APL Machine Learning

RESEARCH ARTICLE | MARCH 12 2024

Random forests for detecting weak signals and extracting physical information: A case study of magnetic navigation

Mohammadamin Moradi ; Zheng-Meng Zhai ; Aaron Nielsen ; Ying-Cheng Lai

APL Mach. Learn. 2, 016118 (2024)
https://doi.org/10.1063/5.0189564

Random forests for detecting weak signals and extracting physical information: A case study of magnetic navigation

Cite as: APL Mach. Learn. 2, 016118 (2024); doi: 10.1063/5.0189564
Submitted: 29 November 2023 • Accepted: 20 February 2024 •
Published Online: 12 March 2024

Mohammadamin Moradi,[1] Zheng-Meng Zhai,[1] Aaron Nielsen,[2] and Ying-Cheng Lai[1,3,a)]

AFFILIATIONS

[1] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA
[2] ANT Center, Air Force Institute of Technology, Wright-Patterson AFB, 2950 Hobson Way, Ohio 45433, USA
[3] Department of Physics, Arizona State University, Tempe, Arizona 85287, USA

[a)] **Author to whom correspondence should be addressed:** Ying-Cheng.Lai@asu.edu

ABSTRACT

It has been recently demonstrated that two machine-learning architectures, reservoir computing and time-delayed feed-forward neural networks, can be exploited for detecting the Earth's anomaly magnetic field immersed in overwhelming complex signals for magnetic navigation in a GPS-denied environment. The accuracy of the detected anomaly field corresponds to a positioning accuracy in the range of 10–40 m. To increase the accuracy and reduce the uncertainty of weak signal detection as well as to directly obtain the position information, we exploit the machine-learning model of random forests that combines the output of multiple decision trees to give optimal values of the physical quantities of interest. In particular, from time-series data gathered from the cockpit of a flying airplane during various maneuvering stages, where strong background complex signals are caused by other elements of the Earth's magnetic field and the fields produced by the electronic systems in the cockpit, we demonstrate that the random-forest algorithm performs remarkably well in detecting the weak anomaly field and in filtering the position of the aircraft. With the aid of the conventional inertial navigation system, the positioning error can be reduced to less than 10 m. We also find that, contrary to the conventional wisdom, the classic Tolles–Lawson model for calibrating and removing the magnetic field generated by the body of the aircraft is not necessary and may even be detrimental for the success of the random-forest method.

I. INTRODUCTION

Random forest[1–3] is a supervised machine-learning method for solving classification and regression problems based on noisy feature data. The "forest" consists of a number of decision trees, each trained using a different subset of the characteristics and data points. A decision tree[4,5] recursively splits the feature space into two halves, which is done for a specific feature input using a threshold, and the depth of the tree is the number of available input feature signals. At the end of the tree construction, the whole feature space has been divided into a large number of small subspaces, each associated with a particular value of the physical quantity of interest (the target variable), leading to a "leaf." Given a number of feature signals in the form of, e.g., time series and the value of the target variable, supervised training can be done through some standard optimization methods to determine a proper set of the threshold values required for splitting the feature signals. After training is done, when a new set of feature signals is presented to the tree, a branch of the tree can be quickly identified, which closely matches the corresponding feature values, and the leaf end of this branch gives the predicted value of the physical quantity for the particular set of feature values. In principle, a decision tree can be used for predicting the values of the target variable corresponding to different combinations of the values of the feature signals, but a single tree is susceptible to overfitting, especially when the feature space becomes large. Random forest solves this overfitting problem by combining a number of random decision trees, each responsible for a subset of the features and a portion of the training data.

More specifically, the working of the random-forest algorithm can be described as follows. For each tree in the forest, a subset of the features and a portion of the training data are randomly chosen, where each tree is given a slightly different perspective on the data, known as bootstrap sampling.[6] The algorithm then creates a decision tree utilizing the characteristics and data that have been chosen using a splitting criterion, such as the Gini index or information gain. The new tree is then added to the forest, and the process is repeated until a pre-specified requirement is met, at which point no more trees are produced. During the testing or prediction phase, the model combines the predictions from all the trees in the forest to provide a forecast. The predicted value of the target variable is the average or median of the predictions of all trees in the ensemble. Random forest is resilient to noise and outliers, and it alleviates overfitting since the ensemble of trees balances out the noise and variability in the data. Moreover, random forests is capable of handling missing data and can offer metrics of feature relevance. Random forest regression has been applied in various fields, such as healthcare[7,8] and transportation,[9,10] for tasks such as predicting traffic flow[11] and forecasting commodity prices.[12] Various extensions and modifications of the random-forest algorithm, such as extremely randomized trees[13] and quantile regression forests,[14] have been proposed to further improve the accuracy and robustness of the method.

In this paper, we exploit random forests for accurate detection of weak physical signals and for predicting the values of a small number of physical variables of interest based on a relatively large number of noisy feature signals. In particular, we consider the situation where the feature signals can be continuously measured at all times, but the weak signal and the target variables can be assessed only in a special and well controlled environment with certain additional measurements—the calibration phase. The question is, in the deployment phase where the additional measurements for extracting the weak signal are no longer available and the target variables are not accessible any more, whether the weak signal and the variables can be predicted based on the available feature signals.

A directly relevant application is magnetic navigation in a GPS (global positioning system) denied environment, where the goal is to use the Earth's anomaly magnetic field for precise positioning of an airplane, as this field is position-dependent.[16,17] In such an application, various sensors in the cockpit of the airplane are employed to generate a large number of feature signals for extracting the Earth's anomaly magnetic field through complex mathematical algorithms. With a predetermined map between the anomaly field and the position for the flying region, if the anomaly field can be accurately detected, precise positioning can be achieved. A key challenge is that the anomaly magnetic field is weak and the feature signals from the sensors are overwhelmingly noisy due to the extensive electronic equipment in the cockpit and the other (dominant) components of the Earth's magnetic field. The problem then becomes one of detecting a weak signal from a noisy background that can be several orders of magnitude stronger than the signal and determining the components of the instantaneous position vector of the airplane (the target variables). In this regard, recently, a machine-learning scheme based on reservoir computing or time-delayed feed-forward neural networks has been developed to detect the Earth's anomaly field, with the implied equivalent positioning accuracy in the range of 10–40 m.[18]

Here, we articulate a random-forest-based machine-learning scheme to detect the Earth anomaly magnetic field and to simultaneously determine the position of the flying airplane. Different from the previous work,[18] we do not combine machine learning with other widely used calibration methods, such as the standard Tolles–Lawson (TL) model. Rather, in the whole process, only the machine-learning model is employed to generate the anomaly field signal *and* the positioning information based on the input feature signals. From time-series data gathered from the cockpit of an airplane, we demonstrate that the random-forest algorithm performs remarkably well in filtering Earth's anomaly magnetic field and generating the instantaneous position of the aircraft. With the aid of the conventional inertial navigation system (INS), the positioning error can be reduced to less than 10 m with negligible standard deviation or uncertainty. We also find that, contrary to the conventional wisdom, the classic TL model for calibrating and removing the magnetic field generated by the body of the aircraft is not necessary and may even be detrimental for the success of the random-forest method.

We remark that, for the task of precise positioning, our proposed random-forest method, in fact, does not require the use of INS sensors for three reasons. First, the INS sensors have a limited range, making it impossible for them to be used for long-distance navigation. Second, due to the accumulation of minute inaccuracies in the readings, the INS sensors are subject to drift errors that reduce the accuracy over time. Third, connecting INS sensors with other navigation systems may be difficult and complex.

We wish to emphasize the dynamical nature of the complex signals in our study. The measured signal comprises various components, including the weak signal generated by the Earth's anomaly magnetic field (the target signal to be detected), the signal from the other (dominant) components of the Earth's magnetic field, and the signals generated by the electronic equipment within the airplane cockpit. While the signals other than the target signal represent some kinds of "noises" to be removed and are typically much stronger, they are, in fact, complex signals with their own dynamics and time scales. To correctly describe these signals that need to be removed, we use the term" strong complex signals" or "overwhelming complex signals." It is the dynamical nature of these strong signals to be removed, which makes the random-forest approach effective. We note a recent work demonstrating that noises in the conventional sense can be filtered out by machine-learning schemes, such as reservoir computing.[15]

In Sec. II, we provide a brief overview of the background of magnetic navigation and the TL model for flight magnetic-field calibration and some previous machine-learning methods. In Sec. III, we describe the flight datasets, machine-learning methods employed in our study (for the purpose of performance comparison), simulation, and data-pre-processing details. Section IV presents results on feature selection, detection of the anomaly magnetic field, and precise positioning. A discussion and potential future research conclude this paper in Sec. V.

II. BACKGROUND

A. Earth's anomaly magnetic field for navigation in a GPS-denied environment

The Earth's magnetic field, or the geomagnetic field, is comprised of several field components.[19] The main component is the

core field generated by the motion of molten iron in the Earth's outer core, with its magnitude ranging from 25 to 65 μT at the surface of the Earth. While the core field makes compasses point north and is responsible for geophysical phenomena, such as the auroras, its magnitude is still quite weak: about 25–65 μT at the surface of the Earth, which is about 100 times weaker than a refrigerator magnet. The second component is the crustal anomaly field generated by the Earth's crust and upper mantle, whose magnitude is about 100 nano-Tesla, which is about 100 times weaker than the core field. While the core field is the dominant component of the geomagnetic field, it is weakly time-dependent and is not sensitive to changes in the position, rendering it unsuitable for precise positioning and navigation. In contrast, the anomaly field is position-dependent and has much stronger spatial variations than the core field. Consequently, in principle, it is possible to use exploit the anomaly field for navigation.

The widely used GPS can achieve the positioning accuracy of less than 10 m worldwide. However, because GPS signals are weak electromagnetic signals and must be transmitted over long distances, it is vulnerable to external interference, such as jamming or spoofing.[20] In a GPS-denied environment, alternative navigation systems are needed for positioning, which include radio-based navigation,[21] computer-vision-based navigation,[22] star-trackers,[23] terrain height matching,[24] and gravity gradiometry.[25] Despite the outstanding performance of these navigation approaches in some specific scenarios, they are unable to work universally under different circumstances. For instance, terrain-aided navigation relies on the unique features of the terrain, which will lose efficacy when working around oceans and deserts, and star-trackers rely on the stars, so it is not workable during the day or in cloudy weather. Different from these methods, the Earth's anomaly field is approximately time-invariant but strongly spatially variant, making magnetic navigation an appealing alternative[16,17] to GPS. Indeed, the anomaly-field-based magnetic navigation is limited neither to terrains now to the time of the day. Another advantage is that, unlike active navigation, such as GPS, magnetic navigation is a kind of passive navigation and, due to the power of the magnetic field decreasing as d^{-3} with distance d, it is not practically possible to disrupt a magnetic navigation device through jamming.[26] A great challenge of magnetic navigation is that the anomaly field is extremely weak and is usually embedded in an overwhelmingly strong noisy background. For example, in the cockpit of a flying aircraft, various types of electronic devices are in active operation.[27] To make magnetic navigation feasible, extracting the weak anomaly-field signal from strong complex signal is essential. With the availability of a predetermined magnetic map, the extracted clean anomaly-field signal can be used to determine the instantaneous position of the aircraft, possibly with the aid of a standard INS.[28]

B. Tolles–Lawson model

To realize magnetic navigation, effective methods to extract the anomaly magnetic-field signal and to obtain the real positioning information are needed. For a flying airplane, the magnetic field generated by the body of the aircraft must be removed from the measured signal to yield the Earth's magnetic field. The Tolles–Lawson (TL) model[29–31] is a linear aeromagnetic compensation method that estimates the magnetic field generated by the aircraft from the total measured magnetic field. When the aircraft is flying ideally in a "magnetically quiet" mode, e.g., there are only limited radio transmissions,[32] the TL model performs well in extracting the anomaly field from the signals measured by using magnetometers placed on the exterior surface of the airplane. However, when the magnetometers are placed inside the cockpit of the airplane, the TL model will not be sufficient to remove the overwhelming complex signals. Despite this, the TL model still represents a state-of-the-art model to calibrate the anomaly field through magnetometers placed outside the airplane and for pre-filtering the data.

C. Previous machine-learning methods

The last thirty years have witnessed the use of machine learning for magnetic navigation. Earlier, neural networks were proposed[33] as a model-free method for aeromagnetic calibration. About three years ago, hundreds of complicated neural networks were trained and it was demonstrated[34] that the nonlinear machine-learning method is capable of reducing the external added noise and extracting the magnetic anomaly signal. It has been proposed[28] recently that the anomaly field can be extracted with small errors by combining the TL model and machine learning. In particular, an extended Kalman filter was used to demonstrate that the extracted anomaly field can lead to low positioning errors. More recently, two machine-learning methods, one based on recurrent neural networks and another using feed forward neural networks with time-delayed inputs, in combination with the TL model, have been articulated[18] for detecting the weak anomaly field from measurements performed inside the cockpit of a flying airplane.

III. METHODS

A. Data source

The datasets used in our work come from the signal enhancement for the magnetic navigation challenge problem.[19] The goal was to extract a "clean" magnetic anomaly field from measured complex signals by using a trained neural network. To achieve this, several test flights were conducted, each containing several segments (or lines). Five magnetometers were used to record the magnetic-field signals, one placed at the tail stinger of the aircraft (in a "magnetically quiet" mode), as shown in Fig. 1. In particular, the tail-stinger signal was calibrated by the TL model, resulting in the ground truth of the true magnetic anomaly field. In our study, GPS signals were also used to provide the positioning information, representing the "ground truth" of the aircraft position.

B. Three machine-learning methods used in this study

The problem to extract the magnetic anomaly signal from noisy measurements belongs to signal filtering. Classical signal-processing methods, such as linear filters or wavelet transformations, are not suitable because the frequency bands of the embedded signal and strong complex signals overlap completely. Machine learning provides a potent and automated way of signal filtering, making it feasible to extract useful information from complex signals. In general, neural networks can be powerful for processing complicated signals with nonlinear properties. For example, convolutional

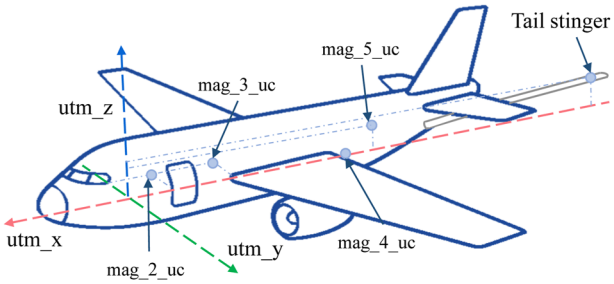

FIG. 1. Configuration of magnetometers on the aircraft. The signals from the magnetometers inside the airplane contain strong complex signals produced by various electronic devices. The signal from the magnetometer placed at the tail stinger is free from these overwhelming complex signals, which, after the TL calibration, leads to the true magnetic anomaly field signal. The measurements from the other four magnetometers contain the anomaly field embedded in strong complex signals. The datasets are from test flights conducted by Sanders Geophysics Ltd. (SGL) near Ottawa, Canada. (a) Random forest—an ensemble of decision trees. (b)An in depth look at a sample decision tree in a random forest.

neural networks (CNNs)[35–37] have been used in signal filtering tasks, including denoising electroencephalography (EEG) signals or removing motion artifacts from magnetic resonance imaging (MRI) data.[38] Here, we describe three machine-learning methods used in our work.

K-nearest neighbor (KNN) approach: The KNN approach[39] is a non-parametric method used for regression and classification. The value of an instance in KNN regression is determined by the mean or median of the k closest instances. The user selecting the parameter k establishes how many nearest neighbors should be utilized to produce a forecast. This enables the method to model nonlinear connections and capture complex patterns in data without making any assumptions about the distribution of the underlying data—a key benefit for certain problems.[40–42] The KNN method has several limitations, such as sensitivity to the choice of the distance metric, curse of dimensionality, and high computational complexity. To address these limitations, the decision-tree and random-forest methods can be used.

Decision tree: A decision tree is also a regressor and classifier that requires iteratively segmenting the feature space into regions and constructing a tree-like structure performing prediction for each smaller portion.[4,5] The feature space is recursively split into two halves, and a tree is constructed using the features that provide the largest sum-squared error reduction. The resulting structure is a binary tree, with the leaves designating the final regions and the projected value being the mean or median of the target variable for the data points in that region. One advantage of decision tree regression is interpretability since the resulting tree structure can be visualized for understanding. Decision trees can also handle nonlinear and multi-dimensional data and are robust to noise and outliers. Yet, decision tree regression is susceptible to overfitting when the tree becomes complex and the data are noisy. To overcome the overfitting problem, pruning, early pausing, and regularization can be used to improve the generalization performance of a decision tree. Decision tree regression has been applied in a variety of fields, including engineering, finance, and environmental sciences, for applications such as building energy consumption modeling and air pollution forecasting.[43–47]

Random forest: As shown in Fig. 2, a random forest is an ensemble of decision trees,[1–3] each trained using a different subset of characteristic features and data points. Techniques such as weighted voting or stacking can also be employed for constructing a random forest. A key benefit of a random forest is its ability to prevent overfitting since the ensemble of trees balances out the noise and variability in the data. Random forest is also resilient against noise and outliers. However, when there are too many or too few trees in the ensemble, a random forest may suffer from bias and correlation

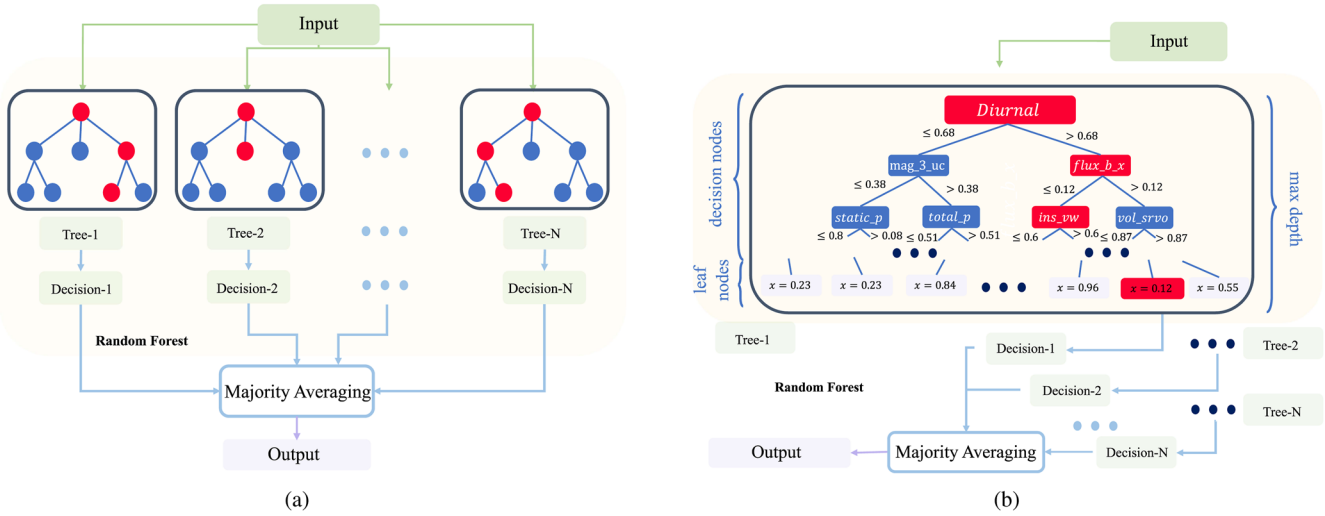

FIG. 2. A schematic illustration of a random forest of decision trees. (a) A random forest and (b) a decision tree in the "forest." Prediction of the target variable of interest is achieved by taking the average of median of the predictions from all the trees in the forest.

and may not work well with data that has complex dependencies or nonlinear interactions.

C. Simulation hardware and software

Our simulations were carried out on a desktop system with one NVIDIA GeForce GTX 750 Ti GPU, an Intel Core i7-6850K CPU @ 3.60GHZ, and 128 GB of RAM. During the training process, the *n_jobs* keyword (from the *model.fit* method) was set to −1, so the simulations were done using all 32 available logical cores. All codes were written in Python, where we used *sklearn*—a machine learning python package—to train and test our algorithms.

D. Data pre-processing

We use real data selected from several flights (number 1002–1007) conducted by Sanders Geophysics Ltd. (SGL) near Ottawa, Canada. For instance, the dataset of flight 1002 consists of 207 580 instances with the sampling time $dt = 0.1$ s, each comprised of 102 features from a collection of various sensor measurements, which are voltage, current, magnetic, and other sensors as well as the positions, INS, and avionics system readings. The position of the aircraft is derived from WGS xyz coordinates included in the dataset, which are GPS positions, and is the predicted target of the model, while the other features [from selected features or those from a principal-component analysis (PCA)] are used as the inputs to the random-forest models. For random-forest models, the number of estimators (trees) is selected to be 100. The dataset is used to train the models in order to filter the position out of the available sensor data. The performance is evaluated using the predicted root-mean square errors (RMSEs) on a head-out unseen test set.

The original data contained missing values, outliers, and other anomalies, rendering necessary extensive pre-processing. In general, machine learning methods require normalizing the data as an essential pre-processing step.[48] A simple method is to normalize the dataset so that all the features have zero mean and unit variance. Another method is minmax, where each feature is scaled individually such that it falls in a given range of the training set, e.g., between zero and one. Normalization in signal filtering also helps in removing bias in the signal that may be brought on by variations in scale or units among the features. In our case, normalizing the data will ensure that any two features with different scales, e.g., one measured in nano-Tesla and the other in Ampere, have the same scale. We find that, for our datasets, the minmax scalar method outperforms the standard scalar normalization. Because of the remaining variance after minmax scaling, it is useful to remove features with low variance since they contribute little to the process of filtering.[49] This can be done by setting a proper threshold value of the variance. Removing the features with variances below the threshold brings additional benefits, such as speeding up feature selection, reducing overfitting, and making the model and results more interpretable. Table I sorts the features of the normalized flight data (including flight number 1002–1007) in terms of their standard deviation. As a reasonable assumption, the exclusion variance threshold is set as 0.0025. For the flight data, this means that the features *cur_flap*, *mag_2_uc*, and *cur_com_1* are removed at this step.

TABLE I. Standard deviations of the features of the entire flight dataset.

Feat. with lowest std	Std	Feat. with highest std	Std
cur_flap	0.0077	*total_p*	0.3438
mag_2_uc	0.0282	*cur_ac_lo*	0.3345
cur_com_1	0.0441	*static_p*	0.3344
ins_acc_z	0.0545	*baro*	0.3340
nrml_acc	0.0569	*ins_alt*	0.3323
roll_rate	0.0590	*utm_z*	0.3322
ltrl_acc	0.0715	*msl*	0.3321
pitch_rate	0.0718	*diurnal*	0.3123
mag_1_igrf	0.0738	*ins_vw*	0.3063
cur_srvo_o	0.0798	*ins_vn*	0.2963

IV. RESULTS

A. Feature selection

A key step is sequential feature selection in which features are added to or removed from the features set in accordance with a predefined criterion, such as RMSE, $R2$, or $F1$ score (depending on the application), until an optimal subset of features is found.[50] Here, optimality means finding the smallest set of features, leading to the required predictive performance, which not only makes the model more understandable but also reduces overfitting, increases prediction accuracy, and results in a computationally efficient training process. Given a set of features, optimal selection can be performed forward, backward, or a mix of both. Because of the relatively large number of features in our dataset, we exploit the forward selection algorithm, which means that, at each step, the algorithm selects the best feature to add or remove based on the cross-validation score of the trained random-forest model. Our computation leads to 12 features: *mag_3_uc*, *mag_4_uc*, *mag_5_uc*, *diurnal*, *flux_b_x*, *flux_b_y*, *flux_c_y*, *ins_vw*, *ins_wander*, *static_p*, *total_p*, and *vol_srvo*. Note that the *ins_lon*, *ins_lat*, and *ins_alt* features are excluded from the pool because they are equivalent to the target positions, as shown in Fig. 3. For the INS free method, the *ins_vw* and *ins_wander* features are removed. Moreover, for the TL aided model, the *mag_3_uc*, *mag_4_uc*, and *mag_5_uc* features are replaced by *mag_3_c*, *mag_4_c*, and *mag_5_c*, which are the compensated values of the magnetometers after application of the TL method. The details and description of the employed features for each method are listed in Table II.

We apply PCA to further process the feature data, where the components are sorted by the eigenvectors in the order of their corresponding eigenvalues from the highest to the lowest. There are two reasons for the PCA analysis. First, for the selected features to be effective, the correlations among them cannot be too high nor too low. In particular, high correlations can produce "multicollinearity," where numerous independent variables in a model are interrelated, making it difficult to predict the individual impact of each feature on the dependent variable. Figure 3 displays the correlation among the selected features, where it can be seen that the features associated with the INS sensors have high correlations. For our magnetic navigation problem, the degree and direction of the association among the sensor readings and the aircraft position are affected by feature correlations. The highly correlated features can be integrated into

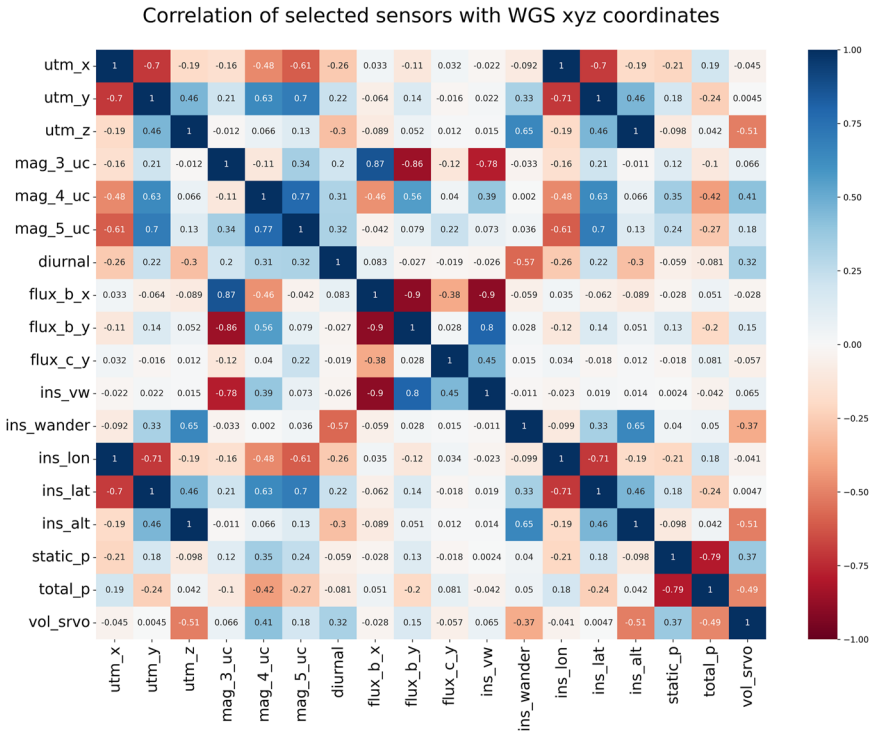

FIG. 3. Correlation among the selected features in Table II. The information from the correlation is used to prune the redundant features. High correlation among the features can produce "multicollinearity," where numerous independent variables in a model are interrelated, making it challenging to predict the individual impact of each feature on the dependent variable.

TABLE II. Selected features using forward sequential feature selection on the entire flight dataset.

Feat	Description	TL INS free	INS aided	TL INS aided
mag_3_uc	Uncomp. mag. sensor 3	✓	✓	✗
mag_4_uc	Uncomp. mag. sensor 4	✓	✓	✗
mag_5_uc	Uncomp. mag. sensor 5	✓	✓	✗
mag_3_c	Comp. mag. sensor 3	✗	✗	✓
mag_4_c	Comp. mag. sensor 4	✗	✗	✓
mag_5_c	Comp. mag. sensor 5	✗	✗	✓
diurnal	Measured diurnal	✓	✓	✓
flux_b_x	Fluxgate B x axis	✓	✓	✓
flux_b_y	Fluxgate B y axis	✓	✓	✓
flux_c_y	Fluxgate C y axis	✓	✓	✓
ins_vw	INS west velocity	✗	✓	✓
ins_wander	INS wander angle	✗	✓	✓
static_p	Avionics static pressure	✓	✓	✓
total_p	Avionics total pressure	✓	✓	✓
vol_srvo	Volt. sensors: servos	✗	✓	✓

new, uncorrelated features named principal components through PCA for dimension reduction, which can then be used as inputs to our machine-learning-based filter to remove the strong complex signals and redundant data. Second, PCA can improve visualization and help better understand the data by transforming the original high-dimensional data into low-dimensional ones. To establish a proper trade-off between dimensionality reduction and information retention, we consider the number of main components beyond which feature reduction can lead to information loss. Figures 4(a) and 4(b) show the first two and three components of PCA applied to the set of selected features (without TL and INS), respectively, where the color elements represent the normalized Euclidean distance of the dataset instances from the origin. It can be seen from the PCA components that there exist distinct clusters in the data in terms of the Euclidean distance.

We remark that nonlinear feature reduction methods, such as Isomap and Kernel PCA, are not suitable for our problem because of the large sizes of the datasets. In particular, to compute the kernel matrix of size $(data\_samples, data\_samples)$, it is necessary to compute and store $data\_samples^2$ number of terms. For our datasets, this requires about 312 GB of computer RAM. A potential solution is to perform clustering on the dataset and fill the kernel with the means of those clusters. However, even this approach might produce a large kernel matrix.

B. Random-forest-based detection of weak anomaly magnetic field

We demonstrate the power of our random-forest method to detect weak anomaly field from data. For comparison, we also

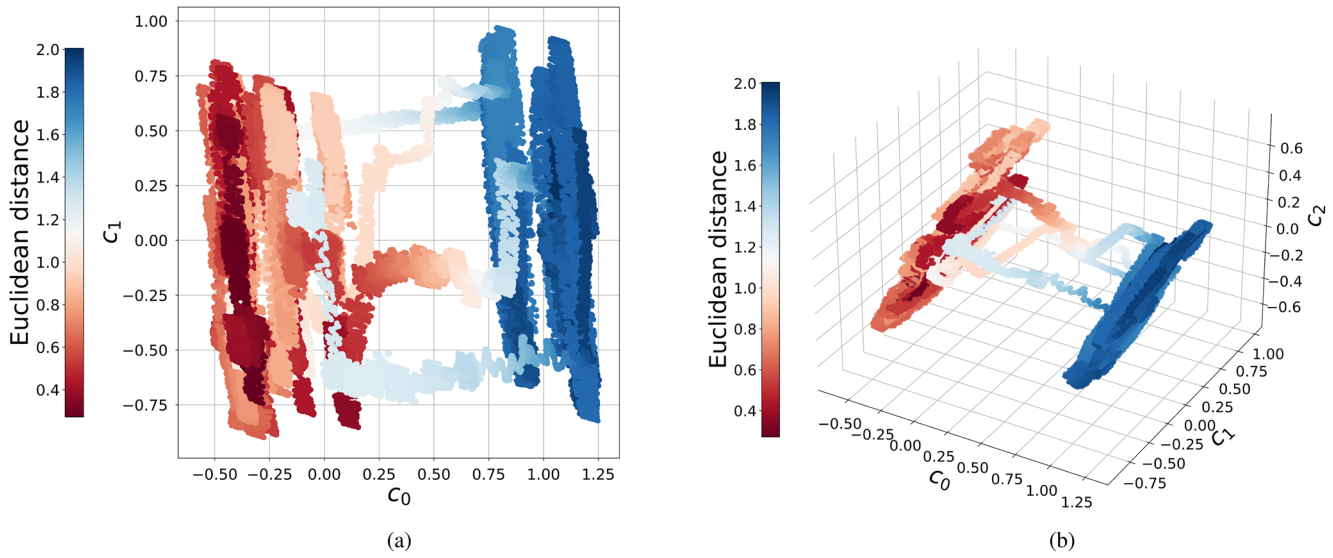

FIG. 4. Colored representation of the normalized Euclidean distance of the dataset instances from the origin. Distinct clusters in the data in a relative Euclidean distance can be observed. (a)The first two components of the PCA transformation of the entire flight dataset in a 2D view. (b)The first three components of the PCA transformation of the entire flight dataset in a 3D view.

display the results from the KNN and decision-tree methods. Each machine-learning model uses the chosen optimal set of 12 features as input and the magnetic anomaly field signal as the output. To ensure a fair comparison, we perform hyperparameter tuning and calculate the average RMSE of the magnetic anomaly field for different flight lines. Figure 5 shows that, for our random-forest model, the average test RMSE is about 1.9 nT for all flights. Table III lists the average RMSEs from the different machine-learning methods. It can be seen that the KNN method has the largest error and the decision tree method suffers from overfitting, but it is overcome by the random-forest method. These results indicate that random forests is a reliable approach for detecting weak magnetic anomaly fields from data.

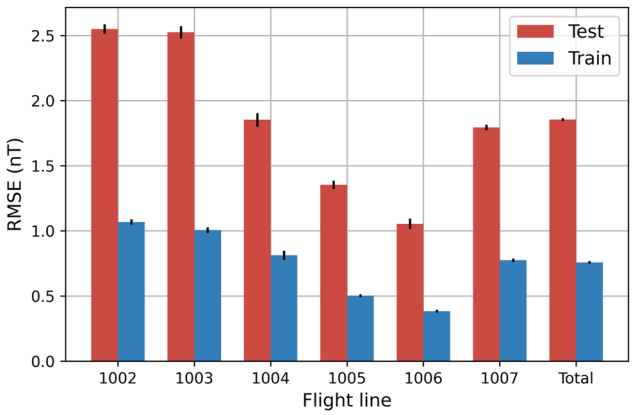

FIG. 5. Results of random-forest-based detection of weak signals. Selected features are used to detect the weak anomaly magnetic field signal. Compared with a recent work on this topic based on the machine-learning methods of reservoir computing and feed forward neural networks,[18] the average RMSEs from the random-forest method are reduced by over 100%.

C. Random-forest-based precise positioning

For navigation positioning, to ensure that the appropriate features are selected, we carry out a feature-importance analysis. For a tree-based method, such as decision tree and random forests, it is practically impossible to analyze the feature importance by the model weights. Feature importance analysis is crucial for decision trees and random forests. Traditional metrics, such as Gini impurity[51] and mean decrease in impurity,[52] are often used. Combining these techniques with domain expertise and cross-validation can lead to a more comprehensive understanding of

TABLE III. Performance comparison among KNN, decision-tree, and random-forest methods in terms of RMSEs using selected features to detect the weak anomaly magnetic field signal (in units of nT).

Flight line	Random forest train	Random forest test	KNN train	KNN test	Decision tree train	Decision tree test
1002	1.05	2.57	40.94	53.23	1.92	5.01
1003	1.05	2.60	19.29	25.68	1.17	3.94
1004	0.82	1.83	15.43	19.98	0.36	3.33
1005	0.50	1.31	9.46	12.74	0.58	3.16
1006	0.36	0.95	12.29	16.35	0.91	2.23
1007	0.80	1.84	18.18	23.34	1.27	3.68
All	0.93	2.58	26.24	33.79	4.01	6.10

feature importance. We use two standard methods: permutation and dropping.[53] In feature permutation, the performance of the model is assessed by randomly permuting the values of a single feature and comparing the results with a baseline model: the more the performance is degraded, the more important that feature is. In the feature-dropping approach, the model's performance is assessed by eliminating one feature at a time and monitoring how the result changes. Each feature's importance is deduced from the performance drop that results from the removal of that certain feature. These methods are computationally efficient for large datasets, and they serve to evaluate the significance of each feature without any knowledge about the intrinsic dynamics of the trained machine learning model, making them suitable for a variety of applications.

Figures 6 and 7 illustrate the feature importance analysis while performing the random-forest filtering of the position of the aircraft using the selected features in Table II. It can be seen that, for the INS free case, both the permutation and dropping methods give that the *diurnal* feature is the most significant. Note that the *diurnal* feature can only be measured independently with an external base-station and will not generally be available on the aircraft. There are some approaches to modeling it based on past history or using a statistical approach that could be used. For the INS aided model, the *ins_wander* feature is the most important since the performance degrades significantly in its absence compared to the baseline method.

We compare the performance of three methods: KNN, decision tree, and random forests, in positioning using both the selected features (Table II) and PCA features in terms of RMSEs from training and testing sets on different flight datasets. An important initial step is to determine the hyperparameter values. In general machine learning, the hyperparameters are those that cannot be learned from the data but must be predetermined before training,

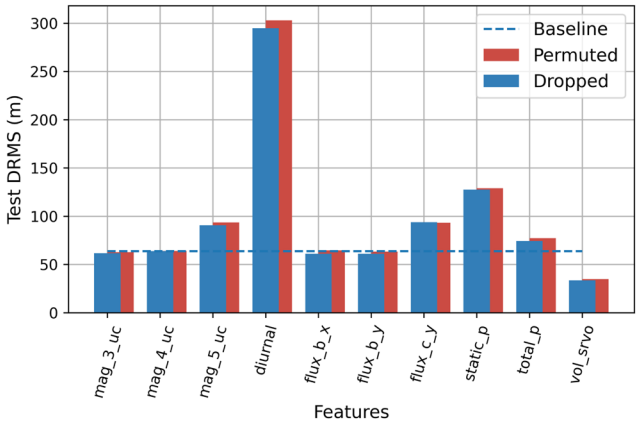

FIG. 6. Importance analysis of selected features in the absence of INS in terms of DRMS (which stands for Distance RMS that is the root mean square of the radial distances from the true position to the predicted positions). Random forest filtering of the position of the aircraft using the selected features in Table II is performed on a testing dataset. Both the permutation and dropping methods give that the feature *diurnal* is the most significant. Other selected features are less significant, and removing the vol_srvo feature can improve the performance.

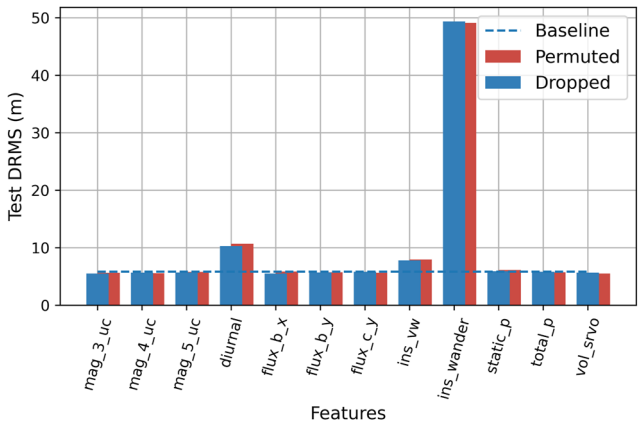

FIG. 7. Importance analysis of selected features with INS. The position of the aircraft was filtered using a random-forest algorithm on a test set utilizing the selected features in Table II. Both the permutation and dropping methods give that the *ins_wander* feature is the most significant. The second most significant feature is *diurnal*.

the selection of which is critical to the success of machine learning. For the decision tree and KNN methods, we use PyGad[54] to tune the hyperparameter values. The random-forest method has a single hyperparameter—max_*depth*. Figure 8 shows the RMSE vs max_*depth*, which gives that the optimal value of max_*depth* is around 25.

Table IV summarizes the best-performance results with INS data in terms of DRMS, the distance between the actual and predicted positions of the aircraft. The *all* flight line means that the entire dataset in the training and testing is used, whereas *all but* 1005 denotes that the entire dataset except that from flight number 1005 is used for training, but the test is performed on the entire dataset, including flight number 1005. The results show that in all cases, the selected features perform better than the PCA features. Remarkably, random forests outperform the other two methods in the testing DRMS. For the decision-tree method, overfitting is pronounced

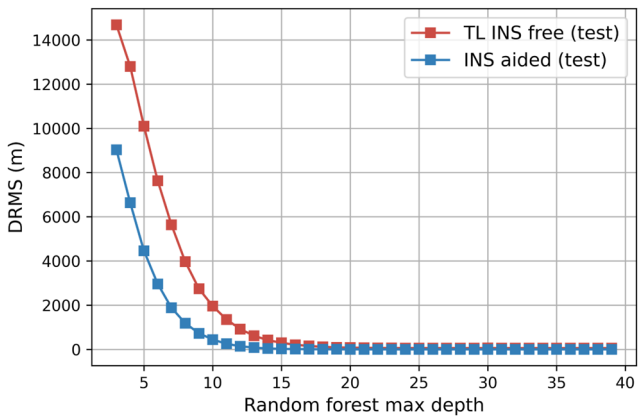

FIG. 8. Determining the hyperparameter for random forests. Shown is the RMSE vs max_*depth*, the depth of the forest (single hyperparameter). The optimal value of max_*depth* is around 25.

TABLE IV. Performance comparison in terms of DRMS among KNN, decision tree, and random-forest methods using PCA and selected features for different flight datasets using the INS aided method (in units of meters).

Flight line	Use PCA	Use selected features	Random forest Train	Random forest Test	KNN Train	KNN Test	Decision tree Train	Decision tree Test
1002	✗	✓	2.12	6.00	30.62	42.14	1.34	8.50
1003	✗	✓	1.94	4.75	37.21	53.11	2.46	9.61
1004	✗	✓	1.70	3.82	33.15	47.63	0.68	7.76
1005	✗	✓	1.35	3.07	19.83	29.11	0.13	7.19
1006	✗	✓	2.02	5.48	25.85	36.78	13.52	20.84
1007	✗	✓	1.55	4.05	38.89	53.26	0.41	7.71
All	✗	✓	2.54	6.51	115.96	159.9	48.48	53.90
All but 1005	✗	✓	2.74	7.02	103.17	125.93	73.25	82.06
1002	✓	✗	726.59	2370.4	869.7	1277.05	745.92	2924.68
1003	✓	✗	252.08	2355.1	417.7	2506.68	466.07	2760.00
1004	✓	✗	171.21	738.1	264.7	577.04	121.29	946.71
1005	✓	✗	213.53	840.0	515.2	742.15	42.10	1107.38
1006	✓	✗	75.48	347.5	139.4	217.58	10.24	415.45
1007	✓	✗	246.63	1759.5	524.7	1823.29	233.97	1830.46
All	✓	✗	663.09	5324.7	1075.9	5582.99	1835.13	6108.65
All but 1005	✓	✗	547.7	17 501.4	783.73	547.75	1949.34	19 254.25

because it performs better than random forests in training but worse in testing. Figure 9 shows the results of the compensated values from magnetic sensors from the TL model and INS sensors. Figures 10 and 11 display the DRMS results when INS data are used and excluded, respectively. There are significant variations in the errors across the three methods. For example, the INS aided method consistently performs better than the other two methods, while the TL INS free aided method outperforms the TL INS aided method. An implication is that incorporating INS sensors generally leads to improved accuracy. For all three methods, the test errors are consistently lower than the training errors, indicating that the models are capable of generalization from training to test data. The INS aided method can be a reliable alternative that provides good accuracy with less reliance on a specific model. The INS free methods, while less accurate, are suitable for applications where INS sensors are unavailable or too costly to implement. Another advantage of the INS free method is that it is independent of the TL model.

Table V summarizes the positioning results in terms of the mean and standard deviations from the entire experiment for different methods. A comparison between the TL INS free and INS aided methods indicates that using the TL model tends to degrade the positioning performance, while the INS aided method gives the best performance.

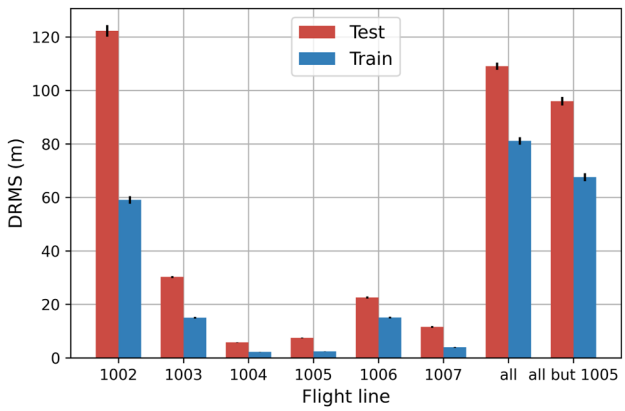

FIG. 9. Results from the TL INS method. The compensated values from the magnetic sensors, the Tolles–Lawson model, and INS sensors are used. The comparison between the TL INS free and INS aided methods suggests that including the TL model tends to degrade the positioning performance.

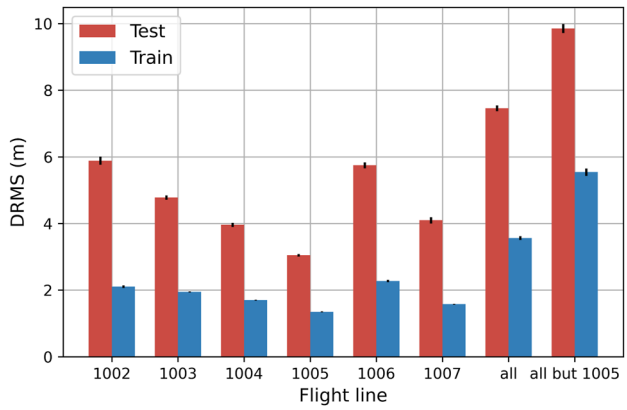

FIG. 10. Results from the INS aided method. The uncompensated values from the magnetic sensors and INS sensors are used. The INS aided method gives the best performance.

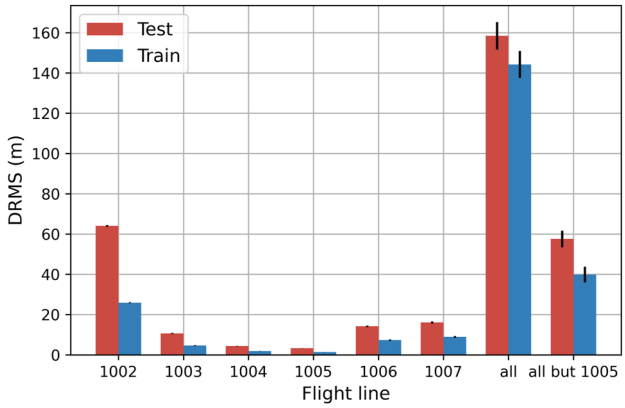

FIG. 11. Results from the INS free method. The uncompensated values from magnetic sensors using the Tolles–Lawson model and INS sensors are not used. While the INS aided method gives the best performance, the INS free method does not rely on the TL model or INS sensors.

TABLE V. Mean and standard deviation of RMSE from the entire experiment for different methods (in units of meters).

Method	Test mean	Test std	Train mean	Train std
TL INS Free	41.08	2.50	29.26	2.46
INS Aided	5.60	0.03	2.51	0.03
TL INS Aided	50.62	0.78	30.79	0.66

V. DISCUSSION

Exploiting machine learning to detect weak physical signals immersed in strong complex signals has attracted a recent interest with applications such as extracting the Earth's anomaly magnetic field from overwhelmingly noisy signals collected from the magnetic sensors installed inside the cockpit of a flying airplane. Previously, the specific machine-learning architectures explored for this application were reservoir computing (recurrent neural networks) and time-delayed feed forward neural networks.[18] It was demonstrated that, when combined with the classical TL model for removing the aircraft magnetic field, the weak anomaly magnetic field can be reliably detected. Detecting the anomaly field has a direct application in magnetic navigation in a GPS-denied environment.[16,17] For the navigation problem, the goal is precise positioning: obtaining the instantaneous position of the flying airplane, a problem that was not addressed in the recent work,[18] raising the need for a more general approach to magnetic navigation.

The present work develops a random-forest-based machine-learning approach to magnetic navigation, where the problems of detecting the weak anomaly magnetic field and determining the position of the flying airplane are solved simultaneously. When many feature signals are present, as in the flying data available to us, a vast and complex "random forest" can be constructed and trained. The random forest contains an enormously large number of distinct paths (from the "root" to the "leafs"), each being associated with a specific value of the anomaly magnetic field and the position of the airplane. In the testing phase when the information about the anomaly field and the position is not available, from a set of features signals, a well-trained random forest can be efficiently searched to yield the anomaly field and the position. As many available feature signals contain redundant information, we performed a process to select the optimal set of feature signals so as to greatly improve the computational efficiency. As random forest is essentially an "intelligent" book-keeping machine, all required for training is the selected set of feature signals and the corresponding anomaly field and position, thereby removing the need of calibration methods, e.g., the TL model. Indeed, we demonstrated that high accuracies in the anomaly field and position can be achieved even without TL calibration.

More generally, the random-forest framework developed here can be applied to signal filtering, an important task in many fields, including image processing, speech recognition, and economic forecasting. There are many challenges in designing effective filtering algorithms, such as dealing with noise, nonstationary signals, and nonlinear dependencies. The success of deploying random forests to magnetic navigation reported here can serve as a starting point to generalize the machine-learning model to other applications. For the broad task of signal filtering, a potential future research direction could be to investigate deep learning methods, such as convolutional neural networks, specially temporal graph convolutional neural networks.[55] Additionally, transfer learning could be explored, where a pre-trained model is fine-tuned on a new dataset for signal filtering. In this case, the knowledge learned from previous flights can be employed in learning new flight data, thereby requiring shorter flight times. Furthermore, the development of online learning algorithms for signal filtering could be explored, which could adapt to changes in the signal over time.

ACKNOWLEDGMENTS

This work was supported by AFOSR under Grant No. FA9550-21-1-0438.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Mohammadamin Moradi: Conceptualization (lead); Data curation (equal); Investigation (lead); Methodology (lead); Writing – original draft (equal); Writing – review & editing (supporting). **Zheng-Meng Zhai**: Conceptualization (supporting); Data curation (equal); Investigation (supporting); Methodology (supporting). **Aaron Nielsen**: Data curation (equal); Investigation (supporting); Methodology (supporting). **Ying-Cheng Lai**: Conceptualization (equal); Funding acquisition (equal); Investigation (supporting); Methodology (supporting); Project administration (lead); Supervision (lead); Writing – original draft (equal); Writing – review & editing (lead).

DATA AVAILABILITY

The data that support the findings of this study are openly available at https://github.com/AminMoradiXL/magnav.

REFERENCES

[1] L. Breiman, "Random forests," Mach. Learn. **45**, 5–32 (2001).
[2] S. J. Rigatti, "Random forest," J. Insur. Med. **47**, 31–39 (2017).
[3] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random forests," Ensemble Mach. Learn.: Methods Appl. **45**, 157–176 (2011).
[4] J. Su and H. Zhang, "A fast decision tree learning algorithm," AAAI **6**, 500–505 (2006).
[5] Y. Freund and L. Mason, "The alternating decision tree learning algorithm," ICML **99**, 124–133 (1999).
[6] T. Hesterberg, "Bootstrap," Wiley Interdiscip. Rev. Comput. Stat. **3**, 497–526 (2011).
[7] M. C. E. Simsekler, A. Qazi, M. A. Alalami, S. Ellahham, and A. Ozonoff, "Evaluation of patient safety culture using a random forest algorithm," Reliab. Eng. Syst. Saf. **204**, 107186 (2020).
[8] C. Iwendi, A. K. Bashir, A. Peshkar, R. Sujatha, J. M. Chatterjee, S. Pasupuleti, R. Mishra, S. Pillai, and O. Jo, "Covid-19 patient health prediction using boosted random forest algorithm," Front. Public Health **8**, 357 (2020).
[9] M. A. Manzoor and Y. Morgan, "Vehicle make and model recognition using random forest classification for intelligent transportation systems," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (IEEE, 2018), pp. 148–154.
[10] J. Urbancic, V. Pejovic, and D. Mladenic, *Transportation Mode Detection using Random Forest* (Information Society, Data Mining and Data Warehouses SiKDD, Ljubljana, Slovenia, 2018).
[11] L. Zhang, N. R. Alharbe, G. Luo, Z. Yao, and Y. Li, "A hybrid forecasting framework based on support vector regression with a modified genetic algorithm and a random forest for traffic flow prediction," Tsinghua Sci. Technol. **23**, 479–492 (2018).
[12] Y. Ma, R. Han, and X. Fu, "Stock prediction based on random forest and LSTM neural network," in *2019 19th International Conference on Control, Automation and Systems (ICCAS)* (IEEE, 2019), pp. 126–130.
[13] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," Mach. Learn. **63**, 3–42 (2006).
[14] N. Meinshausen, "Quantile regression forests," J. Mach. Learn. Res. **7**, 983–999 (2006).
[15] C. Nathe, C. Pappu, N. A. Mecholsky, J. Hart, T. Carroll, and F. Sorrentino, "Reservoir computing with noise," Chaos **33**, 041101 (2023).
[16] A. Canciani and J. Raquet, "Absolute positioning using the earth's magnetic anomaly field," Navigation **63**, 111–126 (2016).
[17] A. J. Canciani, "Magnetic navigation on an F-16 aircraft using online calibration," IEEE Trans. Aerosp. Electron. Syst. **58**(1), 420–434 (2022).
[18] Z.-M. Zhai, M. Moradi, L.-W. Kong, and Y.-C. Lai, "Detecting weak physical signal from noise: A machine-learning approach with applications to magnetic-anomaly-guided navigation," Phys. Rev. Appl. **19**, 034030 (2023).
[19] A. R. Gnadt, J. Belarge, A. Canciani, G. Carl, L. Conger, J. Curro, A. Edelman, P. Morales, A. P. Nielsen, M. F. O'Keeffe *et al.*, "Signal enhancement for magnetic navigation challenge problem," arXiv:2007.12158 (2020).
[20] R. T. Ioannides, T. Pany, and G. Gibbons, "Known vulnerabilities of global navigation satellite systems, status, and potential mitigation techniques," Proc. IEEE **104**, 1174–1194 (2016).
[21] M. Kayton and W. R. Fried, *Avionics Navigation Systems* (John Wiley & Sons, 1997).
[22] M. J. Veth, *Fusion of Imaging and Inertial Sensors for Navigation* (Air Force Institute of Technology, 2006).
[23] C. C. Liebe, "Accuracy performance of star trackers - A tutorial," IEEE Trans. Aerosp. Electron. Syst. **38**, 587–599 (2002).
[24] P. Chen, Y. Li, Y. Su, X. Chen, and Y. Jiang, "Review of AUV underwater terrain matching navigation," J. Navig. **68**, 1155–1172 (2015).
[25] J. A. Richeson, "*Gravity gradiometer aided inertial navigation within non-GNSS environments*," in *Proceedings of 20th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2007)* (The Institute of Navigation, 2007), pp. 1089–1100.
[26] M. Mandea and M. Korte, *Geomagnetic Observations and Models* (Springer, 2010), Vol. 5.
[27] W. Erdmann, H. Kmita, J. Z. Kosicki, and Ł. Kaczmarek, "How the geomagnetic field influences life on earth – An integrated approach to geomagnetobiology," Origins Life Evol. Biospheres **51**, 231–257 (2021).
[28] A. Gnadt, "Machine learning-enhanced magnetic calibration for airborne magnetic anomaly navigation," in *AIAA SciTech 2022 Forum* (American Institute of Aeronautics and Astronautics, 2022), p. 1760.
[29] W. E. Tolles and J. Lawson, *Magnetic Compensation of Mad Equipped Aircraft* (Airb Institute Lab Inc., Mineola, NY, 1950) Rept, 201–1.
[30] W. E. Tolles, "Compensation of aircraft magnetic fields," US Patent 2,692,970, 1954.
[31] Q. Han, Z. Dou, X. Tong, X. Peng, and H. Guo, "A modified Tolles–Lawson model robust to the errors of the three-axis strapdown magnetometer," IEEE Geosci. Remote Sens. Lett. **14**, 334–338 (2017).
[32] A. R. Gnadt, "Advanced aeromagnetic compensation models for airborne magnetic anomaly navigation," Ph.D. thesis, Massachusetts Institute of Technology, 2022.
[33] P. M. Williams, "Aeromagnetic compensation using neural networks," Neural Comput. Appl. **1**, 207–214 (1993).
[34] M. C. Hezel, *Improving Aeromagnetic Calibration Using Artificial Neural Networks* (AFIT Scholar, 2020).
[35] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," arXiv:1511.08458 (2015).
[36] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," IEEE Trans. Neural Netw. Learn. Syst. **33**(12), 6999–7019 (2022).
[37] P. Kim and P. Kim, "Convolutional neural network," *MATLAB Deep Learning: With ML, Neural Networks, and AI* (Springer, 2017), pp. 121–147.
[38] J. Qezelbash-Chamak, S. Badamchizadeh, K. Eshghi, and Y. Asadi, "A survey of machine learning in kidney disease diagnosis," Mach. Learn. Appl. **10**, 100418 (2022).
[39] L. E. Peterson, "K-nearest neighbor," Scholarpedia **4**, 1883 (2009).
[40] R. Goyal, P. Chandra, and Y. Singh, "Suitability of KNN regression in the development of interaction based software fault prediction models," IERI Proc. **6**, 15–21 (2014).
[41] S. Kohli, G. T. Godwin, and S. Urolagin, "Sales prediction using linear and KNN regression," in *Advances in Machine Learning and Computational Intelligence: Proceedings of ICMLCI 2019* (Springer, 2021), pp. 321–329.
[42] Y. Song, J. Liang, J. Lu, and X. Zhao, "An efficient instance selection algorithm for *k* nearest neighbor regression," Neurocomputing **251**, 26–34 (2017).
[43] K. Ramya, Y. Teekaraman, and K. A. Ramesh Kumar, "Fuzzy-based energy management system with decision tree algorithm for power security system," Int. J. Comput. Intell. Syst. **12**, 1173–1178 (2019).
[44] Z. Yu, F. Haghighat, B. C. Fung, and H. Yoshino, "A decision tree method for building energy demand modeling," Energy Build. **42**, 1637–1646 (2010).
[45] X. Luo, J. Xia, and Y. Liu, "Extraction of dynamic operation strategy for standalone solar-based multi-energy systems: A method based on decision tree algorithm," Sustainable Cities Soc. **70**, 102917 (2021).
[46] M.-C. Wu, S.-Y. Lin, and C.-H. Lin, "An effective application of decision tree to stock trading," Expert Syst. Appl. **31**, 270–274 (2006).
[47] B. B. Nair, V. Mohandas, and N. Sakthivel, "A decision tree- rough set hybrid system for stock market trend prediction," Int. J. Comput. Appl. **6**, 1–6 (2010).
[48] B. M. Bolstad, R. A. Irizarry, M. Åstrand, and T. P. Speed, "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias," Bioinformatics **19**, 185–193 (2003).
[49] M. J. Alam, P. Kenny, and D. O'Shaughnessy, "A study of low-variance multitaper features for distributed speech recognition," in *Proceedings of 5th International Conference on Nonlinear Speech Processing, NOLISP 2011* (Springer, 2011), pp. 239–245.
[50] D. W. Aha and R. L. Bankert, "A comparative evaluation of sequential feature selection algorithms," in *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics* (PMLR, 1995), pp. 1–7.
[51] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique," Cybersecurity **5**, 1 (2022).

[52]H. Han, X. Guo, and H. Yu, "Variable selection using mean decrease accuracy and mean decrease Gini based on random forest," in *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)* (IEEE, 2016), pp. 219–224.
[53]H. Kaneko, "Cross-validated permutation feature importance considering correlation between features," Anal. Sci. Adv. **3**, 278–287 (2022).
[54]M. Suganuma, S. Shirakawa, and T. Nagao, "A genetic programming approach to designing convolutional neural network architectures," in *Proceedings of the Genetic and Evolutionary Computation Conference* (Association for Computing Machinery, 2017), pp. 497–504.
[55]L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-GCN: A temporal graph convolutional network for traffic prediction," IEEE Trans. Intell. Trans. Syst. **21**, 3848–3858 (2020).