applied sciences

MDPI

Article

Assessing the Performance of Machine Learning Algorithms for Soil Classification Using Cone Penetration Test Data

Ayele Tesema Chala * and Richard Ray

Structural and Geotechnical Engineering Department, Faculty of Architecture, Civil and Transport Sciences, Szechenyi Istvan University, Egyetem ter 1, H-9026 Gyor, Hungary

* Correspondence: chala.ayele.tesema@hallgato.sze.hu

Abstract: Conventional soil classification methods are expensive and demand extensive field and laboratory work. This research evaluates the efficiency of various machine learning (ML) algorithms in classifying soils based on Robertson's soil behavioral types. This study employs 4 ML algorithms, including artificial neural network (ANN), random forest (RF), support vector machine (SVM), and decision trees (DT), to classify soils from 232 cone penetration test (CPT) datasets. The datasets were randomly split into training and testing datasets to train and test the ML models. Metrics such as overall accuracy, sensitivity, precision, F1_score, and confusion matrices provided quantitative evaluations of each model. Our analysis showed that all the ML models accurately classified most soils. The SVM model achieved the highest accuracy of 99.84%, while the ANN model achieved an overall accuracy of 98.82%. The RF and DT models achieved overall accuracy scores of 99.23% and 95.67%, respectively. Additionally, most of the evaluation metrics indicated high scores, demonstrating that the ML models performed well. The SVM and RF models exhibited outstanding performance on both majority and minority soil classes, while the ANN model achieved lower sensitivity and F1_score for minority soil class. Based on these results, we conclude that the SVM and RF algorithms can be integrated into software programs for rapid and accurate soil classification.

Keywords: cone penetration test; soil classification; machine learning; artificial neural network; support vector machine; random forest; soil behavioral type

check for updates

Citation: Chala, A.T.; Ray, R. Assessing the Performance of Machine Learning Algorithms for Soil Classification Using Cone Penetration Test Data. *Appl. Sci.* **2023**, *13*, 5758. https://doi.org/10.3390/app13095758

Academic Editor: Wei Gao

Received: 29 March 2023
Revised: 2 May 2023
Accepted: 4 May 2023
Published: 6 May 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

For many years, the cone penetration test (CPT) has been the predominant method for conducting field exploration in geotechnical engineering [1–5]. This test requires a cone-shaped instrument to be inserted into the soil at a consistent penetration rate, while measuring the cone tip resistance (qc) and sleeve friction (fs). The CPT continuously provides precise, repeatable results for its entire profile depth. Moreover, the CPT is a relatively quick and inexpensive means of acquiring field data for estimating parameters for many applications, such as soil classification, environmental studies, hydrological analysis, and seismic site response assessments.

Soil classification is essential in geotechnical engineering, especially when evaluating site response to seismic events. Accurate soil classification helps to understand the dynamic properties of soil and the effects of earthquakes on the soil's behavior. The traditional soil classification based on the CPT data involves analyzing 2D charts. Early research was aimed to predict the distribution of soil particles by using the CPT measurements, as outlined in the pioneering work of Begemann [6]. However, later work by Douglas and Olsen [7] suggested that a more useful soil classification approach in practical engineering projects would involve considering soil behavior, rather than solely relying on soil particle distribution. As a result, Robertson developed soil classification charts based on a soil behavior type index using CPT measurements [4,8]. Additionally, there are alternative methods for soil classification. The Unified Soil Classification System (USCS), for example, relies on extensive field and laboratory tests to classify soil [9].

Soil classification and parameter estimation using traditional methods can be costly and time-consuming. Field and laboratory testing is required, and soil samples need to be transported to a laboratory where particle size distribution and Atterberg limits are conducted. These tests take time to complete, and the results may not be immediately available. Additionally, soil properties can significantly change with variations in the temperature and moisture content. However, in recent years, machine learning (ML) techniques have shown great promise in soil classification. Many studies have demonstrated the potential of ML techniques in soil classification based on CPT measurements [10–15].

The study conducted by [15] explored the feasibility of utilizing a general regression neural network (GRNN) to predict soil composition and overall soil type employing CPT data. The research demonstrated that the GRNN model successfully categorized soils as either coarse-grained or fine-grained. Similarly, studies have demonstrated the effectiveness of artificial neural network (ANN) models in predicting complex soil profiles [16–18]. In addition, various machine learning techniques such as random forests (RFs) [19], support vector machines (SVMs) [20,21], decision trees (DTs) [22], gradient boosting machine (GBM) [23–26], and logistic regression (LR) [27] have been utilized for a variety of geotechnical engineering applications including classification and liquefaction.

ML techniques have been widely used in various fields, including image [28–30] and speech recognition, natural language processing, and data analysis, to extract insights from large datasets. For instance, ML algorithms are used to identify objects, faces, and patterns in images, which is crucial in facial recognition, autonomous driving, and object detection in security systems. ML algorithms are also used to transcribe speech into text, enabling the creation of voice assistants, language translation software, and speech-to-text dictation tools. In addition, ML algorithms are used to analyze large datasets, uncovering patterns and insights that would be difficult or impossible to identify manually. This technique has numerous applications in finance, healthcare, and scientific research.

Although machine learning (ML) techniques have been widely applied in various fields, there has been limited research on their use in geotechnical engineering. However, researchers have started exploring the potential of ML techniques for soil classification and estimation of soil parameters using CPT data. Some geotechnical researchers applied ML techniques to predict various geotechnical properties such as landslide [31], slope stability [32–34], soil type [12], and shear wave velocity [13] utilizing CPT data.

In our study, we aim to evaluate the performance of four commonly used ML algorithms, including artificial neural network (ANN), random forest (RF), support vector machine (SVM), and decision tree (DT), for soil classification using CPT data. This study has the potential to address the gap in the existing literature and offer valuable insights into the efficacy of ML algorithms for soil classification through CPT data. Furthermore, the findings of this study could help improve the efficiency and accuracy of soil classification in geotechnical engineering, which could have significant implications for geotechnical engineering.

The selection of the ML model for a classification task is based on several factors, including desired accuracy, dataset size, generalization ability, interpretability, and robustness. For our specific soil classification problem, we chose to evaluate the performance of the ANN, DT, SVM and RF algorithms, each with its own strength and weakness. ANNs are known for their ability to capture complex non-linear relationships in the data [35], while RF is the ML algorithm that utilizes multiple decision trees to enhance the accuracy and robustness of the model [36,37]. SVM can handle high-dimensional data and nonlinear decision boundaries [38,39], and DT is easy to interpret and visualize, and can handle both categorical and numerical data. By selecting these four algorithms, we aimed to strike a balance between complexity and interpretability and compare the performance of the models. Our choice of algorithms provides a diverse set of models that can handle various aspects of the classification task, including complex relationships, high-dimensional data, and interpretability. By evaluating their performances, we hope to gain insights into which algorithm is the most suitable for our specific soil classification problem.

The performance of the ML models can be compromised by various factors if not properly addressed. One of the critical factors that can affect the models' performance is the selection of hyperparameters. By carefully selecting and tuning the hyperparameters, we can improve the models' robustness and ensure that they can perform optimally in real-world applications [40]. Grid search (technique in which sets of predefined hyperparameter values are defined) is one of the most commonly used methods for hyperparameter tuning [41]. Bayesian optimization is another approach that uses a probabilistic model to estimate the performance of different hyperparameter configurations [41,42].

ML models rely on input features to make predictions, and the quality and relevance of those features can have a significant impact on the performance of the models [43]. Performing feature importance such as permutation feature importance [44] and eliminating irrelevant features from the dataset can significantly enhance the performance of ML models. Feature importance is the process of determining the most important features in a dataset for a given model.

In ML, outliers are one of the factors that contribute to the poor performance of ML models. According to the literature [45,46], outliers are data points that deviate significantly from the surrounding data points. Abnormal data readings during CPT operations can primarily occur due to human or procedural errors, such as the addition of a rod [47]. These outliers are not representative of the actual CPT measurements and should be detected and removed during the data preprocessing stage.

The structure of this paper comprises six sections. The first section provides a detailed discussion on the background of soil classification and ML models. In Section 2, the cone penetration test is explained, while Section 3 outlines the dataset preprocessing and methodology utilized. The ML models employed in this study are briefly summarized in Section 4. In Section 5, a detailed discussion is presented on the results obtained from the ML models. Finally, Section 6 provides a summary of the main points of the study and concludes the paper by proposing recommendations for future research.

2. Cone Penetration Tests

The CPT is a widely used in situ geotechnical testing method that involves inserting a cone-shaped penetrometer into the soil and recording the soil's resistance (i.e., qc and fs) to penetration. Figure 1 visually represents a graph that plots the recorded qc, fs, and friction ratio used in this study.

The CPT and its variations, such as the CPT with pore pressure measurement (CPTu) and the seismic cone penetration test with pore pressure measurements (SCPTu), are valuable tools for various engineering applications. These tests can estimate geotechnical parameters and classify soils over a broad range of soil types, from very soft soil to weak rock. Over the past few decades, various soil behavior charts have evolved for soil classification based on CPT-measured data [1–4,48].

One such chart was developed by Robertson [8] and can be used to classify soils into different categories, such as sand, clay, silt mixture, organic soil, and more. An example of such a chart is shown in Figure 2, which illustrates the classification of soil types ranging from sensitive clays to very stiff over-consolidated (OC) clays. The chart categorizes soils into various classes or zones based on their soil behavioral type index (Ic) determined by Equation (1) [48]. Table 1 lists boundaries for classification based on Ic values. In this study, the zone numbers (see Figure 2) are directly used as ML labels as they represent the soil types in a straightforward and intuitive way.

$$I_c = \sqrt{\left(3.47 - log(q_t/p_a)^2 + (logR_f + 1.22)^2\right)} \quad (1)$$

where q_t is corrected cone resistance or CPT cone resistance q_c, p_a is atmospheric pressure in the same unit as q_c, R_f is friction ratio, and f_s is CPT sleeve friction.

$$R_f = (f_s/q_c) \times 100\% \quad (2)$$

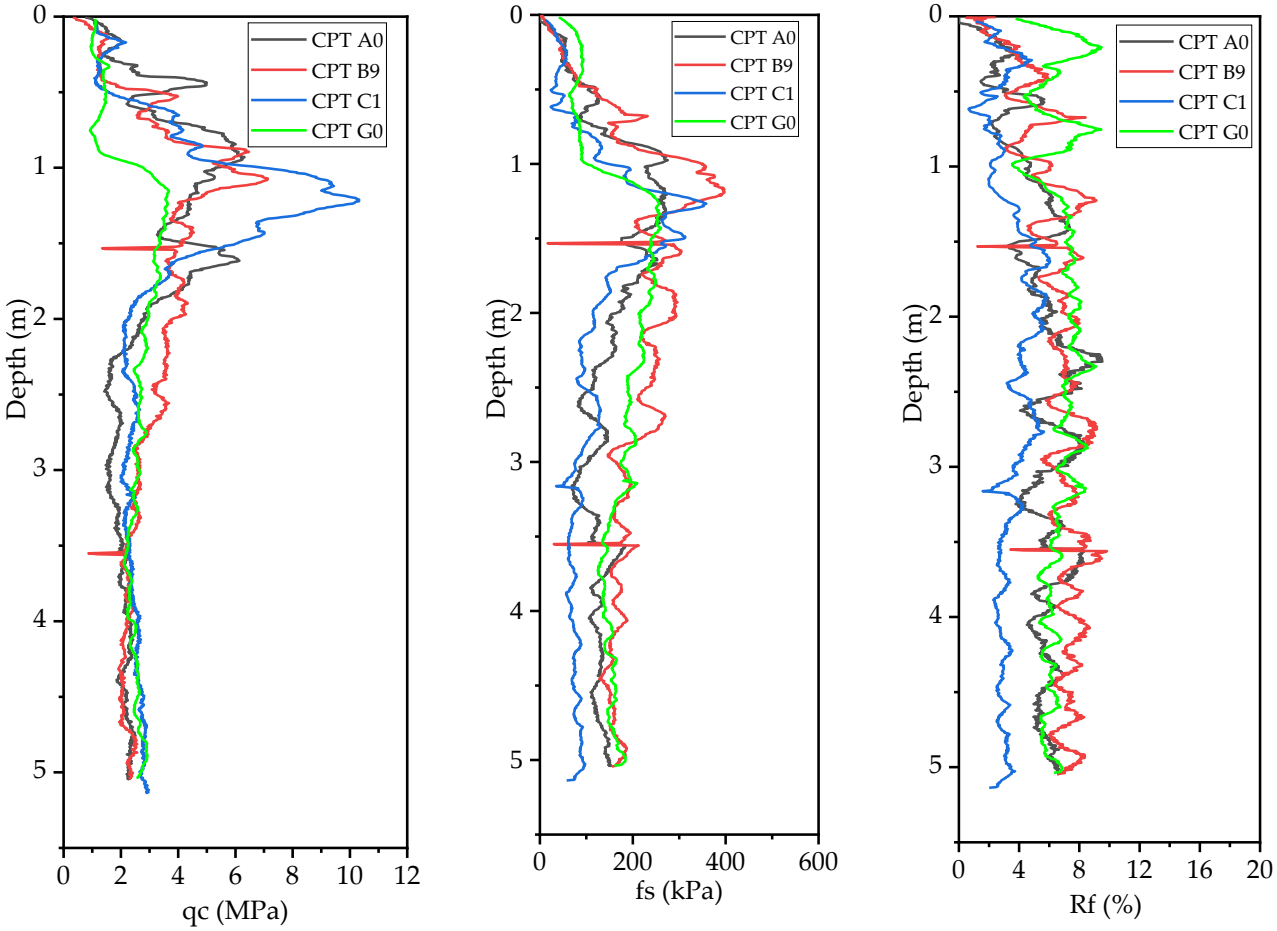


Figure 1. Sample cone tip resistance (qc), sleeve friction (fs), and friction ratio (Rf) vs. depth employed in this study.

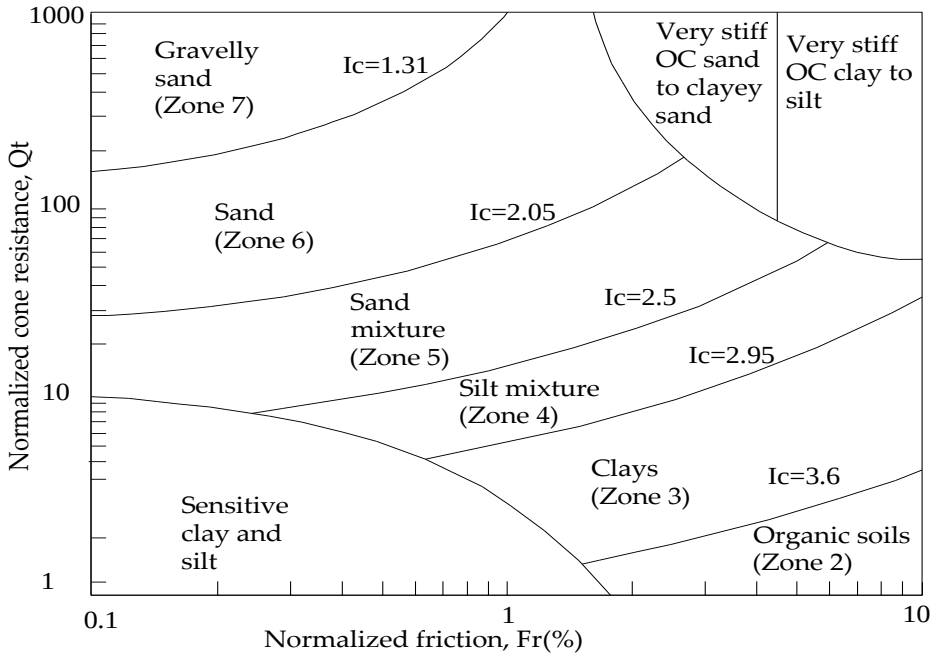


Figure 2. Robertson soil classification chart based on soil behavioral type index, Ic (adapted from [48]).

Table 1. Soil behavior type classification based on Ic boundaries (adapted from [8,48]).

SBT Classification	Ic Boundaries	Soil Type ID/ML Labels
Organic soils: peats	Ic > 3.6	2
Clays: clay to silty clay	2.95 < Ic < 3.6	3
Silt mixtures: clayey silt and silty clay	2.6 < Ic < 2.95	4
Sand mixtures: silty sand to sandy silt	2.05 < Ic < 2.6	5
Sands: clean sand to silty sand	1.31 < Ic < 2.05	6
Gravelly sand to dense sand	Ic < 1.31	7

Although existing empirical correlations work well with the CPT data, their applicability is limited to primarily fine-grained soils. Additionally, CPT and core drilling techniques work together to provide more detailed information about subsurface soil properties [12].

3. Datasets

For our study, we used publicly available CPT datasets contributed by [47], which were accessible in the International Society for Soil Mechanics and Geotechnical Engineering (ISSMGE) database. The CPTs were collected from an area measuring 50 by 50 m. Each CPT was performed to a depth of 5 m below the ground surface, and the measurement spacing of qc and fs was 5 mm. Further information about the specifics of the CPTs can be found in [47,49,50].

We preprocessed the datasets using MS Excel (see Figure 3) to categorize the soil behavior types based on Robertson's classification [48]. In order to reduce bias and ensure that the training and testing datasets are representative of the overall dataset, we shuffled the dataset and divided it into training and testing datasets using (80, 20) ratio.

The steps followed to preprocess data are as follows:

(1) Combine the individual CPT soundings into the appropriate columns (e.g. depth, qc, and fs) using Power Query in MS Excel and remove the missing values. (2) Calculate the inter quartile range (IQR) values for the qc and fs columns using Excel's built-in functions such as QUARTILE. (3) Determine the upper threshold values for outlier detection by multiplying the IQR by three and adding the third quartile. (4) Identify the outlier values in the qc and fs columns using conditional formatting. (5) Remove the outlier from the dataset and replace the values with the threshold value. (6) Estimate the R_f, total vertical stress (σ_v), effective vertical stresses (σ'_v), and Ic.

Table 2 presents a statistical summary of the datasets organized into 222,100 rows and 7 columns. The frequency distribution of each soil type in the dataset is shown in Figure 4. The distribution analysis demonstrated that soil type 5 has the highest frequency and represents over 50% of the total dataset. Soil type 4 has the second highest frequency and represents over 30% of the dataset. Soil types 2, 3, 6, and 7 (minority class) have much lower frequencies and represent less than 20% of the dataset combined, indicating an imbalanced dataset. Balancing this highly imbalanced dataset using oversampling or under sampling techniques may be possible, but it can also affect the natural variability of the soil, potentially leading to biased predictions and incorrect soil classification. To avoid this, we opted to train the ML models on the imbalanced datasets and evaluate their performances using appropriate evaluation metrics such as sensitivity, precision, and F1_score, instead of artificially generating or discarding soil samples that could impact the true variability of the soil.

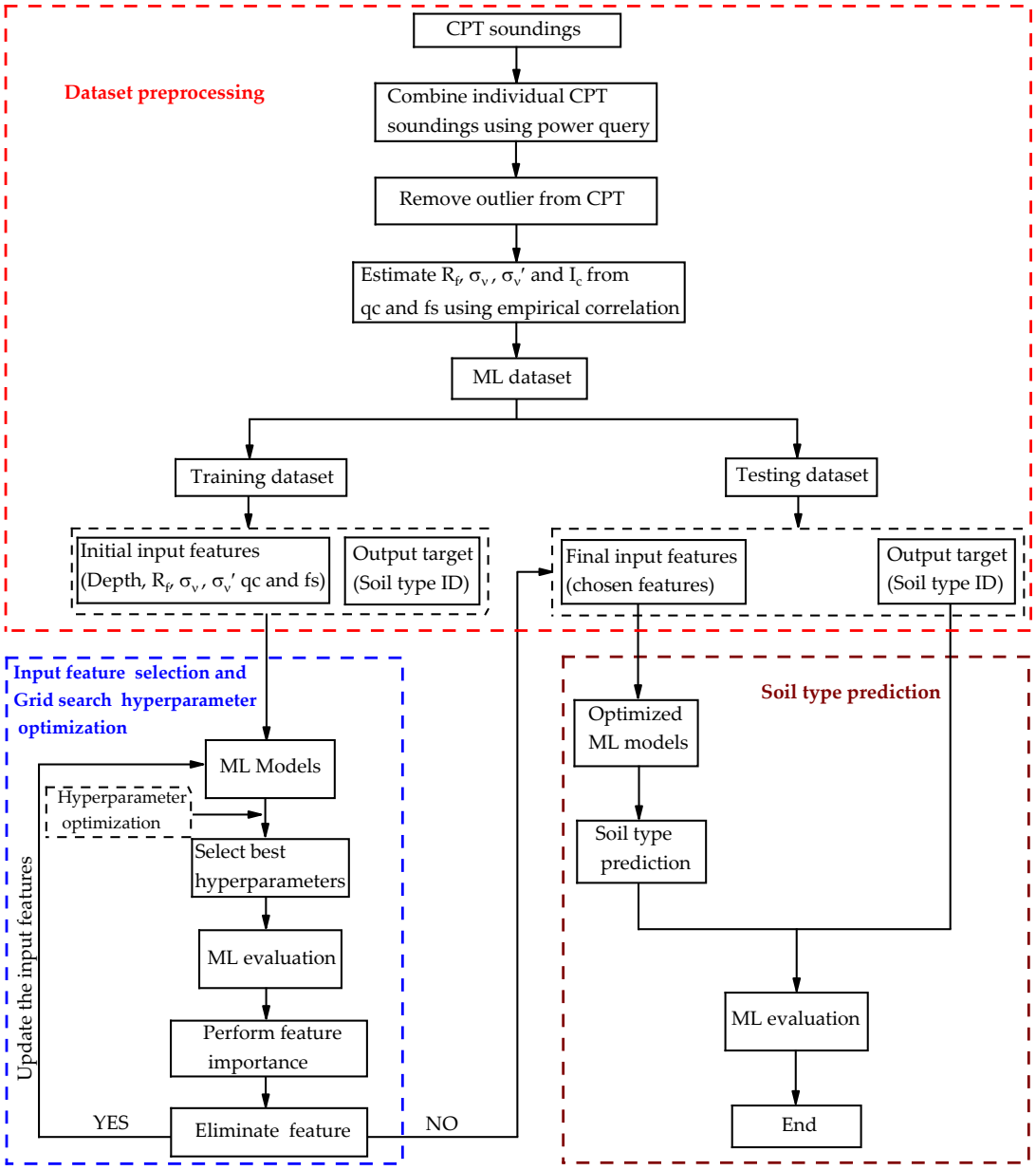


Figure 3. Flow diagram illustrating dataset preprocessing and machine learning model architecture.

Table 2. Summary statistics of dataset.

	Depth (m)	fs (kPa)	qc (kPa)	Rf (%)	σ_v (kPa)	σ'_v (kPa)	Soil Type ID
Mean	2.57	158.91	2535	9.53	49.75	24.00	4.68
Median	2.57	144.70	2360	6.14	49.87	23.89	5
Standard Deviation	1.43	84.31	1181	16.59	27.41	13.22	0.82
Kurtosis	−1.16	1.72	1.79	87.94	−1.11	−1.06	1.23
Skewness	0.00	1.12	0.99	5.86	−0.01	0.01	−0.53
Minimum	0.01	0.30	10.0	0.01	0.07	0.02	2
Maximum	5.63	438.90	6830	899	109.94	54.92	7

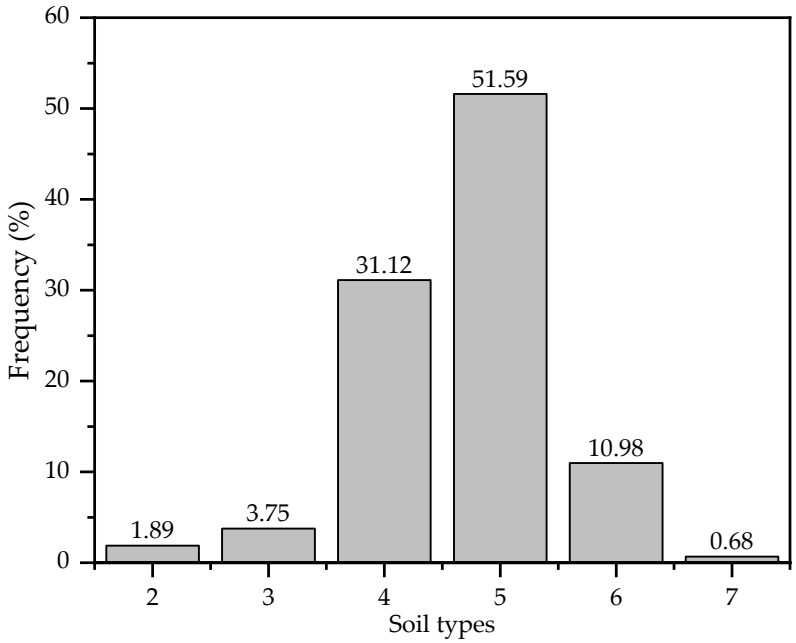


Figure 4. Frequency distribution of soil types considered for ML models.

The input features, which include depth, qc, and fs, are raw data directly obtained from the test. In contrast, the friction ratio R_f, total vertical stress (σ_v), and effective vertical stresses (σ'_v) are results from empirical correlations (Equations (2), (3) and (5), respectively).

$$\sigma_v = \gamma \times h \tag{3}$$

where σ_v is total vertical stress, γ is unit weight of soil and h is depth of soil.

The unit weight of the soil is estimated using the following expression [51]:

$$\gamma = \gamma_w \left[0.27 log R_f + 0.36 log(q_c / p_a) + 1.236\right] \tag{4}$$

where γ is unit weight of soil, γ_w is unit weight of water in the same unit as γ, q_c is cone tip resistance, and p_a is atmospheric pressure in the same unit as q_c.

$$\sigma'_v = \sigma_v - \gamma_w h \tag{5}$$

where σ'_v is effective vertical stress, σ_v is total vertical stress, γ_w is unit weight of water in the same unit as σ_v, and h is depth of soil.

4. Machine Learning Models

ML is a subfield of artificial intelligence (AI) that aims to develop algorithms and statistical models to help computer systems improve their performance on specific tasks by learning from the data [52]. The types of learning include supervised, unsupervised, and reinforcement [53]. While the supervised and reinforcement learning algorithms can involve human supervision, the unsupervised learning algorithms do not rely on labeled data or human guidance.

Our study utilized the supervised ML algorithms to classify soils using the CPT datasets. We trained four different ML algorithms, ANN, RF, SVM, and DT, using training CPT datasets and tested their performance on test datasets via R programming language [54]. In the following section, we discuss each of the ML algorithms to gain insight into their strengths and limitations.

4.1. Artificial Neural Network Model

ANNs are ML models that draw inspiration from the human brain's structure and functions [35]. They comprise interconnected neurons that use weighted connections to process and transmit information. ANNs can learn data patterns and relationships by modifying the connection strength based on the input and output. ANN models typically contain three layers, including input, hidden, and output layers. Figure 5 presents an example of an ANN model that includes an input layer with 6 neurons, 2 hidden layers with 16 and 8 neurons, and an output layer. Deep learning is commonly used to describe neural networks with many hidden layers.

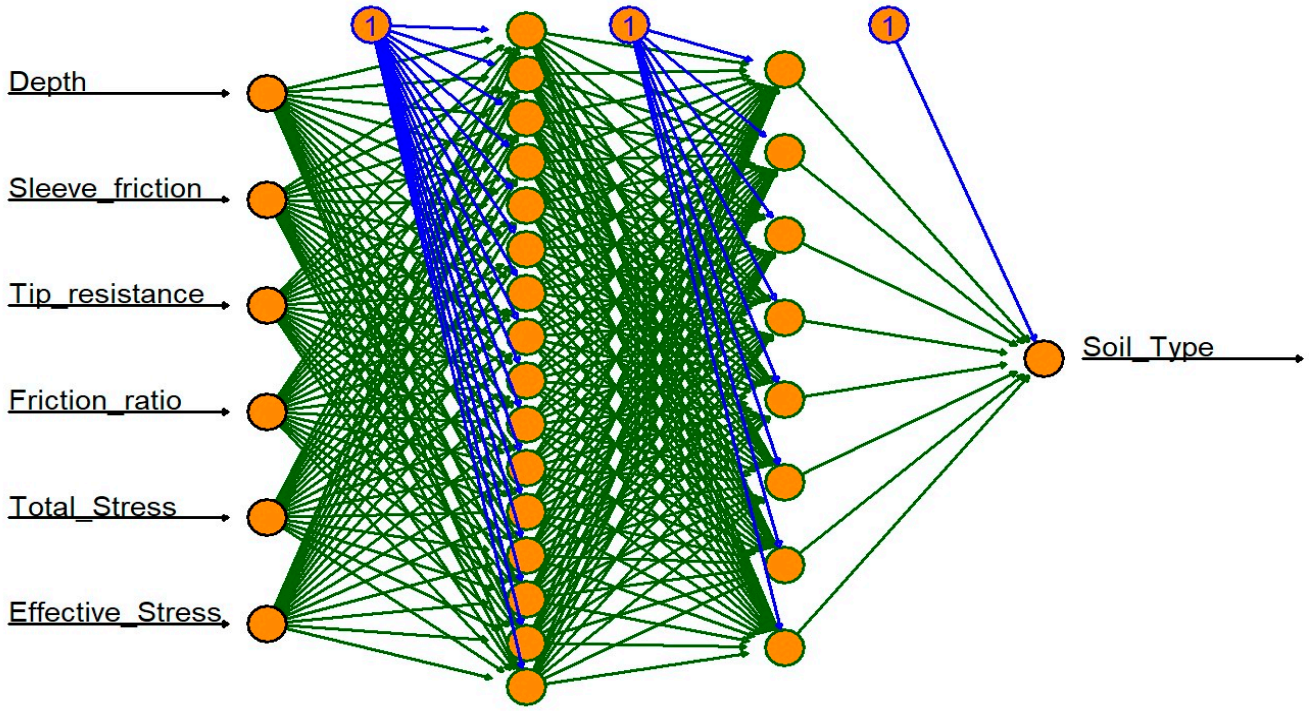


Figure 5. Example of visualized neural network plot with 2 hidden layers, 16 and 6 neurons, and 6 features.

In the ANN models, weights (the connection strength between neurons) and activations (output of a neuron in the network) are fundamental elements that enable the network to learn patterns and relationships in the data. The weights in an ANN are adjusted during the learning process to optimize the model's performance. At the same time, activation applies a mathematical operation to the input and transmits an output to the other neurons in the network.

Choosing the proper activation function is essential when dealing with an ANN model. There are several types of activation functions, namely, Sigmoid function (commonly used for binary classifications), ReLU (rectified linear unit) function, Tanh (hyperbolic tangent) function, and Softmax function (commonly used in the output layer).

Our study considers an ANN model with 2 hidden layers containing 128 and 32 neurons and an output layer. We implemented our models using the Keras package [55], which provides an easy-to-use interface for building and training neural networks. A multi-layer perceptron (MLP) model provided the soil classification with the ReLU activation function in both hidden layers, and the Softmax activation function in the output layer. The Keras library in R aided the model development, which was compiled using the categorical cross-entropy loss function, the Adam optimizer, and accuracy as the evaluation metric. The model was trained on the training data for 200 epochs, using a batch size of 32 and a validation split of 0.2. The model would learn from the data and adjust its weights and

biases to minimize the loss, which measures the difference between the predicted and actual values.

The categorical cross-entropy loss function is used to measure the difference between the predicted and actual values in a classification task. To minimize this loss, the Adam optimizer adjusts the weights and biases of the model during training. Accuracy, on the other hand, is a metric that evaluates how well the model generalizes to new, unseen data by measuring the percentage of correct predictions. The performance of the model was improved through the Bayesian optimization fine-tune of its hyperparameters including dense units 1, dense units 2, dropout 1, dropout 2, and batch size.

4.2. Random Forest Model

Random forest is a widely used ensemble learning algorithm for both classification and regression tasks. The algorithm employs multiple decision trees to improve the model's accuracy and robustness. Unlike individual decision trees, random forest is less prone to overfitting as it combines multiple trees with varying biases and variances. Additionally, it can efficiently handle high-dimensional data with many features by randomly selecting a subset of features for each tree. As a result, the algorithm is capable of handling large and complex datasets [36,37,56].

In our research, we utilized the random forest algorithm to train a model using the random forest package [57] in the R programming language. We fine-tuned the model's hyperparameters, including the number of variables randomly sampled at each split of a decision tree (mtry), the minimum number of internal node size (min.node.size), and the number of decision trees (ntree), using a model-based Bayesian optimization technique. The performance of the model was evaluated using cross-validation, and we selected the optimal values of the hyperparameters based on its best performance.

4.3. Decision Tree Model

Decision tree (DT) is a widely used machine learning algorithm that can be applied to both classification and regression problems. It is a non-parametric algorithm that can handle large and complex datasets without imposing a rigid parametric structure, making it a versatile tool for various applications [57]. The DT algorithm builds a tree-like model where the internal nodes of the tree represent decisions based on input features, while each leaf node represents class labels or target values. DT models are particularly suitable for multi-class classification problems due to their ability to capture non-linear relationships between input features and target variables [58,59].

For our soil classification problem, we utilized the rpart package [60] in the R programming language to implement a decision tree model. We fine-tuned the model's hyperparameters, including the complexity parameter (cp), the maximum depth of trees, the minimum split, and the maximum number of competitor splits, using Bayesian optimization. We used cross-validation to prevent overfitting and improve the model's ability to generalize to new data.

4.4. Support Vector Machine Model

SVM is a well-known supervised ML algorithm frequently utilized for multi-class classification and regression problems [38,39]. The SVM algorithm operates by locating the optimal hyperplane that segregates the input data points into distinct classes. The hyperplane locates itself by maximizing the margin, which is the gap between the hyperplane and the nearest data points of each class. For our study, we employed the e1071 R package [61], which offers an SVM implementation in R. This allowed us to train a model using the training data and assess its effectiveness on the test data. To ensure a well-tuned and generalized model, we used cross-validation to optimize the hyper-parameters (cost and gamma).

5. Results and Discussion

In the following subsections, the results of the ML models are presented and discussed using confusion matrix and various performance metrics such as overall accuracy, sensitivity (ability to detect positive instances), specificity (ability to detect negative instances), negative predicted value (NPV), positive predicted value (PPV), and balanced accuracy (the average of sensitivity and specificity). Due to the imbalanced dataset used for the training and testing purposes, additional informative performance metrics such as precision, recall, and F1_score are utilized to assess the efficacy of the ML models.

$$Overall\ Accuracy = (TP + TN)\ /\ (TP + TN + FP + FN) \tag{6}$$

where TP = True Positive (number of samples correctly predicted as positive), TN = True Negative (number of samples correctly predicted as negative), FP = False Positive (number of samples incorrectly predicted as positive), and FN = False Negative (number of samples incorrectly predicted as negative).

$$Precision = TP\ /\ (TP + FP) \tag{7}$$

$$Sensitivity = TP\ /\ (TP + FN) \tag{8}$$

$$F1_score = \frac{2 \times precision \times Sensitivity}{precision + Sensitivity} \tag{9}$$

5.1. Artificial Neural Network Model Results

The results of the ML model implemented utilizing ANN to classify different soil types are presented here. Figure 6 displays the accuracy and loss of the ANN model for 200 epochs on both the training and validation data. At the beginning of the training, the model has a low accuracy of 0.79 and a high loss of 0.63 values, indicating that it cannot make good predictions. However, as training progresses, the accuracy improves, and the loss decreases, indicating that it gradually improves its ability to make more accurate predictions. When the validation accuracy and loss metrics improve, it suggests that the model is generalizing well to new data, which is a desirable outcome.

Table 3 displays the confusion matrix of the ANN model, which provides insight into the model's performance on the test data. The rows correspond to the predicted values, while the columns correspond to the actual values. The diagonal elements of the confusion matrix represent the number of instances that the model correctly classified, while the off-diagonal elements correspond to the misclassifications made by the model.

Table 3. Artificial neural network confusion matrix.

Prediction	Actual					
	2	**3**	**4**	**5**	**6**	**7**
2	808	8	0	0	0	0
3	31	1594	23	1	0	0
4	0	63	13,708	69	0	0
5	0	0	91	22,781	55	0
6	0	0	0	62	4801	99
7	0	0	0	1	20	202

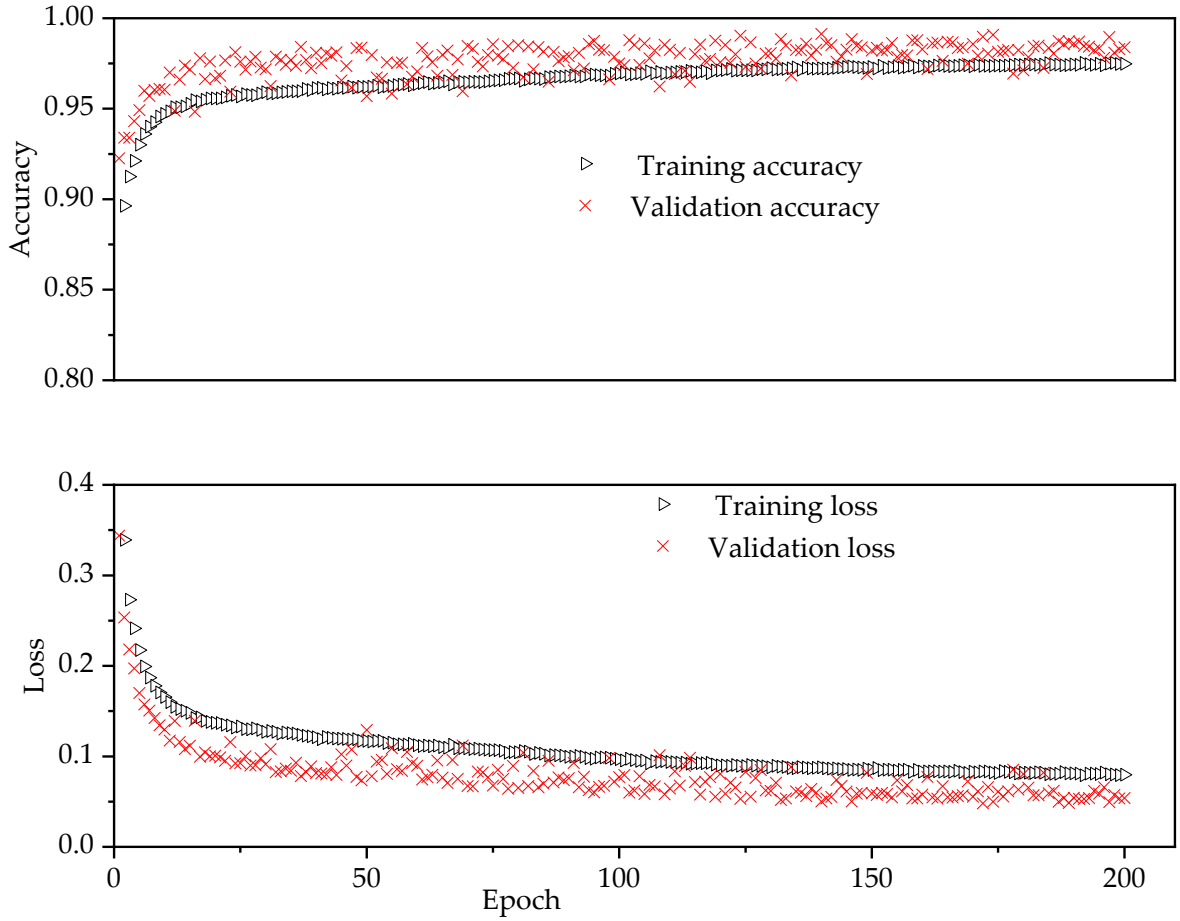


Figure 6. Change in loss and accuracy over epochs for a neural network model.

Statistics by class (Table 4) show that the model has a high sensitivity for all soil types except type 7, with a low sensitivity value of 0.67. Additionally, the model has a high specificity for all classes, with values ranging from 0.99 to 1.0.

Table 4. Class distribution summary for ANN model.

Performance Metrics	2	3	4	5	6	7
Sensitivity	0.96	0.96	0.99	0.99	0.98	0.67
Specificity	1.00	1.00	1.00	0.99	1.00	1.00
PPV	0.99	0.97	0.99	0.99	0.97	0.91
NPV	1.00	1.00	1.00	0.99	1.00	1.00
Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Detection Rate	0.02	0.04	0.31	0.51	0.11	0.00
Detection Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Balanced Accuracy	0.98	0.98	0.99	0.99	0.99	0.84

The positive predicted value (PPV) and negative predicted value (NPV) are important performance metrics in evaluating the effectiveness of a classifier. A high PPV indicates that it is likely correct when the model predicts a sample to belong to a particular class. On the other hand, a high NPV indicates that when the model predicts a sample to not belong to a particular class, it is likely to be correct. The ANN model results show a high PPV for all soil types, with values ranging from 0.91 to 0.99. Similarly, the NPV is high for all soil types, with values ranging from 0.99 to 1.

In summary, the ANN model has an overall accuracy of 98.82%, showing that the model performs well in classification tasks. However, the model struggles to predict class 7 (minority class), given a low sensitivity value.

5.2. Random Forest Model Results

Table 5 displays a confusion matrix that compares the soil types predicted by the RF model with the actual soil types. The confusion matrix shows that the model made some correct and incorrect predictions for each class. For example, the model correctly predicted 835 samples as class 2. The model has high diagonal values, signifying a high number of correct predictions, and low off-diagonal values, implying a low number of misclassifications. The model's overall accuracy is very high, indicating the model's performance. It achieved a 99.23% accuracy, indicating that it effectively predicts soil types.

Table 5. Random forest confusion matrix.

Prediction	Actual					
	2	**3**	**4**	**5**	**6**	**7**
2	835	2	0	0	0	0
3	4	1655	8	0	0	0
4	0	8	13,719	88	0	0
5	0	0	95	22,763	59	0
6	0	0	0	63	4806	5
7	0	0	0	0	11	296

The statistics by class (Table 6) show that the RF model has a high sensitivity and specificity values for all soil types. Overall, it performed well in the classification task, achieving high scores for multiple performance metrics such as PPV, NPV, and balanced accuracy.

Table 6. Class distribution summary for RF model.

Performance Metrics	2	3	4	5	6	7
Sensitivity	1.00	0.99	0.99	0.99	0.99	0.98
Specificity	1.00	1.00	1.00	0.99	1.00	1.00
PPV	1.00	0.99	0.99	0.99	0.99	0.96
NPV	1.00	1.00	1.00	0.99	1.00	1.00
Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Detection Rate	0.02	0.04	0.31	0.51	0.11	0.01
Detection Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Balanced Accuracy	1.00	1.00	0.99	0.99	0.99	0.99

5.3. Decision Tree Model Results

Table 7 presents the confusion matrix for the DT model utilized for the soil classification task. The table evaluates the performance of a predictive model in classifying different soil types based on input features. The number of observations that were accurately predicted by the model (diagonal entries) and the number of misclassifications (off-diagonal entries) appear in the table. Based on the model's confusion matrix, the model performed well in the classification task, as the number of correctly predicted values are significantly higher than the number of misclassifications. The overall accuracy of the model in predicting soil types on the test dataset was 95.67%.

Table 7. Confusion matrix for DT model.

Prediction	Actual					
	2	**3**	**4**	**5**	**6**	**7**
2	819	40	0	0	0	0
3	20	1554	45	0	0	0
4	0	71	13,240	591	0	0
5	0	0	537	22,049	269	0
6	0	0	0	274	4548	19
7	0	0	0	0	59	282

The statistics by class (Table 8) show that the DT model has a high sensitivity and specificity for all soil types. Moreover, the model exhibits a high balanced accuracy with values ranging from 0.96 to 0.99. Overall, the model performed well in the classification task, with high scores for multiple performance metrics across each soil type.

Table 8. Class distribution summary for DT model.

Performance Metrics	2	3	4	5	6	7
Sensitivity	0.98	0.93	0.96	0.96	0.93	0.94
Specificity	1.00	1.00	0.98	0.96	0.99	1.00
PPV	0.95	0.96	0.95	0.96	0.94	0.83
NPV	1.00	1.00	0.98	0.96	0.99	1.00
Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Detection Rate	0.02	0.03	0.30	0.50	0.10	0.01
Detection Prevalence	0.02	0.04	0.31	0.51	0.11	0.01
Balanced Accuracy	0.99	0.97	0.97	0.96	0.96	0.97

5.4. Support Vector Machine Model Results

The results of the ML model implemented utilizing the SVM to classify different soil types are presented here. The confusion matrix computed with the SVM model to evaluate its effectiveness is presented in Table 9. The confusion matrix shows that the model predicted almost all instances correctly, with a few misclassifications in each soil type. The overall accuracy of the model is very high (almost 100%), indicating that it is a high-performing model.

Table 9. Support vector machine confusion matrix.

Prediction	Actual					
	2	3	4	5	6	7
2	837	0	0	0	0	0
3	2	1662	0	0	0	0
4	0	3	13,812	6	0	0
5	0	0	10	22,893	7	0
6	0	0	0	15	4848	9
7	0	0	0	0	21	292

Table 10 shows the class distribution summary for the SVM model. The model's sensitivity is high for each soil type, indicating that the model is good at correctly identifying the positive cases for each soil type. The model's specificity is also high for all soil types, indicating that the model is good at correctly identifying the negative cases for all soil types. Moreover, the model's balanced accuracy (the average of sensitivity and specificity) is remarkably high (almost 1) for all soil types. This shows that the model can accurately identify both positive and negative cases, making it a reliable classifier for the soil classification task.

Table 10. Class distribution summary for SVM model.

Performance Metrics	2	3	4	5	6	7
Sensitivity	1	1	1	1	1	0.97
Specificity	1	1	1	1	1	1
PPV	1	1	1	1	1	0.93
NPV	1	1	1	1	1	1
Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Detection Rate	0.02	0.04	0.31	0.52	0.11	0.01
Detection Prevalence	0.02	0.04	0.31	0.52	0.11	0.01
Balanced Accuracy	1	1	1	1	1	0.98

The model exhibits high PPVs for all soil types, indicating its strong ability to predict the samples of specific soil types accurately. Similarly, the model shows high NPVs for

all soil types, indicating its reliability in predicting the samples that do not belong to a particular soil class. Overall, the model performs exceptionally well on the dataset in terms of multiple performance metrics.

5.5. Comparison of ML Models' Performance

To compare the efficiency of the ML models, different performance metrics such as overall accuracy, sensitivity, precision, and F1_score are utilized. The results of this evaluation are summarized in Tables 11 and 12. Table 11 shows that the SVM model achieved the highest overall accuracy of 99.84%.

Table 11. Performance comparisons of ML models on test datasets.

	ML Models			
	ANN	**RF**	**DT**	**SVM**
Overall Accuracy (%)	98.82	99.23	95.67	99.84

Table 12. Performance metrics of ML models for each soil type.

ML Models	Soil Type	Sensitivity	Precision	F1_Score
ANN	2	0.96	0.99	0.98
	3	0.96	0.97	0.96
	4	0.99	0.99	0.99
	5	0.99	0.99	0.99
	6	0.98	0.97	0.98
	7	0.67	0.91	0.77
RF	2	1	0.99	0.99
	3	1	0.99	0.99
	4	1	0.99	0.99
	5	1	0.99	0.99
	6	1	0.99	0.99
	7	0.98	0.96	0.97
SVM	2	1	1	1
	3	1	1	1
	4	1	1	1
	5	1	1	1
	6	0.99	1	0.99
	7	0.97	0.93	0.95
DT	2	0.98	0.95	0.96
	3	0.93	0.96	0.95
	4	0.96	0.95	0.96
	5	0.96	0.96	0.96
	6	0.93	0.94	0.94
	7	0.94	0.83	0.88

The ANN, RF, and DT models also performed well, achieving overall accuracies of 98.82%, 99.23%, and 95.67%, respectively. It is important to note that the datasets were imbalanced, and therefore, it is necessary to consider both the overall accuracy and other performance metrics for each soil type to accurately assess the ML models' performance.

Table 12 presents the performance metrics of the ML models for each soil type. The table shows the sensitivity, precision, and F1_score values of each model and soil type. These metrics indicate the models' efficiency in correctly identifying the soil type. Across all models, the sensitivity, precision, and F1_score values for each soil type are very high, indicating that the models successfully identified instances of all classes. However, the efficiency of the ANN model on minority class 7 was low compared to the other models. It scored lower sensitivity and F1_score values of 0.67 and 0.77, respectively, compared to the

SVM and RF models with almost perfect scores for all metrics. This indicates that the ANN model needs additional data to better identify minority classes.

The SVM and RF models outperformed the ANN and DT models in terms of sensitivity, precision, and F1_score values for all soil types. These two models achieved almost perfect scores for all performance metrics for all soil types, indicating their high accuracy in the classification task. Overall, the performance of the ML models in classifying soils based on the CPT dataset is consistent with previous similar research carried out on ML techniques (e.g., see [12,21]).

6. Conclusions

In this paper, various ML algorithms, namely, ANN, RF, SVM, and DT are used to classify soils based on Robertson's soil behavioral types. To optimize the performance of these models, hyperparameter tuning was performed using Bayesian optimization. Additionally, cross-validation was employed to ensure the models' generalization ability and optimal performance. The study used 232 CPTs from ISSMGE's database and randomly divided the dataset into training and testing datasets. The performance of each ML model was evaluated using performance metrics such as sensitivity, precision, F1_score, and overall accuracy.

Based on the findings of this study, the following conclusions can be made:

- The ANN model achieved an overall accuracy of 98.39%. It achieved high scores in multiple performance metrics for majority soil classes. However, the model achieved lower sensitivity and F1_score values of 0.67 and 0.77, respectively, for minority soil class 7.
- The DT model achieved an overall accuracy of 95.67%, indicating high performance in classifying soils. Additionally, the model demonstrated excellent sensitivity, precision, and F1_score across all soil types, with scores ranging from 0.88–0.98.
- The SVM model outperformed the other models with the highest overall accuracy of 99.84%. It achieved almost perfect scores for all performance metrics across all soil types.
- The RF model achieved an overall accuracy of 99.23% and demonstrated high performance across all soil types. Similar to the SVM model, the RF model also achieved almost perfect scores for all performance metrics across all soil types.
- In general, the SVM and RF models achieved a high level of overall accuracy (almost 100%) in classifying soils, even when trained with imbalanced CPT datasets. These models exhibited outstanding performance on both majority and minority soil classes, indicating their potential as valuable tools in geotechnical engineering. Integrating these ML models into software programs for rapid and accurate soil classification in real-world projects can aid in making informed decisions.
- Future research could focus on improving the performance of the ANN and DT models in classifying soils based on CPT data. This could involve exploring other approaches such as training the models using balanced datasets.

Author Contributions: Conceptualization, A.T.C.; methodology, A.T.C.; software, A.T.C.; formal analysis, A.T.C.; writing—original draft preparation, A.T.C.; writing—review and editing, A.T.C.; supervision, R.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets can be downloaded at the following link: http://140.112.12.21/issmge/tc304.htm?=6 (Accessed on 20 January 2023).

Acknowledgments: This publication was financially supported by Széchenyi István University.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Robertson, P.K. Interpretation of in-situ tests. In Proceedings of the J.K. Mitchell Lecture-Proceedings of ISC'4, Refice, Brazil, 17–21 September 2012; pp. 1–22.
2. Robertson, P.K. Soil Behaviour Type from the CPT: An Update. In Proceedings of the 2nd International Symposium on Cone Penetration Testing, Huntington Beach, CA, USA, 9–12 May 2010.
3. Robertson, P. Cone penetration test (CPT)-based soil behaviour type (SBT) classification system—An update. *Can. Geotech. J.* **2016**, *53*, 1910–1927. [CrossRef]
4. Robertson, P.K.; Campanella, R.G.; Gillespie, D.; Greig, J. Use of Piezometer Cone Data. In *Use of In Situ Tests in Geotechnical Engineering*; ASCE: Reston, VA, USA, 1986; pp. 1263–1280.
5. Laufer, I. Statistical analysis of CPT tip resistances. *Period. Polytech. Civ. Eng.* **2013**, *57*, 45–61. [CrossRef]
6. Begemann, H.K.S.P. The Friction Jacket Cone as an Aid in Determining the Soil Profile. In Proceedings of the 6th International Conference on Soil Mechanics and Foundation Engineering, Montreal, QC, Canada, 8–15 September 1965; pp. 17–20. Available online: https://cir.nii.ac.jp/crid/1573950399307239936 (accessed on 15 April 2023).
7. Douglas, B.J.; Olsen, R.S. Soil Classification Using Electric Cone Penetrometer. In Proceedings of the Symposium on Cone Penetration Testing and Experience, St. Louis, MO, USA, 26–30 October 1981; pp. 209–227.
8. Robertson, P.K. Soil classification using the cone penetration test. *Can. Geotech. J.* **1990**, *27*, 151–158. [CrossRef]
9. Rock, A.C.D. *Standard Practice for Classification of Soils for Engineering Purposes (Unified Soil Classification System) 1*; ASTM International: West Conshohocken, PA, USA, 2017.
10. Wang, H.; Wang, X.; Wellmann, F.; Liang, R.Y. A Bayesian unsupervised learning approach for identifying soil stratification using cone penetration data. *Can. Geotech. J.* **2019**, *56*, 1184–1205. [CrossRef]
11. Reale, C.; Gavin, K.; Librić, L.; Jurić-Kaćunić, D. Automatic classification of fine-grained soils using CPT measurements and Artificial Neural Networks. *Adv. Eng. Inform.* **2017**, *36*, 207–215. [CrossRef]
12. Rauter, S.; Tschuchnigg, F. CPT Data Interpretation Employing Different Machine Learning Techniques. *Geosciences* **2021**, *11*, 265. [CrossRef]
13. Tsiaousi, D.; Travasarou, T.; Drosos, V.; Ugalde, J.; Chacko, J. Machine Learning Applications for Site Characterization Based on CPT Data. In *Geotechnical Earthquake Engineering and Soil Dynamics V*; American Society of Civil Engineers: Reston, VA, USA, 2018; pp. 461–472.
14. Rogiers, B.; Mallants, D.; Batelaan, O.; Gedeon, M.; Huysmans, M.; Dassargues, A. Model-based classification of CPT data and automated lithostratigraphic mapping for high-resolution characterization of a heterogeneous sedimentary aquifer. *PLoS ONE* **2017**, *12*, e0176656. [CrossRef]
15. Kurup, P.U.; Griffin, E.P. Prediction of Soil Composition from CPT Data Using General Regression Neural Network. *J. Comput. Civ. Eng.* **2006**, *20*, 281–289. [CrossRef]
16. Bhattacharya, B.; Solomatine, D. Machine learning in soil classification. *Neural Netw.* **2006**, *19*, 186–195. [CrossRef]
17. Arel, E. Predicting the spatial distribution of soil profile in Adapazari/Turkey by artificial neural networks using CPT data. *Comput. Geosci.* **2012**, *43*, 90–100. [CrossRef]
18. Carvalho, L.; Ribeiro, D. Application of kernel k-means and kernel x-means clustering to obtain soil classes from cone penetration test data. *Soils Rocks* **2020**, *43*, 607–618. [CrossRef]
19. Kohestani, V.R.; Hassanlourad, M.; Ardakani, A. Evaluation of liquefaction potential based on CPT data using random forest. *Nat. Hazards* **2015**, *79*, 1079–1089. [CrossRef]
20. Goh, A.T.; Goh, S. Support vector machines: Their use in geotechnical engineering as illustrated using seismic liquefaction data. *Comput. Geotech.* **2007**, *34*, 410–421. [CrossRef]
21. Carvalho, L.O.; Ribeiro, D.B. A multiple model machine learning approach for soil classification from cone penetration test data. *Soils Rocks* **2021**, *44*, e2021072121. [CrossRef]
22. Livingston, G.; Piantedosi, M.; Kurup, P.; Sitharam, T.G. Using Decision-Tree Learning to Assess Liquefaction Potential from CPT and V_s. In *Geotechnical Earthquake Engineering and Soil Dynamics IV*; ASCE: Reston, VA, USA, 2008; pp. 1–10. [CrossRef]
23. Nhat-Duc, H.; Van-Duc, T. Comparison of histogram-based gradient boosting classification machine, random Forest, and deep convolutional neural network for pavement raveling severity classification. *Autom. Constr.* **2023**, *148*, 104767. [CrossRef]
24. Aydın, Y.; Işıkdağ, Ü.; Bekdaş, G.; Nigdeli, S.M.; Geem, Z.W. Use of Machine Learning Techniques in Soil Classification. *Sustainability* **2023**, *15*, 2374. [CrossRef]
25. Kang, T.-H.; Choi, S.-W.; Lee, C.; Chang, S.-H. Soil Classification by Machine Learning Using a Tunnel Boring Machine's Operating Parameters. *Appl. Sci.* **2022**, *12*, 11480. [CrossRef]
26. Hikouei, I.S.; Kim, S.S.; Mishra, D.R. Machine-Learning Classification of Soil Bulk Density in Salt Marsh Environments. *Sensors* **2021**, *21*, 4408. [CrossRef]
27. Eyo, E.; Abbey, S. Multiclass stand-alone and ensemble machine learning algorithms utilised to classify soils based on their physico-chemical characteristics. *J. Rock Mech. Geotech. Eng.* **2021**, *14*, 603–615. [CrossRef]
28. Huang, H.-W.; Li, Q.-T.; Zhang, D.-M. Deep learning based image recognition for crack and leakage defects of metro shield tunnel. *Tunn. Undergr. Space Technol.* **2018**, *77*, 166–176. [CrossRef]
29. Cheng, G.; Guo, W. Rock images classification by using deep convolution neural network. *J. Phys. Conf. Ser.* **2017**, *887*, 12089. [CrossRef]

30. Ran, X.; Xue, L.; Zhang, Y.; Liu, Z.; Sang, X.; He, J. Rock Classification from Field Image Patches Analyzed Using a Deep Convolutional Neural Network. *Mathematics* **2019**, *7*, 755. [CrossRef]
31. Xiao, L.; Zhang, Y.; Peng, G. Landslide Susceptibility Assessment Using Integrated Deep Learning Algorithm along the China-Nepal Highway. *Sensors* **2018**, *18*, 4436. [CrossRef] [PubMed]
32. Bui, D.T.; Tsangaratos, P.; Nguyen, V.-T.; Van Liem, N.; Trinh, P.T. Comparing the prediction performance of a Deep Learning Neural Network model with conventional machine learning models in landslide susceptibility assessment. *Catena* **2020**, *188*, 104426. [CrossRef]
33. Chakraborty, A.; Goswami, D. Prediction of slope stability using multiple linear regression (MLR) and artificial neural network (ANN). *Arab. J. Geosci.* **2017**, *10*, 385. [CrossRef]
34. Qi, C.; Tang, X. Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. *Comput. Ind. Eng.* **2018**, *118*, 112–122. [CrossRef]
35. Stock, D.J. *An Introduction to Neural Networks*; CRC Press: Boca Raton, FL, USA, 1992; Volume 23. [CrossRef]
36. Liu, Y.; Wang, Y.; Zhang, J. New machine learning algorithm: Random forest. In Proceedings of the Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, 14–16 September 2012; pp. 246–252.
37. Friedman, J.H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [CrossRef]
38. Petropoulos, G.P.; Kalaitzidis, C.; Vadrevu, K.P. Support vector machines and object-based classification for obtaining land-use/cover cartography from Hyperion hyperspectral imagery. *Comput. Geosci.* **2012**, *41*, 99–107. [CrossRef]
39. Huo, L.-Z.; Tang, P. Spectral and spatial classification of hyperspectral data using SVMs and Gabor textures. *Int. Geosci. Remote Sens. Symp.* **2011**, *46*, 1708–1711. [CrossRef]
40. Meinshausen, N.; Ridgeway, G. Quantile regression forests. *J. Mach. Learn. Res.* **2006**, *7*, 983–999.
41. Sameen, M.I.; Pradhan, B.; Lee, S. Self-Learning Random Forests Model for Mapping Groundwater Yield in Data-Scarce Areas. *Nat. Resour. Res.* **2019**, *28*, 757–775. [CrossRef]
42. Zhang, Y.-M.; Wang, H.; Mao, J.-X.; Xu, Z.-D.; Zhang, Y.-F. Probabilistic Framework with Bayesian Optimization for Predicting Typhoon-Induced Dynamic Responses of a Long-Span Bridge. *J. Struct. Eng.* **2021**, *147*, 04020297. [CrossRef]
43. Stoppiglia, H.; Rémi Dubois, E.; Oussar Yacineoussar, Y. Ranking a Random Feature for Variable and Feature Selection Hervé Stoppiglia Gérard Dreyfus. *J. Mach. Learn. Res.* **2003**, *3*, 1399–1414.
44. Dai, B.; Gu, C.; Zhao, E.; Qin, X. Statistical model optimized random forest regression model for concrete dam deformation monitoring. *Struct. Control Health Monit.* **2018**, *25*, e2170. [CrossRef]
45. Kwak, S.K.; Kim, J.H. Statistical data preparation: Management of missing values and outliers. *Korean J. Anesthesiol.* **2017**, *70*, 407–411. [CrossRef]
46. Barnett, V.; Lewis, T. *Outliers in Statistical Data*; Wiley: New York, NY, USA, 1994; Volume 3.
47. Jaksa, M.B. The Influence of Spatial Variability on the Geotechnical Design Properties of a Stiff, Overconsolidated Clay. Available online: https://digital.library.adelaide.edu.au/dspace/handle/2440/37800 (accessed on 25 January 2023).
48. Robertson, P.K.; Wride, C. Evaluating cyclic liquefaction potential using the cone penetration test. *Can. Geotech. J.* **1998**, *35*, 442–459. [CrossRef]
49. Liu, J.; Liu, J.; Li, Z.; Hou, X.; Dai, G. Estimating CPT Parameters at Unsampled Locations Based on Kriging Interpolation Method. *Appl. Sci.* **2021**, *11*, 11264. [CrossRef]
50. Chala, A.; Ray, R. Generation and Evaluation of CPT Data Using Kriging Interpolation Technique. *Period. Polytech. Civ. Eng.* **2023**, *67*, 545–551. [CrossRef]
51. Robertson, P.K.; Cabal, K.L. Estimating soil unit weight from CPT. In Proceedings of the 2nd International Symposium on Cone Penetration Testing, Huntington Beach, CA, USA, 9–12 May 2010; p. 8. Available online: https://www.mendeley.com/catalogue/4c2ffa47-74a9-3ea8-b17c-5a8843514cd6/?utm_source=desktop&utm_medium=1.19.8&utm_campaign=open_catalog&userDocumentId=%7B2cb2fdcc-bb36-49ee-8cf3-a99ebf60b478%7D (accessed on 14 January 2023).
52. Géron, A.; Courville, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2011; Volume 44.
53. Vemuri, V.K. The Hundred-Page Machine Learning Book. *J. Inf. Technol. Case Appl. Res.* **2020**, *22*, 136–138. [CrossRef]
54. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2022.
55. Venables, W.N.; Ripley, B.D. *Modern Applied Statistics with S*; Springer: New York, NY, USA, 2002.
56. Ren, Q.; Cheng, H.; Han, H. Research on machine learning framework based on random forest algorithm. *AIP Conf. Proc.* **2017**, *1820*, 80020. [CrossRef]
57. Liaw, A.; Wiener, M. Classification and regression by randomForest. *R News* **2002**, *2*, 18–22.
58. Quinlan, J.R. Induction of decision trees. *Mach. Learn.* **1986**, *1*, 81–106. [CrossRef]
59. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: New York, NY, USA, 2009; Volume 2.

60. Therneau, T.; Atkinson, B.; Ripley, B. *rpart: Recursive Partitioning and Regression Trees*; R Package Version; R Foundation for Statistical Computing: Vienna, Austria, 2015; Volume 4, pp. 1–9.
61. Meyer, D.; Dimitriadou, E.; Hornik, K.; Weingessel, A.; Leisch, F. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien_R Package Version 1.7-13. 2023. Available online: https://cran.r-project.org/package=e1071 (accessed on 23 March 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.