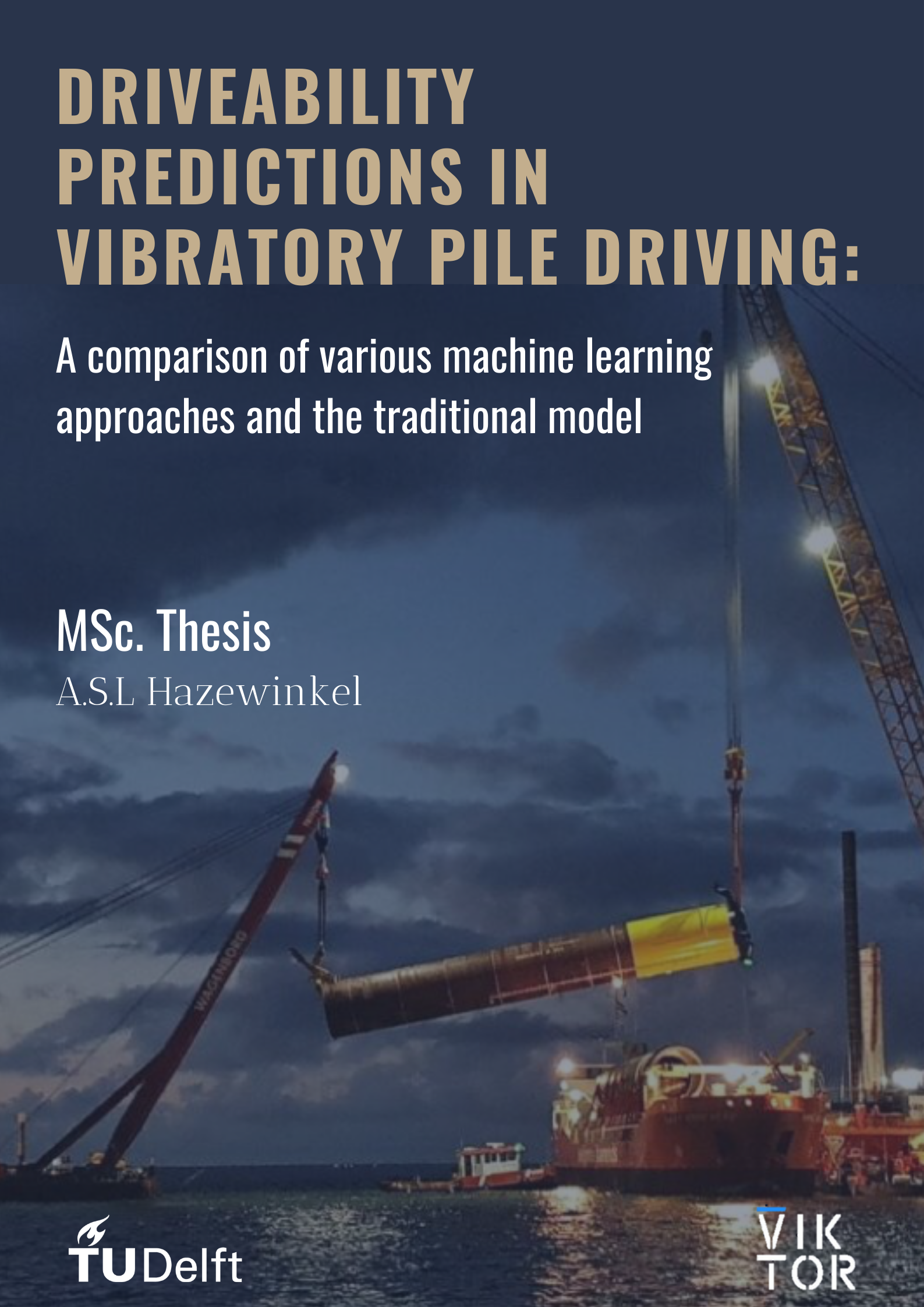
DRIVEABILITY PREDICTIONS IN VIBRATORY PILE DRIVING:

A comparison of various machine learning approaches and the traditional model



MSc. Thesis

A.S.L Hazewinkel

TUDelft

VIKTOR

Delft University of Technology

DRIVEABILITY PREDICTIONS IN VIBRATORY PILE DRIVING:

A comparison of various machine learning approaches and the traditional model

By

A.S.L Hazewinkel
July 18, 2022

To obtain the degree of Master of Science in Robotics at the Delft University of Technology.

Student number: 4478223
Project duration: November 2021 - July 2022
Thesis committee: Prof. dr.ir. J. Hellendoorn, Delft University of Technology, supervisor
M. Slootweg, Viktor, supervisor
Dr. J. Kober, Delft University of Technology
Prof. dr.ir. A. Metrikine, Delft University of Technology

Abstract

Pile driving is a widely used technique for the construction of buildings and infrastructure. A popular technique is to vibrate the pile into the sediment. However, since building sites are increasingly being located in metropolitan areas, there is a growing concern about the environmental impact that vibrations may cause during driving. Therefore, pile driving assessment and prediction have become important for several reasons, including reducing the risk of damage to nearby structures and limiting disturbance to adjacent properties. In particular, pile driveability (i.e., the penetration rate) assessment is immensely useful prior to installation as it increases the construction performance and subsequently reduces costs and environmental impact. The important factors influencing the penetration rate include vibrator characteristics, pile properties, and soil conditions. However, due to assumptions and the lack of methods that accurately represent the complex phenomena at play during vibratory driving, a disparity is obtained between the predictions of modern pile behavior programs and the observed penetration rate.

Recently, the registration of pile driving data has increased significantly. This extended amount of measurement data can potentially be leveraged for an improvement of the prediction of the penetration rate in future projects. Literature review on the application of machine learning (ML) within pile driving, geotechnical engineering and drilling revealed that the artificial neural network (ANN) is a promising alternative method for the prediction of the driveability of vibratory driven piles (i.e., vibro-driveability).

In this work, machine learning methods and the traditional model were utilized to predict vibro-driveability. Promising ML techniques include the multilayer perceptron neural network (MLPNN) and radial basis function neural network (RBFNN). These neural networks were trained with the particle swarm optimization (PSO) algorithm. The backpropagation (BP) algorithm was also incorporated to train the MLPNN and RBFNN models as a conventional method. Based on results obtained with the aforementioned methods, we propose a new model, the Vibratory Driveability (VD) model, that combines the fruitful characteristics of the MLPNN and RBFNN.

The performance of the five different models was compared with the performance of contemporary vibro-driveability prediction software for three test sets. This was done using different performance indices including the mean squared error (MSE), mean absolute error (MAE) and the weighted average percentage error (WAPE). Additionally, the desired characteristics of the predictions based on the geo-engineer's input were examined and compared with the obtained predictions. It was demonstrated that the ANN-based methods achieved drastic improvements in prediction performance and consequently outperformed the traditional model, making ANN-based methods the preferred alternative for the prediction of vibro-driveability. Among the ANN models, the VD model produced the highest performance, as it reflected the desired prediction behavior for all three test cases and showed competitive prediction performance in terms of the performance metrics.

This work leads to the first-ever published research on the application of artificial neural networks for the prediction of vibro-driveability. As such, it could form the foundation for the development of new (vibratory) pile driving behavior assessment and prediction software. The development of these commercial applications could lead to a considerable reduction in costs and environmental impact.

Acknowledgments

This MSc thesis report describes my work over the past year, in which I have been researching "driveability predictions in vibratory pile driving: A comparison of various machine learning approaches and the traditional model.", to conclude my Master of Science in Robotics at Delft University of Technology.

The past year, working on this thesis, has been a challenging but very rewarding experience. Challenging, because my perseverance was put to the test by working individually on this research. And rewarding, because of everything that I have learned over the past year under the guidance of my wonderful supervisors Hans Hellendoorn and Marcel Slootweg. My sincerest appreciation goes to their 24/7 support, guidance and enthusiasm about this topic. The (bi)weekly meetings were always insightful and helped me to take my research and my scientific writing to a higher level. I would also like to thank the companies that made this research possible: application development company Viktor and construction company Voorbij. Their resources allowed me to complete this research.

Furthermore, I would like to thank all my friends and family for patiently listening to my stories about artificial neural networks. I would like to give a special thanks to Edzo Hermans and Merel Hazewinkel for their continuous support and nonstop encouragement. Being able to talk to them about my excitement for this project kept my motivation going and always stirred up new ideas.

Annabel Hazewinkel

Contents

Abstract i

Acknowledgments iii

List of Figures viii

List of Tables ix

1 Introduction 1
1.1 Thesis Outline . . . 2

2 Background and Related Work 4
2.1 Vibratory Pile Driving . . . 4
2.1.1 Important Vibratory Parameters . . . 5
2.2 Vibro-driveability . . . 5
2.2.1 Factors Found to Affect the Vibro-driveability . . . 6
2.2.2 Prediction of Vibro-driveability . . . 8
2.2.3 Measuring . . . 9
2.2.4 Current Practices for Driveability Prediction . . . 10
2.3 Artificial Neural Networks . . . 10
2.3.1 ANN Architectures . . . 11
2.3.2 Training Algorithms . . . 14
2.4 Conclusion . . . 15

3 Data Sets and Data Handling 16
3.1 Data sets . . . 16
3.1.1 Pile Driving Data . . . 17
3.1.2 CPT Data . . . 17
3.2 Data Preprocessing . . . 18
3.3 Feature Selection . . . 19
3.4 Data Scaling . . . 21
3.5 Conclusion . . . 22

4 Model Development 23
4.1 MLPNN-ADAM Development . . . 24
4.1.1 Hyperparameter Optimization . . . 24
4.2 RBFNN-ADAM Development . . . 26
4.2.1 Hyperparameter Optimization . . . 27
4.3 MLPNN-PSO Development . . . 28
4.3.1 Hyperparameter Optimization . . . 29
4.4 RBFNN-PSO Development . . . 30
4.4.1 Hyperparameter Optimization . . . 30
4.5 Vibratory Driveability Model . . . 32
4.5.1 Motivation VD Model . . . 32
4.5.2 Hyperparameter Optimization . . . 36
4.6 Conclusion . . . 37

5 Results 38
5.1 Results Berkel en Rodenrijs . . . 38
5.2 Results Westzaan . . . 40
5.3 Results Moerdijk . . . 41
5.4 Conclusion . . . 43

6 Future Work 44
6.1 Scope of Performance . . . 44
6.1.1 Construction Project Location . . . 44
6.2 Potential for Performance Improvement . . . 45
6.2.1 Mode of Operation . . . 45
6.2.2 Domain Expert Knowledge . . . 46
6.3 Conclusion . . . 46

7 Conclusion **47**

A Feature Selection **48**
A.1 Feature Ranking Calculations 48
A.2 Effect Features on Driveability 49

B Results **50**
B.1 Results Heerlen 50
B.2 External Influence Driveability Moerdijk 52
B.3 Additional Results Berkel en Rodenrijs 53
B.4 Additional Results Heerlen 54

List of Figures

1 Workflow diagram of this research. Each segment is labeled with the corresponding chapter 3
2 Vertical vibrations of the eccentric masses. The horizontal forces cancel each other out. Image taken from Jonker et al. (2017). 4
3 Typical driveability curves of three different piles of length 9 m. The mean is computed for clarity. Image taken from Qin et al. (2017). 5
4 Soil classification chart proposed by Robertson et al. (1986) based on CPT cone tip pressure q_c and friction pressure R_f. Image taken from Robertson (2010). 9
5 Schematic depiction of a typical ANN architecture. Image taken from (de Jong, 2021). . . 11
6 The architecture of MLPNN with one hidden layer. Image taken from Fath et al. (2020). 11
7 Architecture of RBFNN with one hidden layer. Image taken from Fath et al. (2020). . . . 12
8 Schematic of particle swarm optimization algorithm. Image taken from Chen et al. (2020). 15
9 A schematic depiction of the location and size of the pile driving projects used in this research. 16
10 A schematic depiction of the general soil layout of the projects used in this research. . . . 17
11 Example of a pile plan (i.e., 'palenplan' in Dutch). 18
12 Dimensionality reduction performed by PCA. Image taken from Cheng (2022) 19
13 Results of the PCA analysis performed on the independent variables. 20
14 K-fold Cross Validation with k = 5. The blue folds represent the validation subsets and the green folds represent the training subsets. 21
15 An overview of the developed neural networks. The neural networks are composed of the proposed neural network architectures and optimization algorithms. 23
16 A schematic depiction of the SGD method. Image taken from Géron (2019) 24
17 A schematic depiction of the structure of the developed MLPNN optimized with the Adam optimizer. 26
18 RBF Neuron activation for different values of β. Image taken from Mccormick (2013). . . 26
19 A schematic depiction of the structure of the developed RBFNN optimized with the Adam optimizer. 28
20 The Particle Swarm Optimization (PSO) algorithm. Image taken from Le et al. (2019). . 28
21 A schematic depiction of the structure of the developed MLPNN optimized with the PSO optimizer. 30
22 A schematic depiction of the structure of the developed RBFNN optimized with the PSO optimizer. 31
23 A schematic depiction of the development process of the VD model consisting of two parts: the development of the ANN architecture and the development of the optimization algorithm. 32
24 Results of the PCA analysis performed on the independent variables. The blue arrows highlight the input features used in this research. 33
25 Predictions of the MLPNN and RBFNN for two projects and input features of the respective CPT's. 34
26 Schematic depiction of the architecture of the VD mode. 34
27 Predictions of the MLPNN and RBFNN optimized with the Adam and PSO algorithm. . 35
28 A schematic depiction of the structure of the developed VD model. 37
29 Measurements and predictions plotted for the different models for one vibratory driven pile. 39
30 Measurements and predictions plotted for the different models for one vibratory driven pile. 41
31 Measurements and predictions plotted for the different models for one vibratory driven pile. 42
32 Measurements and predictions and corresponding pull-down force plots for the Heerlen project. 45
A.1 Effect of input features on the driveability for the Heerlen test set. The features used for the prediction of the vibro-driveability are shown in bold. 49
B.1 Measurements and predictions plotted for the different models for one vibratory driven pile. 51
B.2 Predictions of the RBFNN-PSO and VD model for the Moerdijk project and input features of the respective CPT's. 52
B.3 Normal distributions of the prediction error for the Berkel en Rodenrijs test set. 53
B.4 Normal distributions of the prediction error for the Heerlen test set. 54

List of Tables

1 Advantages and drawbacks MLPNN and RBFNN . . . 14
2 Pile length ranges used to select and remove piles from the data set. . . . 18
3 Statistical information of input and output parameters . . . 19
4 Ranking of independent variables based on the PCA results. . . . 20
5 Results of the feature selection according to the sequential selection algorithm. The optimal set of features is visualized in bold. . . . 21
6 MLPNN-ADAM hyperparameters. . . . 25
7 Specifications of the MLPNN-ADAM architecture. . . . 25
8 RBFNN-ADAM hyperparameters. . . . 27
9 Specifications of the RBFNN-ADAM architecture. . . . 27
10 MLPNN-PSO hyperparameters. . . . 29
11 Specifications of the MLPNN-PSO architecture. . . . 30
12 RBFNN-PSO hyperparameters. . . . 31
13 Specifications of the RBFNN-PSO architecture. . . . 31
14 VD model hyperparameters. . . . 36
15 Specifications of the VD architecture. . . . 36
16 Division of train and test set. . . . 39
17 Performance metrics values for the different models for the test set. The best-performing model is highlighted. . . . 39
18 Division of train and test set. . . . 40
19 Performance metrics values for the different models for the test set. The best-performing model is highlighted. . . . 40
20 Division of train and test set. . . . 41
21 Performance metrics values for the different models for the test set. The best-performing model is highlighted. . . . 42
22 Performance metrics values for the different models for the projects Berkel en Rodenrijs and Best. . . . 44
A.1 Results of the calculations leading to the ranking of the independent variables based on the PCA results. . . . 48
A.2 Ranking of independent variables based on the PCA results. . . . 49
B.1 Division of train and test set. . . . 50
B.2 Performance metrics values for the different models for the test set. . . . 50

1 Introduction

The first known documents where pile driving has been referred to, date back to the fourth Century BC. Herodotus, a Greek writer and traveler, records how Paeonians lived in dwellings erected on tall piles that were driven into a lake bed (Boyer, 1985). To this day, pile driving is a widely used technique for the construction of buildings and infrastructure. In recent years, the technique has become increasingly important since: (i) more construction sites are being developed in sedimentary basins where the consolidated soil needs support; (ii) the increasing cost of land necessitates the construction of taller structures that must be supported by piles; (iii) More buildings are locating their facilities underground, requiring the use of deeper foundations; (iv) more embankments are having to be protected against erosion (Howe et al., 1995).

One of the most applied techniques is vibratory pile driving. Vibratory pile driving is a method where the pile is vibrated into the sediment by keeping the pile in an oscillating motion throughout the vibratory driving process. Vibratory hammers are popular with pile contractors from an economic perspective as vibratory pile driving is a more efficient installation method in granular soils compared to other pile driving methods. This can be explained by the induced vibrations that serve to reduce the soil resistance, allowing for higher penetration velocities (Holeyman et al., 2002, Rausche, 2002, Rodger and LittleJohn, 1980, Viking, 2002, Warrington, 1992). In addition to being much faster, vibratory pile driving makes much less noise and causes much less acceleration in surrounding soil or conductive structures, resulting in fewer environmental issues (Jonker, 1987).

Even though this method is considered the best option in terms of noise pollution, it has still been one of the most significant sources of vibrations in urban areas. Therefore, pile driving behavior assessment and prediction have been of immediate importance as a result of society's rising awareness about environmental impact and the fact that building projects are increasingly located in urban areas and close to existing structures (Cleary and Steward, 2016). Furthermore, from an economic point of view, the benefit of the vibrating block can only be realized if the contractor correctly predicts which size hammer will bring the pile to the required depth. The use of an unsuitable vibrator will result in extra costs inferred and project delays (Massarsch, 2017).

Several predictions can be performed for vibratory pile driving with contemporary software, such as a vibro-driveability prediction, a maximum penetration depth prediction or a prediction for the maximum pile stresses. These predictions are made based on the Cone Penetration Test (CPT) measurement data and on numerical simulations. According to Middendorp (2021), an expert in pile driving predictions, one of the most difficult but perhaps most important aspects to predict is the driveability. The driveability relates to the penetration speed, or rate of penetration, with which the pile is driven through the soil (Viking, 2002). The term vibro-driveability refers to the rate of penetration for vibratory driven piles specifically. Accurate prediction of the penetration rate of the pile to be driven allows the selection of an appropriate vibratory system and equipment, resulting in increased construction performance. A rapid construction process reduces costs and environmental impact.

Prediction of vibro-driveability is complex as the penetration rate is affected by a large number of variables related to the three main actors involved in the vibratory sinking process (Qin et al., 2017):

- the selected driving system,
- the pile to be driven,
- the imposed soil conditions.

In recent years there have been many researchers attempting to develop vibro-driveability prediction methods. However, Viking (2002) stresses that none of these covers all the fundamental physical phenomena accompanying vibro-driveability prediction.

Recently, the registration of pile driving data has increased significantly. This extended amount of measurement data can potentially be leveraged to improve the predictions of the penetration rate in future projects. According to Shahin (2016), artificial neural networks (ANNs) are by far the most commonly used machine learning (ML) technique in geotechnical engineering. The main objective of ANNs is for an algorithm to learn patterns from a data set to predict the so-called target values from the feature values. This makes the use of ANNs interesting, as assumptions made in current prediction methods can be bypassed. Published work by Ghaboussi (1992) and Ghaboussi et al. (1991) endorse this idea. Research was conducted on the constitutive modeling of material behavior with neural networks and indicated some potential applications of neural networks in geotechnical engineering, such as the modeling of soil behavior, the evaluation of liquefaction potential, seismic ground response and response of geotechnical structures (Chan et al., 1995). Also, according to Momeni et al. (2015), implementing feed-forward backpropagation artificial neural network (ANN) to solve geotechnical problems has recently gained attention mainly due to its ability in finding complex non-linear relationships among different parameters. According to Shahin et al. (2001), the application of ANNs in geotechnical engineering can be found in the prediction of pile capacity, settlement of foundations, soil properties and behavior and liquefaction, among others. However, to the best of the authors' knowledge, no work on the prediction of vibro-driveability using ANNs has been reported.

The objective of this study is to develop a neural network to predict the penetration speed of vibratory driven piles at a certain depth z. The ANN methods include multilayer perceptron neural network (MLPNN) and radial basis function neural network (RBFNN). These neural networks are trained with the particle swarm optimization (PSO) algorithm and the backpropagation (BP) algorithm method. The BP method serves as the conventional method used for reference. Additionally, based on these methods, we introduce a novel ANN, the Vibratory Driveability (VD) model, which combines the fruitful characteristics of the MLPNN and RBFNN. The performance of the developed models is compared with the performance of contemporary vibro-driveability prediction software based on performance metrics and desired prediction behavior.

The workflow of this study, demonstrated in Figure 1, involves 4 main stages including data preprocessing, feature selection, model development and evaluation of the results. The existing models shown in Figure 1 refer to the MLPNN and RBFNN trained with the BP method and the PSO algorithm.

1.1 Thesis Outline

This section provides the structure of this thesis report and intends to give an overview of the topics that will be discussed. Chapter 2 presents the theoretical background for this work by giving an introduction to vibratory pile driving and describing the heritage of vibro-driveability predictions. Then it discusses promising artificial neural network architectures and an optimization algorithm. Chapter 3 gives a detailed explanation of the data sets used for this work, where it elaborates on the preprocessing of the data and the feature selection process. The development process of the models is then discussed in Chapter 4 where the training and hyperparameter tuning procedures are presented. An evaluation of the performance of the trained models is presented in Chapter 5, where prediction results for three different construction projects are shown. Chapter 6 gives an in-depth overview of the scope of the performance and suggests recommendations for future work for this research. Finally, Chapter 7 summarizes the main conclusions of this work.

Appendix A presents additional results of the feature selection process. Additional performance results are contained in Appendix B.

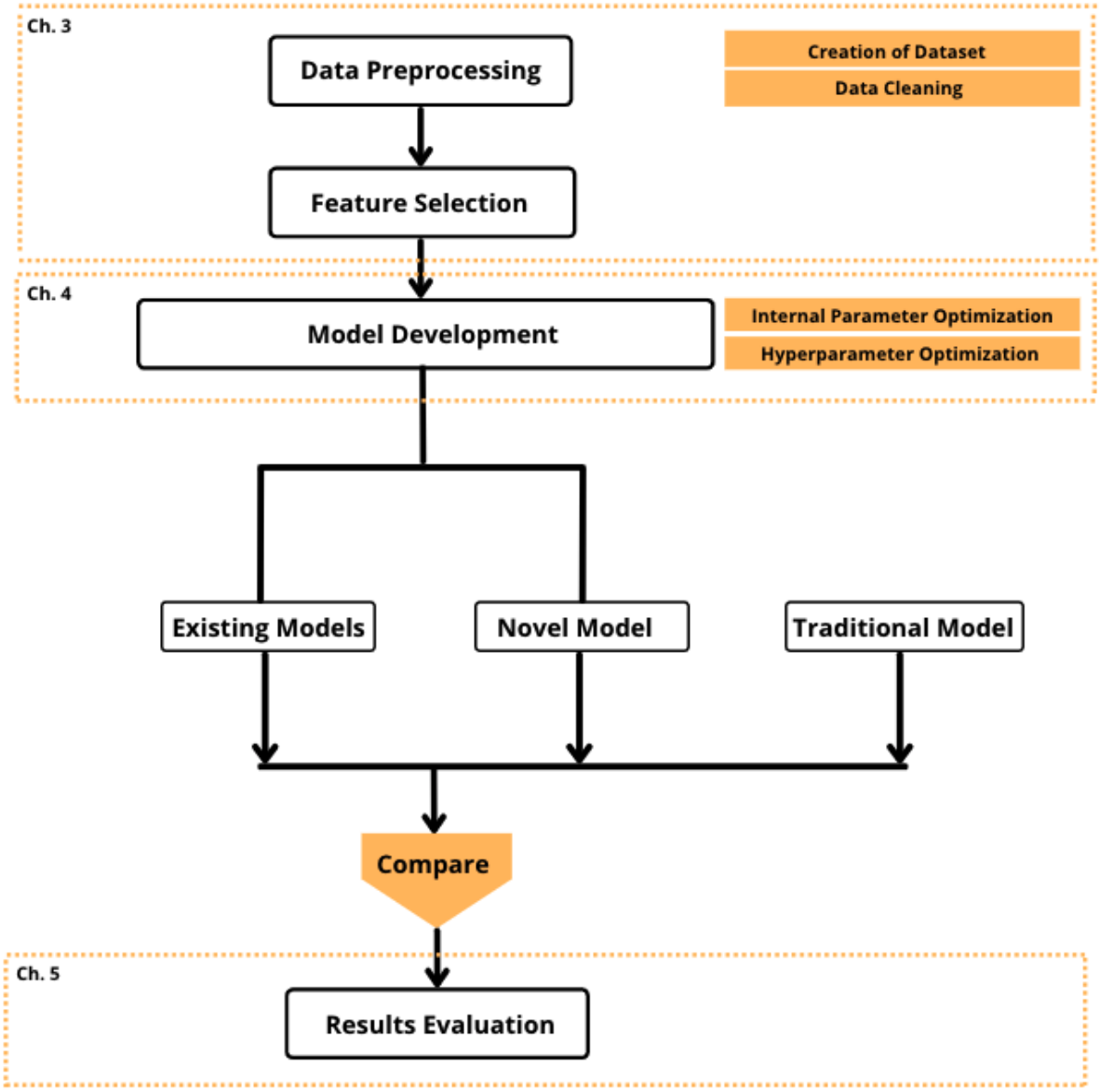


Figure 1: Workflow diagram of this research. Each segment is labeled with the corresponding chapter

2 Background and Related Work

This chapter will provide an overview of the relevant concepts and the associated literature. The items are assumed essential for a good understanding of this study.
In Section 2.1, preliminary information about vibratory pile driving is provided, followed by details about important vibratory parameters. An in-depth overview of vibro-driveability and the prediction of vibro-driveability is provided in Section 2.2. Once this has been clarified, Section 2.3 discusses and compares promising artificial neural network architectures and training algorithms.

2.1 Vibratory Pile Driving

A vibratory hammer is used to drive sheet piles, pipes or other elements into the soil by imparting a small longitudinal vibratory motion of a predetermined frequency and displacement amplitude to the driving unit. At the start of vibratory pile driving, the inside of the pile is empty (it rather looks like an open cylinder). As the pile progresses through the soil, the pile is filled with concrete, resulting in a concrete pile foundation at the desired depth.

The vibratory driver, or the vibro-unit, can be divided into four different parts: the suppressor housing, the elastomer dampers, an excitor block and an hydraulic clamping device, as shown in Figure 2 (Qin et al., 2017). An explanation of the different parts is provided below.

- The suppressor housing, also called the bias mass, is the non-vibrating part of the unit. It is connected to the elastomer dampers that minimize the transmission of the dynamic motion of the excitor block to either the vibrator carrier or the leader mast. The weight of the suppressor housing is called the static surcharge force (F_0) or the bias mass.
- The elastomer dampers connect the excitor block to the suppressor housing and minimize vibration transmission.
- An excitor block contains the rotating eccentric weights, gearbox and hydraulic motor which combined generate the vertical vibrations. The pairs of counter-rotating eccentric masses rotate usually at frequencies below 40 Hz (Jonker and Hartog, 1988). The eccentric masses are phased so that the horizontal components of the centrifugal forces are eliminated and the vertical components are added to each other.
- The hydraulic clamping device transmits the vertical motion of the vibrator to the object being driven. Clamping and releasing the jaw pads as well as holding the object being driven is achieved hydraulically. The coefficient of friction can vary widely, and also energy can be lost in the heating of the pile top due to both hysteresis of the pile material and clamp slippage. To minimize energy loss, it is critical that the jaw pressure in these applications is correct and maintained during the installation phase (Whenham, 2011).

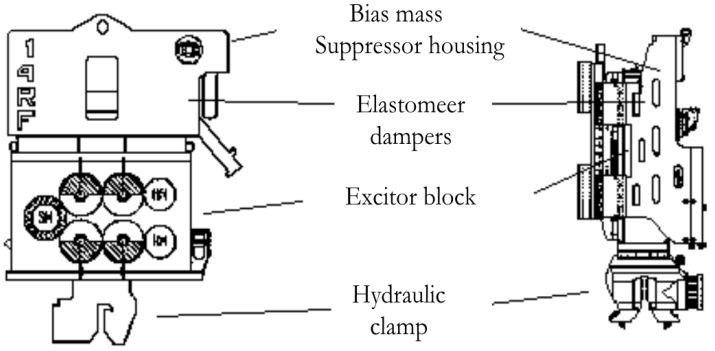


Figure 2: Vertical vibrations of the eccentric masses. The horizontal forces cancel each other out. Image taken from Jonker et al. (2017).

2.1.1 Important Vibratory Parameters

The mechanical action of a vibrator onto a pile consists of two parts: a stationary part called the static surcharge force induced by gravity forces and a vibratory part called the sinusoidal vertical force produced by counter-rotating eccentric masses. The driving force (F_d) is the sum of the surcharge force (F_0) and the vertical force (F_v) and can be seen as the theoretically generated driving capacity of the vibrator.

According to Viking (2002), the main parameters of vibratory systems are the following:

- The static surcharge force (F_0), defined as the stationary action of the vibrator.
- The eccentric moment (M_e), defined as the sum of each single eccentric mass moment.
- The driving frequency (f_d), specified as the number of revolutions per second of the eccentric masses.
- The sinusoidal vertical force (F_v), defined as the vertical component of the centrifugal force F_c generated by the counter-rotating eccentric masses.
- The dynamic mass (m_{dyn}), specified as the mass that represents all total vibrating masses including the mass of the pile, the clamps and the excitor block.
- The free-hanging (double) displacement amplitude (S_0), specified as the total (i.e., double) amplitude of the movement of the vibrator.
- The theoretical power (P_t), defined as the nominal power of the motor driving the eccentric masses.

It should be noted that both the free-hanging (double) displacement amplitude and the theoretical power listed in the specifications are in reality lower. In case of the free-hanging (double) displacement amplitude, this is due to the weight of the pile and the resistance forces that have not been incorporated in the calculation. In case of the theoretical power, this is due to the energy dissipation in the suppressor housing and the energy losses at the pile clamp connection (Holeyman et al., 2002). The authors state that for both parameters, accurate estimation methods are still lacking.

2.2 Vibro-driveability

Generally speaking, the development of the penetration rate with the penetration depth can be divided into three stages: (i) increasing phase, (ii) stable phase and (iii) decreasing phase (see Figure 3). At first, the penetration rate increases due to the low soil resistance and the normalization process of the vibratory system from starting up. Then, the penetration rate remains approximately constant as the soil resistance is remaining stable and the vibrator is operating at a normal state. Finally, as the pile approaches the coarser sand layers, the penetration rate reduces substantially with depth due to the significant increase in the pile shaft and pile toe resistance.

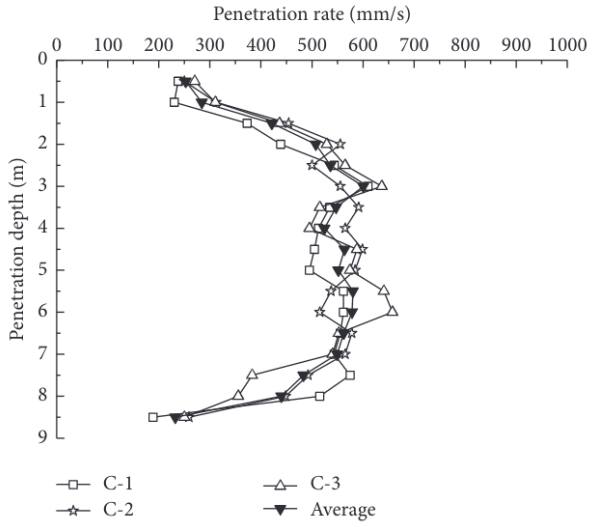


Figure 3: Typical driveability curves of three different piles of length 9 m. The mean is computed for clarity. Image taken from Qin et al. (2017).

2.2.1 Factors Found to Affect the Vibro-driveability

As mentioned before, three actors play a role in the vibro-driveability process; the selected driving system, the pile to be driven and the imposed soil conditions. An overview of the previously documented effects of how various parameters relating to the three actors influence vibro-driveability is presented below.

Vibratory Related Factors
Rodger and LittleJohn (1980) and O'Neill and Vipulanandan (1989) systematically studied how the penetration rate is affected by vibratory related parameters. According to these studies, the following vibratory parameters are strongly related to driveability:

- static surcharge force (F_0),
- eccentric moment (M_e),
- driving frequency (f_d),
- free-hanging double-displacement amplitude (S_0),
- efficiency of the vibrator equipment (ξ),
- mode of operation.

Static surcharge force (F_0)
As explained in Section 2.1.1, the driving force (F_d) is the sum of the surcharge force (F_0) and the vertical force (F_v). An increase in the surcharge force (F_0) should increase the driving force (F_d) and consequently increase the penetration rate (v_p). The static surcharge force can be altered by the so-called pull-down force, or hook load. The hook load represents the amount of static weight of the hammer resting on the pile. A hook load of 100% indicates that the vibro-hammer is fully suspended by the crane, hence only the dynamic weight rests on the pile head. A hook load of 0% indicates no tensile stresses in the hook and therefore the full static and dynamic weights rest on the pile. Decreasing the hook load (i.e., increasing the pull-down force) results in an increase in the static mass on the pile which may cause an increase in the penetration rate. However, according to Viking (2002), how much an increase in (F_0) affects the (v_p), has not yet been established.

Eccentric moment (M_e)
Rodger and LittleJohn (1980) examined the relationship between the eccentric moment and the penetration rate and found this relationship to be linear.

Driving frequency (f_d)
It is widely recognized that the driving frequency has a significant effect on the penetration rate. However, the same applies as for the static surcharge force, it has not yet been established how the optimum driving frequency (f_d) should be determined for an optimal driveability. Contradicting findings can be found in the literature on the effect of the driving frequency on the toe and shaft resistance of the pile and, consequently, on the penetration rate. Viking (2002) concludes that the shaft friction (R_s) is virtually independent of the driving frequency while the toe resistance (R_t) tends to be strongly related to the driving frequency. Further explanation of the shaft friction (R_s) and the toe resistance (R_t) can be found later in this section. In contrast to Viking, Preobrajhenskaja (1956) states that an exponential relationship between the shaft resistance and the driving frequency exists. The author states that drawing conclusions from these tests is difficult, as both tests did not duplicate the actual field conditions.

Free-hanging double-displacement amplitude (S_0)
Not many scientific publications can be found showing how the double-displacement amplitude (S_0) affects the driveability. The few statements found in the literature are from Rodger and LittleJohn (1980), Westerberg (1995) and Denies (2010) who stated that during vibratory driving with low toe resistance, a displacement amplitude of 5-20 mm should be used, while for conditions with high toe resistance this value should be between 20-30 mm. No scientific literature can be found that provides any information on how the displacement amplitude might be appropriately determined under different soil conditions.

Efficiency of the vibrator equipment
As mentioned in Section 2.1.1, the theoretical power of the vibratory driver specified by the manufacturers is always higher than the actual power transmitted. However, it has not been established how much of this power is precisely effective compared to the theoretical power. Further research should be conducted on the efficiency factor to be able to incorporate the power loss into the computation of the actual driving function (Viking, 2002).

Mode of operation
According to Viking (2006), the mode of operation is as important as picking the right equipment. The operator has three different tasks during driving: i) the operator should ensure that the pile reaches the desired depth. However, due to difficulties in penetrating the different soil layers, the pile length can vary significantly. When the pile cannot be driven to the designed depth, the operator can decide to stop the pile driving process. In this case, the operator will insert a second pile next to the first pile's location; ii) the operator should ensure for a correct vertical position of the pile during driving to avoid bending of the pile; iii) the operator can tune different parameters when difficulties arise during vibratory driving to improve the driveability, for example by adjusting the hook load.

Pile Related Factors
Only few literature studies can be found on the pile related parameters influencing the driveability. A distinction between two different factors affecting the vibro-driveability can be made:

- geometric effects,
- longitudinal, transversal and flexural properties.

Geometric effects
In general, one can conclude that the size and shape of the pile have a significant effect on the driveability. Larger-sized piles are heavier and consume more energy (in the case of long piles) and thus a decreased penetration rate can be observed (Qin et al., 2017). Additionally, the shape of the pile affects the penetration rate of piles. For different sizes of open and closed-ended piles (smaller versus larger) an expected and identical trend in the driveability curves ($v_p - z$) can be observed: a decreasing v_p with increasing depth. This can be explained by the increasing resistance due to an increased contact area between pile and soil. However, in case of a smaller H pile, different behavior is observed; the penetration rate increases after which it decreases at the end of the installation phase (Rao, 1993, Wang, 1994).

Longitudinal, transversal and flexural effects
Studies of the longitudinal effects of vibratory driven piles usually relate to the question of whether the pile vibrates as a rigid body or not rather than on the effect on driveability. Prediction of the vibratory behavior is considerably simplified when the pile can be assumed to behave as a rigid body as it allows for the assumption of equal displacement along the length of the pile.

The undesired lateral vibrations during driving normally relate to improper alignment of the driving force and axial symmetry line of the pile. Simply, a bending moment M_{ecc} is introduced at the pile head if the clamp does not hold the pile correctly. This might cause a situation where the pile vibrates as much laterally as axially. Viking (2002) states that these effects have been observed to reduce or increase the driveability depending on different soil types and can result in an increased amount of ground vibrations. The authors state that transversal and flexural effects due to lateral vibrations are often not included in driveability analysis while these effects have been observed to affect the driveability significantly in specific conditions.

Soil Related Factors
A simplified description of the effect of vibratory driving on the soil is that the soil is disturbed by the vibrations which causes the soil resistance to decrease temporarily and locally (Dorp et al., 2019). This effect can be attributed to the cyclic vibrations exerted by the vibrator on the soil. These vibrations serve to reduce the soil resistance allowing penetration under a low vertical thrust. The rate of reduction of the soil resistance is strongly influenced by the soil type; the reduction is the greatest when driving in cohesionless soils and least when driving in cohesive soils. A cohesive soil has an attraction between particles of the same type, origin and nature. As a result, cohesive soils are a type of soil that adhere to each other. Cohesive soils are silts and clays, or fine-grained soils. Cohesionless soils (non-cohesive soil) are soils that do not stick to each other and rely on friction. Sands and gravels, often known as coarse-grained soils, are examples of these soils (Whitlow, 1990).

The soil resistance during vibratory driving is generated simultaneously at the pile toe and along the pile shaft and can be described by the sum of the toe resistance (R_t) and the shaft resistance (R_s).

Toe resistance (R_t)
The toe resistance is found to have a significant effect on the penetration rate (Cudmani, 2001, Deckner, 2017, Holeyman et al., 2002, Viking, 2002). However, up to this day, estimation of the toe resistance of vibratory driven piles remains challenging. For example, the exact influence of the toe shape, which

differs between open-ended pipe piles and closed-ended pipe piles, is still an unresolved issue which makes accurate estimation of the toe resistance impossible.

Shaft resistance (R_s)
The shaft friction is found to reduce significantly in coarse-grained soil (cohesionless soil). Different methods have been developed for the analysis of shaft resistance by Cudmani (2001), Dierssen (1994), Holeyman (2000), Massarsch (2002), Rausche (2002) and Moriyasu et al. (2018). Reasons for the favorable reduction have been proposed by the authors such as liquefaction, material degradation and fluidization. However, exact estimation of the shaft resistance remains a challenge due to the many different phenomena at play in the soil during vibratory loading.

Several key mechanisms influence the favorable reduction of the soil resistance at the toe and along the shaft of the pile during vibratory driving. These mechanisms have been extensively studied within the field of earthquake engineering. Literature review within this field, among others, revealed the following soil related factors that have a major influence on the vibro-driveability:

- Induced cyclic motion: The soil structure will continuously weaken when subjected to cyclic loading. This process is known as cyclic degradation and causes the soil resistance to decrease significantly during vibratory driving.
- Initial relative density: The pressure between the grains (i.e., the initial relative density) reduces when subjected to cyclic loading, however, the amount of reduction depends on the soil type; the effects of vibrations decrease with an increase in normal pressure between the grains (Youd, 1967).
- Saturation level: The presence of water has been found to have a positive influence on driveability, since the oscillating motion of the pile momentarily increases the excess pore pressure (Viking, 2002). This process may eventually lead to liquefaction, a condition where the soil resistance is almost zero. Under drained conditions (i.e., a fully saturated soil) the condition of liquefaction is immediate when subjected to cyclic loading. Under undrained conditions, the build-up of excess pore pressure can potentially lead to the liquefaction condition.

It should be stated that the geotechnical properties of soils are controlled by many factors such as grain size, saturation level and mineralogy and it is difficult to establish the interactions between all these factors solely by theoretical methods due to their interdependence and non-linear behavior (Yang and Rosenbaum, 2002).

2.2.2 Prediction of Vibro-driveability

Several important attempts to develop formulae to predict the vibro-driveability of piles have been made in the past. Not only do the predictive models differ in how they account for mechanical engineering principles, but also in how they model dynamic soil resistance. Modern pile behavior programs used in the industry, such as AllWave-VDP (previously known as TNOWAVE) (Middendorp and Verbeek, 2006) and PVDA (Moulai-Khatir et al., 1994), are based on the wave equation (Smith, 1960). The wave equation models stress wave propagation in a pile which allows for the calculation of accompanying displacements, velocities and acceleration along the entire length of the pile (Josselin De Jong, 1956, Voitus van Hamme et al., 1974). The wave equation found its breakthrough when Smith (1960) developed a numerical solution of the wave equation accompanied by a pile-soil model to calculate the soil resistance. In this method, the soil is assumed to deform in a linearly elastic way up to a limiting deformation, the quake Q [mm]. After the quake threshold is reached, the soil starts deforming plastically. The dampers represent the viscous component of soil resistance. This value is characterized by the coefficients of viscous damping, J [s/m]. Typical values for quake and damping constant, recommended by Smith (1960) based on experience, are $2-3$ mm, and $0.1-0.2$ s/m, respectively (Ismail, 2019). It should be noted that the mass of the soil is usually omitted.

According to Ismail (2019), one of the main disadvantages of Smith's soil model is the simplistic assumptions made about the soil properties. The quake and the damping are unconventional parameters to represent soil fatigue and the model is oversimplified by using them. Also, the recommended values of these parameters were proposed based on Smith's experience, but without any direct experimental field measurements. The accuracy of these values was recognized as relatively crude from a soil mechanics point of view (Wu et al., 1989). Smith concluded: "The knowledge of soil mechanics is incomplete, especially the knowledge of soil mechanics under pile-driving action. This offers a fertile field for future investigations".

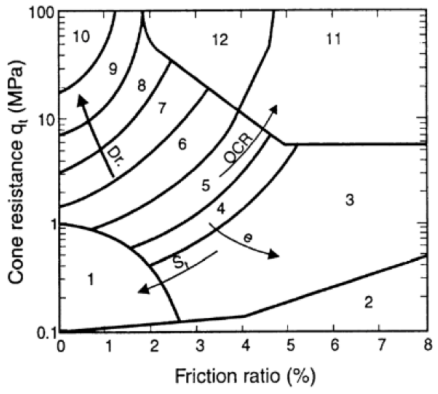
In general, according to Viking (2002), none of the models developed to this date cover all the fundamental phenomena related to vibro-driveability prediction. Assumptions vary significantly from one method to the other, making it difficult to choose an optimal one for a specific project. The assumptions with the largest effect on the driveability are being listed below (Holeyman and Whenham, 2010, Whenham, 2011):

- Vibro-driveability models assume a complete transfer of the driving force generated by the vibrator to the force exerted on the pile.
- The decrease of the soil resistance is influenced by many different factors, such as the initial relative density and the saturation level. One soil model that combines all these factors is still lacking. For example, the saturation level of the soils is often unaccounted for and therefore a standard value is assumed for the build-up of pore pressure.
- Assumptions are made on the vibrator-pile connection as the vibrator is generally assumed to be rigidly clamped to the pile driven and that no flexural effects (i.e., a bending movement) are present.

2.2.3 Measuring

The few measurements performed within the vibratory driving industry are mostly to gain a better understanding of the vibro-related parameters, the soil resistances and the vibrations present. This is accomplished by performing Cone Pressures Tests (CPT).

For more than 40 years, the electric Cone Penetration Test (CPT) has been used. It is favored over traditional field site investigation methods such as drilling and sampling because it is fast and repeatable and thus more economically attractable. These benefits have resulted in a steady increase in the use of CPTs in numerous locations throughout the world. Prior to construction, CPTs are taken at sample locations on the construction site. During a CPT, a cone is hydraulically pushed into the ground at a constant rate, while the pressure at the tip of the cone and on the shaft of the rod is measured. This is done using load cells that measure the pressure force. From these measurements, it is possible to classify the soil composition (e.g., sand or clay). This has typically been accomplished using charts that link cone parameters to soil type. The cone parameters used are the cone tip pressure q_c and the friction ratio R_f where $R_f = \frac{f_s}{q_c} * 100\%$. f_s is the shaft pressure of the cone. The first charts used were proposed by Douglas (1981), however, nowadays the most popular one is the chart proposed by Robertson et al. (1986), see Figure 4. Despite the fact that newer charts based on normalized parameters have been developed, the simple chart based by Robertson remains popular because it is simple to use and can be utilized in real-time during the CPT as it only requires the basic CPT measurements. The reason for this is that the charts that use normalized parameters require other information on soil unit weight and groundwater conditions that is not available during CPT.



Zone	Soil Behavior Type
1	Sensitive fine grained
2	Organic material
3	Clay
4	Silty Clay to clay
5	Clayey silt to silty clay
6	Sandy silt to clayey silt
7	Silty sand to sandy silt
8	Sand to silty sand
9	Sand
10	Gravelly sand to sand
11	Very stiff fine grained*
12	Sand to clayey sand*

* Overconsolidated or cemented

Figure 4: Soil classification chart proposed by Robertson et al. (1986) based on CPT cone tip pressure q_c and friction pressure R_f. Image taken from Robertson (2010).

Modern pile behavior programs, such as AllWave, use Robertson's soil classification chart in combination with the numerical simulation based on the wave equation and Smith's pile-soil model (Smith, 1960). A disadvantage of the CPT is that it cannot be used at sites with high soil density (e.g., rock or cement) as these layers are difficult to penetrate and increase the risk of damage to the cone. As a consequence, the CPT tends to not be the tool of choice when driving in coarse-grained soils or heavily overconsolidated fine-grained soils (Ramsey, 2010).

2.2.4 Current Practices for Driveability Prediction

As explained, none of the conventional pile driving software covers all fundamental phenomena related to vibro-driveability prediction and therefore lack the ability to accurately predict the penetration rate. Subsequently, results obtained with this software are not frequently used, according to geo-engineer W. Koers (Koers, 2022). The standard procedure that Koers uses when making predictions is the following: the measurements of one CPT are used to generate a driveability prediction to get a crude idea of the soil layout of the construction site. Based on this information and his own experience, a more appropriate prediction is made. This experience includes knowledge about soil layout and the effect of external factors such as weather conditions, among others. For example, weather reports are taken into account; weather circumstances such as heavy rain or wind often result in mechanical issues, hence more frequent machine downtime.

According to Koers (2022), the most useful predictions contain the general behavior of the penetration speed rather than specific details such as small fluctuations. As penetration rates get influenced by other aspects than soil layout alone (such as weather circumstances and the operator's mode of operation) very detailed driveability curves generally do not accurately represent penetration speeds. A general representation of the driveability gives a more substantiated overview of the soil layout and can therefore be used by the geo-engineer for fine-tuning. Current predictions lack accurate representation of this general behavior and are therefore little used.

2.3 Artificial Neural Networks

Machine learning (ML) has been effectively used for practically every problem in geotechnical engineering over the last decade. Applications of ANNs in geotechnical engineering include the prediction of liquefaction (Agrawal et al., 1997, Baziar and Ghorbani, 2005, Eldin Ali and Najjar, 1998, Hanna et al., 2007, Javadi et al., 2006, Kim and Kim, 2006), the estimation of several soil properties such as shear strength and stress history (Kurup and Dudani, 2002, Lee et al., 2003, Penumadu and Chameau, 1997) and soil settlement predictions (Chen et al., 2006, Shahin and Jaksa, 2005, Shahin et al., 2003, 2004, 2002).

Examples of the available ML techniques are artificial neural networks (ANNs), genetic programming (GP), evolutionary polynomial regression (EPR), support vector machines (SVM), M5 model trees and k-nearest neighbors (Elshorbagy et al., 2010). According to Shahin (2016), ANNs are by far the most commonly used ML technique in geotechnical engineering. Artificial neural networks' potential is being exploited more and more as their popularity grows. From text translation to image recognition or even generating completely new images from scratch, applications span nearly every field of industry and academia. As mentioned in previous chapters, the lack of physical understanding in traditional mathematical models is usually supplemented by either simplifying the problem or incorporating several assumptions into the models. Consequently, current models fail to cover all the fundamental phenomena related to the prediction of vibro-driveability. In contrast, ANN techniques are data-driven methods in which the structure and parameters of a model are determined by the training of input-output data pairings. In this case, there is less of a need to either simplify the problem or integrate assumptions (Goh, 1996, Momeni et al., 2015). Therefore, it is worthwhile to study the viability of ANNs as an alternative method of analysis.

Artificial Neural Networks (ANNs) are a form of machine learning that can be used to model non-linear and complex input-output relationships. A neural network can be described as a "computational mechanism able to acquire, represent and compute a mapping from one multivariate space of information to another, given a set of data representing that mapping" (Garrett, 1994). An ANN is made up of a series of interconnected processing components known as neurons. As can be seen in Figure 5, the neurons are logically organized into two or more layers and interact with one another through weighted connections. The nature and strength of the influence between the linked neurons are determined by the scalar weights. Each neuron is connected to all the neurons in the next layer. There is an input layer where data are presented to the neural network and an output layer that holds the response of the network to the input. A thorough explanation of the neural network methodology, training and optimization methods is beyond the scope of this research but can be found in Nielsen (2015), Patterson and Gibson (2017) and Géron (2019).

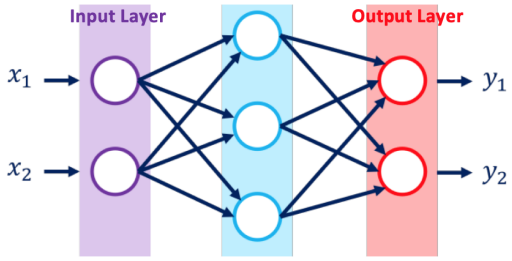


Figure 5: Schematic depiction of a typical ANN architecture. Image taken from (de Jong, 2021).

2.3.1 ANN Architectures

Artificial neural networks (ANN) come in a variety of forms, each of which excels at a different type of task. These different 'forms' of ANNs are distinguished by their 'architecture'. A neural network's architecture outlines how the neurons operate and how they are connected.

As no research is found on useful architectures for the driveability prediction of driven piles, a literature review within the field of drilling was conducted. Some similar phenomena occur during the drilling process such as the build up of pore pressure and wave propagation. Studies conducted by Ashrafi et al. (2019) and Brenjkar and Delijani (2022) pointed out that the multilayer perceptron neural network (MLPNN) and radial basis function neural network (RBFNN) are two promising architectures for the prediction of the penetration rate for drilling operations. A brief description of the fundamentals of MLPNN and RBFNN is given below as the authors conclude that these architectures are the most promising for vibro-driveability predictions.

Multilayer Perceptron Neural Network (MLPNN)
Multilayer perceptron neural network is one of the most common and practical type of ANN and can satisfactorily estimate values for non-linear correlations (Esfe et al., 2015, Moghadassi et al., 2009). MLPNNs are one of the types of feed-forward neural networks and are comprised of an input layer, one or more hidden layers and an output layer. Each layer contains several processing neurons and every neuron is connected by weighted interconnections to subsequent layers. Figure 6 shows the MLPNN architecture.

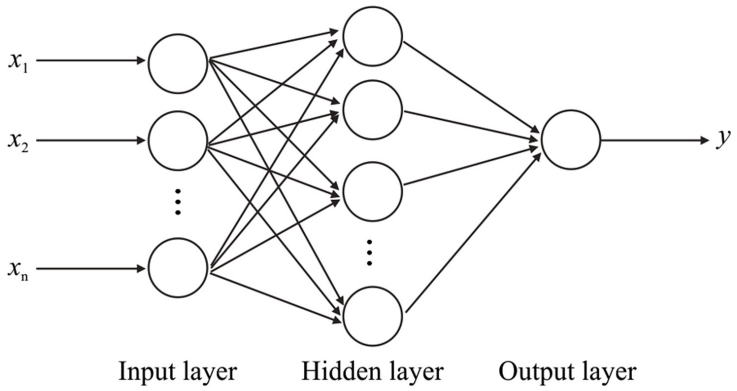


Figure 6: The architecture of MLPNN with one hidden layer. Image taken from Fath et al. (2020).

The number of neurons in the input and output layers is the same as the number of input and output parameters. The number of hidden layers and the number of neurons in the hidden layer(s) define the architecture of the MLPNN and influence the efficiency and accuracy of the model (Hemmati-Sarapardeh et al., 2016). Although an MLPNN with just one hidden layer can usually solve most prediction tasks, more than one hidden layer may be necessary for more complicated systems to produce meaningful results (Sabah et al., 2019). Multiplying the weight w of each node in the previous layer by its calculated value forms one of the inputs to the nodes to which it is connected in the following layer, either another hidden layer or the output layer. A bias value b is also added to the weighted value and that calculated value is fed to the activation level in which a pre-defined activation (transfer) function is used to produce the value of the node in the subsequent layer. This relationship is defined by the following equation:

$$y_{jk} = f_k\left(\sum_{i=1}^{n_{k-1}} w_{ijk} y_{i(k-1)} + b_{ik}\right) \tag{1}$$

Where y_{jk} is the neuron $j's$ output of the MLP $k's$ layer. b_{ik} is the bias term for neuron j in layer k. w_{ijk} is the randomly selected weight applied to the initial training procedure and f_k is the activation function. There are several activation functions such as a binary-step function, linear functions and Gaussian functions (Fausett, 2006). The most frequently used function in the MLPNN is the sigmoid activation function (Ashrafi et al., 2019, Gouda et al., 2022).

Training of MLPNNs is conventionally done with the use of backpropagation (BP) methods such as Levenberg–Marquardt (LM), Gradient Descent (GD) and Resilient backpropagation (RP) (Ghoreishi and Heidari, 2013, Moghadassi et al., 2011). BP methods follow a learning procedure based on the error-correction rule. Network outputs are generated by processing the received input data. Then, the error value is computed by comparing the target values to the network output. The weights and biases are adjusted to reduce the error and the training procedure is repeated until the network reaches a predefined minimum allowable error. Typically, the error function used for this purpose is mean square error (MSE).

Radial Basis Function Neural Network (RBFNN)
RBFNNs are widely used for various applications such as function approximation and pattern classification (Fu and Wang, 2003, Nabney, 2004). RBFNN also belongs to the feed-forward neural networks and its architecture is comparable to the MLPNN's architecture as it is composed of three layers, the input layer, a hidden layer and an output layer. However, the number of hidden layers is not adjustable. This makes the RBFNN much more efficient with fewer computations compared to other types of neural networks with multiple hidden layers (Brenjkar et al., 2021). The number of neurons, also referred to as RBF units, is variable. RBFNN architecture is visualized in Figure 7.

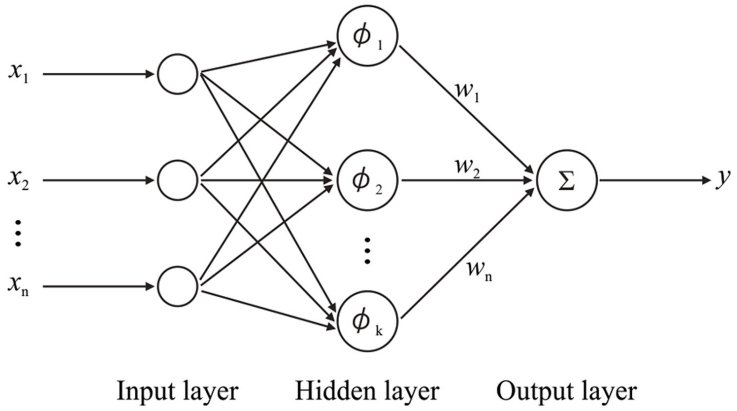


Figure 7: Architecture of RBFNN with one hidden layer. Image taken from Fath et al. (2020).

Each neuron in the hidden layer consists of a radial basis function ϕ. This parameter is a multidimensional radial basis function that describes the difference between an input vector and a predefined center vector. The two tuning parameters that define the efficiency and accuracy of the RBFNN, are the location of the radial basis function's center and its width (spread). The network reaches its highest performance when the measured distance between the input vector and the function's center becomes zero and degrades by increasing the distance (Ashrafi et al., 2019). The output is a weighted sum of outputs from the hidden layer and is formulated as follows:

$$y_i(x) = \sum_{j=1}^{k} w_{ij}\phi\left(\|x - c_j\|\right) \tag{2}$$

Where x is the input vector, y_i is the network's ith output, K is the number of neurons in the hidden layer, c_j denotes the center of the jth hidden neuron, w_{ij} represents the weight of the link from the jth neuron in the hidden layer to the ith neuron in the output layer, $\|\|$ is the Euclidean norm and ϕ is the radial basis function.

In the literature, various forms of radial basis functions are used, with the most common applications referring to the Gaussian Function.

The training of RBF neural networks is composed of two stages:

- (i) find the radial basis function's center and width for each neuron of the hidden layer,
- (ii) find the weights.

There are three different options for finding the centers; using the backpropagation algorithm, selecting random centers and using clustering algorithms. Any clustering algorithm can be used to determine the RBF unit centers, the most common one being the K-means algorithm (Hartigan and Wong, 1979). For training the weight, two options are available; using the backpropagation algorithm or least-squares regression where the error function, the sum of squared errors (SSE), is minimized.

Multilayer Perceptron Neural Network versus Radial Basis Function Neural Network
As mentioned, the MLPNN and RBFNN belong to the same general class of neural networks called the feed-forward neural networks as for both architectures the processing of information follows one direction from the input neurons to output neurons. However, there are a number of fundamental differences between MLPNN and RBFNN (Yu et al., 2011):

- RBFNNs can only have one hidden layer in contrast to MLPNN, which can have more than one hidden layer.
- RBFNNs act as local approximation networks and their outputs are obtained by specified hidden units in certain local receptive fields. On the other hand, MLPNNs work globally and the network outputs are decided by all the neurons.
- The output layer of an RBF network is always linear, whereas in an MLP network it can be linear or non-linear.
- The classification mechanisms for RBF networks and MLP networks are different. RBF clusters are separated by hyper spheres while in MLP networks, hyper surfaces are used for separation.

It is difficult to predict which network architecture, MLPNN or RBFNN, will outperform the other. Each network's performance is largely dependent on the task, thus either architecture can excel in performance depending on the exercise. Little literature is available that makes a fair comparison between multilayer perceptron networks and radial basis function networks. Since no quantitative comparison can be made between the two networks, a qualitative trade-off was made by listing the advantages and disadvantages of each architecture.

It is generally known that MLP networks work well with large input data. Also, the use of multiple layers allows for the utilization of MLPNNs for complex non-linear problems (Akkaya and Çolakoğlu, 2019). Multilayer perceptron networks are more vulnerable to adversarial noise and can make very wrong predictions when fed with such examples as their inputs. This is not the case in RBF networks, which perform more robustly and tolerantly when dealing with noisy input data sets (Montazer et al., 2018). Another advantage of the RBF network is its easy design as the number of hidden layers is decided in case of the RBF network in contrast to MLPNN where there is no analytical approach to determine the number of hidden layers (Nisbet et al., 2009). Also, the training of RBFNN is faster than training MLPNN as RBF networks can be trained in a single direct procedure rather than the iterative process involved in training MLP networks (Ashrafi et al., 2019).

Other advantages of this neural network architecture include its ability to accurately distinguish patterns for non-linear models and its ability to easily generalize to several dimensional spaces (Ashrafi et al., 2019). A limitation of the RBF network is that its centers are determined with reference to the distribution of the input data, but without reference to the prediction task. As a result, representational resources can be wasted in areas of the input space that are irrelevant to the learning task (Nisbet et al., 2009). A summary of the advantages and disadvantages per architecture is captured in Table 1.

Architecture	Advantages	Drawbacks
MLPNN	- Can handle large input data (Akkaya and Çolakoğlu, 2019) - Solves complex non-linear problems (Akkaya and Çolakoğlu, 2019)	- More vulnerable to noise compared to RBFNN (Montazer et al., 2018) - Longer training compared to RBFNN (Ashrafi et al., 2019)
RBFNN	- Solves complex non-linear problems (Ashrafi et al., 2019) - faster training times - Easy design (Nisbet et al., 2009) - Easy generalization to several dimensional space (Ashrafi et al., 2019)	- Centers are determined with reference to the input data (Nisbet et al., 2009)

Table 1: Advantages and drawbacks MLPNN and RBFNN

One of the main advantages of the RBF network over the MLP network is the faster training times. This is beneficial for the development of the model since it will facilitate the large number of training runs required for extensive testing and optimization of the network. Also, the robust character of RBFNN makes the model less susceptible to noise and therefore possibly achieves higher performances compared to MLPNN. However, as stated before, it is very difficult to predict which network architecture will outperform the other. The performance of both architectures is highly dependent on the chosen parameters (number of neurons and number of hidden layers) and the training algorithms used. Therefore, promising training algorithms are also being investigated.

2.3.2 Training Algorithms

Conventionally, the backpropagation (BP) algorithm, a supervised learning algorithm, is used to train neural networks. Backpropagation is a way of propagating the total loss back into the neural network to know how much of the loss every neuron is responsible for. Subsequently, the weights are updated in such a way that the loss is minimized by giving the neurons with higher error rates lower weights and vice versa. However, according to Lee et al. (1991), Wang et al. (2004) and Armaghani et al. (2017), the use of BP algorithms to tune parameters of neural networks has some disadvantages such as slow learning rate and trapping in local minima. These researchers suggest the use of hybrid neural networks where evolutionary algorithms are used for training as they can have much effect in improving prediction performance to its advantage. Evolutionary algorithms are potentially capable of finding optimums that the conventional BP algorithm cannot and thus yield higher performance (Ashrafi et al., 2019, Brenjkar and Delijani, 2022).

Particle Swarm Optimization (PSO)
Studies within the drilling field showed that particle swarm optimization is a very promising evolutionary algorithm (Ashrafi et al., 2019, Brenjkar et al., 2021). PSO is inspired by the flocking behavior of birds and fishes in search of food and escape from danger. It, therefore, exploits simple analogues of social interaction rather than purely individual cognitive abilities (Kennedy and Eberhart, 1995).

In PSO a number of elements, the particles, are placed in the search space of some problem or function and each evaluates the objective function at its current location. Each particle then determines its movement through the search space based on the best position of the particle, the best position already reached by the other particles and the particle's velocity. The next iteration takes place after all particles have been moved. Eventually the swarm as a whole, like a flock of birds collectively foraging for food, is likely to move close to an optimum of the cost function, see Figure 8. In each iteration, the position and velocity of the particles are determined by the following equations:

$$\begin{aligned} v_{ij} &= (t+1) = \omega v_{ij}(t) + c_1 r_{1j}(t)\left[p_{ij} - x_{ij}(t)\right] + c_2 r_{2j}(t)\left[p_{gi} - x_{ij}(t)\right] \\ x_i(t+1) &= x_{ij}(t) + v_{ij}(\mathrm{t}+1) \end{aligned} \tag{3}$$

Where $x_{ij}(t)$ is the current position of the ith particle in the tth iteration, p_{ij} is referred to as the best position discovered by the ith particle and p_{gi} is the best position found by the other particles shared with each particle. The parameters c_1 and c_2 are the cognitive term and the social component, respectively. The constants c1 and c2 are also referred to as trust parameters, where c1 expresses how much confidence a particle has in itself, while c2 expresses how much confidence a particle has in its neighbors. The values r_1 and r_2 are random numbers in the range of [0 1] and ω is the inertia weight of the particle (Karkevandi-Talkhooncheh et al., 2017, Poli et al., 2007).

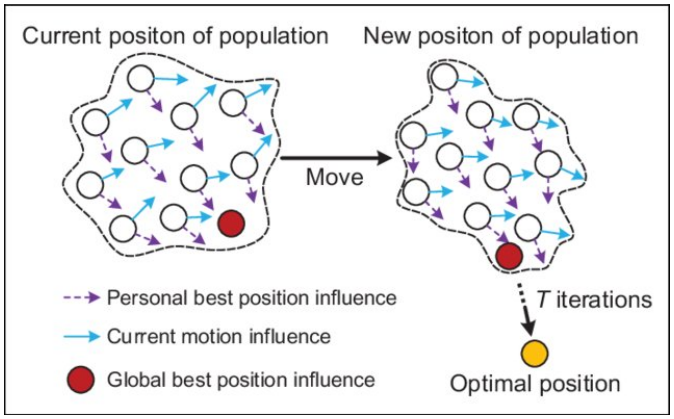


Figure 8: Schematic of particle swarm optimization algorithm. Image taken from Chen et al. (2020).

The PSO algorithm is commonly used as an optimizer as it provides outstanding search capabilities of a multidimensional feasible search space with excellent convergence performance (Kennedy and Eberhart, 1995). Furthermore, PSO is known for its simple implementation and fast convergence (Hassan et al., 2005).

2.4 Conclusion

This chapter introduced the main concepts relevant to this study. Section 2.1 provided an introduction to vibratory pile driving and its most important parameters. Then, Section 2.2 reviewed what factors affect vibro-driveability and how. Also, the performance of modern prediction programs and current practices for using this software were discussed. Additionally, a measuring technique applied within vibratory pile driving, the cone pressure test (CPT), was presented. Moreover, Section 2.3 elaborated on why artificial neural networks can potentially be leveraged for an improvement of the predictions of the penetration rate in future projects. Additionally, two promising ANN architectures and one training algorithm were discussed.

The main conclusions to note from this chapter are the following:

- The mechanisms at play during vibratory driving are controlled by many factors and are related to the driving system, the pile to be driven and the imposed soil conditions. For many of these factors, it is unclear how exactly they affect driveability.
- Methods that accurately represent these mechanisms are lacking due to their interdependence and non-linear behavior. The lack of physical understanding in traditional mathematical models is usually supplemented by either simplifying the problem or incorporating several assumptions into the models.
- AI techniques can potentially be used to improve vibro-driveability predictions as they are data-driven methods in which the structure and parameters of a model are determined by the training of input-output data pairings. In this case, there is less of a need to either simplify the problem or integrate assumptions. MLPNN and RBF are considered promising ANN architectures due to their ability to solve complex non-linear problems, among other things. PSO is considered to be a promising training algorithm as it is known for its simple implementation and provides outstanding search capabilities of a multi-dimensional feasible search space with excellent convergence performance. The BP algorithm can be implemented as a reference as this method is conventionally used to train neural networks.

The lack of accurate vibro-driveability prediction software and the potential of using ANNs to improve these predictions brings us to the coming chapters.

3 Data Sets and Data Handling

This research studies the applicability of machine learning to predict penetration rates of vibratory driven piles. As such, a data set is required that contains vibratory pile driving data. As mentioned earlier, the penetration rate of vibro-driven piles is affected by parameters relating to three different factors: vibratory related factors, pile related factors and soil related factors. For this research, the same pile type, the HSP220 pile, was used with varying diameters. Additionally, the same machine type was used for every project. Therefore, it is assumed that for this research the prediction of vibro-driveability mainly depends on soil properties as the vibrator and pile properties are constant (apart from the pile's diameter and adjustments made by the operator during driving). The approach to predict the penetration speed is to evaluate CPT measurements at a certain penetration depth z, and use these evaluated measurements as input features for the machine learning model to predict the penetration speed at this penetration depth z.

This chapter presents the data sets used for this research and gives a detailed explanation of the data handling process. Section 3.1 gives an overview of the different data sources used to create the data set. Data preprocessing steps are covered in Section 3.2. The feature selection process and the data scaling method are presented in Sections 3.3 and 3.4, respectively.

3.1 Data sets

In the present research, collected data from five vibratory pile construction projects in different locations in the Netherlands performed by the construction company Voorbij were used. Figure 9 shows information about the location and size of the projects.

Figure 9: A schematic depiction of the location and size of the pile driving projects used in this research.

Figure 10 shows the lithology (i.e., soil layout) of the areas. It should be noted that this is a general overview of the soil layout as differences within a soil layer are not shown (e.g., coarse or fine-grained sand). The lithology was obtained from the Dinoloket (Din). The Dinoloket, developed by the Dutch company TNO, provides data on the subsurface in the Netherlands that originate from TNO's DINO repository and BRO (Subsurface Key Register). This includes borehole logs that provide a written record of samples from the drilled part of the subsurface in terms of successive layers of sediment and rock (i.e., the lithology layers). This enables a detailed characterization of soil layers. An important detail to note here is that the borehole logs are an estimate of geological units and properties for an area of 100 x 100 m. Therefore, a synthetic borehole may deviate from the actual borehole log at the same location.

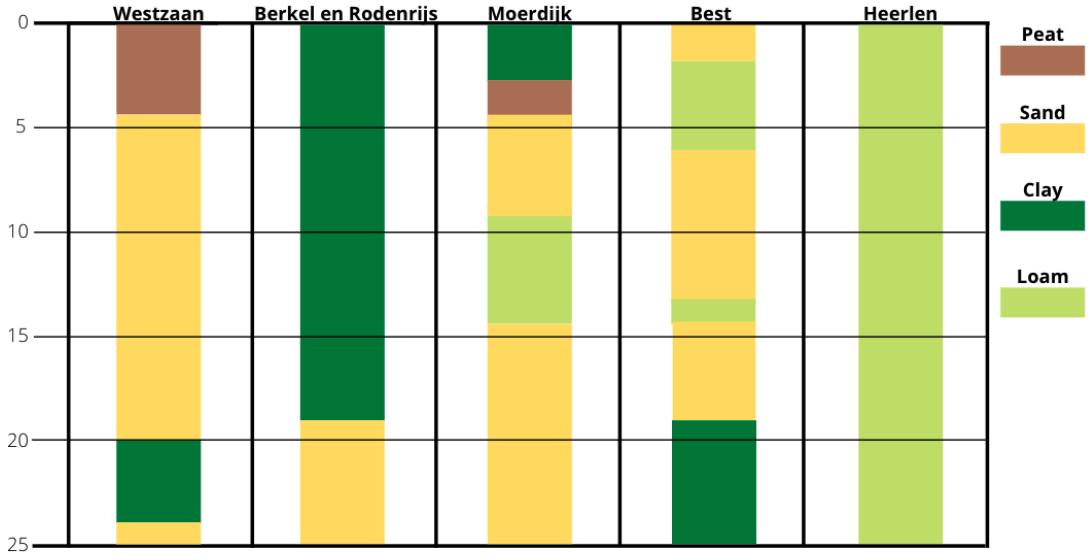


Figure 10: A schematic depiction of the general soil layout of the projects used in this research.

The database consists of 439.989 data points (after preprocessing, discussed in Section 3.2). The available data are comprised of 2 different sources; pile driving data and CPT data.

3.1.1 Pile Driving Data

The pile driving data contain the penetration speed as a function of depth in intervals of 10 cm. This is measured by the Fundie Registration System (FRS). This system consists of several sensors mounted on the pile driving installation. These sensors continuously collect and process into operational variables. Other parameters that are measured by the FRS are the length of pile, pile diameter and pull-down force (i.e., hookload), among others. However, it should be noted that the measurements obtained during the pile driving process cannot be used as input vectors for the relevant models. Examples of this are the length of the pile and the pull-down force.

3.1.2 CPT Data

As explained earlier, during a CPT, a cone is inserted into the ground at a constant rate while measuring multiple parameters: the elevation (z), the friction ratio (R_f), the cone tip pressure (q_c), the shaft pressure (f_s), the inclination of the cone during penetration (i), the cone resistance (F_c), the shaft resistance (F_s) and the total resistance (F_t). The pile driving data of each pile correspond to a specific CPT. The 'pile plan', developed by the construction company, is used in order to select the correct CPT for a pile. The CPT closest to the pile is noted as the corresponding CPT. An example of a pile plan is shown in Figure 11. The CPTs are indicated by $S-1$ and $S-2$ and piles are indicated by the dark colored squares.

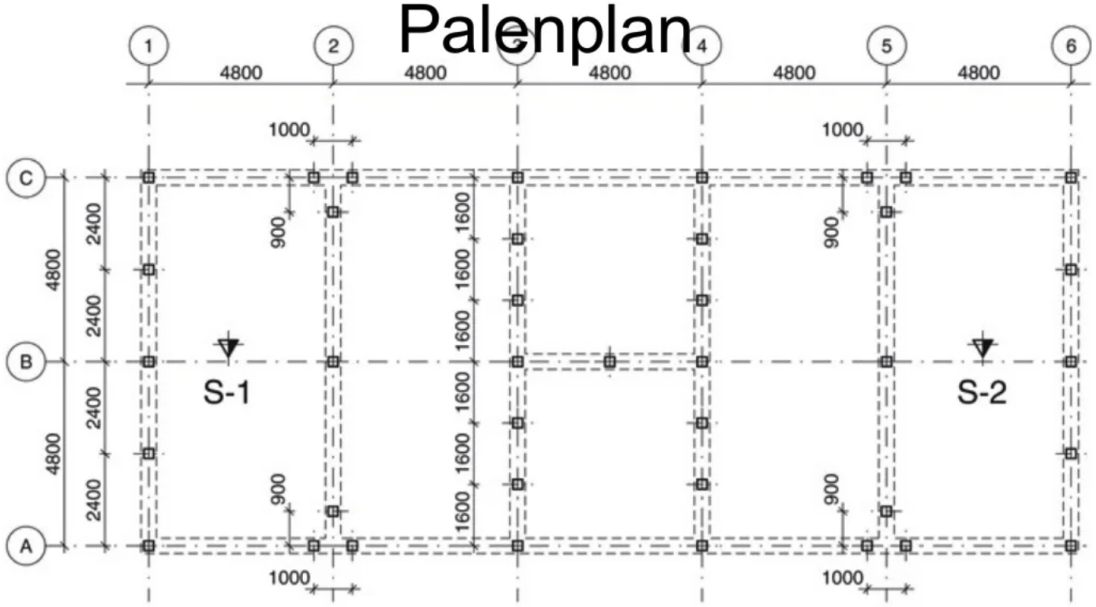


Figure 11: Example of a pile plan (i.e., 'palenplan' in Dutch).

3.2 Data Preprocessing

Although this research aims to utilize the capability of machine learning models to handle raw data (i.e., without the need for manual feature engineering), a few simple data preprocessing steps were taken to optimize model performance and achieve the desired regression behavior. For this research, three different preprocessing steps were undertaken. First, piles not meeting the pile length requirement were removed from the data set. These are piles that are either too long due to operator negligence, or too short due to excessive soil resistance. In this case, a second pile is driven into the ground right next to the unusable pile. Table 2 gives an overview of the length ranges used to select and remove piles.

Project	**Required Length**	**Accepted Pile Length**
Berkel en Rodenrijs	20.6 m	±2m
Best	5.5 m	±1m
Moerdijk	20.1 m	±2m
Heerlen	7.6 m	±1m
Westzaan	14.8 m	±1m

Table 2: Pile length ranges used to select and remove piles from the data set.

Second, measurement errors in pile data can be detected by the following characteristics:

- project number equals zero and or,
- penetration time equals zero and or,
- volume pile equals 0.

Therefore, these measurements were also removed from the data set.

The data are recorded at disparate sampling rates due to the collection of data from two different sources. The pile driving data are recorded at intervals of 10 cm and CPT data are recorded at intervals of 2 cm. Therefore, it was necessary to implement an alignment method to cope with this difficulty. In this approach, the CPT data were aligned with the pile driving data resulting in 10 cm intervals in which part of the CPT data is omitted. Table 3 gives a list of statistical information about the 9 independent variables and the dependent variable (penetration speed).

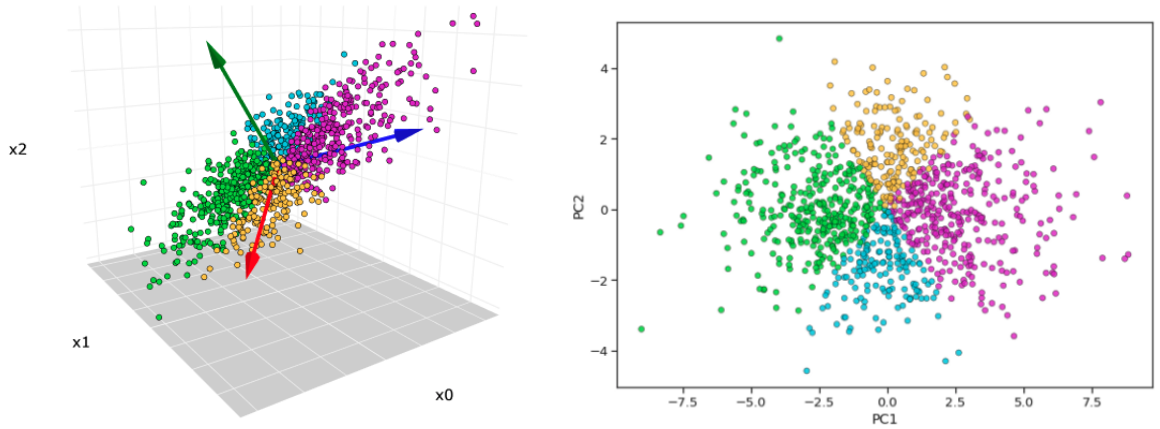
Parameter	Unit	Minimum	Maximum	Average	Standard Deviation
Friction ratio (R_f)	%	-0.021	0.124	0.023	0.019
Shaft pressure (f_s)	MPa	-0.000	1.414	0.312	0.250
Cone pressure (q_c)	MPa	0.000	47.592	5.855	6.578
Elevation (z)	m	-21.800	0.000	8.749	5.944
Inclination (i)	deg	0.000	0.532	0.025	0.023
Pile diameter (D)	m	0.180	0.245	0.214	0.017
Cone resistance (F_c)	MN	0.000	1.578	0.213	0.238
Shaft resistance (F_s)	MN	0.000	1.414	0.312	0.250
Total resistance (F_t)	MN	0.000	2.467	0.525	0.399
Penetration speed (v_p)	m/s	0.000	1.400	0.461	0.223

Table 3: Statistical information of input and output parameters

3.3 Feature Selection

The method of feature selection is the process of choosing a subset of the most influential d parameters of a total set of D characteristics based on the results of an optimization criterion. Reducing the number of input variables is desirable as it shortens the training time, eases the model interpretation and reduces the chance of over-fitting (James et al., 2013). There are various methods to apply the selection of features and choose the most influential independent parameters. Some of the approaches are filter methods, wrapper methods and embedded methods (Chandrashekar and Sahin, 2014). According to Ashrafi et al. (2019), the most robust solutions can be achieved with the use of sequential selection algorithms, a subset of the wrapper methods. These methods use the predictor as a black box and the predictor performance as the objective function to evaluate the subset of variables by sequentially adding features (Chandrashekar and Sahin, 2014).

To start, visualization of the multi-dimensional data is necessary to provide insight into the data and its most influential parameters. Therefore, a Principal Component Analysis (PCA) is performed since it is the most widely used tool to explore similarities and hidden patterns among samples where the relationship in data and grouping is unclear (Granato et al., 2018). PCA is a technique that transforms data in high dimensions into lower dimensions while retaining as much information as possible. This mathematical tool aims to represent the variation present in the data set using a small number of factors. Usually, a two-dimensional or three-dimensional projection of samples is constructed for visual analysis with the axes (principal components, PC) as the factors. A visualization of the reduction in dimensionality by PCA is shown in Figures 12a and 12b.



(a) Original 3-dimensional data set. The red, blue and green arrows are the direction of the first, second, and third principal components.

(b) Scatter plot after PCA reduced from 3-dimensions to 2-dimensions

Figure 12: Dimensionality reduction performed by PCA. Image taken from Cheng (2022)

Results of the PCA analysis on the independent variables used in this research are presented in Figure 13. Variables that contribute similar information are grouped together, that is, they are correlated. Moreover, the distance from the origin also provides information. The further a variable is positioned from the origin, the greater the effect of that variable on the dependent variable. For clarity, the cone pressure (q_c) and the cone resistance ((F_c) are located almost in the same place.

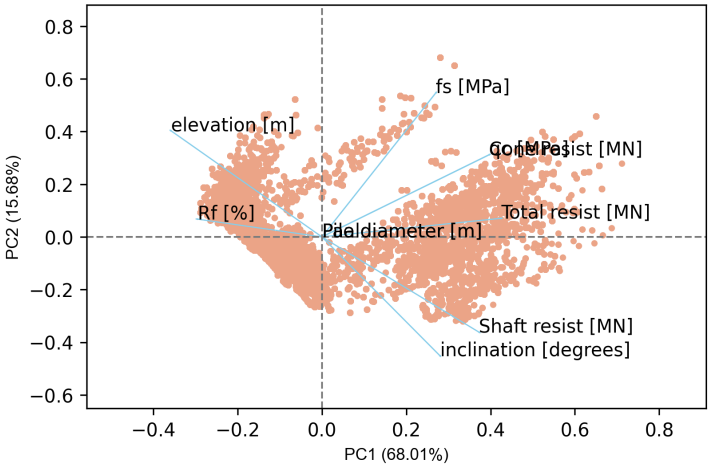


Figure 13: Results of the PCA analysis performed on the independent variables.

A ranking was made based on the absolute values of each feature's location and the importance of each principal component. The ranking, displayed at Table 4, represents the effect of each variable on the dependent variable, starting with the feature with the largest effect (F_1) and ending with the feature with the smallest effect (F_9). A detailed description of the calculations that led to this ranking can be found in Appendix A.1. Appendix A.2 presents how the addition of each feature affects the driveability.

Factor	**Parameter**
F1	Cone pressure (q_c)
F2	Elevation (z)
F3	Shaft resistance (F_s)
F4	Cone resistance (F_c)
F5	Total resistance (F_t)
F6	Shaft pressure (f_s)
F7	Inclination (i)
F8	Friction ratio (R_f)
F9	Pile diameter (D)

Table 4: Ranking of independent variables based on the PCA results.

As mentioned above, the sequential selection algorithm is used to select the optimal set of features. This is done by starting with an empty subset and sequentially adding features. The feature that in combination with the already selected features gives the best result regarding the objective function is permanently added. The search ends when no single attribute expansion improves the current result of the objective function (Aha and Bankert, 1995).

For the feature selection, different architectures of the MLPNN were developed using all available features. The performance of each network was evaluated based on the mean squared error (MSE) (4), using the K-fold Cross Validation (KCV) method.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{4}$$

Where y_i is the ground truth, $\hat{y}_i$ is the model prediction and N is the total number of input values.

The KCV method is widely used to estimate the prediction error as the resampling procedure allows the development of a less biased model (Anguita et al., 2012). By using the KCV method, the data set is split into k subsets; some of the subsets are used to train the model while the remaining subsets are used as validation sets to assess the performance of the model. The performance measure reported by K-fold Cross Validation is then the average of the values computed by the different validation sets. Figure 14 visualizes the KCV method for k = 5. This research considers a fold of 10 as this is the most common value used in machine learning (Ashrafi et al., 2019, Marcot and Hanea, 2021).

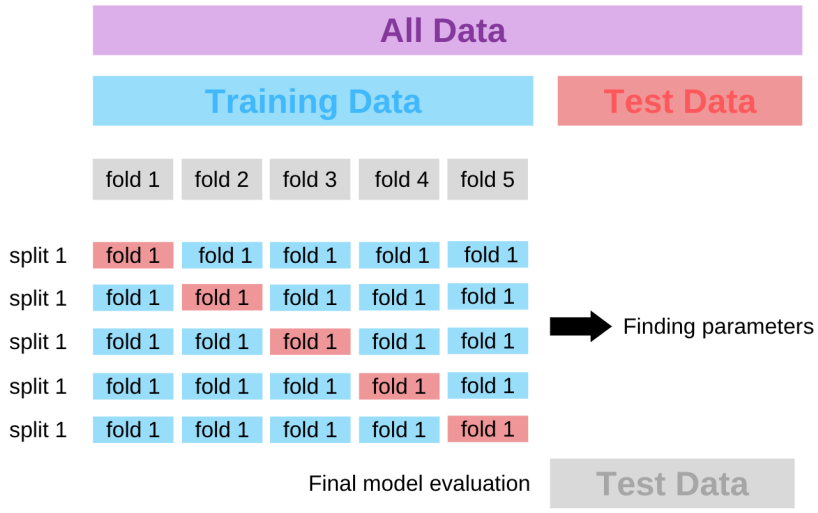


Figure 14: K-fold Cross Validation with k = 5. The blue folds represent the validation subsets and the green folds represent the training subsets.

Through a trial and error approach, it was found that the optimal configuration of MLPNN consists of 3 hidden layers with 6, 7 and 7 neurons, respectively. The sigmoid function was used as the activation function. The results of the feature selection according to the sequential selection algorithm are presented in Table 5. The optimal set of features, indicated in bold, consists of the features cone pressure (q_c), cone resistance (F_c), shaft resistance (F_s) and elevation (z), respectively.

Number of inputs	Selected inputs	Train MSE
1	F1	0,0370
2	F1, F4	0,022
3	F1, F4, F3	0,019
4	**F1, F4, F3, F2**	**0,015**
5	F1, F4, F3, F2, F5	0,020

Table 5: Results of the feature selection according to the sequential selection algorithm. The optimal set of features is visualized in bold.

3.4 Data Scaling

Lastly, the input features must be scaled to improve convergence (LeCun et al., 2012). Specifically, standardization is used to scale the input data, as this method was found to be the most suitable for regression models (García et al., 2016, Marquardt, 1980). Standardization of the data is performed using the following equations:

$$z_i = \frac{x_i - \mu}{\sigma} \tag{5}$$

with mean

$$\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i) \tag{6}$$

and standard deviation

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{7}$$

Where x_i is the network's ith input value and N is the total number of input values. The standardization scaler was fitted on the training features and used to transform the testing features to avoid any data leakage during testing of the model (Anysz et al., 2016).

3.5 Conclusion

This chapter has provided details on the data sets used for this research and the data handling process. First, the different data sets were presented, giving insight into the location, size and lithology. Then, the preprocessing steps were introduced. This was followed by the feature selection process which consisted of a PCA analysis and the utilization of the sequential selection algorithm. The optimal set of input features was found to consist of the cone pressure (q_c), cone resistance (F_c), shaft resistance (F_s) and elevation (z). Lastly, a data scaling method was used to scale the input data.

The next chapter gives a detailed explanation of the development of the ANNs used in this research.

4 Model Development

An important part of the development of a neural network is the optimization of the model. This process consists of two parts: optimization of the trainable parameters and optimization of hyperparameters (Patterson and Gibson, 2017). Trainable parameters are the internal model parameters; they are learned from the data during training as the model attempts to map input to output. Optimization of internal parameters is done by minimizing a predefined loss function. Hyperparameters are parameters whose values control the learning process, these can be tuned to make networks train better and faster. Hyperparameter optimization is done by testing different combinations of hyperparameters to empirically determine which settings result in the best model performance.

This chapter presents the development of the ANNs by providing information on the optimization process. As mentioned in section 2.4, the MLPNN and RBFNN are considered promising neural network architectures. It was also concluded that the particle swarm optimization (PSO) algorithm is a promising training algorithm and that the backpropagation (BP) algorithm should be implemented as a reference as this method is conventionally used to train neural networks. Among the BP algorithms, the Adam optimizer was selected, as it generally achieves higher accuracy than other BP optimizers (Bock and Weiß, 2019). Also, according to the literature, many scholars have confirmed the efficiency of the Adam algorithm in approximation of geotechnical engineering problems (Ashrafi et al., 2019, Brenjkar et al., 2021).

The neural networks developed in this research are composed of the proposed architectures and optimization algorithms. Figure 15 visualizes these neural networks and their composition. The development of each neural network is covered in this chapter.

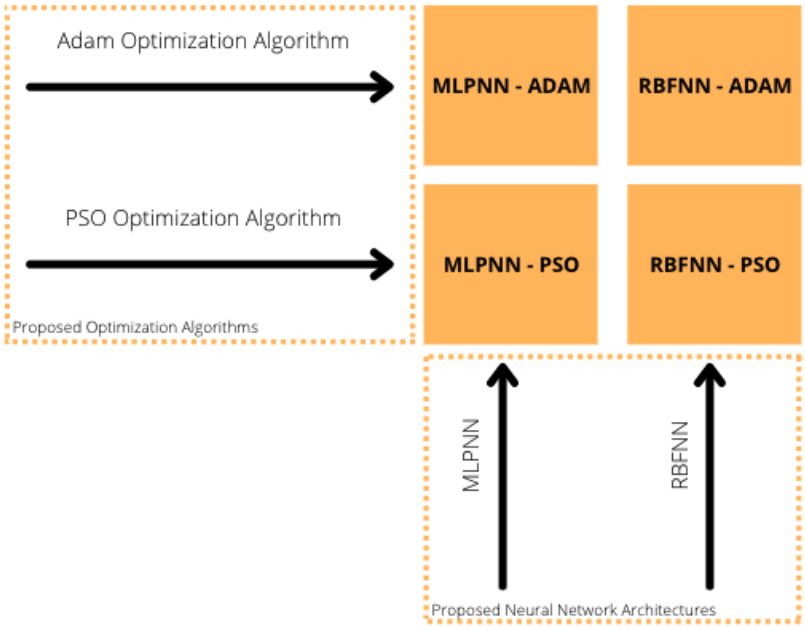


Figure 15: An overview of the developed neural networks. The neural networks are composed of the proposed neural network architectures and optimization algorithms.

The development of the MLPNN and the RBFNN optimized with the Adam optimizer is covered in Section 4.1 and Section 4.2, respectively. The development of the MLPNN and the RBFNN optimized with the particle swarm optimization (PSO) algorithm is covered in Section 4.3 and Section 4.4, respectively. Based on these methods, a novel architecture is presented in Section 4.5.

For the design of the networks, the Keras open-source software library was used within the Python coding environment, which provides built-in tools for the design and training of ML models (Chollet et al., 2015).

4.1 MLPNN-ADAM Development

This research optimizes the internal model parameters of the MLPNN-ADAM by minimizing the MSE, using the Adam algorithm. The Adam optimization algorithm is an extension to stochastic gradient descent (SGD), an optimization algorithm to find the model parameters that correspond to the best fit between predicted and actual outputs. Stochastic gradient descent seeks to minimize a cost function by iteratively updating each parameter θ by a small amount based on the negative gradient of a stochastically selected subset of the given data set (Géron, 2019). A visualization of the SGD method is shown in Figure 16. A detailed explanation of the stochastic gradient descent methodology is beyond the scope of this research, but can be found in Géron (2019)

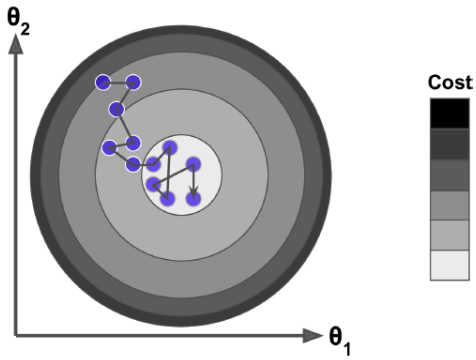


Figure 16: A schematic depiction of the SGD method. Image taken from Géron (2019)

The Adam optimizer involves a combination of two gradient descent methodologies: the Adagrad optimizer Duchi et al. (2011), which works well with sparse gradients, and RMSProp (Tieleman and Hinton, 2012), which works well in on-line and non-stationary settings. A detailed explanation of the Adagrad and RMSP methodologies is beyond the scope of this research but can be found in Géron (2019).

The Adam optimizer inherits the strengths of the above two methods to reach the global minimum efficiently. It does so by taking into account the "exponentially weighted average" of the gradients and the "exponentially moving average" of the squared gradients. An exponential weighted average places a greater weight and significance on the most recent data points allowing faster convergence towards the minima. The exponential moving average allows for an adaptive learning rate α by resolving the diminishing learning rates (Géron, 2019). The updating rule for the Adam optimizer is described as follows:

$$\begin{aligned} m_t &= \beta_1 * m_{t-1} + (1-\beta_1) * (\delta L/\delta w_t) \\ v_t &= \beta_2 * v_{t-1} + (1-\beta_2) * (\delta L/\delta w_t)^2 \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t} \\ w_{t+1} &= w_t - \frac{\alpha}{\sqrt{\hat{v}_t + \epsilon}} * \hat{m}_t \end{aligned} \tag{8}$$

where m_t is the exponential weighted average of the gradient at time t, β_1 is the exponential decay rate for the first moment estimates (i.e., momentum decay), δL is the derivative of the loss (i.e., cost) function, δw_t is the derivative of the weights at time t, v_t is the exponential moving average of the squared gradient at time t, β_2 is the exponential decay rate for the second-moment estimate (i.e., scaling decay), α is the learning rate and ϵ is the smoothing term.
The authors recommend Défossez et al. (2020), Géron (2019) and Haji and Abdulazeez (2021) for further reading on the details of SGD, Adagrad optimizer, RMSProp optimizer and the Adam optimizer.

4.1.1 Hyperparameter Optimization

This section provides an overview of the hyperparameters of MLPNN-ADAM and presents the results of the hyperparameter optimization.

Hyperparameters MLPNN
The model-specific hyperparameters of the MLPNN are the number of hidden layers (n_l), number of hidden neurons (n_h) and the activation function (g). The default hyperparameter settings, as proposed by Géron (2019), Qi et al. (2018), are as follows: number of hidden layers = {1 - 10}, number of neurons = {1 - 20} in each layer and activation function = {sigmoid, hyperbolic tan (i.e., tanh), rectified linear unit (i.e., ReLU)}. Settings found to be most effective in similar studies in the geo-engineering field are used to narrow down the grid search (Ashrafi et al., 2019, Brenjkar et al., 2021).

Hyperparameters ADAM
The hyperparameters of the Adam optimizer are the maximum number of epochs, the batch size, the learning rate (α), the momentum decay (β_1), the scaling decay (β_2) and the smoothing term (ϵ). The default hyperparameter settings are the following: maximum number of epochs = {10 - 250}, batch size = {10 - 450}, learning rate = $\{e^{-1}$ - $e^{-4}\}$, momentum decay = {0.9}, scaling decay = {0.999} and smoothing term = $\{e^{-7}\}$ (Géron, 2019, Qi et al., 2018).

Table 6 summarizes all hyperparameter values for the hyperparameter optimzation of MLPNN-ADAM. The fixed values were found to be most effective in the literature (Géron, 2019, Qi et al., 2018). The baseline values were based on the optimal model settings obtained during the feature selection process.

Setting	**Symbol**	**Baseline value**	**Values to be tested**
Max epochs	-	250	fixed
Batch size	-	256	fixed
Learning rate	α	$5e^{-4}$	fixed
Momentum decay	β_1	0.9	fixed
Scaling decay	β_2	0.999	fixed
Smoothing term	ϵ	e^{-7}	fixed
Number of hidden layers	n_l	3	$[1, 2, 3, 4]$
Number of hidden neurons	n_h	6, 6, 7	$[4, 5, 6, 7, 8]$
Activation function	g	sigmoid	[sigmoid, tanh, ReLU]

Table 6: MLPNN-ADAM hyperparameters.

All possible combinations between the above-presented settings were tested in a hyperparameter grid search optimization. All other hyperparameters (i.e., maximum number of epochs, batch size, learning rate, momentum decay, scaling decay and smoothing term) were kept at their nominal values. To obtain a fair measure of performance, 10-fold cross validation, as described in Section 3.3, was implemented for evaluation. The final hyperparameters found by optimization are presented in Table 7. The optimal structure of MLPNN-ADAM is visualized in Figure 17.

Setting	**Symbol**	**Value or type**
Type of network	-	Feed forward
Training function	-	Adam optimizer
Momentum decay	β_1	0.9
Scaling decay	β_2	0.999
Smoothing term	ϵ	e^{-7}
Number of hidden layers	n_l	2
Number of hidden neurons	n_h	6, 7
Hidden layer activation function	g	Sigmoid
Output layer activation function	g	Linear
Number of inputs	x	4
Number of outputs	y	1
Batch size	-	256
Learning rate	α	$5e^{-4}$
Max epoch	-	250
Stopping condition	-	Achieve minimum error (MSE) or max epoch

Table 7: Specifications of the MLPNN-ADAM architecture.

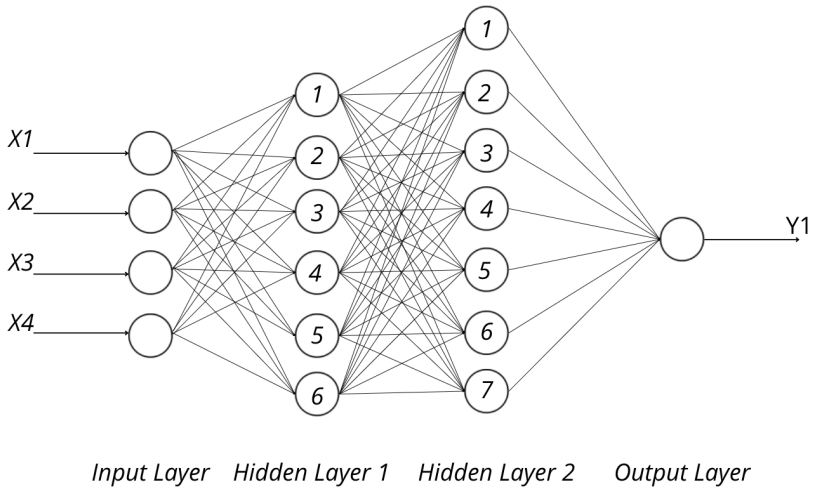


Figure 17: A schematic depiction of the structure of the developed MLPNN optimized with the Adam optimizer.

4.2 RBFNN-ADAM Development

The internal model parameters of the RBFNN were optimized by minimizing the MSE, using the Adam optimizer. As explained in Section 2.3.1, the centers of the radial basis function can be found using the BP algorithm, the selection of random centers, and a clustering algorithm. According to Chen et al. (1992) and Darken and Moody (1991), the K-means clustering algorithms are the most widely used. Therefore, this approach was implemented in this research. A Gaussian function was used to represent the RBF neuron activation function, as this is the conventional and most popular option (Faris et al., 2017). A Gaussian with a one-dimensional input is described as follows:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{9}$$

Where x is the input, μ is the mean, and σ is the standard deviation. The RBF neuron activation function is slightly different and is typically written as:

$$\varphi(x) = e^{-\beta\|x-c_j\|^2} \tag{10}$$

Here, β represents the width of the RBF function. Figure 18 visualizes the effect of β on the width of the activation function.

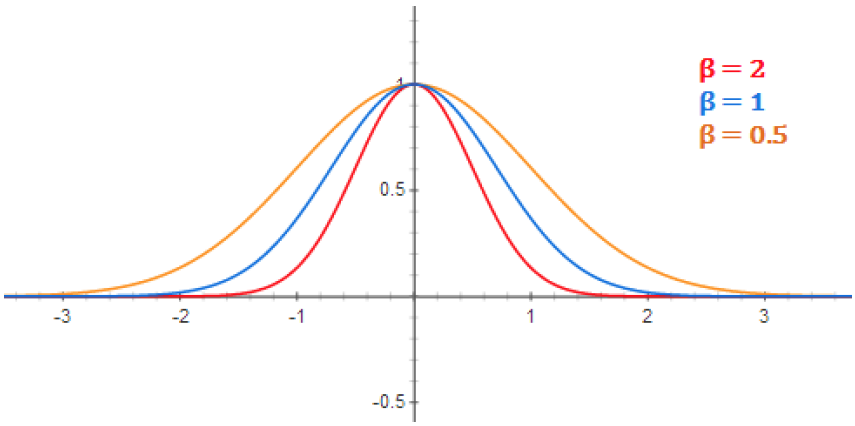


Figure 18: RBF Neuron activation for different values of β. Image taken from Mccormick (2013).

4.2.1 Hyperparameter Optimization

In this section, an overview of the hyperparameters of RBFNN-ADAM is provided and the results of the hyperparameter optimization are presented.

Hyperparameters RBFNN
The number of active RBF centers (n_c) and width of the RBF centers (β) are the model-specific hyperparameters of the RBFNN. Settings found to be most effective in similar studies in the geo-engineering field are as follows: number of active RBF centers = {10 - 100} and width of the RBF centers = {0.5 - 3} (Brenjkar et al., 2021).

Hyperparameters ADAM
As presented above, the hyperparameters of the Adam optimizer include the maximum number of epochs, the batch size, the learning rate (α), the momentum decay (β_1), the scaling decay (β_2) and the smoothing term (ϵ). The default hyperparameter settings were used including the maximum number of epochs = {10 - 250}, batch size = {10 - 450}, learning rate = $\{e^{-1}$ - $e^{-4}\}$, momentum decay = {0.9}, scaling decay = {0.999} and smoothing term = $\{e^{-7}\}$ (Géron, 2019, Qi et al., 2018).

Table 8 summarizes all hyperparameter values. The fixed values were found to be most effective in the literature (Qi et al., 2018).

Setting	**Symbol**	**Baseline value**	**Values to be tested**
Max epochs	-	250	fixed
Batch size	-	256	fixed
Learning rate	α	e^{-3}	fixed
Momentum decay	β_1	0.9	fixed
Scaling decay	β_2	0.999	fixed
Smoothing term	ϵ	e^{-7}	fixed
Number RBF neurons	n_c	-	$[10, 15, 35, 50, 75, 100]$
Width of RBF activation functions	β	-	$[0.5, 1.0, 1.5, 2.0, 2.5, 3.0]$

Table 8: RBFNN-ADAM hyperparameters.

Again, all possible combinations between the above-presented settings were tested in a hyperparameter grid search optimization using 10-fold cross validation. All other hyperparameters (i.e., maximum number of epochs, batch size, learning rate, momentum decay, scaling decay and smoothing term) were kept at their nominal values. The final hyperparameters found by optimization are presented in Table 9. The optimal structure of RBFNN-ADAM is visualized in Figure 19.

Setting	**Symbol**	**Value or type**
Type of network	-	Feed forward
Training function	-	Adam optimizer
Momentum decay	β_1	0.9
Scaling decay	β_2	0.999
Smoothing term	ϵ	e^{-7}
Number RBF neurons	n_c	15
Width of RBF activation functions	β	0.5
Number of inputs	x	4
Number of outputs	y	1
Batch size	-	256
Learning rate	α	e^{-3}
Max epoch	-	250
Stopping condition	-	Achieve minimum error (MSE) or max epoch

Table 9: Specifications of the RBFNN-ADAM architecture.

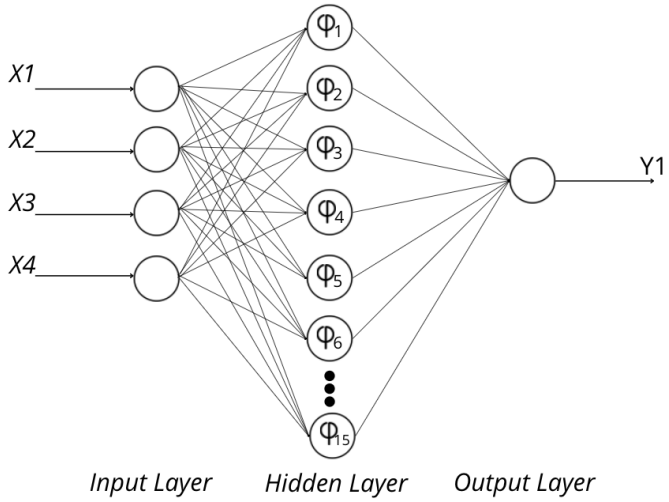


Figure 19: A schematic depiction of the structure of the developed RBFNN optimized with the Adam optimizer.

4.3 MLPNN-PSO Development

The internal model parameters of the MLPNN were optimized by minizing the MSE, using the PSO optimization algorithm. As explained in Section 2.3.2, particle swarm optimization is a population-based stochastic optimization algorithm motivated by intelligent collective behavior of some animals such as flocks of birds or schools of fish (Wang et al., 2018). Figure 20 illustrates the basic algorithm of PSO which consists of three mains steps at each iteration of the process:

1. The best position that each particle has reached thus far, the p_{Best}, is evaluated. The particle's position is updated if the current position is better than the previous position; otherwise, the previous position is kept.

2. The best position of the particles in the entire population, g_{Best} is evaluated.

3. The position and the velocity of the particles is updated using the p_{Best} and g_{Best} (3).

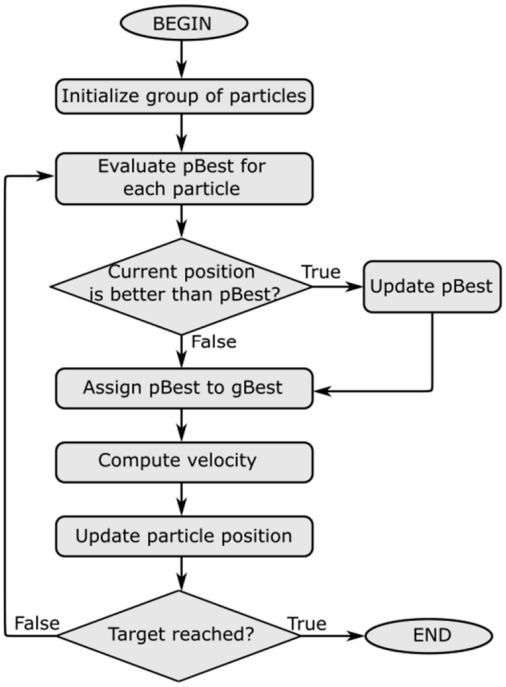


Figure 20: The Particle Swarm Optimization (PSO) algorithm. Image taken from Le et al. (2019).

4.3.1 Hyperparameter Optimization

This section provides an overview of the hyperparameters of MLPNN-PSO and presents the results of the hyperparameter optimization.

Hyperparameters MLPNN
As presented above, the model-specific hyperparameters of the MLPNN are the number of hidden layers (n_l), number of hidden neurons (n_h) and the activation function (g). The hyperparameter settings of the MLPNN were adopted from the MLPNN-ADAM hyperparameter optimization including the number of hidden layers = {1 - 10}, number of neurons = {1 - 20} in each layer and activation function = {sigmoid, hyperbolic tan (i.e., tanh), rectified linear unit (i.e., ReLU)}.

Hyperparameters PSO
The PSO algorithm has five different hyperparameters including the swarm size (n_s), the cognitive term (c_1), the social component (c_2), the inertia weight of the particle (ω) and the number of iterations (N). Each hyperparameter will briefly be discussed below.

The swarm size (n_s): The swarm size is defined by the number of particles in the swarm. Initializing a large swarm allows a larger part of the search space to be covered, however, this increases training times. On the contrary, a very small swarm is likely to take longer to find a solution or even not to find it at all. While empirical studies give a general heuristic of a swarm size of {10 - 30}, the optimal swarm size is problem dependent (Poli et al., 2007). According to Clerc (2010), a swarm size of 15 is suggested as a starting point for experimenting. Similar studies in the geo-engineering field suggest a swarm size of {50 - 80}(Ashrafi et al., 2019, Brenjkar et al., 2021).

The cognitive term (c_1) *and the social component* (c_2): As explained in Section 2.3.2, the constants c_1 and c_2 are also called trust parameters. The c_1 parameter expresses how much confidence a particle has in itself, while c_2 expresses how much confidence a particle has in its neighbors. In case $c_1 > 0$ and $c_2 = 0$, the particles perform a local search and behave as independent hill-climbers. On the contrary, if $c_2 > 0$ and $c_1 = 0$ the swarm turns into one stochastic hill-climber. Therefore, in order to utilize the strength for the cooperative nature of the swarm, c_1 and c_2 should be in good balance (i.e., $c_1 \approx c_2$) (Clerc, 2010). However, Poli et al. (2007) stresses that this ratio is problem dependent. The results of similar studies propose the following ranges: the cognitive term = {0.5 - 3.5} and the social component = {0.5 - 2.5} (Ashrafi et al., 2019, Brenjkar et al., 2021).

The inertia weight (ω): According to Poli et al. (2007), the inertia weight of a particle can be interpreted as the fluidity of the medium in which a particle moves. A high weight value results in a larger step size (i.e., more exploration) whereas a low weight value results in a small step size (i.e., more exploitation behavior). Default setting of the inertia weight = {0.5 - 1.1} (Poli et al., 2007).

The number of iterations (N): A complete iteration corresponds to a single position update of each particle (Clerc, 2010). According to Adıgüzel Mercangöz (2021), the number of iterations is also problem dependent. Too many iterations result in an unnecessary increase in computational complexity whereas a too small number of iterations may terminate the search prematurely. The results of similar studies propose a range of {10 - 100} (Ashrafi et al., 2019, Brenjkar et al., 2021).

Table 10 gives an overview of the hyperparameter values of MLPNN-PSO used for the grid search. Due to time constraints, a selection of the suggestions presented above has been made.

Setting	**Symbol**	**Baseline value**	**Values to be tested**
Swarm size	n_s	-	$[50, 60, 70, 80]$
Cognitive term	c_1	-	$[1.5, 2.5, 3.5]$
Social component	c_2	-	$[0.5, 1.0, 1, 5, 2.0]$
Inertia weight	ω	-	$[0.5, 0.7, 0.9]$
Number of iterations	N	-	$[10, 20, 30, 40, 60, 80]$
Number of hidden layers	n_l	-	$[1, 2, 3, 4]$
Number of hidden neurons	n_h	-	$[4, 5, 6, 7, 8]$
Activation function	g	-	[sigmoid, tanh, ReLU]

Table 10: MLPNN-PSO hyperparameters.

The final hyperparameters that were found through hyperparameter grid search optimization are presented in Table 11. The optimal structure of MLPNN-PSO is visualized in Figure 21. It should be noted that the structure (i.e., number of hidden layers and neurons) of the MLPNN optimized with the Adam optimizer corresponds to the MLPNN optimized with the PSO algorithm.

Setting	**Symbol**	**Value or type**
Type of network	-	Feed forward
Training function	-	PSO optimizer
Swarm size	n_s	80
Cognitive term	c_1	2.5
Social component	c_2	1
Inertia weight	ω	0.7
Number of iterations	N	20
Number of hidden layers	n_l	2
Number of hidden neurons	n_h	6, 7
Hidden layer activation function	g	Sigmoid
Output layer activation function	g	Linear
Number of inputs	x	4
Number of outputs	y	1

Table 11: Specifications of the MLPNN-PSO architecture.

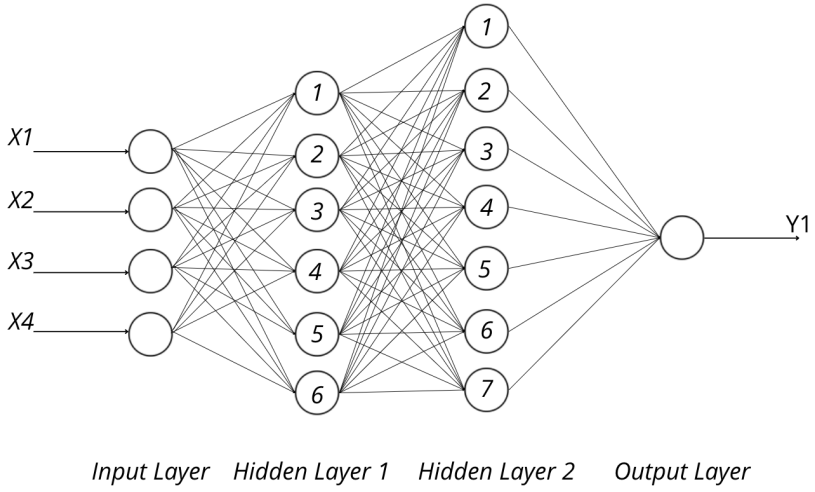


Figure 21: A schematic depiction of the structure of the developed MLPNN optimized with the PSO optimizer.

4.4 RBFNN-PSO Development

The internal model parameters of the RBFNN were optimized by minimizing the MSE, using the PSO optimization algorithm.

4.4.1 Hyperparameter Optimization

In this section, an overview of the hyperparameters of RBFNN-PSO is provided. Additionally, the results of the hyperparameter optimization are presented.

Hyperparameters RBFNN
As presented above, the model-specific hyperparameters of the RBFNN are the number of active RBF centers (n_c) and width of the RBF centers (β). The hyperparameter settings of RBFNN were adopted from the RBFNN-ADAM hyperparameter optimization including the number of active RBF centers = {10 - 100} and width of the RBF centers = {0.5 - 3.0}.

Hyperparameters PSO
The hyperparameters of the PSO algorithm are the swarm size (n_s), the cognitive term (c_1), the social component (c_2), the inertia weight of the particle (ω) and the number of iterations (N). The hyperparameter settings of the PSO were adopted from the MLPNN-PSO hyperparameter optimization including the swarm size = {50 - 80}, the cognitive term = {1.5 - 3.5}, the social component = {0.5 - 2.0}, the inertia weight = {0.5 - 0.9} and the number of iterations = {10 - 80}.

Table 12 gives an overview of the hyperparameter values of RBFNN-PSO used for the grid search.

Setting	**Symbol**	**Baseline value**	**Values to be tested**
Swarm size	n_s	-	$[50, 60, 70, 80]$
Cognitive term	c_1	-	$[1.5, 2.5, 3.5]$
Social component	c_2	-	$[0.5, 1.0, 1, 5, 2.0]$
Inertia weight	ω	-	$[0.5, 0.7, 0.9]$
Number of iterations	N	-	$[10, 20, 30, 40, 60, 80]$
Number RBF neurons	n_c	-	$[10, 15, 35, 50, 75, 100]$
Width of RBF activation functions	β	-	$[0.5, 1.0, 1.5, 2.0, 2.5, 3.0]$

Table 12: RBFNN-PSO hyperparameters.

The final hyperparameters that were found through hyperparameter grid search optimization are presented in Table 13. The optimal structure of the RBFNN is visualized in Figure 22. It should be noted that the structure (i.e., number of RBF neurons and width of the activation functions) of the RBFNN optimized with the Adam optimizer corresponds to the RBFNN optimized with the PSO algorithm.

Setting	**Symbol**	**Value or type**
Type of network	-	Feed forward
Training function	-	PSO optimizer
Swarm size	n_s	80
Cognitive term	c_1	2.5
Social component	c_2	1
Inertia weight	ω	0.7
Number of iterations	N	20
Number RBF neurons	n_c	15
Width of RBF activation functions	β	0.5
Number of inputs	x	4
Number of outputs	y	1

Table 13: Specifications of the RBFNN-PSO architecture.

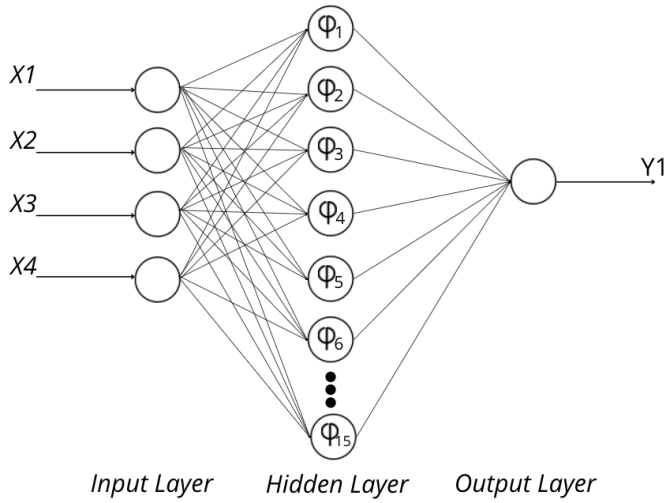


Figure 22: A schematic depiction of the structure of the developed RBFNN optimized with the PSO optimizer.

4.5 Vibratory Driveability Model

As mentioned earlier, this research proposes a novel architecture that utilizes the fruitful characteristics of the models. The motivation behind the development of this model, the Vibratory Driveability (VD) model, is explained in Section 4.5.1. The optimization process of its hyperparameters is presented in Section 4.5.2.

4.5.1 Motivation VD Model

Analysis of the performance of the MLPNN-ADAM, RBFNN-ADAM, MLPNN-PSO and RBFNN-PSO allows for the development of the VD model. The development process, visualized in Figure 23, consists of two parts: the development of the ANN architecture and the development of the optimization algorithm. The motivation behind the choice of architecture and optimization algorithm is explained below.

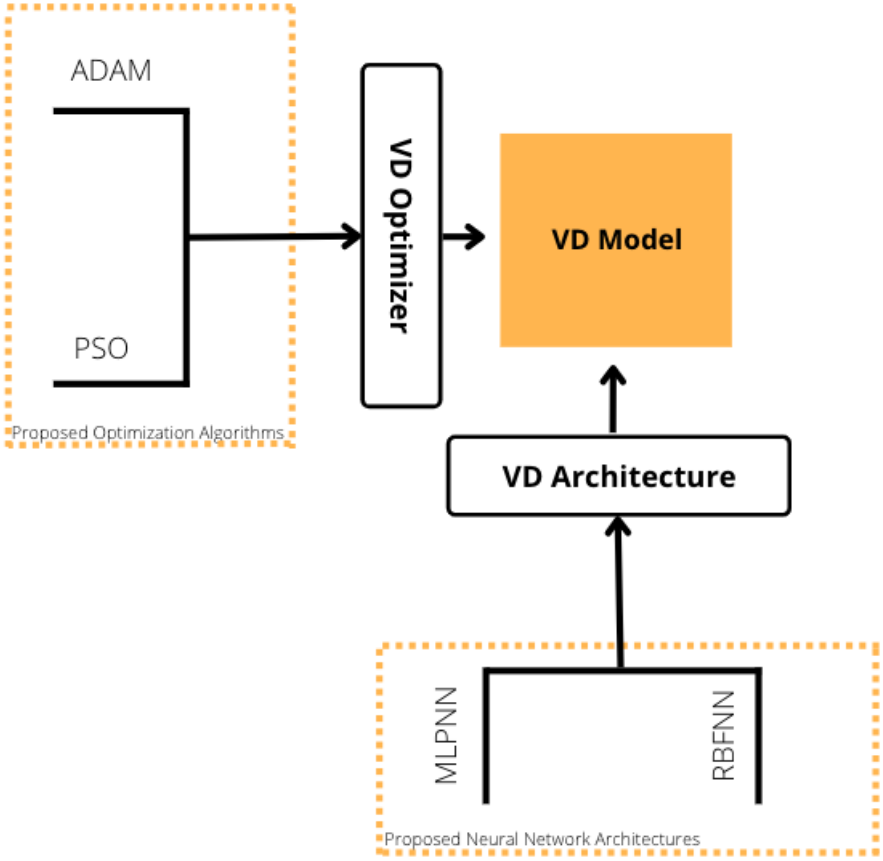


Figure 23: A schematic depiction of the development process of the VD model consisting of two parts: the development of the ANN architecture and the development of the optimization algorithm.

Motivation ANN Architecture

The RBFNN makes use of the radial basis function that describes the difference between an input vector and a predefined center vector. This research uses the K-means clustering algorithm to cluster the data to determine the RBF unit centers. Figure 24 shows the results of the PCA analysis performed on the independent variables, where the blue arrows highlight the direction of the variation associated with the input features used in this research.

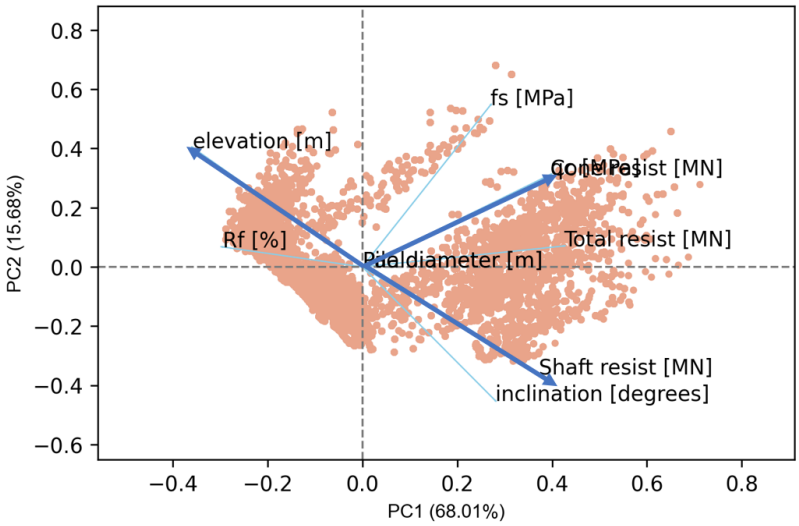


Figure 24: Results of the PCA analysis performed on the independent variables. The blue arrows highlight the input features used in this research.

As can be seen in this figure, the variation associated with input features including the cone pressure (q_c), cone resistance (F_c) and elevation (z) is represented with densely located data points (i.e., many data points are located around the same space). However, this is not the case for data points with variation associated with the shaft resistance (F_c). This results in more clusters being fitted around the data points with variations associated with the cone pressure, cone resistance and elevation. As mentioned, the importance of a data point is determined by its distance to the unit center. Data points with variations related to the shaft resistance are less likely to be close to many unit centers compared to data points related to the other features. Therefore, the predictions computed by the RBF layer are based mainly on the cone pressure, cone resistance and elevation and less on the shaft resistance.

This statement can be substantiated by the results that were obtained during training of MLPNN and RBFNN, both optimized with the Adam optimizer. Figure 25 shows the prediction of the penetration rate for the projects Berkel and Rodenrijs and Westzaan. The predictions produced by MLPNN and RBFNN are shown as well as the input features' measurements from the corresponding CPT. It can be observed that RBFNN is more prone to fluctuations compared to MLPNN. This is because the predictions obtained with the RBFNN are based more on the cone pressure and cone resistance and generally these measurements fluctuate. The shaft resistance is generally less fluctuating hence the predictions obtained with MLPNN contain fewer fluctuations.

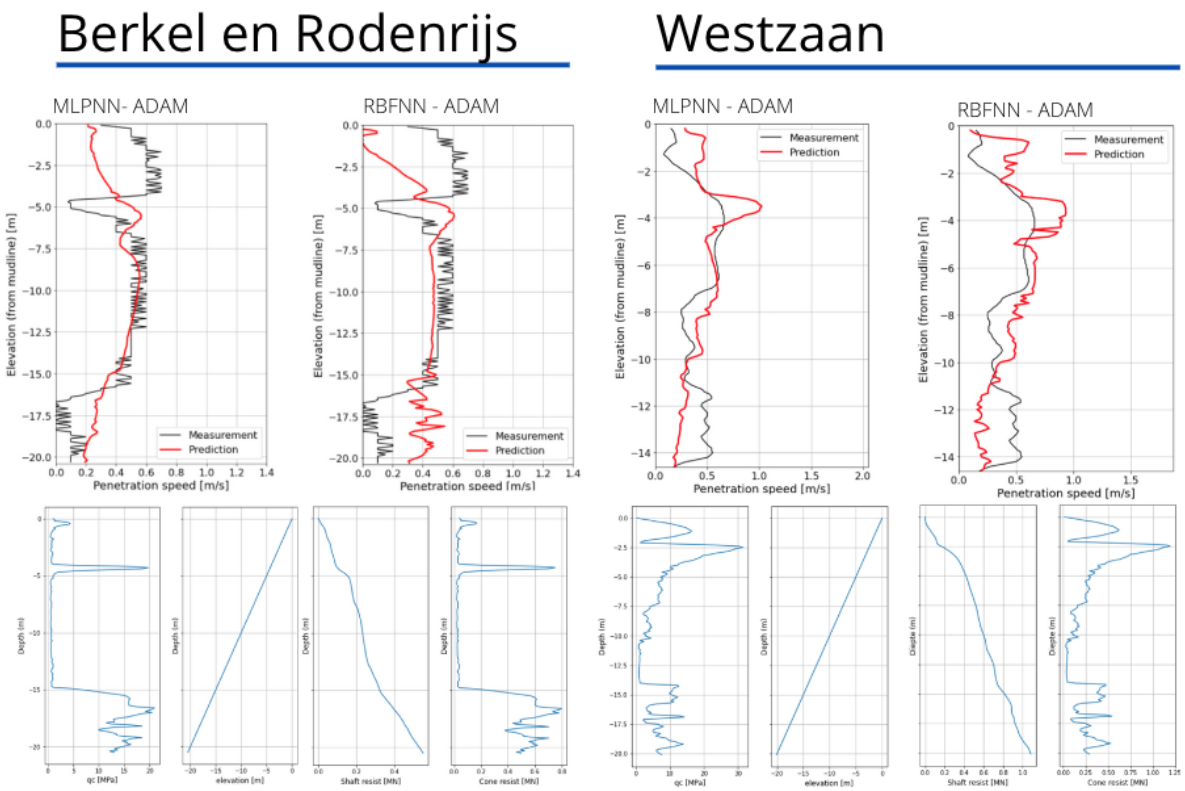


Figure 25: Predictions of the MLPNN and RBFNN for two projects and input features of the respective CPT's.

As mentioned in Section 2.2.1, it could be concluded that the shaft resistance has an impact on the driveability of vibratory driven piles. This would suggest that MLPNN is the preferred architecture to predict the penetration rate. However, according to Jiang et al. (2022), MLPNN is not strong at approximating highly non-linear functions due to the activation functions used. Hence, when applying MLPNN, a common way to enhance its approximation accuracy is to increase the number of hidden layers and/or the number of neurons per layer. This can results in a complex and large network with high computational costs. To overcome this drawback and develop a more efficient network, this research proposes a novel architecture that combines the two methods. RBFNN has shown the capability to predict highly non-linear functions with one hidden layer. Therefore, the combination of the two methods, visualized in Figure 26, ensures a relatively small and simple network, while also making optimal use of the shaft resistance (F_c) for the prediction of vibro-driveability.

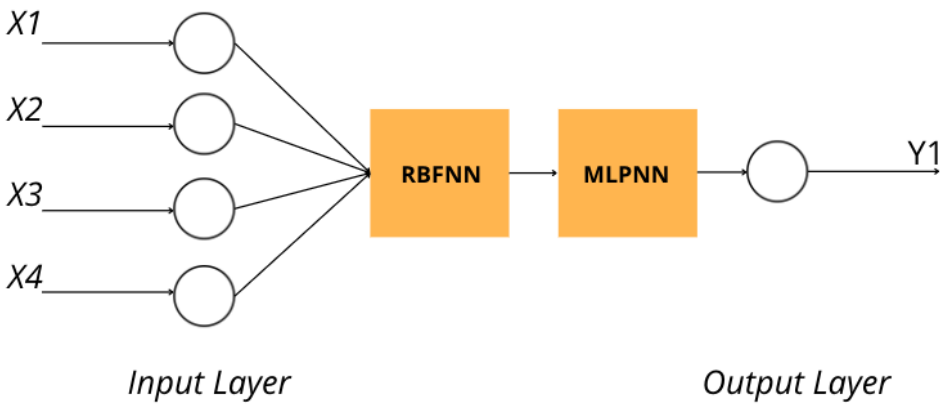


Figure 26: Schematic depiction of the architecture of the VD mode.

Motivation Optimization Algorithm

Gradient descent techniques are known to be limited by a characteristic called the 'local minima' problem. While searching for an optimum solution or global minima, these techniques may encounter local minima from which they cannot escape due to the "steepest descent" nature of the approach. Therefore, they can be far from the optimum or the best solution (Armaghani et al., 2017, Lee et al., 1991, Wang et al., 2004). On the contrary, particle swarm optimization is known for its ability to find global minima. The use of particles increases the probability of finding global minima as particles can jump out of the local minima and exchange information with other particles, hence, converge to optimal solutions. The assumption that the non-linearity and interdepence of the mechanisms at play during vibratory driving result in a complex search space containing multiple local minima, suggests that PSO will outperform Adam.

This intuition is supported by the results that were obtained for two projects optimized with two different algorithms; the Adam algorithm and the particle swarm optimization (PSO) algorithm, shown in Figure 27. It can be observed that the predictions obtained with PSO correspond more to the measured penetration rate compared to the predictions obtained with the Adam optimizer. Additionally, it can be observed that the predictions obtained with PSO generally contain fewer small fluctuations than predictions obtained with the Adam algorithm.

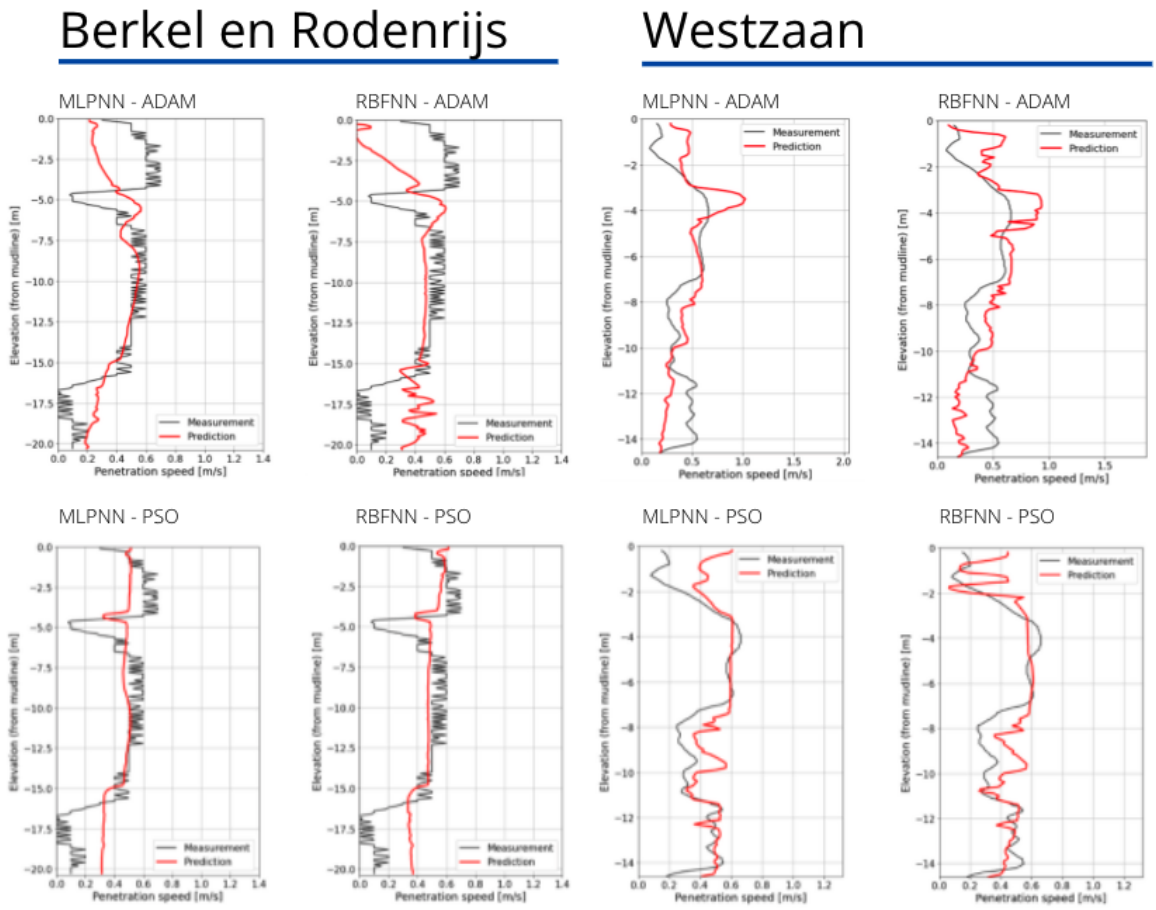


Figure 27: Predictions of the MLPNN and RBFNN optimized with the Adam and PSO algorithm.

As mentioned in Section 2.2.4, the geo-engineer is only interested in the general behavior of the penetration rate rather than specific details such as small fluctuations. Therefore, this research considers PSO as the preferred optimization algorithm as the predictions obtained with this method provide a more accurate representation of the penetration rate and contain fewer small fluctuations compared to the predictions obtained with the Adam optimizer.

4.5.2 Hyperparameter Optimization

The hyperparameters optimized for the VD model include the number of hidden layers (n_l), number of hidden neurons (n_h), number of RBF neurons (n_c), width of the RBF activation functions (β) and number of swarm particles (n_s). Hyperparameter values for the grid search were selected based on the availability of time and on experience obtained during hyperparameter optimization of the MLPNN-ADAM, RBFNN-ADAM, MLPNN-PSO and RBFNN-PSO. The activation function (g) was kept fixed as the sigmoid function was optimal for all neural networks. Also, due to time constraints, the value of the cognitive term (c_1), the social component (c_2), the inertia weight ω and the number of iterations (N) were kept fixed as these hyperparameters contain the widest range of values to be tested. Table 14 gives an overview of the hyperparameter values used for the grid search.

Setting	**Symbol**	**Baseline value**	**Values to be tested**
Swarm size	n_s	-	$[50, 80, 120]$
Cognitive term	c_1	2.5	fixed
Social component	c_2	1.0	fixed
Inertia weight	ω	0.7	fixed
Number of iterations	N	20	fixed
Number RBF neurons	n_c	-	$[3, 10, 15, 20]$
Width of RBF activation functions	β	-	$[0.5, 0.7, 1.0]$
Number of hidden layers	n_l	-	$[1, 2, 3,]$
Number of hidden neurons	n_h	-	$[5, 6, 7, 8]$
Activation function	g	sigmoid	fixed

Table 14: VD model hyperparameters.

All possible combinations were tested in a hyperparameter grid search optimization. The final hyperparameters found by optimization are presented in Table 15. It is noteworthy that the optimal model configuration corresponds to the hyperparameter settings of the MLPNN-PSO and RBFNN-PSO.

Setting	**Symbol**	**Value or type**
Type of network	-	Feed forward
Training function	-	PSO optimizer
Swarm size	n_s	80
Cognitive term	c_1	2.5
Social component	c_2	1.0
Inertia weight	ω	0.7
Number of iterations	N	20
Number RBF neurons	n_c	15
Width of RBF activation functions	β	0.5
Number of hidden layers	n_l	2
Number of hidden neurons	n_h	6, 7
Hidden layer activation function	g	Sigmoid
Output layer activation function	g	Linear
Number of inputs	x	4
Number of outputs	y	1

Table 15: Specifications of the VD architecture.

The optimal structure of the VD model is shown in Figure 28.

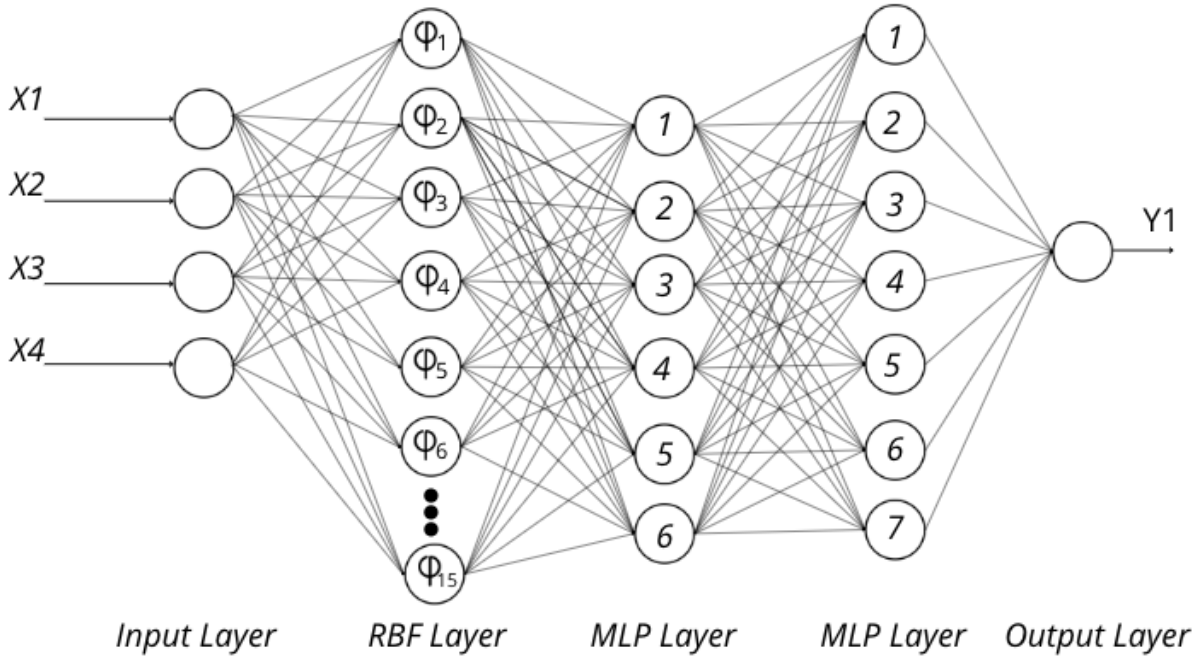


Figure 28: A schematic depiction of the structure of the developed VD model.

4.6 Conclusion

The objective was to develop various machine learning models by optimizing the learnable parameters and the hyperparameters. For optimization of the learnable parameters, the MSE was minimized using the Adam optimizer and the particle swarm optimization method. A grid search optimization using 10-fold cross validation was used to determine the optimal set of hyperparameters for the respective models. Then, a novel ANN was developed that utilizes the fruitful characteristics of the models. Combining MLPNN and RBFNN allows for a relatively small network while also making optimal use of all input features. This architecture is optimized with PSO as the results obtained with this method better match the measured penetration rate and the geo-engineer's preferred prediction behavior compared to the results obtained with the Adam optimizer.

It should be noted that the optimized hyperparameter's values of the VD model correspond to the hyperparameter values of MLPNN-PSO and RBFNN-PSO. What is also interesting to note, is that increasing the complexity of the neural network, by for example increasing the number of hidden layers, hidden neurons and RBF neurons, does not result in increased performance. This can be explained by Occam's razor, which states that the simplest explanation is preferable to a more complex one, a principle that is very much alive today in the science of machine learning (Blumer et al., 1987).

Now that the details of optimization and implementation have been established, this writing is proceeded by presenting the results in the next chapter.

5 Results

As mentioned in Section 3.3, an evaluation of the performance of the trained models was performed using the K-fold Cross Validation method. However, using the validation accuracy to validate the performance may yield inflated results, as the model has been optimized to perform well only on the validation set. Therefore, a third data split was introduced: the test split. This means that the model was trained on the training set, validated on the validation set and tested on the test set to evaluate the prediction performance of the trained model. This research considered three different test sets because of the varying soil compositions; the Berkel en Rodenrijs set, the Westzaan set and the Moerdijk set. The results for a uniform soil composition (i.e., the Heerlen test set) are less interesting to evaluate and therefore presented in Appendix B.1. Additional performance results of the Heerlen test set are contained in Appendix B.4.

For each project, the performance of the five developed models (i.e., MLPNN-ADAM, RBFNN-ADAM, MLPNN-PSO, RBFNN-PSO and VD model) was compared with the performance of contemporary vibrodriveability prediction software. These predictions were obtained by a geo-engineer working at Voorbij using the AllWave-VDP program (Bielefeld, 2020). The method used to generate these predictions is explained in Section 2.2.4. There are various statistical performance metrics that can be used to quantitatively assess the performance of these models. For this research, the mean squared error (MSE), mean absolute error (MAE) and the weighted average percentage error ($WAPE$) were used as these metrics are considered to be the most popular and commonly used metrics for machine learning regression models (Botchkarev, 2018, Spüler et al., 2015).

The MSE (4) represents the averaged squared difference between the model prediction and ground truth. Due to the square, large errors are emphasized and have a relatively greater effect on the value of the performance metric. The MSE equals zero in the event that a model has no error and increases with the model error.

The MAE (11) represents the average of the absolute difference between the ground truth and model prediction. The MAE is less influenced by data outliers compared to MSE (i.e., MSE will become much larger compared to MAE in case of data outliers).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{11}$$

Where y_i is the ground truth, $\hat{y}_i$ is the model prediction and N is the total number of input values.

The WAPE (12) is the sum of the absolute error normalized by the sum of the ground truth. Due to its weighted measure, WAPE equally penalizes for under-forecasting or over-forecasting, and doesn't favor either scenario (Louhichi et al., 2012). The WAPE equals zero percent in the event that a model has no error and increases with the model error.

$$WAPE = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{\sum_{i=1}^{n} y_i} * 100\% \tag{12}$$

Where y_i is the ground truth, $\hat{y}_i$ is the model prediction and N is the total number of input values.

Additionally, the desired characteristics of the predictions based on the input of the geo-engineer were used to assess the performance of each model to ensure useful and valuable predictions for the pile driving industry. The test performance for the Berkel en Rodenrijs, Westzaan and Moerdijk projects are presented in Sections 5.1, 5.2 and 5.3, respectively. Section 5.4 presents the conclusions that were drawn from the results. Additional performance results are presented in Appendix B.

5.1 Results Berkel en Rodenrijs

This section presents the results of the predictions on the Berkel en Rodenrijs test set. The Berkel and Rodenrijs construction project consists of 842 piles with a required depth of 20.6 m. Table 16 presents the specifications of the train and test data.

Data subset	Project	Data ratio
Train	Westzaan, Moerdijk Best, Heerlen	60.4 %
Test	Berkel en Rodenrijs	39.6 %

Table 16: Division of train and test set.

Table 17 presents the MSE, MAE and WAPE values for the different models for the test set. The best-performing model is shown in bold. Based on the performance metrics, it can be concluded that for the Berkel en Rodenrijs test set, the prediction error for the ANNs is significantly lower than the prediction error obtained with the contemporary software AllWave. Second, it can be concluded that the VD model achieves the highest performance in terms of MSE, MAE and WAPE.

Model	MSE	MAE	WAPE
MLPNN-ADAM	0.032	0.144	31.2 %
RBFNN-ADAM	0.061	0.192	41.8 %
MLPNN-PSO	0.024	0.128	27.7 %
RBFNN-PSO	0.027	0.136	29.2 %
VD Model	**0.015**	**0.096**	**20.8** %
AllWave	0.225	0.356	77.3 %

Table 17: Performance metrics values for the different models for the test set. The best-performing model is highlighted.

Figure 29 shows the measurements and predictions plotted for the different models for one driven pile.

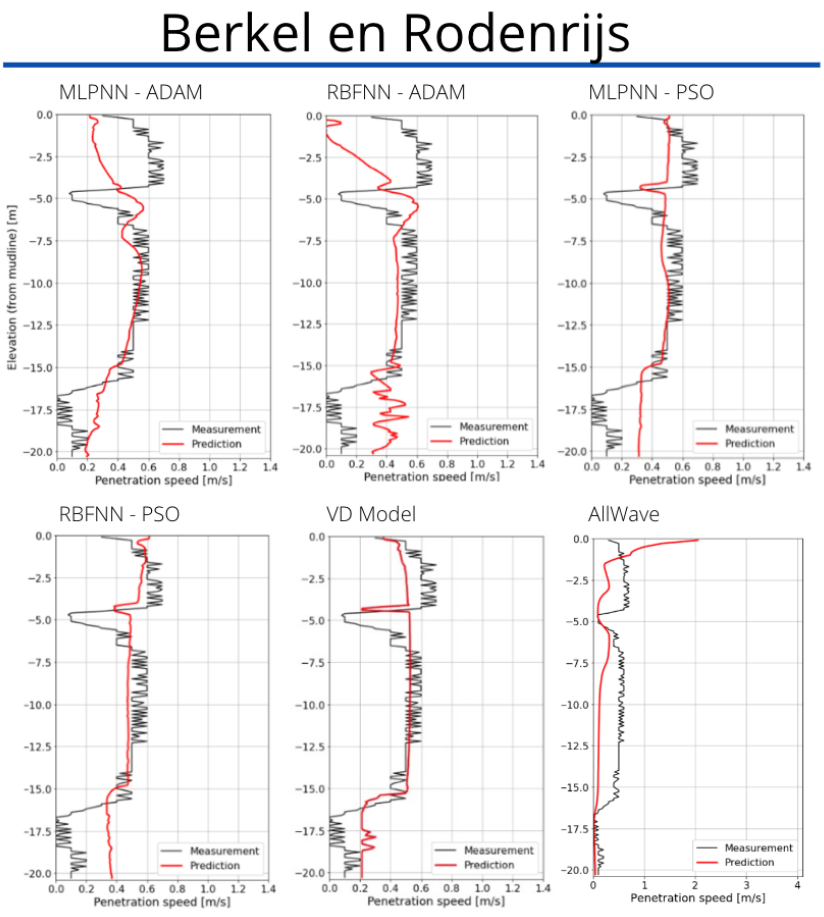


Figure 29: Measurements and predictions plotted for the different models for one vibratory driven pile.

In addition to the performance metrics, the geo-engineer's input should also be taken into consideration when analyzing the performance of the models. As explained earlier, the geo-engineer prefers the model to capture the general behavior of the penetration rate rather than specific details such as small fluctuations. As can be seen in the figure above, the prediction curve obtained with the AllWave software does not match the measured curve, as the model underestimates the penetration rate for most of the curve.

Second, it can be observed that among the ANN models, the VD model best captures the preferences of the geo-engineer. To start, this model only responds to large fluctuations, whereas the models RBFNN-ADAM and MLPNN-ADAM also respond to small fluctuations. Additionally, this model is able to capture the full general behavior of the penetration rate, whereas MLPNN-PSO and RBFNN-PSO only capture part of this behavior (e.g., these models overestimate the deepest measurements).

Additional performance results of the Berkel en Rodenrijs test set are presented in Appendix B.3.

5.2 Results Westzaan

This section presents the results of the predictions on the Westzaan test set. This construction project consists of 126 piles with a required depth of 14.8 m. Table 18 presents specifications on the train and test data.

Data subset	**Project**	**Data ratio**
Train	Berkel en Rodenrijs, Best Moerdijk, Heerlen	95.8 %
Test	Westzaan	4.2 %

Table 18: Division of train and test set.

Table 19 presents MSE, MAE and WAPE values for the different models for the test set where the best-performing model is highlighted. Based on the performance metrics, it is evident that the AllWave model's prediction error is tremendously higher than the prediction error obtained with the artificial neural networks. Second, it can be observed that the RBF-PSO model has the highest performance according to the metrics. However, it should be mentioned that the differences in prediction error between the best scoring model (i.e., the RBFNN-PSO model) and the MLP-PSO model and VD model are minimal.

Model	**MSE**	**MAE**	**WAPE**
MLPNN-ADAM	0.041	0.175	38.8 %
RBFNN-ADAM	0.047	0.178	39.8 %
MLPNN-PSO	0.022	0.118	26.4 %
RBFNN-PSO	**0.020**	**0.116**	**25.7** %
VD Model	0.025	0.130	28.8 %
AllWave	1.855	0.743	165% %

Table 19: Performance metrics values for the different models for the test set. The best-performing model is highlighted.

The performance of each model is also analyzed based on the geo-engineer's preferences. The plotted measurements and predictions for the different models for one driven pile are shown in Figure 30. First, the penetration rate obtained with the AllWave software does not match the measured penetration rate. The model significantly overestimates the penetration rate, making its predictions unusable to the geo-engineer. Second, the prediction curves obtained with the ANN models are evaluated. It can be seen that MLPNN-ADAM and RBFNN-ADAM are unable to accurately capture the general behavior of the curve; the models overestimate the driveability up to a depth of approximately 4 m and underestimate the driveability from a depth of approximately 11 m.

The other neural network models, MLPNN-PSO, RBFNN-PSO and VD model provide a more accurate prediction of the general penetration curve. However, the MLPNN-PSO and RBFNN-PSO contain more small fluctuations compared to the VD model. Additionally, it can be observed that all models fail to predict the first 2 meters accurately. The reason for this is unknown to the authors.

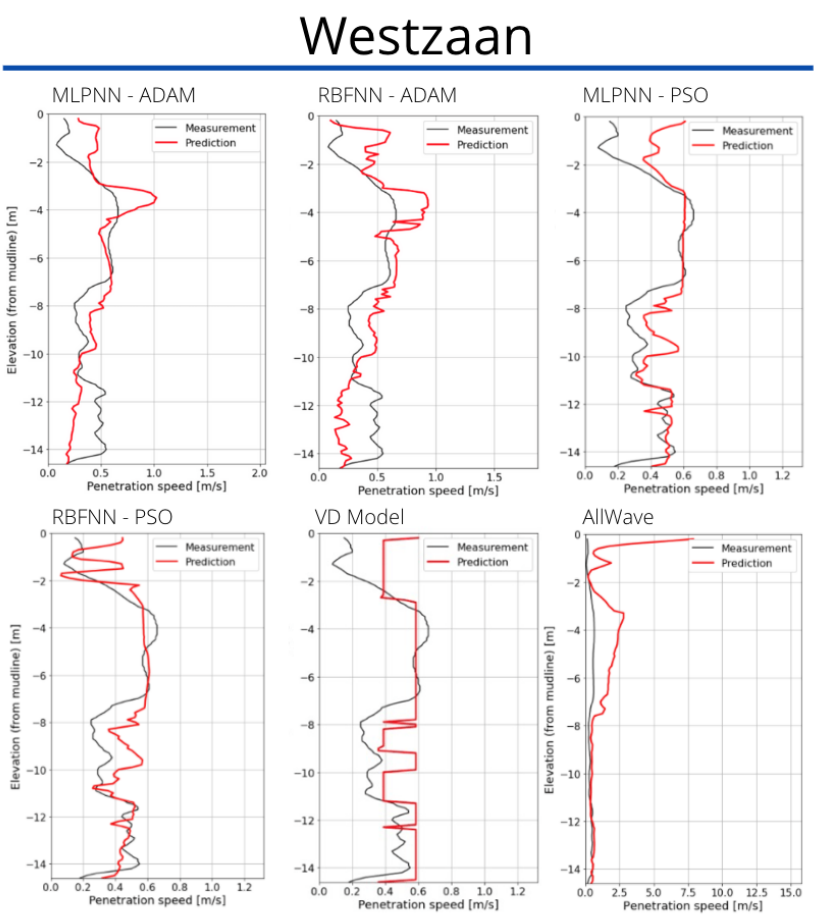


Figure 30: Measurements and predictions plotted for the different models for one vibratory driven pile.

5.3 Results Moerdijk

This section presents the results of the predictions on the Moerdijk test set. This construction project consists of 864 piles with a required depth of 20.1 m. Table 20 presents the specifications on the train and test data.

Data subset	**Project**	**Data ratio**
Train	Berkel en Rodenrijs, Best Heerlen, Westzaan	62.3 %
Test	Moerdijk	37.7 %

Table 20: Division of train and test set.

The MSE, MAE and WAPE values for the different models for the test set are presented in Table 19 where the best-performing model is shown in bold. Also, for the Moerdijk test set, it can be concluded that the prediction error of AllWave is significantly higher compared to the prediction error obtained with the artificial neural networks. Secondly, it can be observed that the RBFNN-PSO model has the highest performance based on the metrics. However, the difference in performance metrics values between the RBFNN-PSO model and the RBFNN-ADAM and VD model is considered to be minimal.

Model	MSE	MAE	WAPE
MLPNN-ADAM	0.055	0.198	50.7 %
RBFNN-ADAM	0.034	0.150	39.1 %
MLPNN-PSO	0.044	0.435	40.9 %
RBFNN-PSO	**0.030**	**0.137**	**35.0** %
VD Model	0.034	0.154	38.7 %
AllWave	2.191	0.992	253.9 %

Table 21: Performance metrics values for the different models for the test set. The best-performing model is highlighted.

Again, the performance of each model is analyzed based on the geo-engineer's preferences. The plotted measurements and predictions for the different models for one driven pile are shown in Figure 31. For this test set, the AllWave software significantly overestimates the penetration rate, and thus the predictions obtained with this software do not provide an accurate representation of the driveability curve. When analyzing the performance of the ANNs, it can be seen that the MLPNN-ADAM and RBFNN-ADAM predictions are not corresponding to the measurements. Combined with the fact that the predictions contain numerous small fluctuations, they are unusable for the geo-engineer. In addition, the MLPNN-PSO is unable to capture the general behavior of the penetration rate. A more accurate representation of the driveability curve is obtained with the RBFNN-PSO and VD model. The difference between these models is the number of small fluctuations; RBFNN-PSO shows small fluctuations whereas the VD model does not respond to small fluctuations. Also, it can be observed that the prediction of both models contains a large fluctuation at -2.5 m. It should be noted that this prediction error is not caused by the model but by an external event influencing the driveability. A more detailed explanation can be found in Appendix B.2.

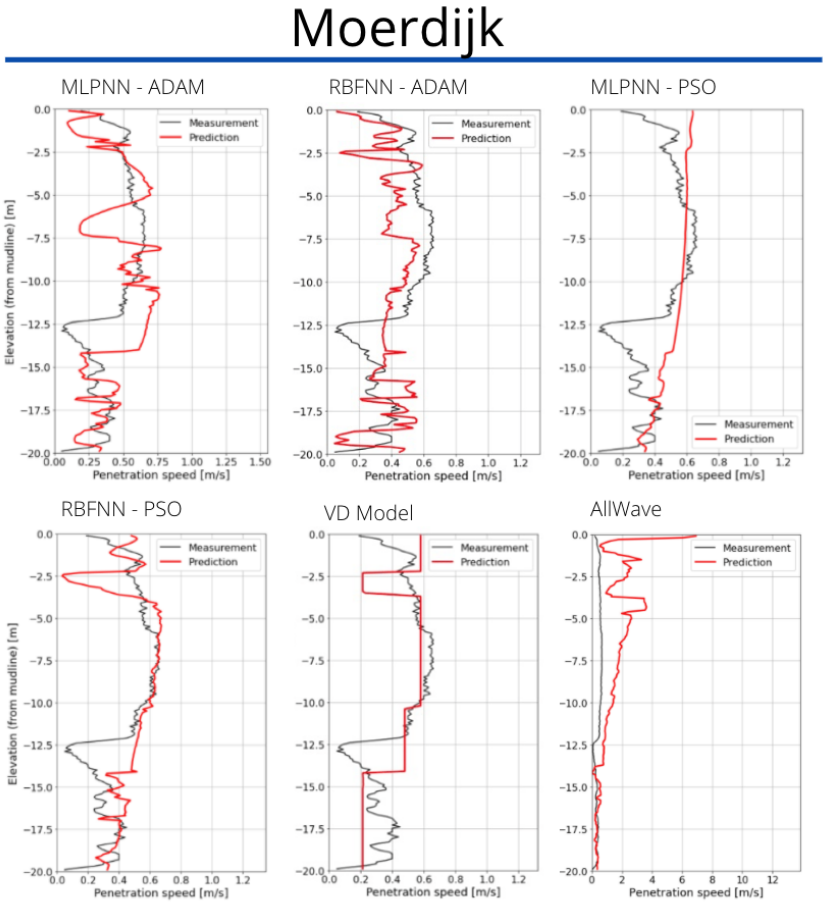


Figure 31: Measurements and predictions plotted for the different models for one vibratory driven pile.

5.4 Conclusion

The results of the developed models were presented for three different projects: the Berkel en Rodenrijs project, the Westzaan project and the Moerdijk project. The performance of the models was analyzed on the basis of statistical performance metrics and the geo-engineer's preference for the prediction behavior.

To start, it can be concluded that the performance of the predictions obtained with artificial neural networks is substantially higher compared to the performance of the predictions obtained with the AllWave software. Results from the AllWave software have a relatively high prediction error with MSE values of {0.225, 1.885, 2.191} and do not capture the general driveability behavior. The performance metrics obtained with the ANNs with MSE varying between {0.015 - 0.061} indicate a significantly higher performance compared to the AllWave software. Moreover, results obtained with the ANNs provide a better representation of the general behavior of the penetration rate.

Second, based on the performance metrics, it can be concluded that the VD model shows the highest performance for the Berkel and Rodenrijs test set while the RBFNN-PSO yields the highest performance for the Westzaan and Moerdijk test sets. Therefore, no model has been developed that consistently performs the best in terms of the selected performance metrics. However, it should be noted that the differences between the best-performing model and the second best-performing model for the Westzaan and Moerdijk test set are minimal with a difference in the MSE value of 0.002 and 0.004, respectively.

To get a more substantiated view of the best-suited model for the prediction of the penetration rate, the input of the geo-engineer is taken into account. Based on this input, it is preferred that the model captures the general behavior of the penetration rate rather than specific details such as small fluctuations. Based on the authors' interpretation, it can be concluded that the models optimized with the PSO method succeed best in this. The models optimized with the Adam optimizer show small fluctuations and generally do not correspond to the driveability curve; hence, the predictions obtained with these models are less useful for the geo-engineer. Second, it can be observed that among the optimized ANNs with PSO, the VD model performs best for the three test cases. This can be substantiated by the following observations:

- In case of the Berkel en Rodenrijs test set, the VD model captures the full general behavior whereas the MLPNN-PSO and RBFNN-PSO only capture part of this behavior. Therefore, the Vibratory Driveability model is considered most useful for the geo-engineer.
- In case of the Westzaan test set, the MLPNN-PSO, RBFNN-PSO and the VD model corresponds to the general behavior of the driveablitiy. However, the VD model shows fewer small fluctuations and is therefore considered the most useful.
- In case of the Moerdijk test set, the RBFNN-PSO and VD model both capture the general behavior, though the Vibratory Driveability model is considered to be most useful as the RBFNN-PSO shows small fluctuations.

Additionally, the Vibratory Driveability model shows robust performance for all three test cases as it was able to generate useful predictions for each project.

The conclusions drawn from the results and other findings of this study allow for the suggestion of improvements and possible extensions to this study. This will be presented in the following chapter.

6 Future Work

This work is the first published research on the application of ANNs for the prediction of vibro-driveability. Therefore, during this research, many new insights have been gained into all aspects of vibratory driving and the effect of these aspects on the penetration rate. To ensure that these insights can be used for the improvement of vibro-driveability prediction performance, a detailed description of possible future work is presented in this section.

6.1 Scope of Performance

The Vibratory Driveability model provides predictions for the Berkel en Rodenrijs, Westzaan en Moerdijk projects that reflect the general behavior of the driveability curve and generally do not contain small fluctuations. Therefore, the predictions obtained with this model can be considered to be useful for the geo-engineer. However, to the best of the authors' knowledge, there is one situation in which the model is not able to provide valuable predictions. Section 6.1.1 explains this situation in detail.

6.1.1 Construction Project Location

The composition of soil layers in the Netherlands varies significantly by location. For example, the soil layout of areas close to the coast or river is fairly monotonous. This means that there is little variation in the composition of the soil layers for a specific area (i.e., the area is composed of the same layers and these are also at the same depth). However, areas inland, especially near Breda, can be composed of many different soil layers. This means that there is much variety in the soil layers and in the composition of these layers. In the first case, CPTs represent the soil layout of the respective area, whereas this does not apply to the second case due to the wide variety of soil layers. This can be seen when comparing the prediction performance for the projects Berkel en Rodenrijs en Best. As shown in Figure 9, Berkel en Rodenrijs is located near the coast and the river Maas (not shown on the map). The composition of the soil layout is relatively monotonous, therefore, CPTs are expected to represent the soil layout of this area accurately. In contrast, Best is located inland, resulting in a varied composition of layers. Therefore, the CPTs are expected to be an inaccurate representation of the soil composition of the entire construction area. The performance metrics for both projects are presented in Table 22. It can be observed that the MSE and MAE values for the Best test set are significantly higher compared to the Berkel en Rodenrijs test set. Therefore, it can be concluded that the prediction performance on the Best test is significantly lower.

	Berkel en Rodenrijs			Best		
Model	**MSE**	**MAE**	**WAPE**	**MSE**	**MAE**	**WAPE**
MLPNN-ADAM	0.032	0.144	31.2 %	0.149	0.314	38.5 %
RBFNN-ADAM	0.061	0.192	41.8 %	0.158	0.325	41.3 %
MLPNN-PSO	0.024	0.128	27.7 %	0.153	0.321	44.6 %
RBFNN-PSO	0.027	0.136	29.2 %	0.163	0.324	45.1 %
VD Model	0.015	0.096	20.8 %	0.181	0.346	48.2 %

Table 22: Performance metrics values for the different models for the projects Berkel en Rodenrijs and Best.

It should be noted that other factors also influence the prediction performance, such as the mode of operation. However, the prediction performance is likely influenced by the degree to which the CPTs represent the soil layout, given the large discrepancies in the performance measurements. Therefore, it is suggested to research in greater detail how well CPTs reflect the soil composition of a particular area and what available options can potentially be leveraged to provide a more accurate representation of the soil layout of a specific area.

In an attempt to create a more accurate representation of the soil composition, lithology measurements from the Dinoloket for every construction project were included in this research. As the lithology measurements include soil classifications (i.e., peat, coarse-grained sand, fine-grained sand), an index was developed that converts the classifications into cone resistances based on characteristic values of soil properties obtained from Backhausen and van der Stoel (2014). The results obtained with the addition of the Dinoloket data were not considered promising, as prediction performance decreased significantly. However, according to the authors, the indexing method used is relatively crude and leaves room for improvement.

6.2 Potential for Performance Improvement

In addition to improving the shortcomings of the model to increase its performance, other aspects can be implemented to reduce the prediction error. Section 6.2.1 and Section 6.2.2 cover these aspects.

6.2.1 Mode of Operation

As explained in Section 2.2.1, the operator can adjust different parameters to improve the penetration rate in case of difficulties during driving. One of the most common methods is to adjust the pull-down force, or hook load. As mentioned in Section 2.2.1, the pull-down force represents the amount of static weight of the hammer resting on the pile. Increasing the pull-down force results in an increased static mass on the pile, which can cause an increase in the penetration rate. During driving, the operator can decide if the pull-down force is used and at what moment it is used. Each operator has his or her behavior in this respect; some use the pull-down force relatively often and some only use it when necessary.

Currently, the driveability predictions are based on CPT measurements, which do not contain any information on the use of the pull-down force. Evidently, the prediction error is relatively high when the use of the pull-down force varies within one construction project. This can be seen in Figure 32, where the measurements and predictions are plotted for several piles with corresponding pull-down force measurements plotted below. The behavior of the operator is visible when studying the leftmost and two rightmost plots: first, the penetration rate increases to a depth of approximately 1 m. Then, when it becomes more difficult to penetrate the soil (i.e., a sudden decrease in penetration rate), the pull-down force is used to increase the driving speed. However, when looking at the second graph on the left, the operator decides to increase the pull-down force at a later stage (i.e., at a depth of 5 m) causing the driveability curve to be different compared to the other plots. The reason for this decision is unknown but can, for example, be due to inattention.

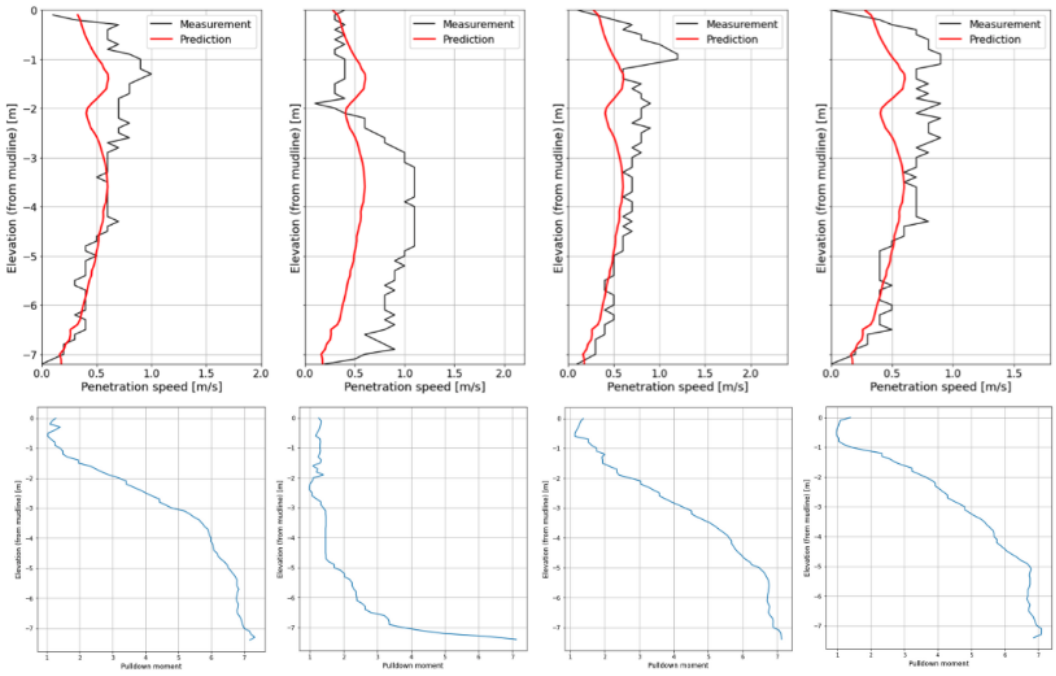


Figure 32: Measurements and predictions and corresponding pull-down force plots for the Heerlen project.

It can be concluded that the mode of operation has an effect on the penetration rate. However, parameters that reflect this mode are measured during the driving process and thus not utilized as input data. Therefore, it is recommended to research in greater detail how the operating behavior of the different operators affects the penetration rate and how this behavior can be included in the model. If the model can "learn" how each operator's mode of operation affects vibro-driveability, prediction performance could be improved.

6.2.2 Domain Expert Knowledge

As mentioned in Section 2.2.4, from the geo-engineer's perspective, a prediction is most useful when it contains the general behavior of the driveability curve. This gives the geo-engineer an overview of the soil structure after which the predictions can be refined based on his or her own experience and knowledge. This includes weather conditions and location-specific information such as saturation level, mineralogy and grain size. The insertion of the domain expert's experience and knowledge is recommended to be researched and implemented. This could result in more accurate driveability predictions and thus reduce or eliminate the need for the geo-engineer's input.

6.3 Conclusion

This chapter presented a detailed description of possible future work. First, the scope of the performance was presented by outlining a limitation of the developed models. Furthermore, possible extensions to this study that could improve the performance of the model were presented.

In the next and final chapter, this work will be concluded.

7 Conclusion

Assessment of vibro-driveability is immensely useful prior to installation as it increases the construction performance and subsequently reduces costs and environmental impact. However, due to assumptions and the lack of methods that accurately represent the complex phenomena at play during vibratory driving, a disparity is obtained between the predictions of modern pile behavior programs and the observed penetration rate. With the significant increase in pile driving data, the usage of machine learning methods is a promising alternative method for the prediction of vibro-driveability. Therefore, this work developed Artificial Neural Network (ANN)-based predictor functions suitable for the prediction of the penetration rate.

The first step was the acquisition of the data set. As such, pile driving data and CPT data were obtained and aligned for five different construction projects. A feature selection process resulted in the optimal set of input features. Then, based on literature review, four promising ANNs were developed by optimizing the internal model parameters and hyperparameters. Based on the fruitful characteristics of these models, a novel architecture was proposed that allows for a relatively simple network while making optimal use of all input features. The performance of the five developed models was compared with the performance of contemporary prediction software based on performance metrics and the geo-eningeer's desired characteristics of the predictions.

Results have shown that the ANN-based predictor functions outperform the traditional model significantly, with MSE of {0.015 - 0.061} and {0.225, 1.885, 2.191}, respectively. As the traditional model fails to cover the non-linear and interdependent interactions during vibratory driving, simplification of the problem and integration of assumptions are needed. The use of data-driven methods allows us to bypass these modifications. Therefore, it can be concluded that the ANN-based method is the preferred alternative for the prediction of vibro-driveability.

In relation to the geo-engineer's desired prediction behavior, it was demonstrated that the Vibratory Driveability model achieved the highest prediction performance for all three test cases, making the predictions useful for the geo-engineer. It can therefore be concluded that, in combination with the model's competitive prediction performance in terms of the performance metrics, the VD model is best suited for the prediction of vibro-driveability. However, a caveat of the proposed method is that a detailed representation of the soil layers must be included for areas in which the CPT measurements do not accurately reflect the soil layout. Furthermore, as observed in the results, the mode of operation should be implemented in the model to improve the prediction performance.

In conclusion, this research established a significant improvement in vibro-driveability predictions by an ANN-based method. As such, it opened doors for further research within the field of predictive modeling in the pile driving industry.

A Feature Selection

In this appendix results on the feature selection process are presented.

A.1 Feature Ranking Calculations

Appendix A.1 presents the results of the calculations leading to the ranking of the independent variables based on the PCA results, shown in Table A.1. The following method was used to compute the importance of each input feature:

$$\text{Total value} = \text{PC1} \cdot \text{PC1 value} + \text{PC2} \cdot \text{PC2 value} = 0.6801 \cdot \text{PC1 value} + 0.1568 \cdot \text{PC2 value} \quad \text{(A.1)}$$

Factor	**Parameter**	**PC1 value**	**PC2 value**	**Total value**
F1	Cone pressure (q_c)	0.398	0.334	0.257
F2	Elevation (z)	0.360	0.380	0.247
F3	Shaft resistance (F_s)	0.389	0.299	0.246
F4	Cone resistance (F_c)	0.414	0.229	0.245
F5	Total resistance (F_t)	0.453	0.025	0.227
F6	Shaft pressure (f_s)	0.190	0.496	0.184
F7	Inclination (i)	0.251	0.229	0.165
F8	Friction ratio (R_f)	0.265	0.169	0.161
F9	Pile diameter (D)	0.128	0.524	0.159

Table A.1: Results of the calculations leading to the ranking of the independent variables based on the PCA results.

A.2 Effect Features on Driveability

Appendix A.2 presents how the addition of each feature affects the driveability using MLPNN-ADAM, starting from feature F1 and ending with features F1 - F9. The ranking of the input features is shown in Table A.2, for clarity. The results for the Heerlen test set are shown in Figure A.1. The features used for the prediction of the vibro-driveability are shown in bold.

Factor	Parameter
F1	Cone pressure (q_c)
F2	Elevation (z)
F3	Shaft resistance (F_s)
F4	Cone resistance (F_c)
F5	Total resistance (F_t)
F6	Shaft pressure (f_s)
F7	Inclination (i)
F8	Friction ratio (R_f)
F9	Pile diameter (D)

Table A.2: Ranking of independent variables based on the PCA results.

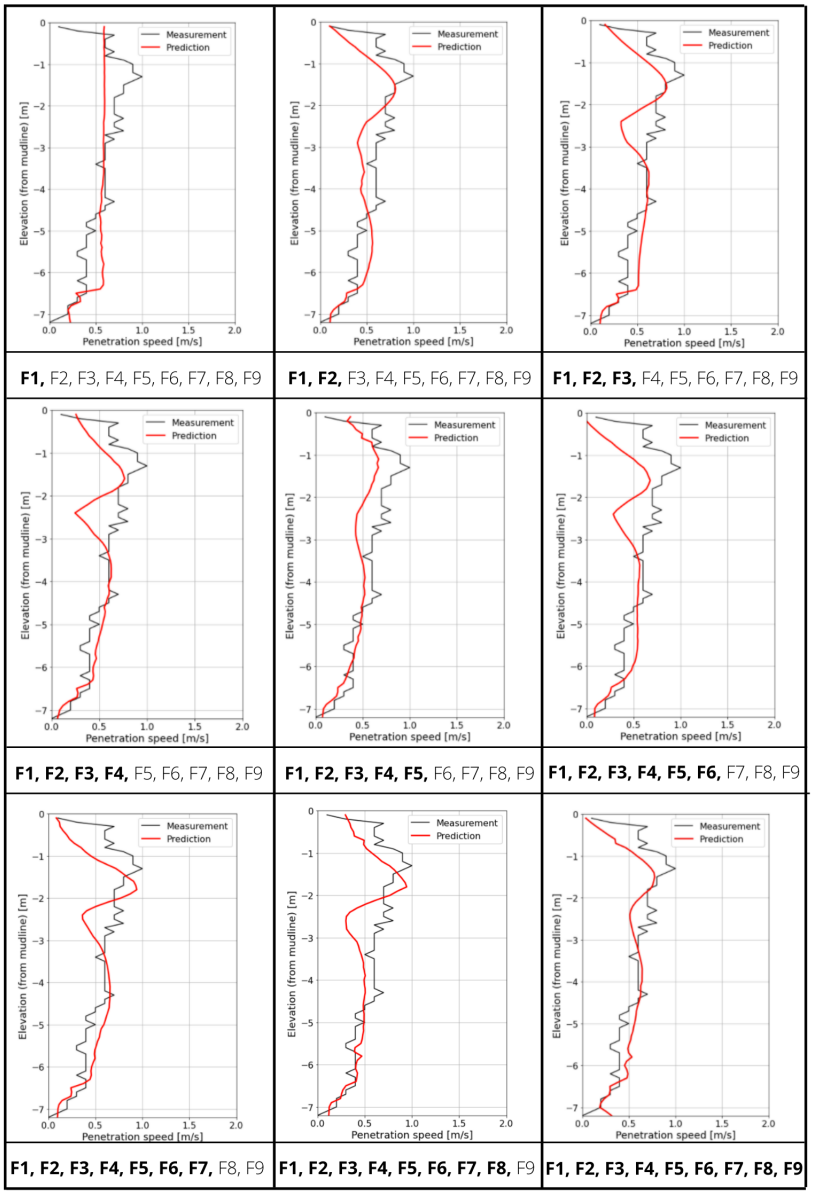


Figure A.1: Effect of input features on the driveability for the Heerlen test set. The features used for the prediction of the vibro-driveability are shown in bold.

B Results

In this appendix additional results as further support to the main matter are presented.

B.1 Results Heerlen

Appendix B.1 presents the results of the predictions on the Heerlen test set. This construction project consists of 515 piles with a required depth of 7.6 m. Table B.1 presents the specifications of the train and test data.

Data subset	**Project**	**Data ratio**
Train	Berkel en Rodenrijs, Best Moerdijk, Westzaan	90.4 %
Test	Heerlen	9.6 %

Table B.1: Division of train and test set.

The MSE, MAE and WAPE values for the different models for the test set are presented in Table B.2. The best-performing model is visualized in bold. The construction company Voorbij did not use the AllWave software to obtain driveability predictions. Therefore, the performance metrics of this model are lacking. It can be observed that the RBFNN-PSO model has the highest performance according to the metrics. However, it should be mentioned that the differences in prediction error between the best scoring model (i.e., the RBFNN-PSO model) and the RBFNN-ADAM and MLPNN-ADAM are minimal.

Model	**MSE**	**MAE**	**WAPE**
MLPNN-ADAM	0.035	0.144	27.8 %
RBFNN-ADAM	0.033	0.146	28.2 %
MLPNN-PSO	0.042	0.168	32.4 %
RBFNN-PSO	**0.031**	**0.137**	**26.3** %
VD Model	0.042	0.171	33.0 %

Table B.2: Performance metrics values for the different models for the test set.

The performance of each model is analyzed based on the geo-engineer's preferences. The plotted measurements and predictions for the different models for one driven pile are shown in Figure B.1. It can be observed that the driveability curves obtained with the ANN models are very similar. All models fail to accurately predict the sudden increase and decrease of the penetration rate measured around -1 m. The CPT measurements do not reflect this change in penetration rate as no sudden decrease and increase in the cone resistance and cone pressure is measured. Therefore, this behavior is very likely to be caused by external events. However, the exact nature of this external event is unknown to the authors. From a depth of 2 m, the models match the measured penetration rate.

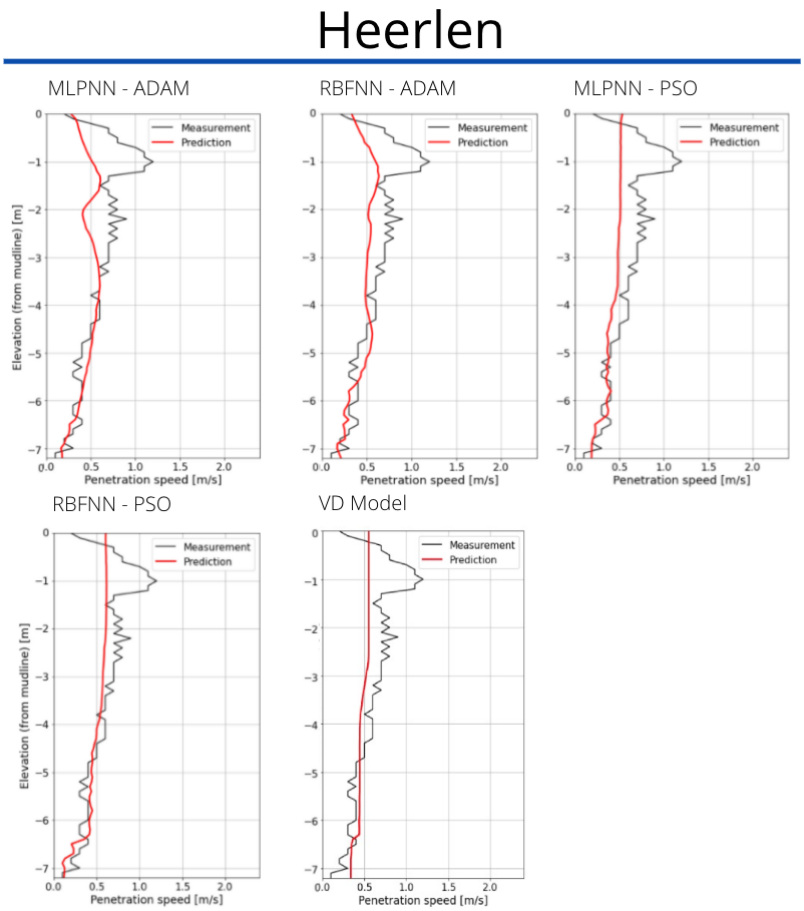


Figure B.1: Measurements and predictions plotted for the different models for one vibratory driven pile.

B.2 External Influence Driveability Moerdijk

Appendix B.2 presents the explanation for the prediction error observed in Section 5.3. In Figure 31, it can be observed that the predictions obtained with RBFNN-PSO and the Vibratory Driveability model contain a relatively large prediction error at a depth of 2.5 m. The authors state that this prediction error is not caused by the models, but rather by an external event influencing the driveability. The proof of this statement is found in the CPT data corresponding to the respective pile, shown in Figure B.4. Here it can be observed that a fluctuation in the cone pressure and cone resistance is measured during the CPT at a depth of 2.5 m. Logically, this would result in a fluctuation in driveability. The reason why this fluctuation was not measured must be due to an external factor influencing driveability. Therefore, it can be concluded that this prediction error was not caused by the models.

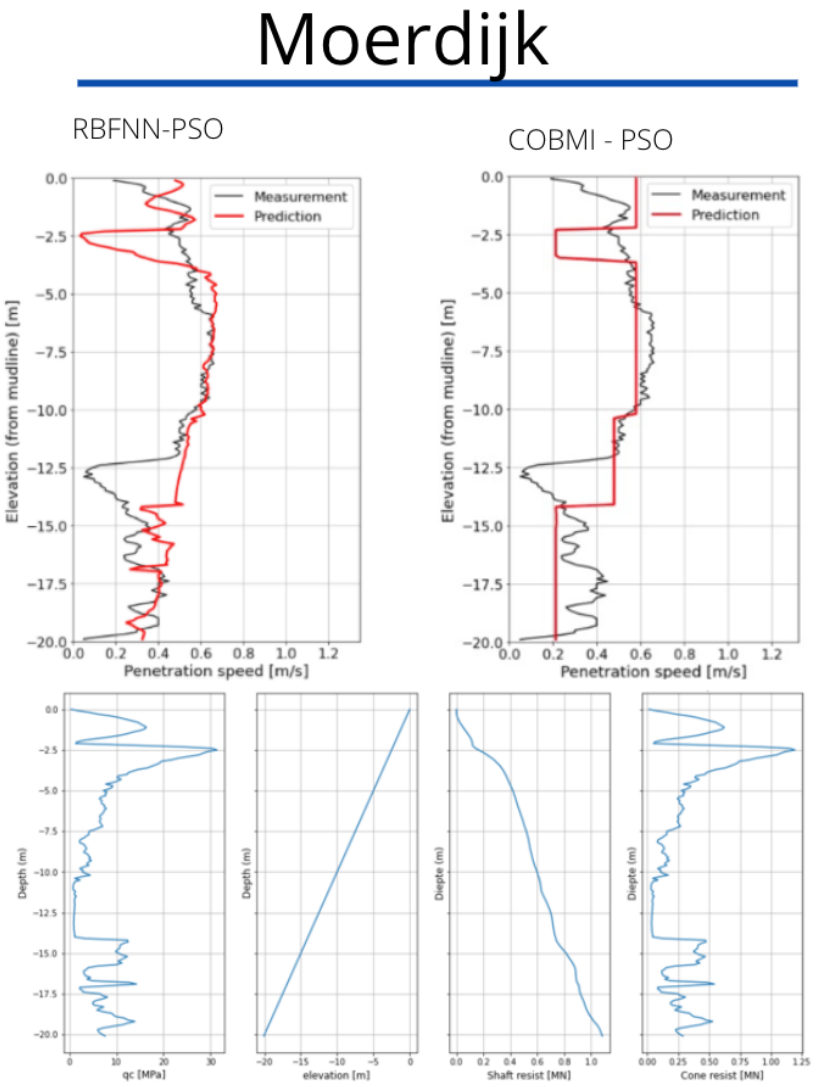


Figure B.2: Predictions of the RBFNN-PSO and VD model for the Moerdijk project and input features of the respective CPT's.

B.3 Additional Results Berkel en Rodenrijs

Appendix B.3 presents the normal distributions of the prediction error for the Berkel en Rodenrijs test set where x is the ground truth and $\hat{x}$ is the model prediction.

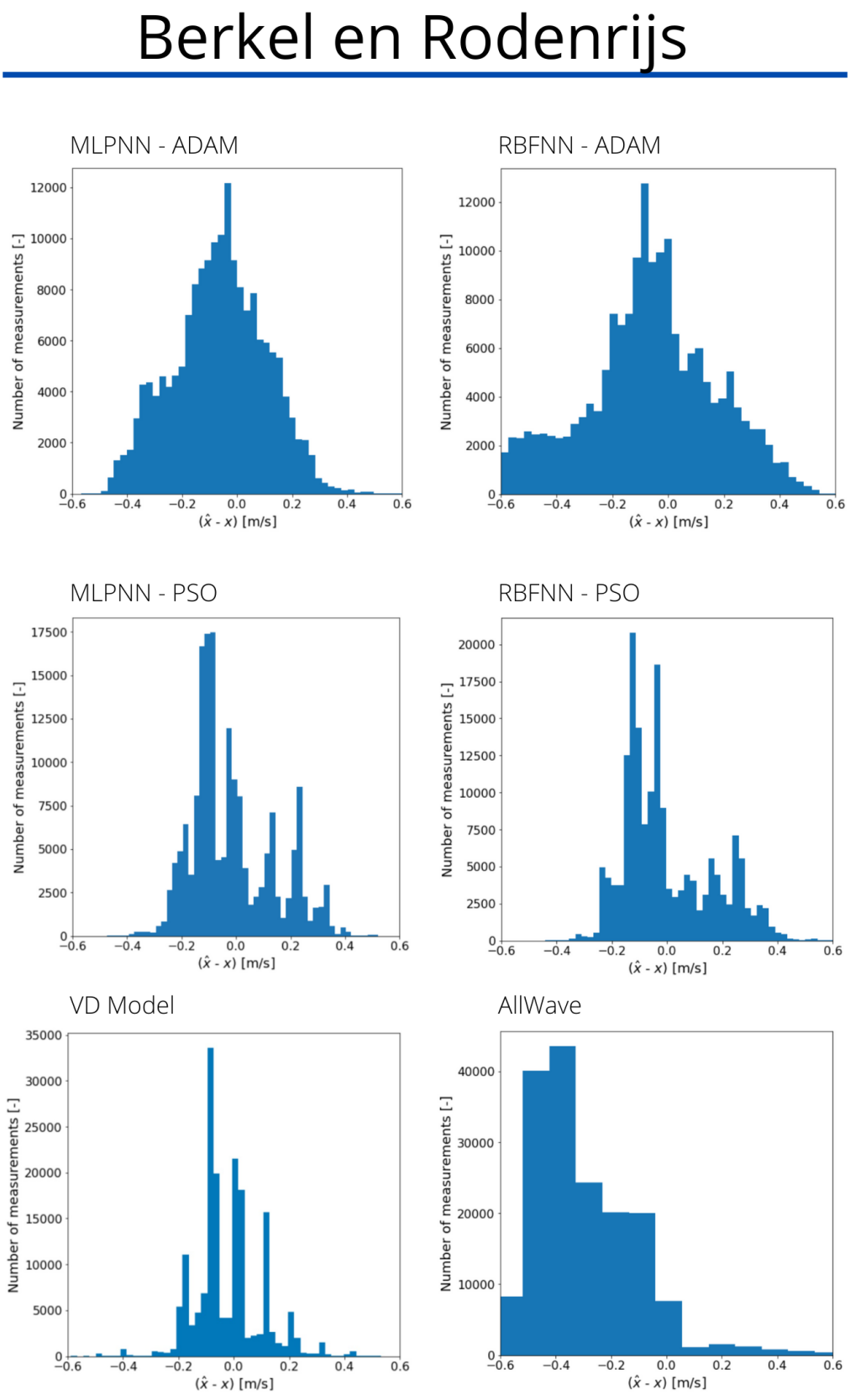


Figure B.3: Normal distributions of the prediction error for the Berkel en Rodenrijs test set.

B.4 Additional Results Heerlen

Appendix B.4 presents the normal distributions of the prediction error for the Heerlen test set where x is the ground truth and $\hat{x}$ is the model prediction.

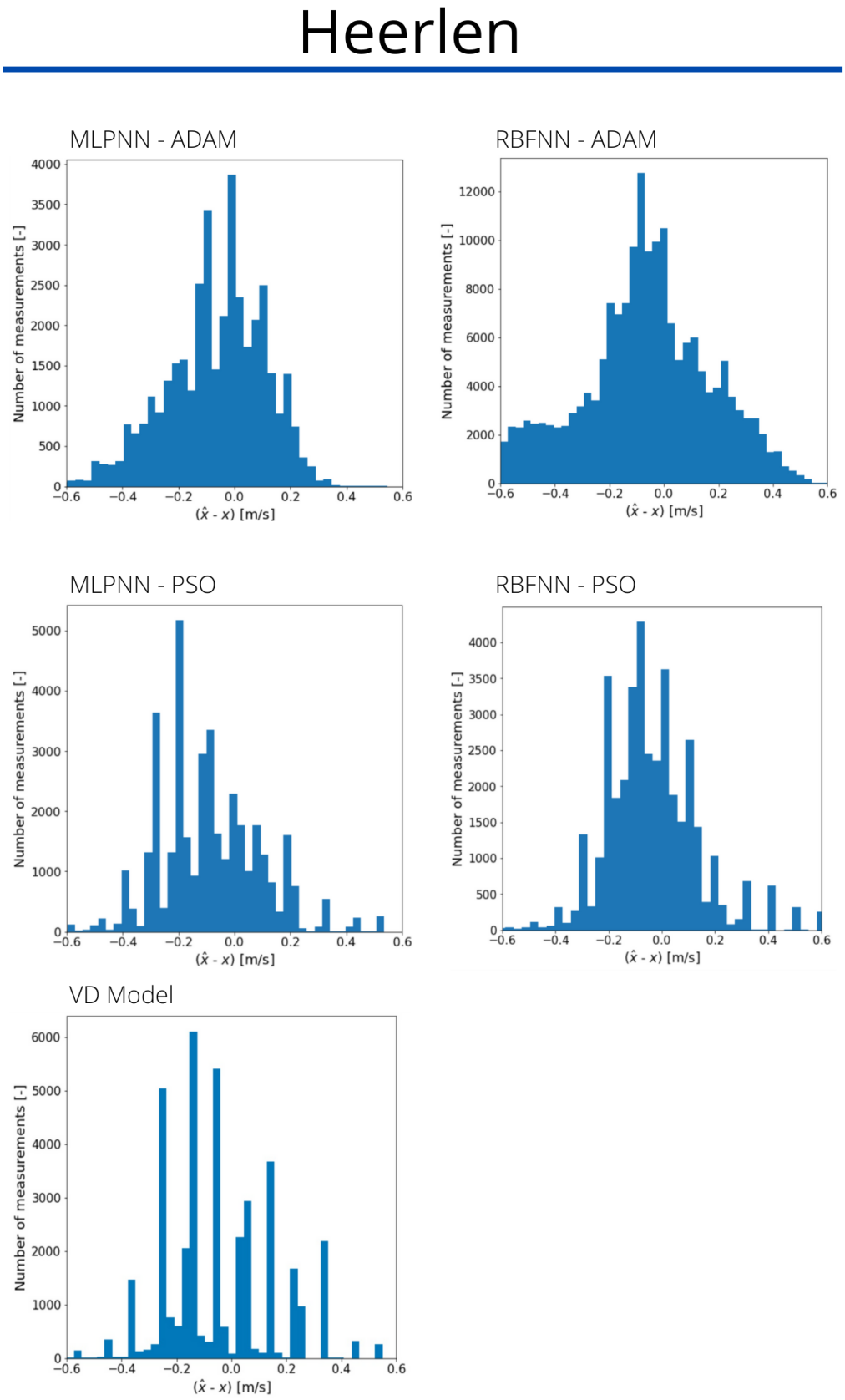


Figure B.4: Normal distributions of the prediction error for the Heerlen test set.

References

Dinoloket, data en informatie van de nederlandse ondergrond. URL `https://www.dinoloket.nl/`.

B. Adıgüzel Mercangöz. Applying particle swarm optimization: New solutions and cases for optimized portfolios. 2021.

G. Agrawal, J.-L. A. Chameau, and P. L. Bourdeau. Assessing the liquefaction susceptibility at a site based on information from penetration testing. In *Artificial neural networks for civil engineers: fundamentals and applications*, pages 185–214. ASCE, 1997.

D. W. Aha and R. L. Bankert. A comparative evaluation of sequential feature selection algorithms. In *Pre-proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, pages 1–7. PMLR, 1995.

B. Akkaya and N. Çolakoğlu. Comparison of multi-class classification algorithms on early diagnosis of heart diseases. 2019.

D. Anguita, L. Ghelardoni, A. Ghio, L. Oneto, and S. Ridella. The 'k'in k-fold cross validation. In *20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, pages 441–446. i6doc. com publ, 2012.

H. Anysz, A. Zbiciak, and N. Ibadov. The influence of input data standardization method on prediction accuracy of artificial neural networks. *Procedia Engineering*, 153:66–70, 2016.

D. J. Armaghani, E. T. Mohamad, M. S. Narayanasamy, N. Narita, and S. Yagiz. Development of hybrid intelligent models for predicting tbm penetration rate in hard rock condition. *Tunnelling and Underground Space Technology*, 63:29–43, 2017.

S. B. Ashrafi, M. Anemangely, M. Sabah, and M. J. Ameri. Application of hybrid artificial neural networks for predicting rate of penetration (rop): A case study from marun oil field. *Journal of petroleum science and engineering*, 175:604–623, 2019.

U. Backhausen and A. van der Stoel. Reader Geotechniek. Technical report, 2014.

M. Baziar and A. Ghorbani. Evaluation of lateral spreading using artificial neural networks. *Soil Dynamics and Earthquake Engineering*, 25(1):1–9, 2005.

M. Bielefeld. Allwave-vdp, 2020. URL `https://allnamics.com`.

A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Occam's razor. *Information processing letters*, 24(6):377–380, 1987.

S. Bock and M. Weiß. A proof of local convergence for the adam optimizer. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

A. Botchkarev. Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. *arXiv preprint arXiv:1809.03006*, 2018.

M. N. Boyer. Resistance to Technological Innovation: The History of the Pile Driver through the 18th Century. Technical Report 1, 1985.

E. Brenjkar and E. B. Delijani. Computational prediction of the drilling rate of penetration (rop): A comparison of various machine learning approaches and traditional models. *Journal of Petroleum Science and Engineering*, 210:110033, 2022.

E. Brenjkar, E. B. Delijani, and K. Karroubi. Prediction of penetration rate in drilling operations: a comparative study of three neural network forecast methods. *Journal of Petroleum Exploration and Production*, 11(2):805–818, 2021.

W. T. Chan, Y. K. Chow, and L. F. Liu. Neural Network: An Alternative to Pile Driving Formulas. Technical report, 1995.

G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

S. Chen, S. Billings, and P. Grant. Recursive hybrid algorithm for non-linear system identification using radial basis function networks. *International Journal of Control*, 55(5):1051–1070, 1992.

Y.-L. Chen, R. Azzam, and F.-B. Zhang. The displacement computation and construction pre-control of a foundation pit in shanghai utilizing fem and intelligent methods. *Geotechnical & Geological Engineering*, 24(6):1781–1801, 2006.

Z. Chen, X. Li, Z. Zhu, Z. Zhao, L. Wang, S. Jiang, and Y. Rong. The optimization of accuracy and efficiency for multistage precision grinding process with an improved particle swarm optimization algorithm. *International Journal of Advanced Robotic Systems*, 17(1):1729881419893508, 2020.

C. Cheng. Principal component analysis (pca) explained visually with zero math, 2022. URL `https://towardsdatascience.com/principal-component-analysis`.

F. Chollet et al. Keras, 2015. URL `https://github.com/fchollet/keras`.

J. C. Cleary and E. J. Steward. Analysis of ground vibrations induced by pile driving and a comparison of vibration prediction methods. *DFI Journal*, 10(3):125–134, 11 2016. ISSN 19375255. doi: 10.1080/19375247.2017.1288855.

M. Clerc. *Particle swarm optimization*, volume 93. John Wiley & Sons, 2010.

R. Cudmani. Static, dynamic and alternating penetration in non-cohesive soils. *Ph.D. Thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany*, 2001.

C. Darken and J. Moody. Towards faster stochastic gradient search. *Advances in neural information processing systems*, 4, 1991.

M. de Jong. Classifying human pilot skill level using deep artificial neural networks. 2021.

F. Deckner. *Vibration transfer process during vibratory sheet pile driving: from source to soil*. PhD thesis, KTH Royal Institute of Technology, 2017.

A. Défossez, L. Bottou, F. Bach, and N. Usunier. On the convergence of adam and adagrad. 2020.

N. Denies. *Dynamic behavior of vibrated dry sand: sphere penetration experiments and discrete element modeling of vibrofluidization*. PhD thesis, UCL-Université Catholique de Louvain, 2010.

G. Dierssen. A soil mechanics model to describe the vibrator in granular soils. *Thesis, University of Karlsruhe,Germany*, 1994.

R. V. Dorp, N. Moscoso, M. Bielefeld, A. Geotechnical, and G. Verbeek. Prediction and Monitoring of Installation of Offshore Foundation Monopiles for Windfarms. Technical report, 2019.

B. Douglas. Soil classificaion using electric cone penetrometer. In *Symp. on Cone Penetration Testing and Experience, Geotech. Engrg. Div.*, pages 209–227. ASCE, 1981.

J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

H. Eldin Ali and Y. M. Najjar. Neuronet-based approach for assessing liquefaction potential of soils. *Transportation Research Record*, 1633(1):3–8, 1998.

A. Elshorbagy, G. Corzo, S. Srinivasulu, and D. Solomatine. Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology-part 1: Concepts and methodology. *Hydrology and Earth System Sciences*, 14(10):1931–1941, 2010.

M. H. Esfe, S. Wongwises, A. Naderi, A. Asadi, M. R. Safaei, H. Rostamian, M. Dahari, and A. Karimipour. Thermal conductivity of cu/tio2–water/eg hybrid nanofluid: Experimental data and modeling using artificial neural network and correlation. *International communications in heat and mass transfer*, 66:100–104, 2015.

H. Faris, I. Aljarah, and S. Mirjalili. Evolving radial basis function networks using moth–flame optimizer. In *Handbook of neural computation*, pages 537–550. Elsevier, 2017.

A. H. Fath, F. Madanifar, and M. Abbasi. Implementation of multilayer perceptron (mlp) and radial basis function (rbf) neural networks to predict solution gas-oil ratio of crude oil systems. *Petroleum*, 6 (1):80–91, 2020.

L. V. Fausett. *Fundamentals of neural networks: architectures, algorithms and applications*. Pearson Education India, 2006.

X. Fu and L. Wang. Data dimensionality reduction with application to simplifying rbf network structure and improving classification performance. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 33(3):399–409, 2003.

J. García, R. Salmerón, C. García, and M. d. M. López Martín. Standardization of variables and collinearity diagnostic in ridge regression. *International Statistical Review*, 84(2):245–266, 2016.

J. H. Garrett. Where and why artificial neural networks are applicable in civil engineering. 1994.

A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

J. Ghaboussi. Potential applications of neuro-biological computational models in geotechnical engineering. *Numerical models in geotechnics*, pages 543–555, 1992.

J. Ghaboussi, J. Garrett Jr, and X. Wu. Knowledge-based modeling of material behavior with neural networks. *Journal of engineering mechanics*, 117(1):132–153, 1991.

S. Ghoreishi and E. Heidari. Extraction of epigallocatechin-3-gallate from green tea via supercritical fluid technology: Neural network modeling and response surface optimization. *The Journal of Supercritical Fluids*, 74:128–136, 2013.

A. T. Goh. Pile driving records reanalyzed using neural networks. *Journal of Geotechnical Engineering*, 122(6):492–495, 1996.

A. Gouda, S. Gomaa, A. Attia, R. Emara, S. Desouky, and A. El-hoshoudy. Development of an artificial neural network model for predicting the dew point pressure of retrograde gas condensate. *Journal of Petroleum Science and Engineering*, 208:109284, 2022.

D. Granato, J. S. Santos, G. B. Escher, B. L. Ferreira, and R. M. Maggio. Use of principal component analysis (pca) and hierarchical cluster analysis (hca) for multivariate association between bioactive compounds and functional properties in foods: A critical perspective. *Trends in Food Science & Technology*, 72:83–90, 2018.

S. H. Haji and A. M. Abdulazeez. Comparison of optimization techniques based on gradient descent algorithm: A review. *PalArch's Journal of Archaeology of Egypt/Egyptology*, 18(4):2715–2743, 2021.

A. M. Hanna, D. Ural, and G. Saygili. Neural network model for liquefaction potential in soil deposits using turkey and taiwan earthquake data. *Soil Dynamics and Earthquake Engineering*, 27(6):521–540, 2007.

J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108, 1979.

R. Hassan, B. Cohanim, O. De Weck, and G. Venter. A comparison of particle swarm optimization and the genetic algorithm. In *46th structures, structural dynamics and materials conference*, page 1897, 2005.

A. Hemmati-Sarapardeh, M.-H. Ghazanfari, S. Ayatollahi, and M. Masihi. Accurate determination of the co2-crude oil minimum miscibility pressure of pure and impure co2 streams: a robust modelling approach. *The Canadian Journal of Chemical Engineering*, 94(2):253–261, 2016.

A. Holeyman and V. Whenham. Vibrodriving Prediction Models vs. Experimental Results. Technical report, 2010.

A. A. Holeyman, J.-F. v. J.-F. Berghe, and N. N. Charue. *Soil behavior under Vibratory Driving*. A.A. Balkema Publishers, 2002. ISBN 9058095215.

A. E. Holeyman. Keynote lecture: Vibratory driving analysis. *Proceedings of Application of Stress-Wave Theory to Piles, Brazil*, pages 479–494, 2000.

D. Howe, D. A. Stone, C. Legrand, and D. J. Van Rompaey Mentcm. High Performance Vibratory Pile-Drivers Based on Novel Systems and Improved Understanding of Soil Dynamics. Technical report, 1995.

A. Ismail. Prediction of resistance to pile driving using evolutionary neural network. *Multiscale and Multidisciplinary Modeling, Experiments and Design*, 2(1):49–61, 2019.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

A. A. Javadi, M. Rezania, and M. M. Nezhad. Evaluation of liquefaction induced lateral displacements using genetic programming. *Computers and Geotechnics*, 33(4-5):222–233, 2006.

Q. Jiang, L. Zhu, C. Shu, and V. Sekar. An efficient multilayer rbf neural network and its application to regression problems. *Neural Computing and Applications*, 34(6):4133–4150, 2022.

F. Jonker, J. de Gijt, and A. van den Thoorn. Handboek Hei-en trilbaarheid. Technical report, 2017.

G. Jonker. Vibratory Pile Driving Hammers for Pile Installations and Soil Improvement Projects. Technical report, 1987.

G. Jonker and S. Hartog. Vibratory Pile Driving Predictions. Technical report, 1988.

G. Josselin De Jong. Wat gebeurt er in de grond tijdens het heien" (what happens in the soil during pile driving). 1956.

A. Karkevandi-Talkhooncheh, S. Hajirezaie, A. Hemmati-Sarapardeh, M. M. Husein, K. Karan, and M. Sharifi. Application of adaptive neuro fuzzy interface system optimized with evolutionary algorithms for modeling co2-crude oil minimum miscibility pressure. *Fuel*, 205:34–45, 2017.

J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE, 1995.

Y. Kim and B. Kim. Use of artificial neural networks in the prediction of liquefaction resistance of sands. *Journal of Geotechnical and Geoenvironmental Engineering*, 132(11):1502–1504, 2006.

W. Koers. Vibro-driveability predictions, April 2022.

P. U. Kurup and N. K. Dudani. Neural networks for profiling stress history of clays from pcpt data. *Journal of Geotechnical and Geoenvironmental Engineering*, 128(7):569–579, 2002.

L. M. Le, H.-B. Ly, B. T. Pham, V. M. Le, T. A. Pham, D.-H. Nguyen, X.-T. Tran, and T.-T. Le. Hybrid artificial intelligence approaches for predicting buckling damage of steel columns under axial compression. *Materials*, 12(10):1670, 2019.

Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012.

S. Lee, S. R. Lee, and Y. Kim. An approach to estimate unsaturated shear strength using artificial neural network and hyperbolic formulation. *Computers and Geotechnics*, 30(6):489–503, 2003.

Y. Lee, S.-H. Oh, and M. W. Kim. The effect of initial weights on premature saturation in back-propagation learning. In *IJCNN-91-Seattle international joint conference on neural networks*, volume 1, pages 765–770. IEEE, 1991.

K. Louhichi, F. Jacquet, and J. P. Butault. Estimating input allocation from heterogeneous data sources: A comparison of alternative estimation approaches. *Agricultural Economics Review*, 13(389-2016-23472):83–102, 2012.

B. G. Marcot and A. M. Hanea. What is an optimal value of k in k-fold cross-validation in discrete bayesian network analysis? *Computational Statistics*, 36(3):2009–2031, 2021.

D. W. Marquardt. Comment: You should standardize the predictor variables in your regression models. *Journal of the American Statistical Association*, 75(369):87–91, 1980.

K. R. Massarsch. Effects of vibratory compaction. In *TransVib 2002–International Conference on Vibratory Pile Driving and Deep Soil Compaction. Louvain-la-Neuve. Keynote Lecture*, pages 33–42, 2002.

K. R. Massarsch. Recent developments in vibratory driving and soil compaction. Technical report, 2017.

C. Mccormick. Radial basis function network tutorial, 2013. URL `https://chrisjmccormick.wordpress.com`.

P. Middendorp. Vibratory pile driving, Dec 2021.

P. Middendorp and G. E. H. Verbeek. 30 years of experience with the wave equation solution based on the method of characteristics. Technical report, 2006.

A. Moghadassi, F. Parvizian, and S. Hosseini. A new approach based on artificial neural networks for prediction of high pressure vapor-liquid equilibrium. *Australian Journal of Basic and Applied Sciences*, 3(3):1851–1862, 2009.

A. Moghadassi, S. M. Hosseini, F. Parvizian, I. Al-Hajri, and M. Talebbeigi. Predicting the supercritical carbon dioxide extraction of oregano bract essential oil. *Songklanakarin Journal of Science & Technology*, 33(5), 2011.

E. Momeni, R. Nazir, D. J. Armaghani, and H. Maizir. Application of artificial neural network for predicting shaft and tip resistances of concrete piles. *Earth Sciences Research Journal*, 19(1):85–93, 2015.

G. A. Montazer, D. Giveki, M. Karami, and H. Rastegar. Radial basis function neural networks: a review. *Computers Reviews Journal*, 1(1):52–74, 2018.

S. Moriyasu, S.-i. Kobayashi, and T. Matsumoto. Experimental study on friction fatigue of vibratory driven piles by in situ model tests. *Soils and foundations*, 58(4):853–865, 2018.

R. Moulai-Khatir, M. W. O'Neill, and C. Vipulanandan. Program vpda wave equation analysis for vibratory driving of piles. *Report to the US Army Corps of Engineerings Waterways Experiments Station., Dept of Civil and Environmental Engineering, UHCE*, pages 94–1, 1994.

I. T. Nabney. Efficient training of rbf networks for classification. *International Journal of Neural Systems*, 14(03):201–208, 2004.

M. A. Nielsen. *Neural networks and deep learning*, volume 25. Determination press San Francisco, CA, 2015.

R. Nisbet, E. John, and M. Gary. Basic algorithms for data mining—a brief overview. handbook of statistical analysis and data mining applications, 2009.

M. W. O'Neill and C. Vipulanandan. Program VPDA Wave Equation Analysis for Vibratory Driving of Piles. *Report to The US.A. Army Corps of Engineers Waterways Experiments Station. Dept. of Civil and Environmental Engineering. Houston, Texas*, 187, 1989.

J. Patterson and A. Gibson. *Deep learning: A practitioner's approach.* " O'Reilly Media, Inc.", 2017.

D. Penumadu and J. Chameau. Geomaterial modeling using artificial neural networks. *Artificial neural networks for civil engineers: fundamentals and applications, N. Kartam, I. Flood, and J. Garrett, eds., ASCE, New York*, pages 160–184, 1997.

R. Poli, J. Kennedy, and T. Blackwell. Particle swarm optimization. *Swarm intelligence*, 1(1):33–57, 2007.

N. Preobrajhenskaja. The influence of vibration factors on the penetration of piles and sheet piles. In *Pap. of Meeting of Inst. of Foundations*, 1956.

C. Qi, A. Fourie, G. Ma, X. Tang, and X. Du. Comparative study of hybrid artificial intelligence approaches for predicting hangingwall stability. *Journal of Computing in Civil Engineering*, 32(2): 04017086, 2018.

Z. Qin, L. Chen, C. Song, and L. Sun. Field Tests to Investigate the Penetration Rate of Piles Driven by Vibratory Installation. *Shock and Vibration*, 2017, 2017. ISSN 10709622. doi: 10.1155/2017/7236956.

N. Ramsey. Some issues related to applications of the cpt. In *2nd International Symposium on Cone Penetration Testing, CPT*, volume 10, 2010.

P. M. Rao. *Effect of pile geometry and soil saturation in the behavior of nondisplacement piles installed by vibration.* PhD thesis, MSc thesis presented to the Faculty of the Dep. of Civil and Environmental Engineering, University of Houston, Texas, 1993.

F. Rausche. Modeling of vibratory pile driving. 2002.

P. K. Robertson. Soil behaviour type from the cpt: an update. In *2nd International symposium on cone penetration testing*, volume 2, pages 575–583. Cone Penetration Testing Organizing Committee, 2010.

P. K. Robertson, R. G. Campanella, D. Gillespie, and J. Greig. Use of piezometer cone data. In *Use of in situ tests in geotechnical engineering*, pages 1263–1280. ASCE, 1986.

A. Rodger and G. S. LittleJohn. A study of vibratory driving in granular soils. Technical Report 3, 1980.

M. Sabah, M. Talebkeikhah, D. A. Wood, R. Khosravanian, M. Anemangely, and A. Younesi. A machine learning approach to predict drilling rate using petrophysical and mud logging data. *Earth Science Informatics*, 12(3):319–339, 2019.

M. Shahin and M. Jaksa. Modelling the pullout capacity of marquee ground anchors using neurofuzzy technique 1. 2005.

M. Shahin, M. Jaksa, and H. Maier. Neurofuzzy networks applied to settlement of shallow foundations on granular soils. In *Proceedings of the 9th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP9, San Francisco, Millpress, Rotterdam*, pages 1379–1383, 2003.

M. Shahin, M. Jaksa, and H. Maier. Application of neural networks in foundation engineering. In *Theme paper to the International e-Conference on Modern Trends in Foundation Engineering: Geotechnical Challenges and Solutions, Theme*, number 5, 2004.

M. A. Shahin. State-of-the-art review of some artificial intelligence applications in pile foundations. *Geoscience Frontiers*, 7(1):33–44, 1 2016. ISSN 16749871. doi: 10.1016/j.gsf.2014.10.002.

M. A. Shahin, M. B. Jaksa, and H. R. Maier. Artificial neural network applications in geotechnical engineering. *Australian geomechanics*, 36(1):49–62, 2001.

M. A. Shahin, M. B. Jaksa, and H. R. Maier. Artificial neural network based settlement prediction formula for shallow foundations on granular soils. *Australian Geomechanics: Journal and News of the Australian Geomechanics Society*, 37(4):45–52, 2002.

E. Smith. Pile-driving analysis by the wave equation. *Journal of the soil mechanics and foundations division*, 86(4):35–61, 1960.

M. Spüler, A. Sarasola-Sanz, N. Birbaumer, W. Rosenstiel, and A. Ramos-Murguialday. Comparing metrics to evaluate performance of regression methods for decoding of neural signals. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1083–1086. IEEE, 2015.

T. Tieleman and G. Hinton. Neural networks for machine learning. *Coursera (Lecture 65-RMSprop)*, 138, 2012.

K. Viking. Vibro-driveability - A field study of vibratory driven sheet piles in non-cohesive soils. 2002. ISSN 1650-9501.

K. Viking. The vibratory pile installation technique. 2006.

G. Voitus van Hamme, J. Jansz, H. Bomer, and D. Arentsen. Hydroblok and improvedpileddriving analysis. 1974.

D. Wang, D. Tan, and L. Liu. Particle swarm optimization algorithm: an overview. *Soft computing*, 22 (2):387–408, 2018.

H. Wang. *Experimental study and finite element analysis of driveability and static behavior of various piles installed by vibratory driving*. PhD thesis, Texas: Faculty of the Department of Civil and Environmental Engineering, 1994.

X. Wang, Z. Tang, H. Tamura, M. Ishii, and W. D. Sun. An improved backpropagation algorithm to avoid the local minima problem. *Neurocomputing*, 56:455–460, 2004.

D. Warrington. Vibratory and impact-vibration pile driving equipment. *Pile Buck, Second October issue 1992*, 1992.

E. Westerberg. Soil resistance during vibratory pile driving. In *Int. Symp. on Cone Penetration Testing, CPT'95*, volume 3. Swedish Geotech. Society, 1995.

V. Whenham. Power Transfer and Vibrator-Pile-Soil Interactions within the framework of vibratory pile driving. Technical report, 2011.

R. Whitlow. *Basic soil mechanics*. 1990.

A. K. H. Wu, R. L. Kuhlemeyer, and C. W. S. To. Validity of smith model in pile driving analysis. Technical report, 1989.

Y. Yang and M. Rosenbaum. The artificial neural network as a tool for assessing geotechnical properties. *Geotechnical & Geological Engineering*, 20(2):149–168, 2002.

T. L. Youd. *The engineering properties of cohesionless materials during vibration.* Iowa State University, 1967.

H. Yu, T. Xie, S. Paszczynski, and B. M. Wilamowski. Advantages of radial basis function networks for dynamic system design. *IEEE Transactions on Industrial Electronics*, 58(12):5438–5450, 2011.