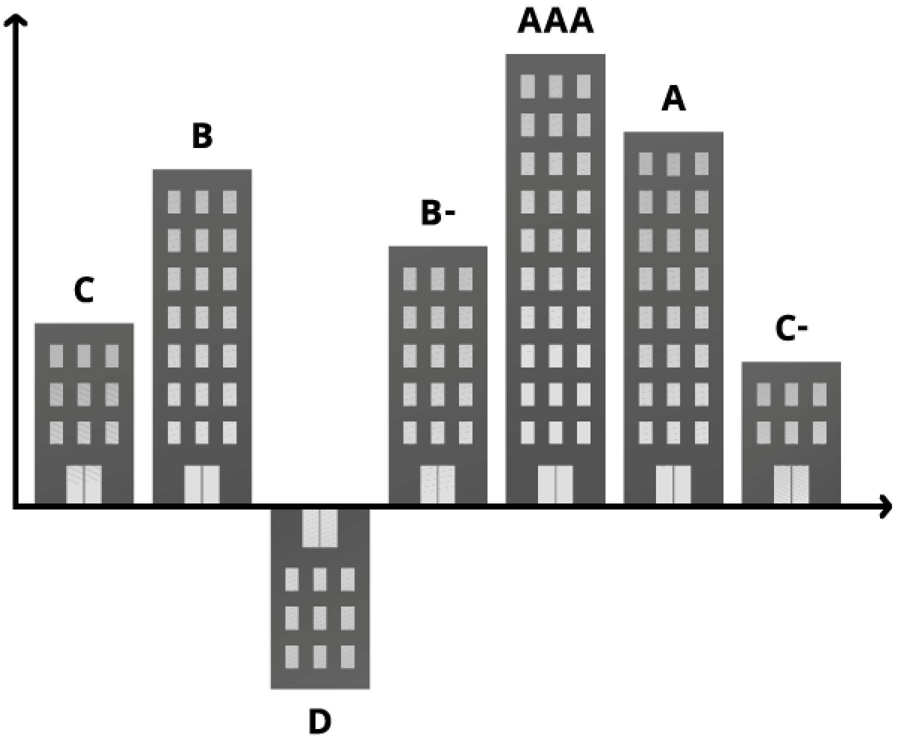
Master's Thesis

MSc in Finance and Investments

Copenhagen Business School



Prediction of Default for Financial Institutions with Machine Learning

Author's	Sindre Falkeid Kommedal (Student number: 108060)
	Anders Lefdal Nordgård (Student number: 88037)
Submission date	15-01-2019
Supervisor	Michael Ahm
Characters (including spaces)	147,922

I. Acknowledgments

Several people have in one way or another contributed to the completion of this thesis.

First of all, we would like to thank Nordea for giving us the opportunity to write this thesis and provide us with insightful information and inspiration. We would also like to send a big thank you to Kristian Salte for proofreading and motivational talks. We would also like to thank our families for their unconditional support and care. Lastly, we would like to thank our girlfriends for their support, patience and encouragement.

II.Abstract

This paper investigates if Artificial Intelligence techniques can be used as an adequate model to calculate a standalone probability of default for counterparties. We start with introducing the legislative framework for the internal rating-based approach, Basel. Furthermore, before introducing the applied methods, we present the elementary concept of machine learning. In pursuance of the best applicable model we have conducted training and testing with three different models. For the chosen models, Neural Networks, Support Vector Machines and Random Forest underlying theory and mechanisms is introduced. With regards to performance assessment of the aforementioned models, several statistical evaluation and comparisons is conducted. The best performing model is the advanced option for decision trees, Random Forest. Nonetheless, the more complex Neural Networks and Support Vector Machines shows disappointing results, which is in conflict with some previous research. In contrary to previous findings this paper concludes that none of the tests can significantly outperform the comparative benchmark - logistic model. We do not wish to neglect the models entirely. Rather, this paper presents the challenges and importance of a satisfactory dataset.

III. Abbrevations

AMA: Advanced Measurement Approach

AUC: Area Under the Curve

BCBS: The Basel Committee on Banking Supervision

BIS: Bank of International Settlement

CART: Classification and Regression Trees

CCR: Counterparty Credit Risk

CVA: Credit Valuation Adjustment

FSB: Financial Stability Board

IRB: Internal Rating Based

MLP: Multi-layered perceptron

NN: Neural Networks

RF: Random Forest

ROC: Receiver Operating Characteristic

SVM: Support Vector Machines

VAR: Value at Risk

List of Tables

List of Figures

Table of Content

1 Introduction

1.1 Background of study

The financial industry has experienced substantial growth over the past decades. More and more complex investment instruments are observed. The banking industry has been characterized by major technological development and liberalization in the asset and credit markets. Banks services are no longer limited to the creation of savings accounts or the granting of mortgages but include complex financial services and products. It follows that profits no longer simply arise from interest rate differentials, but includes income generating activities from more advanced service lines such as Privet Banking and Wealth Management.

The sector is of great importance for both national and international economies. Its intermediation provides settlements between participants providing or in need of capital. All economies and markets rely on a stable and well-functioning bank sector. From a historical perspective, it can be seen, the consequences of financial crashes on the economy.

Beyond any doubt, the last decade's financial crises have caused disastrous consequences. Since the Asian and the Russian crises of the late 1990's where the last one famously caused the disaster and insolvency of Long-Term Capital Management. The latest financial crisis in 2007-2008, were credit agencies where highly involved in the cause of the crisis. After big banks like Bear Sterns, Lehman Brothers and Merrill Lynch faced the disaster of insolvency. The rippling effect was seen in the entire financial market. AIG as one example, needed a bailout of some 180 billion US dollar from the US government, due to trading in credit default swaps on collateralized debt obligations. An event which in a worst case could have triggered the default of major banks worldwide.

The Subprime crisis in 2007 showed that the banking sector not only had unfavorable practices but also that there were major shortcomings in public oversight and regulation (C.A.E.Goodhart, 2008).

In order not to worsen the situation, authorities with central banks in the lead, had to provide liquidity and guarantees packages that transferred the burden from banks to taxpayers (Bank for International Settlements Communications, 2010).

The financial crisis in 2007 was a "crisis for regulation and supervision." Capital requirements were not used adequately to cover important risk exposures and liquidity risk was taken unseriously. Further, poor coordination and monitoring of decisions on financial stability and the uncertain valuation of financial instruments led to instability and the financial crisis became a fact (Jickling, 2009). The world experienced during the financial crisis what the effects and consequences of a weak banking sector bared on the economy and the importance of a strong banking system, to create a stable global economy.

The lack of monitoring and regulation has resulted in new measures that have been taken to prevent and reduce the consequences of the financial crisis. One of the most important measures that was implemented were the preparation of stricter and more concrete requirements for liquidity adjustments and capital adequacy from the Basel Committee.

1.2 Nordea

Nordea is the largest financial services provider in the Nordics. As of 2017, their operating income attributed to some EUR 9.5 billion (Nordea Group, 2018). In the same period, their total assets value was EUR 581.6 billion. Their main business areas are divided into four main services. Personal Banking, Commercial & Business Banking, Wholesale Banking, and Wealth Management.

Risk and capital management is structured in accordance with the Basel III framework published by the Basel Committee on Banking Supervision (Nordea Group, 2018). Their credit decisions are based on the preliminary credit risk assessments used consistently across the Group.

The structure emphasizes different risk exposures so to adjust the scope and weightings of the specific risk components. The exposures used in the risk assessments are also applied as part of their internal rating methods.

Nordea performs risk monitoring and controlling on a regular basis to ensure that all activities remain within acceptable limits. Some of the monitoring is conducted on a daily basis, here especially for market risk, counterparty credit risk, and liquidity risk. Other exposures are assessed on a monthly or quarterly basis (Nordea Group, 2018).

All risks levels within the Nordea Group are defined so to measure any breaches that the bank is not willing to accept in order to attain risk capacity, their business model and overall strategic objectives. The levels of risk are set by constraints reflecting the views of shareholders, debt holders, regulators, and other stakeholders (Nordea Group, 2017).

The framework defines critical risk attributes to Nordea's overall risk exposure regarding all business activities. The terms of risks are "*credit risk, market risk, liquidity risk, operational risk, solvency and compliance/non-negotiable risks*" (Nordea Group, 2017).

With specificity to the objective of the thesis, the relevant risk exposure is categorized as counterparty credit risk, which is a subsection of credit risk. Nordea defines credit risk as:

"Credit risk is defined as the potential for loss due to failure of a borrower(s) to meet its obligations to clear a debt in accordance with agreed terms and conditions. Credit risk includes counterparty credit risk, transfer risk and settlement risk" (Nordea Group, 2018).

The definition of counterparty credit risk follows as:

"Counterparty credit risk is the risk that Nordea's counterpart in an FX, interest, commodity, equity or credit derivative contract defaults prior to maturity of the contract and that Nordea at that time has a claim on the counterpart. Current exposure net (after close-out netting and collateral reduction) represent EUR 8,5B of which 30% was towards financial institutions" (Nordea Group, 2018).

1.3 Research Question

For Nordea the importance of good quality assessments regarding risk management is substantial. Due to both the instability in the banking sector and the implemented legislative frameworks, Nordea seeks to expand their understanding of risk measurements further.

The objective assigned is to develop a Bank Rating Model to initiate in-house credit assessment of the OTC counterparties through an up-to-date model with sufficient predictive ability. Model's purpose is to assign counterparties a standalone probability of default that is valid one year since the analysis date.

With the development of machine learning algorithms, big data capacity and overall improved computing power, they wish to analyze the potential of applying Machine learning for such modeling. Therefore, this paper will examine the potential use of machine learning to calculate the default probability.

Nordea's requirements for satisfactory data analysis is that the prediction is based on their provided dataset. Further, the result should yield the expected probability of default within a one-year period for the entity as a whole. The models should also satisfy relevant legislative frameworks (such as Basel III).

Therefore, the research question is the following:

"Can machine learning be used to develop an adequate model for assigning counterparties a standalone probability of default?"

1.4 Structure of the paper

The paper is divided into 9 chapters with subsections in each chapter. Chapter 1 presents the background for the study and the research question. Chapter 2 will review previous publications in the field of machine learning and related studies. Chapter 3, 4 and 5 will respectively present the legislative, conceptual and theoretical framework used to investigate the research question. In chapter 6, the variable selection, model selection, performance assessment and dataset are described. Chapter 7 describes the preparation process and training done before testing. Results, analysis and performance is presented in chapter 8. Finally, chapter 9 draw conclusions and suggest future research topics related to this study.

2 Literature review

This chapter presents various theories, methods, and models related to the probability of default. “Bankruptcy Prediction in Banks and Firms via Statistical and Intelligent Techniques – A review” by (Kumar & Ravi, 2007) and “Assessing Methodologies for Intelligent Bankruptcy Prediction” by (Kirkos, 2015) provides an overview of what has been done in the industry and the reference articles in these studies are heavily used.

In the literature, there are mainly two types of bankruptcy prediction models, accounting-based models and market-based models (Berg, 2005). Moody's Expected Default Frequency (EDF) model is an example of a market-based model (Nazeran & Dwyer, 2015). This type of model is based on the company's market value where the stock price is usually used as an approach. Models based on market values thus require that the companies are listed on the stock exchange, while Accounting-based models use information from the accounts to predict default.

Before quantitative sizes were obtained on how enterprises performed, they established agencies whose task was to provide qualitative information regarding the creditworthiness of corporations (Altman, 1968).

Formal studies around default chances began around 1930s, and since then several studies have concluded that companies that go bankrupt have significantly different vital financial figures from those companies that continue to operate.

Even though discriminant analyses have restrictive assumptions, it remained the dominant method in the prediction of default. Until the end of the 1970s, when the seminal work of (Martin, 1977) introduced the first method of failure prediction that did not make any restrictive assumptions regarding the distributional properties of the predictive variables. The logistic regression separates from the discriminant analysis in the way that discriminant analysis assumes the financial statement data to be normally distributed.

Later, (Ohlson, 1980) introduced his logistic regression model called the O-score as an alternative to Altman's Z-score. James Ohlson (Ohlson, 1980) in company with William H. Beaver (Beaver, 1966) and Edward I. Altman (Altman E. I., 1968) is today recognized as some of the most notable studies on insolvency using financial figures.

2.1 Standardized Methods

William H. Beavers univariate model from 1966 is recognized as one of the first studies for prediction of default based on key ratios from the financial statements. Univariate analysis views all fundamental financial figures individually, therefore the study assumes that one ratio can be used as a prediction for the health of an entire corporation (Beaver, 1966).

In his study from 1966 Beaver uses a paired selection of 79 solvent and 79 insolvent companies. The corporations were paired based on sector and size. He started out with almost 30 key financial figures, which was shortened to only 6 figures based on the ability to explain the situation of the organizations.

The weakness of the univariate method is that different conclusions can be obtained for different key figures for the same company depending on how much the key figures are weighted (Altman E. I., 1968). This is due to the fact that the model does not consider the relation between the individual financial figures.

(Altman E. I., 1968) developed a multivariate linear discriminant analysis for bankruptcy prediction. Linear discriminant analysis (LDA) is a statistic method suitable for studies where the dependent variable is binary (Hair, 1998). The LDA approach tries to organize and classify the observed objects or events into groupings to create a linear classifier. An advantage of using multivariate as opposed to univariate is that the method tries to find an interaction between the different variables.

In his studies, (Altman E. I., 1968) gathered information from 66 corporations, where the dataset was equally split in 33 default and 33 non-defaults. The model is based on 22 financial figures, popular from earlier studies, as well as a few new. After an iterative process, where all variables were considered he landed on 5 ratios he found most significant for an accumulated bankruptcy prediction.

The Z-score contains a linear combination of the mentioned 5 ratios multiplied by corresponding coefficients from the discriminant analysis. The output gives an indication of distress within a company.

(Ohlson, 1980), chose to use a conditional logistic regression with a “maximum likelihood” estimator. His approach is an alternative procedure for multiple discriminate analysis and is a general linear model. Ohlson’s argument for logistic regression being better than LDA is due to the interpretation of the coefficients (Ohlson, 1980).

Logistic regression is used as a statistical method to analyze data with one or more explanatory variable that control the outcome. It is measured with a binary variable containing two possible values, 0 and 1. The objective is to find the model which best describes the relationship between the binary variable and the independent explanatory variables.

The dataset which Ohlson used is significantly bigger than both Beaver and Altman used in their studies. Containing 2058 companies which did not default, and only 105 entities that did default. Ohlson's model is considered to be more accurate than Altman's Z-score. (Financial Ratios and the Probabilistic Prediction of Bankruptcy) In contradiction to Altman's Z-Score Ohlson applies 9 factors, where 2 factors are dummy variables.

2.2 Static Endogenous Models

In the article by (Kumar & Ravi, 2007) the authors analyze research done on default in the period 1968 to 2005. They provide an overview of the different methods used during the period, distinguishing between two different techniques to solve the bankruptcy problem, statistical techniques, and intelligent techniques.

The broad category of statistical techniques includes several of the methods discussed in this chapter, including linear discriminator analysis, multivariate discriminant analysis, and logistic regression. Intelligent techniques explain various machine learning techniques, including neural networks, support vector machines, k-nearest neighbors and classification trees.

2.3 Implementation of machine learning and Neural Networks

The implementation of Neural Networks in bankruptcy predictions started in the early 1990s. (Odom & Sharda, 1990) was one of the first implementing the Neural Network approach, applying (Altman E. I., 1968) predictive variables. After multiple experiments, they compared the performance of NN against the multivariate discriminant analysis. Analyzing error results of type 1 & 2, they concluded that NN outperformed the more traditional method.

In the next couple of years, (Tam K. , 1991) and (Tam & Kiang, 1992) applied NN for the prediction of banks defaulting. Both studies concluded that NN outperformed the established methods on a one-year term, while the logit performed best for two-year horizons.

(Salchenberger, Mine, & Lash, 1992) came to a familiar conclusion based on thrift failures, Salchenberger concluded that using NN for prediction outperformed logit on an 18-month forecasting horizon.

Further papers can be (Altman, Giancarlo, & Varetto, 1994) “Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks”.

As this paper uses firms within retail, industrial and construction it is not as relevant as others. However, an important takeaway from the paper is the problem of the “black-box” and cases of illogical weightings for indicators, as well as overfitting the data.

Finally, “A Neural Network Approach for Credit Risk Evaluation” (Angelini, di Tollo, & Roli, 2008) is a great fundament for this thesis as it focuses on everything from Credit Risk, and Basel Framework to the Neural Networks.

The background motivation for the paper is the Basel Framework where the Basel Committee on Banking Supervision “proposes a capital adequacy framework that allows banks to calculate capital requirement for their banking books using internal assessments of key risk drivers”. This research article describes a successful application of neural networks to credit risk assessment by using feedforward networks. The application is tested on real-world data, and the paper concludes that “neural networks can be very successful in learning and estimating the bonis/default tendency of a borrower, provided that careful data analysis, data pre-processing and training are performed”

After the establishment of Artificial Intelligence methods, (Huang, Chen, Hsu, Chen, & Wu, 2004) introduce a relatively new approach, Support Vector Machines (SVM). Using backpropagation neural networks (BNN) as their benchmark, their research obtained an accuracy of around 80% for both BNN and SVM when applied for the United States and Taiwan markets. However, another part of their research paper is to improve the interpretability of AI methods. Here they applied recent study results in neural network interpretation and obtained relative importance of the input variables. Which then was applied to create a market comparative analysis on the different determined factors in the chosen markets.

The last type of model applied in this paper is a more simplistic mathematical approach, and therefore do not require the same in depth elaboration as the other presented methods. The core concept of Classification and Regression Trees (CART) was published in Leo Breiman's seminal paper in 1984. Later on, in 2001, Breiman extended the theory with Random Forests.

A number of authors have researched and described Random Forest in their papers, among some of them is (Amaratunga, Cabrera, & Lee, 2008), (Biau, Devroye, & Lugosi, 2008) and (Buja & Stuetzle, 2006).

Even though some of the research already reviewed touches upon variable selection, we have looked a little more into the selection of variables. As in (Derksen & Keselman, 1992), a simple variable selection is well described.

For the use of the more sophisticated Minimum Redundancy Maximum Relevance (MRMR), the mathematical framework is explained in (Peng, Long, & Ding, 2005). Further, the analysis relevant for performance assessment and the tools used in the process is explained by papers such as (Lobo, Jiménez-Valverde, & Real, 2007), (Bhattacharyya, 2000) , (Siddiqi, 2015) and (Fawcett, 2006).

As mentioned above, a lot of research studies has been done about the use of Artificial Intelligence in assessing bankruptcy and/or credit risk assessment. The overall findings from the studies appear to be that machine learning approaches achieve adequate performances regarding the prediction of default. Empirical evidence on the models' predictive performances relative to each other is somewhat mixed. When it comes to variable selection there seems to be a mixed opinion on which factors achieve the highest level of explanation for the end results. However, there seems to be a consensus that capital adequacy, asset quality, earnings, and liquidity are seen as the most important (Kumar & Ravi, 2007).

3 Legislative framework

This chapter presents the underlying motivation behind the establishment of the Basel committee. In addition, the different accords are presented, to introduce the relevance of the internal based approach and its requirements. This is done, in order to oblige the legislative framework required for Nordea's' in-house credit risk assessments.

The section only includes the proposed Basel regulatory framework found relevant for the purpose of this thesis. Thereby excluding sections from the different accords. Some sections not directly intervened with the internal rating model are included to explain concepts or to present the development of the Basel accords.

3.1 Bank of International Settlement (BIS)

The severity of the financial crisis in 2007 and 2008 clarified that the current regulations were not optimal, and the aftermath showed how vulnerable and unstable the financial market was. Market participants, such as banks and other financial institutions, acted in their best interest and established their own paths for routines and risk assessments. One consequence of this was the collapse of Lehman Brothers that illustrated the poor risk management in the banking sector. As important, it illustrated that control and supervision were not optimal.

The response to the international crisis is that the Basel Committee has developed a new regulatory framework, called Basel III. Basel III will be a framework to improve the banking sector, so that the likelihood and consequences of new financial crises will be reduced. The new regulations will form national regulations and be implemented in 2019 at a global level (Financial Stability Board, 2018). The main objective of Basel III is that banks should be better prepared for financial events and handle crisis better.

Bank for International Settlement (BIS) was established in 1930. The international institution is owned by central banks and plays an important role in their international cooperation.

Bank for International Settlement fosters international monetary and financial cooperation and serves as a bank for central banks (Goodhart, 2011).

The Basel Committee on Banking Supervision (BCBS), first established in 1974, is a subcommittee of BIS. Its establishment was motivated by claims from the G10 countries, on the basis of a number of bankruptcies. They were commissioned to develop a regulatory framework and a set of standards to avoid similar bankruptcies.

In 2018, the Basel Committee totaled 45 member entities from a variety of different jurisdictions. The most present participants are central banks, regulatory authorities and other jurisdictions with formal supervision responsibilities in the banking sector (Basel, 2018). The Basel Committee now stands behind the standards underlying the regulation of banks and other credit institutions worldwide.

The committee has no supranational supervisory authority and their proposals for regulations have no legal power in each country. Rather, it is only meant to be a broad wording of supervisory standards and guidelines. Thereby, it is up to each country's authority regarding their decisions on implementations of the published standards and guidelines (Goodhart, 2011) A national implementation of the standards with low discrepancy of the proposals will lead to a convergence of a common standard between member states.

The first publication of the Accord (Basel I) was introduced in 1988 after several bankruptcies over the period 1965 and 1981 (Goodhart, 2011). A decade after the implementation of Basel I, the committee realized the need for a more detailed framework. They proposed a new framework based on three pillars. Firstly, a minimum requirements for solidity, structures for risk management and internal controls and lastly disclosure requirements. (Balin, 2008). Following dialogue and several tests with the member countries, Basel II was introduced in June 2004.

In 2008, the regulations were found to be insufficient to avoid the financial crisis we experienced and realized that further regulation was necessary. At the end of 2010, the Basel

Committee presented a new edition of the regulations that would make banks more prepared for crises that had been experienced.

In order to understand the implementations of Basel III, it is necessary to look at the previous accords, namely Basel I and Basel II.

3.1.1 Basel I

The main objective of Basel I was two-folded. The first objective was to strengthen the international banking system which proved to be narrowed before Basel I was introduced. In addition, the Committee wanted to reduce the disparities between international banks' competitiveness by encouraging a common standard and regulation for the financial sector (Goodhart, 2011).

The reasoning for a common regulation was driven from international banking actors to the authorities for a regulatory race against the bottom.

The banks threatened moving to countries with weaker regulations (Balin, 2008). With these two requirements, the committee wanted to strengthen the banking sector to withstand fluctuations in the real economy. In 1993, the Basel Committee came up with a further development of the current framework. Here with improved guidelines for capital adequacy requirements, in order to reduce losses regarding market risks. Basel I divide itself into four pillars (Balin, 2008).

The Constituents of Capital - The first pillar deals with different types of capital. Basel I shares capital into two "Tiers". The first division of capital is called Tier 1. Tier 1 is the core capital indicating the financial strength for a bank. Core capital includes common stock, retained earnings and various funds. These factors are often referred to as common equity. Banks also have different types of innovative hybrid instruments that can be taken as banks' core capital, given that they meet a number of requirements that are set to qualify as core capital. An example of hybrid instruments is bond mutual funds, but these cannot exceed 15% of common equity (Basel, 1999).

The second tier is called additional capital and consists of reserves to cover potential losses on loans and hybrid debt. It is therefore commonly viewed as banks required reserves. Hybrid capital is a combination of debt and equity. This form of subordinated loan capital is a loan that has a lower priority than other debt.

In case of bankruptcy, this form of debt will first be refunded after other creditors have covered their debts but will be repaid before any payments to the equity holders. At the same time, this form of capital has the characteristic that the distribution of dividend or payment of interest can be postponed if the bank is in need of capital. Additional capital has priority before common equity, which means that losses will first be covered by core capital (Douglas J. Elliott, 2010).

Risk Weighting - Credit risk represents the biggest form of risk a bank holds. This is why the Basel Committee had a major focus in this area (Balin, 2008).

The requirement thus encouraged banks to focus on exercising good risk management, identifying paying customers and being conservative in terms of credit ratings from external agencies.

Assets that are included in the balance sheet are risk weighted so to calculated capital reserves in relation to their credit risk. The exposure is multiplied by a given risk weighting based on the borrower's credit rating. The risk weighting in Basel I is divided into five different risk classes, where the classification goes from risk-free to high risk. The lowest weighted class (risk-free) is weighted with 0% in the calculation in pillar II. While the highest weighted class (high risk) is weighted by 100%.

A Target Standard Ratio - The third pillar is a merging of the two preceding pillars. This pillar provides a universal standard where tier 1 and tier 2 will cover the banks' risk-weighted assets. According to Basel I, the total capital should cover 8% of risk-weighted assets.

Transitional and Implementing Agreements - The fourth and last pillar is the implementation and enforcement of Basel I requirements. Central banks are responsible for the implementation and monitoring. By the end of 1992, all Member States had introduced Basel I with the exception of Japan. In the late 1980s, Japan, experienced a banking crisis that led to major challenges in their banking sector (Balin, 2008). The transition to the new regulations was criticized.

3.1.2 Basel II

According to FSB, member countries started their implementation of Basel III in 2013, with full implementation by 1.th of January 2019 (Financial Stability Board, 2018). In order to understand Basel III, it is important to review essential elements in Basel II. This is because many of these elements have been further developed and transferred into the Basel III regulations. We therefore need a basis to understand how important it is for today's banking sector with updated regulations.

In Basel II, the requirements are defined in three pillars: minimum requirements for subordinated capital, supervisory follow-up and market discipline and publication (Douglas J. Elliott, 2010).

Minimum Capital Requirements - In response to Basel I's criticism, Basel II creates a more sensitive measurement of the banks' risk-weighted assets through the first pillar. With this expansion, it was desired to eliminate the weaknesses that were discovered in retrospect to Basel I and the increasing technological developments in the banking industry (Balin, 2008). The expansion was made through an introduction of operational risk in the calculation basis of minimum capital requirements. This led to capital adequacy requirements for credit risk, operational risk and market risk for banks (Basel, 1999).

Banks have different ways of calculating their credit risk. One of the methods that can be used is rankings from authorized ranking agencies such as Fitch, Moody’s and Standard & Poors.

This method is called "the standardized method" due to an external actor's assessment of the debt. The following table shows different credit ratings and their risk classification provided by Fitch.

Table 1 – Fitch Rating Definition

Fitch Rating definition		
Rank	Rating grade	Risk Characteristic
1	AAA	Prime
2	AA+	High Grade
3	AA	
4	AA-	
5	A+	Upper Medium Grade
6	A	
7	A-	
8	BBB+	Lower Medium Grade
9	BBB	
10	BBB-	
11	BB+	Non-investment grade speculative
12	BB	
13	BB-	
14	B+	Highly Speculative
15	B	
16	B-	
17	CCC+	Substantial Risks
18	CCC	Extremely Speculative
19	CCC-	Default imminent with little prospect for recovery
20	CC	
21	C	
22	D	In Default

Source: (Fitch, 2018)

As an alternative approach to risk calculation, banks can create internal models. The banks themselves can calculate the probability of default with or without regulatory approval (Balin, 2008). This method is called the "Internal Rating Based" approach (IRB).

As mentioned earlier in the thesis, there are three calculation methods for the assessment and protection against operational risk. The methods are mutually exclusive, in other words, banks must choose which method they want to use. The first of the methods, the Basic Indicator Approach, recommends that banks hold capital equivalent to 15% of average gross income over the past three years. Alternatively, banks can divide their business into different business areas, where each area is weighted to their relative size. Banks can then calculate the weighted capital requirements. This is done to hold reserves covering the total operational risk.

The capital holdings required for less risky business lines, such as retail brokerage and asset management, have lower capital requirements than divisions such as the corporate market of a riskier nature. This method of calculating capital requirements around operational risk is called "Standard Approach".

The final method the banks can use is the Advanced Measurement Approach (AMA). This method is more demanding than the two previous methods for both the authorities and the banks. The reason for this is that banks, using this method, must develop own models for calculating capital at operational risk. The supervisory authorities must then approve the models so that they can be used by the banks. This approach has many similarities to IRB, as described in more detail later in this chapter. Both models try to bring more discipline and self-monitoring within the banking legislation and reduce the variance that a regulatory framework often has because of generalization.

The last risk the first pillar is trying to quantify is equity volatility based on their market risk. In assessing market risk, Basel II distinguishes between fixed income and other products such as equity and foreign exchange markets. There exists a variety of different areas for market risk, but the two biggest risks banks face is interest and volatility risk.

When calculating capital requirements for protection against interest rate and volatility risk for interest-bearing assets (government debt, bonds, etc.), "Value at Risk" (VAR) is used.

This has similarities with AMA and IRB, with the development of own internal models from the banks in all three methods.

Supervisory Review Process – The second pillar is the Supervisory Review Process. The authorities have the task of ensuring that banks maintain the minimum capital requirement and they have the authority to impose individual capital adequacy requirements. The individual orders may have different reasons, one reason is that banks can cause major socioeconomic consequences if bankrupted.

Following an evaluation of the banks' risk and capital adequacy process, risk level and the quality of the management and control routines that the banks possess. It will be revealed if there are weaknesses / deficiencies to be addressed. If the error is detected, the authorities intervene at an early stage to avoid financial crises and reduce the consequences.

Market Discipline - The last pillar of Basel II, concerns the transparency of banks' financial position with regard to the minimum capital requirement, in pillar I. By this pillar, both capital requirements and supervision must be disclosed to the market (Basel, 2016). Through publication, pillar III will increase transparency on banks 'financial position. Which will benefit the market because public information will make it easier for the market to assess the banks' capitalization and risk profile.

3.1.3 Basel III

The third accord from the Basel Committee on Regulating Capital and Banking is called Basel III and will start its implementation from 1.1.2019. Basel III will also be based on the same three pillars as Basel II. The work on designing the new guidelines came as a result of Basel II failing to prevent the financial crisis, we experienced in 2007-2008. Basel III thus is a further development of the previous three pillars.

The Basel Committee wanted through the new rules, further increasing the robustness of banks. The reforms strengthen the capital base and increased risk coverage in the capital framework.

The Basel Committee identified several issues regarding the counterparty credit risk (CCR) during the financial crisis. CCR is the risk of counterparties defaulting on their liabilities before the final settlement of the transaction takes place (Bank for International Settlements Communications, 2010). The financial loss of default requires that the transaction or portfolio of the transaction is "*in the money*" at the time of default (Bank for International Settlements Communications, 2010). Unlike credit risk directed at companies by exposure through a loan and where exposure is unilateral, where only the lending bank is running some kind of credit risk.

3.2 Counterparty Credit Risk (CCR)

One problem that was observed during the financial crisis was defaulting counterparties occurring at the same time as volatility in the market was at its highest. This resulted in a higher counterparty risk than otherwise. In addition, it was found that about two thirds of the CCR losses was due to "Credit Valuation Adjustment (CVA)" and that the remaining one-thirds were due to actual defaults (Kroon & Lelyveld, 2018).

The Basel Committee has proposed a number of changes to the Basel III regulations to strengthen the capital requirement for CCR and the proposals are rooted in the reason behind the financial crisis. The CVA supplement is one of the proposals submitted by the Committee to better secure banks against counterparty risk, such as the IRB method that will be discussed in the next section.

3.2.1 Internal Rating-Based (IRB) approach

The internal rating-based approach, as the name implies, is an internal method that approved banks can use to calculate different risk measures.

Such risk measures are then used for the risk-weighted assets calculation in accordance with the necessary capital requirements. In other words, banks under the Basel guidelines, can use own risk measurements for the calculation of regulatory capital.

In order to calculate the capital requirements, there are three elements needed. Firstly, the *Risk parameters.* These parameters include the probability of default, exposure at default, loss given default and maturity. Secondly are the *risk-weight functions*. These functions map the different parameters to the respected risk-weighted assets. Lastly are the *minimum requirements.* The minimum requirements are requirements that a bank must satisfy in order to use the internal rating-based approach.

The Basel accord provides two broad methods that can be used by a bank:

1. Foundation approach
2. Advanced approach

When applying the *foundation approach*, the bank calculates their own probability of default parameter, while the other risk factors are provided by the national supervisors. When the *advanced approach* is used, banks calculate all the risk parameters as long as certain minimum guidelines are satisfied (Bank for International Settlements Communications, 2010).

In the Basel III, there arise some changes to the previous accord regarding what methods can be used. Both Basel II and Basel III differentiates between different classifications of risk exposures and for what methods can be applied when measuring its components.

Table 2 – Revised Scope of IRB

Revised scope of IRB approaches for asset classes		
Portfolio/exposure	Basel II: available approaches	Basel III: available approaches
Large and mid-sized corporates (consolidated revenues > EURm 500	A-IRB, F-IRB, SA	F-IRB, SA
Banks and other financial institutions	A-IRB, F-IRB, SA	F-IRB, SA
Equities	Various IRB approaches	SA
Specialized lending	A-IRB, F-IRB, slotting, SA	A-IRB, F-IRB, slotting, SA

Source: (Basel Committee on Banking Supervision , 2017)

For Nordea, the counterparty risks are classified under "Banks and other financial institutions". In the previous accord the Standard Approach, Advanced Approach and Foundation Approach could be used. However, with the new framework, only the Foundation and Standard approach are accepted methods. (The standard approach uses external rating agencies for the calculation of the risk components.) Hence, with respect to the motivation of this thesis the relevant approach analyzed is the foundation approach.

Nordea's counterparties in relation to the OTC trade arrangements falls under the categorical term "Market risk" stapled by the Basel committee. This is due to that all trades are executed from the trading desk and that the different trades involve different type of financial instruments.

To be specific, Nordea asked for the risk assessment of the OTC counterparties. In accordance with §40 in the Basel Committee publication of the Minimum capital requirements for market risk (BIS, 2016), the following is written:

"Banks will be required to calculate the counterparty credit risk charge for OTC derivatives, repo-style and other transactions booked in the trading book, separate from the capital charge for general market risk" (BIS, 2016).

By adhering this method of risk assessment, Nordea must follow the same approach as for Credit risk in the banking book. This means that the internal rating-based approach is the relevant method for the OTC counterparty risk assessment. Hence, Nordea is able to calculate their own risk components, such as the probability of default.

3.2.2 **Probability of default**

Under the classification of Banks and other financial institutions, the probability of default is defined as the likelihood of a default within a one-year period.

For the calculation of the probability of default with the application of any internal rating-based approach, there are certain requirements that must be attained. Specifically, the estimation must reflect the counterparties involved and transaction characteristics. Also, the estimation must hold a certain consistency and be accurate when estimating the risk. The estimation must be logical and documented, so that replication of the method is possible for regulatory entities. Any scrutiny of such methods should yield a model that in no way provokes a rating system that favor systems minimizing regulatory capital requirements.

In terms of data quality, the internal estimates must take into consideration all possible internal and external data available. The data used for estimation must be based on sound historical and empirical evidence so to limit decisions based purely judgmental. Lastly, for the parameter estimates, a layer of conservatism should be added to reflect potential errors in the estimations that can occur.

The model developed can be based upon the following techniques; internal default experience, mapping to external data and statistical default models.

4 Conceptual framework

As this paper wish to exploit opportunities by using artificial intelligence to calculate the default probability for different counterparties. This chapter will present a short introduction to Artificial Intelligence, before moving on to the subsection Machine Learning and the relevant methods of calculation for probability of default.

4.1 Artificial Intelligence

Artificial intelligence is a common term used for machine intelligence. All sub-terms are based on the same goal, learning or programming a machine to perform or reach a specific objective without any task-specific programming. There exist a variety of Artificial Intelligence fields, such as robotics, voice recognition and machine learning. Common for all is to "mimic" the human brain and how it reacts and responds to problems by observing, analyzing and imminently learning from past experiences (Poole, Mackworth, & Goebel, 1998). In our case, we which to use machine learning.

4.2 Machine learning

Machine learning is a subsection of artificial intelligence. (Samuel, 1959) defines machine learning as a collective term for methods that have the ability to learn without explicitly being programmed. It involves machines learning from historical input data to develop a wanted behavior. By using statistical models, mathematical optimization and algorithms, the machine can find complex patterns in a dataset and take intelligent decisions based on these discoveries.

This learning is then applied when looking at other companies in the future. The goal is similar to the linear approximation, where the network map's the input variables to the dependent variables (McNelis, 2005). When working through the dataset, the system should in the future be able to identify corporations that most likely default.

The three types of learning structures within Machine Learning will be presented in the following sections

4.2.1 Supervised learning

Supervised learning operates with labeled input data. This means that the algorithms learn to predict the given output from the input data. Learning evolves around creating training sets where the algorithm is provided with correct results. The aim is then for the network to learn and find connections between the input and output pairs.

If the algorithm predicts the wrong result, it adjusts the weights in the model. This type of learning is applied in cases were the network has to learn to generalize the given examples. A typical application is classification. In this example, a given input has to be labeled as one of the defined categories. This is done by using the algorithm as a mapping function. During the training process, as the results are known, the process stops if the algorithm achieve an acceptable level of performance.

The two main subsections of supervised learning problems are regression and classification problems:

- Classification: A classification problem is when the output variables is a category, such as red or blue
- Regression: A regression problem is when the output variable is a real value

4.2.2 Unsupervised learning

With unsupervised learning, the data infused to the model is unlabeled. Hence no dependent variables are provided. The algorithm then develops structures and systems to find patterns in the data. This means that the model itself creates desired outputs. Different algorithms can be used with unsupervised learning to guide the networks adaption of its weights and self-organize.

As the learning is unsupervised, there are no correct answers (or penalties given to the model). The algorithm is simply used to discover and resent acknowledgeable patterns or structures. This type of learning is mostly used when the data modeler believes there exists underlying structures as well as distributions in the data, making the process relevant for both data mining and clustering.

The unsupervised learning problems are subdivided into two main objectives. These are association or clustering.

- Clustering: Clustering explore similar patterns for different data points. Which makes it possible to group subsets. One example is using purchasing behavior to classify customers.
- Association: Aims at mapping segments of the data by creating rules to describe patterns. One example could be to find relations between a customer that buys product x also tends to buy product y.

4.2.3 Reinforced learning

Reinforced learning trains the network by introducing prizes and penalties as a function of the network response. Prizes and penalties are then used to modify the weights. Reinforced learning algorithms are applied, for instance, to train adaptive systems which perform a task composed of a sequence of actions. The outcome is the result of this sequence. Therefore, the contribution of each action must be evaluated in the context of the action chain produced.

4.3 Supervised algorithms

As this thesis aims at explaining the relation between input data and the likelihood of default based on historical data, the teaching method with most relevancy is the supervised learning. There exists a great amount of supervised learning algorithms that can be used for prediction problems. A learning algorithm is constructed by a “loss” function and an optimization technique with the goal of finding the correct weightings for the model.

The loss function is the penalty rate, when estimations from the model is too far off the expected result. The optimization technique tries to limit the prediction errors. The different algorithms use different loss functions and different optimization techniques.

Each algorithm has its own style or inductive bias and it is not always possible to know which algorithm is most suitable for one specific problem. Therefore, one need to experiment with several different algorithms to see which algorithm provide satisfactory results.

As the goal of any algorithm used is to find out the probability of a company defaulting or not, the problem falls under the category of binary classification. Thereby, the experiment can be limited so to only use algorithms developed for that purpose.

Algorithms that are used for classification problems are subdivided into two categories. Namely discriminative and generative. Generative models are a statistical model of the joint probability distribution of X·Y where x is the input and y is the prediction.

The prediction is done by applying Bayes theorem to calculate *Px*, and then to select the most likely outcome based on a threshold. (It is the Px value that will yield the wanted probability of default for the counterparties). The discriminative model is a model of the conditional probability that gives *Px*, which makes it possible to predict y when the value of *x* is given.

Generative programming provides a richer model with more insights to how the data was generated. This makes the model more flexible as it is possible to assign conditional relation between data points, generate synthetic data or adjust for missing data.

Discriminant learning does not yield the same insights, as its only focus is to predict the result y given x. As the generic model is richer with respect to data insights, it requires higher computational power. Further, discriminant models are proven superior because it only focuses on the actual task the machine need to solve. Therefore, when not in need of the insights the generative models provide, it is more profitable to use a discriminatory model.

5 Theoretical Framework

The underlying theory for the various models is presented in this chapter. Here with special emphasis on the mathematical construction of the different models. All presented models will be introduced in its simplest form before adding, different relevant components or dimensionalities. Such as the different activation functions, kernels and other tuning parameters. Before any of the models are introduced, as theoretical framework of the different models' objectives is explained. With special emphasis on binary classification where the data is liner or non-linear. Also, the logistic regression will be presented, being the underlying benchmark for the models.

5.1 Linear and Non-Linear Classification

The importance of classification is absolute for finding good and bad counterparties. (Breiman, Random Forests, 2001) defines statistical classification as "the problem of identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations whose category membership is known". Meaning that the ability to split the data, distinguish between groups and differentiate random noise from the data, is evident for all good classification models.

The classifiers main application is for predicting unobserved data. The functions separate two classes by applying a hyper-plane or a line to separate the classes in different dimensions. Standard theory presents two methods for creating such boundaries, either linearly or non-linearly. Which is decided by the shape of the *decision boundary*. For the more formal definition and underlying math, we have focused heavily on the theory provided by (Hastie, Tibshirani, & Friedman, 2013).

Given a data universe R with two classes of observations, say X and Y. The separating hyperplane is constructed by a linear boundary. There can be multiple separating hyperplanes, and we will therefore first present the general theory before moving on to the *optimal separating hyperplane*.

Constructing the separating hyperplane with a *decision boundary* can be formulated as:

5.1.1

$$f(X) = x^T\,\beta + \beta_0 = 0$$

Where β_0 is the intercept, also known as *bias* in machine learning and the *weight vector* is described as x^T. While *x* is the observed values.

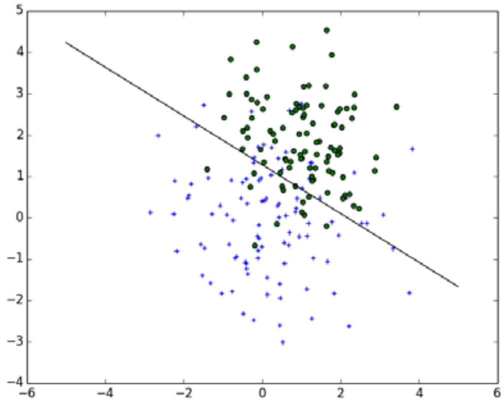


Figure 1 – Linear decision boundary

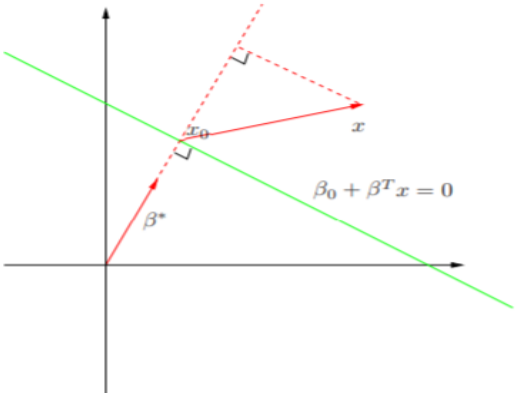


Figure 2 – Linear decision boundary equation

Source: (Hastie, Tibshirani, & Friedman, 2013)

Hastie et.al defines a hyperplane as:

$$\{x : f(x) = x^T * \beta + \beta_0 = 0\} \qquad 5.1.2$$

Where β is a unit vector: β = 1. The classification from here is straight forward, infused data-points yielding a value above 1 belongs to one of the classes as they are above the decision boundary. An observation value that belongs to -1 is thus the opposite and lays underneath the *decision boundary*.

We can therefore construct the decision function accordingly:

$$G(x) = sign(x^T\,\beta + \beta_0) \qquad 5.1.3$$

Where sgn() represents the sign function which produce output above one for positive parameters, -1 for negative parameters. With only two possible outcomes [-1, 1] the problem is known as the statistical binary classification problem.

There exists generalization for finding the optimal *decision boundary*. If the output domain consists of m classes, in example $y = \{1, 2, 3, \dots, \mathrm{m}\}$, the method is m-class classification. The aim is to minimize the variance from the predicted and known outputs by adjusting the weighting of the variable *vector* x^T and the constant term b (Hastie, Tibshirani, & Friedman, 2013).

Discriminative learning is the most common approach for this problem. The discriminative learning aims at finding the optimal relation between the inputted variables and the dependent variables. This without any assumption regarding the underlying distributions of all relevant variables. In other words, the model attempts finding the conditional probability distribution $p\{\mathrm{y} \dots \mathrm{x}\}$ directly. As opposed to generative models, that attempts finding the joint probability distribution. Examples of discriminative models are Neural networks, Logistic regression and Support vector machines.

Discriminative learning uses geometrical interpretation of the linearity and input data, to find the boundaries. Their methods however distinguish by the different linear classifiers creating such boundaries. The conceptual idea of the different models is relatively similar.

Some classifiers are often more applicable, as they tend to outperform other types. There exists some common characteristics for evaluating classifiers. The ability of handling the linearly inseparable data.

- Finding non-linear relationships in the dataset and utilizing these relations
- Handling and classifying non-linearly data
- Capability of generalizing and reducing the impact of outliers as well as noise

In the following sections, some classifiers will lack the ability to deal with some of these points. While other classifiers have the ability to handle these issues.

5.2 Logistic Regression

The mathematical concept of the logistic regression is used in accordance with the theory presented by (Agresti, 2012) *and* (Hosmer, Lemeshow, & Sturdivant, 2013) *and its connection to the probability of default modelling* (Hastie, Tibshirani, & Friedman, 2013).

Ever since David Cox developed the model in 1958 the logistic regression model has become somewhat of an industry standard. Due to its stable performance and easy implementation it has been heavily used in finance and other areas. Further (Ohlson, 1980) pointed out that the logistic regression provides highly interpretable coefficients compared with other models.

In the following section the mathematics underlying the logistic model is presented.

If considering a collection of n independent variables denoted by the vector $x' = (x_1, x_2 \dots, x_n)$. The dependent variable Y_x has a binary distribution:

$$Y_x \begin{cases} 1 = default \\ 0 = non - default \end{cases} \quad 5.2.1$$

Then the conditional probability for the outcome can be denoted:

$$P(\,Y = 1 \mid x\,) = \pi(x) \quad 5.2.2$$

The logit of multiple regression model is given by equation:

$$g(x) = B_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_n x_n \quad 5.2.3$$

In which case the logistic regression model's final form is described by (Hosmer, Lemeshow, & Sturdivant, 2013):

$$\pi(x) = \frac{e^{\beta' x}}{1 + e^{\beta' x}} \quad 5.2.4$$

The primary objective is to define an appropriate model to capture the dependence of the probability of default on the vector for the input variables.

To receive our desired results, we can apply the odds-function as the ratio for the probability of default.

$$OR(x) = \frac{P(Y_x = 1)}{P(Y_x = 0)} = \frac{\pi(x)}{1 - \pi(x)} \tag{5.2.5}$$

However, this function is mapped into the interval $(0, \infty\ )$ because the *Odds-Ratio* can take on any real-value. While in our case, the probability $\pi(x)$ ranges between one and zero. We therefore apply the logit transformation from (Hosmer, Lemeshow, & Sturdivant, 2013).

$$g(x) = \ln(\frac{\pi(x)}{1 - \pi(x)} = B_0 + \ \beta_1\, x \tag{5.2.6}$$

Therefore, we end up with the formula for a logistic regression within our desired interval. The final form is as mentioned in (Hastie, Tibshirani, & Friedman, 2013):

$$\pi(x) = \ \frac{e^{\beta' x}}{1 + \ e^{\beta' x}} \tag{5.2.7}$$

Other possible transformation is an application of the distribution function Φ of standard normal distribution, known as *probit* (Hastie, Tibshirani, & Friedman, 2013):

$$probit(x) = \phi^{-1}\big(\pi(x)\big) \tag{5.2.8}$$

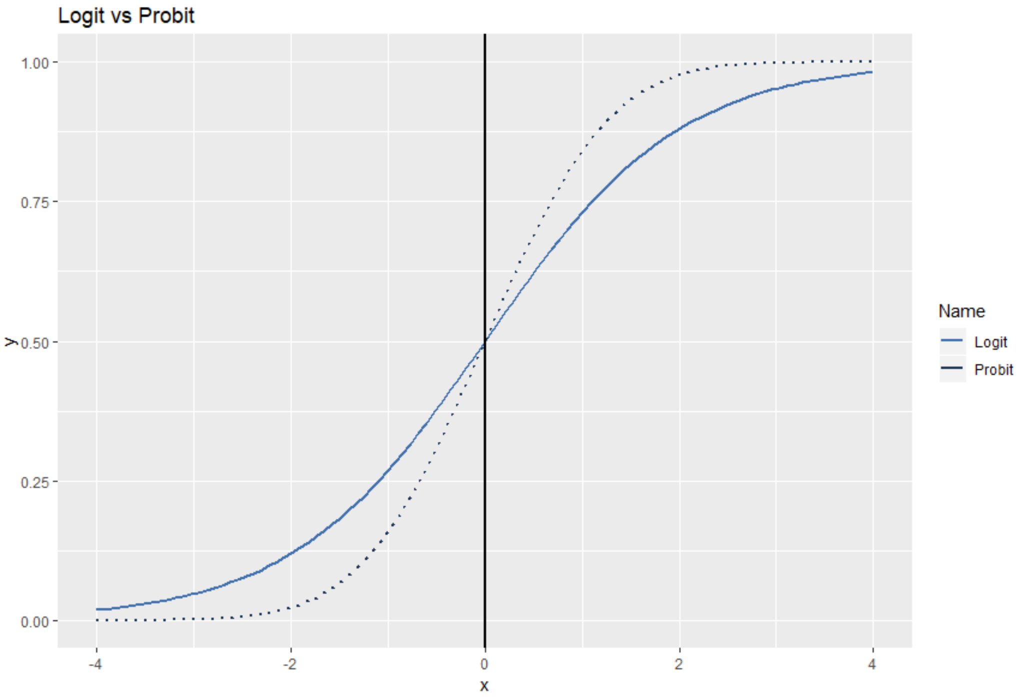


Figure 3 – Logit vs Probit

Source: Produced in RStudio

The main advantage of logit is its closed form. Making it not only easier to compute, but also offering a better understanding and interpretation of change in the parameters. This is suitable when calculating the odds effect for a change in the vector value x_i.

5.3 Neural Networks

The structure and theory in this section is inspired by (McNelis, 2005). Where we have decided to focus on Feedforward Networks, Jump Connections and Multi-layered Feedforward Networks for the task at hand. To start with, a short introduction of Neural Networks as a concept is presented.

Neural networks are machine learning systems based on a simplified model of the biological neuron (Haykin, 2009). Similar to the behaviour of the biological neuron, neural networks modify their internal parameters in order to perform a given computational task. Both linear models and neural networks aim to transform a set of given input variables into a set of output variables.

The difference between other approximation methods and neural networks is that neural networks uses a hidden layer. “In which the input variables are squashed or transformed by a special function, known as a logistic or logismoid transformation” (McNelis, 2005). Where the special function often is referred to as the activation function. The hidden layers can often be hard to understand due to its “black-box” structure, nevertheless they can be extremely powerful in the effort for making nonlinear relationship models.

The two main issues to be defined in a neural network application are the structure of the network typology and the learning algorithm (In example, the procedure used to adapt the network so to solve the computational task at hand).

5.3.1 Feedforward Networks

The figure underneath presents a simplified structure of the feedforward network. It can be seen that the network composes three input variables $\{xi\}$, i=1,2,3….,n, an additional layer, known as the “hidden layer” *n,* and lastly the output variable *y* (McNelis, 2005).

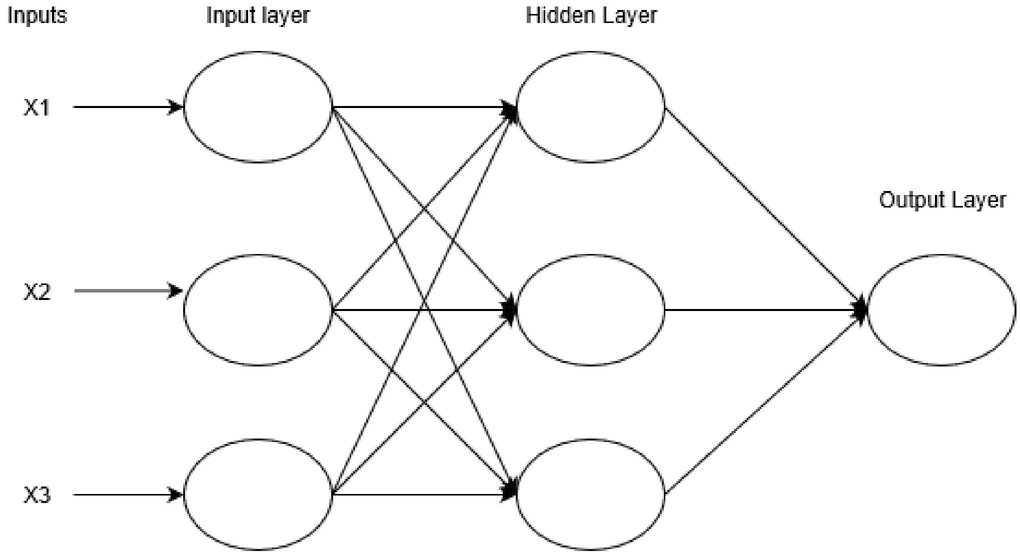


Figure 4 – Architecture of Basic Feedforward Network

Source: Illustration inspired by (McNelis, 2005)*.*

The hidden layer is useful for parallel calculation of information as compared to, in example, linear models applying sequential processing. The parallel processing gives an advantage in the estimation of outputs due to its ability of running several different calculation processes at once. Imminently resulting in multiple outputs that can expose different variations of the final result (McNelis, 2005). Therefore, not being limited by the disadvantage of linear optimization.

Typical tasks neural networks perform efficiently and effectively are: Classification, recognizing patterns in data and prediction.

Neural networks generation of an output is divided in two steps, *weights* and *activation*. Firstly, the weighted sum of inputs are transformed into linear combinations:

$$y_i = \sum_{j \in I} W_{j,i} a_j \quad input\ evaluation \qquad 5.3.1.1$$

Secondly, the linear combinations are processed by an activation neuron.

$$a_i = g(y)\ \ activation\ function\ g \qquad 5.3.1.2$$

Where $W_{j,i}$ is the weight of the link connection neuron j with neuron i and a_i is the activation of neuron j. (Angelini, di Tollo, & Roli, 2008). The activation function *g* can be any function, where the most common is linear, step, Gaussian, tansigmoid or logismoid functions.

The following graph visualises the behaviour for different activation functions from the input layer to the output neutron.

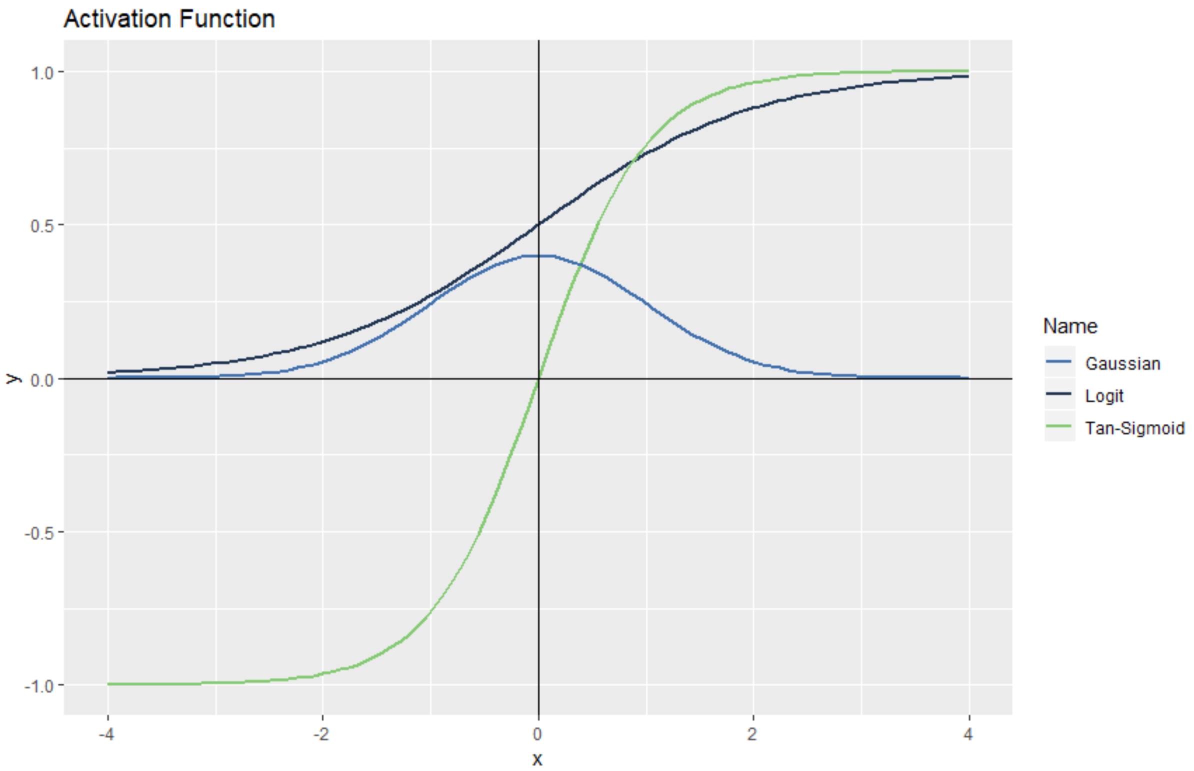


Figure 5 – Different activation functions

Source: Produced in RStudio

The attractiveness of the sigmoid function, or more commonly known as the logit function, is due to its binary behaviour. First of all, it is bounded between zero and one, and its derivative is easy to calculate. Secondly, it can reasonably well describe the majority of the different responses to development in underlying variables.

Hence, the network is able to distinguish between large changes in outliers not relevant for the population as a whole, and rather highlight smaller changes in relevant observations reflecting the majority of the data-set.

The mathematical expression for a simple network with one hidden layer as described by (McNelis, 2005).

$$n_{k,t} = \omega_{k,0} + \sum_{i=1}^{i} \omega_{k,i} x_{i,t} \qquad 5.3.1.3$$

$$N_{k,t} = L(n_{k,t}) = \frac{1}{1 + e^{-n_{k,t}}} \qquad 5.3.1.4$$

$$y_t = \gamma_0 + N_{k,t} = \sum_{k=1}^{k} \gamma_k N_{k,t} \qquad 5.3.1.5$$

If we create an index i containing all of the inputted variables $\{x\}$\{x\}, and also an index k for all of the neurons enabling the parallel processing, we can use the logismoid activation function $L(nk)$. With the application of the activation function a multitude of linear connections of nk is produced with different functions of weights wk, and the constant term wk,0. The transformation of nk by the activation function creates neurons for the different observations t, resulting in the different neurons Nk, for all observations t. The combinations of the linear functions with the specific neurons creates a vector containing all of the coefficients $\{yk\}$ and the constant term y0. The vector is then used to calculate the forecasted $\hat{y}t$ at observation t.

As described earlier, the logismoid is preferred as an activation function for generating outputs of probability. There exist alternatives such as tansigmoid to use as activation functions instead. Depending on the task at hand, the different functions all serve their purpose as activation functions. To illustrate the change of activation function in a mathematical context, we present the calculation described by (McNelis, 2005):

$N_{k,t} = L(n_{k,t})$ is replaced with $N_{k,t} = T(n_{k,t})$ 5.3.1.6

Which gives $N_{k,t} = T(n_{k,t}) = \frac{e^{n_{k,t}} - e^{-n_{k,t}}}{e^{n_{k,t}} + e^{-n_{k,t}}}$ instead of $N_{k,t} = L(n_{k,t}) = \frac{1}{1+e^{-n_{k,t}}}$

As we can see the only change in the end formula is the interpretation of $N_{k,t}$,. We can follow the same procedure for the Gaussian function, also an alternative activation function.

$N_{k,t} = L\left(n_{k,t}\right)$ is replaced with $N_{k,t} = \phi\left(n_{k,t}\right)$ 5.3.1.7

Giving us $N_{k,t} = \phi\left(n_{k,t}\right) = \int_{-\infty}^{n_{k,t}} \sqrt{\frac{1}{2\pi}} e^{\frac{-n_{k,t}^2}{2}}$ 5.3.1.8

As mentioned, neural networks perform classification in a highly effective way. To illustrate decision boundaries for aforementioned *activation functions*, see below.

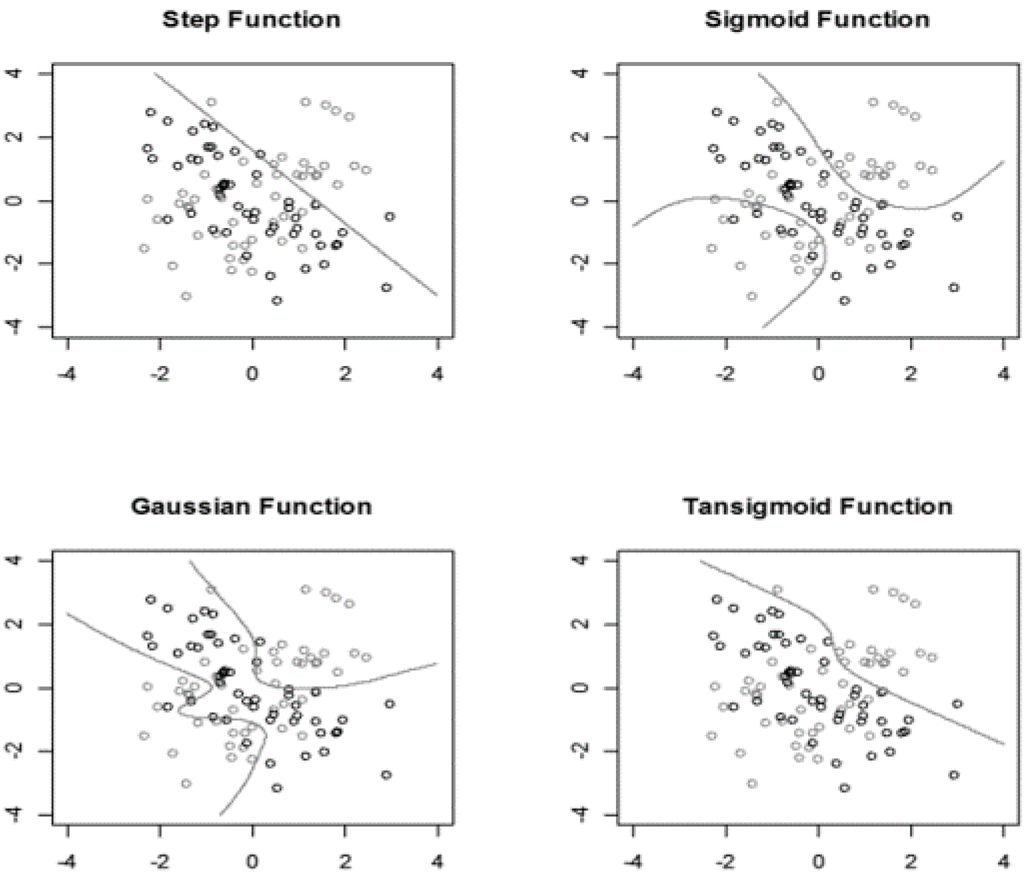


Figure 6 – Separation using Neural Network with Different Activation Functions

Source: Visualisation inspired by (Ozaki, https://tjo-en.hatenablog.com, 2015)

5.3.2 Jump Connections

The previous section introduced a simple feedforward architecture. The following section will present an extension to the simple network. One example, and as shown in the figure underneath, is jump connections. Where the inputs can have a direct linear link to the output *y* (McNelis, 2005).

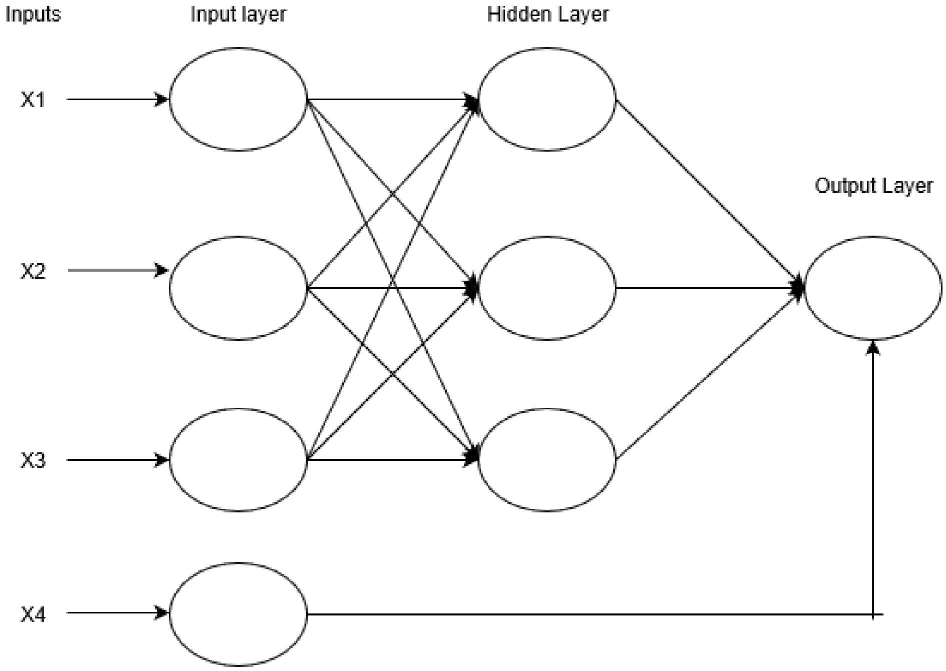


Figure 7 – Feedforward Network with Jump Connections

Source: Illustration inspired by (McNelis, 2005).

$$n_{k,t} = \omega_{k,0} + \sum\nolimits_{i=1}^{i} \omega_{k,i} x_{i,t} \qquad 5.3.2.1$$

$$N_{k,t} = L(n_{k,t}) = \frac{1}{1 + e^{-n_{k,t}}} \qquad 5.3.2.2$$

$$y_t = \gamma_0 + \sum_{k=1}^{k} \gamma_k N_{k,t} + \sum_{i=1}^{i} \beta_i x_{i,t} \qquad 5.3.2.3$$

As we can see, the only difference from the original feedforward network is the additional coefficient β_i. Which represents the direct link from input x with output y. The advantage of this, is that the non-linear and linear calculations gets gathered as a combined version of the non-linear and linear components. However, this also means that computational process is extended due to the increased parameters (McNelis, 2005).

5.3.3 Multi-layered Feedforward Networks.

When opposing problems with increased complexity, we can extend the network by adding multiple hidden layers. This type of feedforward network is known as “Multi-layered Feedforward networks”. The mathematical representation as described by (McNelis, 2005) follows:

$$n_{k,t} = \omega_{k,0} + \sum_{i+1}^{i} \omega_{k,i} x_{i,t} \tag{5.3.3.1}$$

$$N_{k,t} = \frac{1}{1 + e^{-n_{k,t}}} \tag{5.3.3.2}$$

$$p_{l,t} = \rho_{l,0} + \sum_{k=1}^{k} \rho_{l,i} N_{k,t} \tag{5.3.3.3}$$

$$P_{l,t} = \frac{1}{1 + e^{-p_{l,t}}} \tag{5.3.3.4}$$

$$y_t = \gamma_0 + \sum_{l=1}^{l} \gamma_l P_{l,t} \tag{5.3.3.5}$$

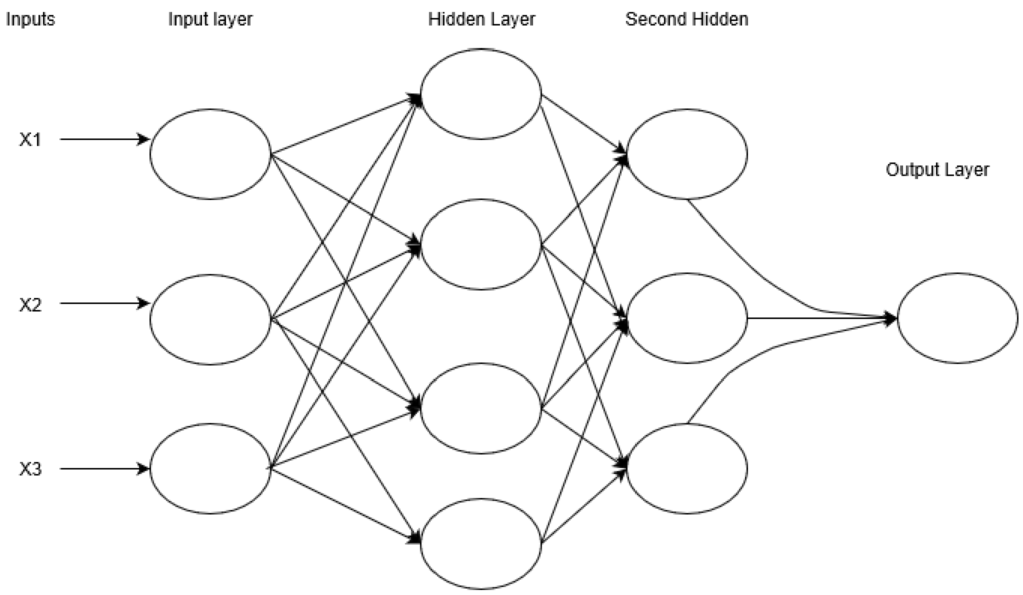


Figure 8 – Architecture of Multilayered Feedforward Network

Soruce: Illustration inspired by (McNelis, 2005).

As seen in Figure 8 and in the mathematical system the second hidden layer is described as P_i. It should be emphasized that adding additional hidden layers increases the number of parameters to be estimated.

A more complex architecture as shown in Figure 8 allows for higher complexity, which can improve productiveness on more advanced problems. However, the negative impacts are that we estimate much more parameters, which increase computation time and effort. With more parameters, there is also the likelihood that the parameter estimates may converge to a local, rather than global optimum (McNelis, 2005).

5.4 Support Vector Machines

Support Vector Machines (SVM) is a relatively new learning method and has since the 1995 been one of the most popular learning methods used for regression and binary classification. Its current structure, introduced in 1992 by Boser, Guyon and Vapnik, is known as a method with high prediction power. In addition, the algorithm can be adjusted so to limit over-fitting the prediction, and thereby improve the accuracy of the model. The following section will introduce the application of the Support Vector Machine.

The crucial idea of the SVM is that it uses the geometrical concept of hyperplanes, which separates multidimensional data into classes. When the data is not linearly separable, SVM introduces the approach of kernel tricks, or “kernel induced feature space” (Cortes & Vapnik, 1995). The kernel trick creates a higher dimensional space, where it is possible to separate the data linearly, thereby splitting positive and negative data points in the multi-dimensional space.

The hyperplane serves as a function for creating a boundary between the different classifications’ groups, so that imaginary margins between classes are maximized. In a two-dimensional space, the positives and negatives are separated by a line. If the space is three-dimensional, a plane separates the different data points. Lastly, if the space is of n-dimensions, the classes are separated using a hyperplane (Elizondo, 2006).

The figure below, illustrates a classification in a two-dimensional space, where the two groups are separated with a line.

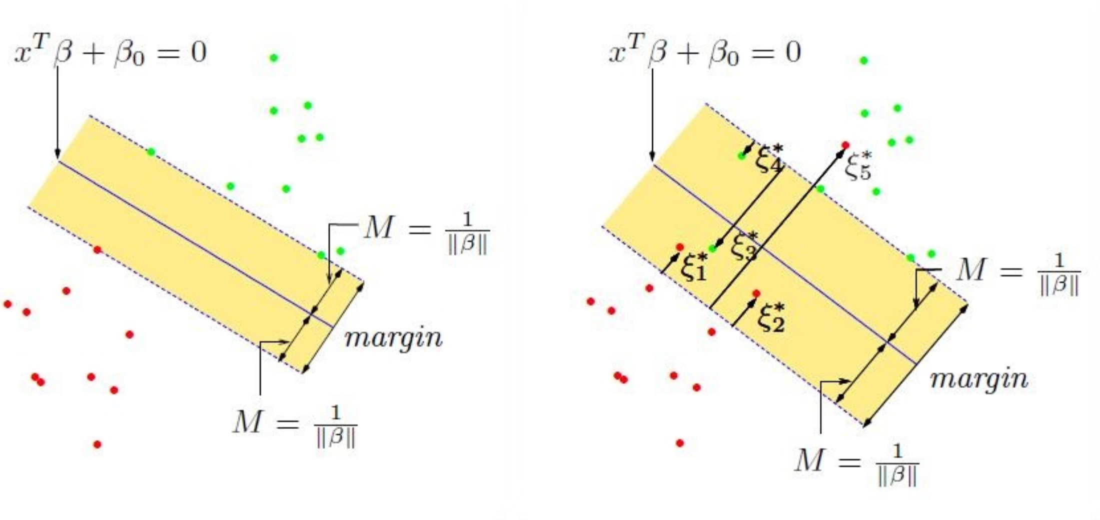


Figure 9 – Separable and Non-separable classification

Source: (Hastie, Tibshirani, & Friedman, 2013)

The figure illustrates the two cases of separable and non-separable classification. The decision boundary is the solid line, whereas the dotted lines bound the maximal margin with distance $2M = \frac{2}{||\beta||}$. For the non-separable case on the right panel, there can be observed data points overlapping the decision boundary. Here the points labeled $\varepsilon^*$are on the wrong side of the boundary with size $\varepsilon_j^* = M\varepsilon_j^*$. (The observations on the correct side of the margin holds the error value $\varepsilon_j^* = 0$) The boundary in the non-separable case are maximized in accordance to a budgeted relaxation equal to $\sum \varepsilon_j^* \leq constant$. In other words, $\sum \varepsilon_j^*$ is the total allowed distance of observations on the wrong side of the decision boundary.

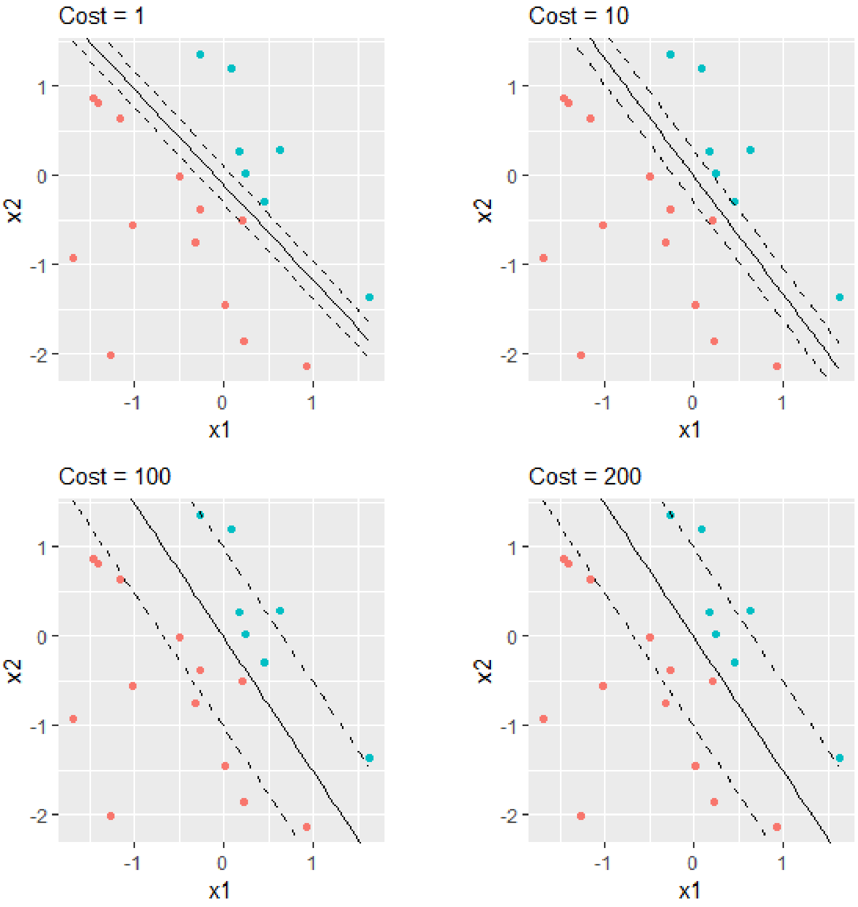


Figure 10 – SVM different cost functions

Source: Produced in RStudio

The graph shows how different cost parameters affect the distance $2M = \frac{2}{||\beta||}$. When $2M$ is increased the size of $\sum \varepsilon_j^*$ can increase. (Will be further explained in this chapter).This can be useful, when there are no clear boundaries, between two subsets. In an attempt of finding a feasible solution, it can be relevant relaxing the cost parameter.

The mathematical approach in the following subsections follows the same approach as presented by (Hastie, Tibshirani, & Friedman, 2013). Assuming a training set of N pairs $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, with $x_i \epsilon \ \mathbb{R}^p$ and $y_i \epsilon \{-1,1\}$. If β is a unit vector: $\left||\beta|\right| = 1$, and $f(x)$ is the induced classification rule, then $G(x)$ for two classes is set as:

$$G(x) = sign[x^T \beta + \beta_0]. \tag{5.4.1}$$

When defining a hyperplane (as in Ch. 5.1):

$$\{x: f(x) = x^T \beta + \beta_0 = 0\} \tag{5.4.2}$$

The induced classification rule $f(x)$ produces a positive or negative vector from the margin M to a point x_i in the data set. If the two classes are separable, there exist a function for the classification rule as follow:

$$f(x) = x^T \beta + \beta_0 \text{ subject to } \min_{\beta,\beta_0} \left||\beta|\right| \rightarrow y_i f(x) \geq 1, i = 1, \dots, N, \tag{5.4.3}$$

If the classes overlap in the future space, we can relax the cost parameter:

$$y_i(x_i^T \beta + \beta_0) \geq M(1 - \varepsilon_i), \forall_i \varepsilon_i y_i \geq 0, \sum_{i=1}^N \varepsilon_i \leq constant. \tag{5.4.4}$$

The idea of $M(1 - \varepsilon_i)$ is to allow some of the observations to be on the other side of the decision boundary. The slack variables are then $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_i)$. Hence ε_i in the constraint $y_i(x_i^T \beta + \beta_0) \geq M(1 - \varepsilon_i)$, is the corresponding amount when $f(x_i) = x_i^T \beta + \beta_0$ is on the wrong side of decision boundary (Hastie, Tibshirani, & Friedman, 2013). In other words, there will be a misclassification when $\varepsilon_i > 1$. By bounding this relaxation at some constant value, puts an acceptance criterion on the total amount of training misclassifications.

The alternative of relaxing the boundary makes the SVM attractive, as it is possible to adjust the effect of outliers. To specify, observations that are far away from the decision boundary are less influential in shaping the hyperplane (Cristianini & Shawe-Taylor, 1999).

5.4.1 Support Vector Classifier

The computation of the Support Vector classifier is complex from a mathematical standpoint. This section is therefore merely to present the mathematical relevancy of the classification problem. A more advanced and in-depth presentation of the issue can be found in (Hastie, Tibshirani, & Friedman, 2013).

As presented above, the modification of the constraint with overlapping classification in the future space can be solved by relaxing the constrain in accordance with $M(1-\varepsilon_i)$. By using the measurement of actual distance from the boundary, the result is a convex optimization problem.

Firstly, the support vector classifier is defined for the non-separable case as follow:

$$\min\left|\left|\beta\right|\right| \, subject\ to \begin{cases} y_i(x_i^T\beta+\beta_0) \geq 1-\varepsilon_i \ \forall_i, \\ \varepsilon_i \geq 0, \sum \varepsilon_i \leq constant. \end{cases} \qquad 5.4.1.1$$

Thus, the programming solution is quadratic with linear inequality constraints. It follows that the solution can be stated using Lagrange multipliers.

$$\underset{\beta,\beta_0}{\text{minimize}} \|\beta\|^2 + C\sum_{i=1}^{N}\varepsilon_i \qquad 5.4.1.2$$

5.4.1.3

$$Subject\ to\ \varepsilon_i \geq 0; y_i(x_i^T\beta+\beta_0) \geq 1-\varepsilon_i$$

It can be observed from the equation that the parameter C replaces the constant. (In the separable case the constant C corresponds to $C=0$.)

The Lagrange (primal) function is

$$\min_{\beta,\beta_0,\varepsilon_i} LP \rightarrow L_p = \tfrac{1}{2}\|\beta\|^2 + C\textstyle\sum_{i=1}^{N}\varepsilon_i - \sum_{i=1}^{N}\alpha_i[y_i(x_i^T\beta + \beta_0) - (1 - \varepsilon_i)] - \sum_{i=1}^{N}\mu_i\varepsilon_i{}' \qquad 5.4.1.4$$

Setting the relevant derivatives to 0 gives us:

$$\beta = \sum_{i=1}^{N}\alpha_i y_i x_i \qquad 5.4.1.5$$

$$0 = \sum_{i=1}^{N}\alpha_i y_i \qquad 5.4.1.6$$

$$\alpha_i = C - \mu_i, \forall_i \qquad 5.4.1.7$$

Further, the positive constraint is set to $\alpha_i, \mu_i, \varepsilon \geq = 0\ \forall_i$. Then, by substituting $\beta\ and\ \alpha_i$ in the primal function, the resulting function is the Lagrangian (Wolfe) dual objective function:

$$L_D = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N}\alpha_i\alpha_{i'}\, y_i y_{i'} x_i^T x_{i'} \qquad 5.4.1.8$$

The presented function produce a lower boundary for all appropriate points in the primal Lagrange function (Hofmann, Scholkopf, & Smola, 2008).

The function Ld is then maximized subject to $0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{N}\alpha_i y_i = 0$. By also employing the Kuhn-Tucker conditions, the following constraints are obtained:

$$\alpha_i[y_i(x_i^T\beta + \beta_0) - (1 - \varepsilon_i)] = 0 \qquad 5.4.1.9$$

$$\mu_i\varepsilon_i = 0 \qquad 5.4.1.10$$

$$y_i(x_i^T\beta + \beta_0) - (1 - \varepsilon_i) \geq 0 \qquad 5.4.1.11$$

For $i = 1, \dots, N$. By the inclusion of the Kuhn-Tucker constraints and the referenced functions above equates the solution of the primal and dual problem, and that the solution for β has the form:

$$\hat{\beta} = \sum_{i=1}^{N} \hat{\alpha}_i y_i x_i \qquad 5.4.1.12$$

For the observations i in which the constraints in (5.4.1.11) are exactly met as from (5.4.1.9) the $\widehat{\alpha_\iota}$ coefficients are nonzero.

These observations are what is known as support vectors, due to the fact that $\hat{\beta}$ is defined by them alone. Some of these support observations, lies on the edge of the boundary, thereby holding a zero distance from the margin. It follows from the equation (5.4.1.10) and (5.4.1.7), that when $\widehat{\varepsilon_\iota} = 0 \; then \; 0 < \widehat{\alpha_\iota} < C$, and that for values were $\widehat{\varepsilon_\iota} > 0$, $\widehat{\alpha}_i = C$.

By looking at (5.4.1.9), any of the points that lies on the margin can be used for solving β_0. For numerical stability, the average value of the solutions are normally applied.

When knowing the solution for both $\hat{\beta}_0 \; and \; \hat{\beta}$, the final function optimizing $\hat{G}(x)$is therefore described as:

$$\hat{G}(x) = sgn\left[\hat{f}(x)\right] = sgn[x^T\hat{\beta} + \hat{\beta}_0] \qquad 5.4.1.13$$

5.4.2 Kernels

Up to this point, the general description of the Support vector machine has been presented. Further, an introduction to the calculation of the support vector classifier and its solution were illustrated. The classifier of the support vectors so far found linear boundaries in the input feature space. However, this is not always the case. By continuing the example of separating datasets in a two-dimensional space. The figure below shows how non-separable datasets can be mapped into a higher dimensional space.

This is done in order to separate the data in the new dimension, thereafter, classify and transform the data back into the original space. In other words, employing a suitable function for mapping the observations into a higher dimensional space, and then separate the classes with a linear hyperplane.

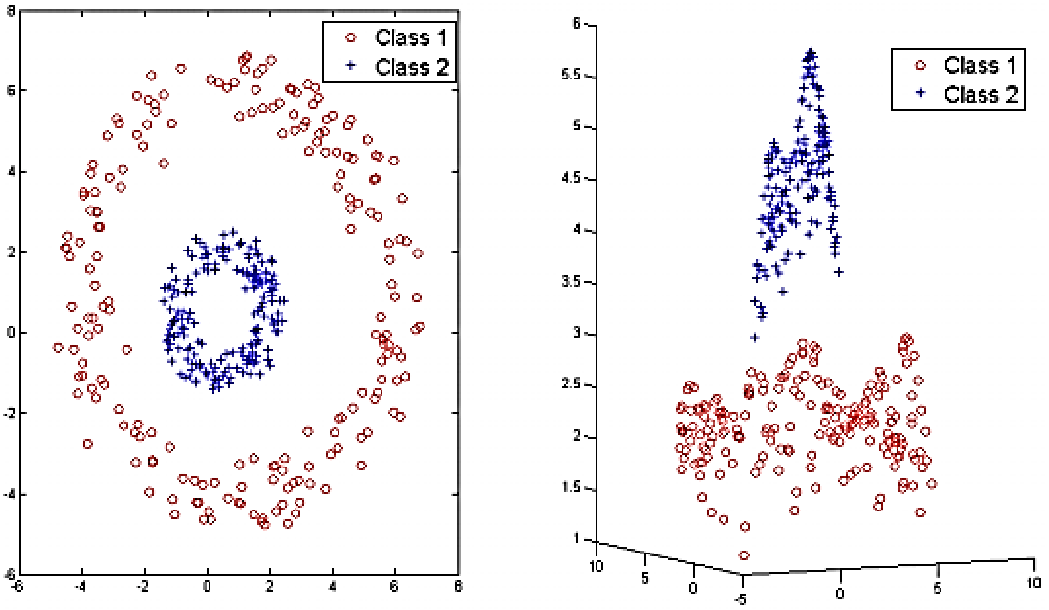


Figure 11 – Example of separating two classes in a new dimension

Source: (Roemer, 2018)

If assuming as before: $\{x: f(x) = x^T\beta + \beta_0 = 0\}\ for\ \beta = \sum_{i=1}^{N} \alpha_i y_i x_i$ the solution function $f(x)$ can be written:

$$f(x) = h(x)^T\beta + \beta_0 = \sum_{i=1}^{N} \alpha_i y_i\ \langle h(x_i), h(x_{i'})\rangle + \beta_0 \tag{5.4.2.1}$$

As before, given α_i, β_0 can be determined by solving $y_i f(x) = 1$ in (ref) for any (or all) x_i for which $0 < \alpha_i < C$.

Then, to fit the Support vector classifier using the input feature space $h(x_i) = \big(h_1(x_i), h_2(x_i), \dots, h_m(x_i)\big),\ i = 1, \dots, N$ includes $h(x)$ only through inner products. Here, examples of inner products can be the length of a vector, or the angle between two vectors.

The main point is that the procedure of the classification in its essence is the same as without any kernel transformation. There is no need to specify the transformation $h(x)$, but simply to understand the kernel function that computes the inner products in the transformed space. Hence, the computational cost is the same.

As the classification procedure is the same, the dual function can be reformulated as:

$$L_d = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N} \alpha_i \alpha_{i'}\ y_i y_{i'}\ \langle h(x_i), h(x_{i'})\rangle \qquad 5.4.2.2$$

And the solution for $f(x)$ can be rewritten

$$f(x) = h(x)^T\beta + \beta_0 = \sum_{i=1}^{N} \alpha_i y_i \langle h(x_i), h(x_{i'})\rangle + \beta_0 \qquad 5.4.2.3$$

As seen, both Ld and $f(x)$ only involve $h(x)$ through inner products. Hence, a mentioned, there is no need to specify the transformation $h(x)$.

If K is defined as a symmetric positive (semi-) kernel function, then the function can be written as:

$$K(x, x') = \langle h(x_i), h(x_{i'})\rangle \qquad 5.4.2.4$$

Where K is the computation of the inner products in the transformed space.

Four popular choices for K in the SVM literature are:

$$Linear: K(x, x') = x \cdot x' \qquad 5.4.2.5$$

$$Polynomial\ of\ d-th\ degree: K(x, x') = (1 + \langle x, x' \rangle)^d \qquad 5.4.2.6$$

$$Radial\ basis: K(x, x') = \exp(-\gamma \|x - x'\|^2) \qquad 5.4.2.7$$

$$Sigmoid: K(x, x') = \tanh(k_1 \langle x, x' \rangle + k_2) \qquad 5.4.2.8$$

The figure below gives an example of the different Kernel functions and their separation:

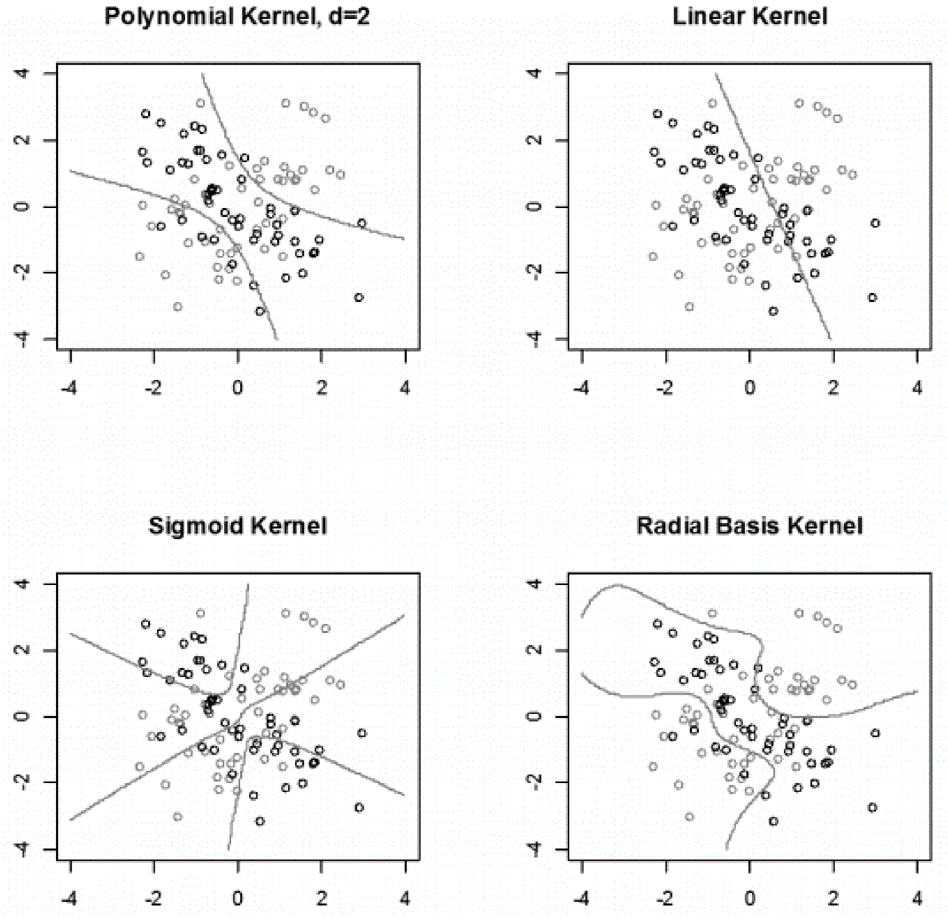


Figure 12 – Kernels and their approach for classification

Source: Visualization inspired by (Ozaki, www.tjo-en.hatenablogg.com, 2015)

To our knowledge, there are no general rules for the selection of an optimal kernel function. The selection varies depending on the complexity of the data. In example, for datasets with linear relationships the Linear Kernel should be adequate. Further, applying Kernels that are more complex should not yield any improvements. If the data are of complex structures, in example no linear relations, there should be significant improvements by applying functions that are more complex. Such complexity in the data often arises from graphical analysis, text analysis or audio analysis. Calculation of the default probability generally is between these extremes.

5.4.3 Regression

In this section, the adaptation of the Support Vector Machines for regression is presented. The adaption is done with a quantitative response in order to employ some of the properties from the classification function. Hence, SVM regression can be used to separate different classes.

To discuss the base line, the linear regression model is defined as:

$$f(x) = x^T\beta + \beta_0 \qquad 5.4.3.1$$

To handle nonlinear generalization, the estimation of β is done by the minimization of:

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V\big(y_i - f(x_i)\big) + \frac{\lambda}{2}\|\beta\|^2 \qquad 5.4.3.2$$

Where

$$V_\varepsilon(r) = \begin{cases} 0 & ; \quad |r| < \varepsilon \\ |r| - \varepsilon & ; otherwise \end{cases} \qquad 5.4.3.3$$

For $V_\varepsilon(r)$ the "$\varepsilon - sensitive$" is an error measure, so to ignore errors of size lower than ε. Thereby limiting points far away from the decision boundary as well as points on the correct side during the optimization. This is similar to the analogy of support vector classification setup.

To specify, the "$\varepsilon - sensitive$" should not be mixed with the calculation of the support vector points. The calculation is similar, but the selection is different. Rather, in terms of regression, theses points are the observations with small residuals. Hence, fitting of the regression model, makes it less sensitive towards outliers in the training-set.

The figure below is a graphical illustration of the $\varepsilon - sensitive$ error function exercised by the SVM. All points outside the decision boundary is not considered during the regression.

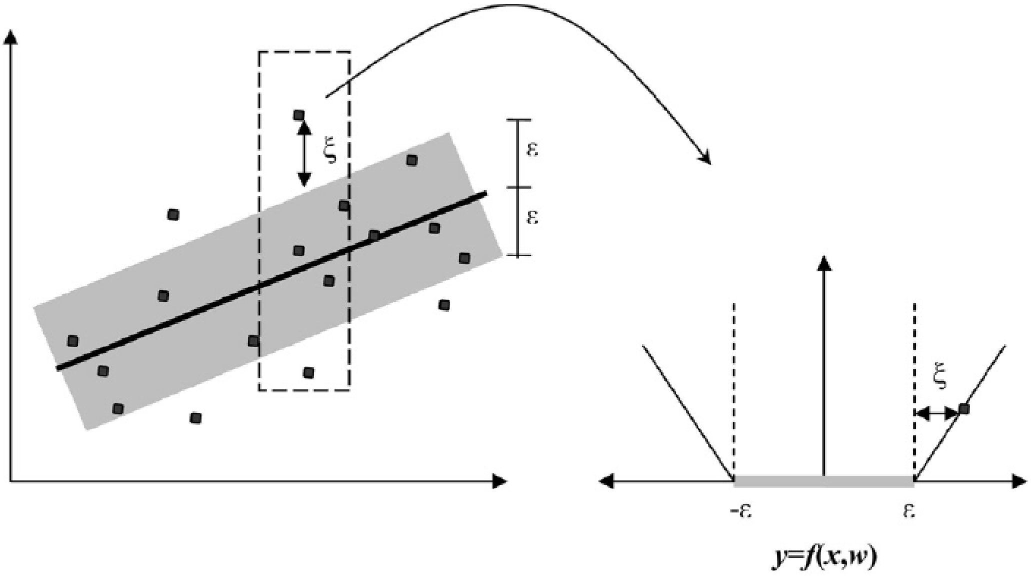


Figure 13 - $\varepsilon - sensitive\ error\ function$

Source: (TwarakaviJiri, Simunek, & Schaap, 2009)

For the successful minimization of $H(\beta, \beta_0)$, the solution for $\hat{\beta}$ can be written

$$\hat{\beta} = \sum_{i=1}^{N} (\hat{\alpha}_i^* - \hat{\alpha}_i) x_i \tag{5.4.3.4}$$

$$\hat{f}(x) = \sum_{i=1}^{N} (\hat{\alpha}_i^* - \hat{\alpha}_i) \langle x, x' \rangle + \beta_0 \tag{5.4.3.5}$$

As $\widehat{\alpha_\iota}, \widehat{\alpha_\iota}^*$ are positive, it is possible to solve the quadratic programming problem

$$\min_{\alpha_i^*,\alpha_i} \varepsilon \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) - y_i \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) + \frac{1}{2}\sum_{i,i\prime=1}^{N}(\alpha_i^* - \alpha_i)(\alpha_{i\prime}^* - \alpha_{i\prime})\,\langle x, x'\rangle \tag{5.4.3.6}$$

Subject to the constraints

$$\alpha_i \geq 0; \alpha_i^* \leq \frac{1}{\lambda}; \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0; \alpha_i\alpha_i^* = 0. \tag{5.4.3.7}$$

The mathematics of the Kernel trick for the support vector machines makes it possible to find solutions on the input values from the inner product $\langle x, x'\rangle$ when data is not-linearly separable. Hence, generalization of the approach is possible by defining an appropriate inner product. Regarding the kernel induced feature space for SVM regressions, the calculations are computational challenging, and excessive compared to the binary classification framework of our dissertation.

5.5 Random Forest and Trees

In this section, we will introduce our last machine learning model. The Random Forest algorithm was developed by (Breiman, Random Forests, 2001), as an extension to his Classification and Regression Trees (Breiman, Friedman, Stone, & Olshen, 1984). The Random Forest algorithm subsist of many other models, which makes it an ensemble method. However, the final prediction and quantities obtained is a combination of outputs from the underlying models.

We will start with the classification and regression trees (CART), before moving onto the Regression Trees.

5.5.1 Classification and Regression Trees

The elemental concept of CART is based on (Breiman, Friedman, Stone, & Olshen, 1984). They present regression trees as a method for binary classification with the use of multiple variable batches from the data-source. They aim at establishing a systematic approach for predicting the classification correctly across the different datasets. We start by structuring the data in a measurement vector;

$$X = \left(x_1, x_2, \dots, x_p\right) \qquad 5.5.1.1$$

Where each x_p represents a variable, in the task at hand, each x_p is a new accounting factor. We can then use the vector of observations,

$$y = (y_1, y_2, \dots, y_n)^T \qquad 5.5.1.2$$

To construct a matrix holding all input data and the corresponding classes. In the effort of systematic predicting the correct classification we can use the rule

$$x_j = \left(x_{1j}, \dots, x_{nj}\right)^T \, for \, j \in \{1, \dots, p\} \qquad 5.5.1.3$$

To assign one of the classes to x_j (Breiman, Friedman, Stone, & Olshen, 1984).

At each branch the objective of the algorithm is to do a binary split. Meaning that the split is a base-2 numeral system (in example 0 or 1, or Yes/No). CART operates by estimating the conditional distribution of the dependent variable based from the different partitioning branch. The resulting output is that each split only contains the base-2 numeral split.

Underneath is a simple example of a Classification and Regression Tree.

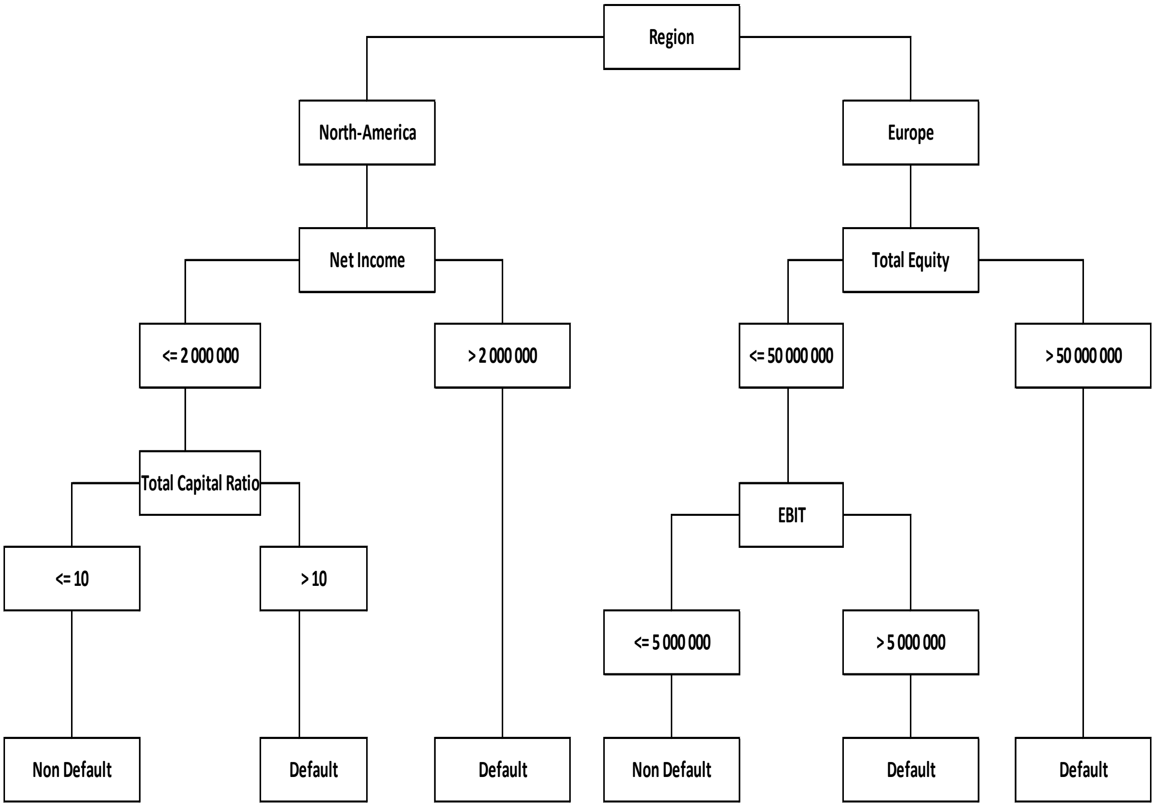


Figure 14 – Flowchart for Classification and regression trees

Source: Visualisation inspired by (Breiman, Friedman, Stone, & Olshen, 1984)

The CART logic can be extended so to arrive at a non-binary classification result for the dependent variables. The main conceptual difference for this approach is the calculation and application of the loss function used in the regression tree. First and foremost, each branch has different nodes, where each node can be expressed as (Breiman, Friedman, Stone, & Olshen, 1984).

$$y^m = (y_1^m, \dots, y_n^m); X^m = \left(x_1^m, \dots, x_p^m\right) \qquad 5.5.1.4$$

At each node split, x_s^m represents the specific split *s,* where C^m is the independent variable

$$C^m = \{x_i^m\}_{i \in \{1,\dots,n^m\}} \qquad 5.5.1.5$$

Each node contains a specific set of variables considered from the element of C^m. The different nodes hence constitute different independent subsets of C^m:

In each node, a new iteration creates a binary split resulting in two new nodes. The new nodes y^{ml} and y^{mr} are based on the previous explanatory variable c from the preceding mother node. The value of y^{ml} and y^{mr} is affected by the value of c corresponding to x_s^m. The proceeding nodes are therefore based on the decision of $\mathrm{x}_s^m \leq c$ at the specific node. After the split, the nodes receive the corresponding values of y^m, depending on the side of the iteration. The optimization for all iterations is obtained by the reduction in the error term for the whole regression as defined by (Breiman, Friedman, Stone, & Olshen, 1984)

$$\Delta(y^m) = L(y^m) - \left[\frac{n^{m_t}}{n^m} L(y^{m_t}) - \frac{n^{m_r}}{n^m} L(y^{m_r})\right] \quad 5.5.1.6$$

Based on the decision criteria $\mathrm{x}_s^m \leq c$, n^{ml} are the number of times the condition is true, whereas n^{mr} accounts for all non-true conditions at the iteration splits. , $\mathrm{L}(\cdot)$ is the loss function occurring from the foregoing node. The loss function is relevant due to the measurement of wrong classification in all nodes.

With classification, the categorical output of the dependent variable can be defined as a set of unique predetermined classes of y^m so that:

$$\mathfrak{D}^m = \{y_i^m\}_{i \in \{1,\dots,n^m\}} \quad 5.5.1.7$$

As mentioned, the loss function $\mathrm{L}(\cdot)$ measures the level of misclassification in each node. In order to calculate the total error value for the model, we need to find the number of times the dependent variable is grouped in the correct class membership $\mathrm{d} \in \mathfrak{D}^m$ and denote it as $\mathrm{p\ m(d)}$. The monotonous class is set as $(\hat{y}^m)$. The total error can then be finalized as following the calculation of (Breiman, Friedman, Stone, & Olshen, 1984):

$$L_{mc}(y^m) = \frac{1}{n^m} \sum_{i=1}^{n^m} \mathbb{1}(y_i^m \neq \hat{y}^m) = 1 - p^m \hat{y}^m \quad 5.5.1.8$$

Where the loss function $\mathrm{L}(\cdot)$ can only obtain a value of 1 if the condition is true, else 0.

The use of the loss function as mentioned is only applicable with categorical regression threes. When dealing with continues outcomes, a different approach is applied. In this event, the error term in each branch is often calculated as the mean squared error (MSE) as proposed by (Ghodselahi & Amirmadhi, 2011).

$$L_{mse}(y^m) = \sum_{i=1}^{n^m} \mathbb{I}(y_i^m - \hat{y}^m)^2 \qquad 5.5.1.9$$

Where $\hat{y}^m$ is the mean for all y_i^m.

After the selection of a suitable loss function, we can find the optimal combination, based on the quantity$\Delta(y^m)$. The calculation containing the highest Δ is chosen, due to its cumulative sum of all correct classifications (for all feasible splits combinations). The readjustments of the weightings are continued until the stopping criterion is met. This continuous iteration process is implemented to avoid overfitting, the node weightings during the model development and to limit excessive usage of complex modelling. It is further relevant for training the model to generalize its interpretations of the independent variables (Hastie, Tibshirani, & Friedman, 2013).

The importance of not overfitting the CART model is true as it may result in high variance of the fitted values making them very volatile to small changes in the training data. In the next section, we will discuss how the introduction of Random Forest deals with these problems.

5.5.2 Random Forest

The problem with high variance for fitted values is discussed in a paper from Breiman. In 1996 he proposed “Bagging”, short for “bootstrap aggregation”. Bagging, is used as a device to reduce the prediction error in CART. Breiman argues that the high variance and problems with overfitting is due to the fact that Classification and Regression Trees are highly unstable functions of the data (Buja & Stuetzle, 2006). Meaning that only small changes in the training sample can result in very different trees. And this is where bagging comes in to play. By doing multiple samples from the training, we can use an average of the models to create a more precise prediction.

Which means that we exclude some parts of the original dataset, known as “out-of-bag” data (Breiman, Bagging Predictors, 1996).

By building multiple samples of CART, we can construct the Random Forest by using the components sampled. To do so, the predicted variables for each observation is used to construct an ensemble estimate. The estimate has a lower variance than only one CART. By producing an average of all CARTs the variance is reduced. (Buja & Stuetzle, 2006).

Other attempts to improve the accuracy was done, and Breiman himself mentions random split selection (Dietterich, 1998) and his own introduction of random noise in the outputs (Breiman, 1998c) as better performing alternatives than Bagging. However, Adaboost (Freund and Schapire, 1996) is believed as the best performing random forest according to Breiman. (Breiman, 2001). Adaboost uses adaptive reweighting, instead of developing completely new trees for each simulation. This means that the weighing of any new model training uses and reweighs previous ensemble trees.

In 2001, Breiman extended his own logic of Random Forest inspired by Adaboost. To improve accuracy, he decided to apply a random selection of independent variable or joint combinations of variables at each node to build each tree. The intention is to minimize correlation from the dependent and independent variables. The random selection of explanatory variables is an effort to analyse a greater variety of variable sets, rather than obtaining a local optimum. With the old architecture, redundant but predictive variables would be left out in advance of higher predictive variables, neglecting the relationship of conditional variables as a whole can be highly explanatory. With the use of adaptive reweighting the trees produced is far more diverse and the predictions, as a consequence of this, yields a lower variance (Breiman, Random Forests, 2001).

Breiman uses bagging complementary to the random feature selection. The out-of-bag sample is used to obtain higher explanatory power, as well as an estimator for generalization error. Even though out-of-bag classifiers often overestimate the current error rate the calculation is an unbiased prediction when reaching adequately variable combinations (Breiman, Bagging Predictors, 1996).

When dealing with continues dependent variables, the simple form of random forest is adequate. The final weightings are achieved by the average value of all modelled trees (Breiman, Random Forests, 2001):

$$\hat{f}(X) = \frac{1}{T}\sum_{t=1}^{T} f^{t}(X_{i\in\hat{\mathrm{B}}^{t}}) \tag{5.5.2.1}$$

The parameter T in the formula is the overall sample of trees in the finalized model. Where $\bar{B}^t$ describes “bagging” for the $f^t(X_{i\in\hat{\mathrm{B}}^t})$- tree at a given time t. $\hat{f}(X)$ is thereby the average from all previous interations.

The overall objective is to minimize the generalization error. In the effort of finding this, the number of trees and variables at each node in the trees are the tuning parameters. The preferred solution depends on the optimization problem, as well as the data. A risk in the process is overfitting the Random Forest. The expected generalization error provided by bootstrap aggregating can be used to find the optimal number of trees, as the error decreases with more combinations.

An outspread resampling method in machine learning is cross-validation. For more details, see (Amaratunga, Cabrera, & Lee, 2008) or (Biau, Devroye, & Lugosi, 2008).

The following figure illustrates classifications with different amounts of *n* trees. This is done to illustrate that excessive usage of trees can result in overfitting. In addition, the figure present the predictive power of only one three (whereas the classification has no chance in a non-linear dataset).

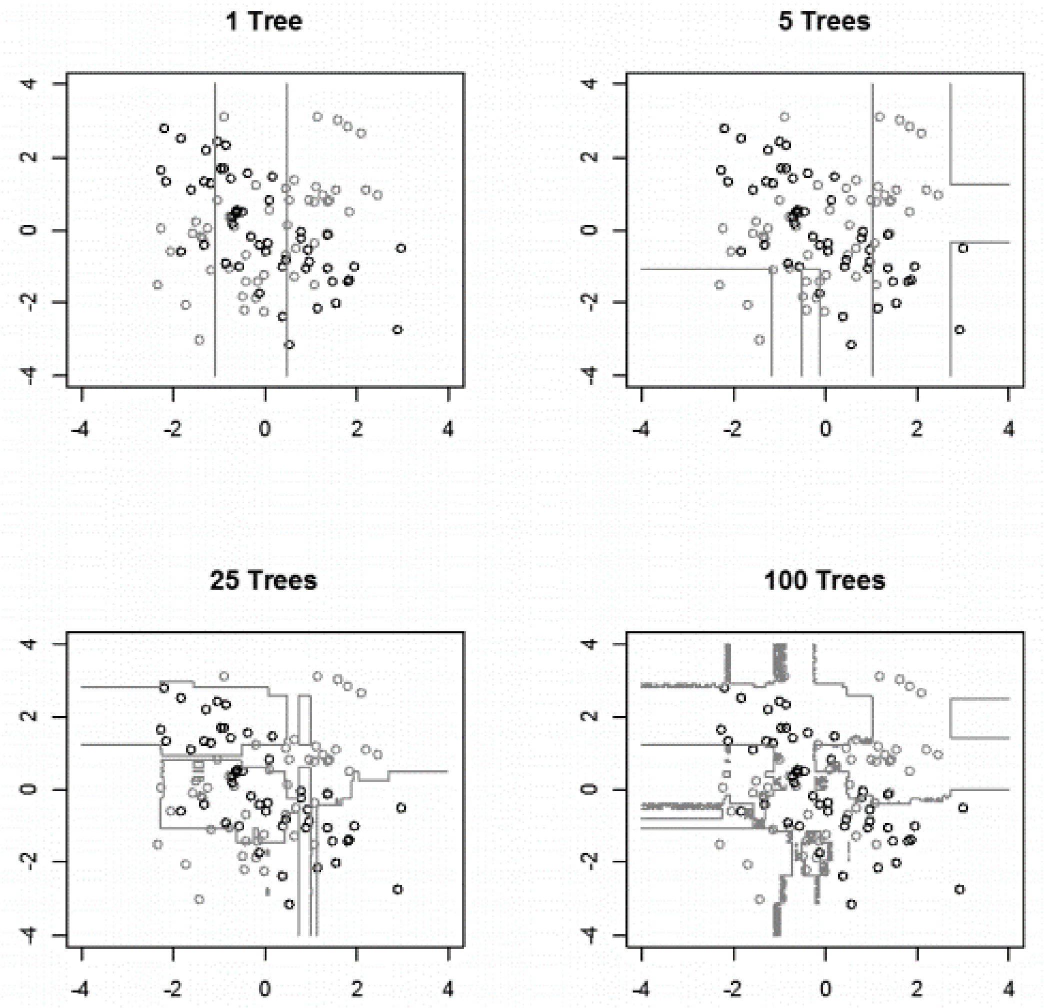


Figure 15 – Comparison of Random Forests

Source: Visualisation inspired by (Ozaki, www.tjo-en.hatenablogg.com, 2015)

6 Model Development

The objective of any model development is to initiate in-house credit assessment of counterparties with sufficient ranking ability. This is accordance with the requirements from Nordea and the legislative framework presented. The ultimate task is to differentiate and distinguish between good and bad counterparties. Hereby, estimating the probability of being a good or bad party. To do so, the algorithms aims at utilizing historical data in such a way that it becomes possible assessing relationships between historical data and future performance with high enough quality to limit false negatives and false positives classifications with satisfactory standards. All of the model developments' will be based on the internal dataset provided by Nordea. Where the expected probability of default is calculated to reflect on a one year expected default basis for the entity as a whole.

This chapter presents how we derived at the different components and their application so to arrive at the final models. This includes underlying theory of variable selection. In addition, theory for assessing the performance of the different machine learning models is presented. Furthermore, a thorough description of the dataset is given. At last, the process of building the aforementioned models is described to give a better view of the structure and computational task.

The following section will differentiate between model, variables and method. A *model* is defined as a simplification of reality, where a dependent variable is explained by one or more independent variables. A *variable set* is defined as a collection of key figures from annual reports that are used as the independent variables in a given model. *Methods* are different techniques for estimating models using a view and a given set of variables.

6.1 Variable Selection

The difficulty of selecting variables arises from the existence of a very large data set, making it a daunting task to select enough but also not too many variables. Other issues relate to the behavior of the variables.

Here especially is the redundancy affected by correlations between potential input variables. Another common issue is including variables with no or very little predictive power.

A bad selection of input variables can affect the model negatively and lowering the predicator power. This again can have huge negative consequences, as Nordea might take unwanted risk from counterparties. The prediction of the dependent variable relies on, for any statistical model, exploitation on relationships between the inputted data and the output. Thereby making the process of good variable selection crucial, to develop a good statistical model.

In view of the thesis, non-parametric methods, are models with no underlying assumption on any factors such as population, distribution or sample size of the independent variables. In comparison, parametric models are models with some physical interpretation of the underlying system. Here, the linear regression is one example. The main difference between the two methods, are the underlying assumptions regarding the structure of the model.

Artificial neural networks or other similar data driven modeling approaches falls under the category of non-parametric methods. For such models, the variables are selected from the available data, and the statistical model is thereafter developed. The complexity and non-parametric structure of Artificial Neural Networks makes the application of many existing variable selection methods inapplicable.

(May, Dandy, & Maier, 2011) present six key considerations for variable selection when using artificial neural networks. These are; relevancy, computational effort, training difficulty, dimensionality and lastly comprehensibility.

Relevance – The most common concern regarding variable selection is to include too few variables, or that the variables included are not sufficiently informative. Consequently, the performance of the model is poor, amid unexplained behavior of the output variable.

A priori – The "*a priori"* assumption evolves around the concept that at least one of the available variables should be capable of finding some, or all of the output behavior.

Hence, the strength of these relations is the unknown and what is disclosed by the models. In the case of low prediction power by the variables, the development is intractable. Resulting in the necessity of reconsider the data set and the choice of model output.

Computational effort – The computational effort can largely be affected by the number of included variables. The immediate consequence is the data cost of querying the network. Which to a large effect, decrease the training speed.

Further, if the model developed is multilayered, the input holds an increased number of incoming connection weights. With kernel-based regression (such as the SVM) and radial basis functions, the increased input will result in more prototype vector calculations due to the higher dimensionality. Overall, excessive usage of variables, place an increased burden on all data pre-processing steps during the model development.

Training difficulty – The training process for the ANN modelling becomes more complex when including variables that are redundant or with low explanatory power. Training sets with redundant variables increases the combination of different parameters that will results in the same locally optimal error term. (In the error function over the parameter space of the model). This is problematic, as the algorithm applies resources adjusting the weights that yield no improved bearing on the output variable. In addition, redundant variables can bring noise, so to mask the relationships of the input-output.

Dimensionality – The challenge of dimensionality is the relation between dimensions and domains. As the dimensionality of a model increases linearly, the total volume of the modelling problem domain increases exponentially (Bellman, 1961).

To solve the challenge of mapping a given function over the parameter space, with satisfactory confidence – the sample size must increase exponentially (Scott, 1992).

Comprehensibility – For most machine learning and neural network modelling, including too many variables can reduce the comprehensibility.

ANN can be seen as a “black box”, where modelers are keen and increasingly concerned with the knowledge discovery during the process of model development. Here, especially to check if the behavior of the modeled input-output response make sense.

The ending goal of any variable selection should therefore obtain a model with the fewest input variables required to describe the behavior of the output variable. Further, the variables selected should hold a minimum degree of redundancy and with no uninformative variables. Successful selection of such, will lead to a cost-effective, more accurate ANN model and make the results and model development more interpretable.

For all the reasons mentioned above, there should occur a process of variable selection or filtering process before the model development can commence. There exist several methods for such filtering, and their application can be used not only for the models presented in this thesis. The variable selection processes used for this thesis are as included in (May, Dandy, & Maier, 2011). For the variable selection there exists different applicable perspective of how the inputted variables are analyzed.

6.1.1 Model based approach

The model-based approach divide itself into two main subtypes for the variable selection. Namely wrapper and embedded algorithms.

Wrapper algorithms is a model-based approach for the input variable selection. The wrapper is an integrated part of the model architecture, where all possible combinations of available variables are tested, so to find the combination that yields the optimal generalization performance of the trained ANN.

In other words, the wrapper approach treats the variable selection as a model selection task, where each model is a unique combination of different variables. The process is illustrated in the following figure (May, Dandy, & Maier, 2011):

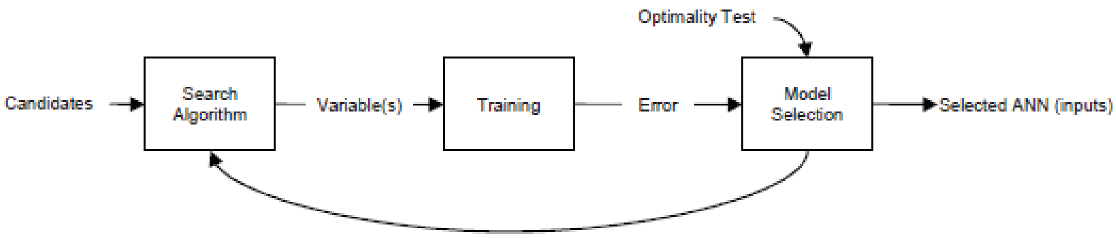


Figure 16 – Wrapper Structure

Embedded algorithms, as the name implies, are directly embodied into the Artificial Neural Network algorithm. The model adjusts the weights of the inputted data to measure the impact of each candidate on the performance of the model. Further, during the training process, redundant and non-explanatory variables are less and less weighted until removed. The process is illustrated in the figure below.

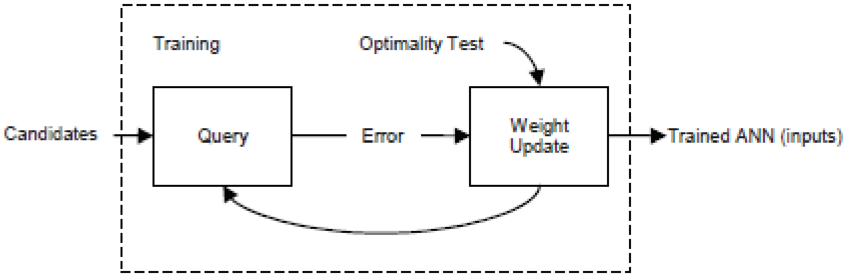


Figure 17 – Embedded algorithm structure

6.1.2 Model-free approach

Filter algorithms are model-free, meaning that the filters operate as a preliminary process externally from the Artificial Neural Network training. The filers adopt an auxiliary statistical analysis technique when looking at the validity of the variables individually or different combinations of the different candidates. The process is illustrated in the figure below.

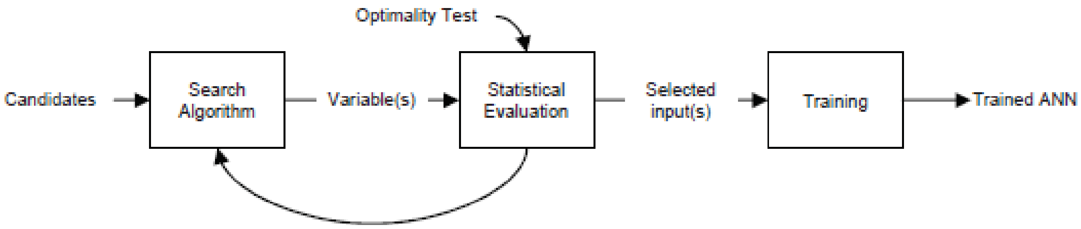


Figure 18 – Model free structure

The analysis of the candidates for the filter algorithm identify preferable candidates by applying the following criteria:

Maximum relevancy (MR), is a criterion for finding variables that are highly informative. This is archived by filtering candidates that have a high degree of correlation with the outputted data. The procedure follows the structure of finding the determined relevancy for each input variable independently, with the output variable. One example of such processing is input ranking schemes. To the maximum relevancy, greedy selection can be applied. The greedy selection puts a limit or threshold to the maximum allowed candidate inclusion.

Minimum redundancy (mR), is an additional criterion to deal with the down side of including the greedy selection. By applying a limit, the candidate variables do not strictly gain an optimal Artificial Neural Network. Hence, the minimum redundancy search for variables to find candidates that are highly dissimilar from each other. This in order to find combinations with minimum redundancy and select sets with maximum containment of relevant variables.

Minimum redundancy-maximum Relevancy (mRMR), the combination of the two criterions lead to the mrMR selection criteria. Here, the inputted variables are evaluated according to both the relevance and dissimilarity compared to the other variables.

6.1.3 Search Strategies

As different Artificial Neural Network models are tested throughout the thesis, the models should be based on the same input of candidates. As both wrappers and the embedded algorithms operates in different ways for each unique mode, the overall selection of inputted variables is based on the model-free approach.

Here by validating the variables externally before inclusion. The variable selection methods applied therefor accounts for this externality decision when comparing the models. Incremental search strategies tend to dominate filter designs, hence the different methods used are forward selection, backward elimination and step-wise regression.

Forward selection is an incremental linear selection strategy, where the individual variables are selected one at a time. The process is continued until adding one extra variable gain no improvement of performance. For filter designs, the process starts by including the most significant variable first. The search strategy then continues by iteratively locating the next most relevant candidate and evaluating if the candidate should be included or not. This process is continued until the optimal criteria is satisfied.

Overall, the process is efficient and with reduced computational costs. Further, the ending result often includes a relatively small set of input variables when the optimal requirement is satisfied. The most significant downside of this method, it that the search strategy does not test for all observable combinations. Consequently, the risk of finding a local optimum, and then kill the search exist. Lastly, as forward selection is an incremental search algorithm, the search may ignore variable combinations that are highly information combined, but do not yield any improvement, when looked at individually.

The backward elimination strategy operates in a similar manner as the forwards selection. Essentially, selecting the potential candidates in reversed order. This means that process starts by including all candidates, and then eliminate one-by-one.

With filter strategies, the least explanatory candidates are iteratively removed up until the optimality threshold is satisfied. Compared to the forward selection, the backward elimination operates with higher computational costs. Especially for many large models, where the data set constitute a large amount of candidates. Also, when starting off with all variables, it can be harder to differentiate the significant importance of the different variables.

Forward selection is said to have fidelity, in that once an input variable is selected, the selection cannot be undone. Step-wise selection is an extension of the forward selection approach, where input variables may also be removed at any subsequent iteration. The formulation of the step-wise approach is aimed at handling redundancy between candidate variables.

Step-wise selection can be viewed as an optimization of the forward selection. Here in the perspective of fidelity. With the previous method once a variable is selected, the selection cannot be undone. However, when employing step-wise search strategies, variables can be removed at each iteration. In other words, variable x1 was selected due to its explanatory power. At a later iteration, the combination of the two new variables x2 and x3 outperform the relevancy of x1. The x1 variable is therefore redundant and will be removed in favor of the combination of the two new variables.

6.2 Model Selection

If the size of the dataset is low, one can use k-fold cross-validation or leave-one-out validation. K-fold cross validation is about splitting the dataset into subjunctive subsets of equal size.

Each subset is used as a test set and the remaining k-1 subsets are joint to be used as training sets. The accuracy rate is the calculated for each test set, and the average of these k scores will be used to measure the accuracy. Popular values for k are 5 and 10. These value for k are statistically likely to provide an estimate that is accurate.

It is important to note that the calculation time increases with the number of subsets created. In other words, 5 subsets will result in 5 times longer prediction time. When using leave-one-out validation, only one of the data items is used for testing and the rest is used as a training set. If the data set is of size m, the process will correspond to m-fold cross validation and this procedure is only used with small datasets’.

6.3 Performance Assessment

In this section, we will focus on a suitable definition for default vs non-default, and subsequently performance indices based on distribution and density functions.

The first step, and potentially the most important is the definition of the dependent variable. Working with credit scoring it is important to define the cases of default vs non-default. We will apply the commonly used binary solution with one as default, and zero as non-defaults. Additionally, it is important to specify the time range. We look at the entities separately on a one-year horizon.

The reason behind a short time-range is due to Basel requirements for the internal rating-based method. Here the probability of default is defined as the likelihood of a default within a one-year period.

After defining the default and non-default entities for building of the model, we will apply multiple statistical measurements. This is done in order to evaluate the overall performance and predictability. We have decided to use assessment factors such as accuracy, sensitivity, specificity, cumulative lift and the AUC coefficient.

We use the following labels classifying the binary output:

$$Company = \begin{cases} 1 = Default \\ 0 = Non - Default \end{cases} \tag{6.3.1}$$

6.3.1 Accuracy

Hit percentage is a percentage of total events that the model predicts correctly. The higher the percentage, the better the model. This way to evaluate models is easy to understand and require no special knowledge of statistics. On the other hand, the percentage of hits in some Cases are a less good evaluation goal, as it does not take into account the class distribution in data selection or error costs.

6.3.2 Receiver Operating Characteristic

ROC is an abbreviation for receiver operating characteristics, and is a technique for visualizing, organizing and categorizing events with only two possible outcomes (Fawcett, 2006).

All models in this task classify events to one of two values; "Default" or "Non default", where the former is considered positive and the latter as negative. Each event the model predicts can be placed in one of the four outcomes as shown:

Table 3 – Four outcomes for classification

	True Company-Positive	**True Company-Negative**
Positive Model	True Positive	False Positive
Negative Model	False Negative	True Negative

As a further explanation of the properties we have attached the underneath matrix.

Table 4 – Outcomes of classification and types of error

Positive class:	Non default
Negative class:	Default

	True Positive	False Positive	False negative	True Negative
True Company	Non Default	Default	Non Default	Default
Model prediction	Non Default	Non-default	Default	Default
Result	Correct decision	Type 1 error	Type 2 error	Correct decision

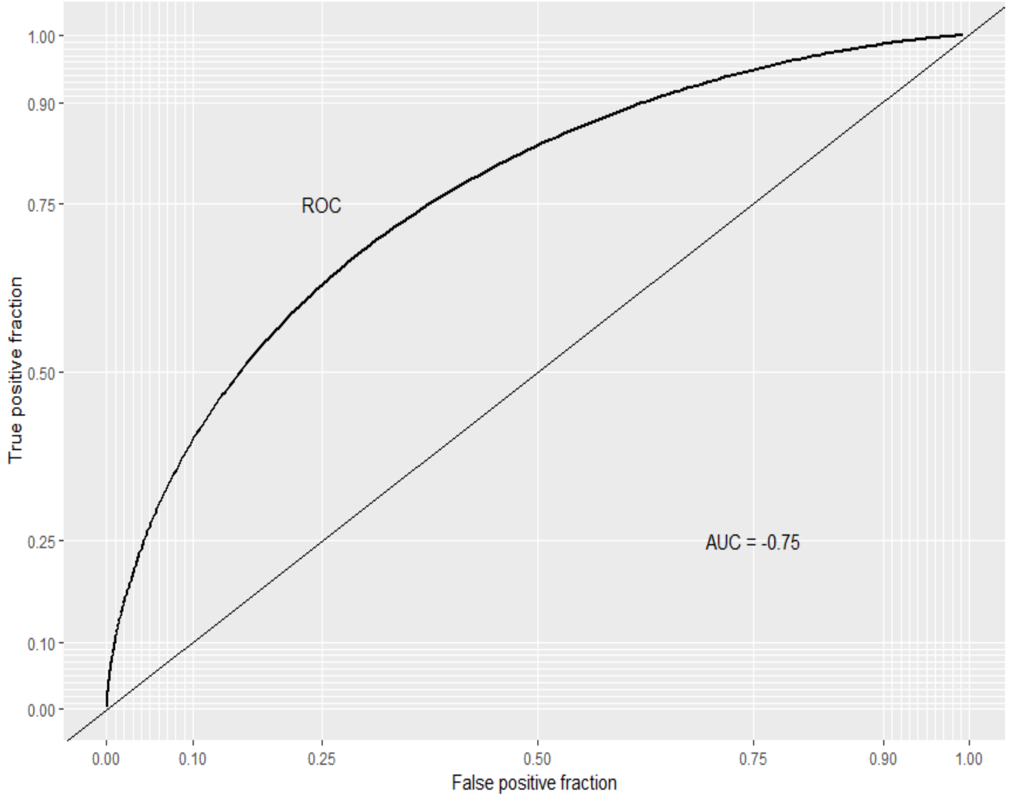


Figure 19 – Receiver Operating Characteristic

Source: Produced in RStudio

The curve for ROC (Receiver Operating Characteristic) can be described as:

$$y = F_{n,good}(a), \qquad 6.3.3.1$$

$$x = F_{m,bad}(a), \qquad a \in [\mathrm{L}, \mathrm{H}] \qquad 6.3.3.2$$

In order to evaluate the performance, we apply some common measurements for statistical binary classification problems, such as Sensitivity and Specificity. Where the Sensitivity represents the prediction of classification for True Positives, or in our case, the non-defaults. While Specificity validates the models ability to correctly distinguish the True Negatives/Defaults.

$$Sensitivity = \frac{True\ Positives}{True\ Positives + \ False\ Negatives} \qquad 6.3.3.3$$

To further explain, this means that a perfect model, in terms of classifying all non-defaults correctly obtain a value of 1 due to all observations being True Positives. In other words, the wrongly classification of non-defaults will increase the proportion of false negatives, thereby reducing the Sensitivity. In example a prediction of 0.7 is due to 0.3 of the non-defaults being classified as false negatives. To recognize defaults, we use the specificity measurement.

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives} \qquad 6.3.3.4$$

The specificity is similar to the sensitivity. However, it measures the accuracy of the default observations, instead of the non-defaults. The sensitivity and specificity measurements are a good way to capture the characteristics of the classification. As an example, a high sensitivity but a low specificity indicates that the classifications cannot correctly identify the defaults in a satisfying manner.

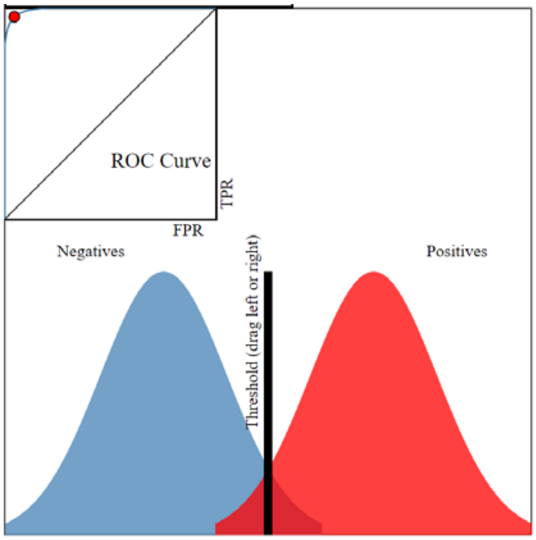
At last, we can use the Likelihood to measure the probability of how many times a non-default correct classification will appear rather than a default being misclassified as a non-default.

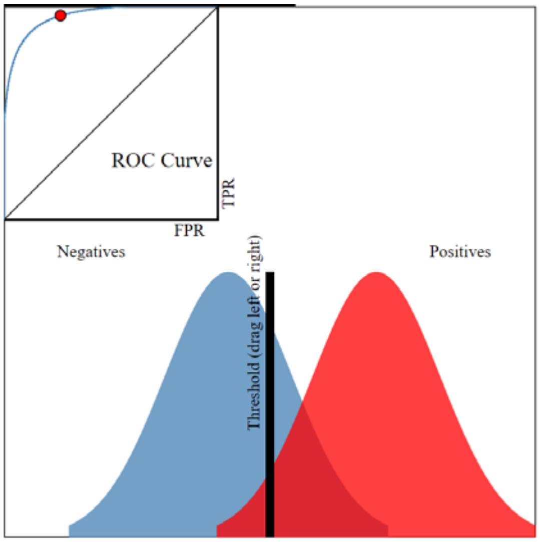
$$Likelihood\ Ratio = \frac{Sensitivity}{1 - Specificity} \qquad 6.3.3.5$$

A great strength with the ROC graph and values is the fact that it visualizes and categorise without taking class classification and error cost (Fawcett, 2006). This feature is especially important when dealing with datasets that have a skew distribution and thus are not normally distributed. It is also important when working with cost-sensitive learning as in this case. It is therefore preferable to use ROC as a measure for how well a model predicts.

6.3.3 Area Under the Curve

As mentioned, an additional measurement for the predictive accomplishment is the AUC (Area Under ROC Curve). The outcome of the model lays in the name, as it represents the value underneath the ROC curve, describing a global performance for the classifier (Lobo, Raimundo, & Jimenez-Valverde, 2008). As the ROC curve measures a binary classifier, the output value for AUC will be between 0 and 1. Where 1 indicates an optimal score. We can see from the four graphs below that a model capable of distinguishing the two class populations will yield a higher AUC. In the bottom right ROC curve, the model has a hard time differentiating the two populations. The AUC is 50%, which can be the equivalent of a random selection or coin flip.





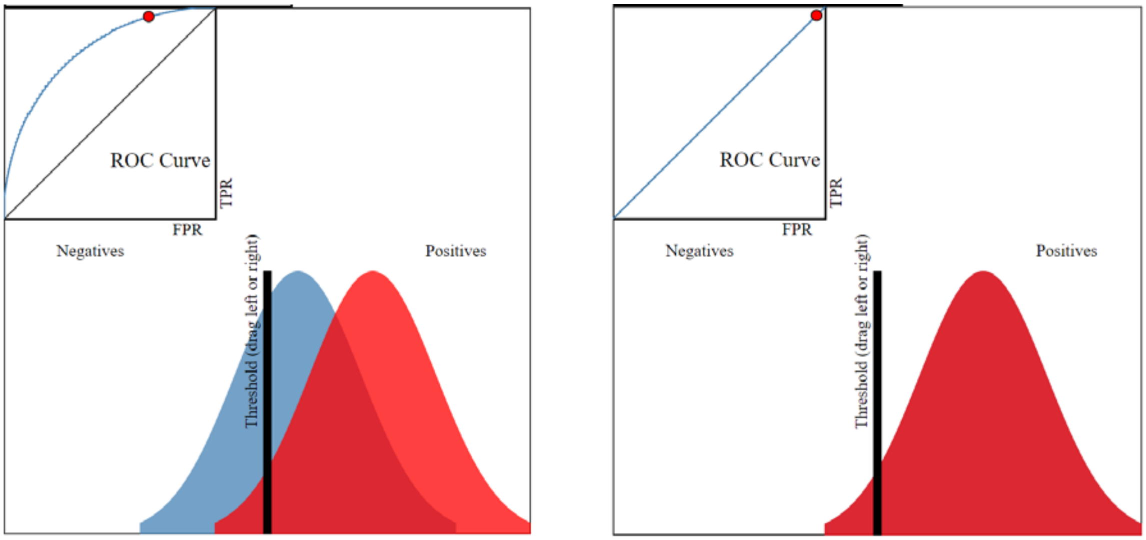


Figure 20 – Different ROC curves

Source: (navan.name, 2018)

6.3.4 Cumulative Lift

Lastly, the Cumulative Lift supplements the other statistical methods with its indication of explanatory power. This is a rather local, not global, performance measure. Which measures the level of acceptance (or rejection) a scoring model is better than a random model.

$$Lift(a) = \frac{BadRate(a)}{BadRate} = \frac{F_{m,bad}(a)}{F_{n+m,all}(a)}, \qquad a \in [\mathrm{L}, \mathrm{H}] \qquad 6.3.5.1$$

The cumulative lift describes the ratio between the predicted results and results using no model. The greater the area between the lift curve and the baseline, the better the model. The Cumulative Lift is nicely demonstrated on the subsequent figure.

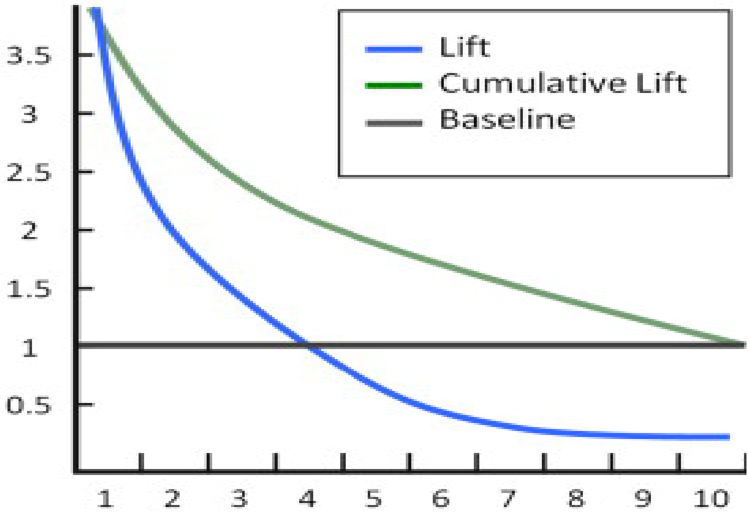


Figure 21 – Cumulative Lift

Source: (Paduaa, Schulzeb, Matkovićb, & Delrieux, 2014)

6.3.5 McNemars test

In (Dietterich, 1997) five different statistical methods for measuring machine learning performance is presented. McNemars test (McNemar, 1947) and (Gillick & Cox, 1989) is viewed as the most precise measurements of the five methods.

Let's assume that the rate for the number of error classifications by a method used is given by:

$$e_1 = \frac{M_1}{n} \qquad 6.3.6.1$$

where n is the number of observations in the dataset, and $M_1$is the number of wrong classifications.

The rate of misclassifications for a comparable method, based on the same sample data is given by:

$$e_2 = \frac{M_2}{n} \qquad 6.3.6.2$$

We then test for the hypotheses that $e_1 = e_2$, furthermore to categorise the test we can apply a similar categorisation as the ROC.

		Method 2	
		Correct	**Wrong**
Method 1	**Correct**	A	B
	Wrong	C	D

McNemars test in its simplified form can be written as this (Gillick & Cox, 1989).

$$W = \frac{|B - C| - 1}{\sqrt{B + C}} \tag{6.3.6.3}$$

The reason behind B and C over A and D, is that B and C are the observations where the two models predict different results. Based on this, the formula can be described as;

$$p = 2p(Z \geq w), where\ Z \sim N(0,1) \tag{6.3.6.4}$$

And w is the actual value for W. Furthermore, a p-value lower than 5% would indicate a significant difference between the models chosen for the task.

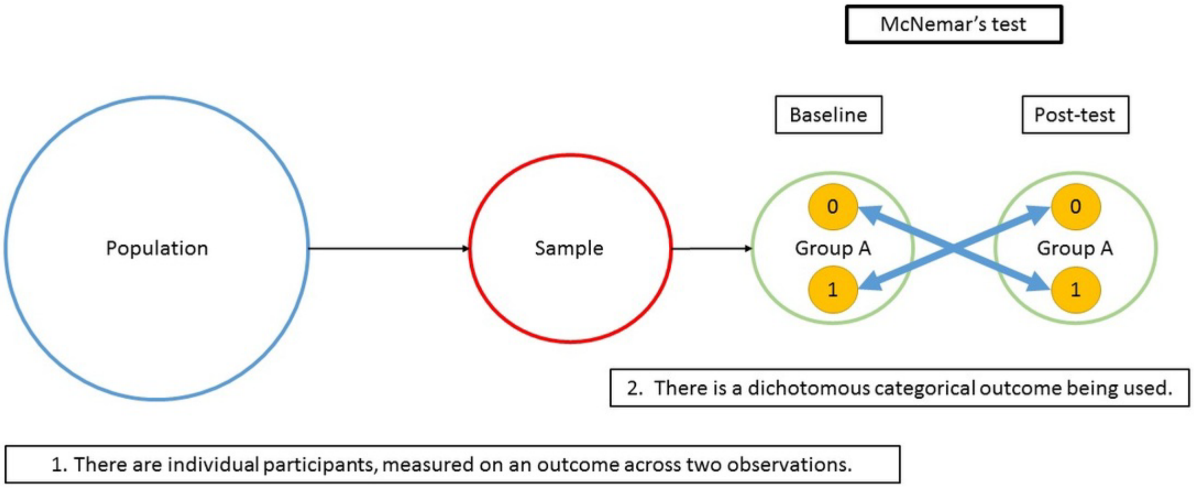

McNemar’s test
Baseline
Post-test
Population
Sample
Group A
0
1
Group A
0
1
2. There is a dichotomous categorical outcome being used.
1. There are individual participants, measured on an outcome across two observations.

Figure 22 – McNemar’s test

Source: (Scalelive, 2018)

6.4 Data Description

Explanatory variables are collected from different sources. But all data is extracted from the same data provider, namely SNL. SNL is the S&P global database for market intelligence. Data collection of the non-defaulted entities was performed by rolling all European commercial banks operating in 2014 and back to 2008. All entities were treated as independent. As for defaulted entities, all defaulted European banks with accounting statements available in SNL were collected. No distinguish was made between bailed out and not bailed out entities. Most observations consist of German Banks.

The dataset as mention is provided by Nordea. The data constitutes a variety of different entities that are relevant for the prediction of counterparty risks. All unique observations hold the same number of variables. To differentiate companies risk profile, all observations include a short-term credit rating by Fitch.

The variables general categorization arises from the respected company's financial statement. Not all of the variables are reported "as is", in the different companies' income statement and balance sheet. This is due to that; Nordea has provided different financial ratios that they find relevant for the prediction of default.

All data is as mentioned collected over the time period 2008 to 2014. This is done to ensure data quantity as well as to cover the business cycle. In total, there are 5,183 unique observations, with 75 independent variables. The lowest grade a company holds in the data set is "D" (which indicate that the company has defaulted or are under liquidation.) The highest rating in the dataset is AAA, indicating a company with the lowest risk possible. The following table illustrates 10 of the first variables in the dataset:

Table 5 – Some variables in the dataset

Example of variables (SNL Dataset provided by Nordea)			
Variable nr.	Variable name	Variable nr.	Variable name
1	Total Assets	6	Net Interest Margin
2	Total Equity	7	Return on Equity
3	Net Income	8	Total Capital Ratio
4	Net Income Growth	9	Net Loans to Assets
5	EBIT to Assets	10	Loans to Deposits

All of the different variable types can be broadly categorized into 7 core groups. These are:

Table 6 – Category of variables

Category of variables	
1	Capitalization
2	Financial flexibility
3	Liquidity
4	Market position
5	Profitability
6	Revenue Mix and Diversity
7	Asset Risk

In total, there are 45 companies that has defaulted. This is 0.95% of the total population. The following table shows the different distributions of the ratings, where the number-value indicate the number of ratings in the different rating grades. (Only ratings with observations higher than zero is presented in the table).

Table 7 – Rating Distribution

Rating Distribution (SNL Dataset provided by Nordea)	
Ratings	Observations
AAA	42
AA+	28
AA	34
AA-	2,832
A+	1,387
A	64
A-	97
BBB+	80
BBB	75
BB+	114
BB	61
BB-	58
B+	65
B	63
B-	74
CCC	6
D	45

7 Model Building

This chapter presents the whole process of model development, here by describing how relevant variables are selected. How the training of the models is conducted and validated. The process starts with the full dataset provided by Nordea and ends up with how the results from the different models are collected.

All calculations, including model development and performance assessments has been done in R-studio.

7.1 Data Preparation

Before any of the tests were started, the dataset was analysed and cleaned for low quality variables and observations. This was done in a two-step process. First, dependent variables missing 5% or more of their values were removed. Thereafter, entities missing any of their independent variables were removed. This was done to insure high quality data for modelling.

The cleaning provides a dataset with no missing observations for any of the variables. This reduced the amount of observation from 5,183 to 3,904. The total number of defaults after the reduction accounts for 1.3% of the entities.

The dependent variables for all observations are the credit rating assigned. All ratings were converted as either 1 or 0. Where 1 indicate a company that has defaulted and 0 for non-defaults. All dependent variables with a rating of higher than D were set as 0, and companies with a rating of D were set as 1.

7.2 Finding relevant variables

When the data cleaning process were completed, the task was to find the relevant variables to be included. For the variable selection process, all three searching methods, as presented in the section 6.1, has been applied. The different methods resulted in three different sets of variables.

Only the best performing set, based on the R^2-value were used for the different algorithms throughout this thesis. The selected variables that yielded the highest result is the step-wise selection process.

The graph below shows the different junctions where different variables are included. The horizontal axis represents all the variables available. On the vertical axis is the R^2-value. Each step on the vertical axis contains a different variable combination with a higher R^2-value than the one underneath. In example, in the first step there are a combination of 3 variables with an R^2-value of 0.0072. In the next step, two more variables are added which obtain a R^2-value of 0.0085. As this is a step-wise regression, each advancement improves the R^2-value, but there can exists different variable combinations than in the previous step. Variables can be added, removed or kept at each advancement. (This is different compared to the backward and forward regression, where an added variable is kept after inclusion).

As mentioned, the vertical axis is the R^2 factor. (Ranging from a minimum of 0 and a maximum of 1. Where 1 indicates the highest possible explanatory power). The process is stopped when there is no improvement of adding an extra variable in terms of the R^2 value. The process stops with R^2 of 0.099. At this point the total variables included are 20.

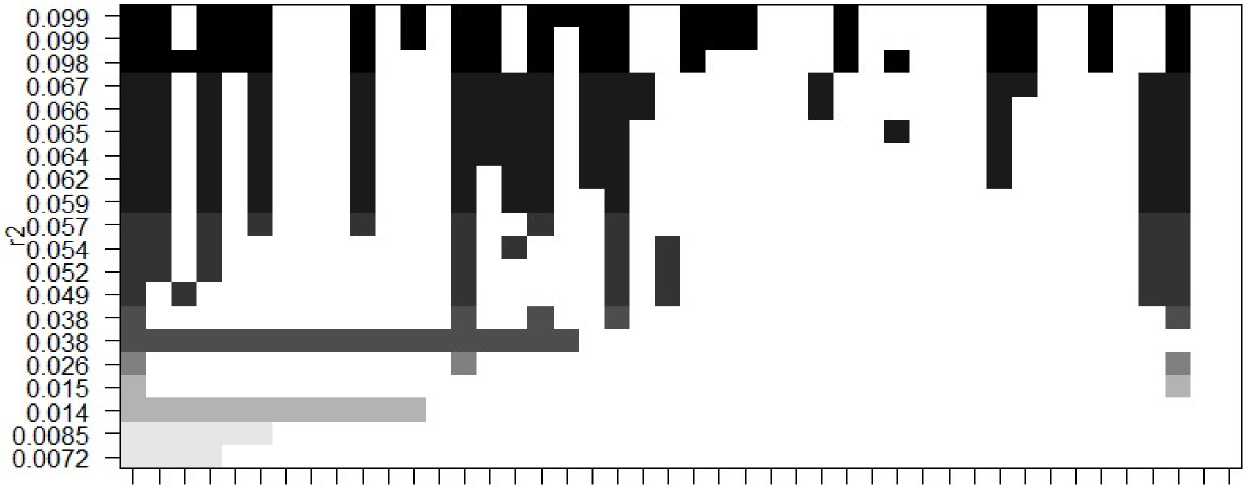
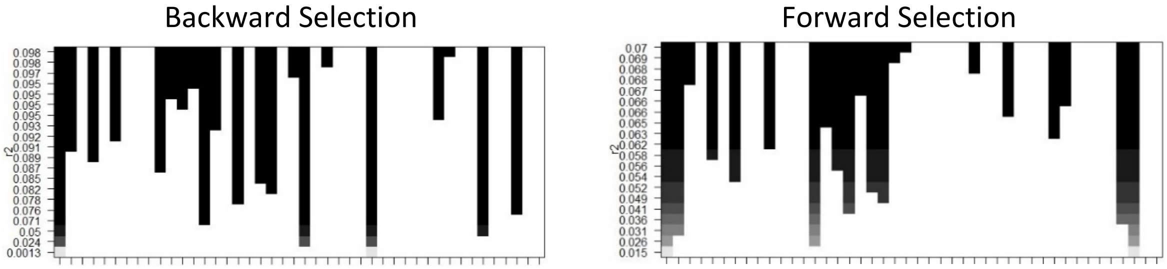


Figure 23 – Step-wise Selection (Output from R-studio)

As a short illustration, the two graphs below present the output from the two other methods. Backward selection has a R^2 value of 0.098 and Forward Selection has a R^2 value of 0.07.



Backward Selection

Forward Selection

Figure 24 – Backward selection & Forward Selection (Output from R-Studio)

In the following table, the 20 selected variables from the step-wise strategy are presented:

Table 8 – Selected Variables

Variables selected after search strategy	
Variable name	Description
Total Assets	Total Assets
Net Int Income To Op Income	Net Interest Income to Operating Income
Net Comm Income to Op Income	Net fee and commission Income / Operating Income (%)
Trd Income To Op Income	Trading Income to Operational income
Cost to Income	Cost to Income
Net Income At To Avg Assets	Net Income Attribution to Average Assets
Net Int Income to Avg Assets	Net Interest Income to Average Assets
Net Income to RWA	Net Income to Risk Weighted Assets
Net Income BTAX To Assets	Net Income Before tax to Assets
Net Income	Net Income
Net Income Growth	Net Income Growth
EBIT To Assets	Earnings before Interest and Taxes to Assets
Tang Equity to Tang Assets	Tangible Equity to Tangible Assets
Tier1 Ratio	Tier1 Ratio
Total Capital Ratio	Total Capital Ratio
Net Loans To Assets	Net Loans to Total Assets
Total Debt to Total Assets	Total Debt to Total Assets
Cash Equiv to Assets	Cash and cash equivalents to Assets
Tang C equity To Assets	Tangible Common Equity to Assets
Government Support Poc	Government support for privately owned companies

To summarize, all three searching algorithms found 20 unique independent variables. Based on their selection, three datasets were created from the cleaned date. Each set represented on of the respected searching strategies. Only the dataset providing the highest result from the statistical evaluation is used as selected inputs for the machine learning algorithms. The selected inputs come from the step-wise selection.

7.3 Training and validation

After the selection of input variables, the dataset needs to be split into different segments. This in order to train the machine learning algorithms and test their performance. The first segment is the training set, used for training the models. The second segment is the validation set. This set is simply used to test the accuracy of the weights found by only using the training set. A common practice is to use 80 percent of the observations for training and the remaining 20 percent for validation. A random selection process was used to allocate the data.

Cross-Validation, sometimes called rotating estimating, use several training and validation datasets. We have decided to use a multiple cross-validation, with 10 as a subset. This means that the process is looped 10 times. Each loop using a different unique partition of the dataset. The ending result for each loop is stored in a data table, and the average value for all validations are used to present the final result. This approach follows the same method as in (Kumar & Ravi, 2007).

7.3.1 Model training

All chosen machines learning algorithms inculcated the training data. This was also done for the logistic regressions. As there are different subtypes of the models, we have decided to use the multi-layer perceptron where jump connections are allowed. This is due to the complexity of the problem, which the MLP has proven to outperform other neural networks. For the same reasons the Support Vector Machine will be used with induced kernel feature space. It is unclear how to find the optimal mapping into different space, hence all different kernels must be tested.

When it comes to decision trees, we have decided to employ the advanced option Random Forest. The other machine learning models are viewed as far more complex structures. Making it compelling to see its performance measured towards the others. The chosen number of trees is based on the error value returned from the algorithm. As shown in the graph below the error slowly declines, and flat out at 80. Which is the number of trees chosen for the training.

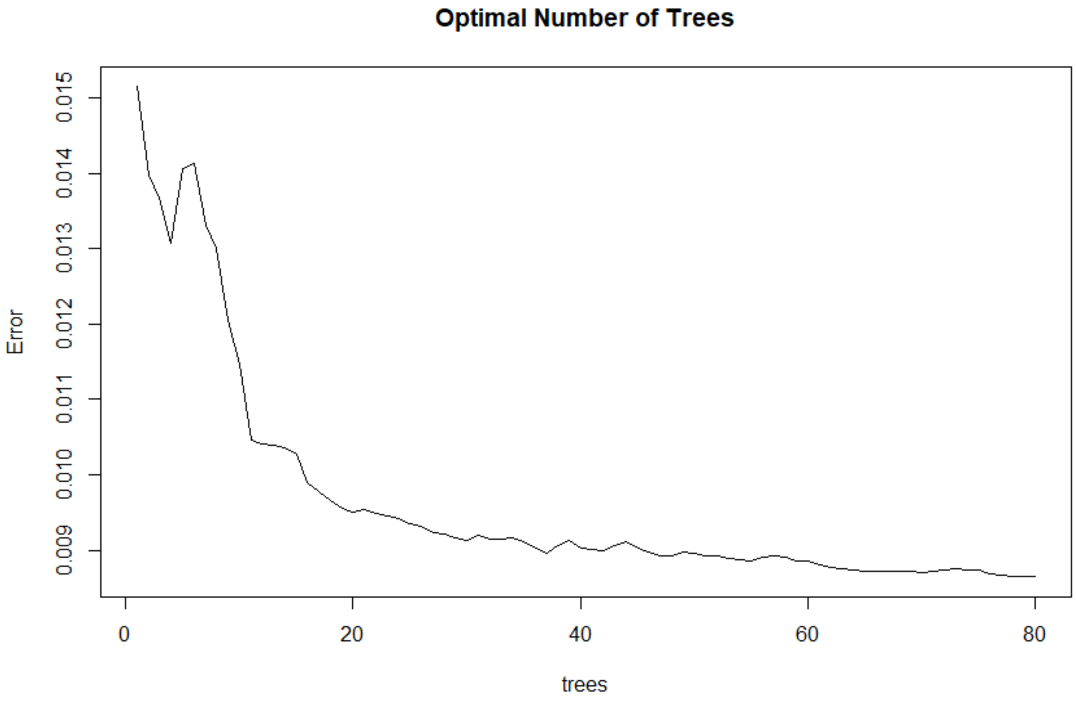


Figure 25 – Finding the optimal number of Trees

Source: Output from RStudio

Lastly, the logit model will be used as benchmark. This is due to the fact that it is a widely used statistical model in the research papers we have read. Its closed form and binary behaviour have made it an attractive alternative method to compare.

There are different parameters and tuning factors that can be used for each algorithm. In academic studies it is not clear when to apply the different adjustments.

The most common approach is trial and errors. After considerable testing, the following parameters gave the highest performance.

Their settings are the following:

- Neural Network: Multi-layer perceptron with 2 hidden layers and logit function as activation function
- Support Vector Machines: Radial kernel for induced feature space. Cost parameter of 10 and gamma value 0.5
- Random Forest: Depth of 5 with 80 trees

7.3.2 Model Validation

When training is completed, all models' weights and parameters are set. Below is the output of the Feed-Forward Neural Network training:

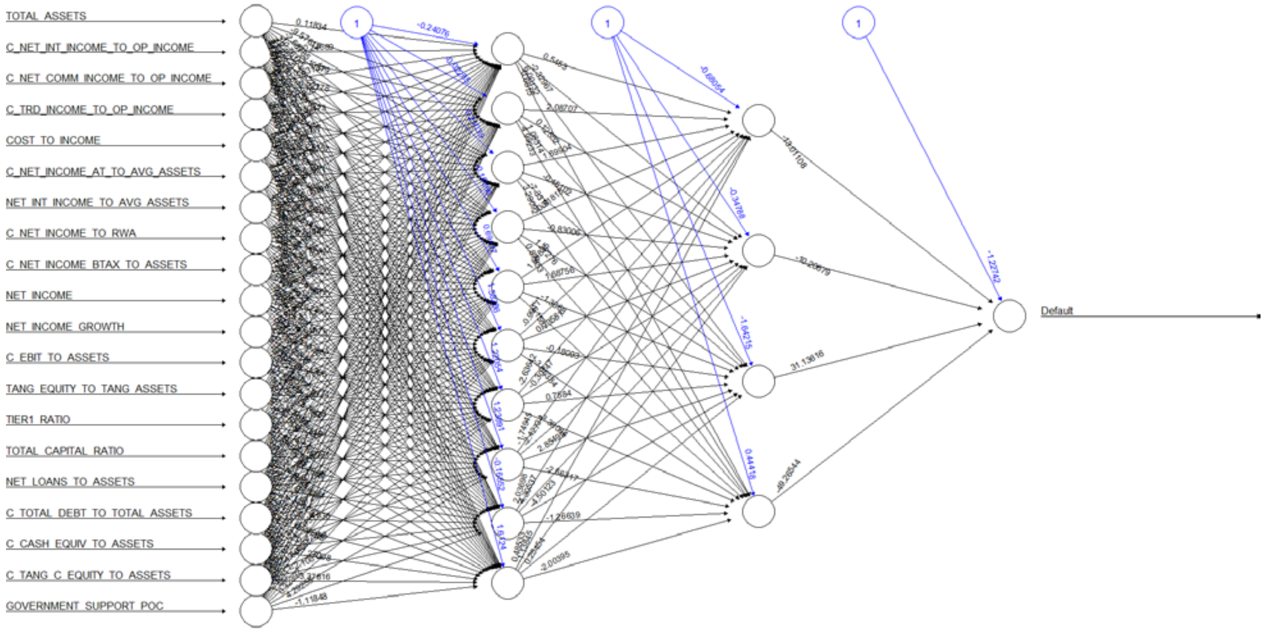


Figure 26 – Output of the Feed-Forward neural Network Training

Source: Output from RStudio

To validate the models, the validation set is used. Simply by infusing the observations into the trained models. The models will then give a value for the probability that one entity belongs to one classification or the other.

Due to the binary classification, these values are rounded given a threshold, so that the output is either 1 or 0. (1 indicating that the company is defaulted).

The point of running the validation set on the trained models is to validate the performance. The aim is to make a forecasting model that can successfully predict a default given some modelled input. Therefor the trained data is tested on “unseen data” where the correct classification is known.

As the classifications are known, it is possible to perform different analysis to validate the overall quality for different algorithms. If the models are approved, the weights are based on the average value of all looped model developments.

The results in the following chapter is based on the difference between the predicted and known classifications in the validation set.

8 Results and analysis

In this section, the results from the model building is presented and the model performance is evaluated. All machine learning algorithms are measured against each other, but more importantly, against the more conventional method- logistic regression. This is done by comparing the different model's performance measurements.

In the table below, the relevant measurements are presented.

Table 9 – Result presentation

	Logit	RF	MLP	SVM
Accuracy	0.99113	0.92522	0.84664	0.98606
Sensitivity	0.99872	0.98649	0.98958	0.98855
Specificity	0.40000	0.32450	0.07564	0.33333
Likelihood	1.66450	1.46040	1.07060	1.48280
AUC	0.96829	0.92432	0.57683	0.96270

By first looking at the accuracy values, there is no huge difference in model performance. The logit is the best performing model, while MLP is the worst performing. However, the accuracy does not differentiate between true-positive and true-negative. It is therefore important to measure these differences. The sensitivity measures the accuracy of true positive, while the specificity measures the true negative classifications of the validation set.

We can see that all models yield a high performance for the sensitivity. On the other hand, with relevancy to the probability of default, the specificity is the most important measurement. The dataset contains very few defaulted observations, and it is often the case, when one class is predominant in the dataset, that the model overgeneralize. In this specific case, the non-defaults. With regards to specificity, the logistic regression is the best performing model followed by the SVM.

The likelihood is an important measurement as well, as it indicates the likelihood of misclassifying an entity. For all models, this value is low. Here especially, is the performance of the MLP. The value is almost 1, meaning that a default has an almost equal probability of being classified as either default or non-default.

The last measurement is the area under the curve (AUC), specifically the area under the ROC curve. The graph below, present the different ROC curves:

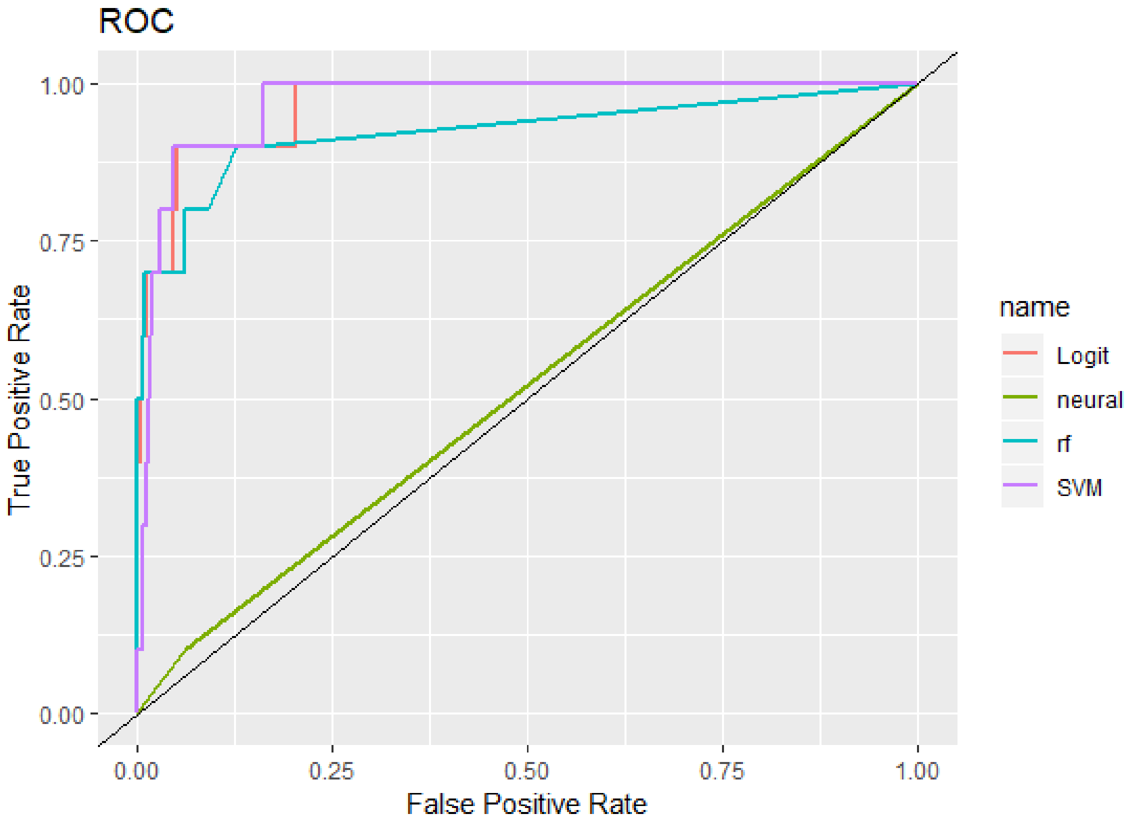


Figure 27 – ROC curve

The ROC curves are a graphical visualization of the True Positive Rate vs the False Positive Rate, in other words Sensitivity and (1-Specificity) as also used in the Likelihood Ratio. Area Under the Curve is the overall area underneath the ROC curve and is the measurable outcome from the ROC curve. As we can see from the graph, the Logit SVM, and Random Forest has the highest discriminatory power, on the other hand the more complex algorithm

- Neural Networks is performing bad and is having a hard time differentiating between the two classifications.

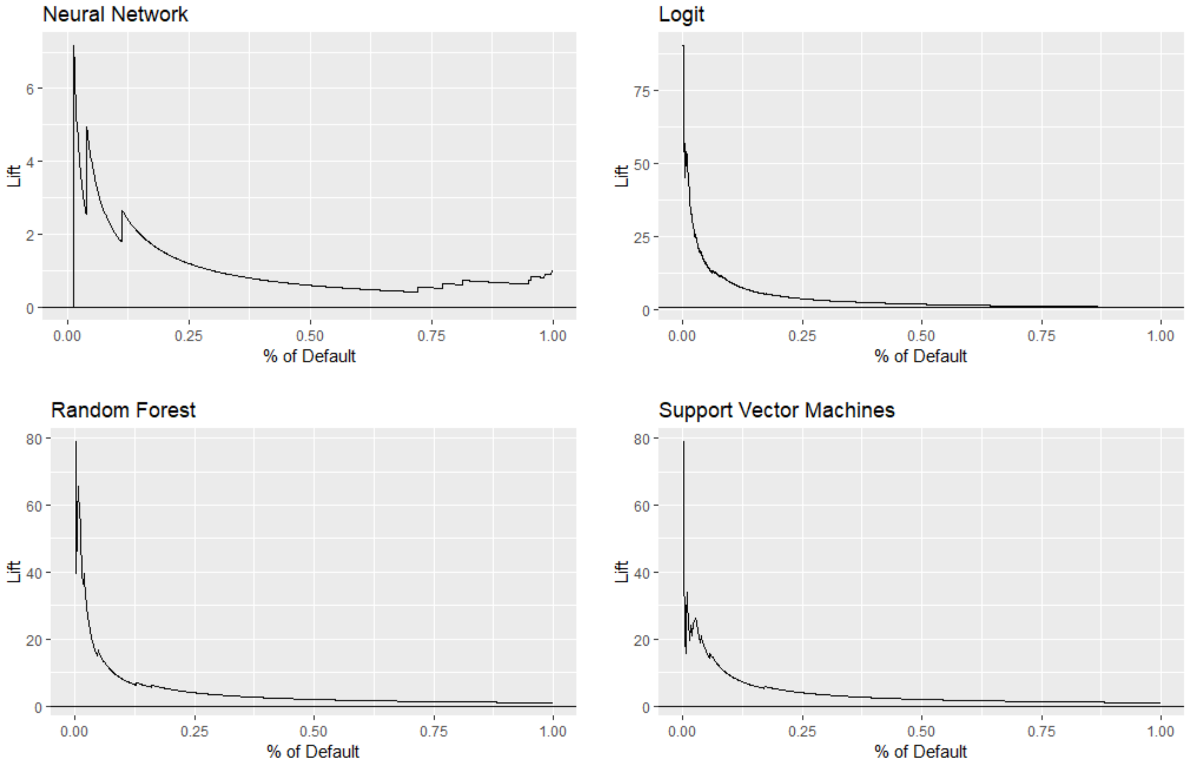


Figure 28 – Cumulative lift

The last performance measurement analyzed is the cumulative lift. As mentioned, it measures the level of acceptance. As also seen from accuracy and sensitivity, all models perform well predicting the non-defaults. However, in the case of predicting defaults, models underperform when compared to no model at all.

The next graph, as an example, presents the average expected probability of default obtained from the logistic regression per rating provided by Fitch. In accordance with the performance analysis, it can be seen how the low classification performance impact the expected default probabilities. In example, the average AAA (Lowest risk) rating has a higher probability of default than the AA+ rating.

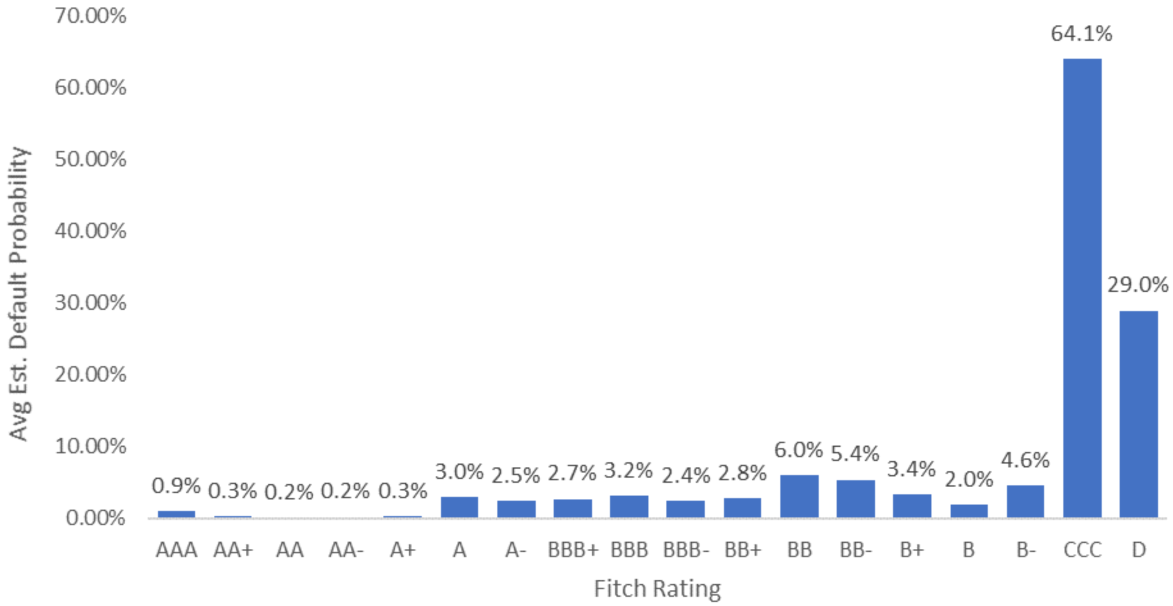


Figure 29 – Average expected probability per rating class provided by Fitch.

9 Conclusion and future research

9.1 Conclusion

The aim of this thesis was to investigate if machine learning can be used to develop an adequate model for assigning counterparties a standalone probability of default. The predictions were entirely based on companies past achievements without incorporation of any forecast of feature performance prospects. This implies that the most accurate assessment would always be achieved on the most recently available company information.

The logistic regression, Multi-layered Feedforward Networks, Support Vector Machines and Random Forest were all used in the attempt to calculate the probability of default for Nordea's counterparties. The variable selection and model development satisfy the requirements of certain estimation consistency underpinned by the Basel III framework.

When evaluating the results, we find little supporting evidence in promoting the use of machine learning algorithms for calculations of the probability of default. We adopted multiple statistical evaluation methods such as Sensitivity, Specificity, Likelihood, AUC, ROC and Cumulative lift in assessing the performance of the models. Even though the accuracy for the models is high, the models have a hard time predicting and classifying the defaulted observations. The models have been tested against logistic regression, and the results are either similar or strongly disappointing. Especially, the performance of neural networks was a big disappointment, as previous research such as (Angelini, di Tolo, & Roli, 2008) has indicated a high performance for neural networks.

We do believe that this paper is important in the extent that it can cast extra light upon the use of machine learning algorithms for default risk assessment for banks. As our models do not perform as wished, we want to point out the importance of securing high quality data for testing. As (Angelini, di Tolo, & Roli, 2008) specify in the success of implementing Neural Networks, "careful data analysis, data pre-processing and training" should be performed.

Preparing, analysing and training the used models has been extremely time consuming. Also, the loss of model development insight is high due to the “black-box” nature of the algorithms, the illogical weighting and overfitting of data as described by (Altman, Giancarlo, & Varetto, 1994). This also concerns the requirements from the Basel Committee, where the estimation must be logical and well documented.

In accordance with the results, we cannot conclude that there exists a clear relation between the input data and the probability of default. As classification perform reasonably well, there occurs miss classifications, which again impact the probability of default. Therefore, the selected models do not perform adequately for assigning counterparties a standalone probability of default.

9.2 Future Research

Overall, we find Random Forest and Support Vector Machine to be the best performing models. We wish to motivate future research to continue testing these models. There are great opportunities in terms of data quality improvements and data size that can address some of the challenges found in this thesis and therefore improve accuracy. Although the results may indicate that machine learning models cannot be used to calculate the expected probability of default, we believe the comparison of probabilities should be done over a larger amount of entities before reaching a final conclusion. Lastly, the data range is relatively short; including longer time horizon could improve data accuracy.

In terms of the legislative framework there must be developed strict guidelines for how datasets should be developed and how training should be conducted. The tuning of parameters must also be addressed, if machine learning should be used for calculating the probability of default. A data-modeller can influence the outputted result by a great amount based on the selection of data points, and overfitting or underfitting the estimations.

References

Agresti, A. (2012). *Categorical Data Analysis, 3rd Edition.* Wiley Series in Probability and Statistics.

Alpaydin, E. (2016). *Machine Learning: The New AI.* The MIT Press Essential Knowledge series.

Altman, E. I. (1968). Financial Ratios, Disriminant Analysis and the Prediction of Corporate Bankruptcy. *The Journal of Finance, Vol 23, Issue 4*, 589-602.

Altman, E., Giancarlo, M., & Varetto, F. (1994). Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience). *Journal of Banking & Finance, Vol 18, Issue 3*, 505-529.

Amaratunga, D., Cabrera, J., & Lee, Y.-S. (2008). Enriched random forests. *Bioinformatics, Volume 24, Issue 18*, 2010-2014.

Angelini, E., di Tollo, G., & Roli, A. (2008). A neural network approach for credit risk evaluation. *The Quarterly Review of Economics and Finance, Elsevier, vol. 48(4)*, 733-755.

Azuaje, F. (2005). *Data Mining: Practical Machine Learning Tools and Techniques 2nd edition.* Morgan Kaufmann Publishers.

Balin, B. J. (2008). *Basel I, Basel II, and Emerging Markets: A Nontechnical Analysis.* The Johns Hopkins University School of Advanced International Studies.

Bank for International Settlements Communications. (2010). *Basel III: A global regulatory framework for more resilient banks and banking system.* BIS.

Basel. (1999). *Core Principles Methodolog.* Basel Committee on Banking Supervision.

Basel. (2016). *Minimum capital requirements for market risk .* Basel Committee on Banking Supervision .

Basel. (2018, 05 31). *BIS*. Retrieved from BIS.org: https://www.bis.org/bcbs/membership.htm

Basel Committee on Banking Supervision . (2017). *High-level summary of Basel III reforms .* BIS.

Beaver, W. H. (1966). Financial Ratios As Predictors of Failure. *Journal of Accounting Research, Vol 4, Empirical Research in Accounting: Selected Studies, Wiley*, 71-111.

Bellman, R. (1961). *Adaptive Control Process: A Guided Tour.* Princeton University Press.

Bellotti, T., & Crook, J. (2009). Support vector machines for credit scoring and discovery of significant features. *Expert Systems with Applications Volume 36, Issue 2, Part 2*, 3302-3308.

Berg, D. (2005). *Bankruptcy Prediction by Generalized Additive Models.* Statistical Research Report No. 1, University of Oslo.

Bhattacharyya, S. (2000). Evolutionary algorithms in data mining: multi-objective performance modeling for direct marketing. *ACM SIGKDD international conference on Knowledge discovery and data mining, Vol 6*, 465-473.

Biau, G., Devroye, L., & Lugosi, G. (2008). Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research, Vol 9*, 2015-2033.

Biau, G., Devroye, L., & Lugosi, G. (2008). Consistency of Random Forests and Other Averaging Classifiers. *The Journal of Machine Learning Research, Vol 9*, 2015-2033.

BIS. (2016). *Minimum capital requirements for market risk.* Basel Committee on Banking Supervision.

Breiman, L. (1996). Bagging Predictors. *Machine Learning, Vol 24, Issue 2*, 123-140.

Breiman, L. (2001). Random Forests. *Machine Learning, Vol 45, Issue 1*, 5-32.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. (1984). *Classification and Regression Trees.* Wadsworth Statistics/Probability 1st Edition.

Buja, A., & Stuetzle, W. (2006). Observations on Bagging. *Statistica Sinica, Vol 16, No. 2*, 323-351.

C.A.E.Goodhart. (2008). The regulatory response to the financial crisis. *Journal of Financial Stability, Vol 4, Issue 4*, 351-358.

Christiani, N., & Scholkopf, B. (2002). Support Vector Machines and Kernel Methods, The New Generation of Learning Machines. *AI Magazine, Vol 23*, 31-42.

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning, Issue 3*, 273-297.

Cristianini, N., & Shawe-Taylor, J. (1999). *An introduction to support Vector Machines: and other kernel-based learning methods.* Cambridge University Press.

Derksen, S., & Keselman, H. J. (1992). Backward, forward and stepwise automated subset selection algorithms: Frequency of obtaining authentic and noise variables. *British Journal of Mathematical and Statistical Psychology, Volume 45, Issue 2*, 265-282.

Dietterich, T. G. (1997). *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms.* Oregon State University, Department of Computer Science.

Ding, C., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology, Vol 3, Issue 2*, 185-205.

Douglas J. Elliott. (2010). *A Primer on Bank Capita.* The Brookings Institution.

Elizondo, D. (2006). The linear separability problem: some testing methods. *IEEE Transactions on Neural Networks, Vol 17, Issue 2*, 330-344.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, Vol 27, Issue 8*, 861-874.

Financial Stability Board. (2018, June 30). *Financial Stability Board*. Retrieved from FSB.org: http://www.fsb.org/what-we-do/implementation-monitoring/monitoring-of-priority-areas/basel-iii/

Fitch. (2018). *Fitch Ratings, Rating Definitions.* Fitch.

Ghodselahi, A., & Amirmadhi, A. (2011). Application of Artificial Intelligence Techniques for Credit Risk Evaluation. *International Journal of Modeling and Optimization, Vol 1, Issue 3*, 243-249.

Gillick, L., & Cox, S. J. (1989). Some statistical issues in the comparison of speech recognition algorithms. *Acoustics, Speech, and Signal Processing*, (pp. 532-535).

Goodhart, C. (2011). *The Basel Committee on Banking Supervision: A History of the Early Years 1974–1997.* London School of Economics and Political Science.

Gouvêa, M., & Gonçalves, E. (2007). Credit risk analysis applying logistic regression, neural networks and genetic algorithms models. *POMS 18th Annual Conference.* Pomsmeetings.

Hair, J. F. (1998). *Multivariate Data Analysis.* Pearson.

Hastie, T., Tibshirani, R., & Friedman, J. (2013). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition.* Springer Series in Statistics.

Haykin, S. O. (2009). *Neural Networks and Learning Machines, 3rd Edition .* Prentice Hall, Neural Networks and Learning Machines sv. 10.

Hofmann, T., Scholkopf, B., & Smola, A. (2008). Kernel methods in machine learning. *The Annals of Statistics, Vol 36, Issue 3*, 1171-1220.

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression, 3rd Edition.* Wiley Series in Probability and Statistics.

Huang, Z., Chen, H., Hsu, C. J., Chen, W. H., & Wu, S. (2004). Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems, Vol 37, Issue 4*, 543-558.

Jickling, M. (2009). *Causes of the Financial Crisis.* Congressional Research Service.

Khashman, A. (2010). Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes. *Expert Systems with Applications: An International Journal, Vol 37, Issue 9*, 6233-6239.

Kirkos, E. (2015). Assessing methodologies for intelligent bankruptcy prediction. *Artificial Intelligence Review, Vol 43, Issue 1*, 83-123.

Kroon, S., & Lelyveld, I. v. (2018). Counterparty credit risk and the effectiveness of banking regulatio. *DNB Working Paper, No. 599*.

Kumar, P. R., & Ravi, V. (2007). *Bankruptcy prediction in banks and firms via statistical and intelligent techniques.* 1-28: European Journal of Operational Research, Vol 180, issue 1.

Lobo, J. M., Jiménez-Valverde, A., & Real, R. (2007). AUC: a misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography, Vol 17, Issue 2*, 145-151.

Lobo, J., Raimundo, R., & Jimenez-Valverde, A. (2008). AUC: A misleading measure of the performance of predictive distribution models. *Global Ecology and Biogeography, Vol 17, Issue 2*, 145-151.

Martin, D. (1977). *Early warning of bank failure: A logit regression approach.* 249-276: Journal of Banking and Finance.

May, R., Dandy, G., & Maier, H. (2011). Review of Input Variable Selection Methods for Artificial Neural Networks. In K. Suzuki, *Artificial Neural Networks - Methodological Advances and Biomedical Applications* (pp. 22-44). InTech.

Mays, E. (2001). *Handbook of Credit Scoring.* Business Series, Global Professional Publishing.

McNelis, P. (2005). *Neural Networks in Finance 1st Edition: Gaining Predictive Edge in the Market.* Academic Press Advanced Finance.

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, Vol 12, Issue 2*, 153-157.

navan.name. (2018, 08 20). *Understanding ROC curves*. Retrieved from Navan.name: http://www.navan.name/roc/

Nazeran, P., & Dwyer, D. (2015). *CreditRiskModelingofPublicFirms: EDF9.* Moody'sAnalytics.

Nordea Group. (2017). *Capital and Risk Management Report 2016.* Nordea.

Nordea Group. (2018). *Annual Report 2017.* Nordea.

Odom, M., & Sharda, R. (1990). A Neural Network for Bankruptcy Prediction. *International Joint Conference on Neural Networks*, 1638-168.

Ohlson, J. A. (1980). Financial Ratios and the Probabilistic Prediction of Bankruptcy. *Journal of Accounting Research, Vol. 18, No. 1*, 109-131.

Ozaki, T. J. (2015, 06 04). *https://tjo-en.hatenablog.com*. Retrieved from Machine learning for package user with R(5): Random Forest: https://tjo-en.hatenablog.com/entry/2015/06/04/190000

Ozaki, T. J. (2015, 05 22). *www.tjo-en.hatenablogg.com*. Retrieved from Machine Learning for package user with R(4): Neural Network: https://tjo-en.hatenablog.com/entry/2015/05/22/190000

Ozaki, T. J. (2015, 04 20). *www.tjo-en.hatenablogg.com*. Retrieved from Machine Learning for package user with R(3): Support Vector Machine: https://.tjo-en.hatenablogg.com/entry/2015/04/20/190000

Paduaa, L., Schulzeb, H., Matkovićb, K., & Delrieux, C. (2014). Interactive exploration of parameter space in data mining: Comprehending the predictive quality of large decision tree collections. *Computers & Graphics, Vol 41*, 99-113.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 27, Issue: 8*, 1226-1238.

Poole, D., Mackworth, A., & Goebel, R. (1998). *Computational Intelligence A Logical Approach.* Oxford University Press, New York.

Random forests. (2001). *Machine Learning, Vol 45, Issue 1*, 5-32.

Rezac, M., & Rezac, F. (2011). How to Measure the Quality of Credit Scoring Models. *Finance a Uver, Vol 61, Issue 5*, 486-507.

Rodríguez, J. D., Martínez, A. P., & Lozano, J. A. (2010). Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 569-575.

Roemer. (2018, 6 12). *Use Gaussian RBF kernel for mapping of 2D data to 3D*. Retrieved from Stackexchange: https://stats.stackexchange.com/questions/63881/use-gaussian-rbf-kernel-for-mapping-of-2d-data-to-3d

Salchenberger, L., Mine, C. E., & Lash, N. A. (1992). Neural Networks: A New Tool for Predicting Thrift Failures. *Decision Sciences Vol 23, No. 4*, 899-916.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development, Vol 44, Issue 1.2*, 206-226.

Scalelive. (2018, 08 30). *McNemar's test, Compare two observations of a dichotomous categorical outcome*. Retrieved from Scalelive: https://www.scalelive.com/mcnemars.html

Schölkopf, B., & Smola, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* Adaptive Computation and Machine Learning series, MIT Press.

Scott, D. W. (1992). *Multivariate Density Estimation: Theory, Pactice and Visualization.* Wiley.

Siddiqi, N. (2015). *Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring Vol 3.* Finance & Investments Special Topics, Wiley.

Souza, C. (2010). Kernel Functions for Machine Learning Applications. *Creative Commons Attribution-Noncommercial-Share Alike*.

Tam, K. (1991). Neural network models and the prediction of bank bankruptcy. *Omega, Elsevier, vol 19, issue 5*, 429-445.

Tam, K. Y., & Kiang, M. Y. (1992). Managerial Applications of Neural Networks: The Case of Bank Failure Predictions. *Management Science Vol 38, Issue 7*, 926-947.

TwarakaviJiri, N. K., Simunek, J., & Schaap. (2009). Development of Pedotransfer Functions for Estimation of Soil Hydraulic Parameters using Support Vector Machines. *Soil Science Society of America Journal, 73*.

Warnock, D., & Peck, C. (2010). A roadmap for biomarker qualification. *Nature Biotechnology, Vol 28, Issue 5*, 444-445.

Witzany, J. (2010). *Credit Risk Management and Modeling.* Oeconomica.

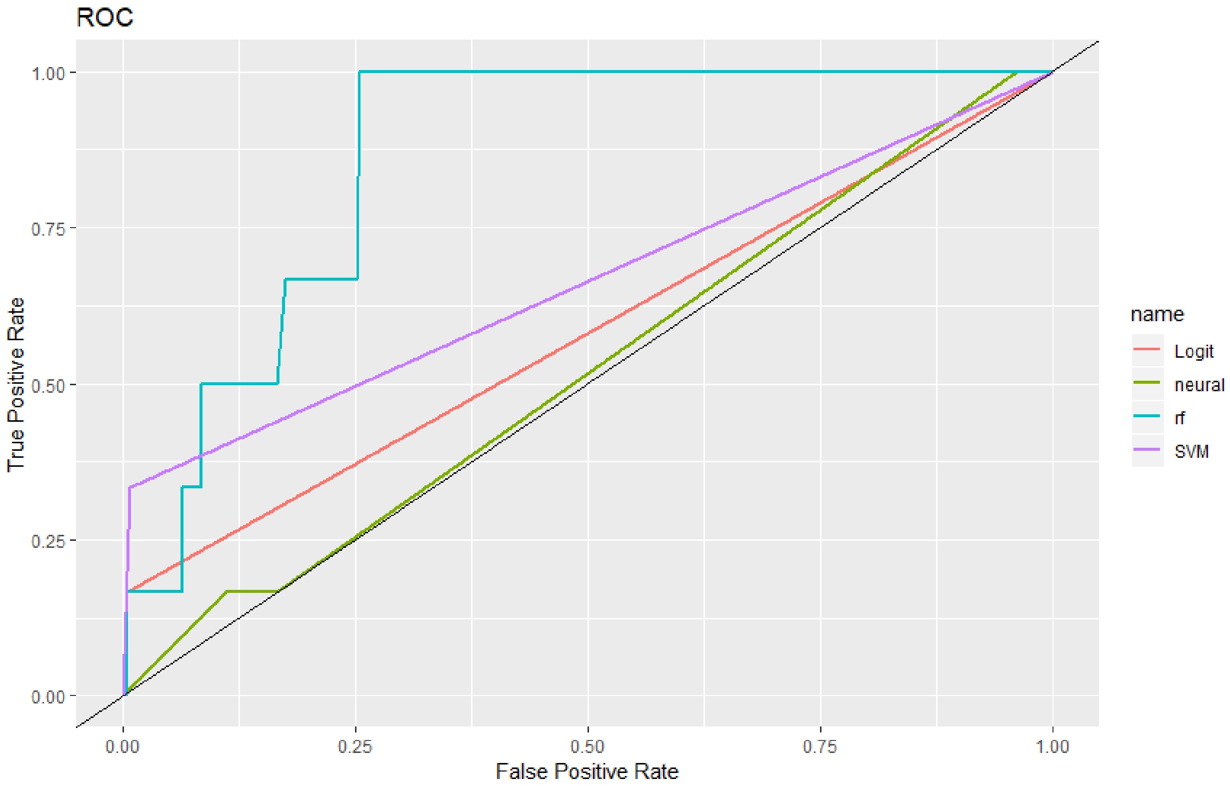
Yu, L., Wang, S., & Lai, K. K. (2008). Credit risk assessment with a multistage neural network ensemble learning approach. *Expert Systems with Applications: An International Journal*, 1434-1444.

Zweig, M. H., & Campbell, G. (1993). Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry, Vol 39, Issue 4*, 561-577.

B. Appendixes

We completed a similar test, with the same approach as for the Nordea Data set. The data was collected from the Wharton (Wharton Research Data Services). The period ranged from 2000 up until 2016, and explanatory variables were based on accounting statements. The rating, indicating the risk profile of the companies where gathered from Standard & Poors. – We obtained similar results as with the Nordea data. Some of the results are presented below:

	Logit	**RF**	**MLP**	**SVM**
Accuracy	0,9751037	0,92013	0,82141	0,9751037
Sensitivity	0,9957447	0,93213	0,96320	0,9831224
Specificity	0,1666667	0,6666667	0,25	0,5
Likelihood	1,1949	2,7964	1,2843	1,9662
AUC	0,58121	0,86170	0,52092	0,66241



C. Rstudio

All code has been written in Rstudio with the help of multiple packages. The code will be made available upon request.