# Forecast of the Global TEC by Nearest neighbour technique

Enric Monte-Moreno[1], Manuel Hernández-Pajares[1], and Heng Yang[2]

[1]Universitat Politècnica de Catalunya
[2]School of Electronic Information and Engineering,Yangtze Normal University

November 24, 2022

**Abstract**

We propose a method for Global Ionospheric Maps of Total Electron Content forecasting using the Nearest Neighbour method. The assumption is that in a database of global ionosphere maps spanning more than two solar cycles, one can select a set of past observations that have similar geomagnetic conditions to those of the current map. The assumption is that the current ionospheric condition can be expressed by a linear combination of conditions seen in the past. The average these maps leads to common geomagnetic components being preserved and those not shared by several maps being reduced. The method is based on searching the historical database for the dates of the maps closest to the current map and using as a prediction the maps in the database that correspond to time shifts on the prediction horizons. In contrast to other methods of machine learning, the implementation only requires a distance computation and does not need a previous step of model training and adjustment for each prediction horizon. Also provides confidence intervals for the forecast. The method has been analyzed for two full years (2015 and 2018), for selected days of 2015 and 2018, i.e., two storm days and two non-storm days and the performance of the system has been compared with CODE (24- and 48-hour forecast horizons).

Posted on 24 Nov 2022 — CC-BY-NC 4 — https://doi.org/10.1002/essoar.10507974.1 — This a preprint and has not been peer reviewed. Data may be preliminary.

# Forecast of the Global TEC by Nearest neighbour technique

**Enric Monte-Moreno [1], Heng Yang [2], Manuel Hernández-Pajares[3,4]**

[1]Universitat Politècnica de Catalunya, (Department of TSC, TALP) Barcelona, Spain
[2]School of Electronic Information and Engineering, Yangtze Normal University, 408100, Chongqing, China.
[3]Universitat Politècnica de Catalunya (UPC-IonSAT), Barcelona, Spain.
[4]Institut d'Estudis Espacials de Catalunya (IEEC), Barcelona, Spain.

**Key Points:**

- A new method for forecasting the Global Ionospheric Maps of Total Electron Content is presented.
- The method is based in searching in a database that encompases two solar cycles.
- The forecasting horizons can be adjusted on real time, without need of retraining the system.

Corresponding author: Enric Monte-Moreno, `enric.monte@upc.edu`

## Abstract

We propose a method for Global Ionospheric Maps of Total Electron Content forecasting using the Nearest Neighbour method. The assumption is that in a database of global ionosphere maps spanning more than two solar cycles, one can select a set of past observations that have similar geomagnetic conditions to those of the current map. The assumption is that the current ionospheric condition can be expressed by a linear combination of conditions seen in the past. The average these maps leads to common geomagnetic components being preserved and those not shared by several maps being reduced. The method is based on searching the historical database for the dates of the maps closest to the current map and using as a prediction the maps in the database that correspond to time shifts on the prediction horizons. In contrast to other methods of machine learning, the implementation only requires a distance computation and does not need a previous step of model training and adjustment for each prediction horizon. Also provides confidence intervals for the forecast. The method has been analyzed for two full years (2015 and 2018), for selected days of 2015 and 2018, i.e., two storm days and two non-storm days and the performance of the system has been compared with CODE (24- and 48-hour forecast horizons).



## Plain Language Summary

In this paper we present a method for the prediction of Global Ionospheric Maps of Total Electron Content. In this paper we argue that the prediction can be performed from information contained in a database spanning two solar cycles. We also show why the use of previous maps with similar properties allows successful prediction. We then compare the performance of the algorithm for various horizons.


## 1 Introduction

The variations in electron density, and correspondingly in its line-of-sight integral, the vertical total ionospheric electron content (TEC) affect satellite telecommunication services and Global Navigation Satellite Systems (GNSS) due to the effect these fluctuations have on radio wave propagation. The TEC variations induce changes that affect the transmission quality either, as reduced transmission rate and positioning errors. This justifies the importance of monitoring and predicting global TEC maps, as the knowledge of the spatial distribution of TEC would allow corrections to be made. The TEC measurement consists of the total number of electrons integrated along a 1 $m^2$ cross-section tube, using as a unit the TECU defined as $= 10^{16} electrons/m^2$. The prediction of Global Ionospheric Maps (GIM) at different horizons is important because the ionospheric delay is main limiting factor in high-accuracy positioning. These predictions may allow achieving sub-meter accuracy for mass-market single-frequency receivers (García-Rigo et al., 2011). In this paper we propose a method for Global Ionospheric Maps of Total Electron Content forecasting using the Nearest Neighbour method which we denote as NNGIM.

## 2 Issues regarding the TEC map prediction.

The difficulty in predicting TEC maps of the ionosphere stems from the fact that the quality of the prediction depends on geomagnetic activity, season, geographical location, ionospheric structures, such as equatorial ionization anomaly (EIA), and storm-enhanced density (SED). Besides, the sparsity in the geographical distribution of stations leads to problems related to interpolation in regions not covered by these stations. Added to the problem of variability and dependence on external factors, the prediction of GIM maps by machine learning techniques is affected by the need for machine learning techniques to infer prediction rules from examples. This means that the database

to train the system has to be rich enough to represent most of the combinations of effects acting on the ionosphere. One intrinsic limitation of machine learning-based systems is the availability of a database that sufficiently covers the multiple forms of phenomena that can occur. In the works cited below, most of the prediction proposals are made using databases covering at most one solar cycle. In this work, we will be using UPC-IonSAT's database, which covers more than two solar cycles. It is important to highlight the importance of having more than one solar cycle to infer the structure and parameters of the forecasting system. Within the long-term solar cycle periodicity, there is large variability. As an example analized in this paper, we can mention two dates when storms occur. I.e., the Saint Patrick storm of 17 March 2015 (maximum of solar cycle C23) and the storm of 25-26 August 2018 (minimum of solar cycle C23). These are dates in different phases of the solar cycle, in which we have high solar and geomagnetic activity superimposed on different basal levels of ionization. In Appendix 9 Tables 5 and 7, summarise the Kp values and the solar flares that occurred on these days. In these two days, the activity in terms of Kp values and magnitude of the flares is similar. Therefore, within the periodicity associated with the solar cycles and the season of the year, there is a high variability that makes it difficult to infer prediction rules. This high variability, in addition to the baseline levels of activity due to the periodicity components, justifies the need for a long enough database.

The need for a database that sufficiently covers the variability of GIMs presents significant technical problems from the point of view of prediction algorithms. In the case of two solar cycles, with maps at a rate of one every 15 minutes, the resulting database consists of more than one million maps. The use of databases of this size makes the hardware requirements demanding, and the computational time requirements to perform topology and parameter tuning of the machine learning system are substantial.

To address the problem the above mentioned problem, ie., of training a machine learning system for forecasting the GIMs, making, there are two approaches.

- **Local approach**: In this case, a specific subset of the database is constructed from the current observation. An example is Monte Moreno et al. (2018), in which maps immediately before the current map are used, and based on these maps and the tangent spaces a linear combination is generated that predicts the maps in the immediate future. This approach assumes that the change in the maps has inertia that determines the future evolution. In C. Wang et al. (2018) they apply a similar idea to calculate the autoregression coefficients that predict the values of the spherical harmonics that allow the GIMs to be reconstructed. Another approach is the one followed in this article, in which prediction is made based on past examples that have a small distance to the current observation. This approach assumes that conditions similar to the one observed in the current map have occurred in the near past and that the temporal evolution of the current map can be inferred from the evolutions seen in the previous history. A noteworthy aspect of the local approximation is that increasing the number of prediction horizons does not lead to a significant increase in computation time, as most of the computation time comes from determining the coefficients in a window that spans a limited amount of time.
- **Global approach**: In this case, the prediction model uses all the historical GIMs. One consequence of this is that to make a reliable prediction, the model has to be estimated from a sufficiently rich set of examples. This leads to problems of implementation. For Support Vector Machines, this approach is infeasible, since it is necessary to create the Gram matrix, which is the square of the number of examples, and it must be kept in memory. In the case of Deep Learning (Goodfellow et al., 2016), the training has to be carried out in Graphical Processing Units (GPU), which have limited memory. In the author's experience (EMM), when trying to solve this problem with Convolutional Neural Networks, on a high-end GPU, train-

ing one model took about a week. This is without taking into account the need to repeat the training to test different topologies and adjust parameters. This computational requirements were for the case where to reduce the model's complexity and take advantage of seasonal similarities in mapping, one model per month of all the years was trained. Even with this partitioning of the database, the resulting model occupied between 1 Gigabyte and 5 Gigabytes depending on the topology. The resources needed to perform the prediction in production, in this case, were significant, as the model has to be loaded into memory and the prediction operations have to be performed.

Another significant limitation in the approach using Deep Learning and similar methods is that either a completely new model or a more complicated topology has to be trained when increasing the number of prediction horizons. In contrast, in the method we propose NNGIM, which is based on finding the nearest map, increasing or changing the values of the horizons has minimal repercussions on the execution time.

A natural model for forecasting the GIM maps that has been used literature (see Section 3) is the Long Short Time Memory (LSTM) (Goodfellow et al., 2016) architecture. A very significant limitation of the LSTM architectures is that they consist of units that have saturating nonlinearities, such as hyperbolic tangent and sigmoid. Since the GIM statistics are long tail (see the last section of Monte Moreno et al. (2018)), the units work much of the time in saturation and cannot model large amplitudes. One consequence is that precisely the regions of interest where there are large TEC gradients cannot be modelled correctly by these units. This is why (EMM), in a first approach to the problem, opted for CNN with Relu-type non-linearities. The complexity of Deep Learning based methods was one of the motivations for seeking a more simple approach to the problem.

## 3 Precedents and limitations of the GIM forecast performances

We will now discuss some precedents to put the NNGIM in context. The features and limitations of other GIM prediction methods will allow us to justify NNGIM design decisions. This section will also serve to establish the limitations of the global approach to forecasting.

- **Global approach**: A first approach to the problem of predicting TEC maps consists of predicting TEC values for specific stations, thus obtaining a local description of the TEC distribution. This is the case of Xiong et al. (2021), where they predict the TEC over China using a variant of the LSTM type networks (ED-LSTM). This type of method differs from ours in the sense that the prediction is done at the station level and there is no interpolation process. One point to note is the use of data from one solar cycle (Jan 2006 to April 2018). The authors use training data from 2006 to 2016, validation between Jan 2017 and April 2018. To avoid the problem of the solar cycle-dependent baseline TEC level, and to adapt the data to the structure of the LSTM grids, the authors normalise the data. This assumes that the variations around the baseline TEC value are similar between different times of the solar cycle. This solves the problem of the variation of the mean TEC level with the solar cycle. One problem related to their approach is that the neural network units they apply have saturation-type non-linearities, which has as a consequence that for extreme values, the units work on saturation. Note that the statistics of the TEC distribution is Leptokurtic, i.e., long tail. On the other hand, an advantage of the type of neural network they employ is that it allows the use of external data naturally in the architecture (solar flux and geomagnetic activity data). In addition to the LSTM architecture (ED-LSTM), the authors explore other architectures and provide a performance hierarchy. The forecast horizons are 2-hour, 3-hour, and 4-hour, using as input a window of past samples between

one day and three days. An important lesson from this work is that the inertia hypothesis, in the sense that the temporal evolution of the TEC follows a trajectory specified by the near past, leads to a prediction barrier at a horizon of a few hours. This limit on the prediction horizon under these conditions was also found in Monte Moreno et al. (2018).

An article reporting a related architecture is Cherrier et al. (2017). Unlike the previous case, the objective was to predict global TEC maps, with a resolution of 5 x 2.5 degrees in longitude and latitude. The temporal resolution was 2 hours. To solve the diurnal cyclicity problem, they use a solar centred reference frame. The authors propose the prediction of global maps with prediction horizons increasing in two-hour steps up to 48 hours. The input data were the maps for the three immediately preceding days. The type of architecture they propose is based on a sequence to sequence, in which CNN-type networks are combined with memory networks, either LSTM or Gated Recurrent Units (GRU), both with saturating nonlinearities. The authors report that prediction at intervals longer than 24 hours did not achieve good results; in fact, in the 24-hour prediction, they obtain a result that improves the cyclic prediction by only 6%. The study was conducted using the data from 1/1/2014 to 12/31/2016. Note also, that the use of LSTM or GRU also suffers from the limitation that the observations are leptokurtic, which means that the nolinearities work in saturation for extreme values.

In Liu et al. (2020) they propose a system based on the use of two LSTM layers followed by a fully connected dense layer for the prediction of the global TEC maps. Unlike the previous cases, the prediction is performed directly on the spherical harmonic (SH) used to build the GIMs. In this approach, in addition to using the information in the recent past (24h) regarding the SH, they also use external information that helps to make the prediction, such as the solar extreme ultraviolet (EUV) flux, the hour of the day, and disturbance storm time (Dst) index. The prediction horizon is set to 1 hour and 2 hours. It is interesting to note that the prediction has an error with respect to frozen maps (persistence) of 60 % at one hour and 63 % at two hours. Note that (although the experiment is not totally comparable) this gain is similar to the obtained by the frozen cyclic approach vs. the persistence hypothesis, see section 7. As a test base, the intervals before and after the interval used for the training base were used. That is, for the training base the interval: 1 January 2015 to 26 May 2016 and for the test base the intervals 19 October to 31 December 2014 and 27 May to 31 December 2016, thus ensuring a similarity between the training and test conditions.

The methodology of the above-mentioned works is correct from the point of view of Deep Learning type network design, however, despite the correctness, it reflects the limitations of this type of technique. These limitations are typical of the general approach to the TEC prediction problem using Deep Learning and do not indicate a misuse of the technique by the authors. Limitations of Deep Learning are the need to process the input data such as normalisation or de-trending of the TEC, the difficulty of performing a test under train-like conditions, the fact that some networks require saturating nonlinearities that are not fit for long-tail input distributions, and the presence of a prediction horizon lower than 24 hours.

- **Local approach**:

  This approach uses information from recent activity to estimate the parameters of the prediction model.

  In C. Wang et al. (2018), the authors describe a system based on autoregressive models, with coefficients computed from a history covering the previous 30 days. The prediction is made on the SH coefficients, which allow the GIM to be reconstructed. By estimating the model locally, they can adapt the system to short-term climatology. This allows them to test the model at different times of the solar cycle, without the need for special partitioning of the database, as is done in the case of deep learning. The performance of the model is tested against CODE, IGS prod-

ucts, and TEC measurements via JASON. The prediction result is different depending on the activity at the time, with worse results at times of high activity. One result is that the RMSE error of prediction during a low activity period was 1.5 TECUs at 24 hours. In Krankowski et al. (2005) the authors use autoregressive moving average (ARMA) for VTEC prediction for stations in Northern Europe. In this article, they use information related to the analysis in wavelets to establish the prediction at 1, 2, and 3-hour horizons, calculating the ARMA coefficients from the last 7 days. The TEC profiles follow a daily pattern, so an ARMA-type method is suitable for modeling the cyclicities.
In García-Rigo et al. (2011), the authors propose a method for the prediction of GIMs with horizons of up to 2 days. It is based on a method that predicts the coefficients of the discrete cosine transform (DCT) by an autoregressive method. The autoregressive coefficients are calculated locally using information from the last week's maps. From the predicted DCT coefficients, the map at the horizon of interest is computed. By calculating the coefficients using a recent past and using the maps of the previous 24 hours for the prediction, the system can adapt to the current weather conditions. The results were validated with JASON measurements.
In Monte Moreno et al. (2018) a prediction system is proposed based on an autoregressive model of the maps of the last 24 hours, together with the components of the tangent spaces associated with each of the previous maps. The forecast horizons range from half an hour to 24 hours. The tangent space information allows to increase the information on the possible trajectory and deformation of the map over time, and in some way to reflect how the ionospheric climatology changes the shape of the high ionisation regions. One feature related to the comparison with other methods, is the improvement in % of the prediction method compared to a frozen reference in a sun-fixed reference frame. The reference will be the prediction error of keeping the map frozen (see section 7 for more information). As shown in Table 1, the prediction performance has a concave profile. The performance is computed using the recent past, and with autoregressive model coefficients calculated with recent values as well, The best prediction compared to frozen is at a 3-hour horizon, increasing thereafter. At 24 hours, the improvement is only 5 %, which is in line with methods based on deep learning. This leads us to think that there is a certain horizon barrier in terms of prediction using the recent past as input.

Table 1: Forcast vs. Frozen (% RMSE) for the Tangent Space.

| Horizon: | 1/2 h | 1 h | 2 h | 3 h | 6 h | 24 h |
|---|---|---|---|---|---|---|
| Forcast vs. Frozen: | 84.99 % | 77.65% | 71.35% | 69.34% | 87.23% | 95.76% |

The analysis of the precedents leads us to the conclusion that the information immediately prior to the current map does not allow reliable predictions of GIM maps at horizons longer than a few hours. They also indicate the limitations and difficulties of training prediction models, and the complexity of the models and partitions of the database.

This leads us to look for a different approach, in which the prediction is made by searching for situations similar to the current one in a sufficiently large database. A byproduct of this approach is that it allows to create confidence margins of the forecast in a natural way (see Section 8.4)

## 4 UPC-IonSAT real-time Global Ionospheric Maps and Data preprocessing

The GIMs are generated from data gathered from several hundred worldwide GNSS stations. This data stream is obtained through the protocol used by the RT IGS working group and the data processing is performed using the UPC-IonSAT ionosphere model.

The streaming protocol referred to as "Networked Transport of Radio Technical Commission for Maritime Services (RTCM) via Internet Protocol" (NTRIP), was developed by the German Federal Agency for Cartography and Geodesy (BKG), enables the streaming of the observation data from the worldwide permanent GNSS receivers (Weber et al., 2005).

The UPC-IonSAT's RT TOMographic IONosphere Model (RT-TOMION) is a 4D (3D+time) model of the global state of the ionosphere, focused on RT estimation of TEC, mainly based on GPS dual-frequency measurements with the hybrid geodetic and tomographic ionospheric model, and robust to various types of deterioration. This model is the extension of the Tomographic Ionospheric Model (TOMION) developed by UPC in the 1990s and has been employed for UPC RT/near-RT ionosphere service of IGS since 2011 (Hernández-Pajares et al., 1999, 2000; Hernández Pajares, 2014; Roma Dollase et al., 2015; Orús et al., 2005).

Also, the VTEC interpolation techniques of the UPC RT- TOMION model is performed either by spherical harmonics or Kriging (Orús et al., 2005) so to fill the gaps where data is lacking. In addition, the most recent maps are interpolated by means of the ADDGIM algorithm presented in Yang et al. (2021). For more details of the processing and interpolation of the GIMs, see Yang et al. (2021).

## 5 NNGIM algorithm

In this section, we will define the Nearest Neighbour GIM (NNGIM) algorithm. This algorithm consists of searching for the $N$ maps closest (in Euclidean metric) to the current one in the database of past maps (more than one solar cycle). Then, from these maps, the GIMs with an offset equal to the prediction horizon are retrieved and averaged.

The assumption underlying the NNGIM algorithm is that in a database that encompasses more than one solar cycle, a small number of maps with a small distance to the current one can be found, and that have ionosphere conditions in common with the current one, might characterize the maps at a time shift equal to the forecast horizon. Although each ionosphere condition is unique, it is assumed that in the past there have been conditions with a similar composition of external features and that the average of all of them will reflect the specific features of the current one. The set of similar maps therefore take into account the cyclical aspects that influence the overall distribution of TEC along with the various external influences. That is, if we select a set of future map values closer to the current one when averaging, common values in subsets of the future maps will be retained, while non-common conditions will be attenuated. Note that the idea behind the assumption is that there will be subsets of maps representing similar ionospheric conditions, and the overall composition of these parts will allow us to approximate previously unseen situations. We assume that these previously unseen situations are composed of subgroups that characterize part of the previous conditions common to the current situation.

The UPC-IonSat GIMs database, which spans over two solar cycles and consists of more than $10^6$ maps, was used to implement the method (see Yang et al. (2021) for details).

In the Algorithm diagram 1 we present the summary of the NNGIM algorithm. A detailed explanation of the algorithm is given below, defining also the variables involved.

The input of the algorithm consists of a database spanning more than two solar cycles ($Db_{AllMaps}$). Note that for consistency in the computation of the distance between maps at different moments, the database and the current map are transformed to sun-fixed geomagnetic coordinates. After the forecast, the inverse transform is performed.

Since the maps have a seasonal component with a mean TEC value that depends on the season of the year (see Figure 4), the search for the nearest map will be carried out in the vicinity of the current month. Therefore, given the date of the current map $Date_{Test}$, the month is extracted ($M_{Tst}$), and maps the current month and a window of $\pm W_{NeighMonths}$ months are selected from the database. In the experiments, a neighbourhood of $W_{NeighMonths} = 1$ was taken. Other parameters are the forecast horizon in hours ($Horizon$) and the number of nearest neighbours ($Num_{NN}$). The next step is to construct a second database ( $Db_{Ima}$), which will consist of the maps with the current map month and the neighbouring months for all years. The Euclidean distance between the current map $Map(Date_{Test})$ and the maps in the $Db_{Ima}$ database is then calculated. (lines 3 to 7 of the Algorithm diagram 1). The vector of distances is then sorted from smallest to largest (line 8 of the Algorithm diagram 1) and assigned to the vector of indices $Index_{MinDist}$.

We define $Num_{NN}$ as the number of maps to be used for prediction estimation. The Algorithm diagram 1, lines 9 to 15 describe the process for generating the prediction. For the nearest $Num_{NN}$ maps, we find the corresponding index $IndexMap$ and the associated date $Date[IndexMap]$. Next, we add the offset $Horizon$ to generate the date $Date_{NNMap}$ associated with each of the maps. The maps associated with each date $Date_{FutMap} \leftarrow Date_{NNMap}+Horizon$ are combined to generate the future map $Forecast_{Map}$.

Finally, from the maps of the horizon shift, the standard deviation at the pixel level is calculated, as shown in line 17.

Various strategies for combining the maps were tested, such as a simple average, a distance-weighted average, or weight that diminishes with the time difference. We also tried a trim mean, defined as the average of the values of each specific pixel in the maps, using only the values between the 25th percentile and the 75th percentile. The median of the pixels of the nearest $Num_{NN}$ maps was also tested. The combination that gave the best results was a simple average of the maps.

One parameter to be adjusted is the number $Num_{NN}$ used to calculate the forecast. This value depends on the forecast horizon and the month of the year. For all experiments we chose a value $Num_{NN} = 500$. The choice was made based on the performance during June 2019 and was explored for values between 1 and 1000. The rationale for the choice of date was to have a date in a cycle (C24) different from the cycle in which the results are presented (C23), and also at a season of low activity. The experiments showed that for this month and horizons between 3 hours and 48 hours the optimum value was between 150 and 700. In the real-time implementation, a look-up table will be used in which the month and horizon will be related to the $Num_{NN}$ value.

An interesting result is that using only the nearest neighbour, i.e., $Num_{NN} = 1$ provided results with a quality equal to using the cyclic version of the map, (defined as $\hat{Map}_{cyclic}(t+\tau) = Map(t-24h+\tau)$). The performance did not improve until using a number of $Num_{NN}$ greater than 50. This leads us to think that the use of a large number of maps allows us to create a representation of the possible contributions of the factors that affect ionisation. The explanation is that the combination of external factors is larger than the number of examples in the database. The underlying assumption is that the current combination of factors affecting ionisation can be expressed as a linear combination of similar situations in the past.

**Algorithm 1:** The NNGIM algorithm

**Data: Inputs to the algorithm:**
$Date_{Test} \leftarrow$ Date of the test GIM;
$Db_{AllMaps} \leftarrow$ All GIMs of two solar cycles in sun-fixed geomagnetic coordinates;
$W_{Neigh} \leftarrow$ Window of Neighbouring Months;
$Num_{NN} \leftarrow$ Number of elements for computing the mean of the Nearest Neighbours;
$Date \leftarrow$ Dictionary of Dates, indexed by Map number;
$Horizon \leftarrow$ Forecast Horizon in hours;
**Result:** $Forecast_{Map}$,$Forecast^{Std}_{Map}$

**Generate the Forecast Database**;
$M_{Tst} \leftarrow GetMonth(Date_{Test})$ ; `/* Month of the current map */`
$Db_{Ima} \leftarrow \emptyset$ ; `/*` $Db_{Ima}$ `Map DataBase of Current and Neighbouring Months */`
**for** $M$=$M_{Tst} - W_{Neigh}$ *to* $M_{Tst} + W_{Neigh}$ **do**
  $Db_{Ima} \leftarrow (Add\,to\,set)Db_{AllMaps}(M)$ ; `/* Add maps for month M */`
**end**
$Mat_{Dist} \leftarrow Distance(Db_{Ima}, Map(Date_{Test}))$ ; `/* Distance from` $Map(Date_{Test})$ `to` $Mat_{Dist}$ `*/`
$Index_{MinDist} = Argsort(Mat_{Dist})$ ; `/* Argsort returns the Indices of the sorted` $Mat_{Dist}$ `*/`
$For_{Map} \leftarrow \emptyset$ ; `/* Compute mean value of the nearest maps at timestamp + horizon */`
**for** *NumMap=1* ***to*** $Num_{NN}$ **do**
  $IndexMap \leftarrow Index_{MinDist}[NumMap]$;
  $Date_{NNMap} \leftarrow Date[IndexMap]$;
  $Date_{FutMap} \leftarrow Date_{NNMap} + Horizon$;
  $For_{Map} \leftarrow For_{Map} + Db_{AllMaps}[Date_{FutMap}]$;
**end**
$Forecast_{Map} \leftarrow For_{Map}/Num_{NN}$;
$Forecast^{Std}_{Map} \leftarrow Compute_{STD}(Db_{AllMaps}, Date, Index_{MinDist}, Horizon)$;

A product of this algorithm is that it can provide confidence intervals for the GIMs, i.e. the local standard deviation of the ionisation values. The estimation of confidence intervals can be done directly, as a collection of several hundred maps is available. One of the features of the maps from which the prediction is constructed is the variability around a central value, as shown in Figure 2. Therefore from the set of maps used to generate the prediction, one can estimate a standard deviation $Forecast_{Map}^{Std}$ at a pixel level, defining this standard deviation as the deviation of the maps from the mean value of the prediction $Forecast_{Map}$. One point that we show in section 8.4 is that the prediction covers most of the area of the reference map $Ref_{Map}$, so we can consider that this variance provides us with an adequate measure of uncertainty for the prediction.

**Improvements**

The improvements we envisage in the next step are to change the average distance, using a metric on the manifold in which the map is located. This is the distance defined in L. Wang et al. (2005) in which coefficients of the angle between coordinates $g_{i,j} =< e_i, e_j >$ are used to weight the Euclidean distance. The advantage of using this distance is that it allows considering in the similarity measure between maps, distortions such as shifts, rotations, etc. The reason why it has not been used in this implementation is that it requires a computational load proportional to the square of the number of map elements. With the current hardware capabilities at 202, the computation of $Mat_{Dist}$ took about ten minutes, so it was not implemented in the final prototype.

Another improvement is to use a heuristic that decreases the computational needs to determine the nearest neighbors. That is, an algorithm with a suitable heuristic for the dimensionality of the maps and with a lower search cost, as is the case of Omohundro (1989). The fact that the GIMs have the ionisation levels distributed in clear and distinct regions makes this algorithm efficient. This might allow implementing a distance with higher computational cost as the nearest neighbour search cost can be decreased.

The computational cost on an iMac i7 using one core of applying the algorithm was as follows. The Euclidean distance $Mat_{Dist}$ from a map $Map(Date_{Test})$ to the database $Db_{Ima}$ consisting of the current month and the two neighbouring months (with 170,000 maps) was of the order of 135 ms, and the cost of sorting the distances $Argsort(Mat_{Dist})$ of 9 ms, the calculation of the average map $Forecast_{Map}$, was less than 1 ms.

The most time-consuming part of the algorithm is the loading into memory of the pre-computed database $Db_{Ima}$, which occupies 2 Gigabytes. The time cost on an SSD is in the order of 2 seconds. However, in a real-time application, the database can be kept permanently in memory.

The real-time prediction of the implementation of this algorithm can be found at the following URL: *NNGIM forecasts at different horizons* (n.d.), with the following naming convention:

The three regions where the forecast was done: Global Forecast (un*g), North-Pole Forecast (un*n), South-Pole Forecast (un*s) And the different horizons that were implemented in real time:

1. un0g/un0n/un0s: 1 hour Forecast
2. un1g/un1n/un1s: 6 hour Forecast
3. un2g/un2n/un2s: 12 hour Forecast
4. un3g/un3n/un3s: 18 hour Forecast
5. un4g/un4n/un4s: 24 hour Forecast
6. un8g/un8n/un8s: 48 hour Forecast

The Polar predictions consist of segments of the global map clipped at 45 degrees of latitude.

## 6 Illustration of how the algorithm works

To understand how the algorithm works, we will consider two points of view.

1. How the dates of the nearest maps are distributed along the solar cycles: *C23*, *C24* and *C25*.
2. Examples of actual maps to understand how is the variability of the nearest neighbours.

We will perform the analysis on day *2019-05-21 16:15:00 UTC* a *C25* cycle day during summer.

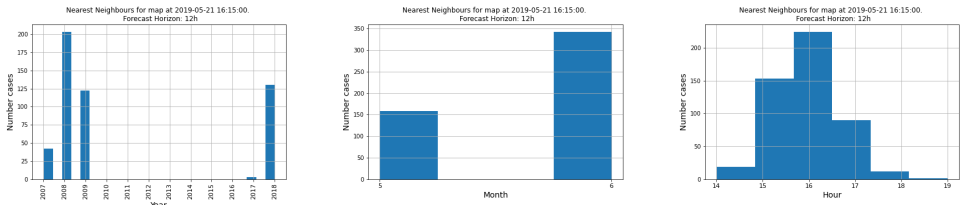



Figure 1: Nearest maps are distributed along solar cycles *C24* and *C25*. Histograms of the years (left), months (center) and time of day (right) of the nearest maps to the map at *2019-05-21 16:15:00 UTC*.

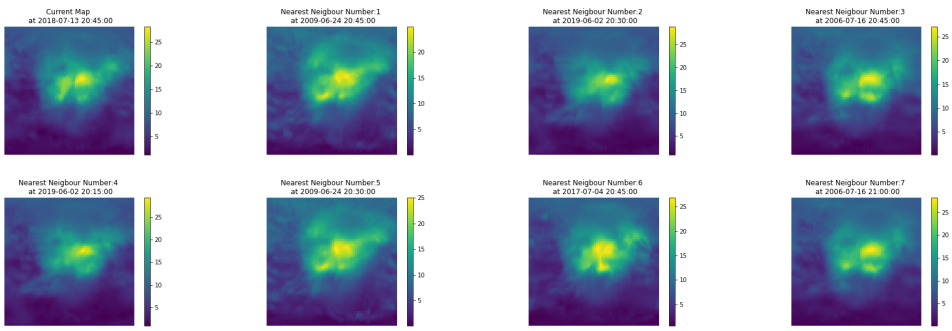



Figure 2: Current map at *2018-07-13 20:45:00 UTC* (subplot at upper left corner), and the seven Nearest Neighbours. All maps in sun-fixed geomagnetic coordinates

1. In Figure 1 we show that the nearest neighbours are distributed over years in the same phase of the cycle. Using only examples from the two cycles *C23* and *C24*. The algorithm does not select any maps from the previous month, and most of the closest maps are from the next month. As we will see later, there is a significant dependence of the behaviour of the algorithm on the month in which the prediction is made. As for the time of day, most of the examples are at the same time of day plus or minus one hour.

2. Next, we consider the variability of the closest maps. The variability of these maps reflects the ionospheric conditions that are common and those that differ. In Figure 2 we show the map for *2018-07-13 20:45:00 UTC* and the seven nearest neighbours in the Euclidean distance sense. To facilitate the comparison, we present the maps in sun-fixed geomagnetic coordinates, which are the setting in which the software computes the distance between maps. The selected maps are from the same time of the year and at similar moments of the solar cycle. On the other hand, the morphology is variable, which indicates that each of the maps reflects ionospheric conditions that have parts in common with the current map as well as specific components. The hypothesis underlying the NNGIM model is that the components common to the current map are preserved by the average, and those that are not common are smoothed out. This variability around common values allows to estimate confidence intervals can capture the most likely ranges in the true reference value. The maps at a future shift equal to the prediction horizon exhibit very similar visual features. For reasons of space and similarity between figures, we do not show them.

## 7 Selection of the Benchmark

In this section, we will define the benchmark to assess the performance of the algorithm. A commonly used reference as benchmark predictor is either a prediction using the current *frozen* map or as a prediction the *cyclic* map, that is, the immediately preceding map of the same time as the time to be predicted. We will formally define the two predictors as follows:

- **Frozen**: $\hat{Map}_{frozen}(t+\tau) = Map(t)$
- **Cyclic**: $\hat{Map}_{cyclic}(t+\tau) = Map(t-24h+\tau)$

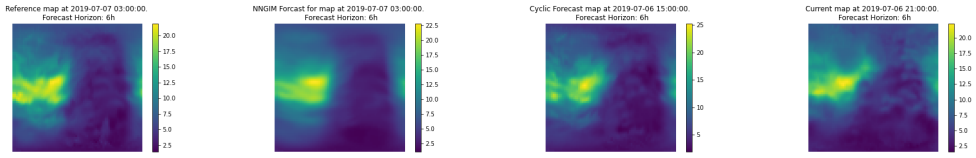
As a benchmark in the following sections, we will use the cyclical prediction $\hat{Map}_{cyclic}(t+\tau)$.

Table 2: Forecasting RMSE (TECU) for $\hat{Map}_{frozen}(t+\tau)$ vs. $\hat{Map}_{cyclic}(t+\tau)$ (June 2019)

| Horizon: $\tau$ (hours) | 3h | 6h | 8h | 12h | 16h | 20h | 24h | 28h | 32h | 36h | 48h |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\hat{Map}_{frozen}(t+\tau)$ (TECU) | 1.87 | 2.35 | 2.51 | 2.59 | 2.51 | 2.18 | 1.42 | 2.19 | 2.57 | 2.61 | 1.54 |
| $\hat{Map}_{cyclic}(t+\tau)$ (TECU) | 1.43 | 1.43 | 1.41 | 1.45 | 1.41 | 1.42 | 1.42 | 1.42 | 1.42 | 1.44 | 1.42 |

We argue this decision through Table 2, in which we show the prediction errors in RMSE (TECU) for prediction horizons ranging from 3 hours to 48 hours. In this case one can see that the prediction cyclic $\hat{Map}_{cyclic}(t+\tau)$ RMSE error and the standard deviation are constant regardless of the prediction horizon, and equal to the 24-hour error of the frozen predictor $\hat{Map}_{frozen}(t+\tau)$. This is to be expected since at all times the cyclic predictor behaves as a 24-hour predictor. On the other hand, an important limitation of the use of the frozen prediction $\hat{Map}_{frozen}(t+\tau)$ as a benchmark is that the comparison is made under non-comparable ionospheric conditions. This results in a sinusoidal behaviour of the RMSE, which increases from 3 hours to 12 hours and then decreases to a minimum at 24 hours. This behaviour is then repeated, reaching a new minimum at 48 hours. Therefore, since the frozen version $\hat{Map}_{frozen}(t+\tau)$ is a very pessimistic benchmark, and has a component that depends on the time of day, we will use as a benchmark only the $\hat{Map}_{cyclic}(t+\tau)$.

To get an idea of the differences between benchmarks and NNGIM prediction, in Figure 3 we present the comparison of the reference map (6-hour a head ground truth),




(a) $Map_{ref}(t+\tau)$ (b) $\hat{M}ap_{nngim}(t+\tau)$ (c) $\hat{M}ap_{cyclic}(t+\tau)$ (d) $\hat{M}ap_{frozen}(t+\tau)$

Figure 3: Comparison of the reference map (a) at *2019-07-07 03:00:00 UTC*, with the NNGIM prediction (b), with the cyclic prediction (c) and with the frozen prediction (i.e., using current map). Note that the maps are in the original coordinates.

with the predictions using the NNGIM algorithms, the cyclic and the frozen reference. The cyclic reference provides local features of the TEC distribution similar to the reference map, while the frozen map has a very different morphology. On the other hand, the NNGIM prediction, despite using maps from other years, captures the structure of the TEC distribution of the reference map.

## 8 Results

For the analysis of the algorithm, we have selected two years of the *C24* cycle and two days of each year. The criterion for selecting the years was to have a sample of one year of high activity in the cycle and one year of low activity. Likewise for the days, in order to contrast the behaviour of the algorithm in the case of storm days vs. quiet days, we chose two storm days of each year and two adjacent days without a storm. In appendix 9, we present a summary of the solar activity on these days (i.e., Kp number and solar flares by the hour).

### 8.1 Analysis of selected years: 2015 and 2018

Figure 4 shows the time series of the average monthly TEC value for the two selected years. The first difference observed in the two years is the underlying monthly average TEC level and the fact that in the most active year (2015), the monthly profile of the TEC level has a marked cyclical component with a minimum in the summer. On the other hand, in the least active year (2018), the cyclical component has a lower amplitude. The mean annual TEC value for 2015 is 20 TECU, while in 2018 it is 8.8 TECU.

First, we show the performance of the NNGIM algorithm in TECU values and then for comparison purposes in percentages concerning the prediction using the frozen cyclic.

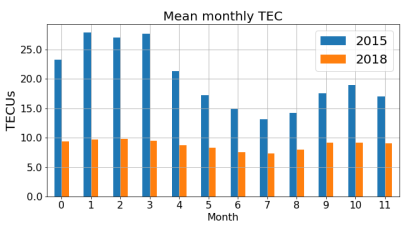



Figure 4: Mean monthly TEC for the years 2015 (in blue) and 2018 (in orange)

In Table 3 we show the average TECU prediction RMSE for 4 prediction horizons. In 2015 the prediction error increases as we increase the horizon from 17% to 20% of the average TEC value. On the other hand, the error in 2018 remains almost constant regardless of the horizon and stands at 18% of the average TEC value in that year. However, as we will see below, the prediction error has an annual cyclical component, being lower in the summer.

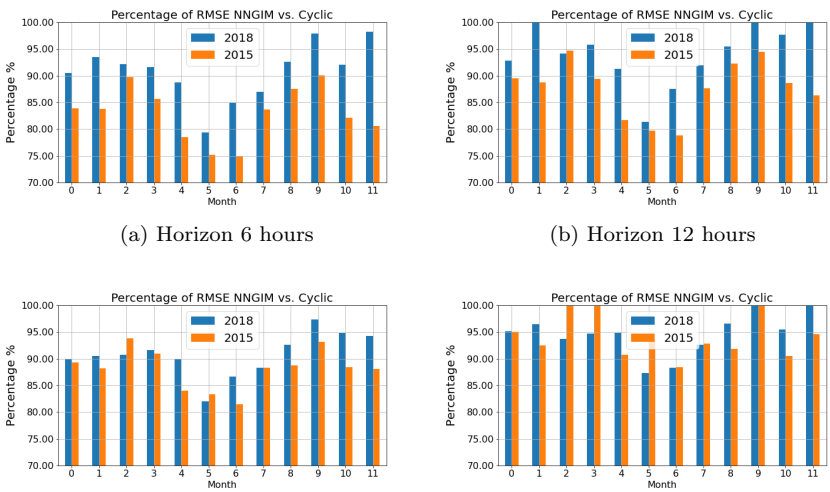
Table 3: RMSE error of the NNGIM algorithm for several horizons

| Horizon | 6h | 12h | 24h | 48h | Mean TECU |
|---|---|---|---|---|---|
| 2015 (TECU) | 3.50 | 3.70 | 3.72 | 4.00 | **20.0** TECU |
| 2018 (TECU) | 1.59 | 1.66 | 1.59 | 1.66 | **8.8** TECU |

In Figure 5 we present the percentage change of the RMSE value for the cyclical prediction vs. NNGIM for various horizons. That is, we plot the ratio

$$\frac{\hat{M}ap_{nngim}(t+\tau)}{\hat{M}ap_{cyclic}(t+\tau)} \times 100\%$$

The first conclusion derived from the figures is that the use of NNGIM provides a decrease that follows an annual pattern and in the summer months for 6 and 12-hour horizons provides a decrease in error in the order of 20% to 25%. This contrasts with the experience with Tangent Spaces predictions (see Monte Moreno et al. (2018)) and Deep Learning based methods (see section 3), where a significant degradation in quality is reported at prediction horizons of the order of 6 hours. The prediction at 24 and 48 hours reported as a percentage of frozen in Cherrier et al. (2017) using Deep Learning is similar to the one shown in the lower row of Figure 5.




(a) Horizon 6 hours

(b) Horizon 12 hours

(c) Horizon 24 hours

(d) Horizon 48 hours

Figure 5: Percentage of RMSE reduction with regard to cyclic freezing for the horizons of 6h, 12h, 24h, 48.

The 12-hour forecast results are worse than the 24-hour ones except for the months of May and June. This is because this is the moment in the interval $(t, t{+}24h)$ when the ionosphere configuration is maximally different from the current state.

On the other hand, 48 hours seems to be a natural limit for the method, as the error reduction for frozen cyclic is on an annual average of 95%.

### 8.2 Performance on selected days of 2015 and 2018

To evaluate the performance of the NNGIM method, we selected two days at the maximum of cycle *24* and two days at the minimum of the same cycle. The criterion for selecting the days was that one of them coincided with a geomagnetic storm and the other one coincided with a nearby day without significant activity. The selected days were:

1. 17 March 2015 (St.Patrick Day storm) and 5 March 2015 (non storm day).
2. 25-26 August 2018 (storm day) and 13-14 August 2018 (non storm day).

In both cases, the *Kp number* and the *solar flares* are shown in appendix 9, Tables 4, 5, 7 and 6

#### *8.2.1 Performance on 5 and 17 March 2015*

In Figure 6 we present the comparison of the NNGIM predictor versus the cyclic frozen for various horizons in the form of a time series, at a rate of one map every 15 minutes.

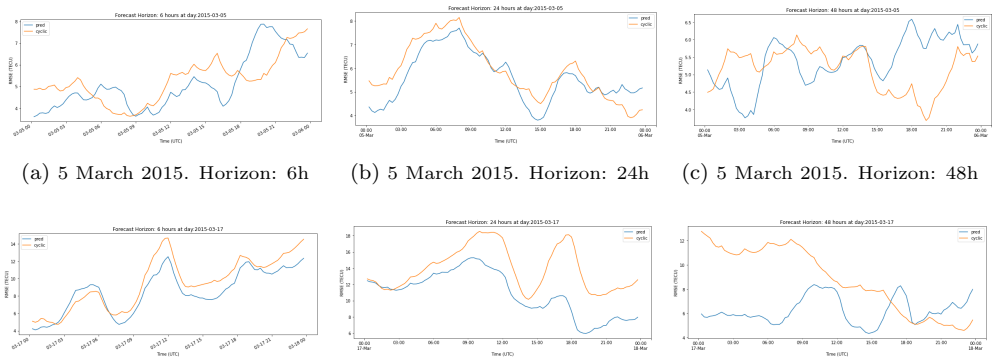
In the top row, the performances of NNGIM vs. frozen cyclic are compared for the 5th of March 2015, which is a day with no significant events (see the Tables 4 and 5). The difference in performance is irregular for the 6-hour forecast, while for the 24-hour forecast the average reduction over the day is a little more than a 10% error. The worse behaviour towards the end of the day could be due to the increase of the Kp indicator and the presence of three solar flares in close temporal proximity. Since the NNGIM method assumes that similar situations have been seen in the past and are used for prediction, the changes in this particular configuration might not have been seen in the past.

In the bottom row, we show the performance throughout the 17th March 2015 ( Saint Patrick's Day storm). The RMSE level compared to the 5 March is between two and three times higher. However, in this case, the NNGIM predictor shows on average a better performance than the cyclic frozen with variations depending on the forecast horizon. For the first hours of the day, the NNGIM predictor performs similarly to cyclic frozen, for the 6 and 24-hour horizons, improving throughout the day. An interesting behaviour is that at 48 hours the RMSE remains at low levels throughout the day, while the frozen cyclic in the early hours provides twice the error.

#### *8.2.2 Performance on 13-14 and 25-26 August 2018*

Figure 7, shows the RMSE time series for the two selected days at a time of the low activity solar cycle. On that day, the RMSE level is similar to that of the 5th of March 2015 analysed above, which was a day of low geomagnetic activity, while being in a high activity phase of the solar cycle.

On 13-14 August 2018, the NNGIM prediction is better or equal to that of the cyclic frozen, except for a brief interval on the 14th of March at a 6-hour horizon. The average improvement over the day is in the order of 25% for 6 hours, 13% for 24 hours, and 18% for 48 hours. However, there are significant fluctuations throughout the day and the slopes/error patterns vary from horizon to horizon.

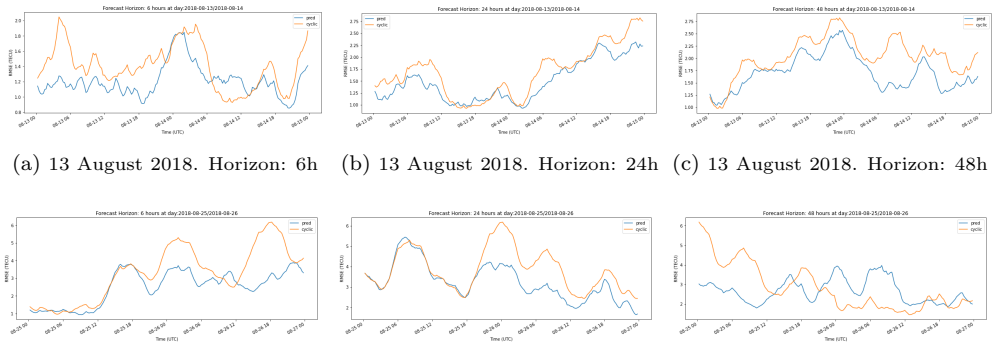


(a) 5 March 2015. Horizon: 6h (b) 5 March 2015. Horizon: 24h (c) 5 March 2015. Horizon: 48h

(d) 17 March 2015. Horizon: 6h (e) 17 March 2015. Horizon: 24h (f) 17 March 2015. Horizon: 48h

Figure 6: Comparison of the NNGIM forecast vs. frozen cyclic RMSE. Upper row: 5 March 2015 (12 days before the storm). Lower row: 17 March 2015 (the St.Patrick storm day)

On 25-26 August 2018 (storm day) for the 6- and 24-hour horizons NNGIM systematically performs better than the frozen cyclical. The performances at the 6- and 24-hour horizons are practically the same for the 25th day, while they differ significantly for the 26th day, with NNGIM being 25-50% better over long time intervals.



(a) 13 August 2018. Horizon: 6h (b) 13 August 2018. Horizon: 24h (c) 13 August 2018. Horizon: 48h

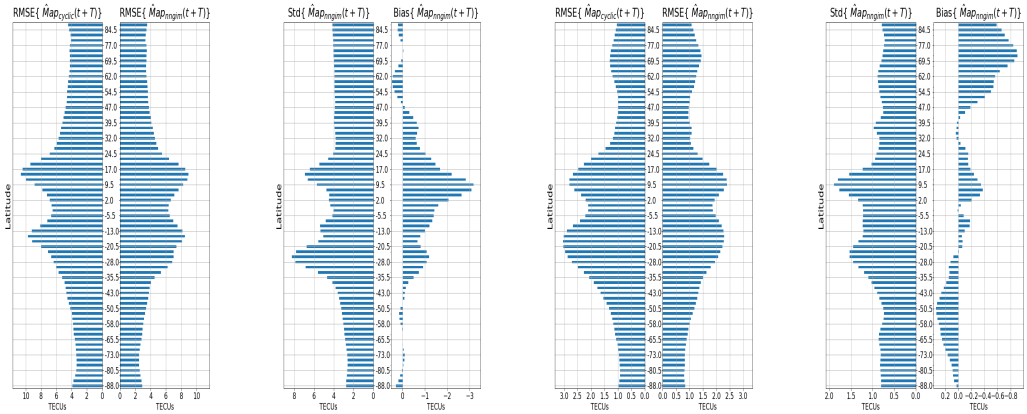
(d) 25-26 August 2018. Horizon: 6h (e) 25-26 August 2018. Horizon: 24h (f) 25-26 August 2018. Horizon: 48h

Figure 7: Comparison of the NNGIM forecast vs. frozen cyclic RMSE. Upper row: 13-14 August 2018 (12 days before the storm). Lower row: 25-26 August 2018 (storm day)

### 8.3 RMSE, Bias and Standard Deviation by latitude

In this section, we will study the relationship of RMSE with standard deviation and bias. In Figure 8, we show the performance for a horizon $T = 6$ hours. In the Figure we present by latitude a) the RMSE of the NNGIM and frozen cyclic predictions and b) the standard deviation and bias components of the NNGIM. The study period consists of the dates studied above, i.e., August 2015 and May 2018. The values were calculated on 3007 maps corresponding to 31 days, with maps every 15 min.

The first observation is that the NNGIM prediction has a lower RMSE at all latitudes on the two studied dates. The RMSE maxima are located in the case of NNGIM at the same latitude, while in the case of frozen cyclic the latitude in one case differs. On the other hand, the maxima in the standard deviation do not coincide with the RMSE maxima, noting that the difference is explained in the case of March 2015 by a very high bias at about 10 degrees north latitude. The bias of -3 TECU observed in this case is rare, in the maps observed by the author, the bias, in general, was less than 1 TECU, as illustrated in the case of August 2018.



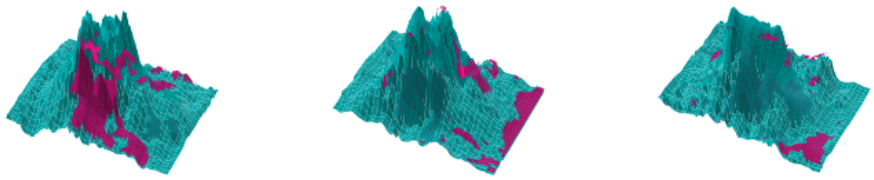

(a) RMSE March 2015 (b) Std/Bias March 2015 (c) RMSE August 2018 (d) Std/Bias August 2018

Figure 8: Performance for a horizon $T = 6$ hours. RMSE, Bias and Standard Deviation by latitude. (a) Comparison of the RMSE between the NNGIM and the frozen cyclic March 2015, (b) Standard Deviation and Bias for the NNGIM March 2015, (a) Comparison of the RMSE between the NNGIM and the frozen cyclic August 2018, (b) Standard Deviation and Bias for the NNGIM August 2018. Note that the Bias and Standard Deviation are not the same scale.

### 8.4 Reliability of the standard deviation estimated from NNGIM

In this section, we will study the reliability of the standard deviation estimated from the nearest neighbours provided by the algorithm. The purpose is to show that the standard deviation computed on the nearest future maps correctly represents the variability of the predicted map. We will show the reliability from two points of view, the first one consists of plotting several maps and showing the regions not covered by the confidence margin given by the standard deviation provided by NNGIM. The second point of view will consist in showing the decrease of the error obtained when the prediction is considered to be included within the confidence margin given by the standard deviation.

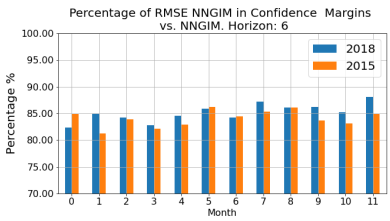
In Figure 9, we show maps for different dates for the month of June 2019, in which we mark in green the region covered by the interval $Forecast_{Map} \pm Forecast^{Std}_{Map}$, and in red the areas of the prediction that fall outside this interval. The images show that the areas of the $Forecast_{ref}$ maps not covered by a standard deviation margin are located in the periphery or at the areas of sharp transition.



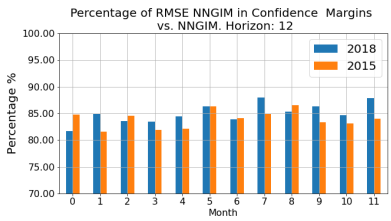
(a) Map at 2019-06-29 23:00:00 (b) Map at 2019-06-22 17:45:00 (c) Map at 2019-06-12 10:15:00

Figure 9: Areas included in the confidence margin of the Forecast map. *Green areas*: show the areas where the reference $Forecast_{ref}$ is included in $Forecast_{Map} \pm Forecast^{Std}_{Map}$. *Red areas*: areas where $Forecast_{ref}$ is outside the margin.

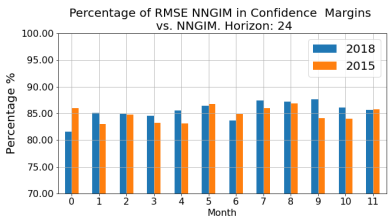
In Figure 10 we show the error decrease regarding the NNGIM prediction if we consider only data outside the interval within the confidence margin. That is, we consider the error to be zero if the predicted map is contained in the margin, i.e., $Forecast_{ref} \subset Forecast_{Map} \pm Forecast^{Std}_{Map}$. It is seen that systematically for the two years and prediction horizons, the error decreases between 15 and 20%. In other words, assuming the correct value is within the confidence interval significantly reduces the error. An interesting feature is that this error reduction does not depend on either the season of the year or the prediction horizon.



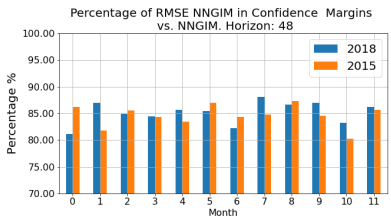

(a) Horizon 6 hours




(b) Horizon 12 hours




(c) Horizon 24 hours




(d) Horizon 48 hours

Figure 10: Performance for $Forecast_{ref} \subset Forecast_{Map} \pm Forecast^{Std}_{Map}$. Percentage of RMSE reduction with regard to cyclic freezing for the horizons of 6h, 12h, 24h, 48.

### 8.5 Validation of the method with JASON3 and CODE data.

Next we show the results of the validation of the NNGIM VTEC in terms of the differences with respect to JASON3 VTEC measurements (see Figure 11) and the comparison with other GNSS VTEC products in terms of Bias, Variance and RMS (see Figure 12).

This part of the study was conducted in the interval of the first 100 days of the year 2021. Note that for the sake of completeness of the analysis of the method, we have performed the experiments at different times of the solar cycle. Given the space limitation, we think that in this way we can provide the maximum information of the algorithm from the point of view of each issue to be evaluated. The CODE data was downloaded from the NOAA website *Code Data* (n.d.).

The comparison was made between the products based on NNGIM prediction at 24 hours (UN4G) and 48 hours (UN8G), vs. IGSG and Center for Orbit Determination in Europe (CODE) VTEC prediction model products, at 24 hours (C1PG) and 48 hours (C2PG).

In Figure 11, we show the histogram of the VTEC residual defined as $\delta V = VTEC_{JASON3} - VTEC_{ForecastGIM}$ on a logarithmic scale to enhance the details in the low-density parts of the histogram, i.e., regions where the number of samples per bin is much lower than at the mode of the distribution. For comparison purposes on the figure, there is a summary of the relevant statistics of each product, i.e., bias, standard deviation, and RMS. Note that the Std. Dev and RMS of the NNGIM prediction at 24 hours (UN4G) and 48 hours (UN8G) are systematically lower than the CODE and IGSG. Note that the tails of the distributions are similar. Also the distribution related to the NNGIM product having a lower width compared with the CODE products. This indicates that the probability of a high-value positive error in the NNGIM products is much lower than the other products.

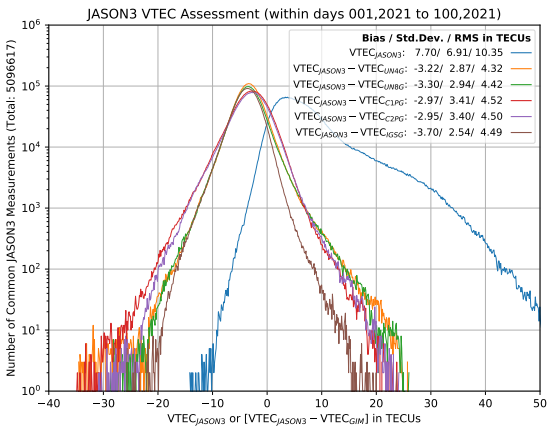


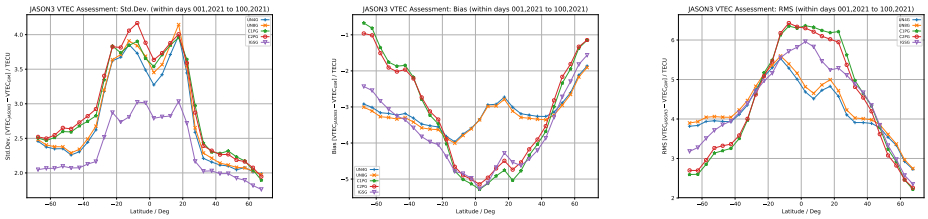

Figure 11: Histogram, in log scale for the number of counts, of VTEC difference of JASON3 measurement minus GIMs value for the first 100 days of 2021, the color code indicates the comparison for different forecasting products. The histogram of the reference values of JASON3 is represented in gray. The corresponding overall bias, standard deviation (Std.Dev.), and RMS are indicated in the upper right legend.

Next, we will compare, concerning the JASON3 measurements, the products by latitude, as a function of the differences in standard deviation, bias, and RMS.

In Figure 12, on the left, we show the standard deviation of the VTEC residual vs. JASON3 at 5-degrees longitudinal intervals. Note that the standard deviation is weighted by the number of JASON3 observations in cells in the same 5-degree latitude range. The 24-hour prediction product based on NNGIM, UN4G consistently has a lower standard deviation than the equivalent CODE, C1PG except for the sample at 15 degrees latitude north where they are the same. The largest differences are observed at the equator and in areas of north/south latitude greater than 35 degrees. In the case of the 48-hour forecast products (UN8G vs. C2PG), the trend is very similar, with NNGIM having a lower standard deviation at all latitudes except at 15 degrees north latitude.

In Figure 12, in the center, we show the bias of the products. In this case, the bias of the NNGIM products is lower, except in the region below -35 degrees south latitude and above 45 degrees north latitude. The explanation for this bias corresponds to the fact that there is a different ionosphere sampling model, as explained in Yang et al. (2021).

Finally, in Figure 12, on the right, we show the RMS value by latitude, in this case, the RMS of the prediction is better for the NNGIM products between -30 degrees south latitude and 50 degrees north latitude. Note that from 50 degrees north latitude the difference concerning CODE is less than half a TECU, and on the other hand in the equatorial region the UN4G and UN8G products provide an improvement of 2 TECUs. The difference in the south polar region could be because there are fewer stations, and therefore the GIMs are less accurate.




(a) Standard deviation (b) Bias (c) RMS

Figure 12: Jason assessment for latitudinal zones, the color representing different products. Note that the measures are weighted by the number of JASON3 observations in cells with the same 5-degree intervals of latitude

Note that the availability of the NNGIM forecasting depends on the delay of generating the GIM maps, which is the case of the UPC-IonSAT is of about half an hour, while the availability of the CODE maps can be with a delay of up to 5 or 7 hours, which makes the effective forecasting horizon shorter.

### 8.6 Considerations about the quality assessment by means of JASON3 VTEC measurements

The importance of the VTEC measures obtained by JASON3 lies in the fact that it provides us with an objective reference of the real value for the comparison purposes. The measures provided by JASON3, allow us to determine whether the estimate made by the prediction product provides a correct value or introduces biases. As the orbit altitude of JASON3 is about ∼1300 km, the altimeter can count almost all the VTEC of

the ionospheric state above the ocean region. It is important to emphasize that over the ocean areas, the GIM used for the prediction might have large interpolating errors appear due to their far distance from GNSS ground stations. Therefore the use of JASON3 VTEC measurement allows for a critical evaluation of the forecast products in adverse circumstances. In this work, the raw observations of the JASON3 VTEC were preprocessed to reduce the measurement noise. The process carried out included the use of a temporal sliding window, removal of outliers, and so on, as explained in Hernández-Pajares et al. (2017) and Roma-Dollase et al. (2018).

Evaluation using dSTEC may be an alternative for evaluating VTEC values of GIM prediction products. However in this particular case the use of dSTEC may not be appropriate because of the following. Typically, the JASON3 VTEC assessment is a validation method for GIMs only over the ocean region, so it may be appropriate to consider the complementary assessment for GIMs over the land region, namely the dSTEC assessment, which compares the difference between the observed STEC along the phase-continuous satellite-station arc and the calculated STEC from GIM, see details in Hernández-Pajares et al. (2017). However, the usage of altimeter VTEC measurements to assess GIMs has been proven to be a good external assessment procedure, consistent with other methods based on GNSS data (behaving similarly to the dSTEC test, Hernández-Pajares et al. (2017)) but independent from GNSS and globally distributed. These are the main reasons behind focusing on altimeter data, being the JASON3 the one available during the whole period of analysis, see the former studies that used JASON2, JASON1, and TOPEX altimeters.

## 9 Conclusion

In this work, we have introduced a method to predict GIMs at various horizons based on the Nearest Neighbour technique. This technique allows to implement predictors without the need to train a model, and the computation time is small. The assumption on which the model is based is that a database covering more than one solar cycle is available, and that the geomagnetic conditions affecting the current map have somehow happened in the past, and that similar geomagnetic effects are distributed among several maps, whose linear combination allows a better approximation of the prediction. An advantage of the method is also that from the similar maps found in the historical database, a confidence margin can be created. The prediction using this confidence margin allows a significant decrease in the prediction error. We have performed a real-time implementation. The computational cost of adding a prediction horizon is very low, so in the implementation, predictions are made with almost no additional cost for arbitrary horizons. The prediction results improve compared to the frozen cyclic up to a 48-hour horizon, which seems to be a natural barrier for this method. Finally the method has been assessed in different moments of the solar cycle, taking into account days with storm and without significant geomagnetic perturbations. Also the method has been assessed by comparing with the forecast at 24 and 48 hours of the Center for Orbit Determination in Europe (CODE) prediction model products.

**Acknowledgments**

This work was been partially supported by the project PID2019-107579RB-I00 (MICINN) and done in the context of PITHIA NRF EU project. The UQRG is openly accessible from IGS server (`https://cddis.nasa.gov/archive/gnss/products/ionex/YEAR/DOY/uqrgDOY0.YYi.Z`) and from UPC server (`https://chapman.upc.es/tomion/rapid/YEAR/DOY_YYMMDD.15min/uqrgDOY0.YYi.Z`) where YEAR and YY the four- and two-digit year identifiers, MM is month number, DD is day of month, and DOY is the day of year. Any missing file can be requested from the authors, in particular from Enric Monte Moreno (enric.monte@upc.edu).

## Appendix: Example of forecasts at several horizons

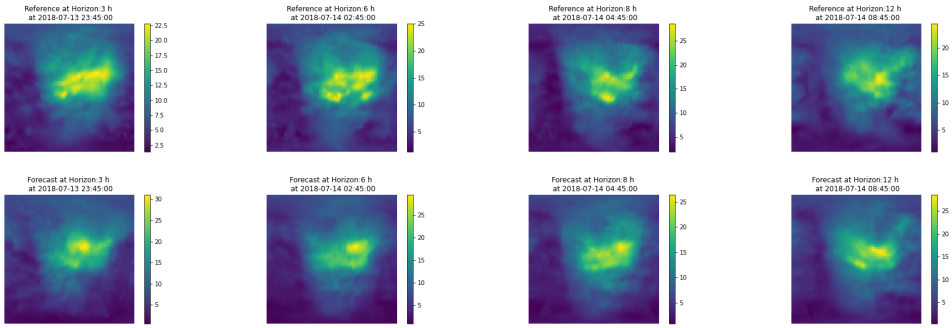



Figure 13: Selected sequence of predictions for the map at 2018-07-14 20:45:00 UT. The upper row shows the reference to 3h, 6h, 8h, and 12h horizons, the second row shows the prediction result. Note that the color bars are not at the same scale.

In Figures 13 and 14 we show a selected sequence of predictions for the map at 2018-07-14 20:45:00 UT, at horizons ranging from 3h to 48h. In the first row we show the reference to 3h, 6h, 8h, and 12h horizons, and in the second row we show the prediction result. The third and fourth rows show the results for horizons of 16h, 20h, 24, 48h. In order to assess the results it has to be taken into account that the colour bars are not at the same scale. This means that local maxima can distort the level of the overall colour gradation. In any case, an indication of the effectiveness of the algorithm lies in comparing the medium/high ionisation regions (not maxima) between reference and prediction. In these cases, the shape of the regions is found to be similar.

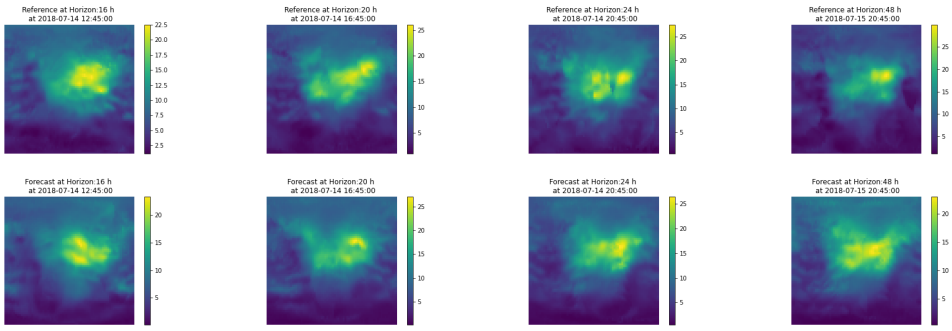



Figure 14: Selected sequence of predictions for the map at 2018-07-14 20:45:00 UT. The upper row shows the reference to 16h, 20h, 24, 48h, the second row shows the prediction result. Note that the color bars are not at the same scale.

## Appendix: Kp number and Solar flares in the analysed days

In this section we present the time distribution of geomagnetic and solar flare activity indices (i.e., Kp and solar flares occurrences) that can be used to study the consistency of the patterns that appear in the temporal distributions of the RMSE forecast error of the method. The data was obtained from internet at *Space Weather Live* (n.d.).

Table 4: Hourly Kp and Solar Flares for the 5 March 2015 (day 64)

Hourly Kp values

| Hour | 00-03h | 03-06h | 06-09h | 09-12h | 12-15h | 15-18h | 18-21h | 21-00h |
|---|---|---|---|---|---|---|---|---|
| Kp | 1 | 0 | 0 | 1 | 2 | 2 | 2 | 1 |

.

Solar Flares 5 March 2015

| Flare | C1.3 | C3.5 | M1.2 | C4 | C1.9 |
|---|---|---|---|---|---|
| Start | 04:13 | 08:46 | 17:06 | 19:35 | 22:45 |
| Maximum | 04:19 | 09:47 | 18:11 | 19:55 | 22:59 |
| End | 04:26 | 10:02 | 18:26 | 20:04 | 23:06 |

Table 5: Hourly Kp and Solar Flares for the 17 March 2015 (days 75,76)

Hourly Kp values

| Hour | 00-03h | 03-06h | 06-09h | 09-12h | 12-15h | 15-18h | 18-21h | 21-00h |
|---|---|---|---|---|---|---|---|---|
| Kp (17 March) | 2 | 5 | 6 | 6 | 8 | 8 | 7 | 8 |

.

Solar Flares 16 March 2015

| Flare | Start | Maximum | End |
|---|---|---|---|
| B8 | 04:17 | 04:21 | 04:24 |
| B6.4 | 07:21 | 07:25 | 07:31 |
| B8.7 | 08:18 | 08:23 | 08:28 |
| C1.8 | 08:33 | 09:52 | 09:59 |
| C1.9 | 09:38 | 09:52 | 09:59 |
| B9.6 | 10:16 | 10:20 | 10:24 |
| M1.6 | 10:39 | 10:58 | 11:17 |
| C1.1 | 12:55 | 12:59 | 13:01 |
| C2.8 | 13:49 | 13:54 | 13:59 |
| B8.7 | 17:55 | 17:59 | 18:01 |
| B6.2 | 18:42 | 18:45 | 18:47 |
| C5.5 | 20:12 | 20:15 | 20:20 |
| C8.1 | 20:38 | 20:49 | 21:00 |
| Solar Flares 17 March 2015 | | | |
| C1.9 | 01:45 | 01:52 | 01:54 |
| C1.1 | 21:14 | 21:19 | 21:25 |
| M1 | 22:49 | 23:34 | 23:48 |

Table 6: Hourly Kp and Solar Flares for the 13,14 August 2018 (days 225, 226)

Hourly Kp values

| Hour | 00-03h | 03-06h | 06-09h | 09-12h | 12-15h | 15-18h | 18-21h | 21-00h |
|---|---|---|---|---|---|---|---|---|
| Kp (13 August) | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| Kp (14 August) | 2 | 1 | 1 | 1 | 1 | 0 | 0 | 2 |

.

Solar Flares 13 August 2018

| | | | | | | |
|---|---|---|---|---|---|---|
| Flare | B3.3 | B3.6 | B5.8 | B2.9 | B4.1 | B4.3 |
| Start | 02:00 | 08:34 | 10:30 | 13:39 | 17:24 | 17:51 |
| Maximum | 02:04 | 08:38 | 10:45 | 13:53 | 17:27 | 18:03 |
| End | 02:07 | 08:51 | 11:00 | 13:57 | 17:32 | 18:10 |

.

Solar Flares 14 August 2018

| | | | | | | |
|---|---|---|---|---|---|---|
| Flare | C1.1 | C1.1 | C1.9 | C1.6 | B4.4 | B5.2 |
| Start | 00:26 | 00:51 | 01:55 | 03:00 | 05:30 | 07:32 |
| Maximum | 00:33 | 01:04 | 02:00 | 03:04 | 05:38 | 07:35 |
| End | 00:37 | 01:10 | 02:04 | 03:08 | 05:40 | 07:37 |

Table 7: Hourly Kp and Solar Flares for the 25-26 August 2018 (day 238)

Hourly Kp values

| Hour | 00-03h | 03-06h | 06-09h | 09-12h | 12-15h | 15-18h | 18-21h | 21-00h |
|---|---|---|---|---|---|---|---|---|
| Kp (25 August) | 1 | 1 | 2 | 2 | 3 | 2 | 4 | 4 |
| Kp (26 August) | 5 | 7 | 7 | 5 | 5 | 6 | 5 | 3 |

.

Solar Flares 25 August 2018

| Flare | Start | Maximum | End |
|---|---|---|---|
| B7.6 | 01:40 | 01:46 | 01:54 |
| B9.2 | 02:00 | 02:03 | 02:06 |
| B8.2 | 02:12 | 02:18 | 02:20 |
| C3.6 | 02:35 | 02:42 | 02:47 |
| B8.7 | 03:26 | 03:30 | 03:32 |
| C1.1 | 04:14 | 04:20 | 04:24 |
| B8.5 | 05:46 | 05:50 | 05:52 |
| C4.3 | 06:18 | 06:31 | 06:37 |
| C2.3 | 07:58 | 08:03 | 08:05 |
| C1.7 | 10:23 | 10:34 | 10:41 |
| C2.3 | 11:55 | 11:59 | 12:03 |
| C1.3 | 12:33 | 12:38 | 12:44 |
| B9.5 | 13:56 | 14:02 | 14:05 |
| C1.1 | 14:48 | 14:53 | 14:58 |
| B9.7 | 15:37 | 15:40 | 15:43 |
| C1 | 17:14 | 17:17 | 17:22 |
| C1.3 | 18:01 | 18:05 | 18:08 |
| C2.2 | 19:23 | 19:27 | 19:31 |
| B8.6 | 19:47 | 19:50 | 19:53 |
| B9.1 | 22:01 | 22:17 | 22:28 |
| C2.8 | 23:40 | 23:53 | 00:04 |

.

Solar Flares 26 August 2018

| Flare | Start | Maximum | End |
|---|---|---|---|
| C1.5 | 02:56 | 03:01 | 03:04 |
| C9.5 | 13:41 | 13:53 | 14:20 |
| C5 | 14:51 | 15:07 | 15:13 |
| C1.7 | 19:20 | 19:22 | 19:24 |
| C1.7 | 19:32 | 19:43 | 19:50 |
| C1.3 | 20:22 | 20:28 | 20:33 |

## References


Cherrier, N., Castaings, T., & Boulch, A. (2017). Forecasting ionospheric total electron content maps with deep neural networks. In *Proc. conf. big data space (bids), esa workshop.*

*Code data.* (n.d.). Accessed on June 2 2021. [Online]. Retrieved from `ftp://ftp.nodc.noaa.gov/pub/data.nodc/`

García-Rigo, A., Monte, E., Hernández-Pajares, M., Juan, J., Sanz, J., Aragón-Angel, A., & Salazar, D. (2011). Global prediction of the vertical total electron content of the ionosphere based on gps data. *Radio science*, *46*(06), 1–3.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning.* MIT press.

Hernández Pajares, M. (2014). Inputs received from ionospheric research groups on activities related with rt global electron content determination. In *Igs rt wg splinter meeting* (pp. 1–15).

Hernández-Pajares, M., Juan, J., & Sanz, J. (1999). New approaches in global ionospheric determination using ground gps data. *Journal of Atmospheric and Solar-Terrestrial Physics*, *61*(16), 1237–1247.

Hernández-Pajares, M., Juan, J., Sanz, J., & Colombo, O. L. (2000). Application of ionospheric tomography to real-time gps carrier-phase ambiguities resolution, at scales of 400–1000 km and with high geomagnetic activity. *Geophysical Research Letters*, *27*(13), 2009–2012.

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.

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.

Liu, L., Zou, S., Yao, Y., & Wang, Z. (2020). Forecasting global ionospheric tec using deep learning approach. *Space Weather*, *18*(11), e2020SW002501.

Monte Moreno, E., García Rigo, A., Hernández-Pajares, M., & Yang, H. (2018). Tec forecasting based on manifold trajectories. *Remote Sensing*, *10*(7), 988.

*Nngim forecasts at different horizons.* (n.d.). Accessed on June 2 2021. [Online]. Retrieved from `http://chapman.upc.es/.trial_urtg/tomion/forecast_nngim/quick/last_results/`

Omohundro, S. M. (1989). *Five balltree construction algorithms.* International Computer Science Institute Berkeley.

Orús, R., Hernández-Pajares, M., Juan, J., & Sanz, J. (2005). Improvement of global ionospheric vtec maps by using kriging interpolation technique. *Journal of Atmospheric and Solar-Terrestrial Physics*, *67*(16), 1598–1609.

Roma-Dollase, D., Hernández-Pajares, M., Krankowski, A., Kotulak, K., Ghoddousi-Fard, R., Yuan, Y., ... others (2018). Consistency of seven different gnss global ionospheric mapping techniques during one solar cycle. *Journal of Geodesy*, *92*(6), 691–706.

Roma Dollase, D., López Cama, J. M., Hernández Pajares, M., García Rigo, A., et al. (2015). Real-time global ionospheric modelling from gnss data with rt-tomion model. In *5th international colloquium scientific and fundamental aspects of the galileo programme* (pp. 27–29).

*Space weather live.* (n.d.). Accessed on June 2 2021. [Online]. Retrieved from `https://www.spaceweatherlive.com/`

Wang, C., Xin, S., Liu, X., Shi, C., & Fan, L. (2018). Prediction of global ionospheric vtec maps using an adaptive autoregressive model. *Earth, Planets and Space*, *70*(1), 1–14.

Wang, L., Zhang, Y., & Feng, J. (2005). On the euclidean distance of images. *IEEE transactions on pattern analysis and machine intelligence*, *27*(8), 1334–1339.

Weber, G., Dettmering, D., & Gebhard, H. (2005). Networked transport of rtcm via internet protocol (ntrip). In *A window on the future of geodesy* (pp. 60–64). Springer.

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.

Yang, H., Monte-Moreno, E., Hernández-Pajares, M., & Roma-Dollase, D. (2021). Real-time interpolation of global ionospheric maps by means of sparse representation. *Journal of Geodesy*, *95*(6), 1–20.