Norwegian School of Economics
Bergen, Spring 2019

Machine Learning in Bankruptcy Prediction

Utilizing machine learning for improved bankruptcy predictions in the Norwegian market with an emphasis on financial, management and sector statements

Eystein Nordby Meese and Torbjørn Viken

Supervisor: Steffen Juranek

Master thesis, MSc in Economics and Business Administration

Major: Business Analysis and Performance Management,

Specialization in Business Analytics

NORWEGIAN SCHOOL OF ECONOMICS

Abstract

In this thesis, we create a new multi-year model for predicting bankruptcies in the Norwegian market. Our emphasis is on utilizing all parts of the financial statements and related information, rather than previously utilized ratios, to predict whether or not companies go bankrupt within the next three years.

Our analysis is based on a database that stems from a collaboration of previous research from the Norwegian School of Economics. After thorough cleaning, our final dataset contains 3 327 405 observations with 159 features related to financial, management and sector statements.

We perform our analysis utilizing nine models based on nine different machine learning techniques. For evaluation, we optimize our models toward the percentage of correct bankruptcy predictions.

Our best model is Random Forest, which yields an overall accuracy and a class independent accuracy of $\sim 78\%$, where the model is able to correctly predict 4/5 bankrupt firms and 4/5 non-bankrupt firms ahead of time. The results we obtain from Neural Network and Mixture Discriminant Analysis are slightly inferior, while the remaining models perform even worse.

Our Random Forest model outperforms other models built on a class distribution that is highly imbalanced. Furthermore, other studies often use ratios as features, and we find that our model assigns considerable importance to some of the individual components of their ratios, in particular, components related to liquidity. We also find components and features that are deemed important which have been neglected in the past ratio-focused research, such as cash flows, sector features and board features.

Keywords – Bankruptcy Prediction, Machine Learning, Norwegian Markets, Support Vector Machines, Random Forest, Generalized Models, Discriminant Analysis, Neural Network, Confusion Matrix, Multi-year Model, Financial- Management- and Sector Statements

Acknowledgements

This thesis was written during the spring of 2019, as part of our Master of Science degree in Economics and Business Administration, majoring in Business Analysis and Performance Management, with a specialization in Business Analytics.

We have found the study to be challenging, but rewarding, especially taking into consideration the vast amount of data. We believe that our findings will contribute to improving bankruptcy predictions in Norway, as well as internationally. To the best of our knowledge, no study of this scale has been created in the past, thus we hope that our results and insights may benefit this field of research.

Throughout this paper, we have had generous help and guidance from several stakeholders. We would like to thank Endre Berner, Aksel Mjøs and Marius Olving at the Centre for Applied Research at NHH (SNF) for providing access to the database. Furthermore, we would also like to thank the IT Department at NHH for providing access to virtual machines capable of handling the vast amount of data. We would also like to thank Dun & Bradstreet for additional data on bankruptcy. In addition, we thank Jonas Andersson, Martin Sørland Festøy, Anna Hjelle, Håkon Otnheim and Andreas Moltke-Hansen Tveten for their contributions.

Lastly, we would like to express our deepest gratitude to our supervisor and mentor, Associate Professor Steffen Juranek. With his support, input and guidance, the quality and actuality of our research have greatly improved.

Norwegian School of Economics

Bergen, May 2019

Eystein Nordby Meese

Torbjørn Viken

Contents

List of Figures

List of Tables

1 Introduction

Bankruptcy is a natural element and part of the foundation which modern economies is built upon. Bankruptcy was one of the reasons limited liability companies started to see the light of day during the seventeenth century in England, allowing the equity holders to only be liable for their own shares, thus decreasing personal risk. This change introduced the possibility of separating owners and management, which introduced new problems such as principal-agent problems, where the management and equity holders have different motives due to misaligned incentives. In addition, the problem is connected to adverse selection, where management and equity holders have different information due to information asymmetry. Therefore, the investors have to rely on information from the management and risk being misled. By creating robust models that help investors to assess the risk, the information asymmetry diminishes.

Bankruptcy empowers the market by eliminating companies which lack sufficient competitive advantages, such as companies with obsolete services or products, or other disadvantages. In such instances, the estate allocates the remaining values into more productive companies at the cost of the equity holders. Thus, the event of bankruptcy is anecdotal evidence for creative destruction, as first proposed by Joseph Schumpeter (Reinert and Reinert, 2006). The primary motivation for improving bankruptcy predictions is rooted in the increasing responsibility that companies face from its stakeholders. The common denominator is that all of these stakeholders are highly interested in the company's future. Equity holders are concerned for their equity placements, while workers are concerned for their jobs. Suppliers are anxious to get paid for their products, and the government wants to maximize job employment and tax revenue.

Therefore, modeling bankruptcy correctly is important for these aforementioned agents. The cost of misclassifying firms, especially bankrupt firms, can potentially be very high. Therefore, the high attention towards developing well-performing bankruptcy prediction models is justified, since these models could contribute to reducing the misclassification costs (Chen et al., 2011).

The financial crisis in 2008-2009 actualized the topic of bankruptcy yet again and the need for accurate models predicting bankruptcy properly. Last year the number of bankruptcy filings was at the highest level in 25 years, despite stable growth in the Norwegian economy[1]. The beginning

[1] https://www.dn.no/marked/ingar-nordmo-olsen/konkurser/bisnode/hoyeste-konkurstall-pa-25-ar/2-1-511553.

of 2019 brought further life to the topic, with a new all-time high bankruptcy figure in January[2]. Hence, bankruptcy is a trending topic, and it would seem that there is a potential for creating new models that better predict bankruptcy, reflecting today's markets and characteristics.

In recent years new methods of machine learning techniques have become available on a larger scale, along with more powerful data manipulation tools, enhancing the possibility of developing new, more advanced models. This enables us to process more data, not limiting the models due to variable selection and linearity. To our best knowledge, machine learning techniques have not been tested on financial, management and sector statements for predicting bankruptcy. Hence, we believe there is potential for developing improved models, utilizing these methods on a large dataset with a considerable number of features.

We optimize our models to correctly classify bankrupt companies at the expense of lower overall accuracy. This is because of the high related cost of misclassifying a firm that indeed goes bankrupt. Therefore, our models are aligned with the point of view from the banks, investors, shareholders and rating agencies.

Results of our study give further life to our belief, that financial, management and sector statements are valid and appropriate data that can be used to predict bankruptcy. In addition, more complex models like Neural Network and Random Forest show potential in bankruptcy prediction, being able to handle and finding patterns in the large set of data. Furthermore, our models utilize a considerable amount of the same components as the previous ratios-based research are built upon. On the other hand, we also find features that have been neglected and ignored in the past, which we find to hold predictive power.

[2]https://www.dn.no/handel/bisnode/konkurser/varehandel/konkursbolge-i-januar-hoyeste-vi-har-sett-noensinne/2-1-538107.

1.1 Objectives and Limitations

The main objective of this thesis is to create a new state-of-the-art multi-year model, customized for the Norwegian market, with information that is easily obtainable to the public at a low cost. We train our models using nine different statistical techniques, based on what techniques has shown promising potential in past studies and new techniques that are suspected to create good predictive models. Some of these techniques have either been non-existing in the past or been computational infeasible to solve until now.

Another key element is that we perform our tests on a large scale data sample on the basis of financial, sector and management statements, rather than previously used ratios. Hence, we are able to objectively observe what information the algorithms indicate holds predictive power, rather than imposing human intuition through a limited number of ratios. Therefore, a sub-objective is to (in)validate ratios utilized in past studies.

As a limitation, we exclude firms prior to 1999 due to accounting standards which are not coinciding. Furthermore, we exclude companies which operate within the financial- or insurance sector, due to significant differences in financial statements. Companies with special ownership, such as municipal or state-owned companies are also excluded. Moreover, this paper does not concern accounting standards and we have therefore made no measure to coincide *IFRS* and *NGAAP*.

1.2 Overview of Sections

This paper is divided into eight sections. Section two reviews previous literature within bankruptcy predictions. Section three describes the algorithms used, and how we create and validate our models, while section four considers data cleaning and hyperparameter tuning. Section five describes our results with the basecase and optimized thresholds, in addition to variable importance. Section six is dedicated to an applied case study of the best model. Section seven is dedicated to a discussion of our findings, while section eight concludes the thesis.

2 Literature Review

In the first part of our literature review, we present the standard models of bankruptcy prediction. The second part presents specialized models with special attributes related to bankruptcy prediction. In the last part of this section, we examine local customized models created for the Norwegian market.

2.1 Standard Models

2.1.1 Early Adaptions

Bankruptcy predictions and credit analysis have been around for a long time. The earliest evidence is from likelihood estimation in the 1890s (Correia, 2018). The analysis was primarily used by privately owned banks to grant loans for companies based on their creditworthiness, thus spreading the idea of ratio analysis. During the early 1900s, the structure became more standardized and contributed to the rise of credit men (Correia, 2018). In 1919, the Federal Reserve in the United States published its first ratio analysis of the federal bank to gain public momentum and start a public discussion about credit risk (Wall, 1919; Correia, 2018).

One of the earliest pioneers in bankruptcy prediction was Beaver (1966). He utilized a univariate analysis to find significant differences in several variables for two categorical groups, bankrupt and non-bankrupt firms. His analysis was conducted on a sample of 706 companies through a period of five years. The sample was selected to exclude certain sectors, and the division between these two groups was approximately 50% for all the years. The 30 variables that were selected were divided into five different subgroups sorted by attributes. These subgroups related to different parts of the companies' financial structure, such as cash flows, ratios related to net income, turnover and acid tests. Based on this model he created four propositions to identify distressed companies and appropriate thresholds for each of these ratios. Today, these thresholds are known as rules of thumb for the aforementioned ratios. In addition, Beaver started the trend of using financial data to systematically rate companies by creditworthiness. Later on, he also introduced alternative ratios (Beaver, 1968*a*) and how investors view distress, viewed from the stock market (Beaver, 1968*b*).

2.1.2 Altman Z-Scores

The most well-renown and applied model in the literature is the Altman Z-score (Altman, 1968). Based on predetermined ratios, he developed the Z-score model for bankruptcy prediction using Multivariate Discriminant Analysis. Based on previous literature and intuition, 22 potential ratios were compiled for evaluation. From the original list, Altman selected five ratios that were the best predictors in terms of overall performance. Altman's final model is presented in Equation 2.1.

$$Z = 0.012X_1 + 0.014X_2 + 0.033X_3 + 0.006X_4 + 0.999X_5 \tag{2.1}$$

Z = Overall index, where companies with a cutoff score[3] above 2.67 are classified as non-bankrupt. The five ratios Altman uses are:

X_1 = Working capital/Total assets

X_2 = Retained earnings/Total assets

X_3 = Earnings before interest and taxes/Total assets

X_4 = Market value of equity/Book value of total debt

X_5 = Sales/Total assets

Altman tested the model on a sample of 66 manufacturing firms, where 50% were categorized as bankrupt and 50% as non-bankrupt. To prevent a skewed dataset, he eliminated very small and very large firms from the sample, based on their total asset value reported in the Q-10 reports. The results one year prior to bankruptcy (t-1) were highly accurate, where 95%[4] were classified correctly. Furthermore, the results for t-2, t-3 and up to t-5 were respectively 72%, 48%, 29% and 36%. This indicates that for more than two years prior to the bankruptcy, guessing will yield better accuracy than utilizing Altman's model. In addition, a criticism toward the accuracy of the model in t-1 and t-2 is that the discriminant coefficients and the group distribution were derived from the same sample. Hence, one should expect a high accuracy and thus argue that a 95% rate is not very impressive in this case.

[3]The cutoff score is the threshold value, which the predictions are classified after. The cutoff Z-score consists of an upper and lower threshold. Below the lower boundary (1.81) there are no errors in bankrupt classification, while over the upper boundary (2.67) there are no errors in non-bankrupt classification. In between the boundaries, the classification is viewed as uncertain.

[4]Overall accuracy is calculated as correct classified observations/all observations. Overall accuracy is often referred to as mean accuracy. However, we consequently use the term overall accuracy in this paper.

To further test the effectiveness of the model, Altman did a hold-out sample test on the original data. While the result of these tests showed a 96% accuracy when tested on bankrupt firms, it performed worse on only non-bankrupt firms with an accuracy of 79%. The overall accuracy of the hold-out sample was 83.5%. However, it should be noted that the non-bankrupt group contained firms under financial distress, which had not yet gone bankrupt.

Later on, Altman et al. (1977) developed a new bankruptcy classification model, known as the ZETA-model, which considered explicit developments in the seventies with respect to business failures. They argued the need for a new model that took into account the change of the sizes of the firms going bankrupt and a broader model that also considered the retail industry, not only the manufacturing. In addition, the new model took into account changes in financial reporting standards and accepted accounting practices. They collected data for 53 bankrupt firms and a matched sample of 58 non-bankrupt firms, where the non-bankrupt group was matched to the bankrupt group by industry and year. In this dataset, the average asset size of the bankrupt firms were approximately $100 million, while the Altman study from 1968 had an average asset size of $6.4 million.

The final model consisted of seven variables (after assessing a total of 27 variables that were considered to be important), which were the return on assets, stability of earnings, debt service, cumulative profitability, liquidity, capitalization and size. In assessing the importance of the variables, all tests indicated that the cumulative profitability was the most important variable, contributing 25% to the total discrimination. Since the model is a proprietary effort, the parameters of the market are unknown.

In addition, the authors made slight adjustments of the financial statements, arguing that the reported values did not reflect actual values correctly. They argued that the most important adjustment was to capitalize all non-cancellable operational and financial leases. The capitalization was included in the firms' assets and liabilities. Other adjustments that they argued would be important included changes to reserves, minority interests, other liabilities, captive finance companies, other non-consolidated subsidiaries, goodwill, intangibles, capitalized R&D, capitalized interest and settled other deferred charges. The ZETA-model was, as the previous model Z-score, developed using a multivariate approach, with an analysis of both linear and quadratic structures.

The results of the ZETA-score presented an overall accuracy score of 92.8% one year prior to

bankruptcy for both the linear and quadratic function. It is interesting to note that the accuracy for bankrupt firms was 96.2% for the linear structure and 94.3% for the quadratic, and at the same, it was lower for the non-bankrupt firms with 89.7% and 91.4% accuracy, respectively.

The authors also applied the data on an extended basis, to make predictions 2-5 years prior to the bankruptcy. The results showed that the accuracy for the non-bankrupt firms stayed high through all these tests, which is not surprising since the data became increasingly imbalanced with an overweight of non-bankrupt firms, as no resampling occurs. The results for the bankrupt firms are more interesting, where the accuracy decreased for each lagged year. The results from the fourth and fifth year prior to bankruptcy showed the accuracy decreased to approximately 70% for both years for the linear structure, while the quadratic decreased considerably more, to approximately 50% (57.4% four years prior and 46.5% five years prior). Therefore, Altman et al. (1977) concluded in their study that the linear structure was superior to the quadratic. Compared to Altman's original Z-score model, the linear structure in the ZETA-model performs considerably better for 3-5 years prior to bankruptcy.

To further compare the accuracy between the two models, the authors also calculated the Z-score for the new sample and the ZETA-score for the 1968 Altman's sample. The findings showed that the ZETA-model outperformed the Z-model on the same sample, for all predictions, except for the five-year prior for the non-bankrupt firms. Furthermore, when applying the ZETA ratios to the 1968 sample, results again showed that the ZETA-model outperformed the older sample. Using the ZETA on the 1968 sample only resulted in slightly more accurate predictions than the older five-variable Z-score model. Hence, using the ZETA on older data did not seem to improve the predictions, concluding the new model is only appropriate to use on the newer data, with newer accounting standards. But, it is difficult to directly compare the models on the data, since the data from Altman et al. (1977)'s paper was adjusted as earlier described.

In the start of the new millennium, Altman (2000) revised his original Z-score and ZETA-model in a new paper. Here, Altman addressed the criticism of ratio analysis in financial predictions (Ohlson, 1980), and argued why ratios should be emphasized rather than downgraded. His concern with ratios was rather to detect which ratios were most important in detecting bankruptcy, what weights to give the selected ratios and how the weights should be objectively established. In subsequent testing of the Z-score on distressed firms over the last 30 years, Altman obtained an accuracy between 82% and 94%, based on one financial reporting period (one year) prior to

the bankruptcy, using an upper boundary cutoff of 2.67. However, the classification of bankrupt firms had increased substantially with 15% - 20% for all firms and 10% of the largest firms, while having a Z-score below 1.81. Therefore, Altman advocated using a lower Z-score cutoff than 1.81, which yielded the lowest overall error in the original tests. The results of the testing across time are shown in Figure 2.1.

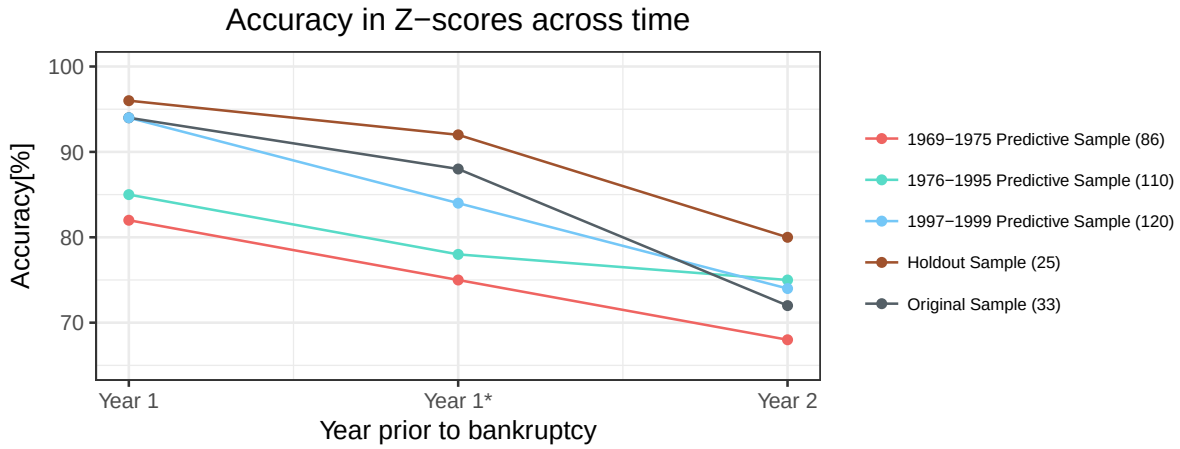

Figure 2.1: For Year 1 and Year 2 the Z-score is used with a cutoff value of 2.675. Year 1 uses a cutoff value of 1.81.*

In addition, Altman conducted a test on a second sample of non-bankrupt firms with 65% of the sample suffering from net deficits in two or three years over the last three years, sampling manufacturing firms from 1958 - 1961. The results showed that the model correctly classified 79% of the firms in the sample. However, this could imply that the result from net income, such as a deficit, does not hold very much predictive power for predicting bankruptcy. As a standalone ratio, it does not contribute enough to determine if a firm goes bankrupt or not.

In the same paper, Altman (2000) also presented the revised Z-score, which was adapted to consider private firms which were not publicly traded. Altman modified X_4 by substituting the market value of equity with the book value of equity. As expected, all the coefficients in the model changed, along with the classification criterion and the cutoff scores. The revised Z-score with the new X_4 variable became:

$$Z' = 0.717X_1 + 0.847X_2 + 3.107X_3 + 0.420X_4 + 0.998X_5 \tag{2.2}$$

Altman tested the model on the original sample from 1968, and the results showed a classification accuracy of 91% for bankrupt firms and 97% for non-bankrupt. In addition, the mean Z-score

for the non-bankrupt firms was lower, and the Z-score boundary became slightly wider than before, concluding that the new model was somewhat slightly less accurate than the original. Additionally, Altman also developed a customized model for non-manufacturing firms, coined Z"-score model, which is presented in it full in appendix A2.

2.1.3 Ohlson O-Score

Another famous bankruptcy predictor was created by Ohlson (1980). This model was created as a critique of the original multivariate models because he claimed that these models violated the assumptions they were built upon. One example was that both categorical classes should have the same variance-covariance matrix. The model was created by a sample of 105 firms that excluded firms in sectors with different financial structures such as financial companies, utility companies and REITs. From the sample, 17% was listed as bankrupt while the remaining part was non-bankrupt. All the firms had at least three years of consecutive financial statements of 10-Q. Based on this he created three models, each coinciding with the time of bankruptcy.

The first model predicted bankruptcy one year prior (t-1). This model obtained an accuracy of 96.12%. The second model predicted two years prior (t-2) and yielded an accuracy of 95.55%. The third model which predicted three years prior (t-3) yielded an accuracy of 92.84% (Ohlson, 1980). In order to optimize the results, Ohlson (1980) performed frontier trading to optimize the trade-off between type I and type II errors. A type I error refers to predicting a firm bankrupt, but it is actually non-bankrupt, while a type II error refers to predicting non-bankrupt, but the firm actually is bankrupt[5]. For model two, the optimal cutoff value was 0.08 yielding an error rate of 20.2% and 8.6% for type I and type II, respectively. Model one yielded similar results but with a slightly lower cutoff value. In hindsight, the models were poorly cross-validated, and as the author pointed out, the data was highly influenced by selection bias and survivorship bias.

The finalized, and applied model consisted of ratios from the financial statement, binary variables and a measure for the relative size of the company. The model is as follows:

$$\begin{aligned}\text{O-score} = &-1.32 - 0.407\log\left(TA_t/GNP\right) + 6.03\frac{TL_t}{TA_t} - 1.43\frac{WC_t}{TA_t} + 0.0757\frac{CL_t}{CA_t} \\ &-1.72OENEG - 2.37\frac{NI_t}{TA_t} - 1.83\frac{FFO_t}{TL_t} + 0.285INTWO - 0.521\frac{NI_t - NI_{t-1}}{|NI_t| + |NI_{t-1}|}\end{aligned} \tag{2.3}$$

[5] A more detailed explanation of the type of errors is referred to section 3.1.1.

where, TA = Total assets

GNP = Gross national product

TL = Total liabilities

WC = Working capital

CL = Current liabilities

CA = Current assets

OENEG = Dichotomous variable, 1 if TL is larger than TA

NI = Net income

FFO = Funds provided by operations

INTWO = Dichotomous variable, 1 if net income was negative for the last two years

Based on this model, he concluded that his financial statement variables were significant in the prediction of bankruptcy and that the relative size of the firm was important. The accuracy of his model is similar to Beaver (1966), but slightly inferior compared to Altman's Z-score.

2.1.4 Meta Analysis

Bellovary et al. (2007) published a thorough meta-analysis that examined over 170 previous studies within bankruptcy prediction, in which they compared the results and techniques from all papers. The first thing that is apparent is the overweight of studies on the US market with US companies and accounting standards, while other markets have been neglected. Of all the different papers only one is included from Northern Europe, written by Skogsvik (1990).

The majority of previous studies have been conducted on a sample consisting of usually manufacturing and retail firms. These models are called *unfocused models*, due to the flexibility built in them (Bellovary et al., 2007). On the other hand, *focused models* are primarily sector dependant either because the sector has special traits or the sector uses specialized accounting rules. Financial companies, insurance agents and REITs are examples of this. However, focused models can still be applied to general sectors to improve predictive accuracy, such as the studies by Gu and Gao (2000), Shah and Murtaza (2000) and Patterson (2001), where they examine bankruptcy predictions on hotels, software companies and casinos, respectively.

The meta-analysis also considers the distribution of frequency of different statistical techniques across time. Overall, Multivariate Discriminant Analysis has been the most used, with an

overweight in the early studies. Logit and Probit models became more popular at the end of the 20th century. For more specialized techniques such as Neural Network (NN), they gained momentum as a consequence of the change in computing power, especially around the 1990s (Bellovary et al., 2007).

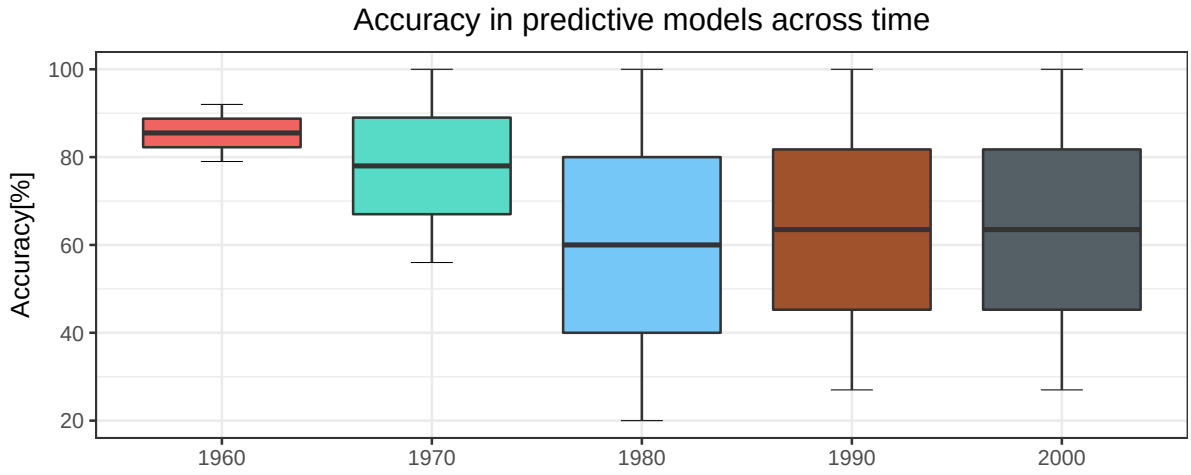

Figure 2.2: Accuracy in predictive models across time, data provided by Bellovary et al. (2007). For each box plot, the square indicates the first and third quartile, where the line shows the median. The vertical lines show the dispersion in the tails.

In Figure 2.2 we can see the change in overall accuracy as decades progressed. It is easy to see that the variation in accuracy has greatly increased with the years. The highest obtained accuracy was 100% in the last four decades, while the lowest was 20% in the 1980s. One possible explanation for this is due to poor cross-validation[6]. The earlier the model was created, the more computational expensive cross-validation was. In other words, the models were trained and tested on the same data. This was the approach that Altman (1968) used. However, this causes the model to have low predictive power out-of-sample. Furthermore, the large change across the decades could also indicate that researchers have relaxed their assumptions, reduced selection bias, increased sample sizes and utilized better cross-validation procedures.

In addition to overall accuracy, another measure that in recent years has become more important is true rates and false rates, where the former is correct classification, and the latter is incorrect classification. The true rates are divided in true positive rate (TPR) and true negative rate (TNR), while false rates are divided in false positive rate (FPR) and false negative rate (FNR)[7]. A

[6]The technical aspect of this will be discussed in section 3.1.2. However, the idea is to formalize the division between the testing and training data, to avoid overfitting.

[7]Throughout this paper, **we define bankrupt firms as "positive"**, in a sense that the observation is "positive" on the event of bankruptcy. Hence, TPR will refer to correctly classifying bankrupt firms.

detailed description of the performance measure is described in section 3.1.1. The two types of false rates are also known as type I and type II error, respectively.

Figure 2.3 plots a summary of the performance of different statistical techniques based on their respective true positive rate and true negative rate. We observe from the figure that all models are capable of producing good predictions. NN is the model with the highest variation in accuracy, while Quadratic Discriminant Analysis (QDA) overall seem inferior. Multivariate Discriminant Analysis is the technique that produces the highest TPR followed by NN.

An inherent question is why researchers still are captivated by bankruptcy prediction when the models yield such accurate results. The obvious answer is that most of the models are either created as a laboratory experiment and hence, that the findings do not coincide with real-life applications. If the models are able to correctly classify almost all bankrupt firms before they go bankrupt, why do companies still go bankrupt? From the applied side of bankruptcy predictions and credit modeling, few of the models are used today. Even worse, the model with the worst cross-validation, Z-score, is the most used in applied finance today.

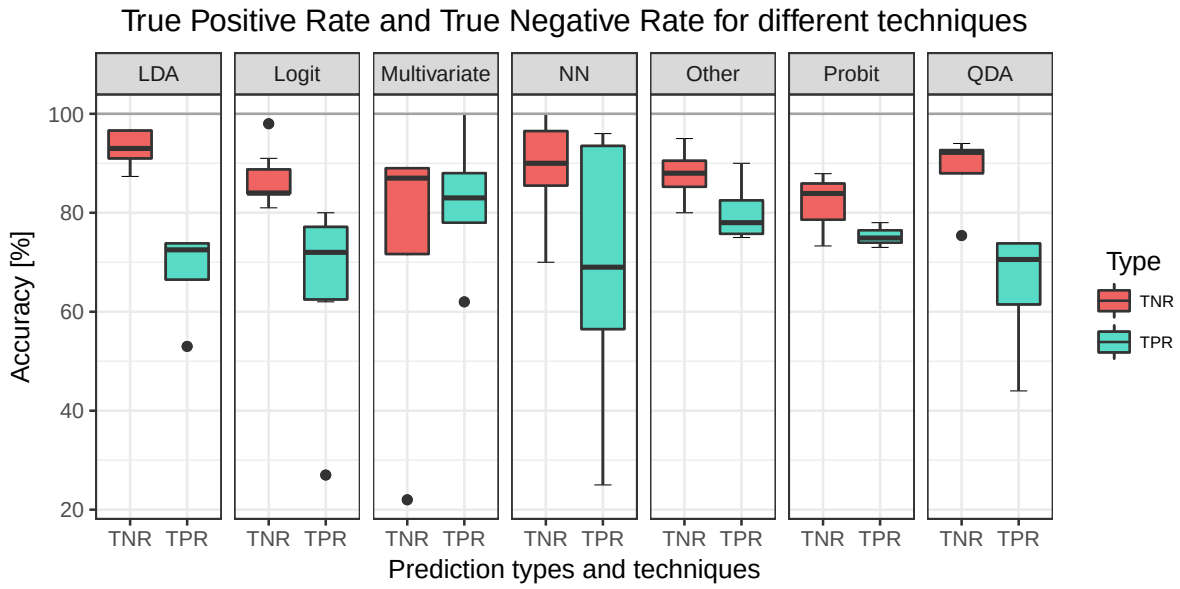

Figure 2.3: Classification rates for different techniques. Based on the best model fit from Anandarajan et al. (2001); Wang (2004); Grover (2003); Agarwal (1993); Gaeremynck and Willekens (2003); Shumway (2001); Lee et al. (1996); Zhang et al. (1999); Kiviluoto (1998); Tam and Kiang (1992); A. Mahmood and C. Lawrence (2007); Gentry et al. (1985); Casey and Bartczak (1985); Patterson (2001); Skogsvik (1990); Bellovary et al. (2007).

The research papers that Figure 2.3 is based on, introduces a new trade-off related to the class frequency. Most of these models are fitted with a balanced response variable, i.e. the frequency of bankrupt and non-bankrupt firms are regulated. This has also been named class balanced

data. This would allow the algorithms to produce better predictions in-sample because there is a clear division between the observations. For real-life applications, these models would perform poorly because the algorithm would classify uncertain companies as bankrupt without hesitation. For such a dataset, consisting of 50% bankrupt and 50% non-bankrupt, this introduces the Naïve decision boundary for classification, where a naïve prediction will consist of 50% to one class, and the remaining to the other class, yielding an accuracy of 50%.

For a class imbalanced, or dominated, dataset, the response variable would reflect the true distribution of bankruptcy, hence take into account the fact that bankruptcy is an extreme event. Therefore, it would produce a lower true positive rate and lower overall accuracy, but for real-life applications, more sensible predictions. Hence, it is easy to see that overall accuracy is a poor measure to evaluate model performance in the case of bankruptcy prediction.

The time aspect of bankruptcy has received varying attention throughout the years. Bellovary et al. (2007) also examined this aspect of the models in their meta-analysis. The general hypothesis is that the closer the observation is to bankruptcy and/or distress, the easier it is to find the common denominator with the highest predictive power. Due to the structure of financial statement revisions and publications of the financial statements, there exists a significant delay that stakeholders are unable to overcome. The delay varies, but the standard procedure consists of filing the financial statements for year *t-1* in the first period of *t*. After this period, the statements are validated and then made public. On average, the annual statements are minimum half a year old at release and up to one and a half years old at the maximum. For quarterly reports, the lag is smaller, usually one quarter. For the financial statements in the latter case, this reveals the possibility to create more accurate trailing financial statements, which could be utilized rather than *old* annual statements. From empirical evidence, El Hennawy and Morris (1983) created a promising model that took this into account. In the study, they achieved an accuracy of $\sim 100\%$ in predicting five years prior to the bankruptcy. This is promising results, indicating, at that time, that the financial statements contain very clear predictors with superior predicting power. However, the accuracy obtained may be too optimistic for out-of-sample observations. Other studies that have similar approaches are Deakin (1972) and Dwyer (1992).

2.1.5 Ratio Analysis

As indicated in earlier sections, ratios are primarily used for bankruptcy predictions. These ratios have been selected on the basis of careful consideration and have been cherry-picked to represent the different financial sides of a complex company[8]. This has been the standard procedure for all bankruptcy predictors to our knowledge. The variation in the ratios are considerable and there exists no common consensus on which ratios to include. In Figure 2.4, we have included the most used ratios based on data from Bellovary et al. (2007).

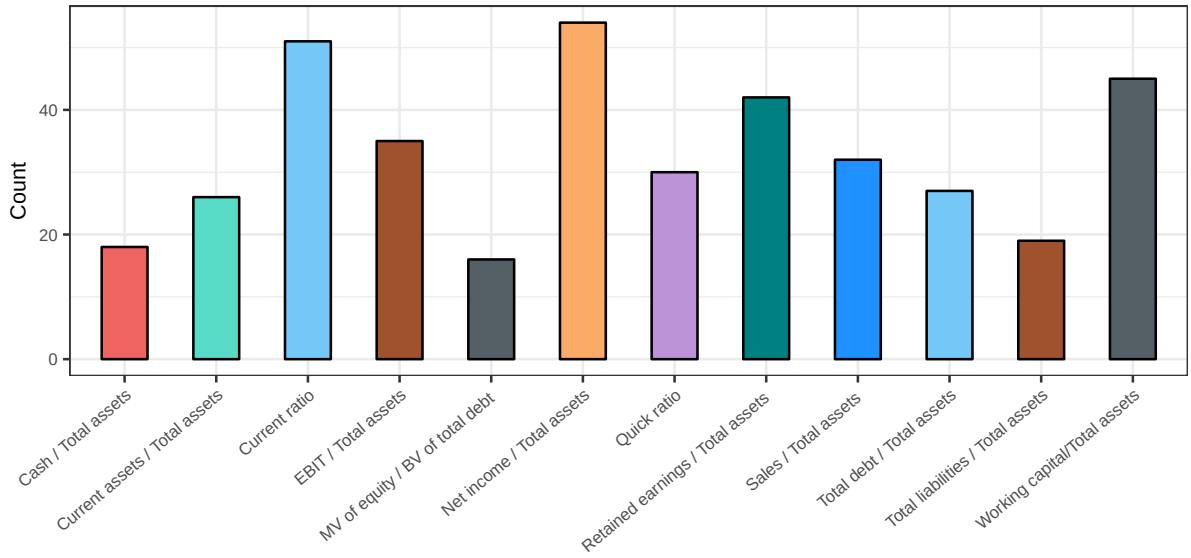

Figure 2.4: Distribution of previously used ratios.

As we observe from Figure 2.4, the most used ratio is Net income/Total assets, which yields a relative profitability measure given the total assets of a company. Perhaps more surprisingly is that previous studies prefer assets compared to liability or equity. One might argue that equity is more interesting because when equity goes toward zero, one would assume that the bankruptcy risk would increase. One possible explanation is that distressed firms generally have negative return on equity, and therefore, return on assets is a better measure to locate distressed companies. For an exhaustive list of these ratios, we refer to Bellovary et al. (2007). In order to maximize the number of correct bankrupt firms the model is able to predict, we believe the models need to be extended to include more than straightforward ratios that are capable of predicting *standard* bankruptcies. Bankruptcy is a complex firm-specific issue that can have

[8] As described, the original Altman Z-score utilized five different ratios related to different parts of the financial statements such as measures related to profitability, efficiency and financial structure.

several explanations and is the result of a series of corporate events and macroeconomic factors. Therefore, we believe it is insufficient to only concentrate on these cherry-picked ratios.

2.2 Specialized Models

2.2.1 Quarterly Predictions

As previously outlined, the timing and time-frame of prediction are crucial if the model is supposed to have any real-world value. A paper by Baldwin and Glezen (1992) could provide valuable insight into this effect. Their paper created a Linear Discriminant Analysis (LDA) model using 24 variables on quarterly data. The sample consisted of 80 companies from 1977 to 1983, where half was classified as bankrupt while the other half was non-bankrupt. The model was fitted on 10-Q reports that were filed with the Security and Exchange Commission (SEC). They predicted seven quarters prior to the bankruptcy, and surprisingly, the accuracy was higher the further away from bankruptcy the prediction was made. Seven quarters prior to the bankruptcy filing, they were able to predict 73% correct for bankrupt firms and 80% for non-bankrupt firms. However, one quarter prior to the bankruptcy filing the accuracy was 61% for bankrupt firms and 87% for non-bankrupt firms. The best predictions that they were able to obtain was six quarters prior to the bankruptcy filing, obtaining an accuracy of 86% and 93% for bankrupt and non-bankrupt firms, respectively. This implies that there is no superior time-frame for the financial statements nor any superiority for annual statements.

2.2.2 Banks and Financial Institutions

A more recent paper by Boyacioglu et al. (2009) gave another framework for bankruptcy predictions. They created a ratio model on Turkish banks from 1988 to 2000. This model was created in the aftermath of the financial crisis of 2007 and 2008 with the idea to predict future bank failures[9]. For the basecase, 20 different variables were calculated for the firms related to liquidity, sensitivity towards the market, asset quality, capital coverage, management and earnings. The dataset consisted of 44 non-failed and 22 failed banks. Their results are displayed in Table 2.1. The predictor performs very well under different statistical techniques. But, due to

[9]The underlying data used by Boyacioglu et al. (2009) is available at the Bank Association of Turkey's website: https://www.tbb.org.tr/english/bulten/yillik/1999/ratios/.

the nature of the problem, the sample size is very limited, which can limit the implication of the study. On the other hand, it provides some evidence for the branch of focused models and their ability to predict bankruptcy. Moreover, this study is an indication that financial companies should be treated differently than other companies.

	Accuracy [%]	
	Training	Test
Neural Network	100.00	95.50
Support Vector Machines	95.34	90.90
Multivariate Discriminant Analysis	88.37	68.18
K-means Clustering	86.04	81.8

Table 2.1: Sample of results from Boyacioglu et al. (2009) table 5, page 3362.

2.2.3 Bloomberg's Bankruptcy Predictor

Another popular bankruptcy predictor within finance is the Bloomberg DRSK models (Cai and Singenellore, 2012). These models are more comparable to credit modeling, in contrast to the previously presented models. The models build on Merton's Distance model, originally proposed for credit modeling and later extended to be included in the famous Black-Scholes model for option pricing. The underlying assumptions are continuous trading, short selling, frictionless trading and, more importantly, that the prices follow the Brownian motion (Merton, 1974)[10]. The models that Cai and Singenellore (2012) created, were a focused and unfocused model, where the former was intended for financial companies. In addition, they also separated the observations based on size, thus creating four mutually exclusive and collectively exhaustive models. The model for non-financial private companies gave an accuracy of 85.6% to 87.8% depending on the out-of-sample years.

The large benefit of this model is the ecosystem the model is applied in, namely the Bloomberg terminal, which has the most recent financial data available with trailing financial statements. For large US companies, Bloomberg also offers reclassified financial statements, which could improve the truthfulness of the financial statement, thus improve the accuracy. These models by Cai and Singenellore (2012) are built on similar assumptions as the models by Crouhy et al. (2000) and Zhang et al. (2009)[11].

[10]The Brownian motion was first proposed by Louis Bachelier in the 1900s, (Davis and Etheridge, 2006) but was perfected by Black and Scholes (1973). Simplified, the Brownian motion dictates that the motion is a random-walk.

[11]Note that these two models are based on default modeling rather than bankruptcy predictions, which is slightly different.

2.3 Local Adaptations

2.3.1 SEBRA

Until now we have primarily considered models that are customized for the American market and American accounting standards. In this section, we consider research completed for the Norwegian market and Norwegian accounting standards. Perhaps the most renowned model was developed by Bernhardsen (2001). This paper was the foundation for the later SEBRA model, utilized by the Norwegian Central Bank and the Financial Supervisory Authority of Norway to evaluate credit and default risk of Norwegian banks and financial institutions (Finanstilsynet, 2017). Therefore, the final model is not disclosed to the public. However, the original model was created on a population of 398 689 companies, where 8 436 went bankrupt ($\sim$ 2.11 % of the population).

Of this population, companies without significant assets (<500 TNOK) and financial statements with accounting and logical errors were excluded in the final sample. The model included several ratios that reflected the different sides of the business, such as liquidity, profitability, solidity, age, size and sector features (Bernhardsen, 2001). From his model, he achieved an *AUROC*[12] value of 0.8973 and an accuracy of 83% for both of the classes (Bernhardsen, 2001). This model has been named SEBRA Basic.

The model was extended and slightly revised further by Bernhardsen and Larsen (2007). This revision aimed to increase the accuracy for correct classification of bankrupt firms at the cost of lower overall accuracy. The revision included three additional variables related to the operations of the firms. The first was the aggregated value of assets, the second was accounts payable as a percent of total assets, while the third was payable public fees as a percent of total assets. The revision was created on a sample of approximately one million unique financial statements in the time period 1990 to 2002, where 20 000 firms were classified as bankrupt. The model solved the time dimensions using binary variables to tick progress towards bankruptcy. Companies that did not report financial statements within a consecutive period of three years were stated as bankrupt. This model has been named SEBRA Extended. The results from Bernhardsen and

[12] AUROC stands for the Area Under the Receiver Operating Characteristic curve, which is one of the most common evaluation measures for testing performance of classification problems at different thresholds. Simply put, it shows how much a model is able to differentiate the classes, i.e. how well the model correctly classifies the observations, regardless of frequency. AUROC will be thoroughly explained in section 3.1.1.

Larsen (2007) showed that the SEBRA Basic with new data achieved an AUROC of 0.88 while the SEBRA Extended achieved an AUROC of 0.89. For the different years, SEBRA Basic was superior in terms of AUROC from 1990 until 1998, while SEBRA Extended was superior for the remaining years[13].

One of the reasons that SEBRA Extended included payable public fees were most likely related to how bankrupt estates are split among its creditors. According to Norwegian legislation, the Norwegian government has a very high priority among creditors (Ministry of Justice & Public Security, 1986). Therefore, creditors that consider declaring a company bankrupt will only start these proceedings as long as the estate has sufficient funds to first cover public charges[14]. In such a setting, creditors would try to find other solutions than losing their claim.

2.3.2 Multi-Year Model

In another study contributing to the Norwegian bankruptcy predictors, Berg (2007) compared several accounting based models, which were developed and tested on a large dataset consisting of annual financial statements by Norwegian limited liability companies. The study compared different supervised learning methods with the hypothesis that Generalized Additive Models (GAM) would outperform the other methods. The methods the study examined, in addition to GAM, were LDA, GLM[15] and NN. In the dataset he examined, only 1% of the data were classified as bankrupt. However, as the author argued, this is representative of bankruptcy prediction since bankruptcy is a rare and extreme event. Furthermore, since the data consist of over 100 000 companies, there should be sufficient data to properly develop and validate a model. The variable selection was mainly based on the paper by Bernhardsen (2001), and in total 13 variables on financial ratios, sector indicator and the number of auditor remarks were selected.

Ten of the financial ratios were lagged on a change basis, indicating the relative change from the opening and closing balance. Thus, the model was fitted on 23 different variables. The model was developed on a training sample consisting of 60% of all observations, while the reminding 40% were used as validation, as an out-of-sample test. When interpreting the results, Berg used AUROC, proposed by Sobehart et al. (2000), to measure the predictive power of the models. In

[13] Note that the difference in AUROC across the years was insignificant.
[14] See appendix A1 for further discussion.
[15] GLM stands for Generalized Linear Models.

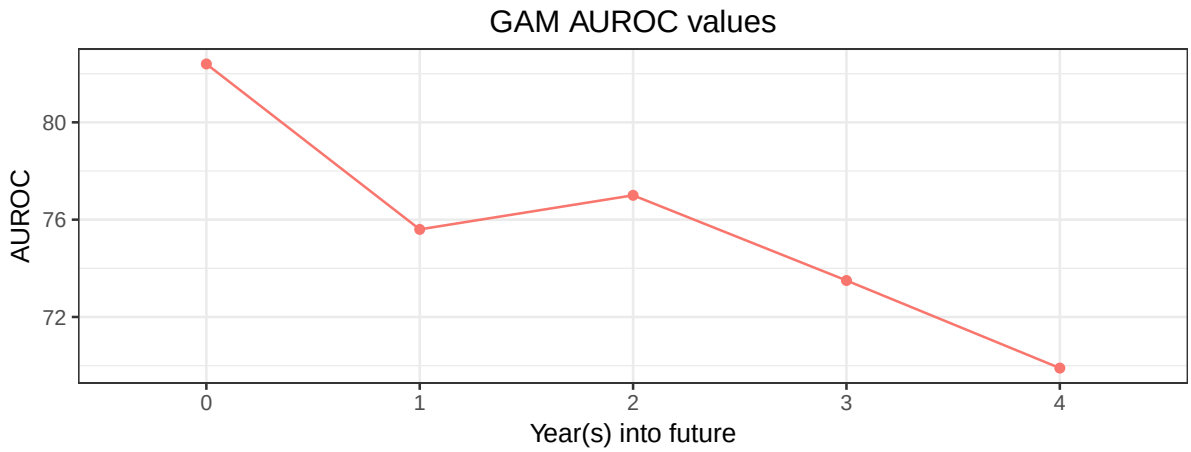

Figure 2.5: AUROC of GAM models n-years into the future, showing performance deprecation as time goes by.

the basecase, with cross-sectional data and a two year default time horizon, LDA, GAM and NN predicted fairly similar, with AUROC of 0.713, 0.720 and 0.723, respectively. The GAM model performed even better, with an AUROC of 0.773. For all the years in the basecase, LDA, GLM and NN performed equally well, while GAM showed superior results. In further testing, Berg developed several GAM models on the same data, testing with different time horizons for bankruptcy. Figure 2.5 displays the results of these tests. What is interesting to observe for Berg's results, is that the models seem to perform better looking two years into the future, compared to one year. This is in contradiction to most of the literature, where the closer in time a firm is to bankruptcy (or not), the higher the prediction accuracy will be.

At last, Berg also considered a multi-year model (consisting of three years), which he compared to a one year model. He argued that a multi-year model is able to consider more data and is less dependent on year-specific macroeconomic conditions, leading to more robust models. The result of these tests showed that the multi-year model outperformed the one year model significantly for a 0-1 year time horizon. On the other hand, comparing a two-year time horizon for the multi-year model against the one year model, he found no significant difference in performance. Berg concluded that data 1-2 years prior added significant information to the model that a one year model did not capture.

2.3.3 Machine Learning on Ratios

The latest addition on the Norwegian markets was completed by Næss et al. (2017), which was based on Wahlstrøm and Helland (2016). The paper harmonized predictions between the famous Altman Z-score, the Norwegian developed SEBRA model and their own sets of variables. For each ratio set, they performed the models on a range of statistical techniques, namely, LDA, QDA, GLM, GAM, KNN, SVM, CT[16] and NN. Regarding the method of NN, they implemented both forward and backward sweeps of back-propagation, in addition to a method involving dimensionality reduction.

Each model was trained and tested on the same sample, consisting of companies with financial statements from 2005 to 2014. Wahlstrøm and Helland (2016) followed the logic of Bernhardsen and Larsen (2007) and excluded companies with low total assets ($<$ 500 TNOK). Furthermore, they also excluded financial firms, in order to be consistent with Bernhardsen and Larsen (2007), thus being able to compare the models (Wahlstrøm and Helland, 2016). Moreover, they manipulated the distribution between the bankrupt firms and non-bankrupt firms, following the logic proposed by Boyacioglu et al. (2009). They divided the data to consist of 2/3 non-bankrupt firms and 1/3 bankrupt firms, where the sampling of non-bankrupt firms was random, while they included all of the bankrupt firms. This was in contrast to Berg (2007), which utilized the true distribution and rather adjusted the probability threshold for classification.

For the model creation, the financial statements were scaled to a mean of zero, thus eliminating the probability that statistical techniques adds importance to size, rather than distribution and distance (Wahlstrøm and Helland, 2016). The hyperparameters of the models were tuned to obtain the best predictions, and the models were cross-validated to ensure reproducibility and that the results were not a product of random chance.

In Figure 2.6, we observe that their unique variable selection outperforms the ratios provided by the Z-score and the SEBRA Extended model. As previous literature also has indicated, NN performs quite well. In line with results presented by Berg (2007), GAM is the technique that gives the best predictions with an AUROC of 0.911. In relation to the time dimension, Næss et al. (2017) performed another analysis on the data as panel data and their results, perhaps counter-intuitive, was that the time-series did not improve the performance.

[16] KNN stands for K-Nearest Neighbor, SVM for Support Vector Machines and CT for Classification Trees.

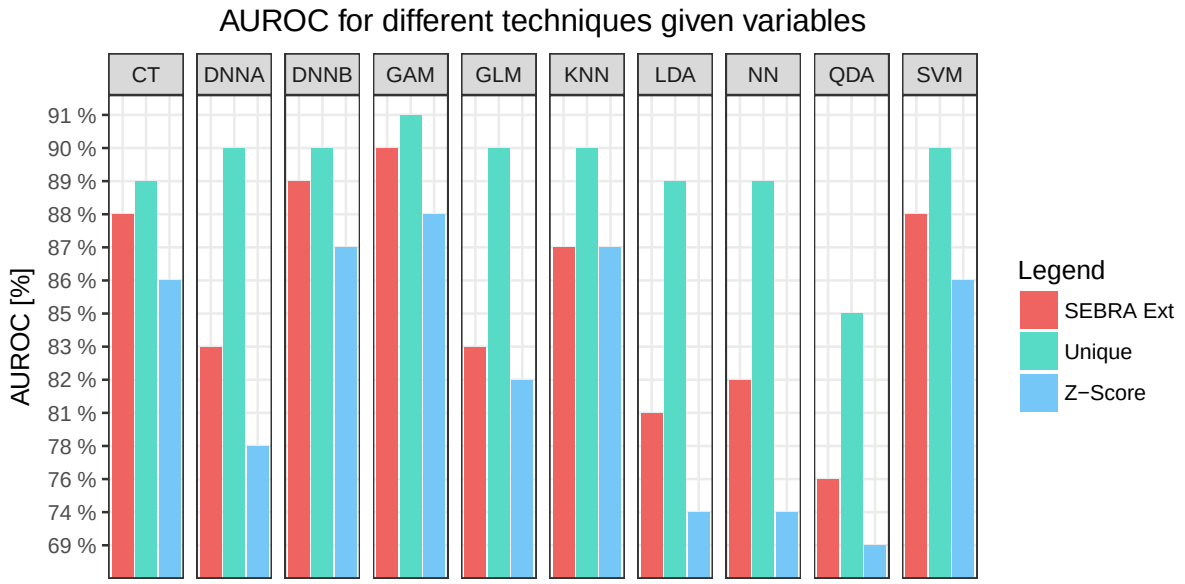

Figure 2.6: Achieved AUROC values for different statistical techniques by Næss et al. (2017) and Wahlstrøm and Helland (2016). SEBRA Ext refers to SEBRA Extended.

2.4 Dominant Responses

As previous research has outlined in sections 2.1.4 and 2.3.2, the event of bankruptcy is very rare and extreme. Therefore, the event in its true nature is skewed. Previous researchers have dealt with the issue in several ways. One approach is to limit the response variables to be equal or increase the relative frequency of the bankrupt companies. In practice, this requires the removal of excess non-bankrupt companies. This is similar to the approaches used by Altman (1968), Ohlson (1980), Beaver (1966) and Wahlstrøm and Helland (2016) to mention a few. On the other hand, one could expect that changing the distribution would affect the usability of the models. As Berg (2007) argued, this limits the accuracy and applicability in the real world. The same logic is used by Cai and Singenellore (2012).

One solution to deal with the class imbalanced response, which is adopted by Cai and Singenellore (2012), Berg (2007) and Bernhardsen (2001), is to change the classifying threshold. As an example, if the probability of observation i for either of the classes, *bankrupt* or *non-bankrupt*, is larger than 50%, the observation is classified as *bankrupt*, and vice versa. The solution requires lowering the threshold for the scarce class.

Cai and Singenellore (2012) solved this by lowering the threshold to 10%, meaning that observations that have a higher or equal probability, would be classified as bankrupt. The following credit scale is sourced from Cai and Singenellore (2012):

		Probabilities	
Credit risk class	Number of Levels	Lower bound	Upper bound
Investment grade	10	0%	0.52%
High Yield	6	0.52%	10%
Default	5	10%	100%

Table 2.2: Rating scale for Bloomberg DRSK. Sourced from Cai and Singenellore (2012).

From Table 2.2, the majority of the distribution is in the lower range on the probability scale, with 16 different credit score levels being assigned to companies with a probability below 10%. On the other hand, the remaining 90% of the probability scale is divided into five levels, where all observations are classified as default. Based on this insight, we also utilize 10% as our basecase threshold.

3 Methodology

This section is divided into two parts, where the first part is related to how the models are evaluated, cross-validated and where we discuss special traits that need consideration. In the last part of this section, we present the theoretical framework for the algorithms and considerations for our implementation.

3.1 Evaluation and Validation

As indicated earlier in this paper, which accuracy measure that is selected affects the applicability of the models and how the models perform out-of-sample. First, we present the performance measures. Furthermore, we will discuss how the cross-validation affects the performance and the interconnected bias-variance trade-off. Lastly, we dwell with the curse of dimensionality and considerations when the response value is imbalanced.

3.1.1 Performance Measures

In classification, the usual evaluation matrix is the confusion matrix shown in Table 3.1.

		Actual	
		Bankrupt	Non-Bankrupt
Prediction	Bankrupt	**TPR**	FPR
	Non-Bankrupt	FNR	**TNR**

Table 3.1: Confusion matrix.

In the matrix, the absolute predictions for each class, bankrupt and non-bankrupt are divided into correct and false predictions. To obtain the rate, the predictions are divided by the actual class. The matrix contains four different rates, where two are true rates, and two are false rates. The former, the *true positive rate* (TPR) and *true negative rate* (TNR), are correct predictions on the response, while the latter consists of *false negative rate* (FNR) and *false positive rate* (FPR). An example of the calculation of the rates is, for TPR:

$$TPR = \frac{True\ positives}{True\ positives + False\ negatives} \tag{3.1}$$

The FNR is often referred to as a type II error, while FPR is a type I error. A prediction marked as FNR is predicted non-bankrupt, but actually is bankrupt, while an FPR is predicting bankruptcy, but the actual observation is non-bankrupt (as we briefly explained in section 2.1.3).

Figure 3.1 display an extension of the confusion matrix in Table 3.1. Here, we plot the true positive rates versus the false positive rates. Any line in the figure is formally called the *Receiver Operating Characteristic curve* (ROC) (Fawcett, 2006). The associated *AUROC* gives the overall performance of a classifier, summarized over all possible thresholds. Therefore, AUROC is a favored classification metric that offers benefits of independence of class frequency or specific false negative/positive costs (Martens and Provost, 2011) and (Moro et al., 2014).

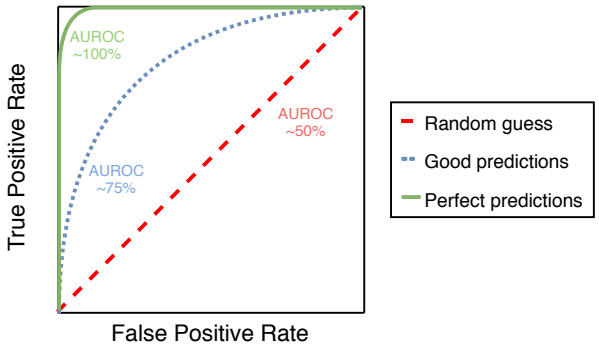

Figure 3.1: Receiver Operating Characteristics Curve.

In Figure 3.1, we see the different ROC curves with associated AUROC values. A naïve approach, by guessing at random, will yield an AUROC of 50%. In other words, it means that the model has no class separation capacity. Furthermore, the blue dotted line in Figure 3.1 represents good predictions that surpass the naïve approach, it has more true predictions and therefore a higher area under the curve. The green line represents perfect predictions and returns an AUROC of 100%[17]. With perfect predictions, the optimal point in the figure is in the top left corner, meaning TPR is 100% and TNR is 100% (TNR = 1 - FPR).

One important aspect of AUROC is that varying the classifier threshold value will also change the TPR and FPR (James et al., 2017). Since we have more non bankrupt firms than bankrupt firms, a threshold of 0.5 will most likely not give the best results in terms of overall error rate and AUROC. This is one advantage of the ROC curve, that the operator can choose the best TPR and FPR trade-off that fits the purpose of the model (Moro et al., 2014).

This trade-off is important in this classification problem since the different error types come at a

[17]This line is slightly skewed to make it visible and should have a 90-degree bend at the corner.

very different cost to different stakeholders. A type I error (FPR) has the implication that the bank does not grant a loan to a *successful* company. In other words, they lose out on potential interest. However, the other type of error, type II error (FNR), is viewed as much more severe. A result of this type of error is that the creditor looses all or parts of the capital lent to the company. Therefore, it is important to optimize the threshold with regards to maximizing TPR, which consequently minimizes FNR. The process of optimizing the threshold is explained in subsection, 3.1.5.

A more frequently used performance measure in the literature, but more problematic, is the overall accuracy, i.e. overall accuracy for both positive and negative rates. As a motivating example, we would achieve an accuracy of 97% in a sample of 100 000 observations, when only predicting the companies are non-bankrupt. This because the likely outcome dominates the output, and thus, a large overall accuracy could indicate the model only predicts the dominated response. Therefore, it is misleading to use this measure on a class imbalanced dataset w.r.t. to the response. Acknowledging this, we will not put too much weight on the overall accuracy in our models and mainly utilize the AUROC to assess the performance of our models.

3.1.2 Cross-Validation

Cross-validation (CV) is important to ensure good validity of the models that are created. However, as previous literature indicates, it has not always been that important or feasible. The earliest adoptions would usually test and train the model on the same data because information and computational power were scarce. This is a classic setting of overfitting the model, or in other words, customizing the model to the degree that new observations introduced would most likely be falsely classified. The first attempt on primitive cross-validation was introduced with the strict division between testing and training data. This reduces the problem of overfitting and thus, creates more robust models that hold predictive power outside the training environment. This method is still used today, but more advanced methods of CV are often preferred.

The method that we utilize to cross-validate our models is *k-fold cross-validation*. The method divides the data into different folds, with the intention of varying what data is tested and trained. For any given k, the model will be trained on k-1 folds and tested on the remaining fold. If the associated k is high, this will cause the model to be fit on many different combinations of the data, while if the k is low, few model fits are generated. The associated error rate for k-folds are

calculated as (James et al., 2017):

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} I(y_i \neq \hat{y}_i) \tag{3.2}$$

Where $\hat{y}_i$ is the predicted class and $I(y_i \neq \hat{y}_i)$ is an indicator equal to 1 if predicted outcome and actual outcome is the same, and 0 if not. A special case is if $k = n$, implying that all observations are trained once, which is called *Leave-one-out-cross-validation (LOOCV)*.

Another aspect of cross-validation that has become more important the recent years is the contribution towards hyperparameter tuning. In a nutshell, hyperparameter tuning is the introduction of integers to correct certain behaviors imposed by the algorithms. Thus, the CV procedure contributes to validate the hyperparameters across folds, which help to reduce the bias-variance trade-off, which we discuss in the next section, 3.1.3.

3.1.3 The Bias-Variance Trade-Off

The bias-variance trade-off is an important aspect of statistical learning methods (SLM) because it is rooted in the problem of overfitting and underfitting. The trade-off consists of how much variance and bias the model contains. To much bias will cause the model to tend to linearity, while too much variation will cause the model to find patterns that do not exist. The problem lies in the flexibility of the SLM at hand (James et al., 2017). In general, more flexibility in the method means higher variance and lower bias, where extremely flexible methods essentially can eliminate the bias. However, at some point, increasing flexibility will have a small effect on the bias and a notable effect on the variance, which leads to increasing test error rates. Thus, it is easy to obtain a method with low bias, but high variance and vice versa. A good performance of SLM requires low variance as well as low squared bias, and here lies the trade-off, finding a method that results in both being low.

With respect to *k*-fold cross-validation, this trade-off is represented in the selection of the *k*, the number of folds. Because a lower *k* results in lower bias, but higher variance, while increasing the *k* yields a lower variance, but a higher bias (James et al., 2017). From previous research, such as the classical paper Kohavi (2001), he showed that the bias and variance significantly improves when *k* is at least 5. For *k*-values of 10 and 20 certain improvements occur. For larger

k-values (k>20), the accuracy stops improving. James et al. (2017) conclude with a similar finding, that one should choose $k = 5$ or $k = 10$, as these have been empirically tested to result in estimations suffering from neither very high bias or variance.

Since our concern is primarily related to data size, and therefore computational time, we choose to pick the lowest *k* value that the literature indicates still is reasonable. Hence, we perform 5-fold cross-validation on our models.

3.1.4 Curse of Dimensionality

The expression ”Curse of dimensionality” was coined by Bellman (1957), and is a phenomenon that is present in high dimension space. Our data do not fit the strict definition of high dimensionality ($p \gg n$) (Hastie et al., 2009), but some of the insight is still applicable. As dimensions increases, this introduces a critical problem for computational algorithms as the number of nodes increases with exponential growth. Thus, the marginal computational cost will be severe when dimensions increases. Of our nine algorithms, the problem is most severe for KNN and SVM, where the number of interactions is high.

There are several methods that one could pursue to lift the curse, where the most popular is *Principal Component Analysis* (PCA) and *Partial Least Squares* (PLS), where the latter is usually better suited for classification (Hastie et al., 2009). PLS is an unsupervised learning technique that identifies an array of new features which are a linear combination of the original features. These new features are then fit using a linear model, using least squares, and then compared to the response to ensure they are credible and have predictive power. In essence, PLS tries to capture as much information in as few variables as possible, thus lifting the curse. Our KNN model utilizes PLS to reduce the dimensionality. The underlying mathematics of PLS is not in the scope of this paper, for more details refer to Hastie et al. (2009) and James et al. (2017).

3.1.5 Classification Thresholds

As outlined in section 2.4, the issue of dominant responses has been solved differently in the past. For creating a model that is applicable in the real world, we believe that dominant responses are inevitable and a necessary evil. However, there exist several solutions to the problem, such as

adding weight to observations that are prone to be neglected or by optimizing the classification threshold. For our paper, we believe that optimizing the threshold is superior because it is computationally expensive to train the data several times to find the optimal weights while the threshold can be optimized after the models have been created. By optimizing the threshold we will rather produce classification probabilities than class predictions.

From previous literature, a classification threshold of 0.1, or 10%, has been used with good results on applied models, shown in Table 2.2. The benefit of using the same threshold on all models is that it gives the possibility of easy comparison across models and that the threshold is logical. Therefore, we will fit our models on this threshold as a basecase.

However, this would probably not lead to optimal predictions for the TPR and TNR. Because, as discussed, it is costly to misclassify a firm that goes bankrupt (FNR), we want to optimize the TPR, and thereby predict the bankrupt firms to the best extent possible, which consequently minimizes the FNR. To deal with this, we optimize the TPR and TNR using AUROC, by calculating the Euclidean distance[18] between the TPR and TNR and the origin (0,0). Then, using the resulting smallest Euclidean distance value, we find the optimal threshold value to be used in classification separation, which we test on the best models attained from the basecase.

3.2 Machine Learning Algorithms

In the following section, we present the theoretical framework that the models are built upon. Primarily, we present the methods from a classification view. The scope of this paper is not to derive nor to show every mathematical aspect of the models. Therefore, for interested readers, we refer to other papers that elaborate on the details.

3.2.1 Generalized Models for Classification

Generalized linear model (GLM) is a flexible generalization of standard linear regression, which allows the response variable to have an error distribution model other than a normal distribution. This was first developed by Nelder and Wedderburn (1972), as a way of uniting different statistical models. In other words, we seek to model *p(X)* using a function that returns values for all *X* in the interval $\epsilon[0, 1]$. The following paragraphs are based on James et al. (2017). There are

[18]Euclidean distance is calculated as $d(p, q) = \sqrt{(p_1 - q_1)^2 + ... + (p_i - q_i)^2 + ... + (p_n - q_n)^2}$.

several ways of generalizing the linear model, but the most typical link function is the logistic (logit) model. In logistic regression, we use a logistic function, defined as:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}} \tag{3.3}$$

on the linear regression, where the *p(X)* is the probability distribution. The logistic function will always produce an *S-shaped* curve in the [0,1] space, see Figure 3.2 for an illustration. Hence, we always obtain sensible predictions, given the probability scale. After a bit of manipulation Equation 3.3 becomes:

$$\log\left(\frac{p(X)}{1-p(X)}\right) \equiv \beta_0 + \beta_1 X_1 + ... + \beta_p X_p \tag{3.4}$$

Where the left-hand side is called the logit. In this function, the coefficients β_i's are unknown, and will, therefore, be estimated based on the training data inputted to the model. The general method to estimate these coefficients in GLM is through *maximum likelihood*. The intuition is to estimate β_i, in order to predict the probability $\hat{p}(x_i)$ for each observation, such that the probability corresponds as closely as possible to actual response of the observation. Therefore, the objective is to find the optimal $\hat{\beta}_i$'s that when utilized in Equation 3.3, yields a probability as close to the binary outcome as possible. In other words, that the probability goes toward 1 (bankrupt) or 0 (non-bankrupt), respective of class. James et al. (2017) formalized this intuition into the likelihood function:

$$\ell\left(\beta_0, \beta_1\right) = \prod_{i:y_i=1} p\left(x_i\right) \prod_{i':y_i,t=0} \left(1 - p\left(x_{i'}\right)\right) \tag{3.5}$$

Where $\prod$ represents the product multiplication of the terms. Since β_0 and β_1 are not observable, we utilize the estimates represented as $\hat{\beta}_0$ and $\hat{\beta}_1$. These two estimates are chosen to *maximize* the function. Further mathematical details of maximum likelihood are beyond the scope of this paper, but for the interested reader, we refer to Dempster et al. (1977) and Hastie et al. (2009).

The method of *Generalized additive models* (GAMs) was first proposed by Hastie and Tibshirani (1986). The intuition was seeking to extend the standard linear model, and provide a general framework for this, allowing non-linearity in the functions of each variable, and still conserve

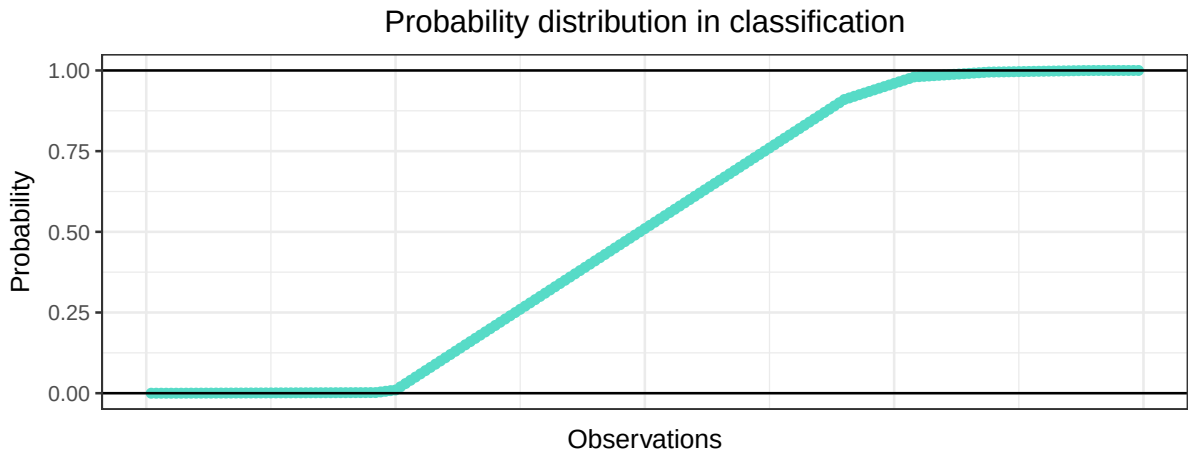

Figure 3.2: Illustration of a probability distribution in a classification setting.

additivity (James et al., 2017). The logistic regression of GAM is based on the GLM logistic regression, and is extended from Equation 3.3 to allow for non-linear relationships:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p) \tag{3.6}$$

Each linear component $\beta_i X_i$ in Equation 3.4 is replaced with a smooth non-linear function $f_i(X_i)$. The model calculates a separate f_i for each X_i, and then adding together all their contributions, therefore the regression is called an *additive* model. This is one of the advantages of a GAM, that it automatically model non-linear relationships that would have been missed with a standard linear regression. Furthermore, because the model is additive, GAM provide a useful representation if we are interested in inference since the method can examine the effect of each variable on the predicted outcome individually (X_i on Y), ceteris paribus. In addition, it is possible to summarize the smoothness of the function f_i for the variable X_i through degrees of freedom. The main limitation of the model is, ironically, the additivity, because, with many variables, there is a possibility that important interactions can be missed. However, it is possible to add low-dimensional interaction functions and/or manually add interaction terms to the model, to mitigate the problem. This is an important aspect we need to be aware off since our model is developed using an extensive number of variables. We add a number of binary and ordinal variables to try to mitigate the problem, as explained in section 4.

When performing these aforementioned methods, variable selection is important, due to the simplicity of the models. In addition, to the earlier mentioned PLS, *subset selection* and *shrinkage* are popular methods (James et al., 2017). Subset selection comes in a lot of varieties,

but the essence is to try different subsets of features to identify which is relevant and choose the features with the highest performance (James et al., 2017). Shrinkage methods rather utilize all predictors and shrink the coefficients towards zero (or equal to, depending on method), with the intention of ranking them (James et al., 2017). For our paper, we have not performed these operations, because we want to compare models across all features and more importantly, be able to (in)validate components of previously used ratios. As a consequence, the performance of GAM and GLM could perform worse than optimal.

3.2.2 Discriminant Analysis

For discriminant analysis, we introduce three types of discriminant analysis with different properties. Most prominent is the fit of the decision boundaries. *Linear Discriminant Analysis* (LDA), is an alternative to logistic regression that involves modeling the distribution, rather than directly calculating probabilities. Based on the distribution the observations are classified after a linear decision boundary. For the distributions, we utilize Bayes theorem to obtain the probabilities that observation *i* is either bankrupt or non-bankrupt (James et al., 2017)[19].

The Bayesian probability for each class is shown in Equation 3.7, where *Y* represents the response variable for *k* classes, such as bankrupt and non-bankrupt. *X* represents the density function for that particular observation, and which class it belongs in. This is formalized into the following formula by James et al. (2017):

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{t=1}^{K} \pi_l f_l(x)} \tag{3.7}$$

where π_k denominates the *a priori* probability that a randomly sampled observation belongs in class *k*. The $f_k(x)$ is the modeled density function, e.g. the normal or Gaussian distribution[20]. Simply put, each observation is tested against its peers on the distribution and classified thereafter. However, the standard framework of the distributions only allows for using one feature. Since we need to utilize more than one feature, we solve this by introducing a multivariate Gaussian density function, $f(x)$ represented as (James et al., 2017):

[19] As a reminder, Bayes rule dictates: $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$.

[20] The mathematical representation of this is not shown here since we will focus on multiple predictors instead, represented as $f(x)$.

$$f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\mathbf{\Sigma}^{-1}(x-\mu)\right) \tag{3.8}$$

First, we see that the suffix *k* disappears since the function is not dependent on *k*. In Equation 3.8 the *p* represents the number of predictors, Σ is a common covariance matrix for all responses of class *k*, and μ is the mean of each vector with *p* predictors. By plugging Equation 3.8 into Equation 3.7 we obtain an updated Bayes Classifier, or discriminant function, for each class *k*:

$$\delta_k(x) = x^T\mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T\mathbf{\Sigma}^{-1}\mu_k + \log\pi_k \tag{3.9}$$

By using Equation 3.9 for each *k*, observation *i* will be classified to the *k* with the coherent highest probability. I.e, which mathematical operator that holds true in the following relationship:

$$Pr(Y = Bankrupt|X = x) \leq \vee \geq Pr(Y = Not\ Bankrupt|X = x) \tag{3.10}$$

Until this point, we have only considered linear decision boundaries imposed by LDA. For the next paragraphs, we will relax this assumption by introducing *Quadratic Discriminant Analysis* (QDA) and *Mixture Discriminant Analysis* (MDA). QDA is built upon the same framework as LDA but with the intention of relaxing the assumption of linearity, thus allowing the boundaries to be more flexible. Using the same notation as in LDA, the discriminant function of QDA can be represented as follows (Hastie et al., 2009):

$$\delta_k(x) = -\frac{1}{2}log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + log\pi_k \tag{3.11}$$

where the largest difference from Equation 3.9 is that μ_k and Σ, the covariance matrix, is estimated for each class separately, thus allowing for different variances across different classes (Hastie et al., 2009) and (Taylor, 2018). The decision boundaries are given as quadratic functions given the *x*.

Another popular variety of discriminant analysis is MDA, also known as a Gaussian Mixture model. The general idea is to allow for more flexibility in the model compared to LDA. This is accomplished by allowing multiple prototypes on the feature space, i.e. several *regions* that yield the same prediction on the same plane. For LDA this is only one, given by the linear decision

boundary. The Gaussian density function for class *k*, is defined according to Hastie et al. (2009) as:

$$P(X|Y=k) = \sum_{r=1}^{R_k} \pi_{kr} \phi(X; \mu_{kr}, \Sigma) \tag{3.12}$$

where R_k represents the prototypes for class *k*. The π_k now represents the mixing proportion and the term will add up to one. The Σ is the same covariance matrix. ϕ represents the Gaussian density function. Calculating this for each class *k* will yield the posterior probabilities, formalized as (Hastie et al., 2009):

$$P(Y=k|X=x) = \frac{\sum_{r=1}^{R_k} \pi_{kr} \phi(X; \mu_{kr}, \Sigma) \prod_k}{\sum_{\ell=1}^{K} \sum_{r=1}^{R_\ell} \pi_{\ell r} \phi(X; \mu_{\ell r}, \Sigma) \prod_\ell} \tag{3.13}$$

As the aforementioned Equation shows, this is in a similar form to Equation 3.7, but more nuanced. The $\prod_k$ represents *a priori* probabilities for class *k*. The remaining parameters in the density function is estimated through a maximum likelihood method, similar to those introduced in subsection 3.2.1 on a joint log-likelihood basis given by *P(Y,X)*, which is reflected in the following expression (Hastie et al., 2009):

$$\sum_{k=1}^{K} \sum_{y_i=k} \log \left[\sum_{r=1}^{R_k} \pi_{kr} \phi\left(x_i; \mu_{kr}, \mathbf{\Sigma}\right) \Pi_k \right] \tag{3.14}$$

According to Hastie et al. (2009), the Equation is best solved by an *Expectation–Maximization algorithm*[21] since the properties of the algorithms increases the chances of convergence. In essence, the algorithm takes the parameters estimated by Equation 3.14 and by an iteration, calculate responsibilities and weighted maximum-likelihood estimates, converging towards a solution.

3.2.3 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a classification method that utilizes the relationship between observation *i* and observation *j*, and based on their relative distance, classify the observations to

[21] The technical and mathematical aspects of this algorithm are out of the scope of this paper, but for further reading consult Dempster et al. (1977) and Hastie et al. (2009).

a class such as bankrupt or non-bankrupt (James et al., 2017). The general formula is given by Equation 3.15:

$$Pr(Y = k|X = x_0) = \frac{1}{K} \sum_{i \in \eta_0} I(y_i = k) \tag{3.15}$$

Where x_0 is a random observation and *K* is the hyperparameter for how many neighbors each x_0 should have. Given the x_0 and *K*, the algorithm locates the relevant neighbors, represented in η_0. Then, evaluating the response value of these points in η_0, and estimating the conditional probability for each class, represented as *k*. Thus, the neighbors contribute to the classification of x_0. In the last step, KNN applies Bayes rule to get a probability that x_0 belongs in class *k*. The flexibility in KNN comes from the selection of *K* and composes the bias-variance trade-off. Usually, lowering the *K* leads to a lower bias at the cost of higher variance. In essence, when *K* is very small, it will pick up patterns that do not exist out-of-sample. If *K* is too large the model is unable to introduce sufficient flexibility, and therefore unable to follow the actual underlying patterns, i.e., it becomes linear. However, this problem can be managed through cross-validation and hyperparameter tuning.

The distance measure can be of any geometrical unit, as long as it indicates the length between two points. Alternatives such as Manhattan-, Euclidean- and Chebyshev distance are all feasible options. However, in practice Euclidean distance is most used, and the distance for an n-dimensional space in Euclidean distance is calculated as:

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + ... + (p_i - q_i)^2 + ... + (p_n - q_n)^2} \tag{3.16}$$

As outlined in section 3.1.4, KNN is computational demanding because the growth in computational nodes are exponential. Therefore, we will utilize PLS to reduce the dimensions with a stratified sample, which contain less data, making the technique computational feasible.

3.2.4 Classification Trees

Classification trees provide another variation on how to perform classification, by mimicking the human decision process (James et al., 2017). The versatility of the trees has caused them to gain popularity the recent years. The underlying logic of classification trees is very simple and can

easily be visualized. The technique consists of an iterative algorithm that divides the prediction surface based on different features as a measure of node purity. The regions are represented as R_m, while the selected array of observations is represented as N_m. The node, or branch, is then split as the following formula suggests (Hastie et al., 2009):

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k) \tag{3.17}$$

Where $\hat{p}_{mk}$ is the proportion (p) of observations in class k on branch m. The objective is to maximize the class separation across the prediction space. To validate this separation, and measure node purity, several methods could be applied. A selection that Hastie et al. (2009) proposes are misclassification error rate, Gini index and deviance. Throughout the paper we focus on the Gini index for classification trees, which is a measure of node purity and variance and is given by (James et al., 2017):

$$G = \sum_{k=1}^{K} \hat{p}_{mk} \left(1 - \hat{p}_{mk}\right) \tag{3.18}$$

In Figure 3.3 we show an illustrative tree where observations enter at the top of the tree and are classified based on its features. Firstly, the net cash flow is measured against the coefficient. If the observation (y_i) is below the corresponding value, the observation is measured against net income and so on. The length of each branch indicates the purity, given by the error rates (Ripley, 2018).

The vanilla flavor of classification trees is often criticized for being too unreliable and without sufficient predictive power. However, by utilizing methods to aggregate predictions from many trees with methods such as *bagging*, *boosting* and *Random Forest* (RF), it is possible to improve the robustness of the trees. We only consider Random Forest, since we believe it is superior compared to the other techniques due to the treatment of correlation. Random Forest de-correlates each subtree in the forest by imposing constraints on each branch, giving the model only a sub-selection of predictors to choose from. For each iterative split, the selection is shuffled at random. This forces the tree to consider all alternatives on an equal basis, contributing to a robust ensemble tree[22]. Our model is based on the logic proposed by Hastie et al. (2009), James

[22]A special case is if the sample of predictors is equal to all available predictors, then, the tree is simply a bagged decision tree.

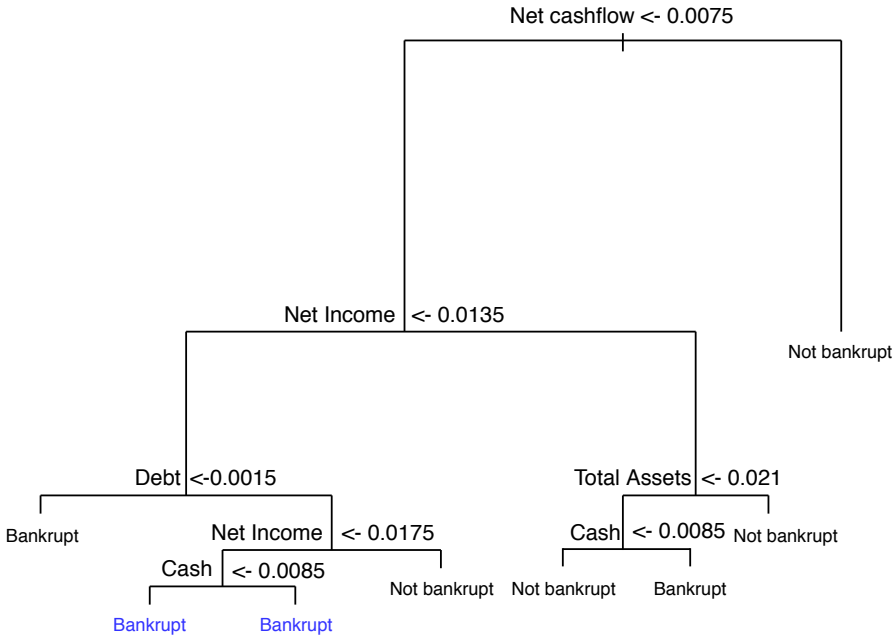

Figure 3.3: An illustrative example of a very simple classification tree, produced by 606 observations with normalized numbers and five features. The last split that is performed on the right leaf, in blue, yields the same prediction. This is performed to increase the node purity measured by the Gini index (James et al., 2017).

et al. (2017) and Breiman (2001), displayed in Table 3.2.

Random Forest Algorithm
This procedure is performed for fold $f \epsilon F$
1) For subsample $b \epsilon B$
a) Draw a subsample from b of size N
b) Grow the Random Forest trees on the data from a) by recursively repeating these steps until the nodes reach the minimum size, n_{min}
i) At random select m variables from the selection of p variables
ii) Pick the best split, were the classes are as separable as possible
iii) Split the node from ii) into two regions.
iv) Repeat i), ii), iii) until the minimum node size, n_{min}, is achieved.
c) Cache the fitted Random Forest
2) Repeat until all subsamples, B, have been trained, then ensemble the trees into one.

Table 3.2: The profound logic our Random Forest model is built upon. Influenced by Hastie et al. (2009),James et al. (2017) and Breiman (2001).

For the algorithm presented in Table 3.2, regarding step i), the menu of available variables at each split is $\sqrt{p}$, as proposed by (James et al., 2017). When aggregating each subtree to the ensemble tree, one easy approach is to take the average of all subtrees. However, this ignores the confidence in the votes from each tree. One alternate approach which we follow is to perform a soft decision boundary, i.e., taking the votes from each tree and adjust for their confidence. The

different Random Forests are then ensembled into one model[23].

3.2.5 Support Vectors

Support Vector Machines (SVM) is a relatively new method for classification. In this section, we first define a hyperspace, then consider the support vector classifier before presenting the support vector machines.

The general idea behind the approach is to place observations on a hyperplane of *p-1* dimensions, and use the observations for the different responses to guide vectors across the hyperplane. This is done with the intention of dividing the feature space between the responses. A good way to visualize this is through comparing it to the predictor space in classification trees in two dimensions. The hyperplane is, by definition, for *p* dimensions (James et al., 2017) and (Hastie et al., 2009),

$$\beta_0 + \beta_1 X_1 + ... + \beta_p X_p = 0 \tag{3.19}$$

where β_i's are the coefficients and X_i's are points on the hyperplane. In theory, if the data is separable, it is possible to divide the hyperplane into regions based on the response value. This would produce linear vectors across the hyperplane customized to the training data and inherently produce perfect predictions. Should this be true, it implies that the support vector could be drawn on infinitely many points on the hyperplane because there are infinite deviations that will not break this separation. Now, the question becomes which of these infinite vectors are the *true vector*. Therefore, the solution is to create an equal margin on both sides of the vector, where the support vector will be drawn at the equal distance between the two classes on the hyperplane.

Unfortunately, the data is seldom separable. The support vector classifier solves this by allowing a *soft-margin* of error. This implies that it allows for some misclassification, or violation of the margin, on the hyperplane (James et al., 2017). By performing this simplification, one is able to punish the model for violating the margin, by allocating a cost parameter, c. This hyperparameter is crucial ensuring the optimal bias-variance trade-off.

[23]Technically, the trees not aggregated, but ensembled. This ensemble tree considers the adjusted probability from the entire forest compared to an aggregated tree.

The support vector machines build on the solutions presented by the support vector classifier by enlarging the feature space using kernels. As a simplification, the enlarged feature space can be thought of like a stretched feature space, so that the vectors are linear. However, in reality, without this stretch, the vectors would be non-linear. Since this operation is very computationally demanding, the solution is to use kernels to simplify the curvature of the vectors. The underlying mathematics is not in the scope of this paper, and are only partially disclosed. For a full review, we refer to James et al. (2017), Hastie et al. (2009) and Steinwart and Thomann (2017).

The method will regardless of position on the hyperplane calculate the *inner products*. For two observations, x_i and x_i', it can be calculated as $\langle x_i, x_i' \rangle = \sum_{j=1}^{p} x_{ij} x_{i'j}$, where *p* is features. This expression is also known as the linear *kernel* (James et al., 2017). In general, we can define the kernel as a vessel that states the resemblance of two observations on the hyperplane. The kernel is a function related to the chosen distribution of the decision boundary. There exist several varieties of the kernel, while the most used are polynomial and radial kernels. In practice, it is uncommon to know the distribution precisely and therefore one of these two are often preferred. The function for these two kernels is stated below (James et al., 2017; Hastie et al., 2009):

$$K(x_i, x_i') = \overbrace{(1 + \sum_{j=1}^{p} x_{ij} x_{i'j})^d}^{\text{polynomial}} \qquad or \qquad K(x_i, x_i') = \overbrace{exp(-\gamma \sum_{j=1}^{p} (x_{ij} x_{i'j})^2)}^{\text{radial}} \tag{3.20}$$

Where the γ is a positive constant. For the radial kernel, it is important to calculate the proximity of the observation. Therefore, the standard procedure is to calculate the Euclidean distance between the training and testing points and afterward rank them. A special case for the polynomial kernel occurs if *d*=1, as the polynomial kernel then is linear and the method turns into the support vector classifier. We utilize the radial kernel from Equation 3.20 in the calculation of the support vector machine (James et al., 2017):

$$f(x) = \beta_0 + \sum_{i \in \varsigma} \alpha_i K(x, x_i) \tag{3.21}$$

where α can be thought of as a weight and ς represents a vector of indices based on the *coordinates* of the points. α and β are estimated through the inner products for all pairs of training observations (James et al., 2017). The benefit of utilizing kernels, rather than enlarging

the feature space of the original variables, is that kernels provide superiority in relation to computational time and complexity.

So far we have been concerned with vanilla SVM, but this approach is computationally expensive when training on large data. Therefore, we adopt and introduce an SVM variety proposed by Steinwart and Thomann (2017), called *liquidSVM*, which is less computationally demanding. The theoretical framework this model is built upon is similar to the one presented in Equation 3.21. However, some additional properties this alternative introduce are very advantageous for our problem and data. The main benefit is that the algorithm de-aggregates the feature space into several chunks, meaning that it manages each chunk separately before they are ensembled[24]. This allows for less demanding calculations and faster training of the models. This introduces the second benefit, that the chunks are already processed, which speeds up the cross-validation and simultaneously performs hyperparameter tuning. The tuning is performed over an adaptive grid of different parameter values where the algorithm chooses the best tuning parameters from a menu of possible options.

3.2.6 Neural Networks

Hopfield (1982) described *Neural Networks* (NN) as

> *A network or circuit of neurons, or in a modern sense, an artificial neural network, composed of artificial neurons or nodes.*

Or as more recently described by Rahbari (2014):

> *Artificial neural network is an interconnected group of natural or artificial neurons that uses a mathematical or computational model for information processing.*

This model is based on an approach to computation called *connectionism* (Garson, 2018). In a more practical sense, NNs are non-linear statistical data modeling, or decision-making tools, which are used to find patterns in the data in order to model complex relationships between the inputs and the outputs.

A NN is a two-stage model, which can typically be presented by a network diagram, as the

[24]The ensemble is similar to Random Forest, where the vector for each chunk is created using a least-square SVM model where each chunk is divided into neighborhoods, similar to KNN. Thus, each observation is considered against its peers and included in the chunk, as a consequence of its original position (Vapnik and Bottou, 1993). The technical process is not presented here, refer to Steinwart and Thomann (2017).

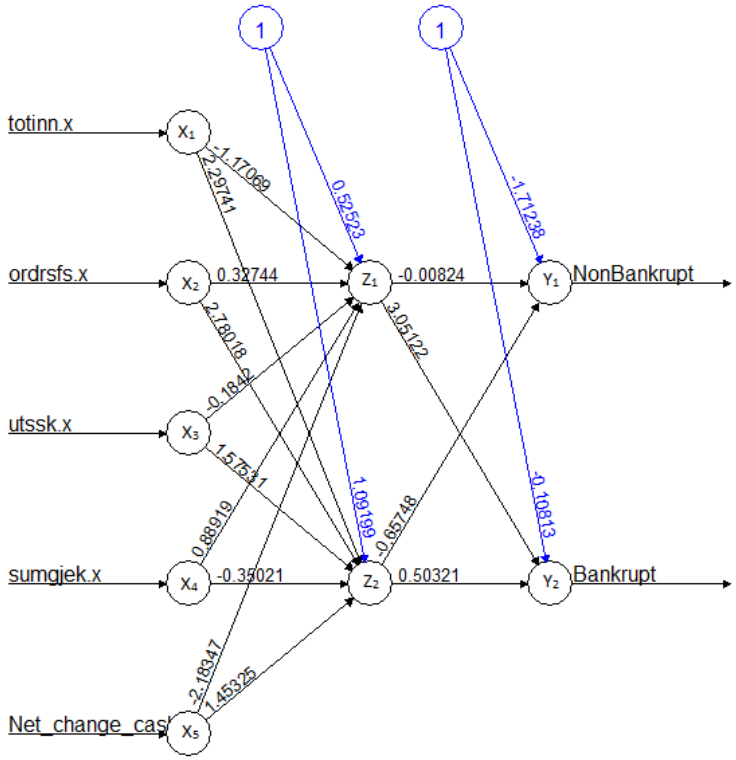

Figure 3.4: Illustrative example of a Neural Network, created using 5 variables and 10 observations, with 1 hidden layer (l) and 2 hidden nodes (Z_m) from our dataset. The blue line represents the additional bias unit, feeding into every unit in the hidden and output layers.

example in Figure 3.4 shows. Referring to the figure, there are *two* units at the most-right in a binary classification, Y_k, where the probability of class *k* is modeled by the *k*th unit (Hastie et al., 2009). These two target values, Y_k, are classified as either bankrupt or non-bankrupt for the two classes in Figure 3.4. In the figure, there is one *hidden layer* (*l*) of *Z*, but note that a NN could consist of unlimited hidden layers. The derived features, Z_m, called *hidden units* because they are not observed directly, are created from linear fusions of the inputs x_i in the figure, which can be thought of as a *black-box*. The output Y_k is modeled as a function of linear fusions of Z_m, where sub-score *m* is an index referring to the hidden unit (Hastie et al., 2009):

$$\begin{aligned} Z_m &= \sigma\left(\alpha_{0m} + \alpha_m^T X_i\right), m = 1, \ldots, M, \\ Y_k &= \beta_{0k} + \beta_k^T Z_m, k = 1, 2, \end{aligned} \tag{3.22}$$

where, $Z = (Z_1, Z_2 \ldots, Z_M)$

$Y = (Y_1, Y_2)$

α_m^T = weight to each hidden unit

X_i = input variables

β_k^T = weight to each output class

α_{0m} and β_{0k} = intercept terms, including bias unit[25]

$\sigma(z)$, the activation function, most commonly set to be *sigmoid:* $\sigma(z) = 1/(1+\epsilon^{-z})$.

Now, in our case of binary classification, we make use of the *softmax* function (Hastie et al., 2009), that allows for a final transformation of the vector of targets Y:

$$g_k(Y) = \frac{e^{Y_k}}{\sum_{\ell=1}^{K} e^{Y_\ell}}, k = 1, 2, \tag{3.23}$$

which we clearly observe is the same transformation used in the logit model and produces estimated probabilities that sum to one.

When fitting the NN-model, we seek values for the unknown parameters α_m and β_k, called weights, that make the model fit well with the training data. The complete set of weights are denoted θ. In section 3.2.4 we introduced several measures of fit, and for NN we use cross-entropy (deviance) (Hastie et al., 2009):

$$R(\theta) = -\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik} \log g_k\left(y_i\right) \tag{3.24}$$

Where *G(x)* = $argmax_k g_k(y)$ is the corresponding classifier. Now, all the parameters are estimated by maximum likelihood in the NN, with the softmax algorithm and cross-entropy error, since it is a linear logit model in the hidden units. Furthermore, we want to minimize the error term $R(\theta)$, but we do not want to find the global minimum since this is likely to be an overfitted solution. Hence, the minimizing approach is accomplished by a gradient descent method, which is called *back-propagation*, which can be computed by a *forward* or *backward* sweep of the network[26]. We use the feed-forward back-propagation approach in our model. We illuminate parts of the process of back-propagation, that involves *batch learning*, where the parameters are summed over all the training cases. The *learning rate* γ_r for batch learning controls the speed of how fast (or slow) the model learns the problem at hand and is an important hyperparameter in order to achieve good performance. The learning rate updates the model for each step (i.e. for each *number of rounds* the model is trained). Another important hyperparameter in the technique is the *momentum* (*m*), which is designed to accelerate the

[25] constant "1", drawn in blue in figure 3.4.

[26] The technical details of the back-propagation are quite advanced and not necessary for this paper to examine. The interested reader is referred to Hastie et al. (2009) chapter 11 or Hecht-Nielsen (1992).

learning process, especially in cases with high curvature and noisy gradients (Goodfellow et al., 2016).

In addition, there are also some issues with training neural networks that we need to be aware of since the model is, unless we take some precaution, overparameterized and the optimization problem is unstable and non-convex (Hastie et al., 2009). One issue regards the starting values of the weights[27]. If one uses weights of exactly zero, this leads to zero derivatives and perfect symmetry, hence the algorithm will never move. In the other end, starting with large weights will, more often than not, lead to poor predictions. Usually, one chooses a starting value at random, which is near zero, increasing the weights from there, finding the value that achieves the lowest error. This could be tedious and cumbersome. Therefore, Ripley (1996) proposes that a better approach is to take the average predictions over the complete collection of the NN and use this prediction as the final result. He argues that this is preferable compared to averaging the weights since the models non-linearity suggests this solution could be unsatisfactory.

The number of hidden nodes, Z_m's, is also an aspect to be aware of when training a NN. Problems can arise with both too few and too many hidden nodes; the former may lead to the model having insufficient flexibility to capture the non-linearity, while the latter may lead to additional weights that are shrunk towards zero, as earlier explained. Therefore, a rule of thumb in the literature seems to be a number of hidden nodes between 5 and 100, with an increasing number of hidden nodes, given an increasing number of inputs and observations. The choice of hidden nodes should be guided by background knowledge and experimentation (Hastie et al., 2009). According to learned scholars, two simple methods to determine the number of hidden nodes are either to take the square root of both input nodes (N_i) + output nodes (N_o) (Masters, 1993), or the "rule of thumb" method to weight according to the size of the sample (N_s):

$$N_h = \sqrt{N_i + N_o} \quad \text{or} \quad N_h = \frac{N_s}{(\alpha * (N_i + N_o))} \tag{3.25}$$

where α is an arbitrary scaling factor, usually between 2 and 10. We find our optimal values for hidden nodes and layers using these methods, as explained in section 4.2. There we also explain how we tune the other hyperparameters we need for NN, as well as for KNN, RF and SVM.

[27] It is important to be aware of the non-convexity of the error function $R(\theta)$. This means that the error function contains multiple local minima, which means that the final solution is very dependent on the starting weights.

4 Data and Tuning

This section is divided into three parts, where the first part is related to data cleaning, while the second part considers hyperparameter tuning of each technique introduced in the previous section. The third part considers randomization and reproduction.

4.1 Introduction to the Data

Our data was provided by the Centre for Applied Research (SNF) at the Norwegian School of Economics (NHH). The database was the result of a merger of several earlier research projects at SNF (Berner et al., 2016). The majority of the accounting data comes from Dun & Bradstreet (D&B), stock exchange information was sampled from NHH's database of security prices (Norwegian School of Economics, 2019), while other miscellaneous information was provided by governmental institutions in Norway. The data set has been quality checked by Menon Business Economics AS.

The database consists of all Norwegian corporate accounts from 1991 until 2016 regardless of size, legal structure or type. The data consists of approximately 4 600 000 company records, 150 000 group records and the associated data related to the sector, management, and miscellaneous information for each company and group. For the companies and groups, the number of features is 231 for each. Thus, the matrix dimensions of our data are approximately 4.6 million x 462 variables before data cleaning.

4.1.1 Data Cleaning

The amount and quality of the data are generally good, however, due to data inconsistencies, we are required to remove a few observations. We remove observations that deviate on two different logical tests, one related to the balance sheet and one for the income statement. The primary source of error is due to rounding followed by accounting errors. The first test is to ensure that total assets equal total liabilities and equity. For large companies with total assets above 100 000 TNOK, we allow a deviation between the total assets and total liabilities and equity up to 5%. Observations with deviation above this are removed. For smaller firms, we allow deviance of 10%. The second test we perform is ensuring reported net income equals

calculated net income. For companies with a total income above 100 000 TNOK, we allow an income statement deviance of 5%, and 10% for smaller firms. The reason that we allow for larger absolute deviance for large firms compared to smaller firms is that some companies report in MNOK rather than TNOK. However, this deviance cannot be to high, since this would include firms with accounting errors.

Furthermore, we choose to remove observations before 1999, because at the end of the year in 1998 a new accounting standard and legislation were implemented. We have evaluated these two standards to be too different and therefore decided not to attempt to harmonize them.

We have information on a very diverse selection of legal structures, sectors, and entities. The legal form of incorporation is a leading indicator of the bankruptcy risk for a company. Therefore, it is unfortunate to include observations that are either improbable to go bankrupt or companies with a different incentive structure. These could be state-owned schools, governmental organizations or foundations without owners and operations. Therefore, we remove observations with legal forms that have unusual incentive structures that could skew our data. In appendix A5 we have listed all legal forms that are kept, and those who are removed.

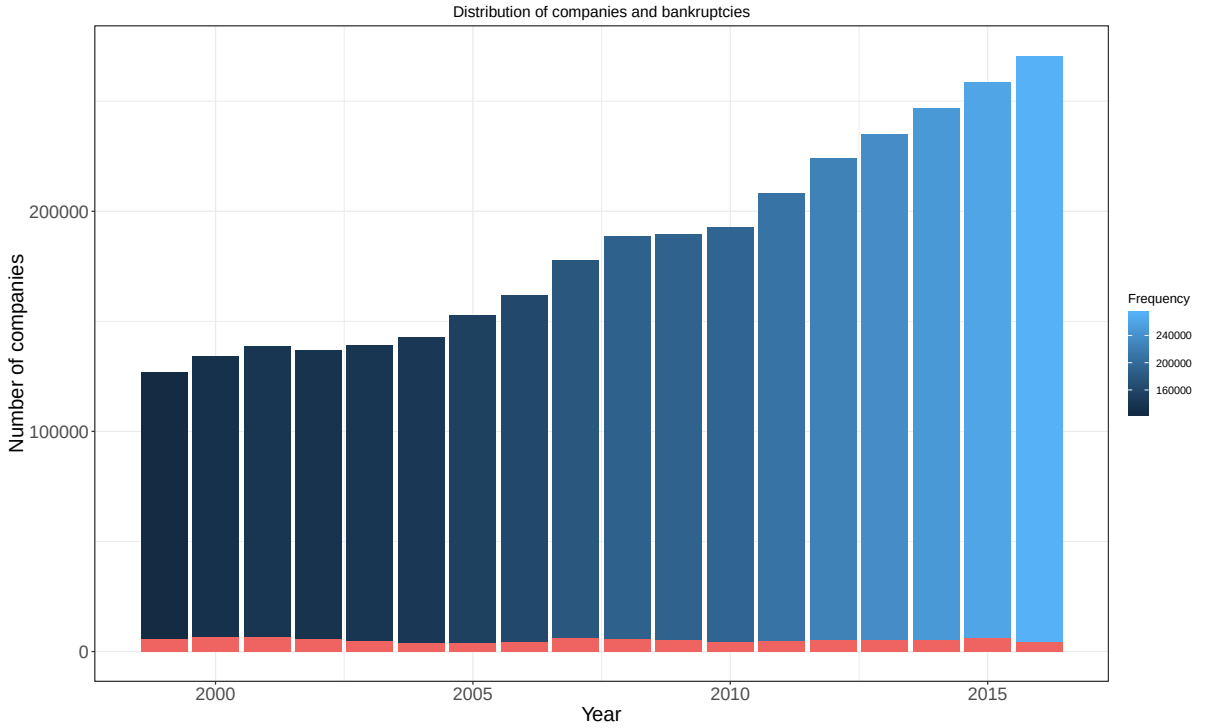

Figure 4.1: The final distribution of companies and bankruptcies after all cleaning steps. The red portion indicates the bankrupt portion.

As the previous literature indicates, when creating unfocused models, sectors with special traits

such as financial companies and insurance agents are often excluded. This is primarily because there is a concern that the model could add to much weight on special accounting in these sectors, and neglect the underlying patterns, which are more important. Therefore, by unifying these sectors it could lead to a significant decrease in overall performance. For our models, we remove these companies, based on the sector variable, introduced in section 4.1.3.

Thus, our finalized selection of company records is 3 327 405 across all years. In Figure 4.1 we have illustrated the frequency of bankrupt and non-bankrupt firms.

4.1.2 Response Variable

The rare event of bankruptcy is a fluid event and it is a matter of definition when a firm is officially bankrupt. There are several proxies that could work as the binary trigger, such as court date, registration date and finalization date. Our paper defines a company bankrupt on the date (i.e year) the company filed for bankruptcy, prior to the bankruptcy proceedings[28]. In the original data, the response variable is built around this trigger. However, we observe some noise in the responses. This is primarily due to other corporate actions, indicating that the original response quality is mixed and that the response values are slightly overstated. In addition, filing for bankruptcy is not a stable response variable since the bankruptcy could be solved, and overturned in court. For observations that either has missing or inconsistent values, they are unified for each unique company. Furthermore, we have patched our data with updated information for the last five years, by the courtesy of Bisnode. Unfortunately, due to legislation, older data related to bankruptcies are not easily available to researchers, since the legislation imposes clear constraints on how long the data can be stored. Hence, this indicates that the model is prone to suffer from more false positives in the early years compared to later years.

In the literature review in section 2, we encountered different solutions on how the panel data is treated. We believe that the superior alternative is a multi-year model, similar to the logic proposed by Berg (2007). The benefit of creating a multi-year model is the ability to utilize more data, giving our models a better basis for detecting signs of distress. The predictive power of the variables in a multi-year model is less likely to be affected by short-term bias that is present in the data. This could be fluctuating macroeconomic conditions and short term disturbances in the markets, which indicates that a one-year-model could predict worse out-of-time. A multi-year

[28] See Appendix A1 for details.

model is better suited for assigning less weight to these events.

For our application, we decide to lag the patched response variable three consecutive periods. We believe that this time-frame is optimal because there is a trade-off between early and late bankruptcy predictions. A too long time prior to the bankruptcy could train the model on healthy companies, which could introduce noise in the predictions. Lastly, given a time-frame of three years, it should be sufficient for stakeholders to act on this information, while less would make it more difficult to act on. Hence, if a firm filed for bankruptcy in year *n*, we would classify the firm as bankrupt for the years *n-2*, *n-1* and *n*, but for earlier years the response variable is non-bankrupt.

4.1.3 Variable Selection and Transformation

Previous research within the field seems to be obsessed with ratios. However, ratios are just a simplification of the financial statement. Therefore, we prefer that the algorithms themselves choose which variables are more important. A critical aspect of this approach is to find the right balance between details and computational power and time. One solution is to slightly reorganize the financial statements from the standard structure provided by *IFRS* and *NGAAP* to a structure similar to the analytical balance sheet suggested by Revsine et al. (2018), Petersen et al. (2017) and Koller et al. (2015). This is also called the *reclassified financial statement*. This allows us to separate the operations and financing of the company but still conserve sufficient detail. For our purpose, we do not perform the whole transformation, but rather stop after creating the division between financial and operating assets. See Table A4.1 in appendix A4 for the financial structure of the companies.

Throughout this paper, we have made no measure to adjust nor improve the accounting standards. We simply attempt to harmonize the standards, forcing them into the same framework. This means we have not changed the accounting of our *IFRS* observations, but rather forced them into the same framework as *NGAAP*. The overwhelmingly large majority of our observations use *NGAAP*, therefore we conclude this not to be problematic.

As a consequence of performing tests on the financial, management and sector statements rather than ratios, we are faced with a problem related to correlation. The financial statements are built on double entry, and logic flows from the income statement to the different accounts in

the balance sheet. Therefore, we are forced to remove variables that introduce a significant correlation. After considerable evaluation, we set a high threshold of *+/- 0.9*, removing those with a higher correlation than this threshold.

We also remove variables that we deem to have lower predictive power on bankruptcy, such as inter-company information or highly detailed information about management, the board of directors and excess information about the auditor/accountant. Furthermore, we have calculated the respective cash flows from operation, investing and financing, by the logic proposed by Langli (2010). Note that we are missing a few accounting records, such as changes in currency reserves, hence there is some room for error in the cash flows. In addition, we choose to include the D&B rating scale for the firms, since the rating would decrease the problem of response domination, yet not limit the model, since less than half of the population are rated.

In the original data, we encountered two different standards for the sector codes. In 2002 and 2007, there were reclassifications of industry sectors conducted by the Norwegian Government, such that old standards were renewed to reflect the developed market better. Therefore, we have two different standards of industry sector codes included in our dataset[29]. To decode this problem, our solution is to compare the link between the different standards, which can be found on the home page of Statistics Norway[30], and develop a new list of 17 common industry sectors that, in our view, reflects the different types of sectors in the most appropriate way. The details of this process are presented in appendix A6.

For the financial statements on the parent, we encountered a dilemma, because only a small proportion of the companies also have a parent. By introducing this data for all companies this would introduce a lot of *NA's*. On the other hand, we believe that some parent variables could have predictive power. Hence, our solution is to include a few key variables from the parent that we deem to be important. For observations without a group, we replace the appropriate *NA* with a zero.

Since we are working with panel data, we introduce a simplification that allows us to treat it as cross-sectional data, without regarding the time difference. This was discussed in the previous section 4.1.2. However, to still capture spill-over effects across years, we choose to lag ten variables. These variables are selected by the algorithms themselves and are summarized in

[29]The data before 2002 is replaced with 2002 standards where data was not missing.
[30]https://www.ssb.no/virksomheter-foretak-og-regnskap/naeringsstandard-og-naeringskoder.

appendix A3. The variables are lagged[31] for two consecutive years, *t-1* and *t-2*, calculated as the percent change from each year, as previous studies have done.

Another problem we encounter is occasional missing data for a number of variables, where data is present in the previous year(s) or later year(s). We replace the missing data with either lagged or forecasted values for the variables: *year of bankruptcy*, *region*, *legal form*, *incorporation category* and *structure of ownership*.

Next, we transformed the data into appropriate variables such as dummies and ordinal variables, see appendix A7 for our variable treatment. For continuous variables, we normalized them to a mean of zero with the intention of removing the actual distance between the number, but not the distribution. This is called standardizing the variables, and is performed for each observation *i* on each variable *j*, by the following equation:

$$z_{i,j} = \frac{x_{i,j} - \bar{x}_j}{\sigma_j} \tag{4.1}$$

where $\bar{x}_j$ represents the average for variable *j* and σ_j is the standard deviation for variable *j*.

Based on these assumptions and cleaning steps, our finalized variable count is 159, giving us a finalized matrix of 3 327 405 x 159.

4.2 Hyperparameter Tuning

As discussed in section 3, more flexible methods require hyperparameter tuning. For our models, this regards KNN, RF, SVM and NN.

For KNN, the objective is to find the *K* that gives the best training predictions. Due to the large amount of data we process, we are required to perform tuning on a smaller sample. This has the obvious implication of changing the number of observations and possible neighbors. Regardless, the computational power available to us is insufficient to solve this in an appropriate time-frame. The tuning parameter is selected using a two-fold CV and a sample of 250 000 observations. The selected attributes are chosen to make the computational time reasonable. As we observe from Figure 4.2, the optimal *K* is 51.

[31] For observations that start with a value of zero, thus introducing an infinitely large increase, the observations are replaced with a large finite number to offset the infinity. This number is chosen based on other numeric responses and is set to 1000, approximately at the break-point for the third quartile, on average.

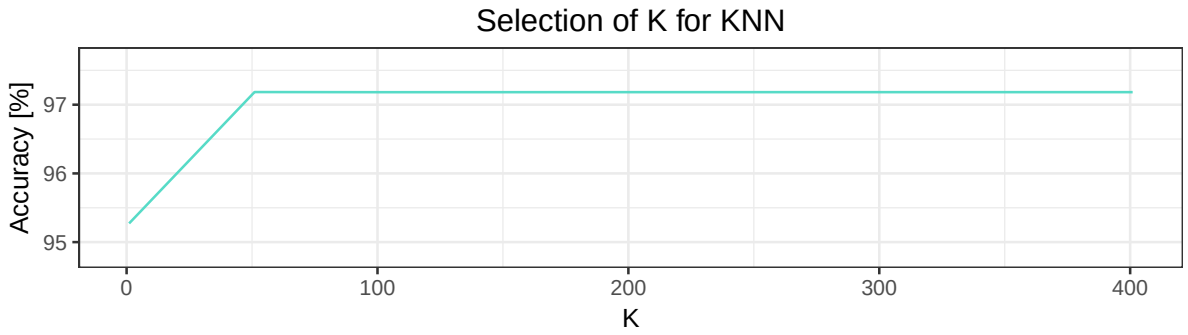

Figure 4.2: Different accuracy levels for different K's. Due to the low number of runs the accuracy seems more stable than it actually is.

Regarding Random Forest, the hyperparameter that requires consideration is n, the number of trees that are fitted for each fold. Given that the available features are $\sqrt{p}$ and minimum node size n_{min} is set to 1, this imposes computational constraints on n. This is because an increase in n causes the computational time to grow exponentially. From Figure 4.3 we conclude that when n is above 30, most of the variance is exhausted. Therefore, we select 30 as our n. Based on these three aforementioned properties, we calculate the appropriate size of b based on computational time. We find that an b of 23 000 is the optimal trade-off. For $b \epsilon B$ this creates 115 subsamples, while for $f \epsilon F$ this creates 575 subsamples containing 17 250 trees.

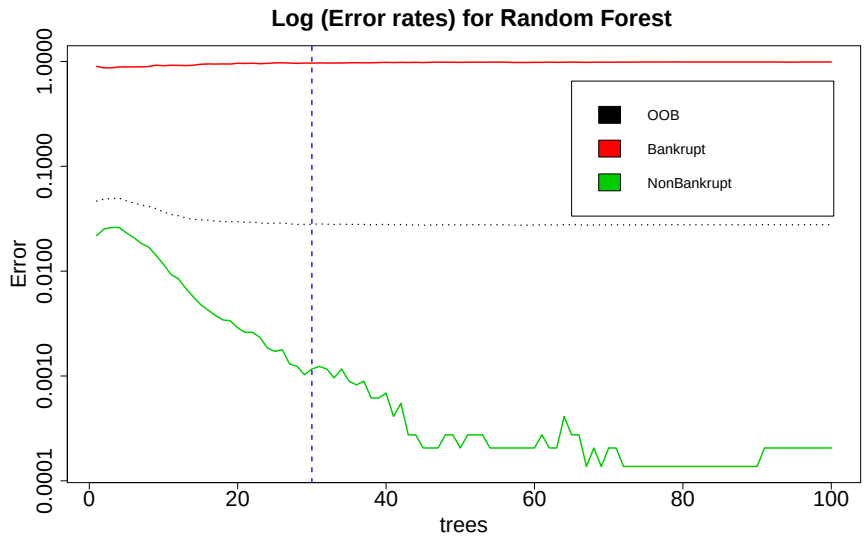

Figure 4.3: Illustration of the error rates on a stratified sample based on 15 000 observations. The black line represents the logged out-of-bag error (OOB), which is equivalent to training error. The green and red line display logged class error and the dotted blue represents the selected n.

For SVM the tuning of the hyperparameters is very convenient because it happens simultaneously as the model is fitted. The available tuning values are given by an adaptive grid where the grid is scaled to each chunk and the algorithm selects the optimal parameters from this adaptive grid.

The unscaled grid is shown below in Table 4.1. The scaled parameters are visualized in appendix A9.

Adaptive grid of hyperparameter tuning			
	Gamma (γ)	Lambda (λ)	Cost (c)
Maximum value	0.01	5	0.1
Minimum value	0.2	0.001	100
Steps	10	10	10

Table 4.1: The unscaled adaptive grid for SVM Hyperparameters. The scaled values are displayed in appendix A9.

For the Neural Network, there are several hyperparameters that require tuning, that is; Number of hidden nodes (Z_m) and hidden layers (l), learning rate (γ_r) and momentum (m). In addition, there are other parameters such as the number of rounds and array batch size (batch learning) we need to be aware of. As with KNN, due to our large dataset, we are required to perform the tuning on a smaller sample. The tuning parameters are selected with extensive testing, using a stratified sample of 100 000 observations. The number of rounds and the array batch size is set to 20 and 50, respectively, based on the past literature and common practice with NN fit on large data.

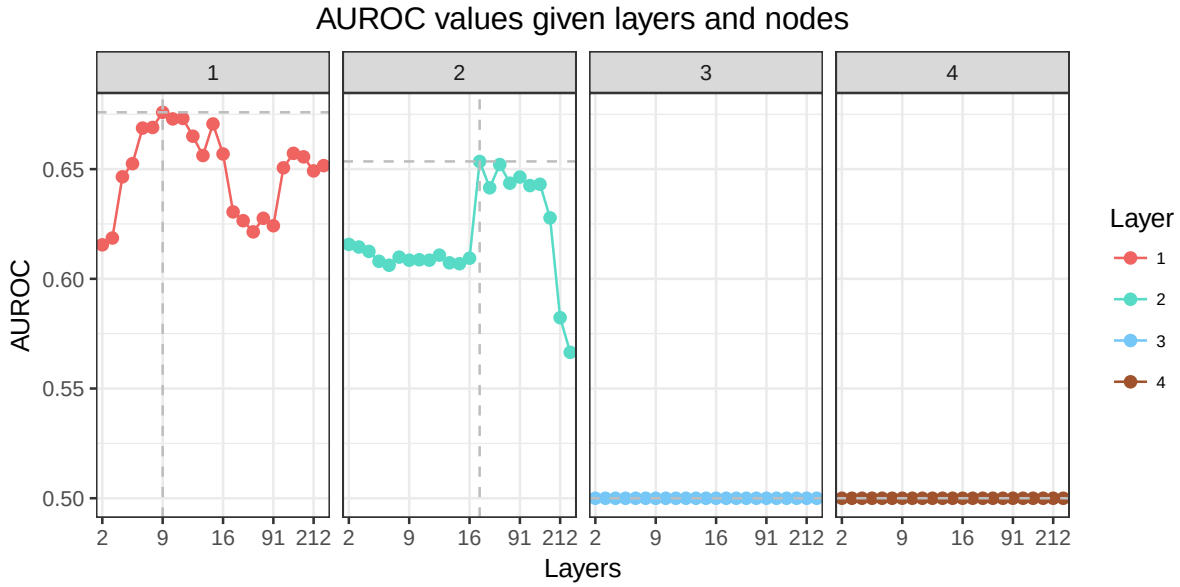

Figure 4.4: Different AUROC for different numbers of hidden nodes and layers.

Based on the methods introduced in section 3.2.6, we perform a number of tests to find i) number of hidden nodes and ii) the number of hidden layers. In these tests, we use both methods from Equation 3.25 and a number of different values of α from the right equation in 3.25 to find the

optimal choice of parameters. Figure 4.4 shows that AUROC is highest for 1 hidden layer and 9 hidden nodes, hence we use those parameters in our NN model. We do not test more than 4 layers, because the AUROC is stabilized at 0.5 after 2 layers.

Having determined the number of hidden nodes and layers, we tune for γ_r and *m*, and find that the optimal choice in our case is a γ_r value of 0.3 and *m* value of 0.9. The entire technical process of hyperparameter tuning is explained in appendix A10, where the overall accuracy, AUROC and TPR are recorded for the hyperparameters in the tuning process.

4.3 Randomization and Reproduction

To ensure reproducibility and easy comparison across different techniques we utilize the same random number stream for all models. The seed is selected to be ”1”. This ensures that folds are equal, with the same observations in each fold for all models. For computational heavy techniques, such as KNN and SVM, the composition of the folds is slightly different, but based on the same number stream.

5 Results

This section is divided into three major parts. For the first part, we present results for each model at a basecase threshold of 10%, before comparing their performance against each other. For the second part, we take the best fold presented in the first part and optimize the threshold. In the last part, we perform an analysis of the importance of the variables on our best models.

5.1 Basecase Threshold

In this subsection, we present our results using the basecase threshold of 10%. This empirically tested threshold helps us to decide which fold to optimize and enables us to compare the models at the same threshold. The threshold implies that given a predicted probability for an observation, the observation is classified as bankrupt if $p \geq 0.1$.

5.1.1 Generalized Linear Model

GLM is created with the *stats* package from the R Development Core Team (2008), using the *glm* function. The results are displayed in Figure 5.1, 5.2 and Table 5.1. Five folds are used in the cross-validation. The stacked barplot is a visualization of the confusion matrix, where each facet is representing a fold, consisting of two true rates (TPR and TNR) in green, and two false rates (FNR and FPR) in red.

As we observe from the figures, the performance of the model is weak, where we are able to predict only 16% of the bankrupt firms correct for the best model, fold 2. Fold 3 and fold 5 performs worst, yielding no correct classification of bankrupt firms. We also observe the fold with the highest AUROC and best classification of bankrupt firms give the lowest overall accuracy. This is primarily due to the trade-off between classifying bankrupt versus non-bankrupt firms correctly. From Figure 5.2 and Table 5.1 we observe that the AUROC for fold 2 is the highest at 0.5435, while fold 3 and 5 have values of ~ 0.5, implying that the fitted models only predict the over-represented category, i.e. predicts all observations as non-bankrupt.

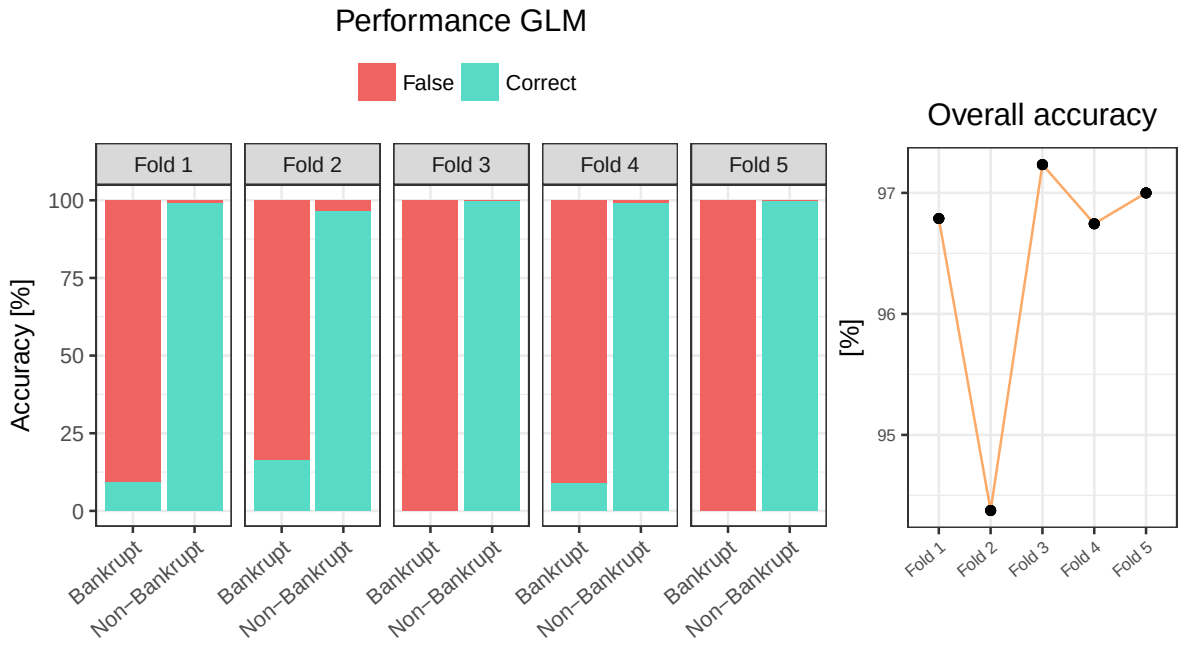

Figure 5.1: Performance of GLM.

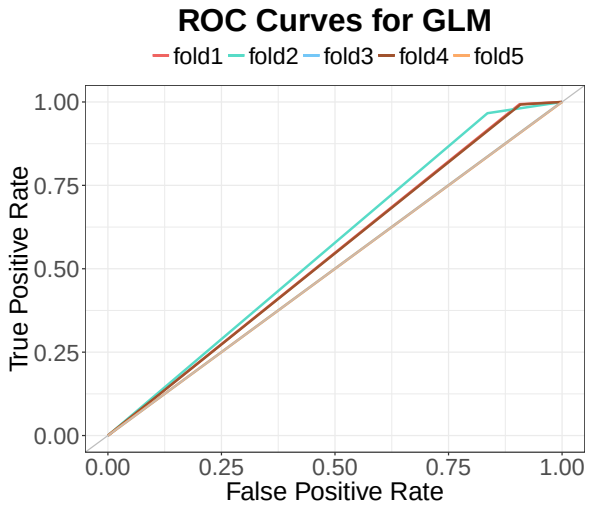

Figure 5.2: ROC curves GLM.

Fold	AUROC
Fold 1	0.5435
Fold 2	0.5565
Fold 3	0.5003
Fold 4	0.5420
Fold 5	0.5

Table 5.1: AUROC GLM.

5.1.2 Generalized Additive Model

GAM is cross-validated using five folds on the finalized dataset. We make use of the function *gam* from Hastie (2018) available in R, first presented in Hastie and Tibshirani (1990). Results of these models are presented in Figure 5.3, 5.4 and Table 5.2.

GAM, like GLM, yields rather weak predictions, where only fold 3 and 5 classifies some observations correctly in the bankrupt class (TPR), while fold 1, 2 and 4 have close to none correct predictions in the bankrupt class. As we also observe from Figure 5.4 and Table 5.2, the

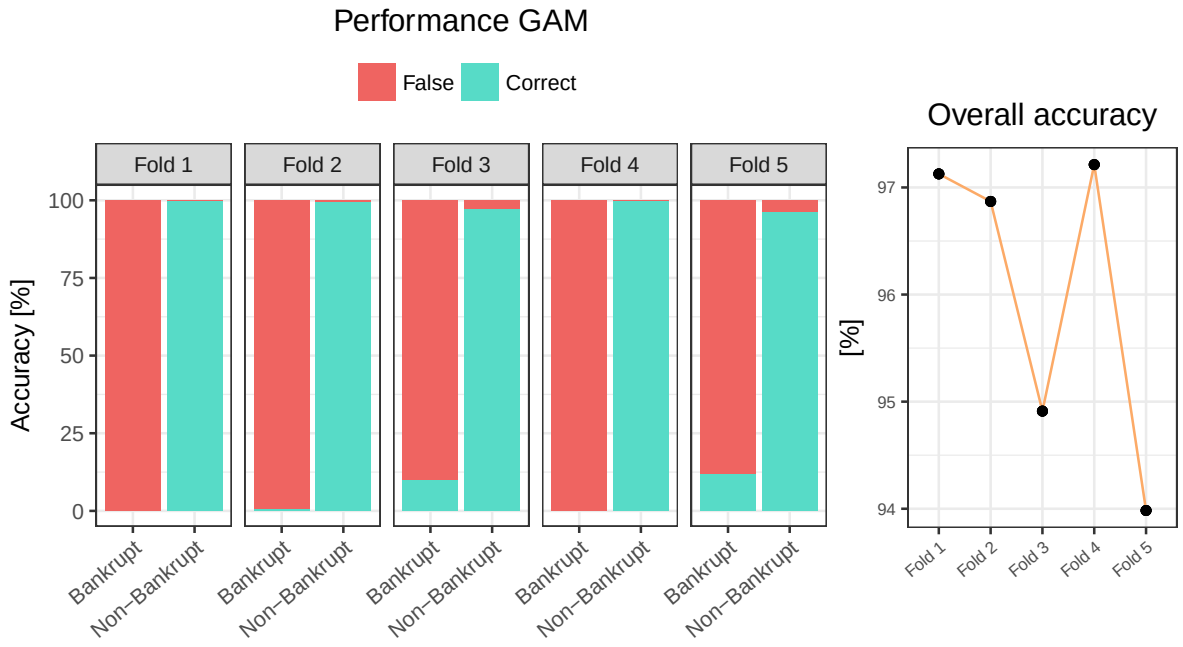

Figure 5.3: Performance of GAM.

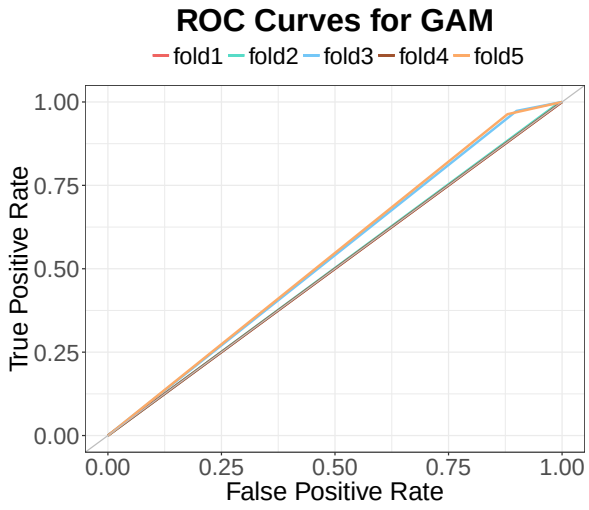

Figure 5.4: ROC curves GAM.

Fold	AUROC
Fold 1	0.5001
Fold 2	0.5023
Fold 3	0.5367
Fold 4	0.5
Fold 5	0.5417

Table 5.2: AUROC GAM.

ROC curves and AUROC values are respectively low, with ~ 0.5 for fold 1, 2 and 4. Fold 5 performs best for GAM, with an AUROC of 0.5417. In this fold, the model correctly classifies 12% of the bankrupt firms, and have an overall accuracy of 93.98%. The overall accuracy is quite good, but, as already mentioned, this is not a suitable indicator for our data. Fold 1 and 4 are good examples of how a model is able to attain high accuracy, simply by neglecting those predictions that matter to us; the bankruptcy predictions. GAM is inferior to GLM at the basecase threshold.

5.1.3 Linear Discriminant Analysis

Our LDA model was created with a five-fold CV using the model by Venables and Ripley (2002), available in R.

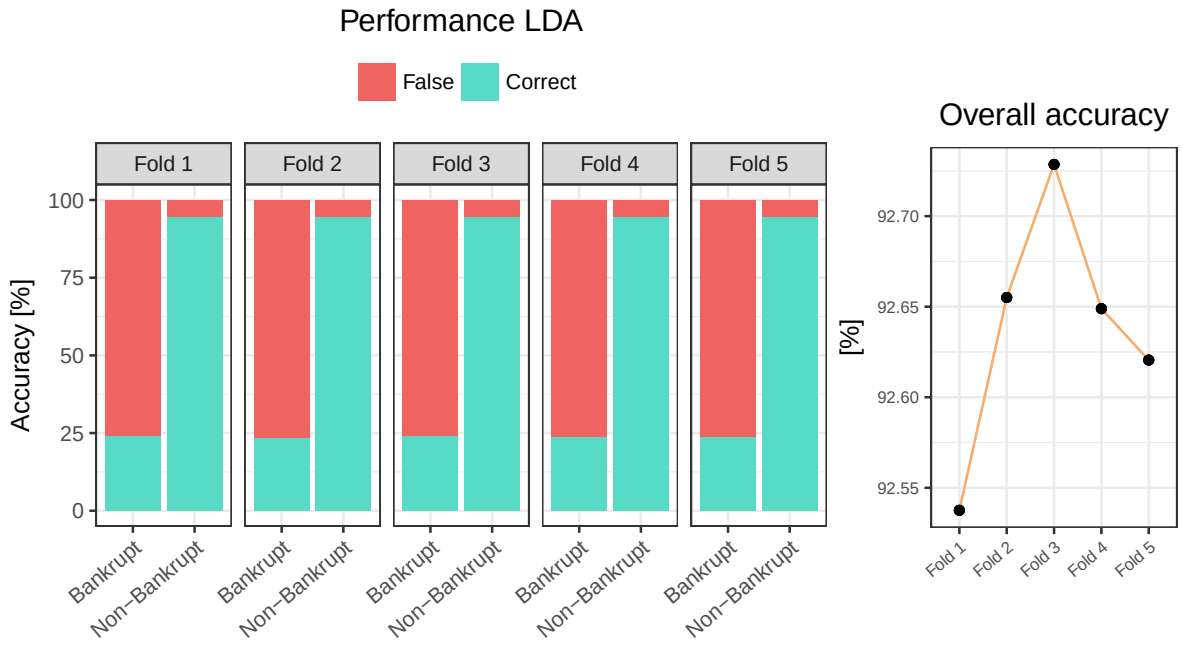

Figure 5.5: Performance of LDA.

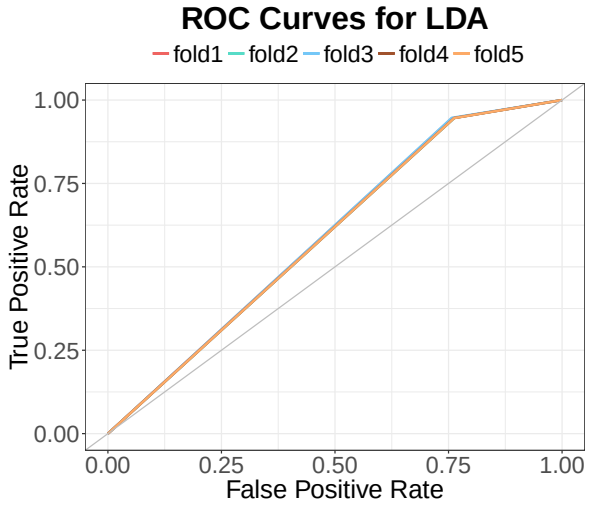

Figure 5.6: ROC curves LDA.

Fold	AUROC
Fold 1	0.5940
Fold 2	0.5917
Fold 3	0.5948
Fold 4	0.5919
Fold 5	0.5918

Table 5.3: AUROC LDA.

The results are shown in Figure 5.5, 5.6 and Table 5.3. LDA performs intermediate in prediction of bankruptcies, at the cost of higher false classifications of non-bankrupt firms compared to GLM and GAM. Fold 3 yields the highest achieved AUROC value of 0.5948. On the other hand, all folds yield approximately the same AUROC, indicating a robustness of the model. For

a large majority of the probabilities LDA produces, the observations cluster around the same probabilities, thus, the chosen threshold is important to optimize the results. Regardless, LDA is superior to both GLM and GAM in the basecase.

5.1.4 Quadratic Discriminant Analysis

For the QDA model, we utilized the R package by Venables and Ripley (2002). The quadratic function of QDA causes it to be averse of highly correlated features, but due to the nature of our data, it is evident, and required, that some of the features of our data correlate. Therefore, we hold out a few features that are highly correlated, that otherwise would make QDA infeasible. The features removed are primarily aggregates of financial accounts such as total liabilities. We are required to remove a total of four features. Thus, the model is not entirely comparable to the other models. The model is cross-validated using five folds.

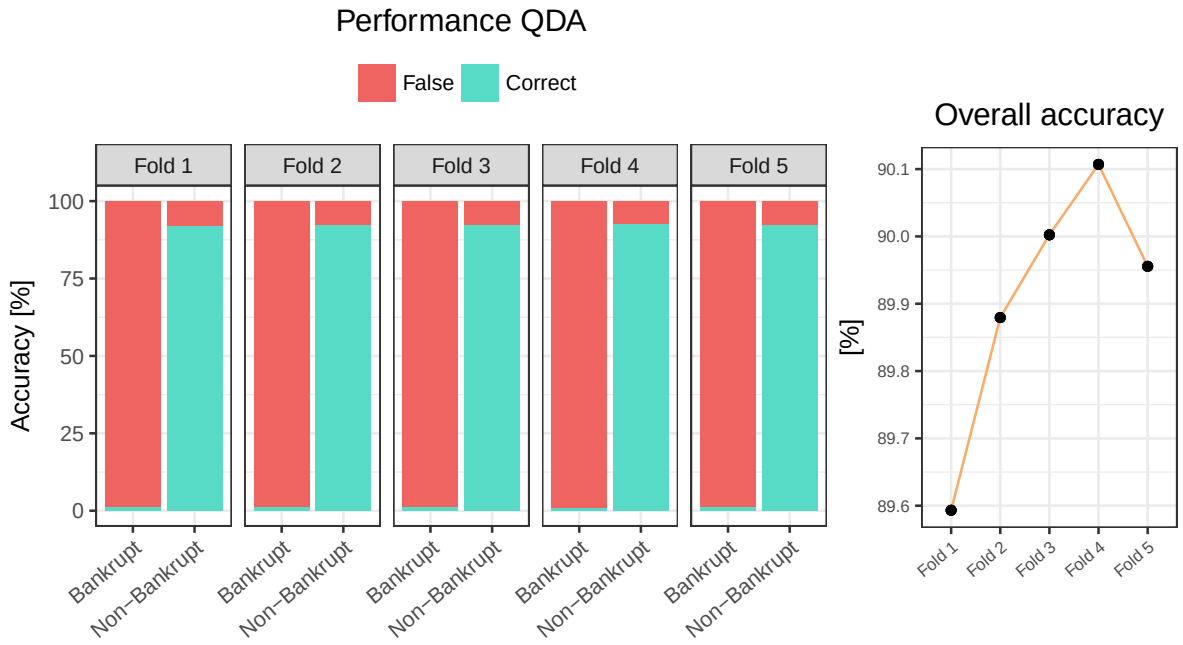

Figure 5.7: Performance of QDA.

From Figure 5.7, 5.8 and Table 5.4, we observe that QDA performs weakly, with all folds resulting in AUROC values below 0.5. The highest AUROC is achieved for fold 4. The probabilities of QDA is clustered around very small probabilities, indicating that the basecase threshold of 10% is too high. QDA is inferior compared to GLM, GAM and LDA in the basecase.

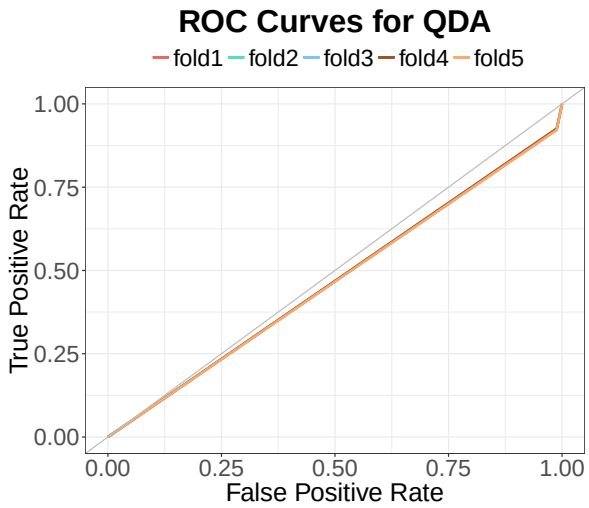

Figure 5.8: ROC curves QDA.

Fold	AUROC
Fold 1	0.4669
Fold 2	0.4684
Fold 3	0.4686
Fold 4	0.4689
Fold 5	0.4689

Table 5.4: AUROC QDA.

5.1.5 Mixture Discriminant Analysis

We utilize the MDA package by Hastie et al. (2017) to train and test the model. We create Gaussian probabilities for each class and the model is cross-validation, using five folds. In the results displayed in Figure 5.9, 5.10 and Table 5.5 we observe MDA yield consistent predictions across folds.

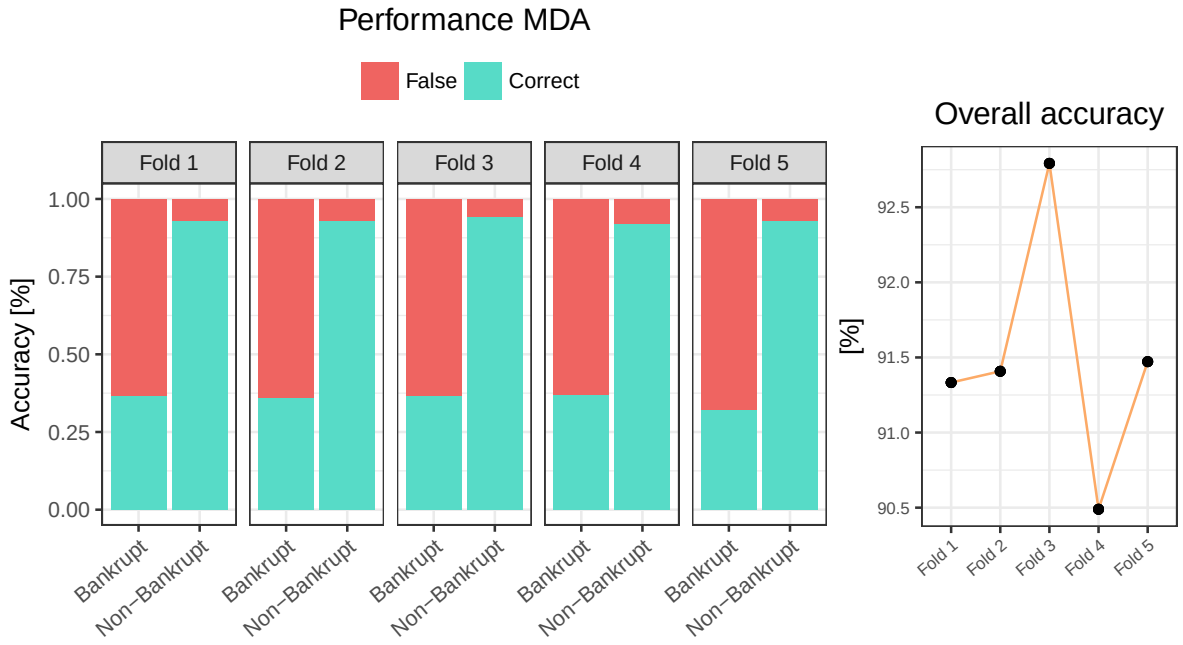

Figure 5.9: Performance of MDA.

The results indicate that MDA outperforms all previous models w.r.t AUROC, given the threshold. From Table 5.5 we conclude fold 3 is superior compared to the other folds in terms of the AUROC.

Figure 5.10: ROC curves MDA.

Fold	AUROC
Fold 1	0.6481
Fold 2	0.6454
Fold 3	0.6554
Fold 4	0.6460
Fold 5	0.6266

Table 5.5: AUROC MDA.

Fold 3 yields a TPR of 36.7% and an AUROC of 0.6554. Furthermore, this fold also produces the best overall accuracy of 92.8%, which is higher than both QDA and LDA. Furthermore, we observe from Figure 5.9 that the correct classification of bankrupt firms is stable at 34% $\sim$ 37% for all folds. Due to this stable classification, it indicates the model is able to consistently predict on different data, thus, the underlying selected features is important across time and data variation. In the basecase, MDA outperform all the other models we have presented so far.

5.1.6 K-Nearest Neighbors

As outlined in the previous chapter we treat KNN as a special case because of the computational time this method would require for a full-scale model. For our implementation, we have followed the logic and framework built by Cacciatore et al. (2018) where each point is fitted on a plane with the 51 nearest neighbors. PLS is used to reduce the dimensionality of the data.

The data consist of a stratified sample based on 1/6th of the original data, with the same features as the previous models. As a measure to reduce the computational time further, KNN is set to only produce label predictions, i.e. bankrupt or non-bankrupt. This means that KNN does not classify according to the 10% threshold and hence, the method will not be optimized in the next section. The results from KNN is presented in Figure 5.11, 5.12 and Table 5.6.

From Figure 5.11 we observe we are able to obtain very high accuracies, with fold 1 yielding the highest accuracy of 97.23%. This accuracy is produced at the high cost of a low TPR, i.e., KNN is struggling to classify the bankrupt firms correctly. A possible reason for this is that the K is

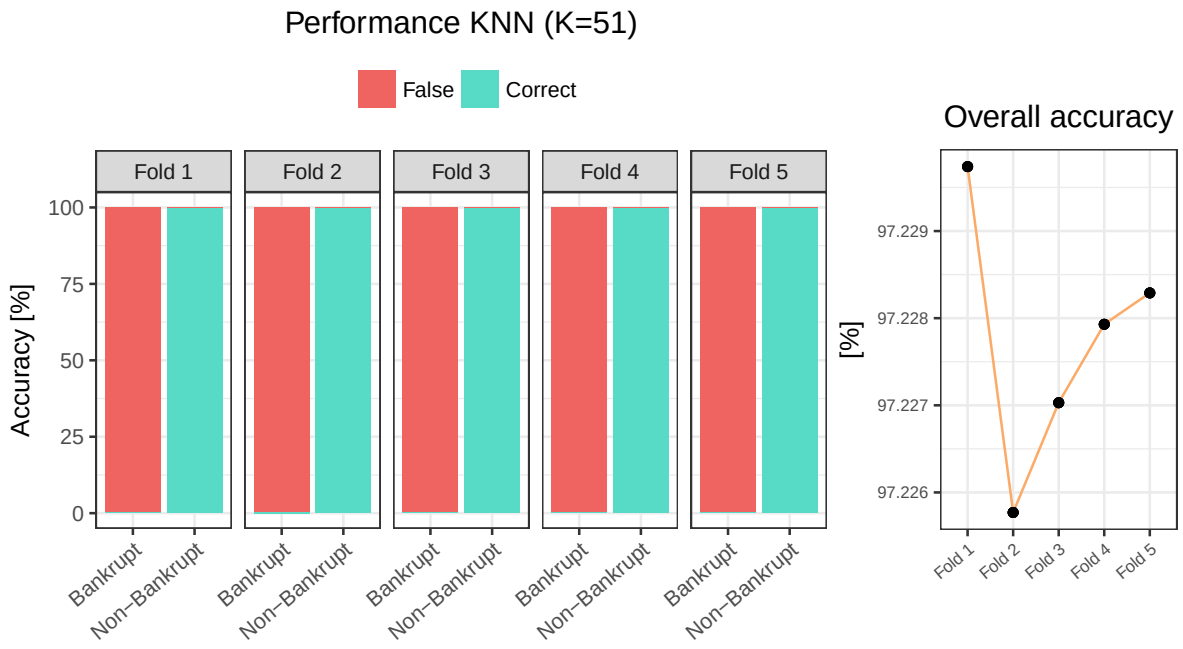

Figure 5.11: Performance of KNN with K=51.

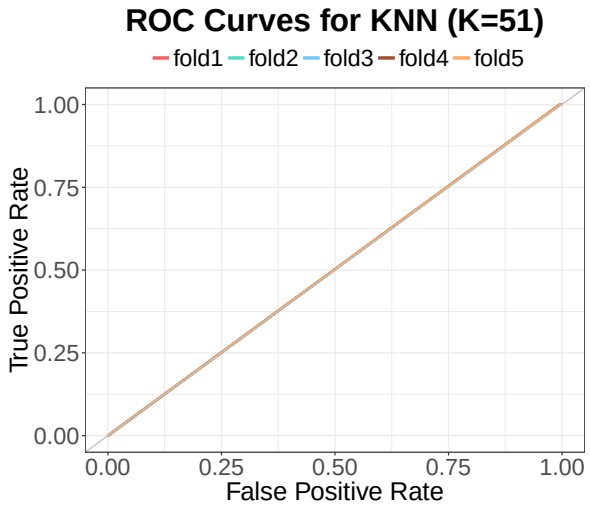

Figure 5.12: ROC curves KNN.

Fold	AUROC
Fold 1	0.5020
Fold 2	0.5020
Fold 3	0.5021
Fold 4	0.5020
Fold 5	0.5019

Table 5.6: AUROC KNN.

too low and that the data points cluster around each other. A possible solution is to make KNN more aggressive, by increasing the K above what our tuning indicates, which causes a larger radius of neighbors to be classified as bankrupt. This happens at the cost of overall accuracy but can improve TPR. Fold 1 produces the highest TPR at 0.513%, while the TNR is close to 100%. However, the AUROC values for all folds are at approximately 0.5. In terms of AUROC, KNN is inferior to all other models except QDA.

5.1.7 Random Forest

For RF, we have utilized the algorithm described in section 3.2, while parts of the R implementation is solved using the R package by Liaw and Wiener (2002), influenced by Breiman (2001).

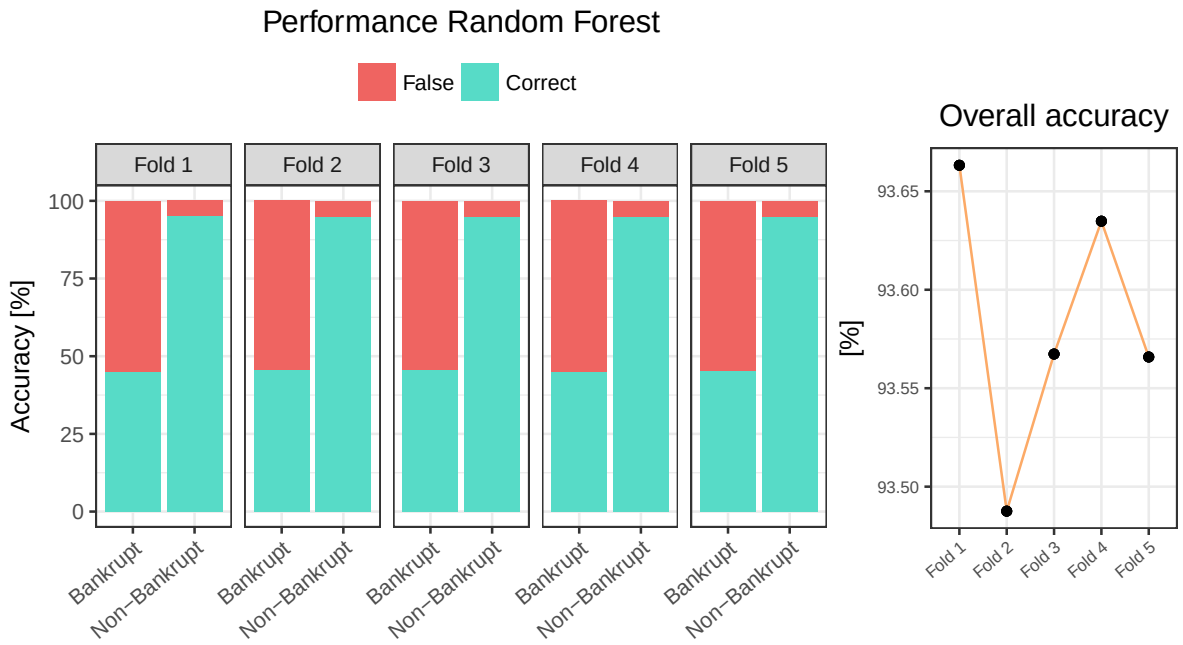

Figure 5.13: Performance of Random Forest with 17 250 trees.

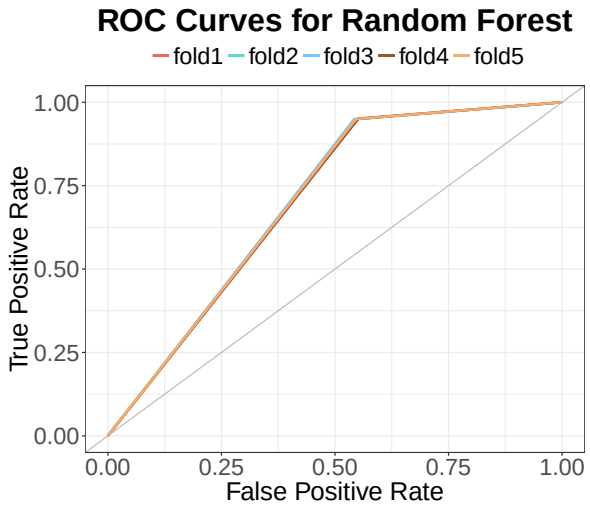

Figure 5.14: ROC curves Random Forest.

Fold	AUROC
Fold 1	0.7005
Fold 2	0.7021
Fold 3	0.7035
Fold 4	0.6995
Fold 5	0.7015

Table 5.7: AUROC RF.

From Figure 5.13, 5.14 and Table 5.7, we see that RF produce very good and consistent predictions. The forests, or folds, are able to predict very well for the TPR without introducing a significantly large cost for the TNR. The best predictions in terms of the AUROC is for fold 3

with a value of 0.7035, but we observe all forests yield consistent values. The overall accuracy is very stable for all forests as well, with a range of 93.50 % $\sim$ 93.65 %. Fold 1 yield the best accuracy, while the fold with highest TPR, fold 3, yields an intermediate accuracy. The excellent performance and consistency across the forests indicate that the bias-variance trade-off has been appropriately addressed. However, this could be expected due to the large forests and decorrelated trees. We conclude that RF is superior compared to MDA at this threshold, and all the other models presented so far.

5.1.8 Support Vector Machines

As with KNN, SVM is also computational demanding and expensive. Due to the vast amount of data that should be fitted and processed, we have to be creative. Therefore, we have utilized the framework proposed by Christmann and Steinwart (2008), Farooq and Steinwart (2017) and the R package created by Steinwart and Thomann (2017). Rather than fitting all observations on the same hyperplane, the data is divided into chunks. For our implementation, the cell size is equal to 10x10 and yields approximately 1770 cells for each fold. As mentioned in section 3.2.5 the optimal hyperparameters are determined on an as-you-go basis.

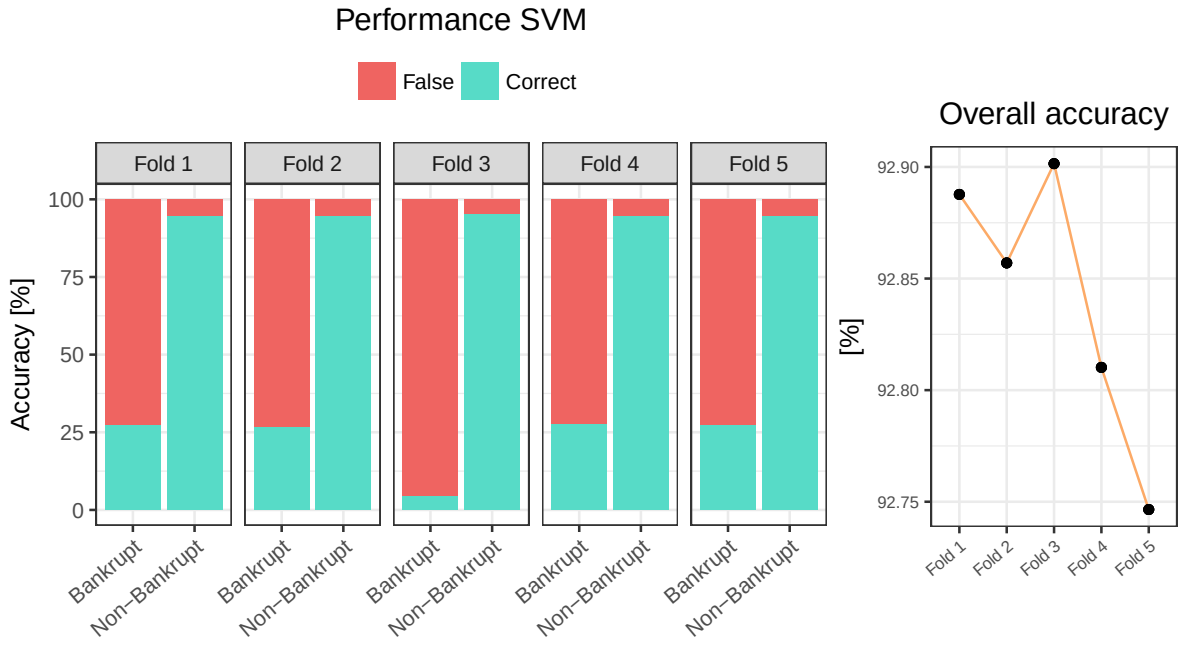

Figure 5.15: Performance of SVM with an adaptive hyperparameter grid.

The results are presented in Figure 5.15, 5.16 and Table 5.8. SVM produces intermediate

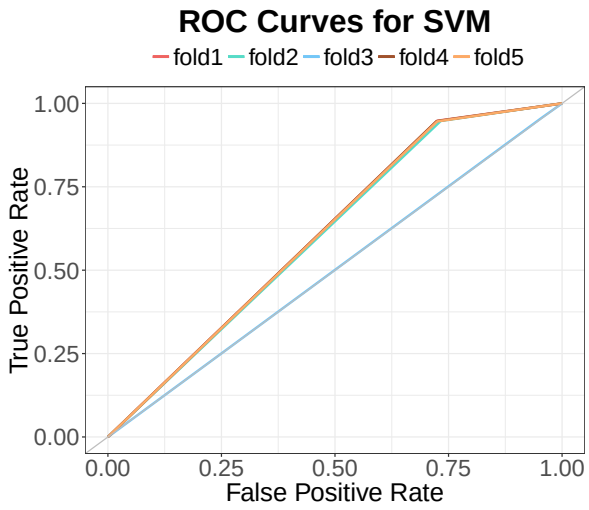

Figure 5.16: ROC curves SVM.

Fold	AUROC
Fold 1	0.6114
Fold 2	0.6072
Fold 3	0.5008
Fold 4	0.6116
Fold 5	0.6105

Table 5.8: AUROC SVM.

predictions, where fold 4 is the superior fold yielding an AUROC of 0.6116. Fold 1, 2 and 5 produces similar AUROC values, while fold 3 yields the worst AUROC. The overall accuracy yields an average of approximately 92.8%. Fold 4 yields a TPR of 27.63% and a TNR of 94.68%. SVM is inferior to MDA and RF, but superior compared with LDA, GLM, GAM, QDA and KNN.

5.1.9 Neural Network

The method of NN is not as straightforward as some of the other methods, as described in section 3.2.6. Therefore, when fitting the model, we utilize the logic proposed by Chen et al. (2017), making use of the convenient wrapper *mx.mlp* in their package. The results are presented in Figure 5.17, 5.18 and Table 5.9.

We observe that NN performs very well compared to most of the other methods, with the best fold, fold 3, having an AUROC of 0.7088, a TPR of approximately 50% and overall accuracy of 91.7%, which is very impressive results. However, the folds present some variation in the results, with a range in the AUROC from approximately 62% to 70%, where fold 1 and 5 performs worst and fold 3 best. Regarding the TPR, the measure range from 26% in the worst performing fold to approximately 50% in the best fold. The variation in TPR across folds could indicate that the threshold is not optimal, or worse, that the model contains too much variation, compared to bias. However, the performance is very good, and it is superior to all other techniques at this threshold.

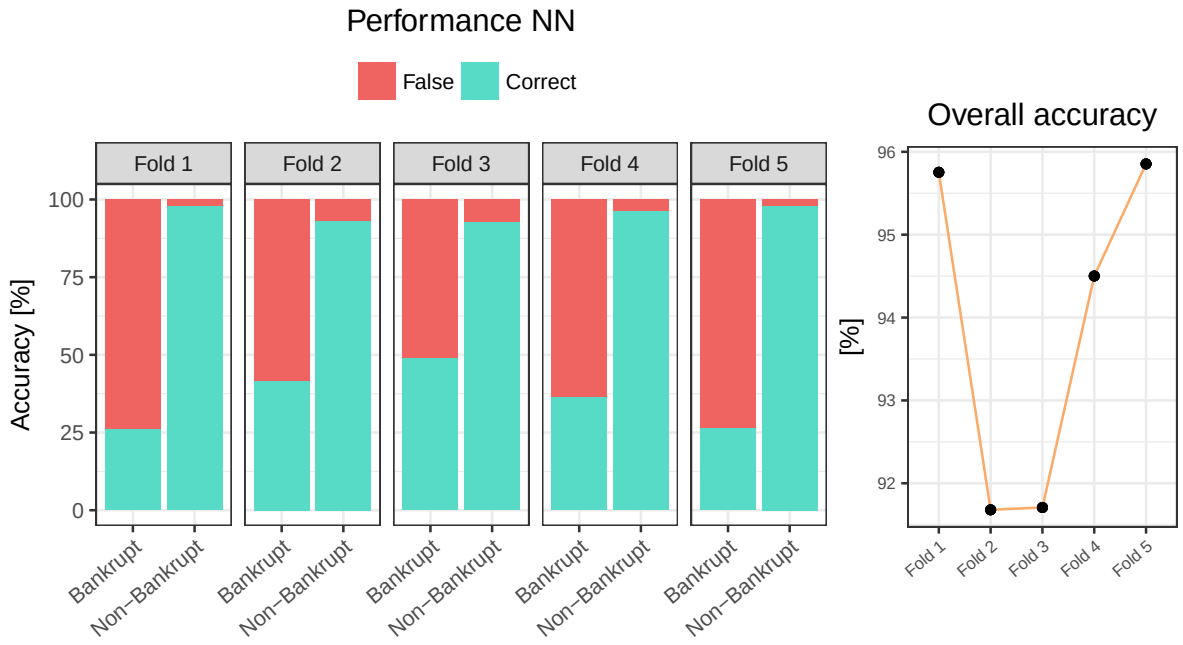

Figure 5.17: Performance of NN.

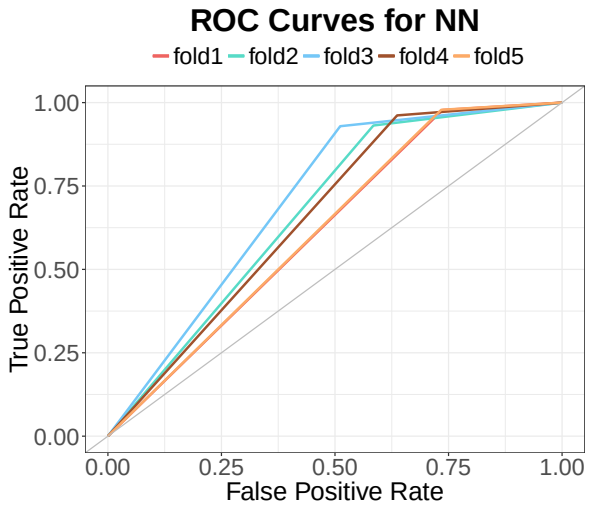

Figure 5.18: ROC curves NN.

Fold	AUROC
Fold 1	0.6204
Fold 2	0.6734
Fold 3	0.7088
Fold 4	0.6624
Fold 5	0.6222

Table 5.9: AUROC NN.

5.1.10 Comparison

Here we present a comparison of all the models, where each model is represented with the best performing fold from the basecase.

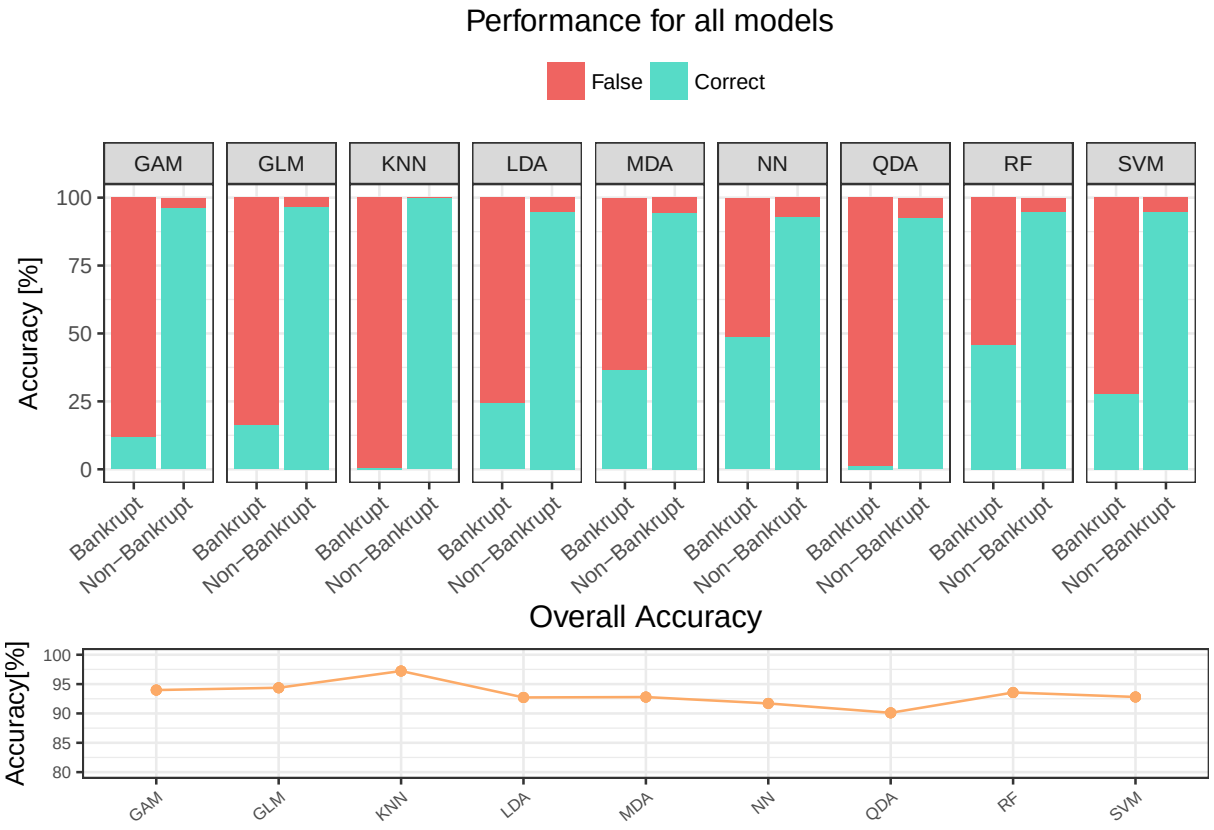

Figure 5.19: Comparison of all models.

Observing Figure 5.19, we conclude all our models performs well in terms of accuracy, where the prediction is stable well above 90%. However, as we outlined in section 3.1.1 it makes more sense to optimize the AUROC at the cost of a lower accuracy. The AUROC values presented in Table 5.10 shows that the best models are Neural Network and Random Forest, with an AUROC of 0.7088 and slightly worse of 0.7035, respectively.

Model	GAM	GLM	KNN	LDA	MDA	**NN**	QDA	RF	SVM
AUROC	0.5417	0.5565	0.5021	0.5948	0.6554	**0.7088**	0.4689	0.7035	0.6116

Table 5.10: AUROC for all models - basecase.

As indicated throughout this chapter, the consistency of the results is important. Based on consistency and the TPR, Random Forest is the superior technique compared to other techniques.

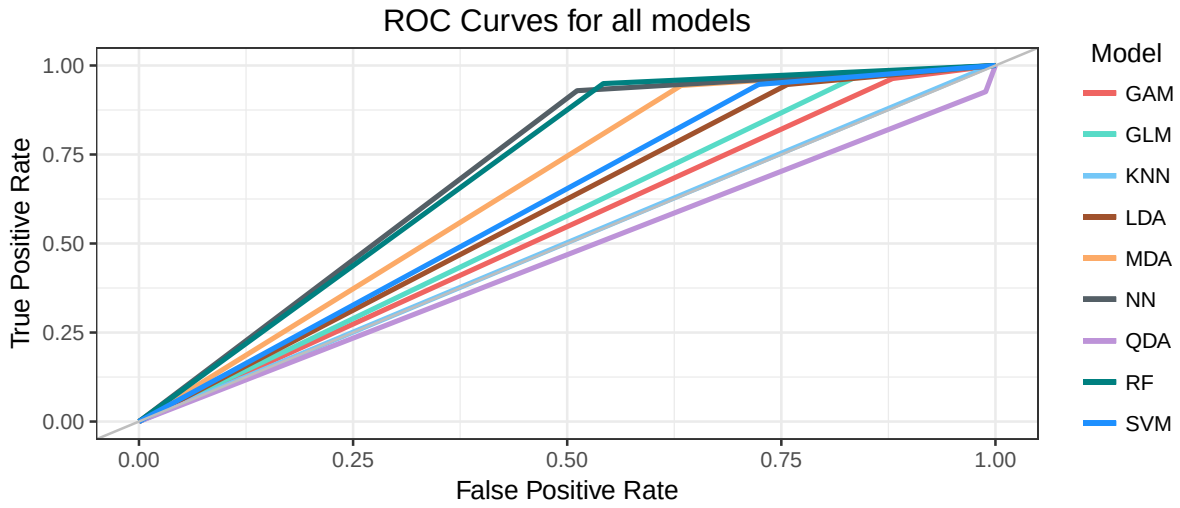

Figure 5.20: ROC curves for all models.

The model that yields the highest overall accuracy is also the model that yields the lowest TPR, which is KNN. For the TNR all models perform fairly well. Figure 5.20 display the ROC curves of the best fold for each model.

5.2 Optimized Threshold

In this section, we optimize the best fold from each technique, presented in the previous subsection, with the intention of improving the AUROC, by tuning the threshold imposed on the probability.

5.2.1 Optimized Threshold Values

Initially, we ran the models with an empirically tested threshold value of 0.1, meaning that predicted probabilities higher than 0.1 are classified as bankrupt. This value results in an overall very high accuracy, but the correction is insufficient to capture the full effect of the class imbalance. In line with our expectations, as results show in section 5.1, the TPR is relatively low compared to very high TNR's. To deal with this, we optimize the TPR and TNR using the AUROC, as explained in section 3.1.5. We repeat this process for the best fold for each model from section 5.1 and compare the new optimized results. Figure 5.21 display the optimization process for the best NN fold (fold 3).

We observe from Figure 5.21 that the optimal threshold value for NN is very low, 0.0309 to be

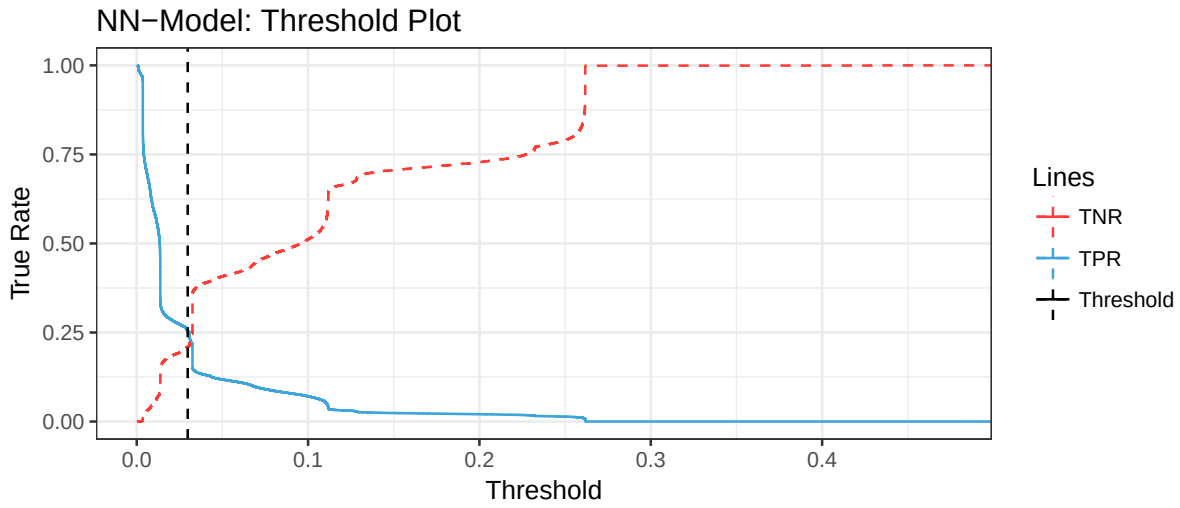

Figure 5.21: Threshold trade-off figure for the best NN model.

Model	GAM	GLM	KNN	LDA	MDA	NN	QDA	RF	SVM
Threshold	0.0031	0.0615	*NA*	0.0216	0.0324	0.0309	7.5e-122	0.04372	0.02282

Table 5.11: Threshold value for all models optimizing TPR.

exact. The interpretation is that a firm with a larger bankruptcy probability than 3.1% is classified as bankrupt. All threshold values are as expected very low because of the class imbalance in our data, as Table 5.11 show. One point worth noting in Table 5.11 is the remarkably low value for the QDA model validating our suspicion in the previous section, that the basecase threshold is too high.

5.2.2 Optimized Results

Results of performance after optimizing the threshold values are presented in Figure 5.22, while the resulting ROC curves are displayed in Figure 5.23 and the associated AUROC values are presented in Table 5.12. We observe that after the optimization, Random Forest is the model that performs best of all the models, with an AUROC of 0.7837, followed by the former best model, NN, with an AUROC of 0.7698. In other words, the models are sharply enhanced, especially with respect to the TPR.

Regarding the overall accuracy, the two models, RF and NN, result in an overall accuracy of 77.7% and 76.5% respectively, which is lower than in the basecase, but as stated the measure is somewhat misleading. Nevertheless, we state the measure in Table 5.12, since it has been widely used in previous research.

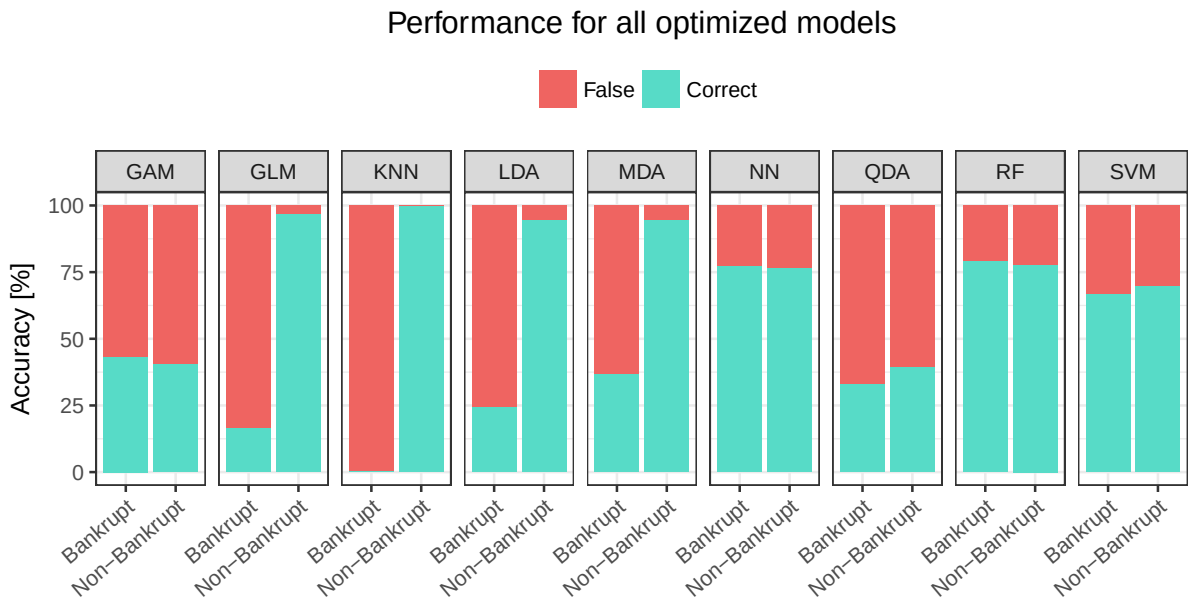

Figure 5.22: Barplot of all optimized models, KNN results are from the basecase.

Model	GAM	GLM	KNN	LDA	MDA	NN	QDA	**RF**	SVM
AUROC	0.5817	0.5653	0.5021	0.6833	0.7524	0.7698	0.6364	**0.7837**	0.6827
TPR	43.2	16.8	0.512	71.5	76.3	77.5	33.5	**79.0**	66.9
TNR	40.5	96.6	**99.5**	65.1	74.2	76.5	39.5	77.7	69.7
Accuracy	40.6	94.4	**97.2**	65.3	74.2	76.5	39.3	77.7	69.6

Table 5.12: AUROC, TPR, TNR and accuracy for all models with optimized TPR.

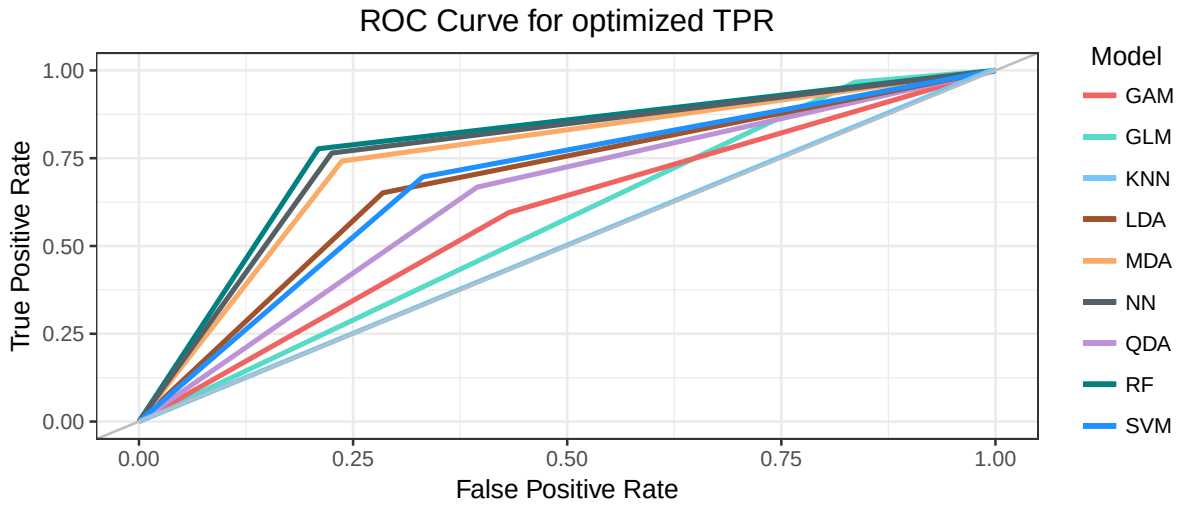

Figure 5.23: ROC curves for optimized TPR.

As mentioned, the threshold value for QDA is remarkably low. This could be interesting to further investigate, even though the prediction accuracy is improved from the basecase and seems reasonable compared to the other models. In addition, the AUROC of QDA is performing

extremely poor in the basecase, below 0.5, while it is improved to 0.6364 with the new threshold. Hence, the low threshold value seems to be valid.

Table 5.12 also displays the TPR and TNR of the models, showing a TPR of 79% for RF, compared to 45.8% in the basecase, which is an improvement in the TRP of 33.2%. The TNR is, of course, suffering by this as $77.7\% < 94.9\%$, but this is the trade-off one must deal with. All our techniques perform better in terms of the AUROC compared to the basecase.

		Actual					
		Random Forest		**Neural Network**		**Mixture Discriminant**	
		Bankrupt	Non-Bankrupt	Bankrupt	Non-Bankrupt	Bankrupt	Non-Bankrupt
Prediction	Bankrupt	14 497	144 399	14 211	152 280	13 991	167 096
	Non-Bankrupt	3 840	502 745	4 126	494 864	4 346	480 048

Table 5.13: Confusion matrix for the optimized best fold for the three best performing models.

Table 5.13 displays the confusion matrix for the three best models, showing the absolute values of the best performing models. We observe, not surprisingly, that in absolute units, RF yields the highest number of true positives and true negatives.

5.3 Variable Importance

In this section, we briefly present the variables the final models seem to rely most on when predicting the probabilities. We collected the coefficients for LDA, MDA, GLM and GAM, the importance of the variables in RF, and the weights for each hidden node in the NN-model. For the remaining models, QDA, KNN and SVM, we are not able to extract this information from the models. Since MDA is iterated five times, and results in five lists of coefficients, building on each other, we use the sum of these five as the final coefficient. We use the same procedure to get one final weight from the nine different weights in NN.

We utilize *three* different approaches to find the importance of the individual variables, two where we normalize the data and one with a top/bottom approach. The technical procedure is explained in appendix A12, for the interested reader. Finally, we aggregate the results for these three approaches, based on these six models, to interpret which variables overall seem to be most important in our study. All the variables which are selected two or more times in our analysis are presented in Table 5.14. We observe that cash flow from investing activities seems to be most important across all models, resulting in 11 out of 18 possible votes.

The net cash flow variables seem to overall be very important across models, along with total interest-bearing liabilities, provisions, profits and both short and long term libilities, among others. There are also several non-financial data that seem to be important, like the number of employees, industry sectors, legal form of company, number of board members and auditor remarks. Thus, we have a strong indication that not only financial data is relevant for prediction purposes with respect to bankruptcy, but also non-financial features.

Variable Selection by Coefficients, Importance and Weights

Votes	Final selection	Votes	Final selection
11	net.cashflow.invest	3	sector.07.transport
9	tot.ib.liab.max	3	total.income
8	net.change.cash	3	personnel.expense
7	other.provisions	3	other.opex
7	provisions.for.commit	3	accounts.receivables
7	lt.provisions.commitments	3	other.st.assets
6	net.cashflow.oper	3	total.equity
6	ib.st.liab.max	3	total.liabilities
5	taxrate	3	lt.operational.liabilities
5	dbrating	3	general.manager.pay
5	result.to.equity	2	legal.form
5	other.lt.liabilities	2	largest.ownershare
5	long.term.liabilities	2	chair.change
5	short.term.liabilities	2	complex.ownership
5	total.equity.liabilities	2	parent
5	st.marketbased.securities	2	region.nord.norge
4	paid.tax.rate	2	boardmemb.2.3
4	transferred.equity	2	sector.07.primsec.y
4	current.assets	2	sector.07.tert.y
4	total.assets	2	sector.07.health
4	other.st.liabilities	2	sector.07.realstate
4	net.cashflow.finance	2	net.cogs
4	employees	2	nopat
3	auditor.remarks	2	dividend
3	incorp.cat.lim.liability	2	public.duties.payable
3	boardmemb.1	2	financial.invest.pension
3	sector.07.construction	2	auditors.fee
3	sector.07.manufacturing	2	accounts.payable.t-1
3	sector.07.trade		

Table 5.14: Table of variable importance after extensive analysis of coefficients, importance and weights of the models LDA, MDA, GLM, GAM, RF and NN.

6 Case Study

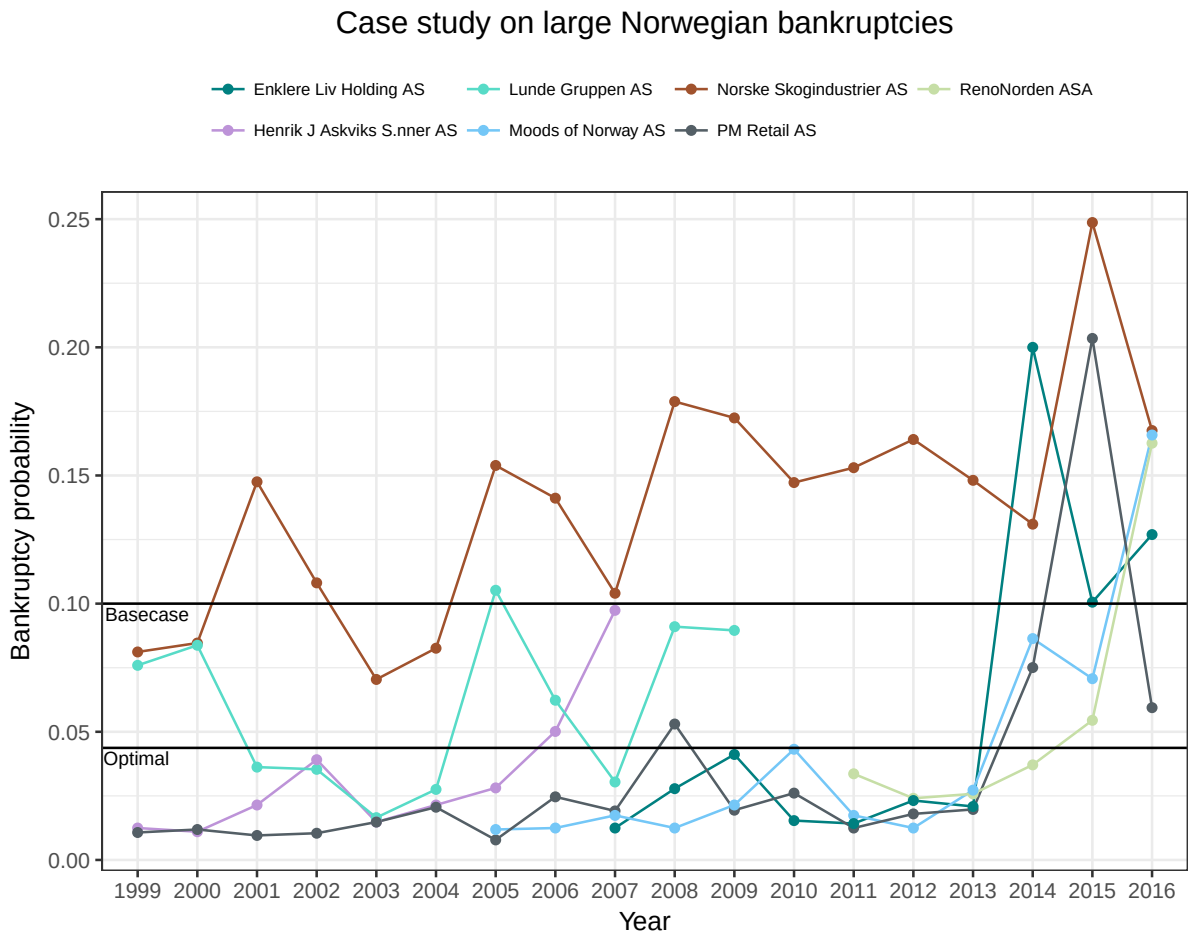

Figure 6.1: Case study of bankrupt Norwegian companies.

Based on the best Random Forest fold, we have completed a case study that visualizes our model and gives anecdotal evidence for our findings. The sample our model is built on is large bankruptcies in the Norwegian market from 1999 until 2016[32]. Furthermore, we would like to point out that some of the data points are used for fitting the model, reducing the credibility of this case study. We emphasize that this case study should not be viewed as validation of our model.

The results are shown in Figure 6.1, where each line corresponds to a specific firm and the associated bankruptcy risk in a particular year. The horizontal lines show the different classification thresholds. For *Norske Skogindustrier*, which went bankrupt in 2017 with last filings in 2016, the model would classify the firm as bankrupt over the entire time period. This indicates a very high bankruptcy probability. Another large bankruptcy in Norway was *Lunde*

[32]Note that some of the bankruptcies have been filed after 2016.

Gruppen with last filings in 2009 and official proceedings starting in 2011. Due to the reversal in 2007, our model is able to predict two years prior to the bankruptcy.

A more impressive case is the bankruptcy of the traditional shipbuilder *Henrik J Askviks Sønner AS*, which went bankrupt in 2008 with last filings in 2007, where the figure shows a consistent upward sloping probability until it went bankrupt. The multinational garbage collector *RenoNorden* went into bankruptcy proceeding in 2017 with the last filing one year prior. Our model indicates that two years prior we would classify the firm as bankrupt. For the bankruptcy in *PM Retail*, we are missing data on 2017. However, we classify the firm as bankrupt up to two years prior. As for the remaining companies we observe the same trend. In general, the model is able to correctly predict bankruptcies ahead of time. However, this is at the cost of a higher FPR and a lower overall accuracy. For the two different thresholds, the figure show they yield different TPRs.

7 Discussion

In this section, we first discuss our findings and compare with previous literature introduced in section 2. In addition, we discuss variable selection in our study compared to past utilized ratios, and other issues regarding our data and models. Lastly, we give our opinion on future applications for bankruptcy predictions.

For our findings, we obtain an AUROC of 78% and 77% for the optimized models based on Random Forest and Neural Network, respectively. This indicates that our models are able to predict the class separation. Regarding the true rates, Random Forest is able to predict 79% correctly for bankrupt firms, while 78% correctly for non-bankrupt firms. The performance by Neural Network is slightly worse with 78% and 77% correctly classified, respectively. In other words, the models are able to correctly classify 4/5 in both classes. The results indicate that 80% of the variation in bankruptcies, at this threshold, can be explained by the variables that we have utilized, based on financial, management and sector statements. The remaining 20% may be related to other underlying trends our data is unable to capture such as macroeconomic conditions and black swans.

Overall, the performance of RF and NN is very good, showing that more complex models perform better than other models when presented with the financial, management and sector statements.

7.1 Comparison to the Literature

Based on the literature, we compare the performance of our model to previously created models. Not surprisingly, Altman Z, Z' and Z"-score outperforms our model in terms of overall accuracy. However, our basecase model, with a threshold of 10% outperforms certain years within the training range of the different Z-scores. Regarding the FPR and FNR, it is natural to compare the error rates at three years prior since this is what our model is trained on. For three years prior, across all of the different models and samples, the worst FPR is 10.3% while the best is 8.6%. For the worst error, FNR, the worst is 52% while the best is 25.5%. Our best RF fold yields an FPR of 20.9 % and 22.3% for FNR. Hence, our model produces a lower FNR and a higher FPR. Ohlson's O-score also provides the FPR and FNR, which is 20.2% and 8.6%, respectively. Indicating that the O-score is superior to our results.

These aforementioned models are created on a balanced response variable, i.e. a artificial distribution of the classes, which we believe is not correct to utilize on real-world applications. Therefore, these models can be viewed as a gold standard that is impossible to surpass on class imbalanced problems. We have chosen not to run these models on our data because there is empirical evidence that these ratios are not optimal, as indicated by Pelja and Stemland (2017) and Næss et al. (2017).

The best SEBRA model, SEBRA Extended, achieves an AUROC of 0.897 compared to ours at 0.783, indicating that the SEBRA Extended is superior. The best model created by Næss et al. (2017) is also superior with an AUROC of 0.911. However, as mentioned in the previous section these models are not entirely comparable to ours since they are based on a balanced response variable[33]. Both of these methods exclude small and large firms, which eliminates the extreme cases, making it easier for the model to perform well. For new data, these models are likely to yield a lower AUROC. Compared with the Bloomberg DRSK model our model outperforms this model in both accuracy and TPR, indicating that our model is superior on class imbalanced data. Lastly, the model created by Berg (2007) obtains an AUROC of 0.773 in GAM, while our highest AUROC is 0.783, indicating that our model is slightly better.

Compared to Figure 2.3 on page 12, where we show the TPR and TNR for different techniques, the TPR for LDA is similar to the average of previous studies. For MDA our results are in the lower quartile compared to the aforementioned figure. Our QDA performs significantly worse than previous papers, while NN are in the upper middle quartile.

7.2 Discussion of Dimensionality

The data our models are built upon suffers slightly from the curse of dimensionality. The method of KNN imposes unsupervised learning to reduce the dimensions through PLS. However, this is not done for the rest of the models. The obvious candidates that suffer from this are the generalized models, GLM and GAM, based on their weak performance. Most likely, the former method suffers from the linearity of its parameters, while the latter suffers from the additivity, which may limit the inherent flexibility and potentially miss important interactions, as we discussed in section 3.2.1. In addition, this may be a consequence of not performing variable

[33] Both models remove observations with very high or low assets thus skewing the class imbalance. In addition, Næss et al. (2017) uses a distribution of 1/3 bankrupt firms and 2/3 non-bankrupt firms.

selection on these two models. Our effort to mitigate the limitation of the model seems to have failed, based on the weak performance of the model and the fact that GAM has been performing well in past studies by Berg (2007) and Næss et al. (2017).

The techniques from discriminant analysis perform with our expectations, where MDA is superior, followed by LDA and lastly QDA. One possible explanation why LDA and QDA are inferior to MDA is that these two models struggle to find reliable decision boundaries. As our literature review indicates, QDA seem to perform poorly, indicating that QDA is not optimal for predicting bankruptcies.

For the three remaining model our results show that Random Forest is better suited for processing a lot of features, due to the decorrelation procedure that the model imposes on each subtree. This is indicated in the RF model yielding the highest TPR. Support Vector Machines produce fair predictions but struggle with many features. We solved this by using a cell-grid-system, which works as a regulator between optimal accuracy and computational time. Our results indicate that even with the regulator the performance decay is too large to compete with Random Forest. Lastly, Neural Network also performs well with many features. This is rooted due to the neuron system, where the model is able to find important interactions across the feature space and properly update the nodes after each iteration.

7.3 Comparison of Variable Selection

In section 5.3 we tested for which variables our models emphasize as the most important for predicting bankruptcy. The most important variables in Table 5.14 are presented in Figure 7.1.

We notice that the net cash flow measures are recurring. Comparing to previous literature, the most used ratios, presented in Table 2.4 on page 14, do not include cash flows. Hence, we argue that cash flows should have been included to a greater extent in the past since it apparently is a good predictor and thus most likely would improve bankruptcy predictions. Other types of financial variables that are neglected in the past, which are deemed important in our models are provisions, commitments, cash equivalents such as market based securities, tax rates, paid tax rate and expenses like payroll and miscellaneous.

We also observe variables such as short and long term liabilities, total income, result to equity and both total and current assets are recurring in Table 5.14. This is interesting, since ratios

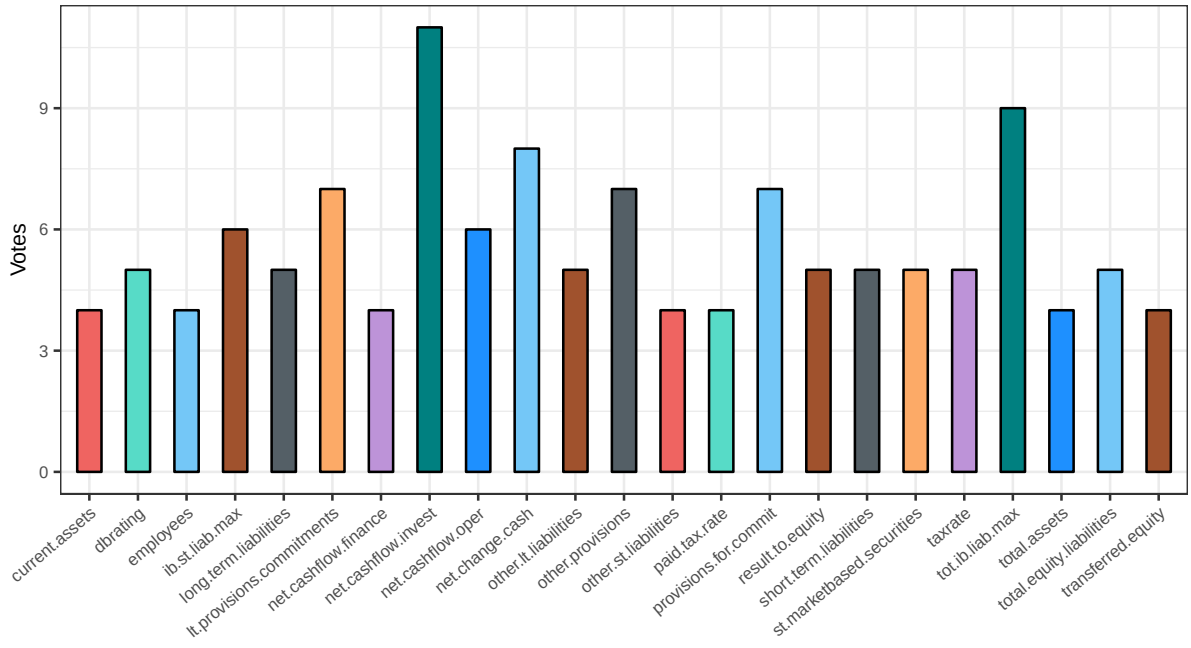

Figure 7.1: Distribution of variables our models emphasizes.

for liquidity, CATA (Current assets/Total assets), NITI (Net income/Total assets), TDTA (Total debt/Total assets), TLTA (Total liabilities/Total assets) and EBIT/TA (Earnings before interest and taxes/Total assets) are highly used in the past, as Table 2.4 shows.

Hence, our results indicate that these measures are valid to some degree. This implies that the previous ratio-intuition seems to be somewhat correct, in terms of selected ratios. The previously used ratios are primarily driven by assets, while our results show that liabilities are much more important. In Figure 7.1, only 3 of the features are related to assets, while 10 of the features are related to liabilities. This indicates that liabilities have a higher predicting power of bankruptcies than assets.

On the other side, features related to management and sector have mainly been neglected and ignored in the past. In our variable importance test, we find that several of these features are deemed important across methods, especially auditor remarks, number of employees, number of board members, industry sectors, incorporated types and salary of the general manager. Therefore, we suggest that these types of variables have predicting power and should have been included to a greater extent, since they, like the financial statement, also reflect the state of the company.

7.4 Discussion of Time-Validation and Applications

If the application of our models should have any real-world value, it is evident that the models are capable of predicting a firm bankrupt ahead of time. Due to the structure of the multi-year model, this indicates the model will have a higher FPR compared to a one-year model ceteris paribus. This is also indicated in the results of our simple case study in section 6, where the model classifies firms as bankrupt ahead of time and that a proportion of our FPR is actually firms that will go bankrupt over the lifetime of the company, but not in that particular year the prediction was made. This is a trade-off between the aggressiveness of the model, and how long the model will delay sending the *distress signal*.

For the application of our model, the thresholds should be seen as guides, not definitive solutions. This was illustrated in the case study in section 6. Therefore, the threshold should vary according to the purpose and intention of the prediction. Viewed from an external stakeholder position, for example a bank manager, it is in the interest to correctly predict most of the firms that go bankrupt, since bankrupt firms could be very costly. This implies that perhaps an even lower threshold should be used. For suppliers, where the cost of bankruptcy is not that severe a higher threshold than the optimized may be suitable.

7.5 Further Research

Further research which we would like to pursue is an out-of-time validation for our models, similar to the approach perform by Berg (2007). Hence, we would to a greater extent be able to quantify how well the model performs 1-3 years ahead of time and thus be able to compare the models with past literature. Another application we would have liked to pursue is to create independent models for each sector, to identify sector-specific features that are important.

Furthermore, we think it would be very interesting to look into the application of sentiment analysis and text mining. It would be very interesting to test if annual reports and their wording has any predictive power on the bankruptcy prediction. To the best of our knowledge, this has not been tested in the bankruptcy literature yet, even though it has become an increasingly more popular application. Lastly, it could be interesting to look at the relationship between bankruptcy predictions and the associated cost of capital and credit rating, because a higher default probability should lead to a higher cost of capital and worse rating.

8 Conclusion

The field of bankruptcy prediction has been evolving since the earliest adoptions at the beginning of the twentieth century. The field has gone through a reformation as computational power has excelled. The most renowned and applied models today are *simple* ratio models which were developed in the last part of the twentieth century. Through our thesis, we have focused on our objective to create a new multi-year model for bankruptcy prediction, customized for the Norwegian market, which focuses on financial, management and sector statements.

Previous research indicates that eight different machine learning techniques are suitable for bankruptcy prediction and that the previously untested Random Forest may be superior. Therefore, we have created nine different models, one for each technique, with different statistical properties. All of these nine models are unfocused models that are applicable to all sectors except for the financial and insurance sector.

The models have been tested under two different cut-off thresholds, one that is empirically tested and one that is optimized for each technique. The optimization is done through maximizing true positive predictions, which consequently minimizes false negative predictions, where the optimal threshold is dependent on the technique and coherent probability structure.

Our research indicates that Random Forest, Neural Network, and Mixture Discriminant Analysis are the three best performing models, obtaining an accuracy for bankrupt companies at 79.0%, 77.5%, and 76.3 %, respectively. Overall, these models obtain an AUROC value of 0.783, 0.778 and 0.753, respectively, where the AUROC is a measure of class independent accuracy. This indicates that we are able to correctly predict 4/5 of both bankrupt and non-bankrupt companies, or in other words that 80% of the variation in bankruptcies can be explained by our data at this threshold. Our best model is superior compared to comparable models that are created on class imbalanced data, while the model is inferior compared to models built on balanced class data.

The models provide some evidence toward the appropriateness of previously used ratios, where our models assign high importance to some of the individual components the ratios are built upon, such as liquidity. However, the models also indicate that other important factors have been neglected in the past, such as all cash flow measures and the size of the board. These traits are important and have predictive power. Furthermore, our models indicate that liabilities are more important than assets, in contrast to the asset-driven ratios utilized in the past.

References

A. Mahmood, M. and C. Lawrence, E. (2007), 'A performance analysis of parametric and nonparametric discriminant approaches to business decision making', *Decision Sciences* **18**, 308 – 326.

Agarwal, A. (1993), Neural networks and their extensions for business decision making, PhD thesis, Ohio State University.

Altman, E. (2000), 'Predicting financial distress of companies: Revisiting the z-score and zeta', *Handbook of Research Methods and Applications in Empirical Finance* **5**.

Altman, E., Hartzell, J. and Peck, M. (1995), 'Emerging market corporate bonds — a scoring system', *The Future of Emerging Market Flows* .

Altman, E. I. (1968), 'Financial ratios, discriminant analysis and the prediction of corporate bankruptcy', *The Journal of Finance* **23**(4), 589.

Altman, E. I., Haldeman, R. G. and Narayanan, P. (1977), 'Zetatm analysis a new model to identify bankruptcy risk of corporations', *Journal of Banking & Finance* **1**(1), 29–54.

Anandarajan, M., Lee, P. and Anandarajan, A. (2001), 'Bankruptcy prediction of financially stressed firms: an examination of the predictive accuracy of artificial neural networks', *Intelligent Systems in Accounting, Finance and Management* **10**(2), 69–81.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/isaf.199*

Baldwin, J. and Glezen, G. W. (1992), 'Bankruptcy prediction using quarterly financial statement data', *Journal of Accounting, Auditing & Finance* **7**(3), 269–285.
URL: *https://doi.org/10.1177/0148558X9200700301*

Beaver, W. H. (1966), 'Financial ratios as predictors of failure', *Journal of Accounting Research* **4**, 71–111.
URL: *http://www.jstor.org/stable/2490171*

Beaver, W. H. (1968*a*), 'Alternative accounting measures as predictors of failure', *The Accounting Review* **43**(1), 113–122.
URL: *http://www.jstor.org/stable/244122*

Beaver, W. H. (1968*b*), 'Market prices, financial ratios, and the prediction of failure', *Journal of Accounting Research* **6**(2), 179–192.
URL: *http://www.jstor.org/stable/2490233*

Bellman, R. (1957), *Dynamic Programming*, Dover Books on Computer Science Series, Dover Publications.
URL: *https://books.google.no/books?id=fyVtp3EMxasC*

Bellovary, J. L., Giacomino, D. E. and Akers, M. D. (2007), 'A review of bankruptcy prediction studies: 1930 to present', *Journal of Financial Education* **33**, 1–42.
URL: *http://www.jstor.org/stable/41948574*

Berg, D. (2007), 'Bankruptcy prediction by generalized additive models', *Applied Stochastic Models in Business and Industry* **23**(2), 129–143.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/asmb.658*

Berner, E., Mjøs, A. and Olving, M. (2016), Norwegian Corporate Accounts : Documentation and quality assurance of SNF's and NHH's database of accounting and company information for Norwegian companies. Unpublished Working paper.

Bernhardsen, E. (2001), A Model of Bankruptcy Prediction, PhD thesis, University of Oslo.

Bernhardsen, E. and Larsen, K. (2007), 'Modellering av kredittrisiko i foretakssektoren – videreutvikling av sebra-modellen', *Penger og Kreditt* **35**(2), 60–67.

Black, F. and Scholes, M. (1973), 'The pricing of options and corporate liabilities', *Journal of Political Economy* **81**(3), 637–654.
URL: *https://doi.org/10.1086/260062*

Boyacioglu, M. A., Kara, Y. and Ömer Kaan Baykan (2009), 'Predicting bank financial failures using neural networks, support vector machines and multivariate statistical methods: A comparative analysis in the sample of savings deposit insurance fund (sdif) transferred banks in turkey', *Expert Systems with Applications* **36**(2, Part 2), 3355 – 3366.
URL: *http://www.sciencedirect.com/science/article/pii/S095741740800078X*

Breiman, L. (2001), 'Random forests', *Machine Learning* **45**(1), 5–32.
URL: *https://doi.org/10.1023/A:1010933404324*

Cacciatore, S., Tenori, L., Luchinat, C., Bennett, P. R. and MacIntyre, D. A. (2018), *KODAMA: Knowledge Discovery by Accuracy Maximization*. R package version 1.5.
URL: *https://CRAN.R-project.org/package=KODAMA*

Cai, L. and Singenellore, R. (2012), *Framework, Methodology and Usage*, Bloomberg Professional Service. Bloomberg Terminal: DRSK "Go".

Casey, C. and Bartczak, N. (1985), 'Using operating cash flow data to predict financial distress: Some extensions', *Journal of Accounting Research* **23**(1), 384–401.
URL: *http://www.jstor.org/stable/2490926*

Chen, N., Ribeiro, B., Vieira, A., Duarte, J. and Carvalho das Neves, J. (2011), 'A genetic algorithm-based approach to cost-sensitive bankruptcy prediction', *Expert Syst. Appl.* **38**, 12939–12945.

Chen, T., Kou, Q. and He, T. (2017), *mxnet: MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems*. R package version 1.3.0.
URL: *https://github.com/dmlc/mxnet/tree/master/R-package*

Christmann, A. and Steinwart, I. (2008), *Support Vector Machines*, Springer Science+Business Media.

Correia, M. M. (2018), 'After 50 years - what is the future of bankruptcy prediction?'. The Frøstein Gjesdal Lecture.

Crouhy, M., Galai, D. and Mark, R. (2000), 'A comparative analysis of current credit risk models', *Journal of Banking & Finance* **24**, 59–117.

Davis, M. and Etheridge, A. (2006), *Louis Bachelier's Theory of Speculation: The Origins of Modern Finance*, Princeton University Press.
URL: *http://www.jstor.org/stable/j.ctt7scn4*

Deakin, E. B. (1972), 'A discriminant analysis of predictors of business failure', *Journal of Accounting Research* **10**(1), 167–179.
URL: *http://www.jstor.org/stable/2490225*

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the em algorithm', *Journal of the Royal Statistical Society. Series B (Methodological)* **39**(1), 1–38.
URL: *http://www.jstor.org/stable/2984875*

Dwyer, M. (1992), A comparison of statistical techniques and artificial neural network models in corporate bankruptcy prediction, PhD thesis, University of Wisconsin.

El Hennawy, R. and Morris, R. (1983), 'The significance of base year in developing failure prediction models', *Journal of Business Finance & Accounting* **10**(2), 209–223.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1468-5957.1983.tb00424.x*

Farooq, M. and Steinwart, I. (2017), 'An svm-like approach for expectile regression', *Computational Statistics & Data Analysis* **109**(C), 159–181.
URL: *https://EconPapers.repec.org/RePEc:eee:csdana:v:109:y:2017:i:c:p:159-181*

Fawcett, T. (2006), 'An introduction to roc analysis', *Pattern Recognition Letters* **27**(8), 861–874.

Finanstilsynet (2017), 'Sebra-rapportering (krt-1115)'.
URL: *https://www.finanstilsynet.no/rapportering/fellesrapporteringer/sebra-rapportering/*

Gaeremynck, A. and Willekens, M. (2003), 'The endogenous relationship between audit-report type and business termination: Evidence on private firms in a non-litigious environment', *Accounting and Business Research* **33**, 65–79.

Garson, J. (2018), Connectionism, *in* E. N. Zalta, ed., 'The Stanford Encyclopedia of Philosophy', fall 2018 edn, Metaphysics Research Lab, Stanford University.

Gentry, J. A., Newbold, P. and Whitford, D. T. (1985), 'Classifying bankrupt firms with funds flow components', *Journal of Accounting Research* **23**(1), 146–160.
URL: *http://www.jstor.org/stable/2490911*

Goodfellow, I., Bengio, Y. and Courville, A. (2016), *Deep Learning*, MIT Press.
URL: *http://www.deeplearningbook.org*

Grover, J. (2003), Validation of a cash flow model: A non-bankruptcy approach, PhD thesis, Nova Southeastern University.

Gu, Z. and Gao, L. (2000), 'A multivariate model for predicting business failures of hospitality firms', *Tourism and Hospitality Research* **2**(1), 37–49.
URL: *http://www.jstor.org/stable/23743573*

Hastie, T. (2018), *gam: Generalized Additive Models*. R package version 1.16.
URL: *https://CRAN.R-project.org/package=gam*

Hastie, T. and Tibshirani, R. (1986), 'Generalized additive models', *Statist. Sci.* **1**(3), 297–310.
URL: *https://doi.org/10.1214/ss/1177013604*

Hastie, T. and Tibshirani, R. (1990), *Generalized additive models*, 1 edn, Chapman and Hall/CRC. Monographs on Statistics and Applied Probability 43.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2 edn, Springer.

Hastie, T., Tibshirani, R., Leisch, F., Hornik, K. and Ripley, B. D. (2017), *mda: Mixture and Flexible Discriminant Analysis*. ,R package version 0.4-10.
URL: *https://CRAN.R-project.org/package=mda*

Hecht-Nielsen, R. (1992), 'Theory of the backpropagation neural network**based on "nonindent" by robert hecht-nielsen, which appeared in proceedings of the international joint conference on neural networks 1, 593–611, june 1989.', *Neural Networks for Perception* pp. 65–93.

Hopfield, J. J. (1982), 'Neural networks and physical systems with emergent collective computational abilities', *Proceedings of the National Academy of Sciences* **79**(8), 2554–2558.
URL: *https://www.pnas.org/content/79/8/2554*

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2017), *An introduction to statistical learning*, 8 edn, Springer.

Kiviluoto, K. (1998), 'Predicting bankruptcies with the self-organizing map', *Neurocomputing* **21**, 191–201.

Kohavi, R. (2001), 'A study of cross-validation and bootstrap for accuracy estimation and model selection', **14**.

Koller, T., Goedhart, M., Wessels, D., Schwimmer, B. and Manoury, F. (2015), *Valuation: Measuring and managing the value of companies*, Wiley. McKinsey and Company.

Langli, J. C. (2010), *Årsregnskapet*, 9 edn, Gyldendal.

Lee, K. C., Han, I. and Kwon, Y. (1996), 'Hybrid neural network models for bankruptcy predictions', *Decision Support Systems* **18**(1), 63 – 72. Artificial Intelligence and Expert Systems.
URL: *http://www.sciencedirect.com/science/article/pii/0167923696000188*

Liaw, A. and Wiener, M. (2002), 'Classification and regression by randomforest', *R News* **2**(3), 18–22. Fortran original by Leo Breiman.
URL: *https://CRAN.R-project.org/doc/Rnews/*

Martens, D. and Provost, F. (2011), 'Pseudo-social network targeting from consumer transaction data', *NYU Working Paper No. CEDER-11-05* .

Masters, T. (1993), *Practical neural network recipes in C++*, 1 edn, Morgan Kaufmann.

Merton, R. C. (1974), 'On the pricing of corporate debt: The risk structure of interest rates', *The Journal of Finance* **29**(2), 449–470.
URL: *http://www.jstor.org/stable/2978814*

Ministry of Justice & Public Security (1986), *The Bankruptcy Act – kkl*, Ministry of Justice & Public Security.
URL: *http://www.konkursradet.no/konkursloven-oversatt-til-engelsk.5798530-300331.html*

Moro, S., Cortez, P. and Rita, P. (2014), 'A data-driven approach to predict the success of bank telemarketing', *Decision Support Systems* **62**.

Nelder, J. A. and Wedderburn, R. W. M. (1972), 'Generalized linear models', *Journal of the Royal Statistical Society. Series A (General)* **135**(3), 370–384.
URL: *http://www.jstor.org/stable/2344614*

Norwegian School of Economics (2019), 'Børsprosjektet'.
URL: *https://www.nhh.no/forskning/borsprosjektet/*

Næss, A. B., Wahlstrøm, R. R., Forbod, F. and Kjærland, F. (2017), *Konkursprediksjon for norske selskaper - En sammenligning av regresjonsmodeller og maskinlæringsteknikker.*, Fagbokforlaget, NTNU, pp. 313–330.

Ohlson, J. A. (1980), 'Financial ratios and the probabilistic prediction of bankruptcy', *Journal of Accounting Research* **18**(1), 109–131.
URL: *http://www.jstor.org/stable/2490395*

Patterson, D. W. (2001), Bankruptcy prediction: A model for the casino industry, PhD thesis, University of Nevada.
URL: *https://digitalscholarship.unlv.edu/rtds/2457*

Pelja, I. and Stemland, T. B. (2017), 'Predicting bankruptcy for norwegian firms : a study of altman's z´´-model using alternative ratios'. Master thesis, Norwegian School of Economics.

Petersen, C., Plenborg, T. and Kinserdal, F. (2017), *Financial statement analysis*, Fagbokforlaget.

R Development Core Team (2008), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
URL: *http://www.R-project.org*

Rahbari, D. (2014), 'A novel approach in classification by evolutionary neural networks', *International Journal of Computer Science and Network Security (IJCSNS)* **14**, 86.
URL: *http://paper.ijcsns.org/07_book/201409/20140916.pdf*

Reinert, H. and Reinert, E. S. (2006), *Creative Destruction in Economics: Nietzsche, Sombart,Schumpeter*, Springer US, Boston, MA, pp. 55–85.

Revsine, L., Collins, D., Johnson, B., Mittelstaedt, F. and Soffer, L. (2018), *Financial Reporting and Analysis*, 7 edn, McGraw-Hill Education.

Ripley, B. (2018), *tree: Classification and Regression Trees*. R package version 1.0-39.
URL: *https://CRAN.R-project.org/package=tree*

Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge University Press, p. 1–16.

Shah, J. and Murtaza, M. (2000), 'A neural network based clustering procedure for bankruptcy prediction', *American Business Review* **18**.

Shumway, T. (2001), 'Forecasting bankruptcy more accurately: A simple hazard model', *The Journal of Business* **74**(1), 101–124.
URL: *http://www.jstor.org/stable/10.1086/209665*

Skogsvik, K. (1990), 'Current cost accounting ratios as predictors of business failure: The swedish case', *Journal of Business Finance & Accounting* **17**(1), 137–160.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1468-5957.1990.tb00554.x*

Sobehart, J., Keenan, S. and Stein, R. (2000), 'Validation methodologies for default risk models', *Credit* **1**, 51–56.

Steinwart, I. and Thomann, P. (2017), 'liquidSVM: A fast and versatile svm package', *ArXiv e-prints 1702.06899* .
URL: *http://www.isa.uni-stuttgart.de/software*

Tam, K. Y. and Kiang, M. Y. (1992), 'Managerial applications of neural networks: The case of bank failure predictions', *Management Science* **38**(7), 926–947.
URL: *http://www.jstor.org/stable/2632376*

Taylor, J. (2018), 'Classification, lda'. Stanford Statistics.
URL: *https://web.stanford.edu/class/stats202/content/lec9.pdf*

Vapnik, V. and Bottou, L. (1993), 'Local algorithms for pattern recognition and dependencies estimation', *Neural Computation* **5**(6), 893–909.
URL: *https://doi.org/10.1162/neco.1993.5.6.893*

Venables, W. N. and Ripley, B. D. (2002), *Modern Applied Statistics with S*, fourth edn, Springer, New York. ISBN 0-387-95457-0.
URL: *http://www.stats.ox.ac.uk/pub/MASS4*

Wahlstrøm, R. R. and Helland, F. F. (2016), 'Konkursprediksjon for norske selskaper – en analyse ved maskinlæringsteknikker og tradisjonelle statistiske metoder'. Master thesis, Norwegian University of Science and Technology.

Wall, A. (1919), 'Study of credit barometers', *Federal Reserve Bulletin* **5**(3), 229 – 242.

Wang, B. (2004), Strategy changes and internet firm survival, PhD thesis, University of Minnesota.

Zhang, B. Y., Zhou, H. and Zhu, H. (2009), 'Explaining Credit Default Swap Spreads with the Equity Volatility and Jump Risks of Individual Firms', *The Review of Financial Studies* **22**(12), 5099–5131.
URL: *https://doi.org/10.1093/rfs/hhp004*

Zhang, G., Hu, M. Y., Patuwo, B. E. and Indro, D. C. (1999), 'Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis', *European Journal of Operational Research* **116**(1), 16 – 32.
URL: *http://www.sciencedirect.com/science/article/pii/S0377221798000514*

Appendix

A1 Technical Notes on Bankruptcy

At the end of the lifespan of a company usually three things could happen. For inactive companies, it could disband without legal repercussions. The second option is that the court disbands the company due to neglect. This could manifest itself through a lack of reporting, missing information from the company or inadequate company structures. Lastly, the company could disappear due to bankruptcy. The technical definition of bankruptcy is that the overall liabilities have surpassed assets and that the entity is unable to cover the difference from operational income. The formal definition is that an entity does not have enough liquidity to cover short-term debt thus is insolvent.

According to Norwegian legislation; ”If the debtor is insolvent, (...) the estate shall be subject to bankruptcy proceedings (...)” (Ministry of Justice & Public Security, 1986). Furthermore, the legislation also defines insolvency; ”The debtor is insolvent when he/she cannot meet his/her obligations as they fall(...)”- (Ministry of Justice & Public Security, 1986). Furthermore, the creditor is obliged to pay collateral for the maximum liability the court may need. Thus, increasing the threshold of submitting a bankruptcy petition. This creditor is able to recollect this collateral if the liquidated estate contains liaison-free assets of this amount. For assets with covenants or liaisons, these will be honored first while the remaining assets will be allocated at a pro-rata basis.

The Norwegian bankruptcy laws are compliant with international standards such as the famous Chapter 11 and Chapter 7. The latter is a regulation that controls the liquidation of the company, a sort of estate sale, while the former gives the company time to resolve their insolvency issues. Therefore, when a company goes bankrupt it could spark a long-lasting process that could take years to settle, thus the company stops financial reporting. This could impose noise and inconsistencies in our data, which could cause problems since the company enters a bankruptcy-like state, resolve their insolvency issues and continue to exist. However, as outlined, this is solved through our data patch. The domain of bankruptcy filings has traditionally been a domain of attorneys, which has primarily been driven by legislation within the field and requirements for the trustee. Economists, however, have to a greater extent focused on bankruptcy predictions.

A2 Altman Z”-Score

Altman (2000) also created another revised Z-score model, called Z” that was customized for non-manufacturing firms. In this model, the X_5 variable, sales/total assets were removed in order to minimize the potential industry effect. The model was also used to assess the financial health of non-U.S. corporations (Altman et al., 1995). This model used the book value of equity as X_4. The Z”- Score model for non-manufacturing (and emerging markets (Altman et al., 1995)) were given by the following coefficients:

$$Z'' = (3.25) + 6.56X_1 + 3.26X_2 + 6.72X_3 + 1.05X_4 \tag{.1}$$

As with the revised Z-score model, the cutoff score value changed in this model. For the model with emerging markets, the constant term +3.25 was added, to standardize the scores with a score of zero equated to a D (default) rated bond.

A3 Lag of Features

	LDA	RF	GAM	GLM	MDA	NN
Std	146	13	51	4.40E+20	2211	90
Mean of sample	-7	21	3	1.12E+19	-58	-8
other.opex	0.040	0.618	-0.065	-0.026	0.026	0.772
nopat	0.145	0.797	-0.065	-0.026	0.026	0.302
result.to.equity	-0.032	0.643	-0.065	-0.026	0.026	-0.553
total.receivables	-0.037	0.349	-0.012	-0.026	0.050	0.087
cash	0.057	1.101	-0.067	-0.026	0.033	0.291
current.assets	0.031	0.576	0.136	-0.026	-0.657	0.105
total.assets	-1.083	0.888	0.739	-0.026	0.109	0.104
paid.in.equity	0.465	-0.143	-0.068	-0.026	0.032	0.119
retained.earnings	0.873	1.394	-0.069	-0.026	0.032	0.538
pension.liabilities	0.265	-1.609	-0.268	-0.026	0.042	0.130
deferred.tax.liabilities	0.449	-1.205	-0.291	-0.026	0.043	0.141
other.provisions	1.187	-1.596	-0.215	-0.026	0.193	0.088
it.commitments	0.124	-1.549	-0.628	-0.026	0.063	0.088
provisions.for.commit	1.491	-1.200	0.702	-0.025	-0.146	0.092
other.lt.liabilities	0.352	-0.506	-0.412	-0.025	0.046	0.156
long.term.liabilities	-3.916	-0.472	-0.135	-0.025	-0.152	0.108
accounts.payable	0.015	1.441	-0.041	-0.026	0.040	-0.571
taxes.payable	0.036	-0.672	-0.061	-0.026	0.031	-1.091
short.term.liabilities	1.552	0.490	1.162	-0.019	-3.957	0.097
total.liabilities	-1.208	0.643	-3.851	-0.026	2.762	0.100
total.equity.liabilities	-1.018	0.925	0.232	-0.026	-0.076	0.101
st.marketbased.securities	0.047	-1.421	-0.131	-0.026	0.326	0.510
bond.loans	0.514	-1.564	-0.044	-0.026	NA	0.127
other.st.liabilities	0.011	0.601	1.043	-0.031	0.029	0.091
net.cashflow.oper	0.056	0.886	-0.065	0.939	1.025	0.068
net.cashflow.invest	0.043	0.504	-0.065	3.459	0.029	2.224
net.change.cash	0.049	0.451	NA	-3.760	0.026	-4.325
tot.ib.liab.max	-0.506	-0.370	2.603	-0.026	NA	0.099

Table A3.1: In this table we observe different importance or coefficients for different models. For most models, the values are generated based on a sample of 200 000 observations. All observations are scaled to a mean of zero for easy comparison, see Equation 4.1. Selected lag variables are bolded out. The selection of variables is based on a hard-decision vote between each model.

A4 Reclassification of the Balance Sheet and Income Statement

Structure of financial statements							
Assets			**Liabilities and equity**			**Income Statement**	
Time	*Class*	*Accounts*	*Time*	*Class*	*Accounts*	*Accounts*	
			LT	O	Paid in equity	Total income	
				F	Retained earnings	Net COGS	
				F	Other equity	Personnel expenses	
					Total equity	Depreciation	
			LT	B	Pension liabilities	Impairments	
			LT	B	Deferred taxes	Expenses for bad debt	
			LT	B	Provisions and commitment	other OPEX	
LT	B	Deferred tax assets			Total provisions	Tax on ordinary result	
LT	O	Intangible assets	LT	F	Subordinated loan capital	Net operating profit after tax	
		Total intangible fixed assets	LT	O	Other long-term liabilities	Income from investments	
LT	O	PP&E	LT	O	Operational liabilities	Gross interest income	
LT	O	Other fixed receivables	LT	B	Bond loans	Transitory financial income	
ST	O	Inventory			Long-term liabilities	Impairments of financial assets	
ST	O	Net other claims	ST	F	Alternative loans	Gross interest expense	
ST	O	Accounts receivables	ST	F	Liabilities to financial institutions	Transitory financial loss	
		Total receivables	ST	F	Convertible loans	Net financial accounts	
LT	F	Financial assets, JV and group	ST	O	Accounts payable	Ordinarily result before XO	
ST	F	Market-based securities	ST	O	Taxes payable	XO - Income	
LT	F	Financial investments and pensions	ST	F	Dividends payable	XO - Expenses	
ST	F	Derivatives and intercompany securities	ST	O	Public duties payable	Results to equity	
ST	B	Cash	ST	B	Other liabilities	Dividends	
ST	B	Other current assets			Short term liabilities	Transferred to equity	
		Total current assets			Total liabilities	Transferred to reserves	
		Total assets			Total equity and liabilities	Group contributions	

Table A4.1: Financial statement after reclassification, without net calculations. The indicative classes are highly debatable for such a generalized sample. The income statement is structured to divide operational, financial and transitory effects. F represents financing, O represents operations and B can be both.

A5 Legal Form of Incorporation

Legal form of Incorporation

Legal form	Description	Incorporation category	Action	Number of firms dropped
ANN	Other legal entity	Ownerless	Removed	473
ANNA	Other legal entity	Ownerless	Removed	661
ANS	Unlimited company	Partners	Kept	0
AS	Limited share company	Limited	Kept	0
ASA	Public limited company	Limited	Kept	0
BA	Limited company	Limited	Removed	8107
BBL	Coop. Building association	Ownerless	Removed	1203
BRL	Housing cooperative	Ownerless	Removed	89633
DA	Shared liability	Partners	Kept	0
ENK	Sole proprietorship	Sole property	Kept	0
ESE	Real estate partnership	Partners	Removed	3556
ESEK	Real estate partnership	Partners	Removed	39697
FKF	Country company	Public	Removed	14
FLI	Associations etc.	Ownerless	Removed	13495
GFS	Mutual insurance company	Ownerless	Removed	860
IKS	Inter municipal company	Public	Removed	1538
KOMM	Municipality	Public	Removed	3
KS	Municipality	Public	Kept	0
NUF	Joint office	Partners	Kept	0
PK	Organizational unit	Ownerless	Removed	585
PRE	Pension trust	Ownerless	Kept	0
REV	Shipowning partnership	Partners	Kept	0
SÆR	Other legally defined	Public	Removed	969
SA	Sole proprietorship(acc)	Sole property	Kept	0
SAM	Partnership	Partners	Removed	1751
STI	Trust/foundation	Ownerless	Removed	100883
UTBG	Real estate		Kept	0
VPF	Mutual fund	Ownerless	Kept	0
VPFO	Mutual fund	Ownerless	Kept	0
IKJ	Other non-legal persons	Ownerless	Removed	4
IKJP	Other non-legal persons	Ownerless	Removed	8
KF	Municipal company	Public	Removed	293
KIR	Church council	Public	Removed	3
KIRK	Church council	Public	Removed	2
KTR	Limited partnership	Limited	Kept	0
KTRF	Office	Partners	Kept	0
ORG	Organizational unit	Limited	Removed	10
ORGL	Organizational unit	Limited	Removed	10
SE	Partnership	Partners	Kept	0
SF	State company	Public	Removed	104
SPA	Savings bank	Ownerless	Kept	0
NA/(blank)			Removed	10660

Table A5.1: The legal form of incorporation with description and categories. The two last columns describe if the legal form is kept or removed, and if removed, how many observations are removed.

In table A5.1 we have presented all types of legal forms of incorporation that is included in the original data. Based on the legal form type, we decide to remove a number of them from our finalized data, because this is a leading indicator of the bankruptcy risk for a company, where some types either are improbable to go bankrupt or the company type has different incentive structure than other.

For the different incorporation forms, we have evaluated if the entity is a special purpose vehicle or a special purpose entity. Entities that we believe are defined under this expression are removed, such as BBL (Co Operational Building Association). Furthermore, we have classified the incorporation types based on operations. Entities without any operational parts are removed, such as churches. Lastly, we have evaluated if these incorporation types are in any form related to profits. Our threshold is, therefore, to remove entities without incentives to create a profit or break even. I.e. Municipal and state-owned entities are removed, while e.g. operational non-profit companies are kept. In total, 274 522 observations are removed from the original dataset based on these logical reasons.

A6 Reclassification of Industry Sector Codes

As briefly explained in the main text, there exist two different variables containing two different types of industry sector classification in the original dataset, one standard from 2002, and a new standard from 2007. The reason for this is that the Norwegian Government implemented a new standard for industry sector classification in 2007, reflecting the sector division more accurate, also including new types of businesses. Since this variable is very important regarding which type of industry we examine, and both variables contain NA's, we decided to propose the following solution, resulting in one list of industry sectors.

First, we find the link between the two standards[34], and based on this we compute a new standard list of 17 common industry sectors and codes, reflecting both "old" standards, with an emphasizes on the 2007-standard. This resulted in two new variables, *sector.02* and *sector.07*.

Secondly, updating each variable, we replace NA's with the sector code of the other variable. Hence, the new variable should be complete, including the industry sector for each observation. Lastly, we delete the variable *sector.02*, since we only need one variable of the industry sector.

Below, in Table A6.1 and A6.2, the reclassification is summarized for both standards. A small number of observations, where both variables are NA's, are deleted from the dataset.

[34]Found at Statistics Norway: https://www.ssb.no/virksomheter-foretak-og-regnskap/naeringsstandard-og-naeringskoder.

Industry Sector Classification

Original				Reclassified			
Number	sector.02	From	To	Number	Reclassify To	From	To
1	Primary industries	0	10000	1	Primary industries	0	10000
2	Manufacturing industries	10000	11000	2	Oil/Gas/Mining	10000	15000
		12000	40000	3	Manufacturing industries	15000	22000
3	Oil/Gas	11000	12000	9	Telecom/IT/Media	22000	22200
5	Construction/Energy	40000	50000	3	Manufacturing industries	22200	37000
6	Trade	50000	60000	4	Energy/Water/Sewage/Util.	37000	45000
7	Transport, Tourism	60000	65000	5	Construction & Property Development	45000	50000
8	Shipping	60300	60400	6	Trade	50000	55000
		61100	61200	7	Transport, Tourism	55000	61000
9	Finance, Insurance	65000	70000	8	Shipping	61000	62000
10	Services/Real Estate/Advisors	70000	75000	7	Transport, Tourism	62000	64000
		90000	91000	9	Telecom/IT/Media	64000	65000
11	Health, Care	85000	90000	10	Finance, Insurance	65000	70000
12	Culture, Media	92000	95000	11	Real Estate	70000	71000
13	IT/Telecom	30020	31000	12	General Services	71000	72000
		64200	65000	9	Telecom/IT/Media	72000	73000
		71330	71340	13	Research & Development	73000	74000
		72000	73000	14	Advisors	74000	75000
				15	Public Services	75000	85000
				16	Health, Care	85000	90000
				4	Energy/Water/Sewage/Util.	90000	91000
				17	Culture	91000	92000
				9	Telecom/IT/Media	92000	92200
				17	Culture	92300	93000
				6	Trade	93000	99000

Table A6.1: Reclassification of industry codes from the original variable main.sector.code.02 to our standard in the variable named sector.02.

Industry Sector Classificiation

Original				Reclassified			
Number	sector.07	From	To	Number	Reclassify To	From	To
1	Primary industries	0	5000	1	Primary industries	0	5000
2	Oil/Gas/Mining	5000	10000	2	Oil/Gas/Mining	5000	10000
3	Manufacturing industries	10000	35000	3	Manufacturing industries	10000	35000
4	Energy/Water/Sewage/Util.	35000	40000	4	Energy/Water/Sewage/Util.	35000	40000
5	Construction & Property Development	40000	45000	5	Construction & Property Development	40000	45000
6	Trade	45000	49000	6	Trade	45000	49000
7	Shipping	50000	51000	7	Shipping	50000	51000
8	Transport, Tourism (excl. Shipping)	49000	58000	8	Transport, Tourism	49000	58000
9	Telecom/IT/Media	58000	64000	9	Telecom/IT/Media	58000	64000
10	Finance, Insurance	64000	68000	10	Finance, Insurance	64000	68000
11	Real Estate, Services	68000	69000	11	Real Estate	68000	69000
12	General services (ecxl. R&D)	69000	84000	14	Advisors	69000	72000
13	Research & Development	72000	73000	13	Research & Development	72000	73000
14	Public sector/Culture	84000	-	12	General Services	73000	79000
				8	Transport, Tourism	79000	80000
				12	General Services	80000	84000
				15	Public Services	84000	86000
				16	Health, Care	86000	90000
				17	Culture	90000	96000
				6	Trade	96000	99000

Table A6.2: Reclassification of industry codes from the original variable main.sector.code.07 to our standard in the variable named sector.07.

A7 Variable Treatment

Our finalized dataset consists of 159 variables, where most of them are continuous or discrete accounting data. In addition, some of them are categorical types, i.e. dummy and ordinal. Hence, we need to deal with these types of variables. Below, in Table A7.1 and A7.2, we describe which variables this regards, which type and scale it has and a description of how these variables are dealt with. Regarding the variables *dbrating* and *dbrating.y*[35], the original scale was 0,1,2,3,4,5,9, where 0 is not rated, 1 is rating C, 5 is rating AAA and 9 is categorized as bankrupt/dissolved/liquidated. This variable is not that frequently stated, which means that the majority are not rated. This implies that this variable does not affect the model to a very high degree.

Ordinal and Binary Variable Treatment

Variable Name	Description of variable	Type	Scale	Description			
auditor.remarks	Auditor remark	Ordinal	[1:4]	1 = NA	2 = Acceptable error	3 = Minor critical error	4 = Critical error
auditor	Auditor name	Ordinal	[1:3]	1 = NA	2 = Other	3 = "Five-big" auditor houses[36]	
auditor.change	Change of Auditor	Binary	[0:1]	0 = No change of auditor		1 = Change of auditor	
accountant.change	Change of Accountant	Binary	[0:1]	0 = No change of accountant		1 = Change of accountant	
complex.ownership	Complex Ownership of Company	Binary	[0:1]	0 = Not a complex ownership structure		1 = Complex ownership structure	
parent	Company part of Group	Binary	[0:1]	0 = Is not part of a group		1 = Part of group	
ifrs	IFRS	Binary	[0:1]	0 = Do not use IFRS standards		1 = Use IFRS standards	
listed.shares	Listed securities	Binary	[0:1]	0 = No listed securities or derivates at Oslo Stock Exchange		1 = Have listed securities or derivates at Oslo Stock Exchange	
chair.change	Change of Chairperson	Binary	[0:1]	0 = No change of chairperson		1 = Change of chairperson	
ceo.change	Change of CEO	Binary	[0:1]	0 = No change of chairperson		1 = Change of chairperson	
legal.form	Legal form of Incorporation	Binary	[0:1]	0 = All others	1 = AS, ASA and NUF		
chair.sex	Sex of Chairperson	Binary	[0:1]	0 = Male	1 = Female		
ceo.sex	Sex of CEO	Binary	[0:1]	0 = Male	1 = Female		
dbrating	D&Bs credit rating company	Ordinal	[1:7]	1 = 0 and NA	2 = 5	3=4 4=3	5=2 6=1 7=9
dbrating.y	D&Bs credit rating parent company	Ordinal	[1:7]	1 = 0 and NA	2 = 5	3=4 4=3	5=2 6=1 7=9

Table A7.1: Table of ordinal and binary variables in the dataset, and the associated treatment for each variable, with description.

[35] D&Bs credit rating of (group) company.

[36] "Five-big" auditor houses is a reference to the following companies: BDO, Deloitte, EY, KPMG and PwC.

Dummy Variable Treatment

Variable Name	Description of variable	Type	Scale	New Variable	Description	
region	Region in Norway	Dummy	[1:8]	region.innlandet	1 = Region "Innlandet"	2 = Not region "Innlandet"
				region.nord.norge	1 = Region "Nord-Norge"	2 = Not region "Nord-Norge"
				region.sorlandet	1 = Region "Sorlandet"	2 = Not region "Sorlandet"
				region.trondelag	1 = Region "Trondelag"	2 = Not region "Trondelag"
				region.vest.viken	1 = Region "Vest-Viken"	2 = Not region "Vest-Viken"
				region.vestlandet	1 = Region "Vestlandet"	2 = Not region "Vestlandet"
				region.ostviken	1 = Region "Ostviken"	2 = Not region "Ostviken"
					If all 7 is 2 = Missing value for region (NA)	
legal.form	Incorporation Category[37]	Dummy	[1:4]	incorp.cat.lim.liability	1 = Category of company is limited liability	2 = Not limited liability
				incorp.cat.partnership	1 = Category of company is partnership	2 = Not partnership
				incorp.cat.ownerless	1 = Category of company is ownerless	2 = Not ownerless
					If all three is 2 = Category of company is sole property	
legal.form.mother	Legal form of Group Company	Dummy	[1:3]	legal.form.mother.as	1 = Group company is of legal form "AS"	2 = Not of legal form "AS"
				legal.form.mother.other	1 = Group company is of legal form other than "AS"	2 = Not of legal form other than "AS"
					If both is 2 = Company does not belong to a group	
boardmemb	Number of Board Members	Dummy	[1:4]	boardmemb.1	1 = Board consist of 1 member	2 = Not 1 member
				boardmemb.2.3	1 = Board consist of 2 or 3 members	2 = Not 2 or 3 members
				boardmemb.4+	1 = Board consist of 4 or more members	2 = Not 4 or more members
					If all three is 2 = Missing value for observation (NA)	
sector.07	Industry Sector of Company	Dummy	[1:18]	sector.07.advisory	1 = Company industry sector is advisory	2 = Not advisory
				sector.07.construction	1 = Company industry sector is construction	2 = Not construction
				sector.07.culture	1 = Company industry sector is culture	2 = Not culture
				sector.07.energy	1 = Company industry sector is energy	2 = Not energy
				sector.07.general	1 = Company industry sector is general	2 = Not general
				sector.07.health	1 = Company industry sector is health	2 = Not health
				sector.07.manufacturing	1 = Company industry sector is manufacturing	2 = Not manufacturing
				sector.07.oil	1 = Company industry sector is oil	2 = Not oil
				sector.07.primary	1 = Company industry sector is primary	2 = Not primary
				sector.07.public	1 = Company industry sector is public	2 = Not public
				sector.07.realstate	1 = Company industry sector is realestate	2 = Not realestate
				sector.07.rnd	1 = Company industry sector is rnd	2 = Not rnd
				sector.07.shipping	1 = Company industry sector is shipping	2 = Not shipping
				sector.07.it	1 = Company industry sector is it	2 = Not it
				sector.07.trade	1 = Company industry sector is trade	2 = Not trade
				sector.07.transport	1 = Company industry sector is transport	2 = Not transport
				sector.07.finance	1 = Company industry sector is finance	2 = Not finance
					If all 17 is 2 = Missing observation (NA)	
sector.07.y	Industry Sector of Group Company	Dummy	[1:3]	sector.07.primsec.y	1 = Group company is either in primary or secondary industry sector	2 = Not in primary or secondary
				sector.07.tert.y	1 = Group company is in tertiary industry sector	2 = Not in tertiary
					If both is 2 = Company does not belong to a group	

Table A7.2: Table of dummy variables and how they are treated, i.e. creation of the dummies and description of what each stands for.

[37] Incorporation category original consist of 5 categories, but in this case, we simplify and categorizes public sector as ownerless.

A8 Correlation Matrix

Correlation matrix	oper.result	Fin.rev	RTE	Fin.assets	Tot.assets	Other rec	Invest	PC	FJSG	LTPPE	FIP	STLFI	STLIMAX
nopat	0.9895	0.2403	0.8813	0.0704	0.0687	0.0063	0.0577	0.0572	0.0707	-0.0036	0.0579	0.0556	0.0397
tax.nopat	0.997	0.103	0.7986	-0.0014	-0.0016	0.0001	0.0055	-0.0011	-0.0014	-0.0013	-0.0014	-0.0015	-0.0022
result.to.equity	0.8128	0.6268	1	0.2812	0.2749	0.0245	0.2099	0.2294	0.2822	-0.01	0.2319	0.2232	0.1628
gross.interest.income	0.9657	0.1965	0.8248	0.0179	0.021	0.0459	0.0002	0.0073	0.018	0.0192	0.0138	0.0125	0.0323
transitory.financial.gain	-0.002	0.9778	0.5396	0.0781	0.0777	0.0136	0.3887	-0.0015	0.0797	0.0059	-0.0008	0.0005	0.0015
Fixed assets	0.0123	0.0095	-0.0102	0.0182	0.1854	0.3225	-0.0019	0.0875	0.0175	1	0.0501	0.1045	0.577
Fin.assets	0.0727	0.0867	0.2812	1	0.9859	0.0258	0.1099	0.953	1	0.0182	0.9842	0.972	0.7584
Tot.assets	0.0735	0.0872	0.2748	0.9859	1	0.0793	0.1077	0.9514	0.9857	0.1854	0.9757	0.9726	0.8418
net.other.claims	0.0709	0.0408	0.2499	0.9758	0.9613	0.0193	0.1071	0.939	0.9758	0.0117	0.9576	0.9472	0.7285
other.st.assets	0.0062	0.0205	0.0245	0.0253	0.0796	0.9886	0.0357	0.1134	0.0251	0.327	0.0361	0.0231	0.3561
current.assets	0.0138	0.3701	0.2273	0.196	0.1989	0.1099	0.9785	0.207	0.1958	0.0369	0.201	0.2047	0.1754
pension.liabilities	0.0718	-0.001	0.2283	0.953	0.9514	0.0809	0.1068	1	0.9526	0.0875	0.9544	0.9413	0.7467
other.provisions	0.0044	0.0031	0.0115	0.0076	0.0484	0.9717	-0.0012	0.0567	0.0076	0.2454	0.0043	-0.0015	0.3186
other.lt.liabilities	0.061	0.0061	0.1628	0.7584	0.8418	0.3608	0.0814	0.7467	0.7582	0.577	0.7587	0.7967	1
long.term.liabilities	0.0619	0.0058	0.1671	0.7699	0.8514	0.3753	0.0827	0.765	0.7696	0.5669	0.7713	0.8048	0.9991
fin.assets.jv.sub.group	0.0727	0.0886	0.2822	1	0.9857	0.0256	0.1099	0.9526	1	0.0175	0.9836	0.9717	0.7582
st.marketbased.securities	0.007	0.3766	0.2097	0.1089	0.1067	0.0363	1	0.1058	0.1089	-0.0019	0.1093	0.1091	0.0806
lt.ppe.operating	0.0123	0.0095	-0.0102	0.0182	0.1854	0.3225	-0.0019	0.0875	0.0175	1	0.0501	0.1045	0.577
financial.invest.pension	0.0727	0.0004	0.2319	0.9842	0.9757	0.035	0.1102	0.9544	0.9836	0.0501	1	0.9706	0.7587
lt.provisions.commitments	0.0055	0.0032	0.0148	0.008	0.056	0.9606	-0.0013	0.0735	0.008	0.2868	0.0057	-0.0018	0.3154
lt.operational.liabilities	0.0611	0.0062	0.163	0.7576	0.8409	0.3642	0.0811	0.7455	0.7573	0.5771	0.7575	0.7954	0.9991
st.liab.to.financial.inst	0.0728	0.0022	0.2233	0.972	0.9726	0.0236	0.1101	0.9413	0.9717	0.1045	0.9706	1	0.7967
net.cashflow.oper	0.9786	0.1993	0.8467	0.0501	0.0507	0.0226	0.0408	0.0404	0.0503	0.0083	0.041	0.0401	0.0357
net.cashflow.invest	0.9869	0.0423	0.7543	-0.0316	-0.0401	-0.0021	-0.0035	-0.0272	-0.0316	-0.0547	-0.0298	-0.0273	-0.0506
net.change.cash	0.9839	0.1258	0.8067	0.0351	0.0331	0.0211	0.0786	0.0346	0.0352	-0.0087	0.0325	0.0341	0.0288
tot.ib.liab.max	0.0659	0.0124	0.1862	0.8389	0.9081	0.2782	0.0905	0.8285	0.8385	0.4997	0.8442	0.872	0.9785
number.of.shareholders	0.0731	-0.001	0.23	0.9897	0.9742	0.0165	0.1107	0.9546	0.9895	0.0102	0.9834	0.9734	0.7488

Table A8.1: Correlation matrix for most correlated variables. Abbreviations: RTE - Result to equity, PC - Pension Commitment, FJSG - fin.assets.jv.sub.group, LTPPE - lt.ppe.operating, FIP - financial.invest.pension, STLFI - st.liab.to.financial.inst, STLIMAX - ib.st.liab.max.

A9 Support Vector Machines Hyperparameter Tuning

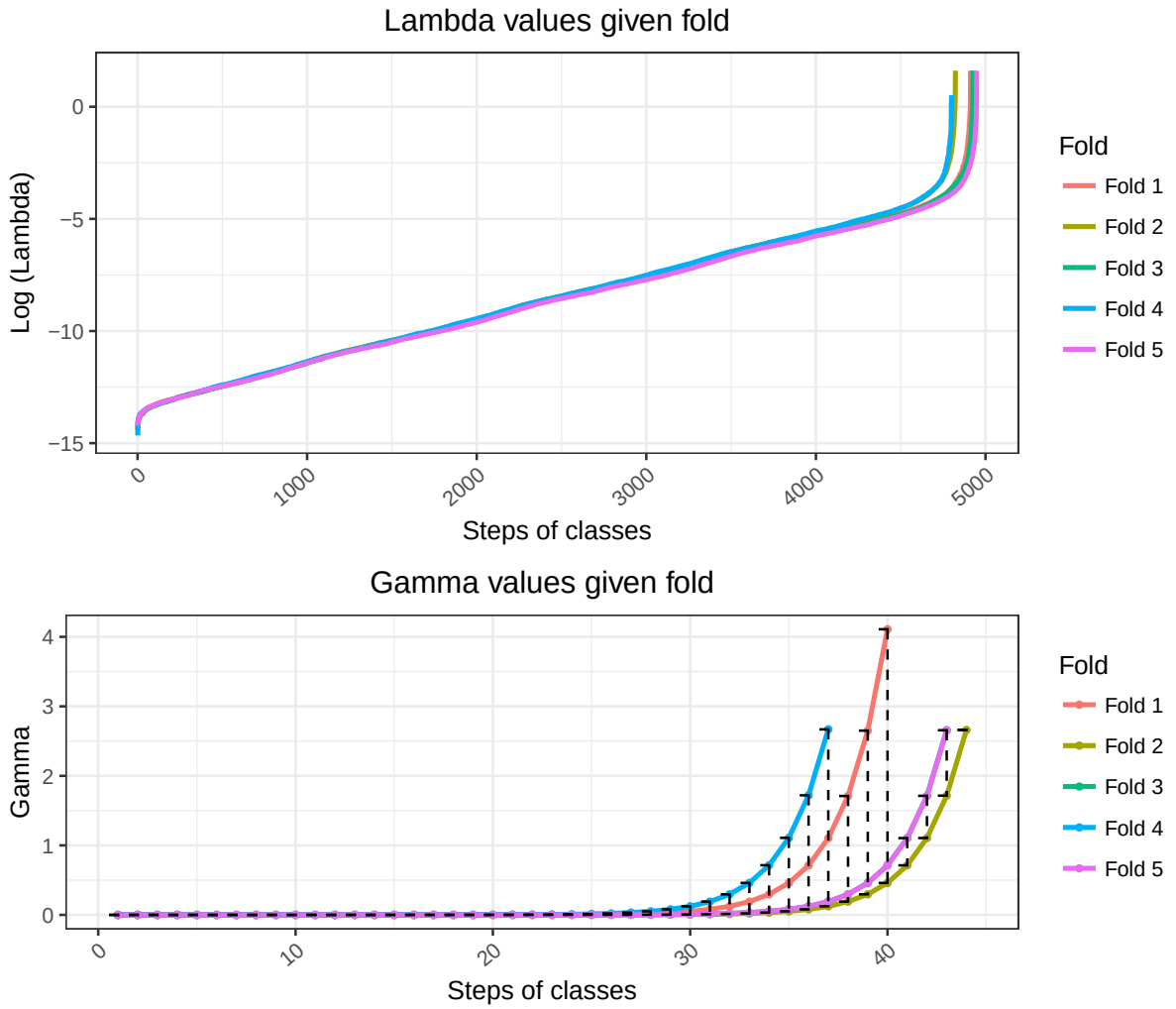

Figure A9.1: λ and γ values for different folds for SVM under a cost scheme of 0.1,1,10,100 that is scaled for each fold. We can observe that the parameters are within reasonable ranges. Difficult to interpret the stand-alone values because SVM decides the optimal values to maximize predictions. The grid is shown in Table 4.1.

As mentioned in 3.2.5, the adaptive grid controls the hyperparameters lambda (λ) and gamma (γ) based on the properties of the data, and therefore the whole scale is used on a pro-rata basis. In Figure A9.1, we observe the fitted and scaled tuning parameters for the different folds. The hyperparameter values are fitted on a Gaussian radial basis function kernel where the λ is kernel bandwidth and γ is the regulator that has the purpose of avoiding under/overfitting the model.

A10 Neural Network Hyperparameter Tuning

In this section of the appendix, we explain the tuning process of the hyperparameters learning rate (γ_r), momentum (m), number of hidden nodes (Z_m) and number of hidden layers (l) for Neural Network. Based on previous literature on NN and research, we started the tuning process with a test for m, with γ_r = 0.05, Z_m = 128 and l = 1. Z_m was selected based on the second method presented in Equation 3.25, with an α value of 5. Table A10.1 shows that the optimal AUROC and TPR is achieved with an m value of 0.7. Hence, for the next parameter tuning for Z_m and l, we use m = 0.7. Part two of the process is explained under section 4.2, where figure 4.4 displays the result of the testing. There, we find that the optimal number of hidden nodes and layers are Z_m = 9 and l = 1.

In the third step of our tuning process, we back-test for the correct m and γ_r, with the optimal values for Z_m and l. The results of this iterative process are displayed in Figure A10.1, displaying the overall accuracy, TPR and AUROC values for the tests performed. Interpreting the results, we find that there are two solutions that give, approximately the same, best performance with respect to AUROC and TPR. These are the combination γ_r = 0.3 and m = 0.9 and the combination γ_r = 0.9 and m = 0.2. Hence, for the fourth, and final step of the tuning process, we back-test these two combinations for the number of hidden nodes again, to ensure that the tuning is optimal. The results are displayed in Table A10.2 and show that the optimal solution is the same as found in step three. Hence, our final hyperparameter selection for NN is displayed in Table A10.3.

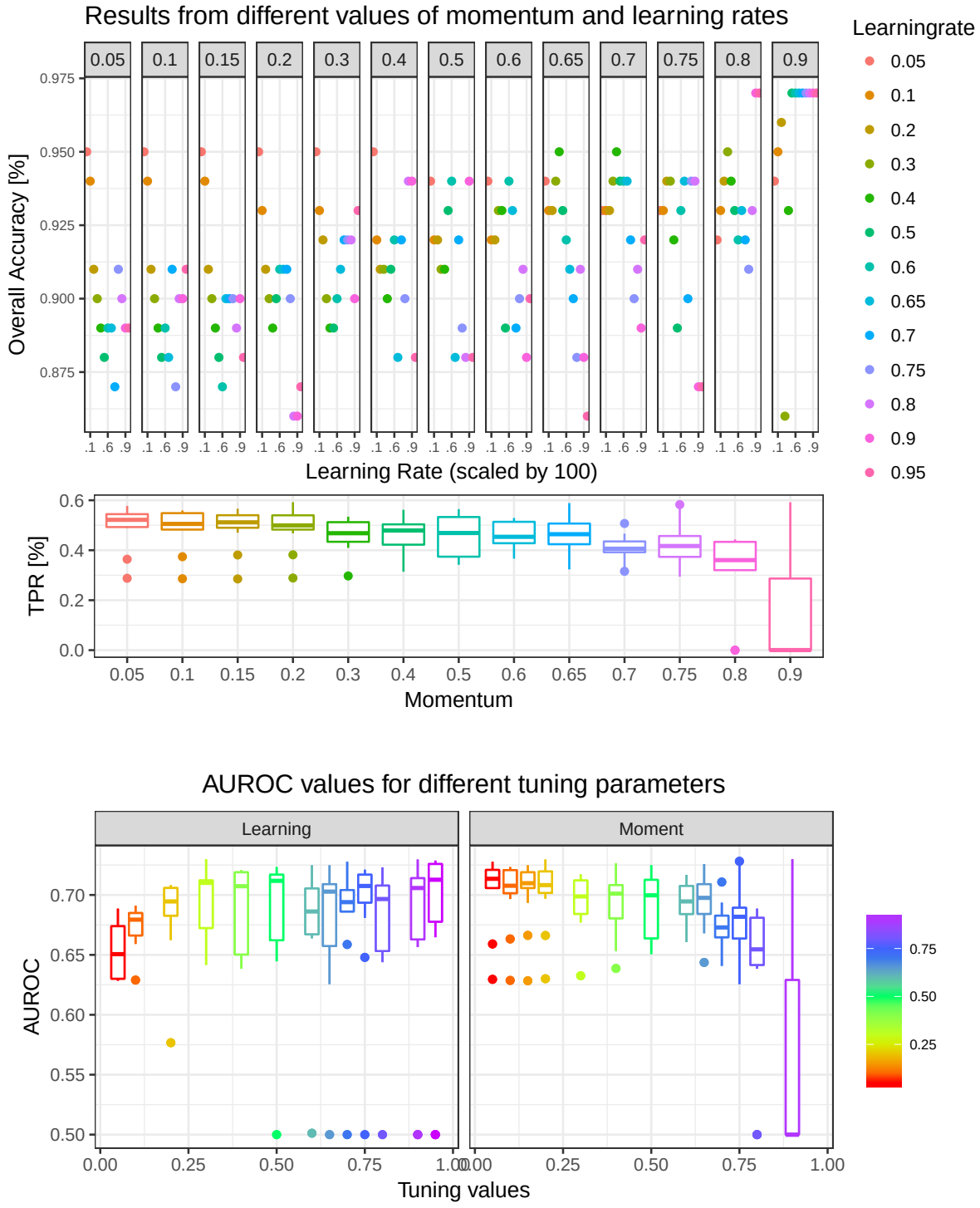

Figure A10.1: Tuning of the Neural Network.

Momentum Test			
Z_m = 128, l = 1, γ_r= 0.05			
m	Accuracy	AUROC	TPR
0.05	0.9455	0.5826	0.1984
0.10	0.9456	0.5845	0.2020
0.20	0.9459	0.5842	0.2013
0.30	0.9467	0.5868	0.2057
0.40	0.9432	0.6224	0.2827
0.50	0.9410	0.6429	0.3271
0.60	0.9420	0.6499	0.3405
0.65	0.9425	0.6512	0.3427
0.70	0.9388	**0.6572**	**0.3590**
0.75	0.9395	0.6389	0.3206
0.80	0.9485	0.6193	0.2707
0.90	0.9472	0.5672	0.1648
0.95	0.9457	0.5712	0.1746

Table A10.1: Momentum test for Neural Network.

Final test of the number of hidden nodes, Z_m						
l = 1	γ_r = 0.3 and m = 0.9			γ_r = 0.9 and m = 0.2		
Z_m	Accuracy	AUROC	TPR	Accuracy	AUROC	TPR
8	0.9536	0.6144	0.2552	0.8705	0.7260	0.5730
9	0.8601	**0.7298**	0.5918	0.8590	0.7296	**0.5925**
10	0.8972	0.6904	0.4714	0.9052	0.7049	0.4927
11	0.9241	0.6521	0.3640	0.8976	0.7192	0.5303
12	0.8900	0.7130	0.5256	0.8863	0.7210	0.5459
13	0.8891	0.7111	0.5227	0.9180	0.6928	0.4544
14	0.9282	0.6563	0.3684	0.8855	0.7216	0.5480
15	0.9328	0.6632	0.3778	0.8850	0.7228	0.5509
16	0.9431	0.6443	0.3279	0.8771	0.7235	0.5607
64	0.9277	0.6790	0.4157	0.9402	0.5971	0.2339
128	**0.9612**	0.5809	0.1782	0.9383	0.5828	0.2064

Table A10.2: Final test for the number of hidden nodes for Neural Network.

Final Selection	
Parameter	Value
Z_m	9
l	1
γ_r	0.3
m	0.9

Table A10.3: Final hyperparameter values for Neural Network.

A11 Random Forest Tree

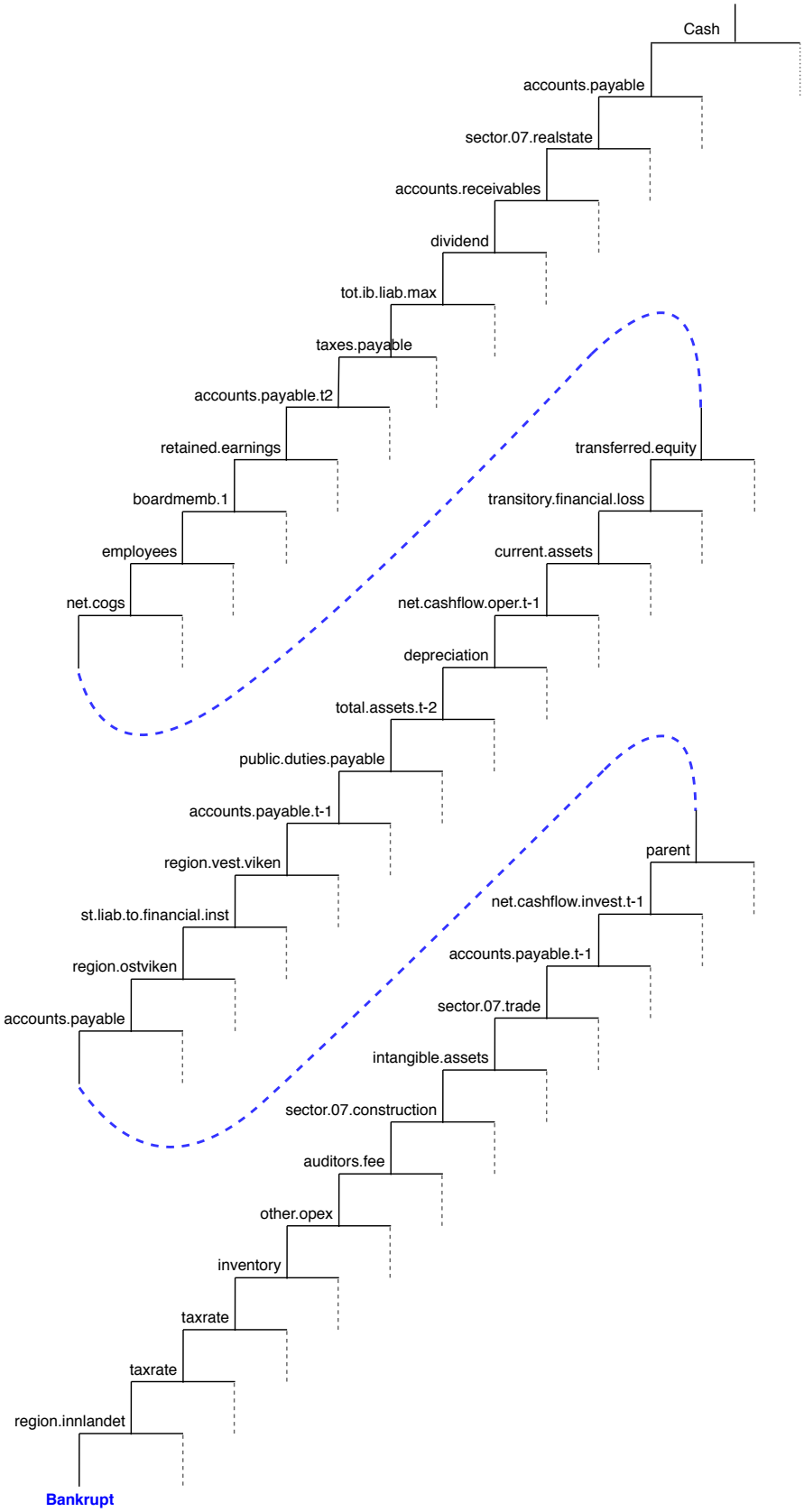

Figure A11.1: A standalone Random Forest tree.

In Figure A11.1 we observe a standalone Random Forest tree. This is the first of the 17 250 trees. Due to the large trees we have focused only on the right hand leaf of the tree. The blue dotted lines shows where the connection in the tree. Another observation we note from the figure is that the same variable are included several times, due to the random shuffling of the features.

For the interested reader the splitting rules of the tree in Figure A11.1 are as follows:

cash $<$ -0.0075 & accounts.payable $>$ -0.0055 & sector.07.realstate $>$ 1.5 & accounts.receivables $<$ -0.0025 & dividend $<$ -0.0145 & tot.ib.liab.max $<$ -0.0115 & taxes.payable $<$ -0.0075 & accounts.payable.t-2 $>$ -0.1095 & retained.earnings $>$ -0.0105 & boardmemb.1 $<$ 1.5 & employees $<$ -0.009 & net.cogs $<$ 0.0055 & transferred.equity $>$ -0.0045 & transitory.financial.loss $<$ -0.0125 & current.assets $<$ -0.0045 & net.cashflow.oper.t-1 $>$ -5e-04 & depreciation $<$ -0.0115 & total.assets.t-2 $>$ -0.0075 & public.duties.payable $<$ -0.0305 & accounts.payable.t-1 $<$ -0.1065 & region.vest.viken $>$ 1.5 & st.liab.to.financial.inst $<$ -0.0025 & region.ostviken $>$ 1.5 & accounts.payable $<$ -0.0045 & parent $<$ 0.5 & net.cashflow.invest.t-1 $>$ -0.098 & accounts.payable.t-1 $>$ -0.1075 & sector.07.trade $>$ 1.5 & intangible.assets $<$ -0.007 & sector.07.construction $>$ 1.5 & auditors.fee $<$ -0.026 & other.opex $>$ -0.0205 & inventory $<$ -0.0065 & taxrate $<$ 0.269 & taxrate $<$ 0.25657895 & region.innlandet $>$ 1.5 $=>$ Bankrupt

A12 Variable Importance

As presented in section 5.3, we test for the importance of variables using coefficients, importance and weighting values of the final optimized model of LDA, MDA, GLM, GAM, RF and NN. Unfortunately, its not possible to extract any sensible values for the remaining models; QDA, KNN and SVM. After extracting the values, we perform three different tests on each model.

For the first approach, all the values are normalized to a *mean of zero* for easy comparison, using Equation 4.1. Then we use a hard-decision threshold of +/- 1 to determine if the variable should be selected as important for the model at hand. I.e., if the normalized value is less than -1 or larger than +1, the variable is selected and given a vote of *1* for each time it is selected. Finally, all the variables with at least one vote (and the number of votes) are collected for use in the final variable selection. Table A12.1 displays a sample of variables from the test.

Normalization with a mean score of zero

	GLM	GAM	LDA	MDA	RF	NN
Std	1.18E+17	1.59E+16	1.10E+14	0.3876	0.1132	12.8684
Mean	1.45E+15	1.10E+14	0.3876	0.1132	7.9238	2.2430
taxrate	-0.0122	-0.0069	-0.0071	-0.0027	**2.6337**	**1.42647**
paid.tax.rate	-0.0122	-0.0069	-0.0071	-0.0028	**1.3195**	**3.6499**
auditor.remarks	-0.0123	-0.0151	-0.0222	-0.0144	**1.2420**	**-5.3702**
auditor	-0.0122	-0.0105	-0.0054	-0.0021	0.3379	0.3806
auditor.change	-0.0123	-0.0045	-0.0092	-0.0041	-0.2658	-0.5732
intangible.assets	-0.0118	-0.0177	-0.0072	0.0031	-0.3197	0.1250
other.st.liabilities	0.0057	-0.6968	-0.5235	0.3890	-0.4349	-0.1246
net.cashflow.oper	**1.0199**	0.8429	**1.0020**	0.1335	0.2878	-0.4153
net.cashflow.invest	**8.7047**	**7.1796**	**8.5152**	**1.1585**	-0.7821	-0.1712
deferred.tax.liabilities	-0.0158	0.0229	-0.0078	0.0117	-0.7117	0.0603
other.provisions	-0.8418	**-1.9839**	-0.0493	**9.1391**	-0.8788	-0.1720

Table A12.1: Sample of normalized coefficient scores with a mean of zero. Variables viewed as important are bolded (score of +/- 1).

For the second approach, we again make use of a normalization approach, but in this case, we use the *max-min* approach:

$$z_i = \frac{x_i - min(x)}{max(x) - min(x)} \tag{.2}$$

For this approach we use a hard-decision threshold which is set to +/- 0.6. Otherwise we follow the same procedure as in the first approach. A sample from the analysis is drawn in Table A12.2.

For the third approach, we simply use a top/bottom rule for the values for each model. I.e., we

Normalization with Max-Min

	GLM	GAM	LDA	MDA	RF	NN
Max	1.03E+18	1.14E+17	461.57	379.74	64.65	49.21
Min	-1.02E+18	-1.1E+17	-458.95	-333.12	0.035	-66.86
taxes.payable	0.4989	0.4968	0.4980	0.4677	0.1044	**0.6144**
public.duties.payable	0.4989	0.4977	0.4985	0.4674	0.5910	0.5020
short.term.liabilities	0.4970	**0.6500**	0.5450	0.4512	0.0458	0.5807
ib.st.liab.max	0.4787	0.3642	0.4207	0.4385	0.0612	0.5753
tot.ib.liab.max	0.5231	**0.6345**	**0.6023**	0.5053	0.0937	**0.6152**
sector.07.trade	0.4994	0.5049	0.4984	0.4669	0.1100	**0.8966**
lt.provisions.commitments	0.5304	**0.8021**	0.4970	0.0000	0.0009	0.5763
total.income	0.4989	0.5131	0.4988	0.4672	0.2626	**0.6400**
net.cogs	0.4988	0.4859	0.4984	0.4674	0.2322	**0.6104**
complex.ownership	0.4992	0.4987	0.4987	0.4672	0.0029	**0.7174**
other.provisions	**-0.8418**	**-1.9839**	-0.0493	**9.1391**	**-0.8788**	-0.1720

Table A12.2: Sample of normalized coefficient scores with the max-min approach. Variables viewed as important are bolded (score of +/- 0.6).

find the top 10 and bottom 10 values, in all 20 values and the corresponding variable for each model. For each time a variable is observed, it is given a vote of 1. To conclude the approach, we count the number of times each variable is selected in a table, saving it for the final step.

In the final step, after all three approaches are finalized, we merge the three tables together, and summarizes the number of times each variable is selected. Hence, the maximum number a variable can be selected is 18 (3x6). The merger and summation is presented in Table A12.3. Then, using a hard-decision threshold of *2*, meaning that only variables selected two or more times, we reach our final estimate of 57 variables that are viewed as important across the models. These 57 variables and their corresponding score of importance were neatly presented in section 5.3, in Table 5.14. The entire table of all variables and their score is presented in this section, in Table A12.3.

We have chosen not to analyze the inference because our best performing models do not yield coefficients that are easy to interpret. The *coefficients* from NN and Random Forest are weights and importance, respectively, which are not directly comparable to the coefficients of the other models. However, by performing the aforementioned rankings, we still are able to get some insight from the selected variables, and we choose to focus on this ranking instead.

Variable Importance In Optimized Models

Variable	Mean	Max-Min	Top/Bottom	Sum	Variable	Mean	Max-Min	Top/Bottom	Sum
taxrate	2	1	2	5	sector.07.general	0	0	0	0
paid.tax.rate	2	1	1	4	sector.07.health	1	0	1	2
auditor.remarks	2	0	1	3	sector.07.manufacturing	1	1	1	3
auditor	0	1	0	1	sector.07.oil	0	0	0	0
auditor.change	0	0	0	0	sector.07.primary	0	0	0	0
accountant.change	0	0	0	0	sector.07.public	0	0	0	0
dbrating	2	1	2	5	sector.07.realstate	1	0	1	2
ifrs	0	0	1	1	sector.07.rnd	0	0	0	0
listed.shares	0	0	1	1	sector.07.shipping	0	0	0	0
legal.form	1	0	1	2	sector.07.it	0	0	0	0
largest.ownershare	1	1	0	2	sector.07.trade	1	1	1	3
chair.change	1	0	1	2	sector.07.transport	1	1	1	3
chair.sex	0	0	0	0	total.income	1	1	1	3
ceo.change	0	0	0	0	net.cogs	0	1	1	2
ceo.sex	0	0	0	0	personnel.expense	1	1	1	3
complex.ownership	1	1	0	2	depreciation	0	1	0	1
parent	1	1	0	2	impairments	0	0	0	0
dbrating.y	0	0	0	0	exp.bad.debt	0	0	0	0
region.innlandet	0	1	0	1	other.opex	1	1	1	3
region.nord.norge	1	1	0	2	finance.expense	0	0	0	0
region.sorlandet	0	0	0	0	nopat	1	1	0	2
region.trondelag	0	1	0	1	tax.nopat	0	0	0	0
region.vest.viken	0	0	0	0	xo.revenue	0	0	0	0
region.vestlandet	0	1	0	1	xo.costs	0	0	0	0
region.ostviken	0	0	0	0	xo.tax	0	1	0	1
incorp.cat.lim.liability	1	1	1	3	result.to.equity	2	1	2	5
incorp.cat.partnership	0	0	1	1	dividend	1	1	0	2
incorp.cat.ownerless	0	0	1	1	transferred.equity	2	1	1	4
boardmemb.1	1	1	1	3	transferred.reserve	0	0	0	0
boardmemb.2.3	1	1	0	2	group.contribution	0	0	0	0
boardmemb.4+	0	1	0	1	revenue.fininvest.jv	0	1	0	1
sector.07.primsec.y	1	0	1	2	gross.interest.income	0	1	0	1
sector.07.tert.y	1	0	1	2	transitory.financial.gain	0	0	0	0
legal.form.mother.as	0	1	0	1	writedown.financials	0	0	0	0
legal.form.mother.other	0	1	0	1	gross.interest.expense	1	0	0	1
sector.07.advisory	0	0	0	0	transitory.financial.loss	0	0	0	0
sector.07.construction	1	1	1	3	deferred.tax.assets	0	0	0	0
sector.07.culture	0	0	0	0	total intangible.assets	0	1	0	1
sector.07.energy	0	0	0	0	other.fixed.receivables	0	0	0	0
inventory	1	0	0	1	dividends.payable	0	1	0	1
net.other.claims	0	0	0	0	other.st.liabilities	0	0	4	4
accounts.receivables	0	0	3	3	net.cashflow.oper	2	0	4	6
total.receivables	0	0	1	1	net.cashflow.invest	4	3	4	11
cash	0	1	0	1	net.cashflow.finance	0	0	4	4
other.st.assets	0	0	3	3	net.change.cash	4	0	4	8
current.assets	0	0	4	4	ib.st.liab.max	2	0	4	6
total.assets	1	0	3	4	tot.ib.liab.max	2	3	4	9
paid.in.equity	0	1	0	1	employees	2	0	2	4
retained.earnings	0	1	0	1	year.foundation	0	0	0	0
other.equity	0	0	1	1	number.of.shareholders	0	0	0	0
total.equity	0	1	2	3	general.manager.pay	1	0	2	3
pension.liabilities	0	1	0	1	directors.fee	0	0	0	0
deferred.tax.liabilities	0	1	0	1	auditors.fee	1	0	1	2
other.provisions	2	1	4	7	advice.auditor.fee	0	0	0	0

Variable	Mean	Max-Min	Top/Bottom	Sum	Variable	Mean	Max-Min	Top/Bottom	Sum
it.commitments	0	0	0	0	result.to.equity.y	0	1	0	1
provisions.for.commit	2	0	5	7	total.assets.y	0	1	0	1
subordinated.loan.cap	0	1	0	1	total.equity.y	0	1	0	1
other.lt.liabilities	0	1	4	5	result.to.equity.t-1	1	0	0	1
long.term.liabilities	1	0	4	5	result.to.equity.t-2	1	0	0	1
convertible.loans	0	0	0	0	total.assets.t-1	0	0	0	0
accounts.payable	1	0	0	1	total.assets.t-2	0	0	0	0
taxes.payable	0	1	0	1	deferred.tax.assets.t-1	0	0	0	0
public.duties.payable	1	0	1	2	deferred.tax.assets.t-2	0	1	0	1
short.term.liabilities	1	1	3	5	accounts.payable.t-1	1	1	0	2
total.liabilities	0	0	3	3	accounts.payable.t-2	0	1	0	1
total.equity.liabilities	1	0	4	5	short.term.liabilities.t-1	0	1	0	1
fin.assets.jv.sub.group	0	1	0	1	short.term.liabilities.t-2	0	1	0	1
st.marketbased.securities	0	0	5	5	total.liabilities.t-1	0	0	0	0
intangible.assets	0	1	0	1	total.liabilities.t-2	0	0	0	0
lt.ppe.operating	0	1	0	1	total.equity.liabilities.t-1	0	0	0	0
financial.invest.pension	0	0	2	2	total.equity.liabilities.t-2	0	0	0	0
derivatives.and.other	0	0	0	0	net.cashflow.oper.t-1	1	0	0	1
lt.provisions.commitments	2	1	4	7	net.cashflow.oper.t-2	1	0	0	1
bond.loans	0	0	0	0	net.cashflow.invest.t-1	0	1	0	1
lt.operational.liabilities	1	1	1	3	net.cashflow.invest.t-2	0	0	0	0
st.alterative.loans	0	1	0	1	net.change.cash.t-1	0	0	0	0
st.liab.to.financial.inst	0	0	1	1	net.change.cash.t-2	0	1	0	1

Table A12.3: Summation of the final variable selection of importance. Abbreviation lt and st denominates long-term and short-term. Liab represents liabilities, subfix ".y" indicates that the row is related to the group/parent. t-1 and t-2 denominate lagged variables of 1 and 2 years, respectively.

A13 Computer Specifications

Our analysis was completed using two virtual machines with the following specifications:

R version 3.5.1 (2018-07-02)

Running under: Windows Server 2012 x64 (build 9200)

Intel(R) Xeon(R) CPU X7550 @ 2.00 GHz, 1998 Mhz, 8 cores, RAM: 48 GB

Intel(R) Xeon(R) CPU X7550 @ 2.00 GHz, 1998 Mhz, 4 cores, RAM: 36 GB