Journal of the Civil Engineering Forum, January 2026, 12(1):23-39
DOI 10.22146/jcef.21347
Available Online at https://jurnal.ugm.ac.id/v3/jcef/issue/archive

Enhancing Soil Liquefaction Prediction: Overcoming Data Challenges in SPT-Based Machine Learning with Imputation Technique

Fandi Fadliansyah[1], Fikri Faris[1,4*], Wahyu Wilopo[2,4], Ardiansyah[3]
[1]Department of Civil and Environmental Engineering, Universitas Gadjah Mada, Yogyakarta, INDONESIA
[2]Department of Geological Engineering, Universitas Gadjah Mada, Yogyakarta, INDONESIA
[3]Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Lampung, INDONESIA
[4]Center for Disaster Mitigation and Technological Innovation (GAMA-InaTEK), Universitas Gadjah Mada, Yogyakarta, INDONESIA
*Corresponding author: fikri.faris@ugm.ac.id

SUBMITTED 05 May 2025 **REVISED** 20 July 2025 **ACCEPTED** 23 July 2025

ABSTRACT In addition to the adverse effects of earthquakes, the loss of soil-bearing capacity during liquefaction can exacerbate damage to buildings. Liquefaction phenomena involve many parameters, making it more complex to evaluate. Machine learning has been studied to deal with liquefaction complexity in recent decades. However, incomplete liquefaction data can result in missing information, complicating model development across various datasets. Therefore, this study aims to assess the capability of machine learning models to predict liquefaction by implementing the missing value imputation technique. Seismicity, soil properties, and soil condition parameters were utilized to develop models. Random Forest (RF), k-Nearest Neighbor (k-NN), and eXtreme Gradient Boosting (XGBoost) were trained by applying feature selection and parameter optimization based on standard penetration test (SPT) data. The confusion matrix was used to assess the performance of the model based on the performance matrix of Overall Accuracy (*OA*), Precision (*Prec*), Recall (*Rec*), F1-Score (*F1*), and Area Under the Curve (*AUC*). In addition, the preprocessing stage included data normalization and outlier treatment to enhance the reliability of model predictions, ensuring consistent learning behavior across different variable scales. The results show that the RF achieved the highest performance (*OA* = 90.71%), which is comparable to findings from other previous studies. The *AUC* results indicate that the models deliver excellent classification performance. These findings suggest that the integration of imputation and preprocessing techniques can significantly improve data-driven approaches in geotechnical earthquake engineering. In conclusion, the missing imputation is quite effective in the predictive model. Finally, this study offers a new perspective on developing machine learning models using a more user-friendly software and applying imputation techniques to handle missing data.

KEYWORDS Machine learning; Missing value imputation; Soil liquefaction; Earthquake; Standard penetration test.

1 INTRODUCTION

1.1 Seismic-Induced Liquefaction and Advances in Identification Methods

The movement of tectonic plates can lead to vibrations known as earthquakes. The intensity of an earthquake can lead to devastating effects. In addition, earthquakes, earthquakes can trigger other natural disasters, such as soil liquefaction. Soil liquefaction can worsen the damage to building infrastructure. This phenomenon caused extensive damage, as happened in the 1964 Niigata earthquake in Japan, the 1964 Alaska earthquake, the 1999 Chi-Chi and Kocaeli earthquakes, and the 2018 Palu earthquake. Soil liquefaction can cause the building's foundation to crack and collapse, resulting in fatalities. As a result of liquefaction, formerly solid soil transforms into a liquid, causing structures above it to sink and damaging underground infrastructure such as pipes and cable networks. Therefore, identifying the vulnerability of soil liquefaction is critical.

A comprehensive knowledge of liquefaction empowers engineers, planners, and policymakers to identify areas that may undergo liquefaction effectively. Liquefaction assessment is particularly challenging because many variables that affect liquefaction do not always correlate directly. In addition, identifying liquefaction potential becomes more complex because there is no single conclusive indicator. Experts assess liquefaction susceptibility by analyzing several factors, such as seismicity and soil conditions. This approach gained acceptance until the simplified procedure, an empirical method, emerged.

The simplified procedure, introduced by Seed and Idriss (1971), is frequently used to evaluate liquefaction potential. The simplified approach for assessing liquefaction potential involves determining the safety factor by comparing the cyclic resistance ratio (*CRR*) and the cyclic stress ratio (*CSR*). This method has been continuously updated (Youd

et al.,2001; Idriss and Boulanger,2008; Boulanger and Idriss,2014) through empirical evaluation, laboratory test results, and the availability of field test data. Soil investigation data using standard penetration test (SPT), cone penetration test (CPT), and shear wave velocity (V_S) techniques are examples of field test data that are commonly used for empirical or semi-empirical analysis. The SPT method is the most widely used in estimating liquefaction among the three methods.

1.2 Overview of Machine Learning Models in Liquefaction Assessment

Geological processes, composition, depth, and other factors influence soil characteristics. The simplified procedure method occasionally cannot accommodate the variability of complex soil properties. Furthermore, the parameters used to assess liquefaction tend to have abstract or non-linear relationships with each other (Zhang and Wang, 2021; Demir and Sahin, 2022*a*). Inappropriate handling of soil property variability may lead to a reduction in the accuracy of liquefaction assessment. Thus, an alternative method is needed to deal with such conditions. Machine learning algorithms can handle complex problems by determining the hidden patterns in data that humans may not notice (Xie et al., 2020). This capability allows for a better understanding of the factors inducing liquefaction, including the interactions between various geotechnical variables.

Machine learning is increasingly utilized across various fields to address complex issues and identify hidden information in large datasets. It has proven reliable in solving geotechnical problems (Puri et al., 2018; Tang and Na, 2021; Galupino and Dungca, 2022; Torres and Dungcaa, 2024). In recent decades, various researchers have used machine learning to estimate liquefaction vulnerabilities. A previous study conducted by (Gandomi et al., 2013) used 620 data points from the last liquefaction occurrences to compare the performance of the decision tree (DT) approach and the logistic regression (LR) algorithm. They concluded that DT could successfully predict liquefaction and outperform the LR model. Using 24 field observations from the 1976 Tangshan earthquake and the 1997 Sanshui earthquake, Xue et al. (2017) employed a hybrid probabilistic neural network (PNN) and particle swarm optimization (PSO) method to predict liquefaction. The result indicates that PSO-PNN is an effective instrument for analyzing the complex relationship between soil and seismic properties in liquefaction evaluation.

Some probabilistic approaches were introduced by Zhao et al. (2021, 2022, 2024) to evaluate the liquefaction potential. The models were developed using Python programming software. The probabilistic method was found to be reliable in predicting liquefaction using CPT-based, as well as the combination of V_S and CPT-based datasets. Kumar et al. (2022) developed machine learning models using soft computing techniques based on SPT data. The result implies that the XGBoost model is the most efficient of the four models. Three ensemble algorithms, AdaBoost, Gradient Boosting Machine, and XGBoost, were performed using 620 SPT-based data from the 1999 Chi-Chi and Kocaeli earthquakes by Demir and Sahin (2023) to estimate liquefaction susceptibility. The study used pseudocode through the R software with the *"caret"* library package. The findings imply that all the generated models perform well in predicting soil liquefaction. Khatti et al. (2024) compared conventional and hybrid models in predicting liquefaction using CPT-based data. The results suggest that hybrid models outperform conventional ones, while the k-NN model performs over 90%.

1.3 Research Gap, Objectives, and Novelty of the Study

Although substantial research has focused on liquefaction prediction using machine learning, a thorough literature review reveals that no comparative study specifically addresses missing value imputation for liquefaction prediction with these techniques. Several previous studies, such as those conducted by Hu et al. (2015); Hu and Wang (2024) have eliminated the data containing missing values, which may lead to the loss of important information. Also, it can impede the further development of the models when combining data from different sources. In addition, some earlier studies developed machine learning models by using coding techniques (Demir and Sahin,2022*a*; 2022*b*; 2023; Zhao et al.,2024; Khatti et al.,2024), which may be difficult for non-expert users or practitioners with no or limited computer programming expertise. Therefore, developing the machine learning model using other relatively large liquefaction world case data and utilizing the missing value imputation technique to deal with incomplete data using a user-friendly method for non-expert users is necessary.

Considering the limitations of previous studies, this study aims to assess the reliability of machine learning in predicting soil liquefaction by utilizing a missing value imputation technique to address incomplete data. The research findings are expected to provide a new perspective for future studies in developing machine learning models to evaluate liquefaction phenomena by employing the missing value imputation technique, allowing larger datasets and a more diverse set of features or parameters, even with varying data completeness. Using a more user-friendly method, this study can hopefully provide new insight into developing machine learn-

ing models for practitioners or non-experts with no or limited experience in computer programming.

2 LIQUEFACTION HISTORICAL DATA

Proper data is essential for developing robust machine learning models that achieve high levels of accuracy. The liquefaction prediction model was developed using historical earthquake and liquefaction data. This study used 1116 SPT-based data compiled from various earthquakes around the world. The data utilized in this paper exceeds that of other previous studies, such as those conducted by Gandomi et al. (2013); Xue et al. (2017); Demir and Sahin (2023). The data used mainly originates from three previous studies, grouped as Dataset A, Dataset B, and Dataset C.

Dataset A consists of 288 liquefaction data from the Chi-chi earthquake, as compiled by Hwang and Yang (2001). The record contains 164 liquefaction and 124 non-liquefaction data from soil investigations conducted before and after the earthquake. Dataset B includes 620 sets of data, consisting of 256 liquefaction and 364 non-liquefaction data, compiled by Hanna et al. (2007). The dataset is a compilation of soil investigations studied at many sites in Taiwan and Turkey following the 1999 Chi-Chi and Kocaeli earthquakes that caused liquefaction in various areas. Dataset C contains historical earthquake and liquefaction data cases compiled by Cetin et al. (2018). It includes soil investigation data from major Japanese earthquakes, such as Niigata (1964), Tokachi-Oki (1968), Miyagiken-Oki (1978), Nihonkai-Chubu (1983), and Hyogoken-Nambu (1995). It also covers seismic events in the Americas, like the Argentina earthquake (1977), the Imperial Valley earthquake (1979), the Loma Prieta earthquake (1989), and others with moment magnitudes (M_w) ranging from 5.9 to 8.3. Approximately 208 data points (113 liquefaction and 95 non-liquefaction) were used out of an entire set of 210, since two were identified as marginal liquefaction. Data selection considered reliable sources, complete parameters, and liquefaction potential.

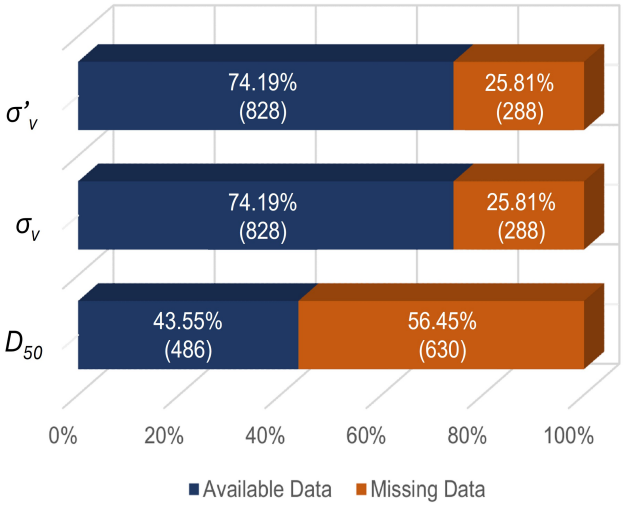

Figure 1 The distribution of missing data for each parameter.

Ten parameters were utilized in this study, including earthquake magnitude (M_w), peak ground acceleration (a_{max}), SPT blow number ($(N_1)_{60}$), fine content (FC), median grain size (D_{50}), depth of soil layer (z), groundwater table depth (d_w), total vertical stress (σ_v), effective vertical stress (σ_v'), and cyclic stress ratio (CSR). Among these, D_{50}, σ_v, and σ_v' contain missing values Figure 1. Descriptive statistical and correlation analyses were conducted to determine the structure, distribution, and relationships between the variables, following the approach of several previous studies, such as Kumar, Samui and Burman (2023); Kumar, Samui, Burman, Wipulanusat and Keawsawasvong (2023). The statistical

Table 1. Statistical descriptions of the dataset

Parameter	Notation	Mean	Median	Mode	Standard Deviation	Minimum	Maximum
Magnitude	M_w	7.44	7.60	7.60	0.30	5.90	8.30
Peak Horizontal Acceleration (g)	a_{max}	0.33	0.40	0.40	0.17	0.05	1.00
SPT blow number	$(N_1)_{60}$	14.46	11.39	7.00	10.72	0.93	75.00
Fine content (%)	FC	44.33	32.00	99.00	34.80	0.00	100.00
Median grain size (mm)	D_{50}	0.20	0.15	0.00	0.21	0.00	2.00
Depth of soil layer (m)	z	7.42	6.50	9.00	4.44	0.80	20.30
Groundwater Table Depth (m)	d_w	1.82	1.40	1.10	1.61	0.00	15.30
Total Vertical Stress (kPa)	σ_v	131.99	107.95	72.06	90.41	12.10	408.90
Effective Vertical Stress (kPa)	σ_v'	77.99	64.30	50.00	48.69	7.50	233.70
Cyclic Stress Ratio	CSR	0.31	0.30	0.44	0.16	0.04	0.82

Table 2. Correlation matrix of the parameters

	M_w	a_{max} (g)	$(N_1)_{60}$	FC (%)	D_{50} (mm)	z (m)	d_w (m)	σ_v (kPa)	σ_v' (kPa)	CSR
M_w	1.00	-0.02	-0.05	0.06	0.09	0.27	-0.03	0.27	0.22	0.03
a_{max} (g)	-0.02	1.00	0.24	0.12	-0.13	-0.05	0.03	0.05	0.09	0.90
$(N_1)_{60}$	-0.05	0.24	1.00	-0.39	-0.05	0.27	0.04	0.32	0.35	0.20
FC (%)	0.06	0.12	-0.39	1.00	-0.30	-0.09	-0.17	-0.07	-0.13	0.20
D_{50} (mm)	0.09	-0.13	-0.05	-0.30	1.00	-0.05	-0.07	-0.07	-0.12	-0.13
z (m)	0.27	-0.05	0.27	-0.09	-0.05	1.00	0.20	0.99	0.96	-0.07
d_w (m)	-0.03	0.03	0.04	-0.17	-0.07	0.20	1.00	0.05	0.24	-0.18
σ_v (kPa)	0.27	0.05	0.32	-0.07	-0.07	0.99	0.05	1.00	0.97	0.03
σ_v' (kPa)	0.22	0.09	0.35	-0.13	-0.12	0.96	0.24	0.97	1.00	0.00
CSR	0.03	0.90	0.20	0.20	-0.13	-0.07	-0.18	0.03	0.00	1.00

description of the dataset is described in Table 1. M_w and a_{max} are seismic parameters mostly used in liquefaction susceptibility assessment. Liquefaction susceptibility increases with an increase in M_w and a_{max}. The dataset contained a maximum peak acceleration of 1g and a minimum acceleration of 0.051g.

Saturated, loose sandy soils with low fine-grain contents are prone to liquefaction. However, sandy soils with a high fine-grain content may also liquefy under certain circumstances. Consequently, liquefaction potential evaluation depends extensively on $(N_1)_{60}$ and FC.

The descriptive statistical analysis of the FC and D_{50} results indicated that the historical liquefaction data used to develop the machine learning model represent liquefaction on clean sand and liquefaction that occurs in soil layers with high fine content. The dataset contains liquefaction in soil layers with SPT values less than 29. The sandy soil layer will undergo liquefaction when located below the groundwater table because it can increase the potential of excess pore water pressure. A study conducted by Zakariya et al. (2023) show a correlation between liquefaction potential and excess pore water pressure. In addition, the results also show that liquefaction commonly occurred in a shallow soil layer. Therefore, the soil layer depth and groundwater table depth are important in evaluating liquefaction vulnerability.

Table 2 displays the correlation matrix values between the used parameters. Red represents a positive correlation, and the color green represents a negative correlation. A higher intensity of color shows a stronger connection among the factors. The weak correlation between parameters indicates that each parameter has no linear relationship. The correlation between parameters is expressed in the R-value as the correlation coefficient. The R-value classification consists of (i) > 0.8, which shows a very strong correlation; (ii) 0.61 – 0.80 shows a strong correlation; (iii) 0.41 – 0.60 presents a moderate relationship; (iv) 0.21 – 0.40 represents a weak correlation; and (v) < 0.20 shows no correlation between the parameters (Khatti and Grover,2024*a*; 2024*b*; Samadi et al., 2024). Generally, the parameters used have no or weak correlation, which may indicate that the correlation between the parameters is not linear but may be non-linear. The non-linear have a strong correlation. For example, the correlation between total vertical stress and effective vertical stress with the depth of the soil layer, as well as the peak horizontal acceleration with the cyclic stress ratio. Based on Table 2, total vertical stress and effective vertical stress strongly correlate with the soil layer depth (R > 0.8). Theoretically, the effective vertical and total vertical stress values increase with soil layer depth. The peak ground acceleration parameter and cyclic stress ratio have a strong positive correlation (R = 0.9), whereby an increase in the peak ground acceleration value is correlated with an increase in the cyclic stress ratio value.

3 METHODS

The study began with collecting historical liquefaction data from various sources. The data was then prepared before being used to develop the model, train it with different techniques, and evaluate its liquefaction prediction performance. The processes of this study are represented in Figure 2. The model-building processes were conducted using RapidMiner Studio 10.3.001 software, provided with various extra extensions. The computer that used to run the models was powered by an 11th Gen Intel(R) Core™ i7-1165G7 @2.80GHz processor, 16 GB of RAM, and an Nvidia GeForce MX450 graphics card with 2 GB of VRAM running on Windows 10 Home 64-bit.

3.1 Preprocessing data

Preprocessing data is an early stage in machine learning that significantly impacts model performance. This stage involves cleaning, transforming, and preparing raw data for the effective construction of prediction models by machine learning algorithms. In general, preprocessing data can consist of multiple operations, such as missing value imputation, outlier identification, and normalization.

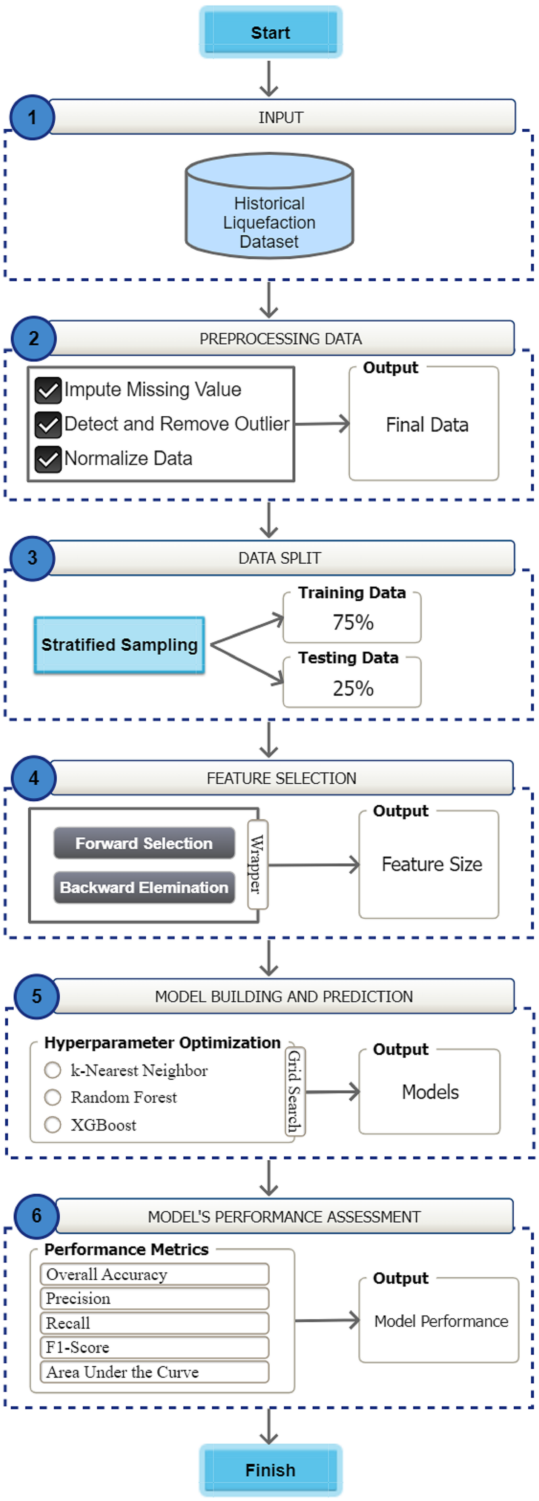

Figure 2 The process chart of the methodology used in this study.

3.1.1 Missing value imputation

Missing value imputation approaches can be grouped into statistical and machine learning-based techniques (Aittokallio, 2010; García-Laencina et al., 2010). Statistical approaches work by filling in missing values with the available data's mean, median, or mode for the same attribute. Machine learning approaches detect missing values using algorithms, including decision trees, neural networks, and k-NN. Statistical methods are often simpler and faster than machine learning methods, while machine learning methods tend to provide more accurate results but are more computationally intensive. One of the most popular imputation techniques is k-NN machine learning-based imputation, built on the notion that data with comparable features will be close together in the feature space (Lin and Tsai, 2020).

Several classification methods can perform better using the k-NN imputation approach (Pan et al., 2015). In the context of missing value imputation, k-NN looks for the k-nearest neighbors of a data point with a missing value and utilizes the values from these neighbors to replace the missing value. The number of the closest data affected the prediction of the missing value. Several trial-and-error experiments were done to obtain the ideal k value. The missing value imputation process used the *"Impute Missing Values"* operator in RapidMiner.

3.1.2 Outlier detection

Outlier identification is critical in data sets because it eliminates data that may affect model performance. It can be done using fundamental statistical-based methods, such as z-score and interquartile range, or classification methods employing algorithms, one of which is a distance-based outlier detection approach (Aggarwal, 2017; Mandhare and Idate, 2017). The output of outlier detection can be an outlier score and a binary label.

The k-NN concept is used for distance-based outlier detection. This method is suitable for high-dimensional data, offering low computational costs and good efficiency (Mandhare and Idate, 2017). The outlier detection and filtering process was done using the *"Detect Outlier (Distance)"* and *"Filter Examples"* operators in RapidMiner. The operator identifies n outliers in the given dataset based on the distance to their k nearest neighbors by applying several experimental trial-and-error methods to obtain the optimal k value.

3.1.3 Data normalization

The data collection process may lead to non-uniform value scales for each attribute. Therefore, data normalization is a necessary step in machine learning data preprocessing. Its objective is to rescale or range the input variables, resulting in a uniform or standardized distribution of the data. In the previous study, Kumar, Samui, Burman, Biswas and Vanapalli (2024); Kumar, Samui, Burman and Kumar (2024) used datasets with different scales and standardized them into a 0 – 1 range to minimize the dimensional effect of the input parameters. Therefore, in this study, all input and output variables were normalized between 0 and 1 during preprocessing to prevent issues with machine learning algorithms' learning rates. The standard data normalization formula used in the current study is shown in Equation (1).

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

X_{min} and X_{max} stand for the maximum and minimum values of each given feature, respectively, and X represents the value of the actual feature. X_{norm} stands for the value of the normalized feature.

3.2 Stratified Random Sampling

The sampling technique is a fundamental component of machine learning. It is carried out at the data processing stage before it is used to train and evaluate the model. Sampling techniques divide the dataset into training and testing data samples in a certain ratio, commonly known as the data split process. The two most common sampling techniques used in machine learning are Simple Random Sampling (SRS) and Stratified Random Sampling (StRS) (Demir and Sahin, 2022*b*).

StRS has various advantages over SRS despite being more complex due to the need for a data division process and requires accurate information on the proportions of each stratum. This technique can generate representative and proportional sampling results from each stratum, making it appropriate for use on datasets with class imbalance. Based on the results of a study conducted by Ye et al. (2013), StRS can reduce test error and boost *AUC* for high-dimensional data. Furthermore, because each stratum is represented well (Acharya et al., 2013), this technique can perform well on heterogeneous data and reduce bias. Due to the class imbalance in the data used and to ensure the uniform distribution of the dataset, the sampling process was done using the StRS technique through the *"Split Data"* operator in RapidMiner.

3.3 Feature Selection Method

Feature selection is an important stage that could simplify the model-building process. Although not every feature selection can enhance the model's performance, this technique might help the model perform better by considering several factors, including the data and method used (Theng and Bhoyar, 2024). This Feature selection might reduce the dataset's dimensionality to improve the model's performance by selecting the most relevant features or parameters to use in model development (Shi et al., 2021). Reducing the dimensions of the input data can decrease the complexity of the algorithm's modeling, which may reduce the time required for the model to execute. One of the widely used feature selection methods is the wrapper. The wrapper feature selection method is prone to overfitting (Dhal and Azad, 2022).

Forward Selection (FS) and Backward Elimination (BE) are the most common wrapper methods used in feature selection (Sahin and Demir, 2023). FS begins with an empty model and iteratively adds more variables to the model individually, depending on specific criteria. On the other hand, BE begins with a model containing all variables and proceeds to remove variables one by one that are considered unimportant according to certain criteria. The feature selection processes used the *"Forward Selection"* and *"Backward Elimination"* operators in RapidMiner software.

3.4 Hyperparameter Optimization

Hyperparameter optimization is one of the important processes in machine learning. Even though not every hyperparameter tuning can enhance the model performance, this procedure should help the model perform better when predicting new data by finding the best hyperparameter combination. Each algorithm has its hyperparameters that should be adjusted before training to achieve the best performance (Demir and Sahin, 2023). It is essential to avoid using the test data when tuning parameters (Probst et al., 2019). The most common method used in this process is grid search. In addition to grid search, there are several other methods, such as random search and Bayesian optimization.

Grid search is a systematic search technique that finds the optimal combination of hyperparameters by thoroughly searching over a predetermined range (Ranjan et al., 2019). Grid search is an exhaustive optimization method. This method performs hyperparameter tuning by training and evaluating the model for each specified combination of hyperparameters. Hence, this method incurs a high computational cost as compensation. In contrast, this method ensures that all hyperparameter combina-

tions are examined and analyzed, resulting in the best and most optimal results. The grid search hyperparameter selection processes used the *"Optimize Parameters (Grid)"* operator in RapidMiner, enabling parallel execution.

3.5 Machine Learning Classifiers

3.5.1 k-nearest neighbour (k-NN)

The k-NN algorithm is a fundamental approach for handling classification and regression issues in machine learning that belongs to the supervised learning category. The core premise of k-NN is to create predictions based on data that is most similar to the data to be predicted. Classification is performed by identifying an instance's label using the most frequent labels from its k-nearest neighbors as a reference. The number of nearest neighbors and distance used as references can be adjusted to meet those criteria and produce the best results.

The k-NN process involves two stages: selecting the nearest neighbor as a reference and determining the class based on this neighbor to identify a value or feature of the data (Cunningham and Delany, 2022). This algorithm is noise-resistant since it relies on most of the nearest neighbors and is known to handle non-linear data, making it more adaptable to various sorts of data. According to Manzali et al. (2024), the k-NN performance can be improved by assigning a greater weight to the significant characteristic, as it can be beneficial when performed using a high-dimensional dataset.

This study applies the k-NN algorithm through the "k-NN" operator in RapidMiner software. This operator provides four measure types: *mixed measures*, *nominal measures*, *numerical measures*, and *Bregman divergences*. Several parameters may be tuned to increase model performance, including the number of nearest neighbors (*k*), distance metrics (Euclidean, Manhattan, etc.), and weighting methods.

3.5.2 Random forest (RF)

RF algorithm is an ensemble learning approach capable of handling data complexity and variability. Similar to how several kinds of trees make up a forest, RF comprises several decision trees belonging to various data subsets. RF combines the concept of "bagging" (bootstrap aggregating) with decision trees, which are groups of randomly generated decision trees. It was established by Breiman (2001) and has become one of the most used machine learning algorithms due to its excellent classification and regression capabilities.

According to various studies, RF has several benefits over similar machine learning methods, including the capacity to avoid overfitting, producing consistently excellent performance across datasets, and being insensitive to outliers (Cutler et al., 2007; Genuer et al., 2010; Roy and Larocque, 2012). Besides, since multiple decision trees are made, RF tends to be less sensitive to high variability and heterogeneous data. According to Gregorutti et al. (2017), RF performs well even with high-dimensional data and can handle multi-class output even when using imbalanced data. Some previous studies show that RF generally performs well in predicting liquefaction (Demir and Sahin, 2022*a*,*b*). Therefore, RF is utilized in this work and will be compared with other ensemble and basic algorithms. In this study, the default configuration of the RF algorithm used was the *"majority vote"* with the *"information_gain"* criterion. RF capability can be improved by optimizing several hyperparameters, such as *the number of trees*, *mtry*, *sample size*, *node size*, and *max depth* (Probst et al., 2019; Demir and Sahin, 2022*b*).

3.5.3 eXtreme gradient boosting (XGBoost)

XGBoost has recently emerged as one of the most powerful and widely used machine learning methods. This algorithm, which uses a decision tree as the basis model, was introduced by Chen and Guestrin (2016). It implements the Gradient Boosting algorithm to increase speed, performance, and accuracy. Until then, XGBoost was one of the most precise and accurate decision tree-based algorithms for handling classification and regression issues. The basic concept of XGBoost's learning method is ensemble learning, which combines several weak learners to build a more powerful learner. XGBoost applied a gradient boosting approach, where each subsequent model focuses on reducing the previous model's prediction error.

XGBoost's performance has been evaluated in multiple studies utilizing a variety of datasets, including public, medical, and geotechnical datasets, with great results Can et al. (2021); Paleczek et al. (2021); Zhang et al. (2022). This method can perform effectively with imbalanced data by adjusting the class weights. XGBoost provides many adjustable hyperparameters that users can customize to increase model performance and improve accuracy. Various prior studies have done the research and discovered various hyperparameters that have significant effects on XGBoost performance, such as *learning rate*, *subsample*, *max_leave*, *max_depth*, *min_child_weight*, and *number of rounds* (Wang and Sherry Ni, 2019; Demir and Sahin, 2023).

3.6 Performance Evaluation Criteria

The performance of a model must be assessed to determine its ability to make predictions. Machine learning can be evaluated using various approaches. This study employs the Confusion Matrix (CM) approach to evaluate model performance. The CM is a table presented in matrix form, depicting the model's performance by comparing its prediction results to the actual values of the test data. It includes four parameters: true positive (TP), true negative (TN), false positive (FP), and false negative (FN) Figure 3.

Confusion Matrix		Actual	
		Liquefaction (Yes)	Non-Liquefaction (No)
Predicted	Liquefaction (Yes)	TP	FP
	Non-Liquefaction (No)	FN	TN

Figure 3 Confusion matrix illustration.

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Prec = \frac{TP}{TP + FP} \tag{3}$$

$$Rec = \frac{TP}{TP + FN} \tag{4}$$

$$F1 = 2 \cdot \frac{Prec \cdot Rec}{Prec + Rec} \tag{5}$$

The model performance matrix can be calculated based on the CM parameters, which include Overall Accuracy (*OA*), Precision (*Prec*), Recall (*Rec*), and F1-score (*F1*). *OA* indicates the rate at which the model correctly predicts the class Equation (2). At the same time, *Prec* represents the proportion of correct positive predictions among all positive predictions Equation (3). *Rec* determines how often the model correctly identifies the correct class out of any instance that should be considered positive Equation (4). *F1* is the harmonic mean of *Prec* and *Rec*, providing a balanced assessment of the model's performance Equation (5).

Receiver Operating Characteristic (ROC) and Area Under the Curve (*AUC*) are curve-based tools to measure model performance. Both are commonly used metrics in binary classification. The ROC curve shows how well the model performs at different classification thresholds. The curve is created by plotting the true positive rate (TPR) on the y-axis against the false positive rate (FPR) on the x-axis. TPR, also known as Recall or sensitivity, can be calculated using Equation (4), while FPR can be calculated using Equation (6). The *AUC* value can be calculated from the total area under the ROC curve (Gorunescu, 2011). Its value will always be between the range of 0.0 and 1.0. As the *AUC* value gets closer to 1.0, it indicates a better classification performance of the model; conversely, it indicates a worse performance when it gets closer to 0.0.

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

4 RESULTS AND DISCUSSION

4.1 Preprocessing Data

The data preparation stage is the initial and most important stage in developing a machine learning model. This process includes identifying and managing missing data, detecting and eliminating outliers, and normalizing the data before splitting the dataset into training and testing data.

The first step is to detect and manage any missing values in the liquefaction dataset. Missing data values were identified for the parameters median grain size (D_{50}), total vertical stress (σ_v), and effective vertical stress (σ_v'). These missing values were addressed using the *"impute missing values"* operator in the RapidMiner application, which employs an algorithmic approach to estimate missing values. Specifically, the k-NN method was used to estimate and fill in the missing values for these parameters. The number of neighbors was selected by applying several experiments of trial-and-error methods using numbers ranging from 5 to 30 in multiples of 5, considering the size of the dataset. Some other missing value imputation techniques were considered. These included replacing the missing data with minimum, maximum, and mean values. However, the k-NN-based method still outperformed the others. Once the missing data was processed, the *"detect outliers (distances)"* operator was used to identify outliers using the k-NN approach. The identified outliers were then removed from the modeling process. Next, the data was normalized to ensure that large-scale values did not dominate the model development process.

Finally, the processed dataset was split into training and testing data. Some previous studies showed that the data ratio affects the performance of the developed model (Pham et al., 2020; Nguyen et al., 2021). A study conducted by Pham et al. (2020) showed an increase in model performance as the amount of training data increased from 30% to 80%, while the model began to decrease in performance when using 80% to 90% of training data. Therefore,

considering those findings, several trial-and-error experiments were applied, and 75:25 was obtained as the optimum data ratio. It means that 75% of the total processed data was used to train the model, while the remaining 25% was used to test it.

4.2 Feature Selection Using the Wrapper Method

The feature selection process aims to choose the most relevant and significant subset of features from the provided dataset. By reducing the amount of data, feature selection is attempted to enhance the model's predictive accuracy. This study utilizes a feature selection approach on a dataset with ten features (M_w, a_{max}, $(N_1)_{60}$, FC, D_{50}, z, d_w, σ_v, σ_v', and CSR). FS and BE were employed to identify the most significant and relevant features to be included in the model's construction. Both methods were implemented using the RF algorithm and k-fold cross-validation.

The k-fold cross-validation approach involves splitting the dataset into k subsets. Each subset is used as training data k-1 times and testing data once. This process is repeated k times, ensuring each subset serves as test data once. The average assessment outcomes from each iteration estimate the model's overall performance. This approach helps reduce overfitting and delivers consistent prediction performance. This study utilized a k-value of 5, meaning the dataset was uniformly divided into five subsets, alternately used as testing and training data.

Table 3. Selected parameters based on the result of feature selection

Parameters	Feature Selection Method		
	FS	BE	RAW
M_w	✗	✗	✓
a_{max}	✓	✓	✓
$(N_1)_{60}$	✓	✓	✓
FC	✓	✓	✓
D_{50}	✗	✓	✓
z	✓	✓	✓
d_w	✗	✓	✓
σ_v	✗	✓	✓
σ_v'	✓	✓	✓
CSR	✓	✓	✓

**FS* : forward selection; *B*E : back elimination; *R*AW : standard algorithm without feature selection.

Table 3 displays the result of the feature selection method employing BE and FS. The parameters a_{max}, $(N_1)_{60}$, FC, z, σ_v', and CSR were identified as having the most important role to be utilized in developing the liquefaction prediction model based on the results of running feature selection. In contrast, M_w is regarded as the least significant among the ten parameters, which runs contrary to the findings of the study carried out by Hu (2021) and Hu et al. (2021). The FS method selected six different parameters (a_{max}, $(N_1)_{60}$, FC,z, σ_v', and CSR), while the BE method only eliminated the parameter M_w from being used in building the model. Nonetheless, the feature selection results are generally consistent with the results of previous studies. Hu (2021) suggests that a_{max}, FC, d_w, and σ_v' are relatively significant factors; other than that, Hu et al. (2021) imply that a_{max}, FC, D_{50}, d_w, z, are the key factors, and σ_v' is relatively important. Furthermore, the model will be constructed with all parameters (RAW) to compare the implications of the feature selection approach.

4.3 Hyperparameter Optimization Using Grid Search

This study utilizes the *"optimize parameters (grid)"* operator to optimize hyperparameters using 5-fold cross-validation. The hyperparameter combinations optimized in the RF algorithm include *"number_of_trees"*, *"maximal_depth"*, and *"criterion"*. The XGBoost algorithm focuses on optimizing the *"max_depth"*, *"min_child_weight"*, and *"subsample"* while the k-NN algorithm optimizes the hyperparameters k-value, *"measure_type"*, *"weighted_vote"*, and *"numerical_measure"*. The hyperparameter optimization results are summarized in Table 4. Based on the evaluation of these results, the XGBoost_BE model's hyperparameter combination achieved the highest *OA* score, while the k-NN_BE model had the lowest score.

4.4 Liquefaction Prediction and Model's Performance Evaluation

RF, XGBoost, and k-NN algorithms predict liquefaction based on historical earthquake and liquefaction datasets. The performance of each model is evaluated using three schemes: standard without feature selection (RAW), and utilizing feature selection (FS and BE). According to previous studies (Demir and Sahin, 2022*b*, 2023; Hu and Wang, 2024), the five and 10-fold cross-validation were frequently used in liquefaction prediction. Therefore, after experimenting with both 5 and 10 k values, it was determined that the 5-fold cross-validation yielded better performance and lower computational cost. This approach was then utilized to assess the model's performance. The model's performance was measured by assessing each model's *OA*, *Prec*, *Rec*, and *F1* values. The summary performance measurement results are shown in Table 5.

The confusion matrix evaluation results show that the model can predict liquefaction using the missing value imputation technique. In the FS scheme,

Table 4. Result of the best hyperparameter combination

Feature selection method	Algorithm	*OA*	*Prec*	*Rec*	*F1*	Best hyperparameters
FS	RF	85.00%	84.29%	85.35%	84.82%	'number_of_trees': 61, 'maximal_depth': 8, 'criterion': information_gain
	XGBoost	85.50%	85.86%	84.34%	85.10%	'max_depth': 3, 'min_child_weight': 0.2, 'subsample': 0.45
	k-NN	82.28%	81.08%	83.33%	82.21%	'k': 8, 'measure_type': NumericalMeasures, 'weighted_vote': true, 'numerical_measure': ManhattanDistance
BE	RF	85.00%	84.29%	85.35%	84.82%	'number_of_trees': 160, 'maximal_depth': 18, 'criterion': information_gain
	XGBoost	85.76%	85.04%	86.11%	85.58%	'max_depth': 3, 'min_child_weight': 1.0, 'subsample': 0.8
	k-NN	81.04%	78.45%	84.60%	81.53%	'k': 10, 'measure_type': NumericalMeasures, 'weighted_vote': true, 'numerical_measure': EuclideanDistance
RAW	RF	85.62%	84.15%	87.12%	85.64%	'number_of_trees': 41, 'maximal_depth': 24, 'criterion': information_gain
	XGBoost	85.62%	85.00%	85.86%	85.43%	'max_depth': 4, 'min_child_weight': 1.0, 'subsample': 0.475
	k-NN	81.91%	81.41%	81.82%	81.62%	'k': 4, 'measure_type': NumericalMeasures, 'weighted_vote': true, 'numerical_measure': EuclideanDistance

the XGBoost model has the highest *OA* value (89.59%) compared to the other algorithms, indicating that the model can predict liquefaction with just six parameters. However, in the BE and RAW schemes, RF achieved the highest scores for *OA* (90.33% and 90.71%, respectively). The RF model generally has the best *OA* score in almost all schemes, with the RF_RAW model having the highest *OA* (Figure 4).

Overall, the k-NN model without feature selection has the lowest *OA* performance of any model, with a score of 81.04%. Regarding *OA*, feature selection improves the performance of the k-NN and, under certain conditions, can improve XGBoost performance. The RF algorithm typically has a desirable *OA* value without feature selection. This result showed that feature selection does not always improve model performance. In certain cases, removing seemingly insignificant features may lead to losing important information. Therefore, it is highly recommended that a thorough model performance evaluation be conducted after the feature selection is applied. The findings suggest that the missing values could still be handled properly since the model's performance utilizing missing value imputation is still reasonably comparable to the study conducted by Demir and Sahin (2022*b*), especially the RF model. The RF model developed by Demir and Sahin (2022*b*) achieved 90.54% accuracy using stratified random sampling and 93.24% accuracy using SMOTE. The performance differences stay within a reasonably acceptable range due to the difference in the use of data, software, and some methods, such as missing value imputation.

Table 5. Result of the model's performance evaluation

Algorithm	Performance matrix	Feature Selection Method		
		FS	BE	RAW
RF	*OA*	89.22%	90.33%	**90.71%**
	Prec	89.92%	**91.41%**	90.23%
	Rec	87.88%	88.64%	**90.91%**
	F1	88.89%	90.00%	**90.57%**
	AUC	0.947	0.952	0.943
XGBoost	*OA*	89.59%	86.62%	88.48%
	Prec	91.27%	87.50%	89.76%
	Rec	87.12%	84.85%	86.36%
	F1	89.15%	86.15%	88.03%
	AUC	**0.959**	0.955	0.957
k-NN	*OA*	87.36%	86.99%	81.04%
	Prec	87.12%	84.89%	80.45%
	Rec	87.12%	89.39%	81.06%
	F1	87.12%	87.08%	80.75%
	AUC	0.922	0.925	0.892

*The best results are shown in bold.

Based on the feature selection result, the FS and BE methods excluded the M_w parameter, and even the FS method excluded D_{50}. The correlation analysis result shows that these two parameters do not significantly correlate with each other, which may be contrary to the technical perspective. From a technical standpoint, excluding highly correlated parameters can be advantageous for various reasons, such as avoiding redundant information and multicollinearity in models. However, attention to various aspects and conditions, such as domain knowledge and model performance, is also appropriate. Apart from that, the exclusion should also be justified by evaluating the model's performance. Based on the performance analysis, the RF models tend to achieve the highest performance when using all the parameters. This might suggest that the RF model is less sensitive to correlated parameters. This study's findings imply that the RF algorithm has the best overall accuracy value without feature selection. This result may indicate that feature selection does not always improve model performance. In certain cases, removing seemingly insignificant features may lead to losing important information. Therefore, it is recommended that a comprehensive evaluation of model performance be conducted after applying feature selection. It should also be noted that several factors affect the impact of the feature selection on model performance, including the data used (Theng and Bhoyar, 2024).

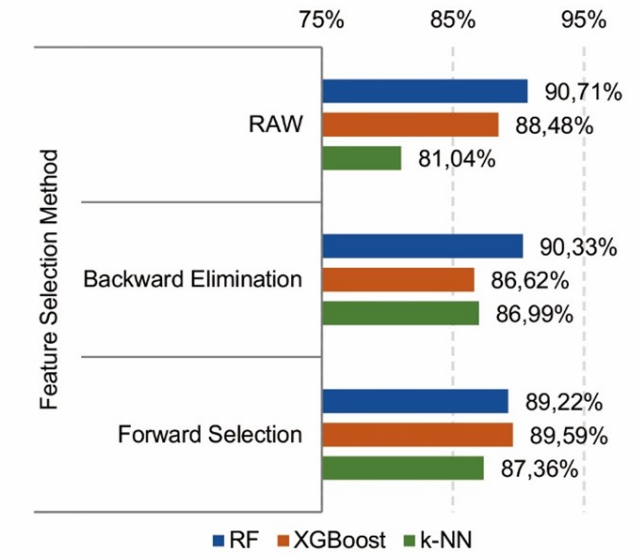

Figure 4 Overall accuracy performance of the models.

Precision is the ratio of true positive predictions to the overall data that is predicted to be positive. The higher the precision value, the fewer instances of *"non-liquefaction"* are classified as *"liquefaction"* by the model. The RF_BE model attained the highest price value of 91.41%, while the k-NN_RAW model acquired the lowest *Prec* value of 80.45% (Figure 5). Based on the result, the *Prec* performance of XGBoost and k-NN increased when using the FS method, while the BE method increased RF's *Prec* performance.

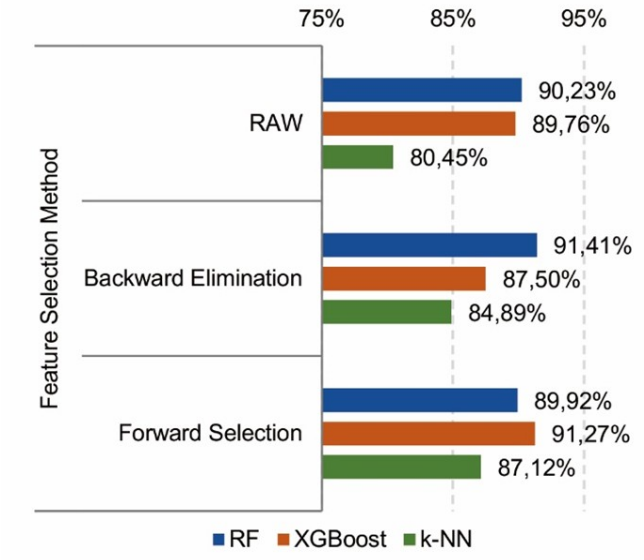

Figure 5 Precision performance of the models.

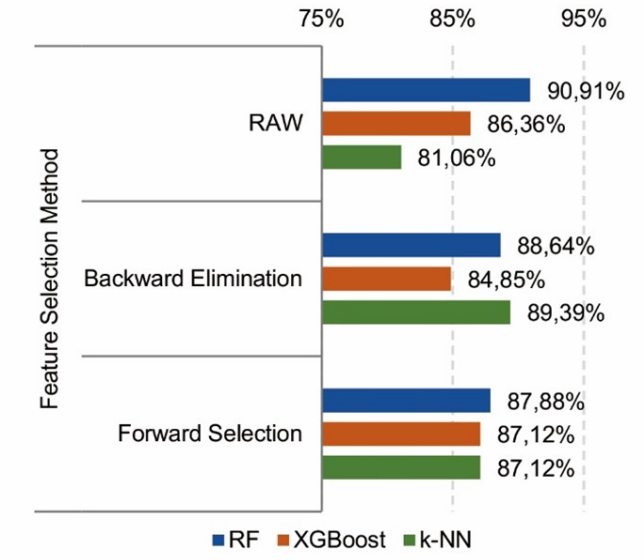

Figure 6 Recall performance of the models.

In the case of estimating liquefaction susceptibility, a false negative condition is when the model identifies data as *"non-liquefaction"* in actual conditions that should be *"liquefaction"*, which is more dangerous than a false positive condition. Model errors in predicting conditions that should be liquefaction but are identified as non-liquefaction can be fatal in planning and construction. Thus, in the liquefaction context, it is generally preferable to minimize false negative conditions, making recall a good choice to utilize as a benchmark when evaluating model performance.

The recall is the ratio of the true positive value to the total of the true positive and false negative values; therefore, the lower the false negative value, the higher the *Rec* value; in other words, the lower the proportion of liquefied conditions that the model fails to detect. In the current study, the RF_RAW model achieved a *Rec* score of 90.91%, the highest among the other models (Figure 6). The RF_RAW model can accurately identify up to 120 out of the 132 actual liquefaction conditions that were evaluated (Figure 7). There is an imbalance in the amount of *"liquefaction"* class and *"non-liquefaction"* class data used in this study, with more *"non-liquefaction"* data than *"liquefaction"* data. As a result, there may be an imbalance in the model's ability to forecast the data, with the model being stronger at predicting the *"non-liquefaction"* condition than the *"liquefaction"* condition.

A performance matrix called the F1-Score can be used to evaluate how well models perform when given unbalanced data. The *F1* is a performance metric that combines precision and recall into a single value; in other words, it is the harmonic average of *Prec* and *Rec*. The test results showed that the RF_RAW model, with a total of 90.57%, had the greatest *F1* value (Figure 8). The RF_RAW model can

Confusion Matrix		Actual	
		Liquefaction (Yes)	Non-Liquefaction (No)
Predicted	Liquefaction (Yes)	120	13
	Non-Liquefaction (No)	12	124

Figure 7 Confusion matrix of RF_RAW model.

predict liquefied and non-liquefied conditions in a generally balanced way, with results in the form of FP and FN values that are typically balanced in percentages with actual conditions. Figure 9 shows the *AUC* value of the models. The RF and k-NN models achieved the highest *AUC* values of 0.952 and 0.925, respectively, while using backward elimination. On the other hand, XGBoost obtained the highest *AUC* value (0.959) using the forward selection scheme.

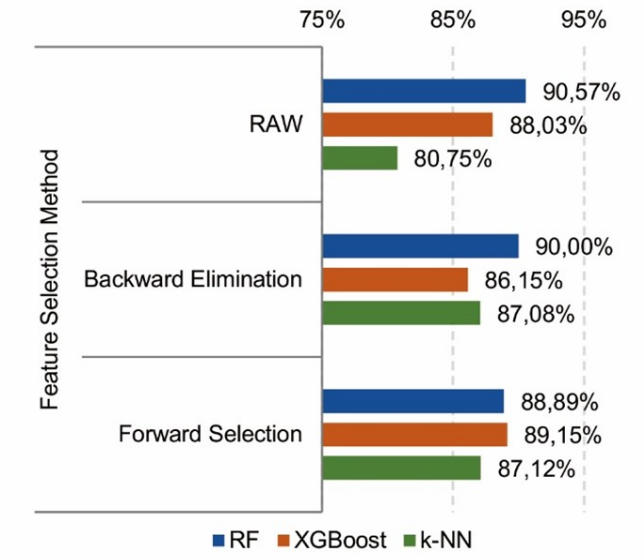

Figure 8 F1-Score performance of the models.

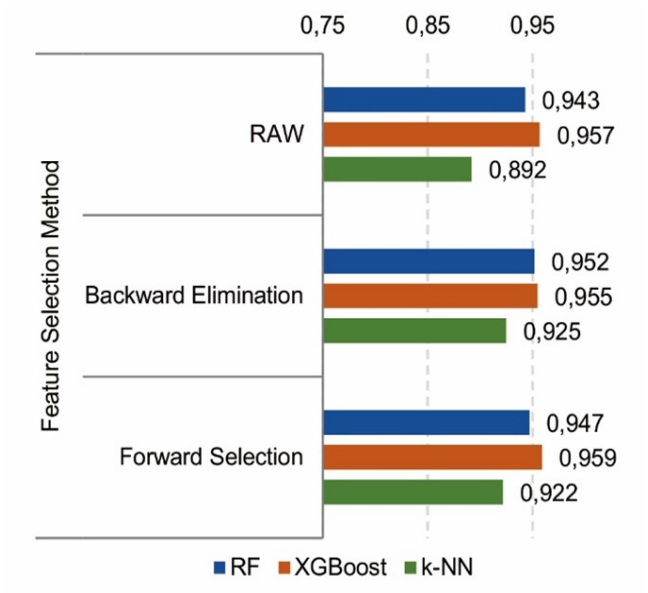

Figure 9 *AUC* score of the models.

5 LIMITATIONS AND FUTURE WORKS

Although the findings indicate that the missing value imputation technique could handle missing data properly, this study still has some limitations that need further investigation. This study was done using only an SPT-based dataset. Therefore, further research using other data types, such as V_S and CPT-based data, is necessary to investigate the model's performance using the missing value imputation technique. The k-NN model seems more sensitive to highly correlated features; hence, exploring other algorithms besides using other hyperparameter techniques is necessary. Future studies should incorporate a larger variety of liquefaction data, and another method should be used to assess the model's performance in more detail. A more diverse dataset may provide new, important information, but could impact the model's generalizability. Therefore, further investigation is needed by applying other feature selection techniques and hyperparameter optimization techniques in combination with other robust ensemble algorithms to anticipate this possibility. In addition, it is important to explore various missing value imputation techniques to identify the most suitable technique for the liquefaction dataset.

6 CONCLUSIONS

Identifying soil liquefaction susceptibility is critical for managing seismic disaster risks. Various methods remain to be developed to identify liquefaction vulnerability and reduce hazards accurately. One extensively studied method for liquefaction prediction is machine learning. In this study, liquefaction prediction was conducted using a larger amount of liquefaction historical data, with a missing value imputation technique to handle missing data, allowing greater variety in the dataset. To identify the best-performing model, three algorithms, namely k-NN, RF, and XGBoost, were evaluated using different feature selection and parameter optimization techniques.

The overall results indicate that all models are still effective in predicting liquefaction, especially the RF model. The RF_RAW model achieved the best performance (*OA* = 90.71%) and is still reasonably comparable to the previous study. It may suggest that missing value imputation using the nearest-neighbor approach could still handle missing data properly. In general, the RF algorithm outperformed nearly every modeling scheme tested. The RF model performed best when incorporating all data considered, while feature selection generally improved performance for the XGBoost and k-NN models. It may indicate that RF is less sensitive to correlated data. In the context of liquefaction prediction, *Rec* is an important metric because reducing false negatives helps prevent larger losses due to inaccuracies in building design and liquefaction mitigation planning. Additionally, the *AUC* results demonstrate that the models deliver excellent classification performance.

Finally, despite the differences in accuracy reported from previous studies, this study hopefully can provide a beneficial perspective for further research into managing missing data using imputation techniques to assess liquefaction vulnerability. This approach allows for the combination of data from many data sources to accommodate other significant information that may be missed. In addition, this study is expected to provide a new perspective for future studies in developing machine learning models to evaluate liquefaction phenomena by using a more user-friendly method that is more usable for non-expert users with no or limited computer programming experience.

DISCLAIMER

The authors declare no conflict of interest.

ACKNOWLEDGMENTS

The authors express their sincere gratitude to the Ministry of Public Works for funding this study.

REFERENCES

Acharya, A., Prakash, A., Saxena, P. and Nigam, A. (2013), 'Sampling: why and how of it?', *Indian Journal of Medical Specialities* **4**(2).
URL: *https://doi.org/10.7713/ijms.2013.0032*

Aggarwal, C. (2017), *Outlier Analysis*, Springer International Publishing, Cham.
URL: *https://doi.org/10.1007/978-3-319-47578-3*

Aittokallio, T. (2010), 'Dealing with missing values in large-scale studies: microarray data imputation and beyond', *Briefings in Bioinformatics* **11**(2), 253–264.
URL: *https://doi.org/10.1093/bib/bbp059*

Boulanger, R. and Idriss, I. (2014), Cpt and spt based liquefaction triggering procedures, Technical Report Report No. UCD/CGM-14/01, Center for Geotechnical Modeling, University of California, Davis.

Breiman, L. (2001), 'Random forests', *Machine Learning* **45**(1), 5–32.
URL: *https://doi.org/10.1023/A:1010933404324*

Can, R., Kocaman, S. and Gokceoglu, C. (2021), 'A comprehensive assessment of xgboost algorithm for landslide susceptibility mapping in the upper basin of ataturk dam, turkey', *Applied Sciences* **11**(11), 4993.
URL: *https://doi.org/10.3390/app11114993*

Cetin, K., Seed, R., Kayen, R., Moss, R., Bilge, H., Ilgac, M. and Chowdhury, K. (2018), 'Dataset on spt-based seismic soil liquefaction', *Data in Brief* **20**, 544–548.
URL: *https://doi.org/10.1016/j.dib.2018.08.043*

Chen, T. and Guestrin, C. (2016), Xgboost: A scalable tree boosting system, *in* 'Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)', ACM, San Francisco, California, USA, pp. 785–794.
URL: *https://doi.org/10.1145/2939672.2939785*

Cunningham, P. and Delany, S. (2022), 'k-nearest neighbour classifiers', *ACM Computing Surveys* **54**(6), 1–25.
URL: *https://doi.org/10.1145/3459665*

Cutler, D. R., Edwards, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J. and Lawler, J. J. (2007), 'Random forests for classification in ecology', *Ecology* **88**(11), 2783–2792.
URL: *https://doi.org/10.1890/07-0539.1*

Demir, S. and Sahin, E. K. (2022*a*), '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* **154**, 107130.
URL: *https://doi.org/10.1016/j.soildyn.2021.107130*

Demir, S. and Sahin, E. K. (2022*b*), 'Liquefaction prediction with robust machine learning algorithms (svm, rf, and xgboost) supported by genetic algorithm-based feature selection and parameter optimization from the perspective of data processing', *Environmental Earth Sciences* **81**(18), 459.
URL: *https://doi.org/10.1007/s12665-022-10578-4*

Demir, S. and Sahin, E. K. (2023), 'An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using adaboost, gradient boosting, and xgboost', *Neural Computing and Applications* **35**(4), 3173–3190.
URL: *https://doi.org/10.1007/s00521-022-07856-4*

Dhal, P. and Azad, C. (2022), 'A comprehensive survey on feature selection in the various fields of machine learning', *Applied Intelligence* **52**(4), 4543–4581.
URL: *https://doi.org/10.1007/s10489-021-02550-9*

Galupino, J. and Dungca, J. (2022), 'Machine learning models to generate a subsurface soil profile: A case of makati city, philippines', *International Journal of GEOMATE* **23**(95).
URL: *https://doi.org/10.21660/2022.95.3372*

Gandomi, A. H., Fridline, M. M. and Roke, D. A. (2013), 'Decision tree approach for soil liquefaction assessment', *The Scientific World Journal* **2013**, 1–8.
URL: *https://doi.org/10.1155/2013/346285*

García-Laencina, P. J., Sancho-Gómez, J.-L. and Figueiras-Vidal, A. R. (2010), 'Pattern classification with missing data: a review', *Neural Computing and Applications* **19**(2), 263–282.
URL: *https://doi.org/10.1007/s00521-009-0295-6*

Genuer, R., Poggi, J.-M. and Tuleau-Malot, C. (2010), 'Variable selection using random forests', *Pattern Recognition Letters* **31**(14), 2225–2236.
URL: *https://doi.org/10.1016/j.patrec.2010.03.014*

Gorunescu, F. (2011), *Data Mining*, Intelligent Systems Reference Library, Springer Berlin Heidelberg, Berlin, Heidelberg.
URL: *https://doi.org/10.1007/978-3-642-19721-5*

Gregorutti, B., Michel, B. and Saint-Pierre, P. (2017), 'Correlation and variable importance in random forests', *Statistics and Computing* **27**(3), 659–678.
URL: *https://doi.org/10.1007/s11222-016-9646-1*

Hanna, A. M., Ural, D. and Saygili, G. (2007), 'Neural network model for liquefaction potential in soil deposits using turkey and taiwan earthquake data', *Soil Dynamics and Earthquake Engineering* **27**(6), 521–540.
URL: *https://doi.org/10.1016/j.soildyn.2006.11.001*

Hu, J. (2021), 'Data cleaning and feature selection for gravelly soil liquefaction', *Soil Dynamics and Earthquake Engineering* **145**, 106711.
URL: *https://doi.org/10.1016/j.soildyn.2021.106711*

Hu, J.-L., Tang, X.-W. and Qiu, J.-N. (2015), 'A bayesian network approach for predicting seismic liquefaction based on interpretive structural modeling', *Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards* **9**(3), 200–217.
URL: *https://doi.org/10.1080/17499518.2015.1076570*

Hu, J., Tan, Y. and Zou, W. (2021), 'Key factors influencing earthquake-induced liquefaction and their direct and mediation effects', *PLOS ONE* **16**(2), e0246387.
URL: *https://doi.org/10.1371/journal.pone.0246387*

Hu, J. and Wang, J. (2024), 'A data extension framework of seismic-induced gravelly soil liquefaction based on semi-supervised methods', *Advanced Engineering Informatics* **59**, 102295.
URL: *https://doi.org/10.1016/j.aei.2023.102295*

Hwang, J.-H. and Yang, C.-W. (2001), 'Verification of critical cyclic strength curve by taiwan chi-chi earthquake data', *Soil Dynamics and Earthquake Engineering* **21**(3), 237–257.
URL: *https://doi.org/10.1016/S0267-7261(01)00002-1*

Idriss, I. M. and Boulanger, R. W. (2008), *Soil Liquefaction During Earthquakes*, Earthquake Engineering Research Institute (EERI).

Khatti, J., Fissha, Y., Grover, K. S., Ikeda, H., Toriya, H., Adachi, T. and Kawamura, Y. (2024), 'Cone penetration test-based assessment of liquefaction potential using machine and hybrid learning approaches', *Multiscale and Multidisciplinary Modeling, Experiments and Design* **7**(4), 3841–3864.
URL: *https://doi.org/10.1007/s41939-024-00447-x*

Khatti, J. and Grover, K. S. (2024*a*), 'Assessment of uniaxial strength of rocks: A critical comparison between evolutionary and swarm optimized relevance vector machine models', *Transportation Infrastructure Geotechnology* .
URL: *https://doi.org/10.1007/s40515-024-00433-3*

Khatti, J. and Grover, K. S. (2024*b*), 'Prediction of uniaxial strength of rocks using relevance vector machine improved with dual kernels and metaheuristic algorithms', *Rock Mechanics and Rock Engineering* **57**(8), 6227–6258.
URL: *https://doi.org/10.1007/s00603-024-03849-y*

Kumar, D. R., Samui, P. and Burman, A. (2022), 'Prediction of probability of liquefaction using soft computing techniques', *Journal of The Institution of Engineers (India): Series A* **103**(4), 1195–1208.
URL: *https://doi.org/10.1007/s40030-022-00683-9*

Kumar, D. R., Samui, P. and Burman, A. (2023), 'Suitability assessment of the best liquefaction analysis procedure based on spt data', *Multiscale and Multidisciplinary Modeling, Experiments and Design* **6**(2), 319–329.
URL: *https://doi.org/10.1007/s41939-023-00148-x*

Kumar, D. R., Samui, P., Burman, A., Biswas, R. and Vanapalli, S. (2024), 'A novel approach for assessment of seismic induced liquefaction susceptibility of soil', *Journal of Earth System Science* **133**(3), 128.
URL: *https://doi.org/10.1007/s12040-024-02341-z*

Kumar, D. R., Samui, P., Burman, A. and Kumar, S. (2024), 'Seismically induced liquefaction potential assessment by different artificial intelligence procedures', *Transportation Infrastructure Geotechnology* **11**(3), 1272–1293.
URL: *https://doi.org/10.1007/s40515-023-00327-w*

Kumar, D. R., Samui, P., Burman, A., Wipulanusat, W. and Keawsawasvong, S. (2023), 'Liquefaction susceptibility using machine learning based on spt data', *Intelligent Systems with Applications* **20**, 200281.
URL: *https://doi.org/10.1016/j.iswa.2023.200281*

Lin, W.-C. and Tsai, C.-F. (2020), 'Missing value imputation: a review and analysis of the literature (2006–2017)', *Artificial Intelligence Review* **53**(2), 1487–1509.
URL: *https://doi.org/10.1007/s10462-019-09709-4*

Mandhare, H. C. and Idate, S. R. (2017), A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques, *in* '2017 International Conference on Intelligent Computing and Control Systems (ICICCS)', IEEE, Madurai, pp. 931–935.
URL: *https://doi.org/10.1109/ICCONS.2017.8250601*

Manzali, Y., Barry, K., Flouchi, R., Balouki, Y. and Elfar, M. (2024), 'A feature weighted k-nearest neighbor algorithm based on association rules', *Journal of Ambient Intelligence and Humanized Computing* **15**, 1–14.
URL: *https://doi.org/10.1007/s12652-024-04793-z*

Nguyen, Q. H., Ly, H.-B., Ho, L. S., Al-Ansari, N., Le, H. V., Tran, V. Q., Prakash, I. and Pham, B. T. (2021), 'Influence of data splitting on performance of machine learning models in prediction of shear strength of soil', *Mathematical Problems in Engineering* **2021**(1), 4832864.
URL: *https://doi.org/10.1155/2021/4832864*

Paleczek, A., Grochala, D. and Rydosz, A. (2021), 'Artificial breath classification using xgboost algorithm for diabetes detection', *Sensors* **21**(12), 4187.
URL: *https://doi.org/10.3390/s21124187*

Pan, R., Yang, T., Cao, J., Lu, K. and Zhang, Z. (2015), 'Missing data imputation by k nearest neighbours based on grey relational structure and mutual information', *Applied Intelligence* **43**(3), 614–632.
URL: *https://doi.org/10.1007/s10489-015-0666-x*

Pham, B. T., Qi, C., Ho, L. S., Nguyen-Thoi, T., Al-Ansari, N., Nguyen, M. D., Nguyen, H. D., Ly, H.-B., Le, H. V. and Prakash, I. (2020), 'A novel hybrid soft computing model using random forest and particle swarm optimization for estimation of undrained shear strength of soil', *Sustainability* **12**(6), 2218.
URL: *https://doi.org/10.3390/su12062218*

Probst, P., Wright, M. N. and Boulesteix, A. (2019), 'Hyperparameters and tuning strategies for random forest', *WIREs Data Mining and Knowledge Discovery* **9**(3), e1301.
URL: *https://doi.org/10.1002/widm.1301*

Puri, N., Prasad, H. D. and Jain, A. (2018), 'Prediction of geotechnical parameters using machine learning techniques', *Procedia Computer Science* **125**, 509–517.
URL: *https://doi.org/10.1016/j.procs.2017.12.066*

Ranjan, G. S. K., Kumar Verma, A. and Radhika, S. (2019), K-nearest neighbors and grid search cv based real time fault monitoring system for industries, *in* '2019 IEEE 5th International Conference for Convergence in Technology (I2CT)', IEEE, Bombay, India, pp. 1–5.
URL: *https://doi.org/10.1109/I2CT45611.2019.9033691*

Roy, M.-H. and Larocque, D. (2012), 'Robustness of random forests for regression', *Journal of Nonparametric Statistics* **24**(4), 993–1006.
URL: *https://doi.org/10.1080/10485252.2012.715161*

Sahin, E. K. and Demir, S. (2023), 'Greedy-automl: A novel greedy-based stacking ensemble learning framework for assessing soil liquefaction potential', *Engineering Applications of Artificial Intelligence* **119**, 105732.
URL: *https://doi.org/10.1016/j.engappai.2022.105732*

Samadi, H., Hassanpour, J., Rostami, J. and Khatti, J. (2024), Application of supervised learning algorithms to predict engineering characteristics of soft to strong rock masses using actual tbm performance data, *in* '58th U.S. Rock Mechanics/Geomechanics Symposium', ARMA, Golden, Colorado, USA, p. D022S023R001.
URL: *https://doi.org/10.56952/ARMA-2024-0036*

Seed, H. B. and Idriss, I. M. (1971), 'Simplified procedure for evaluating soil liquefaction potential', *Journal of the Soil Mechanics and Foundations Division* **97**(9), 1249–1273.
URL: *https://doi.org/10.1061/JSFEAQ.0001662*

Shi, X., Wong, Y. D., Chai, C. and Li, M. Z.-F. (2021), 'An automated machine learning (automl) method of risk prediction for decision-making of autonomous vehicles', *IEEE Transactions on Intelligent Transportation Systems* **22**(11), 7145–7154.
URL: *https://doi.org/10.1109/TITS.2020.3002419*

Tang, L. and Na, S. (2021), 'Comparison of machine learning methods for ground settlement prediction with different tunneling datasets', *Journal of Rock Mechanics and Geotechnical Engineering* **13**(6), 1274–1289.
URL: *https://doi.org/10.1016/j.jrmge.2021.08.006*

Theng, D. and Bhoyar, K. K. (2024), 'Feature selection techniques for machine learning: a survey of more than two decades of research', *Knowledge and Information Systems* **66**(3), 1575–1637.
URL: *https://doi.org/10.1007/s10115-023-02010-5*

Torres, E. S. and Dungcaa, J. R. (2024), 'An interpretable machine learning approach in understanding lateral spreading case histories', *International Journal of GEOMATE* **26**(116).
URL: *https://doi.org/10.21660/2024.116.g13159*

Wang, Y. and Sherry Ni, X. (2019), 'A xgboost risk model via feature selection and bayesian hyperparameter optimization', *International Journal of Database Management Systems* **11**(01), 01–17.
URL: *https://doi.org/10.5121/ijdms.2019.11101*

Xie, Y., Ebad Sichani, M., Padgett, J. E. and DesRoches, R. (2020), 'The promise of implementing machine learning in earthquake engineering: A state-of-the-art review', *Earthquake Spectra* **36**(4), 1769–1801.
URL: *https://doi.org/10.1177/8755293020919419*

Xue, X., Yang, X. and Li, P. (2017), 'Application of a probabilistic neural network for liquefaction assessment', *Neural Network World* **27**(6), 557–567.
URL: *https://doi.org/10.14311/NNW.2017.27.030*

Ye, Y., Wu, Q., Huang, J. Z., Ng, M. K. and Li, X. (2013), 'Stratified sampling for feature subspace selection in random forests for high dimensional data', *Pattern Recognition* **46**(3), 769–787.
URL: *https://doi.org/10.1016/j.patcog.2012.09.022*

Youd, T. L., Idriss, I. M., Andrus, R. D., Arango, I., Castro, G., Christian, J. T., Dobry, R., Finn, W. D. L., Harder, L. F., J., Hynes, M. E., Ishihara, K., Koester, J. P., Liao, S. S. C., Marcuson, W. F., I., Martin, G. R., Mitchell, J. K., Moriwaki, Y., Power, M. S., Robertson, P. K., Seed, R. B. and Stokoe, K. H., I. (2001), 'Liquefaction resistance of soils: Summary report from the 1996 nceer and 1998 nceer/nsf workshops on evaluation of liquefaction resistance of soils', *Journal of Geotechnical and Geoenvironmental Engineering* **127**(4), 297–313.
URL: *https://doi.org/10.1061/(ASCE)1090-0241(2001)127:4(297)*

Zakariya, A., Rifa'i, A. and Ismanti, S. (2023), 'The correlation of liquefaction potential and probability on excess pore water pressure in kretek 2 bridge area', *Journal of the Civil Engineering Forum* pp. 39–48.
URL: *https://doi.org/10.22146/jcef.7002*

Zhang, J. and Wang, Y. (2021), 'An ensemble method to improve prediction of earthquake-induced soil liquefaction: a multi-dataset study', *Neural Computing and Applications* **33**(5), 1533–1546.
URL: *https://doi.org/10.1007/s00521-020-05086-6*

Zhang, P., Jia, Y. and Shang, Y. (2022), 'Research and application of xgboost in imbalanced data', *International Journal of Distributed Sensor Networks* **18**(6), 155013292211069.
URL: *https://doi.org/10.1155/2022/1550132*

Zhao, Z., Duan, W. and Cai, G. (2021), 'A novel pso-kelm based soil liquefaction potential evaluation system using cpt and vs measurements', *Soil Dynamics and Earthquake Engineering* **150**, 106930.
URL: *https://doi.org/10.1016/j.soildyn.2021.106930*

Zhao, Z., Duan, W., Cai, G., Wu, M. and Liu, S. (2022), 'Cpt-based fully probabilistic seismic liquefaction potential assessment to reduce uncertainty: Integrating xgboost algorithm with bayesian theorem', *Computers and Geotechnics* **149**, 104868.
URL: *https://doi.org/10.1016/j.compgeo.2022.104868*

Zhao, Z., Duan, W., Cai, G., Wu, M., Liu, S. and Puppala, A. J. (2024), 'Probabilistic capacity energy-based machine learning models for soil liquefaction reliability analysis', *Engineering Geology* **338**, 107613.
URL: *https://doi.org/10.1016/j.enggeo.2024.107613*