Dissertation presented to the Dean of Graduate Studies of the Technological Institute of Aeronautics, as part of the requirements for obtaining the Master of Science degree in the Graduate Program in Engineering of Aeronautical Infrastructure Aeronautical.

Marcelo Saraiva Peres

Machine Learning Techniques Application for Installation Torque Prediction of Helical Piles

Dissertation approved in its preliminary version by the undersigned:

Prof. Dr. Dimas Betioli Ribeiro
Orientador

Prof. Dr. José Antonio Schiavon
Coorientador

Prof. Dr. Nome Completo
Dean of Graduate Studies

Campo Montenegro
São José dos Campos, SP – Brasil
2021

Machine Learning Techniques Application for Installation Torque Prediction of Helical Piles

Marcelo Saraiva Peres

Banca Examinadora:

Prof. Dr. xxxxx	Presidente	- ITA
Prof. Dr. xxxxx	Orientador	- ITA
Prof. Dr. xxxxx	Coorientador	- ITA
Prof. Dr. xxxxx		- UFRGS
Prof. Dr. xxxxx		- UNIPAMPA
Prof. Dr. xxxxx		- USP

ITA

Agradecimentos

To Cristina de Hollanda Cavalcanti Tsuha for making available the dataset used in this work and Bruno Oliveira da Silva for helping with data curation and the Coordination of Improvement of Higher Education Personnel (CAPES) for granting the first author's scholarship.

If I have seen farther than others, it is because I stood on the shoulders of giants.

Resumo

xxxxx.

Abstract

Installation torque is a key parameter to define installation equipment and helical piles final depth; however, installation torque estimation is still a challenge. In this work, machine learning (ML) techniques are applied for helical piles installation torque prediction based on information from 707 helical piles installation reports, including SPT data. This information was used to build three datasets used by the eight machine learning techniques evaluated in this study. The input attributes include variables taken directly from the reports, and also other informative variables generated for the ML techniques. The relevance of each attribute was evaluated considering its correlation with others and its effective importance within the ML procedures. The performance in predicting torque was evaluated using classical statistical measures and the confidence interval concept. Cubist, random forest and boosting techniques showed the best performance in torque prediction and the best performing model (cubist) was used in a case study. The proposed approach shows potential to aid in the anchoring design and in the equipment choice for helical pile installation

List of Figures

List of Tables

Contents

Lista de Abreviaturas e Siglas

CAPES: Coordination of Improvement of Higher Education Personnel

Lista de Símbolos

α: alfa

β: Beta;

ε: varepsilon;

φ:varphi;

Sobre o Autor

Aqui posso criar algum texto

1 Introduction

Helical piles are composed of steel circular plates welded to a central shaft, which usually has a tubular or a square solid cross section. Usually, the helical plates are equally spaced, with equal or increasing diameter from the tip to the top of the pile. Helical piles are installed into the ground at a constant rotation rate associated with a vertical downward (crowd) force mainly at the initial depths. It is recommended to proceed the pile penetration at a vertical advance of approximately one helix pitch length each rotational cycle, which aims at minimizing the penetrated soil disturbance [1, 2, 3, 4]. A torque indicator is coupled between the hydraulic motor and pile to measure the torque values during installation and, therefore, the pile capacity can be estimated via torque-to-capacity correlations [5].

The use of helical piles began in Brazil in the late 1990s, serving mainly as anchors for cables of guyed transmission towers, therefore, subjected to tensile loading [6]. Their use increased in the following years [7], which could be attributed to characteristics such as fast and easy installation, possibility of immediate loading after driving [8], negligible vibration and noise during installation, and the possibility of estimating load capacity from the torque values measured at the end of the installation [9].

Several methods are available in the literature to estimate the pile capacity [5], including a recent study that applies machine learning techniques for tdesignhis purpose [10]. Additionally, the empirical correlation between torque and uplift capacity (K_t method) is an approach usually adopted in the practice of validating the pile installation depth [9]. Since correlation between torque and pile capacity is well developed in the literature [2], some studies seek to predict torque as an indirect way to determine the pile capacity using, for example, information from the cone penetration test [11]. Nonetheless, in most Brazilian projects, the only *in situ* test available is the Standard Penetration Test (SPT).

In this context, the objective of this work is to apply eight machine learning techniques to produce models capable of predicting helical piles installation torque from basic preliminary design information, which includes only SPT as *in situ* test. This is an

extension of the study of [12], which used linear regressions applied to a similar dataset. To improve the model accuracy, new variables are obtained from raw data. The performance of different machine learning techniques is compared using the mean absolute error (MAE) the root of the mean squared error (RMSE) and the coefficient of determination (R^2). The variable importance is measured using the three best performing models, which are also evaluated using the concept of confidence interval.

2 General methodology

The methodology proposed to obtain the final machine learning models to be used is divided into three basic steps:

- Pre-processing,
- Machine learning techniques calibration and
- Evaluation of models.

The strategies employed into each of these steps are summarized hereafter. This Section also presents brief concepts of multiple linear regression and variable importance.

3 Pre-Processing

The raw dataset used in this study contains information from the tower foundations construction on a 350 kilometer stretch of a power transmission line located between the towns Paranatinga and Cláudia, Mato Grosso State, Central-West Region of Brazil. The region presents a very diversified geological context. Sandy sediments, partially covered by a sand-clay layer, predominate the northern portion of this region. In the southern portion, old rocks and recent sediments are found [13]. The predominant soil types in the study area are latosols, eutrophic podzolic, cambisol, hydromorphic laterite, low humic glei, quartz sands, alluvial soils, litholic soils and concretion soils. The great variety of soils is a result of the geomorphological units diversity and lithologies found in the study area [14].

The information on soil type was provided on the SPT reports firstly used for the transmission line design. The soil classification was based upon information on particle size (sieve analysis), soil plasticity, color, and soil formation (e.g. sedimentary, residual), with such procedures established in the Brazilian standard for SPT test. Most samples were identified as clayey sand (65.8%), followed by clayey silt (13.8%), sandy silt (9.6%), sandy clay (5.9%), and sand (0.3%). Fig. **??**fig: fig-1) XXXXXXX presents the SPT blow counts occurrence (hereinafter referred to as SPT index) versus depth. The SPT tests were

carried out according to the Brazilian standard [15], which describes a testing procedure similar to the international procedure [16] . For Brazilian standards equipment, the SPT efficiency ranges between 70 and 80%, 72% the most common value [17] and, therefore, this is the considered value at this work.

The helical piles installed in the study area were comprised of a 101,6 mm diameter steel tubular shaft and six helical plates with diameters of 254, 305, 366, 366, 366, and 366 mm, with increasing diameter from the tip to the top of the pile. The helix spacing was three times the diameter of the smaller helix [12]. The piles were installed using a hydraulic drive head attached to a backhoe loader. The torque measurement at each meter of pile penetration was provided by a torque indicator coupled between the driving head and the pile top. The pile advance rate was not included in the installation reports. In addition to soil type and SPT information, the raw dataset used in this study includes 707 helical piles installation torque, the pile inclination and the pile final length.

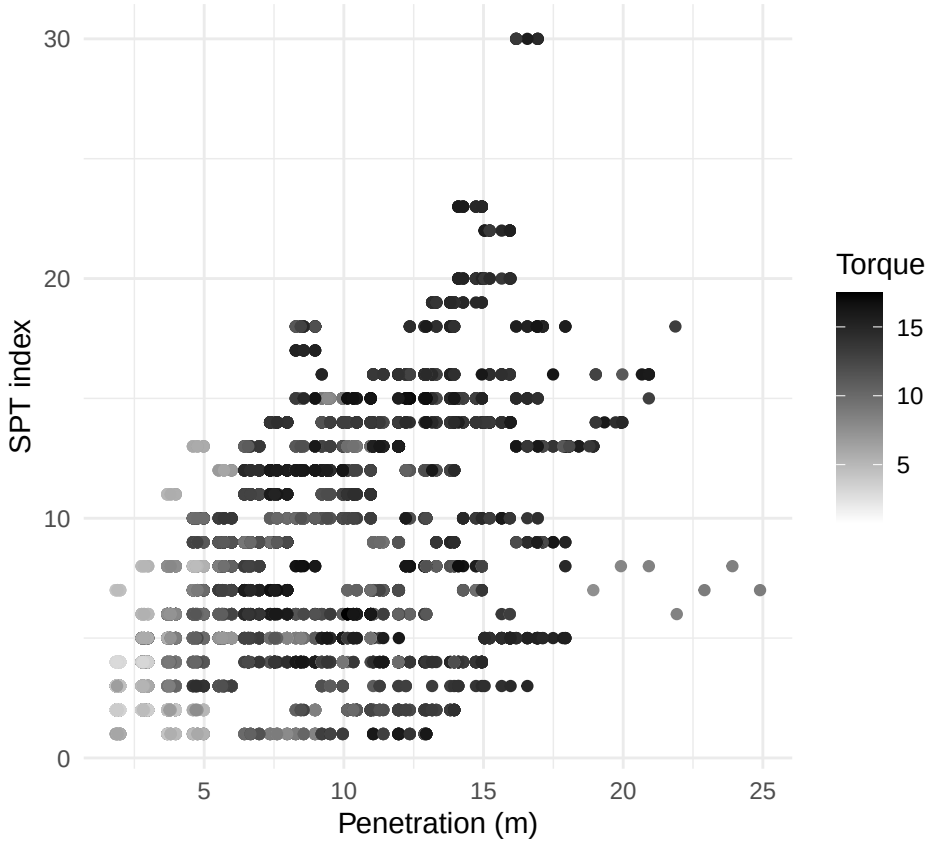

Figure 1: SPT index values versus occurrence depth

The first dataset assembled, which totaled 9355 samples, was pre-processed to improve the machine learning techniques performance. Each sample correspond to a torque value for a specific pile number (identification) with a specific penetrated length, which consequently can be associated with the soil type and the SPT index.

The first concern is cleaning data to eliminate errors like missing values and inconsistencies, a procedure that reduced the original dataset to 7632 samples. Next, variables must be chosen as inputs for predicting the installation torque.

The variable P refers to the pile tip depth at the moment in which torque is measured, as shown in Fig. XXXXXX. The variable $NSPT_{tip}$ represents the SPT index at the pile tip depth. The so called Initial Dataset is composed only of these two inputs plus the measured torque, which is simply referred to as $Torque$. Table xxxxxx presents an Initial Dataset sample.

Table 1: Initial Dataset sample

Penetration	NSPTtip	Torque
1.99	2	1.76
2.99	3	3.66
3.98	6	4.88
4.98	5	6.24
5.98	5	7.46
6.97	7	8.41
7.97	9	8.54
8.97	6	8.68
9.96	6	10.30
10.96	6	11.80

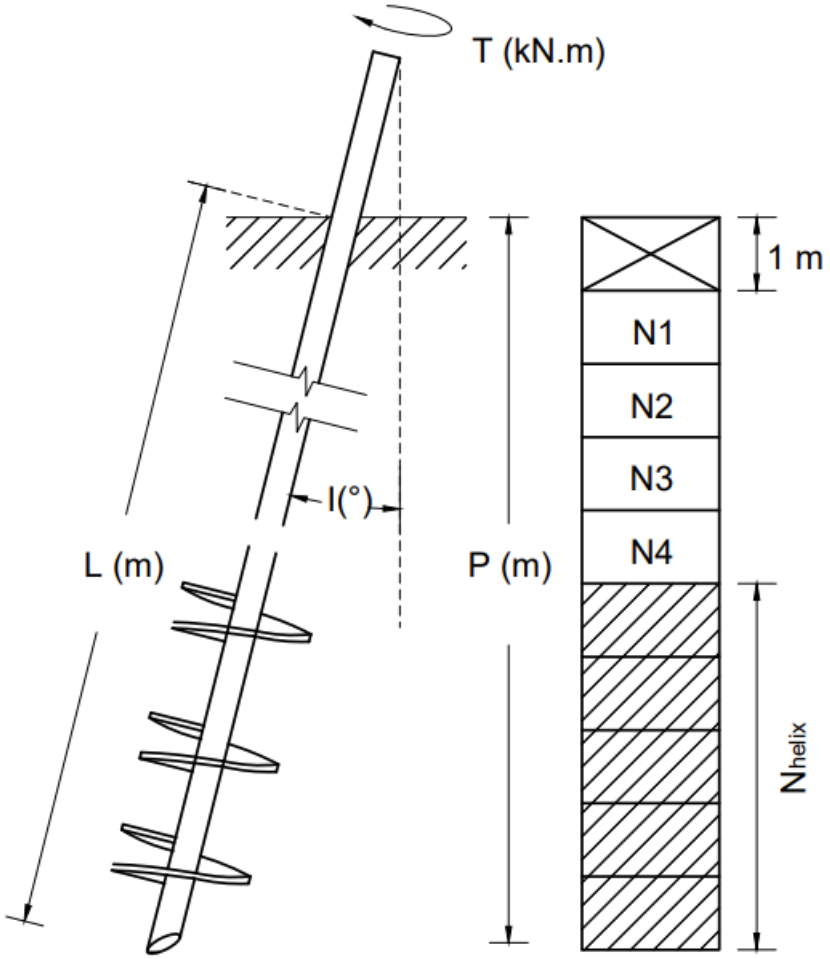

Figure 2: Helical pile schematic representation

Other proposed inputs are N_1, N_2, N_3 and N_4, representing the SPT index of first 4 meter depth after excluding the first meter (see Fig. XXXXXXXX). It is well known in the Brazilian practice that the pile penetration causes a gap around the pile shaft with at least 1 meter depth in the ground. Therefore, the soil contribution to the torque at 1 meter depth is disregarded.

The soil type is also included using six predefined classes, represented by the binary variables S_1, S_2, S_3, S_4, S_5 and S_6. These soil types and their encoding are listed in Table XXXXXX. The mean SPT index for each depth of helical plates is also included as the input $NSPT_{helix}$. The variable N_{shaft} is the summation of all SPT indexes measured above the helical plates, representing the shaft resistance. The Initial Dataset combined

with these new inputs constitutes the so called Complete Dataset, with 14 inputs. All these variables and the output *Torque* are summarized in Table

The pre-processing final step is evaluating input correlation, it measures the association between variable pairs. Values close to ±1 indicate perfectly related variables, while 0 indicate they are independent. Highly correlated inputs should be avoided because they cause redundancies that can destabilize regression techniques [18]. In this work, all correlation indexes belong to the interval $[-0.9, 0.9]$, which can be considered reasonable [19]. This is illustrated in Fig. XXXXX, which shows the so called correlation matrix for the Complete Dataset.

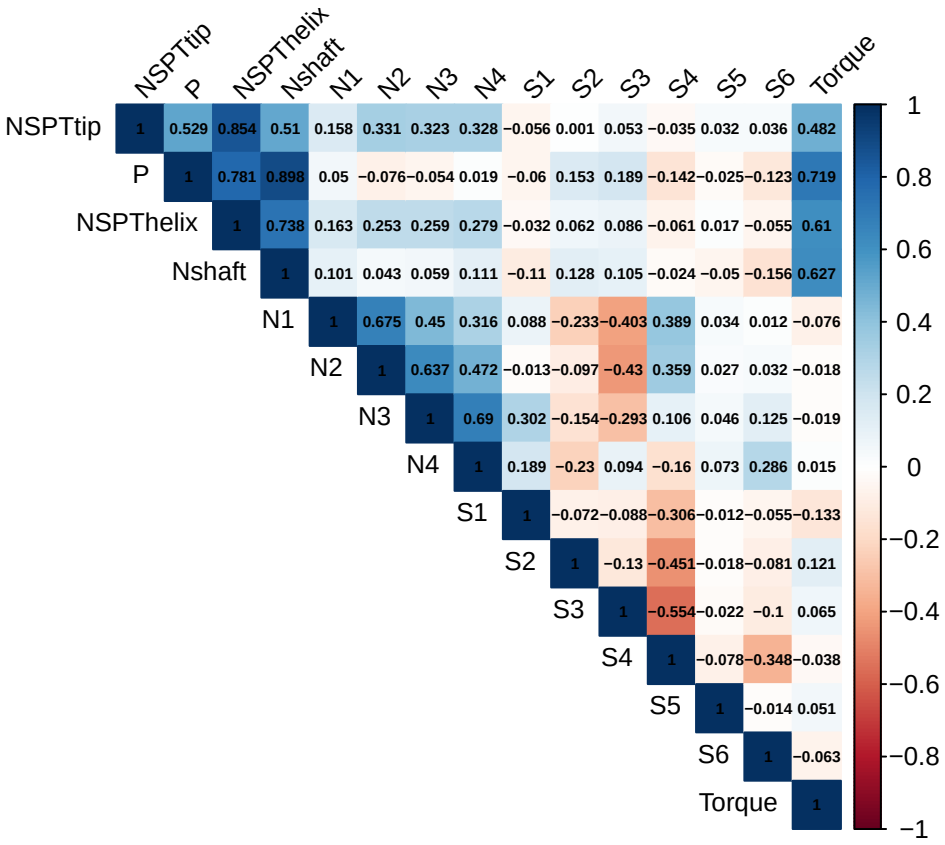

Figure 3: Correlation between variables

Many other inputs were considered in this study; however, they are omitted here for conciseness sake. The ones selected to be presented in this section are those that provided the most interesting conclusions, including the most accurate results. It was also considered that the three main parts of the helical pile (shaft above helical plates, helical

plates region and pile tip) should be well represented. Further detail such as different helix diameter was not considered, because it would require knowing the rupture mechanism during installation.

4 Machine Learning Techniques Calibration

The dataset is divided into two stages, named training and test. The first is used to calibrate the machine learning techniques and the second is used to measure their performance. Considering literature recommendations [19], 80% of all examples are used for training and 20% for the testing stage in this work.

The machine learning techniques performance is evaluated using a cross-validation strategy [20]. In this strategy, the training dataset is randomly divided into k folds of the same size. In each cross-validation step, one fold is separated for evaluation without replacement, while the remaining $k-1$ folds are used for training. All folds are evaluated after k steps and the overall performance is considered through the individual performances mean [19]. In this work, $k = 3$ was fixed after initial tests, as presented by Fig.xxxxx . The training dataset is divided into 15 folds for better visualization of the procedure.

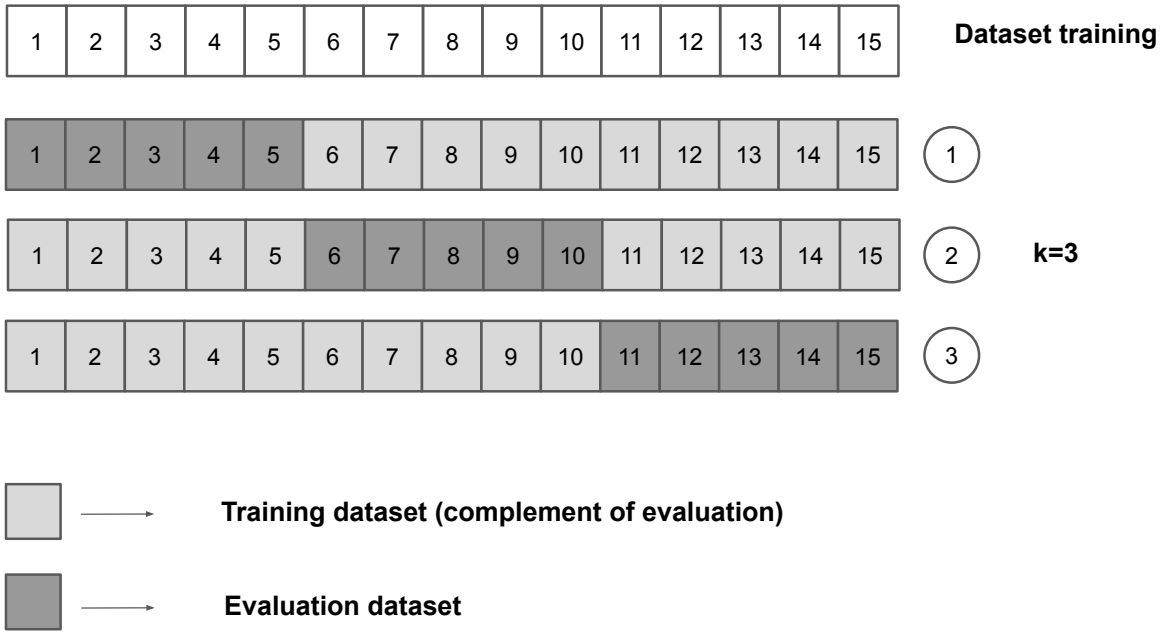

Figure 4: Cross validation with k=3

The same performance measures are used to calibrate the machine learning techniques during training and testing [19]. Next Section presents these measures.

5 Evaluating the Models

After calibration, the machine learning models are tested to estimate torque for new data. The testing dataset, which represents 20% of the main dataset, is kept apart from the training procedures to reproduce practical situations in which the new data is completely unknown. For both, training and test, the metrics described below use inputs x_i to compare the predictions $f(x_i)$ with the known target values y_i.

The mean absolute error MAE is calculated using the absolute difference between

known and predicted values. For n examples, it can be obtained by

$$MAE = \frac{\sum_{i=1}^{n} |y_i - f(x_i)|}{n} \tag{1}$$

The root of the mean squared error $RMSE$ represents the standard deviation of the difference between the observed and predicted values, using

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2} \tag{2}$$

Finally, the coefficient of determination R^2 is a dimensionless measure that represents the correlation between predicted and observed values. It can be written as

$$R^2 = \frac{\sum_{i=1}^{n}(y_i - f(x_i))^2}{\sum_{i=1}^{n}(y_i - \bar{y_i})^2} \tag{3}$$

where

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{4}$$

These measures serve not only to evaluate the performance of the models during the testing stage, but also to evaluate their generalization during the training stage. Very precise values in training when compared to the test can indicate overfitting, which means that even noisy data from the training dataset is being reproduced by the model. An over specialized model tends to become inaccurate when applied to new data. On the other hand, the model losing trends during training indicates underfitting. It means the model is ignoring important information from the training dataset, which also compromises generalization.

Models are calibrated in training using hyperparameters, which are specific configurations for each machine learning technique. One strategy to set these hyperparameters is training several models with random values and choose the one with the best accuracy. However, this strategy is not practical and tends to result in high computational cost. An

alternative is the so called “pick the best” technique [18], which randomly chooses different parameters in training, decreasing the computational cost to adjust the parameters.

6 Multiple Linear Regression – LM

Linear regressions work minimizing error to approximate data using linear functions. Eq. 5 represents its general form:

$$f(x_i) = \alpha + \beta.x_i \tag{5}$$

where x_i represent the input, $f(x_i)$ is the output and α and β represent parameters to be calibrated. The best fit corresponds to the solution of an optimization problem for minimizing error. Using the least squares technique, the error J is measured using Eq. 6:

$$J = \sum_{i=1}^{n} (y_i - f(x_i))^2 \tag{6}$$

where n is the number of examples. Fig. **??** illustrates an example of a linear regression, with the plane representing the approximation, the points the observations and the distance between them the minimized errors $y_i - f(x_i)$.

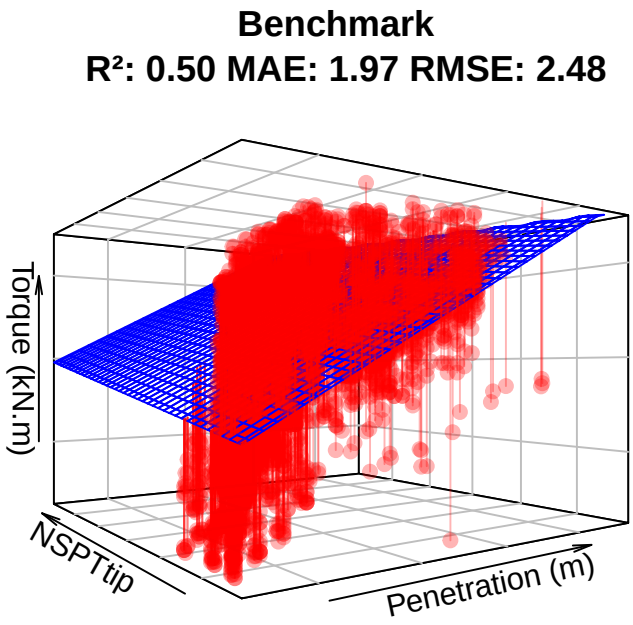

Figure 5: Multiple linear regression representation

The concept is easily extendable to a general number of inputs. More advanced techniques related to linear regression are: partial least squares, least absolute shrinkage and selection operator, ridge regression and least angle regression [21].

In this work, LM is used as one of the references to evaluate the predictive performance of the machine learning models.

The concept is easily extendable to a general number of inputs. More advanced techniques related to linear regression are: partial least squares, least absolute shrinkage and selection operator, ridge regression and least angle regression [21].

In this work, LM is used as one of the references to evaluate the predictive performance of the machine learning models.

Table 2: Eight models used for torque prediction

Algorithm	Method
ANN	nnet
DT	rpart
KNN	kknn
RF	rf
SVM	svmRadial
BAG	bagEarth
BOO	xgbLinear
CUB	cubist

7 Machine learning

Table 2 presents all machine learning techniques used in this work, being ANN for artificial neural networks, DT for decision tree, KNN for k-nearest neighbor, RF for random forest, SVM for support vector machines, BAG for DT associated with bagging, BOO for LM associated with boosting, and CUB for cubist. The R Caret package method used for each technique is also presented.

The following sections provide a brief description of each of these algorithms. Input scaling to the $[0, 1]$ interval is mandatory for KNN and ANN, and optional for all other techniques. No output scaling was used.

8 DT and CUB

A DT is a unidirectional graph that starts at a root node, proceeds to decision nodes, which divide the dataset using predefined rules and ends at leaf nodes, where a value is assigned to the output [22]. When used for regressions, the model uses error measures during training to define the tree architecture and calibrate the rules to be used in decision nodes. Fig. **??** presents a DT example that uses inputs P and $N1$. The number of samples received by each node is indicated as n, together with the percentage that it represents among all samples. Note that all leaf nodes percentage sum is 100%.

```
## Warning: package 'rpart.plot' was built under R version 3.6.3
```

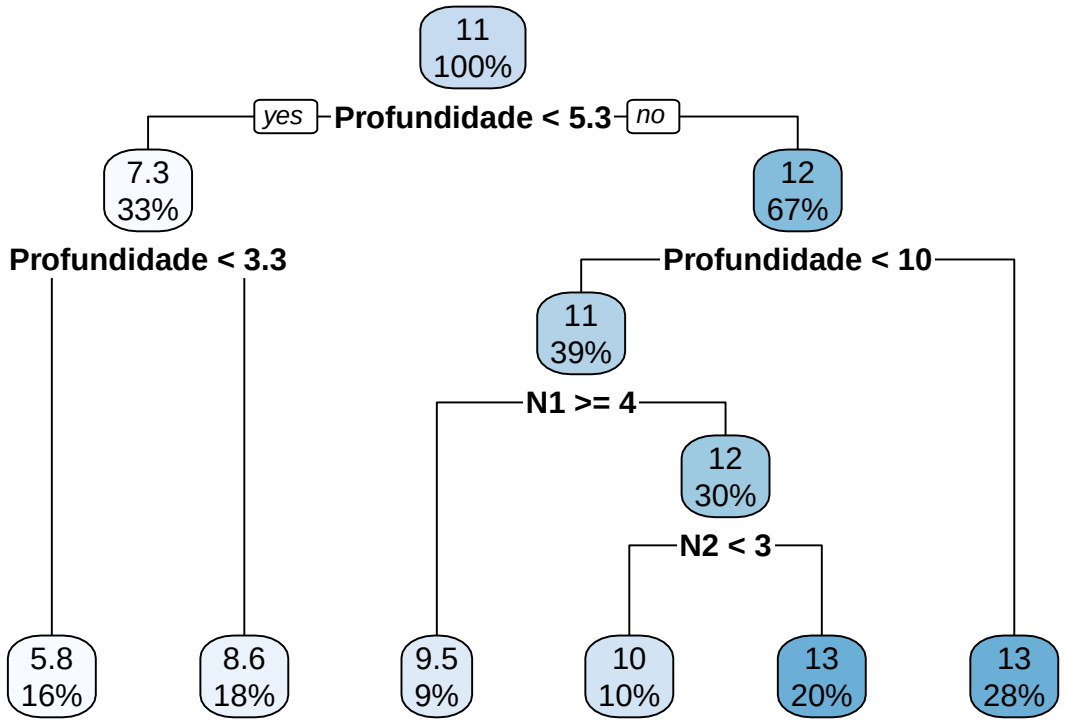

Figure 6: Example of a decision tree

Most decision trees are binary, meaning that each decision node distributes examples exactly to two nodes. The rule used in each node should be calibrated to maximize the precision of the DT, considering first the leaf node and then the previous ones recursively.

DT models have the advantage of being interpretable and their main disadvantage is overfitting. In this work this is avoided by pruning the trees, which is a procedure that eliminates redundant and irrelevant branches. They are also associated with linear regressions for the tests performed at each node, which leads to the CUB model [23]. Linear regressions are calibrated considering the previous nodes prediction, recursively to the root node. Pruning also applies to the CUB model.

9 KNN

The k-nearest neighbors is a non-parametric machine learning technique, meaning that the model is determined from the existing dataset structure [24]. It includes the following steps:

```
## Warning: package 'FNN' was built under R version 3.6.3
```

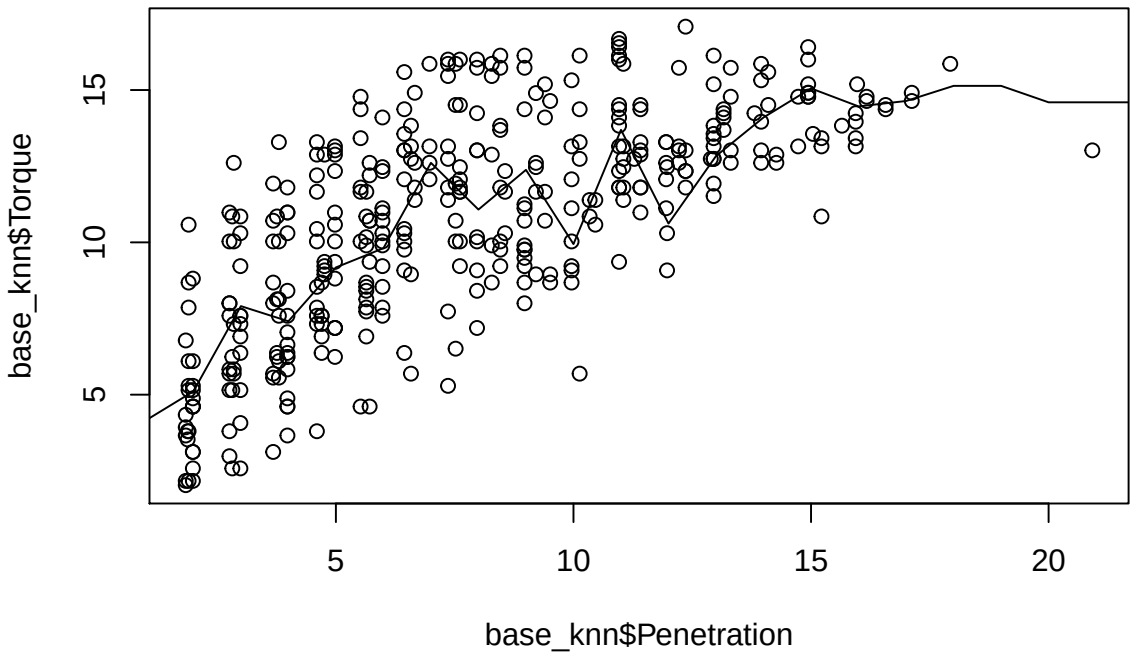

Figure 7: Example of KNN Torque x Penetrarion

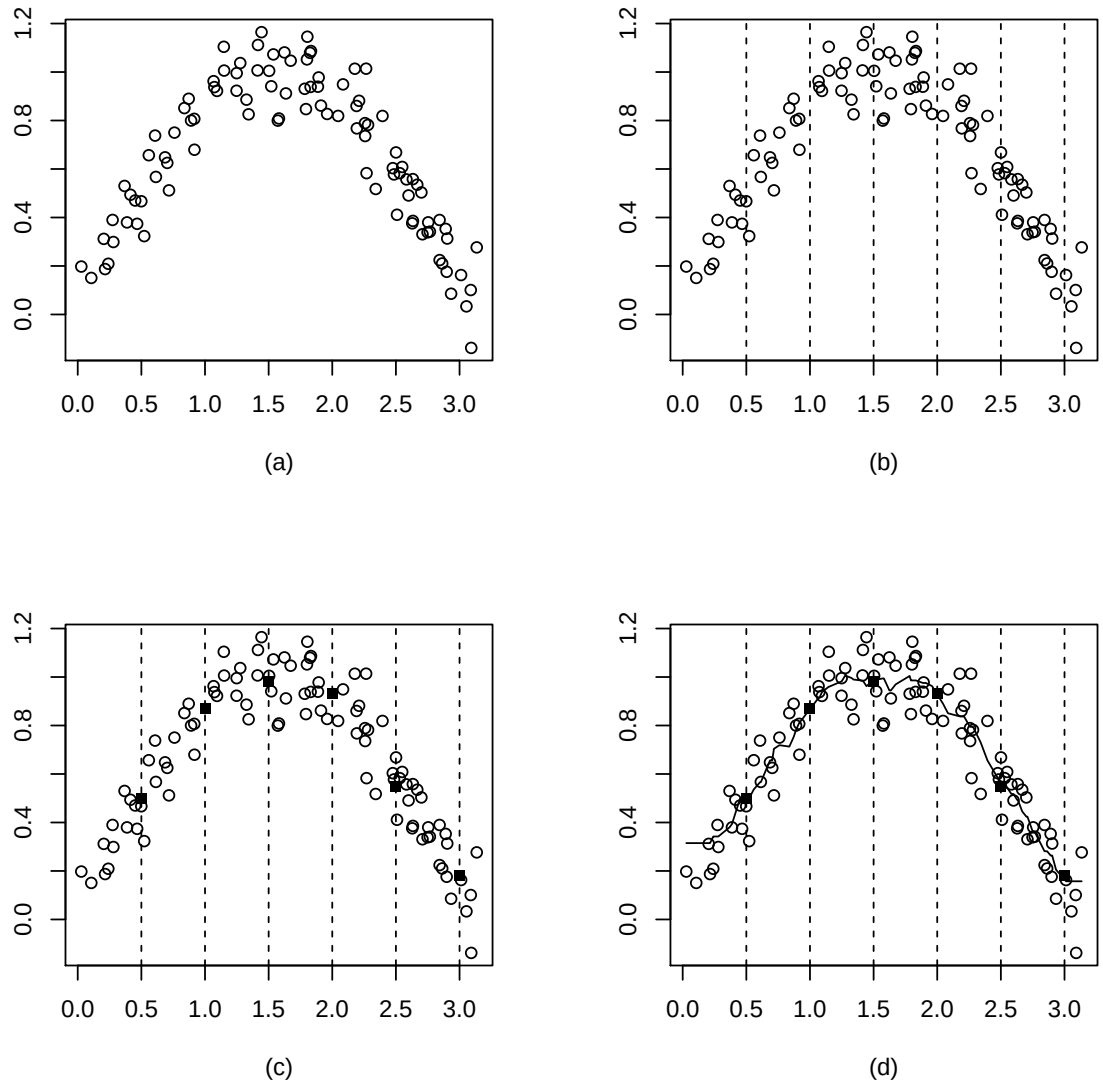

Figure 8: KNN - (a) training dataset available; (b) value range (c) calculation of new points depending on the value of k; (d) KNN regression

1. The model starts with the training dataset Fig. 8(a);

2. It includes value ranges to predict new points Fig. 8(b);

3. Selects the k nearest neighbors (in this example $k = 3$) from the range limits Fig. 8(c) and

4. The mean value of all neighbors is given to the new unknown point 8(d).

The inputs are used as coordinates to calculate the distance to the nearest

neighbors. The Minkowsky metric is popular in the literature and uses the following expression:

$$d\left(A,B\right)=\left(\sum_{i=1}^{dim}\left|a_i-b_i\right|^{exp}\right)^{1/exp} \tag{7}$$

where A and B are points between which distance d is calculated, dim is the space dimension, a_i and b_i are the i^{th} coordinates of points A and B, respectively, and exp is a parameter to be chosen. This work uses the Euclidean distance, with $exp = 2$. One way of improving accuracy is calculating a weighted average by the distance of each neighbour value to the new example.

10 ANN

Neural networks are inspired in the human neurological system. The basic process of an artificial neuron is shown in Fig. 9.

The neuron receives x_i inputs, or signals, weighted by w_i. These contributions sum is subjected to an activation function f_{ativ} that gives the neuron output y, which can be an input to another neuron. Connections between neurons and their weights determine the architecture of the ANN [25].

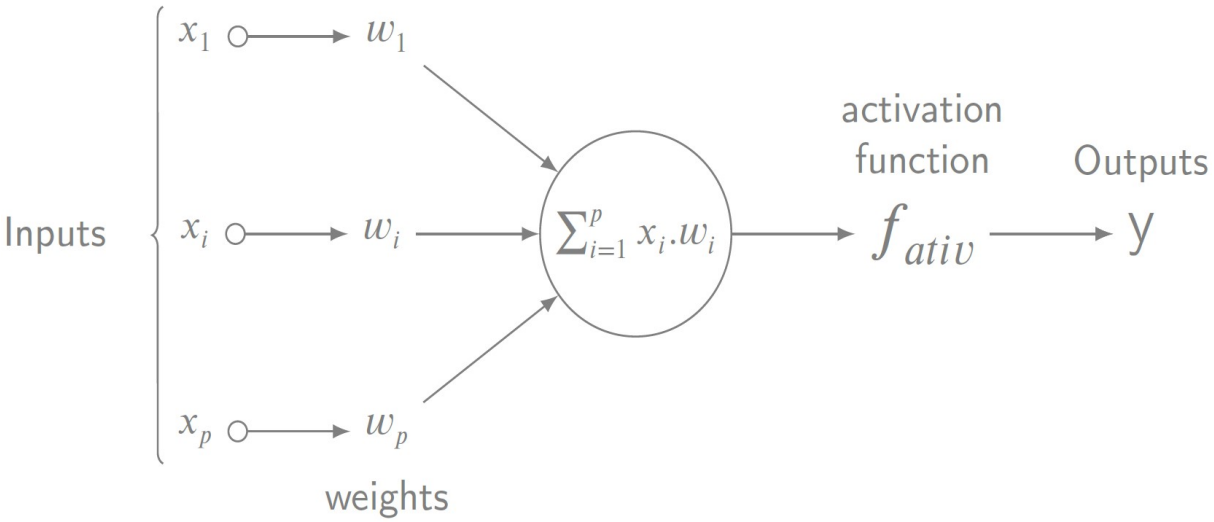

Figure 9: Neuron

The sigmoid function is usually used for f_{ativ}. For a total input u and a calibration parameter λ, it is given by

$$f_{ativ}(u) = \frac{1}{1 + e^{-\lambda u}} \tag{8}$$

The training of an ANN model is based on adapting the architecture to minimize predictive error. Neurons are normally organized in layers, including an input layer, one or more hidden layers and an output layer. This is shown in Fig. 10.

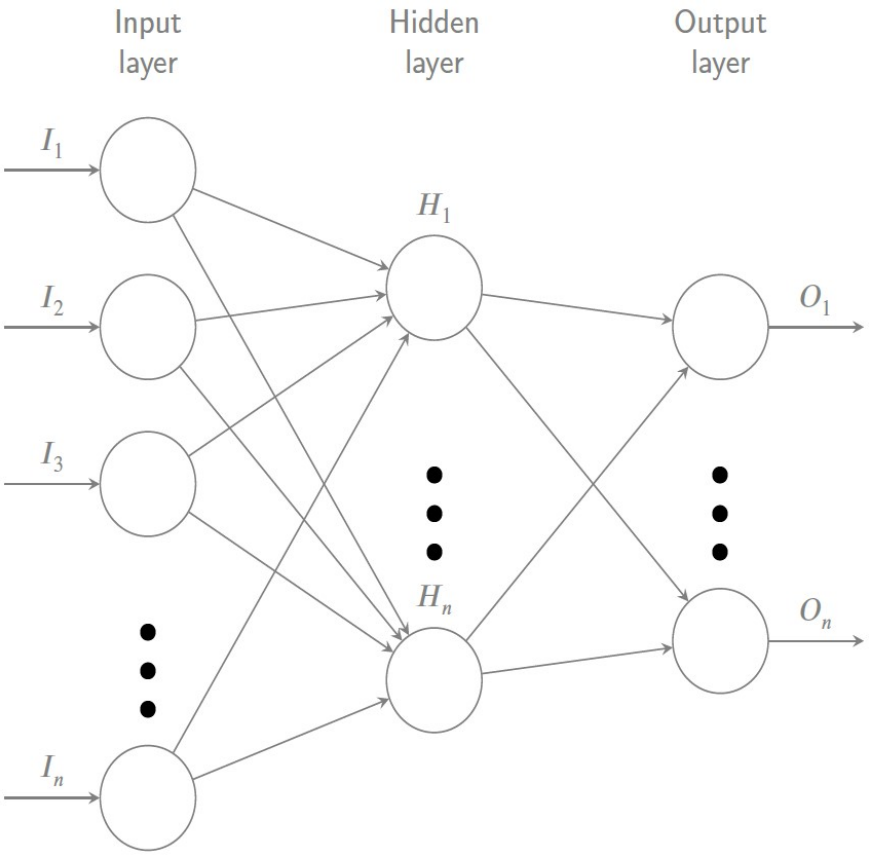

Figure 10: Neural network

The input layer contains one neuron for each input I_i and feeds the first hidden layer. There is no limit for the number of hidden layers or neurons H_i used for each of them, except computational cost. In classification problems, the output layer contains one neuron for each possible output O_i and only one neuron for regression problems [26].

An ANN can approximate any linear function with a single neuron [27], any continuous function with only one hidden layer and any function with two or more hidden layers [**?**].

11 SVM

SVM basic idea is creating hyper planes separating positive and negative data, ensuring global minimums and maximizing the model generalization [28]. In regression problems, boundary lines create a margin around the hyper plane in order to contain all points of the training dataset, and support vectors are the data points placed at the

boundary lines.

Considering a margin ϵ, the hyper plane and boundary lines can be formulated by:

$$f(x) = wx + b \pm \epsilon \tag{9}$$

The solution of the optimum hyper plane that minimizes ϵ is unique. It is possible to smooth the margins, allowing some training points be outside the margin limits. It contributes to avoid overfitting. It is also possible to embrace non-linear problems, like the one presented in Fig. **??**, by the use of kernels - functions that map the input space into a higher dimension space. The objective is obtaining a problem in this higher dimension that can be evaluated using a linear SVM, as shown in Fig. 12.

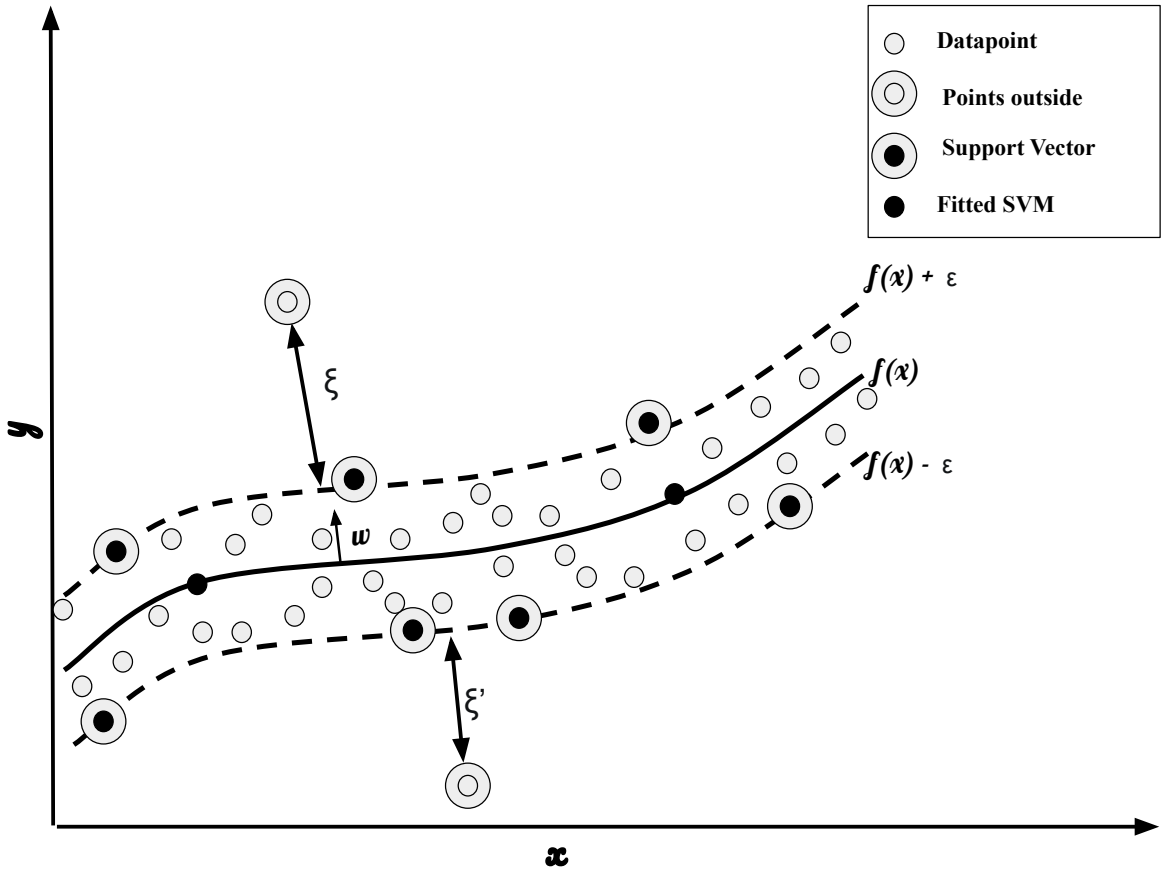

Figure 11: SVM regression

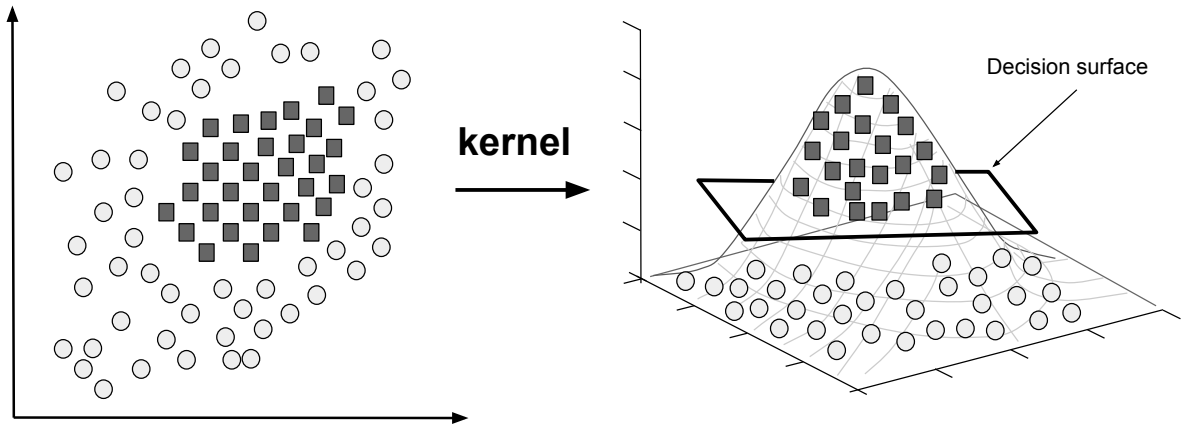

Figure 12: Kernel using

Penalties are imposed upon training data outside the limits, considering their distances to the margin (ξ and ξ' in Fig. 11). These penalties are controlled by a C parameter, using the following expression:

$$\frac{1}{2}||w||^2 + C\sum_{i=1}^{N}[\xi + \xi'] \tag{10}$$

Model generalization is increased minimizing $||w||^2$ and C weights the training error.

In the literature, the most popular kernels use linear, polynomial or radial functions. In this work, after preliminary tests, the polynomial kernel was chosen. It is given by

$$(\delta\,(x_i \cdot x_j) + \kappa)^{\zeta} \tag{11}$$

where δ, κ and ζ are calibration parameters.

12 Ensemble Techniques

Ensemble techniques work combining two or more predictors. Such definition embraces three techniques used in this work: bagging (BAG), random forest (RF) and boosting (BOO).

[29] presented BAG using a bootstrapping approach, which collects samples from the training dataset with replacement and, therefore, trains the model with each sample. For regression, the mean value obtained from all trained models gives the final result. This procedure tends to improve accuracy by reducing variation within unstable models. BAG can be combined with any regression model. In this work, references to BAG mean bagging associated to DT.

The RF technique uses DT improved by BAG, the difference is selecting random input combinations for each BAG technique model [30, 31]. The random inputs number is a calibration parameter.

BOO technique basic idea is focusing on the model weaknesses to make it stronger [32]. The model is trained repetitively. During each run, samples which the model is inaccurate are identified then additional weight is given to these samples for the next run. The final model is a linear combination of all trained models, so that additional weight is given to more accurate models. The BOO technique can also be combined with any regression model; however, in this work, BOO references mean boosting applied to LM.

13 Ensemble techniques

Ensemble techniques work combining two or more predictors. Such definition embraces three techniques used in this work: bagging (BAG), random forest (RF) and boosting (BOO).

[29] presented BAG using a bootstrapping approach, which collects samples from

the training dataset with replacement and, therefore, trains the model with each sample. For regression, the mean value obtained from all trained models gives the final result. This procedure tends to improve accuracy by reducing variation within unstable models. BAG can be combined with any regression model. In this work, references to BAG mean bagging associated to DT.

The RF technique uses DT improved by BAG, the difference is selecting random input combinations for each BAG technique model [30, 31]. The random inputs number is a calibration parameter.

BOO technique basic idea is focusing on the model weaknesses to make it stronger [32]. The model is trained repetitively. During each run, samples which the model is inaccurate are identified then additional weight is given to these samples for the next run. The final model is a linear combination of all trained models, so that additional weight is given to more accurate models. The BOO technique can also be combined with any regression model; however, in this work, BOO references mean boosting applied to LM.

14 Results and duscussion

All results presented in this section refer to the regression techniques applied to Initial Dataset, Complete Data-set and Contribution Dataset, all of them described in Section 2.1. They resulted in 7632 observations after the pre-processing stage, which were divided into training dataset (with 6108 6,108 observations, 80%) and test data-set (with 1524 observations, 20%). All models are evaluated using the metrics explained in Section 2.3, named MAE, $RMSE$ and R^2, along with "pick the best" approach.

Table 3: Test stage for Initial Dataset

Algoritmo	MAE	RMSE	Rsquared	Rank
ANN	1.79	2.29	0.58	6
BAG	1.81	2.29	0.58	7
BOO	1.59	2.05	0.66	1

Algoritmo	MAE	RMSE	Rsquared	Rank
CUB	1.63	2.09	0.65	3
DT	1.90	2.39	0.54	8
KNN	1.68	2.19	0.62	4
LM	1.97	2.48	0.50	9
RF	1.61	2.07	0.66	2
SVM	1.70	2.22	0.60	5

Table **??** presents the evaluation metrics obtained during the eight ML techniques training stage. This analysis used the Initial Dataset and includes values obtained with LM for comparison purposes. The testing stage results shown in Table **??** are analogous to the training stage results. For both tables, column Rank orders the regression models from the greatest to the lowest R^2 value. Each technique performance is similar when comparing the training and test stages, which is an expected behavior.

Table 4: Test stage for Initial Dataset

Algoritmo	MAE	RMSE	R2	Rank
ANN	1.82	2.32	0.57	6
BAG	1.83	2.34	0.56	7
BOO	1.58	2.06	0.66	1
CUB	1.64	2.16	0.63	3
DT	1.92	2.44	0.52	8
KNN	1.70	2.27	0.59	4
LM	1.99	2.50	0.50	9
RF	1.60	2.08	0.65	2
SVM	1.73	2.27	0.59	5

In both tables, LM was inferior than all ML techniques, which indicates the torque

non-linearity and justifies ML techniques usage. DT model showed the second poorest performance, which is explained by its simple form, since RF and CUB, which are improved techniques also based on trees, resulted in the second and third greater R^2 respectively. BOO model completes the top three list of models with the best performance in both training and test stages. It is worth noting that BOO is a LM technique associated to boosting.

After that, an analysis was undertaken to evaluate the performance of the models in training and test regarding to the Complete Dataset. Based on training results shown in Table **??** and comparing to Table **??**, all models had an improved performance on Complete Dataset. Similar conclusion is drawn comparing Table **??** to Table **??**, which are valid for test stage.

For Complete Dataset, each model performance was similar again when the results of both training and testing stages are evaluated. LM and DT models had the poorest performance, exhibiting very close values for the evaluation parameters. The other tree-based models, RF and CUB, performed well again, which emphasizes that DT model poor performance was due to its simple form. CUB model showed the best performance in both stages, followed by SVM in the training stage and BOO in the test stage.

Table 5: Training stage for Complete Datase

Algoritmo	MAE	RMSE	Rsquared	Rank
ANN	1.41	1.81	0.74	6
BAG	1.46	1.83	0.73	7
BOO	1.19	1.61	0.79	3
CUB	1.15	1.52	0.81	1
DT	1.84	2.30	0.57	9
KNN	1.22	1.64	0.78	5
LM	1.77	2.23	0.60	8
RF	1.19	1.61	0.79	4

Algoritmo	MAE	RMSE	Rsquared	Rank
SVM	1.17	1.58	0.80	2

Table 6: Test stage for Complete Dataset

Algoritmo	MAE	RMSE	R2	Rank
ANN	1.35	1.77	0.75	6
BAG	1.46	1.85	0.72	7
BOO	1.15	1.57	0.80	2
CUB	1.11	1.51	0.82	1
DT	1.81	2.28	0.58	9
KNN	1.21	1.66	0.78	5
LM	1.77	2.23	0.60	8
RF	1.16	1.59	0.80	3
SVM	1.15	1.59	0.80	4

As presented in Section 2.3, this work employs the "pick the best" technique to calibrate ML techniques hyperparameters. To verify if their full potentialities are being explored, a more careful hyperparameter selection was performed only for CUB, BOO, and RF, which are the models that better performed on test stage using the Complete Dataset. Compared to previous analysis results (Table **??**), no significant improvement was noticed.

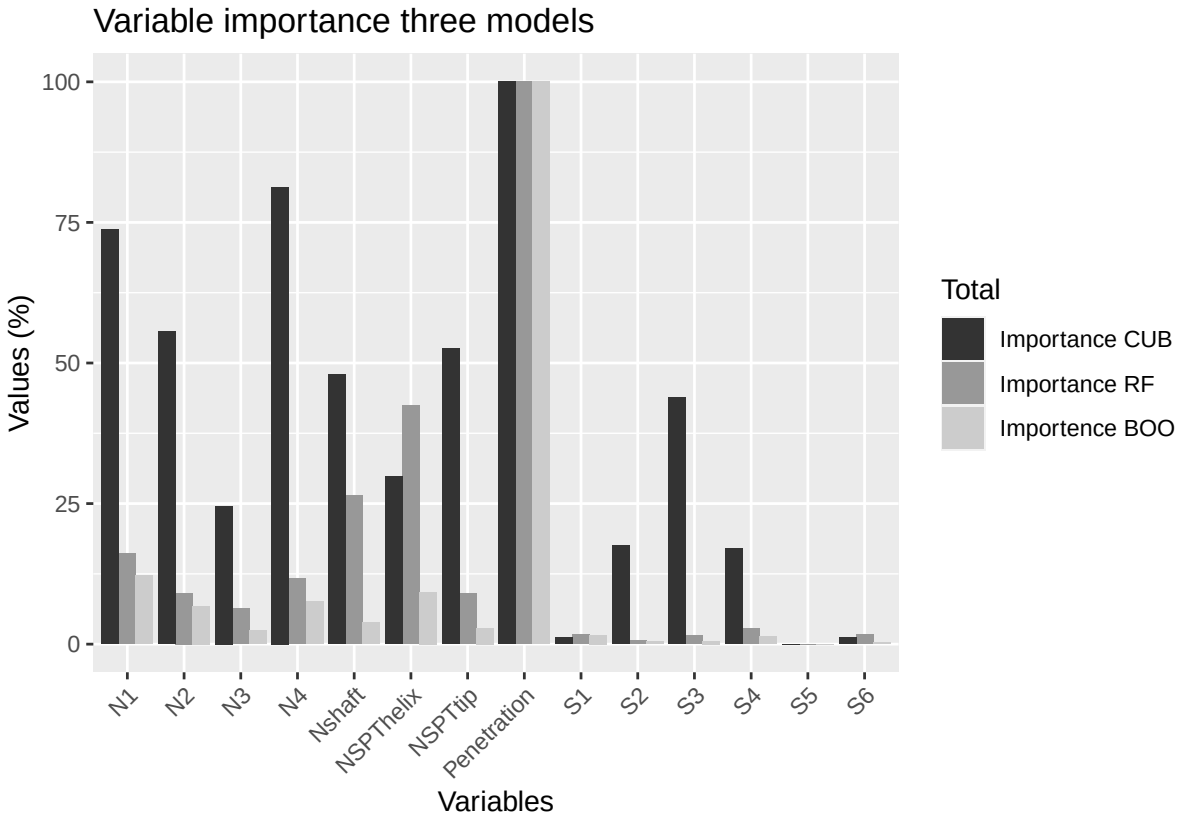

Figure 13: Variabel importance for three models

Fig. **??** presents the variable importance (according to **??** item of this work) determined for the three best performing models in the test stage with Complete Dataset (BOO, CUB and RF). The pile depth (P) is the most important variable for the three models and, therefore, was used in all predictions (100%). P prevalence importance is clearer for BOO, considering that the second most important variable for this model is $N1$, which was used in approximately 13% of all predictions. On the other hand, the CUB model showed the most balanced usage of variables, with 8 variables being used in more than 40% of all predictions.

The soil type, which is represented by $S1$, $S2$, $S3$, $S4$, $S5$ and $S6$, had little importance for the three models. This can be explained by the predominance of clayey sand within the dataset (65.8%), not allowing much difference between examples. Considering that irrelevant data can jeopardize the model performance, a new dataset named Contribution Dataset was created excluding the soil type variable from the Complete

Dataset. Table **??** presents the accuracy of the models BOO, CUB and RF obtained with the new dataset in the training stage. Table **??** indicates that similar results were obtained from test stage. Comparing these tables with the ones obtained using Complete Dataset (see Table **??** and Table **??**) it can be noticed very similar accuracies, with a slight improvement of BOO and RF and a slight deterioration of CUB. It is worth noting that reducing the number of inputs reduces computational cost. This further indicates that the soil type information has low relevance in this problem, which justifies the removal from the dataset.

Table 7: Training stage for Contribution Dataset

	Algoritm	MAE	RMSE	Rsquared
25	BOO	1.18	1.59	0.80
13	CUB	1.14	1.52	0.81
2	RF	1.17	1.56	0.80

Table 8: Test stage for Contribution Dataset

Algoritmo	MAE	RMSE	R2
BOO	1.14	1.56	0.80
CUB	1.08	1.47	0.83
RF	1.14	1.55	0.81

Fig. **??** shows torque measured values compared with the predictions made with RF, BOO, and CUB models for test stage using Contribution Dataset. Considering that the dashed line represents predictions equal to measured values, it can be concluded that the three models final accuracy is reasonable.

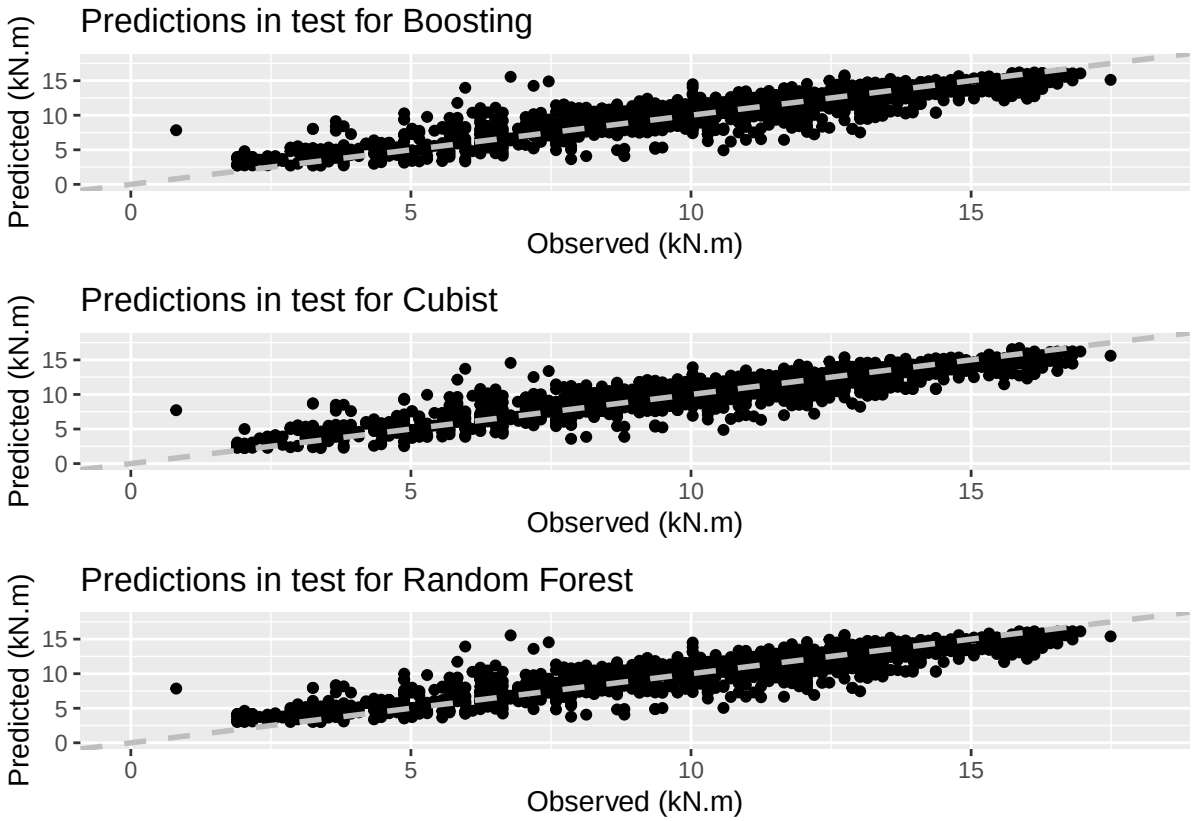

Figure 14: Predicted and observed values.

A complement to the above analyses is given with the definition of a confidence interval. Confidence interval is a data percentage expected to be within a given value range. This range is here proposed using a factor given by the ratio of the predicted torque to the measured torque, as presented in Eq. (12)

$$Factor = \frac{Predicted}{Observed} \tag{12}$$

This factor was calculated for all 6108 xxxxxxxxx 7,632 predictions obtained using RF, BOO and CUB models, with the results organized in ascending order to construct a histogram for each model. Fig. **??** presents the histograms which show the frequency distribution of these data in intervals of 0.01.

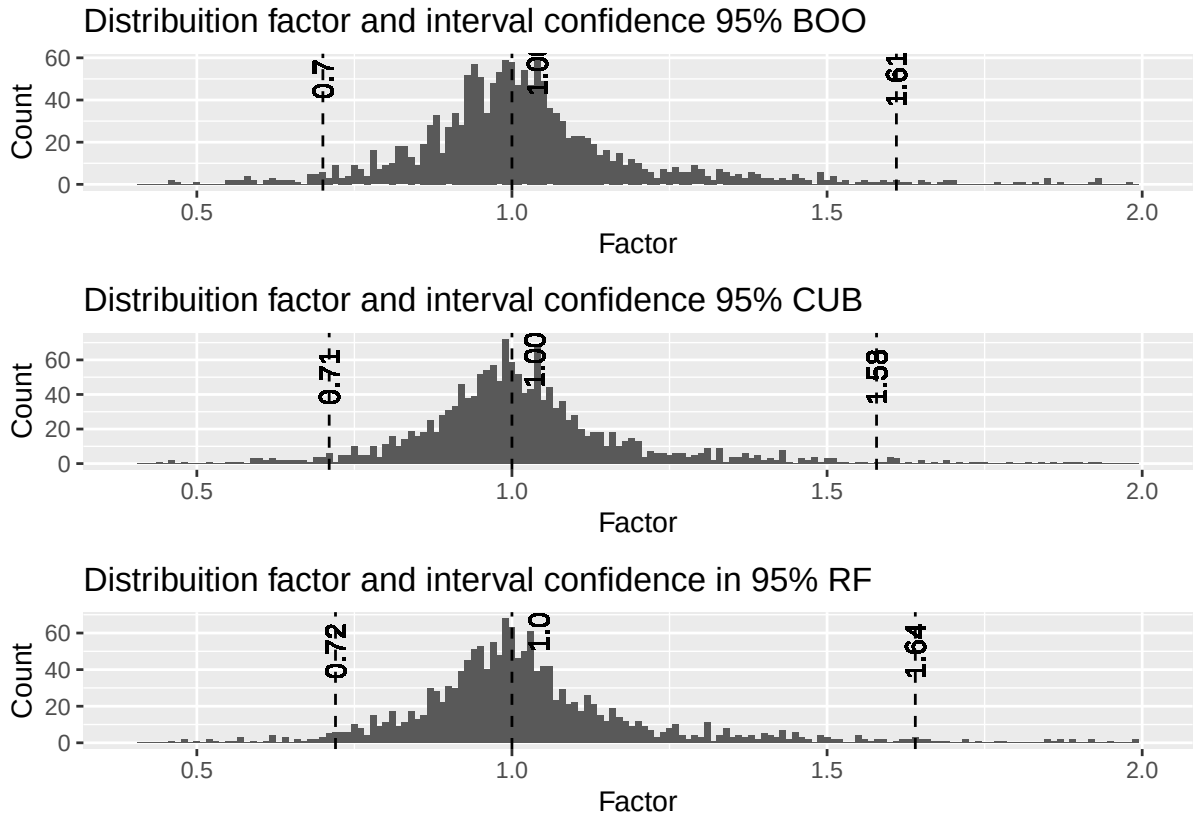

Figure 15: Factor distribution for RF, BOO and CUB

It can be observed that at the median, represented by the middle dashed line, the factor is equal to 1.0 for the three models. It means that the predicted value is equal to the observed one at the median. Other dashed lines represent the 0.25% percentile on the left and the 97.5% percentile on the right, which means that these lines contain 95% of all factors. This corresponds to a 95% confidence interval.

For RF model, 95% of all predicted values are between 0.72 and 1.64 times the observed value, which means there is 95% confidence that the interval $[0.72, 1.641.64]$ will contain the factor for RF. For BOO and CUB models, the confidence intervals are $[0.7, 1.61]$ and $[0.71, 1.58xxxxx1.61]$, respectively. Table **??** presents these intervals size. CUB model has the smallest interval size, which means it is the most accurate.

Table 9: 95% confidence intervals

Algorithm	Left	Right	Size
BOO	0.70	1.61	0.91
CUB	0.71	1.58	0.87
RF	0.72	1.64	0.92

In this work, 95% confidence interval was chosen because most applications in statistics use this percentage. However, other values could be used depending on the engineering application. Smaller confidence percentages such as 90% or 80% would lead to smaller interval sizes and vice-versa.

15 Case Study

This case study uses one pile installation case from the Contribution Dataset to predict the installation torque at each meter depth aiming to determine the maximum pile length based on a limit torque value, which is dependent on the installation equipment capacity. A limit torque value of 13 kN×m was chosen based on the available machinery low capacity in Brazilian market for helical pile installation. Table **??** presents the results obtained with CUB model trained with Contribution Dataset, which is one of the models that led to the best results, as shown in the previous sections. Notice that all required inputs can be deduced from SPT index and its related depth. The limits of the intervals were calculated using Table **??** values, the left limit was given by 0.69 times the predicted value and the right limit by 1.56 times the predicted value. The first column starts with $L = 2$ m because torque is not measured for $L = 1$ m.

Table 10: Results for the case study

L (m)	NSPTtip	Obs (kN.m)	Pred (kN.m)	Tmin (kN.m)	Tmax(kN.m)
2	2	6.10	4.64	3.20	7
3	3	7.86	8.12	5.60	13
4	6	9.08	10.41	7.18	16
5	6	9.63	11.50	7.93	18
6	7	11.66	12.51	8.63	20
7	6	13.15	13.48	9.30	21
8	7	13.69	14.64	10.10	23

In this example, CUB model seems to become more accurate when the pile length reaches 4 m (the shortest piles in the the datasets of the current work had 8 m length). Considering that the pile bearing capacity is usually estimated using the final torque during installation, higher accuracy with the increase in pile length can be considered an advantage. Moreover, if the machinery for pile installation has 13 kN×m maximum torque capacity, the pile will not reach 7 m length in the field. Supposing that in a preliminary design stage only SPT results are available, Table **??** indicates that similar conclusion could be reached with CUB model usage. In other words, the maximum pile length would be 6 m for both observed and predicted values.

16 Conclusions

Eight different machine learning techniques were applied to predict helical piles installation torque, using a dataset with 9,355 torque measurements and SPT data from the pile location. After pre-processing procedures, 7,632 data samples remained, from which three different datasets were produced. The dataset which provided the best results was constructed based on the importance of the variables, only informative inputs are selected in order to achieve the best performance with reasonable computational cost. Thus, this

work used “pick the best” technique, which was compared with the procedure of careful hyperparameter selection and have shown equivalent accuracy with less computational cost.

For the most informative dataset, the most accurate prediction was obtained with cubist (CUB), followed by random forest (RF) and boosting (BOO). The most important input for the three techniques was helical pile vertical penetration; the soil type was considered low importance, which is explained by dataset low variability (65.8% cases correspond to sandy soil). Nevertheless, as in the case of torque-to-capacity correlations used in practice (e.g. [5]), one advantage of eliminating the soil type is subjectivity reduction, since all remaining inputs can be deduced from SPT index with its respective depth.

A complement to the main analysis was presented using confidence interval concept. It was shown that 95% of CUB model predictions are between 0.69 and 1.56 times the observed measurements. To better illustrate the application of this range in real engineering projects, one pile from the dataset was used in a case study aiming at predicting the pile installation length. The CUB model resulted in a maximum pile length similar to the one which would be expected in field. Thus, confidence intervals were provided to assist the project planning. This approach tends to be more informative than simply providing a safety factor because it includes not only a mean value, but also standard deviation information.

17 Data Availability Statement

- Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request (R codes used to obtain the results).
- Some or all data, models, or code used during the study were provided by a third party (helical piles installation reports). Direct requests for these materials may be made to the provider as indicated in the Acknowledgements.

18 Acknowledgements

To Cristina de Hollanda Cavalcanti Tsuha for making available the dataset used in this work and Bruno Oliveira da Silva for helping with data curation and the Coordination of Improvement of Higher Education Personnel (CAPES) for granting the first author's scholarship.

References

[1] MITSCH, M. P.; CLEMENCE, S. P. Uplift capacity of helix anchors in sand. In: CLEMENCE, S. P. (Ed.). *Uplift Behavior of Anchor Foundations in Soil*. New York: American Society of Civil Engineers, 1985. p. 26–47. Disponível em: <http://www.scopus.com/inward/record.url?scp=0022223105partnerID=8YFLogxK>.

[2] GHALY, A.; HANNA, A.; HANNA, M. Uplift behavior of screw anchors in sand. i: dry sand. *J. Geotech. Eng.-ASCE*, American Society of Civil Engineers, v. 117, n. 5, p. 773–793, 1991.

[3] LUTENEGGER, A. J. Screw piles and helical anchors–what we know and what we don't know: an academic perspective–2019. In: *Proc. 1st Int. S. on Screw Piles for Energy Applications*. Dundee, United Kingdom: University of Dundee, 2019. p. 15–28.

[4] ELSHERBINY, Z. H.; NAGGAR, M. H. E. Axial compressive capacity of helical piles from field tests and numerical study. *Can. Geotech. J.*, NRC Research Press, v. 50, n. 12, p. 1191–1203, 2013.

[5] HOYT, R. M.; CLEMENCE, S. P. Uplift capacity of helical anchors in soil. In: *Proc., 12th Int. Conf. on Soil Mechanics and Foundation Engineering*. Rio de Janeiro, Brazil: [s.n.], 1989. p. 1–7. Disponível em: <http://pascal-francis.inist.fr/vibad/index.php?action=getRecordDetailidt=6664015>.

[6] TSUHA, C. H. C.; FILHO, J. M. S. M. dos S.; SANTOS, T. C. Helical piles in unsaturated structured soil: a case study. *Can. Geotech. J.*, NRC Research Press, v. 53, n. 1, p. 103–117, 2015.

[7] TSUHA, C. H. C. *Modelo teórico para controle da capacidade de carga à tração de estacas metálicas helicoidais em solo arenoso [theoretical model to control on site the uplift capacity of helical screw piles embedded in sandy soil]*. Tese (Doutorado) — Universidade de São Paulo, São Carlos, SP, Brazil, November 2007. (in Portuguese). Disponível em: <https://teses.usp.br/teses/disponiveis/18/18132/tde-06052008-151518/en.php>.

[8] LUTENEGGER, A. J. Historical development of iron screw-pile foundations: 1836–1900. *Int. J. Hist. Eng. Tech.*, Taylor and Francis, v. 81, n. 1, p. 108–128, 2011.

[9] PERKO, H. A. *Helical piles: a practical guide to design and installation*. New Jersey, USA: John Wiley and Sons, 2009.

[10] WANG, B. et al. Feasibility of a novel predictive technique based on artificial neural network optimized with particle swarm optimization estimating pullout bearing capacity of helical piles. *Eng. with Comput.*, Springer, v. 36, p. 1315–1324, September 2020.

[11] SPAGNOLI, G. et al. Estimation of uplift capacity and installation power of helical piles in sand for offshore structures. *J. of Waterw., Port, Coast., and Ocean Eng.*, v. 144, n. 6, p. 04018019, November 2018.

[12] SILVA, B. O. *Estimativa do torque de instalação de fundações por estacas helicoidais por meio de resultados de ensaio SPT [estimation of the installation torque of helical piles using SPT data]*. Dissertação (M.S. thesis) — Universidade de São Paulo, São Carlos, SP, Brazil, 2018. (in Portuguese). Disponível em: <https://www.teses.usp.br/teses/disponiveis/18/18132/tde-20122018-160049/pt-br.php>.

[13] ROSS, J. L. S.; SANTOS, L. Geomorfologia. In: *Projeto Radam Brasil, Folha Cuiabá CD [project Radam Brazil, Cuiabá sheet CD]*. [S.l.]: Ministério das Minas e Energia, 1982. v. 21, p. 222. (in Portuguese).

[14] BARROS, A. M. et al. Folha sd. cuiabá [sd cuiabá sheet]. In: *Projeto Radam Brasil [project Radam Brazil]*. [S.l.]: Ministério das Minas e Energia, 1982. v. 21. (in Portuguese).

[15] ABNT. Solo - sondagens de simples reconhecimento com spt - método de ensaio [soil - standard penetration test - spt - soil sampling and classification]. In: *NBR 6484*. Rio de Janeiro, Brazil: ASTM, 2001. (in Portuguese).

[16] ASTM. Standard test method for standard penetration test (spt) and split barrel sampling of soils. In: *D1586 / D1586M - 18*. [S.l.]: ASTM, 2008.

[17] LUKIANTCHUKI, J. A.; BERNARDES, G. P.; ESQUIVEL, E. R. Energy ratio (e^r) for the standard penetration test based on measured field tests. *Soils and Rocks*, Brazilian Association for Soil Mechanics and Geotechnical Engineering and Portuguese Geotechnical Society, v. 40, n. 2, p. 77–91, 2017.

[18] KUHN, M.; JOHNSON, K. *Applied predictive modeling*. [S.l.]: Springer, 2013. v. 26.

[19] JAMES, G. et al. *An introduction to statistical learning: with applications in R*. [S.l.]: Springer, 2013. v. 112.

[20] GOETZ, J. N. et al. Evaluating machine learning and statistical prediction techniques for landslide susceptibility modeling. *Comput. Geosci.*, Elsevier, v. 81, p. 1–11, 2015.

[21] FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. *The elements of statistical learning*. Philadelphia, Pa.: Springer Series in Statistics, 2001.

[22] QUINLAN, J. R. Induction of decision trees. *Mach. Learn.*, Springer, v. 1, p. 81–106, 1986.

[23] QUINLAN, J. R. Combining instance-based and model-based learning. In: *Proc.,* 10^{th} *Int. Conf. on Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993. p. 236–243.

[24] DUDANI, S. A. The distance-weighted k-nearest-neighbor rule. *IEEE T. Syst. Man. Cyb.*, SMC-6, n. 4, p. 325–327, 1976. ISSN 0018-9472.

[25] MASSART, D. L. et al. *Handbook of Chemometrics and Qualimetrics*. 1^{st}. [S.l.]: Elsevier, 1997.

[26] RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *Nature*, v. 323, p. 533–536, 1986.

[27] MINSKY, M. L.; PAPERT, S. A. *Perceptrons: An introduction to computational geometry*. [S.l.]: MIT press, 2017.

[28] VAPNIK, V. N. An overview of statistical learning theory. *IEEE T. Neural Networ.*, IEEE, v. 10, n. 5, p. 988–999, September 1999.

[29] BREIMAN, L. Bagging predictors. *Mach. Learn.*, Springer, v. 24, n. 2, p. 123–140, 1996.

[30] BREIMAN, L. Random forests. *Mach. Learn.*, Springer, v. 45, n. 1, p. 5–32, 2001.

[31] HO, T. K. Random decision forests. In: IEEE. *Proc., 3rd Int. Conf. on Document Analysis and Recognition*. Montréal, Canada, 1995. p. 278–282.

[32] ABNEY, S.; SCHAPIRE, R. E.; SINGER, Y. Boosting applied to tagging and pp attachment. In: *Proc., Joint SIGDAT Conf. on Emp. Meth. Nat. Lang. Proc. V. L. Corpora*. [s.n.], 1999. p. 38–45. Disponível em: <https://www.aclweb.org/anthology/W99-0606>.