Data-Driven Modeling of Seismic Energy Dissipation of Rocking Foundations Using Decision Tree-Based Ensemble Machine Learning Algorithms

Sivapalan Gajan, Ph.D., M.ASCE[1]; Wakeley Banker[2]; and Alexander Bonacci[3]

[1]Associate Professor, College of Engineering, SUNY Polytechnic Institute, Utica, NY (corresponding author). Email: gajans@sunypoly.edu
[2]Undergraduate Student Researcher, College of Engineering, SUNY Polytechnic Institute. Email: bankerw@sunypoly.edu
[3]Undergraduate Student Researcher, College of Engineering, SUNY Polytechnic Institute. Email: bonaccav@sunypoly.edu

ABSTRACT

The objective of this study is to develop data-driven predictive models for seismic energy dissipation of rocking shallow foundations during earthquake loading using decision tree-based ensemble machine learning algorithms and supervised learning technique. Data from a rocking foundation's database consisting of dynamic base shaking experiments conducted on centrifuges and shaking tables have been used for the development of a base decision tree regression (DTR) model and four ensemble models: bagging, random forest, adaptive boosting, and gradient boosting. Based on k-fold cross-validation tests of models and mean absolute percentage errors in predictions, it is found that the overall average accuracy of all four ensemble models is improved by about 25%–37% when compared to base DTR model. Among the four ensemble models, gradient boosting and adaptive boosting models perform better than the other two models in terms of accuracy and variance in predictions for the problem considered.

INTRODUCTION

It has been shown that properly designed shallow foundations, with controlled rocking during earthquakes, have beneficial effects on the seismic performance of structures by dissipating seismic energy in soil and by effectively acting as geotechnical seismic isolation mechanisms (e.g., Gavras et al. 2020 and Gajan et al. 2021). Despite the mounting experimental evidences, foundation rocking and soil yielding is still perceived as an unreliable or unproven energy dissipation mechanism for reducing seismic force and ductility demands on the structure. The lack of practical, reliable dynamic soil-foundation interaction models for rocking foundations is among primary concerns that hinder the use of foundation rocking as a designed mechanism for improving the seismic performance of structural systems. As globally available experimental databases become increasingly common, machine learning algorithms in predictive modeling have become efficient in many fields (e.g., Geron 2019). Models based on machine learning algorithms have the ability to learn directly from experimental data and generalize experimental behavior, capture the effects and propagation of uncertainties, and hence can be used in combination with mechanics-based and physics-based models as complementary measures in practical applications.

The objective of this research is to develop well-trained and tested data-driven predictive models for seismic energy dissipation of rocking foundations using multiple decision-tree based ensemble machine learning algorithms and supervised learning technique. Data from a rocking

foundation database, consisting of 140 dynamic base shaking experiments conducted on centrifuges and shaking tables in the US and Greece are used for training and testing of machine learning models. The input features to machine learning models include critical contact area ratio of foundation, slenderness ratio and rocking coefficient of rocking system, and peak ground acceleration and Arias intensity of earthquake motion. The base machine learning algorithm used in this study is decision tree regression (DTR), a nonlinear, nonparametric algorithm that uses supervised learning technique to build a tree-like data structure. Multiple DTR models are then combined together to develop four ensemble models using bootstrap aggregating (bagging), random forest, adaptive boosting (ada-boosting), and gradient boosting ensemble methods.

DATA MINING AND DATA PREPARATION

Database. The results obtained from five series of centrifuge experiments and four series of shake table experiments (altogether 140 individual experiments) have been used in this study. The centrifuge experiments were conducted in University of California at Davis (Gajan and Kutter 2008, Deng et al. 2012, Deng and Kutter 2012, and Hakhamaneshi et al. 2012) and the shake table experiments were conducted in University of California at San Diego (Antonellis et al. 2015) and the National Technical University of Athens in Greece (Drosos et al. 2012, Anastasopoulos et al. 2013, and Tsatsis and Anastasopoulos 2015). Details of these experiments, including types of soils, foundations, structures, and ground motions, number of shaking events, raw data, and meta data, are available in a database (Gavras et al. 2020). A summary of processed data from these experiments in terms of meaningful engineering parameters and the relationships among them are published in Gajan et al. (2021).

Input features. Input features for machine learning algorithms considered in this study include three rocking system parameters (critical contact area ratio (A/A_c), slenderness ratio (h/B), and rocking coefficient (C_r)) and two earthquake ground motion parameters (peak ground acceleration (a_{max}) and Arias intensity of ground motion (I_a)). A/A_c is conceptually a factor of safety for rocking with respect to vertical loading (where A is the total base area of the footing when in full contact with the soil and A_c is the minimum footing contact area required to support the applied vertical load). The moment capacity (M_{ult}) of a rocking foundation has been shown to correlate with A/A_c through the following relationship (Gajan and Kutter 2008),

$$M_{ult} = \frac{V \cdot B}{2} \cdot \left[1 - \frac{A_c}{A}\right]$$

The rocking coefficient of a soil-foundation-structure system, C_r, is the ratio of ultimate rocking moment capacity of the foundation to the weight (V) of the structure normalized by the effective height (h) of the structure (Deng et al. 2012),

$$C_r = \frac{B}{2 \cdot h} \cdot \left[1 - \frac{A_c}{A}\right]$$

where, B is the width of the footing in the direction of shaking. These five input features have been selected based on their close relationships with seismic energy dissipation presented in Gajan et al. (2021). The range of values, the mean, and the standard deviation of all five input

features are presented in Table 1. As can be seen from Table 1, experimental results utilized in this study cover a wide range of rocking system capacity parameters and ground motion demand parameters.

Table 1. Details of input features used in machine learning algorithms

Input feature	A/A_c	h/B	C_r	a_{max} (g)	I_a (m/s)
Range of values	1.9 - 17.1	1.2 - 2.83	0.08 - 0.36	0.04 - 1.28	0.03 - 26.4
Mean value	8.17	1.89	0.24	0.43	2.31
Standard deviation	4.27	0.53	0.08	0.26	4.37

Performance parameter. Seismic energy dissipation (ED) in soil during foundation rocking is the performance (prediction) parameter considered in this study and is calculated from the total area enclosed by the cyclic moment-rotation hysteretic loops of foundation. ED is normalized by the applied vertical load on the foundation and the width of the foundation in order to obtain a nondimensional parameter called normalized seismic energy dissipation [NED = ED/(V·B)] and to make comparisons meaningful across different experiments. The data from the database for 140 individual experiments have been processed to obtain the variation of NED with I_a of earthquake for different clusters (C_r) of rocking systems and for sandy soils and clays. The experimental data used in this study is shown in Figure 1 (Gajan et al. 2021). The data presented in Figure 1 generally indicate that (i) the amount of scatter in data is relatively high and (ii) simple statistics-based models are not capable of correlating the data with reasonable accuracy. This hypothesis is verified when the accuracy of machine learning models developed in this study is compared with the accuracy of simple, statistics-based linear regression models (presented in Results section).

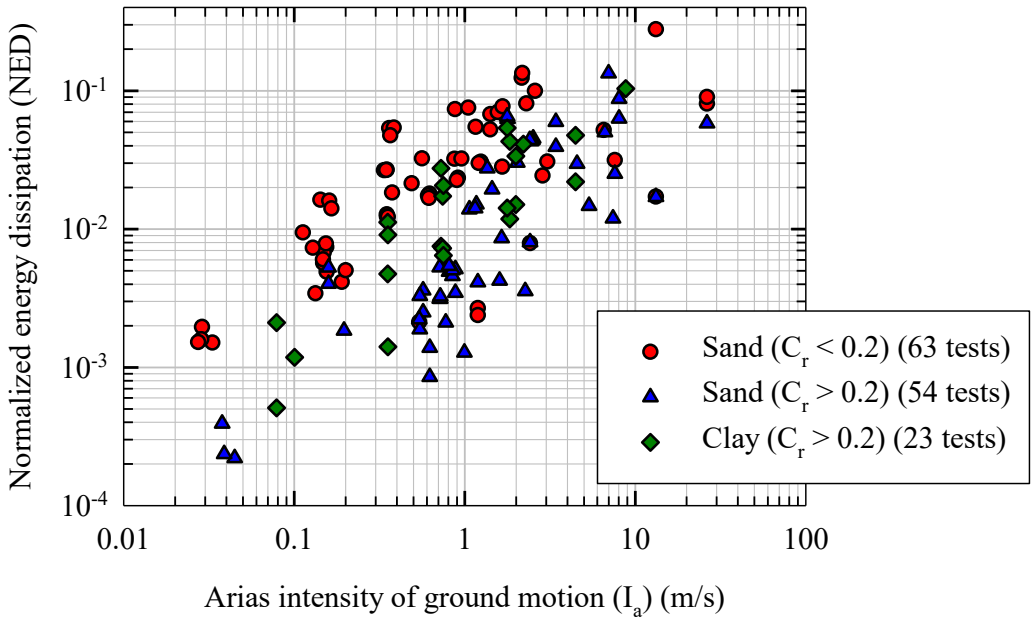

Figure 1. Variation of normalized seismic energy dissipation in foundation soil with Arias intensity of earthquake and rocking coefficient of foundation (Gajan et al. 2021)

Feature transformation and normalization. As can be seen from Figure 1, the variation in NED and I_a is relatively high, and hence the data is plotted in log – log space. For this reason, these two parameters (NED and I_a) are transformed to logarithmic values (base 10) before the training and testing phases of machine learning models. In addition, in order to make reliable predictions using models developed by different machine learning algorithms, all the input feature data are normalized by using min-max feature scaling so that each input feature value varies between 0.0 and 1.0.

MACHINE LEARNING (ML) ALGORITHMS

Decision tree regression (DTR). The DTR is a nonlinear, nonparametric ML algorithm that uses supervised learning technique to build a tree-like data structure by employing a top-down, greedy search through the space of possible branches using information gain as a measure of reduction in uncertainty in data (reduction in entropy in data). While building the tree, the DTR algorithm minimizes a cost function to choose a single feature (k) and a threshold value (t_k) for that feature when deciding a split, and the process is repeated until it reaches the maximum depth of the tree or if it cannot find a split that would reduce the uncertainty further. The cost function that the DTR algorithm minimizes is given by $J(k, t_k)$ (Geron 2019),

$$J(k, t_k) = \frac{m_{left}}{m}.E_{left} + \frac{m_{right}}{m}.E_{right}$$

where E_{left} and E_{right} measure the mean absolute error of the left and right subsets of the splitting node, respectively, and m_{left} and m_{right} are the number of instances in the left and right subsets, respectively ($m = m_{left} + m_{right}$). The major hyperparameter of the DTR model is the maximum depth of the tree, as deeper trees tend to overfit the training data while relatively shorter (shallow) trees tend to underfit the training data. During testing or prediction phase, the DTR model arrives at an estimate by asking a series of questions to the data related to input feature values, each question narrowing the possible values until the model gets confident enough to make a prediction. The final prediction is the average value of the dependent (prediction) variable in that leaf node.

Bagging regression (BAR) and random forest regression (RFR). A bagging regression is an ensemble model that fits multiple base DTR models each on random subsets of the training dataset and then aggregates each individual DTR's predictions on testing dataset by averaging to form a final prediction. Bagging is typically used as a way to reduce the variance of the base model by introducing randomization into its construction procedure and making an ensemble out of it. A special case of general bagging model is called a random forest model, where random number of input features are used to build multiple DTR models of different depths (hence the name random forest). For the base DTR model and the BAR model developed in this study, the number of features is equal to 5 and the maximum depth of the tree is equal to 6. For RFR model, the maximum number of features for a base DTR is limited to 2 (randomly chosen 2 features out of total 5 features) and the maximum depth of the tree is limited to 6 (random depths that can vary from 1 to 6). Therefore, the RFR algorithm, in general, introduces more randomness in the model and is expected to perform better than the bagging ensemble model. The optimum value of number of trees needed (base DTR models) for all four ensemble models is found to be around 50 (discussed further in hyperparameter tuning section).

Adaptive boosting regression (ABR) and gradient boosting regression (GBR). The idea of boosting method is to train multiple individual base DTR models sequentially, each trying to correct the predecessor. In the process, the boosting method combine several weak learners into a strong learner (Geron 2019). In adaptive boosting, each base DTR model has a weight (predictor weight) and each training data instance also has a weight (instance weight). The ABR model begins by fitting a DTR on the original dataset and then fits additional DTR models on the same dataset but where the weights of instances of data are adjusted according to the error of the current prediction (using the current predictor weight). As such, subsequent DTR models focus more on difficult cases of data. When ABR model makes predictions on a new data instance (from test dataset), it simply computes the predictions of all individual base DTR models and weighs them using the predictor weights. Similar to ABR, gradient boosting (GBR) works by sequentially adding base DTR models to an ensemble, each one correcting its predecessor. However, instead of tweaking the data instance weights at every iteration, the GBR fits the new predictor to the residual errors made by the previous DTR model. When GBR model makes predictions on a new data instance (from test dataset), it simply computes the predictions of all individual base DTR models and adds them up together.

RESULTS AND DISCUSSION

Initial evaluation of ML models. Altogether 140 experimental data (instances) are extracted from the above-mentioned database and are randomly split into two groups for the purpose of initial training (supervised learning) and testing (validation) of ML models. A 70%– 30% split is used to randomly create a training dataset (98 instances) and a testing dataset (42 instances) using the train-test-split function available in "scikit-learn" library of modules in Python (https://scikit-learn.org/stable/). In addition to the decision tree-based ML models described above, for the purpose of comparisons, a multivariate linear regression (MLR) model is also developed using the same input features and supervised learning technique. Since MLR model is a linear, parametric model, it is used as the baseline model for comparison of model performances. All ML models are first trained using the same training dataset (98 instances) and then tested using previously unseen testing dataset (42 instances). The variation of predicted NED with their corresponding experimental NED values on top of 1:1 prediction-to-experiment comparison lines during initial testing phase are presented in Figure 2 for all four decision tree-based ensemble ML models (in all four cases, the maximum depth of tree = 6 and the number of trees in the ensemble = 50). Also included in Figure 2 are the mean absolute percentage errors (MAPE) in predictions of each model during testing.

For the same train-test split of data, the testing MAPE of baseline MLR model and the base DTR model are 0.76 and 0.83, respectively. As can be seen from the results presented in Figure 2 (a) through (d), it is clear that all four ensemble models perform better than the base DTR model and the baseline MLR model. The MAPE of predictions of the ensemble models vary from 0.49 to 0.60, which is a significant improvement (28% to 41% increase in accuracy on average) from the corresponding MAPE of base DTR model (0.83). As can be seen from Figure 2 (a) and (b), the predictions of BAR and RFR models look very similar (but not identical) with a MAPE of about 0.52 and 0.53, respectively, on test data. This is because of the low number of features used in this study (random forest model is typically more efficient compared to bagging when the number of features are relatively high (Geron 2019)). On the other hand, the prediction results and the MAPE values of BAR and RFR models reinforce the consistency of the

algorithms (RFR is a special case of BAR). Both ABR and GBR models also show similar trends (Figure 2 (c) and (d)), but with different MAPE values (about 20% difference). It appears that ABR model performs better than the GBR model on this test data, however, as will be seen later, the GBR model turns out to be the best ensemble model in terms of overall average MAPE and the variance in MAPE in k-fold cross validation tests (presented later).

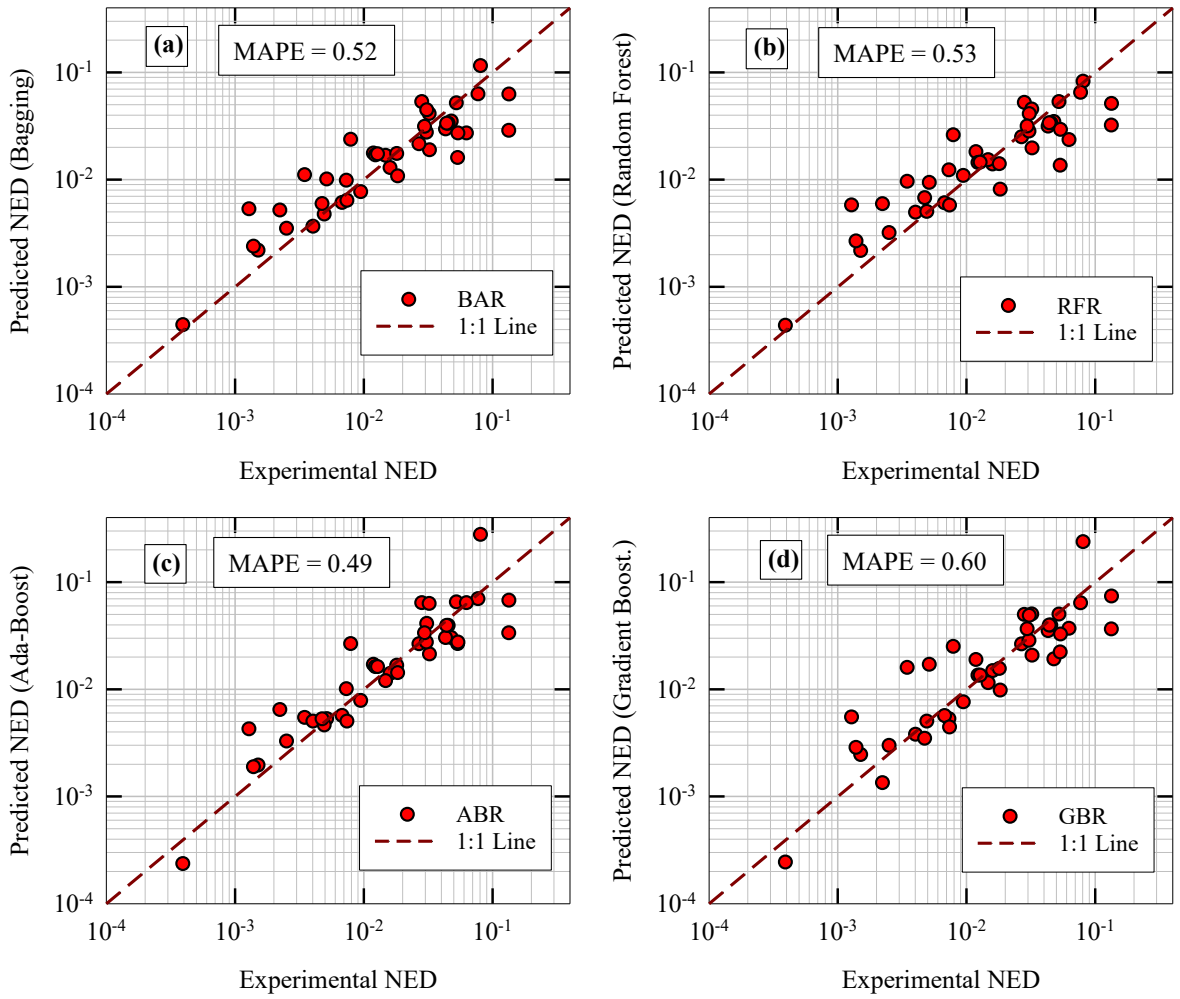

Figure 2. Comparisons of model predictions of normalized energy dissipation (NED) with experimental results for ensemble ML models during initial testing phase

Hyperparameter tuning of models. For the base DTR model and four decision tree-based ensemble models presented above, hyperparameter tuning is carried out (i) to investigate the sensitivity of model predictions to their hyperparameters, (ii) to determine the optimum values of these parameters for the problem considered and data analyzed, and (iii) to ensure that the models neither overfit nor underfit the training data. The maximum depth of the tree of the base DTR model is varied first, and once an optimum maximum depth is found, the number of trees is varied for the ensemble models while the maximum depth of base DTR model is kept at its optimum value. In each case, the corresponding MAPE values in predictions during initial testing phase are calculated and the results are presented for all five models in Figure 3.

For base DTR model (Figure 3 (a)), the MAPE values during testing phase first decreases and then increases as the maximum depth of tree increases, with the optimum maximum depth of the tree being 6. Any value smaller than 6 for maximum depth of tree would underfit the training data and any value greater than 6 for maximum depth of tree would overfit the training data. Based on this observation, maximum depth of tree for optimum performance of DTR model is found to be 6 for the problem considered, and all other results of DTR models presented in this paper are obtained using maximum depth of tree = 6. For all four decision tree-based ensemble models, as can be seen from Figure 3 (b) and (c), the MAPE during testing decreases as number of trees increases, indicating that the accuracy of the models increases with the number of trees. However, after about 50 to 100 trees in the ensemble, the MAPE of all four ensemble model do not seem to decrease much further, indicating that the optimum number of trees required is within this range. This is remarkably consistent for all four ensemble models. The optimum value for number of trees in the ensemble is chosen to be 50 and all other results of the four ensemble models presented in this paper are obtained using this value.

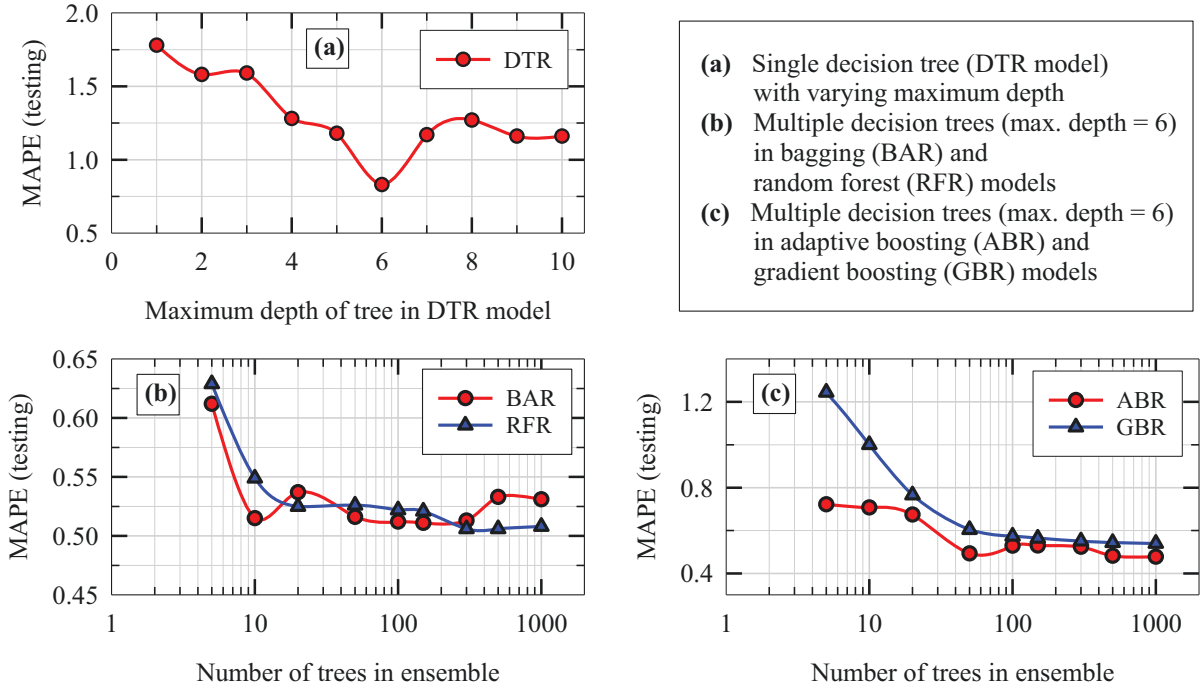

Figure 3. Results of hyperparameter tuning of decision tree-based ML models

Importance of input features. The implementations of RFR, ABR, and GBR algorithms in scikit-learn (in Python) have a function called "feature importances", which outputs relative importance of each input feature. This function measures a feature's importance by looking at how much the tree nodes that use that feature reduce uncertainty in data on average (across all trees in the ensemble). The function computes this score for each feature after training and normalizes the results so that the sum of all feature importances is equal to 1.0. Figure 4 presents the results of normalized feature importance values obtained from RFR, ABR, and GBR models after initial training phase. As can be seen from Figure 4, the earthquake ground motion intensity parameters (I_a and a_{max}) have higher scores for feature importance, compared to rocking system

parameters. This indicates that the seismic energy dissipation is more sensitive to ground motion parameters than to rocking foundation capacity parameters. A/A_c and h/B have feature importance scores of 10% to 15% each, while the feature importance of C_r is 7% to 10%, indicating that none of the input features are redundant. These observations are remarkably consistent over all three ensemble models, except that the feature importance values in GBR model for I_a and a_{max} are slightly different compared to the other two models.

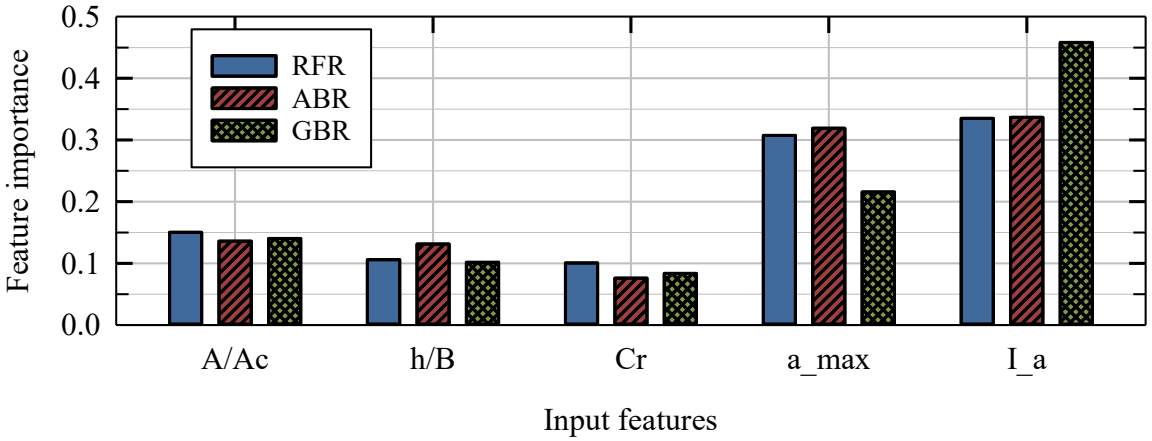

Figure 4. Normalized feature importance scores obtained from three ensemble ML models

K-fold cross validation tests of models. The k-fold cross validation is a technique for evaluating the performance of machine learning models on multiple, different train-test splits of dataset. It provides a measure of how good the model performance is both in terms of bias and variance. In this study, the entire dataset is randomly shuffled and split into different groups of train-test sets for 7-fold cross validation. In a 7-fold cross validation, six folds of data are used for training of ML models and one fold is used for testing, and this process is repeated 7 times. On top of this, the whole process is repeated 3 times with different randomization of the data in each repetition. With this setup, for each model, the 7-fold cross validation tests result in 21 sets of predictions and 21 different values for testing MAPE. Figure 5 presents the results of 7-fold cross validation test for base DTR model and all four ensemble models developed in this study along with baseline MLR model. In each case, results obtained for testing MAPE are plotted in the form of boxplots. For each ML model, the boxplot plots the average of MAPE, median of MAPE (the horizontal line inside the box), 25^{th} and 75^{th} percentile values of MAPE (bottom and top edges of the box), and the 10^{th} and 90^{th} percentile values of MAPE (bottom and top horizontal lines outside the box).

As can be seen from Figure 5, the average MAPE values of all five decision tree-based models are smaller than that of base MLR model (average MAPE = 0.9), and the average MAPE values of all four decision tree-based ensemble models (average MAPE = 0.5 to 0.6) are smaller than that of the base DTR model (average MAPE = 0.8). This indicates that by combining 50 DTR base models together in four different ensemble ML models, the average accuracy is improved by 25% to 37% for the problem considered. The average MAPE values of the models alone suggest that, for the problem considered, the overall accuracy of model predictions follows the following order, from the most accurate to the least accurate: GBR, ABR, RFR, BAR, DTR, and MLR. The variance of DTR model predictions is the highest among all six models.

However, by combining only 50 DTR base models, the four ensemble methods reduce the variance in model predictions quite significantly. More specifically, the GBR model reduces the variance in predictions by about 47% when compared to base DTR model.

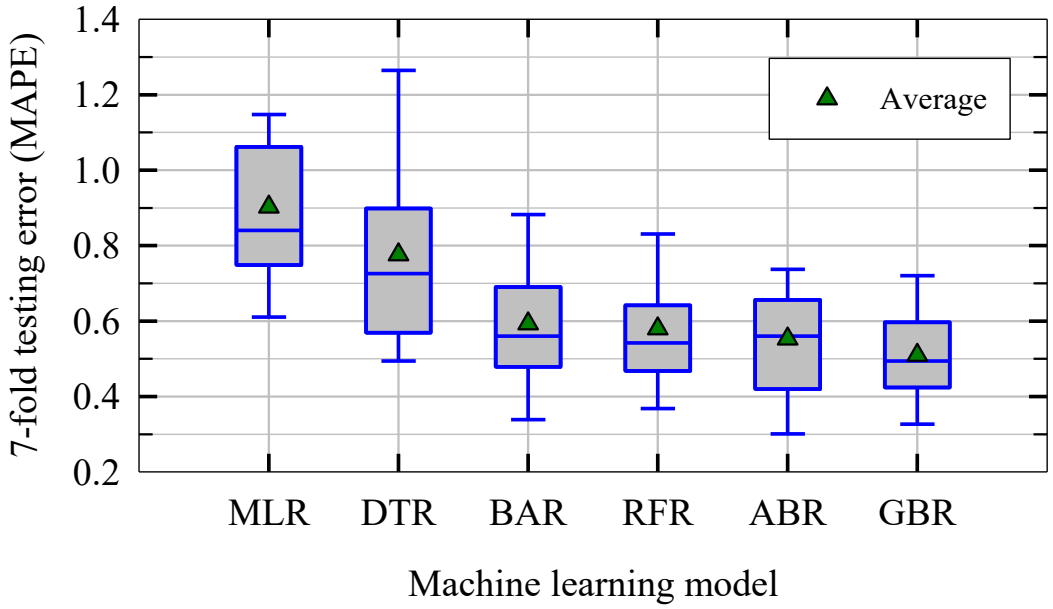

Figure 5. Summary results of 7-fold cross validation of ML models in terms of the average and variance of mean absolute percentage errors (MAPE)

Comparisons with simple linear regression models. When the experimental data presented in Figure 1 (all 140 instances) are run through a simple linear regression (SLR) algorithm, with log (I_a) as independent variable and log (NED) as dependent variable, it results in a coefficient of determination (R^2 value) of 0.44. When the data is categorized into four groups (all sand, sand $C_r < 0.2$, sand $C_r > 0.2$, and clay $C_r > 0.2$) and run through the SLR model, the following R^2 values are obtained: 0.41, 0.50, 0.68, and 0.77, respectively. For comparison, the R^2 values of all four DTR-based ensemble ML models developed in this vary from 0.92 to 0.99 on training data, and the R^2 values of predictions of previously unseen test data are 0.8 or greater. This indicates that the ML models developed in this study are better than the SLR models in terms of accuracy.

CONCLUSIONS

Based on the results presented in this paper, the following major conclusions are derived:

- All four decision tree-based ensemble machine learning models developed in this study (BAR, RFR, ABR, and GBR) perform better than a baseline MLR model and simple statistics-based SLR models in terms of accuracy and variance in predictions.
- The average accuracy of all four decision tree-based ensemble models, with only 50 trees in each ensemble, is improved by about 25% to 37% when compared to base DTR model. The variance in model predictions is reduced by as much as 47% (GBR model) when compared to base DTR model.

- For the problem considered and data analyzed, the overall average accuracy of model predictions follows the following order, from the most accurate to the least accurate: GBR, ABR, RFR, BAR, DTR, and MLR. Hyperparameter tuning of models is carried out to make sure that models neither underfit nor overfit the training data.
- Based on the "feature importance" values obtained from three ensemble models, it can be concluded that the five chosen input features capture the rocking induced energy dissipation satisfactorily and that the energy dissipation is more sensitive to ground motion demand parameters than to rocking foundation capacity parameters.
- The data-driven predictive models developed in this study can be used in combination with other mechanics-based models to complement each other in modeling of rocking foundations in practical applications. One such approach would be theory-guided machine learning, an emerging field that combines physics with data science.

ACKNOWLEDGEMENTS

The authors acknowledge the National Science Foundation (NSF) for funding the research presented in this paper through award number CMMI-2138631.

REFERENCES

Anastasopoulos, I., Loli, M., Georgarakos, T., and Drosos, V. (2013). "Shaking table testing of rocking-isolated bridge pier on sand." *J. Earthquake Eng.* 17, 1–32.

Antonellis, G., Gavras, A. G., Panagiotou, M., Kutter, B. L., Guerrini, G., Sander, A. C., and Fox, P. J. (2015). "Shake table test of large-scale bridge columns supported on rocking shallow foundations." *J.Geotech. Geoenviron. Eng.* 141.

Deng, L., and Kutter, B. L. (2012). "Characterization of rocking shallow foundations using centrifuge model tests." *Earthquake Engng Struct. Dyn.* 41, 1043–1060.

Deng, L., Kutter, B. L., and Kunnath, S. K. (2012). "Centrifuge modeling of bridge systems designed for rocking foundations." *J. Geotech. Geoenviron. Eng.* 138, 335–344.

Drosos, V., Georgarakos, T., Loli, M., Anastsopoulos, I., Zarzouras, O., and Gazetas, G. (2012). "Soil-Foundation-Structure Interaction with Mobilization of Bearing Capacity: Experimental Study on Sand." *J. Geotech. Geoenviron*, 138(11), 1369–1386.

Gavras, A., Kutter, B. L., Hakhamaneshi, M., Gajan, S., Tsatsis, A., Sharma, K., Kouno, T., Deng, L., Anastasopoulos, I., and Gazetas, G. (2020). "Database of rocking shallow foundation performance: Dynamic shaking." *Earthquake Spectra*.

Gajan, S. (2021). "Modeling of seismic energy dissipation of rocking foundations using nonparametric machine learning algorithms." *Geotechnics*, 1, 534–557.

Gajan, S., and Kutter, B. L. (2008). "Capacity, settlement, and energy dissipation of shallow footings subjected to rocking." *J. Geotech. Geoenviron. Eng.* 134, 1129–1141.

Gajan, S., Soundararajan, S., Yang, M., and Akchurin, D. (2021). "Effects of rocking coefficient and critical contact area ratio on the performance of rocking foundations from centrifuge and shake table experimental results." *Soil Dyn. Earthquake Engng*, 141.

Geron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools and techniques to build intelligent systems*. 2nd ed. O'Reilly Media, Inc., Sebastopol, CA.

Hakhamaneshi, M., Kutter, B. L., Deng, L., Hutchinson, T. C., and Liu, W. (2012). "New findings from centrifuge modeling of rocking shallow foundations in clayey ground." in *Proc, Geo-Congress 2012*, 25–29 March, 2012, Oakland, CA.

Tsatsis, A., and Anastasopoulos, I. (2015). "Performance of rocking systems on shallow improved sand: shaking table testing." *Front. Built Environ.* 1:9.