Kp < 4, $4 \leq$ Kp < 5, and Kp ≥ 5 denoting quiet, moderate, active and storm conditions, respectively, in the geomagnetic field, over solar cycle 24 (2009–2019) (top left) and for years 2015, 2016 and 2017 (bottom). Top right: Number of hours of the Kp data with values Kp≥ 5 vs. the maximum values of the sunspot number R and the solar radio flux F10.7 (both referring to the right y-axis) from 2009 to 2019.

As highlighted in Figure 7 (top left), 85% of the 3 h Kp data from 2009 to 2019 indicates quiet conditions in the geomagnetic field, while only 2% of the Kp data have reached an index of 5 or higher. This indicates that geomagnetic storms are strongly underrepresented and occur rather rarely, leading to imbalanced examples. On the other hand, these examples are of special interest as they contain useful knowledge and important information for forecasting purposes. Machine learning boosting algorithms have been shown to be suitable for applications with imbalanced data [47]. The number of geomagnetic storm conditions (Kp $\geq$ 5) was the highest in the years after the solar maximum (reached in April 2014), i.e., from 2015 to 2017, and in 2012, before the solar maximum (Figure 7, top right). Years 2015 and 2016 have more of these events than other years (Figure 7, bottom). In addition, they are near the solar maximum. Therefore, they have been chosen for training and cross-validating the models to have more examples of storm events and near-solar maximum conditions. The subsequent year 2017 is selected for testing as it includes the strongest storm of solar cycle 24 (in September 2017). In the training dataset, there are 99 days with reported Kp $\geq$ 5, with 56 days in 2015 and 43 days in 2016. In the test dataset, Kp $\geq$ 5 applies for 37 days.

3.2. K-Fold Selection for Cross-Validation

To achieve optimal results, the appropriate k-fold size was analyzed with respect to the accuracy of two machine learning models, namely Decision Tree and Random Forest. For the analysis VTEC is predicted for high-latitude, mid-latitude and low-latitude ionospheric regions using the varying k-fold sizes: $k = (6, 10, 20, 30, 40, 50)$ (Figure 8). The graph to the left presents the RMSE for cross-validation and test datasets with Decision Tree and Random Forest for 6 and 20 folds. The low-latitude VTEC forecast (RMSE) is improved for about 1 TECU by increasing the k-fold size from 6 to 20, while the RMSE for the high and mid-latitude VTEC is similar for both k-folds. Only for Decision Tree, a slight RMSE degradation for the high-latitude VTEC is observed for test data with 20 folds, which may be due to overfitting as a single tree tends to overfit the data. However, the improvement in the low-latitude VTEC is significantly higher. The graph to the right illustrates the RMSE as a function of the number of k-folds for the low-latitude VTEC, where $k = 20$ folds appear to be optimal. The low-latitude VTEC has more complex variations, such as those due to the equatorial anomaly. Using a smaller k-fold size, the model is trained and cross-validated with larger data samples in a single k-fold with a smaller number of iterations. In that case, it may overlook some VTEC variations that are not much represented in the single split. By increasing the k-folds size, the model is trained on smaller data subsets with more iterations. Thus, the signal can be learned better. However, too large k-fold will result in very small data subsets that can lead to overfitting during training, resulting in higher RMSE on the cross-validation set, as for $k = 50$. Similar behavior is observed for the boosting models. However, the goal is to neither overfit nor underfit the data. From Figure 8 it is apparent that $k = 20$ folds are optimal for a 2-year cross-validation period (2015–2016).

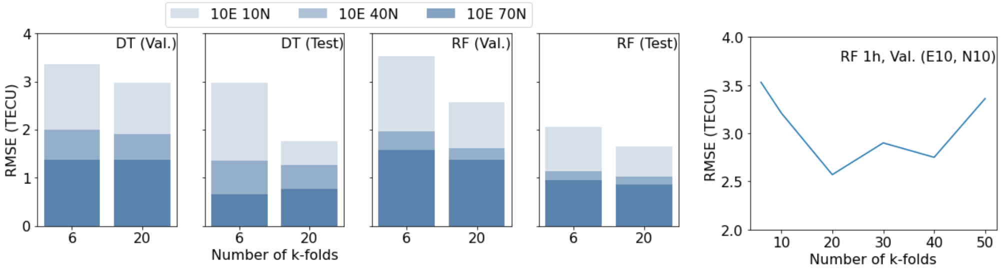


Figure 8. Number of time series k-folds splits as function of RMSE. **Left**: RMSE for Decision Tree (DT) and Random Forest (RF) on cross-validation and test datasets for *k* = 6 and 20. **Right**: RMSE for the RF model on coss-validation data for *k* = 6, 10, 20, 30, 40, 50.

RMSE on training and cross-validation data sets is presented for each k-fold for the model VR1 in Figure 9. Training curves are mostly constant after the 5th fold, while cross-validation curves are changing more significantly. The training set is small in the first folds, which results in lower RMSE during training, as it is easier to fit the smaller dataset. On the other hand, the larger RMSE values are for cross-validation, as such a small training data set is not representative. While increasing the number of k-folds, the training data set becomes larger, which slightly increases RMSE during training, while decreasing RMSE during cross-validation. Thus, the largest errors for the cross-validation are mostly in the first 5 folds. After the 10th fold, RMSE values of cross-validation are similar to RMSE values of training or even smaller as k further increases. The average RMSE values for training and cross-validation for all k-folds are summarized in the bar graphs in Figure 9. Differences in RMSE between training and cross-validation are larger for non-differenced data than for differenced data. Differenced data have a lower training RMSE, while the cross-validation RMSE is mostly similar between differenced and non-differenced data.

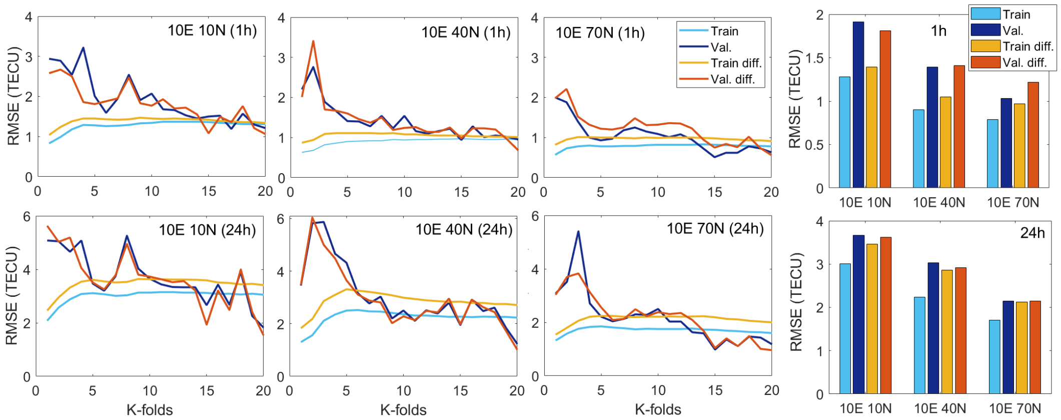


Figure 9. RMSE on training and validation data from 1st to 20th k-fold (**left**). The bar graph represents the average RMSE for all 20 k-folds (**right**). Top: 1-h, bottom: 24 h forecast. Results are provided for the model VR1 for both non-differenced and differenced (diff.) data.

3.3. Relative Importance of Input Variables to VTEC Forecast

Relative importance of the input variables for the 1 h and 24 h VTEC forecasts using non-differenced and differenced data for the period 2015–2016 is estimated (Figure 10), including an analysis for geomagnetic activity conditions (Kp $\geq$ 5) (Figure 11).

The results demonstrate that the previous VTEC information as an input variable has the largest contribution to the VTEC prediction. Its significance for non-differenced data is about 60%, while for differenced data it is from 60% to 70% for 1 h and 30% to 50% for 24 h forecasts (Figure 10). Another important input variables for non-differenced data are exponential moving averages, especially over the last four days (96 h). On the other hand, time-derivatives of VTEC are more important for 1 h forecast with differenced data, in particular the first derivative. For 24 h forecast with differenced data, exponential moving averages have higher importance. For the high-latitude ionospheric region (10°70°), the second derivative has also a higher contribution. For models with non-differenced data, other dominant input variables are temporal information (hour and DOY), followed by solar activity data (F10.7 index), while other variables (solar wind and magnetic field) have little or no influence on the VTEC forecast. In the case of differenced data, on the other hand, the contribution of the temporal information decreased. At the same time, the contribution of other input variables increased, namely solar wind speed (SW) and indices of magnetic field (AE, Kp, Dst, Bz). Their significance is larger for 24 h forecasts.

During geomagnetic storm conditions, the relative importance of input variables, describing the solar activity, solar wind, and magnetic activity, increased for almost all ionospheric regions, especially for non-differenced data, while the contribution of previous VTEC value mostly decreased (Figure 11). In addition, the contribution of the first time derivative for non-differenced data increased. It is especially interesting to see the higher significance of the AE index for high-latitude, the Kp index for mid-latitude, and the Dst index for low-latitude ionospheric regions for differenced data, having in mind that these indices are measured in those regions.

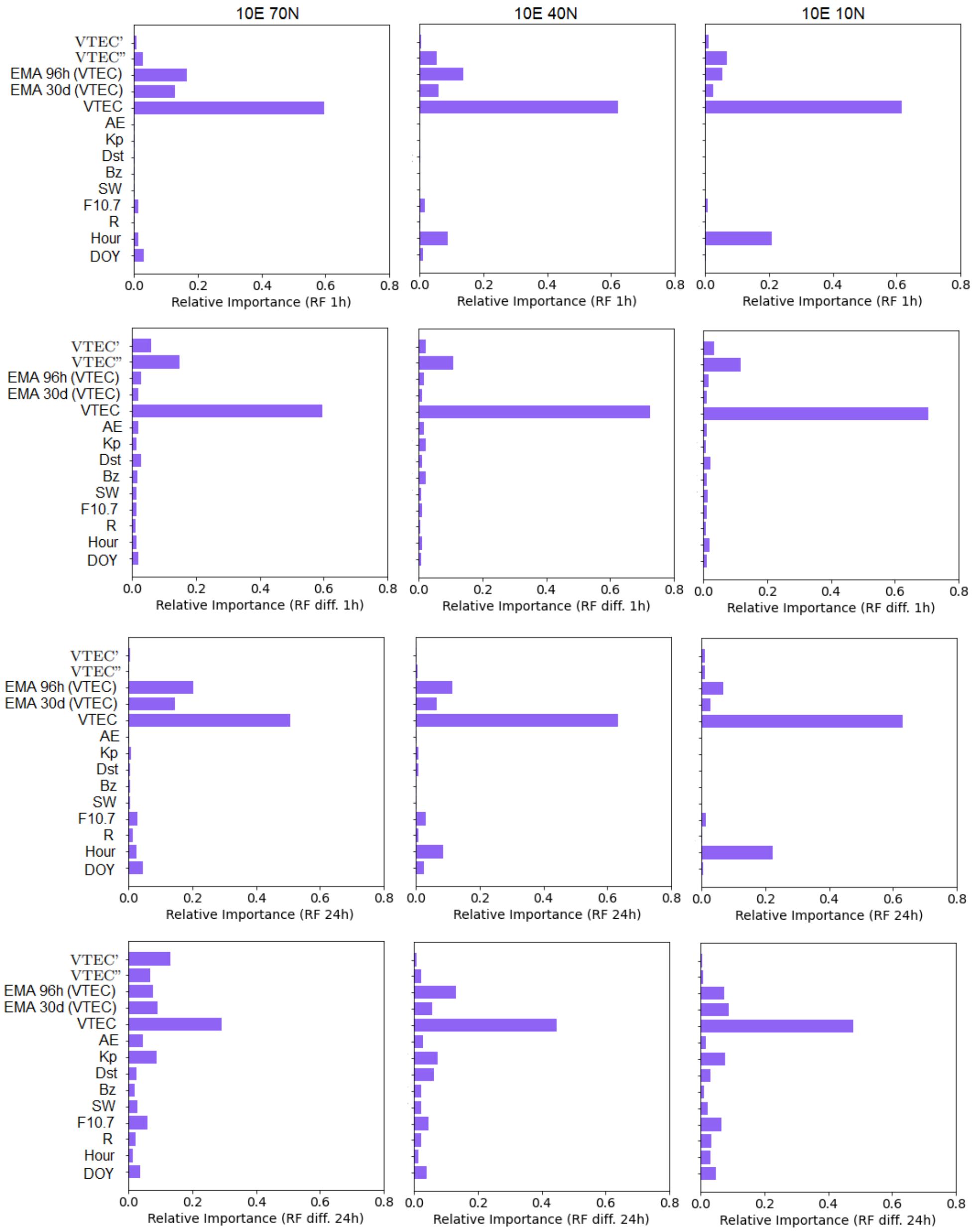


Figure 10. Relative importance of input variables to VTEC forecast estimated from the Random Forest models. Results are presented for 1 h forecast with non-differenced data (**first row**) and differenced data (**second row**), and for 24 h forecast with non-differenced data (**third row**) and differenced data (**fourth row**) for high-latitude (**left**), mid-latitude (**middle**) and low-latitude (**right**) VTEC.

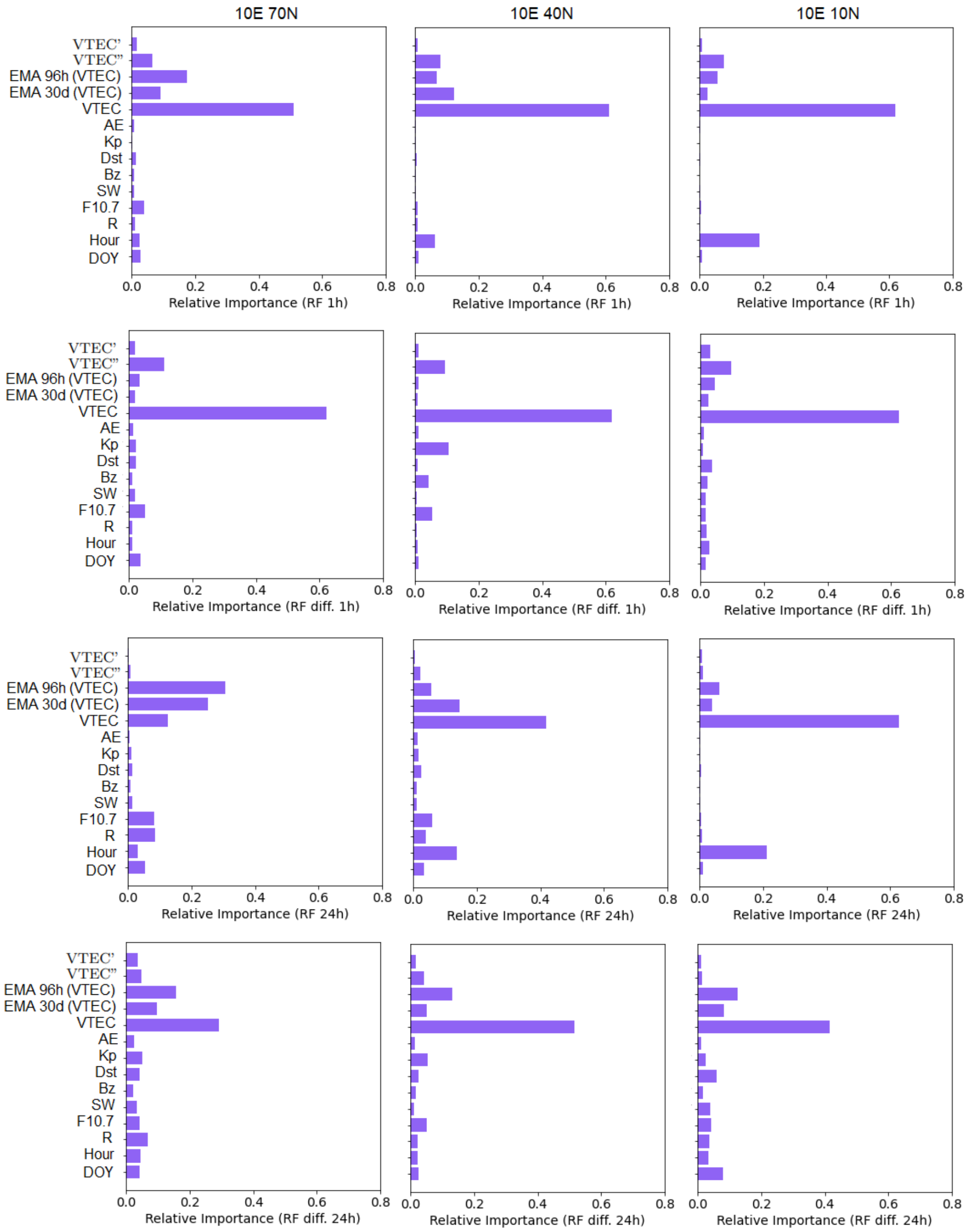


Figure 11. Relative importance of input variables to VTEC forecast for geomagnetic storm conditions (Kp ≥ 5) estimated from the Random Forest models. Results are presented for 1 h forecast with non-differenced data (**first row**) and differenced data (**second row**), and for 24 h forecast with non-differenced data (**third row**) and differenced data (**fourth row**) for high-latitude (**left**), mid-latitude (**middle**) and low-latitude (**right**) VTEC.

3.4. Accuracy Performance of Machine Learning Models

The RMSE and correlation coefficients for cross-validation, test, and geomagnetic storm (7–10 September 2017) datasets for the 1 h and 24 h forecasts with different machine learning models, namely Decision Tree and ensemble learning (Random Forest, AdaBoost, XGBoost and Voting Regressors), using two types of data (non-differenced and differenced) are presented in Figure 12. The period of the severe geomagnetic storm (7–10 September 2017) covers the main and recovery phase of the storm. In addition, an overview of the RMSE for the year 2017 and for the storm in September 2017 is shown in Table 5.

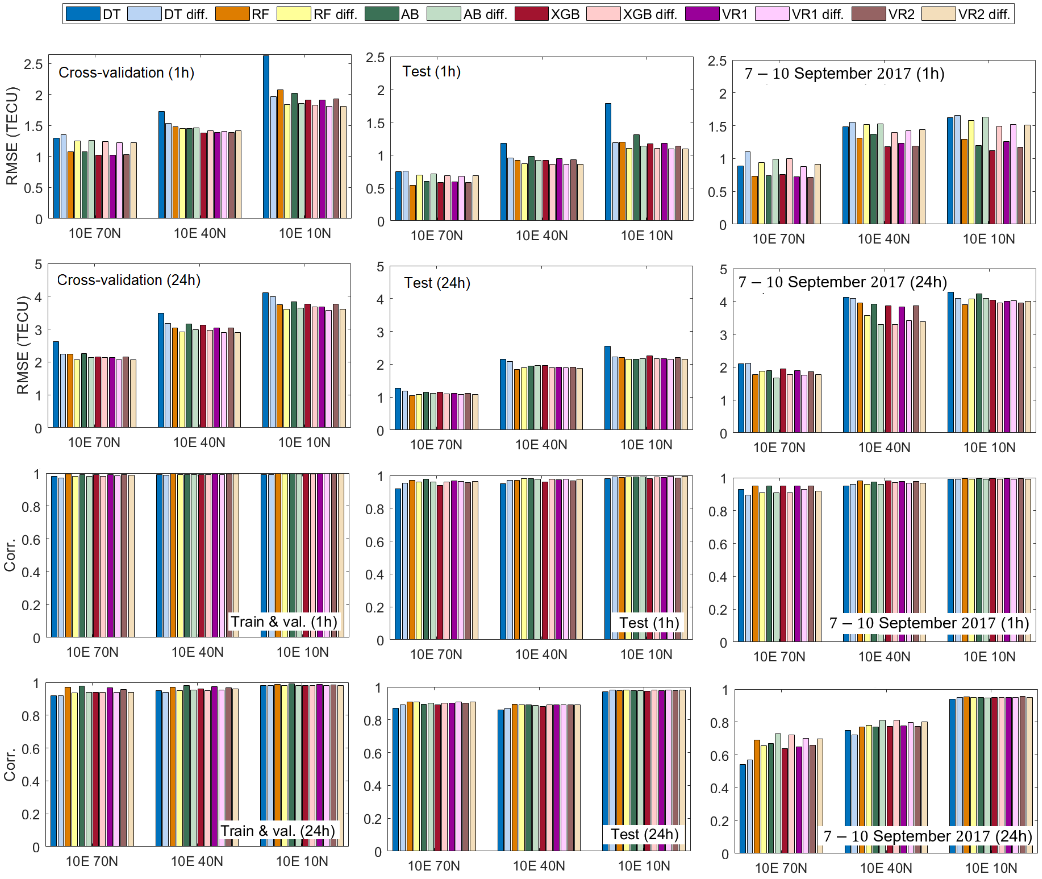


Figure 12. The RMSE and correlation coefficient (Corr.) for cross-validation (**first column**), test (**second column**) and geomagnetic storm (7–10 September 2017) (**third column**) datasets, for 1 h and 24 h forecast horizons for Decision tree (DT), Random forest (RF), Adaboost (AB), XGBoost (XGB) and two Voting regressor models (VR1 and VR2). Models trained of differenced data are marked with "diff" The evaluation on the test dataset is a measure of the models' generalization to new examples.

The RMSE for the cross-validation dataset is higher than for the test dataset because the training period (2015–2016) includes larger absolute VTEC values as it is closer to the solar maximum (in April 2014). The RMSE is about twice higher for 1 day forecast than for 1 h forecast for all the models. The VTEC forecast with a single Decision Tree tends to have the highest RMSE and the lowest correlation coefficients for all datasets. Ensemble learning improved the accuracy. The lowest RMSE is mostly achieved with the Random Forest, XGBoost and Voting Regressor models. During the severe geomagnetic storm in September 2017, the RMSE is about 0.3 to 0.5 times higher for the 1 h forecast and 0.7 to 1 times higher for the 24 h forecast than for the entire test year 2017 (Table 5). The boosting method provided the lowest RMSE during the space weather event (XGBoost in particular), demonstrating its usefulness for predicting rare events. From the cross-validation and test results (Figure 12) it can be seen that the lower RMSE for the mid- and low- latitudinal ionospheric regions for 1 h and 24 h forecasts mainly have models trained on differenced data. This can be also observed for the high-latitude ionosphere for the 24 h forecast. The correlation coefficients are above 90% for the 1 h and 24 h forecast horizons for the training and cross-validation dataset (2015–2016), while the correlation coefficients for the test dataset (2017) are above 90% for the 1 h and above 85% for the 24 h forecast horizons. For the year 2017, the lowest RMSE for the high-latitude VTEC is 0.54 TECU and 1.6 TECU for the 1 h forecast (non-differenced data) and the 24 h forecast (differenced data), respectively (Table 5). The lowest RMSE for VTEC in the mid-latitude is 0.86 TECU and 1.86 TECU for the 1 h (differenced data), and the 24 h (non-differenced data) forecast horizons, respectively. The highest accuracy of the VTEC forecast for the low-latitude region is the RMSE of 1.09 TECU and 2.15 TECU for the 1 h and 24 h forecast horizons, respectively, both with differenced data.

On the other hand, during the storm, models trained on non-differenced data have a smaller RMSE and a higher correlation coefficient for the 1 h forecast, while for the 24 h forecast the smallest RMSE and the highest correlation coefficients are mostly for models with differenced data. Correlations between the ground-truth VTEC and predicted VTEC for the storm test period are from 90% for the 1 h forecast, while for the 24 h forecast they are from 55% (the highest 73%), 72% (the highest 81%) and 94% (the highest 95%) for the high-latitude, mid-latitude, and low-latitude ionospheric regions, respectively. For the 1 h forecast, the highest correlation coefficients are achieved with Voting Regressor and non-differenced data, while for the 24 h forecast the highest correlation coefficients are with boosting methods (AdaBoost and XGBoost) and differenced data. The lowest RMSE for the high-latitude VTEC is 0.71 TECU and 1.67 TECU for the 1 h (non-differenced data) and 24 h (differenced data) forecast, respectively (Table 5). In terms of the mid-latitude VTEC, the lowest RMSE is 1.18 TECU and 3.29 TECU for the 1 h (non-differenced data) and 24 h (differenced data) forecast, respectively. The low-latitude VTEC forecast achieved the lowest RMSE of 1.12 TECU and 3.96 TECU for the 1 h (non-differenced data) and 24 h (non-differenced and differenced data) forecast, respectively.

Table 5. Overview of RMSE for different machine learning models for test period of year 2017 and severe geomagnetic storm 7–10 September, 2017 for high-latitude (10E 70N), mid-latitude (10E 40N) and low-latitude (10E 10N) VTEC. The subscript diff. indicates models trained on differenced data. The maximum and minimum values of the RMSE are marked in red and green, respectively.

	DT	**RF**	**AB**	**XGB**	**VR1**	**VR2**
	70N, 40N, 10N	70N, 40N, 10N	70N, 40N, 10N	70N, 40N, 10N	70N, 40N, 10N	70N, 40N, 10N
2017	RMSE (TECU)	RMSE (TECU)	RMSE (TECU)	RMSE (TECU)	RMSE (TECU)	RMSE (TECU)
1 h	0.75, 1.18, 1.79	0.54, 0.92, 1.20	0.60, 0.98, 1.31	0.59, 0.92, 1.17	0.59, 0.95, 1.18	0.58, 0.93, 1.14
1 h$_{\text{diff.}}$	0.76, 0.96, 1.19	0.70, 0.87, 1.10	0.71, 0.92, 1.14	0.69, 0.86, 1.10	0.68, 0.86, 1.09	0.69, 0.86, 1.09
24 h	1.28, 2.15, 2.55	1.06, 1.86, 2.20	1.15, 1.95, 2.26	1.15, 1.96, 2.25	1.11, 1.91, 2.17	1.11, 1.92, 2.21
24 h$_{\text{diff.}}$	1.18, 2.08, 2.22	1.08, 1.89, 2.15	1.12, 1.96, 2.17	1.10, 1.89, 2.17	1.08, 1.89, 2.15	1.08, 1.88, 2.15
7–10 September						
1 h	0.89, 1.48, 1.62	0.73, 1.31, 1.29	0.74, 1.37, 1.20	0.76, 1.18, 1.12	0.72, 1.23, 1.16	0.71, 1.19, 1.17
1 h$_{\text{diff.}}$	1.10, 1.55, 1.66	0.94, 1.52, 1.58	0.99, 1.53, 1.63	1.00, 1.40, 1.49	0.88, 1.42, 1.52	0.91, 1.44, 1.51
24 h	2.10, 4.12, 4.29	1.77, 3.95, 3.95	1.89, 3.92, 4.23	1.95, 3.87, 4.04	1.90, 3.84, 4.01	1.86, 3.87, 3.96
24 h$_{\text{diff.}}$	2.12, 4.09, 4.10	1.87, 3.57, 4.08	1.67, 3.29, 4.09	1.77, 3.29, 3.96	1.76, 3.41, 4.02	1.78, 3.39, 4.00

The relative RMSE change with respect to the persistence (naive) forecast is presented in Figure 13. The persistence model considers that the VTEC(i + t) is equal to the VTEC(i), where t takes values of 1 and 24 for the 1 h and 24 h forecasting, respectively, i.e., we assume the state of the frozen ionosphere with respect to the previous hour or previous day. Persistence forecast is the most common baseline method to measure the forecast performance in supervised machine learning, as well as, in data-driven, physics-based and traditional statistical VTEC forecasting [24,31,48–50]. The models for the 1 h forecast of the low-latitude VTEC have reduced RMSE of about 60% for the test period and about 70% during the geomagnetic storm with respect to the baseline. The relative RMSE reductions with respect to the persistence forecast are up to 20%, near 40%, and around 60% for the high-, mid-, and low- latitude VTEC points, respectively, for the 1 h forecast with a non-differenced RF model. For the 24 h forecast, the machine learning models have a lower RMSE by 10–25%, 15–25%, and 5–10% for the high-, mid-, and low- latitude VTEC points, respectively. For the 1 h forecast of the high-latitude VTEC, the RMSE is increased when using differenced data. However, in the case of the 24 h forecast, the models with differenced data mostly forecast the high-latitude VTEC by 5% to 10% lower RMSE than the models with non-differenced data. In addition, they improve the 1 h and 24 h mid- and low-latitude VTEC forecasts by about 2–5% with respect to the models with non-differenced data in 2017. For the storm period, models with non-differenced data provide 1 h forecast for the VTEC points at high-, mid-, and low-latitudes with an RMSE of about 10–15% lower than the models with differenced data. On the other hand, for the longer (24-h) forecast, the models with differenced data mostly outperform the models with non-differenced data, especially during the storm, when the relative RMSE decrease is up to 10% for the high-latitude VTEC and 10% to 15% for the mid-latitude VTEC. The differences in the low-latitude 24 h VTEC forecast are smaller (<4%) between different models and data.

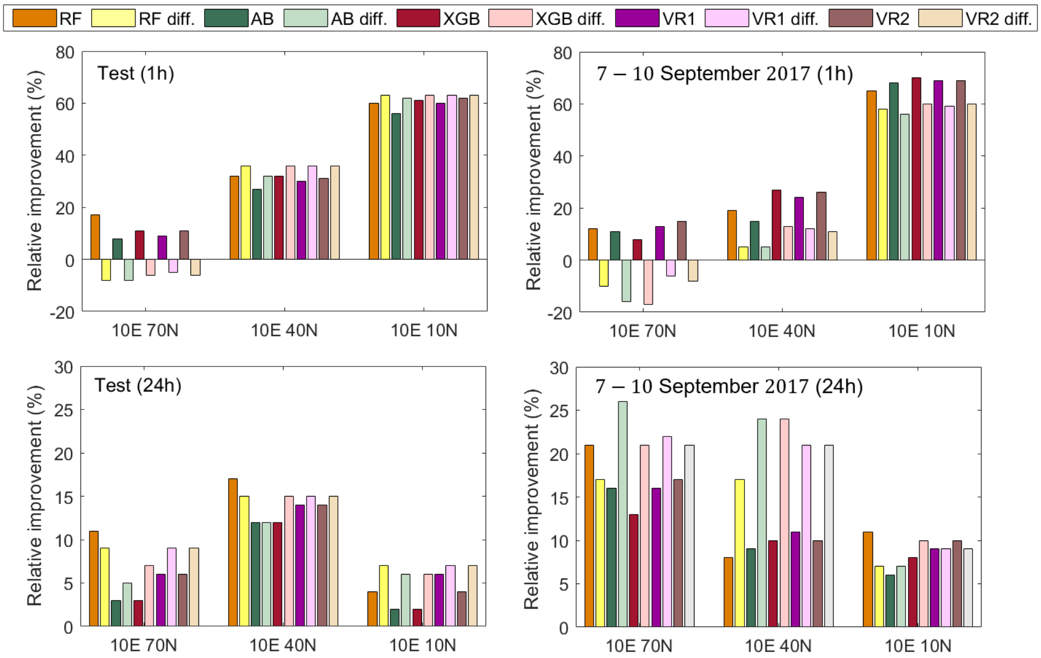


Figure 13. Relative RMSE change with respect to the persistence (naive) forecast. **Top**: 1 h forecast, **bottom**: 24 h forecast. **Left**: test data (2017), right geomagnetic storm (7–10 September 2017). Models trained of differenced data are marked with "diff.".

Machine learning performance depends highly on data and, therefore, data should be prepared in a way to enhance learning. Figure 14 shows the VR1 model improvement by including the inputs such as exponential moving averages and time derivatives of VTEC. First, models were trained with the first ten input data in Table 1, i.e., without exponential moving averages and time derivatives, denoted as Data1. Later those inputs are added to improve learning, referred to as Data2. For both non-differenced and differenced data, the RMSE is reduced (by 0.2 to 0.5 TECU) as additional inputs of moving averages and time derivatives are added.

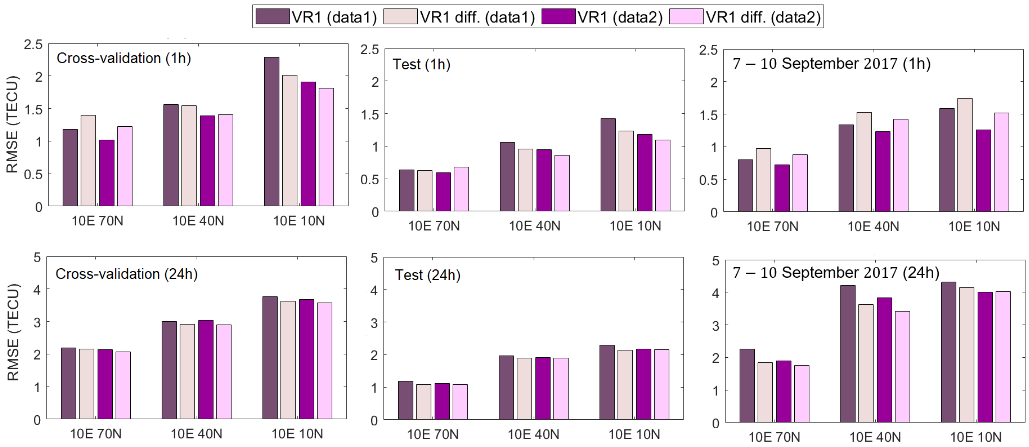


Figure 14. The RMSE for cross-validation (**left**), test (**mid**) and geomagnetic storm (**right**) datasets for 1 h (**top**) and 24 h (**bottom**) forecasts for the VR1 model. Models trained with differenced data are marked with "diff". Data2 refers to data in Table 1, while Data1 comprises Data2 excluding inputs of EMA, VTEC′ and VTEC″.

Further, differenced and undifferenced data were combined, where exponential moving averages and time derivatives were calculated from non-differenced VTEC, to forecast non-differenced VTEC (Figure 15). The analysis was done for the XGBoost model, because it is fast compared to other models (Table 4), while accuracy is comparable between the models (Table 5). The RMSE for the 1 h VTEC forecast for test data including the severe geomagnetic storm is similar for models with non-differenced data and data combination. For the 24 h forecast, the RMSE for mid- and low-latitude VTEC with data combination is lower than for the non-differenced data, while more similar to the RMSE for differenced data (Figure 15). Data combination for forecasting non-differenced VTEC improved the model accuracy compared to the non-differenced data for the 24 h forecast and during the space weather event.

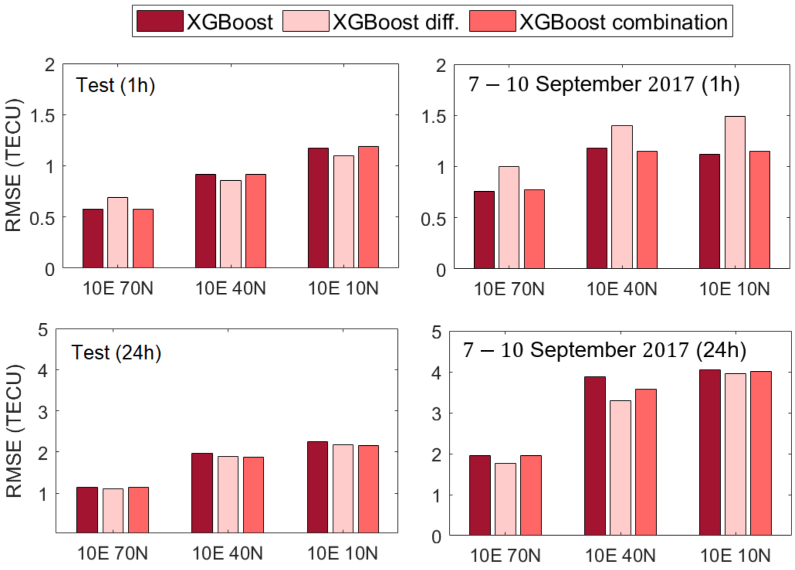


Figure 15. The RMSE for test dataset (**right**) and geomagnetic storm (7–10 September 2017) (**left**) for 1 h (**top**) and 24 h (**bottom**) forecasts for the XGBoost model. Models trained using differenced data are marked with "diff", while "combination" denotes differenced and non-differenced data together, where exponantial moving averages and time derivatives are calculated from non-differenced VTEC, while the model output is non-differenced VTEC.

Values of VTEC from machine learning models and GIM CODE VTEC were analyzed in more detail for the severe space weather event in September 2017 (Figure 16, left) including their differences (Figure 16, right).

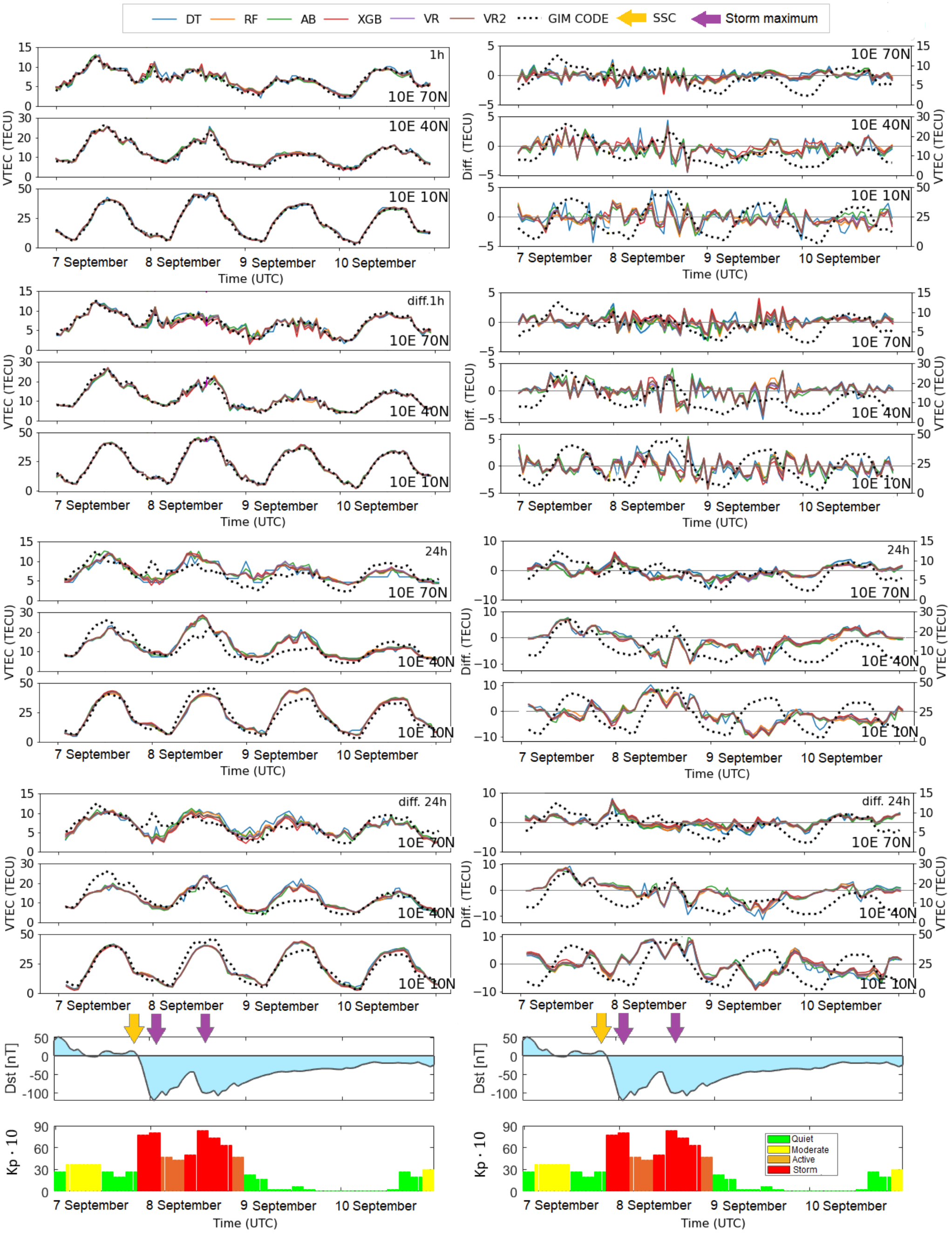


Figure 16. Left: VTEC values from machine learning models and ground truth VTEC from GIM CODE for the severe geomagnetic storm (7–11 September 2017). **Right**: Differences between VTEC from machine learning models and GIM CODE VTEC. From top to bottom: 1 h forecast with non-differenced data, 1 h forecast with differenced data, 24 h forecast with non-differenced data, 24 h forecast with differenced data, Dst index, and Kp index ·10. The yellow arrow denotes the sudden storm commencement (SSC) time, while the purple arrows point to two Dst minima, which correspond to the maximum phase of the geomagnetic storms.

The CMEs with earthward trajectories were emitted from the Sun on 4 and 6 September 2017 [51]. The first CME arrived on 6 September at about 23:43 UT leading to moderate geomagnetic conditions on 7 September (Figure 16). The solar wind shock from the second CME, originating from the intense X9.3 solar flare of September 6, caused a sudden storm commencement (SSC) at about 23 UT on 7 September. This led to severe geomagnetic storms in September with the maximum value of Kp = 8. The main phase of the storm was characterized by the two pronounced minima of Disturbance storm time (Dst) index at about 1:00 and 14 UT on 8 September. Afterward, the recovery phase started and lasted for about 3 days until 11 September [51]. Shortly after SSC, there is a sudden VTEC increase in high latitude. This peak is reproduced in a 1 h forecast and slightly in a 24 h forecast with differenced data. The largest differences are visible for the 1 day forecast for low-latitude VTEC on 8 September, i.e., during the maximum intensity of geomagnetic storms. On the other hand, 1 h forecast much better adapts to rapid changes in the ionosphere and can reproduce sudden intense variations during this space weather event. On 10 September, both 1 h and 1 day forecasts stabilize and have much lower differences from the ground truth during the recovery storm phase and a decrease in VTEC values. Overall, differences for the 1 h forecast are up to about 5 TECU, while for the 24 h forecast they are about twice as high, i.e., up to 10 TECU for all three latitudinal regions during sudden and intense irregular VTEC variations resulting from the space weather event. Figure 17 presents the scatter plot of the predicted and ground truth (GIM CODE) VTEC for the datasets from January 2015 to December 2016 (training) and January-December 2017 (testing) for the VR1 model. The highest correlations between predicted and GIM VTEC in test data mostly have the models with differenced data, while having lower correlation coefficients for training data than models with non-differenced data. This suggests that models with non-differenced data show a slightly better fit during training than models with differenced-data, resulting in a slightly lower correlation for test data. VTEC forecast (green) from the VR1 model for 1 h and 24 h are shown as time series in Figure 18, and as 2D maps as a function of *DOY* and *HOD* in Figure 19 for year 2017. It is interesting to observe that the models trained on non-differenced data overestimate the lowest VTEC values, particularly for the high-latitude grid point (Figure 18). In contrast, the models trained with differenced data are able to predict the lowest VTEC values. Sudden VTEC peaks are better captured with the 1 h forecast models than 24 h forecast models (Figures 18 and 19). Daily, seasonal and semi-annual VTEC variations are well predicted with the VR1 model (Figure 19). The maximum absolute differences between the GIM CODE and VR1 model are about 4 TECU for high-latitude and 7 TECU for the mid- and low-latitude points for the 1 h forecast, while for the 24 h forecast they are up to 8 TECU for high-latitude and to 12 TECU for the mid- and low-latitude points. However, most of the time, the differences are within 1 and 2.5 TECU for the 1 h forecast and within 2.5 and 5 TECU for the 1 day forecast for the high-latitude and mid-/low-latitude grid points, respectively. 1 day predicted GIM CODE (C1PG) provides VTEC with mostly lower values than the final GIM CODE and VR1 model (Figure 19). In addition, the GIM C1PG is mostly unable to predict VTEC peaks with the maximum absolute differences from the final GIM CODE up to 9 TECU, 17 TECU, and 14 TECU for high-, mid-, and low-latitude grid points.

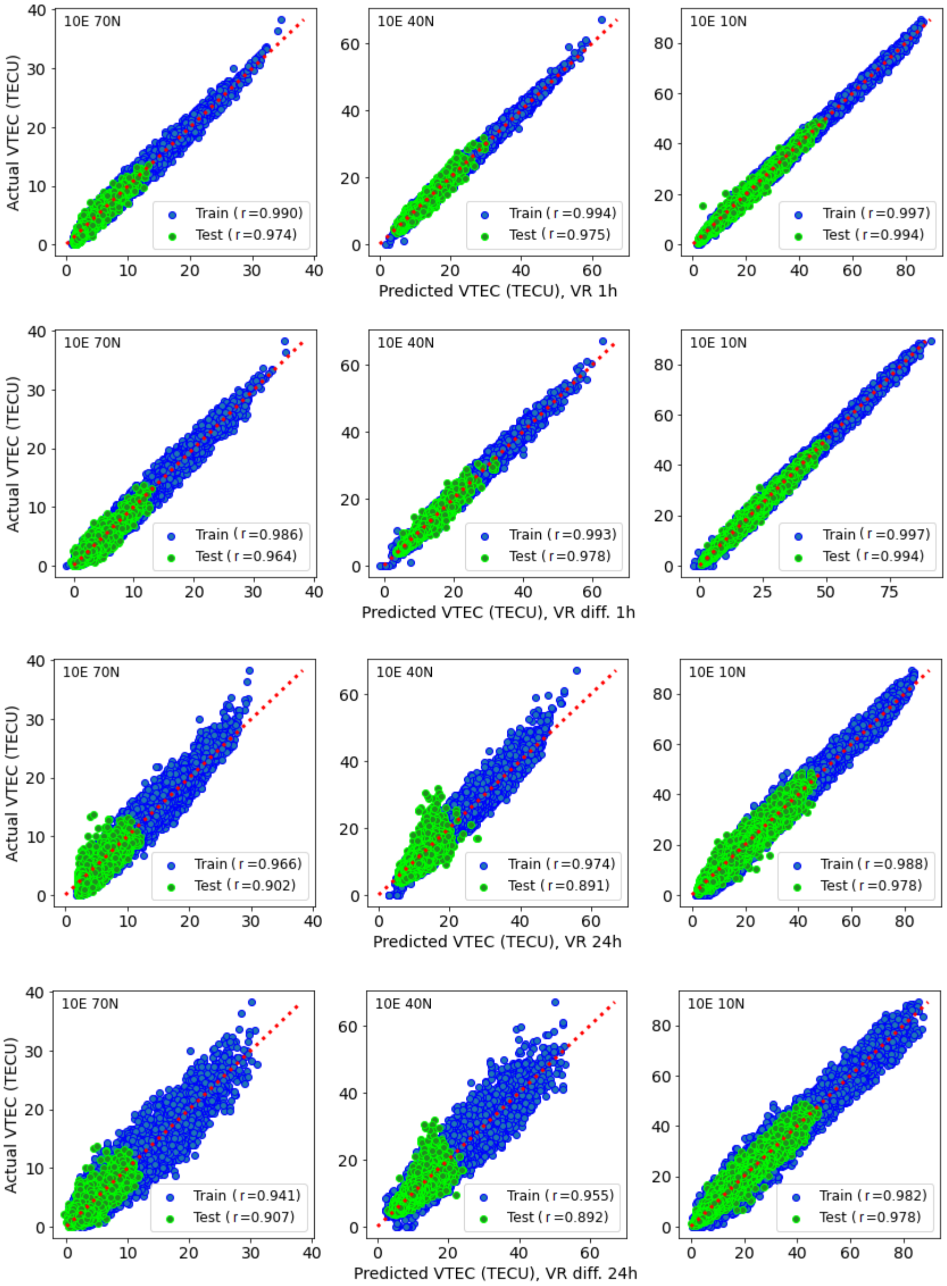


Figure 17. Predicted VTEC by the VR1 models vs. ground truth VTEC for training and cross-validation (blue) and test (green) datasets. **First row**: 1 h forecast with non-differenced data, **second row**: 1 h forecast with differenced data, **third row**: 24 h forecast with non-differenced data, **fourth row**: 24 h forecast with differenced data for high-latitude (**left**), mid-latitude (**middle**) and low-latitude (**right**) VTEC grid points.

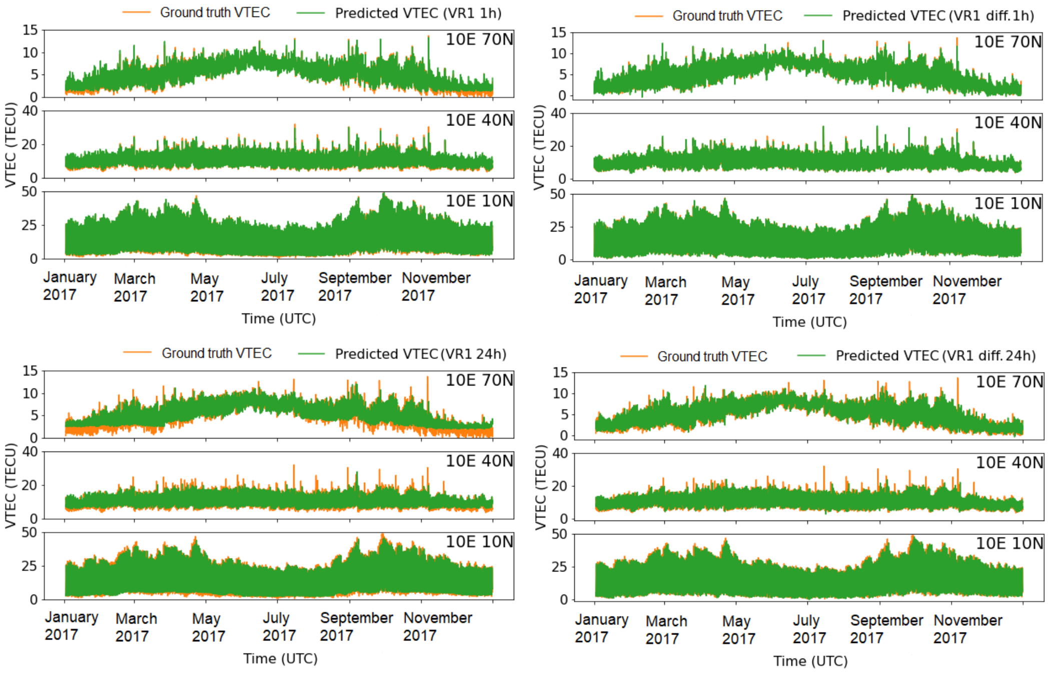


Figure 18. Forecasted VTEC by the VR1 model (green) and ground truth VTEC (orange) during year 2017. **Top**: 1 h, **bottom**: 24 h forecast. **Left**: non-differenced data, **right**: differenced data.

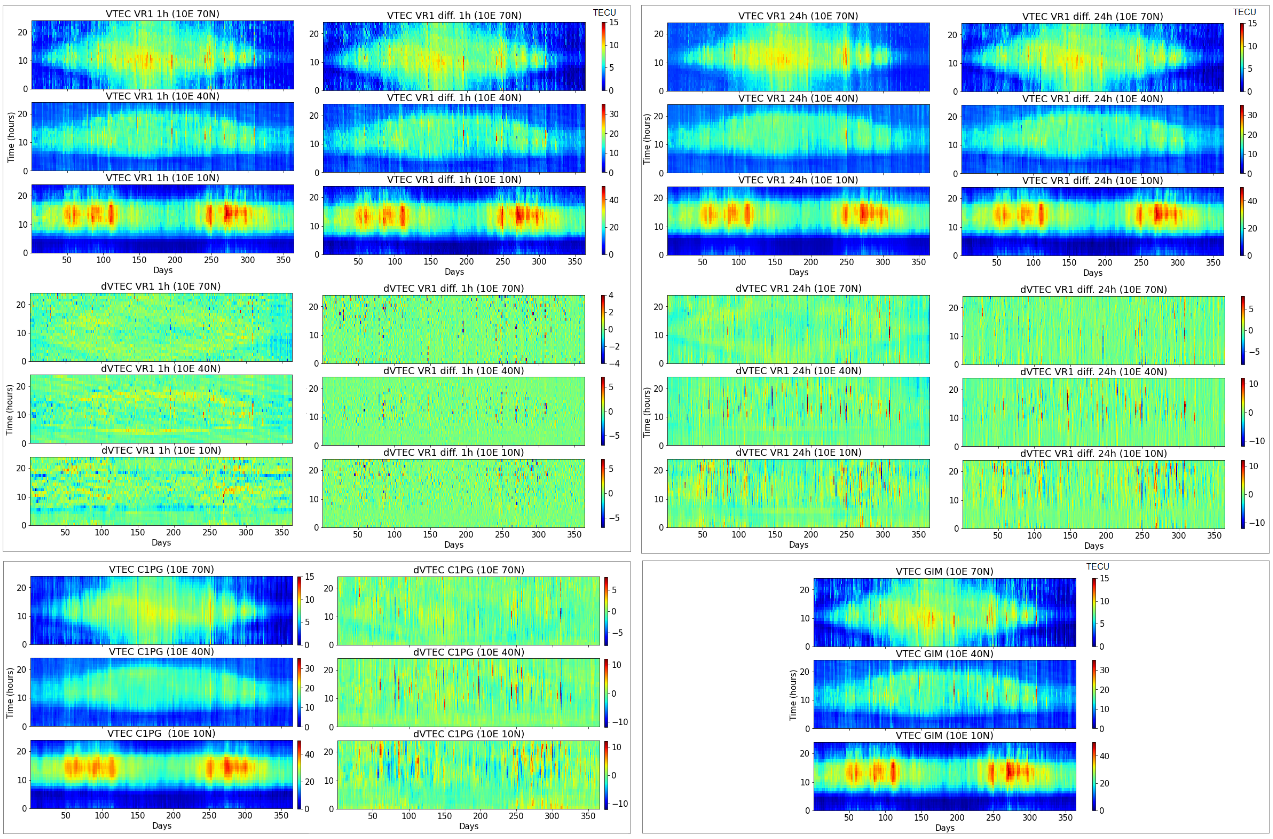


Figure 19. Upper panel left: VR1 model 1 h forecast (**top**) with non-differenced data (**left**) and differenced data (**right**), corresponding VTEC differences ($dVTEC = VTEC_{GIM} - VTEC_{VR1}$) (**bottom**). Upper panel right: VR1 model 24 h forecast (**top**) with non-differenced data (**left**) and differenced data (**right**), $dVTEC$ (**bottom**). Bottom panel left: 1 day predicted GIM CODE (C1PG) (**left**), $dVTEC = VTEC_{GIM} - VTEC_{C1PG}$ (**right**). Bottom panel right: GIM CODE.

4. Discussion

The relative importance of the input variables shows that the VTEC forecast with non-differenced data is mostly based on previous VTEC values, temporal information (h), and solar index (F10.7), especially for mid- and low- latitude VTEC. The contribution of other variables (solar wind speed, Bz, AE, Kp, Dst) is extremely small in those models or is not represented at all. The reason can be the strong correlations of VTEC(t) with VTEC(t + 1 h) and VTEC(t + 24 h) and with temporal information. In addition, the correlation between solar activity indices (F10.7 and R) and VTEC is much higher for non-differenced data than for differenced data. On the other hand, the correlations between VTEC, and solar wind, and magnetic activity increase for differenced data, leading to their higher contributions to the VTEC forecast. As a result, the models trained on differenced data use almost all of the input data for forecasting VTEC, especially during storm periods.

The performance of machine learning models is slightly different for non-differenced and differenced data. Using differenced data, the RMSE for cross-validation and test datasets are mostly consistent between different ensemble models. On the other hand, the differences between the models with the non-differenced data are more pronounced. Models with differenced data provided mostly better results on the test dataset for both the 1 h and 24 h forecast. During space weather events, improvements are observed over the longer forecasting horizon.

Ensemble learning improved the VTEC forecast from about 25% to 50% over the single tree. Larger improvements are visible for models with non-differenced data. Using differenced data there is less disagreement between Decision Tree and ensemble models. This suggests that differentiation facilitated learning of structural patterns in the data even using less optimal models such as Decision Tree. During the storm event, ensemble learning majorly improves forecast, especially over the 24 h forecasting horizon.

Regarding different machine learning models, Voting Regressor meta-ensemble models provided the lowest RMSE and the highest correlation coefficients. The lowest RMSE values provided are 0.6, 0.9, and 1.1 TECU for the high-, mid-, and low-latitude VTEC, respectively, for the 1 h forecast. For the 24 h forecast, the RMSE is about twice higher, resulting in 1.1, 1.9 and 2.1 TECU for the high-, mid- and low-latitude VTEC, respectively. During the severe storm in September 2017, the lowest RMSE is 0.7, 1.2 and 1.2 TECU for the high-, mid-, and low-latitude 1 h VTEC forecast, respectively, with Voting Regressor models. For the 24 h forecast horizon, the RMSE is higher, reaching 1.8, 3.4 and about 4 TECU for the high-, mid- and low-latitude VTEC, respectively. During the storm, the RMSE for the 24 h mid-latitude VTEC forecast with AdaBoost and non-differenced data is 3.9 TECU, while for the differenced data is 3.3 TECU. This shows an improvement of more than 0.50 TECU during the severe storm when the model is trained on the differenced data. Other ensemble models show similar results. Our ensemble learning models provide the high-latitude VTEC forecast (10°, 70°) with a lower RMSE than the LSTM model in [20], i.e., around 1 TECU and 2 TECU for the 1 h and 24 h forecast, respectively, during the September 2017 storm. The LSTM model [20] for the high-latitude VTEC forecast (138°, 57°) resulted in an RMSE of about 5 TECU during storm events (-150 nT $\leq Dst \leq -100$ nT). The feed-forward ANN provides a mid-latitude VTEC forecast with an average RMSE of about 5 TECU during geomagnetic storms [34], while our best performing Voting Regressor model provides the 1 h and 24 h mid-latitude VTEC forecasts with an RMSE of 1.20 and 3.40 TECU, respectively, during the severe geomagnetic storm. In 2017, its mid-latitude VTEC forecast is below 1 TECU, being better than the SVM model in 2018 (RMSE 1.5 TECU) [30]. The 1 day mid-latitude VTEC forecast values with our models have RMSE below 2 TECU which is in line with the LSTM model performance in 2016 [32]. The 1 h low-latitude VTEC forecast from the Voting Regressor model has twice lower RMSE (about 1.1 TECU) in 2017 than the LSTM-CNN models in 2016 [22] and in 2018 [21] and slightly better (for 0.4 TECU) than the SVM model in 2018 [30]. The GBDT model provides 1 h low-latitude VTEC forecasting with an RMSE of about 3 TECU in 2015 and 3–4 TECU during the geomagnetic storm [27]. Our XGBoost model has RMSE below 1.5 TECU for the 1 h low-latitude VTEC forecast during severe storm. Moreover, our XGBoost model provides the VTEC forecast for three ionospheric grid points with the mean RMSE below 1 and 2 TECU for the 1 h and 24 h forecast horizons, respectively, in 2017, and about 3 TECU for 24 h forecast during the September 2017 storm. The XGBoost model [29] has a global average RMSE of around 2.5 TECU in 2017, while during the September storm has a higher RMSE of about 8 TECU. The autoregressive neural network model [24] provides the 1 day global VTEC forecast with an RMSE from 3.4 to 5.1 TECU from May 2017 to February 2018, while the cGAN model [26] has an RMSE of about 1.74 TECU from 2017 to 2018. The nearest neighbour [31] resulted in an RMSE of about 4 TECU in 2015 and 2 TECU in 2018. Our ensemble learning models have the average RMSE for three ionospheric regions below 2 TECU for the 1 day forecast from January to December 2017. It should be taken into account that the discussed studies use different datasets for forecasting VTEC for different locations or regions during similar or different time periods.

5. Conclusions

This paper presents the development of machine learning models for ionosphere VTEC forecasting exploiting different learning algorithms from single Decision Tree to ensemble learning. The approach is presented for three grid points of different latitudes along the same longitudinal band. Of course, the presented methodology can be extended to cover larger regions by training the models on VTEC data from different grid points. The models are data-driven, gaining insights and knowledge from the data describing the solar activity, solar wind speed, interplanetary and Earth's magnetic field, and the ionosphere. In addition, a time series cross-validation method is implemented and the impact of the different sizes of k-folds on the VTEC forecasting is analyzed, especially for the low-latitude region. The study has further investigated the performance of machine learning models in terms of the data, where original and transformed data were used. The second approach was a differentiation with respect to values of the previous day to remove/reduce trends related to daily variations.

Looking at the different models, combining a large number of trees in an ensemble, such as Random Forest and boosting, significantly improve the accuracy and even outperform a single Decision Tree solution. The optimal accuracy and generalization are achieved by combining tree-based ensemble models in a meta-model of Voting Regressor. The use of differenced data instead of original data results in an RMSE improvement of more than 0.5 TECU for 24 h forecast during a severe storm. Such improvements are also visible for the 1 h and 24 h forecasts in 2017. Only in the case of the 1 h VTEC forecast during the storm, the models with non-differenced data perform clearly better, i.e., have a smaller RMSE value. Including additional input such as exponential moving averages and time derivatives further reduces the RMSE by up to 0.5 TECU. Relative RMSE decrease with respect to the persistence (naive) forecasting is from 15% to more than 70% for the 1 h forecast, and from 5% to 25% for the 24 h forecast. Differences to the final GIM CODE are mostly within 2.5 TECU for the 1 h forecast, and within 5 TECU for the 1 day forecast.

Based on these results, we can answer the questions raised at the beginning of this paper:

1. The new, proposed learning VTEC models can capture variations in electron content consistent with ground truth for both 1 h and 1 day forecasts.
2. The ensemble meta-models (VR1 and VR2) improve the VTEC forecasting over each individual model in the ensemble and deliver optimal results.
3. Including additional input features, such as moving averages and time derivatives, is beneficial to increase the accuracy of the models.
4. Data modification in the form of differencing enhances the VTEC model performance for a longer (24-h) forecast, including a geomagnetic storm.

The proposed VTEC models have perspectives to be used as a useful source of information for single-frequency GNSS users to mitigate the ionospheric delay and thereby, directly reduce the ionospheric range error. For instance, the final GIM CODE improves 3D single-frequency position estimates by about 5.5 m, 1 m, and 2.5 m on average in high-, mid-, and low-latitude regions, respectively, compared to the real-time available Klobuchar model [52]. However, considering that the final GIMs are usually provided with a time delay of 1–2 weeks, and the rapid GIMs with 1–2 days [53], their application in real-time is not possible. There are also ultra-rapid GIMs provided with latency of 2–3 h [54] and real-time (RT) GIMs [53,55]. The accuracy of RT GIMs is typically worse than final, post-processed GIMs due to the shorter span of observations, higher noise in carrier-to-code leveling, and difficulty in carrier ambiguity estimation in real-time processing mode [53]. Considering that 1 TECU corresponds to 0.162 m in L1 signal delay, differences between our developed models and the GIM CODE of 1 and 2.5 TECU for the high-latitude, and 2.5 and 5 TECU for the mid-/low-latitude grid points for the 1 h and 1 day forecast, respectively, result in L1 delay difference of about 0.2 m to 0.8 m. This suggests that our 1 h and 1 day VTEC forecast models are expected to improve the GNSS position estimates much more than the Klobuchar model in the studied locations. Thus, forecasted VTEC information can be used to support positioning applications. For a regional or global application, the models should of course be spatially expanded. For operational purposes, the model needs to use VTEC input from the rapid or RT GIMs or estimate it directly from GNSS observations.

The study shows promising results for the application of tree-based ensemble machine learning for VTEC forecasts. This approach has the potential to forecast VTEC in different ionospheric regions during quiet and storm periods. In further work, we plan to extend the models to additional locations to forecast VTEC at the regional or global level. Furthermore, the results support the idea of data importance, which is the core of machine learning and one of the major drivers of machine learning performance. Therefore, future studies will concentrate on further data exploration and modification in order to find the most optimal dataset from which the model can learn, especially over longer forecasting horizons and during space weather. The integration of additional input data that can further characterize space weather in a form useful for learning is intended. In addition, models with longer forecast horizons and multi-epoch predictions are to be developed. An investigation that includes more space weather events will be undertaken and the results should be validated for the latest time period as a new solar cycle is progressing. Another point of interest is the comparison with the neural network-based approach that has so far mainly been used in the area of ionospheric VTEC forecasting. To achieve an objective comparison, the same datasets should be used, and testing should be carried out for the same time period and locations. In addition, the uncertainty of VTEC predictions needs to be quantified, such as in [33], in order to better define the efficiency of the models, provide trustworthy results, and increase the reliability of the VTEC predictions.

Author Contributions: Conceptualization: R.N.; Data curation: R.N.; Formal analysis: R.N.; Methodology, R.N.; Software, R.N.; Investigation: R.N.; Validation: R.N.; Visualization: R.N.; Writing—original draft: R.N.; Writing—review and editing: R.N., B.S. and M.S.; Supervision: B.S. and M.S.; Funding acquisition: R.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Research Grants—Doctoral Programmes in Germany from German Academic Exchange Service (Deutscher Akademischer Austauschdienst, DAAD). Open Access funding is enabled by the Technical University of Munich (TUM) in the framework of the Open Access Publishing Program.

Data Availability Statement: GIM products are available at https://cddis.nasa.gov/archive/gnss/products/ionex, accessed on 10 March 2022. Other used data can be obtained from NASA/GSFC's OMNIWeb https://omniweb.gsfc.nasa.gov/form/dx1.html, accessed on 15 July 2022.

Acknowledgments: We acknowledge the use of NASA/GSFC's Space Physics Data Facility's OMNIWeb (or CDAWeb or ftp) service and OMNI data, providers of OMNI data namely: Belgium SILSO Center, GFZ Potsdam, World Data Center for Geomagnetism Kyoto, as well as Center for Orbit Determination in Europe (CODE) of the University of Bern for the GIM data. We thank the reviewers whose valuable comments and suggestions helped improve and clarify this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AdaBoost	Adaptive Boosting
AB	AdaBoost
DOY	Day Of Year
HOD	Hour Of Day
DT	Decision Tree
GIM	Global Ionosphere Map
GNSS	Global Navigation Satellite System
LSTM	Long Short-Term Memory
r	correlation coefficient
RMSE	Root Mean Square Error
RF	Random Forest
SW	Solar Wind speed
TECU	Total Electron Content Unit
VTEC	Vertical Total Electron Content
VR	Votting Regressor
XGBoost	eXtreme Gradient Boosting
XGB	XGBoost

References

1. Coster, A.; Komjathy, A. Space Weather and the Global Positioning System. *Space Weather* **2008**, *6*, 1–6. [CrossRef]
2. Klobuchar, J.A. Ionospheric Time-Delay Algorithm for Single-Frequency GPS Users. *IEEE Trans. Aerosp. Electron. Syst.* **1987**, *AES-23*, 325–331. [CrossRef]
3. Roma, D.; Pajares, M.; Krankowski, A.; Kotulak, K.; Ghoddousi-Fard, R.; Yuan, Y.; Li, Z.; Zhang, H.; Shi, C.; Wang, C.; et al. Consistency of seven different GNSS global ionospheric mapping techniques during one solar cycle. *J. Geod.* **2017**, *92*, 691–706. [CrossRef]
4. Yuan, Y.; Wang, N.; Li, Z.; Huo, X. The BeiDou global broadcast ionospheric delay correction model (BDGIM) and its preliminary performance evaluation results. *Navigation* **2019**, *66*, 55–69. [CrossRef]
5. Cander, L.R. Ionospheric Variability. In *Ionospheric Space Weather*; Springer: Berlin, Germany, 2019; pp. 59–93.
6. Nishimura, Y.; Verkhoglyadova, O.; Deng, Y.; Zhang, S.R. (Eds.) *Cross-Scale Coupling and Energy Transfer in the Magnetosphere-Ionosphere-Thermosphere SYSTEM*; Elsevier: Amsterdam, The Netherlands, 2021. [CrossRef]
7. Pulinets, S.; Ouzounov, D. Lithosphere–Atmosphere–Ionosphere Coupling (LAIC) model—An unified concept for earthquake precursors validation. *J. Asian Earth Sci.* **2011**, *41*, 371–382. [CrossRef]
8. Luo, X.; Du, J.; Lou, Y.; Gu, S.; Yue, X.; Liu, J.; Chen, B. A Method to Mitigate the Effects of Strong Geomagnetic Storm on GNSS Precise Point Positioning. *Space Weather* **2022**, *20*, e2021SW002908. [CrossRef]
9. Luo, X.; Gu, S.; Lou, Y.; Xiong, C.; Chen, B.; Jin, X. Assessing the Performance of GPS Precise Point Positioning Under Different Geomagnetic Storm Conditions during Solar Cycle 24. *Sensors* **2018**, *18*, 1784. [CrossRef]
10. Natras, R.; Horozovic, D.; Mulic, M. Strong solar flare detection and its impact on ionospheric layers and on coordinates accuracy in the Western Balkans in October 2014. *SN Appl. Sci.* **2019**, *1*, 1–14. [CrossRef]
11. Yuan, Y.; Ou, J. An improvement to ionospheric delay correction for single-frequency GPS users—The APR-I scheme. *J. Geod.* **2001**, *75*, 331–336. [CrossRef]
12. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. [CrossRef]
13. Natras, R.; Schmidt, M. Machine Learning Model Development for Space Weather Forecasting in the Ionosphere. In Proceedings of the CEUR Workshop, Gold Coast, Australia, 1–5 November 2021; Volume 3052.
14. Camporeale, E.; Wing, S.; Johnson, J. *Machine Learning Techniques for Space Weather*; Elsevier: Amsterdam, The Netherlands, 2018.
15. Adolfs, M.; Hoque, M.M. A Neural Network-Based TEC Model Capable of Reproducing Nighttime Winter Anomaly. *Remote Sens.* **2021**, *13*, 4559. [CrossRef]
16. Natras, R.; Goss, A.; Halilovic, D.; Magnet, N.; Mulic, M.; Schmidt, M.; Weber, R. Regional ionosphere delay models based on CORS data and machine learning. *Navig. J. Inst. Navig.* 2022, *in review*.
17. Tebabal, A.; Radicella, S.; Damtie, B.; Migoya-Orue', Y.; Nigussie, M.; Nava, B. Feed forward neural network based ionospheric model for the East African region. *J. Atmos. Sol.-Terr. Phys.* **2019**, *191*, 105052. [CrossRef]
18. Liu, L.; Zou, S.; Yao, Y.; Wang, Z. Forecasting Global Ionospheric TEC Using Deep Learning Approach. *Space Weather* **2020**, *18*, e2020SW002501. [CrossRef]
19. Srivani, I.; Siva Vara Prasad, G.; Venkata Ratnam, D. A Deep Learning-Based Approach to Forecast Ionospheric Delays for GPS Signals. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1180–1184. [CrossRef]
20. Tang, R.; Zeng, F.; Chen, Z.; Wang, J.S.; Huang, C.M.; Wu, Z. The Comparison of Predicting Storm-Time Ionospheric TEC by Three Methods: ARIMA, LSTM, and Seq2Seq. *Atmosphere* **2020**, *11*, 316. [CrossRef]

21. Kaselimi, M.; Voulodimos, A.; Doulamis, N.; Doulamis, A.; Delikaraoglou, D. Deep Recurrent Neural Networks for Ionospheric Variations Estimation Using GNSS Measurements. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [CrossRef]
22. Ruwali, A.; Kumar, A.J.S.; Prakash, K.B.; Sivavaraprasad, G.; Ratnam, D.V. Implementation of Hybrid Deep Learning Model (LSTM-CNN) for Ionospheric TEC Forecasting Using GPS Data. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 1004–1008. [CrossRef]
23. Xiong, P.; Zhai, D.; Long, C.; Zhou, H.; Zhang, X.; Shen, X. Long Short-Term Memory Neural Network for Ionospheric Total Electron Content Forecasting Over China. *Space Weather* **2021**, *19*, e2020SW002706. [CrossRef]
24. Cesaroni, C.; Spogli, L.; Aragon-Angel, A.; Fiocca, M.; Dear, V.; De Franceschi, G.; Romano, V. Neural network based model for global Total Electron Content forecasting. *J. Space Weather Space Clim.* **2020**, *10*, 11. [CrossRef]
25. Sivavaraprasad, G.; Lakshmi Mallika, I.; Sivakrishna, K.; Venkata Ratnam, D. A novel hybrid Machine learning model to forecast ionospheric TEC over Low-latitude GNSS stations. *Adv. Space Res.* **2022**, *69*, 1366–1379. [CrossRef]
26. Lee, S.; Ji, E.Y.; Moon, Y.J.; Park, E. One day Forecasting of Global TEC Using a Novel Deep Learning Model. *Space Weather* **2020**, *19*, 2020SW002600. [CrossRef]
27. Han, Y.; Wang, L.; Fu, W.; Zhou, H.; Li, T.; Chen, R. Machine Learning-Based Short-Term GPS TEC Forecasting During High Solar Activity and Magnetic Storm Periods. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 115–126. [CrossRef]
28. Ghaffari Razin, M.R.; Voosoghi, B. Ionosphere time series modeling using adaptive neuro-fuzzy inference system and principal component analysis. *GPS Solut.* **2020**, *24*, 1–13. [CrossRef]
29. Zhukov, A.V.; Yasyukevich, Y.V.; Bykov, A.E. Correction to: GIMLi: Global Ionospheric total electron content model based on machine learning. *GPS Solut.* **2021**, *25*, 21. [CrossRef]
30. Xia, G.; Liu, Y.; Wei, T.; Wang, Z.; Huang, W.; Du, Z.; Zhang, Z.; Wang, X.; Zhou, C. Ionospheric TEC forecast model based on support vector machine with GPU acceleration in the China region. *Adv. Space Res.* **2021**, *68*, 1377–1389. [CrossRef]
31. Monte-Moreno, E.; Yang, H.; Hernández-Pajares, M. Forecast of the Global TEC by Nearest Neighbour Technique. *Remote Sens.* **2022**, *14*, 1361. [CrossRef]
32. Wen, Z.; Li, S.; Li, L.; Wu, B.; Fu, J. Ionospheric TEC prediction using Long Short-Term Memory deep learning network. *Astrophys. Space Sci.* **2021**, *366*, 1–11. [CrossRef]
33. Natras, R.; Soja, B.; Schmidt, M. Machine Learning Ensemble Approach for Ionosphere and Space Weather Forecasting with Uncertainty Quantification. In Proceedings of the 2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC), Gran Canaria, Spain, 30 May–4 June 2022; pp. 1–4. [CrossRef]
34. Uwamahoro, J.C.; Habarulema, J.B. Modelling total electron content during geomagnetic storm conditions using empirical orthogonal functions and neural networks. *J. Geophys. Res. Space Phys.* **2015**, *120*, 11000–11012. [CrossRef]
35. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd ed.; Springer: Berlin, Germany, 2009. [CrossRef]
36. Blum, A.; Kalai, A.; Langford, J. Beating the Hold-out: Bounds for K-Fold and Progressive Cross-Validation. In Proceedings of the Twelfth Annual Conference on Computational Learning Theory, Santa Cruz, CA, USA, 7–9 July 1999; Association for Computing Machinery: New York, NY, USA, 1999; pp. 203–208. [CrossRef]
37. Arlot, S.; Celisse, A. A survey of cross-validation procedures for model selection. *Stat. Surv.* **2010**, *4*, 40–79. [CrossRef]
38. Hyndman, R.J.; Athanasopoulos, G. *Forecasting: Principles and Practice*, 3rd ed.; OTexts: Melbourne, Australia, 2021.
39. King, J.H.; Papitashvili, N.E. Solar wind spatial scales in and comparisons of hourly Wind and ACE plasma and magnetic field data. *J. Geophys. Res. Space Phys.* **2005**, *110*, 1–9. [CrossRef]
40. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
41. Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
42. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
43. Breiman, L.; Friedman, J.; Stone, C.; Olshen, R. *Classification and Regression Trees*; Taylor & Francis: Abingdon, UK, 1984.
44. Wong, T.T.; Yeh, P.Y. Reliable Accuracy Estimates from k-Fold Cross Validation. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 1586–1594. [CrossRef]
45. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
46. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
47. Esposito, D. *Introducing Machine Learning*, 1st ed.; Microsoft Press: Redmond, WA, USA; Safari: Boston, MA, USA, 2020.
48. Badeke, R.; Borries, C.; Hoque, M.M.; Minkwitz, D. Empirical forecast of quiet time ionospheric Total Electron Content maps over Europe. *Adv. Space Res.* **2018**, *61*, 2881–2890. [CrossRef]
49. García-Rigo, A.; Monte, E.; Hernández-Pajares, M.; Juan, J.M.; Sanz, J.; Aragón-Angel, A.; Salazar, D. Global prediction of the vertical total electron content of the ionosphere based on GPS data. *Radio Sci.* **2011**, *46*, 1–3. [CrossRef]
50. Verkhoglyadova, O.; Meng, X.; Mannucci, A.J.; Shim, J.S.; McGranaghan, R. Evaluation of Total Electron Content Prediction Using Three Ionosphere-Thermosphere Models. *Space Weather* **2020**, *18*, e2020SW002452. [CrossRef]
51. Imtiaz, N.; Younas, W.; Khan, M. Response of the low- to mid-latitude ionosphere to the geomagnetic storm of September 2017. *Ann. Geophys.* **2020**, *38*, 359–372. [CrossRef]

52. Wang, G.; Yin, Z.; Hu, Z.; Chen, G.; Li, W.; Bo, Y. Analysis of the BDGIM Performance in BDS Single Point Positioning. *Remote Sens.* **2021**, *13*, 3888. [CrossRef]
53. Liu, Q.; Hernández-Pajares, M.; Lyu, H.; Goss, A. Influence of temporal resolution on the performance of global ionospheric maps. *J. Geod.* **2021**, *95*, 34. [CrossRef]
54. Goss, A.; Schmidt, M.; Erdogan, E.; Görres, B.; Seitz, F. High-resolution vertical total electron content maps based on multi-scale B-spline representations. *Ann. Geophys.* **2019**, *37*, 699–717. [CrossRef]
55. Erdogan, E.; Schmidt, M.; Goss, A.; Görres, B.; Seitz, F. Real-Time Monitoring of Ionosphere VTEC Using Multi-GNSS Carrier-Phase Observations and B-Splines. *Space Weather* **2021**, *19*, e2021SW002858. [CrossRef]

A3 Publication III

Reference

Natras, R., Soja, B., and Schmidt, M. (2023). **Uncertainty Quantification for Machine Learning-based Ionosphere and Space Weather Forecasting: Ensemble, Bayesian Neural Network and Quantile Gradient Boosting**. Space Weather, 21, e2023SW003483. https://doi.org/10.1029/2023SW003483

Copyright

This paper was published in the open access journal Space Weather, AGU Wiley. It is available under the Creative Commons license at https://agupubs.onlinelibrary.wiley.com/journal/15427390. The copyrights remain with the authors.

Declaration of own contribution

The idea for this publication originated from **P-II**, where ensemble modeling is applied, which provided an opportunity to evaluate uncertainties and raised the question of other approaches to quantify uncertainties in machine learning-based VTEC models. Thus, the idea of the paper has its source in previous works of R. Natras and was extended to different approaches to perform a detailed analysis and develop a reliable probabilistic VTEC forecast model. The concept and outline of the paper were proposed by R. Natras and discussed with all co-authors. The part of the software comes from **P-II**, realized by R. Natras. Other parts of the software were implemented in the Python programming language and the open-source machine learning software libraries ScikitLearn (Pedregosa et al., 2011) and TensorFlow (Abadi et al., 2015) by R. Natras. The computations, analyses, and evaluations were performed by R. Natras with valuable input from the co-authors. All co-authors supported the progress through discussions of the results and suggestions regarding the validation and analysis. All figures and tables included in the paper were compiled by R. Natras with suggestions from co-authors. The figures were created in Python using Seaborn (Waskom, 2021) and Matplotlib (Hunter, 2007) and in MATLAB (MATLAB, 2020). The manuscript was written by R. Natras and reviewed and edited by the co-authors.
The overall contribution of R. Natras to P-III is estimated to be 89%.

Space Weather®

RESEARCH ARTICLE
10.1029/2023SW003483

Key Points:

- Machine learning-based Vertical Total Electron Content models with 95% confidence intervals (CI) are developed for the first time using four approaches to quantify uncertainties
- Bayesian Neural Network quantifying model and data uncertainties contains ground truth within CIs, but is computationally intensive
- Quantile Gradient Boosting is fastest with comparable performance in terms of uncertainty; CIs largely determined from space weather indices

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

R. Natras,
randa.natras@tum.de

Citation:

Natras, R., Soja, B., & Schmidt, M. (2023). Uncertainty quantification for machine learning-based ionosphere and space weather forecasting: Ensemble, Bayesian neural network and quantile gradient boosting. *Space Weather*, *21*, e2023SW003483. https://doi.org/10.1029/2023SW003483

Received 9 MAR 2023
Accepted 12 SEP 2023

Author Contributions:

Conceptualization: Randa Natras, Benedikt Soja, Michael Schmidt
Data curation: Randa Natras
Formal analysis: Randa Natras
Funding acquisition: Randa Natras
Investigation: Randa Natras
Methodology: Randa Natras, Benedikt Soja
Software: Randa Natras
Supervision: Michael Schmidt
Validation: Randa Natras
Visualization: Randa Natras
Writing – original draft: Randa Natras

Uncertainty Quantification for Machine Learning-Based Ionosphere and Space Weather Forecasting: Ensemble, Bayesian Neural Network, and Quantile Gradient Boosting

Randa Natras[1], Benedikt Soja[2], and Michael Schmidt[1]

[1]Deutsches Geodätisches Forschungsinstitut (DGFI-TUM), TUM School of Engineering and Design, Technical University of Munich, Munich, Germany, [2]Institute of Geodesy and Photogrammetry, ETH Zurich, Zurich, Switzerland

Abstract Machine learning (ML) has been increasingly applied to space weather and ionosphere problems in recent years, with the goal of improving modeling and forecasting capabilities through a data-driven modeling approach of nonlinear relationships. However, little work has been done to quantify the uncertainty of the results, lacking an indication of how confident and reliable the results of an ML system are. In this paper, we implement and analyze several uncertainty quantification approaches for an ML-based model to forecast Vertical Total Electron Content (VTEC) 1-day ahead and corresponding uncertainties with 95% confidence intervals (CI): (a) Super-Ensemble of ML-based VTEC models (SE), (b) Gradient Tree Boosting with quantile loss function (Quantile Gradient Boosting, QGB), (c) Bayesian neural network (BNN), and (d) BNN including data uncertainty (BNN + D). Techniques that consider only model parameter uncertainties (a and c) predict narrow CI and over-optimistic results, whereas accounting for both model parameter and data uncertainties with the BNN + D approach leads to a wider CI and the most realistic uncertainties quantification of VTEC forecast. However, the BNN + D approach suffers from a high computational burden, while the QGB approach is the most computationally efficient solution with slightly less realistic uncertainties. The QGB CI are determined to a large extent from space weather indices, as revealed by the feature analysis. They exhibit variations related to daytime/nightime, solar irradiance, geomagnetic activity, and post-sunset low-latitude ionosphere enhancement.

Plain Language Summary Space weather describes the varying conditions in the space environment between the Sun and Earth that can affect satellites and technologies on Earth, such as navigation systems, power grids, radio, and satellite communications. The manifestation of space weather in the ionosphere can be characterized using the Vertical Total Electron Content (VTEC) derived from Global Navigation Satellite Systems observations. In this study, the machine learning (ML) approach is applied to approximate the nonlinear relationships of Sun-Earth processes using data on solar activity, solar wind, magnetic field, and VTEC. However, the measurements and the modeling approaches are subject to errors, increasing the uncertainty of the results when forecasting future instances. For reliable forecasting, it is necessary to quantify the uncertainties. Quantifying the uncertainty is also helpful for understanding the ML-based model and the problem of VTEC and space weather forecasting. Therefore, in this study, ML-based models are developed to forecast VTEC within the ionosphere, including the manifestation of space weather, while the degree of reliability is quantified with a target value of 95% confidence.

1. Introduction

Space weather has been identified as a natural hazard to the modern technical infrastructure on which our society is highly dependent. Its accurate and reliable modeling and forecast are therefore essential and rely on modeling nonlinear solar-terrestrial coupling processes. The last few years have witnessed a huge growth in the use of machine learning (ML) and deep learning (DL) to predict complex space weather phenomena, from conditions on the Sun to their effects on Earth (including the ionosphere). Over the next decade, we expect continued rapid development and adaptation of emerging ML/DL tools for operational forecasting systems. However, there is considerable concern about trusting the results of ML/DL models and treating them as a black box because they are difficult to interpret. One of the main issues of previous work is the lack of transparency, as there is no indication when those results should not be trusted, which may lead to scientific skepticism toward ML and DL. Despite their widespread use, there has been little discussion on probabilistic ML/DL and uncertainty quantification (UQ) in the space weather domain. Most studies have focused on providing a single

Writing – review & editing: Randa Natras, Benedikt Soja, Michael Schmidt

prediction/forecast for each input (deterministic), while probabilistic predictions/forecasts have usually not been addressed. The present paper aims to alleviate these issues by extending ML-based models to quantify uncertainty in order to develop the probabilistic model for Vertical Total Electron Content (VTEC) within the ionosphere.

The uncertainty can be classified into two main categories (Abdar et al., 2021; Hüllermeier & Waegeman, 2021; Siddique et al., 2022):

- Model parameter uncertainty: it occurs due to incomplete knowledge, which can be due to a lack of training data or training data information poor. This is the deterministic part of uncertainty, which can be reduced with more knowledge about the system, for example, by adding more information-rich data.
- Data uncertainty: it is related to uncertainty in measurements, which is due to the noise inherent in the data or the stochastic nature of the process generating the data. This is the stochastic part of uncertainty, caused by randomness, and therefore irreducible.

In the ML literature, these uncertainties are often referred to as epistemic and aleatoric uncertainties, respectively (Abdar et al., 2021; Hüllermeier & Waegeman, 2021). The third source of uncertainty relates to the limitation of the learning model to approximate the target function. This is not easy to quantify accurately. For example, model selection involves a particular choice of hyperparameters, and it is impossible to fully explore the hyperparameter space. The choice of hyperparameters can significantly impact the model's accuracy, complexity, and computational cost. Thus, it comes down to a trade-off between the complexity of the model to capture higher-order nonlinear functions and its ability to generalize to unseen data. Ultimately, the ML process consists of various steps of learning and approximating an unknown mapping function from input to output, and the errors and uncertainties associated with these steps may contribute to the uncertainty of the model output.

The most commonly used UQ approaches for ML, in general, are the deep ensembles technique and the Bayesian approximation (Abdar et al., 2021; Kendall & Gal, 2017; Rahaman & Thiery, 2021). There are few examples of UQ studies for estimating a continuous variable in the space weather domain, such as artificial neural network (ANN) with Monte Carlo (MC) dropout as a Bayesian approximation and the negative log-likelihood (NLL) loss function for thermospheric density prediction (Licata & Mehta, 2022), BNN for the geomagnetically induced currents (Siddique et al., 2022), as well as, a least squares-based ensemble of convolutional neural networks (CNN) for the geomagnetic Dst index prediction (Hu et al., 2022). It has been shown that ANN with MC dropout and NLL loss requires much more computational time than ANN with NLL loss and direct probability prediction, but both approaches demonstrated similar accuracy (Licata & Mehta, 2022). Siddique et al. (2022) highlight that estimating uncertainties allows quantifying the degree of reliability of the ML-based model but does not necessarily increase the model accuracy. The least squares-based weighting of the CNN ensemble with a class-balanced cost function was used to account for the imbalance between storm and non-storm cases and provide probabilistic Dst prediction (Hu et al., 2022). The least squares inclusion of both input and output data uncertainties with Bayesian learning using a Long Short-Term Memory neural network resulted in better generalization when applied to the prediction of Earth orientation parameters and Global Navigation Satellite Systems station coordinates (Kiani Shahvandi & Soja, 2022). The initial study of an ML ensemble approach to VTEC forecast in Natras et al. (2022b) showed higher accuracy and improved generalization compared to a single-model approach, with uncertainties estimated as ensemble spread. Other studies on ML-based VTEC modeling and forecasting, such as Y. Han et al. (2022), Kaselimi et al. (2022), L. Liu et al. (2020), Lee et al. (2020), to name a few, have not quantified the uncertainties or provided confidence intervals (CI) of VTEC output, leading to a lack of information on how certain and reliable their ML-based VTEC results are; Natras et al. (2022a) provides an overview of these studies and their results.

Based on the review of existing literature, there has been little discussion on probabilistic ML/DL for VTEC and space weather in general. In this study, we aim to fill this gap by developing and adapting UQ techniques for ML-based VTEC forecasting to produce a probabilistic VTEC model. With this in mind, we analyze and discuss the effectiveness of various techniques for estimating uncertainties and 95% CI of 1-day VTEC forecasting for both quiet and extreme space weather conditions. Section 2 begins with an overview of data preparation, then describes four methods for estimating uncertainty, and ends with an outline of models and hyperparameters optimization. Section 3 provides a detailed analysis of the UQ models for test case studies. Our conclusions are drawn in the final section.

2. Methodology

2.1. Data

This study deals with supervised learning, in which a set of both input and output data is clearly specified and prepared, called training data, which is needed to learn the function that maps the input variables to an output variable. A training sample consists of the vector $\mathbf{x}_i$ and an output $y_i = F(\mathbf{x}_i)$ with $i = \{1, 2, \ldots, N\}$. The vectors $\mathbf{x}_i$ can be interpreted as the rows of the $N \times P$ predictor matrix $\mathbf{X} = \left(\mathbf{X}_i^T\right)$, whereas the columns represent the input features $\widetilde{\mathbf{x}}_p$ with $p = \{0, 1, 2, \ldots, P - 1\}$ (Natras et al., 2022a).

In this study, VTEC is obtained from Center for Orbit Determination in Europe (CODE) global ionospheric maps (GIM), computed via spherical harmonics up to degree 15 (Schaer, 1999), and interpreted as GT. Because the temporal resolution of the CODE GIM was updated from 2 hr to 1 hr in 2015, we used data starting from January 2015 to develop the VTEC model with 1-hr intervals. September 2017 was an extremely active space weather period, with the Sun emitting 27 M-class and 4 X-class flares, as well as several earthward-directed coronal mass ejections (CME) (https://www.nasa.gov/feature/goddard/2017/september-2017s-intense-solar-activity-viewed-from-space). Therefore, the year 2017 is selected for testing, and data from January 2015 to December 2016 are used for training and cross-validation, The training set consists of a total of 17,544 samples, while the test set contains 8,760 samples. To model the solar-terrestrial processes and the impact of space weather on VTEC, data on solar and geomagnetic activity were downloaded from the OMNIWeb NASA Service and added as input features. The data set is prepared with a 1-hr resolution, denoted D1, corresponding to Table 1 of Natras et al. (2022a). It consists of the following input features of $\mathbf{x}_i$ at timestamp i:

(a) VTEC for grid points at 10° of longitude, and 10°, 40°, and 70° of latitude;
(b) OMNIWeb data: sunspot number, F10.7 solar radio flux, solar wind plasma speed, interplanetary magnetic field Bz index, geomagnetic field (GMF) Dst index, GMF Kp index, auroral electrojet (AE) index;
(c) Derived VTEC features: exponential moving average (EMA) of VTEC over the previous 30 and 4 days, first and second VTEC derivatives;
(d) Hour of the day (HoD) and day of the year (DoY),

and the output $\mathbf{y}_i = \mathbf{VTEC}(i + 24)$ for the 1-day forecast. Grid points for VTEC were selected along the same longitude (10°) to represent VTEC latitudinal variations alongside other VTEC variability. Separate models were developed for each grid point.

The input data for artificial neural networks were standardized to obtain data with a mean of zero and a standard deviation of one. Learning algorithms based on decision trees (Sections 2.2.1 and 2.2.2) do not require data normalization since they are not sensitive to the scale of input features, and the data were not standardized in these cases. Moreover, a neural network benefits from transforming time information to preserve its cyclic significance:

$$HoD_{\sin} = \sin\left(\frac{2\pi \cdot HoD}{24}\right), \quad HoD_{\cos} = \cos\left(\frac{2\pi \cdot HoD}{24}\right)$$

$$DoY_{\sin} = \sin\left(\frac{2\pi \cdot DoY}{365.25}\right), \quad DoY_{\cos} = \cos\left(\frac{2\pi \cdot DoY}{365.25}\right). \tag{1}$$

2.2. Methods

In the following, we present different approaches to determine model and data uncertainties in ML-based UQ VTEC models.

2.2.1. Ensemble Approach

Ensemble modeling combines multiple diverse models to predict an outcome using either different algorithms or different data sets. The ensemble model, called Super-Ensemble (SE) (Natras et al., 2022b), aggregates the mean result across all base models to produce a final prediction with reduced generalization error. This approach improves the prediction compared to the single base model within the ensemble by averaging the results over a set of functions of well-performing models (Natras et al., 2022b). In this study, ensemble modeling combines three learning algorithms, namely Random Forest (Breiman, 2001), Adaptive Boosting (AdaBoost) (Freund &

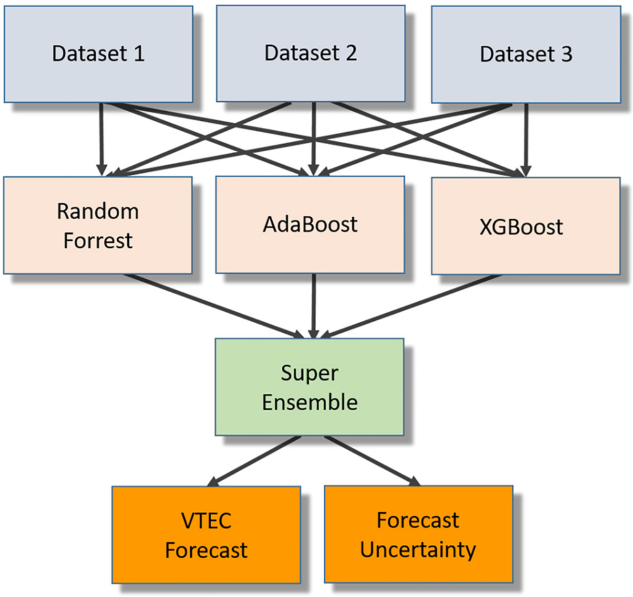


Figure 1. Flowchart of the ensemble modeling procedure trained on three different data sets using the three different learning algorithms Random Forest, AdaBoost, and eXtreme Gradient Boosting (XGBoost). After the nine individual model runs, the results are combined into a Super-Ensemble model, which provides the Vertical Total Electron Content forecast values and their uncertainty.

Schapire, 1997) and Gradient Boosting (Friedman, 2001) on three data sets consisting of different versions of input features and output. These algorithms are based on decision tree learning but follow different computation strategies. Random Forest belongs to the bagging approach of learning many diverse random trees, while the other two algorithms are boosting approaches of sequential learning that aim to reduce the errors of the tree from the previous step. Moreover, boosting is realized differently in AdaBoost by assigning different weights to observations depending on the model performance in the previous step and in Gradient Boosting by training the models on the gradient of the objective cost function of the previous step. For more details, see Natras et al. (2022a).

In addition to training the ensemble members with different learning algorithms, further randomness is introduced into the ensemble by training with different versions of the data set to increase the number of ensemble members and increase the diversity between them, as shown in Figure 1. Therefore, we created three sets of data from the D1 data set introduced in Section 2.1:

1. Data set D1 with $\mathbf{x}_i$, y_i for $i = 1, 2, \ldots, N$;
2. Daily differences for the input features and output: The data, except HoD and DoY, are time-differenced with $\Delta\mathbf{x}_i$, Δy_i by calculating the difference between an observation at time step $i + 24$ and observation at time step i so that $\Delta\mathbf{x}_i = \mathbf{x}_{i+24} - \mathbf{x}_i$ and $\Delta y_i = y_{i+24} - y_i$. The EMA and time derivatives of VTEC are calculated from the differenced VTEC values. At the end, the VTEC forecast is reconstructed from the forecasted VTEC daily difference by adding the VTEC value from 24 hr ago.
3. The input of the data set from point 1 and the input of the daily differenced data set from point 2 are used as input features, while the output comes from the data set from point 1.

Daily differences remove the dominant daily VTEC variations so that the model can learn the remaining signatures associated with other sources of VTEC fluctuations. Such a data strategy demonstrated improved generalization and accuracy of 1-day VTEC forecasting in ensemble tree learning (Natras et al., 2022a), as well as in the convLSTM VTEC model (L. Liu et al., 2022). In addition, differencing reduces temporal dependencies and trends and stabilizes the mean of the data set, which can improve modeling.

The employed cost function is the mean squared error (MSE), defined as

$$Cost_m = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(y_i, \hat{F}(\mathbf{x}_i)\right) = \frac{1}{N}\sum_{i=1}^{N} e_i^2 = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_{m_i}\right)^2, \tag{2}$$

where $\mathcal{L}$ is the loss function, y_i is the GT VTEC, $\hat{F}(\mathbf{x}_i)$ is an approximation function of the function $F(\mathbf{x}_i)$ that maps the input $\mathbf{x}_i$ to the output y_i, and $\hat{y}_{m_i}$ is the VTEC forecast of the mth model with $m = \{1, 2, \ldots, M\}$.

The ensemble approach can be viewed as an approximation of a distribution, and thus, its diversity can be used as an indicator of the model parameter uncertainty (Hüllermeier & Waegeman, 2021). In this case, the results of M independently trained models are averaged, forming a joined distribution $p(\mathbf{y}|\mathbf{X})$ as

$$p(\mathbf{y}|\mathbf{X}) = \frac{1}{M}\sum_{m=1}^{M} p(\mathbf{y}|\mathbf{X}, \theta), \tag{3}$$

where θ represents a set of model parameters. Nine models, $M = 9$, are developed for each of the 3 VTEC grid points, resulting in a total of 27 models. The randomness in the nine models in this study is introduced by the learning algorithms and the data. More specifically, by training the three algorithms mentioned above on each of the three data sets individually. The final output $\hat{y}_i$ is estimated as the ensemble mean μ_i

$$\hat{y}_i = \mu_i = \frac{1}{M}\sum_{m=1}^{M}\hat{y}_{m_i} \tag{4}$$

and the standard deviation across the ensemble for observation time i is defined as

$$\sigma_i = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left(\hat{y}_{m_i} - \hat{y}_i\right)^2}. \tag{5}$$

The standard deviation of the ensemble members with respect to the ensemble mean, known as the ensemble spread, provides an estimate of the uncertainties. The ensemble spread is represented as a probabilistic prediction in terms of lower bounds (LB) and upper bounds (UB) with 95% confidence, defined by

$$UB = \hat{y}_i + 2\sigma_i, \quad LB = \hat{y}_i - 2\sigma_i. \tag{6}$$

2.2.2. Quantile Gradient Boosting

Quantile methods (Koenker & Hallock, 2001) can be seen as an extension of classical least squares model estimation for the conditional mean function to the estimation of models for the conditional median function and the full range of other conditional quantile functions. The quantile function does not require a specification of variance changes and can thus model heterogeneous variation in the objective loss distribution (Chan, 2021). Moreover, this approach avoids the distributional assumption, that is, it does not assume a Gaussian error distribution (unlike most traditional methods) and can be used when the error distribution is non-Gaussian (Chan, 2021). Quantiles can be estimated by multiplying different quantile values β by positive and negative residuals in the loss function to obtain the quantile loss as

$$\begin{aligned}\mathcal{L}(e_i|\beta) \quad &= \begin{cases}\beta \cdot e_i & \text{if} \quad e_i \geq 0, \\ (\beta - 1)\cdot e_i & \text{if} \quad e_i < 0\end{cases} \qquad e_i = y_i - \hat{y}_i \\ Cost(\mathbf{e}|\beta) &= \tfrac{1}{N}\sum_{i=1}^{N}\mathcal{L}(e_i|\beta).\end{aligned} \tag{7}$$

The quantile values of β are set to 0.025 and 0.975 for estimating the lower and upper confidence bounds, respectively, to obtain a CI of 95%. The mean quantile $\beta = 0.50$ provides the median VTEC forecast. To estimate other CI levels of 90% and 99%, the quantile values must be changed to $\beta = \{0.05, 0.95\}$ and $\beta = \{0.005, 0.995\}$, respectively.

Quantile loss has been shown to model data uncertainty in neural networks (Amell et al., 2022; Tagasovska & Lopez-Paz, 2019). In this study, we applied quantile loss with a Gradient Boosting tree. The Gradient Boosting algorithm and its implementation for VTEC forecast are explained in Natras et al. (2022a). We chose Gradient Boosting because it is fast (Natras et al., 2022a), performs well on structured input data even for relatively small data sets (Duan et al., 2020), and has proven to be a powerful method in many data science competitions (Chen & Guestrin, 2016). Moreover, Vasseur and Aznarte (2021) compared the performance of 10 ML algorithms with quantile loss for predicting NO_2 pollution and found that Gradient Boosting outperformed the other models with better results for all metrics examined.

2.2.3. Bayesian Neural Network

The Bayesian neural network (BNN) represents a modification of an ANN in which the deterministic network parameters or weights are replaced by probability distributions of those weights (Abdar et al., 2021; Blundell et al., 2015; Kendall & Gal, 2017); for more details on the architecture and computation of an ANN, see Natras, et al. (2023a). The probability distributions are used to model the uncertainty in the weights and consequently can be used to estimate the uncertainty due to the model parameter uncertainty based on Bayes' theorem. The posterior parameters θ to be trained are the mean μ and standard deviation σ of the posterior weight distribution. They can be learned by variational Bayesian inference during the training process, facilitated by a standard neural network backpropagation technique during the training process (Blundell et al., 2015). That technique is called Bayes by Backprop and is implemented in this study.

Given a training data set $D = (x_i, y_i)$ with $i = 1, 2, \ldots N$, the likelihood function $p(D|w)$ can be constructed, which is a function of the weights w. Maximizing the likelihood function yields the maximum likelihood estimate of w. The usual optimization objective in ML training is to minimize the NLL. Multiplying the likelihood by a prior distribution $p(w)$ is proportional to the posterior distribution $p(w|D) \propto p(D|w)p(w)$ according to Bayes' theorem (Koch, 2018). An analytical solution for the posterior $p(w|D)$ in neural networks is not feasible. We can approximate the true posterior with a variational distribution $q(w|\theta)$ of the function whose parameters we want to estimate. This can be done by minimizing the Kullback-Leibler (KL) divergence between $q(w|\theta)$ and the true posterior $p(w|D)$.

KL divergence measures how close the variational probability distribution of the weights $q(w|\theta)$ is to the posterior probability distribution of the weights $p(w|D)$. It is also called relative entropy in probability and information theory (Murphy, 2012). Normally, the reverse KL divergence is used

$$\begin{aligned} KL(q(w|\theta)\|p(w|D)) &= q(w|\theta)\cdot\log\frac{q(w|\theta)}{p(w|D)} \\ &= -\log p(D|w) + KL[q(w|\theta)\|p(w)]. \end{aligned} \tag{8}$$

The idea behind variational inference is to choose an approximation $q(w|\theta)$ to the distribution and then try to make this approximation as close as possible to the true posterior $p(w|D)$. This reduces variational inference to an optimization problem, and from Equation 8, the objective cost function can be defined as follows

$$Cost = \frac{1}{N}\sum_{i=1}^{N}(-\log p(D|w) + KL[q(w|\theta)\|p(w)]), \tag{9}$$

which can be split into two parts: the left term of the loss function on the right side corresponds to the NLL, and the right term is the KL divergence between the variational distribution $q(w|\theta)$ and the prior $p(w)$, which can also be seen as the regularization term.

The prior weight distribution is a Gaussian distribution with a mean $\mu = 0$ and a diagonal covariance with a standard deviation $\sigma = 1$. A sample of the weights w is obtained by randomly sampling ϵ from $\mathcal{N}(0, 1)$, then scaling it by a standard deviation σ, and shifting it by a mean μ as

$$w = \mu + \sigma\cdot\epsilon. \tag{10}$$

For numerical stability, the network is parametrized with ρ instead of σ. ρ is transformed with the so-called softplus activation function as

$$\sigma = \log(1 + \exp(\rho)) \tag{11}$$

to ensure that σ is always non-negative (Blundell et al., 2015). The algorithm proceeds by sampling from the variational posterior distribution, computing a forward pass through a network, and then backpropagating through the model parameters to update them. The gradients are calculated with respect to the mean and the standard deviation to update the previous distribution parameters using the stochastic gradient descent optimization algorithm (Bottou, 2012). The parameters are updated stepwise, controlled by the learning rate, along a preferred direction, which is a function of the previous gradient.

The Gaussian likelihood is assumed in this study, parameterized by the mean and standard deviation as

$$p(D|w) = l(y|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2}. \tag{12}$$

The NLL loss is defined as

$$\begin{aligned} \mathcal{L} &= -\log l(y_i|\mu,\sigma) \\ &= \frac{1}{2}\left[\log\left(\sigma^2\right) + \frac{(y_i-\mu)^2}{\sigma^2} + \log(2\pi)\right] \\ &= \frac{1}{2}\left[\log\left(\sigma^2\right) + \frac{(y_i-\mu)^2}{\sigma^2} + C\right], \end{aligned} \tag{13}$$

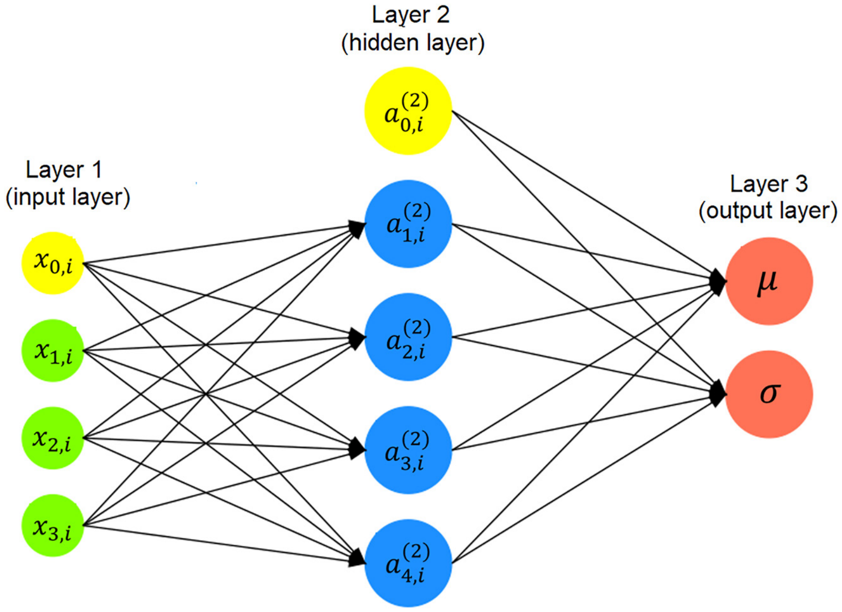


Figure 2. Simplified representation of the Bayesian neural network (BNN) architecture with probabilistic implementation to output the probability distribution parameters: μ and σ. The yellow circles represent the bias neurons. The green circles are the input neurons (simplified here to 3, but 14 in the BNN + D VTEC model), the blue circles are hidden neurons (4 here, but 32 in the BNN + D VTEC model), and the orange circles are the output neurons.

where y_i is the observed value or GT, μ is the predicted mean, and σ is the standard deviation. C is a constant equal to log(2π), which can be neglected. The loss function in Equation 13 is also known as the negative logarithm of predictive density (Licata & Mehta, 2022).

For BNN, we assume a fixed data noise, as usual. Then the loss in Equation 13 corresponds to the squared error loss, and the left term on the right side in Equation 9 becomes the standard MSE cost, similar to Equation 2. This Bayesian approach to an ANN aims to capture the model parameter uncertainty due to limited training data. Each time the BNN model is run with the same input variables, a new set of parameters is sampled from the distribution, and a result is produced. In this study, the VTEC forecast is estimated as the mean of an ensemble of results from 100 iterations, while the 95% CI is calculated as in Equation 6.

The BNN implementation described so far is deterministic, that is, it produces a single VTEC forecast for each run, and the uncertainty is calculated from an ensemble of many iterations. The BNN can be extended to a probabilistic implementation by enabling the model to output a distribution and quantify the data uncertainty. In this case, the data noise is assumed to be data-dependent rather than fixed, and thus, it is learned as a function of the data. Therefore, the NLL from Equation 13, which accounts for the observation noise, is used in Equation 9 to compute how likely the GT values are to deviate from the estimated distribution produced by the model. The model can then provide a probability distribution as an output, that is, μ and σ, instead of a single point estimate. To provide μ and σ as output values, a custom output layer is created with two neurons, shown in Figure 2: one for mean output and one for standard deviation output. The 95% CI is computed from the predicted standard deviation according to Equation 6.

All approaches used in this study are summarized in Table 1. The 95% CI in the SE, BNN, and BNN + D approaches is approximated by multiplying the standard deviation by 2. If necessary, other CI can also be estimated. For example, multiplying the standard deviation by 1.64 gives a 90% confidence level, and by 2.58 gives a 99% confidence level. For the QGB approach, the quantile values must be adjusted accordingly to estimate 90% and 99% CI, as already mentioned in Section 2.2.2.

2.3. Models Optimization and Hyperparameters

Optimization of a ML model includes adjusting the hyperparameters to minimize the objective cost function. In this study, the hyperparameters were tuned using 20-fold time-series cross-validation (Natras et al., 2022a) and

Table 1
Approaches of Applying Different Uncertainty Quantification Methods on Different Learning Algorithms and Their Abbreviations

Approaches	Cost/Loss	Learning algorithms	Abbreviations
Ensemble modeling: Super-Ensemble	MSE	Bagging and Boosting	SE
Quantile Gradient Boosting	Quantile	Gradient Boosting	QGB
Bayesian inference	MSE + KL	Bayesian Neural Network	BNN
Bayesian inference	NLL + KL	Bayesian Neural Network	BNN + D

grid search within the hyperparameter range, see Table S1 in Supporting Information S1. Table 2 summarizes the selected values for the hyperparameters.

3. Results

The analysis is performed for the year 2017 (1 January–31 December 2017), for a period with space weather events (6–10 September 2017), and for a quiet period concerning solar and geomagnetic activity (25–29 April 2017). Figures 3 and 4 show the 1-day VTEC forecast in orange with a 95% CI in green using the SE and BNN approaches, and the QGB and BNN + D approaches, respectively, for the quiet period (left) and the storm period (right) in 2017. The results of mean/median VTEC from different ML-based UQ VTEC models are summarized in Table 3.

As the baseline models, we use the frozen ionosphere and the Multi-Layer Perceptron (MLP) model. For the frozen ionosphere, we define that VTEC(i + 24) equals VTEC(i), that is, we assume the state of the frozen ionosphere with respect to the previous day. This assumption is consistent with the prevailing diurnal VTEC variability, where the next day's VTEC should not be significantly different from the previous day's VTEC under quiet conditions. The MLP model is the classical type of neural network and represents a fully connected ANN consisting of one or more hidden layers of neurons. MLP is the most commonly used ML method for VTEC modeling and forecasting (e.g., Ferreira et al., 2017; Orus Perez, 2019; Özkan, 2022). The International Reference Ionosphere (IRI) 2016 is used as a third baseline, where VTEC was extracted at the height of 450 km, and the upper height for TEC integration was set at 20,000 km. The IRI analysis was conducted for two study periods: one in April and a second in September 2017, and the detailed analysis is shown in Figure 5.

The period 6–10 September 2017, represents the most intense solar activity period with the strongest solar flare of X9.3 class, which peaked at 12:02 UT on 6 September. Earthward-directed CMEs were emitted from the Sun on 4 and 6 September (Imtiaz et al., 2020). The first CME arrived at about 23:43 UT on 6 September and caused moderate geomagnetic conditions on 7 September, while the second CME from the X9.3 solar flare triggered a sudden storm commencement at 23 UT on 7 September. This resulted in severe geomagnetic storms on 8 September with a maximum Kp = 8 (Figure 4, bottom). The main phase of the storm was characterized by the two pronounced minima of the Dst index at around 1 and 14 UT on 8 September. Thereafter, the recovery phase began and lasted for about 3 days, that is, until 11 September.

The SE and BNN methods provide narrow CI ranging from less than 1 TECU to 2 TECU from the mean VTEC forecast, as shown in Figure 3. However, about 50% of GT VTEC values in 2017 are outside their 95% CI in Table 3. This indicates that the approaches that capture only the model uncertainties produce over-confident VTEC CI. During the disturbed space weather period in September 2017, the GT is outside the CI by up to 4

Table 2
Selected Hyperparameters (hl Stands for Hidden Layer and lr for Learning Rate)

Model	Hyper-parameters
SE	see Table 2 in Natras et al. (2022a)
QGB	tree depth = {3, 4, 5}, number of trees = {50, 100, 150}, lr = 0.1
BNN	batch = 500, epoch = {500, 1,000}, lr = 0.001, 1 hl with 32 neurons
BNN + D	batch = 500, epoch = 2,000, lr = {0.01, 0.1}, 1 hl with 32 neurons

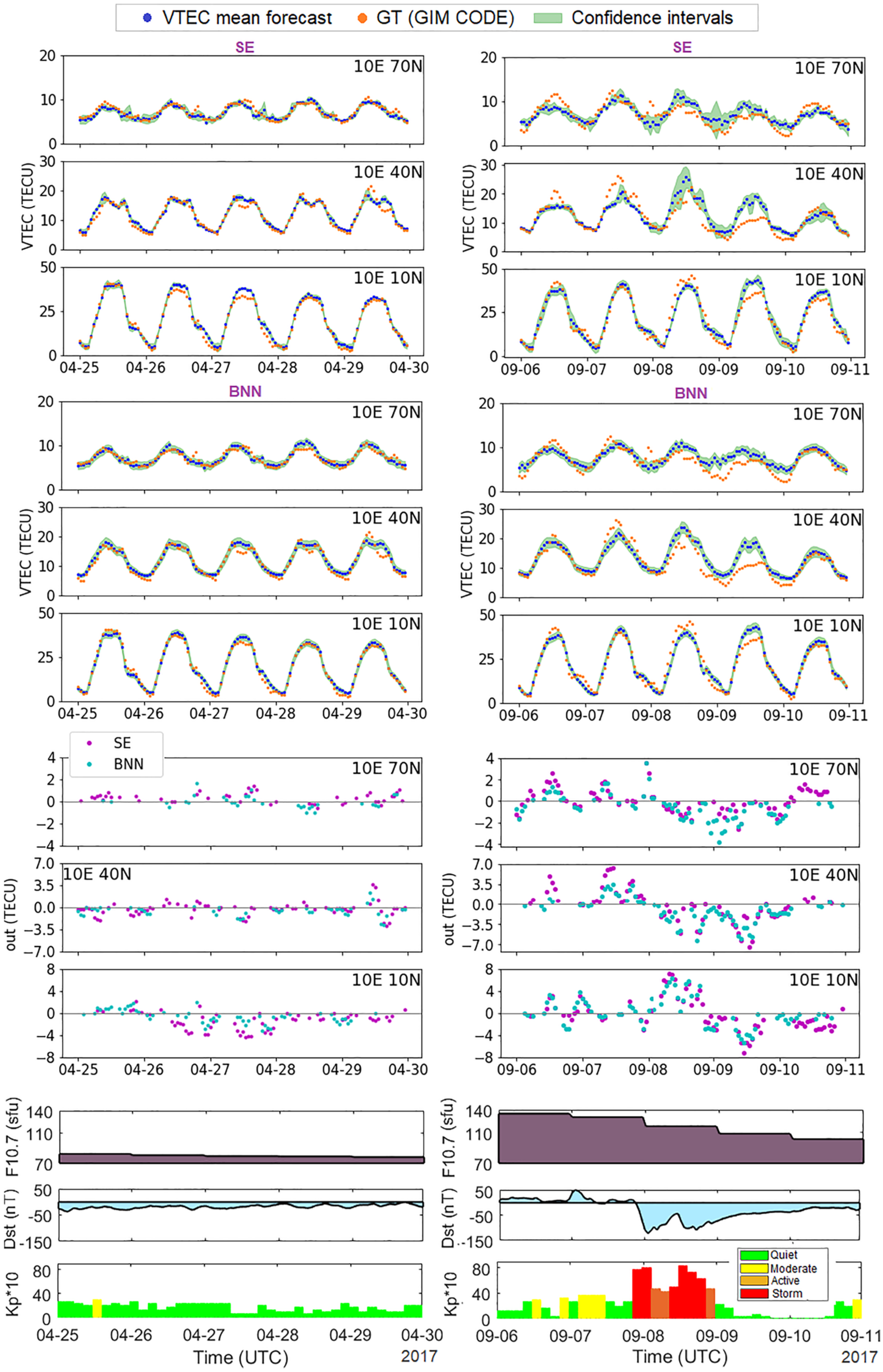


Figure 3. Mean Vertical Total Electron Content (VTEC) forecast and 95% confidence intervals (CI): SE (first panel) and Bayesian neural network (second panel) for selected grid points. Third panel: ground truth (GT) VTEC outside CI (positive value: the amount by which GT is higher than upper CI, negative value: the amount by which GT is lower than lower CI). Fourth panel: indices of F10.7, Dst, and Kp · 10 ($Kp < 3$, $3 \leq Kp < 4$, $4 \leq Kp < 5$, and $Kp \geq 5$ denote quiet, moderate, active and storm conditions, respectively). Left: 25–29 April 2017, right: 6–10 September 2017.

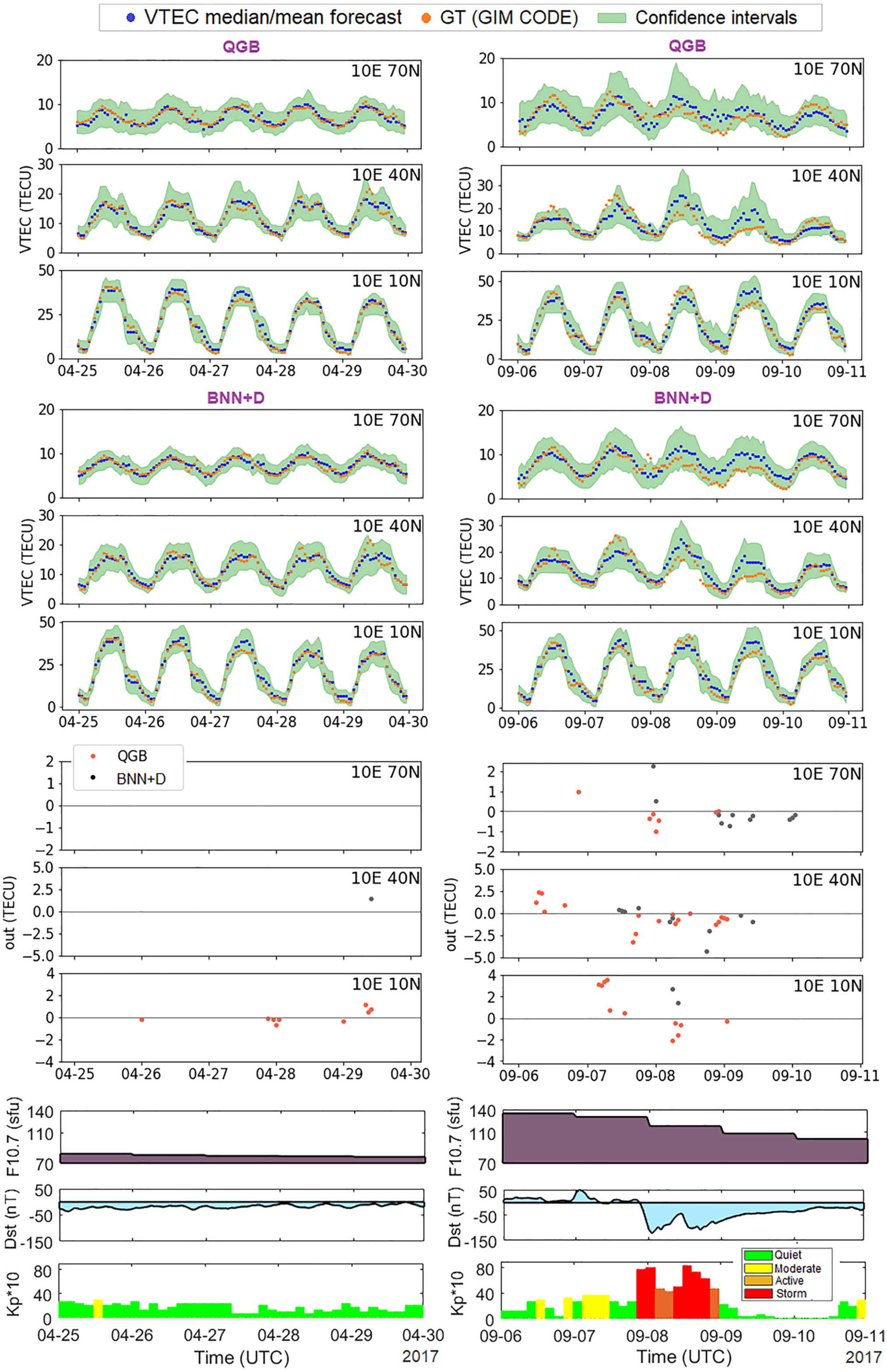


Figure 4. Median/mean Vertical Total Electron Content (VTEC) forecast and 95% confidence intervals (CI): Quantile Gradient Boosting (first panel) and BNN + D (second panel) for three selected grid points. Third panel: ground truth (GT) VTEC outside CI (positive value: amount by which GT is higher than upper CI limit, negative value: amount by which GT is lower than lower CI limit). Fourth panel: indices of F10.7, Dst, and Kp · 10. Left: 25–29 April 2017; right: 6–10 September 2017.

Table 3
Statistics on the Test Data Set for 1-Day Probabilistic Vertical Total Electron Content Forecast

	1 January–31 December 2017	6–10 September 2017	25–29 April 2017
Model	RMS, Corr., CI_{avg}, In(%)	RMS, Corr., CI_{avg}, In(%)	RMS, Corr., CI_{avg}, In(%)
VTEC:10°70°			
SE	1.03, 0.92, 0.74, 51.89	1.73, 0.71, 1.06, 33.33	0.71, 0.89, 0.62, 60.83
QGB	1.05, 0.91, 2.29, 94.63	1.73, 0.71, 3.99, 94.17	0.77, 0.88, 2.54, 100.0
BNN	1.18, 0.91, 0.78, 48.82	1.79, 0.73, 0.98, 40.0	0.73, 0.91, 0.78, 73.33
BNN + D	1.07, 0.91, 2.20, 96.75	1.90, 0.74, 3.20, 90.83	0.69, 0.90, 2.04, 100.0
Baseline MLP	1.09, 0.92, /, / ,	2.10, 0.80, /, / ,	0.77, 0.90, / , / ,
Baseline Frozen	1.18, 0.89, / , / ,	2.17, 0.58, / , / ,	0.81, 0.85, / , / ,
Baseline IRI 2016		3.39, 0.82, / , / ,	1.89, 0.92, / , / ,
VTEC:10°40°			
SE	1.83, 0.90, 0.92, 43.46	3.31, 0.80, 1.44, 41.67	1.32, 0.96, 0.66, 36.67
QGB	1.89, 0.89, 3.45, 94.17	3.35, 0.80, 4.59, 82.50	1.27, 0.96, 3.61, 100.0
BNN	1.95, 0.90, 1.20, 47.48	3.09, 0.85, 1.44, 39.17	1.40, 0.96, 1.24, 60.83
BNN + D	1.89, 0.90, 3.78, 95.11	2.94, 0.86, 4.24, 93.33	1.53, 0.94, 3.90, 99.17
Baseline MLP	1.92, 0.89, / , / ,	3.50, 0.85, / , / ,	1.48, 0.96, / , / ,
Baseline Frozen	2.22, 0.86, / , / ,	4.00, 0.72, / , / ,	1.33, 0.95, / , / ,
Baseline IRI 2016		5.63, 0.78, / , / ,	2.78, 0.95, / , / ,
VTEC:10°10°			
SE	2.08, 0.98, 1.32, 53.50	3.71, 0.96, 2.10, 39.12	2.19, 0.99, 1.22, 47.50
QGB	2.22, 0.98, 5.53, 96.21	3.98, 0.95, 6.51, 89.17	2.09, 0.99, 5.18, 95.83
BNN	2.28, 0.98, 1.66, 52.56	3.45, 0.96, 1.78, 40.00	1.90, 0.99. 1.60, 55.00
BNN + D	2.67, 0.97, 5.70, 97.38	3.63, 0.96, 7.02, 98.33	2.07, 0.99, 6.16, 100.0
Baseline MLP	2.34, 0.97, / , / ,	4.19, 0.96, / , / ,	2.16, 0.99, / , / ,
Baseline Frozen	2.40, 0.97, / , / ,	4.21, 0.94, / , / ,	2.31, 0.99, / , / ,
Baseline IRI 2016		8.41, 0.91, / , / ,	4.76, 0.94, / , / ,

Note. RMS stands for Root Mean Square, and Corr. for the correlation coefficient. RMS and Corr. are calculated between the median (QGB) or mean (SE, BNN, BNN + D) VTEC and ground truth. CI_{avg} represents the average distance of the lower and upper bounds from the forecast median (QGB) or mean (SE, BNN, BNN + D) VTEC. In(%) represents the percentage of ground truth within the 95% confidence intervals. The best results are highlighted in green. When all developed models have the same correlation coefficients, no values are highlighted, that is, all are black.

TECU for the high latitude point and 8 TECU for the low latitude point, while it is half lower during the quiet period in April 2017. The largest absolute GT VTEC values outside the CI occur during the strongest solar flare on 6 September, during moderate geomagnetic conditions on 7 September, during geomagnetic storms on 8 September, and at the beginning of the recovery period on 9 September. These results show that the forecast CI of the SE and BNN approaches exclude most of the sudden and intense VTEC variability during space weather events. However, the mean VTEC from the SE approach mostly achieves the lowest RMS for the entire test year. On the other hand, the QGB and BNN + D approaches provide 3 to 4 times wider CI, as shown in Figure 4, that contain more than 95% of GT in 2017 and even 100% during the quiet period. The largest absolute values of GT outside the CI are on 7 September, as well as during the first and second Dst minima, that is, the maximum intensity of the geomagnetic storm. The magnitude of the GT outside the CI is 4–5 TECU during the September 2017 space weather events and less than 2 TECU with much lower frequency during the quiet period. For instance, in the BNN + D approach, there is only one GT value outside the CI during the quiet period in April 2017. The median VTEC from the QGB approach mostly has a slightly lower correlation with GT than the other approaches, while the mean VTEC of the BNN + D approach has the highest correlation during intense space weather in Table 3. For the SE, BNN, and BNN + D approaches, the average width of the 90% CI from the VTEC mean

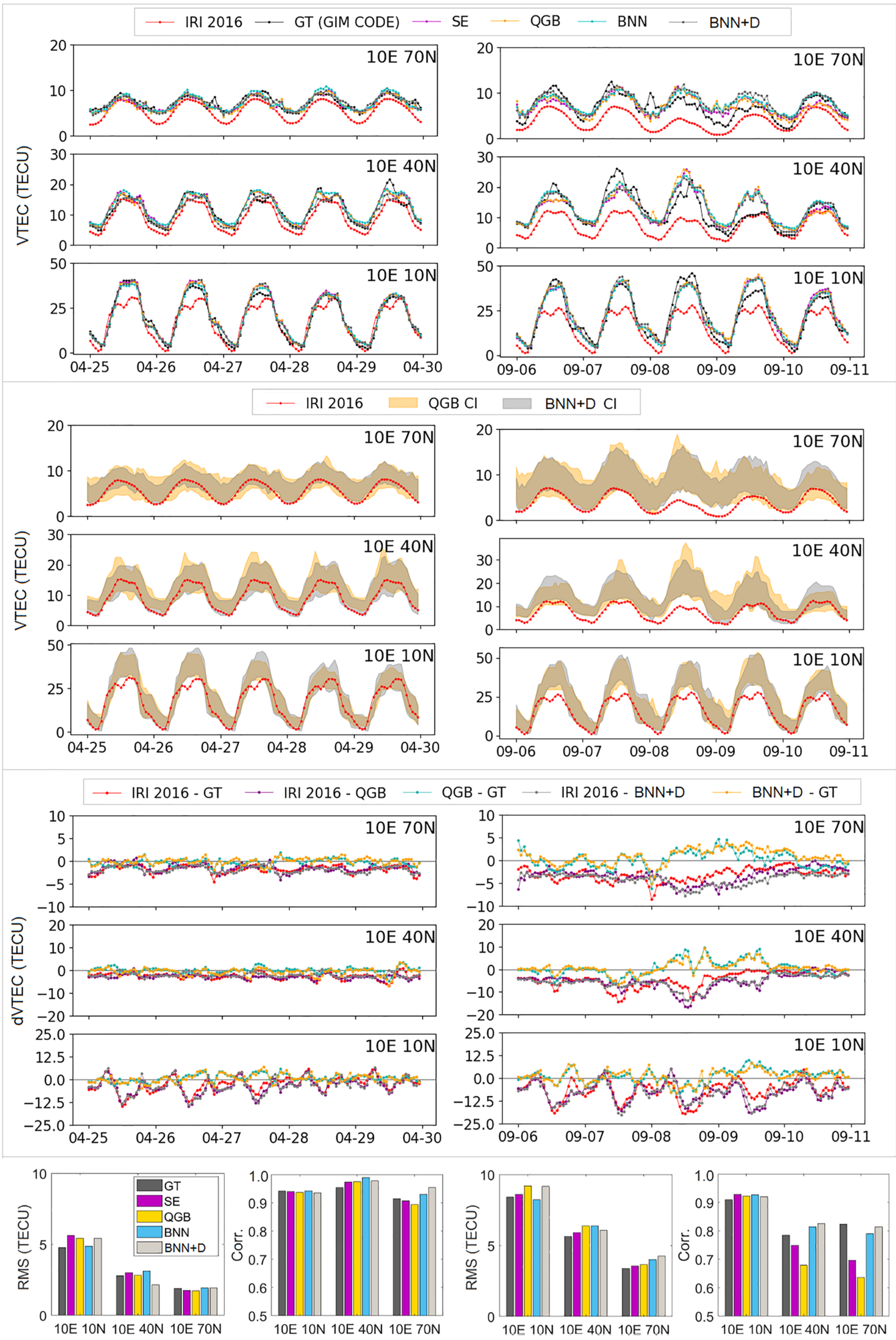

IRI 2016
GT (GIM CODE)
SE
QGB
BNN
BNN+D
VTEC (TECU)
10E 70N
10E 40N
10E 10N
04-25
04-26
04-27
04-28
04-29
04-30
09-06
09-07
09-08
09-09
09-10
09-11
QGB CI
BNN+D CI
IRI 2016 - GT
IRI 2016 - QGB
QGB - GT
IRI 2016 - BNN+D
BNN+D - GT
dVTEC (TECU)
GT
RMS (TECU)
Corr.
10E 10N
10E 40N
10E 70N

Figure 5.

can be estimated by multiplying the value for CI_{avg} in Table 3 by 0.8, and for the 99% CI by multiplying it by 1.3. Accordingly, the 90% CI will be 20% narrower, and the 99% CI will be 30% wider than the 95% CI.

The mean/median VTEC of the studied approaches mostly outperforms the baseline frozen ionosphere, with the most significant improvement from around 20%–30% during severe space weather (6–10 September) for all three investigated VTEC grid points. Their differences are not as significant for the quiet period in April 2017, that is, from 0.1 to 0.3 RMS. The mean/median VTEC values of the developed models have lower RMS than the baseline MLP model for most of the study cases. In particular, for the storm period, they improve the RMS by about 0.50 TECU or 15%, while the MLP model still maintains high correlations with GT. The RMS of the IRI 2016 is about twice as large as the RMS for the mean/median VTEC of the developed ML-based UQ VTEC models for both quiet and storm study cases.

The detailed analysis for the IRI 2016 is presented in Figure 5. In April 2017, the IRI is mainly within the CI of QGB or BNN + D, while in September 2017, it is sometimes at the edge and sometimes outside these intervals. The differences between the IRI VTEC values and the GT VTEC, and between IRI and the median VTEC of QGB and mean VTEC of BNN + D models are similar. They are mostly more prominent than the differences between the median/mean VTEC of QGB/BNN + D and the GT data. The enormous IRI differences exist for the low-latitude VTEC position, where VTEC from IRI is underestimated by up to more than 20 TECU. Vertical Total Electron Content from IRI agrees much better with the mean/median VTEC values from the ML-based UQ VTEC models and the GT data in April 2017 than September 2017. Consequently, the RMS values between IRI and GT and ML-based UQ VTEC models are smaller in April 2017, that is, when the ionosphere is quiet, while they are twice as significant when the ionosphere is disturbed in September 2017. This space weather effect is also reflected in the correlation coefficients between IRI and GT, as well as ML-based UQ VTEC models.

The upper and LB of the 95% CI estimated using different approaches are visualized in Figure 6 without the VTEC mean/median, that is, they are adjusted around $y = 0$. In the case of the quiet period (Figure 6, left), the QGB and BNN + D CI are similar in size for the mid-latitude grid point, while the QGB CI is wider for the high-latitude point, and the BNN + D confidence upper bound is slightly larger for the low-latitude point. The SE and BNN CI are of a similar order of magnitude. The main difference is that the BNN CI is smoother and more constant over the study period, while the SE CI is variable. During the storm case (Figure 6, right), the CI become wider as the changes in the GMF occur. The SE and QGB CI are about two times wider and more variable on the day of the geomagnetic storm maximum (8 September) and the following day of the recovery phase (9 September), while the BNN and BNN + D CI slightly increase. The largest upper confidence bound for high- and mid-latitude points in this period comes from the QGB approach, while for the low-latitude point, the QGB and BNN + D upper confidence bounds are similar in size. For both study cases, it can be seen that the QGB and SE CI are more variable and have frequent peaks, while for BNN and BNN + D, they are smoother and more consistent from day-to-day. The CI of all approaches are wider around local noon for the mid-latitude point, while for the low-latitude point, an additional increase in the upper bound is visible after sunset and lasts for several hours. Post-sunset increase in the QGB and BNN + D upper low-latitude VTEC bounds is visible for 6 to 9 September with F10.7 > 110 sfu, and from 25 to 29 April during a period of low geomagnetic activity, with both periods close to equinox. The effect is more pronounced in QGB. The post-sunset VTEC enhancement has been detected at low latitudes within the equatorial ionization anomaly using actual VTEC observations in Dashora et al. (2019), Kutiev et al. (2007), Kumar et al. (2022), J. Liu et al. (2020). It develops 2–3 hr after sunset, with a peak around 7:00–8:00 p.m. local time (Kumar et al., 2022; Kutiev et al., 2007), and occurs during prolonged periods of low geomagnetic activity (Kutiev et al., 2007), as well as during geomagnetic storms (Dashora et al., 2019), with stronger intensity around equinoxes (J. Liu et al., 2020), and when the F10.7 solar flux exceeds 110 sfu (Kumar et al., 2022). Therefore, the patterns of increase in the upper low-latitude VTEC bounds after sunset are consistent with observations of the low-latitude VTEC post-sunset enhancement reported in previous studies.

The results of the analysis in Figure 6 show that the CI exhibit variations depending on daytime/nighttime, solar irradiance, space weather conditions, that is, geomagnetic storms, and the post-sunset ionosphere enhancement at

Figure 5. First panel: Vertical Total Electron Content (VTEC) from International Reference Ionosphere (IRI) 2016, ground truth (GT) data, and the mean/median VTEC values from uncertainty quantification ML-based VTEC models. Second panel: IRI 2016 and confidence intervals of Quantile Gradient Boosting (QGB) and BNN + D. Third panel: VTEC differences between IRI 2016 and GT, the QGB median VTEC, and the BNN + D mean VTEC, as well as the differences between QGB/BNN + D and GT. Fourth panel: RMS and correlation of GT and ML-based mean/median VTEC values with respect to IRI 2016. Left: 25–29 April 2017, right: 6–10 September 2017.

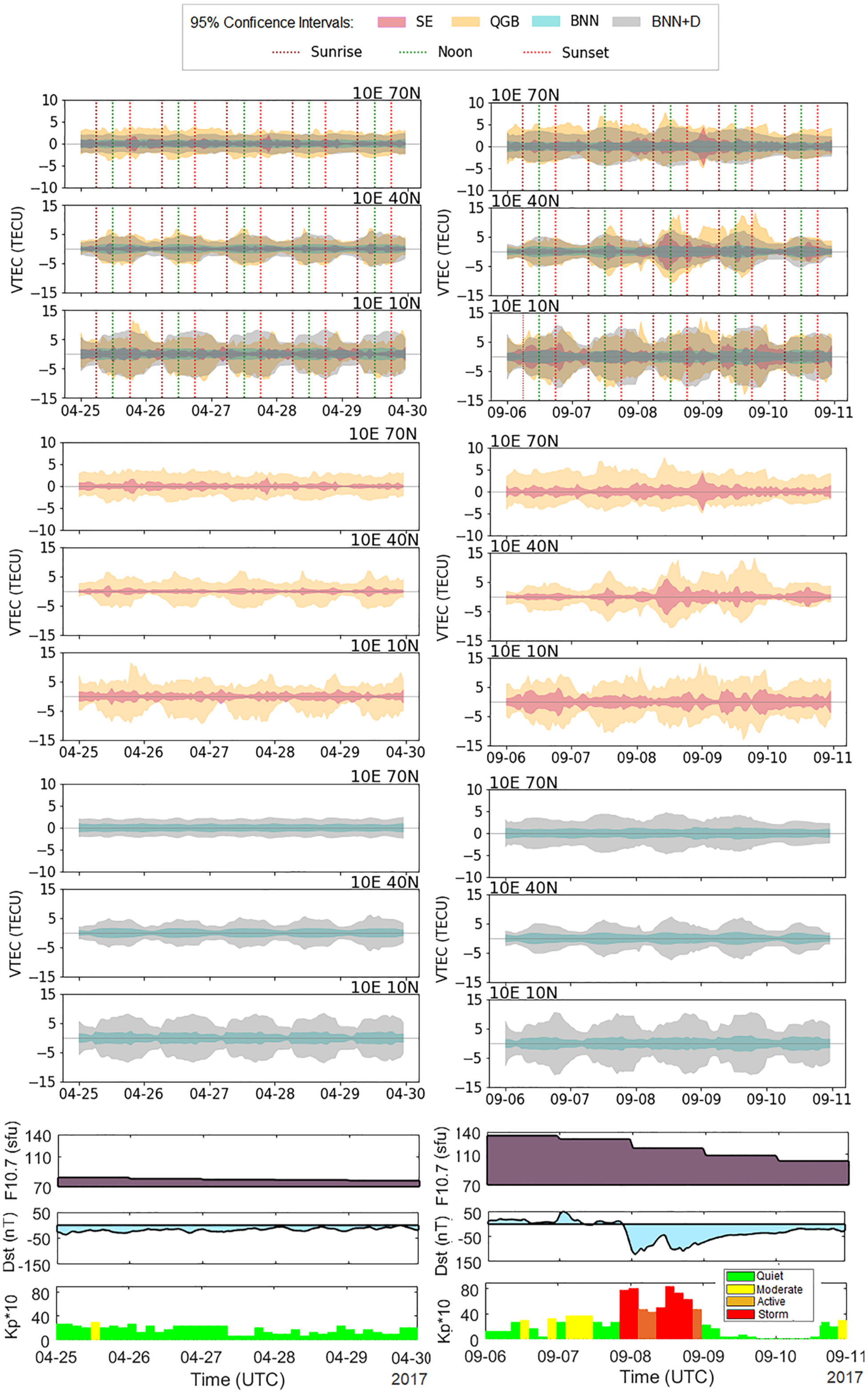


Figure 6. 95% confidence interval of all developed uncertainty quantification ML-based Vertical Total Electron Content models (first panel), SE and Quantile Gradient Boosting (second panel), Bayesian neural network (BNN) and BNN + D (third panel) for three selected grid points. Fourth panel: indices of F10.7, Dst, and Kp · 10. Left: 25–29 April 2017; right: 6–10 September 2017.

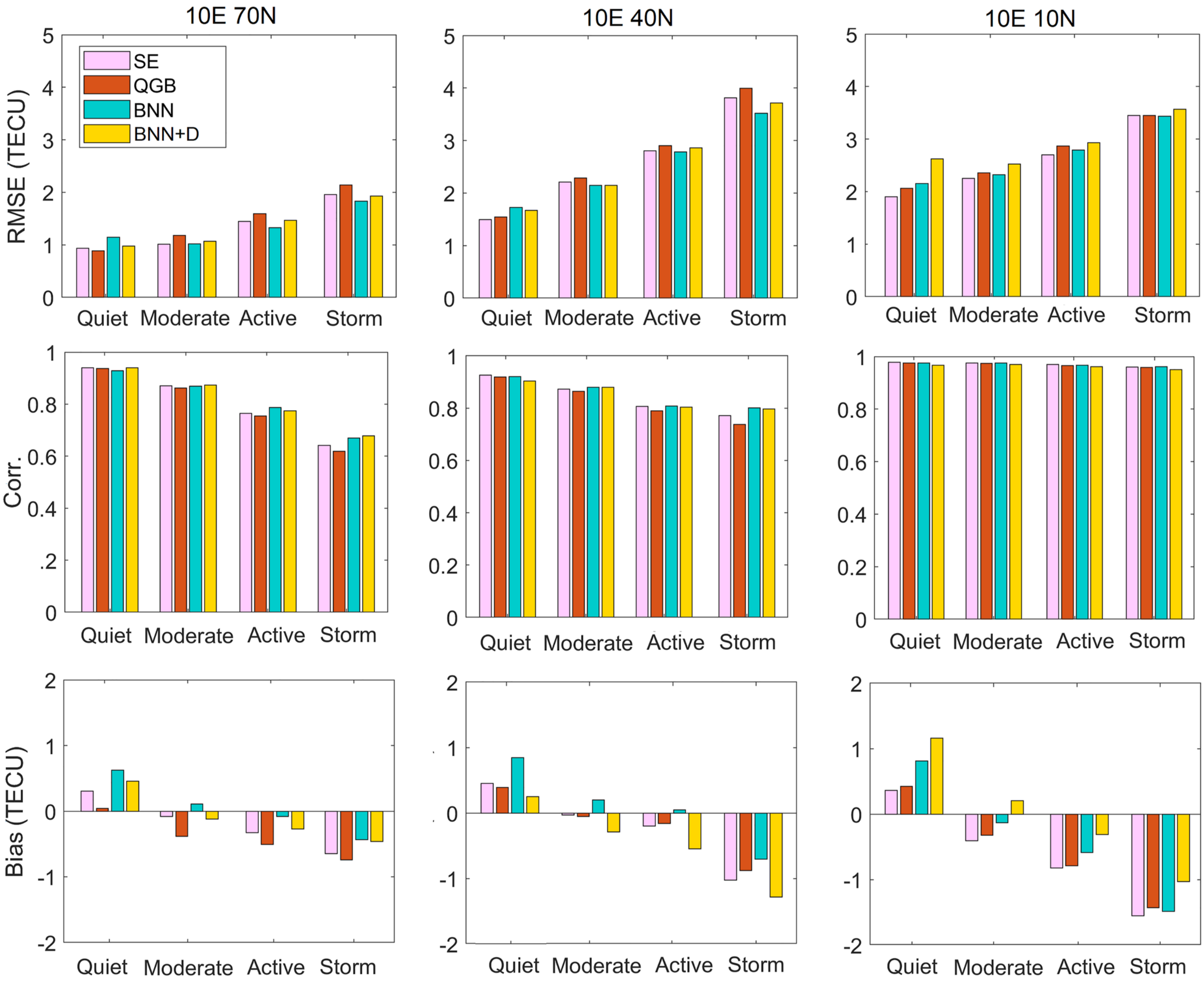


Figure 7. Statistics of mean/median Vertical Total Electron Content forecast from the developed models to ground truth versus Kp index for 2017. Top: RMS, mid: correlation coefficients (Corr.), bottom: bias. The labels quiet, moderate, active and storm correspond to $Kp < 3$, $3 \leq Kp < 4$, $4 \leq Kp < 5$, and $Kp \geq 5$, respectively.

low latitudes. Therefore, they are narrower during the night, wider around local noon for the mid-latitude point, wider and more variable with the change of Kp and Dst indices, and wider after sunset for the low latitude point under certain conditions mentioned above.

Further analysis is performed regarding geomagnetic activity in Figure 7. The forecast mean/median VTEC accuracy in terms of RMS and correlation coefficients decreases with increasing geomagnetic activity. The biases are largest and negative during storms, suggesting that the models underestimate the mean/median VTEC for storms. Due to the complex, distinct VTEC irregularities during different geomagnetic storms, the lack of VTEC samples covering different geomagnetic storms under different dependent factors such as storm intensity, season, magnetic local time, storm onset time, magnetic latitude and solar cycle phase (Greer et al., 2017; J. Liu et al., 2010; Vijaya Lekshmi et al., 2011), as well as the overall presence of storm events in the data set, resulting in a high imbalance compared to the quiet condition samples (see Figure 7 in Natras et al. (2022a)), the developed ML-based UQ VTEC models have lower accuracy in forecasting the mean/median VTEC during storms. To evaluate the full performance of the ML-based UQ VTEC models and achieve realistic accuracy representation, we need to consider the full probabilistic prediction, that is, the CI.

Figure 8 represents boxplots of the GT VTEC outside the forecast 95% CI concerning different geomagnetic activity levels. The analysis is performed only for data samples where the GT falls outside the forecast CI, which

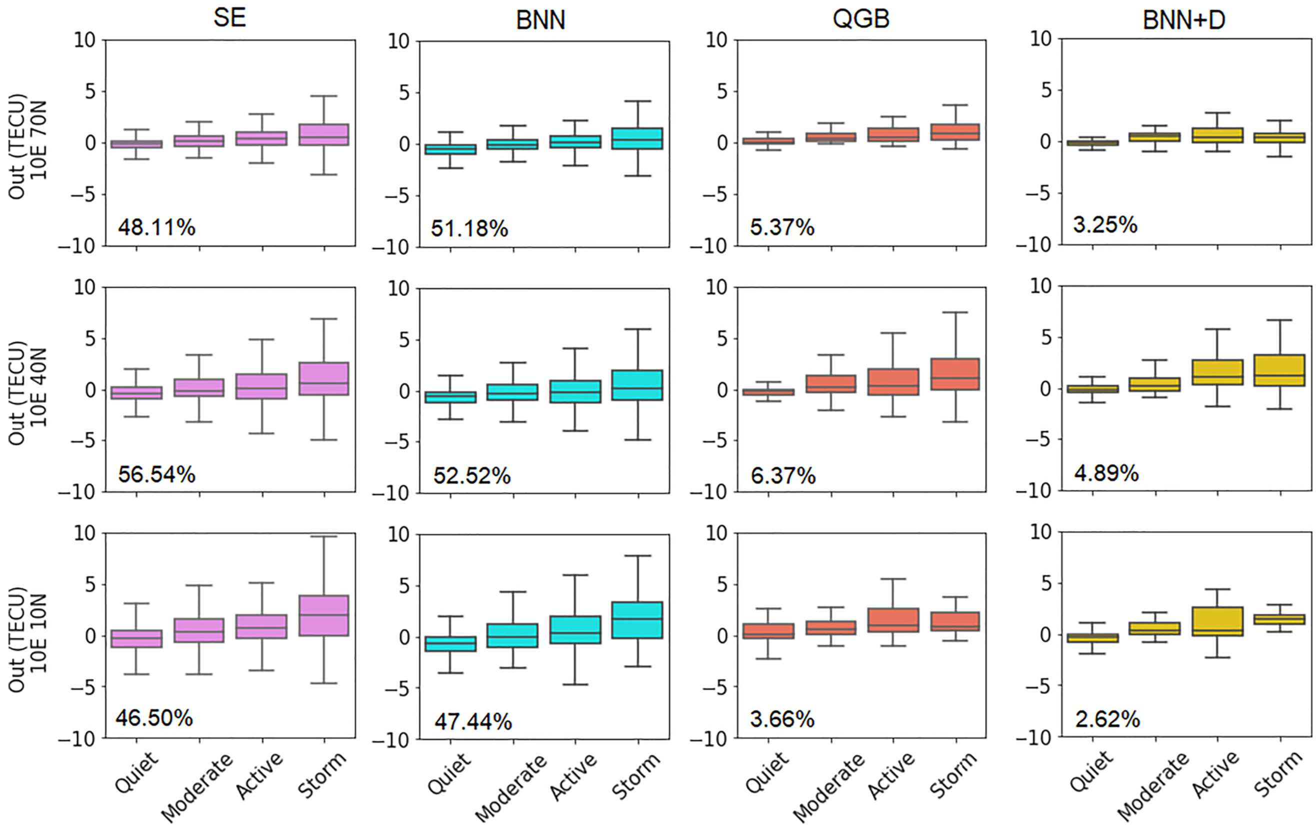


Figure 8. Boxplots for ground truth (GT) outside the forecast confidence intervals (CI) versus different geomagnetic conditions (quiet, moderate, active, and storm correspond to $Kp < 3$, $3 \leq Kp < 4$, $4 \leq Kp < 5$, and $Kp \geq 5$, respectively). In each graph, the percentage of outliers in 2017 is given (bottom left). It corresponds to the data samples for which GT falls outside the forecast CI. Positive value: the amount by which GT is higher than the upper CI bound; negative value: the amount by which GT is lower than the lower CI bound. The boxes (the interquartile range) represent the range between the 25th (first quartile) and 75th percentile (third quartile), that is, the middle 50%. The gray line in each box corresponds to the median. The gray lines outside the boxes represent the lower 25% and the upper 25% of the values, with the ends of the line representing the minimum and maximum values. First row: 10°70°, mid row: 10°40°, third column: 10°10°. From left to right: SE, Quantile Gradient Boosting, Bayesian neural network (BNN), BNN + D.

we can refer to here as outliers from the CI. Most outliers are between 0 and 1 TECU outside of the CI. There is a clear tendency for the interquartile range and the maximum absolute values of the outliers to increase with increasing geomagnetic activity in SE and BNN models. The interquartile range and the maximum and minimum outliers also tend to be the largest in these models. In contrast, the BNN + D and QGB models have the lowest percentage of outliers: 3%–5% and 4%–6%, respectively. Considering that the CI are set at 95%, outliers up to 5% from the CI indicate reliable performance. According to these results, both BNN + D and QGB approaches achieve the target of 95% confidence. Moreover, the amount of outliers from the CI for these two methods is less affected by geomagnetic conditions.

The relative importance of input features for probabilistic VTEC forecasting by the QGB model (Figure 9) is estimated for the upper confidence bound (top), median VTEC (middle), and lower confidence bound (bottom) using the methodology presented in Text S1 in Supporting Information S1. For the median VTEC, the most important input feature is the lagged VTEC at time step t_i for forecasting VTEC at time step t_{i+24h}. This is due to the prevailing diurnal VTEC variations, where day-to-day VTEC usually does not change much during quiet conditions. On the other hand, other input features have much greater importance in estimating the lower and upper limits, such as the AE, Kp, Dst, and SW indices. Here, the objective function minimizes the positive and negative residuals between the GT and the model results for the upper and LB, respectively; see Equation 7. These residuals are more strongly influenced by solar and geomagnetic activity than the median VTEC. Thus, the lagged VTEC contributes 20%–50% less to the confidence bounds estimate than to the median VTEC estimate, while the space weather input features increase their contribution. These results suggest that the CI are determined by the space weather features in addition to the VTEC-related features.

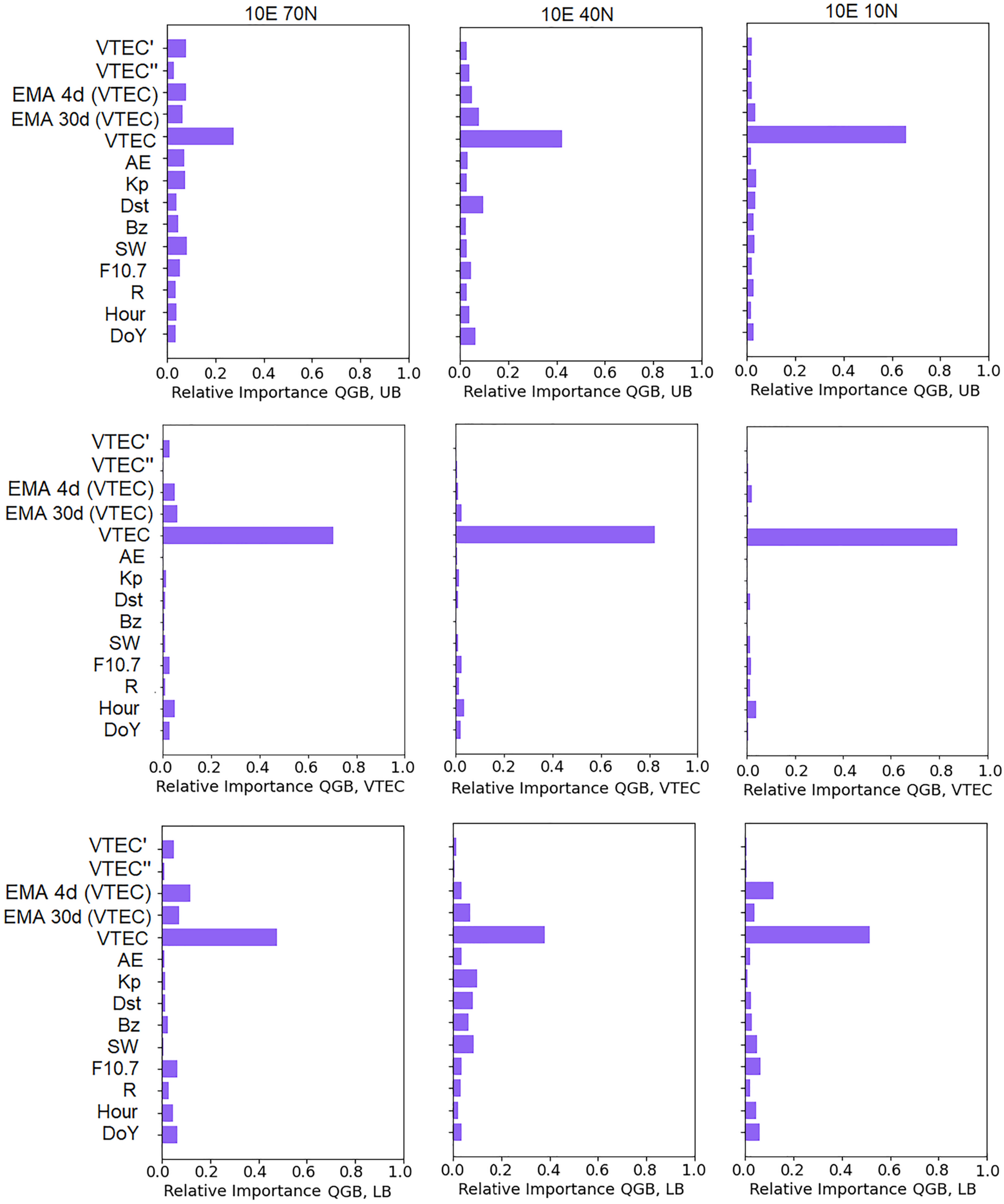


Figure 9. The relative importance of input features for the 1-day Quantile Gradient Boosting Vertical Total Electron Content (VTEC) forecast, consisting of upper bound (top), median VTEC (mid), and lower bound (bottom). VTEC' and VTEC" represent the first and second derivatives, respectively; exponential moving average (EMA) 4d (VTEC) and EMA 30d (VTEC) represent EMAs of VTEC over 4 and 30 days, respectively. The input features refer to time step i, while the output or forecast is the VTEC at time step i + 24, that is, $y_i = \text{VTEC}_{(i+24)}$. Left column: 10° 70°, mid column: 10° 40°, right column: 10° 10°.

Table 4
Computational Cost for a Single Training Run (Mid) and a Test Run (Left) for a Single Vertical Total Electron Content Grid Point Using the NVIDIA Tesla P100 GPU With 16 GB Memory

Model	Training	Testing
SE	1,275 s (~20 min)	1.35 s
QGB	50–80 s (~1 min)	<0.1 s
BNN	1,917–3,648 s (~30–60 min)	1.37 s
BNN + D	5,900 s (~100 min)	1.40 s

As for the computational complexity analysis in Table 4, the two BNN-based approaches are the most computationally intensive. A single training iteration with two years of data takes about 1 hr or more on the NVIDIA Tesla P100 GPU computing processor with 16 GB, which can be considered a disadvantage of the BNN method. The most computationally efficient model is QGB, which takes only 1 min for a training iteration with 2 years of data. When the models are trained and optimized, the execution is fast and takes 1 s for 1 year of data.

4. Conclusion

This work is the first to thoroughly examine probabilistic VTEC forecasting using ML techniques and quantifying uncertainties. In addition to forecasting a single VTEC value, the models estimate 95% CI to provide information on how confident and reliable results are by considering the uncertainties in the model parameters and/or data. In summary, we have implemented and analyzed several approaches for 1-day UQ ML-based VTEC forecasting, including:

- SE of multiple models trained with different tree-based learning algorithms and data sets to estimate uncertainties as ensemble spread,
- QGB, in which probabilistic output is estimated by minimizing quantile loss, with quantiles set at 0.025 and 0.975, for the lower and upper confidence bound, respectively, and 0.50 for median VTEC to capture the data uncertainties,
- Bayesian Neural Network (BNN), where the probability distributions of the parameters are learned to estimate the model uncertainty,
- BNN including data uncertainty (BNN + D) to capture the data uncertainty.

The findings can be summarized as follows:

1. The SE and BNN approaches provide the lowest uncertainties and, thus, overconfident results. In reality, the GT VTEC in 2017 is outside the forecast CI about 50% of the time.
2. The approaches that capture data uncertainties, QGB and BNN + D, provide wider CI that contain GT around 95% of the time and are, therefore, more realistic and reliable.
3. As for the forecasting of the mean/median VTEC, the SE approach often yields the lowest RMS value, demonstrating the power of an ensemble to improve the accuracy of the deterministic estimate. On the other hand, BNN tends to provide the highest correlations to GT, especially during the storm.
4. The relative importance of the input features shows that the CI for the QGB model are determined by space weather indices in addition to VTEC-related input features, with lagged VTEC dominating.
5. CI, especially of QGB, exhibit variations depending on the daytime/nightime, solar irradiance, geomagnetic activity, and post-sunset low-latitude ionosphere enhancement.
6. The most computationally intensive method is BNN + D, while QGB is the fastest.
7. The data uncertainties are at least three times larger than the model parameter uncertainties.

The advantages and disadvantages of each investigated UQ method for VTEC forecasting are outlined in Table 5.

Based on these findings, the probabilistic VTEC forecasting that only considers the model parameter uncertainties are insufficient. An ML-based model trained with different learning algorithms using the same/similar data sets performs similarly because it learns an approximation function from similar data, resulting in smaller discrepancies between the solutions of different ML-based models in an ensemble. The ensemble approach for UQ could be improved by training the base models on different subsets of data covering different study cases, which would increase diversity and randomness among ensemble members and may better describe uncertainties. Probabilistic VTEC modeling and forecasting, which accounts for both model parameters and data uncertainties, would be the optimal solution, as shown by the BNN + D results. Due to the computational complexity of the BNN + D approach, modification may be required to obtain a computationally efficient and accurate model. In this context, the advantage of fast gradient boosting computation on decision trees can be exploited. The QGB model could be improved by adding the model uncertainties, for example, via an ensemble of multiple models (with data uncertainty-informed base models) or virtual ensembles (Malinin et al., 2021) using a single gradient boosting model. Instead of estimating multiple quantile functions separately, the method can be modified to

Table 5
Advantages and Disadvantages of Different Investigated Uncertainty Quantification Approaches for Vertical Total Electron Content Forecasting

	SE	QGB	BNN	BNN + D
PROS	Improved mean VTEC No distribution assumption	Fast to train CI ~ 95% GT No distribution assumption	Higher Corr. to GT	CI > 95% GT
CONS	Many models to train Uncertainty too small	Estimate each quantile	Slow to train Gaussian distribution Uncertainty too small	Slow to train Gaussian distribution

Note. CI stands for confidence interval, GT for ground truth, and Corr. for correlation coefficients.

estimate them simultaneously (e.g., X. Han et al., 2021; Y. Liu & Wu, 2011). Moreover, adding information about the uncertainty of the input data directly into a model can further improve the probabilistic estimation of output and provide a more realistic representation of the uncertainties (e.g., Kiani Shahvandi & Soja, 2022). It is also important to note that we assumed GIM CODE data to be GT, which is not error-free. In further work, GT uncertainty information may also be included, for example, as an additional input value to the model as in Kiani Shahvandi and Soja (2022).

The results from this study show that the uncertainty arising from the data is much larger than that of the model parameters. Therefore, the input data of an ML-based ionosphere model are much more important to be considered for future improvements. Further steps may include investigating and incorporating new input observations, extracting new input features for VTEC modeling and forecasting that can characterize the effects of space weather on the ionosphere in a way that is more helpful to the learning process. Some input observations, such as the F10.7 and Kp indexes, have lower resolution. Including data with higher temporal resolution and minimizing the need to interpolate values may also reduce uncertainties.

As can be seen from the results, the uncertainties during the space weather event in September 2017 are up to 1.5 to 2 times larger than during the quiet period in April 2017. The ionospheric response to a geomagnetic storm depends on several factors that lead to distinct VTEC irregularities during different storms, as well as on the overall presence of storm events in the data set. The VTEC response to geomagnetic disturbances depends not only on the intensity of the storm, but also on the season, magnetic local time, storm onset time, magnetic latitude, and solar cycle phase (Greer et al., 2017; J. Liu et al., 2010; Vijaya Lekshmi et al., 2011). Therefore, it varies from one storm to another, making it difficult for a learning algorithm to find an approximation function that generalizes to all storms. Another challenge is the small number of storm samples in the training data. The analysis by Natras et al. (2022a) shows that only around 11% of the samples from January 2015 to December 2016 belong to geomagnetic active and storm conditions, even though these years contain the highest number of geomagnetic storms in solar cycle 24. If the training data set contained balanced instances of quiet and storm periods, the forecast accuracy during a space weather event could be improved and the associated uncertainties reduced.

Recommended solutions for the imbalanced data set to be explored in future work include improving the input features for learning rare space weather-related VTEC signatures, training on the balanced data set achieved with oversampling or undersampling, or developing a cost-sensitive solution that can adjust the penalty for the degree of importance assigned to the minority case. Another possible solution is combining physical laws and equations with ML to develop a physically informed ML-based VTEC model, which could improve space weather modeling when only few training examples of space weather events exist and reduce uncertainties.

Since dynamic solar-terrestrial processes and space weather govern the ionosphere, and the VTEC quantity is essential for positioning applications and early-warning systems of space weather effects, it is crucial to include reliability and confidence information in VTEC and space weather forecasting. Moreover, such information will increase the explain ability and interpretability of ML-based ionosphere modeling and forecasting, and trust in ML results in general. Therefore, we encourage further work on uncertainty estimation to produce trustworthy probabilistic ionosphere and space weather forecasts. We hope that the research community will begin to incorporate probabilistic frameworks into their ML solutions alongside the tremendous amount of work exploring various learning algorithms for VTEC approximation. This study is a starting point for discussing and integrating UQ solutions into ML-based VTEC forecasting and will hopefully lead to further ML-based ionosphere and space weather studies that take uncertainties into account.

Data Availability Statement

Software used to implement machine learning (ML) approaches are ScikitLearn (Pedregosa et al., 2011) and TensorFlow (Abadi et al., 2015). The figures were created in Python using Seaborn (Waskom, 2021) and Matplotlib (Hunter, 2007), and in Matlab (MATLAB, 2020). Global ionosphere maps (GIM) produced by the Center for Orbit Determination in Europe (CODE) at the University of Bern, available in Dach et al. (2020), were used to prepare the VTEC data in this study. Other input data to the ML-based VTEC models: sunspot number, F10.7 solar radio flux, solar wind plasma speed, Bz index, Dst index, Kp index, and AE index are publicly available via NASA/GSFC's OMNIWeb (King & Papitashvili, 2005). The IRI 2016 was retrieved from the Community Coordinated Modeling Center (CCMC) Instant-Run System of NASA Goddard Space Flight Center at https://kauai.ccmc.gsfc.nasa.gov/instantrun/iri. The dataset containing the probabilistic VTEC forecast results for the year 2017 from the four ML uncertainty quantification approaches presented and discussed in this study is openly available under the Creative Commons Attribution 4.0 International license at Zenodo (Natras et al., 2023b). The codes defining the architecture of the BNN and BNN + D VTEC models, the model development process using training and cross-validation data, and their evaluation using test data can be found in Natras (2023a). The codes for loading the QGB VTEC models and evaluating them using test data are provided along with the developed QGB VTEC models in Natras (2023b).

Acknowledgments

This research was funded by Research Grants—Doctoral Programmes in Germany from the German Academic Exchange Service (in German: Deutscher Akademischer Austauschdienst, DAAD). Open access funding enabled and organized by Projekt DEAL. The authors acknowledge the use of GIM products of CODE from University of Bern; the OMNIWeb CDAWeb service of NASA/GSFC's Space Physics Data Facility and OMNI data; CCMC Instant-Run System of NASA Goddard Space Flight Center and IRI 2016 data. Open Access funding enabled and organized by Projekt DEAL.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Retrieved from https://www.tensorflow.org/

Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., et al. (2021). A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, *76*, 243–297. https://doi.org/10.1016/j.inffus.2021.05.008

Amell, A., Eriksson, P., & Pfreundschuh, S. (2022). Ice water path retrievals from meteosat-9 using quantile regression neural networks. *Atmospheric Measurement Techniques*, *15*(19), 5701–5717. https://doi.org/10.5194/amt-15-5701-2022

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. In *Proceedings of the 32nd international conference on machine learning* (Vol. 37, pp. 1613–1622).

Bottou, L. (2012). Stochastic gradient descent tricks. In G. Montavon, G. B. Orr, & K.-R. Müller (Eds.), *Neural networks: Tricks of the trade* (2nd ed., pp. 421–436). Springer. https://doi.org/10.1007/978-3-642-35289-8_25

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Chan, J. S. K. (2021). Predicting loss reserves using quantile regression running title: Quantile regression loss reserve models. *Journal of Data Science*, *13*(1), 127–156. https://doi.org/10.6339/JDS.201501_13(1).0008

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794).

Dach, R., Schaer, S., Arnold, D., Kalarus, M. S., Prange, L., Stebler, P., et al. (2020). Code final product series for the IGS [Dataset]. Astronomical Institute, University of Bern. Retrieved from http://www.aiub.unibe.ch/download/CODE

Dashora, N., Suresh, S., & Niranjan, K. (2019). Interhemispheric asymmetry in response of low-latitude ionosphere to perturbation electric fields in the main phase of geomagnetic storms. *Journal of Geophysical Research: Space Physics*, *124*(8), 7256–7282. https://doi.org/10.1029/2019JA026671

Duan, T., Anand, A., Ding, D. Y., Thai, K. K., Basu, S., Ng, A., & Schuler, A. (2020). Ngboost: Natural gradient boosting for probabilistic prediction. In *International conference on machine learning* (pp. 2690–2700).

Ferreira, A. A., Borges, R. A., Paparini, C., Ciraolo, L., & Radicella, S. M. (2017). Short-term estimation of GNSS TEC using a neural network model in Brazil. (Studies on mesosphere, thermosphere and ionosphere from equatorial to mid latitudes—Recent investigations and improvements—Part 1). *Advances in Space Research*, *60*(8), 1765–1776. https://doi.org/10.1016/j.asr.2017.06.001

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, *55*(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Greer, K. R., Immel, T., & Ridley, A. (2017). On the variation in the ionospheric response to geomagnetic storms with time of onset. *Journal of Geophysical Research: Space Physics*, *122*(4), 4512–4525. https://doi.org/10.1002/2016JA023457

Han, X., Dasgupta, S., & Ghosh, J. (2021). Simultaneously reconciled quantile forecasting of hierarchically related time series. In A. Banerjee & K. Fukumizu (Eds.), *Proceedings of the 24th international conference on artificial intelligence and statistics* (Vol. 130, pp. 190–198). PMLR. Retrieved from https://proceedings.mlr.press/v130/han21a.html

Han, Y., Wang, L., Fu, W., Zhou, H., Li, T., & Chen, R. (2022). Machine learning-based short-term gps tec forecasting during high solar activity and magnetic storm periods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *15*, 115–126. https://doi.org/10.1109/JSTARS.2021.3132049

Hu, A., Shneider, C., Tiwari, A., & Camporeale, E. (2022). Probabilistic prediction of dst storms one-day-ahead using full-disk soho images. *Space Weather*, *20*(8), e2022SW003064. https://doi.org/10.1029/2022SW003064

Hüllermeier, E., & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, *110*(3), 457–506. https://doi.org/10.1007/s10994-021-05946-3

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Imtiaz, N., Younas, W., & Khan, M. (2020). Response of the low- to mid-latitude ionosphere to the geomagnetic storm of September 2017. *Annales Geophysicae*, *38*(2), 359–372. https://doi.org/10.5194/angeo-38-359-2020

Kaselimi, M., Voulodimos, A., Doulamis, N., Doulamis, A., & Delikaraoglou, D. (2022). Deep recurrent neural networks for ionospheric variations estimation using GNSS measurements. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–15. https://doi.org/10.1109/TGRS.2021.3090856

Kendall, A., & Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? In *Proceedings of the 31st international conference on neural information processing systems* (Vol. 30, pp. 5580–5590).

Kiani Shahvandi, M., & Soja, B. (2022). Inclusion of data uncertainty in machine learning and its application in geodetic data science, with case studies for the prediction of earth orientation parameters and GNSS station coordinate time series. *Advances in Space Research*, *70*(3), 563–575. https://doi.org/10.1016/j.asr.2022.05.042

King, J. H., & Papitashvili, N. E. (2005). Solar wind spatial scales in and comparisons of hourly wind and ace plasma and magnetic field data. *Journal of Geophysical Research*, *110*(A2), A02104. https://doi.org/10.1029/2004JA010649

Koch, K. R. (2018). Bayesian statistics and Monte Carlo methods. *Journal of Geodetic Science*, *8*(1), 18–29. https://doi.org/10.1515/jogs-2018-0003

Koenker, R., & Hallock, K. F. (2001). Quantile regression. *The Journal of Economic Perspectives*, *15*(4), 143–156. https://doi.org/10.1257/jep.15.4.143

Kumar, A., Chakrabarty, D., Pandey, K., & Yadav, A. K. (2022). Solar flux dependence of post-sunset enhancement in vertical total electron content over the crest region of equatorial ionization anomaly. *Journal of Geophysical Research: Space Physics*, *127*(5), e2021JA030156. https://doi.org/10.1029/2021JA030156

Kutiev, I., Otsuka, Y., Saito, A., & Tsugawa, T. (2007). Low-latitude total electron content enhancement at low geomagnetic activity observed over Japan. *Journal of Geophysical Research*, *112*(A7), 893. https://doi.org/10.1029/2007JA012385

Lee, S., Ji, E.-Y., Moon, Y.-J., & Park, E. (2020). One-day forecasting of global TEC using a novel deep learning model. *Space Weather*, *19*(1), 2020SW002600. https://doi.org/10.1029/2020SW002600

Licata, R. J., & Mehta, P. M. (2022). Uncertainty quantification techniques for data-driven space weather modeling: Thermospheric density application. *Scientific Reports*, *12*(1), 1–17. https://doi.org/10.1038/s41598-022-11049-3

Liu, J., Zhang, D., Mo, X., Xiong, C., Hao, Y., & Xiao, Z. (2020). Morphological differences of the northern equatorial ionization anomaly between the eastern Asian and American sectors. *Journal of Geophysical Research: Space Physics*, *125*(3), e2019JA027506. https://doi.org/10.1029/2019JA027506

Liu, J., Zhao, B., & Liu, L. (2010). Time delay and duration of ionospheric total electron content responses to geomagnetic disturbances. *Annales Geophysicae*, *28*(3), 795–805. https://doi.org/10.5194/angeo-28-795-2010

Liu, L., Morton, Y. J., & Liu, Y. (2022). Ml prediction of global ionospheric TEC maps. *Space Weather*, *20*(9), e2022SW003135. https://doi.org/10.1029/2022SW003135

Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020). Forecasting global ionospheric TEC using deep learning approach. *Space Weather*, *18*(11), e2020SW002501. https://doi.org/10.1029/2020SW002501

Liu, Y., & Wu, Y. (2011). Simultaneous multiple non-crossing quantile regression estimation using kernel constraints. *Journal of Nonparametric Statistics*, *23*(2), 415–437. https://doi.org/10.1080/10485252.2010.537

Malinin, A., Prokhorenkova, L., & Ustimenko, A. (2021). Uncertainty in gradient boosting via ensembles. In *International conference on learning representations*. Retrieved from https://openreview.net/forum?id=1Jv6b0Zq3qi

MATLAB. (2020). *Version 2020a*. The MathWorks Inc.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.

Natras, R. (2023a). Randa-lab/Bayesian_Neural_Network_Probabilistic_Io nosphere_VTEC: Bayesian_Neural_Network_Probabilistic_Ionosphere. *Zenodo*. https://doi.org/10.5281/zenodo.7858906

Natras, R. (2023b). Randa-lab/Quantile_Gradient_Boosting_for_Probabili stic_VTEC: Quantile_Gradient_Boosting_Probabilisti c_Ionosphere_Evaluation. *Zenodo*. https://doi.org/10.5281/zenodo.7858661

Natras, R., Goss, A., Halilovic, D., Magnet, N., Mulic, M., Schmidt, M., & Weber, R. (2023a). Regional ionosphere delay models based on cors data and machine learning. *NAVIGATION: Journal of the Institute of Navigation*, *70*(3), navi.577. https://doi.org/10.33012/navi.577

Natras, R., Soja, B., & Schmidt, M. (2022a). Ensemble machine learning of Random Forest, AdaBoost and XGBoost for vertical total electron content forecasting. *Remote Sensing*, *14*(15), 3547. https://doi.org/10.3390/rs14153547

Natras, R., Soja, B., & Schmidt, M. (2022b). Machine learning ensemble approach for ionosphere and space weather forecasting with uncertainty quantification. In *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)* (pp. 1–4). https://doi.org/10.23919/AT-AP-RASC54737.2022.9814334

Natras, R., Soja, B., & Schmidt, M. (2023b). Dataset of machine learning forecasted VTEC from paper: Uncertainty quantification for machine learning-based ionosphere and space weather forecasting. *Zenodo*. https://doi.org/10.5281/zenodo.7741342

Orus Perez, R. (2019). Using tensorflow-based neural network to estimate GNSS single frequency ionospheric delay (iononet). *Advances in Space Research*, *63*(5), 1607–1618. https://doi.org/10.1016/j.asr.2018.11.011

Özkan, A. (2022). An artificial neural network model in predicting VTEC over central Anatolia in Turkey. *Geodesy and Geodynamics*, *14*(2), 130–142. https://doi.org/10.1016/j.geog.2022.07.004

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Rahaman, R., & Thiery, A. (2021). Uncertainty quantification and deep ensembles. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan (Eds.), *Advances in neural information processing systems* (Vol. 34, pp. 20063–20075). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2021/file/a70dc40477bc2adceef4d2c90f47eb82-Paper.pdf

Schaer, S. (1999). *Mapping and predicting the earth's ionosphere using the global positioning system* (Vol. 59). Institut für Geodäsie und Photogrammetrie, Eidg. Technische Hochschule.

Siddique, T., Mahmud, M. S., Keesee, A. M., Ngwira, C. M., & Connor, H. (2022). A survey of uncertainty quantification in machine learning for space weather prediction. *Geosciences*, *12*(1), 27. https://doi.org/10.3390/geosciences12010027

Tagasovska, N., & Lopez-Paz, D. (2019). Single-model uncertainties for deep learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 32). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2019/file/73c03186765e199c116224b68adc5fa0-Paper.pdf

Vasseur, S., & Aznarte, J. (2021). Comparing quantile regression methods for probabilistic forecasting of NO_2 pollution levels. *Scientific Reports*, *11*(1), 11592. https://doi.org/10.1038/s41598-021-90063-3

Vijaya Lekshmi, D., Balan, N., Tulasi Ram, S., & Liu, J. Y. (2011). Statistics of geomagnetic storms and ionospheric storms at low and mid latitudes in two solar cycles. *Journal of Geophysical Research*, *116*(A11), 530. https://doi.org/10.1029/2011JA017042

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021

A4 Conference Publications

CP-I

Reference: Natras, R., Schmidt, M. (2021). **Machine Learning Model Development for Space Weather Forecasting in the Ionosphere**. In: CEUR Workshop Proceedings of the CIKM 2021 Workshops co-located with the 30th ACM International Conference on Information and Knowledge Management (CIKM 2021), Vol. 3052, pp. 1-7, https://ceur-ws.org/Vol-3052/short10.pdf

Summary: This publication presents the workflow of the machine learning model development for the space weather forecast in the Earth's ionosphere. The problem of space weather forecasting using traditional approaches is discussed, as well as the advantages of using machine learning instead. In addition, the methods and approaches for building a machine learning model are presented, together with challenges related to data and algorithms. The machine learning workflow for the problem of space weather forecast is discussed, from problem formulation and data acquisition, data preparation and feature engineering, learning algorithms, model training, evaluation and deployment, to challenges and open issues.
The estimated contribution of R. Natras to CP-I is 90%.

CP-II

Reference: Natras, R., Soja, B., Schmidt, M. (2022): **Machine Learning Ensemble Approach for Ionosphere and Space Weather Forecasting with Uncertainty Quantification**. In: IEEE Xplore Proceedings of 2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC), Gran Canaria, Spain, pp. 1-4, https://doi.org/10.23919/AT-AP-RASC54737.2022.9814334

Summary: This publication presents a novel machine learning approach to ionosphere forecasting, including forecasting the space weather impact on the ionosphere. It exploits a data-driven approach in which the models learn underlying processes and relationships from data describing solar activity, solar wind, interplanetary and Earth's magnetic fields, and the ionosphere. A multi-model and multi-data ensemble forecasting approach using diverse models of different learning algorithms with different training datasets is developed to generate 1-day VTEC forecasts. This approach improved forecasting accuracy compared to a single-model-based approach. In addition, the forecast uncertainty was assessed by estimating an ensemble spread. The results show potential for forecasting VTEC in different ionospheric regions during quiet and storm periods while quantifying the uncertainties.
The estimated contribution of R. Natras to CP-II is 85%.

CP-III

Reference: Natras, R., Halilovic, Dz., Mulic, M., Schmidt M. (2023): **Mid-latitude Ionosphere Variability (2013–2016), and Space Weather Impact on VTEC and Precise Point Positioning**. In: Ademović, N., Mujčić, E., Mulić, M., Kevrić, J., Akšamija, Z., editors, Advanced Technologies, Systems, and Applications VII. IAT 2022. Lecture Notes in Networks and Systems, vol 539. Springer, Cham., pp. 471–491, https://doi.org/10.1007/978-3-031-17697-5_37

Summary: This publication presents a detailed study of the VTEC variability for Bosnia and Herzegovina by analyzing daily, monthly, seasonal, solar cycle, and space weather-related VTEC variations for a 4-year period, i.e. from 2013 to 2016. VTEC values were derived from GNSS observations of the EPN station SRJV, located in Sarajevo. VTEC varied according to solar activity and solar cycle progression. The largest VTEC values and standard deviations occurred during the spring equinox. Moreover, ionospheric winter anomaly was observed during high solar activity. Furthermore, the impact of a severe St. Patrick's geomagnetic storm in March 2015 was studied in terms of VTEC variations and dual-frequency PPP. During a storm, VTEC values change from about 50% to 150% with respect to the mean VTEC of quiet days. Cordinate variations were visible in all three components, north, east and up, where the highest position error was up to 0.2 m in the up component.
The estimated contribution of R. Natras to CP-III is 70%.

A5 Co-author Publications

Reference: Barta, V., Natras, R., Srećković, V., Koronczay, D., Schmidt, M., Šulic D. (2006): **Multi-instrumental investigation of the solar flares impact on the ionosphere on 05–06 December 2006**. In: Frontiers in Environmental Science, 10. https://doi.org/10.3389/fenvs.2022.904335

Summary: This publication studies the solar flare impact on the ionosphere above Europe on 05 and 06 December 2006 using ground-based (ionosonde and VLF) and satellite-based data (Vertical Total Electron Content (VTEC) derived from GNSS observations and VLF measurements from DEMETER satellite). Based on the Kp and Dst indices, 05 December 2006 was a quiet day, while there was a geomagnetic storm on 06 December 2006. The total fade-out of the EM waves emitted by the ionosondes was experienced at all investigated stations during an X9 class flare on 05 December 2006. The variation of the fmin parameter (first echo trace observed on ionograms, it is a rough measure of the "non-deviative" absorption) and its difference between the quiet period and during the flares have been analyzed. A latitude-dependent enhancement of fmin (2–9 MHz) and fmin (relative change of about 150%–300%) was observed at every station at the time of the X9 (on 05 December) and M6 (on 06 December) flares. Furthermore, we analyzed VTEC changes during and after the flare events with respect to the mean VTEC values of reference quiet days. During the X9 solar flare, VTEC increased depending on the latitude (2–3 TECU and 5%–20%). On 06 December 2006, the geomagnetic storm increased ionization (5–10 TECU), representing a "positive" ionospheric storm. However, an additional peak in VTEC related to the M6 flare could not be detected. We have also observed a quantifiable change in transionospheric VLF absorption of signals from ground transmitters detected in low Earth orbit associated with the X9 and M6 flare events on 05 and 06 December in the DEMETER data. Moreover, the amplitude and phase of ground-based, subionospherically propagating VLF signals were measured simultaneously during the investigated flares to analyze ionosphere reaction and to evaluate the electron density profile versus altitude. For the X9 and M6 flare events, we have also calculated the ionospheric parameters important for the description and modeling of this medium under forced additional ionization.
The estimated contribution of R. Natras to CP-III is 30%.