Design optimization for flood defenses using machine learning

by

Louise Guichelaar

to obtain the degree of Master of Science
in Civil Engineering at Delft University of Technology,
to be defended publicly on 30-08-2024.

Student number:	4929055	
Thesis committee:	Dr. Ir. R.B.J. Brinkgreve,	TU Delft, committee chair
	Dr. G. Rongier,	TU Delft
	Ir. R.A. van der Eijk,	TU Delft
	Dr. Ir. J.P. Aguilar-López,	TU Delft
	Ir. A.K. de Jong,	Witteveen+Bos

Preface

Fighting against water is something that the Dutch inhabitants have been doing for generations. The incident from 1953 demonstrated the importance of flood defenses and still the Dutch people are aware of what dikes mean to us. In 2021, a famous Dutch singer, Davina Michelle, gave a beautiful performance during the Eurovision Song Contest with her hit 'Sweet water' which highlights the power of water and the Deltaworks that protects the Netherlands from floods. Water is an integral part of the Netherlands and we have to learn to live with water.

During primary school, I was already fascinated by the incident in 1953. How could our country fill up with water in such a short period of time? Are the Dutch dikes sturdy enough and should I still be concerned about any floods that may occur now?

When I went to high school, I had geography lessons for the first 3 years. During those years, I was amazed how strong the water is. We are not the only ones who shape the land, but water also plays a major role in this. Think for example of meandering rivers or strengthening dikes to resist the power of water.

In the last year of high school I had to make my education's choice. Since I like to take on complex technical challenges and enjoy performing calculations, I have opted for the bachelor and master civil engineering. During my bachelor, I chose the minor 'Delta Expert Water for the Future'. In the year of my minor, the Netherlands became increasingly familiar with rip currents. I had to make a presentation and I found it particularly interesting to make a presentation about this topic, because I wanted to know how this natural phenomenon arose. When I watched the news, it seemed that rip currents were only a problem in the Netherlands, but it is a natural phenomenon over the whole world. Rip currents are very strong and shape the sea bottom, similar to the meandering phenomenon.

Another assignment during this minor was searching for special flood defenses. I was impressed by the storm surge barrier 'Ramspol'. The Ramspol is an inflatable rubber dam and is blown up at high tide. It is fascinating to see how this storm surge barrier works.

The ethics assignments I wrote, were mainly about dike breaches. The incident in 1953 is still in the back of my head and I wanted to understand how this disaster could have arisen. What happened in 1953 was enormous, but still dike breaches occur and I wanted to know why these problems still occur. Are the dikes in the Netherlands strong enough and are they able to cope with the climate change we are undergoing? When a dike breach takes place, we learn a lot from it. It helps us to understand dikes better and better. However, given the climate change that is taking place, we must ensure that such dike breaches do not take place (too much), as the sea level rises and a dike breach at high tide can cause a lot of damage.

When I heard that many embankments need to be revised for railway during a presentation of the course 'Site Investigation and Monitoring', I was wondering what about the Dutch dikes. In September last year I organized a company dinner with our Ondergrondse board. I visited the Witteveen+Bos table and they showed me some papers with graduation subjects. When I saw the paper with a surrogate model that assesses the safety of dikes and integrated model, I was sold immediately. If such an integral model works, what a big step is this in the right direction and in this way dikes can be easily tested if they have to be optimized.

In conclusion, I look forward to working on this graduation research!

Louise Guichelaar

Deventer, August 2024

Acknowledgement

This research is the final part to obtain my master's degree in civil engineering at TU Delft, specializing in geotechical engineering. I am very grateful that Witteveen+Bos gave me the opportunity to do my research with them. I am thankfull that they believed in me and gave me the chance to develop myself in different ways.

This research could not be conducted without the help of countless people. Starting with the members of my graduation committee. My sincere gratitude goes to Koen de Jong at Witteveen+Bos for the weekly meetings, for welcoming me into group, introducing me to colleagues who shared their experiences with me, explaining the thesis topic, and many more things that made my time at Witteveen+Bos very enjoyable and educational. Secondly, I would like to thank Ronald Brinkgreve, the chair of my thesis committee and associate professor at the geo-engineering department, for his time and guidance to properly conduct this research. When you are doing your research, sometimes you end up in a tunnel vision and you only see certain aspects in one way. Ronald Brinkgreve throw a different light on these aspects, which made me look at this differently and think differently. This part helped me a lot in doing this research. I am grateful for the help of Juan Aguilar-López and Guillaume Rongier who have both given me advice on and information about machine learning (models). I would also like to thank Ryan van der Eijk for sharing his knowledge about dikes with me. The progress meetings were very valuable to me with useful feedback during the meetings and on my intermediate reports. I am very thankful for their help and guidance.

Besides my graduation committee, I would like to express my appreciation to my colleagues at Witteveen+Bos who shared their work with me, gave me idea's for my thesis, their enthusiasm, and made the graduation period very enjoyable for me. I had a very nice time at Witteveen+Bos and I am grateful to everyone for that.

Very unusual to report this, but I would like to thank my English high school teacher. Because I was very bad at English in high school, she did not think I could ever go to college or finish the pre-university education. I wanted to prove to her that I can do it and this research is the proof of it.

Last, but certainly not least, I would like to thank my family, friends and acquaintances for their listening ear and support. They believed in me and without their support I wouldn't have gotten anywhere.

I look back on many educational years at university with academic and personal development and opportunities. Last year, I was part of the Ondergrondse board and learned and laughed a lot during this year. I wouldn't miss it for the world. I am ready to learn more from the geotechnical world and I am curious what the future has to offer me!

Abstract

Currently, many dikes need to be reinforced to be future-proof. Therefore many dike designs have to be realized for which one of the main failure mechanisms is macro-stability of the inner slope of dikes. Nowadays, designing dikes requires to consider not just different failure mechanisms, but also how to best integrate a flood defense in a given area. To assess these different design aspects, different software, methods, and models should be used and combined to get an integrated dike design. Each engineering aspect has its own step-by-step plan to get the best outcome and this result can affect the outcome of another engineering aspect. This makes it very hard to create many integral dike designs in a short amount of time.

To address these issues, the following research question is answered: How can a surrogate model for inner slope stability of a flood defense be created for use in combination with other (machine learning) models to generate conceptual flood defenses, making it possible to optimize the design using an interdisciplinary Multi Criteria Analysis (MCA)?

To get optimal dike designs this thesis is divided into two parts. The first part is about deriving a surrogate model for macro-stability of dikes and the second part about combining the macro-stability model with the equations/models for the other engineering aspects related to dike design.

In the first part, simple, basic dikes geometries are considered first. The training data for the model is generated based on conventional slope stability analysis: input parameters: geometry (and soil) parameters; output parameters: Factor of Safety (FoS), macro-stability failure mechanism (centroid, radius, depth). The data is sampled with Random Sampling (RS) and Latin Hypercube Sampling (LHS) creating different dike geometries. Several machine learning (ML) methods are trained on the datasets derived from the two sampling methods. The Support Vector Regression (SVR) model trained on the sampled dataset with RS has the best model's performance according to the assessment of various regression metrics. After deriving the surrogate model for the simple dike geometries, a surrogate model for complex dike geometries can be developed. The simple dike geometries are extended by an inner berm and a ditch next to the dike, adding three geometric input parameters to the training data. This time, the data is sampled with LHS since this sampling method resulted in higher model performances for the different ML models for the simple dike geometries. After training the same ML models as for the simple dike geometries, the model's performance for the surrogate model SVR was the best for the complex dike geometries according to the considered various regression metrics. To be able to predict the slip circle, output parameters related to the macro-stability failure mechanism were added to the training dataset. The training dataset was changed many times in order to predict the slip circle accurately, but the performance of the surrogate model remained not as accurate as the performance of the Simple and Complex surrogate model which predicts the value for the FoS.

In the second part, the derived surrogate model for macro-stability of dikes is combined with equations/models for the engineering aspects of ecology, costs, environmental cost indicator (ECI), and wave overtopping. The five engineering aspects are combined in a multi criteria analysis (MCA), in which the surrogate model for macro-stability and the equation for wave overtopping act as constraints and the models for ecology, ECI, and costs act as criteria. An objective function, which is the total score of a certain dike geometry in the MCA, is formulated as the sum of the weights for the criteria multiplied by the score of the corresponding criteria for the considered dike geometry. The objective function has as input the dike geometry and as output the scores for the criteria ecology, ECI, costs of the dike, and the total score for the dike geometry. In order to find the best score in the MCA and therefore the optimal dike geometry, an optimization function is defined. This function takes as input the objective function and has as output the optimized dike geometry and the lowest value for the objective function. The optimization function ensures that thousands of dike geometries should not be compared by hand in a MCA, but the optimization function compares dike geometries, which saves time.

Contents

List of Figures

List of Tables

Nomenclature

γ_{sat}	Saturated unit weight $[kN/m^3]$
γ_{unsat}	Unsaturated unit weight $[kN/m^3]$
ψ	Dilatancy angle [°]
σ'_{n}	Normal effective stress [kN/m^2]
σ'_{v}	Vertical effective stress [kN/m^2]
σ'_{y}	Maximum experienced vertical yield stress [kN/m^2]
τ	Maximum available shear strength [kN/m^2]
φ	Friction angle [°]
c	Cohesion [kN/m^2]
m	Strength increase exponent as a result of over-consolidation [-]
OCR	Over Consolidation Ratio [-]
POP	Pre Overburden Pressure $[kN/m^2]$
S	Ratio between the undrained shear strength s_u and the vertical effective stress [-]

Acronyms

ANN Artificial Neural Network

CV cross-validation

ECI Environmental Cost Indicator

XGB eXtreme Gradient Boosting

FNN Feedforward Neural Network

FoS Factor of Safety

GB Gradient Boosting

GS Grid Sampling

HWBP Hoogwater beschermingsprogramma

HWL Hoogwaterbescherming and Landinrichting

LEM Limit Equilibrium Method

LHS Latin Hypercube Sampling

MAE Mean Absolute Error

MCA Multi Criteria Analysis

ML Machine Learning

MSE Mean Squared Error

post python ondergrond schematisatie toolkit

RFR Random Forest Regressor

RS Random Sampling

R^2 Coefficient of Determination

SSK Standaardsystematiek voor Kostenramingen

SVC Support Vector Classification

SVM Support Vector Machine

SVR Support Vector Regression

WBI 2017 Legal Assessment Instrument 2017

W+B Witteveen+Bos

Chapter 1 Introduction

1.1 Background information

The Netherlands lies for a large part below sea level. 60% of the Netherlands will be flooded without the existing dikes and dunes [18]. A great natural disaster, the Watersnoodramp in 1953, showed the Netherlands how important dikes are. The disaster revealed the devastating effects of flooding, such as loss of life and economic damage. Before this disaster, people just believed that the dikes were stable and no money was made available to maintain the flood defenses. However, the dikes were too low and weak and could not withstand high water and wind loads during the storm surge, which led to many dike breaches and consequently the Watersnoodramp. First, the dikes failed due to wave overtopping and overflow of water and after some time the dikes became unstable, because they became saturated due to long-term high water and rain. The Watersnoodramp was a turning point in the Dutch history regarding water safety. After the Watersnoodramp, stricter requirements were imposed on primary flood defenses and the Delta Commission was founded [31] to prevent such disaster from happening again. The Delta Commission made the Delta plan and as a result of this Delta plan, many flood defenses were realized along the coast on the North Sea.

In 2008, the latest Delta Commission was founded. Back then, the water threat was not as high as at the time of the first Delta Commission, but the Netherlands should be concerned about the future. To be well prepared for the expected climate change, the flood defenses should be reinforced and the spatial environment should be adjusted [12]. The advice from the Delta Commission to the Dutch government resulted in a new Delta plan: 'The Netherlands should be climate-proof and water-robust by 2050' [67]. As a result of this Delta plan, Rijkswaterstaat and the 21 water boards in the Netherlands came together and made the High Water Protection Program in 2014, also known as the Hoogwater beschermingsprogramma (HWBP) in Dutch. The HWBP is the biggest flood defense operation ever since the Delta plan after the Waternoodsramp and the amount of dike that needs to be reinforced is comparable to the amount after the disaster. Initially, Rijkswaterstaat and the water boards had to reinforce 1500 km of dikes. However, 9 years later, in 2023, they discover that they have to reinforce a total amount of 2000 km of dikes [45]. This increase of the length of dike that needs to be reinforced is caused by the new standards in the field of safety against water. The new standards are derived from Legal Assessment Instrument 2017 (WBI 2017). It is the newest legal assessment method for primary flood defenses in the Netherlands. Considering the duration of the (re)construction of dikes and the Delta Works after the Watersnoodramp, many different calculations on dikes, based on the principles of WBI2017, must be performed in a short time to achieve the goal by 2050.

1.2 Motivation

Dikes can fail in various ways, as seen during the Watersnoodramp. When designing a flood defense, all potential failure mechanisms need to be considered. On top of that, the current reinforcement projects in the Netherlands require more demands regarding the integration of a flood defense in a certain area compared to the past. Those demands include ecology, space limitations, and the influence on surroundings. On top of that, climate change calls for more robust dike designs. As a result of climate change, the Netherlands is increasingly suffering from extreme weather and sea-level rise. Over the years, periods of downpours and heavy storms or long dry and hot periods occurred in the Netherlands [44]. Due to both wet and dry periods, the water level varies enormously and this has a significant impact on the strength of the soil. If the strength reduction is quite high, the dike can not resist the different magnitude of water loads, and parts of the dike can slide [61]. This failure mechanism is called 'Macro-instability'. Macro-instability is an important failure mechanism, which is often cost-determining in dike reinforcement projects. That is why the water board searches for indications for the possible occurrence of macro-instability of dikes during it's dike inspections. As a result of the HWBP and the dike inspections, many dike designs and dike calculations should be performed.

Due to climate change more and stricter requirements are needed to comply with the safety standards for dike stability combined with an increasing complexity of test methods, large amounts of stability calculations on dikes have to be carried out to make sure that all influences are correctly taken into account. This is only for the stability of dikes, but dikes have much more design aspects. To cover all design aspects, many dikes design should be created and tested. This is a time-consuming process that makes optimization of integrated designs complex and costly. Often it turns out to be difficult to deliver good integrated dike designs and this should be improved. That is where this research comes into play. By combining different (machine learning) models for different design aspects, the process of optimizing integrated dike designs becomes faster and less costly compared to the current situation.

1.2.1 Problem statement and Relevance

Over time, many methods, models, and software have been developed to make optimal designs for flood defenses, given a set of criteria. Software programs like PLAXIS and Deltares' software product for macro-stability 'D-stability' may be used to estimate the Factor of Safety (FoS) for the design of a dike. The FoS is a measure of how stable the slopes of a dike are. These existing programs evaluate just design aspects related to processes that can lead to one of the failure mechanisms of a dike, such as instability of a dike and dike settlement, which leads to overflow of water. However, design aspects such as cost and Environmental Cost Indicator (ECI) are not directly taken into account. Nowadays, to assess these different design aspects for a dike design, different design software, methods and models should be used and combined to get an integrated design. Each different design aspects has its own tool to get the best outcome and this result can affect the outcome of the tools for the other engineering aspects, which requires the other engineering aspects to be re-optimized. This is a time-consuming process to run and adjust all these different types of software, methods and models one by one for each design aspect to get to a fully integrated design.

That is why nowadays, in an early design stage, dike calculations are done with some guideline values, a rough approach for a dike design. Later, when more information is gained about the

dike project, it often appears that incorrect or insufficiently detailed assumptions have been made in an early design stage and that previous conclusions need to be revised. Being able to perform more calculations for different design aspects would help explore a wider range of assumptions at an early stage. However, this takes too much time. A possible solution for the stability calculations in this phase is to replace the calculations with a surrogate model that has been trained on many dike stability calculations to reach a more accurate first estimate compared to the current guideline values and integrate this model with the equations/models for the cost of the dike, ECI, ecology, and wave overtopping, which may be improve design process.

The surrogate model will make sure that the integral part of dike designs will be visualized in an early stage. This model can use guide values, but there is direct visibility on the sensitivity to certain parameters, and how this influences other design aspects. This prevents making wrong conclusions in an early design stage.

1.3 Research objective

This research aims to find the best way to construct a surrogate model for macro-stability of inner slopes of dikes and how it can be combined with other models in a Multi Criteria Analysis (MCA) to obtain interdisciplinary dike designs. Looking at the High Water Protection Program, many dike sections need to be assessed to be future proof and this surrogate model helps the program by quickly generating stable dikes. The surrogate model must run faster than the current stability calculations for dikes, so that many flood defense designs with as many interactions as possible can be generated and compared in the early stage of a project. Many dike designs should be generated to perform a MCA, which is cross-disciplinary due to the considered engineering aspects. This research is in collaboration with Witteveen+Bos (W+B) and TU Delft.

To achieve this objective, the following research question will be answered:

How can a surrogate model for inner slope stability of a flood defense be created for use in combination with other (machine learning) models to generate conceptual flood defenses, making it possible to optimize the design using an interdisciplinary MCA?

To support answering the main research question, a number of sub-questions are posed to contribute to the answering of the main research question.

SQ1: Which input and output variables of a stability model does the surrogate model need to train the machine learning on certain combinations of input and output variables for accurate prediction of the model outcome for macro-stability of dikes in the Netherlands?

SQ2: Which sampling method is needed to have enough information about dike stability captured in the dataset to train, validate, and test the surrogate model?

SQ3: Which type of ML algorithms perform best for emulating macro-stability models best?

SQ4: What are the equations / models for the other relevant aspects for dike design and

what do these equations / models look like?

SQ5: How can the macro-stability surrogate model be combined with the equations/models for the other relevant engineering aspects related to dike design to get to an optimal design?

1.4 Methodology

The method used to achieve an optimal dike design is divided into two parts. The first part is about deriving a surrogate model for macro-stability of dikes. In this part, subquestions SQ1 to SQ3 will be answered. To answer each subquestion, the following method is used:

- SQ1: First, simple dike geometry will be considered with constant soil properties, also named as the 'simple input parameters'. The machine learning algorithms will be trained on these simple input parameters with the corresponding output from D-stability. After training the ML model, it can be observed which simple input parameter influences the FoS the most and least. Based on the results of this analysis and further data investigation, changes will be made to the dataset, ML models, or training procedures in order to improve the ML models' performance. After having an accurate ML model for the simple dike geometries, other parameters related to the dike geometry will be added to increase the complexity of the dike geometry such that the dike geometries become more realistic. The machine learning algorithms will be trained on the input parameters from the complex dike geometry with the corresponding output from D-stability and again by performing sensitivity analysis and data investigation should lead to an increase in ML models' performance.
- SQ2: Two sampling methods will be considered in this research. To decide which sampling method is better, both methods are used to get a dataset containing simple input parameters with corresponding output for training the machine learning algorithm. Both datasets, one dataset obtained from one sampling method and the other dataset from the other sampling method, will be used to train the different machine learning algorithms that are considered in this research. Based on the three different validation metrics that are considered, the sampling method that performs the best and has enough information about dike stability can be chosen. The best sampling method will be applied to create the dataset containing the more complex dike geometries.
- SQ3: In this research, different type of machine learning models will be considered. First, four different machine learning models will be trained on the dataset containing the simple input parameters with their corresponding output. After hyper parameter tuning for each machine learning model and training the final ML models, the validation metrics will be applied to see which machine learning model predicts macro-stability the best. When changing the input parameter sets from simple dike geometries to complex dike geometries, a similar procedure as for choosing the ML model for the simple input parameters will be applied in order to get the best surrogate model for macro-stability of the inner slope of a dike. However, the best sampling method will be applied to get the new dataset, containing more input parameters, instead of using the two sampling methods mentioned in the SQ2 section.

The second part is about combining the macro-stability model with the equations/models for the other engineering aspects related to dike design. In this part, sub questions SQ4 and SQ5

will be answered. To answer each sub question, the following method is used:

- SQ4: For some of the engineering aspects, equations and models already exist to approximate the engineering aspect and this equation or model can be used for the integral dike design. However, for some, the existing equations or models should be simplified. The simplification is based on the dike geometries that are considered for the surrogate model and on generalization of the project-specific situations since the dike designs that will be generated in this research do not have project-specific aspects. The equations and models for the other engineering aspects are derived from available reports and discussions with experts at Witteveen+Bos.
- SQ5: The macro-stability model will be combined with the equations/models for other engineering aspects via a multi criteria analysis to get to an optimal integrated dike design. The dike design will be optimized based on the other relevant engineering aspects that are considered in this research. The flood defenses are not optimized based on macro-stability and wave overtopping since these two failure mechanisms acts as constraints in the MCA. In the MCA, changing the weights for the ecology, costs, and ECI models will result in different optimized dike geometries and the values for the weights are in line with the wishes of the water board or Rijkswaterstaat, who commissions the reinforcement of a dike.

1.5 Research scope

In this research, the particular focus is on primary flood defenses in the Netherlands.

A surrogate model is a model that is an approximation of a computationally expensive model. In this research, the surrogate model is a ML model since one ML model is considered to assess the dike stability. Surrogate model and ML model will be used interchangeably in this report.

As mentioned in section 1.2, many engineering aspects interact with the design of a dike. It is not possible to incorporate all factors since there exist many. The relevant engineering aspects, besides macro-stability of a dike, that will be considered in this research are: Ecology, Wave overtopping, Cost of the dike, and Environmental Cost Indicator, also known as MKI.

1.6 Thesis outline

This chapter has introduced the problem and the backgrounds and approach of this research topic. Following the introduction in chapter 1, the contents of the thesis are divided into the following sections with corresponding chapters:

I Macro-stability model

- Chapter 2 (Literature review): starts with explaining the different failure mechanisms related to dikes and highlights the ones that are applicable in this research. After this explanation, the shear strength models will be elaborated and the ones that are used in this research will be explained in detail. In D-stability, three types of Limit Equilibrium Method (LEM) exist and all three will be clarified. After explaining the models used in D-Stability, the basic concepts of machine learning will be clarified together with explaining the Machine Learning (ML) algorithms that are used in this research.

- Chapter 3 (Database surrogate model): describes the database that is used to train, validate, and test the surrogate model. First, a simple dike geometry will be considered with five input parameters and one output parameter for the surrogate model to train on. In the following section, more complex dike geometries will be considered which require more input parameters. These additional input parameters can be considered to increase models prediction and accuracy. Finally, two different sampling methods are explained that will be applied in this research in order to create appropriate datasets.
- Chapter 4 (Methodology): explains the way to achieve a good surrogate model. The chapter starts with explaining how the D-Stability calculations are run in order to get the FoS for the different dike geometries. The section that follows is about the techniques and methods to clean, normalize and split the dataset into a training, validation, and test set. After the D-Stability calculations, the hyper parameter tuning techniques and the different hyper parameters that are considered for each ML model are explained. After hyper parameter tuning, the loss functions and the validation metrics will be described. The validation metrics are applied on the outcome of the ML models and from the validation metrics, a decision will be made which sampling method will be used for the more complex dike geometries and which ML model approaches the macro-stability of the inner slope of a dike the best. The surrogate model will be optimized by changing the dataset, ML models, or training procedures.
- Chapter 5 (Macro-stability model): is about applying the methods discussed in chapter 4. The surrogate model will be derived in this chapter. In addition to deriving the surrogate model for the simple input parameters, a sensitivity analysis will be performed to check which input parameters have the most influence on the FoS. The data is investigated to see what changes should be made in order to get an appropriate surrogate model that assess the macro-stability of the inner slope of a dike. In this chapter, sub-questions 1, 2 and 3 will be answered.

II Integrated dike design

- Chapter 6 (Integrated model): introduces the equations/models for the other engineering aspects that are considered in this research. The input and output parameters for each model will be clarified and sub-question 4 will be answered.
- Chapter 7 (Multi Criteria Analysis (MCA)): explains what a multi criteria analysis is and how it is applied in this study. Sub-question 5 will be answered in this chapter.
- Chapter 8 (Results integrated model): shows the results of an integrated dike design. For a certain dike geometry, the weights in the objective function are varied in order to see what the impact is of changing the weights to the outcome of the objective function.

III Conclusions and Recommendations

- Chapter 9 (Conclusions): summarizes the final conclusions of this research.
- Chapter 10 (Recommendations): highlights the aspects related to this research for further investigation. In addition to the recommendations for further research, some recommendations are made from the results of this research.

Chapter 2 Literature review

2.1 Introduction

The literature review explains the failure mechanisms macro-stability and wave overtopping and provides an overview of previous related research on the topics shear strength models, limit equilibrium methods (LEM), and machine learning (algorithms). For each topic, the connection of that topic and this research will be explained. In this research, the principles of WBI2017 are used for assessing the macro-stability of the dikes.

WBI2017 is the legal instrumentation for the assessment of dikes and consists of the regulations for determining the hydraulic loads and the strength and safety of primary flood defenses and for assessing the safety of primary flood defenses. WBI2017 is used for assessing the dikes since dikes must meet this new safety standards.

2.2 Failure mechanisms

A failure mechanism is defined as a certain action that occurs on a dike and as a consequence of this action, the dike will lose its retaining function and fails. Different types of failure mechanisms related to dikes exist. These are shown in Figure 2.1. In this research, the main focus is on the macro-stability of dikes since this failure mechanism is common and an important expense of dike sections which do not satisfy the standard [19]. The failure mechanism wave overtopping will be treated too since many dike sections also need to be reinforced based on this failure mechanism in dike reinforcement projects [1]. However, wave overtopping will be treated less in depth compared to dike instability since for macro-stability a surrogate model will be developed and for wave overtopping an existing equation/model will be used. These two failure mechanisms are highlighted in red in Figure 2.1.

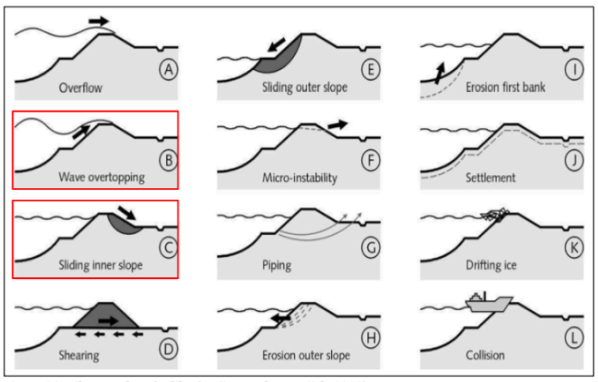


Figure 2.1: Dike failure mechanisms [68].

2.2.1 Macro-stability

When designing new dikes or assessing existing dikes, the stability of the slopes has to be checked. The exact definition of this macro-stability is: 'The ability to withstand loads without loss of function or excessive deformations' [71]. For macro-stability, information about the dike geometry, the soil layers in the underground, unit weight and strength characteristics of each layer, and the loads is required. When the dike slope is not stable due to decrease in strength of a dike, large parts of soil mass slide along straight or curved slip surfaces. The macro-stability is assessed by means of the Factor of Safety (FoS). The FoS is the ratio of the soil strength that is present and the soil strength that is required such that the soil body is exactly in equilibrium. To assess the macro-stability, two different components are considered: the driving and the resisting force. The driving force is caused by soil mass within the slip circle, water forces acting on the top and side of a soil slice within the slip circle, (parts of) the external load(s) within the slip plane, and the influences of earthquake forces. The resisting force is caused by shear stresses and normal effective stresses along the slip circle [59], possible soil improvement, the shear strength at the edges of the sliding section, and the weight of the soil at the passive side of the slip circle. The shear stress that will develop along the slip plane, is shown by the grey arrows in Figure 2.2.

Four types of failure mechanisms exist related to this main type; macro-stability inner slope, macro-stability outer slope, stability of the area foreshore of the dike, and stability during construction of the dike. In this research, the focus is only on the first mechanism; macro-stability of the inner slope of the dike since this type of failure mechanism resulted in one of the most registered failure cases [1].

The inner slope of the dike is the landward slope, as depicted in Figure 2.2. The inner slope of the dike can be in danger if the core of the dike and its foundation have become saturated. The dike can become very wet due to prolonged heavy rainfall and high water level. The rising water level together with the rainfall causes an increase of pore water pressures inside the dike. The increase of pore water pressures leads to a decrease in effective stress in the soil and a decrease in shear resistance of the soil. This can lead to failure of the inner slope of a dike.

Figure 2.2: Schematization of the failure mechanism: macro-stability inner slope [28]. When the inner slope of a dike is not stable, the whole dike will fail.

2.2.2 Wave overtopping

The failure mechanism wave overtopping occurs due to a combination of a high waterlevel and a large wave height. This combination can lead to water flowing over the crest of the dike going land inward. A dike won't fail directly when water hits the dike and flows over the dike crest. This failure mechanism develops over time. When wave overtopping lasts for a long time, erosion of the inner slope of the dike will take place, as depicted in Figure 2.3, and dike breach can occur.

Wave overtopping develops also over time in the sense that dikes encounter settlements over the years, which may decrease the crest height of the dike. The decrease of the crest height will increase the risk for wave overtopping, together with the increase of the water level. Besides that, wave overtopping can also lead to complete dike failure when dike instability is approached. The combination of approaching dike instability with associated settlements and wave overtopping can lead to dike failure.

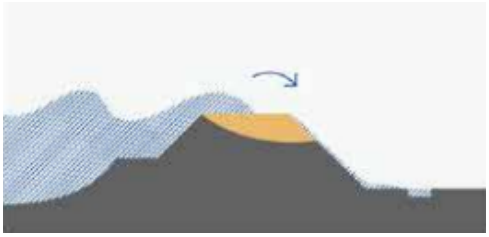


Figure 2.3: Schematization of the failure mechanism: wave overtopping [28].

2.3 Shear strength models

When a dike slope failure occurs, the soil will be sheared. During shearing, the soil can behave in 3 ways: completely undrained, partially drained or completely drained. If the soil is sheared sufficiently fast that no drainage of shear induced pore pressure arises, the undrained strength is mobilised [15]. The pore water pressure will increase and the effective stress decrease, which can result in a reduction in the soil's shear strength. If the shearing is at a very slow rate, such that the shear induced pore pressures dissipate, the drained strength is mobilised. First, when shearing is induced, the shear induced pore pressures are built up and the effective stress and the soil's shear strength decreases, but over time the pore pressures decrease and the effective stress and the soil's shear strength increases. If the shearing is not very slow, but also not fast enough, partially drained conditions will occur and the mobilised strength is in between the undrained and drained conditions. These different shearing modes with the corresponding types of soil behavior have an influence on the factor of safety against slope failure.

To assess slope stability of a dike, shear strength models are developed for different types of soils. Each type of soil will have a model that corresponds to the behavior the soil can express. The mobilisation of undrained strengths can only occur when undrained conditions develop. The types of soils that are susceptible to this undrained shearing are the fine-grained soils due to it's low permeability. That is why fine-grained soils are often modelled with an undrained shear strength model and coarse-grained soils with a drained shear strength model.

In D-stability, four types of shear strength models exist, also known as material models, to describe the strength of the soils. The four types of material models are Mohr Coulomb, Mohr Coulomb Advanced, SHANSEP, and SU Table. The first two models are for drained soil conditions and the last two for undrained soil conditions. The shear strength models that are used in this research are Mohr Coulomb Advanced and SHANSEP. Elaboration on the shear strength models will follow in the next sections.

2.3.1 SHANSEP

The SU Table and SHANSEP models are used for soils that are susceptible to undrained shearing. These soils are the cohesive soils. WBI2017 determines the undrained shear strength of cohesive

soils preferably with the SHANSEP model. The SU Table method is used for poorly permeable clay with relatively high saturated unit weights, unit weights greater than 17 kN/m^3. Since the unit weight of the cohesive soil is lower than 17 kN/m^3 in this research, the SHANSEP model will be used for cohesive soils. More detailed explanation on the unit weight is stated in section 3.2.2.

The shear stress along the slip plane from ratio S between the undrained shear strength and the vertical effective stress can be calculated with equation 2.1.

$$\tau = s_u = \begin{cases} \sigma'_v > 0 & \sigma'_v \cdot S \cdot (\frac{\sigma'_y}{\sigma'_v})^m \\ \sigma'_v = 0 & 0 \end{cases} \tag{2.1}$$

where:

S is the shear strength ratio;

m is the strength increase exponent as a result of over-consolidation [-];

σ'_v is the vertical effective stress [kN/m^2];

σ'_y is the maximum experienced vertical yield stress [kN/m^2];

S is the undrained shear strength divided by the effective vertical consolidation stress at an overconsolidation ratio (OCR) of 1 [47]. This soil parameter characterizes the undrained shear strength of a soil under normally consolidated conditions. Normally consolidated conditions means that the soil has never been subjected to a stress level higher than it's current vertical effective stress. The relation between maximum experienced vertical effective stress and the current vertical effective stress can defined by the OCR or pre overburden pressure (POP), which can be observed in Equation 2.2 and 2.3. In this research, this relation is expressed in POP in kN/m^2. If the POP value is high for a certain soil, the material is compacted which results in higher effective stresses, an increased shear strength, and decreased permeability.

$$OCR = (\frac{\sigma'_y}{\sigma'_v}) \tag{2.2}$$

$$POP = \sigma'_y - \sigma'_v = (OCR - 1) \cdot \sigma'_v \tag{2.3}$$

The strength increase exponent determines to which extent the loading history has an impact on the undrained shear strength. For the strength increase exponent, the correlation: m = 1 - Cs / Cc exists [47]. The value for Cs (swell index) and Cc (compression index) are derived from lab tests and differences per type of clay. The ratio Cs/Cc is typically around 0.2.

2.3.2 Mohr-Coulomb Advanced

The Mohr-Coulomb and Mohr-Coulomb advanced model are both for drained soils. The only difference between the two models is that Mohr-Coulomb Advanced has one more input parameter, namely the dilatancy angle. The dilatancy angle describes the plastic volume change of a soil during shearing [47]. In D-Stability, the dilatancy angle is set equal to the friction angle for the Mohr-Coulomb model. This is the well-known Mohr-Coulomb definition of the failure line

[59]. The failure line is a relationship between the normal and shear stress. The shear stress can not exceed the failure line, because then local failure will occur. Macro-instability of dikes occurs when the failure line, the maximum shear stress, is surpassed for all stress points along the slip circle.

The dilatancy angle is a measure of the tendency of a soil to dilate or expand when it is subjected to shear stresses. WBI2017 assumes a dilatancy angle of 0 °. When the dilatancy angle is 0 °, no plastic volume strains are developed and the soil does not experience any volume change during shear deformation. The shear strength is significantly lower compared to associative soil behavior, when the dilatancy angle is set equal to the friction angle. When a material dilates during shearing ($\psi > 0°$), it tends to strengthen (increase in shear strength) because the dilation allows for a greater distribution of stresses within the material, leading to increased interlocking and resistance to shearing. Given the dilatancy angle of 0 degrees, the shear strength will not be overestimated, which leads to safer circumstances.

For soft soils, the dilatancy can be negative, which indicates that during shearing the soil contracts. When the soil contracts, the pore pressure increases and the shear stress reduces which can lead to slope failure eventually. Especially in seismic areas, the pore pressure can increase enormously such that the effective stress approaches zero, leading to loss of shear strength. This process is known as liquefaction in which the soil behaves like a liquid. When a dike is built on soft soils, measures like preloading the subsoil, should take place in order to increase the shear stress.

If the friction angle is equal to the dilatancy angle, the soil should undergo a big volume change, which is never observed. A negative dilatancy angle is not common for most soil types in the Netherlands. For these reasons, the dilatancy angle is set to 0 °in this research.

In the Mohr-Coulomb model, the shear stress is calculated with equation 2.4 [59].

$$\tau = c \cdot \frac{\cos\psi \cdot \cos\varphi}{1 - \sin\psi \cdot \sin\varphi} + \sigma'_n \cdot \frac{\cos\psi \cdot \sin\varphi}{1 - \sin\psi \cdot \sin\varphi} \tag{2.4}$$

When the dilatancy angle equals 0 °, the shear strength becomes [59]:

$$\tau = c \cdot \cos\varphi + \sigma'_n \cdot \sin\varphi \tag{2.5}$$

where:

- c is the cohesion [kN/m^2];
- φ is the friction angle [°];
- σ'_{n} is the normal effective stress [kN/m^2];
- ψ is the dilatancy angle [°];

The cohesion, friction angle and the normal effective stress can vary within a soil layer.

2.4 Calculation methods for LEM

In this section, different Limit Equilibrium Methods (LEM's) will be evaluated. The macro-stability of a dike is assessed by the FoS, as mentioned in section 2.2.1. Various equilibrium methods exist that consider a slip plane that goes through the soil profile, for which the FoS will be calculated. This factor will give an indication if the slip plane is stable or not.

In D-stability, three different LEM's are available; Bishop's method, Uplift-Van, and Spencer- Van der Meij. To evaluate if a slope is stable, it should meet the equilibrium criteria, which depends on the limit equilibrium method. The different LEM's will be explained in the next sections.

2.4.1 Bishop

In the Dutch consultancy, Bishop's method is commonly used to calculate the macro-stability of dikes using circular slip planes. In this model, vertical forces and moment equilibrium around the center of the slip surface will be considered. The soil is divided into slices. For each slice, the resultant of the forces in vertical direction is equal to zero and the maximum shear stress will be calculated. In this theory, it is assumed that the maximum mobilized shear stress along the slip surface is indeed mobilized. Achieving a failure state is assessed based on the normative load against the maximum available stress in the soil.

Bishop's method will be used for the 'Simple' model in this research. The Simple model is explained in the next chapter. This Simple model does not have a complex dike geometry. The dike consists only of slopes and no berms or something else added to it, therefore it is expected that the slip plane will be circular.

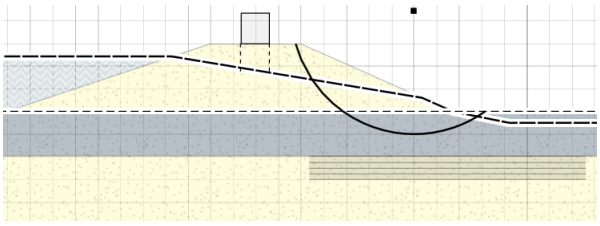


Figure 2.4: A schematized dike with a developed slip circle using Bishop's method.

2.4.2 Uplift-Van

The Uplift-Van method, is the circular slip plane from Bishop's method combined with a horizontal slip plane. The slip plane consists of an active circle, passive circle and a horizontal bar between the circles [59]. This horizontal bar is loaded horizontally by the dike and vertically by the upward water pressure. Similar to Bishop's method, moment equilibrium is considered for the circular parts of the slip plane. For the horizontal part of the slip plane, horizontal balance will be considered. High pore pressures at the horizontal interface of weak layers with an underlying permeable sand layer can cause reduction or even complete loss of shear resistance at this plane [59]. This can result in an uplift failure mechanism. FoS is determined by the horizontal forces equilibrium acting on the horizontal bar in between the circular slip planes.

In the schematization manual of WBI2017 for macro-stability, it is stated that the Uplift-Van method is used to analyse the macro-stability of the inner slopes of dikes. Besides that, this method commonly used to assess dike slopes. For these reasons, the Uplift-Van method is used to evaluate slope stability for the 'Complex' model, the end surrogate model for macro-stability

of dikes, in this research.

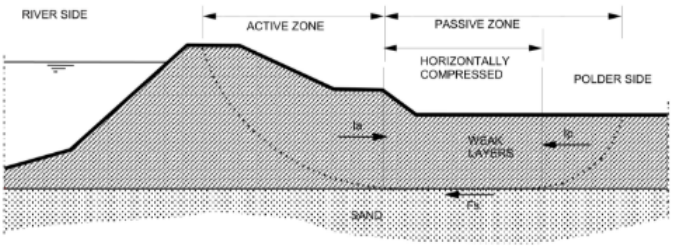


Figure 2.5: A schematized dike with a developed slip circle using Uplift Van method.

2.4.3 Spencer- Van der Meij

Bishop's method calculates the macro-stability of dikes using circular slip planes. However, the circular shape does not result in the most unfavorable case. If a soil layer in the underground has a limited thickness with low strength properties, then the stability should be checked for another shape of slip plane shear due to the bad quality of the soil layer. Spencer's method satisfies besides the moment equilibrium and vertical forces equilibrium (Bishop's method), the horizontal forces equilibrium too on each slice.

Spencer's method ensures that the critical slip surface that is found is actually the slip surface which leads to the lowest safety. In this method, the slip surfaces are not limited to circular shapes, but can take any shape [64].

This method is not used in this research, since the calculation time is longer for this method compared to Bishop's method and Uplift-Van. This model is used less compared to the other two. In addition to that, simple dike geometries will be created to assess macro-stability of dikes in this research. Bishop and Uplift Van models already capture the relevant slip surfaces and therefore it is not necessary to divert to a more complex limit equilibrium method.

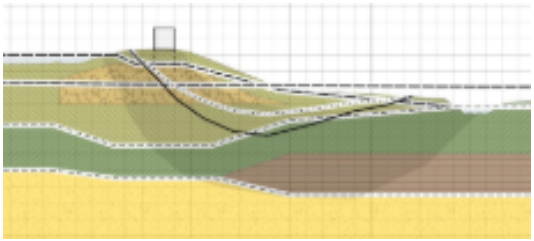


Figure 2.6: A schematized dike with a developed slip circle using Spencer- Van der Meij method.

2.5 Basic concepts of machine learning

The basic concepts of machine learning contains the types of machine learning algorithms that exist and the bias-variance trade-off in machine learning. These two aspects will be elaborated in this chapter.

2.5.1 Different types of machine learning

The machine learning algorithms can be divided in three broad categories: supervised, unsupervised, and reinforcement learning. These three categories with the associated subcategories and models can be observed in Figure 2.7. In this research, supervised machine learning is used

only. Supervised machine learning is a type of machine learning in which models are trained on a labeled dataset to predict outcomes. The labeled dataset is the input data along with corresponding target value(s).

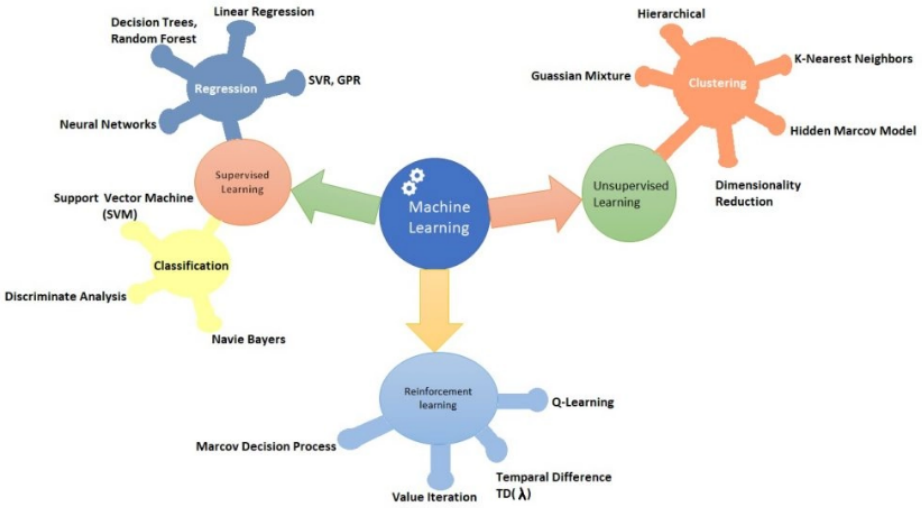


Figure 2.7: Different types of machine learning with associated subcategories and models [34].

In this research, the data derived from the D-stability calculations is labelled data, since each combination of input parameters have a corresponding FoS. That is why supervised machine learning is applied.

2.5.2 Bias and variance trade-off

The bias and variance trade-off is a concept in machine learning that refers to the balance between two sources of error: bias and variance, which cover the reducible error. Irreducible error also exists and this comes from the data due to noisy observations.

Bias is the difference between the expectation of the machine learning (ML) model and the optimal prediction, such that it will minimize the loss function. If the model is biased, it tends to underfit the training data. This means that the model fails to capture underlying patterns and relations in the data and oversimplifies the ML model, which results in bad performance on both the training and test dataset.

Variance refers to the magnitude of deviation of the current machine learning model to the average model. If the variance is large, the machine learning model tends to overfit the training data. This means that the machine learning model captures noise or random fluctuations in the training data as it were useful information and training a ML model on a new dataset would result in a very different model. The ML model performs well on the training dataset, but poorly on the test dataset.

There is a trade-off between bias and variance. When a ML model is biased, but has a low variance, the ML model is a simple model that has a similar performance across different training sets, but the overall performance is not well due to underfitting. When a ML has a high variance, but is not biased, the ML model is a complex model that fit the training data well, but performs bad on unseen (test) data.

Managing the bias and variance trade-off is crucial for building ML models since it has an impact on how well the model performs with unseen data. The bias and variance trade-off can be observed in Figure 2.8. The best balance between the bias and variance is the point where the generalization error is the lowest.

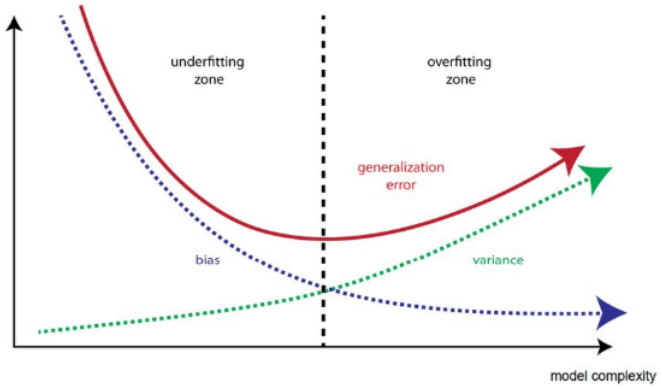


Figure 2.8: The bias and variance trade-off [70].

2.6 Machine learning algorithms

The development of machine learning algorithms and techniques has become faster in recent years and therefore many algorithms are established, each suited to different types of tasks, data, and problem domains. The choice of the best machine learning algorithm depends on many factors including data characteristics, the problem being addressed, computational resources, prior experience, and model complexity. Therefore, it often requires experimentation and evaluation of multiple algorithms to determine the most suitable algorithm for a given problem.

Based on most commonly cited machine learning algorithms in literature, recent thesis reports related to this thesis topic [24][70], and the ML algorithms that are often mentioned by different users of the online platform Kaggle, four different ML algorithms are chosen in this research. Four different ML algorithms are chosen in order to test a limited, varied amount of ML algorithms. Random Forest Regression (RFR), Support Vector Regression (SVR), and the Feedforward Neural Network (FNN) are three complete different ML algorithms and eXtreme Gradient Boosting (XGB) is quite similar to RFR, but it operates based on different principles and mechanisms compared to RFR and is very popular in the field of machine learning due to its exceptional performance, efficiency, and flexibility as a consequence of its advanced features.

Based on literature, one type of machine learning algorithm can not be chosen, since the performance of the machine learning model depends on the data on which it is trained. That is why, for each new generated dataset with the D-stability calculations, the four different machine learning algorithms should be trained again. In order to fully understand how these different ML models work, the main take-aways of online video's about the different ML models are written down in this chapter.

2.6.1 Random Forest Regression (RFR)

Random Forest Regressor (RFR) [8] is an ensemble learning method, which combines multiple individual models to make a better prediction and to reduce over-fitting. Combining the predictions of the individual models is called aggregation. When a ML model is over-fitted it performs well on the training data, but bad on the test data, as mentioned in section 2.5.2. The random forest consists of various decision trees and these trees represent each a model that tries to learn the relationship between the input features and the target variable. Each tree in the forest is trained on a random subset of the training data and a random subset of the available features, which indicates that two random processes are going on in a tree; bootstrapping and random feature selection [21]. In Figure 2.11, the exact definition of a random forest is shown, in which it can be observed that it is a bagging-based algorithm by combining bootstrapping with aggregation. The random forest method can be used for classification and regression. Since the surrogate model for macro-stability will predict the FoS, the random forest algorithm should be based on an ensemble of regression trees instead of classification trees.

In RFR, the trees run in parallel, as can be observed in Figure 2.9. Each individual tree is a model, which has branches, nodes, and leaves. Each decision tree is built independently of the others, since no interaction exists between the trees. At each node in a regression tree, the data is split based on a chosen feature and threshold value such that the difference between the predicted and the actual target value is minimized. A root node is the entry node on top of the regression tree, where a first decision boundary is set by asking if the first selected feature is less than or greater than an optimal threshold [22]. The decision tree will expand until the decision is made about the target value in the last node, the leaf node, the predicted FoS. The leaf node is reached when the stopping criterion is met, such as the maximum depth of the tree is reached by having a minimum number of samples in the nodes or when further splitting do not result in improving model's performance.

RFR builds a number of independent regression trees based on random subsets of training data and combines the results of all individual trees to make a final prediction for the FoS. This process is called bagging. To improve the generalization performance even more, a random subset of features are selected for splitting at each node in the decision tree. This reduces the correlation between trees in the forest.

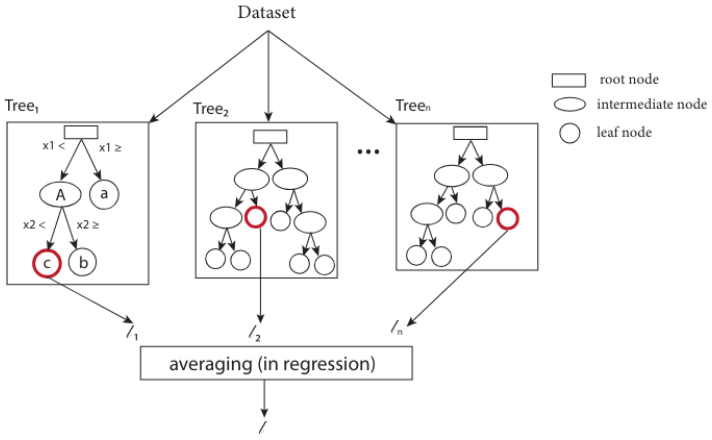


Figure 2.9: Schematization of the RFR structure, in which x are the features, n the number of trees, and l the predicted value [22].

2.6.2 eXtreme Gradient Boosting (XGB)

eXtreme Gradient Boosting (XGB) [9] is an ensemble learning algorithm based on decision trees, just like RFR, but they differ in their approach. XGB uses a gradient boosting framework, which can be observed in Figure 2.11, and RFR uses bagging to get the final prediction. XGB looks similar to RFR, but it has been added to the machine learning models list in this research, since this algorithm has become a popular algorithm in the ML branch, such as on online platforms like Kaggle, an online community of data scientists and machine learning engineers. RFR is quite robust to its hyper parameter values, but XGB often achieves higher predictive performance, which makes it very popular. The performance of XGB with respect to other regression algorithms is shown in Figure 2.10.

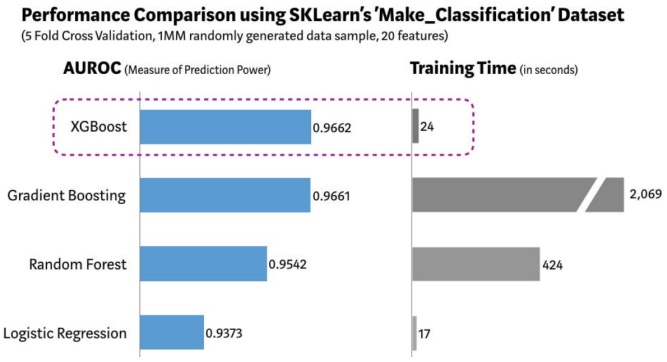


Figure 2.10: Performance comparison of different ML regression models [38].

XGB uses boosting instead of bagging, which RFR uses. In RFR, multiple decision trees are built independently and the prediction of each tree is combined to get the final prediction. This process is called bagging. However, in XGB, boosting makes sure that the decision trees are built sequentially, which means that each new tree in the sequence is trained to correct the errors made by the previous tree. The final prediction is made by combining the predictions of all trees, often weighted by their individual performance. Gradient Boosting (GB) is a type of boosting algorithm that combines boosting with gradient descent optimization to control over-fitting and improve generalization performance by minimizing the loss function. In GB, each new tree in the sequence is trained to correct the errors made by the previous tree, just like boosting, but this time the new model is fitted to the residuals, which are the differences between the 'final' predictions so far and the target values. The loss function is optimized by minimizing the residuals. eXtreme Gradient Boosting is built upon the foundation of GB. XGB is an optimized implementation of GB and is often faster than the traditional gradient boosting due to certain optimizations such as parallel processing to speed up the training and prediction processes and tree-pruning and regularization to prevent overfitting, to control the complexity of the model, and to improve the generalization of the model. The definitions for boosting, GB, and XGB are also stated in Figure 2.11.

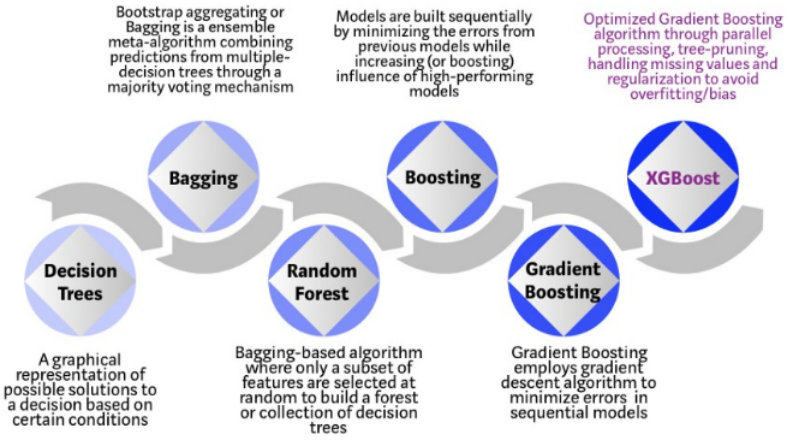


Figure 2.11: Evolution of decision-tree-based algorithms[38].

2.6.3 Support Vector Regression (SVR)

Support Vector Regression (SVR) [11][25], is also known as Support Vector Machine (SVM) for regression. SVM is mainly used for classification problems, but can also be used for regression. For classification problems, Support Vector Classification (SVC) tries to find a line (data is 2D) or hyperplane (data is multi-dimentional) that separates two classes. A new point will be assigned to the class depending on which side of the line/hyperplane the point is. The optimal decision boundary is one that maximizes the margin, which is the distance between the line/hyperplane and the closest data point from each class, the support vectors [43]. The same principle for SVC is used for SVR, but SVR is more difficult since a real number must be predicted.

SVR also uses the concepts of hyperplane and margin, but the definitions are not the same. SVR aims to find a function that approximates the relationship between the input variables and a continuous target variable, while minimizing the prediction error [52]. This function, also known as hyperplane, should fit the data points in a continuous space by mapping the input parameters to a high dimensional feature space. After this mapping, the hyperplane is found by maximizing the margin ε between the hyperplane and the closest data points to ensure that most data points lie within this margin and at the same time minimizing the prediction error. The decision boundary, depicted in Figure 2.12, has a distance ε from the hyperplane and this distance acts as a regularization term to avoid overfitting. The data points that lie on or within the margin boundary are only taken and these data point are called the support vectors. A prediction of the target variable is made by using the function that fit to these support vectors.

SVR can be applied to linear and non-linear relationships between the input and output variables by using different kernel functions. When data is non-linearly separable, the kernel helps to find a function in a higher dimensional space where a linear regression problem can be solved [43].

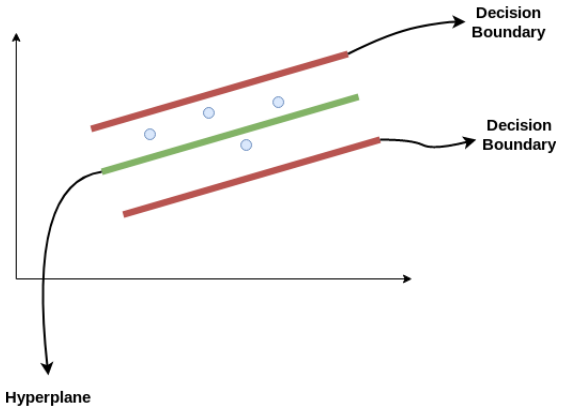


Figure 2.12: The best fitted hyperplane to the support vectors within the margin [52].

2.6.4 Feedforward Neural Network (FNN)

Feedforward Neural Network (FNN) [32] is a type of Artificial Neural Network (ANN), in which information flows in one direction without feedback. A FNN consists of neurons connected by weights, as shown in Figure 2.13. The information flows from the input neurons, purple in Figure 2.13, towards the output neurons, turquoise in Figure 2.13. The input neurons are the input parameters of the model and the output neuron is the output parameter. The yellow dots in Figure 2.13 are the hidden neurons, forming hidden layers. Each vector/column of neurons in Figure 2.13 forms a layer in the neural network. The neurons of one layer depend on (at least) the neurons of the previous layer [49]. FNN learns the pattern in the data by updating the weights based on the error of the output.

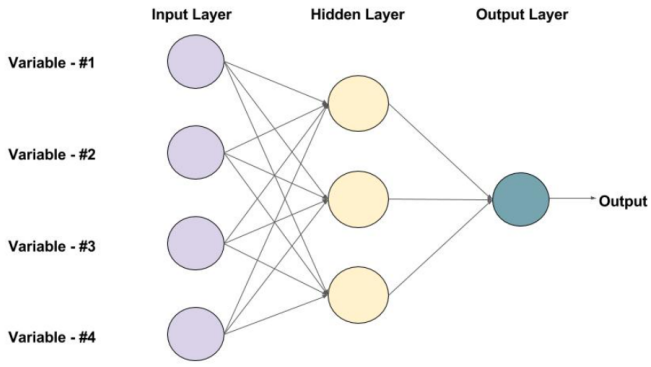


Figure 2.13: The structure of a FNN consisting of 4 input neurons, 1 hidden layer with 3 neurons, and 1 output neuron [29].

Training the FNN involves two phases: the feed forward and the back propagation phase.

During the feed forward phase, the input data is fed into the network and moves from layer to layer to reach the output layer. This moving forward can be broken down into 2 basic steps. First, for an arbitrary neuron of any given layer, a linear combination of a bias term and the values coming from the previous layer with weights is performed (the weighted sum of the inputs), which results in a value which is called A in Figure 2.14. The second step is introducing some non-linearity in the network. Value A will be passed through a predefined non-linear activation function (hyper parameter), resulting in neuron value Z. In this way the model tries to

learn and approximate more complex relationships in the data, which means that the model can represent highly non-linear functions in order to get the desired degree of accuracy. Without the activation functions between the hidden layers, the FNN just performs linear transformations. After passing through the activation function, value Z will move to the next layer, and so on till the final value, the output parameter, is reached and a first prediction is made.

The back propagation phase starts with the first prediction that is made. The difference between the predicted output from the FNN and the actual output is then calculated by a predefined error function. The back propagation is part of the training process when different epochs, multiple passes through the training dataset, are performed so that the calculated error of the training prediction is used to adjust the network's weights and to improve the model's performance.

The feed forward and back propagation phase is done multiple times until the difference between the predicted and the actual output will converge to a state where the error is very small and the network performs satisfactorily on the training data.

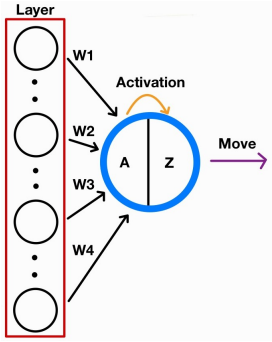


Figure 2.14: Schematization of the feedforward phase of one arbitrary neuron, in blue. The red box depicts the neurons of one layer, the black arrows are the weights, the orange arrow shows the activation function, and the purple arrow the information transportation to the next layer.

2.7 Conclusion

This chapter shows the important aspects from the literature review. Two failure mechanisms are discussed in this chapter, macro-stability of the inner slope and wave overtopping which are captured by a (surrogate) model. Two of the four shear strength models from D-Stability are used in this research, namely SHANSEP and Mohr-Coulomb Advanced. Two of the three LEM's from D-Stability are implemented in this research, Bishop in the Simple model and Uplift-Van in the Complex model.

Sections 2.2 until 2.4 covers the literature review that is needed to create the data for deriving the surrogate model. What this data looks like and how it is created will be explained in chapter 3. Sections 2.5 and 2.6 are about the machine learning part, which type of machine learning is used in this research and what the different ML models look like. Each ML model can have multiple input parameters to assess the FoS. For visualisation purposes, the slip circle can also be predicted. When the slip circle needs to be predicted, next to the FoS, each ML model should be trained on more output parameters related to geometry of the slip circle. Every ML model that is considered in this research can have multiple outputs, except for the standard SVR model. This model should be adjusted in order to predict multiple outputs.

Chapter 3 Database surrogate model

3.1 Introduction

In this chapter, the data is discussed that is needed to calculate the macro-stability of the inner slopes of dikes and to train the surrogate model for macro-stability later. To calculate the macro-stability of the inner slopes of dikes, data on soil properties, dike geometry, groundwater conditions, hydrology, and loading conditions are needed.

In dike reinforcement projects, the soil properties are usually fixed. Soil investigation is performed and the soil properties derived from this investigation are used. When existing dikes have to be reinforced, then the dike geometry is examined first in order make the inner slopes of dikes more stable. That is why the focus of this research is on varying dike geometries to create stable inner slopes of dikes. This means that the dataset that will be created consists of different dike geometries with the same soil properties and for these dike geometries the FoS will be calculated with D-Stability.

In addition to varying dike geometries, the hydrological conditions and the thicknesses of the soil layers underneath the dike will also vary for the different dike geometries since these differ per location. The groundwater table and the uniform traffic load are constant. The average groundwater level in the Netherlands is 1 m below ground level [16]. In the technical report of water retaining soil structures, it is stated that a traffic load of 13.3 kPa operating over a distance of 2.5 meters should be applied on top of the crest of the dike, even when there is no road on top of it [53]. The average groundwater level is used as value for the groundwater table and the traffic load of 13.3 kPa as value for the load on top of the crest of the dike.

The different dike geometries with the different hydrological conditions and thicknesses of the soil layers underneath the dike, input, together with the corresponding FoS, output, will serve as dataset to train the ML algorithms.

In the Netherlands, a few main types of dike constructions exist, namely clay, sand, peat and sand dike with clay cover [62]. In this research, the focus is on primary flood defences, which are mostly sand and clay dikes, therefore peat dikes are not considered.

For the macro-stability analysis, WBI2017 compiled one dike cross-section with several characteristic points. This cross-section is shown in Figure 3.1. This schematization of a dike cross-section is the starting point for the different cross-sections that are considered in this research.

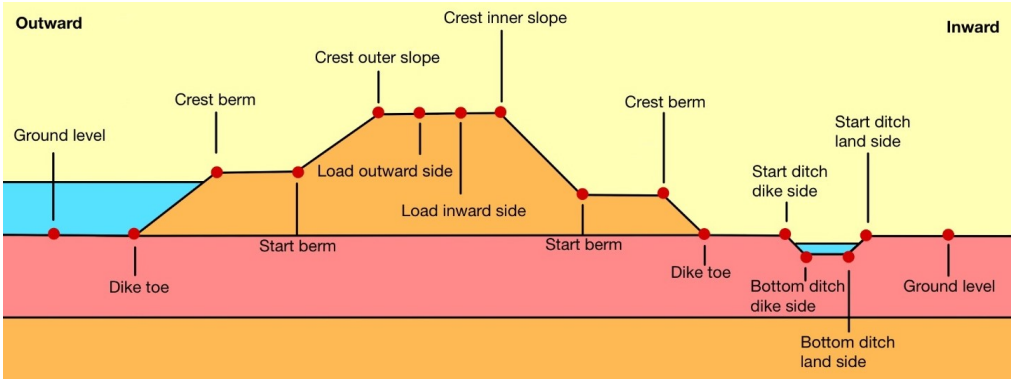


Figure 3.1: Schematization of a dike with characteristic points. The characteristic points load outward side and load inward side refers to start and end point where the uniform traffic load is located.

In section 3.2, a simplification of the cross-section in Figure 3.1 is considered first. Simple dike geometries are considered first in order to see if the ML algorithms, that are trained on the dataset containing information of dikes with this geometry, can predict the FoS accurately. In section 3.3, a similar cross-section with the corresponding characteristic points as in Figure 3.1 is considered. The ML algorithms that are trained on the simple dike cross-sections are referred to as 'Simple models' and the algorithms trained on similar cross-sections as Figure 3.1 are referred to as 'Complex models' hereafter. In sections 3.4 and 3.5, the strategy is explained to get the values for the different input parameters that correspond to the different dike geometries.

3.2 Simple model

3.2.1 Input parameters

The dike geometry for the Simple model can be observed in Figure 3.2. It consists of an outer and inner slope and no outer or inner berm are attached to the dike and no ditch is added to the dike geometry. To create a dataset containing different dike geometries, values for the three parameters related to the geometry, one parameter related to the hydrological conditions, and one parameter related to the schematization of the underground, which can be observed in Figure 3.2 by the grey boxes, are varied. The outer slope of the dike will not be varied since the main focus is on inner slope stability of the dike and is equal to 1:3. The space before the outer dike toe (5 m) is smaller than the space behind the inner dike toe (20 m) since the main focus of this research is on developing a slip plane on the inner side of a dike (inner slope). The space behind the inner dike toe is enough to develop the slip circle near the inner slopes of the different dike geometries. The output of the surrogate model is the FoS. In Figure 3.3, a clear overview of the different input parameters and the output parameter for the Simple model is displayed.

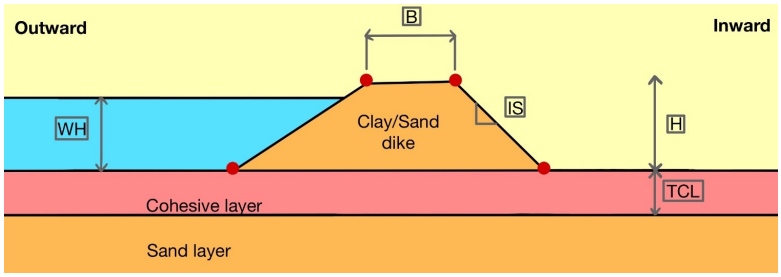


Figure 3.2: Schematization of the simple dike with the input parameters to train, validate, and test the surrogate model.

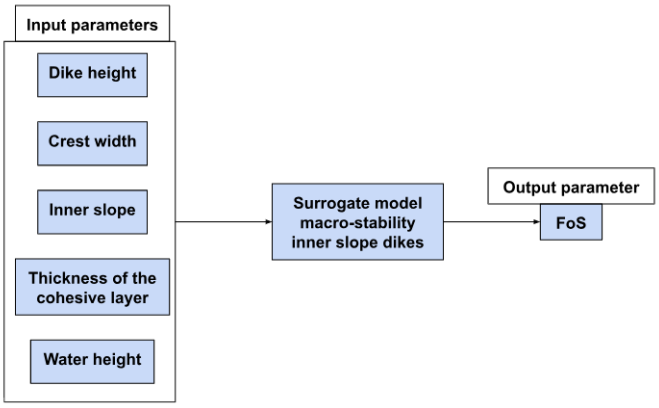


Figure 3.3: The different input parameters and the output parameter for the surrogate model trained on simple dike geometries.

These five parameters are part of the input parameter sets. In Table 3.1, the minimum and maximum value for each input parameters is stated. The different input parameters can take any value that is at the corresponding range. The different values for the different input parameters will result in different dike geometries and input parameter sets. In Table 3.1, the references for the values of the input parameters are shown too.

Parameter	Symbol	Min. value	Max. value	Unit	References
Crest width	B	3	10	m	[57] [40]
Inner slope	IS	1:2*	1:4	-	[40] [46], current dike reinforcement projects, soil type
Dike height	H	3.5	6	m	current dike reinforcement projects
Water height	WH	3	5.5	m	current dike reinforcement projects
Thickness cohesive layer	TCL	0	8	m	[13]

Table 3.1: The minimum and maximum values for the input parameters to train, validate, and test the surrogate model for the Simple model.

*The minimum value for the inner slope of the dike is rounded in Table 3.1. This value is calculated with the lowest friction angle for the considered soils in this research. Clay has a lower friction angle than sand in this research, as can be observed in the section 3.2.2, which means that the minimum value for the inner slope of a dike is related to the friction angle of clay. Clay has a lower friction angle than sand since clay particles are much smaller than sand particles allowing that clay particles can slide over each other more easily, reducing the resistance to shear force. This movement results in a lower friction angle.

3.2.2 Soil parameters

In section 2.3, the two shear strength models from D-stability are mentioned that are used in this research; SHANSEP and Mohr Coulomb Advanced. For clay, Mohr Coulomb Advanced is used above the phreatic level and SHANSEP below the phreatic level. The dike core is not fully saturated, which makes the Mohr Coulomb Advanced model more suitable for this soil than SHANSEP. For sand, Mohr Coulomb Advanced is used.

These shear strength models require parameters related to the soil conditions. In this research, one type of sand and one type of clay is chosen to keep things simple. Sand will be placed in the dike core and underneath the cohesive layer and clay will also be placed in the dike core and serve as the cohesive layer underneath the dike core. The parameters related to the soil conditions mainly come from Table 2B of the NEN 9997-1.

The clay type that is considered in this research is called klei-schoon-matig in Table 2B of the NEN 9997-1, since this material has a unit weight that is common in dike cores and does not contain large amount of silt, sand or other soil particles. The unit weight of this clay type is 17 kN/m^3, but this unit weight has been reduced in this study to 16 kN/m^3 since this is a boundary given for use of the SHANSEP model in WBI2017. The unit weight is the same for saturated and unsaturated clay since the unit weight for different types of clay in saturated and unsaturated conditions is also the same in Table 2B of the NEN 9997-1. The soil properties friction angle and cohesion are derived from Table 2B of the NEN 9997-1. However, the friction angle belongs actually to the soil klei-schoon-vast in Table 2B from the NEN 9997-1, since a higher friction angle is more common for a dike body as can be observed in the macro-stability manual WBI2017. The dilatancy angle is set to 0°as mentioned in section 2.3.2. No soil data is considered in this research, which makes it difficult to determine appropriate values for the strength increase exponent (m), shear strength ratio (S), and pre overburden pressure (POP).

By using the correlation m = 1 - Cs/Cc with a value of 0.2 for the ratio Cs/Cc as mentioned in section 2.3.1, a value of 0.8 is derived for m. According to the manual 'Handreiking voor het bepalen van schuifsterkte parameters WTI 2017 Toetsregels Stabiliteit' from Deltares, when no soil data is available, a safe value for m is 0.8 [60], which is in line with the correlation.

The lowest expected value for the POP, according to the macro-stability manual of WBI2017, is around 15 kN/m^2 and this value is used for clay. The expectation value for S for clean clay, according to the macro-stability manual is 0.25 [47].

The clay parameters for the SHANSEP model and Mohr Coulomb Advanced can be observed in Table 3.2 and Table 3.3.

Soil parameter	Symbol	Value	Unit
Unit weight saturated	γ_{sat}	16	kN/m^3
Unit weight unsaturated	γ_{unsat}	16	kN/m^3
Shear strength ratio	S	0.25	-
Strength increase exponent	m	0.8	-
Pre Overburden Pressure	POP	15	kN/m^2

Table 3.2: The clay parameters for the cohesive layer under the dike. The soil parameters correspond to the input parameters for the SHANSEP model.

Soil parameter	Symbol	Value	Unit
Unit weight saturated	γ_{sat}	16	kN/m^3
Unit weight unsaturated	γ_{unsat}	16	kN/m^3
Friction angle	φ	25	°
Cohesion	c	5	kPa
Dilatancy angle	ψ	0	°

Table 3.3: The clay parameters for the dike itself, in the case if the core of the dike contains clay. The soil parameters correspond to the input parameters for the Mohr Coulomb Advanced model.

Reflecting on the POP value of clay, a value of 15 kN/m^2 leads to a very high over consolidation ratio (OCR) near ground level. This means that the stress experienced in the past is much higher than the stress that is experienced now. In combination with the value for the strength increase exponent will lead this to a very high undrained shear strength. This is known in this study. The soil parameters for clay are chosen in order to get a reasonable proof of concept. In practice, some cohesion is added to the top soil layers to avoid unrealistic shallow slip circles instead of having a high undrained shear strength near ground level, but this is not applied in this research since this is not the main purpose of this study. The main purpose of this study is to create a conceptual model that assesses the macro-stability of inner slopes of dikes. That is why the POP value is the same for all amounts of clay that is considered in each dike geometry.

The sand type that is considered in this research is called zand-schoon-slap in Table 2B from the NEN 9997-1. Like the clay type, this sand type is also clean and does not contain any other soil particles. The soil properties unit weight, friction angle, and cohesion are derived from Table 2B of the NEN 9997-1. The dilatancy angle is set to 0 °as mentioned in section 2.3.2. The sand parameters for the Mohr Coulomb Advanced model can be observed in Table 3.4.

Soil parameter	Symbol	Value	Unit
Unit weight saturated	γ_{sat}	19	kN/m^3
Unit weight unsaturated	γ_{unsat}	17	kN/m^3
Friction angle	φ	30	°
Cohesion	c	0	kPa
Dilatancy angle	ψ	0	°

Table 3.4: The sand parameters for the sand layer under the cohesive layer and for the dike itself, in the case if the core of the dike contains sand. The soil parameters correspond to the input parameters for the Mohr Coulomb Advanced model.

3.2.3 Phreatic level

The phreatic level inside and outside the dike for a dike core containing clay or sand is schematized in Figure 3.4 and Figure 3.5. Initially, the phreatic line in-/and outside the dikes were schematized following the rules from the Technische Adviescommissie voor de Waterkeringen [54]. However, when using the input parameters from sections 3.2.1 and 3.2.2, the phreatic line was schematized very high in the dike core, which resulted in high water stresses inside the dike body and low FoS's. That is why a different approach for the phreatic line has been chosen. The phreatic line schematization from the D-Stability manual with corresponding offset values results in a lower phreatic line inside the dike core, lower water stresses inside the dike body and higher FoS's. These schematizations can be observed in Figures 3.4 and 3.5 for dikes containing clay or sand inside its core and both will be used in the same training dataset in this research.

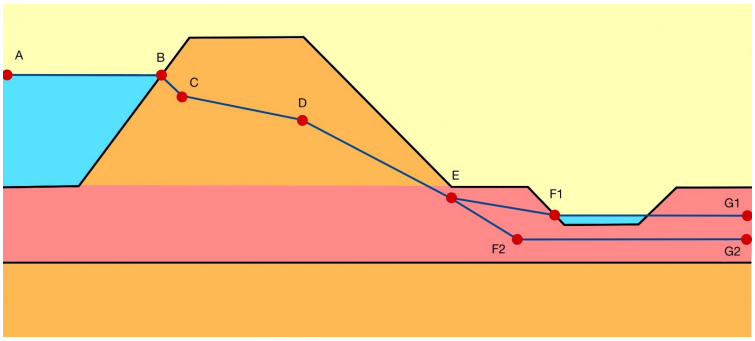


Figure 3.4: Schematization of the phreatic line inside and outside a dike with a clay core.

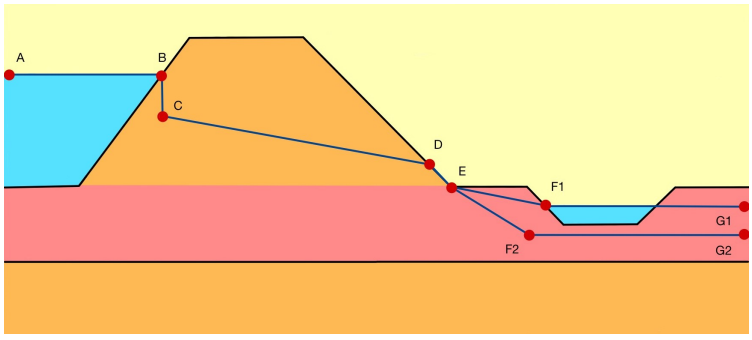


Figure 3.5: Schematization of the phreatic line inside and outside a dike with a sand core.

In the D-Stability manual, the derivation of the coordinates of the different points through which the phreatic line passes is explained. In Table 3.5 and 3.6, the points through which the phreatic line passes with the corresponding characteristics for a dike core containing clay and sand are stated. The phreatic line runs from point A until G, depending on whether a ditch is present or not. For the Simple model, a ditch is not present and therefore the phreatic line goes through points F2 and G2.

Point	x coordinate	y coordinate
A	$x_{start\_dike\_geometry}$	$y_{outer-waterlevel}$
B	$x_{intersection\_outer-waterlevel\_and\_outer-dikeslope}$	$y_{outer-waterlevel}$
C	x_B - 1	y_B - C_{offset}
D	$x_{crest\_inner\_slope}$	y_C - D_{offset}
E	$x_{inner\_toe\_dike}$	$y_{inner\_toe\_dike}$ - (0.25 * $y_{outer-waterlevel}$)
F1	$x_{intersection\_ditch-waterlevel\_and\_ditchslope}$	$y_{ditch-waterlevel}$
F2	$x_{inner\_toe\_dike}$ + F_{offset}	$y_{groundwaterlevel}$
G1	$x_{end\_dike\_geometry}$	$y_{ditch-waterlevel}$
G2	$x_{end\_dike\_geometry}$	$y_{groundwaterlevel}$

Table 3.5: The coordinates of the points with characteristics of the dike through which the phreatic line passes for a dike with a clay core.

Point	x coordinate	y coordinate
A	$x_{start\_dike\_geometry}$	$y_{outer-waterlevel}$
B	$x_{intersection\_outer-waterlevel\_and\_outer-dikeslope}$	$y_{outer-waterlevel}$
C	x_B	y_B - (0.5 * $y_{outer-waterlevel}$)
D	$x_{intersection\_0.25*outer-waterlevel\_and\_inner-dikeslope}$	y_E + (0.25 * $y_{outer-waterlevel}$)
E	$x_{inner\_toe\_dike}$	$y_{inner\_toe\_dike}$
F1	$x_{intersection\_ditch-waterlevel\_and\_ditchslope}$	$y_{ditch-waterlevel}$
F2	$x_{inner\_toe\_dike}$ + F_{offset}	$y_{groundwaterlevel}$
G1	$x_{end\_dike\_geometry}$	$y_{ditch-waterlevel}$
G2	$x_{end\_dike\_geometry}$	$y_{groundwaterlevel}$

Table 3.6: The coordinates of the points with characteristics of the dike through which the phreatic line passes for a dike with a sand core.

In the D-Stability manual, two default values are mentioned; C_{offset} and D_{offset}. The values for these points are 1 and 1.5 m.

The average groundwater level in the Netherlands is 1 m below ground level [16] as mentioned in the introduction of this chapter. Therefore it is assumed that for the dikes geometries with no ditch on the land side of the dike, after point E, the phreatic level reaches a depth of 1 m below ground level over a horizontal distance of 5 m (F_{offset}) after reaching the inner toe of the dike.

3.3 Complex model

3.3.1 Input parameters

The dike geometry for the Complex model can be observed in Figure 3.6. It consists of an outer and inner slope, an inner berm attached to the dike and a ditch added to the dike geometry. To create a dataset containing different dike geometries, values for the six parameters related to the geometry, one parameter related to the hydrological conditions, and one parameter related to the schematization of the underground, which can be observed in Figure 3.6 and Table 3.8, are varied. The constant parameters related to the dike geometry can be observed Figure 3.6 and Table 3.7. The output of the surrogate model is the FoS. A clear overview of the different input parameters and the output parameters for the surrogate model trained on complex dike geometries can be observed in Figure 3.7

The space before the outer dike toe (5 m) is smaller than the space behind the ditch (20 m) since the main focus of this research is on developing a slip plane on the inner side of a dike (inner slope). The space behind the inner dike toe is enough to develop the slip circle near the inner slopes of the different dike geometries.

The difference with regard to the dike core in the Simple model is that the dikes with the sand core have a clay cover over the sand core now. In the Netherlands, dike cores containing only sand is not common, but with the clay cover it is.

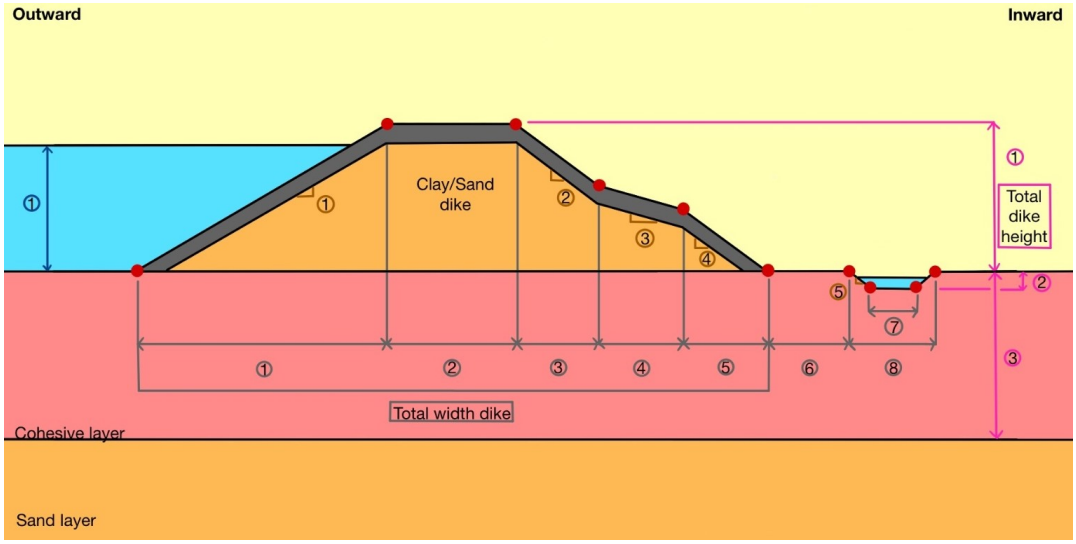


Figure 3.6: Schematization of the complex dike with the different width, height, and slope components the dike consists of and the water level to train, validate, and test the surrogate model.

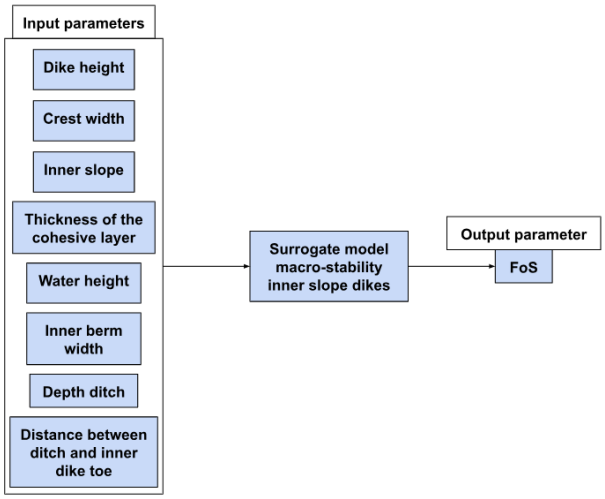


Figure 3.7: The different input parameters and the output parameter for the surrogate model trained on complex dike geometries

The eight parameters stated in Table 3.8 are part of the input parameter sets. In Table 3.8, the minimum and maximum value for each input parameters is stated. The different input parameters can take any value that is at the corresponding range. The different values for the different input parameters will result in different dike geometries and input parameter sets. In Tables 3.7 and 3.8, the references for the values of the input parameters and the constant parameters related to the dike geometry are shown too. The same minimum and maximum values for the input parameters that are the same for the Simple and the Complex model are used, which can be observed in Table 3.8.

Parameter	Symbol	Number in Figure 3.5	Value	Unit	References
Outer slope*	OS	Brown 1	1:3	-	current dike reinforcement projects
Ditch slope	DS	Brown 5	1:3	-	[56]
Width bottom ditch	WBD	Grey 7	0.5	m	[11]
Inner berm slope	IBS	Brown 3	1:20	-	[48], expert knowledge
Thickness clay cover**	TCC	-	1.5	m	[63] [54]

Table 3.7: The constant input parameters to train, validate, and test the surrogate model for the Complex model.

*Berms will not be added to the outer slope and the outer slope will still be constant for the Complex model, since the main focus of this research is on developing a slip plane on the inner side of a dike (inner slope).

** The clay cover is placed over the entire dike core with the same thickness. In reality, when a sand dike has a clay cover, the clay cover is only placed on the outer slope of the dike or placed over the entire dike core but with a decreasing thickness.

Parameter	Symbol	Number in Figure 3.5	Min. value	Max. value	Unit	References
Crest width	B	Grey 2	3	10	m	[57] [40]
Inner slope	IS	Brown 2 and Brown 4	1:2	1:4	-	[40] [46], current dike reinforcement projects, soil type
Dike height	H	Pink 1	3.5	6	m	current dike reinforcement projects
Water height	WH	Dark blue 1	3	5.5	m	current dike reinforcement projects
Thickness cohesive layer	TCL	Pink 3	0	8	m	[13]
Inner berm	IB	Grey 4	0	15	m	[26]
Distance ditch-dike	DDD	Grey 6	0	5	m	[66]
Depth ditch***	DD	Pink 2	0.5	1.25	m	[11] [65]

Table 3.8: The minimum and maximum value for the input parameters to train, validate, and test the surrogate model for the Complex model.

***Since the ditch slope and the width of the bottom of the ditch are constant, only the depth or the total width of the ditch can vary. The maximum width of ditches next to a dike is 8 m [11], which leads to a maximum ditch depth of 1.25 m.

3.3.2 Soil parameters

In section 3.2.2, the soil parameters are stated for the SHANSEP and Mohr Coulomb Advanced model for the Simple model. For the Complex model, the same soil types and soil parameters are used.

3.3.3 Phreatic level

For the complex model, dikes can have a clay core and a sand core with a clay cover. In addition to the different dike core, a ditch is also present in the Complex model. The phreatic line runs from point A until G, including points F1 and G1. The coordinates of these points can be observed in Tables 3.5 and 3.6.

The water level inside the ditch is kept constant at height of 0.4 m below ground level in order to always have some water inside the ditch for different dike geometries.

3.4 Training dataset

Data sampling aims at capturing as best as possible the variability of the original model to train a suitable surrogate model. Data sampling is used to select a subset of input parameter values from the input parameter range with a minimum and maximum value mentioned in previous sections of this chapter. In general, it is difficult for surrogate models to extrapolate, which means that the sampling method should aim to be space-filling and broader than the expected values to predict. Data sampling strategies aim to maximize the amount of information per number of samples [24].

The minimum and maximum values for the input parameters are derived from different sources, as can be observed in Tables 3.1, 3.7, and 3.8. In reality, all input parameters values do follow a statistical distribution. However, in this research, it is assumed that each data

point/input parameter in the dataset has an equal chance of being selected. Due to this assumption, the data points are sampled from a uniform distribution with the minimum and maximum value for each input parameter described in previous sections of this chapter.

Different smart sampling strategies exist in literature, with the simplest method being the Grid Sampling.

3.4.1 Grid Sampling (GS)

In Grid Sampling (GS), a multi-dimensional space is divided into a regular grid of equally spaced points along each dimension. This sampling strategy is pretty straightforward and easy to implement, but is computationally demanding if it has to capture variability in the data when the underlying function is not linear.

The relationship between the input parameters and the FoS is most likely not linear, which means that the grid sampling strategy will not gain enough information to train the surrogate model. Therefore it is necessary to go for more advanced strategies. In this research, two common types of sampling strategies are considered; Random Sampling and Latin Hypercube Sampling. These two will be elaborated in the following sections. In Figure 3.8, the difference between the sampling techniques can be observed.

3.4.2 Random Sampling (RS)

Random Sampling (RS) is a sampling technique in which an input variable is selected independently from the entire range. This technique does not guarantee that the taken samples are well-spaced, which can affect the accuracy and efficiency of the simulation. However, random sampling makes sure that the selection bias is reduced, since certain ranges in the data are not over represented or under represented.

3.4.3 Latin Hypercube Sampling (LHS)

Latin Hypercube Sampling (LHS) is a sampling method that ensures that the samples are both evenly distributed within each dimension and are also as independent as possible across dimensions. This sampling strategy divides the range of each input variable into equal intervals and selects one value from each interval. After doing this, the values are paired randomly. This way of sampling makes sure that the data points are more evenly distributed than Random sampling and is beneficial since less samples are needed in comparison with Random sampling to get an appropriate ML model. The ML models will be trained using both sampling methods in order to select the sampling method with the best performance for the data used in this research.

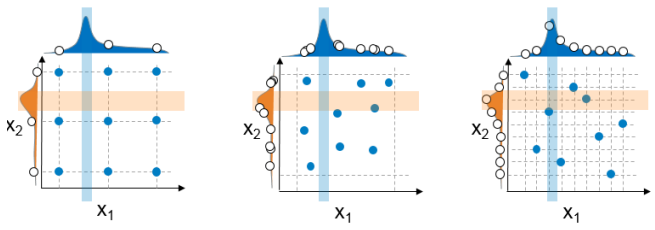


Figure 3.8: From left to right: Grid Sampling, Random Sampling, and Latin Hypercube Sampling [50].

3.4.4 The dataset

Certain requirements regarding the size of the dataset and the values of the input parameters have to be considered before sampling the input parameters.

The dataset containing about 10.000 input parameter sets with the corresponding FoS, calculated by D-Stability, should be used to create a surrogate model for the macro-stability of dikes. From 10.000 samples of the input parameters propagate in a Bishop model on a Monte Carlo simulation, it is expected that the FoS estimation error of its mean and standard deviation is around 1%. The error is estimated to be proportional to the square root of the number of samples [30]. 1% estimation error is often the limit, since a lower error will result in higher computational costs, because more samples are needed. This means that 10.000 samples are taken with Random Sampling and with Latin Hypercube Sampling for each input parameter following an uniform distribution.

The 10.000 input parameter sets are an initial guess for the amount of D-Stability calculations that need to be performed. If the input parameter space is not completely filled, after running the 10.000 D-Stability calculations, then more calculations need to be performed and more D-Stability calculations will be executed.

To create the dataset, different combinations of input parameters are collected by sampling to get the FoS. It can happen that certain combinations of input parameters contain values for the water height that are higher or equal to the crest height of the dike. The input parameter sets with water heights higher than the crest of the dike are removed from the input parameter dataset since the water acts as an additional load on top of the dike for these combinations of the input parameters and the schematization of the phreatic line is not properly schematized for these combinations.

In reality, dikes are not designed for water levels equal to the crest level. However, the input parameter combinations with these water levels help the surrogate model to approximate the FoS better for situations when the water level is very high with respect to the crest level of the dike and therefore these combinations have not been removed from the dataset.

3.5 Data sampling in Python

In the previous section, the data sampling methods are described. How these two sampling methods are implemented and how to get the different dike geometries from the input parameters is explained in this section.

For data sampling, two Python files are created; one for Random Sampling (RS) and one for Latin Hypercube Sampling (LHS) to keep the methods separate. For RS, the input parameters are sampled with the function random.uniform() from the random module and for LHS, the class LatinHypercube is used from module scipy.stats.qmc. After deriving the 10.000 input parameter sets with the specific sampling method, the dike geometries can be created. For the dike geometries, a function is defined that describes the dike geometry, which returns a numpy array containing the x-, y-, and z-coordinates of the characteristic points along the dike. Two different CSV-files should be created containing the characteristic points along the dike as input for D-Stability. One CSV-file contains the x-, y-, and z-coordinates of the characteristic points with the column names X1, Y1, Z1, X2, ... etc. and the other CSV-file contains also all coordinates, but with column names X_name characteristic point, Y_name characteristic point, Z_characteristic point, ... etc. until all characteristic points along the dike have been selected.

First, the characteristic points are stored in two separate data frames called surfacelines and characteristic_point and later both data frames are saved in two separate CSV-files. The sampled input parameters are saved in a similar way.

3.6 Conclusion

This chapter presents what the training dataset looks like and how it can be created. First, the dataset of the Simple model is described. The Simple model is created in order to see if it is possible to create a surrogate model based on D-Stability calculations with high model performance before moving on to the more complex dike geometry. The values for the input parameters come mainly from current dike reinforcement projects in order to eventually apply the surrogate model in current projects. The values for the soil parameters are derived from Table 2B of the NEN 9997-1 and the macro-stability manual of WBI2017. If soil investigation is not performed, this approach is often done to have an idea about the soil conditions.

Some input parameters have more influence on the FoS than the others. This will result from the outcome of the Simple and the Complex model. Table 3.1 and Table 3.8 just gives an overview of the initial ranges of input parameters. The Simple and the Complex model are optimized by changing the input parameter ranges such that the range of input parameters, that influences the FoS the most, and the right values for the FoS are chosen. Optimizing the Complex model is done in a similar way as optimizing the Simple model. That is why the same input parameter ranges as the initial ranges for the Simple model are observed in Table 3.8. The sensitivity for certain input parameters will be elaborated in detail in chapter 5.

After describing the 'ingredients' of the dataset, the techniques for generating the data and certain requirements related to the dataset are elaborated in section 3.4 and 3.5. Three different sampling techniques are explained. For the Simple model, both methods are used. For the Complex model, the best sampling strategy, derived from the Simple model, is applied. e

Chapter 4 Methodology

4.1 Introduction

The methodology is about discussing the strategy to derive an appropriate surrogate model for the macro-stability of dikes. Step by step, the different processes are explained until a final ML model is created. In between different sections of this chapter, the python implementation of the previous sections is explained. A clear overview of the strategy of deriving an appropriate surrogate model for macro-stability of dikes is shown in Figure 4.1.

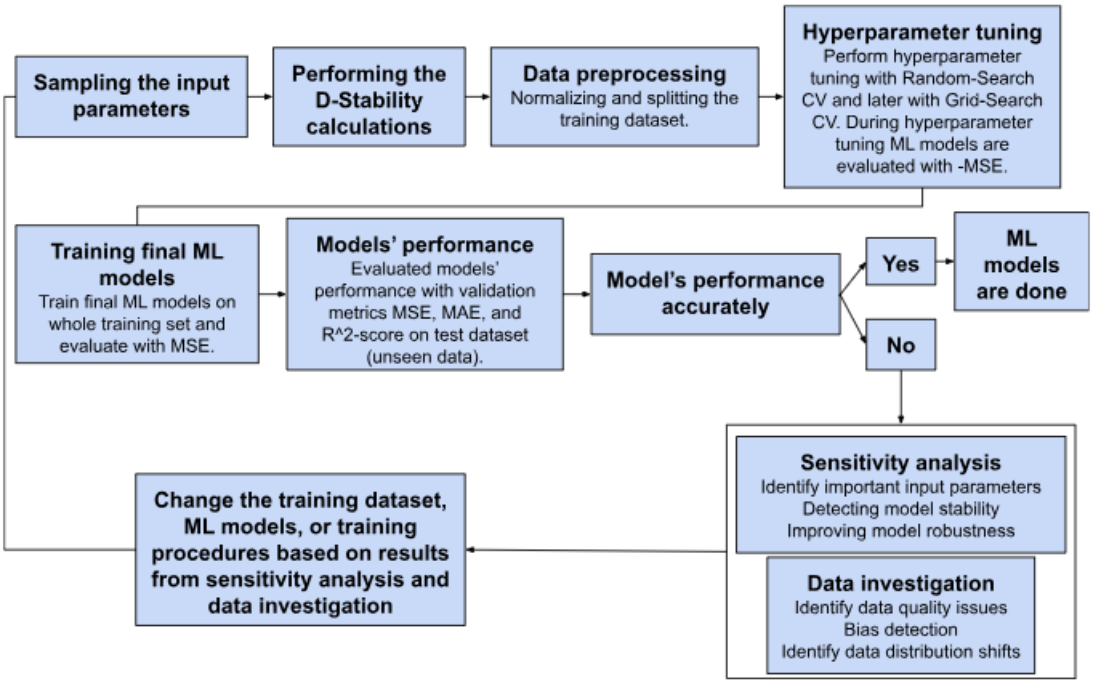


Figure 4.1: Overview of the strategy to derive an appropriate surrogate model for the macro-stability of dikes.

4.2 HWL datamodel

Before training the ML model, the dataset should be generated. The dataset is generated by sampling the different input parameters that create different dike cross-sections and implement these cross-sections in D-Stability one by one to get the corresponding FoS. Doing this one by one to get 10.000 D-Stability calculations will take a lot of time. That is why Witteveen+Bos (W+B)

already created a model that automates this process, the High Water protection and Land development model, in Dutch the Hoogwaterbescherming and Landinrichting (HWL) model, by combining Python with D-stability. Each dike section is stored in a database after the python tooling checks if all the properties of the different dike geometries are correctly implemented.

The automated process starts with defining the cross section of the dike, containing the dike geometry and the underground schematization. The dike geometry is defined by a Linestring containing the x- and y-coordinates of the characteristic points along the dike. These characteristic points are shown in Figure 3.1. The subsoil is schematized as a 1D borehole. This 1D borehole contains all the soil properties and the depth of each soil layer. The information of the dike geometry and the subsoil will be sent to HWL datamodel, which will store and prepare the data for the python underground schematization toolkit, in Dutch python ondergrond schematisatie toolkit (post). After storing the information in HWL datamodel, post will create a 2D schematization of the dike cross section with the data from the dike geometry and subsoil schematization. Post has created an embankment, but the water is still missing. The phreatic line is created as an additional part of the HWL datamodel. The created dike with the phreatic line (waternet) will be sent to D-Stability to calculate the FoS. After the D-Stability calculation, the input parameter set with the corresponding FoS will be stored in a tabular dataset encoded into a pandas' dataframe [36]. The values for the input parameters will be changed and the same process follows for another dike cross-section. A lay-out of the automated process containing the HWL datamodel is depicted in Figure 4.2. After doing the 10.000 D-Stability calculations, the different input parameter sets with the corresponding FoS will be stored in a CSV-file for easy access to train the machine learning (ML) model.

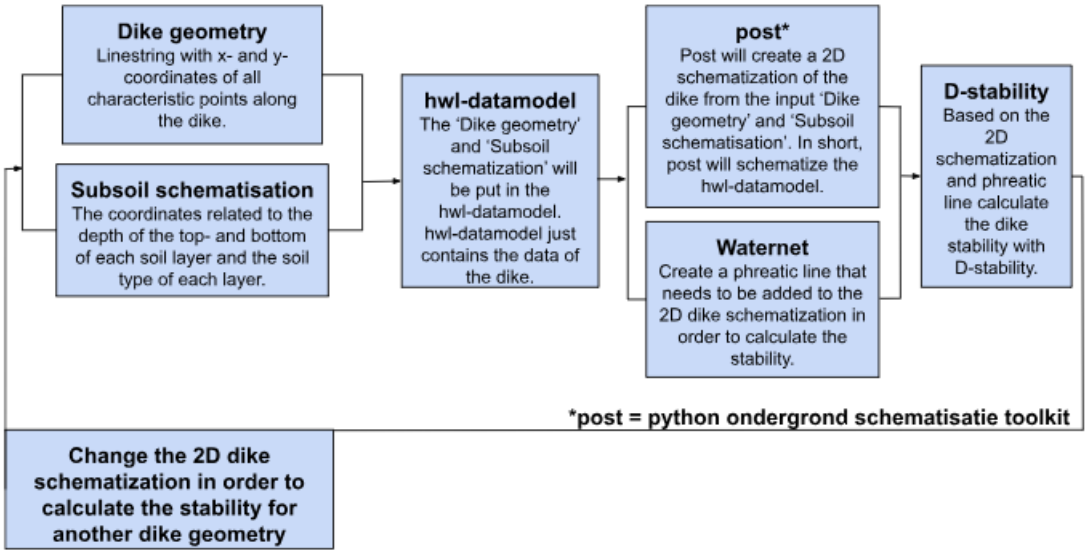


Figure 4.2: Explanation of the HWL datamodel.

4.3 Python set-up for FoS

In the previous section, HWL datamodel is explained that is used in this research to get the 10.000 D-Stability calculations quickly. In this section, the implementation of HWL datamodel in Python is elaborated. The soil data implementation in Python is also explained in this section.

4.3.1 Soil data

Some requirements related to the soil data exist in order to use this data for the D-Stability calculations. In sections 3.2.2. and 3.3.2, the characteristics of clay and sand are elaborated and these characteristics should be named in a way that the soil data is imported correctly. In a separate Python file, the soil data is stored in a pandas data frame with the correct column names according to the classes HWL datamodel. This data frame is saved as an CSV-file.

4.3.2 HWL datamodel

The steps that should be taken before running the 10.000 D-Stability calculations are as follows:

1. The four CSV-files, surfacelines, characteristic_point, soil data, and input parameters, are imported in the Python file.
2. After getting the data, 10.000 1D soil profiles are created with the soil data and the values for the different dike heights and thicknesses of the cohesive layer, from the input parameter file, with class SoilProfile1D from HWL datamodel.
3. 10.000 CSV-files will be created for each surfaceline of the 10.000 dikes from CSV-file surfacelines and similar for characteristic_point CSV-file.
4. A 2D profile will be created for each dike geometry from the created CSV-files in step 3 with class Profile2D from HWL datamodel.
5. For each dike geometry, a block of soil will be created from the 1D soil profiles in step 1 and the minimum and maximum x-coordinate from the 2D profile in step 4 with class _create_block from post.
6. After deriving a block of soil, the dike geometry from the 2D profile is cut from the block of soil with _cut_profile from post.
7. The schematization of the dike is finished using the class SubsoilSchematisation from HWL datamodel with the input from step 6.
8. Right now, the base schematization is established and is ready to be exported to D-Stability. First, an empty DStability model should be created. The soils and the schematisation will be exported to the model using convert_soils() and convert_subsoil_schematisation() from the wb_geolib package, which is a wrapper around the Deltares GEOLIB package. This package supplies adapters to convert data between HWL datamodel and Geolib.
9. After converting the soils and schematization of the dike, the waternet should be created. The waternet is not part of the subsoil schematization of the dike profile, as can be seen in Figure 4.2. A class will be defined in the Python file to create the waternet. The phreatic line that is schematized in section 3.2.3 is part of this class. The waternet is exported to the D-Stability model using convert_waternet from wb_geolib.
10. The uniform traffic load on top of the crest of the dike and the state points in the different soil layers are added to the D-Stability model with classes UniformLoad and StatePoint from Geolib.
11. After performing step 1 until 10, the D-Stability model is done. The only thing that needs to be done is defining the Analysis method (DStabilityBishopBruteForceAnalysisMethod

for the Simple model / DStabilityUpliftVanParticleSwarmAnalysisMethod for the Complex model) from Geolib, set it to the model, and performing the D-Stability calculation. To run the model, the model needs to be serialized (saved) with the function serialize() first and then the function execute() is called to run the D-Stability calculation. The output file should be defined in order to save the D-Stability calculation.

12. After running the D-Stability calculations, the results from the calculations are saved in an CSV-file together with the corresponding sampled input parameters. The results from the D-Stability calculations are the FoS and the midpoint and the radius of the slip circle for the Simple model with analysis method DStabilityBishopBruteForceAnalysisMethod, is Limit Equilibrium Method (LEM) Bishop's method. The results from the D-Stability calculations are the FoS, the midpoint and the radius of the active slip circle, and the midpoint and radius of the passive slip circle for the Complex model with analysis method DStabilityUpliftVanParticleSwarmAnalysisMethod, is LEM Uplift-Van method. The FoS is needed as output parameter to train the surrogate model and the characteristics of the slip circle are needed for the data investigation in order to improve the surrogate model for the simple and complex dike geometries.

4.4 Data preprocessing

The ML models can not be trained directly on the data obtained from the D-stability calculations with the different input parameter sets. Some input parameter sets will result in shallow slip circles. These need to be removed from the total dataset since these shallow slip circles will not result in the dike failure due to the macro-stability failure mechanism. After removing these calculations, the dataset will be normalized and split into a training, validation, and test set.

4.4.1 Normalization of data

Some ML models are sensitive to feature scaling. Therefore it is necessary to scale or standardize the data. In addition to that, a non-normalized dataset often leads to numerically more challenging optimization problem. Since the input and output parameters have different scales and differ in orders of magnitude, the best method to scale the data is MinMaxScaler and is an alternative to mean-variance scaling. The MinMaxScaler scales the data in such a way that the minimum value in the dataset will be set to zero and the maximum to one.

4.4.2 Splitting the dataset

Splitting the dataset is crucial to assess the models' performance. The dataset containing the input parameters with corresponding output parameters should be split into a training, validation, and test dataset.

The training dataset is applied to train the ML model. The ML model tries to find the patterns captured inside the data and directly improves its parameters. The training dataset is often 60-80 % of the total dataset, and even 90% if the dataset is relatively large [41].

The validation dataset is used to calibrate the hyper parameters of the ML model and to provide an unbiased evaluation of the model fit on the training dataset while tuning the hyper parameters. The validation dataset is often 10-20 % of the total dataset.

The test dataset is used to provide an unbiased evaluation of the final model fit on the training dataset. The test dataset is an independent dataset to evaluate the model's performance after training and tuning the hyper parameters. The test dataset is often 10-20 % of the

total dataset.
In this research, the dataset is split using the function train_test_split from scikit-learn [55]. A common split is that 80% of the total dataset is for training and validating the ML model and 20% is for testing the ML model [49]. These percentages are used in this research. Since the dataset is relatively large, the training and testing dataset have the same statistical properties. The size of the validation dataset varies due to the value for CV, which is explained in section 4.5.

4.5 Hyperparameters

Before training the final ML models, the hyper parameters for each ML model will be tuned first. Hyper parameters are settings of the ML model, which will not be changed during training of the ML model. The hyper parameters associated with each type of machine learning model have a crucial impact on determining the bias and variance.

Hyper parameter tuning is the process to optimize the hyper parameters of the ML algorithms. The validation set, described in section 4.4.2, is used to evaluate the ML algorithms during hyper parameter tuning. During hyper parameter tuning, the ML models are evaluated based on negative mean squared error, which is a commonly used scoring parameter [17] [35]. Two common used hyper parameter tuning techniques are Grid-Search Cross-Validation and Random-Search Cross-Validation. The ML models are evaluated based on the negative mean squared error since the classes in scikit-learn for Grid-Search Cross-Validation and Random-Search Cross-Validation are designed to maximize the scoring metric. Since MSE needs to be minimized, using negative mean squared error allows the optimizer to effectively minimize MSE by maximizing the negative MSE. Grid-Search Cross-Validation and Random-Search Cross-Validation will be elaborated in the following section and afterwards the hyper parameter tuning process for the different ML algorithms with their corresponding hyper parameters.

4.5.1 Hyper parameter tuning techniques

In this section, the two considered hyper parameter tuning techniques are elaborated. The hyper parameter tuning process will be explained in the next section. However, one parameter is important to mention in this section and that is the cross-validation (CV).

Both hyper parameter techniques evaluate the ML model's performance using CV. CV is a technique to create better fitting models by splitting the training dataset into multiple random subsets, called folds, holding one group out as validation set and the remaining groups serve as training set. If cv equals 5, then 5 folds are created and the model is trained on 4 folds and evaluated on 1 fold. This process is repeated five times, with each fold serving as validation set exactly once. The model's performance is evaluated based on the validation fold and after 5 iterations, the average score of all folds for a certain set of hyper parameters is calculated to get a robust estimate of the model's performance. This iterative process can be observed in Figure 4.3. The default value for cv equals 5 in both techniques. However, a debate is going on which value and strategy is the best for cv and the value 10 for cv for a small total dataset seems to be better [42]. Tuning the hyper parameters for a cv of 10 takes longer, but is doable for a dataset consisting of 10.000 data points. That is why a cv value of 5 and 10 are both used in this research.

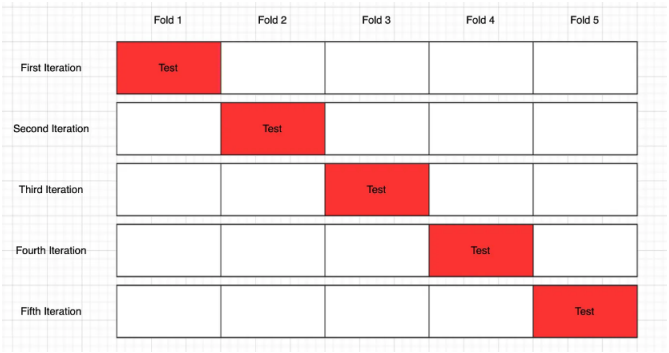


Figure 4.3: Cross validation strategy with cv=5 [6].

Grid Search Cross-Validation

Grid Search Cross-Validation is a technique for finding the optimal hyper parameter values from a given set of parameters in a grid to enhance the model's performance. This technique works in the same way as grid sampling, which is explained in section 3.4. Grid Search Cross-Validation explores each possible combination of hyper parameters that is provided in the grid that is defined by the researcher. This method evaluates the model's performance by testing it on various sections of the dataset. After trying out all possible hyper parameter combinations, Grid Search Cross-Validation presents the best combination of hyper parameters with the corresponding score. This method performs an exhaustive search to get the right hyper parameters. In Figure 4.4, the grid lay-out of Grid Search Cross-Validation can be observed. This technique is extremely costly in computing power and time when the number of hyper parameters and the hyper parameter grid is large.

Random Search Cross-Validation

Random Search Cross-Validation is a technique for finding the optimal hyper parameter values by selecting random combinations of hyper parameters from given hyper parameter grids to enhance the model's performance. The default value for sampling the hyper parameter combinations is 10 and this value is used in this research. This method also evaluates the model's performance by testing it on various sections of the dataset, just like Grid Search Cross-Validation. After trying out the 10 random sampled hyper parameter combinations, Random Search Cross-Validation presents the best combination of hyper parameters out of the 10 considered combinations with the corresponding score. This method is less costly in computer power and time since not all possible hyper parameter combinations are assessed. In Figure 4.4, the grid lay-out of Random Search Cross-Validation can be observed.

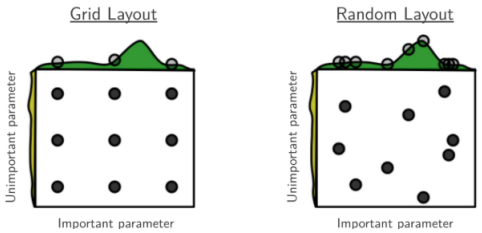


Figure 4.4: Grid Search Cross-Validation vs Random Search Cross-Validation [70].

4.5.2 Hyper parameter tuning

Before tuning the hyper parameters for each ML algorithm, the hyper parameters and the corresponding ranges must be chosen first. The hyper parameters and their ranges have been chosen based on literature. Some of the used literature is cited in Tables 4.1 to 4.4. The most common hyper parameters with the ranges are used in this research.

After defining the hyper parameters and the range, the base ML model will be created with the default hyper parameters that belongs to that model. When the ML models are created, the Random Search Cross-Validation object is set up using the defined hyper parameter grids, which are observed in Tables 4.1 until 4.4 for the corresponding ML model, and a cv value of 5 and 10 is used. For Random Search Cross-Validation, the amount of iterations is the default value, 10. After setting up Random Search Cross-Validation object, the Random Search Cross-Validation is fitted to the data, iterates over combinations of hyper parameters and assesses the hyper parameter combination by the scoring parameter, negative mean squared error. Eventually, the Random Search Cross-Validation returns a score of the model's performance for a cv value of 5 and 10 and the best hyper parameters. Based on this score and the best hyper parameters, the hyper parameter grid will be changed and a decision is made which value for cv is used for Grid Search Cross-Validation.

The cv value with the best score is selected for the second search for the best hyper parameters with Grid Search Cross-Validation. Grid Search Cross-Validation is selected for the second search since this method considers all possible hyper parameter combinations. For every hyper parameter, 3 values are selected, which are around the best hyper parameter value that is derived from the Random Search Cross-Validation. Finding the best hyper parameters is done in the same way as for Random Search Cross-Validation, but the number of iterations is higher since all possible hyper parameter combinations are considered. The hyper parameter combination that is derived from Grid Search Cross-Validation is the one that is used for training the final ML model.

Hyper parameter	Definition	Range of values for Random Search Cross-Validation
n_estimators	number of trees in the forest	(50, 1000, stepsize=50)
max_features	considering number of features when looking for best split	['log2', 'sqrt', 'None']
min_samples_split	min. number of samples required to split an internal node	(2, 10, stepsize=1)
min_samples_leaf	min. number of samples required at a leaf node	(1, 5, stepsize=1)
max_depth	max. depth of a tree	(10, 100, stepsize=10)

Table 4.1: The hyper parameters that will be tuned for RFR based on literature [14] [27] [5].

Hyper parameter	Definition	Range of values for Random Search Cross-Validation
n_estimators	number of trees in the forest	(50, 250, stepsize=50)
max_depth	max. depth of a tree	(3, 10, stepsize=2)
η (learning_rate)	contribution of each tree	[0.01, 0.02, 0.05, 0.1, 0.2, 0.3]
subsample	fraction of training data used for training each tree	(0.5, 1, stepsize=0.1)
colsample_bytree	fraction of features used for training each tree	(0.5, 1, stepsize=0.1)
γ	min. reduction loss for creating new split in leaf node	(0.1, 0.2, stepsize=0.01)
α	L1 regularization term on weights	(0, 1, stepsize=0.1)
λ	L2 regularization term on weights	(0, 1, stepsize=0.1)
min_child_weight	min. required weight to create a new node in tree	(1, 10, stepsize=2)

Table 4.2: The hyper parameters that will be tuned for XGB based on literature [20] [4] [2].

Hyper parameter	Definition	Range of values for Random Search Cross-Validation
C	regularization parameter	[0.1, 1, 10]
γ	kernel coefficient	[0.01, 0.1, 1]
kernel	kernel type	['linear', 'rbf', 'poly']
ε	specifies the ε tube; trade-off model complexity and error tolerance	[0.01, 0.1, 0.2, 0.3, 0.5, 0.7]

Table 4.3: The hyper parameters that will be tuned for SVR based on literature [33] [39] [51].

Hyper parameter	Definition	Range of values for Random Search Cross-Validation
hidden_layer_sizes	number of neurons in the hidden layer(s)	[(3), (5), (10), (20), (3, 3), (5, 5), (10, 10), (20, 20), (3, 3, 3), (5, 5, 5), (10, 10, 10), (20, 20, 20)]
batch_size	size of the mini batches for optimizer	[100]
learning_rate_init	controls the step size in updating the weights	[0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1]
optimizer	solver for weight optimization	['Adam', 'SGD']
activation_function	activation function for the hidden layer	['ReLU', 'sigmoid', 'tanh']
alpha	strength of the L2 regularization	[0.00001, 0.0001, 0.001, 0.01]

Table 4.4: The hyper parameters that will be tuned for FNN based on literature [3] [37].

4.6 Loss function

After hyper parameter tuning, the final ML models will be trained. Loss functions are used to evaluate the model's performance during training the model and help to improve the model's performance. For each ML algorithm in this research, the basis loss function is the Mean Squared Error (MSE), except for SVR since this ML algorithm uses the ε - Insensitive Loss Function. For SVR, the goal is to find a regression function that has a prediction error no larger than a specified margin ε for as many training samples as possible.

MSE measures the average squared difference between the predicted target values and actual values and can be observed in equation 4.1. This loss function is modified for certain ML algorithms in this research if the chosen hyper parameters for the considered ML algorithm are related to regularization terms. These regularization terms help to prevent overfitting.

$$MSE = \frac{1}{n} \cdot \sum (y_i - \hat{y}_i)^2 \tag{4.1}$$

where:

- n is the number of data points;
- y_i is the actual target value;
- $\hat{y}_i$ is the predicted target value;

4.7 Validation metrics

After training the ML models with the training dataset, the model's performance should be evaluated with the test dataset in order to see how well the ML models perform to new, unseen data. The model's performance will be assessed using validation metrics. Validation metrics measures the model's performance with the test dataset. In this research, three different validation metrics are considered for testing the ML models on unseen data; MSE, Mean Absolute Error (MAE), and Coefficient of Determination (R^2). MAE measures the average of the absolute differences between the predicted target values and actual values and can be observed in equation 4.2. R^2 gives an indication of the quality of fit of the ML model and can be observed in equation 4.3. The value of R-squared ranges from 0 to 1, the higher the value, the better the model predicts the target value. How MSE works, is already explained in section 4.6.

$$MAE = \frac{1}{n} \cdot \sum \mid y_i - \hat{y}_i \mid \tag{4.2}$$

where:

- n is the number of data points;
- y_i is the actual target value;
- $\hat{y}_i$ is the predicted target value;

$$R^2 = 1 - \frac{SSE}{SST} \tag{4.3}$$

where:

$SSE = \sum(y_i - \hat{y}_i)^2$; y_i is the actual target value;
$\hat{y}_i$ is the predicted target value;
$SST = \sum(y_i - \bar{y})^2$; y_i is the actual target value;
$\bar{y}$ is the mean of the actual values;

4.8 Python set-up for Machine learning

In the previous sections of this chapter and the chapters before, the different steps are elaborated to develop ML models and to train them in a way that the target values are predicted well. How these steps are performed in Python that will be explained in this section.

First, the CSV-file with the sampled input parameter together with the results from the D-Stability calculations is imported in the Python file. The data is normalized using an own defined MinMaxNormalizer class and split into training and test datasets using train_test_split() [55]. After data normalization and splitting the dataset, hyper parameter tuning is performed. The ML algorithms are trained with different sets of hyper parameters in order to see which combination of values for the different hyper parameters will result in the best ML model. First, hyper parameter tuning is done with RandomizedSearchCV and later with GridSearchCV from module sklearn.model_selection. After training the different ML models, the model's performance is evaluated with the test dataset and assessed using the different validation metrics from section 4.8. The best model is saved with function joblib.dump().

4.9 Conclusion

This chapter shows how to perform the 10.000 D-Stability calculations and the steps that need to be taken to achieve a ML model that performs well. First, HWL datamodel is explained. HWL datamodel is established by W+B and can be used to store, validate, serialize and communicate data. HWL datamodel makes sure that performing the D-Stability calculations with it is quicker than doing it by hand. How HWL datamodel is implemented in Python, is elaborated in the section 4.3. The data is saved after performing the D-Stability calculations and should be normalized and split into a training and test dataset. Normalizing and splitting the data is better for the ML model's performance. This is explained in section 4.4. The same yields for hyper parameter tuning, which is elaborated in section 4.5. Tuning the hyper parameters will increase the ML performance. Many hyper parameters exist for each ML model, but the most used ones are selected to improve the ML model's performance. After hyper parameter tuning with Random Search Cross-Validation and Grid Search Cross-Validation the final models can be trained. The ML models will be evaluated during training with a loss function, which is explained in section 4.6. The basis for the loss function is the Mean Squared Error, except for the SVR model. The loss function for the SVR model is the ε - Insensitive Loss Function. The ML models XGB, SVR, and FNN adjust the loss function by adding regularization terms, which are hyper parameters for each ML algorithm, to prevent overfitting. After training the ML models, the model's performance can be evaluated with the three different validation metrics that are discussed in section 4.7. How the derived data from the D-Stability calculations results in different ML models in Python, will be discussed in the last section of this chapter. In the end, the best ML model will be used in the Multi Criteria Analysis (MCA) to optimize dike designs.

Chapter 5 Surrogate model results

5.1 Introduction

In chapter 3 and 4, the set-up for the database and the methodology for creating a surrogate model is explained. In this chapter, the results regarding the database and the model's performance are elaborated. First, the Simple model will be explained with all the needed steps to get the best macro-stability model for simple dike geometries. After getting an appropriate ML model for the simple dike geometries, a surrogate model will be created for the complex dike geometries. The best ML model for the complex dike geometries will be used in the Multi Criteria Analysis (MCA) in part 2 of this research.

5.2 'Simple' model

After running the 10.000 D-Stability calculations, the training dataset is created to train the ML algorithms. This dataset is normalized and splitted into a training and test dataset. Before training the ML algorithms, two steps must be taken first. First, a quick check is performed whether the size of the training set is good. When the different input parameters are plotted against each other, then the whole space is covered by data points. This applies to both the train and test dataset, which means that the amount of samples is enough to train the ML algorithms.

Secondly, the training dataset should be cleaned by removing outliers. Part of data preprocessing is removing the D-Stability calculations which has shallow slip circles as result and therefore lead to outliers in the training dataset. After checking the training dataset, the ML algorithms can be trained.

In section 5.2.1, the model's performance for each ML algorithm is shown and in section 5.2.2 the correlations in the training data can be observed. In section 5.2.3, the database is investigated and important changes in the parameters are elaborated in order to increase the model's performance for the Simple model.

5.2.1 First results training ML models

After normalizing and splitting the database into a training and test dataset, hyper parameter tuning and training the final ML models can take place. First, hyper parameter tuning will take place to find the best values for the hyper parameters with Random Search and Grid Search Cross-Validation and after that the final ML models are trained on the whole training dataset and tested with the test dataset. The final ML models are evaluated with the different validation metrics and the results after the final training are stated in Tables 5.1 and 5.2. The differences in results for the different ML models with the different sampling methods are small, with a maximum deviation of $0.5 \cdot 10^{-3}$ for MSE, $0.3 \cdot 10^{-2}$ for MAE, and 0.03 for R^2-score. At first

glance, it looks like the sampling method and the type of ML algorithm does not matter for the performance of the ML model. The values for the MSE and MAE are small and the values for the R-squared are high after the final training, but still the values can be improved since the dike geometries are simple. In the next sections, the training data derived from the D-Stability calculations will be investigated, the input parameters will be adjusted and some small changes are made in order to improve the model's performance.

ML models	MSE	MAE	R^2-score
RFR	$3.0 \cdot 10^{-3}$	$4.8 \cdot 10^{-2}$	0.83
XGB	$3.1 \cdot 10^{-3}$	$4.9 \cdot 10^{-2}$	0.82
SVR	$\mathbf{2.7 \cdot 10^{-3}}$	$\mathbf{4.7 \cdot 10^{-2}}$	**0.84**
FNN	$\mathbf{2.7 \cdot 10^{-3}}$	$\mathbf{4.7 \cdot 10^{-2}}$	**0.84**

Table 5.1: Results of the validation metrics for Random Sampling for the different ML algorithms after the last training on the whole training dataset.

ML models	MSE	MAE	R^2-score
RFR	$3.2 \cdot 10^{-3}$	$\mathbf{5.0 \cdot 10^{-2}}$	0.81
XGB	$3.2 \cdot 10^{-3}$	$\mathbf{5.0 \cdot 10^{-2}}$	0.81
SVR	$\mathbf{3.0 \cdot 10^{-3}}$	$\mathbf{5.0 \cdot 10^{-2}}$	**0.82**
FNN	$3.1 \cdot 10^{-3}$	$\mathbf{5.0 \cdot 10^{-2}}$	**0.82**

Table 5.2: Results of the validation metrics for LHS for the different ML algorithms after the last training on the whole training dataset.

5.2.2 Sensitivity analysis

After training the first ML models, sensitivity analysis and data investigation have to take place in order to improve the ML models. From section 5.2.1, it could be concluded that, at first glance, the sampling method and type of ML algorithm do not have a major influence on the performance of the ML model so far. That is why the training dataset is investigated in order to improve the ML models.

Before creating the first ML models, the training dataset was already checked on outliers and whether the size was good. A logical step, after data preprocessing and creating the first ML models, is creating a pairplot. A pairplot is a type of data visualization that shows pairwise relationships in the dataset. This plot is typically used for exploring the relationships between different variables (input parameters) in a multivariate dataset. The off-diagonal scatter plots show the relationships between two input parameters, and the diagonal histograms show the distribution of individual input parameters. The different input parameters were sampled following an uniform distribution and therefore the histograms along the diagonals should be flat. However, the histograms along the diagonal for the dike height and the outer water level are not flat. All the input parameters are sampled following an uniform distribution, but during sampling, the samples were removed for which the water height was higher than the dike height resulting in a histogram which is not flat. This way of correcting the water height is not right since the histogram of the dike height should be flat. That is why for the improved Simple model the water height should be corrected by having the minimum value between the sampled dike height and the sampled water height. This means that, during sampling, when the value

for one water height sample is higher than the value for one dike height sample that the value for the water height sample is corrected to the value for the dike height sample.

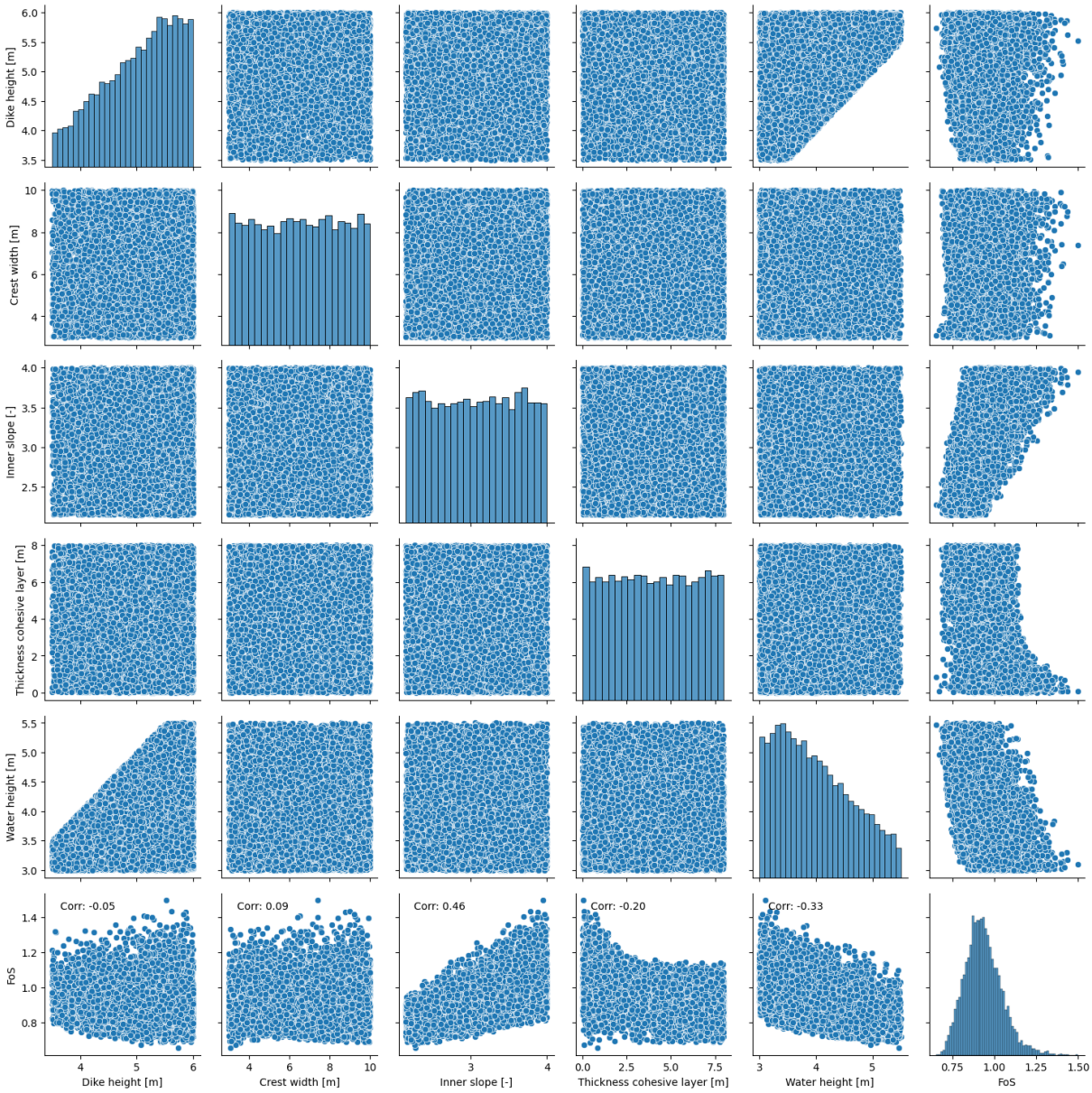


Figure 5.1: Pairplots of the different input parameters and the FoS for the simple dike geometries sampled with Random Sampling.

In the pairplots, the correlation between the different input parameters and the target value (FoS) can be computed. This is shown in the last row of Figure 5.1. The correlations between the different input parameters and the FoS are computed with Kendall's correlation. Kendall's correlation is a non-parametric test procedure, which means that Kendall's tau does not assume that the data follows any specific distribution [10]. The considered relations for which Kendall's tau should be calculated are non-linear and that is why Kendall's correlation is used. The correlations between the different input parameters and the FoS for the different sampling methods are of the same size, which can be observed in Table 5.3. That is why the pairplot of

the different input parameters and the FoS for the simple dike geometries sampled with RS is only shown in Figure 5.1.

When the different input parameters are plotted against each other, then the whole space is covered by data points. This can be observed in Figure 5.1. This is also the case when the dataset is splitted into training and test dataset, as mentioned at the beginning of this section.

Parameter	Correlation w.r.t. FoS for Random Sampling	Correlation w.r.t. FoS for Latin Hypercube Sampling
Dike height	-0.05	-0.06
Crest width	0.09	0.08
Inner slope	0.46	0.45
Thickness of the cohesive layer	-0.20	-0.19
Water height*	-0.33	-0.33

Table 5.3: Correlation of the input parameters of the simple surrogate model w.r.t the FoS.

*This correlation is calculated between the outer water level and the FoS and not the difference between the water level in the foreland and the water level in the hinterland and the FoS, while the last one, the head difference, is the main driver. However, the correlation is taken between the outer water level and the FoS since the outer water level is, just like the other input parameters, a sampled parameter, and the value for the correlation remains the same, when the water height is replaced by the head difference.

The input parameters inner slope and the water height are strongly correlated with the FoS. If the inner slope is very steep, the slope is less stable, and the FoS decreases. If the outer water level is high, the phreatic line inside the dike is high, which means that the effective stresses inside the dike decreases, and the FoS decreases.

The input parameters dike height and crest width are less correlated with the FoS. Widening the crest of the dike has a small positive influence on the FoS. A wider crest helps to improve the stability of the inner slope by spreading the loads, but this influence on the FoS is not as significant as other input parameters. The dike height is an independent input parameter, but is in reality connected to the water height and the inner slope. The dike height is negatively correlated with the FoS, although the dike height is not strongly correlated with the FoS. When the dike height increases, the stresses on both the dike material and the soil underneath the dike increase, leading to increase in shear stress along potential slip surfaces within the dike or the soil layers underneath the dike and reducing the stability of the inner slope of the dike. When the difference between the dike height and water level is plotted against the FoS, then the difference is positively correlated with the FoS. However, since the dike height is an independent input parameter, this input parameter itself does not influence the FoS that much.

Looking at Table 5.3, when the current dike geometry is not stable, the most effective way to make the dike more stable is to increase the inner slope of the dike.

Polynomial functions input parameters vs FoS

In order to see what the exact impact of one input parameter is on the FoS, one input parameter will be varied and the others will be constant and the FoS is calculated with the best model coming from Table 5.1 or 5.2. The model with the best values for MSE, MAE, and R^2-score is the Feedforward Neural Network from sampling with RS. This model will be used in order to

see the relation between the specific input parameter and the FoS. The values for the constant input parameters is the mean value of the specific input parameter range derived from Table 3.1.

The mean value for the different input parameters are:

- Crest width: 6.5 m
- Dike height: 4.75 m
- Inner slope: 1:3.1
- Thickness cohesive layer: 4 m
- Water level: 4.25 m

In Figure 5.2, the relations between the different input parameters and the FoS is shown. The input parameters from Figure 5.2 are sampled from RS, but similar relations between the input parameters and the FoS can be found for LHS. The polynomial function for each plot is stated below this figure.

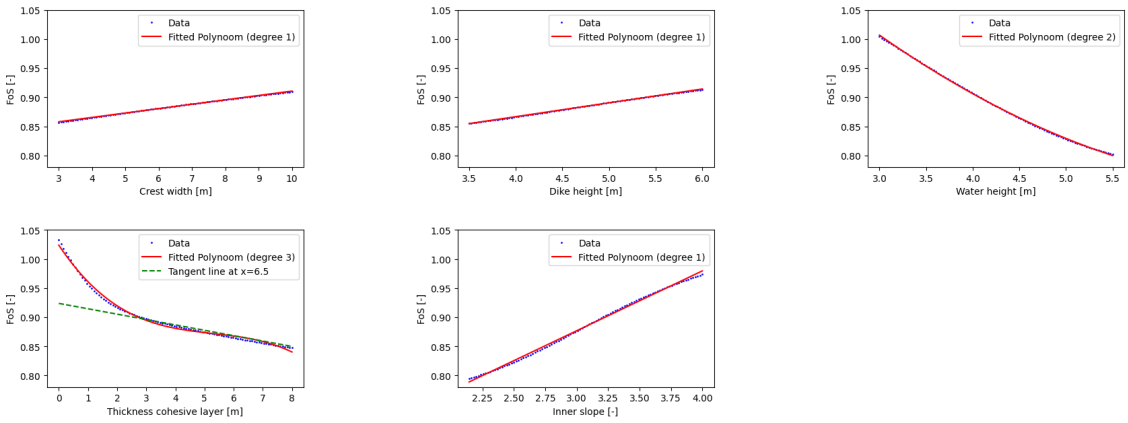


Figure 5.2: Relations between the different input parameters and the FoS sampled with Random sampling.

The polynomial function for each plot is, from left to right and top to bottom:

- Crest width: $0.0075 \cdot x + 0.8353$
- Dike height: $0.2380 \cdot x + 0.7714$
- Water level: $0.0119 \cdot x^2 - 0.1840 \cdot x + 1.4520$
- Thickness cohesive layer: $-0.0008 \cdot x^3 + 0.0131 \cdot x^2 - 0.0750 \cdot x + 1.0240$
- Inner slope: $0.1028 \cdot x + 0.5684$

In the plot thickness cohesive layer against the FoS, a tangent line is plotted near x = 6.5 m in order to see if a linear relationship exists between the thickness of the cohesive layer and the FoS for greater thicknesses.

When the plot in Figure 5.2 is compared with the correlations in Table 5.3, most relations that can be observed in Figure 5.2 are in line with what is expected.

In Table 5.3, the crest width of the dike is weak positive correlated with the FoS. The polynomial function for the crest width is also a gently increasing linear function, which is in line with the computed correlation between the two.

The dike height is weak negative correlated with the FoS in Table 5.3, but the polynomial function is a gently increasing linear function. It would be expected that the polynomial function is a gently decreasing linear function. When changing the values for the constant input parameters or changing the ML model, the polynomial function still remains a gently increasing linear function. It seems that the ML model can not capture the relation between the dike height and the FoS that well. As mentioned at the start of this section, the distribution of the dike height went wrong. This could influence the relation between the dike height and the FoS that is shown in Figure 5.4. The sampling procedure for the dike height will change for the improved Simple model and this will affect the relation between the dike height and the FoS too. The relation between the dike height and the FoS will be discussed further in section 5.3 - Improved 'Simple' model.

The water height is strongly negative correlated with the FoS in Table 5.3 and a negative trend can be observed in the plot of the water height against the FoS in Figure 5.2, which is in line of what is expected.

The plot that stands out is the plot of the thickness cohesive layer against the FoS. A sharp decrease in the FoS for the first meters of thickness of the cohesive layer can be observed and for greater thicknesses the decrease in the FoS is less. In Table 5.3, the thickness of the cohesive layer is negatively correlated with the FoS. Overall, a negative trend can be observed in the plot of the thickness of the cohesive layer against the FoS, but for the first meters a strong decrease. Looking at the correlation coefficient for the thickness of the cohesive layer and the FoS, at first glance it is not expected that the relationship between these two parameters proceeds in this way, but a negative trend can be observed which is in line with what is expected.

The correlation coefficient between the inner slope of the dike and FoS is extremely positive and this relationship can also be observed in Figure 5.2. In comparison with the other input parameters, the resulted range for the FoS is the biggest for the input parameter inner slope. This is expected since the correlation coefficient is the highest for the correlation between the inner slope of the dike and the FoS.

5.2.3 Data investigation

In Figure 5.1, each input parameter, sampled with RS, is plotted w.r.t. the FoS and the positive/negative trends in some plots can be observed. The values for the FoS are not that high in Figure 5.1, the minimum value for the FoS is 0.6 and the maximum 1.5. The average FoS value for the complete dataset, 0.9, is lower than the value for the FoS for which a dike is called stable. In this research, a dike is called stable when the FoS is at least 1.2. In order to increase the values for the FoS, to have more stable dikes in the training dataset, some changes in the input parameters must be made.

As mentioned in the previous section, some changes in the input parameters should also be made in order to improve the model's performance. Before introducing the changes with the corresponding explanation, the data is investigated first.

For the Simple model, a slip plane constraint is applied. The slip plane constraint is that the minimum circle depth is 1 m. This constraint is applied since shallow slip circles arose on the inner slope of sand dikes with the lowest point of the slip circle being 0.5 m above the ground level, which means that the slip circle did not enter the subsoil and only developed in the dike

core. To be sure that no shallow slip circles occur and that the slip circles will develop in the subsoil, a minimum circle depth of 1 m is applied. That deeper slip circles will appear, can be confirmed by plotting the bottom of the slip circle. The lowest point of the slip circle is calculated by extracting the radius of the slip circle from the y-coordinated of the midpoint of the slip circle. To see if a correlation with the FoS exists, the bottom of the slip circle is plotted against the FoS. The bottom of the slip circle w.r.t. the FoS can be observed in Figure 5.3.

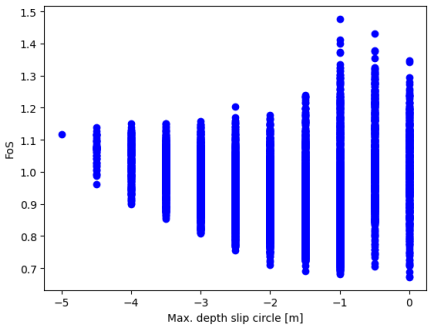


Figure 5.3: The maximum depth of the slip circle in the subsoil w.r.t. the FoS. The data in the plot is sampled with LHS.

The pattern that is observed in the plot of the bottom of the slip circle vs the FoS in Figure 5.3 is due to the resolution that is initially set for the D-Stability calculations. In this plot, a difference of 0.5 m between the lowest points of the slip circles can be observed, which is the set resolution. If the radius of the slip circle is plotted against the FoS and the y-coordinate of the mid point of the slip circle against the FoS, both plots looks similar, but have a small shift. This can be observed, when the same D-Stability calculation is highlighted with a red dot in both plots, see Figure 5.4. The shift is due to depth of the slip circle into the subsoil and, as can be observed from Figure 5.4, the radius of the slip circle is always higher than the y-coordinate of the mid point of the slip circle, indicating that the slip circle always enters the subsoil in this training dataset.

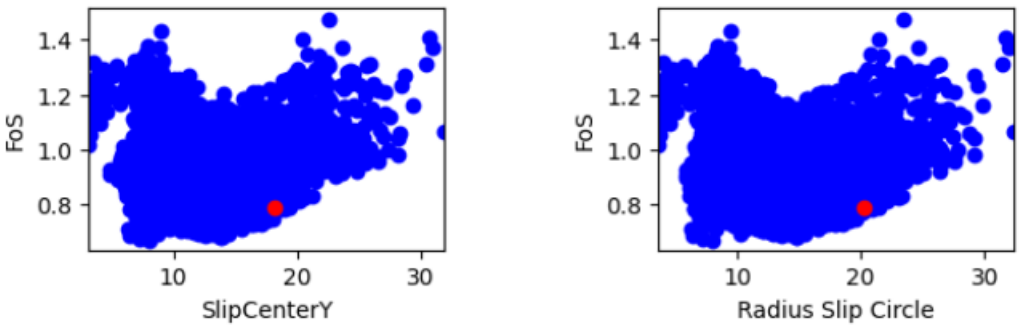


Figure 5.4: The values for the different y-coordinates of the mid point of the slip circle and the radius of the different slip circles are plotted w.r.t. the FoS. Both results from the D-Stability calculations are derived from the input parameters sampled with LHS.

Error plot for the first best simple model

As mentioned in section 5.2.2 - Polynomial functions input parameters vs FoS, the Feedforward Neural Network derived from sampling with Random sampling is the best model yet. For this

model, the errors are plotted in a histogram. The values on the x-axis are the differences between the actual FoS and the predicted FoS by the best model, the residuals.

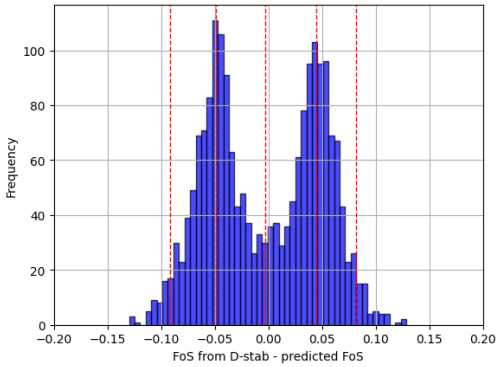


Figure 5.5: Histogram of the difference between the actual FoS and the predicted FoS for the Simple model. The red lines belong to the residual values [-0.09, -0.05, 0, 0.04, 0.08] that corresponds to the 2.5%, 25%, 50%, 75%, and 97.5% of the data.

Looking at Figure 5.5, the size of the residuals is not that big for a first training, without adjusting the input parameters, and no significant outliers can be observed. Another interesting thing is the shape of the histogram. It makes more sense if the frequency is the highest around 0, like a normal distribution, where the predicted FoS equals the actual FoS. Right now, the plot looks like two normal distributions that overlaps each other.

The shape of the histogram can be explained by printing the index values of the test dataset. Index values from 0 to 4999 belongs to a dike containing a clay core and index values 5000 to 9999 belong to a dike containing a sand core. When the index values are stored for residuals values that are lower than the 25 percentile and higher than the 75 percentile, which more or less corresponds to the mean of the two normal distributions, the cause of the shape of the plot in Figure 5.4 can be explained. 94% of the residuals that are lower than the 25 percentile belongs to a dike containing clay and 94% of the residuals that are higher than the 75 percentile belongs to a dike containing sand.

The model is not trained on the dike core type. When using the best model and create a dataset containing only clay dike core. The shape of the histogram is the shape that it is expected to have, see Figure 5.5.

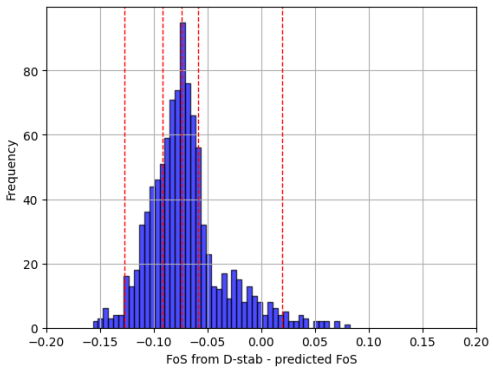


Figure 5.6: Histogram of the difference between the actual FoS and the predicted FoS for the Simple model. The data contain only clay core dikes. The red lines belongs to the residual values [-0.13, -0.09, -0.07, -0.06, 0.02] that corresponds to the 2.5%, 25%, 50%, 75%, and 97.5% of the data.

The whole dataset contains dikes having a clay or a sand core. This a parameter that is varied, but it is not taken into account when the ML models are trained. It is expected that the models' performances are slightly worse when this parameter is not taken into account in training the ML models since some knowledge is taken away from the training process. That is why this parameter should be added to the next training process.

Changes based on data investigation

Based on the conclusions drawn in the previous sections of this chapter, some changes will be applied to training dataset. The changes and explanation of the changes are summarized in this section. These changes will be implemented to improve the Simple model. The mentioned changes to the training dataset are not necessarily about improving the values for the FoS, but it is more about obtaining a representative dataset, which has a good distribution of stable and unstable dikes.

- Range inner slope: **1:2.5 - 1:4** instead of 1:2 - 1:4. In the plot inner slope vs FoS in Figure 5.1, the values for the FoS for an inner slope between 1:2 and approximately 1:2.5 are never higher than 1.0.

  In general, a dike is called stable when the FoS is equal to or higher than 1.0. However, in dike reinforcement projects, a dike is never built with a safety factor that is equal to 1.0 since in that case there is no safety margin. That is why a dike is called stable when the FoS is equal to or higher than 1.2 in this research.

  Since many dike geometries that are considered in this research have a FoS lower than 1.0, average FoS for the dataset is 0.9, the FoS should be increased in order to have more stable dikes in the training dataset such that the surrogate model can also predict the FoS accurately for stable dikes. The value 1.0 for the FoS is highlighted in this part since it emphasizes that the training dataset consists of a few stable dikes. In order to increase the values for the FoS, the inner slope range is adjusted.

- Range thickness cohesive layer: **0 - 5 meters** instead of 0 - 8 meters. The initial range 0 - 8 m is based on DINOloket [13]. A cross-section of the substrate from north to south and from west to east in the Netherlands is created in DINOloket in order to see that the cohesive layer is often between 0 and 8 m.

For the thickness of the cohesive layer is a strong negative correlation between the FOS and thickness of cohesive layer for a range of 0 - 2 m for the thickness of the cohesive layer, as can be observed in Figure 5.2. It seems that a smaller thickness influences the FoS more than a larger thickness. The range of the thickness of the cohesive layer should be at least from 0 - 2 m to capture the critical zone where small variations can have significant effects on the stability of dikes.

Between 0 - 3 m thickness of the cohesive layer, the lowest point of the slip circle can be found under the cohesive layer, i.e. in the sand layer. This happens about 1000 times out of 10.000 D-Stability calculations. For the model's performance, it would be good to include all cases where the slip circle is below the cohesive layer. This means that the thickness of the cohesive layer must certainly be taken up to 3 m.

The lowest point of the slip circle can be found at - 4.5 m below ground level for a range of 0 - 8 m cohesive layer, only once this is -5 m. To include the different depths of the slip circle in the improved model, the changed range should be at least 0 - 5 m for the thickness of the cohesive layer.

The FoS decreases a lot when the difference between the thickness of the cohesive layer and the lowest point of slip circle is bigger than 5 m. High values for the difference between thickness cohesive layer and the lowest point of the slip circle indicates that the cohesive layer is thick and that the slip circle develops very shallow in the cohesive layer. Higher thicknesses of the cohesive layer does not influence the progression of the slip circle since it will develop very shallow in the subsoil, but higher thicknesses of the cohesive layer result in lower values for the FoS. In order to create a dataset with higher values for FoS, values for the thickness of the cohesive layer higher than 5 m is not included in the new range.

For all the reasons mentioned, the thickness of the cohesive layer will vary between 0 and 5 meters. That is why a range of 0 to 5 meter is more suitable in this research than a range of 0 to 8 meter.

- There is a strong negative correlation between the FoS and the thickness of the cohesive layer for a range of 0 - 2 m, as can be observed in Figure 5.2. More samples are taken in this range of the cohesive layer for the improved simple model in order to capture this strong negative correlation; 6000 samples for range 0 - 2 m and 4000 samples for range 2 - 5 m are taken.

- Low values of the FoS is also caused by the schematization of the phreatic line. First, the phreatic line was schematized by the rules from the Technische Adviescommissie voor de Waterkeringen. This resulted in a high phreatic level inside the dike core. Therefore another schematization is considered to get a lower phreatic level inside the dike core, which is also elaborated in section 3.2.3, in order to get a more realistic representation of the phreatic line.

- In general, dike cores containing only sand are never built. Most sand dikes have a partial or complete clay cover. For simplicity, the clay cover was not schematized in the Simple model and only schematized for the Complex model. However, it is better to schematize it for the Simple model too. Adding a clay cover on top of a sand dike core lowers the phreatic level inside the dike by reducing water infiltration, which is beneficial for the FoS. The thickness of the clay cover can be observed from Table 3.7.

- In section 5.2.2 - Sensitivity analysis, it was mentioned that the sampling of the dike

height went wrong. The samples were removed for which the water height exceeded the dike height. In this way, the distribution for the dike height was changed. In order to sample the dike height following an uniform distribution, only the values for the water height needs to be changed. This means that, during sampling, when the value for one water height sample is higher than the value for one dike height sample that the value for the water height sample is corrected to the value for the dike height sample. This correction for the water height is implemented in the improved simple model.

- The last change is to add the input parameter whether the dike core contains clay or sand. The schematization of the phreatic line is different for a clay and sand dike and if the model know whether the dike contains clay or sand, the model performance may also increase. The number 0 is assigned to a dike containing clay and the number 1 is assigned to the dike containing sand.

  In addition, looking at the error plots in section 5.2.3 - Error plot for the first best model, the error plot makes more sense when including the dike core.

5.3 Improved 'Simple' model

The changes that are mentioned at the end of section 5.2.3 are implemented in the code and the 10.000 D-Stability calculations are performed again. In Tables 5.4 and 5.5, the results after training the ML algorithms can be observed with adding the extra input parameter the type of soil in the dike core. Adding this input parameter has a significant impact on MSE, MAE, and R^2-score. The values for the MSE and MAE are quite lower and the values for R-squared are higher compared to the previous results for the validation metrics, which means that the ML models improved and can predict the FoS better.

ML models	MSE	MAE	R^2-score
RFR	$4.0 \cdot 10^{-3}$	$1.4 \cdot 10^{-2}$	0.98
XGB	$6.4 \cdot 10^{-3}$	$1.7 \cdot 10^{-2}$	0.97
SVR	$\mathbf{2.2 \cdot 10^{-3}}$	$\mathbf{1.0 \cdot 10^{-2}}$	**0.99**
FNN	$2.3 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$	**0.99**

Table 5.4: Results of the validation metrics for Random Sampling for the different ML models for the improved simple model with extra column dike core.

ML models	MSE	MAE	R^2-score
RFR	$3.6 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$	0.98
XGB	$6.9 \cdot 10^{-3}$	$1.9 \cdot 10^{-2}$	0.97
SVR	$\mathbf{2.5 \cdot 10^{-3}}$	$\mathbf{1.1 \cdot 10^{-2}}$	**0.99**
FNN	$\mathbf{2.5 \cdot 10^{-3}}$	$1.2 \cdot 10^{-2}$	**0.99**

Table 5.5: Results of the validation metrics for LHS for the different ML models for the improved simple model with extra column dike core.

According to Tables 5.4 and 5.5, the best model for the improved Simple model is Support Vector Regression sampled with RS. In Figure 5.7, a histogram can be observed containing the differences between the actual and the predicted FoS.

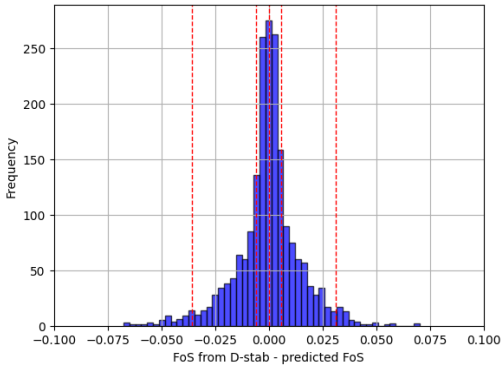


Figure 5.7: Histogram of the difference between the actual FoS and the predicted FoS for the improved Simple model. The red lines belong to the residual values [-0.04, -0.01, 0.00, 0.01, 0.03] that corresponds to the 2.5%, 25%, 50%, 75%, and 97.5% of the data

As can be observed from Figures 5.5 and 5.7, the errors are smaller and the mean of the error lies around 0, as expected. 95% of the data have a maximum deviation of 0.04 from the actual FoS. This is a very small deviation. In reality, at an early stage of dike design, a maximum deviation of 0.1 is allowed, which indicates that a deviation of 0.04 from the FoS is quite accurately.

5.3.1 Sensivity analysis

The correlations between the input parameters and the FoS for the improved Simple model are shown in a pairplot in Appendix A. The values for Kendall's tau are slightly changed, which makes sense because the ranges for the input parameters did change and this time the water heights are only corrected in order to have the water heights equal to or lower than the dike height. However still the different input parameters are correlated in a similar way with the FoS as in section 5.2.2 - Sensitivity analysis.

However, as for the polynomial functions between the different input parameters and the FoS, the polynomial function between the dike height and the FoS did change. The polynomial function is not a gently increasing linear function anymore, as can be observed in Figure 5.8. The polynomial function is a gently decreasing polynomial function, which is in line with the Kendall's tau, -0.14, as can be observed in Appendix A.

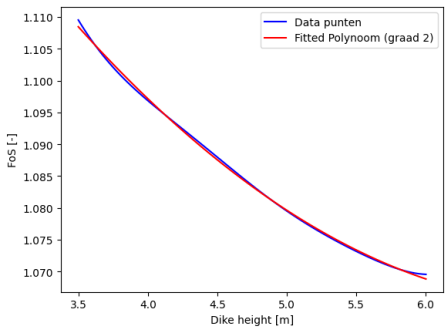


Figure 5.8: Relations between the dike height and the FoS sampled with Random sampling.

5.3.2 Best sampling method

Looking at Tables 5.4 and 5.5, the difference in model's performance between the two different sampling methods is quite small. In general, Random Sampling has lower values for the MSE and MAE and Latin Hypercube Sampling has slightly higher values for the R^2-score, but these difference are almost negligible. The relative differences between the R^2-scores are a bit higher than for the MSE and MAE for the two different sampling techniques. The differences between the R^2-scores are in the order of 10^{-2} - 10^{-3}. Since the data for the Complex model consists of more complex dike geometries with more input parameters, it is good to have a high R^2-score that the model fits the data well. In addition to that, 10.000 D-Stability calculations will be performed to create the Complex model, the same amount as for the Simple model, but with more input parameters. In order to gain as much as information from the data to create a good surrogate model, LHS is a better sampling method than RS.

For these reasons, the Complex model will be created with Latin Hypercube Sampling.

5.4 'Complex' model

Since it is possible to create a surrogate model with D-Stability calculations for simple dike geometries, complex dike geometries are considered. Complex dike geometries are the simple dike geometries extended with a berm and a ditch. The input parameters for the Complex model can be observed from Tables 3.7 and 3.8. In section 5.3, the ranges for the input parameters thickness cohesive layer and inner slope were changed in order to improve the model's performance and get higher values for the FoS for the simple dike geometries, but the ranges for the thickness cohesive layer and inner slope for the Complex model will be the initial, old ranges, as a starting point. The difference w.r.t. the initial input parameter ranges of the Simple model is that the Complex model has added input parameter dike core since this resulted in a better error distribution.

In addition to the dike core, the thickness of the clay cover on top of the sand dike also changed. A thickness of 1.5 m for the clay cover was initially chosen because this resulted in another schematization of the phreatic line for the sand dike. However, the schematization of the phreatic line has changed from the rules from the Technische Adviescommissie voor de Waterkeringen to the schematization from the D-Stability manual. The thickness of the clay cover does not influence the schematization of the phreatic line according to the D-Stability manual. The initial thickness of 1.5 m is a bit on the high side and a more common thickness for the clay cover is 0.8 m. On top of the clay cover, a certain type of soil is situated with a thickness of 0.1 m in order to let grass grow. This results in a total thickness of the clay cover of 0.9 m for the Complex model instead of 1.5 m that was considered for the Simple model.

Initially, the code for deriving the D-Stability calculations for the Simple model was copied and extended with the complex dike geometries. However, the Python code could not be copied since error arose due to the fact that some points were overlapping each other, while this error did not occur during running the 10.000 D-Stability calculations for the improved Simple model. In the meantime, the HWL datamodel was modified and a new function was developed, Modification_ClayLayer(). This function applies a clay cover on top of the sand dike without changing the geometry of the dike. This clay cover is a bit extended in the subsurface, which can not be removed, otherwise errors will occur in the code. After implementing this function, the D-Stability calculations can be performed.

As a starting point, 10.000 D-Stability calculations are performed for the Complex model. The

amount of D-Stability calculations was enough for the Simple model and the Complex model has only three additional input parameters compared to the Simple model, so that is why 10.000 D-Stability calculations are performed initially.

After running the 10.000 D-Stability calculations, several outliers can be observed in the dataset. These outliers have a very low FoS. When looking at the D-Stability calculations of these outliers, it is observed that these low values for the FoS result from the shallow slip circles. These shallow slip circles are not the interesting slip circles for the failure mechanism inner slope stability. That is why these D-Stability calculations are removed from the dataset.

After removing the outliers in the dataset, a pairplot can be created for the complex dike geometries. Since the figure is large, the pairplot, together with the correlations between the different input parameters and the FoS, can be observed in Appendix A. The values for Kendall's tau that are computed in the pairplot can also be observed in Table 5.6.

Parameter	Correlation w.r.t. FoS
Dike height	-0.19
Crest width	0.03
Inner slope	0.27
Thickness of the cohesive layer	-0.17
Water height	-0.23
Inner berm width	0.41
Distance ditch - inner toe dike	0.17
Depth ditch	-0.04
Dike core	-0.12

Table 5.6: Kendall's correlation between the input parameters of the complex surrogate model and the FoS.

As for the Simple model, the crest width is still less correlated with the FoS. The dike height is slightly more negatively correlated with the FoS compared to the Simple model. The inner slope and the water height are less correlated with the FoS for the Complex model than for the Simple model. The correlation between the thickness of the cohesive layer and the FoS remains more or less the same. The input parameter that influences the FoS the most is the inner berm width. When the width of the inner berm increases, the stability of the inner slope increases as well. The input parameters distance ditch - inner toe of the dike and depth of the ditch influences the schematization of the phreatic line a bit. The schematization of the phreatic line has a significant impact on the value for the FoS and that is why these two input parameters have a slight effect on the FoS.

When the ditch is further away, the schematization of the phreatic line is more gradual and not sharp. No sudden drop exists in the phreatic line, which results in a more stable inner slope of the dike.

The water level inside the ditch is given, 0.4 m below ground level, and only the depth of the ditch varies. The pore water pressure remains the same when varying the depth of the ditch. This means that the water level inside the ditch does not influence the height of the phreatic line near the ditch. If the water level inside the ditch would vary, then the water level would have some impact on the FoS. For these reasons, the depth of the ditch has less impact on the value for the FoS.

After computing the correlations between the different input parameters and the FoS, the different ML models are trained and evaluated with the different validation metrics. The results after the training are stated in Table 5.7.

ML models	MSE	MAE	R^2-score
RFR	$9.0 \cdot 10^{-4}$	$1.8 \cdot 10^{-2}$	0.93
XGB	$1.2 \cdot 10^{-3}$	$2.3 \cdot 10^{-2}$	0.90
SVR	$\mathbf{6.0 \cdot 10^{-4}}$	$\mathbf{1.2 \cdot 10^{-2}}$	**0.95**
FNN	$9.0 \cdot 10^{-4}$	$1.7 \cdot 10^{-2}$	0.93

Table 5.7: Results of the validation metrics for the different ML models for the Complex model with the FoS as output.

The results after training for the different ML models are already quite good. From Table 5.7, Support Vector Regression is the best ML model. In order to see how good this ML model the FoS predicts, a histogram is made, containing the differences between the actual FoS and the predicted one by SVR model. The histogram is shown in Figure 5.9.

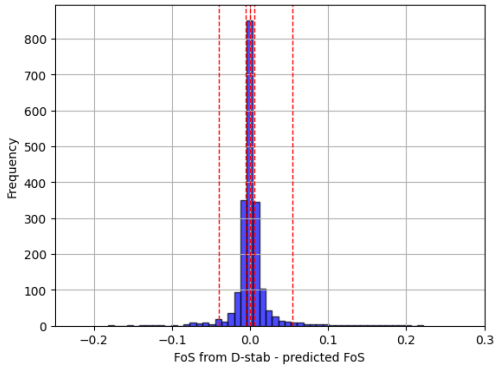


Figure 5.9: Histogram of the difference between the actual FoS and the predicted FoS for the Complex model. The red lines belong to the residual values [-0.04, -0.01, 0.0, 0.01, 0.05] that corresponds to the 2.5 %, 25%, 50%, 75%, and 97.5% of the data.

As can be observed from Figure 5.9, 95% of the data has a maximum deviation of 0.05 from the actual FoS. Figure 5.9 consists of some outliers in the errors. When the corresponding D-Stability calculations are investigated, a clear pattern can not be recognized why these outliers occur. However, in comparison with the best model's performance for the Simple model, the model performs well for the complex dike geometries. The dike geometries are much more complex than the dike geometries considered for the Simple model, but that does not influence the model's performance that much. The SVR model will be used to assess the inner slope stability of dikes.

5.5 'Complex' model - slip plane

In the previous section, the surrogate model for the complex dike geometries is created. For visualisation purposes, it would be nice if the surrogate model can also assess the slip circle. To visualize the slip circle, the surrogate model should be trained at least on the outputs the

coordinates of the midpoints of the active and passive circle, the radius of the active and passive slip circle, and the FoS. The coordinates of the midpoints of the active and the passive slip circle are relative coordinates. The dike geometries are built from a reference point in D-Stability with a x-coordinate that is equal to the x-coordinate of the crest of the inner slope and a y-coordinate that is equal to the ground level.

For training the surrogate model on more outputs, the same dike geometries as the Complex model with only the FoS as output is used. The same ML models can be used for training the models on more output parameters, except for the SVR model. The SVR model is typically used for single output regression problems and it can only predict multiple outputs when the models, for each output parameter, are combined in another type of model. Since three ML algorithms are left from the previous sections, which can predict multiple outputs, it is not necessary to introduce a new type of ML algorithm. Therefore only the RFR, XGB, and FNN models are trained first. The results after training the different ML models can be observed in Table 5.8.

ML models	MSE	MAE	R^2-score
RFR	$\mathbf{3.2 \cdot 10^{-3}}$	$\mathbf{3.1 \cdot 10^{-2}}$	**0.65**
XGB	$3.6 \cdot 10^{-3}$	$3.5 \cdot 10^{-2}$	0.59
FNN	$3.7 \cdot 10^{-3}$	$3.6 \cdot 10^{-2}$	0.60

Table 5.8: Average results of the validation metrics for the different ML models with the complex dike geometries with more outputs.

As can be observed from Table 5.8, the values for the MSE and MAE are higher and the values for the R^2-score are lower compared to the values from Table 5.7, where the FoS was the only output parameter. This makes sense because the amount of output parameters has increased, leading to higher complexity and requires more data, computational resources. In Table 5.8, the RFR-model has the best performance. In order to see why the ML models' performance is bad and which output parameter causes poor models' performance, the values for MSE, MAE, and R^2-score are calculated separately for each output parameter for the RFR-model. These results are shown in Table 5.9.

Output parameter	MSE	MAE	R^2-score
FoS	$1.4 \cdot 10^{-3}$	$2.5 \cdot 10^{-2}$	0.89
M1x	$8.5 \cdot 10^{-3}$	$5.4 \cdot 10^{-2}$	0.79
M1y	$8.0 \cdot 10^{-4}$	$2.0 \cdot 10^{-2}$	0.77
M2x	$1.8 \cdot 10^{-3}$	$3.2 \cdot 10^{-2}$	0.93
M2y	$2.0 \cdot 10^{-3}$	$1.2 \cdot 10^{-2}$	0.15
R1	$1.1 \cdot 10^{-3}$	$2.3 \cdot 10^{-2}$	0.81
R2	$2.0 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$	0.19

Table 5.9: Results of the validation metrics for the RFR-model with the complex dike geometries and different output parameters.

In Table 5.9, many interesting aspects can be observed. The values for the MSE and MAE is very high for the output parameter M1x, when these values are compared to the values for the MSE and MAE for the other output parameters. However, the value for the R^2-score for output parameter M1x is not low. When the values for the MSE and MAE are high and the R^2-score

is not low, the ML model captures the variability in the data well, but the target variable has a large range or high variance. Comparing the variance of M1x with the variance of other output parameters, then the variance is of similar size and does not stand out. The high values for MSE and MAE and low values for the R^2-score can also be explained the fact that the model is complex. The ML model is trained on 9 input parameters and 7 output parameters. If the relationship is complex, a simple ML model can capture the overall variance, but can still have large errors for specific predictions.

When looking at the values for the R^2-score in Table 5.9, then the output parameters M2y and R2 score very low. For the output parameters M2y and R2, the values for MSE and MAE are low and the R^2-score is low. The parameters M2y and R2 are related to each other since the value for R2 can be calculated when the value for M2y is known. That is why the scores for MSE, MAE, and R^2-score are of similar size for these two output parameters. When the values for the MSE and MAE are low and the R^2-score is low, the model's predictions are close the the actual target values, but the model can not capture the variability in the data well. Looking closely at the relative y-coordinates of the midpoint of the active and the passive slip circle, 0.1% of whole dataset for the relative y-coordinates of the active slip circle and 0.2% of the whole dataset for the relative y-coordinates of the passive slip circle are outliers. When the test dataset is investigated, this dataset does not contain outliers for the M1y values, but it does contain outliers for the M2y values. A few outliers are in the test dataset causing large errors, but not that many outliers are present in the test dataset to significantly increase the MSE value. Since outliers are present in the test dataset, the model can not predict these outliers well resulting in higher values for SSE and SST in Equation 4.3 and thus a lower R^2-score. The total variance can not be explained by the ML model due to outliers resulting in a low R^2-score.

The outliers in the values for M1y come mainly from the clay core dike, only 1 outliers is coming from a sand core dike. The outliers in the values for M2y come only from the sand core dikes. For both M1y and M2y, the outliers are very high values for the relative y-coordinates of the midpoints of the active and passive slip circle. Looking at the sand core D-Stability calculations with high values for M2y, the cohesive layer underneath the dike is between 0 and 1 meter for all calculations and the slip circle ends almost horizontally in the ditch. This indicates that the radius of the passive slip circle does not really matter as long as the slip circle ends in the ditch. The slip circle does not start developing near the crest of the dike, but at the end of the inner berm.

For the outliers in the values for M1y are not so clear. It is a combination of input parameters that result in high values for M1y. Due to high values for the width of the inner berm, high thickness of the cohesive layer underneath the dike and a large distance between the inner toe of the dike and the ditch, the value for M1y is high. Another case where the value of M1y is high is when the thickness of the cohesive layer underneath the dike is low.

When the outliers for M1y and M2y are kept away from the test dataset and only exists in the training dataset, then the ML models do not improve. When the outliers for M1y and M2y are removed from the dataset, the values for MSE and MAE increase and the R^2-score's increase. The results after training the ML models, without the outliers for M1y and M2y, can be observed in Table 5.10.

ML models	MSE	MAE	R^2-score
RFR	$\mathbf{4.5 \cdot 10^{-3}}$	$\mathbf{4.3 \cdot 10^{-2}}$	**0.73**
XGB	$4.9 \cdot 10^{-3}$	$4.5 \cdot 10^{-2}$	0.69
FNN	$4.8 \cdot 10^{-3}$	$4.5 \cdot 10^{-2}$	0.70

Table 5.10: Average results of the validation metrics for the different ML models with the complex dike geometries with more outputs. The complete dataset does not contain the outliers for M1y and M2y.

The low models' performances are mainly caused by the outliers of M2y. To improve the models' performances, the ML models need to better understand this parameter. Therefore it has been decided that more samples are taken from a low thickness for the cohesive layer since this causes high values for M2y. In Table 5.11, the results are shown after getting 5.000 additional samples, 2.000 for clay core dike and 3.000 for sand core dike, with a low thickness of the cohesive layer.

ML models	MSE	MAE	R^2-score
RFR	$2.7 \cdot 10^{-3}$	$3.1 \cdot 10^{-2}$	**0.68**
XGB	$3.0 \cdot 10^{-3}$	$3.3 \cdot 10^{-2}$	0.63
FNN	$\mathbf{2.5 \cdot 10^{-3}}$	$\mathbf{2.9 \cdot 10^{-2}}$	0.66

Table 5.11: Average results of the validation metrics for the different ML models with the complex dike geometries with more outputs. The previous total dataset is increased with 5.000 additional samples.

Looking at Table 5.11, training the ML models with 1.5 times the original dataset, the models' performances do not improve significantly. Still the RFR model is the best model, even though the MSE and MAE values are not the lowest for this model, but that is due to rounding off the values for MSE and MAE and the difference is very small compared to the difference between the R^2-scores. The individual value for the MSE, MAE and R^2-score for each output parameter is a bit improved, slightly less improved, or not improved at all compared to initial training with 10.000 D-Stability calculations.

All steps that are taken in order to improve ML models based on data investigation did not work out yet. In general, the performance of ML models increase when the amount of output parameters decreases. When the ML algorithms are trained on all input parameters and on a single output parameter, the models' performance remain in similar size as the results in this section.

Looking at the results of the performance of the different ML models, then the differences in validation metrics are quite small, indicating that another type of ML algorithm won't help increasing the models' performance.

For the Simple model, when the dike core input parameter was added to the dataset, the models' performance increased significantly, meaning that changing the training dataset has a significant impact on the performance of the ML models. In order to increase the models' performance, it would be beneficial if an input parameter can be added to training dataset, which gives some information about the slip circle development.

However, when the predicted and the actual slip plane are computed for a dike containing clay and sand in its core, then the difference between the two does not look bad, as can be seen

in Figures 5.10 and 5.11. Despite the fact that the relative distances of the coordinates of the midpoints of the active and passive slip circles are predicted wrong, the representation of the slip plane is still accurate.

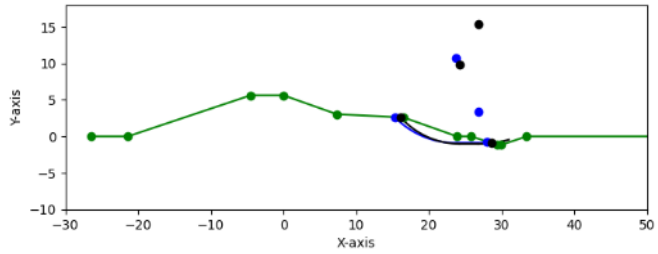


Figure 5.10: Plot of the difference between the actual and the predicted slip circle for a dike containing sand in its core.

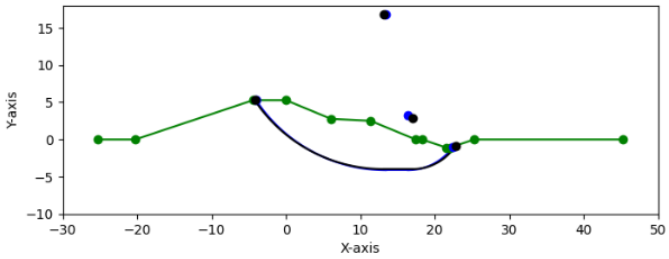


Figure 5.11: Plot of the difference between the actual and the predicted slip circle for a dike containing clay in its core.

5.6 Conclusion

This chapter presents all the needed steps to derive a suitable surrogate model for the macro-stability of the inner slopes of dikes. First, the Simple model is evaluated. After running the first 10.000 D-Stability calculations, the correlation matrix and the validation metrics are printed for the dataset. The inner slope is the most influencing parameter on the FoS. When increasing the inner slope of a dike, the FoS increases. The model's performance is already quite good, but it can still be improved. In the next subsection, the data is investigated in order to see which input parameter needs to be adjusted to increase the model's performance. Interesting to see in the plots of the different input parameters vs the FoS is that the values for the FoS are not that high. For the improved Simple model, the values for the FoS need to be higher since for this research the dike is safe at a FoS of 1.2 and most of the schematized dikes have a FoS lower than 1.2. That is why the input parameter ranges inner slope and thickness of the cohesive layer are adjusted. After changing these input parameter ranges, the model's performance is good and predicts the FoS well.

After deriving the Simple model, the dike geometries are adjusted. The 10.000 D-Stability calculations are performed again, but this time the input parameters are only sampled with Latin Hypercube Sampling since the results from this sampling method was a bit better compared to Random Sampling for the Simple model. After running the 10.000 D-Stability calculations, the correlation matrix is plotted and the validation metrics are printed. The most influencing input

parameter on the FoS is the width of the inner berm for the complex dike geometries. When increasing the width of the berm, the FoS increases the most. The values for the MSE and MAE in Table 5.7 are very low and the value for the R^2-score is quite high after training the different ML models. When looking at Figure 5.9, it is not needed to improve the best model from Table 5.7 since 95 % of the predicted FoS's have a deviation of 0.05.

For visualisation purposes, it would be nice to assess the slip circle too. More output parameters have been added to the dataset in order to train the ML such that it can predict the slip circle too for complex dike geometries. Many steps have been taken in this chapter to improve the ML models, but the models' performance for some output parameters remains bad. However, when plotting the actual slip plane against the predicted slip plane for a dike containing clay or sand in its core, the representation of the slip plane is still accurate.

When comparing the approach of improving performance of the ML models for the complex dike geometries with the simple dike geometries, adding additional input parameters to the training dataset did increase the models' performance for the simple dike geometries. Looking at approach, it would be beneficial if an input parameter can be added to training dataset for complex dike geometries, which gives some information about the slip circle development.

Chapter 6 Integrated model

6.1 Introduction

In the first part of this report, a surrogate model for the macro-stability of the inner slope is derived. In the second part of this report, the models for the other engineering aspects related to dike design are explained and the Multi Criteria Analysis (MCA) is explained. This chapter focuses on explaining the models for ecology, wave overtopping, costs of the dike, and the Environmental Cost Indicator (ECI) for dike design. These models will be used, together with the macro-stability model, to get to an integrated dike design. At the end of this chapter, the implementation of the ditch in the cost and ECI model is explained separately since it needed a further detailed explanation about changing the ditch geometry.

6.2 Ecology

Surface water quality is under significant pressure due to some natural processes and human activities. Surface water is polluted by industrial discharges, agricultural intensification, and urbanization and is influenced by climate change due to changed rainfall patterns, increased temperature, and more extreme weather events. When the surface water is too polluted, it is less suitable for drinking water, irrigation, recreation, and other functions, while an increasing demand exists for clean water [7].

It is important that the surface water remains clear and healthy and when the water is not clear, that measures are taken to achieve clean water. Which measures have the most effect on improving the water quality for a certain ditch, is shown by the PCDitch model. PCDitch is a metamodel developed by W+B, STOWA, NIOO-KNAW, Netherlands Environmental Assessment Agency and the WUR and is based on a dataset with 360.000 runs by the PCDitch model developed by Jan Janse [69]. The PCDitch model is the ecology model in this research and will be implemented in the MCA. It is a trained neural network, which requires a value for the water depth of the ditch, flow rate, and sediment type as input parameters. The output parameter of this model is the critical nutrient load. The critical nutrient load is the maximum amount of nutrients that the ditch can receive without negative effects on the ecological value of the ditch.

When the nutrient load increases in a ditch, the system can initially handle this. When the increase in the nutrient load is too large, the critical nutrient load is exceeded, plants and fish are saturated and algae take advantage of this. The amount of algae increases in the ditch and blocks the sunlight. Plants can receive less sunlight and the amount of plants decreases. The diversity of fish is also decreasing and the (submerged) plant dominates. The system changes from a clear to a turbid state. The system can change to a clear state with a plant-rich environment if the nutrient load is greatly reduced. The PCDitch model can be used to derive the right measures to restore the system in order to the plan dominated clear state. The ranges

for the input parameters for the ecology model are shown in Table 6.1. The output parameter of the Ecology model is the critical nutrient load. To have a stable environment in the ditch, the critical nutrient load should exceed the external nutrient load. When optimizing the dike designs, the critical nutrient load should be maximized.

Input parameter	Value/Range	Unit	References
Water depth ditch	depth ditch - 0.4*	m	-
Flow rate	20	mm/d	[23]
Sediment type	Clay Sand	- -	- -

Table 6.1: Values for the input parameters for the ecology model.

*The water level inside the ditch is kept constant at a height of 0.4 m below ground level. The water depth inside the ditch is calculated by taking the difference of the depth of the ditch and the water level inside the ditch.

The surrogate model for macro-stability and the ecology model have two sheared input parameter, the depth of the ditch and the thickness of the cohesive layer.

When the thickness of the cohesive layer is equal to or bigger than the depth of the ditch, the sediment type in the ditch is clay. The sediment type is sand when the thickness of the cohesive layer is smaller than the depth of the ditch.

The water level inside the ditch is kept constant at a height of 0.4 m below ground level and the depth of the ditch varies, which results that the water depth in the ditch also varies.

In general, for ditches with the sediment type clay, the more shallow the ditch, the higher critical nutrient load. For ditches with the sediment type sand, the optimal ditch depth is approximately 0.96 m. In Figure 6.1, the depth of the ditch is plotted against the critical nutrient load for sediment type clay and sand.

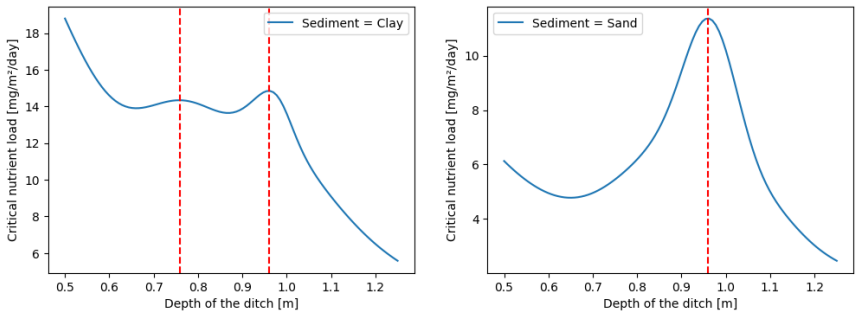


Figure 6.1: Depth of the ditch plotted against the critical nutrient load for sediment types clay and sand. The red lines in both plots show the local maximum of the critical nutrient load.

For macro-stability of the inner slope of the dike yields, the more shallow the ditch, a higher value for the FoS. However, the pore water pressure remains the same in this research since the water level in the ditch is kept constant at a level of 0.4 m below ground level.

6.3 Wave overtopping

This failure mechanism is already explained in section 2.2.2 and the amount of wave overtopping is calculated in this section. According to the requirements for the inner slope of a dike, the maximum amount of wave overtopping is 10 L/m/s for a dike with a clay cover and grass on top of that [58]. When the amount of wave overtopping is higher than 10 L/m/s, erosion on the inner slope of the dike will take place, which can affect the macro-stability of the inner slope of a dike.

The amount of wave overtopping is calculated with a formula derived from a report of the Technische Adviescommissie voor de Waterkeringen and can be observed in Equation 6.1 [58]. Equation 6.1 is a simplification of the complete formula from this report and is the maximum of the complete formula.

$$q = 0.2 \cdot \exp(-2.3 \cdot \frac{h_k}{H_{m0}} \cdot \frac{1}{\gamma_f \cdot \gamma_\beta}) \cdot \sqrt{g \cdot H_{m0}^3} \cdot 10^3 \tag{6.1}$$

where:

h_k is the free crest height above the still water line [m];

H_{m0} is the significant wave height at the toe of the dike [m];

γ_f is the influence factor for the roughness on the dike slope [-];

γ_β is the influence factor for the angle of wave attack on the dike slope [-];

g is the acceleration due to gravity [m/s^2];

The formula in Equation 6.1 will serve as the wave overtopping model in this research. For this model, parameters h_k, H_{m0}, and γ_β will vary and γ_f will be constant. The values for the different input parameters for the wave overtopping model are stated in Table 6.2. The parameters stated in Table 6.2 are the input parameter of the wave overtopping model and the amount of wave overtopping is the output parameter.

Input parameter	Value/Range	Unit	References
h_k*	dike height - water level	m	-
H_{m0}	[0 - 1.5]	m	Expert knowledge
γ_f	1.0	-	[58]
β	[0 - 80]	°	[58]
γ_β**	$1 - 0.0033 \cdot \lvert\beta\rvert$	-	[58]
g	9.81	m/s^2	-

Table 6.2: Values for the input parameters for the wave overtopping model.

*The dike height and the water level both vary, which results that this parameter also varies.

**γ_β varies because the angle of wave attack varies, indicated by the parameter β, which is dependent on the location where the dike is situated. The formula that is used to calculate γ_β is derived from the same report as Equation 6.1.

For the dike optimization, only the dike height can be optimized. H_{m0}, β, and the water level outside the dike are given for a dike on a specific location and can not be varied for that location.

For these input parameters, Tables 3.8 and 6.2 give the minimum and maximum value that the specific input parameter can have.

Wave overtopping is often accompanied by the wind set-up of water. The wind blows over water and the water level will be elevated, depending on the "free length" of the water surface. This should be taken into account in the calculation of the free crest height above the still water line, h_k. However, the dike geometries that are considered in this research are not location specific. This means that the "free length" of the water surface can not be determined and it is hard to incorporate the effect of wind in the calculation of h_k. That is why the difference between the dike height and the water level is used to calculate the free crest height above the still water line, but in reality the impact of wind should be taken into account in this parameter.

The surrogate model for macro-stability and the wave overtopping model have two sheared input parameter, the height of the dike and the outer water level. However, the two shared input parameters result in one input parameter for the wave overtopping model by taking the difference between the dike height and the outer water level.

When the dike is high and the water height is low such that the difference between the dike height and the water height is big, the amount of wave overtopping is small. A larger difference between the dike height and the outer water level is also more favorable for the surrogate model since a larger difference result in a lower phreatic level inside the dike core, higher effective stress, and therefore a higher value for the FoS.

6.4 Environmental Cost Indicator (ECI)

Sustainability is not only playing an increasingly important role in our lives, sustainability is also becoming increasingly important in projects. The environmental cost indicator is a value that indicates the size of the environmental impact for a certain project. All environmental impacts are included in the ECI, from the material- and energy consumption during extraction of raw materials up to the demolition and reuse phase. The impacts are calculated over the entire life of a product. All these different impacts are converted to one single value, the ECI. The lower the ECI, the more durable the dike design is.

To calculate the ECI for a certain project, W+B uses the program DuboCalc. DuboCalc is a calculation tool that assesses projects on sustainable material and energy use. This tool is developed by Rijkswaterstaat, who is responsible for the management and maintenance of the Dutch infrastructure. Before calculating the ECI value for a certain dike reinforcement project, the duration of the project must be indicated first in DuboCalc, in other words, how long the dike will remain in place. For a green dike, W+B often takes a lifespan of 50 years and this is also the lifespan of dikes that are considered in this research.

DuboCalc has a library in which objects for a dike reinforcement project can be found under 'HWBP Dijkversterking PU-fase'. Under 'HWBP Dijkversterking PU-fase' values for the ECI can be found for supplying, applying, excavating, and disposing of soil in a dike reinforcement project. These values are taken from DuboCalc to calculate the ECI and can be found in Table 6.3.

Categories in earthmoving	ECI including allowances for clay	ECI including allowances for sand	Unit
Supplying soil	1.7405	3.0758	ECI/m^3
Applying soil	0.1481	0.1444	ECI/m^3
Excavating soil	0.1037	0.1000	ECI/m^3
Disposing soil	0.8924	0.4216	ECI/m^3

Table 6.3: Values for the ECI for different processes in a dike reinforcement project. The values are including allowances, which is an uncertainty margin since a LCA expert did not look at this project.

In DuboCalc the ECI is expressed in MKI, the dutch word for ECI and shows some kind of environmental impact points. The soil is transported by axle and not by ship in Table 6.3 and these values will be implemented in this research. In Equation 6.2, environmental cost indicator model for this reseach is displayed and the ECI value is calculated per meter dike in length direction.

$$\begin{aligned} \text{ECI} = {} & 1.7405 \cdot \text{Qsc} + 3.0758 \cdot \text{Qss} \\ & + 0.1481 \cdot \text{Qac} + 0.1444 \cdot \text{Qas} \\ & + 0.1037 \cdot \text{Qec} + 0.1000 \cdot \text{Qes} \\ & + 0.8924 \cdot \text{Qdc} + 0.4216 \cdot \text{Qds} \end{aligned} \tag{6.2}$$

where:

Qsc is the quantity clay that is supplied from an external location [m^3];

Qss is the quantity sand that is supplied from an external location [m^3];

Qac is the quantity clay that is moved from a project depot to the place of processing and is processed within the project [m^3];

Qas is the quantity sand that is moved from a project depot to the place of processing and is processed within the project [m^3];

Qec is the quantity clay that is excavated from the dike and is brought to the project depot [m^3];

Qes is the quantity sand that is excavated from the dike and is brought to the project depot [m^3];

Qdc is the quantity clay that is excavated from project depot and is moved to an external location [m^3];

Qds is the quantity sand that is excavated from project depot and is moved to an external location [m^3];

All the soil quantities in Equation 6.2 can differ and that is why they are all taken apart. The quantities indicated in Equation 6.2 should be as small as possible in order to have a sustainable project.

The formula in Equation 6.2 will serve as the ECI model in this research. The input parameters of this model are the same as the input parameters for the surrogate model for macro-stability, except for the input parameter water height. The change in soil volume and the shape of the dike and the ditch lead to movement of soil and may lead to supply and removal of soil and this influences the ECI value (output parameter ECI model). The more the dike and ditch change, the greater the ECI, leading to a higher environmental impact.

A wider dike and a shallower ditch at a certain distance from the dike is the most optimal situation for the macro stability of the inner slope of a dike. Changing the dike and ditch geometry too much can be favorable for the macro-stability, but unfavorable for the ECI-value.

In this research, the dikes already exist and a new dike geometry is designed based on the old one. When calculating the ECI value, the difference in soil volume between the old and new dike geometry (after dike reinforcement), including the ditch, is taken. Moving soil and adding or removing soil also has an impact on the ECI. This is further explained in chapter 7.

6.5 Costs of the dike

The Standaardsystematiek voor Kostenramingen (SSK) is used for cost estimates in projects. SSK is a method developed by CROW and is a conceptual framework and a framework for project division. SSK provides a clear insight into what is and is not included in a project and improves the communication about the financial management in projects. The CROW does not provide any guidelines in the SSK for the unit prices and the percentages that are considered in dike reinforcement projects. These are based on the experience of W+B.

The schematization of the dikes that are considered in this research are not at a detailed level, the dikes are coarsely schematized. According to SSK, the costs that are considered in this research is the sum of the direct costs, the indirect costs, and the contingencies. The direct costs are all the effort and material supply involved in certain activities. The indirect costs are non-reoccurring costs like mobilization/demobilization, site facilities, site management, general costs. Sometimes indirect costs are referred to as 'contractors overhead'. The contingencies are the uncertainties involved in the timeline of the construction, the project's costs or outcome. The total construction costs are divided into direct costs, indirect costs, and the contingencies and how each part of the costs is calculated can be observed in Equations 6.3, 6.4, 6.5, and the total construction costs in Equation 6.6. All mentioned costs are excluding VAT and the costs are per meter dike in length direction.

$$\text{Direct costs (Dc)} = (Qc \cdot Uc) + (Qs \cdot Us) \tag{6.3}$$

where:

- Qc is the quantity clay that is needed to reinforce the dike [m^3];
- Uc is the unit price for clay [€/m^3];
- Qs is the quantity sand that is needed to reinforce the dike [m^3];
- Us is the unit price for sand [€/m^3];

The unit price of the different soils is divided into several parts. How the unit price is constructed for the different soil types can be seen in the Table 6.4.

Components direct costs	Value for clay	Value for sand	Unit
Supply soil	19	12	€/m^3
Transport and loading soil in depot	5	5	€/m^3
Excavate soil from depot and transport to working platform	2	2	€/m^3
Processing soil in dike core	2.5	2	€/m^3
Unit price soil	28.5	21	€/m^3

Table 6.4: Components of the direct costs in order to calculate the unit price for the different soils, excluding VAT.

The direct costs schematized in Equation 6.3 are without considering allowances. To include the allowances in the direct costs, the direct costs should be multiplied by 1.15; 15% of the calculated direct costs are added to the total direct costs in order to account for uncertainties in the costs that needs to be further detailed.

The indirect costs consists of non-reoccurring costs, site facilities, site organisation, general costs, profit, and risks. In Equation 6.4, the indirect costs are split into 3 parts in order to make the indirect costs more clear. Indirect costs part 1 consists of the non-reoccurring costs, site facilities costs, and the site organisation costs. Indirect costs part 2 consists of general costs and indirect costs part 3 consists of profit and risks. The formula to calculate the total indirect costs can be observed in Equation 6.4 and how the different components of the total indirect costs are constructed can be seen in the Table 6.5. All the percentages that are used in Table 6.5 are derived from W+B's own experience.

$$\begin{aligned}\text{Total indirect costs (TIDc)} = \text{Indirect costs part 1 (IDc1)} + \text{Indirect costs part 2 (IDc2)} \\ + \text{Indirect costs part 3 (IDc3)}\end{aligned} \tag{6.4}$$

Components indirect costs	Value for clay	Unit
Non-reoccurring costs (NRc)	0.01 ·DCa∗	€
Site facilities (SFc)	0.02 ·DCa∗	€
Site organisation (SOc)	0.12 ·DCa∗	€
Indirect costs part 1 (IDc1)	NRc + SFc + SOc	€
General costs (Gc)	0.08 ·(DCa ∗ +IDc1)	€
Indirect costs part 2 (IDc2)	Gc	€
Profit (Pc)	0.03 ·(DCa ∗ +IDc1 + IDc2)	€
Risks (Rc)	0.02 ·(DCa ∗ +IDc1 + IDc2)	€
Indirect costs part 3 (IDc3)	Pc + Rc	€
Total Indirect costs (TIDc)	NRc + SFc + SOc + Gc + Pc + Rc	€

Table 6.5: Components of the indirect costs in order to calculate the total indirect costs in Equation 6.4, excluding VAT.

*DCa are the direct costs including allowances.

The last component of the total construction costs are the contingencies. The input for calculating the contingencies is the direct costs and the indirect costs and the formula for the contingencies can be seen in Equation 6.5. 15% of the sum of the direct costs and the total indirect costs are the contingencies and this percentage is derived from W+B's own experience.

$$\begin{aligned}\text{Contingencies (Cc)} = 0.15 \cdot (&\text{Direct costs including allowances (DCa)} \\ &+ \text{Total Indirect costs (TIDc)})\end{aligned} \tag{6.5}$$

All things considered, the total construction costs are calculated in Equation 6.6 in euro's. The formula in Equation 6.6, together with the equations that are needed for this formula, will serve as the cost model in this research. The input parameters of this model are the same as the input parameters for the surrogate model for macro-stability, except for the input parameter water height. The change in soil volume and the shape of the dike and the ditch lead to movement of soil and may lead to supply and removal of soil and this influences the costs (output parameter cost model). The more the dike and ditch change, the more costs are incurred.

A wider dike and a shallower ditch at a certain distance from the dike is the most optimal situation for the macro stability of the inner slope of a dike. Changing the dike and ditch geometry too much can be favorable for the macro-stability, but unfavorable for the costs.

$$\begin{aligned}\text{Total construction costs} = &\text{Direct costs including allowances (DCa)} \\ &+ \text{Total Indirect costs (TIDc)} + \text{Contingencies (Cc)}\end{aligned} \tag{6.6}$$

The total construction costs in Equation 6.6 are the costs when the dike is built from scratch. However, in this research, the dikes already exist and a new dike geometry is designed based on the old one. When calculating the costs, the difference in soil volume between the old and new dike geometry (after dike reinforcement), including the ditch, is taken. Moving soil and adding or removing soil also has an impact on the costs. This is further explained in chapter 7.

6.6 The ditch

Supplying, excavating, and transporting soil is not only necessary for the dike core, but is also needed to change the ditch geometry. As mentioned in section 6.2, the critical nutrient load is dependent on the depth of the ditch. In order to optimize a dike design, the depth of the ditch can also change in order to meet the ecological requirements. On top of that, the distance between the ditch and the inner toe of the dike is also a varying parameter in the optimization. When the ditch geometry is changed, soil has to be excavated from the ditch to deepen the ditch or soil has to be supplied to the ditch the make the ditch less deep. This soil volume has to be added to the considered soil volumes in the costs and eci calculation. To keep the calculation for the amount of soil coming from or going to the ditch simple, it is assumed that the soil that is excavated from the ditch is clay since in most cases the ditch lies in the cohesive layer. In reality, the amount of clay and sand should be calculated when soil has to be excavated for the ditch. Since the amount of excavated soil is often a small quantity and a detailed calculation

of amount of excavated soil can be computer demanding, it is assumed that the soil that is excavated from the ditch is clay. When more soil is needed for the ditch, sand will be supplied since this a cheaper solution and is also done in reality.

6.7 Conclusion

This chapter elaborates the models for the other engineering aspects that are considered in this research. All the equations that are shown in this chapter are implemented in Python in order to perform the multi criteria analysis (MCA) there. The PCDitch metamodel is already computed in Python. The last section is about how the movement of soil volumes regarding the ditch is implemented in the cost and ECI model since this can be a rather complicated calculation.

In each section of this chapter, the relation between the surrogate model for macro-stability of the inner slope of dikes and the models for the different engineering aspects are elaborated. A clear overview of the shared input parameters between the different models for the engineering aspects and the surrogate model can be observed in Figure 6.2. The input parameters dike height, crest width, inner slope, inner berm width, distance between the ditch and the inner toe of the dike, and the depth of the ditch are highlighted with a red box since these input parameters of the surrogate model for macro-stability will be the varying parameters in the MCA.

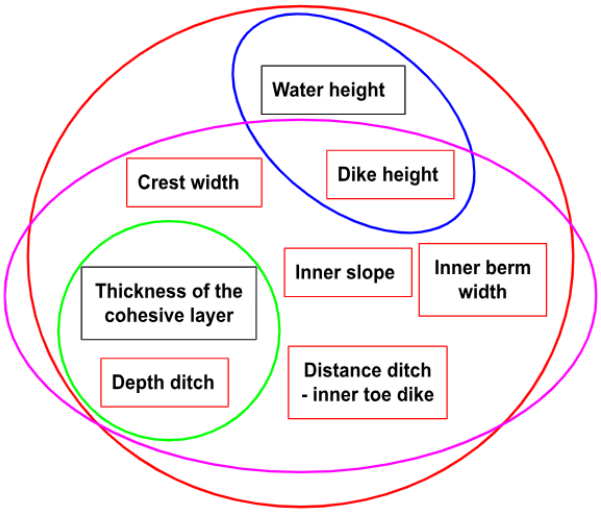


Figure 6.2: Plot of the shared input parameters between the ecology, wave overtopping, ECI, and cost models and the surrogate model for macro-stability. The red circle contains all the considered input parameters for the MCA and these are the input parameters for the surrogate model for macro-stability of the inner slope of dikes. The blue ellipse shows the shared input parameters between the wave overtopping model and the surrogate model. The green circle indicates the shared input parameters between the ecology model and the surrogate model. The purple ellipse shows the shared input parameters between the ECI and the surrogate model and the cost model and the surrogate model.

Chapter 7 Multi criteria analysis (MCA)

7.1 Introduction

In the previous chapter, the models are explained for the other engineering aspects that are considered in this research. These models will be combined together with the surrogate model for macro-stability in a Multi Criteria Analysis (MCA) in order to find the most optimal dike geometry. Since all the models are clear, the MCA can be elaborated in order to get the most optimal dike design.

7.2 Multi criteria analysis

A multi criteria analysis is performed in order to get the most optimal dike design. A MCA is a method that is used to support decisions in complex issues. For a specific case, multiple criteria should be gathered that belong to certain objectives. These criteria needs to be standardised in order to have the different criteria on the same level. Some criteria are more important than others, that is why the criteria are weighted. After doing all these steps, the different alternatives for the specific case can be ranked from best to worst alternative. The best alternative is selected to solve the complex issue.

The criteria for an optimal dike design are the costs of the dike, the ECI, and Ecology model in this research. The dike designs won't be optimized based on the wave overtopping model and the surrogate model for macro-stability of the inner slope. These two criteria forms a constraint for the MCA. The considered dike geometries should have a value for the FoS that is equal to or higher than 1.2 and the amount of wave overtopping should be equal to or less than 10 L/m/s, otherwise these dike geometries can lead to dike failure. The weights for the MCA can be adjusted depending on the priorities in different dike reinforcement projects in order to meet the wishes of the different water boards and Rijkswaterstaat, so they are not fixed. The MCA is performed in Python and to perform the MCA certain steps must first be taken. The needed steps are written down below and a clear overview can be observed in Figure 8.1.

1. First the five models, surrogate model for macro-stability inner slope dike, ecology, costs, wave overtopping, and ECI model, are implemented in the Python file. The surrogate model and the wave overtopping model are needed for defining constraints for the MCA and the ecology, costs, and ECI model are the criteria inside the MCA.

2. The criteria needs to be standardised in order to have the different criteria on the same level. Standardizing the criteria is done with the minimum and maximum value that each model for a certain criteria can generate. The scaling formula can be observed in Equation 7.1.

   The minimum value for the ECI and cost model is 0. This is when the dike and ditch

geometry do not have to change. The maximum value for these models is that the dike and ditch geometry have to undergo the maximum change that is possible. This means that the dike height, crest width, inner slope, and the width of the berm changes from the minimum value to the maximum value from the range that can be observed in Table 3.8 and when the ditch changes from the biggest possible ditch to the smallest and the distance between the ditch and the inner toe of the dike is changed from the minimum value to the maximum from Table 3.8. Since the minimum value for the ECI and cost model is 0, the result from these criteria can also be multiplied by a scaling factor. The formula for the scaling factor is shown in Equation 7.2.

The minimum and maximum value for the Ecology model is derived by calculating the critical nutrient load for different water depths inside the ditch and sediment type. A distinction has been made between clay and sand as a sediment type since the size of the critical nutrient load varies enormously for sediment type clay and sand, as can be observed in Figure **??**. The minimum and maximum value for clay and sand as sediment type is saved for scaling the result from the Ecology model.

3. After getting the values for the scaling formula, the objective function is defined. The formula for the objective function is shown in Equation 7.3. The score for the cost and ECI model is calculated by calculating the cost and the ECI value for the current dike geometry first. After that, the value for the cost and ECI model is calculated for the new dike geometry. The score for the cost and ECI model is derived by taking any difference between the new dike and ditch geometry and the current geometry. Every movement of soil and any amount of soil that needs to be supplied or removed from the dike and ditch result in higher costs and a higher ECI score. The score for the cost and ECI model is multiplied by the scaling factor associated with the corresponding model.

   The score for the ecology model is derived by calculating the critical nutrient load for the new dike geometry with the ecology model and put this value as X_{original} in Equation 7.1 to calculate the scaled score.

   The scores for the different criteria/models are multiplied by the weight that are chosen by the client. The minus sign for the ecology part in the objective function is explained in step 7.

4. For the integrated dike design, another aspect of the cost of a dike has been added to the MCA. If a dike needs to be made wider, then the dike can sometimes be made so wide that a part of the dike enters the ground of another parcel owner. If this is the case, then extra costs are incurred to pay the parcel owner. The water board applies 80,000 euros per ha and this is €8 per extra meter of dike. This aspect is an fixed extra input parameter in the objective function and varies per location of the dike. If the width of the new dike exceed it's own parcel, then additional costs are calculated and added to the current costs for the new dike design.

5. In step 3 and 4 is the objective function defined. After defining the objective function, the constraints needs to be defined. If a constraint is not met, a penalty is counted on top of the outcome of the objective function to make sure that every constraint is met. The penalty is equal to 10 when one constraint is not met. This penalty should be high compared to the outcome of the objective function, otherwise the constraints are not that important to the objective function and the constraints will be easily exceeded. The outcome of the objective function is in the order of 10^{-1} and thus is a penalty of 10 relatively high.

In this research, 3 constraints exist:

Constraint 1: The result from the wave overtopping model should be equal or lower than 10 L/m/s. By applying this constraint, the the water level is always lower than the crest level and therefore the constraint that the water level should be equal or lower than the crest level of the dike does not have to be implemented.

Constraint 2: This is a hydrological precondition, the wetted cross-sectional area of a ditch should be equal or higher than 0.625 m^2. This constraint belongs to a ditch with the following standard dimensions for water boards: water depth is 0.5 m, slope ditch is 1:1.5, and the width of the bottom of the ditch is 0.5 m.

Constraint 3: The FoS of the dike should be equal to or higher than 1.2. The value for the FoS is derived from the surrogate model for the macro-stability of the inner slope of a dike.

6. The input parameters for the different models are defined in this step. Some input parameters are fixed and some will vary when the dike geometry will change. The input parameters that will vary in order to optimize the dike design are: dike height, crest width, the inner slope of the dike, the width of the berm, the depth of the ditch, and the distance between the ditch and the inner toe of the dike. The parameters that are fixed, but can be changed dependent on the location of the dike and the client's wishes, are: the weights in the objective function, the wave height, the angle of wave attack, the outer water level, the thickness of the cohesive layer underneath the dike, the type of soil in the dike core, and the width of the dike that is on the parcel of another owner. The values for the fixed parameters can just be filled in and for the varying parameters, an initial guess is made by filling in the parameters that corresponds to the current dike geometry. For the varying parameters, the bounds are defined which corresponds with the values from Table 3.8.

7. In order to optimize the dike designs, an optimization function is defined. This optimization function is used to iterate over possible improved dike and ditch geometries that are close to the current dike and ditch geometry. In this way, the MCA is not performed by weighing different dike and ditch geometries against each other by hand, but the optimization function returns the most optimal dike and ditch geometry. It is important that the optimize function will find the global minimum, instead of the local minimum.

   Another important aspect of the optimization function is the initial guess. The initial guess can be implemented in some optimization functions, but not all. For this research, it is important that the initial guess can be implemented in the optimization function since the initial guess is the current dike geometry.

   The last important aspect is the constraints in the initial guess. In some cases, the current dike geometry does not meet the set constraints of the MCA. When the current dike geometry does not meet all the constraints, the objective function returns a high score for the current dike geometry since penalties are added on top of the outcome of the objective function. When this is the case, the optimization function is looking for optimized dike and ditch geometries close to the current geometry, the initial guess, and thus also look for dike geometries that exceed the constraints. This is not the intention. Therefore the initial dike geometries should be updated first in order to get closer to meeting the constraints. The following will explain exactly how the geometries should be updated in order to get closer to meeting the constraints.

Constraint 1: When the amount of wave overtopping is larger than 10 L/m/s, exceeding Constraint 1, the dike height needs to be increased. As can be observed from Figure 6.2, the dike height is the only influencing input parameter in the objective function for the wave overtopping model and therefore the value for this input parameter needs to be changed in order to get closer to meeting Constraint 1.
Constraint 2: When the wetted cross-sectional area of the ditch is smaller than 0.625 m^2, the depth of the ditch should be increased in order to get closer to meeting Constraint 2.
Constraint 3: When the value for the FoS is lower than 1.2, the inner berm width should be increased first in order to get closer to meeting Constraint 3. Changing the width of the inner berm has the largest impact on increasing the value for the FoS since the inner berm width has strongest positive correlation with the FoS. When the inner berm width can not be increased further, because the width of the inner berm is equal to the maximum value of the inner berm width in Table 3.8, the inner slope of the dike should be increased in order to get closer to meeting Constraint 3. The inner slope of the dike is also strongly positive correlated with the FoS, but less strong compared to the width of the inner berm. The initial dike geometries are changed based on the three constraints and when the geometries are getting closer to meet the constraints, these geometries are implemented in the objective function. This change in the geometries has been processed in the objective function in order to calculate what the impact is on the score for the costs, ecology, and ECI and this impact should be taken into account in the MCA.

Multiple optimization functions are available and many have been tried in this research, but the scipy.optimize.basinhopper() function works the best for optimizing dike designs. The input for this function is the objective function, the initial guess, the bounds of the of varying parameters, the number of iterations, the temperature, the step size, and method. The bounds of the varying parameters are the minimum and maximum value of the input parameters and can be observed in Table 3.8. A higher value for the temperature parameter can help the optimization function to escape local minima. A higher value for the step size parameter helps the objective function to explore the solution space. The method parameter specifies the algorithm that is used to perform local minimizations. Basinhopper searches for the global optimum by combining random jumps that explore the solution space with local minimization to find the precise minimum. For an optimal solution, the results from the cost and the ECI model are low and the result from the ecology model is high. That is why a minus sign is in front of the ecology part in the objective function. The number of iterations is set to 200, the temperature to 5, step size 1, and the method SLSQP. The method SLSQP is chosen since this method can be used when complex constraints are incorporated in the objective function, including non-linear (in)equality constraints.

$$X_{\text{scaled}} = \frac{X_{\text{original}} - \text{min. result model}}{\text{max. result model} - \text{min. result model}} \tag{7.1}$$

$$\text{Scaling factor} = \frac{1}{\text{max. result model}} \tag{7.2}$$

$$\text{Objective function} = (w_{\text{c}} \cdot s_{\text{c}}) - (w_{\text{e}} \cdot s_{\text{e}}) + (w_{\text{eci}} \cdot s_{\text{eci}}) \tag{7.3}$$

where:

w_c is the weight for the cost model [-];

s_c is the score of the cost model [-];

w_e is the weight for the ecology model [-];

s_e is the score of the ecology model [-];

w_{eci} is the weight for the ECI model [-];

s_{eci} is the score of the ECI model [-];

A clear overview of the MCA procedure can be observed in Figure 8.1. In Figure 8.1, the loop of the updated dike and ditch geometry can be observed. When the initial dike and ditch geometry does not meet all constraints, then the optimizer will only search for optimized dike and ditch geometries that do not meet all constraints. Therefore the dike and ditch geometry should be updated, to have optimized geometries without exceeding the constraints.

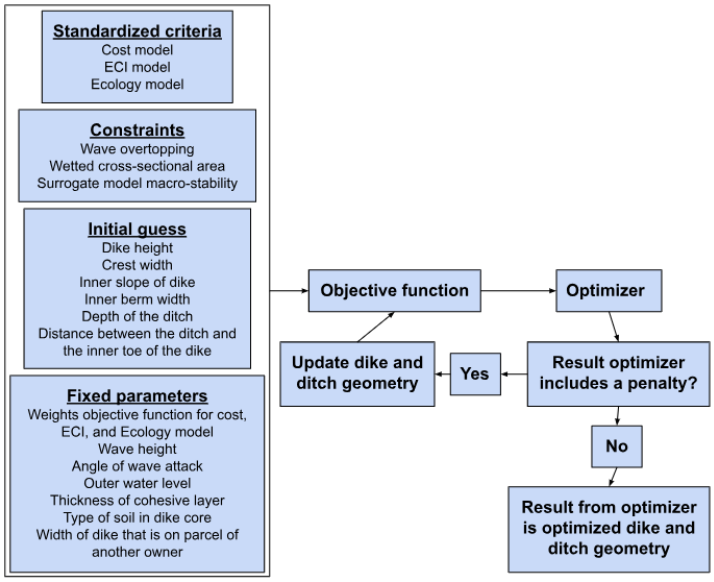


Figure 7.1: MCA procedure to derive the optimal dike and ditch geometries.

7.3 Conclusion

This chapter shows what a MCA is and how it is used to get the most optimal dike designs. It is important that the optimize function returns a global minimum instead of a local minimum.

Initially, the function scipy.optimize.minimize() with multiple solvers was used since this optimization function worked for the Simple model. When the amount of input parameters increased, the most optimal result was the same, regardless of the initial guess. This meant that another optimization function must be implemented. After trying the several optimization algorithms for finding the global minimum with multiple initial guesses, the function scipy.optimize.basinhopper() returns one optimized result for one dike reinforcement project.

Chapter 8 Results integrated model

In chapters 6 and 7, the models for the other engineering aspects and the MCA procedure is elaborated. In this chapter, results from the MCA are shown. The impact of changing the weights of the objective function on optimizing dike geometries is explained.

8.1 Varying weights

In order to see what the impact of the different weight factors is on the optimized dike geometries, one weight factor increases and the other weight factors are kept similar. The sum of the weight factors for ecology, costs, and ECI always equals 1. The initial dike geometry is the minimum values for the varying input parameters from the ranges that can be observed in Table 3.8. For the fixed parameters in the objective function, the average values are used. This means that Hmo equals 0.75 m, β equals 40 °, the outer water level is 4.25 m, the thickness of the cohesive layer is 4 m and the dike core contains clay. The results can be observed in Figure 8.1.

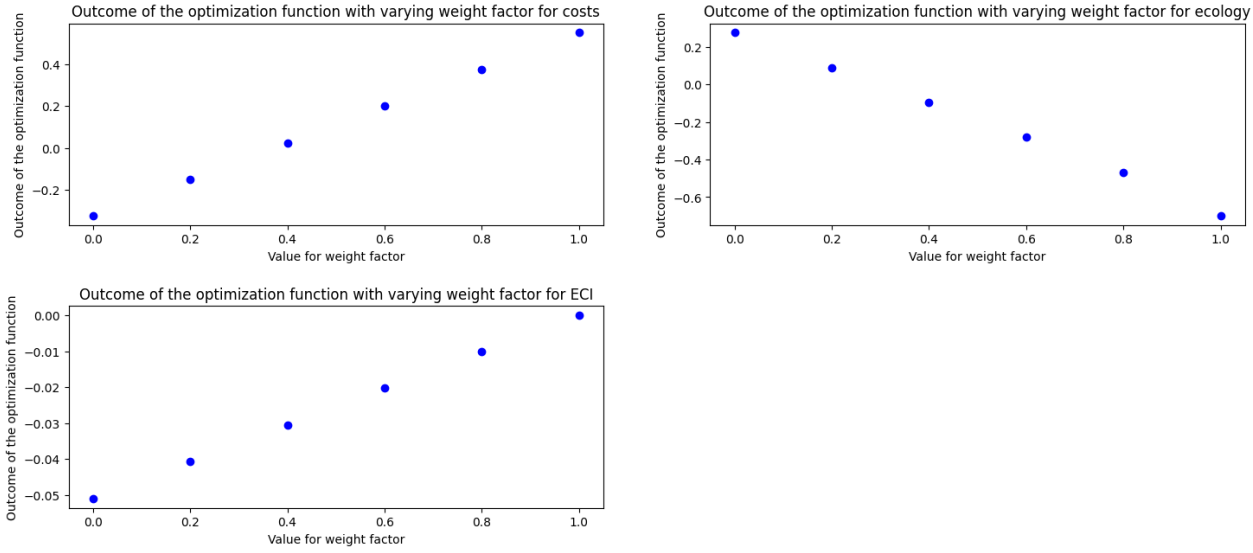


Figure 8.1: Varying weights in the objective function against the outcome of the objective function.

When the varying weight is set to 0.2, the other weight is set to 0.4 (0.8/2) to create a total weight of 1.0. This means that when the varying weight is set to 0, this criteria is not taken into account in the MCA, and when it is set to 1.0, the other criteria are not taken into account. The negative value of the objective function is derived from the impact of ecology criteria, which

is explained in the previous section. When the impact of the ecology model is not big on the outcome of the objective function, then the outcome of the objective function is positive.

The MCA is performed multiple times with the same initial dike geometry and the same weights for the criteria in order to see if the optimization function returns the optimized result. A simple calculation can be performed in order to check one value from Figure 8.1, if the calculated outcome is the same as the resulted outcome.

When the weight for the ecology criteria is set to 1.0, the weights for the costs and ECI are equal to 0. This indicates that the optimization function should look for highest value for the critical nutrient load. In Figure 8.2, the depth of the ditch is plotted against the critical nutrient load. To meet the constraint of the minimum wetted cross-sectional area of a ditch, the depth of the ditch should be bigger than values left from the green line in Figure 8.2. The optimal depth of the ditch derived from the optimization function is 0.96 m, which is indicated by the red line in Figure 8.2. Looking at the area on the right side of the green line, the red line crosses indeed the maximum of the curve, which means that the returned value from the optimization function is indeed the optimized result.

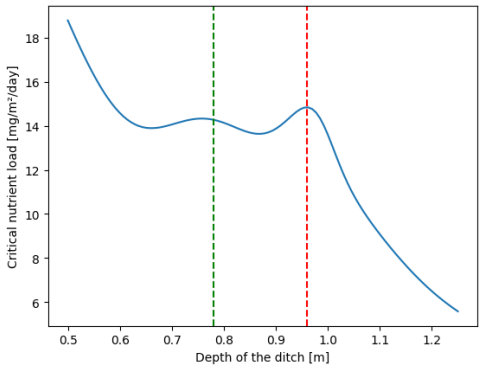


Figure 8.2: Plot of the depth of the ditch vs critical nutrient load for sediment type clay.

The optimized dike geometries from Figure 8.1 are more or less the same, but results in another value for the optimization function due to the different weights. The optimized dike geometries are the same due to the implementation of the changed intial dike geometries, which is explained in step 7 in section 7.2.

The input parameters for initial dike geometry that is considered for the plots in Figure 8.1 were the minimum values from the ranges in Table 3.8. This means that the considered constraints in this research were not met. The optimization function is strongly bounded to the initial dike geometry and when the initial dike geometry does not meet the constraints, the optimization function searches for an optimal dike geometry that does not meet all constraints, despite the fact that penalties are added. To make sure that the optimization function searches for optimal dike geometries that meet all constraints, the initial dike geometry should be updated. This means that the initial dike geometry is changed first and then implemented in the optimization function in order to get dike geometry which are closer to meeting all constraints. This updated version is implemented in the optimization function. However, after the optimization, the returned dike geometry is more or less similar to the updated version of the dike geometry. This makes sense, because the updated dike and ditch geometry just meets all constraints, which is the most optimal dike geometry since the amount of wave overtopping is

close to 10 L/m/s, the wetted cross-sectional are of the ditch is the minimum value, and the FoS is just 1.2.

8.2 Conclusion

In this chapter, a Multi Criteria Analysis is performed by having an initial dike geometry that is optimized by a deterministic approach and an optimization function. The deterministic approach makes sure that the initial dike geometry is updated first, otherwise the optimization function will look for dike geometries, which will exceed the constraints. In many cases, this deterministic approach is the optimal dike geometry since all contraints are just met, which means that the optimization function is not needed. Only when the influence of the Ecology model is high, the dike geometry, which is returned by the optimization function, will sometimes deviate from the dike geometry derived from the deterministic approach.

Chapter 9 Conclusions

In the upcoming year, many dikes need to be reassessed as a result of Hoogwater Beschermingsprogramma HWBP and stricter requirements regarding macro-stability of dikes. The failure mechanism macro-stability is very important since this failure mechanism is often cost-determining in dike reinforcement projects. To achieve the goal of the HWBP by 2050, many stability calculations need to be performed in a short amount of time. Current dike reinforcement projects check the failure mechanisms, regarding the integration of a flood defense in a certain area and other design aspects. This results in testing and designing a lot of dikes and this costs a lot of money and takes a lot of time and therefore it often proves difficult to deliver good integral designs. In order to improve this, this research has been set up.

To achieve this objective, the following research question and sub-questions will be answered:

RQ:How can a surrogate model for inner slope stability of a flood defense be created for use in combination with other (machine learning) models to generate conceptual flood defenses, making it possible to optimize the design using an interdisciplinary MCA?

> **SQ1**: Which input and output variables of a stability model does the surrogate model need to train the machine learning on certain combinations of input and output variables for accurate prediction of the model outcome for macro-stability of dikes in the Netherlands?
>
> **SQ2**: Which sampling method is needed to have enough information about dike stability captured in the dataset to train, validate, and test the surrogate model?
>
> **SQ3**: Which type of ML algorithms perform best for emulating macro-stability models best?
>
> **SQ4**: What are the equations / models for the other relevant aspects for dike design and what do these equations / models look like?
>
> **SQ5**: How can the macro-stability surrogate model be combined with the equations/models for the other relevant engineering aspects related to dike design to get to an optimal design?

This chapter will answer the sub-questions based on the conclusions made in the previous chapters. The first part of this research answers sub-questions 1 to 3 and the second part answers sub-questions 4 and 5. The answer to the main research question will derive from these sub-questions.

9.1 Macro-stability model

In this section, the surrogate model for the macro-stability of dikes is developed. This is done by answering sub-questions 1 to 3. First, simple dike geometries were considered in order to see

if a surrogate model can be developed based on D-Stability calculations and later complex dike geometries were considered.

SQ1: Which input and output variables of a stability model does the surrogate model need to train the machine learning on certain combinations of input and output variables for accurate prediction of the model outcome for macro-stability of dikes in the Netherlands?

Initially, for the simple model, five varying parameters are chosen to create simple dike geometries. These parameters are the height, the crest width and the inner slope of the dike, the thickness of the cohesive layer underneath the dike and the outer water level. These parameters serve as input parameters for ML model. The output parameter is the FoS.

After the first training, the models' performances were already quite good, having a MSE value of $2.7 \cdot 10^{-3}$, MAE value of $4.7 \cdot 10^{-2}$, and a R^2-score of 0.84 for the best ML model sampled with Random Sampling. The mean value for the FoS for the complete dataset is lower than the FoS for which dikes are considered stable. In this research, dikes are stable when the FoS is equal to or larger than 1.2 to have some margin. In order to have a representative training dataset with a sufficient amount of unstable and stable dikes, the values for the FoS needed to be increased.

To increase the FoS, the ranges of the varying input parameter were changed based on data investigation. The models' performances were improved by adding another input parameter, the type of soil in the dike core. After plotting the residuals for the FoS, two normal distributions could be observed belonging to the two different dike cores. This knowledge was kept away in the training process.

After the second training, the models' performances were quite accurate, having a MSE value of $2.2 \cdot 10^{-3}$, MAE value of $1.1 \cdot 10^{-2}$, and a R^2-score of 0.99 for the best ML model sampled with Random Sampling. The best ML model is the Support Vector Regression. 95% of the predicted FoS having a maximum deviation of 0.04 from the actual FoS, which is an accurated prediction of the FoS at an early stage of dike design.

Since the surrogate model for the simple dike geometries is accurate, a surrogate model for complex dike geometries could be made. The amount of varying input parameters were extended by three parameters, namely the inner berm width, the distance between the ditch and the inner toe of the dike, and the depth of the ditch. The dike core parameter was also taken into account in the varying input parameters since this parameter was needed to have improved models' performances. The sand core dikes were made more realistic by adding a clay cover of 0.9 m on top of the sand core. This clay cover was included in the dike height. The output parameter of the surrogate model for complex dike geometries is the FoS.

After training the ML models, the models' performances were already quite good, having a MSE value of $6.0 \cdot 10^{-4}$, MAE value of $1.2 \cdot 10^{-2}$, and a R^2-score of 0.95 for the best ML model. These values for the validation metrics were comparable with the values for the validation metrics for the simple dike geometries. The best ML model is again the Support Vector Regression. 95% of the predicted FoS having a maximum deviation of 0.05 from the actual FoS, which is again an accurated prediction of the FoS at an early stage of dike design.

For visualisation purposes, the ML models were also trained on more output parameters to predict the slip circle. Since the Limit Equilibrium Method is Uplift-Van for assessing the slope stability for complex dike geometries, eight additional output parameters were needed in order to predict the slip circle. The minimum amount of additional output parameters was 6 in order

to approach the slip circle and the additional output parameters are the x- and y-coordinates of the active and passive slip circle and the radii of the active and passive slip circle. The same input parameters are considered when the ML models were trained only on the output parameter FoS for the complex dike geometries.

After training the ML models, the models' performances were bad in comparison with the models' performances for the complex dike geometries with the FoS as only output. The averaged best MSE value was 0.0032, MAE value 0.0313, and R^2-score 0.6531. The best model is the Random Forest Regressor. The bad performance was mainly caused by y-coordinate of the passive slip circle and consequently the radius of the passive slip circle. Very great values for the relative y-coordinate of the passive slip circle were the cause of the bad performance, because these values formed outliers for the dataset. The great relative values for the y-coordinates of the passive slip circle were the result of thin thicknesses of the cohesive layer underneath the dike.

In order to improve the models' performances, more samples were taken for a low value of the thickness of the cohesive layer. However, this didn't improve the models' performances. In order to improve the models' performance, the ML algorithms might need more information about developing the slip circle, which should be translated into an input parameter. Adding more information to the training dataset resulted in a higher models' performance for the Simple model too.

In summary, for the simple dike geometries the input parameters are the dike height, the crest width, and the inner slope of the dike, the thickness of the cohesive layer, the outer water level, and the soil in the dike core. The output parameter is the FoS. For the complex dike geometries, the input parameters are the dike height, the crest width, and the inner slope of the dike, the thickness of the cohesive layer, the outer water level, the width of the inner berm, the distance between the ditch and the inner toe of the dike, the depth of the ditch and the soil in the dike core. The output parameter is the FoS. For visualisation purposes, the output parameters are the FoS, the x- and y-coordinates of the active and passive slip circle, and the radii of the active and passive slip circle.

SQ2: Which sampling method is needed to have enough information about dike stability captured in the dataset to train, validate, and test the surrogate model?

The input parameter sets for the simple dike geometries are derived by sampling with the methods Random Sampling and Latin Hypercube Sampling from the uniform distributions for the input parameters. This means that 2 times 10.000 input parameter sets are created, with 10.000 input parameter sets sampled with Random Sampling and 10.000 input parameter sets sampled with Latin Hypercube Sampling. After the first training, the differences in models' performances between the sampling methods were small, but Random Sampling performs better for each ML model. When the input parameters are changed, a second training is performed. The difference in models' performances between the two different sampling methods are quite small. Random Sampling performs better on the values of MSE and MAE and Latin Hypercube Sampling on the values for the R^2-score. Since the differences between the values for the MSE and MAE are a bit smaller compared to the differences between the values for the R^2-score and a high R^2-score is important to capture the underlying relationships in the data, the Latin Hypercube Sampling method is the best method.

SQ3: Which type of machine learning model assesses macro-stability of dikes the

best?

The best ML models are already mentioned under sub-question 1 with the corresponding explanation. For the simple dike geometries, the Support Vector Regression was the best ML and this type of ML model was also the best model for the complex dike geometries with the FoS as output parameter. The Support Vector Regression model was not used for training on more output parameters since this ML model is typically used for single output regression problems. For the complex dike geometries with more output parameters was the Random Forest Regressor the best ML model.

9.2 Integrated dike design

In this section, the multi criteria analysis is set up and performed to get the optimized dike designs. This is done by answering sub-questions 4 and 5. The surrogate model for macro-stability is combined with the equations/models for the ecology, costs of the dike, environmental cost indicator, and wave overtopping in a MCA to get the optimized dike designs.

SQ4: What are the equations / models for the other relevant aspects for dike design and what do these equations / models look like?

The relevant aspects that are considered in this research for dike design are ecology, the costs of the dike, the environmental cost indicator (ECI), and wave overtopping. For each relevant design aspect, the equation/model is explained below.

- Ecology: The ecology model that is used in this research is a metamodel developed by W+B, STOWA, NIOO-KNAW, Netherlands Environmental Assessment Agency and WUR. It is based on a dataset with 360.000 runs by the PCDitch model. This model assesses water quality in ditches and show which measures have the most effect on improving the water quality in a certain ditch. The input parameters of this model are water depth in the ditch, the flow rate, and the sediment type. The output parameter is the critical nutrient load, which is the maximum amount of nutrients that the ditch can receive without negative effects on the ecological value of the ditch. The input parameters that vary are the water depth in the ditch and the sediment type. The water level inside the ditch is kept constant at a height of 0.4 m below ground level, but the depth of the ditch varies, which results that the water depth in the ditch also varies. The sediment type can vary, because the thickness of the cohesive layer underneath the dike can be thinner than the depth of the ditch, resulting in sand as sediment type. In other cases the sediment type is clay. The flow rate is constant and equal to 20 mm/d.
- Wave overtopping: Wave overtopping is schematized by an equation derived from the Technische Rapport Golfoploop en Golfoverslag bij Dijken from the Technische Adviescommissie voor de Waterkeringen. The equation that is used in this research is a simplification of the complete formula from this report and is the maximum of the complete formula. The input parameters of this equation are the free crest height above the still water line, the significant wave height at the toe of the dike, the influence factor for the roughness on the dike slope, the influence factor for the angle of wave attack on the dike slope, and the acceleration due to gravity. The output parameter amount of wave overtopping. The varying input parameters are the free crest height above the still water line, the significant wave height at the toe of the dike, and the angle of wave attack on the dike slope. The influence factor for the roughness on the dike slope is equal to 1.0 since it is a green dike.

The free crest height above the still water line is calculated without the impact of the wind since the dike geometries that are considered in this research are not location specific and is calculated by taking the difference between the dike height and the water level. The significant wave height varies between 0 and 1.5 m and the angle of wave attack between 0 and 80 °. The influence factor for the angle of wave attack on the dike slope can be calculated with the following formula: $1 - 0.0033 \cdot |\beta|$. The maximum allowable amount of wave overtopping is 10 L/m/s.

- Environmental Cost Indicator (ECI): The model that is set up for ECI makes use of the program DuboCalc. This is a tool developed by Rijkswaterstaat. DuboCalc has a library in which objects for a dike reinforcement project can be found under 'HWBP Dijkversterking PU-fase'. Under 'HWBP Dijkversterking PU-fase' values for the ECI can be found for supplying of soil, applying of soil, excavating of soil, and disposing of soil in a dike reinforcement project and these values can be observed in Table 9.1. A distinction is made between clay and sand and these values belong to a green dike that was considered with a lifespan of 50 years. The ECI is calculated by multiplying the quantities of clay and sand with the corresponding value in Table 9.1.

Categories in earthmoving	ECI including allowances for clay	ECI including allowances for sand	Unit
Supplying soil	1.7405	3.0758	ECI/m^3
Applying soil	0.1481	0.1444	ECI/m^3
Excavating soil	0.1037	0.1000	ECI/m^3
Disposing soil	0.8924	0.4216	ECI/m^3

Table 9.1: Values for the ECI for different processes in a dike reinforcement project. The values are including allowances, which is an uncertainty margin since a LCA expert did not look at this project.

- Costs of the dike: The Standaardsystematiek voor Kostenramingen (SSK) is used as basis for the cost estimate in this research project. The dikes that are considered in this research are coarsely schematized and according to SSK, the costs that are considered in this research is the sum of the direct costs, the indirect costs, and the contingencies. The direct costs are all the effort and material supply involved in certain activities. The indirect costs are non-reoccurring costs like mobilization/demobilization, site facilities, site management, general costs. Sometimes indirect costs are referred to as 'contractors overhead'. The contingencies are the uncertainties involved in the timeline of the construction, the project's costs or outcome. The unit prices and the percentages that belongs to these costs are based on experience of W+B. The costs are excluding VAT. A detailed elaboration of the cost estimation can be found in section 6.5.

SQ5: How can the macro-stability surrogate model be combined with the equations/models for the other relevant engineering aspects related to dike design to get to an optimal design?

A multi criteria analysis is performed in order to get an optimal dike design. In this MCA, the five models, macro-stability, wave overtopping, ecology, costs, and ECI will be combined. The macro-stability model and the wave overtopping equation will serve as constraints in the optimization function since a dike geometry should meet these requirements. When the FoS is

high, the dike geometry is not the most optimal one. The same applies to wave overtopping, when the dike is very high, waves won't go over the crest of the dike, but the dike geometry is not optimized. The ecology, costs, and ECI model will serve as criteria in the objective function. The input parameters for the optimization function are the objective function, constraints, initial dike geometry (is the initial guess), and the hyper parameters of the optimization function. The optimization function is scipy.optimize.basinhopper() which searches for the global minimum based on the initial dike geometry and returns the value of the objective function and the optimal dike geometry.

9.3 Research question

The answers to the sub-questions, from the previous section, will help answering the research question. In this section, a summary is made to answer the research question.

The surrogate model for the inner slope stability of a dike is created step by step. First, simple dike geometries were considered with as few parameters as possible in order to see if a ML model can assess the FoS. The input parameters for training the ML models on simple dike geometries are the dike height, crest width, inner slope, the thickness of the cohesive layer underneath the dike and the outer water level. Combinations of values for these five input parameters are created by the two sampling methods Random Sampling and Latin Hypercube Sampling. After deriving the input parameter set, the D-Stability calculations can be performed in order to get the FoS for the corresponding dike geometry. HWL datamodel, which is a python tool developed by Witteveen+Bos, is used to help running the D-Stability calculations as quickly as possible. After deriving the FoS for each input parameter combination, the whole dataset is normalized using MinMaxScaler and is split into a training, validation, and test dataset using train_test_split() from scikit-learn with a train_size of 0.8 and test_size of 0.2. Before training the final ML models, hyper parameter tuning will take place in order to optimize the hyper parameters for the concerned ML model to improve the models' performances. Random Search Cross-Validation and Grid Search Cross-Validation are used to find the best hyper parameters for the corresponding ML model with cv-values that equals 5 and 10, resulting in 64% training and 16% validation and 72% training and 8 % validation dataset. After hyper parameter tuning, the final ML models can be trained and can be assessed by the three validation metrics, MSE, MAE, and R^2-score. In this way, a surrogate model for the macro-stability of inner slope is created. Data investigation, resampling, retraining is needed to increase the surrogate performance. The surrogate model for complex dike geometries is developed in the same way as the simple dike geometries. The only difference is that the complex dike geometries do contain more input parameters, namely the inner berm width, ditch depth and the distance between the ditch and the inner toe of the dike. Some snippets of the code has been added to the Appendix B in order to see how the functions are implemented in python.

The surrogate model for inner slope stability of a dike is combined with the models/equations for ecology, ECI, costs, and wave overtopping via a multi criteria analysis. The wave overtopping equation and the macro-stability model serve as constraints for the MCA and the ecology, ECI, and costs model serve as criteria. The criteria is normalized and included in an objective function. The scores for the criteria are multiplied by a weight factor that varies depending on the stakeholders in the dike reinforcement project. To get optimal dike designs, the objective function is minimized in an optimization function. The optimization function that is used in this research is scipy.optimize.basinhopper(). Another constraint that is set in this research are the minimum wetted cross-sectional area in the ditch. The last element that is included in the

optimization process is the parcel owner. When the dike should be extended, the dike can enter ground of another parcel owner. If this is the case, extra costs of €8,- per extra meter dike should be taken into account. All constraints, the objective function, the initial dike geometry (is the initial guess), and the hyper parameters of the optimization function are implemented in the optimization function. The optimization function searches for the global minimum based on the initial dike geometry and returns the value of the objective function and the optimal dike geometry.

Chapter 10 Recommendations

During this research, some recommendations for further research were found. In addition to the recommendations for further research, some recommendations are about the results derived from this research.

- **Simple models:** The models for the engineering aspects that are considered in this research are very simplified w.r.t. the reality. In order to implement the MCA in projects, complex models and more design aspects should be implemented in the MCA to have a more accurate outcome of the optimized dike geometry. For now, simple versions of those complex models and a few design aspects are considered to see whether the MCA performs well in familiar territory.
- **Wave overtopping equation:** The wave overtopping equation that is used in this research is picked out since this equation is very simplified. As mentioned in section 6.3, the wind set-up of water is not taken into account. However, not only the wind set-up has a large impact on the wave overtopping, also the wind direction, the outer slope of the dike, and more aspects. In a certain situation, due to wind, the wave height can be up to 1 m for the west side of a river and at the east side, the wave height is negligible. This example shows how important wind is and such parameter should be included in calculating the amount of wave overtopping in this case.
- **Subsoil variation:** To implement the MCA for different project across the Netherlands, the subsoil should also be varied. Right now, the only parameter related to the subsoil is the thickness of the cohesive layer underneath the dike. The subsoil in the Netherlands varies enormously with varying soil parameters. In order to implement the MCA in more projects, subsoil variation should be implemented in the MCA.
- **POP value:** In this research, one value for the POP-value is used. This leads to very high over consolidation ratio near ground level. This is also explained in section 3.2.2. In practice, lower POP-values near ground level need to be considered in order to have a more realistic subsoil.
- **Current dike reinforcement projects:** For further research, it would be nice if the current MCA is used in a current dike reinforcement project in order to see how accurate the MCA is.
- **MCA:** Currently the MCA is strongly bounded to the initial dike geometry and an updated dike geometry is needed in the optimization function when the initial dike geometry doesn't meet the constraints. This means that the optimization is helped upfront with a deterministic approach to update the dike geometries based on constraints. For further research, it would be nice if the optimization function can also find the optimized dike geometries without this deterministic approach.

When using a logarithmic function for the penalties instead of adding just the value 10 on top of the outcome of the objective function when the optimized dike geometry exceeds one of the constraints, the optimization function can approach an optimal dike geometry better without the need of a deterministic approach. This further explained in Appendix C.

Bibliography

[1] Juan Pablo Aguilar-López. "Probabilistic safety assessment of multi-functional flood defences". PhD thesis. Enschede, The Netherlands: University of Twente, Dec. 2016. ISBN: 9789036542586. DOI: 10.3990/1.9789036542586. URL: http://purl.org/utwente/doi/10.3990/1.9789036542586.

[2] N. Arya. *Tuning XGBoost Hyperparameters*. Aug. 2022. URL: https://www.kdnuggets.com/2022/08/tuning-xgboost-hyperparameters.html.

[3] J.H. Bae et al. "Evaluation of sediment trapping efficiency of vegetative filter strips using machine learning models". In: 11.24 (Dec. 2019). ISSN: 20711050. DOI: 10.3390/SU11247212.

[4] P. Banerjee. *A Guide on XGBoost hyperparameters tuning*. 2020. URL: https://www.kaggle.com/code/prashant111/a-guide-on-xgboost-hyperparameters-tuning.

[5] N. Begum. *Hyperparameter tuning in random forest*. Sept. 2023. URL: https://www.kaggle.com/code/nargisbegum82/hyperparameter-tuning-in-random-forests.

[6] N. Beheshti. *Cross Validation and Grid Search*. Feb. 2022. URL: https://towardsdatascience.com/cross-validation-and-grid-search-efa64b127c1b.

[7] B. Brederveld. *PCLake & PCDitch*. URL: https://www.witteveenbos.com/digital-solutions/pcditch-metamodel.

[8] Leo Breiman. *Random Forests*. Tech. rep. Oct. 2001, pp. 5–32.

[9] Tianqi Chen and Carlos Guestrin. "XGBoost: A scalable tree boosting system". 2016. DOI: 10.1145/2939672.2939785.

[10] DATAtab. *Kendall's Tau*. URL: https://datatab.net/tutorial/kendalls-tau.

[11] J. De Vree. *Sloot*. URL: https://www.joostdevree.nl/shtmls/sloot.shtml.

[12] Deltacommissie. *Samen werken met water*. Tech. rep. Sept. 2008.

[13] *DINOloket, Ondergrondmodellen*. URL: https://www.dinoloket.nl/ondergrondmodellen/kaart.

[14] geeksforgeeks. *Random Forest Hyperparameter Tuning in Python*. Dec. 2022. URL: https://www.geeksforgeeks.org/random-forest-hyperparameter-tuning-in-python/.

[15] L. Geldenhuys, F. Hörtkorn, and Y. Narainsamy. *Use of mode of shear for undrained slope stability analysis*. Tech. rep.

[16] Geologische Dienst Nederland. *Grondwaterstanden in Beeld*. 2024. URL: https://www.grondwatertools.nl/gwsinbeeld/.

[17] Happy C. *Machine Learning4GridSearch and Cross-Validation*. Jan. 2024. URL: https://medium.com/@cynthia202312/machine-learning-4-gridsearch-and-cross-validation-7650b3b7a779.

[18] Hoogwaterbeschermingsprogramma. *Wie we zijn en wat we doen*.

[19] HWBP. *Projectoverstijgende Verkenning Macrostabiliteit (POVM)*. URL: https://www.hwbp.nl/kennisbank/pov-macrostabiliteit.

[20] A. Jain. *XGBoost Parameters Tuning: A Complete Guide with Python Codes*. July 2024. URL: https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/.

[21] Jainvidip. *Understanding Random Forest and OOB error*. July 2024. URL: https://medium.com/@jainvidip/understanding-random-forest-0ca15aaa443c.

[22] Elahe Jamalinia et al. "A data-driven surrogate approach for the temporal stability forecasting of vegetation covered dikes". In: *Water (Switzerland)* 13.1 (Jan. 2021). ISSN: 20734441. DOI: 10.3390/w13010107.

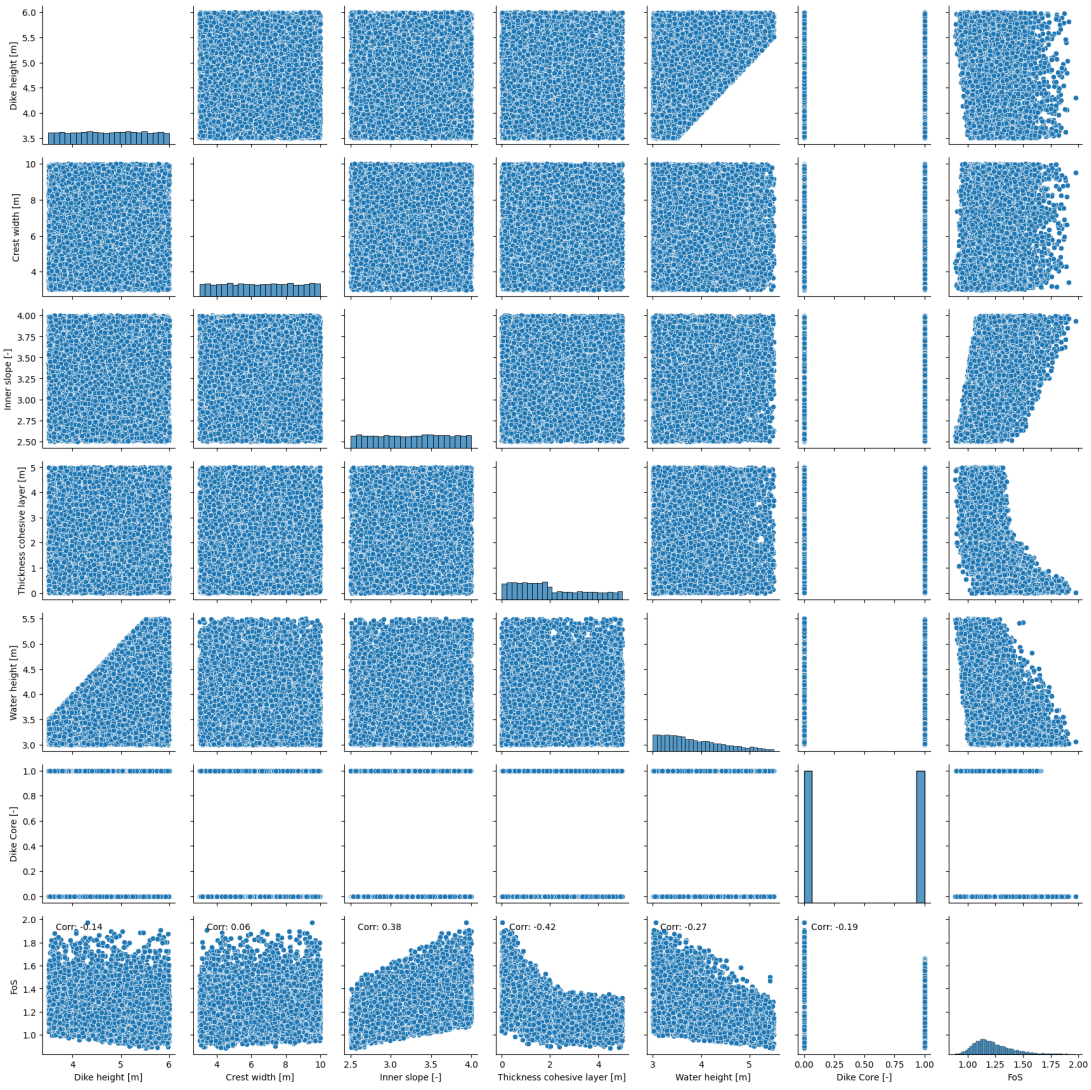
[23] J.H. Janse. *Model studies on the eutrophication of shallow lakes and ditches*. Tech. rep. May 2005.

[24] L.A. Kamphuis. *Surrogate modelling framework for probabilistic assessment of slope stability of dikes on heterogeneous soils*. Tech. rep. Delft: TU Delft, 2022.
[25] M. Kao. *Encyclopedia of Algorithms*. July 2008.
[26] F. Klijn et al. *Doorbraakvrije dijken: een nadere verkenning*. Tech. rep. Deltares, 2014.
[27] W. Koehrsen. *Hyperparameter Tuning the Random Forest in Python*. Jan. 2018. URL: https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74.
[28] M Kok et al. *Grondslagen voor hoogwaterbescherming*. Tech. rep. Expertisenetwerk waterveiligheid, Nov. 2017.
[29] S. Mallick. *Understanding Feedforward Neural Networks*. URL: https://learnopencv.com/understanding-feedforward-neural-networks/.
[30] J.W. Miller. "Chapter 5: Monte Carlo Approximation". In: 2015. Chap. 5.
[31] H.B. Misérus. *Inventaris van het archief van de Delta-Commissie, (1936) 1953-1961*. Tech. rep. Ministerie van Onderwijs, Cultuur en Wetenschap, 1975.
[32] H.S. Murat. "A brief review of feed-forward neural networks". In: *Communications Faculty Of Science University of Ankara* 50.1 (Feb. 2006), pp. 11–17. ISSN: 1303-6009. DOI: 10.1501/commua1-2{_}0000000026.
[33] R. Nair. *SVM-Hyperparameter Tuning*. 2022. URL: https://www.kaggle.com/code/rajeevnair676/svm-hyperparameter-tuning.
[34] Neova. *Machine Learning Algorithms: Beginners Guide Part 1*. June 2018. URL: https://www.neovasolutions.com/.
[35] Nick. *Grid Search*. June 2018. URL: https://napsterinblue.github.io/notes/machine_learning/model_selection/grid_search/.
[36] *pandas.DataFrame*. Apr. 2024. URL: https://pandas.pydata.org/docs/index.html#.
[37] Panjeh. *scikit learn hyperparameter optimization for MLPClassifier*. June 2020. URL: https://panjeh.medium.com/scikit-learn-hyperparameter-optimization-for-mlpclassifier-4d670413042b.
[38] Preston Fan. *What is XGBOOST?* 2020. URL: https://www.kaggle.com/discussions/getting-started/145362.
[39] B. Priya. "Hyperparameter Tuning: GridSearchCV and RandomizedSearchCV, Explained". In: *Machine Learning* (Nov. 2023). URL: https://www.kdnuggets.com/hyperparameter-tuning-gridsearchcv-and-randomizedsearchcv-explained.
[40] Projektgroep 'TAWB Constructief'. *Handreiking Constructief ontwerpen*. Tech. rep. Delft: Technische Adviescommissie voor de Waterkeringen (TAW), Apr. 1994.
[41] S. Raschka. *STAT 479: Machine Learning Lecture Notes*. Tech. rep. Delft: Technische Adviescommissie voor de Waterkeringen, May 2019. URL: http://stat.wisc.edu/%E2%88%BCsraschka/teaching/stat479-fs2019/.
[42] Sebastian Raschka. *STAT 451: Intro to Machine Learning Lecture Notes*. Tech. rep. 2020. URL: http://stat.wisc.edu/%E2%88%BCsraschka/teaching/stat451-fs2020/.
[43] N. Rasifaghihi. *From Theory to Practice: Implementing Support Vector Regression for Predictions in Python*. Apr. 2023. URL: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927.
[44] Rijksoverheid. *Klimaatverandering en gevolgen*. URL: https://www.rijksoverheid.nl/onderwerpen/klimaatverandering/gevolgen-klimaatverandering.
[45] Rijksoverheid. *Onderzoek: meer dijken moeten worden versterkt*. Nov. 2023.
[46] Water Verkeer en Leefomgeving Rijkswaterstaat. *Regeling veiligheid primaire waterkeringen 2017*. Jan. 2017.
[47] Water Verkeer en Leefomgeving Rijkswaterstaat. *Schematiseringshandleiding macrostabiliteit*. Tech. rep. Ministerie van Infrastructuur en Waterstaat, May 2021. URL: www.iplo.nl.
[48] Rijkswaterstaat and Water Verkeer en Leefomgeving. *Schematiseringshandleiding hoogte*. Tech. rep. Ministerie van Infrastructuur en Waterstaat, Nov. 2019. URL: www.helpdeskwater.nl.
[49] L. Rocha, R. Taormina, and H. Wang. *Data Science and Artificial Intelligence for Engineers*. Delft, 2023. URL: https://interactivetextbooks.citg.tudelft.nl/dsaie/intro.html.
[50] S. Rowland. *Maximize model performance without maximizing effort*. Apr. 2020. URL: https://blogs.sas.com/content/subconsciousmusings/2020/04/08/maximize-model-performance/.

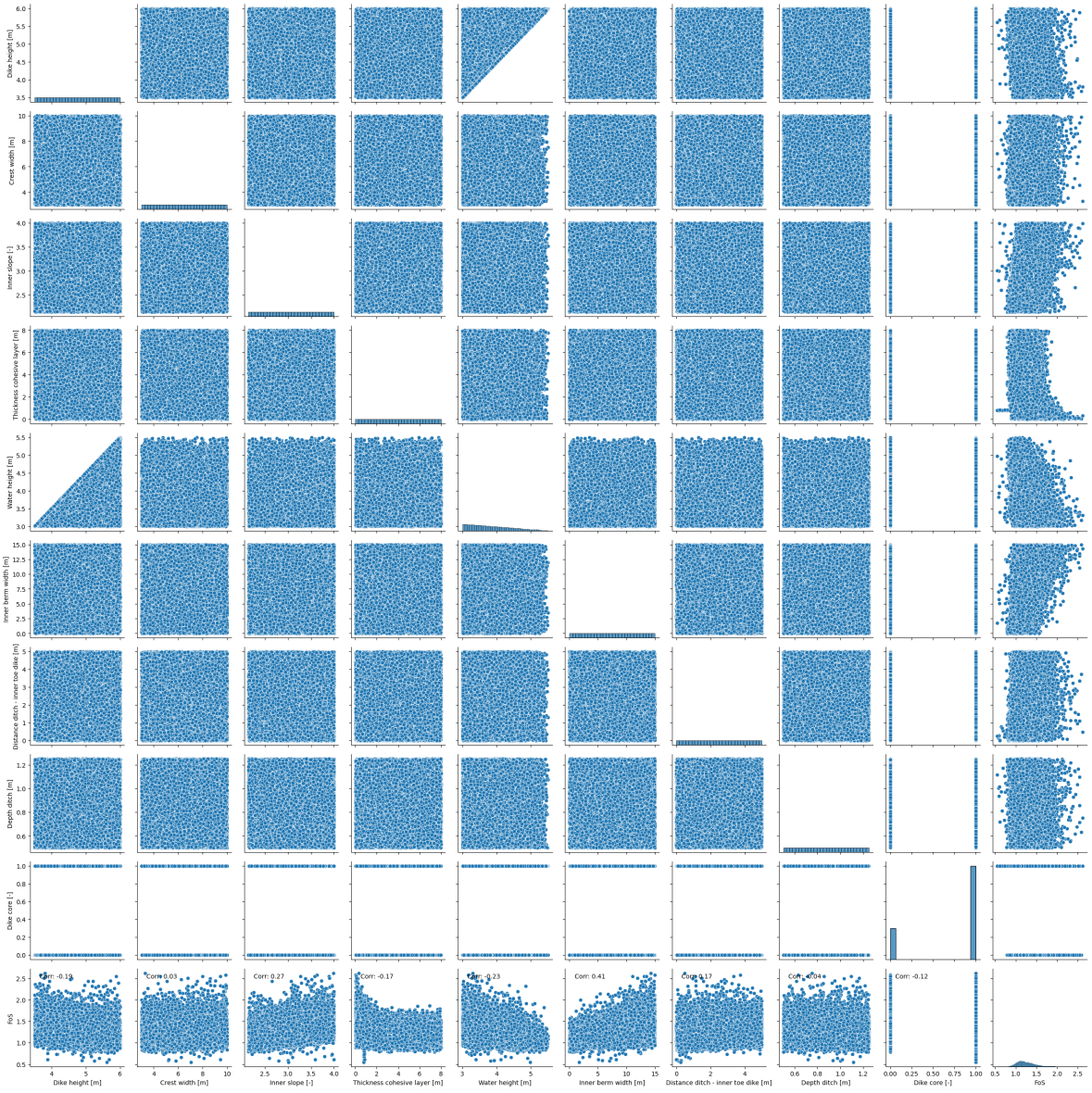
[51] scikit-learn. *SVR*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html.
[52] A. Sethi. *Support Vector Regression Tutorial for Machine Learning*. Feb. 2024. URL: https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/.
[53] Technische Adviescommissie voor de Waterkeringen. *Technisch Rapport Waterkerende Grondconstructies*. Tech. rep. 2001.
[54] Technische Adviescommissie voor de Waterkeringen. *Technisch rapport waterspanningen bij dijken*. TAW, Technische Adviescommissie voor de Waterkeringen, 2004. ISBN: 9036955653.
[55] *train_test_split*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html.
[56] S. Van Breukelen et al. *Natuurvriendelijke Oevers Handreiking*. Tech. rep. July 2003.
[57] M. Van Buuren et al. *Krachtige IJsseldijken Krimpenerwaard, Esthetisch Programma van Eisen*. Tech. rep. Hoogheemraadschap van Schieland Krimpenerwaard, Sept. 2019.
[58] J.W. Van Der Meer. *Technisch Rapport Golfoploop en Golfoverslag bij Dijken*. Tech. rep. Delft: Technische Adviescommissie voor de Waterkeringen, May 2002.
[59] R. Van Der Meij. *User Manual_D-stability*. Tech. rep. Deltares, June 2023.
[60] A. Van Duinen. *Handreiking voor het bepalen van schuifsterkte parameters WTI 2017 Toetsregels Stabiliteit*. Tech. rep. Deltares, Dec. 2014.
[61] Alexander Van Duinen. *Memo Aan Bianca Hardeman (RWS WVL)*. Tech. rep. Deltares, July 2014.
[62] Britt Van Haastregt. *STOWA 2019-10 Droogte zomer 2018 v2*. Tech. rep. Amersfoort: STOWA, May 2019.
[63] G.A.M. Van Meurs and G.A.M. Kruse. *Update inzichten gebruik van klei voor ontwerp en uitvoering van dijkversterkingen*. Tech. rep. Deltares, Jan. 2017.
[64] Mick Van Montfoort. *Safety assessment method for macro-stability of dikes with high foreshores*. Tech. rep. TU Delft, Jan. 2018. URL: www.zuider-ijsseldijk.nl.
[65] P J T M Van Puijenbroek and J Clement. *Basiskaart Aquatisch: de Watertypenkaart Het oppervlaktewater in de TOP10NL geclassificeerd naar watertype*. Tech. rep. Den Haag/Bilthoven: Planbureau voor de Leefomgeving, 2010. URL: www.pbl.nl,.
[66] T. Van Woerkom et al. "Global sensitivity analysis of groundwater related dike stability under extreme loading conditions". In: *Water* 13.21 (Nov. 2021). ISSN: 20734441. DOI: 10.3390/w13213041.
[67] Co Verdaas. *Wat is het nationaal Deltaprogramma?*
[68] J K Vrijling, T Schweckendiek, and W Kanning. *Safety standards of flood defenses*. Tech. rep. 2011. URL: https://www.researchgate.net/publication/254908290.
[69] Witteveen+Bos. *PCDitch metamodel*. URL: https://www.witteveenbos.com/digital-solutions/pcditch-metamodel.
[70] Beiyang Yu. *Machine learning for prediction of undrained shear strength from cone penetration test data*. Tech. rep. Delft: TU Delft, Sept. 2022.
[71] Cor Zwanenburg, Alexander Van Duinen, and Arno Rozing. *Technisch Rapport Macrostabiliteit*. Tech. rep. Deltares, Feb. 2013.

Appendix A - Pairplots

Pairplot of the improved Simple model



Pairplot of the Complex model



Appendix B - Code snippets

Code snippet of MinMaxNormalizer

```
class MinMaxNormalizer:
    def __init__(self, data_training):
        self.max = data_training.max()
        self.min = data_training.min()

    def normalize(self, data):
        normalized_data = (data - self.min) / (self.max - self.min)
        return normalized_data

    def denormalize(self, data):
        return data * (self.max - self.min) + self.min

X_normalizer = MinMaxNormalizer(train_data.iloc[:, :9])
y_normalizer = MinMaxNormalizer(train_data.iloc[:, 9])

X_normalized = X_normalizer.normalize(train_data.iloc[:, :9])
y_normalized = y_normalizer.normalize(train_data.iloc[:, 9])

# 80% of dataset is trainingset and 20% of dataset is testset
X_train, X_test, y_train, y_test = train_test_split(X_normalized, y_normalized, test_size=0.20, random_state=42)

model = load('best_model_SVR_Complex.joblib')
```

Code snippet of hyper parameter tuning

cv = 5, RandomizedSearchCV

```
params = {
    "n_estimators": list(np.arange(50, 1050, 50)),
    "max_features": [None, "sqrt", "log2"],
    "min_samples_split": list(np.arange(2, 11, 1)),
    "min_samples_leaf": list(np.arange(1, 6, 1)),
    "max_depth": list(np.arange(10, 110, 10)),
    "bootstrap": [True] # default
}

# Create your RandomForestRegressor
regr = RandomForestRegressor(random_state=42, criterion="squared_error")

# Create your RandomizedSearchCV object
rfr_reg1 = RandomizedSearchCV(regr, params, cv=5, scoring="neg_mean_squared_error")

# Fit RandomizedSearchCV object to data
rfr_reg1.fit(X_train, y_train)

# Access best hyperparameters and score
print("Best hyperparameters found:")
print(rfr_reg1.best_params_)
print("Best MSE score:", rfr_reg1.best_score_)
```

```
Best hyperparameters found:
{'n_estimators': 700, 'min_samples_split': 4, 'min_samples_leaf': 3, 'max_features': None, 'max_depth': 60, 'bootstrap': True}
Best MSE score: -0.003120462271961431
```

Code snippet of validation metrics

```
best_model_RFR = rfr_reg3.best_estimator_
best_model_RFR.fit(X_train, y_train)
y_pred_RFR = best_model_RFR.predict(X_test)

mse_RFR = mean_squared_error(y_test, y_pred_RFR)
mae_RFR = mean_absolute_error(y_test, y_pred_RFR)
r2_RFR = r2_score(y_test, y_pred_RFR)

print("Mean Squared Error:", mse_RFR)
print("Mean Absolute Error:", mae_RFR)
print("R2 score:", r2_RFR)
```

```
Mean Squared Error: 0.0027480700265921513
Mean Absolute Error: 0.03054106901718148
R2 score: 0.6843449310940856
```

Code snippet of the defined constraints

```
# Define the constraints
def constraint_ineq(vars, fixed_param):
    H, B, IS, Bb, dd, Dtoe = vars
    gamma_beta = 1 - (0.0033 * fixed_param[4])
    hk = H - fixed_param[5]
    return 10 - wave_overtopping(hk, fixed_param[3], gamma_beta, 1.0, 9.81)

def constraint_ineq2(vars):
    H, B, IS, Bb, dd, Dtoe = vars
    wl = dd - 0.4
    return ((0.5 * wl) + (wl * 3 * wl)) - 0.625

def constraint_ineq3(vars, fixed_param):
    H, B, IS, Bb, dd, Dtoe = vars

    if fixed_param[9] == 1:
        dike_core = 0
    elif fixed_param[9] == 6:
        dike_core = 1

    x_data = pd.DataFrame({
        'Dike height [m]': [H],
        'Crest width [m]': [B],
        'Inner slope [-]': [IS],
        'Thickness cohesive layer [m]': [fixed_param[7]],
        'Water height [m]': [fixed_param[5]],
        'Inner berm width [m]': [Bb],
        'Distance ditch - inner toe dike [m]': [Dtoe],
        'Depth ditch [m]': [dd],
        'Dike core': [fixed_param[8]],
    })

    x_norm = X_normalizer.normalize(x_data)
    y_pred = model.predict(x_norm)
    FoS = y_normalizer.denormalize(y_pred)
    return FoS - 1.2
```

Appendix C - Improved optimization process

In the main report, a penalty of a value of 10 was applied on top of the outcome of the objective function. This resulted in a very high jump in the value of the objective function, which was needed, otherwise the optimization function did not care about the constraints. However, when a logarithmic function is used instead of a value of 10 for exceeding the constraints, the optimization function can approach optimal dike designs better without the help of a deterministic approach. The logarithmic function should have a shape like the one that is indicated in the figure. The resulting values for the objective function are of similar size with a maximum deviation of 0.2 m for the input parameters inner berm width and distance between the ditch and the inner toe of the dike for the dike geometry.