



Classifying continuous GNSS stations using integrated machine learning

Nhung Le^{1,2,3} · Benjamin Männel¹ · Luyen K. Bui^{4,5} · Harald Schuh^{1,2}

Received: 20 May 2024 / Accepted: 29 November 2024 / Published online: 19 December 2024
© The Author(s) 2024

Abstract

The development of Global Navigation Satellite Systems (GNSS) results in large spatial geodetic networks with a distinct range of accuracy. Thus, classification of the GNSS stations is needed to determine which stations are appropriate for geodetic applications. Additionally, advanced Machine Learning (ML) techniques have been proposed. However, ML algorithms may sometimes be less sensitive due to a lack of samples or anomalies in input data. Therefore, this study introduces an approach in which human-based supervision is integrated into ML processes to improve the ML model's performance in classifying the continuous GNSS stations. The human factor influences the ML processes through two sampling strategies: “*suggest-decide*” and “*correct-retrain*”, where the accuracy of ML models will be improved via human-based corrections. The idea is that the unsupervised ML-based clustering techniques are driven by human-based supervision to create samples for training the supervised ML-based classification models. In this study, we develop a MATLAB app to automate the clustering and labeling processes. Our finding demonstrates that applying these sampling strategies can enhance the accuracy of the ML-based classification models from under 50 % up to ~99 % after re-training. Also, this study categorizes almost 9000 continuous monitoring stations in the Nevada database, of which 1900 stations in Europe serve as samples for training the ML-based classification models. Furthermore, the methodologies developed in this study can be applied to warning systems, which utilize internal and external human resources to correct errors, address unusual situations, and provide timely feedback for better performance of ML-based forecasts.

Keywords GNSS · GNSS station classification · Machine learning · Integrated machine learning · Human and machine learning combination

Introduction

In the 1930s, the concept of a machine being able to think and learn like a human began to take shape (Ross 1933; Silaparasetty 2020). As a realization of this idea, Artificial

Intelligence (AI) was first pronounced in 1956 and later quickly evolved into the global scientific and technological revolution (Silaparasetty 2020). Together with data science, AI has been developed and plays an essential role in data mining, such as modeling, clustering, classification, and anomaly detection (Kantardzic 2020; Bali et al. 2022). Also, advanced algorithms have been suggested to train AI (Mazinan 2013; Carrino and Giove 2021; Hagendorff 2021; Moura et al. 2021). However, despite operating on supercomputers, AI is still limited in some aspects; especially, errors and biases in decision-making (Welser and Osoba 2017), or potential risks to society and cybersecurity, like deepfakes (Schick 2020; Kalpokas and Kalpokiene 2022), lie AI bots (Braun et al. 2020), and out of control of Super AI (Alfonseca et al. 2021; Choi 2021). Thus, to enhance AI's performance, the notion of combining humans and Machine Learning (ML) emerged in various applications, e.g., image segmentation, entertainment, biomedicine, or

✉ Nhung Le
nhung@gfz-potsdam.de

¹ GFZ German Research Centre for Geosciences, Potsdam, Germany

² Technische Universität Berlin, Berlin, Germany

³ Hanoi University of Natural Resources and Environment, Hanoi, Vietnam

⁴ Faculty of Geomatics and Land Administration, Hanoi University of Mining and Geology, Hanoi, Vietnam

⁵ Innovations for Sustainable and Responsible Mining (ISRM) Research Group, Hanoi University of Mining and Geology, Hanoi 100000, Vietnam

robots (Amershi et al. 2014; Natsume et al. 2018; Hagen-dorff 2021; Arslan et al. 2022). The synergy of natural and artificial intelligence has been optimized through human-based supervision to correct AI-made errors. Motivated by this research stream, we implement the Human and Machine learning (H&M) combination approach in the applied geodetic domain to address a specific topic of classifying continuous Global Navigation Satellite Systems (GNSS) stations. Thus, the aim is to concretize an existing approach for the first time in the geodetic field.

As a part of space geodesy, GNSS is widely employed not only in general geodetic applications but also in deformation tracking, which enables determining motions in 3-dimensional space with high accuracy and integrating flexibly with other techniques (Saria et al. 2013; Kreemer et al. 2014; Yu et al. 2020; Serpelloni et al. 2022). Also, a significant increase in worldwide GNSS stations over the last decades has yielded high-space-resolution networks (Vollerthun and Wieser 2002; Ge et al. 2006; Wang et al. 2022). These stations cover an extensive range of accuracy (Altamimi et al. 2013). Moreover, the instability of reference stations and the accuracy inequivalence of monitoring stations can lead to bias in deformation analyses due to systematic errors in observations obtained from these stations (Bastos et al. 2014; Karaim et al. 2028). Hence, it is necessary to classify GNSS stations for two main purposes: (1) identifying high-precision reference stations and (2) selecting appropriate monitoring stations for better interpretations of deformation information.

So far, only ~400 stations from the EUREF Permanent GNSS Network (EPN¹) in Europe have been classified into seven orders and provided by the Royal Observatory of Belgium (ROB). Their classification tool² is programmed based on a traditional approach that uses statistical criteria and thresholds (Bruyninx et al. 2019; Legrand and Bruyninx 2020, 2021; Legrand 2022). It is designed to specialize in categorizing the EPN-GNSS network with the fixed criteria³. Thus, applying ML will be an alternative solution that can perform flexibly on datasets and criteria without adjusting algorithms or core codes of tools. However, one challenge arises in applying this ML approach: the 400 samples derived from these EPN-GNSS stations for training ML models are not yet big enough, resulting in low classification performance with <50 %. Further details will be demonstrated in Sect. 3.3. Hence, overcoming the limitation in

sample size promotes the actualization of the human and ML combination approach in this study.

Below is the implementation of the H&M approach, where both the unsupervised and supervised ML techniques are employed to cluster data and train ML models. The classification performance of the ML models will be enhanced step by step via two sampling strategies: “*suggest-decide*” and “*correct-retrain*”, in which human-based supervision will be integrated into the ML processes. We structure this article as follows. Section 2 presents experimental data and methodologies. Section 3 shows results and discussions about the sampling, clustering, and classification techniques in ML. Finally, conclusions and outlook are summarized in Sect. 4.

Data and methodologies

Data

We obtain the GNSS coordinate time series from 8975 monitoring stations in the Nevada database⁴ (Blewitt et al. 2018) to generate features for classifying the GNSS station quality (Fig. 1a). These stations were observed continuously from January 1994 to December 2022. The longest continuous monitoring period is 29.5 years with 10729 daily coordinate solutions, and the shortest is 2.0 years with 688 daily solutions. Of those, we select 1900 European stations as samples for training the ML-based classification models, with an average monitoring timespan equal to 12.7 years (Fig. 1b). These European stations are selected in this study so that comparative investigations can be conducted based on ~400 EPN⁵ stations, which have been classified by the Royal Observatory of Belgium.

In addition, we use the auxiliary data, including seismic activities (Fig. 2) and GNSS metadata (e.g., antenna changes and firmware installation), to detect anomalies and correct discontinuities in the GNSS coordinate time series. These data are accessible at the United States Geological Survey (USGS⁶), GEOFON⁷, and Semisys (Bradke 2020) archives. Please notice that the earthquake occurrence frequency is currently denser, especially after 2010 (Fig. 2b, d). According to NASA⁸, climate changes can trigger earthquakes (i.e., weather earthquakes). Another reason is the increasing number of global seismological stations and advanced

¹ <https://epncb.oma.be/>

² https://www.epncb.oma.be/_productsservices/ReferenceFrame/Tool.php

³ https://www.epncb.oma.be/_productsservices/ReferenceFrame/Station_Classification.php

⁴ <http://geodesy.unr.edu/>

⁵ https://www.epncb.oma.be/_networkdata/stationlist.php

⁶ <https://www.usgs.gov/>

⁷ <https://geofon.gfz-potsdam.de/>

⁸ <https://climate.nasa.gov/news/2926/can-climate-affect-earthquakes-or-are-the-connections-shaky/>

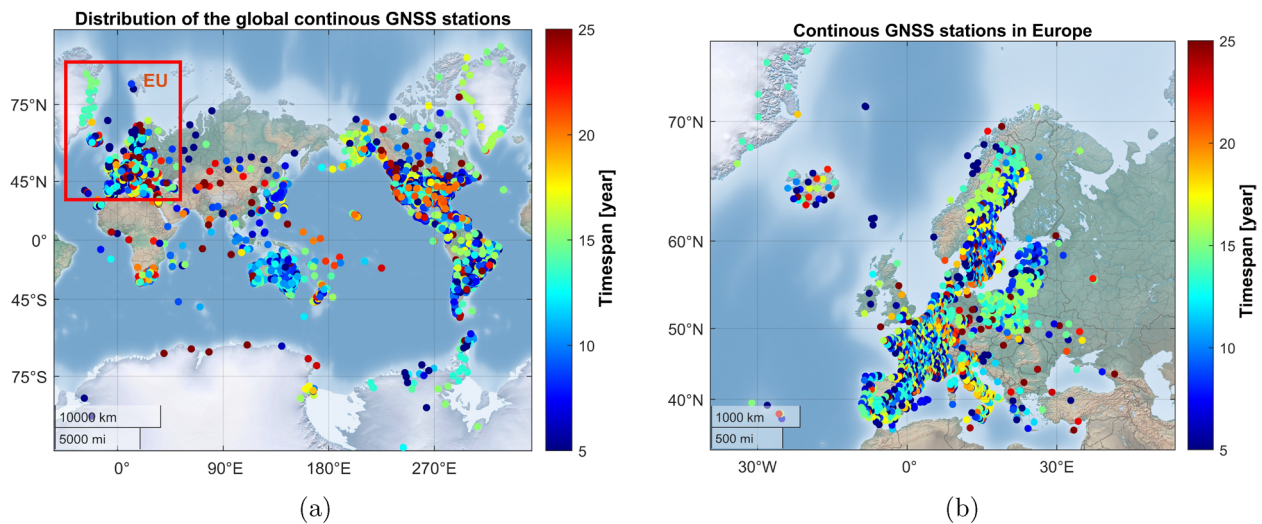


Fig. 1 Almost 9000 globally continuous GNSS stations obtained from the Nevada database were investigated in this study (a), in which 1900 European GNSS stations were used as samples to train the ML models (b)

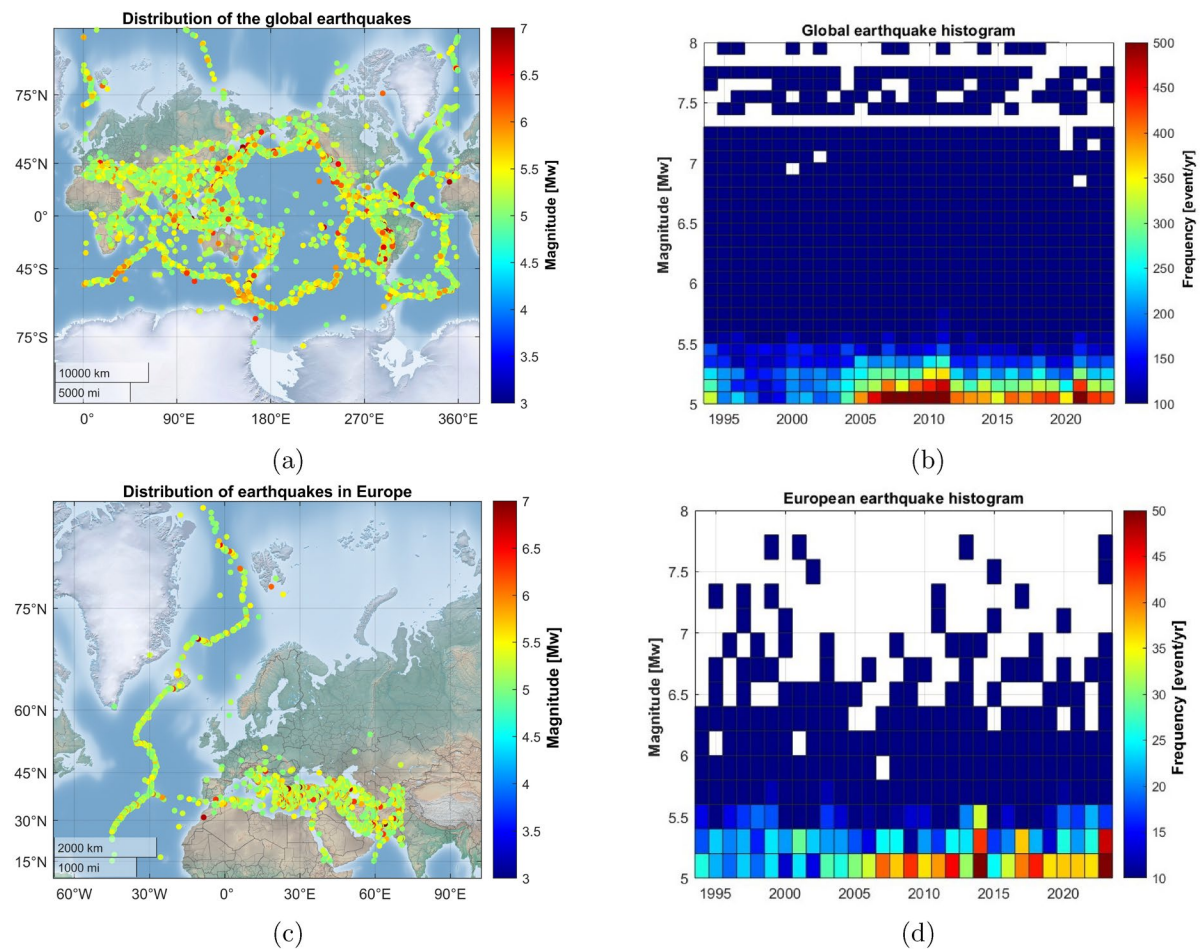


Fig. 2 The magnitude and frequency of the global earthquakes (a & b) and in the European region (c & d)

monitoring techniques, enabling the tracking of more seismic events (Li 2021). These will predict more GNSS stations might be found in the effect radii of earthquakes after 2010. For example, on average, ~1800 earthquakes (≥ 5 Mw) per year were recorded globally after 2010, compared to only ~1500 earthquakes per year before 2010. A similar data pattern was observed in European countries, where earthquake events were recorded, with 96 events of ≥ 5 Mw per year after 2010, compared to only 76 events per year before 2010.

Methodologies

The workflow of classifying GNSS stations is divided into three main steps: preprocessing, clustering, and training ML models (Fig. 3). In the first step, the GNSS coordinate time series will be processed to gain features for training ML. The outputs, including monitoring timespans, velocity vectors, accuracy, discontinuities, and outlier records, together with the auxiliary data, will be tested to determine the best-fit features. The GNSS time series processing techniques in the preprocessing step have been addressed in our previous study (Le et al. 2024). Thus, this study will describe in detail the methods to classify the GNSS stations based on the H&M approach. Human interactions will influence ML processes by setting statistical thresholds (e.g., cleaning and

clustering data), selecting features, and tuning hyper-parameters (e.g., cross-validations, hidden layers, and iterations).

Clustering data is conducted in the second step to create samples for training the ML-based classification models. The unsupervised ML can cluster the data, but it assigns labels arbitrarily for these clusters. Hence, the sampling strategy of “*suggest-decide*” will be implemented. Here, depending on the data characteristics, human-based supervision is to label the clusters. More explicitly, ML “*suggests*” on how the data is clustered, while humans “*decide*” on which classes these clusters belong to based on users’ experience or knowledge of stations. Moving clusters into actual classes can be performed manually or automatically but must be under human-based supervision. Labels are clustered manually by examining the outcomes and recording them on datasheets. On the other hand, the automatic approach is based on the distance from the origin to the centroid of the clusters. Based on the characteristics of features and the origin-centroid distances, we then rearrange the order of clusters. In our application, together with error correction and decision-making, hyper-parameter tuning is also done by humans, as usually done when working with machine learning (Fig. 3). This step finishes by categorizing the sample GNSS stations, i.e., clustering and labeling samples.

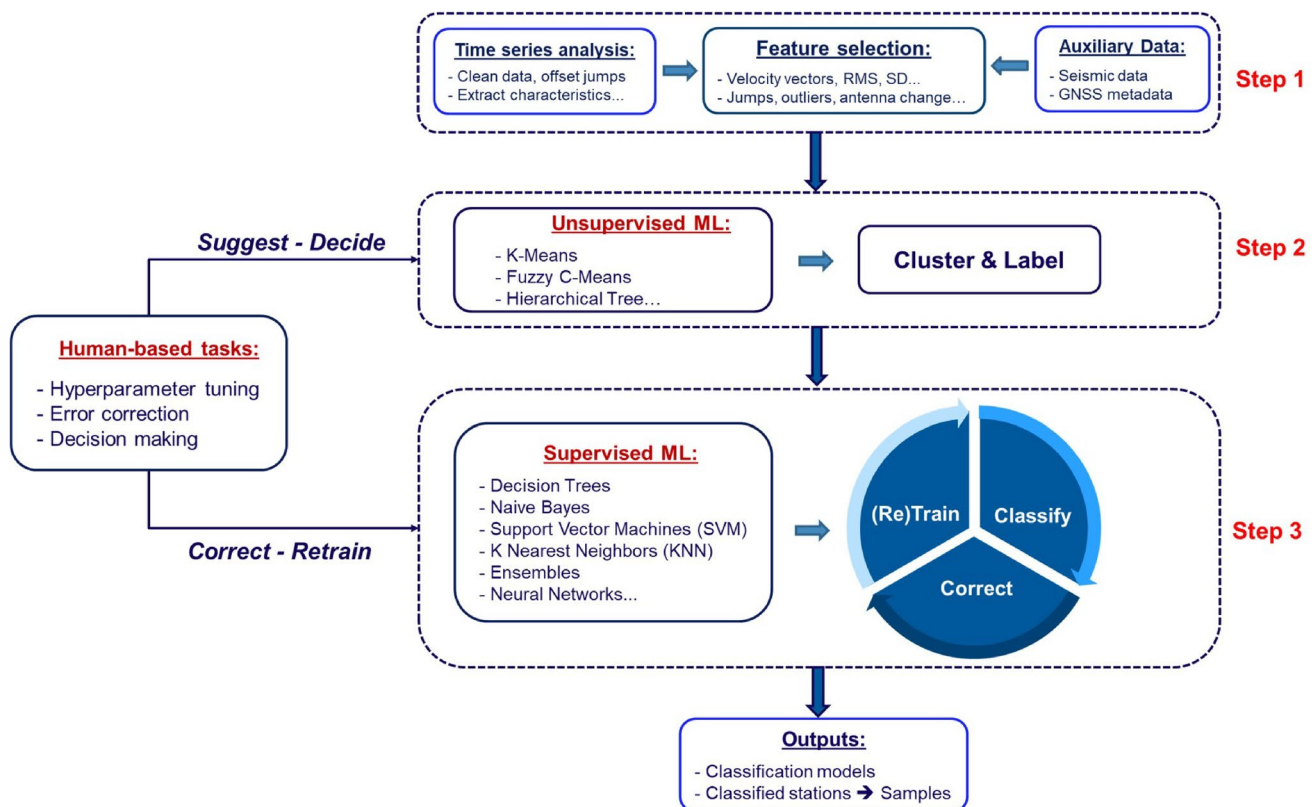


Fig. 3 Flowchart of the GNSS station classification using the H&M approach

We develop a MATLAB application named “GNSS_Cluster” to automate both tasks: clustering and labeling (Appendix A). Five unsupervised ML methods: K-means, Fuzzy C-means, Hierarchy, Spectrum, and Gaussian, have been coded in the App, which are among the most commonly used classification methods (e.g., Shi and Malik 2000; Kanungo et al. 2002; Nayak et al. 2015; Gordon 2018; Villacampa-Calvo et al. 2021). Additionally, based on our previous investigations of ~1300 EU-GNSS stations in the GFZ-GNSS database (Le et al. 2022), K-means, Fuzzy C-means, and Hierarchical clustering are best suited for clustering data with evenly distributed stratification, while the Spectral and Gaussian methods are more effective for detecting anomalies and outliers. Therefore, this study focuses on three methods: K-means, Fuzzy C-means, and Hierarchical clustering. GNSS users can set hyper-parameters for ML models, shown on the left-hand side of the App’s interface (Appendix A). The options of the clustering models are also found on the right-hand side, corresponding to Tab “Network architecture” (Appendix B). The “Sort_data” option enables users to re-order input data and assign clusters’ labels as desired. In the case users do not alter input settings, the “default” mode will be executed. For example, the “default” mode of “Sort_data” is “Off”, where the Boolean value (so-called logical variable) is equal to “0” or “False”. That means the data structure will remain as the original input, and the clusters will not be sorted. Optimization processing, considering both accuracy and convergence speed, is applied to determine the convergence point (Appendix C). To achieve this, iterative loops are performed to identify the convergence point that offers an optimal solution with high reliability and rapid outcomes. The determination of the convergence point involves counting the number of consecutive iterations executed (using a “moving window”) where the loss percentage after each loop remains below the “convergence” threshold. The classification of GNSS stations into groups is determined by analyzing the confidence interval and loss percentage for each cluster. The GNSS stations are assigned to the cluster with the highest confidence level (expressed in percentage). In this approach, we can also mitigate human biases in setting hyper-parameters for the ML processes. Further details will be described in Sect. 3.2.

The last step involves building ML-based classification models and categorizing ~9000 GNSS stations in the Nevada database. These outcomes can also be considered as samples to serve similar tasks. This study categorizes the Nevada-GNSS stations into three classes, with Class 1 representing the highest quality and Class 3 the lowest. The second sampling strategy of “correct-retrain” is involved in this step for better performance of the ML models. Human intervention is required to correct the stations assigned to inappropriate classes, and then the ML models are retrained to achieve better accuracy. That means the outputs of the

previous loop will become the inputs of the next loop. In this way, the sample size for training ML models increases after each loop, depending on the number of stations utilized. In this study, 1900 stations are employed in the training phase, after which the trained model is applied to the remaining 7075 stations. Should additional data become available in the future, subsequent work will involve training the models with the full set of 9000 stations, and the newly added stations will be adopted in the application phase. Alternatively, the entire workflow can be applied to the combined dataset of existing and newly updated data via two or three loops. As a result, the accuracy of the ML models will be enhanced step by step. More details will be shown in Sect. 3.3.

Results and discussions

Feature selection

Forty-six parameters are obtained from the ~9000 GNSS time series processing in the three-dimensional coordinate system, e.g., observation frequencies, outliers, discontinuities, variations, velocities, Root Mean Square Errors (RMSE), Standard Deviations (SD), and entropy of the GNSS time series are regarded as features. Figure 4 shows four specific features used for training ML models. On average, each station coordinate time series has 13 outliers per year (Fig. 4a). Outliers are potentially linked to the observation tracking state at the GNSS stations (e.g., weather conditions or the stability state of measurement devices). Thus, they will be considered as a criterion for the GNSS station classification. We would note that the number of identified outliers depends on the filtering thresholds and the GNSS coordinate solutions. For this reason, all observations should be processed consistently on the same statistical threshold and GNSS coordinate solution.

This study divides discontinuities in the GNSS time series into two types: (1) those caused by measurement devices, e.g., antenna changes, and (2) those associated with seismic activities, e.g., earthquakes. The GNSS metadata obtained from the Nevada database is used to verify the causes of discontinuities. In ~9000 global stations, there are ~4900 stations with 17924 discontinuities in all three coordinate components of the time series (Fig. 4b). Among that, 1596 stations are identified within the earthquake-perception radii. Please see our previous study in Le et al. (2024) for further details on the method to detect discontinuities and how to distinguish the two types of discontinuities.

We extract four parameters, including the amplitude of the annual variations, the number of extreme values in the GNSS time series, the range of the extreme variations, and noise (Fig. 5). Figure 4c illustrates the annual variations of the GNSS stations. We combine the sine functions to

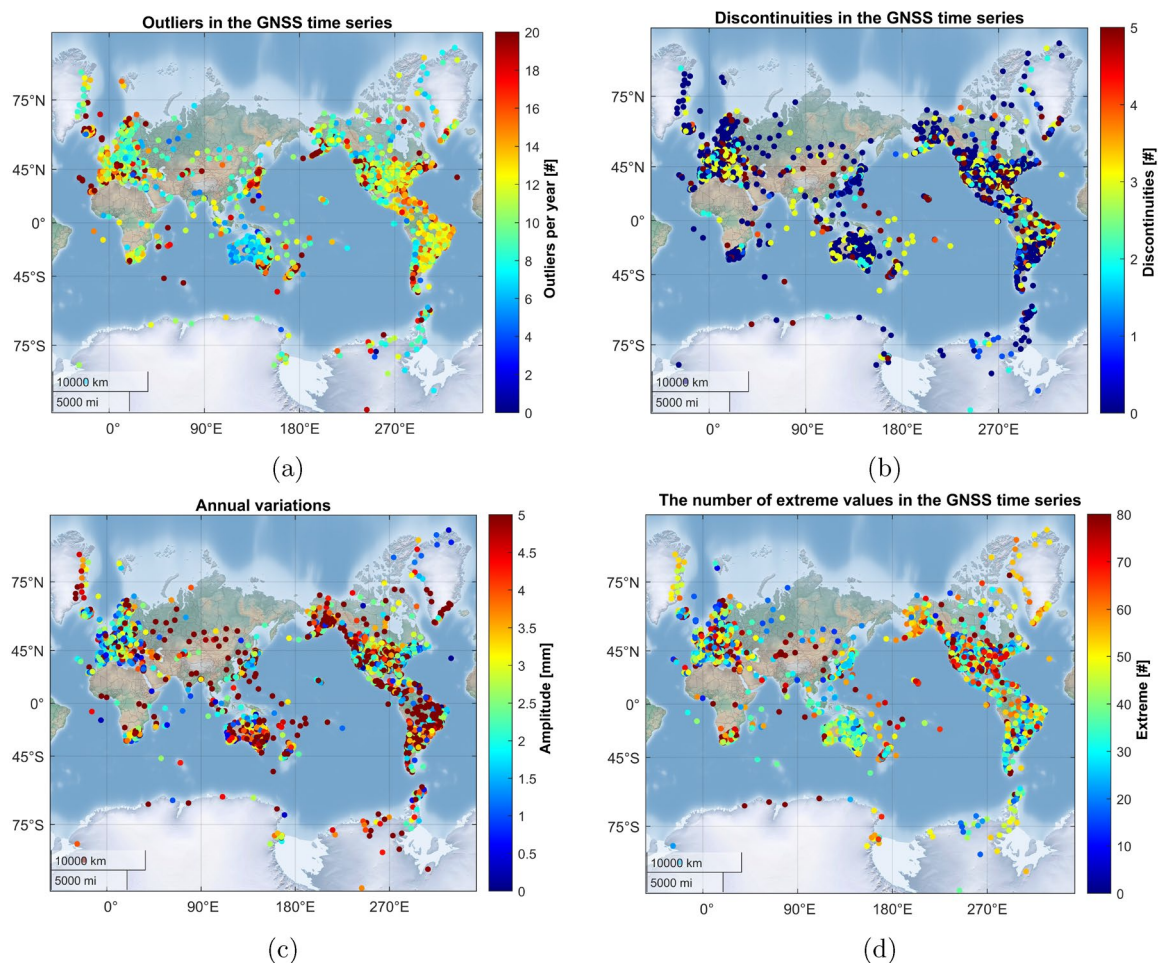


Fig. 4 Four specific features for training the classification ML models, including outliers (a), discontinuities (b), annual variations (c), and extreme values (d) in the station coordinate time series

determine the amplitude of these variations. As seen clearly on the map, the GNSS stations located in active tectonics, like the Ring of Fire region⁹, will measure a larger amplitude than other regions. Globally, the average amplitude is ~ 3.1 mm. Furthermore, the uncertainty of the GNSS station coordinate time series can be presented via the entropy index, which reflects the unpredictability of fluctuations over a time series. Finally, Fig. 4d shows the number of extreme values in the three components of the coordinate time series at a moving window = 365 days. On average, 46 extreme points are identified in each GNSS station coordinate time series, with four stations reaching the maximum number of 112 extreme points.

The coordinate time series are processed linear regression to gain three features: RMSE, SD, and velocity vectors. The monitoring timespans and observation records are also counted in this regression process. In data mining, a

denser observation frequency is often expected to enhance reliability, especially for long-term monitoring that requires stable and consistent observations. Hence, the monitoring timespan and observation frequency are included in the classification criteria. To simplify the calculation, we reduce the data dimensionality of the 46 parameters into 15 features for training ML models, as shown in Fig. 5. The features are plotted in parallel coordinates corresponding to 15 y-axes, with the feature names listed along the x-axis. The values on the y-axes represent the data ranges of the respective features.

Data clustering and labeling

We create samples for training the supervised ML-based classification models using the unsupervised ML techniques to cluster 15 selected features (Fig. 5) corresponding to the seven categories (Table 1). Here, we grouped features with similar data ranges or characteristics into a single category for statistical testing at once. To achieve

⁹ https://en.wikipedia.org/wiki/Ring_of_Fire

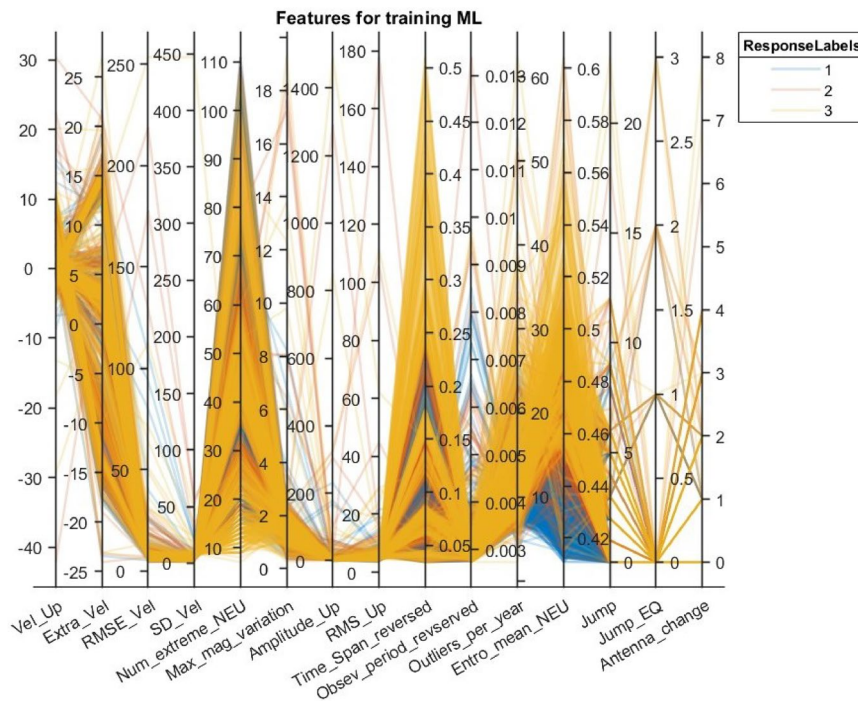


Fig. 5 Fifteen features derived from the 1900 GNSS stations in the European countries for training ML. From left to right along the x-axis, the selected features include the velocity of the Up coordinate, intra-plate motions, the RMSE of the velocity fields, the standard deviation (SD) of the velocity fields, the number of extreme values in the NEU coordinates, the maximum variation amplitude of extreme values, the seasonal variation amplitude of the Up coordinate, the

RMS residuals of the Up coordinate, the monitoring timespan, the number of daily observations (daily coordinate solutions) per year, the number of outliers per year, the average entropy in the NEU coordinates, the number of jumps in the NEU related to antenna changes, the number of jumps in the NEU related to earthquakes, and the number of antenna changes

Table 1 Estimation of the feature separation by the silhouette scores

Methods	Outliers_1P	Entropy_3P	Timespans_2P	Velocity_2P	Discontinuities_3P	RMSE_2P	Variations_4P
Hierarchy	0.60	0.73	0.87	0.99	0.98	0.99	0.99
K_means	0.60	0.71	0.84	0.83	0.76	0.99	0.99
Fuzzy C_means	0.49	0.70	0.83	0.51	0.68	0.98	0.99

robust solutions, we investigate three unsupervised ML techniques, namely Hierarchy, K-means, and Fuzzy C-means. The silhouette criterion is applied to test the dissimilarity in each feature. The silhouette score ranges from -1 to 1, where 1 indicates a strong alignment between data points and their corresponding clusters, while -1 signifies misassigned clusters. Silhouette scores will vary based on the number of set clusters. There is no hard threshold for silhouette scores, but a common standard of ≥ 0.5 is often specified (Rousseeuw 1987; Leonard and Rousseeuw 1990; Campello and Hruschka 2006; Shutaywi and Kachouie 2021). Table 1 displays a substantial variation range in silhouette scores across the three clustering methods, with Hierarchical clustering generally yielding higher scores than the others. The features of seasonal variations and RMSE exhibit a significant influence, while the features of outliers and entropy consistently register lower

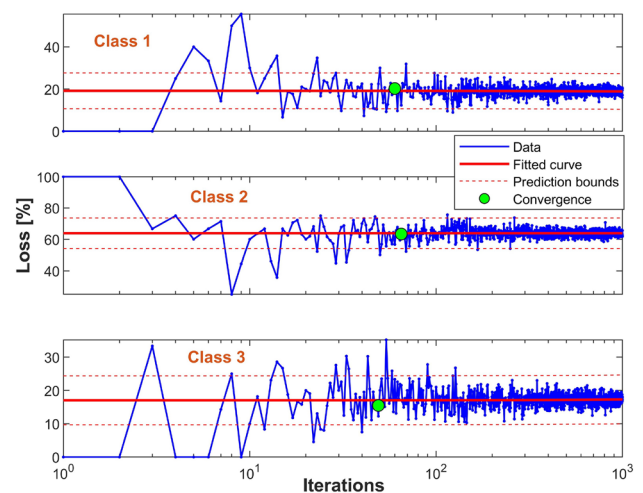


Fig. 6 The loss patterns of the entropy feature over 1000 optimization loops at the POTS station in Potsdam, Germany

scores on all three clustering methods. Even so, all the selected features stay at a silhouette score range from 0.47 to 0.99. This implies the features have robust statistics for the clustering process.

Clustering the features is then executed by the optimization process. The convergence point is determined within a window = 5 moving across iterations. That means the clustering process will stop and suggest a final solution when the certainty percentage of the cluster does not exceed a statistical threshold after five loops. As depicted in Fig. 6, the entropy feature of the GNSS coordinate time series at the POTS station in Potsdam, Germany, stays in Class 2 with the highest confidence interval compared to Classes 1 and 3. Here, we used an exponential function to fit the loss patterns of the clustering process. The convergence point is observed at the 65th iteration, regarding the accuracy and convergence speed at its best.

Figure 7 shows the ML-based feature clustering and their linkage correlation. The Cophenetic coefficient (C), ranging from 0 to 1, indicates the levels of distinction among the clusters. The Fi values indicate the fuzzy threshold between two linkages. The cutoff threshold is limited within the upper and lower boundaries. The cutoffs are determined as the median values of the distance matrices. A greater distance between the Fi values indicates a higher reliability in clustering. Our study sets $C \geq 0.7$, regarded as a high probability, to ensure that samples are distributed almost evenly into each cluster.

In this step, human-based supervision involves re-ordering and labeling the clusters assigned by the *unsupervised* ML process. Based on the cluster mean values, Cluster 2, in green scatters, will be designated Class 3 (the lowest quality). In contrast, Cluster 3, in yellow scatters, will be classified as Class 2 (Fig. 7b). This label re-order process will be

performed on the “GNSS_Cluster” App via the “Sort_data” option.

Model training

We first use the 1900 stations in European countries as samples to train ML models. Here, the “train-classify-correct-(re)train” loops are executed (please refer to Flowchart’s last step). The performance of the ML-based classification models is estimated via the K-fold cross-validations, with $K = 5$. The accuracy of the ML models in the first loop “train-classify-correct” is depicted in Fig. 8. Among these, more errors are identified in predictions related to the features of variations and discontinuities in the coordinate time series (Fig. 8a). That might be because these features were affected by anomalies and/or extreme values. Even so, the average true prediction rate, Positive Predictive Value (PPV), over three classes still archives $\sim 97.7\%$ (Fig. 8b). Class 1 reaches the highest accuracy, with a true prediction rate of up to 99.2% . Meanwhile, the average false prediction rate, False Negative Rate (FNR), is $\sim 0.8\%$, with the highest wrong prediction of only 3.4% for Class 3.

Because of higher accuracy, the Boosted-Tree Ensemble model is selected to classify the global 7075 GNSS stations in the Nevada database (Table 2). As the 1900 Europe-GNSS stations have already been involved in training the classification models as mentioned above, they are now excluded from the global GNSS dataset that needs to be classified. Our findings reveal that the percentage of the high-quality stations labeled Class 1 in the European countries accounts for 39.7% , slightly higher than the other global stations with 35.9% (Fig. 9). In contrast, the Europe-GNSS stations labeled Class 3 account for 24.6% , compared to 26.2% for the global GNSS stations. Notably, the GNSS stations in

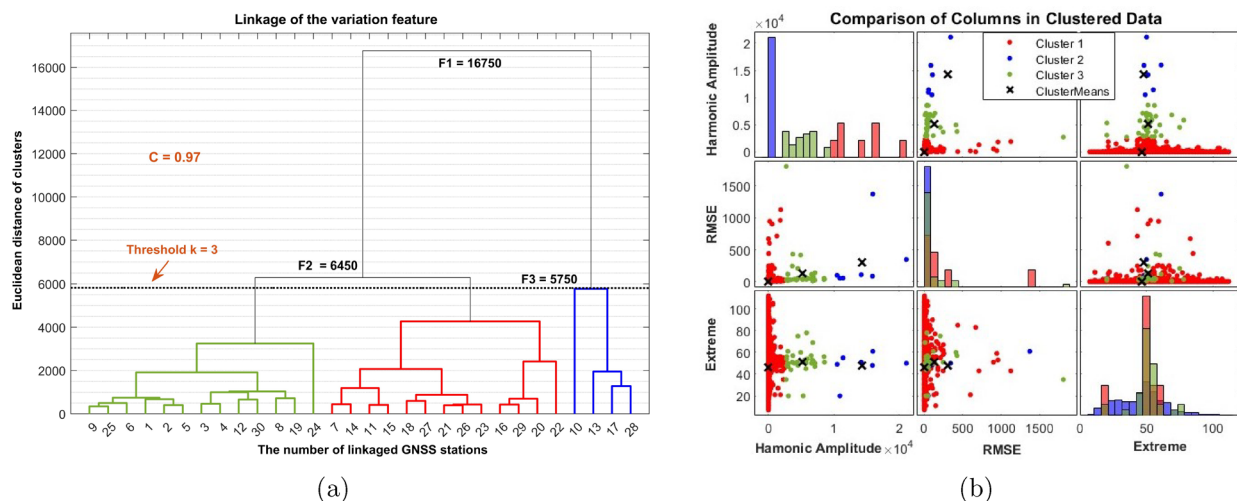


Fig. 7 The linkage (a) and clustering matrix (b) of the variation feature

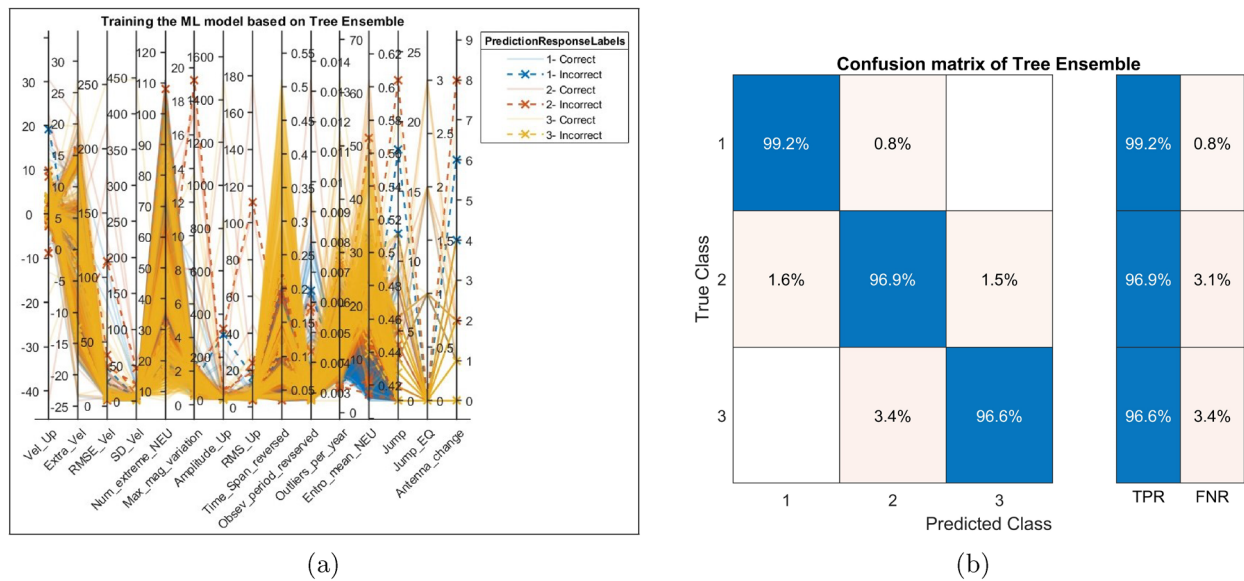


Fig. 8 Training the Ensemble-based classification model using the features from the 1900 EU-GNSS stations **(a)** and its confusion matrix **(b)**

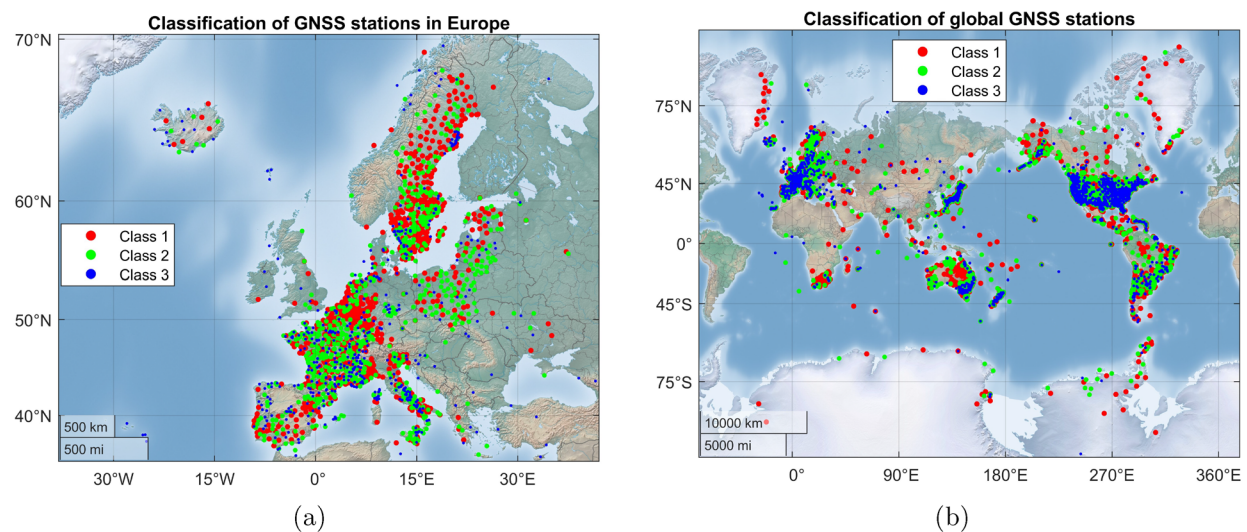
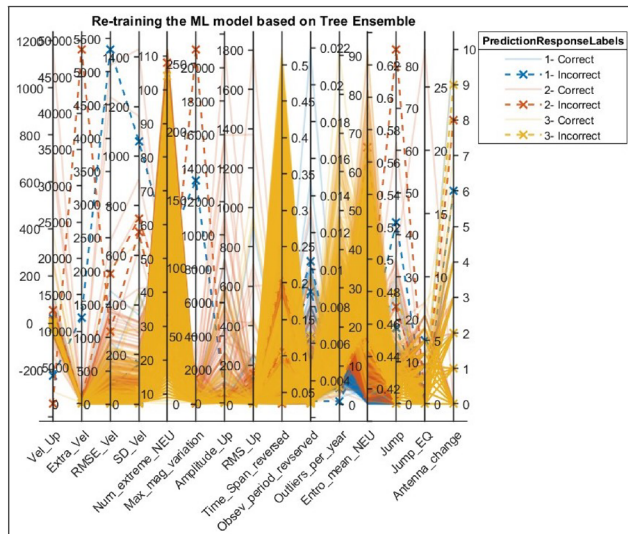


Fig. 9 Distributing the continuous GNSS stations in European countries **(a)** and worldwide **(b)** in three classes

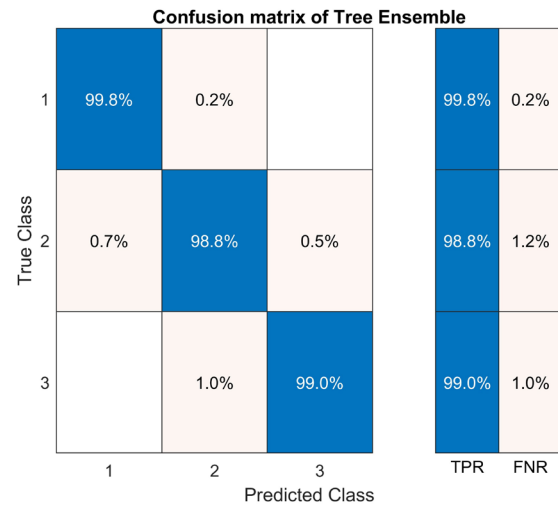
Greenland are mainly defined as Classes 1 and 2. This is because our classification strategy is based on 15 features/indices, and velocity is only one of these 15 features. For example, except for the velocity fields, the data pattern obtained from station LBIB in Greenland shows the data quality indices (e.g., jumps, outliers, observation frequency, RMSE, and extreme values in the coordinate time series) better than station POTS in Potsdam, Germany. Please note that the GNSS stations have been observed continuously; these classification results might change if the inputs are updated (e.g., investigated timespans, observations, and GNSS metadata) or adjusted (e.g., selected features).

However, the methodology, as well as the model training approach, remains consistent.

To improve the classification performance, all the classified features from the ~9000 global GNSS stations above will be inputs for retraining the ML models. Figure 10 shows the classification outputs of the retrained ML model and its corresponding confusion matrix. The true predictions in this re-training round reach an average rate of ~99.2 % (Fig. 10b), compared to ~97.7 % in the first training round above (Fig. 8b). The average false prediction rate on all three classes in this retraining round is ~2.4 %, in which Class 1 reaches the highest with only 0.2 %, corresponding to 18



(a)



(b)

Fig. 10 Retraining the Ensemble-based classification model using the features from ~9000 GNSS stations (a) and its confusion matrix (b)

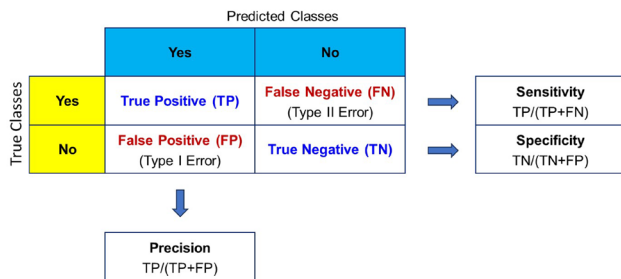


Fig. 11 Error types of a binary classification model

wrong prediction stations in the total of the classified 8975 stations.

Usually, when designing strategies for training ML models, the goal is to balance specificity and sensitivity (Fig. 11). However, in some cases, it may be desirable for

either the Type-I error (False Positive Rate - the ML model's specificity) or the Type-II error (False Negative Rate - the ML model's sensitivity) in each class to be more regarded. For example, minimizing the Type-II error (False Negative Rate) is crucial in hazard warnings. This implies that forecast models should maximize sensitivity to ensure no potential risk of hazards is skipped. On the other hand, in this study, stations in Class 1 are considered to be of high quality to serve high-precision tasks. Thus, the Type-I error in this class will be expected to be the lowest possible, so there will be the least wrong decisions in selecting the high-precision required stations. As indicated in Fig. 10b, the Type-II error (FNR) in Class 1 is 0.8 % while the Type-I error (FPR) remains lower, with only 1.6 %. Depending on specific tasks, we will decide on a trade-off between either specificity or sensitivity by selecting samples, adjusting statistical thresholds, or tuning hyper-parameters of the ML models.

Table 2 The accuracy improvement of the classification models

Model Type	Training (400 EPN-ROB samples)		Training (1900 Nevada samples)		Re-training (9000 Nevada samples)		Accuracy improvement
	Accuracy (%)	Total cost	Accuracy (%)	Total cost	Accuracy (%)	Total cost	
Naive Bayes	47.8	192	71.2	547	68.3	2843	↓ -2.89
Subspace KNN	41.6	215	70.4	563	70.2	2672	↓ -0.14
Subspace Discriminant	45.9	199	72.7	518	71.0	2603	↓ -1.74
Linear SVM	44.6	204	78.9	400	75.1	2238	↓ -3.88
Kernel SVM	46.7	196	82.2	338	84.2	1417	↑ 2.00
Fine KNN	43.8	207	83.3	318	89.7	921	↑ 6.48
Trilayered Neural Network	43.8	207	90.6	179	96.7	293	↑ 6.16
Boosted-Tree Ensemble	48.6	189	97.7	43	98.9	96	↑ 1.19
Fine Tree	41.3	216	97.4	50	99.1	79	↑ 1.75
Bagged-Tree Ensemble	48.6	189	97.3	52	99.2	71	↑ 1.95

Comparison of the ML models

Table 2 compares the ML models' accuracy, which lists fifteen classification models based on the six supervised ML techniques. The number of mathematical models is more than that of the ML techniques because, for one ML technique, we can build different mathematical models by adjusting hyper-parameters, distance matrices, or loss functions. Using the three-class features of the ROB-EPN stations for training the ML models gains an accuracy of only under 50 %. We would note that the low classification performance is due to the limited sample size of only 400 stations, not related to the ROB-EPN data quality. Our preliminary investigations also showed that the proposed method requires at least 1000 GNSS stations to achieve stable and high-accuracy performance (Le et al. 2022). In addition, this study used the features for classification derived from the high-precision time series processing, as shown in Le et al. (2024). In comparison, the samples created by the H&M approach can double the classification performance of ML models. The Tree-based Ensemble models can reach an accuracy of over 97 %, at a statistical significance, and around 99 % after retraining.

The performance of the Tree-based models is outstanding, with >97 %. Retraining these Tree-based models can improve their accuracy from 1.2 to 2.0 %. The improvement of the Fine K-Nearest Neighbor (KNN) model reaches up to ~6.5 %. The Naive Bayes model performs the lowest accuracy, with <70 %. The distinction of the accuracy range between the Naive Bayes models and the others might result from the principle of the Bayes algorithms, which is based on a naive assumption of independence among features. This assumption is often challenging to satisfy in real applications, especially in classification issues. For example, we can only statistically test to exclude weak correlation features but cannot eliminate multicollinearity among features. Here, linear correlations exist between the pair of features, such as the pair of discontinuities and antenna changes or the pair of timespan and observation frequency. However, each feature plays a specific role in estimating the GNSS station quality, so we included these features in the classification criteria. Thus, once this initial assumption is not satisfied, the Naive Bayes-based models might be biased. In contrast, the conditions of initial inputs for training the Decision Tree-based models are less strict, and the branches of the Decision Tree models are also developed flexibly. As a result, the classification performance of the Tree-based ML models is often higher than others.

Again, the ML-based classification models obtained from this study can also find applications in classifying the IGS stations or national Continuously Operating Reference Stations (CORS). In that case, human-based supervision in improving AI's performance can be utilized from both

internal (i.e., GNSS station operators) and external (i.e., GNSS users) sources. More specifically, GNSS users can provide feedback on the status of monuments or observations. The station operators will check the feedback's reliability and consider it as updated samples for retraining. This approach not only enhances the accuracy of ML models over time but also timely reflects the actual status of the GNSS stations.

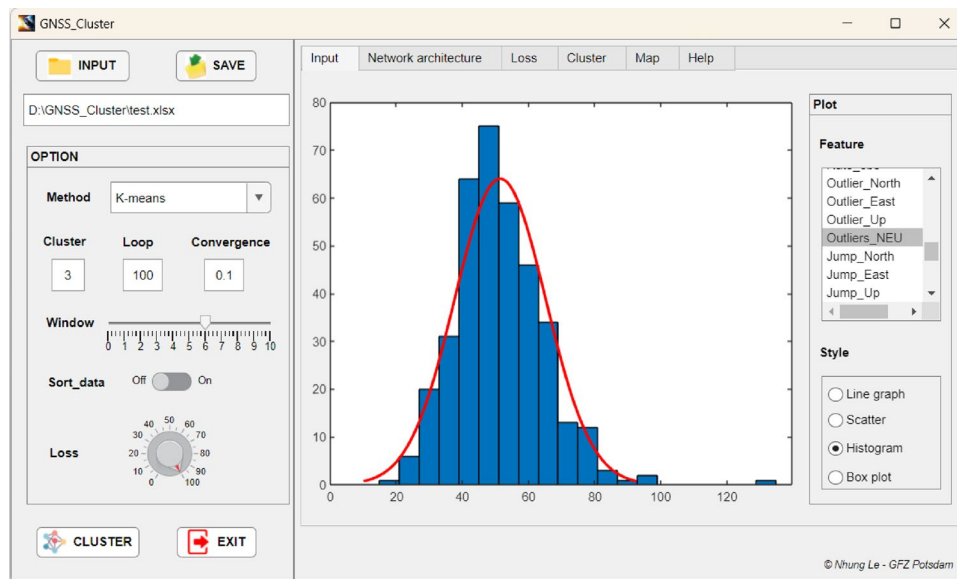
Conclusions and outlook

While AI (via ML techniques) can process big data efficiently and promote an extensive application spectrum, humans can better solve the problems associated with anomalies and mutations. Also, in the context of potential risks, AI is warned not to work beyond the control of natural intelligence - of human beings. This study again advocates that combining natural and artificial intelligence can enhance AI's performance. Our findings have demonstrated the feasibility of the H&M approach in improving the ML models' accuracy using two sampling strategies: “*suggest-decide*” and “*correct-retrain*”. To mitigate humans' biases, we applied the optimization processing technique, which has been included in our MATLAB app “*GNSS_Cluster*”. The Decision Tree models gain outstanding accuracy among the fifteen investigated models, with under 100 false predictions in the ~9000 stations. The Tree-based Ensemble models with a classification performance of up to 99.2 % are highly recommended for applications. The ~9000 GNSS stations have been classified, which can be considered as samples for training other ML models and addressing similar tasks. Finally, our methodology potentially tackles other topical issues, such as early warning of natural hazards or clustering high-risk areas. In that case, users become a part of the warning systems, contributing insights into data characteristics and rectifying AI-made errors.

Appendices

A The MATLAB App for clustering and labeling GNSS data

The “*GNSS_Cluster*” App is designed with five main Tabs: “*Input*”, “*Network architecture*”, “*Loss*”, “*Cluster*”, and “*Map*”. GNSS users can copy, edit, and save figures displayed in these tabs in four formats: *.png, *.jpg, *.tif, and *.pdf. The outputs can be saved in three formats: *.xls, *.xlsx, and *.csv. Tab “*Help*” is a further introduction to the App.

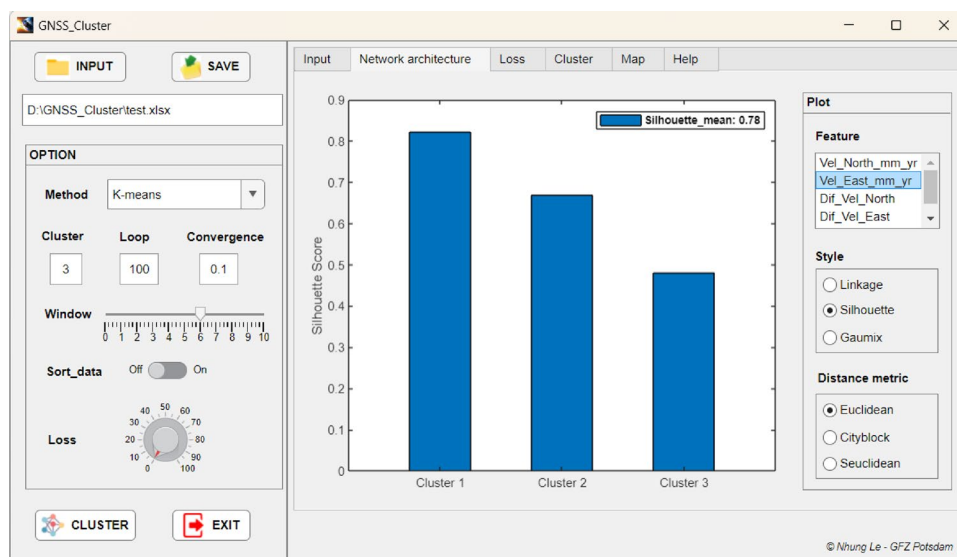


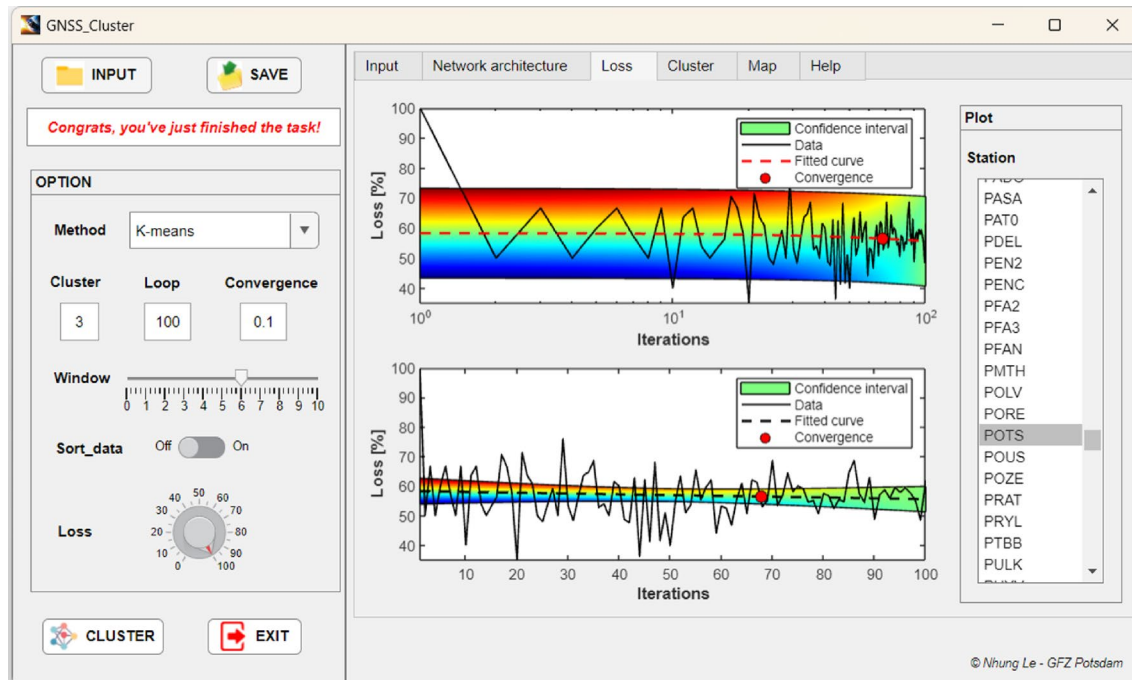
B Setting hyper-parameters in the MATLAB App

GNSS users can set hyper-parameters for ML models, as shown on the left-hand side of the app's interface. The options of the clustering models are also found on the right-hand side, corresponding to Tab “*Network architecture*”.

C Performing optimization loops for clustering GNSS data

Optimization loops are executed to determine the convergence point. This point is the optimal number of loops that should be performed to gain the highest probability for a certain cluster.





Acknowledgements The authors gratefully acknowledge the Nevada Geodetic Laboratory for providing the GNSS coordinate time series, the Royal Observatory of Belgium for providing the classified GNSS station data, the GEOFON Seismic Program in Germany, and the USGS Earthquake Hazards Program in the US for providing the seismic data. We have also applied the ML toolboxes in MATLAB® (Release 2023b) to train the ML models in this study. We especially would like to thank Dr. Mihaela Jarema in MathWorks, München, Germany, and Dr. Dedong Wang in the GFZ, Potsdam, Germany, for their valuable suggestions and constructive discussions on our MATLAB App of “GNSS_Cluster”.

Author contributions BM and NL designed the study. NL developed the methodologies, coded the computation programs, and analyzed the results. BM, LB, and HS interpreted and verified the results. NL wrote the manuscript, and all the authors revised it for article preparation.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability The raw GNSS coordinate time series can be downloaded at <http://geodesy.unr.edu>. The classified EPN stations have been provided by the Royal Observatory of Belgium at https://www.epncb.oma.be/_networkdata/stationlist.php. The seismic data are available at <https://www.usgs.gov/> and <https://geofon.gfz-potsdam.de/>.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Alfonseca M, Cebrian M, Anta AF, Coviello L, Abeliuk A, Rahwan I (2021) Superintelligence cannot be contained: lessons from computability theory. *J Artif Intell Res* 70:433–460. <https://doi.org/10.1613/jair.1.12202>
- Altamimi Z, Collilieux X, Metivier L (2013) Reference frames for applications in geosciences. *Int Assoc Geodesy Symp* 138:137–145. <https://doi.org/10.1007/978-3-642-32998-2>
- Amershi S, Cakmak M, Knox WB, Kulesza T (2014) Power to the people: the role of humans in interactive machine learning. *AI Mag* 35(4):105–120. <https://doi.org/10.1609/aimag.v35i4.2513>

- Arslan A, Cooper C, Khan Z, Golgeci I, Ali I (2022) Artificial intelligence and human workers interaction at team level: a conceptual assessment of the challenges and potential HRM strategies. *Int J Manpow* 43(1):75–88. <https://doi.org/10.1108/IJM-01-2021-0052>
- Bali V, Banerjee K, Kumar N, Gour S, Chawla SK (eds) (2022) Demystifying technologies for computational excellence moving towards society 5.0: Industry 4.0, AI, and data science - Research trends and challenges (1st, 1st edn. CRC Press, Taylor & Francis Group LLC., London. <https://doi.org/10.1201/9781003097181>
- Bastos L, Bos M, Fernandes RM (2014) Sciences of geodesy - Chapter 5: Deformation and tectonics: contribution of GPS measurements to plate tectonics "Overview and recent developments," 1st edn. Springer, Berlin Heidelberg. https://doi.org/10.1007/978-3-642381-11741-1_5
- Blewitt G, Hammond W, Kreemer C (2018) Harnessing the GPS data explosion for interdisciplinary science. *Eos* 99(2):e2020943118. <https://doi.org/10.1029/2018eo104623>
- Bradke M (2020) SEMISYS—sensor meta information. System. <https://doi.org/10.5880/GFZ.1.1.2020.005>
- Braun D, Bhat M, Biesdorf A, Matthes F (2020) Would you lie to me bot? Supporting decision-making processes with deceiving virtual agents. *Proc Comput Sci* 177:587–592. <https://doi.org/10.1016/j.procs.2020.10.083>
- Bruyninx C, Legrand J, Fabian A, Pottiaux E (2019) GNSS metadata and data validation in the EUREF permanent network. *GPS Solut* 23(4):1–14. <https://doi.org/10.1007/s10291-019-0880-9>
- Campello RJ, Hruschka ER (2006) A fuzzy extension of the silhouette width criterion for cluster analysis. *Fuzzy Sets Syst* 157(21):2858–2875. <https://doi.org/10.1016/j.fss.2006.07.006>
- Carrino L, Givoe S (2021) Progresses in artificial intelligence and neural systems. In: Howlett RJ, Jain LC (eds), vol 184. Springer Nature Singapore Pte Ltd. https://doi.org/10.1007/978-981395-15-5093-5_29
- Choi CQ (2021) Superintelligent AI may be impossible to control; that's the good news. Retrieved from <https://spectrum.ieee.org/super-artificialintelligence>
- Ge M, Gendt G, Dick G, Zhang FP, Rothacher M (2006) A new data processing strategy for huge GNSS global networks. *J Geodesy* 80(4):199–203. <https://doi.org/10.1007/s00190-006-0044-x>
- Gordon AD (2018) A review of hierarchical classification. *R Stat Soc J Ser A Gen* 150(2):119–137. <https://doi.org/10.2307/2981629>
- Hagendorff T (2021) Linking human and machine behavior: a new approach to evaluate training data quality for beneficial machine learning. *Mind Mach* 31(4):563–593. <https://doi.org/10.1007/s11023-021-09573-8>
- Kalpoka I, Kalpokiene J (2022) Deepfakes: a realistic assessment of potentials risks and policy regulation. Springer Nature, Switzerland. <https://doi.org/10.1007/978-3-030-93802-4>
- Kantardzic M (2020) Data mining: concepts, models, methods, and algorithms, 3rd edn. Wiley, Hoboken, New Jersey. <https://doi.org/10.1080/07408170490426107>
- Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY (2002) An efficient k-means clustering algorithm: analysis and implementation. *IEEE Trans Pattern Anal Mach Intell* 24(7):881–892. <https://doi.org/10.1109/TPAMI.2002.1017616>
- Karaim M, Elsheikh M, Noureldin A (2018) Multifunctional operation and application of GPS - Chapter 4: GNSS error sources, 1st edn. Hashimov RBR, AM (eds). IntechOpen. <https://doi.org/10.5772/intechopen.75493>
- Kreemer C, Geoffrey B, Elliot CK (2014) A geodetic plate motion and global strain rate model. *Geochem Geophys Geosyst* 15:3849–3889. <https://doi.org/10.1002/2014GC005407>
- Le N, Männel B, Bui LK, Schuh H (2024) Identifying neotectonic motions in Germany using discontinuity-corrected GNSS data. *Pure Appl Geophys* 181(1):87–108. <https://doi.org/10.1007/s00024-023-03390-z>
- Le N, Männel B, Deng Z, Luong T, Schuh H (2022) Improving classification performance of continuous GNSS stations using a combination of Human and Machine Learning. In: 2nd symposium of IAG (international association of geodesy) commission 4 positioning and applications, p 5194. <https://doi.org/10.5194/iag-comm4-2022-20>
- Legrand J (2022) EPN multi-year position and velocity solutions. Royal observatory of Belgium. Retrieved from <https://www.epncb.oma.be/DOI/ROB-EUREF-CWWW.php?Solution=CWWW-Whhttps://doi.org/10.24414/ROB-EUREF-CWWW>
- Legrand J, Bruyninx C (2020) Quality assessment of GNSS reference stations: criteria and thresholds. *EGU Gen Assem* 2020:20320. <https://doi.org/10.5194/egusphere-egu2020-20320>
- Legrand J, Bruyninx C (2021) Station classification and reference station selection. *EGU Gen Assem* 2021:14190. <https://doi.org/10.5194/egusphere-egu21-14190>
- Leonard K, Rousseeuw PJ (1990) Finding groups in data: an introduction to cluster analysis. Wiley. <https://doi.org/10.2307/2532178>
- Li Z (2021) Recent advances in earthquake monitoring: ongoing revolution of seismic instrumentation. *Earthq Sci* 34(2):177–188. <https://doi.org/10.29382/eqs-2021-0011>
- Mazinan AH (2013) A new algorithm to AI-based predictive control scheme for a distillation column system. *Int J Adv Manuf Technol* 66(9–12):1379–1388. <https://doi.org/10.1007/s00170-012-4415-6>
- Moura P, Paulo O, Paulo L, Eds R, Goebel R (eds) (2021) Progress in artificial intelligence, 1st edn. Springer Nature, Switzerland. <https://doi.org/10.1007/978-3-030-86230-5>
- Natsume Y, Oaku K, Takahashi K, Nakamura W, Oono A, Hamada S, Sasano T (2018) Combined analysis of human and experimental murine samples identified novel circulating micRNAs as biomarkers for atrial fibrillation. *Circ J* 82(4):965–973. <https://doi.org/10.1253/circj.CJ-17-1194>
- Nayak J, Naik B, Behera H (2015) Fuzzy C-means (FCM) clustering algorithm: a decade review from 2000 to 2014. In: Jain LC, Behera HS, Mandal JK, Mohapatra DP (eds), Computational intelligence in data mining-volume 2: Proceedings of the international conference on CIDM, 2014 (vol 32, pp 133–149). https://doi.org/10.1007/978-81-322451-2208-8_14
- Ross T (1935) Machines that think—a further statement. *Psychol Rev* 42(4):387. <https://doi.org/10.1037/h0059970>
- Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 20(C):53–65. [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7)
- Saria E, Calais E, Altamimi Z, Willis P, Farah H (2013) A new velocity field for Africa from combined GPS and DORIS space geodetic solutions: contribution to the definition of the African reference frame (AFREF). *J Geophys Res: Solid Earth* 118(4):1677–1697. <https://doi.org/10.1002/jgrb.50137>
- Schick N (2020) Deepfakes: the coming infocalypse. Twelve Hachette Book Group. Retrieved from <https://www.hachettebookgroup.com/titles/nina-schick/deepfakes/9781538754313/?lens=twelve>
- Serpelloni E, Cavaliere A, Martelli L, Pintori F, Anderlini L, Borghi A, Cacciaguerra S (2022) Surface velocities and strain-rates in the Euro-Mediterranean region from massive GPS data processing. *Front Earth Sci* 10:1–21. <https://doi.org/10.3389/feart.2022.907897>
- Shi J, Malik J (2000) Normalized cuts and image segmentation. *IEEE Trans Pattern Anal Mach Intell* 22(8):888–905. <https://doi.org/10.1109/34.868688>
- Shutaywi M, Kachouie NN (2021) Silhouette analysis for performance evaluation in machine learning with applications to clustering. *Entropy* 23(6):1–17. <https://doi.org/10.3390/e23060759>
- Silaparasetty N (2020) Machine learning concepts with python and the jupyter notebook environment. In: Spahr W, Black A, Markham J, Vakili J (eds). 2nd edn. Apress, Bangalore, India. <https://doi.org/10.1007/978-1-4842-5967-2>

- Villacampa-Calvo C, Zaldívar B, Garrido-Merchán EC, Hernández-Lobato D (2021) Multi-class Gaussian process classification with noisy inputs. *J Mach Learn Res* 22(36):1–52
- Vollerthun A, Wieser M (2002) A European global navigation satellite system—the German market and value adding chain effects. *Acta Astronaut* 50(5):325–333. [https://doi.org/10.1016/S0094-5765\(01\)00173-4](https://doi.org/10.1016/S0094-5765(01)00173-4)
- Wang S, Huang S, Fang H (2022) Ground GNSS station selection to generate the global ionosphere maps using the information content. *Space Weather* 20(1):e2020SW002675. <https://doi.org/10.1029/2020SW002675>
- Welser W, Osoba O (2017) An intelligence in our image—the risks of bias and errors in artificial intelligence, 1st edn. The RAND Corporation, Santa Monica, California. <https://doi.org/10.7249/RR1744>
- Yu J, Meng X, Yan B, Xu B, Fan Q, Xie Y (2020) Global navigation satellite system based positioning technology for structural health monitoring: a review. *Struct Control Health Monit* 27(1):e2467. <https://doi.org/10.1002/stc.2467>
- Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.