

Article

Not peer-reviewed version

Comparison of Liquefaction Potential Prediction Results Using Machine Learning Methods Based on CPT Data

[Sercan Tekeođlu](#) * and [Ender Bařarı](#)

Posted Date: 14 April 2026

doi: 10.20944/preprints202604.0966.v1

Keywords: liquefaction; machine learning; random forest; artificial neural network; CPT



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

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.

Article

Comparison of Liquefaction Potential Prediction Results Using Machine Learning Methods Based on CPT Data

Sercan Tekeoğlu * and Ender Başarı

Manisa Celal Bayar University, Faculty of Engineering and Natural Sciences, Department of Civil Engineering, Manisa 45140, Türkiye

* Correspondence: 211291005@ogr.cbu.edu.tr

Abstract

Soil liquefaction is a significant geotechnical hazard that can lead to severe structural damage during seismic events. Traditional liquefaction assessment methods, such as those based on the Standard Penetration Test (SPT) and Cone Penetration Test (CPT), rely on empirical correlations but often struggle to capture the complex, nonlinear interactions between soil properties and seismic parameters. Recent advancements in machine learning (ML) offer data-driven approaches that can improve liquefaction prediction accuracy. This study evaluates and compares the performance of Random Forest (RF) and Artificial Neural Networks (ANNs) for liquefaction potential prediction using a dataset containing 480 field observations derived from CPT-based studies. The dataset was preprocessed using min-max normalization, and models were trained and optimized through hyperparameter tuning. Model performance was assessed using accuracy, precision, recall, F-measure, Cohen's kappa, and AUC-ROC analysis. The results show that RF achieved the highest accuracy (89%), outperforming both ANN (86%) and the traditional CPT-based liquefaction assessment method (87%). Additionally, ROC-AUC values of 0.932 for RF and 0.872 for ANN indicate the superior classification capability of machine learning models. Feature importance analysis in RF revealed that cone tip resistance (q_c), cyclic stress ratio (CSR), and peak ground acceleration (a_{max}) are the most influential factors in liquefaction prediction. These findings demonstrate that machine learning techniques, particularly RF, provide more reliable liquefaction predictions compared to conventional empirical methods. The study highlights the potential of ML models in improving seismic risk assessments and guiding engineering decision-making processes.

Keywords: liquefaction; machine learning; random forest; artificial neural network; CPT

1. Introduction

Soil liquefaction is an earthquake phenomenon in which saturated granular soils lose their strength and stiffness under seismic loading and behave more like a viscous fluid than a solid medium. This transformation, recognized by the 1964 Niigata and Great Alaska events, has been observed and reported repeatedly in many earthquakes for about half a century. After the awareness of liquefaction, a significant portion of earthquake damage has been attributed to soil liquefaction, and many studies and procedures have been included in the literature to estimate liquefaction potential and prevent liquefaction hazards [1–4].

At the core of liquefaction lies the mechanism of cyclic shearing under seismic forces. As strong ground shaking occurs, cyclic loading causes a progressive buildup of pore water pressure in the saturated soils. When pore pressure nearly equals the initial confining stress, the effective stress—and hence the soil's shear strength—drops dramatically. While loose sands often experience complete loss of strength and significant deformations, medium-density sands, silty sands, and sandy silts may not fail entirely but still undergo considerable softening and cyclic strain [5–7].

Traditional liquefaction potential assessments are semi-empirical methods based on both field tests and past liquefaction observations due to the difficulty of taking undisturbed samples from frequently liquefaction-occurred loose sandy soils. These methods rely on key parameters such as soil properties (e.g., relative density and gradation), environmental conditions (e.g., groundwater depth), and earthquake (e.g., peak ground acceleration) to classify a site as susceptible or resistant to liquefaction. One of the seminal approaches in this area is based on the standard penetration test (SPT) or cone penetration test (CPT) combined with the computation of the cyclic stress ratio (CSR), which utilizes parameters like peak ground acceleration to quantify the potential for liquefaction [6,8,9].

Despite their widespread use and proven track record, traditional methods face inherent challenges. In particular, the influence of various in situ state parameters—such as fines content, plasticity index, cementation, and overconsolidation—on q_c and V_s is complex and not fully captured by deterministic approaches. For example, while q_c may decrease with increasing fines content and increase with cementation and overconsolidation, V_s tends to behave inversely when fines content exceeds a critical threshold. Such complexities underscore the limitations of conventional methods in accurately modeling the multifaceted nature of soil behavior under seismic loading [9].

Although these empirical techniques have formed the backbone of liquefaction hazard assessment for decades, in recent years, machine learning (ML) methods have emerged as transformative tools in civil and geotechnical engineering, offering innovative solutions to complex problems that traditional techniques often struggle to address. One of the key advantages of machine learning techniques over conventional empirical methods is their ability to model nonlinear and multidimensional interactions without the need for a predetermined mathematical form between input and output variables. Traditional approaches typically rely on simplified assumptions and empirical correlations that may overlook the intricate interplay among various soil properties, environmental conditions, and earthquake characteristics. In contrast, ML algorithms can flexibly adapt to the complex behavior of soils under seismic loading, capturing subtle variations that influence liquefaction potential and thereby enhancing prediction accuracy [10]. ML methods can automatically learn from large datasets—such as those derived from standard penetration tests (SPT), cone penetration tests (CPT), and shear wave velocity (V_s) measurements—by adjusting internal parameters to detect underlying patterns and relationships in the data [5].

Moreover, the adaptive learning process inherent in ML techniques offers a dynamic framework that evolves with emerging research and changing site conditions. This continuous updating capacity makes ML-based approaches particularly well-suited for capturing the transient and variable nature of soil behavior under seismic loading, ultimately leading to more resilient and cost-effective engineering designs. Consequently, machine learning is paving the way for a new era of innovation in geotechnical engineering by overcoming traditional limitations through advanced data analytics and computational intelligence [10,11].

2. Materials and Methods

2.1. Dataset

Within the scope of the study, the dataset is prepared for loose sandy soils that is known as susceptible to liquefaction, soils such as gravel, clay, unsaturated soils are excluded. Dataset used in the analyses consists of both liquefied and non-liquefied cases due to earthquakes [12–14]. The selected sources, which reported observed liquefaction and non-liquefaction cases for the 22 earthquake events, are listed in Table 1. The number of cases reported as liquefied and non-liquefied for each earthquake event is given in Figure 1. The total number of data points is 480, and proportions of observed liquefaction and non-liquefaction cases are 66% (316 cases) and 34% (164 cases), respectively, as shown in Figure 2.

Table 1. Selected data sources.

Source	Location
Boulanger et al., 1997	USA (Loma Prieta)
Juang et al., 2003	China (Haicheng), Taiwan (Chi-Chi), USA (Imperial Valley, Loma Prieta, San Fernando Valley)
Boulanger et al., 2014	China (Haicheng, Tangshan), Japan (Hyogo-Ken Nanbu, Nihonkai-Chubu, Niigata, Tohoku), Mexico (Victoria), New Zealand (Christchurch, Darfield, Edgumbe, Inangahua), Taiwan (Chi-Chi), Türkiye (Kocaeli), USA (Borah Peak, Imperial Valley, Loma Prieta, Northridge, Superstition Hills, Westmorland)

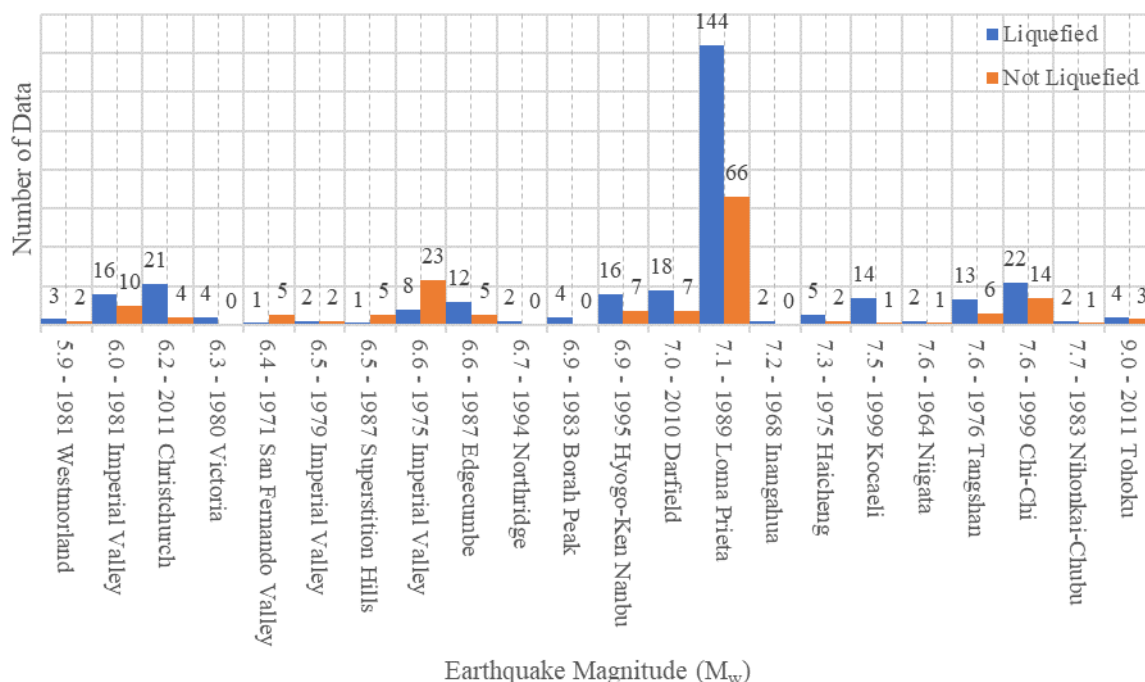


Figure 1. Earthquake events and number of observed liquefaction and non-liquefaction cases.

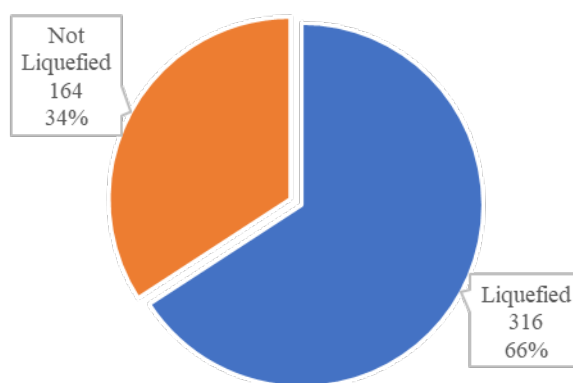


Figure 2. Proportions of observed liquefaction and non-liquefaction cases.

Each data point in the created dataset contains nine quantities, which are earthquake moment magnitude (M_w), peak ground acceleration (a_{max}), measured depth of q_c and f_s (d), cone tip resistance (q_c), cone friction resistance (f_s), cyclic stress ratio (CSR), soil behavior type index (I_c), total stress (σ_v) and effective stress (σ'_v). M_w , a_{max} , d , q_c , f_s , σ_v , σ'_v quantities are determined by measurement, and CSR, I_c quantities are determined depending on the measured quantities.

Researchers conduct studies in cases where liquefaction is observed as a result of earthquakes, and they provide valuable contributions to the literature by reporting their results. Some of these studies added to the literature may include laboratory study results such as dynamic triaxial, monotonic triaxial, consistency limit testing, and sieve analyses. Even simple tests such as sieve analysis and consistency limits require laboratory work. Studies based on field data results can be completed faster and more economically. Therefore, some studies in the literature may not contain laboratory results. Thus, approximately 40% of the studies evaluated within the scope of this study do not have fine material data [6,9–11,14–16]. Field tests are performed as a standard practice in studies about cases of liquefaction after an earthquake. Therefore, almost all of the case studies evaluated within the scope of this study have field test data. The number of data points is one of the major factors that governs the accuracy and reliability of the machine learning methods [17]. For this reason, the high number of input data in machine learning methods was prioritized, in this study. For this purpose, it has been preferred to use the soil type (I_c) parameter instead of fine material content (FC) parameter that may be missing in some studies. I_c is a parameter that indicates the soil type (gravel, sand, silt, silty sand, etc.), and it can be determined based on CPT test results for all data points [3].

The following parameters earthquake moment magnitude (M_w), peak ground acceleration (a_{max}), depth (d), cone tip resistance (q_c), cone friction resistance (f_s), total stress (σ_v), effective stress (σ'_v), cyclic stress ratio (CSR), soil behavior type index (I_c) were evaluated as input parameters of the prediction models. The liquefaction status in the database, which is evaluated as the output parameter, is considered as a binary classification problem to separating patterns between samples where liquefaction occurs (positive) and those where liquefaction does not occur (negative). The histogram and scatter plots of the input parameters used in the study are given in Figure 3.

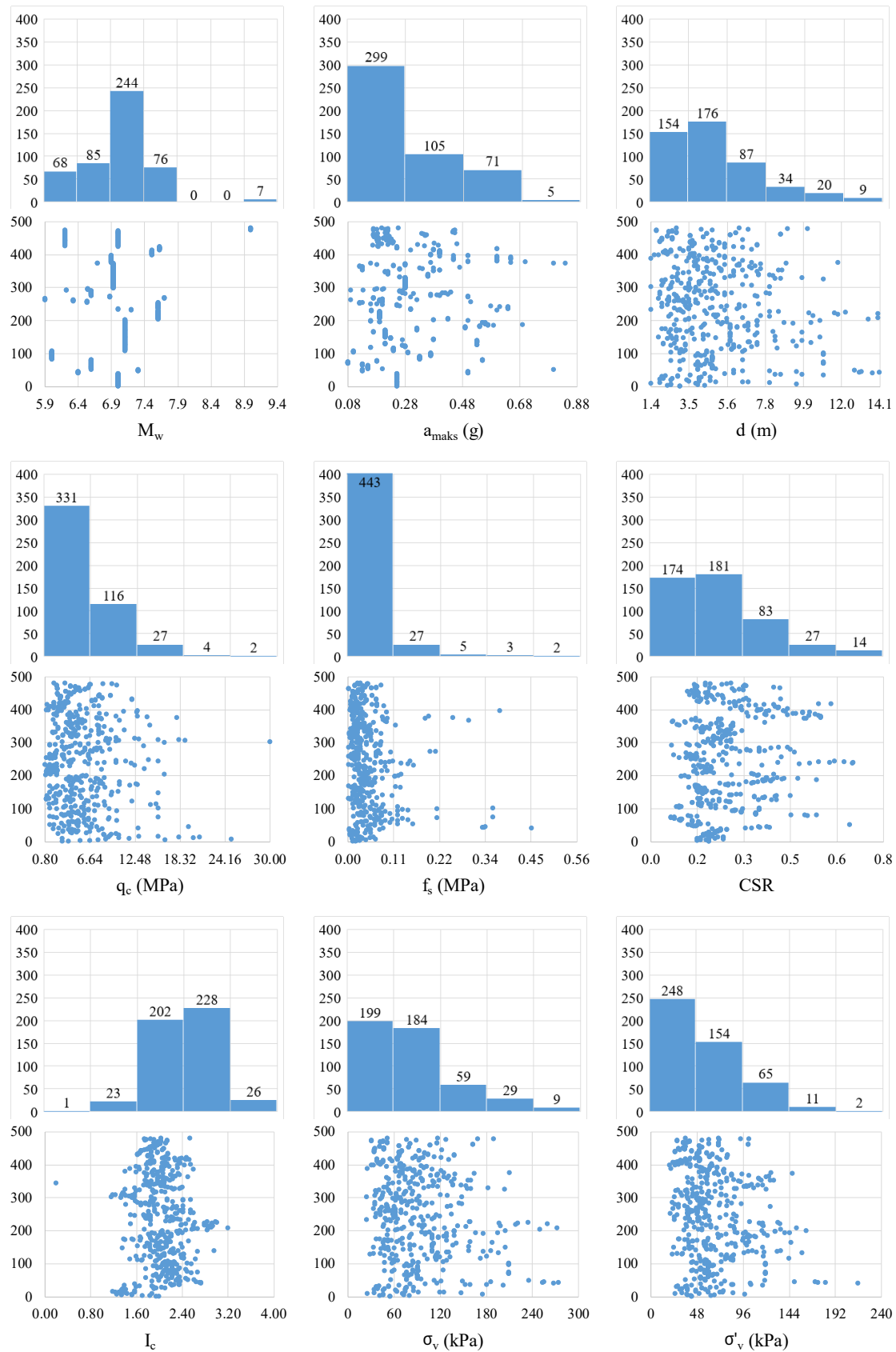


Figure 3. Scatter and histogram of input parameters.

In machine learning models, variables with different scales can negatively impact model performance. Specifically, some algorithms may be biased toward variables with larger values, causing the model to become unbalanced. To prevent this issue, a normalization process is applied

during the data preprocessing stage. In this study, the min-max normalization method given in Equation 1 was used to scale the variables to a certain range. Min-max normalization ensures that all values fall between 0 and 1, allowing variables of different magnitudes to contribute equally.

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Here X represents the original value, X_{min} represents the lowest value in the dataset, and X_{max} represents the highest value [18–21].

As part of this study, min-max normalization was applied to all independent variables used for liquefaction prediction. This process helped balance the effect of variables with different scales, increased the learning efficiency of the model, and minimized errors due to differences in scale between variables [7,17,22]. Traditional liquefaction potential prediction method [3].

Liquefaction potential has been computed for several decades by evaluating field tests such as SPT, CPT and shear wave velocity tests [23]. Cyclic stress ratio (CSR) is calculated using Equation 2, was proposed by Seed and Idriss [24], and it is a worldwide accepted and frequently preferred index for liquefaction computations. The cyclic resistance ratio (CRR) can be computed is calculated using the CPT data [3].

$$CSR = 0.65 \frac{a_{max}}{g} \left(\frac{\sigma_{v0}}{\sigma'_{v0}} \right) r_d \quad (2)$$

Here a_{max} is maximum horizontal surface ground acceleration, g is ground acceleration, σ_{v0} is total vertical stress, σ'_{v0} is effective vertical stress and r_d is stress reduction coefficient.

The Stress Reduction Coefficient r_d , which is included in Equation 2 and allows the flexibility of the soil profile to be considered, is calculated by Equation 3. The z value in the equation indicates the depth (m).

$$r_d = \frac{(1 - 0.4113z^{0.5} + 0.04052z + 0.001753z^{1.5})}{(1 - 0.4177z^{0.5} + 0.05729z - 0.006205z^{1.5} + 0.001210z^2)} \quad (3)$$

The curve separating the areas where liquefaction is observed/not observed, shown in Figure 4, which was presented by combining the studies in the literature to be used for estimating CRR using CPT tip resistance, is approximately given by Equation 10. This graph is known to provide more valid results for clean sands [3], so the values used in the calculations need to be converted to values suitable for clean sands.

The soil behavior table given in Figure 5 was developed using the appropriate value of $n = 1.0$ for clayey soil types. However, a value of $n = 0.5$ would be appropriate for clean sands, and a value between 0.5 and 1.0 for silts and sandy silts [3]. The soil behavior type boundaries shown in Figure 5 were determined according to the I_c value ranges given in Table 2. The soil behavior type index I_c is not valid for zones 1, 8 or 9. Throughout the normally consolidated zone, the soil behavior type index increases in direct proportion to the fines content and plasticity. This relationship is also shown in Figure 6.

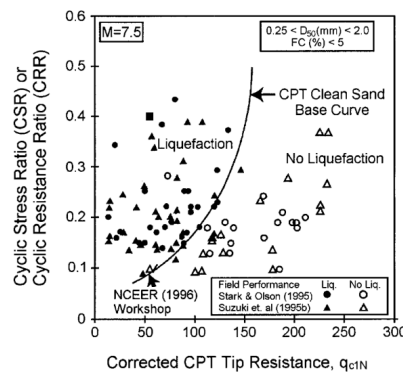


Figure 4. Recommended clean sand curve for use in CRR calculations [3].

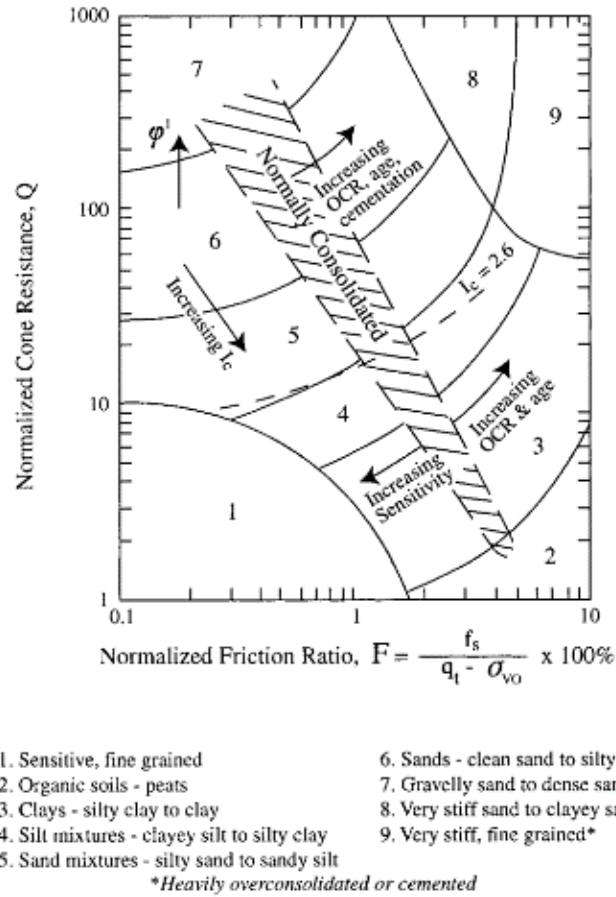


Figure 5. Determination of soil behavior and type with CPT data [3].

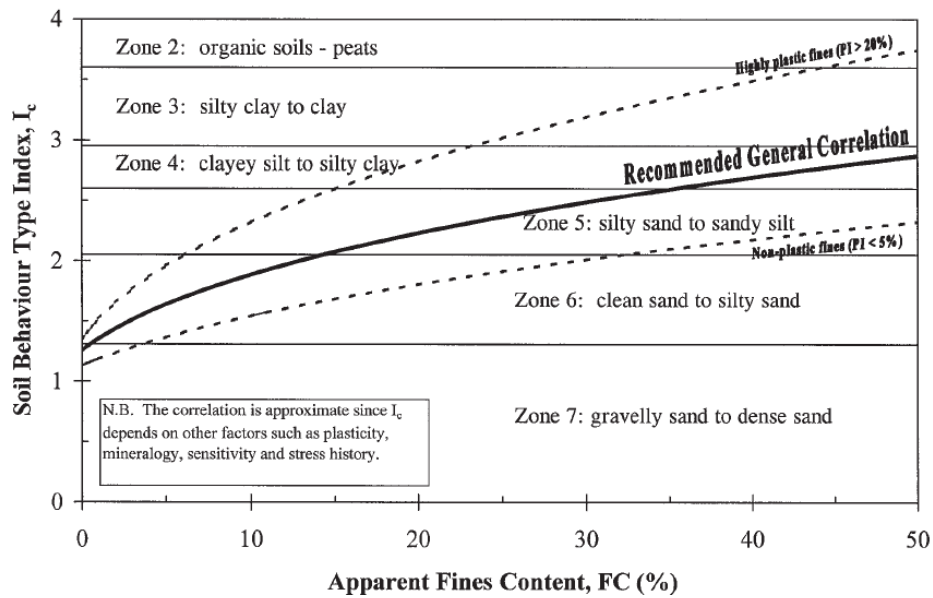


Figure 6. Changes in soil behavior type index according to fines ratio and soil regions (Table 2) [3].

Table 2. Soil Behavior type limits.

Soil Behavior Type Index (I_c)	Region	Soil Behavior Type Limits
$I_c < 1.31$	7	Gravelly sand to dense sand
$1.31 < I_c < 2.05$	6	Sands: clean sand to silty sand
$2.05 < I_c < 2.60$	5	Sand mixtures: silty sand to sandy silt
$2.60 < I_c < 2.95$	4	Silt mixtures: clayey silt to silty clay
$2.95 < I_c < 3.60$	3	Clays: silty clay to clay
$I_c > 3.60$	2	Organic soils: peats

The Cyclic Stress Ratio (CRR) is calculated by following the process steps explained below and summarized in the flow chart given in Figure 8 [3].

Firstly, the soil types defined as clay are distinguished from sand and silt. This distinction is calculated by assuming $n = 1.0$ in the equation given by Equation 4 and calculating the Q value.

$$Q = [(q_c - \sigma_{vo})/P_a][(P_a/\sigma'_{vo})^n] \quad (4)$$

Here, Q is the dimensionless CPT end resistance, q_c is the CPT end resistance, σ_{vo} is the vertical total stress, P_a is the air pressure (100 kPa = 1 atm), σ'_{vo} is the vertical effective stress, and n is a value ranging from 0.5 to 1.0 representing the soil type.

$$F = [f_s/(q_c - \sigma_{vo})] \times 100\% \quad (5)$$

The soil behavior type index I_c given by Equation 6 is calculated using the Q value calculated by Equation 4 and the F value calculated by Equation 5. Here, F is the dimensionless friction ratio and f_s is the CPT friction resistance.

$$I_c = [(3.47 - \log Q)^2 + (1.22 + \log F)^2]^{0.5} \quad (6)$$

If the calculated I_c value is greater than 2.6, the soil is classified as clayey, and the analysis is completed by assuming that it is too rich in clay to liquefy. However, soil samples should be taken and tested with laboratory tests to verify the soil type and liquefaction resistance. If the calculated I_c value is less than 2.6, the soil is classified as granular and therefore, Q and I_c values are recalculated by accepting $n = 0.5$ in the equation given by Equation 4.

If the recalculated I_c value is less than 2.6, the soil is classified as non-plastic and granular. This I_c value is used to estimate the liquefaction resistance in the following steps. If the recalculated I_c value is greater than 2.6, the soil is very silty and probably plastic. In this case, the normalized CPT tip resistance q_{c1N} value for silty sands given by Equation 7 should be recalculated by taking $n = 0.7$.

$$q_{c1N} = \left(\frac{q_c}{P_a}\right) \left(\frac{P_a}{\sigma'_{vo}}\right)^n \quad (7)$$

The I_c value is then recalculated using the recalculated q_{c1N} value instead of the Q value in Equation 6. This I_c value is then used to calculate the liquefaction resistance.

For silty sands, the normalized CPT tip resistance q_{c1N} is corrected to the equivalent clean sand value $(q_{c1N})_{cs}$ using Equation 8.

$$(q_{c1N})_{cs} = K_c \times q_{c1N} \quad (8)$$

Here K_c is the correction factor for the grain characteristics calculated by Equation 9.

$$\text{if } I_c \leq 1.64, \quad K_c = 1.0$$

$$\text{if } I_c > 1.64, \quad K_c = -0.403I_c^4 + 5.581I_c^3 - 21.630I_c^2 + 33.750I_c - 17.880 \quad (9)$$

The K_c curve given by Equation 9 is shown in Figure 7. For $I_c > 2.6$, the curve shown as a dashed line indicates that the soils in this I_c range must have a high clay content or be plastic to liquefy.

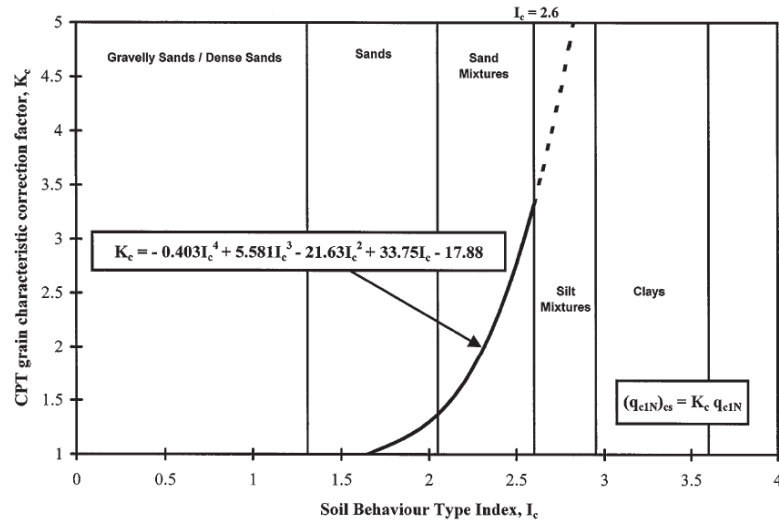


Figure 7. Recommended grain characteristic correction to achieve clean sand equivalent CPT tip resistance in sandy soils [3].

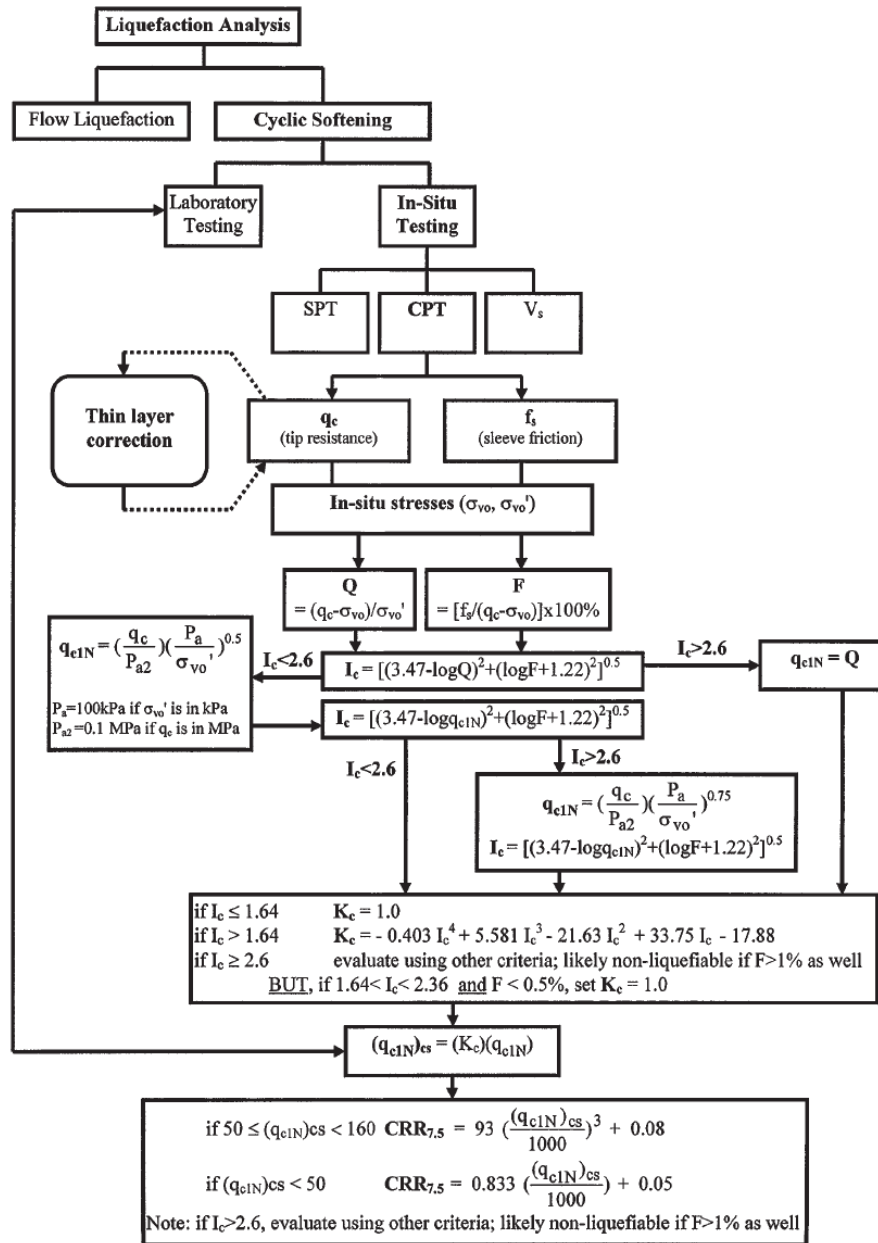


Figure 1. Calculating CRR using CPT data [3].

The obtained $(q_{c1N})_{cs}$ value is used in the relationship given by Equation 10 and as a result of calculating the $CRR_{7.5}$ value, the cyclic resistance ratio (CRR) required for the liquefaction of the soil and the cyclic stress ratio (CSR) created by the earthquake are compared with each other and the liquefaction potential can be estimated with the ratio obtained. As a result of these calculations, it is concluded that there is a risk of liquefaction in the layer if the FS safety factor is less than 1.0. Here $(q_{c1N})_{cs}$ is the clean sand CPT tip resistance normalized to 100 kPa (1 atm).

$$\text{if } (q_{c1N})_{cs} < 50, \quad CRR_{7.5} = 0.833[(q_{c1N})_{cs}/1000] + 0.05$$

$$\text{if } 50 \leq (q_{c1N})_{cs} < 160, \quad CRR_{7.5} = 93[(q_{c1N})_{cs}/1000]^3 + 0.08 \quad (10)$$

2.2. Machine Learning Methods

Liquefaction, defined as the loss of soil strength during earthquakes, poses a significant risk to engineering structures. While traditional empirical and experimental methods can be used for

liquefaction prediction, machine learning methods have recently emerged as a powerful alternative in this field.

Machine learning enables the development of data-driven models that can learn patterns from large datasets to make predictions. Various machine learning algorithms, such as logistic regression, decision trees, support vector machines (SVM), random forest, and artificial neural networks (ANN), have been utilized for liquefaction prediction.

Machine learning-based approaches can predict liquefaction potential by utilizing different variables, including soil properties (e.g., grain size, void ratio), groundwater level, and seismic parameters (e.g., acceleration, magnitude). The advantages of these methods include:

More flexible and faster than traditional methods.

Capable of making high-accuracy predictions with large datasets.

Easily updatable with new data.

In this study, different models were created using Random Forest (RF) and Artificial Neural Networks (ANN) methods for liquefaction prediction and their performances were evaluated and compared with the traditional method. The data set used in the modeling includes field observations with and without liquefaction, and the algorithms are aimed to learn these patterns and make accurate predictions. It is thought that machine learning-based predictions can be a useful tool in engineering decision-making processes.

2.2.1. Random Forest (RF)

The Random Forest method was developed by Leo Breiman in 2001 [25]. While developing the Random Forest method, Breiman was inspired by several previous studies and combined the ideas in these studies to create a new approach. As Breiman stated in his article [25], this method is based on the combination of two main concepts: Bagging (Bootstrap Aggregating) [26] and Random Subspace Method [27].

Random Forest is a machine learning algorithm used for both classification and regression problems that works by combining multiple decision trees. It is based on the basic principle that individual trees may be weak, but when they work together, they can form a strong predictive model.

One of the significant differences between decision trees and random forest methods is the management and reduction of overfitting problem. Overfitting problem is the situation where the model makes predictions that are very close to the learning data, in other words, it memorizes the problem. In this case, the model can make very high accuracy predictions with the learning data, but it can make very low accuracy predictions with the test data. While this problem is evident in the decision tree method, it is controlled in the Random Forest method with the techniques of "Bagging" [26] and "Random Subspace" [27]. These techniques reduce the variance of random forests, increase the generalization ability of the model, and reduce the risk of overfitting. In this way, the random forest method generally provides the creation of high-performance and reliable models.

2.2.1.1. Bagging (Bootstrap Aggregating) [26]

Bagging is an ensemble learning technique used particularly in statistical learning. (Figure 9)

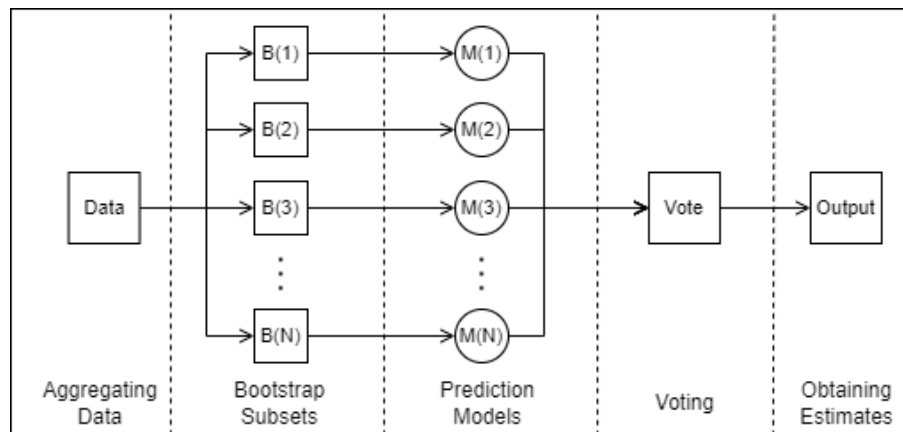


Figure 9. Bagging.

The main purpose of Bagging is to increase generalization power and reduce overfitting by combining many independent models. In this context, Bagging consists of Bootstrap Sampling, Training of Independent Models and Aggregating Model Results steps.

Bootstrap Sampling is the process of creating a new training data set by taking random samples from the original training data. The feature of this sampling is that each sample is selected by replacing the place in the training set. Therefore, the same data point can be selected more than once.

Training of Independent Models is the process of creating a model that is trained independently for each Bootstrap Sampling and each learns on a different data set.

Combining Model Results is the process of combining the predictions of each model after the training phase is completed. For classification problems, the class that receives the most votes from the class predictions made by each model is determined, and for regression problems, the final prediction is made by calculating the arithmetic average of the predictions of all models.

2.2.1.2. Random Subspace Method [27]

The Random Subspace Method is also used to reduce over-learning and increase model generalization power, like the Begging Technique. This method ensures that each model is trained by selecting a random subset from all available features (input parameters). The main differences between the two methods are shown in Table 3.

Table 1. Comparison of begging and random subspace method.

Stages	Bootstrap Aggregating	Random Subspace Method
I. Data Sampling	Data subsets are created using bootstrap sampling.	Feature sampling is performed to select a different subset of features for each tree.
II. Training the Models	Each tree is trained with all features.	Each tree is trained with a subset of features.
III. Combining Results	The prediction with the most votes is determined for classification, and the average of the predictions is determined for regression.	The prediction with the most votes is determined for classification, and the average of the predictions is determined for regression.

2.2.1.3. Structure of Random Forest Method

Instead of branching each node using the best branch among all variables, Random Forest branches each node using the best branch among randomly selected variables at each node (Figure 10). Each dataset is generated from the original dataset with replacement. Then, trees are developed

using random feature (input parameter) selection. In simple terms, Random Forest creates multiple decision trees and combines them to obtain a more accurate and stable prediction.

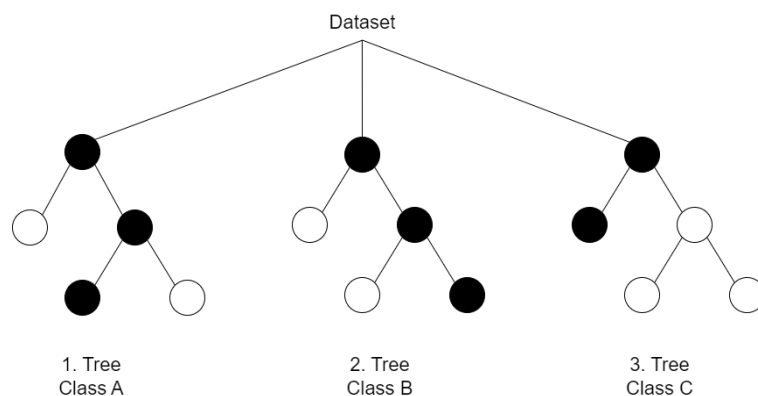


Figure 10. Randomly Selected Variables at Each Node.

Decision trees consist of three main structures: root, decision node and leaf (Figure 11).

Root is the top node of the decision tree and is where the analysis of the entire data set begins. Each branch of the decision tree is assumed to start from this root.

Decision Node is the node where a decision rule is applied and branches. Decision nodes divide the data set by means of a feature and a threshold value determined for this feature. (For example, a decision node can check whether a feature value is greater or less than a certain threshold value. This divides the data set into two subsets.)

Leafs are the last nodes of the decision tree and contain the results. A class label (for classification problems) or a prediction value (for regression problems) is found in the leaf nodes. Leafs are where a branch ends, and the results are obtained. In other words, a leaf node is reached at the end of each branch.

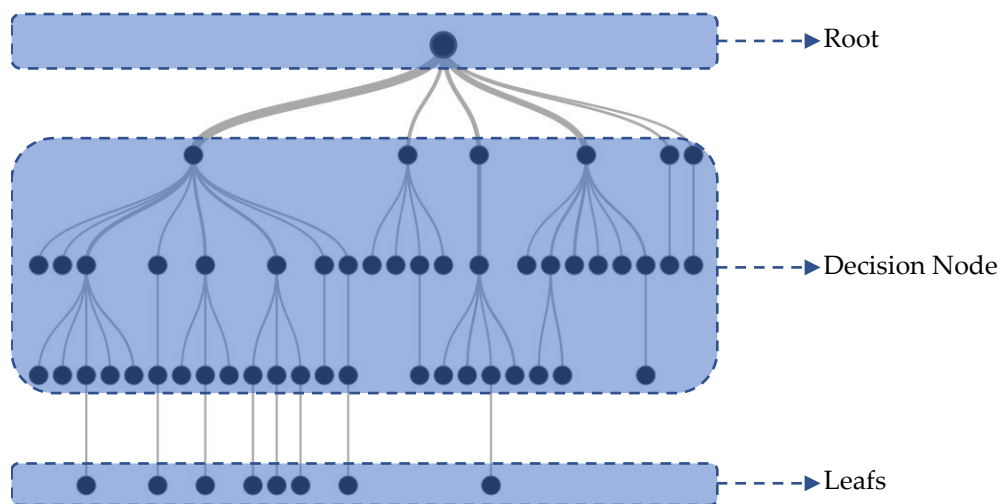


Figure 11. Basic Components of a Decision Tree.

2.2.1.4. Splitting Criteria

Splitting criteria are a crucial step in the construction of decision trees. These criteria decide which feature and which equality persists at a node. The goal is to maximize the information gain at each node by splitting the dataset into purer (homogeneous) subgroups. Different categories can apply different splitting criteria. The most used splitting criteria are:

Gini impurity is a metric that measures the mixing of classes in a node and is calculated by Equation 11. Here C is the number of classes and p_i is the proportion of observations belonging to

class i . Gini impurity is widely used by decision trees (e.g. CART - Classification and Regression Trees). When a node is pure (i.e. contains a single class), its Gini value is 0.

$$Gini = 1 - \sum_{i=1}^c p_i^2 \quad (11)$$

Information gain measures the difference in entropy before and after the split. Entropy is used to measure the confusion or uncertainty in the data set (Equation 12 and Equation 13). Here D is the dataset, D_k are the subsets after the split. Information gain measures the difference in entropy before and after the split. Entropy is used to measure the confusion or uncertainty in the dataset.

$$Entropy = - \sum_{i=1}^c p_i \cdot \log_2(p_i) \quad (12)$$

$$Information\ Gain = Entropy(D) - \sum_k \frac{|D_k|}{|D|} \cdot Entropy(D_k) \quad (13)$$

The information gain can be biased for features (input parameters) with many classes. For example, if a feature has a different value for each observation, the information gain can be maximum. The information gain ratio normalizes the information gain by dividing it by the partition entropy to correct for such biases (Equation 14). This method is widely used in the C4.5 algorithm.

$$Information\ Gain\ Ratio = \frac{Information\ Gain}{Entropy(split)} \quad (14)$$

In this study, since the Random Forest method, which is a decision tree method, was preferred, the Gini Impurity method was used as the division criterion applied at the nodes.

2.2.2. Artificial Neural Network (ANN)

Artificial neural networks (ANNs) are programming models inspired by the principles of biological neural networks. These models, especially in developmental problems, learn complex and immutable relationships in the entered data and determine whether it belongs to the registered separated categories, and produce effective results.

Multilayer perceptron (MLP), one of the most basic examples of these approaches, has a fully connected, feedforward network architecture. MLP usually consists of three main layers: input, hidden and output layer. In the input field, it consists of the lengths representing the features of the data in the first step of the model. These distributions, while carrying a data feature to the model, process the incoming information of the hidden layers and consist of more abstract and complex representation forms. Thanks to the nonlinear activation amount (amount, sigmoid, ReLU, tanh) used in the hidden layers, the model can verify the complexity ratio of the interval intervals. The output distribution in the last stage, the result of the termination, emerges; Here, SoftMax is generally preferred for multi-class problems, and sigmoid activation function is preferred for binary development problems. This power determines which category the data belongs to by calculating the prediction probabilities of each class.

The training process of MLP is carried out with backpropagation applications (backpropagation) that allow the weight to be adjusted appropriately. First, the input data is processed through forward programming (forward propagation) throughout the network and is produced as predicted because of the programming performed in each layer. This produced output is checked with the correct labels in the training record and the error (loss) value is specified. Cross entropy loss is usually used for these error images in classification problems. In addition, the calculated error is derived according to the layers by means of the chain rule (chain rule) and the weight of each connection is updated by the method of regime methods such as gradient descent (gradient descent). The training process is optimized by repeating these cycles until the performance and model error values reach the recorded tolerances with an epoch that occurs when all the training data is persisted from the network once.

MLP also includes bias terms that increase the expansion of the model. Bias is added to the activation function of each part, contributing to better adaptation during the learning process of the model. Hyperparameters such as ratio are also principal elements of the extent of updating the weights and the direct learning characteristics of the training processes. Correct adjustment of these hyperparameters helps to ensure generalization by preventing excessive or insufficient updates during the training of the model.

As a result, artificial neural network models such as MLP stand out with their superior ability to reveal hidden patterns and nonlinear relationships in data in complex solved problems. This structure and operations allow MLP to reduce errors during training and then produce successful production results on test data.

3. Results

3.1. Hyperparameter Optimization

Hyperparameters are predefined settings that directly affect the learning process and overall accuracy of the model. Wrongly selected hyperparameters can lead to problems such as overfitting or underfitting, causing the model to perform poorly on real-world data. Therefore, determining the optimal hyperparameters is a critical step to increase the generalization ability of the model and maximize its accuracy.

In this study, hyperparameter optimization was applied to improve the performance of RF and ANN models. The optimized hyperparameters used in the RF model are shown in Table 4, and the optimized hyperparameters used in the ANN model are shown in Table 5. This optimization process increased the generalization ability of the models and provided more reliable results in liquefaction prediction.

Table 2. Hyperparameters for RF model.

Number of trees	190
Tree depth	11

Table 3. Hyperparameters for ANN model.

Number of iterations	160
Number of hidden layers	5
Number of hidden neurons per layer	10

3.2. Correct and Incorrect Estimated Values

True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) concepts are the basic parameters used when evaluating the prediction results of a model in classification problems. These concepts, summarized in Table 6, are calculated because of comparing the results predicted by the model with the actual results.

Classification models have four key outcomes. A True Positive (TP) occurs when the model correctly identifies a positive class as positive, accurately recognizing the presence of a condition. Conversely, a False Positive (FP) happens when the model incorrectly predicts a negative class as positive, mistakenly indicating the presence of a condition when it is absent. Similarly, a True Negative (TN) is observed when the model accurately classifies a negative class as negative, correctly acknowledging the absence of a condition. Lastly, a False Negative (FN) takes place when the model incorrectly predicts a positive class as negative, thereby failing to detect a condition that is present.

Table 4. Comparing model predicted results with actual results.

	Positive Prediction	Negative Prediction
Positive Observation	True Positive (TP)	False Negative (FN)
Negative Observation	False Positive (FP)	True Negative (TN)

3.3. Performance Metrics

The accuracy and effectiveness of predict results need to be measured with various metrics.

3.3.1. Recall

Recall expresses the rate at which the model correctly predicts positive classes (Equation 15). It is also known as sensitivity.

$$Recall = \frac{TP}{TP + FN} \quad (15)$$

3.3.2. Precision

Precision measures how accurate the positive predictions of the model are. It is expressed as the ratio of true positive predictions to all positive predictions (Equation 16).

$$Precision = \frac{TP}{TP + FP} \quad (16)$$

3.3.3. Specificity

Specificity refers to the rate at which the model correctly predicts negative classes as negative (Equation 17).

$$Specificity = \frac{TN}{TN + FP} \quad (17)$$

3.3.4. F-Measure

It represents the harmonic mean of precision and recall (Equation 18). It is preferred especially in cases where the positive class is more important.

$$F \text{ measure} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (18)$$

3.3.5. Accuracy

The proportion of classes that the model predicts correctly (Equation 19) is defined as accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (19)$$

3.3.6. Cohen's Kappa

Cohen's Kappa compares the classification performance of the model to the probability of making a random guess. It eliminates the effect of random guess. It is calculated by Equation 20. Here P_o is defined as the observed accuracy rate $((TP + TN)/Sum)$, P_e is defined as the expected accuracy rate (probability of random guess).

$$\kappa = \frac{P_o - P_e}{1 - P_e} \quad (20)$$

3.4. Evaluation of Performance Metrics

The number of correct and incorrect predictions and performance metrics obtained as a result of the study are shown in Table 7 for RF, Table 8 for ANN, and Table 9 for the traditional method; all results are summarized in Table 10.

Table 5. Random forest accuracy statistics.

Fold No	TP	FP	TN	FN	Recall	Precision	Sensitivity	Specificity	F-measure	Cohen's Kappa	Accuracy
0	30	4	12	2	0.94	0.88	0.94	0.75	0.91	0.71	0.88
1	29	4	12	3	0.91	0.88	0.91	0.75	0.89	0.67	0.85
2	30	5	11	2	0.94	0.86	0.94	0.69	0.90	0.66	0.85
3	30	2	14	2	0.94	0.94	0.94	0.88	0.94	0.81	0.92
4	31	4	12	1	0.97	0.89	0.97	0.75	0.93	0.75	0.90
5	31	6	10	1	0.97	0.84	0.97	0.63	0.90	0.64	0.85
6	29	0	17	2	0.94	1.00	0.94	1.00	0.97	0.91	0.96
7	29	5	12	2	0.94	0.85	0.94	0.71	0.89	0.67	0.85
8	29	3	14	2	0.94	0.91	0.94	0.82	0.92	0.77	0.90
9	30	3	14	1	0.97	0.91	0.97	0.82	0.94	0.81	0.92
	Mean				0.94	0.89	0.94	0.78	0.92	0.74	0.89

Table 6. Artificial neural network accuracy statistics.

Fold No	TP	FP	TN	FN	Recall	Precision	Sensitivity	Specificity	F-measure	Cohen's Kappa	Accuracy
0	27	3	13	5	0.84	0.90	0.84	0.81	0.87	0.64	0.83
1	29	5	11	3	0.91	0.85	0.91	0.69	0.88	0.61	0.83
2	28	7	9	4	0.88	0.80	0.88	0.56	0.84	0.46	0.77
3	29	2	14	3	0.91	0.94	0.91	0.88	0.92	0.77	0.90
4	30	3	13	2	0.94	0.91	0.94	0.81	0.92	0.76	0.90
5	30	8	8	2	0.94	0.79	0.94	0.50	0.86	0.48	0.79
6	30	5	12	1	0.97	0.86	0.97	0.71	0.91	0.71	0.88
7	30	6	11	1	0.97	0.83	0.97	0.65	0.90	0.66	0.85
8	27	1	16	4	0.87	0.96	0.87	0.94	0.92	0.78	0.90
9	30	3	14	1	0.97	0.91	0.97	0.82	0.94	0.81	0.92
	Mean				0.92	0.88	0.92	0.74	0.89	0.67	0.86

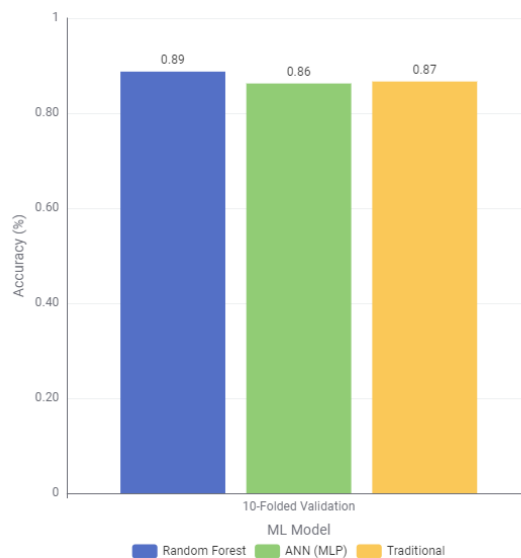
Table 7. Traditional method accuracy statistics.

TP	FP	TN	FN	Recall	Precision	Sensitivity	Specificity	F-measure	Cohen's Kappa	Accuracy
290	38	126	26	0.92	0.88	0.92	0.77	0.90	0.70	0.87

Table 8. Comparison of accuracy statistics obtained with all models.

Model	Recall	Precision	Sensitivity	Specificity	F-measure	Cohen's Kappa	Accuracy
RF	0.94	0.89	0.94	0.78	0.92	0.74	0.89
ANN	0.92	0.88	0.92	0.74	0.89	0.67	0.86
Traditional	0.92	0.88	0.92	0.77	0.90	0.70	0.87

The accuracy values obtained for RF, ANN and the traditional method are given in Figure 12. The highest accuracy value was reached with the RF model with 89%. Accuracy rates of 87% were obtained for the traditional method and 86% for ANN.

**Figure 2.** Comparison of accuracy values obtained with all models.

3.5. ROC Curve and AUC

ROC curve (Receiver Operating Characteristic Curve) is a graph used to analyze how a classification model performs at different threshold values. It is drawn by calculating the True Positive Rate (TPR) and False Positive Rate (FPR) values. TPR (Sensitivity) is the rate at which the model correctly predicts true positives (Equation 21). FPR (False Positive Rate - Fall-out) is the rate at which the model falsely predicts true negatives as positives (Equation 22).

$$TPR = \frac{TP}{TP + FN} \quad (21)$$

$$FPR = \frac{FP}{FP + TN} \quad (22)$$

If the ROC curve is close to the upper left corner, it indicates that the model is performing well. If the curve is close to the 45-degree diagonal line, it means that the model is making random predictions. The ROC curve is given in Figure 13 for RF and in Figure 14 for ANN.

AUC (Area Under the Curve) refers to the area under the ROC curve and measures the overall classification performance of the model. If the AUC value is close to 1, the model makes excellent discrimination, if AUC = 0.5, the model makes random predictions, and if AUC < 0.5, the model makes

poor classification. The AUC value is one of the most important metrics that show how well the model distinguishes positive and negative classes. A value between 0.7-0.8 is considered good, 0.8-0.9 is considered very good, and greater than 0.9 is considered an excellent model. The ROC curve and AUC are important metrics for comparing different models, evaluating the model in datasets with class imbalance, and determining the optimal threshold value. The ROC curves given in Figure 13 and Figure 14 were evaluated and the area under the graph was calculated as 0.932 for RF and 0.872 for ANN.

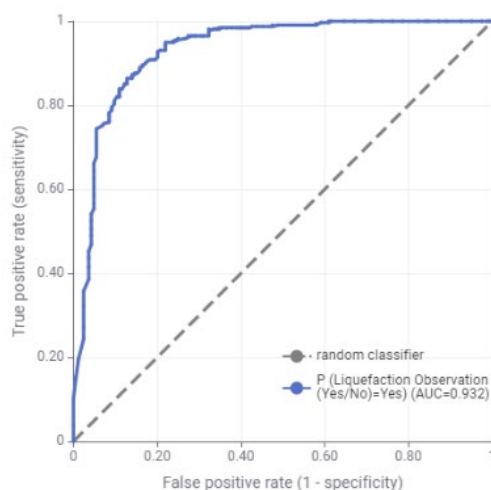


Figure 3. ROC Curve of RF.

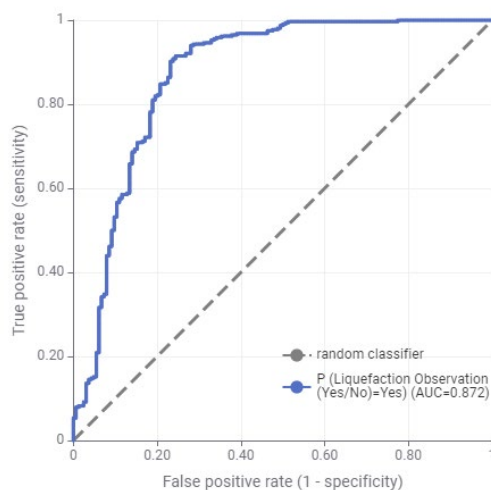


Figure 14. ROC Curve of ANN.

3.6. Feature Importance

The split number indicates how many times a feature is used for the split operation in the nodes in the trees. In each decision tree, the nodes divide the data into two subsets. During this splitting operation, one of the features is selected and splits the data according to a certain threshold value. The split number indicates how often the model uses that feature in the decision processes. A high split number indicates that the feature plays an important role in the classification or prediction

process of the model. If the split number of a feature is very low (or even zero), this may indicate that the feature is generally unimportant to the model or is used less than other features.

The number of candidates indicates how many times a feature is considered a candidate for splitting. At each node, the model makes the best split decision from a randomly selected subset of features (candidate features). The number of candidates indicates how many times a feature is included in this subset. This number usually depends on the total number of nodes and the size of the randomly selected subset. A high number of candidates indicates that the feature is frequently included in randomly selected subsets and therefore frequently considered for splitting. However, being a candidate does not necessarily mean that a feature will be used. The model selects the feature that best splits.

Candidate > Split: Not every feature may be used for splitting in every case it is selected as a candidate. In other words, a feature may be frequently nominated in nodes, but may be used less because it does not perform better than other features for splitting.

Split \approx Candidate: If a feature is both frequently nominated and frequently used for splitting, this means that the feature is very important for the model.

Split \ll Candidate: If the candidate number of a feature is high but the number of splits is low, this may indicate that the feature generally falls short of the splitting criterion.

The split and candidate numbers in the first three levels of Random Forest trees are shown in Table 11. Here, the ratio of the split numbers in the first three levels to the total number of splits for each feature (input parameter) is significant in terms of feature importance. When the split number ratios given in Figure 15 are evaluated; It is seen that the parameters q_c , a_{max} , f_s , CSR and I_c are more important than other parameters in the Random Forest model, respectively. Since feature importance is not directly calculated in ANN models compared to the RF module, only feature importance for the RF model was evaluated within the scope of the study.

Table 9. Number of split and candidate.

	Split			Candidate		
	Level 0	Level 1	Level 2	Level 0	Level 1	Level 2
M_w	0	20	23	60	139	246
a_{max}	36	80	77	70	153	251
d	7	13	53	72	135	248
q_c	52	76	105	52	115	249
f_s	34	55	101	62	124	265
CSR	29	53	100	80	113	266
I_c	25	36	85	55	115	255
σ_v	4	20	59	53	131	248
σ'_v	3	21	47	66	115	216

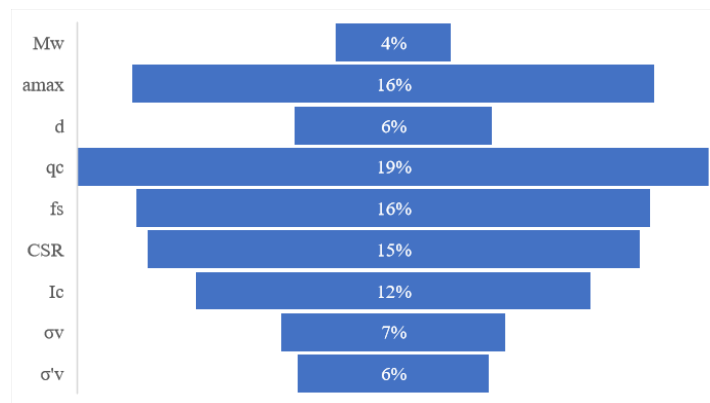


Figure 15. Feature importance ratios.

4. Discussion

In this study, the potential of machine learning (ML) techniques for predicting seismic soil liquefaction was evaluated using Random Forest (RF) and Artificial Neural Network (ANN) models, and their performances were compared with the traditional Cone Penetration Test (CPT) based method. The findings clearly demonstrate the capability of ML algorithms in analyzing complex soil behaviors under seismic loading and highlight their advantages over conventional empirical approaches.

The analyses conducted to compare the ML models and the traditional method reveal significant differences in classification success:

Accuracy Rates: According to the performance metrics, the RF model achieved the highest accuracy rate at 89%. This was followed by the traditional CPT-based method with 87% and the ANN model with 86%.

ROC-AUC Analysis: When evaluating the Area Under the Curve (AUC) values, the RF model yielded an AUC of 0.932, whereas the ANN model scored 0.872. An AUC value greater than 0.9 indicates that the RF model possesses an "excellent" classification capability in distinguishing between liquefied and non-liquefied cases. Although the ANN model is considered a "very good" model with an AUC between 0.8 and 0.9, it fell behind the RF model.

Overfitting Control: The overfitting problem, which is frequently encountered in decision tree methodologies, was successfully managed in the RF model through the use of "Bagging" and "Random Subspace" techniques. This structural advantage effectively reduces the variance of the model and increases its generalization capability, confirming why RF provided more robust predictions compared to the ANN model.

The fact that the traditional method achieved a relatively high accuracy of 87% proves its long-standing empirical validity. However, traditional deterministic approaches often struggle to fully capture the complex, nonlinear interactions between soil properties and seismic parameters.

Conventional methods typically rely on simplified assumptions and deterministic correlations to account for multifaceted parameters such as fines content or plasticity.

In contrast, machine learning algorithms can model nonlinear and multidimensional interactions flexibly, without the need for a predetermined mathematical form between input and output variables.

Furthermore, the dynamic framework of ML techniques allows them to continuously update and adapt as new data emerges, offering a distinct advantage over static traditional models in capturing the transient and variable nature of soil behavior.

Identifying which parameters play a more significant role in prediction models directly impacts engineering decision-making processes. The feature importance analysis applied to the RF model revealed the parameters that the model relied on most heavily:

The analysis demonstrated that the most influential variables used in the splitting operations of the trees were, respectively, cone tip resistance (q_c), peak ground acceleration (a_{max}), cone friction resistance (f_s), cyclic stress ratio (CSR), and soil behavior type index (I_c).

Considering that the fundamental mechanism underlying liquefaction is the cyclic shearing induced by seismic forces, it is theoretically consistent that a_{max} and CSR are among the most critical parameters for the model.

Similarly, the high importance of direct field measurement metrics such as q_c and f_s , which characterize the in-situ physical state of the soil, provides a strong physical validation that reinforces the model's accuracy.

Author Contributions: Methodology, S.T. and E.B.; Validation, E.B.; Formal analysis, S.T.; Investigation, E.B.; Resources, S.T.; Writing—original draft, S.T.; Writing—review & editing, E.B.; Visualization, S.T.; Supervision, E.B.; Project administration, E.B. 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: Data is contained within the article.

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

References

1. Baziar, M.H.; Nilipour, N. Evaluation of Liquefaction Potential Using Neural-Networks and CPT Results. *Soil Dynamics and Earthquake Engineering* **2003**, *23*, 631–636, doi:10.1016/S0267-7261(03)00068-X.
2. Liu, B.Y.; Ye, L.Y.; Xiao, M.L.; Miao, S. Artificial Neural Network Methodology for Soil Liquefaction Evaluation Using CPT Values. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2006**, *4113 LNCS-I*, 329–336, doi:10.1007/11816157_36.
3. Robertson, P.K.; Wride, C.E. Evaluating Cyclic Liquefaction Potential Using the Cone Penetration Test. *Canadian Geotechnical Journal* **1998**, *35*, 442–459, doi:10.1139/T98-017.
4. Seed, R.; Cetin, K.; Moss, R.; Kammerer, A.; Wu, J.; Pestana, J.; Riemer, M.; Sancio, R.; Bray, J.; Kayen, R.; et al. Recent Advances in Soil Liquefaction Engineering: A Unified and Consistent Framework. *Proceedings of the 26th Annual ASCE Los Angeles Geotechnical Spring Seminar: Long Beach, CA* **2003**.
5. Hanna, A.M.; Ural, D.; Saygili, G. Neural Network Model for Liquefaction Potential in Soil Deposits Using Turkey and Taiwan Earthquake Data. *Soil Dynamics and Earthquake Engineering* **2007**, *27*, 521–540, doi:10.1016/J.SOILDYN.2006.11.001.
6. Tung, A.T.Y.; Ya Yung Wang; Wong, F.S. Assessment of Liquefaction Potential Using Neural Networks. *Soil Dynamics and Earthquake Engineering* **1993**, *12*, 325–335, doi:10.1016/0267-7261(93)90035-P.
7. Wang, J.; Rahman, M.S. A Neural Network Model for Liquefaction-Induced Horizontal Ground Displacement. *Soil Dynamics and Earthquake Engineering* **1999**, *18*, 555–568, doi:10.1016/S0267-7261(99)00027-5.
8. Erzin, Y.; Ecemis, N. The Use of Neural Networks for CPT-Based Liquefaction Screening. *Bulletin of Engineering Geology and the Environment* **2015**, *74*, 103–116, doi:10.1007/S10064-014-0606-8/TABLES/13.
9. Livingston, G.; Piantedosi, M.; Kurup, P.; Sitharam, T.G. Using Decision-Tree Learning to Assess Liquefaction Potential from CPT and Vs. *Geotechnical Earthquake Engineering and Soil Dynamics IV* **2008**, 1–10.
10. Kohestani, V.R.; Hassanlourad, M.; Ardakani, A. Evaluation of Liquefaction Potential Based on CPT Data Using Random Forest. *Natural Hazards* **2015**, *79*, 1079–1089, doi:10.1007/S11069-015-1893-5/FIGURES/3.
11. Demir, S.; Sahin, E.K. Comparison of Tree-Based Machine Learning Algorithms for Predicting Liquefaction Potential Using Canonical Correlation Forest, Rotation Forest, and Random Forest Based on CPT Data. *Soil Dynamics and Earthquake Engineering* **2022**, *154*, 107130, doi:10.1016/J.SOILDYN.2021.107130.
12. Boulanger, R.W.; Idriss, I.M.; Boulanger, R.W. *CPT and SPT Based Liquefaction Triggering Procedures*; 2014;
13. Boulanger, R.W.; Meija, L.H.; Idriss, I.M. Liquefaction at Moss Landing during Loma Prieta Earthquake. *Journal of Geotechnical and Geoenvironmental Engineering* **1997**, *123*, 453–467, doi:10.1061/(ASCE)1090-0241(1997)123:5(453).
14. Juang, C.H.; Yuan, H.; Lee, D.-H.; Lin, P.-S. Simplified Cone Penetration Test-Based Method for Evaluating Liquefaction Resistance of Soils. *Journal of Geotechnical and Geoenvironmental Engineering* **2003**, *129*, 66–80, doi:10.1061/(ASCE)1090-0241(2003)129:1(66).
15. Nejad, A.S.; Guler, E.; Ozturan, M. *Evaluation of Liquefaction Potential Using Random Forest Method and Shear Wave Velocity Results*; Institute of Electrical and Electronics Engineers Inc.: Budapest, Hungary, 2018;
16. Juang, C.H.; Chen, C.J.; Tien, Y.-M. Appraising Cone Penetration Test Based Liquefaction Resistance Evaluation Methods: Artificial Neural Network Approach. <https://doi.org/10.1139/t99-011> **1999**, *36*, 443–454, doi:10.1139/T99-011.

17. Pacheco, V.L.; Bragagnolo, L.; Dalla Rosa, F.; Thomé, A. Cone Penetration Test Prediction Based on Random Forest Models and Deep Neural Networks. *Geotechnical and Geological Engineering* **2023**, *41*, 4595–4628, doi:10.1007/S10706-023-02535-0/TABLES/2.
18. Kumar, D.R.; Samui, P.; Burman, A.; Wipulanusat, W.; Keawsawasvong, S. Liquefaction Susceptibility Using Machine Learning Based on SPT Data. *Intelligent Systems with Applications* **2023**, *20*, 200281, doi:10.1016/J.ISWA.2023.200281.
19. Sui, Q. ru; Chen, Q. huang; Wang, D. dan; Tao, Z. gang Application of Machine Learning to the Vs-Based Soil Liquefaction Potential Assessment. *J. Mt. Sci.* **2023**, *20*, 2197–2213, doi:10.1007/S11629-022-7809-4/METRICS.
20. Ozsagir, M.; Erden, C.; Bol, E.; Sert, S.; Özocak, A. Machine Learning Approaches for Prediction of Fine-Grained Soils Liquefaction. *Comput. Geotech.* **2022**, *152*, 105014, doi:10.1016/J.COMPGEO.2022.105014.
21. Huang, S.; Huang, M.; Lyu, Y. An Improved KNN-Based Slope Stability Prediction Model. *Advances in Civil Engineering* **2020**, *2020*, 8894109, doi:10.1155/2020/8894109.
22. Kuran, F.; Tanurcan, G.; Pashaei, E. Developing Machine Learning-Based Ground Motion Models to Predict Peak Ground Velocity in Turkiye. *J. Seismol.* **2024**, *28*, 1183–1204, doi:10.1007/S10950-024-10239-Y/FIGURES/18.
23. Youd, T.L.; Idriss, I.M. Liquefaction Resistance of Soils: Summary Report From The 1996 NCEER and 1998 NCEER/NSF Workshops of Evaluation of Liquefaction Resistance of Soils. *Journal of Geotechnical and Geoenvironmental Engineering* **2001**, *127*, 297–313, doi:/10.1061/(ASCE)1090-0241(2001)127:4(297).
24. Seed, H.B.; Idriss, I.M. Ground Motions and Soil Liquefaction during Earthquakes; Oakland, California, 1982;
25. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32, doi:10.1023/A:1010933404324.
26. Breiman, L. Bagging Predictors. **1996**, *24*, 123–140, doi:10.1007/BF00058655.
27. Ho, T.K. The Random Subspace Method for Constructing Decision Forests. *IEEE Trans. Pattern Anal. Mach. Intell.* **1998**, *20*, 832–844, doi:10.1109/34.709601.

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.