Using Neural Networks and Support Vector Machines for Default Prediction in South Africa

by
FRANCES MELTZER

A thesis submitted to the Faculty of Computer Science and Applied Mathematics,
University of Witwatersrand,
in fulfillment of the requirements for the
Master of Science (MSc)

Johannesburg
Feb 2017

ABSTRACT

This is a thesis on credit risk and in particular bankruptcy prediction. It investigates the application of machine learning techniques such as support vector machines and neural networks for this purpose. This is not a thesis on support vector machines and neural networks, it simply looks at using these functions as tools to preform the analysis.

Neural networks are a type of machine learning algorithm. They are nonlinear models inspired from biological network of neurons found in the human central nervous system. They involve a cascade of simple nonlinear computations that when aggregated can implement robust and complex nonlinear functions. Neural networks can approximate most nonlinear functions, making them a quite powerful class of models.

Support vector machines (SVM) are the most recent development from the machine learning community. In machine learning, support vector machines (SVMs) are supervised learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. SVM takes a set of input data and predicts, for each given input, which of two possible classes comprises the input, making the SVM a non-probabilistic binary linear classifier. A support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite dimensional space, which can be used for classification into the two different data classes.

Traditional bankruptcy prediction medelling has been criticised as it makes certain underlying assumptions on the underlying data. For instance, a frequent requirement for multivarate analysis is a joint normal distribution and independence of variables. Support vector machines (and neural networks) are a useful tool for default analysis because they make far fewer assumptions on the underlying data.

In this framework support vector machines are used as a classifier to discriminate defaulting and non defaulting companies in a South African context. The input data required is a set of financial ratios constructed from the company's historic financial statements. The data is then Divided into the two groups: a company that has defaulted and a company that is healthy (non default). The final data sample used for this thesis consists of 23 financial ratios from 67 companies listed on the jse. Furthermore for each company the company's probability of default is predicted.

The results are benchmarked against more classical methods that are commonly used for bankruptcy prediction such as linear discriminate analysis and logistic regression. Then the results of the support vector machines, neural networks, linear discriminate analysis and logistic regression are assessed via their receiver operator curves and profitability ratios to figure out which model is more successful at predicting default.

DECLARATION

I declare that this dissertation is my own unaided work. It is being submitted for the degree of MSC at the University of Witwatersrand, Johannesburg, South Africa. It has not been submitted before for any degree or examination at any other University.

_____
Frances Meltzer

_____ day of _____, 2017.

ACKNOWLEDGEMENTS

I wish to express sincerest gratitude to

- My supervisor Antonie Kotze without whose endless wisdom and support I could not have completed this thesis.
- My husband Alan for his constant love and support.
- My beautiful baby boy and beautiful baby girl who brightens up my day everyday.

Contents

List of Figures

Chapter 1

Aim and Introduction

1.1 Aim

The sudden global financial crisis in 2008, where financial market liquidity dried up, caused numerous companies with historically strong financial standing to go out of business because they were caught off guard and could not meet their financial obligations. Globally the recent bankruptcies of large joint stock companies in U.S. and Europe, such as Enron, WorldCom, Lehman Brothers, Swissair and Parmalat shook investors across the world and underlined the importance of failure prediction modelling both in academia and industry.

The demise of these companies, which not only had extremely large balance sheets but also came from a developed economy, highlighted the importance of bankruptcy prediction modelling. In developed markets, the number of financially distressed companies and among them the number of bankrupt companies is fewer than emerging markets (Aktan, 2011a). This further shows the importance of a bankruptcy prediction model in an emerging market such as South Africa which is a key motivation for this paper. A successful failure prediction model will help understand the reasons behind the collapse of a firm, as this will enable timely bankruptcy preventative action to be taken as precaution. This thesis looks at developing a failure prediction model for South African countries. It is a thesis on credit risk that utilises certain machine learning techniques to preform the analysis.

An example of a JSE listed company to enter into financial distress is African Bank (ABL). ABL is South Africas biggest provider of unsecured loans and was part of the

FTSE/JSE Top 40 Index[1]. In May 2013 ABL wrote off R445m of non-preforming loans. Other JSE listed companies to have financial distress include 1Time Airlines, VOX Telecoms and Uranium One Inc.

To predict whether a company will become bankrupt or reach financial distress is not only significant to everyone involved in the running of the company including company management, auditors, creditors, customers, suppliers and employees. It is relevant to the investors in the company for both private and publicly listed companies.

Over the last decade, a number of the worlds largest banks have developed sophisticated models in an attempt to model counterparty credit risk. The outputs of these models also play increasingly important roles in banks risk management and performance measurement processes, including performance-based compensation, customer profitability analysis, risk based pricing, active portfolio management and capital structure decisions (Basel, 1999).

The environment of credit and the circumstances of lending have changed substantially in the last two decades. Traditional banking, which used to be reliant more on intuition and experience to judge good and bad customers is being sidelined for more analytical methods. Similarly the traditional accounting analysts are being replaced by analysts with solid knowledge not only of accounting and finance, but also of other related areas, such as statistics, forecasting, data mining and econometrics (Minussi et al., 2006).

The New Basel Capital Accord (Basel II/III) was introduced to allow banks to reduce unexpected losses, improve profitability and increase risk-carrying capacity. The Basel II/III framework is a set of guidelines issued by the Basel Committee on Banking Supervision; it contains recommendations on banking laws and regulations. Basel II/III sets international standards on how much capital a bank must hold in order to protect itself against expected losses. Banks must have a credit scoring model to validate the accuracy of estimated probabilities of default but that Basel II does not specify what type of model. This puts further importance and emphasis on credit scoring models because an accurate model is needed to set aside the correct amount of capital (Minussi et al., 2006). Therefore there is more of a need for people with quantitative backgrounds and a tougher grounding in Probability of Default(PD), Loss Given Default(LGD), Exposure at Default(EAD), VAR and Expected Tail Loss Models. (on Banking Supervision, 2016a) (on Banking Supervision, 2016b) (on Banking Supervision, 2014).

Corporate default prediction has long been important and studied widely by financial

[1]The FTSE/JSE Top 40 Index consists of the largest 40 companies ranked by full market value in the FTSE/JSE All-Share Index

literature under the name of bankruptcy prediction, loan default risk, default prediction, credit analysis, failure prediction, credit risk and financial distress prediction. This subject involves developing models that attempt to forecast financial failure before it actually happens. Bankruptcy prediction models use appropriate independent variables to predict whether a company (or individual) is financially healthy or bankrupt. Financial ratios are constructed from the companys financial statements. These financial ratios are used as the independent variables for the bankruptcy prediction models.

The literature on bankruptcy prediction can be classified into four broad approaches: discriminate analysis, structural form models (contingent claims analysis), reduced-form or intensity-based approach and modern quantitative credit risk modelling (Georgakopoulos, 2004) (Altman et al., 2014).

The first bankruptcy prediction studies date back to the 1930's beginning with the initial studies concerning the use of single ratio analysis to predict future bankruptcy. These studies used individual ratios and would compare ratios of bankrupt companies to those of non-bankrupt companies. These single ratio studies were known as univariate studies. The univarite studies laid the foundation for multivariate bankruptcy prediction models. Beaver (1966) is one of the earliest researchers to study the prediction of credit risk. Beavers analysis is the most widely recognised form of univariate analysis, which includes studying one financial ratio at each time and deciding a cut-off threshold for every financial ratio.

After the mid-1960's bankruptcy prediction studies were focused on multivariate discriminate analysis. In 1968, Altman published the first multivariate discriminate study in which financial ratios can be used for classification (Altman, 1968b). There have been many further studies on multivariate discriminate analysis of different permutations of Altmans study. Studies on multivariate discriminate analysis remain popular even in modern times. After multi discriminate analysis the logit and probit models for predicting bankruptcy were developed. The logit model is in a way a generalised Discriminate Model, because it does not assume multivariate normality and equal covariance matrices. Structural form models employ modern option pricing theory in corporate debt valuation. The original framework developed by Merton (1974) using the principles of option pricing formulae, is the cornerstone for all other structural models.

Unlike in the structural default framework, the reduced form approach does not consider in an explicit way to model the relation between default and the companys underlying value. The reduced form approach depends on market prices of deflectable derivatives and other market indicators and not simply on balance sheet information.

Around the late 1980s more modern methods for default prediction were developed in

order to improve prediction accuracy. These methods are still active today. These include machine learning techniques, one of the artificial intelligence techniques (Hardle, 2010). Machine learning has become a common alternative for credit scoring models and has achieved higher accuracy than traditional statistical methods. Machine learning is a form of artificial intelligence that provides the algorithm the ability to learn without being explicitly programmed. It analyses patterns in the underlying data in order to extract information from a data set in order to make predictions about future events. These models are non-parametric techniques for the prediction, so they can overcome some constraints of the traditional statistical models.

Traditional bankruptcy prediction modelling has been criticized as it makes certain underlying assumptions on the underlying data. For instance, a frequent requirement for multivarate analysis is a joint normal distribution and independence of variables. Support vector machines and neural networks are a useful tool for default analysis because they make far fewer assumptions on the underlying data.

Before developing the model, the study posed the question whether newer modelling techniques such as support vector machines and neural networks were a better tool for predicting bankruptcy than traditional modelling techniques such as LDA analysis and logit analysis in a South African context with limited data availability. For this, the hypotheses was raised and tested:

$H1 : Support\ Vector\ Machines\ and\ Neural\ Networks\ predict\ default\ with\ a\ higher\ accuracy\ than\ LDA\ and\ LogitAnalysis$

Due to lack of data when analysing the research question $H1$ a different research question was posed: $Will\ it\ be\ possible\ to\ answer\ this\ question\ with\ the\ data\ availabe\ for\ analysis$?

This raised another hypothesis: $H2 : The\ data\ available\ is\ sufficient\ to\ predict\ default\ 2\ years\ prior\ to\ default$

Further analysis regarding small data sets led to another research question: $Does\ feature\ selection\ increase\ the\ algorithm's\ accuracy$?

Translating into another hypothesis test:

$H3 :\ Feature\ selection\ increases\ the\ algorithm's\ accuracy$

It is with these three hypotheses $H1$, $H2$, $H3$ that the study is constructed.

Neural networks are a type of machine learning algorithm. They are nonlinear models inspired from biological network of neurons found in the human central nervous system. They involve a cascade of simple nonlinear computations that when aggregated

can implement robust and complex nonlinear functions. Neural networks can approximate most nonlinear functions, making them a quite powerful class of models. They have a wide range of uses including forecasting, classification and statistical pattern recognition. Research studies on using neural networks for default prediction started in 1990, and are still active now. Neural networks have generally outperformed the other existing methods (Atiya, 2001).

Support vector machines (SVM) are the most recent development from the machine learning community. In machine learning, support vector machines (SVMs) are supervised learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. SVM takes a set of input data and predicts, for each given input, which of two possible classes comprises the input, making the SVM a non-probabilistic binary linear classifier.

All the mathematical modelling for this thesis is preformed in R-Studio using R Studio's built in functions. In order to correctly implement these functions the theory is needed for all functions used. This is provided in Chapter 6, Chapter 7 and Chapter 8.

There has been a lot of research on support vector machines and neural networks for bankruptcy prediction in developed countries. Most studies have shown that both support vector machines and neural networks predict bankruptcy with a higher accuracy than traditional methods. There has been less research done on this topic in a South African Context, in particular the research on Support vector machines in a South African Context is minimal.

Previous studies have focused on developed economies; therefore there are few studies that have been conducted for developing countries, which is a key motivation for this paper. The research findings from developed economies may not be suitable for application in South Africa due to differences in market structures, socio-economic factors, politics, legal frameworks and accounting standards (Hlahla, 2010a).

1.2 Thesis Overview

This thesis is organised as follows: The remainder of this chapter will describe the data used for the analysis as well as a literature review for neural networks and support vector machines applied to bankruptcy prediction. Chapter 2 is an overview of bankruptcy prediction modeling in Developing and Frontier Countries including South Africa. It also describes The Basel II/III framework and the IRB Approach. Chapter 3 describes the definition of "Default" applied to this thesis as the definition

of "Default" will effect the classification output of the algorithm. Chapter 4 provides an overview of the different types of default prediction models. It describes the different approaches for default prediction models as well as highlighting the main advantages and disadvantages of each method. Chapter 5 is a history of bankruptcy prediction modelling. It discusses a few of the most pioneering works for Multivariate Scoring Models and Structural Form Models that have been cited the most in previous Bankruptcy literature. Chapter 6 is a description of machine learning. It describes the implementation of general machine learning algorithms as well as different methods of interpreting the results. Chapter 7 and Chapter 8 describe the theory behind neural networks and support vector machines. This theory is needed in order to correctly implement the algorithms. Chapter 9 explores the advantages and disadvantages of support vector machines and neural networks as well as providing a comparison of neural networks and support vector machines. Chapter 10 describes feature selection and it's importance as well providing a summary of the different feature selection methods. It goes on to describe the feature selection method used for this thesis. Chapter 11 is a chapter on validation and Chapter 12 is the results and Chapter 14 is the conclusion.

1.3 Data

The input data for the default prediction models described above is a set of financial ratios constructed from the companys historic financial statements. The data is then divided into the two groups: a company that has defaulted and a company that is healthy (non default).

It needs to be mentioned, that great difficulty in obtaining any information on defaulted companies for both public (JSE listed) companies and private companies in South Africa. Attempts to get information from banks, liquidators, private equity firms, micro lenders, small cap analysts, hedge funds and banks all proved futile with cross referrals to each other.

In developed countries such as the USA information on financial distress for companies is more readily available and can be easily obtained from places such as the Wall Street Journal Index (Altman et al., 2014).

The difficulty in obtaining financial data for defaulted South African Companies was common in default prediction literature in South Africa. Naidoo (2006b) and Hlahla (2010a) experienced similar frustrations obtaining the correct financial data. Most of the prior research on default prediction has been on JSE listed companies, see Chapter 2 Section 2.3.2 for a summary of the data used in each study.

This thesis develops a prediction model for company default for JSE listed companies using historic data from companies which defaulted from 2008 to 2014. Obtaining the historic information for companies defaulting prior to 2008 was very difficult and was not available on the most prevalent trading platforms used for stock analysis in South Africa, Bloomberg and Inet Bridge. The JSE was contacted directly. A database of the full set of financial information for companies defaulting prior to 2008 did not exist. The financials were only available in soft copy, kept in storage. The only way to obtain the financials was to create the database by hand, which would take longer than the length it took to write this thesis.

The final list of companies in financial distress was composed by researching the companies on the internet and speaking to people in industry (JSE, trading companies and banks). The raw data was extracted from Bloomberg using Bloomberg Microsoft Excel Addin. The raw data included information from the income statement, balance sheet and cash flow statement for companies from 2008 to 2014. The data was filtered, cleaned and put in the correct format for analysis in Microsoft Excel.

Figure 1.1 is a list of the companies used in this thesis that are classified as being in financial distress. For a company to be classified as falling into financial distress, one of the following conditions as stated in Chapter 3 Section 3.3 need to have occurred. Included in this Table is the year the company became in financial distress as well as the reason for the financial distress.

Long Name	Short Name	Reason for Financial Distress	Year of Financial Distress
1time Holdings Ltd	1tm	Liquidated	2012
Pals Holding	PAL	Delisted - Financial distress	2009
Square One Solutions Group	SQE	Liquidated - Cash Flow problems	2010
Seakay Holdings	SKY	Provisional Liquidation	2012
Brikor	BIK	Provisional Liquidation	2013
Sanyati Holdings Ltd	SAN	Liquidated due to Corruption	2013
Johnnic Holdings Ltd	JNC	Voluntary Liquidation/ Corruption	2007
Jasco	JSC	Financial Distress	2013
VOX Telecoms	VOX	Debt written off	2008
Racec	Rac	Financial Distress	2011
Esorfranki Ltd	ESR	Rights Offer Prevent Default	2011
First Uranium Corp	fuu	Sale of Assets to prevent Liquidatuion	2012
Uranium One Inc	uuu	Financial Distress; large losses	2013
Best Cut	BCH	Financial Distress	2010
Beget Holdings Ltd	BEE	Liquidation	2011
Pamonzi Gold	PZG	Provisional Liquidation	2010
Super Group Ltd	SPG	Voluntary Liquidation	2010
Murray and Roberts	MUR	Issue Righs prevent Liquidation	2012
Ellerine Holdings	ELR	Financial Distress/ Large Debt	2011
Murray and Roberts	MCU	Issue Righs prevent Liquidation	2010
Platmin Ltd	PPN	Financial Distress/ Voluntary Business Re	2011
Simmer & Jack Mines Ltd	SIM	Liquidation	2011
Sentula Mining Ltd	SNU	Sentula Mining Ltd	2013
Tiger Wheel	TIW	Financial Distress/ Liquidation	2010
Wesizwe Platinum Ltd	WEZ	Liquidation	2012

Figure 1.1: Final List of Companies that are in Financial Distress

Figure 1.1 is the final list of healthy companies used in this thesis. They are all the members of the Top40 Index in January 2014.

Long Name	Short Name	Reason	Year
African Rainbow Minerals Ltd	ARI	Top 40 Member	2014
Anglo American Platinum Ltd	AMS	Top 40 Member	2014
Anglo American PLC	AGL	Top 40 Member	2014
AngloGold Ashanti Ltd	ANG	Top 40 Member	2014
Aspen Pharmacare Holdings Ltd	APN	Top 40 Member	2014
Assore Ltd	ASR	Top 40 Member	2014
Barclays Africa Group Ltd	BGA	Top 40 Member	2014
BHP Billiton PLC	BIL	Top 40 Member	2014
Bidvest Group Ltd	BVT	Top 40 Member	2014
British American Tobacco PLC	BTI	Top 40 Member	2014
Capital & Counties Properties	CCO	Top 40 Member	2014
Cie Financiere Richemont SA	CFR	Top 40 Member	2014
Discovery Ltd	DSY	Top 40 Member	2014
Exxaro Resources Ltd	EXX	Top 40 Member	2014
FirstRand Ltd	FSR	Top 40 Member	2014
Growthpoint Properties Ltd	GRT	Top 40 Member	2014
Impala Platinum Holdings Ltd	IMP	Top 40 Member	2014
Imperial Holdings Ltd	IPL	Top 40 Member	2014
Intu Properties PLC	ITU	Top 40 Member	2014
Investec Ltd	INL	Top 40 Member	2014
Investec PLC	INP	Top 40 Member	2014
Kumba Iron Ore Ltd	KIO	Top 40 Member	2014
Life Healthcare Group Holdings Ltd	LHC	Top 40 Member	2014
Mediclinic International Ltd	MDC	Top 40 Member	2014
Mondi Ltd	MND	Top 40 Member	2014
Mondi PLC	MNP	Top 40 Member	2014
MTN Group Ltd	MTN	Top 40 Member	2014
Naspers Ltd	NPN	Top 40 Member	2014
Nedbank Group Ltd	NED	Top 40 Member	2014
Old Mutual PLC	OML	Top 40 Member	2014
Remgro Ltd	REM	Top 40 Member	2014
RMB Holdings Ltd	RMH	Top 40 Member	2014
SABMiller PLC	SAB	Top 40 Member	2014
Sanlam Ltd	SLM	Top 40 Member	2014
Sasol Ltd	SOL	Top 40 Member	2014
Shoprite Holdings Ltd	SHP	Top 40 Member	2014
Standard Bank Group Ltd	SBK	Top 40 Member	2014
Steinhoff International Holdings Ltd	SHF	Top 40 Member	2014
Tiger Brands Ltd	TBS	Top 40 Member	2014
Truworths International Ltd	TRU	Top 40 Member	2014
Vodacom Group Ltd	VOD	Top 40 Member	2014
Woolworths Holdings Ltd/South Afr	WHL	Top 40 Member	2014

Figure 1.2: Final List of Healthy Companies

1.4 Support vector machines and Neural Networks Literature Review

In recent years literature neural networks and support vector machines have been applied to bankruptcy prediction problems. There have been many studies comparing support vector machines and neural networks with the older statistical methods as described above. The reason for this is in part methodological and partly because neural networks and support vector machines are the latest machine learning methods, which are increasingly applied in other disciplines in addition to bankruptcy prediction.

Multivariate statistical approaches in practice often pose unachievable criteria in respect of data subject to analysis. One instance of this is that many statistical techniques require a joint normal distribution and independence of variables.

Recently, SVMs have gained popularity due to many attractive features and generalization performance on a wide range of problems. SVMs have become a focus of interest for failure prediction and the applications of SVMs into financial failure prediction began in 2005. SVM are relatively new and are used less frequently in financial literature.

1.4.1 Support vector machines

The most popular traditional techniques for bankruptcy prediction include statistical methods like linear discriminant analysis (DA) and logit or probit models. There have been many studies comparing SVM's to these techniques. Below is a summary of these studies. In most of the studies below the discriminative power of the models is measured on the basis of their accuracy ratio (AR) and percentage of correctly classified observations.

- Min and Lee (2005) proposed support vector machines for bankruptcy prediction. They compared the support vector machines and neural networks in terms of prediction accuracy. They concluded that the support vector machines outperformed neural networks for the training and validation data based on prediction accuracy of both models.

- Aliakbari (2009) did a comparison of support vector machines and logit regression for bankruptcy prediction. He concluded that logit regression out preforms support vector machines. Analysis was based on comparing the accuracy ra-

tios for each model. Furthermore results showed that profitability and leverage indicators have discriminating power in bankruptcy prediction.

- Wu et al. (2006) examined bankruptcy predictive accuracy in five statistics models discriminant analysis: logistic regression, probit regression, neural networks, support vector machines, and genetic-based support vector machines (the genetic algorithm for variable selection). Refer to chapter 10 for a discussion on the different methods for variable selection. Empirical results indicate that the support vector machine based models are superior models for predicting financial failure. The accuracy ratio was used as a predictor of model accuracy.

- Razvan-Alexandru (2009) did a comparison of support vector machines and logit analysis. The results indicate that the support vector machines performed better than the logit analysis. In the comparison the accuracy ratio was used as main indicator of performance.

- Virag and Nyitrai (2013) applied support vector machines and neural networks to the database of Hungarian banks in order to predict bankruptcy. Results were then compared with the results of earlier bankruptcy models. A crossvalidation was preformed with the average classification accuracy and the average area under the roc curve as the main indicators of performance. The results of the models assembled suggest that there was a significant improvement of classification accuracy achieved using support vector machines. Furthermore support vector machines showed a higher accuracy ratio than neural networks.

- Chen (2011) compared some traditional statistical methods for predicting financial distress to some more 'unconventional' methods, such as decision tree classification, neural networks, and support vector machines. Analysis was done on 200 taiwan firms. He revealed that support vector machines provided a good balance of high-accuracy short and long-term performance predictions for healthy and failed firms. Results were based on a performance measures including accuracy ratio, precision, true positive rate and true negative rate.

- Dikkers (2005) applied linear regression, ordinal logistic regression, and support vector machine techniques for default prediction. Findings were that support vector machines can significantly outperform other techniques like linear and logistic discriminate analysis. Results were based on mcnemars test, a statistic based on the classification table.

- Einarsson (2008) compared support vector machines to more traditional statistical techniques for bankruptcy prediction. Findings were that logistic regression was a better predictor of bankruptcy than support vector machines. Analysis was based on the statistic auc.

- Hardle et al. (2007) compared support vector machines with more traditional approaches such as discriminant analysis and logit regression. The results demonstrate that the support vector machines has clear advantages over the other methods. Results were based on the accuracy ratio (ar), which was used as a criterion for model selection.

- Wang et al. (2013) apply support vector machines to discriminate between healthy and defaulting companies. The particle swarm optimization algorithm is used to optimize parameters of the support vector machines. In particle swarm optimization a number of simple entitiesthe particlesare placed in the search space of some problem or function, and each evaluates the objective function at its current location (R Poli and Blackwell, 2007). The analysis was preformed on manufacturing companies listed on shanghai and shenzhen stock markets. Result were that article partial swarm optimization with support vector machines has a distinct improvement to support vector machines and neural networks with out partial swarm optimization. Results were based on the accuracy ratio.

- Huang et al. (2004) apply support vector machines and neutral networks to the problem of bankruptcy prediction. Analysis was preformed on two data sets for taiwan financial institutes and united states commercial banks. The results showed that support vector machines achieved accuracy comparable to that of neural networks based on the accuracy ratio.

- Huang et al. (2007) compared support vector machines (with the genetic algorithm for variable selection) with neural networks, genetic programming, and decision tree classifiers for credit scoring. A ten fold cross validation was done. Results were that support vector machines achieved an identical classification accuracy with relatively few input features when compared to the other models, based on the accuracy ratio.

- Hardle et al. (2011) analyzed credit default risk for firms in the Asian and pacific region by applying support vector machines and a logistic regression. Among the different financial ratios used as predictors of default leverage ratios and the company size display a higher discriminating power compared to others. Variable selection was performed using the forward selection procedure. The accuracy ratio was used as a predictor for model accuracy. Results were that support vector machines have a lower model risk (higher accuracy) than the logit analysis.

- Auria and Moro (2008) compared of the support vector machines with more traditional approaches such as logistic regression and discriminant analysis. Analysis was done on the Deutsche Bundesbank data of annual income statements and balance sheets of German companies. The discriminative power of the

models is measured on the basis of their accuracy ratio (AR) and percentage of correctly classified observations. Results were that Support vector machines outperformed both discriminate analysis and logit analysis.

- Mahdi (2013) compared support vector machines and the genetic algorithm for bankcrupcy prediction. Results were that support vector machines predicted default with higher accuracy.

1.4.2 Neural Networks

There is a significant amount of research on neural networks applied for business forecasting and prediction. A summary of comparisons of neural networks with other classification methods approaches with the application of business forecasting and prediction can be seen in Adya and Collopy (1998). Another review of neural network approaches for bankruptcy prediction post the 1990's is provided by Atiya (2001)

A few of the more significant studies will be described below:

- several of the major commercial loan default prediction products are based on neural networks. For instance moodys public firm risk model [2]is based on neural networks as the main technology Atiya (2001).
- Kim et al. (1993): compared the neural network approach with linear regression, discriminant analysis, logistic analysis, and a rule-based system for bond rating. Results were that neural networks achieved better performance than other methods in terms of classification.
- Altman et al. (1994): preformed a study comparing neural networks to discriminate analysis. The best results were from a three layer neural network. Neural networks were a better predictor than discriminate analysis for company distress prediction.
- Al-Osaimy (1998) uses neural networks for predicting islamic banks performance. His findings were that neural network methods show significant promise for providing useful early warning signals for various types of performance including bankruptcy, insolvency and failure.

Many researchers in bankruptcy forecasting report that neural networks produce significantly better prediction accuracy than classical statistical techniques (Zhang et al., 1999).

[2]moodys public firm model has been designed to act as an early warning system to monitor changes in the credit quality of corporate obligors

Chapter 2

General Default Prediction

2.1 Introduction

Both in academia and industry, people have shown a great interest in the prediction of bankruptcy and financial distress. It is a very deep and well researched field of study. The bulk of this research has been done on companies that have been traded in developed markets, in particular the US. There is much less evidence of bankruptcy and financial distress prediction in developing countries, in particular how the existing models perform on samples of companies from developing markets as well as whether different predictors are needed (Charalambakis and Garrett, 2015)

In this thesis the term developing countries will be used to describe any country that is not highly developed. Highly developed countries can be measured by the Human Development Index (HDI, 2009). According to the index the highly developed countries are Australia, Switzerland, Netherlands, United States, Germany, New Zealand, Canada, Singapore, Denmark, Ireland, Sweden, Iceland, United Kingdom, Korea, Hong Kong and Japan. A full list of countries rated by the Human Development Index can be seen in (HDI, 2009).

The characteristics of developing countries are (Damodaran, 2009):

- Currency volatility - In many emerging markets, the local currency is volatile, both in terms of exchange rate and inflation.
- Country risk - There is substantial growth in emerging market economies, but this growth is accompanied by significant macroeconomic risk.

- Information gaps and accounting differences- While information disclosure requirements have become more stringent globally, the rules still require that much less information be disclosed in emerging markets than in developed markets.

- Corporate governance - Many emerging market companies used to be family-owned businesses and while they might have made the transition to being publicly traded companies, the families retain control through a variety of devices – shares with different voting rights, pyramid holdings and cross holdings across companies.

- Discontinuous risk - In some emerging markets, there is an added layer of risk that can cause sudden and significant changes in a firm's fortunes. Included here would be the threat of nationalisation or terrorism or political risk.

A frontier market is a type of developing country which is more developed than the least developing countries, but too small to be generally considered an emerging market (Warburton, 2017)

2.2 Developed Countries

There is a wealth of evidence of the ability of models to predict bankruptcy in developed markets and in particular the US. The research ranges from the seminal works of Beaver (1966) and Altman (1968a) using multivariate scoring models to the later more modern prediction models using neural networks and support vector machines. A summary of this is found in Section 6.

2.3 Developing and Frontier Countries

Most of the default prediction research has been in highly developed countries. This section will cover the default prediction models in developing countries (emerging markets and frontier markets).

2.3.1 History of default prediction models in developing countries

There have been numerous studies on default prediction for developing countries. As far back as 2005 Altman developed a scoring model for emerging corporate bonds known as the Emerging Market Score Model (EMS Model) (Altman, 2005). The EMS Model is an enhancement of the Altman Z-Score Model. The EMS Model can be used to score and rate corporate bonds based on Altman initial Multivariate Model. The model first rates the bond on Altman's initial model and then modifies it based on the companys vulnerability to currency devaluation, industry affiliation and competitive position in the industry when rating the bond.

Lin (2010) develops a two stage hybrid model using logistic regression and neural networks to develop a default prediction model for companies in the banking sector in Emerging Market Countries. The analysis was performed on financial ratios from data from the banks from Turkey, China, Pakistan, Taiwan, Indonesia, India, Russian, South Africa, Hong Kong, Thailand and Mexico. Results were that the model could accurately predict default in the banks from emerging markets. The hybrid model has a higher accuracy than a regular neural network model.

Triandafil and Brezeanu (2010) did a study on default prediction in Emerging Markets vs. Developed Markets. A multivariate linear regression was applied to companies from both emerging markets and developed markets. Results showed that companies in developed countries took on a higher leverage than countries in emerging markets.

Further bankruptcy prediction models for developing countries including Thailand, Malaysia, India, Kenya, Tunisia, Russia and Taiwan will be described below.

- Thailand - Sirirattanaphonkun and Pattarathammas (2012) apply multivariate discriminant analysis and logistic regression analysis to small medium enterprises (SMEs) in Thailand.
- Malaysia - Yap et al. (2010) and Yap et al. (2013) used multivariate discriminant analysis for default prediction on Malaysian Countries. Findings in both studies were that the multivariate discriminant analysis predicted default with a high accuracy. Yap et al. (2012) investigates the ability of logistic regression in anticipating corporate failures in Malaysia. Results again showed default prediction with a high accuracy.
- India - Abhishek and Jayesh (2012) use multiple discriminant analysis do develop a bankruptcy prediction model for companies listed on the National Stock Exchange India (NSE). Results showed default prediction with a high accuracy.

Purohit et al. (2012) develop a default prediction model for assessing loan applications using logistic regression, decision trees, neural networks and support vector machines. findings indicate that support vector machines, decision tree and logistic regression are the best methodology for classifying the loan applications.

- Nigeria - Pam (2013) applies multivariate discriminate analysis to listed Nigerian companies.

- Kenya - Mamo (2011) applies multivariate discriminate analysis to commercial banks in Kenya. Findings were that the discriminate model predicted default accurately.

- Tunisia - Yap et al. (2012) develop a default prediction model for assessing corporate credit loans in a Tunisian bank. A neural network model and a logistic regression were used for default prediction. Results were that the neural network model produced superior results.

- Russia - Sharma and Shebalkov (2013) use neural networks to analyze and to predict the performance of Russian Banks. Results were that neural networks could accurately predict failure in Russian Banks.

- Taiwan - Chen and Chen (2011) develop a prediction model using support vector machines on Taiwanese listed companies. Results showed that default was predicted with a high accuracy.

2.3.2 History of Default Prediction Models in South Africa

South Africa is a developing country but since the focus of this thesis is on default prediction in South Africa a section has been dedicated to the existing research on default prediction in South Africa. A summary of South African Research on default prediction is provided below.

De La Rey's 1981 multivariate K-score model is South Africa's most well-known failure prediction model (Reeves, 2001) (Naidoo, 2006a). De La Rey was inspired by the work of Altman applying the Z-score to financial statement analysis. He developed a model for bankruptcy prediction in South Africa by applying multivarite discriminate analysis to JSE listed companies. This was known as the K-Score. The K-score model has been used on private companies as well as publically listed companies in South Africa.

The K-Score is constructed from the discriminate function is:

$$-0.01662A + 0.0111B + 0.0529C - 0.086D0.0174E + 0.01071F - 0.06881 \quad (2.1)$$

Where:
A - Total outside financing Total assets.
B - Earnings before interest and tax Average total assets.
C - Total current assets and listed investments Total current liabilities.
D - Earnings after tax Average total assets.
E - Net cash flow Average total assets.
F - Stock Inflation adjusted total assets.

A k-score smaller than -0.19 implies potential failure, with a k-score larger than 0.20 implying a Healthy company and a zone of ignorance exists between a score of -0.19 and +0.20 implying that a company cannot be classified as either healthy or a candidate for potential failure.

Muller (2008) assessed the accuracy of multiple discriminant analysis, logit analysis, recursive partitioning and neural networks to predict financial distress. Analysis was done on companies listed on the Johannesburg Stock Exchange(JSE). Muller reached the conclusion that each technique produces a different predictive accuracy. Multiple discriminate analysis and recursive partitioning techniques correctly predict the most number of failed companies. But that logit analysis and neural networks provide the best overall predictive accuracy.

Rama (2012) did an analysis on whether the Altman (1968a) failure prediction model is effective in predicting the failure of South African companies. Analysis was done by using the Altman Z-Score model on JSE listed companies. Findings were that The Altman failure prediction model can be used as a tool to predict company failure in South Africa, however the model has certain limitations being that the model is not accurate when Z scores are negative or if Z-Scores are in an area of uncertainty.

Tshitanga (2010) analyzed the cost of financial distress in South Africa on JSE listed companies. He evaluated the cost of financial distress by using the Altman Z-Score Model.

Naidoo (2006a) developed a two stage approach to identifying (first stage) and analyzing (second stage) the Financial Health of a company. In the first stage macroeconomic variables were used to determine a Business Failure Rate. In the second stage three different types of models were used for prediction purposes: a Naive model

using the simple Share Holder Value Added (SVA) ratio [1], a Multiple Discriminant Analysis (MDA) model and a Chi-squared Automatic Interaction Detection(CHAID) model. Analysis was performed on JSE listed companies. He concluded that both the Nave and CHAID models produced overall superior results to the more complicated MDA statistical method.

Jacobs (2007) investigated the accuracy of prediction models when applied to South African non-listed companies. Analysis was done on non-listed companies in South Africa. Data was obtained from the Coface Database. The Coface database was provided by The Coface Group, an international Credit Provider. Findings were that the Altman-Z Score Model can be effectively applied to South African Non-Listed Companies.

Cort and Radloff (1993) developed a failure prediction model for South African listed companies using both macro and micro economic variables. A two stage failure prediction model was developed. In the first stage regression analysis was performed on fourteen macroeconomic variables, four macro-economic variables were found to explain 95 percent of the variability of failure rate. In the second stage, discriminant analysis was used on twenty financial and non-financial variables.

Reeves (2001) applied two international models and two South African Models to predict the probability of default on JSE listed companies. The international models applied were Beaver's univariate model and Altman's Z-score model. The South African models were De La Rey's K-score model and Amiras, Aston and Cohen's A-ratio model. The A-Ratio model preforms multiple regression analysis on 17 financial ratios. Reeves concluded that Beaver, De La Rey and the A-ratio models were good predictors on bankruptcy in South Africa. The Altman model should not be applied to local samples as the results of this model had a higher failure rate.

Boltman (2009) and Moonasar (2007) investigated the suitability and accuracy of artificial neural networks (ANNs) and logistic regression respectively as a tool for consumer credit defaults in South Africa.

Senoto (2012) investigated predicting the probability of default in South Africa using macro and micro economic variables. He uses the KMV Merton model applied to JSE Listed companies to predict the probability of default and then uses a Factor Model to analyse the relationship between macro-economic variables and probability of default. Findings were that the KMV Merton Model can be used to predict probability of default in South Africa and furthermore probability of default is correlated to Macro economic trends.

[1]The SVA ratio represents a company's worth to shareholders in the absence of liabilities and capital costs.

Hlahla (2010a) used multivariate discriminate analysis as well as univariate analysis as a bankruptcy prediction tool for South African Companies. Analysis was performed on JSE Listed Companies. Findings were that univarite analysis can be used as a default predictor on JSE Listed companies and that failure can be directly linked to size of the companies listed on the JSE. The multivariate discriminate analysis proved to be able to predict corporate bankruptcy on JSE Listed companies with an accuracy ratio greater than 70 percent.

Kidane (2004) investigated the ability of the Altman Z-Score models in predicting financial distress in Information Technology and Service sector in South Africa. Analysis was preformed on JSE Listed Companies in the information technology sector and in the service sector. Results were that the Altman model is reasonably accurate to classify the failed companies correctly but was unsuccessful in classifying the non-failed companies correctly.

Maeteletsa and Kruger (1993) investigated if the time period a bankruptcy prediction model was developed has any effect on the accuracy of the model. Analysis was done by applying the Altman Z-Score Model on JSE listed companies. Results were that classification accuracy of the bankruptcy prediction model is not affected by the period the model was created in.

Arron and Sandler (1995) investigated the use of neural networks for bankruptcy prediction in South Africa by comparing the accuracy of neural networks to multiple discriminate analysis and logistic regression. Analysis was done on JSE Listed companies. Findings were that the predictions of neural networks and logistic regression were more accurate than MDA Analysis.

Strebel and Andrews (1977) applied Beavers univariate analysis model to South African Businesses. Findings were that the Cashflow to Total Debt ratio was a powerful predictor of corporate failure in South Africa. Furthermore findings showed that this ratio reflected problems four years before corporate failure. The power of the cashflow to debt ratio to predict bankruptcy was consistent with Beaver's findings.

Daya (1977) applied Beavers univariate analysis model to South African Businesses. He found that Cash flow to Average Total Current Liabilities ratio to be the best predictor of failure in the short term but the best overall predictor over a longer period was Net Income to Total Assets.

The pioneering failure prediction work of Beaver (1966) univariate model, and Altman (1968a) multivariate model, provided the platform for the development of local South African models, most notably De La Rey's (multivariate) K-Score model (1981) and Daya (1977). Predominantly most of the South African models have been applying multivariate discriminate analysis to South African Companies, in particular to com-

panies listed on the JSE. There have been some studies applying neural networks for probability of default prediction but there have been very limited research on support vector machines for default prediction in a South African context.

2.4 Basel II/III

The Basel II/III framework is a set of guidelines issued by the Basel Committee on Banking Supervision, it contains recommendations on banking laws and regulations. Basel II/III sets international standards on how much capital a bank must hold in order to protect itself against expected losses. While it is never possible to know in advance the losses a bank will suffer in a particular year, a bank can forecast the average level of credit losses it can reasonably expect to experience. These losses are referred to as Expected Losses (EL).

The amount of capital is calculated from the Expected Loss (EL). The EL is the value of a possible loss times the probability of that loss occurring. Basel II does not specify what type of model should be adopted by banks to calculate the EL and the probability of that loss (Minussi et al., 2007). Instead banks (and investment firms are permitted to calculate capital requirements on the basis of internally produced bankruptcy prediction models subject to certain minimum requirements and agreement by the relevant member state regulator. This is known as the Internal Ratings Based Approach (IRB).

Under the IRB approach, banks must assign banking-book exposures into one of six broad classes of exposures with different underlying credit risk characteristics: corporates, sovereigns, banks, retail, project finance, and equity (Basel, 2001).

2.4.1 Risk Parameters for the IRB Approach

There are three risk components or factors within the IRB Approach to corporate, bank, and sovereign exposures, which build off the structure of banks rating systems. These are (Basel, 2005):

- Probability Of Default (PD) - This gives the average percentage of obligators that default in this rating grade in the course of one year
- Exposure At Default (EAD) - An estimate of the amount outstanding (drawn amounts plus likely future drawdowns of yet undrawn lines) in case the borrower defaults

- Loss Given Default (LGD - The percentage of exposure the bank might lose in case the borrower defaults. These losses are usually shown as a percentage of EAD

The Expected Loss (in currency amounts) can then be written as

$$EL = PD \cdot EAD \cdot LGD \tag{2.2}$$

Or if expressed as a percentage figure of the EAD, as

$$EL = PD \cdot LGD \tag{2.3}$$

With the Internal Ratings Approach the need for reliable quantitative models that predict defaults accurately is imperative for their minimum regulatory capital calculation. This is because banks have an incentive to minimise the capital they hold, because reducing capital frees up economic resources that can be directed to profitable investments. On the other hand, the less capital a bank holds, the greater is the likelihood that it will not be able to meet its own debt obligations, i.e. that losses in a given year will not be covered by profit plus available capital, and that the bank will become insolvent (Basel, 2005).

Noting that Basel II/III also brings its own definition of defaulting companies, which differs from the definition used in most of the previous studies on bankruptcy prediction as well as in this thesis.

2.5 Summary

The main focus on work in developing countries has been using multivariate discriminate analysis for bankruptcy prediction. There have been some studies using neural networks and logit analysis for bankruptcy prediction and fewer using support vector machines. There have been more studies in developing countries on support vector machines to predict company default than in South Africa. Assessing both the past studies on developing markets and in South Africa highlights the relevance of this thesis, developing a model using support vector machines and neural networks to predict bankruptcy in the developing country South Africa.

All the above studies have developed the model on input data from the relevant

country and not just applied a model previously developed from input data from a developing country for bankruptcy prediction.

Results from the above studies illustrate that all of these models can be utilized for bankruptcy prediction in developing countries assuming the correct input data is utilized in the model and the model is set up and executed correctly.

Chapter 3

Definition of Default

3.1 Introduction

It has been shown that an issue of growing importance in both industry and academia is predicting financial failure or financial distress. Financial distress can lead to financial failure or be used to describe financial failure itself. The definitions of both financial failure and financial distress are diverse, and are not uniform in the past literature. Financial failure can refer to bankruptcy, insolvency, the ceasing of current business operations or liquidation each having a different technical definition.

3.2 Bankruptcy Definitions in Previous Studies

The different definitions of bankruptcy date back to the 1930s where Fitzpatrick (1934) made the distinction between failing and failed companies and described the different transitions of financial distress that occur before as a company fails (K Poston, 1994)
(Naidoo, 2006b) . The five stages that lead to business failure according to Fitzpatrick were:

1. Incubation - The first stage incubation is likely to go unnoticed it is when the company's financial difficulties are just developing.

2. Financial Embarrassment - This is when management and other people are likely to notice the company's financial position, it is when the company is unable to meet immediate cash needs. There are remedies for financial embarrassment including borrowing to meet the immediate cash needs.

3. Financial Insolvency - This happens if the company is unable to acquire the funds to meet its obligations. Like Financial Embarrassment this is curable but the remedies are usually long term in nature, for example new management or long term debt.

4. Total Insolvency - This is when liabilities are greater than total assets. The company can no longer avoid the confession of failure.

5. Confirmed Insolvency - This is when legal steps are taken to protect the company's creditors

Ward and Foster (1987) noted that previous researchers such as Altman (1968a), Deakin (1972), Ohlson (1980) used legal bankruptcy as the response variable for economic financial distress, or included legal bankruptcy with other events in dichotomous prediction models. Altman (1968a) definition of bankruptcy is if a company filed a bankruptcy petition under the National Bankruptcy Act during the period $1946 - 1965$ (Altman, 1968a). Ward and Foster (1987) argues that the results of bankruptcy prediction studies should be interpreted as a description of distress rather than of bankruptcy. He suggests that financial distress or economic bankruptcy is often described by one of the following circumstances:

1. A condition of negative net worth

2. An inability to pay debts as they are due

3. The legal definition of bankruptcy

Ward and Foster (1987) found that the amount of time a firm can remain distressed before filing for bankruptcy can stretch up to 7 years. Financial distress can be a critical event and not a fatal event.

In past studies on bankruptcy prediction in South Africa definitions of bankruptcy have differed slightly to the bankruptcy definitions for global studies. One of the reasons for this is due to the difficulty in obtaining adequate data on companies that have declared bankruptcy, refer to Chapter 1. Therefore different definitions of bankruptcy have been adopted. Hlahla (2010b) used the Sharenet [1] definition of default:

1. Firms that have agreed to undertake a restructuring scheme to revive their financial conditions by the South African authorities.

2. Firms that were put under receivership.

3. Companies that have been incurring losses for three years continuously or more.

4. Companies that have exhibited negative position in cash flow for three years continuously or more.

3.3 Bankruptcy Definitions in This Study

For the purpose of this thesis the terminology "Default" and "Non-Default" will be used to classify a company in financial distress rather than the formal definition of bankruptcy. This thesis uses companies listed on the JSE. For a company to fall into the category default, one of the following conditions need to have occurred:

1. The company may undergo financial restructuring that it would not have taken if it had sufficient cash flow (Rao, 2017). An example would be issuing rights which it would not have ordinarily done to raise capital to repay loans.

2. Asset Restructuring - The company has sold off its assets to repay debt (Gaughan, 2015)

[1] Sharenet is a common trading platform used in South Africa http://www.sharenet.co.za/

3. The company has been suspended from trading on the JSE. In this study all suspended companies selected would have been selected because they were suspended because they did not submit financial statements as required under their listing conditions [2]

4. The company has been liquidated

5. The company has delisted - firms delist voluntarily when they fail to benefit from listing. Overall, these firms destroyed shareholder value and they shouldnt have come to the market (Pour, 2013). In this study delisted firms have been investigated on the reason for delisting. If by delisting they have destroyed share holder value they have been selected

6. The share price of the company has dropped significantly (more than 50 percent) whilst the share price of the other companies in that sector remain constant.

3.4 Summary

In previous bankruptcy prediction studies in which the algorithm involved predicts a binary outcome of "Default" or "Non Default" there have been many different definitions of "Default". The definition of "Default" will effect the classification output of the algorithm. For the purpose of this thesis the terminology "Default" and "Non-default" will be used to classify a company in financial distress. Financial distress can refer to bankruptcy, insolvency, the ceasing of current business operations or liquidation each having a different technical definition.

[2] https://www.jse.co.za/content/JSEEducationItems/Service

Chapter 4

Overview of Bankruptcy Prediction Models

4.1 Introduction

This chapter provides an overview of the different types of default prediction models. It describes the different approaches for default prediction models as well as highlighting the main advantages and disadvantages of each method. It excludes the most recent quantitative default prediction modelling as this analysis forms the basis for this thesis and is discussed in more detail in Chapter 6, Chaper 7 and Chapter 8. This chapter is not meant as a complete summary of such a vast body of default prediction and credit risk modelling literature, but is intended instead to motivate the analysis provided later in this thesis.

Modern credit risk literature identifies two primary types of models that describe default processes; structural models and reduced-form models. Structural models are based on modern option pricing theory. They use the companys balance sheet items to develop a value process through which one can determine the time of default. Unlike in the structural default framework, the reduced form approach does not consider in an explicit way to model the relation between default and the companys underlying value. The reduced form approach depends on market prices of deflectable derivatives and other market indicators and not simply on balance sheet information (JeanBlanc and Lecam, 2008). Prior to the structural and reduced form models default prediction modeling was done using multivariate discriminate analysis. One of the most prominent models to predict firm-level bankruptcy was offered by Altman (1968a), who applied multiple discriminant analysis (MDA) to a sample of US industrial firms.

Altmans Model was known as the Z-Score model and is still being used today. This section will describe multivariate scoring models, structural form models and reduced form models.

4.2 Multivariate Scoring Models

The multivariate scoring model is a statistical process for estimating the relationships among variables, in particular the relationship between a dependent variable and one or more independent variables. For the purpose of bankruptcy prediction, the dependent variable is Default or No default. See Chapter 3 for the different definitions of default.

There are only two outcomes for the dependent variable, "Default" and "Non Default". Therefore, the dependent variable is a binary variable, that can take on two variables, which can be denoted by a 0 or a 1. The independent variables are accounting variables or financial ratios from the balance sheet, cashflow statement and income statement. A multivariate scoring model takes key accounting variables or financial ratios and combines and weights them to provide either a credit score or a probability of default.

Multivariate Scoring Models can be classified by at least four different approaches:

1. the univariate analysis model

2. the multivariate discriminant analysis model

3. the logit model

4. the probit model

The most dominant of these models has been Discriminate Analysis followed by Logit Analysis (Altman and Saunders, 1998). Linear regression analysis will be introduced but the linearity assumption can easily be relaxed.

4.2.1 Description of Univariate Analysis and Multivariate Analysis Models

The linear regression model will model the dependent response variable "Default" or "Non Default" as a function of the independent financial ratios with an error term:

$$Default\,Status = Model(financial\,ratios) + error \tag{4.1}$$

In the univariate case the default status is modelled in terms of a single variable:

$$Y(\mathbf{x_1}) = \beta_0 + \beta_1 x_1 + \varepsilon \tag{4.2}$$

where:
Y is the dependant response variable
x_1 is the independent financial ratio
β_0 is a constant
β_1 is regression co-efficient
ε is the error term

For a multiple regression the default status is modelled in terms of a multiple number of variables.

$$Y(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon \tag{4.3}$$

where:
n is the number of financial ratios
Y is the dependant response variable
β_0 is a constant
ε is the error term
$x = x_1, x_2, ..., x_n$ are the independent financial ratios
$\beta_1, \beta_2, ..., \beta_n$ are the regression co-efficients

discriminate analysis was first introduced by Fisher (1936). A detailed discussion of discriminate analysis can be seen in Koiov (2014). In this context discriminate analysis is a form of supervised learning, see Chapter 6 Section 6.3 for details on supervised learning. Linear discriminant analysis is based on the estimation of a linear discriminant function with the task of separating individual groups (in this case the defaulting companies and non-defaulting companies) according to specific characteristics. The aim of Discriminate Analysis is to combine the variable scores in some way so that a single new composite variable, the discriminant score or the discriminate function $Y(x)$ is determined.

The discriminant function is:

$$Y(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon \tag{4.4}$$

where:
n is the number of financial ratios
Y is the dependant response variable
β_0 is a constant
ε is the error term
$x = x_1, x_2, ..., x_n$ are the independent financial ratios
$\beta_1, \beta_2, ..., \beta_n$ are the regression coefficients

The discriminant function will adhere to the following principles:

- Maximisation of the spread between the groups ("Default" and "Non Default")
- Minimisation of the spread within individual groups

multivariate discriminant analysis is similar to the multiple linear regression model given in Equation 4.3 and Equation 4.4. In fact, the proportions among the coefficients of the regression model are equal to the optimal proportion according to the discriminant analysis. The difference between the two methods is a theoretical one, in the regression model the characteristics are deterministic, and the default state is the realization of a random variable, for discriminant analysis the opposite is true. In this example the groups ("Default" or "Non-Default") are deterministic and the characteristics of the discriminant function are realizations from a random variable. For practical use this difference is virtually irrelevant (Engelmann and Rauhmeier, 2008).

The major underlying assumptions of Discriminate Analysis are:

1. The observations are a random sample with each predictor variable normally distributed.
2. Each of the allocations for the dependent categories in the initial training set is correctly classified.
3. There must be at least two groups or categories, with each case belonging to only one group so that the groups are mutually exclusive.

If the discriminate analysis is successful each group will have a normal distribution of discriminant scores. The degree of overlap between the discriminant score distributions can then be used as a measure of the success of the technique (Engelmann and Rauhmeier, 2004).

Figure 4.1 is a graphical representation of the different degrees of overlap between two Discriminate Scores. The degree of overlap between the discriminant score distributions can be used as a measure of the success of the technique. The top two distributions overlap too much and do not discriminate well compared to the bottom set. Misclassification will be minimal in the lower pair, whereas many will be misclassified in the top pair (Engelmann and Rauhmeier, 2004).

If an observations discriminant score is less than or equal to some cutoff value, then assign it to "Default"; otherwise assign it to "Non Default".

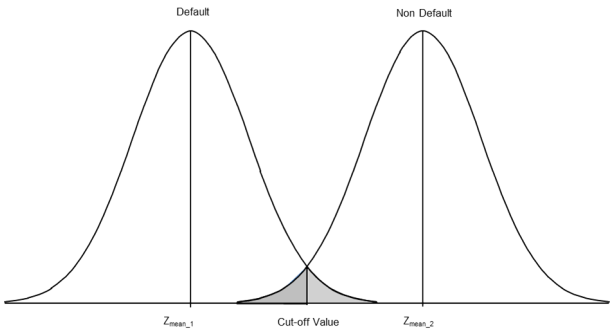


Figure 4.1: The Cutoff Value

Figure 4.1 is a graphical representation of a cutoff value. Calculation of the cut off value is:

$$\text{Cutoff value} = \frac{Z_{mean1} + Z_{mean2}}{2} \text{ for equal sample sizes for the two groups} \quad (4.5)$$

$$\text{Cutoff value} = \frac{Z_{mean1} + Z_{mean2}}{n_1 + n_2} \text{ for unequal sample sizes} \quad (4.6)$$

where:
$Zmean_1$ = the average discriminant score for "Default"
$Zmean_2$ = the average discriminant score for "Non Default"
n_1 = number in sample for "Default"

n_2 = number in sample for "Non Default"

discriminate analysis can be used to produce a direct estimate of the probability of Default. If "Default" is the event that the company has defaulted then the probability of default is Gurny and Gurny (2013):

$$\text{Probability of Default} = \text{PD} = P(\text{Default}|x_i) = \frac{1}{1 + \frac{1-\Pi_\beta}{\Pi_\beta} e^{Y-\alpha}} \tag{4.7}$$

where:
Y is the function defined in 4.3
Π_β is the prior probability of default (relative frequencies)
$\Pi_\beta = \frac{\text{number of samples in default class}}{\text{total number of sample size}}$
$\alpha = 0.5\gamma(x_A x_B)$
x_A and x_B are vectors containing the mean values of the n independent variables
γ is the vector of gamma coefficients
$\gamma = \Sigma^{-1}[x_A - x_B]$
Σ is the matrix of variances and covariances between the n independent variables

4.2.2 Advantages and Disadvantages of Univariate Analysis and Multivariate Analysis Models

The advantages of univariate analysis and multivariate analysis are

- Easy to implement
- Easy to interpret, once the cutoff discriminate score is calculated, but the absolute value of the discriminant function cannot be interpreted easily.

The disadvantages of univariate analysis and multivariate analysis are:

- the multivariate analysis needs strong assumptions that the distribution of predictors within each class is a multivariate normal distribution.
- unordered categorical predictors cannot be used as the input variables.
- multivariate analysis is very sensitive to outliers compared to logistic regression.

4.2.3 Description of Logit Models

The logit model has become one of the most commonly applied parametric failure prediction models in both the academic literature as well as in the banking regulation (van der Ploeg, 2010). The logit model is used for modeling binary outcome variables, it is assumed that the binary response, Y takes on the values of 0 and 1 with 0 representing failure and 1 representing success. The main difference between discriminate analysis and logit analysis is that logit analysis does not assume multivariate normality.

For each company i, the dependent response variable, y which is a function of the financial ratios x will equal to 0 if default occurs (with probability P_i) and to 1 if default does not occur (with probability $1 - P_i$). In the logit model we wish to model the probability P_i that the default will occur. The logit model by relates the score y directly to the probability of default.

To sum up: we have a binary output variable Y and we wish to model the conditional probability $Pr(Y = 1|X = x) = p(x)$ as a function of x. We do this by making $log(\frac{p(x)}{1-p(x)})$ be a linear function of x.

Hence:

$$log(\frac{p(x)}{1-p(x)}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{4.8}$$

where β_i are regression co-efficients as seen in Figure 4.3

Let:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{4.9}$$

Solving for $p(x)$ gives:

$$p(x) = e^{\frac{z}{1+e^z}} = \frac{1}{1+e^{-z}} \tag{4.10}$$

This is the Cumulative Density Function (CDF) for the logistic distribution

4.2.4 Description of Probit Models

The probit model is closely related to the logit model described above. The probit model also predicts a binary dependant variable Y. The main difference between the probit model and the logit model is that the conditional probability $p(x)$ of the logit model can be captured by the CDF of the logistic distribution and the conditional

probability $p(x)$ of the probit model can be captured by the CDF of the standard normal distribution.

Let $z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$ then the conditional probability is given by:

$$Pr(Y = 1|X = x) = p(x) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\Pi}} e^{-0.5t^2} dt \tag{4.11}$$

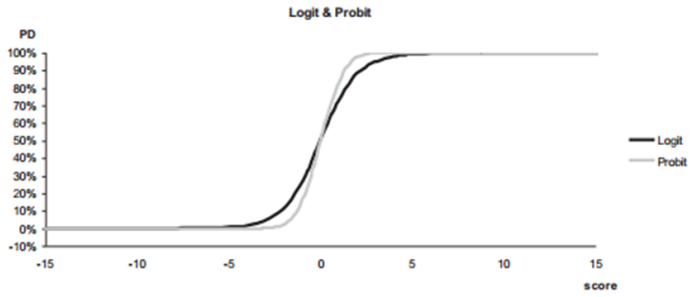


Figure 4.2: Logit Model has slightly heavier tails than the Probit Model

Figure 4.2 shows that logit and probit models give qualitatively similar results and the main difference between logit and probit model is that logit has slightly heavier tails (Petr Gurn, 2013).

4.2.5 Advantages and Disadvantages of Logit and Probit Models

The advantages of the logit and probit model are:

- the independent variables dont have to be normally distributed.
- It does not require that the independents be unbounded.

The Disadvantages of the Logit and Probit Model are

- It does not assume a linear relationship between the dependant variables and the independent variables but it assumes a linear relationship between the logit transformation of the dependant variables and independent variables.
- The dependent variable has to be binary.

4.3 Structural Form Models

Structural form models employ modern option pricing theory in corporate debt valuation. The original framework developed on 1974 by Robert Merton Merton (1974). The theory was based on the principles of option pricing formulae introduced by Black and Scholes in 1973 (Black and Scholes, 1973). The Merton Model laid the cornerstone for all other structural models.

The basis of the structural model approach asses the probability of default of a company by characterizing the companys equity as a call option on its assets. That is, debt and equity are contingent claims on the firms assets. The model assumes that a company has a certain amount of zero-coupon debt that expires at a time T. Default occurs when the firm value falls to a low level such that the issuer cannot meet the par payment at maturity time T. Hence if the value of its assets is less than the value of the debt at time T, the company will default.

The equity of the company is a European call option on the assets of the company with maturity T and a strike price equal to the value of the debt. The model can be used to estimate either the risk-neutral probability that the company will default (Hull et al., 2003).

4.3.1 The Merton Model

The theory behind the Merton Model was introduced in Merton (1974).

At time t, for a company define:

V_t Value of assets (Value of company)
E_t - Value of Equity
D_t - Zero coupon dept. It might be necessary to transform the debt structure of the firm into a zero-coupon bond with maturity T and face value K
K - Face value of Zero Coupon Bond
T - The maturity T of the zero-coupon bond can be chosen either to represent the maturity structure of the debt, for $T > t$

The value of the companys assets at time t can be measured by the price at which the total of the firms liabilities can be bought or sold (Wang, 2009).

It them follows that:

$$\text{Market Value of Assets} = \text{Market Value of Equity} + \text{Market Value of Bonds} \tag{4.12}$$

or

$$V_t = E_t + D_t \tag{4.13}$$

Now if:

- $V_T > K$ the companys debtholders can be paid the full amount K, and shareholders equity still has value $V_T - K$.
- $V_T < K$ debtholders have the first claim on the assets V_T, take hold of the company and shareholders are left with nothing and the company defaults.

Therefor at time T, equity value E_t can be written as $E_T = max(V_T - K; 0)$

This is the payoff of a European call option written on underlying asset V_t with strike price K maturing at time T. The Black and Scholes option price formulae can be applied under the Black and Scholes assumptions (Black and Scholes, 1973).

Applying the Black and Scholes option pricing formulae at any time t the market value of the companys equity is:

$$E_t = V_t\phi(d_+) - Ke^{-r(T-t)}\phi(d_-) \tag{4.14}$$

Where $\phi(.)$ denotes the N(0,1) cumulative distribution function with d_+ and d_- given by:

$$d_+ = \frac{ln(\frac{V_t}{K}) + (r + \frac{1}{2}\sigma_V^2)(T-t)}{\sigma_V\sqrt{T-t}} \tag{4.15}$$

$$d_- = \frac{ln(\frac{V_t}{K}) + (r - \frac{1}{2}\sigma_V^2)(T-t)}{\sigma_V\sqrt{T-t}} \tag{4.16}$$

where:
σ_V is the standard deviation of V_t
r is the standard riskfree rate

4.3.2 Assumptions of the Merton Model

Structural Models (including the Merton Model) have many advantages; firstly they model default on the realistic assumption that it is a result of the firms assets falling below the value of the debt. Disadvantages are that they are difficult to calibrate and are computationally intense (Focardi and Fabozzi, 2004)

The following key assumptions are set in the Merton Model (Sundaresan, 2013) (Mria Miankov, 1977):

- There are no transactions costs, taxes, or problems with indivisibilities of assets.
- The liabilities of the company consist of one zero-coupon bond.
- Debt structure is static, it doesnt change.
- There are a sufficient number of investors with comparable wealth levels such that each investor believes that he can buy and sell as much. of an asset as he wants at the market price.
- The riskiness of the investment will not be influenced by how close the company is to default.
- There exists an exchange market for borrowing and lending at the same rate of interest.
- Short sales of all assets, with full use of the proceeds, are allowed.
- Trading in assets takes place continuously in time.
- The value of the company in not dependent on its capital structure.
- The debt structure of the firm is modelled as a zero-coupon bond.
- The company assets has lognormal distribution, it cannot be negative.
- The dynamics for the value of the firm, V_t can be described through a stochastic process.

Ever since the introduction of the Merton Model there have been many researchers who have criticized the underlying assumptions. There have been many proposed extensions to the original Merton model to relax some of the assumptions stated above. A description of these extensions will be explored in Chapter 5 Section 5.3.2.

4.4 Reduced Form Models

Reduced form models assume default occurs exogenously (usually by a Poisson process) and a separately specified recovery is paid upon default. the reduced form approach uses a modelling technique that does not consider the relationship between whether a company defaults and the companys underlying value. This is unlike both structural form and multivariate discriminate Analysis which use information off the companys balance sheet to determine the time of default.

The reduced form method uses inputs such as market prices of defaultable derivatives such as Credit Default Swaps and other market indicators to model the rate of default through an intensity or hazard rate process. The reduced form approach lies on the assumption that the bankruptcy or default occurs by surprise (by hitting an unobserved boundary). The reduced firm model consists of modelling of this random time. The parameters governing the default time are inferred from market data (Elizalde, 2006) (JeanBlanc and Lecam, 2008). The reduced form framework has been split so far into two different approaches: the Intensity based approach and the Hazard Approach.

4.4.1 Intensity based approach

In the Intensity Based Approach economic uncertainty is modelled with the specification of a filtered probability space Π (Elizalde, 2006).

$$\Pi = (\Omega, F, (F_t), P) \tag{4.17}$$

where Ω is the set of possible states of the economic world P is a probability measure The filtration (F_t) represents the flow of information over time and contains all the needed economic information

$$F = \sigma(\bigcup_{t \geq 0} F_t) \tag{4.18}$$

where F is a σ-algebra, a family of events at which we can assign probabilities in a consistent way

Other assumptions are that a financial market is said to have the following characteristics:

- T is the maturity date
- A default event occurs at some random time τ,
- A promised contingent claim X, paid at time τ if $\tau < T$
- A recovery process $Z : Z_\tau$ is the recovery payoff at time of default, if it occurs prior to or at the maturity date T.

The default time τ is modelled as a random time in this given filtration (F_t). The modelling is based on the assumption of existence of an intensity rate process: a non-negative process satisfying a certain properties. Default intensities may be allowed to depend on observable input variables that are linked with the likelihood of default, such as financial ratios, volatility measures, market equity prices, bond yield spreads, industry performance measures, and macroeconomic variables related to the business cycle. The full modelling of this process can be seen in JeanBlanc and Lecam (2008).

The intensity process then calculates the default probability of a given firm at the default time τ (Elizalde, 2003).

4.4.2 Hazard Process Models

The Hazard approach is based on the computation of the "hazard process". A pricing rule known as the "Hazard Based Pricing Rule" is derived. The hazard approach starts with the same assumptions as the intensity approach: economic uncertainty is modelled with the specification of a filtered probability space Π as defined in 4.17. There exists a σ-algebra F as defined in Equation 4.18.

The Hazard Based approach is a version of Reduced Form Models, which is slightly more general than the Intensity Based Approach (T Bielecki, 2009). The Hazard process introduces a second σ-algebra G.

At time t the filtration G is defined by:

$$G_t = F_t \bigvee H_t \tag{4.19}$$

$$H_t = 1_{\tau \leq t, t \geq 0} \tag{4.20}$$

where τ is the stopping time F_t is the filtration defined in Equation 4.18

The intensity process then calculates the default probability of a given firm at the default time τ under the more general conditions introduced. A full mathematical proof can be seen in Elizalde (2003).

4.4.3 Credit Default Swaps

A Credit Default Swap(CDS) contract is made between two parties - the protection buyer and the protection seller. is a contract that provides insurance against the risk of a default by particular company. The company is known as the reference entity and a default by the company is known as a credit event. The buyer of the insurance obtains the right to sell a particular bond issued by the company for its par value when a credit event occurs. The bond is known as the reference obligation and the total par value of the bond that can be sold is known as the swaps notional principal. The market standard valuation of a credit default swap requires estimates of the risk-neutral probability that the reference entity will default at different future times. This probability of default is calculated by an intensity or hazard rate (White, 2014) (Hull and White, 2000). The full pricing of a CDS contract can be seen in White (2014).

4.4.4 Advantages and Disadvantages of Intensity and Hazard Based Models

In both Intensity and Hazard models are relatively easy to calibrate, this is the main reason why reduced form models are strongly favoured in industry (Focardi and Fabozzi, 2004).

Other advantages are (T Bielecki, 2009):

- They can be made consistent with the risk-neutral probabilities of default backed out from corporate bond prices or CDS spreads.
- No restrictions are imposed on the way monetary policy affects the economy.
- The default time is modeled as an exogenous random variable with no reference to any particular economic background.
-

Disadvantages of Intensity and Hazard Models are (T Bielecki, 2009):

- The range of default correlations that can be achieved is limited. Even when there is a perfect correlation between two hazard rates, the corresponding correlation between defaults in any chosen period of time is usually very low. This is can be a problem in some circumstances, for example, when two companies operate in the same industry and the same country or when the financial health of one company is for some reason heavily dependent on the financial health of another company.
- A lack of clear reference to economic fundamentals, such as the company's asset-to-debt ratio.
- Structural Form Models are better suited for inference of reliable debt information, such as: risk-neutral default probabilities or the present value of the firm's debt.

4.5 Summary

This chapter provides a brief overview of previous bankruptcy prediction models as well as discussing the advantages and disadvantages of each method. Multivariate Scoring Models, Structural Form Models and Reduced Form Models are discussed in this chapter.

The multivariate scoring model is a statistical process for estimating the relationships among variables, in particular the relationship between a dependent variable and one or more independent variables. multivariate scoring models can be classified by at least four different approaches; the univariate analysis model, the multivariate discriminant analysis model, the logit model and the probit model. The most dominant of these models has been discriminate analysis followed by logit analysis.

Structural form models employ modern option pricing theory in corporate debt valuation. The basis of the structural model approach asses the probability of default of a company by characterizing the company's equity as a call option on its assets.

Reduced Form Models assume default occurs exogenously (usually by a Poisson process) and a separately specified recovery is paid upon default. The main difference between Reduced Form Modelling and the other two methods is that Reduced Form Modelling uses a modelling technique that does not consider the relationship between whether a company defaults and the company's underlying value.

Chapter 5

History of Bankruptcy Prediction Models

5.1 Introduction

The literature on default prediction methodologies is substantial. There have been many studies on default analysis in the past 40 years. A timeline of all studies done can be seen in Bellovary et al. (2007). This chapter will discuss a few of the most pioneering works for multivariate scoring models and structural form models that have been cited the most in previous bankruptcy literature.

5.2 History of Multivariate Scoring Models

5.2.1 Beaver

Beaver (1966) laid the framework for predictive models. He used univariate analysis for a number of bankruptcy predictors. His use of univariate analysis for bankruptcy prediction set the stage for the multivariate attempts. Beaver used financial ratios of 79 pairs of bankrupt and non-bankcrupt us companies one year to failure. The failed firms were from 38 different industries. The non-default companies were chosen to be in the same industries as the failed firms.

Beaver then selected 30 financial ratios based on performance in previous studies as well as being defined in terms of cashflow. They were analysed in terms of mean

values and a dichotomous test was used to make a prediction whether a firm had defaulted or not. The best single predictors of bankruptcy were:

- Cash Flow/Total Debt
- Net Income/Total Assets
- Total Debt/Total Assets

With Cash Flow/Total Debt as the best single predictor of bankruptcy (Beaver, 1966). Beaver was able to classify 78% of the companies correctly (Deakin, 1972).

5.2.2 Altman

Altman (1968b) developed the most widely recognized and applied model for predicting financial distress, known as the Altman Z-Score Model. The famous Altman Z-Score modelled bankruptcy prediction using multivariate discriminate analysis, this analysis was based on Beavers univariate model.

Discriminate analysis was performed on 66 US companies, 33 that have defaulted and 33 that have not. All the companies on this study were from the manufacturing sector. The mean asset size of these companies was $6.4 million.

For the variable selection Altman considered the variables (financial ratios) used in other studies. He made a subset of 22 these financial ratios for evaluation. These financial ratios were chosen on the basis of their popularity in other studies and relevancy to the study. The 22 financial ratios were narrowed down to 5. The 5 financial ratios were chosen on their ability to predict probability of default together and not on their independent ability. Altman did not consider the Cashflow to Debt ratio because of the lack of consistent and precise depreciation and cash flow data.

The Z-Score constructed from the discriminate function is:

$$Z = 0.021X1 + 0.041X2 + 0.033X3 + 0.06X4 + 0.99X5 \tag{5.1}$$

where:
$X1$ = Working capital/Total Assets
$X2$= Retained Earnings/Total Assets
$X3$= EBIT/total Assets
$X4$= Market Value of Equity/Book Value of Liabilities
$X5$= Sales/Total Assets

Interpretation of the Z-Score is that a score below 1.8 means the company will default, while companies with scores above 3.0 are not likely to default. The lower (higher) the score, the lower (higher) the likelihood of bankruptcy.

Financial ratios one year prior to default and two years prior to default were tested. Findings were that there was a 95% accuracy one year prior to default and 72% accuracy two years prior to default. The results obtained were superior to the results Beaver attained with his single best ratio analysis.

Since Altman published his z-score model logistic regression has replaced discriminate analyses as the preferred tool for binary outcomes as the outcomes are less restrictive (Frade, 2008).

5.2.3 Deaken

Deakin (1972) compared Beavers and Altmans methods using the same sample. He first replicated Beaver studies using the same ratios that Beaver had used. Next, he searched for the linear combination of the 14 ratios used by Beaver which best predicts potential failure in each of five years prior to failure. He then devised a decision rule which was validated over a sample of firms (Deakin, 1972).

Deaken used the same method of company and ratio selection as Beaver but with one major difference being by his definition of default. Beaver defined default as experiencing bankruptcy, insolvency, liquidation as well as defaulting on loan obligations. Deaken defined default as experiencing bankruptcy, insolvency or liquidation.

Deakins findings were in favor of the discriminant analysis, which compared to the univariate analysis, is a better classifier for potential bankrupt firms.

5.2.4 Edmister

Edmister (1972) tested a number of methods of analysing financial ratios to predict small business failures. Even though he found that not all methods and ratios could be used as predictors of failure, he confirmed that some ratios could be used to predict failure of small companies. Finally, Edmister recommended using at least three consecutive years financial statement to predict the default of small companies.

5.2.5 Ohlson

Ohlson (1980) used conditional logit analysis to provide an alternative method to multivariate discrimimate analysis for predicting default. Ohlson describes the shortcomings of multivariate discriminate analysis to predict default as:

- There are certain statistical requirements imposed on the variables used in the analysis.
- The output of multivariate discriminate analysis is a score which has little intuitive interpretation
- Multivariate analysis relies on the companies chosen for the analysis to be "matched" to each other on certain criterion such as asset size, industry etc. but these should rather be input variables into the equation.

Ohlson used a data set consisting of 105 bankrupt firms and 2058 surviving firms for the logit analysis.

Nine financial ratios were selected for the logit analysis:
1. Size
2. Total liabilities/total assets
3. Working capital/total assets
4. Current liabilities/current assets
5. Binary operator 1 if total liabilities ¿ total assets, 0 otherwise
6. Net income/total assets
7. Cash from operations/total liabilities
8. Binary operator 2 1 if net income was negative for last two years, 0 otherwise
9. Change in net income

For the variable selection Ohlson considered variables used in previous research. The variables were chosen because he perceived them to be the ones most frequently mentioned in previous literature.

A new variable "size" which is the size of the company was introduced. The "size" variable allows for more companies to be used in the analysis as the sample does not need to be chosen within certain matching criterion. Ohlson claimed that his study had one important advantage; the financial reports indicated at what point in time the financial statements were released to the public. Therefore one could check whether a company entered bankruptcy prior to or after the date of release for the financials.

Findings of his study were:
i. The four best predictors were size of company, a measure of the companies financial structure, a measure of performance and a measure of liquidity
ii. Previous studies overstated the predictive power of models because financial statements were released after bankruptcy was filed for.
From a statistical perspective Ohlson claimed logit analysis should be preferable to multivariate analysis.

5.3 History of Structural Form Models

5.3.1 Merton

The original framework developed for Structural Form Models was developed in 1974 by Robert Merton in Merton (1974). A description of the Merton Model was described in 4 as it formed the cornerstone for Structural Form Models.

5.3.2 Modifications to the Merton Model

Ever since the Merton Model there have been many researchers who have proposed extensions to the original Merton model to relax some of the assumptions stated in Chapter 4 Section 4.3 Subsection 4.3.2. Some of these models are:

- F Black (1976) modified the Merton framework to allow for default before time T. In Mertons model a company could only default at its debt maturity date. The Black and Cox model can be modified to allow for early defaults by specifying a threshold level. A default event occurs when asset value Vt falls below this critical level. Equity is no longer a European call option on the borrowing firms assets. Rather, equity is a down-and-out call option on the firms assets. They also consider senior and subordinated debt and interpret their valuations.
- A Loigstaff (1995) incorporated a stochastic interest rate rather than Mertons fixed interest rate.
- Lehland (1998) introduced taxes and bankruptcy costs which may be interpreted as liquidation costs. Thus, he formalized the trade-off framework and provided a way to determine both the optimal default boundary and the value maximising optimal capital structure.

- Geske (1977) uses the assumption that the firms debt is a coupon bond instead of a zero-coupon one.

5.3.3 KMV Model

The KMV model is based on the Merton model, it is otherwise known as the KMV Merton model. It was developed by KMV corporation in the late 1980s. Moodys Analytics acquired KMV in 2002 to expand its credit risk management product offering. It was called the KMV Merton model because it is a nontrivial extension of Merton Model. The KMV Model was adapted by Vasicek (1984). It was then applied by KMV Corporation.

The KMV-Merton Model applies the framework of Merton (1974), in which the equity of the company is a call option on the underlying value of the firm with a strike price equal to the face value of the firms debt. The model recognizes that neither the underlying value of the company nor its volatility are directly observable but can be calculated by solving a system of non-linear simultaneous equations (Bharath and Shumway, 2004).

Vasicek (1984) introduces the distinction between short and long term liabilities as well as the probability of default. In the KMV model, default occurs when the firms asset value goes below a threshold represented by the sum of the total amount of short term liabilities and half of the amount of long term liabilities (Sreedhar T Bharath, 2004).

5.4 Summary

the multivariate scoring models discussed in this chapter are Beaver (1966), Altman (1968a), Deakin (1972), Edmister (1972) and Ohlson (1980). Beaver (1966) was one of the first researchers to apply a multivariate scoring model for bankruptcy prediction and laid the framework for the other predictive models. Altman (1968a) developed the most widely recognised and applied model for predicting financial distress, known as the Altman Z-Score Model.

The most relevant structural form model was Merton (1974)'s Merton Model as it formed the cornerstone for structural form models. Ever since the Merton Model there have been many researchers who have proposed extensions to the original Merton model to relax some of the model assumptions.

Chapter 6

Machine Learning Description

6.1 Introduction

The purpose of this study is to compare the performance of support vector machines and neural networks, to traditional statistical models (logit analysis and multivariate discriminate analysis) for credit loss estimation in South Africa. Neural networks, support vector machines, logit analysis and multivariate discriminate analysis are all used to classify data into groups based on historic data. This type of algorithm is knows as machine learning.

In order to implement the above algorithms and correctly interpret and compare the results it is necessary to understand the general concept of machine learning. This chapter will describe the implementation of general machine learning algorithms as well as different methods of interpreting the results.

6.2 Machine Learning Description

Machine learning is programming computers to optimize a performance criterion using example data or past experience. There is a model defined up to some parameters and *learning* is the selection of these parameters to optimize the model based on the example data or past experience. The model may be predictive to make predictions in the future or descriptive to gain knowledge from data (Alpaydin, 2004). Machine learning is a form of artificial intelligence that provides the algorithm the ability to learn without being explicitly programmed (Bell, 2014). It is data-driven, and the

output (prediction) of the algorithm is based on the input data.

There is sometimes some overlap with data mining and machine learning as both techniques use data analysis. Data mining and machine learning have been used as alternative terms in the same research, H Liu (2016) explains the overlap between the two area. Even though data mining and machine learning involve data processing, data mining aims to find something new from *unknown* properties and machine learning aims to find something new from *known* properties. In particular data mining has patterns which are previously unknown and the aim is to find patterns in the unknown data. In contrast machine learning has some patterns which are known in general but not known to the machine and the aim is to make the machine learn the patterns.

Data mining focuses on finding patters, associations or relationships in data in order to draw a conclusion. Machine learning finds an outcome which is a prediction based on the training data. There are many applications of machine learning including bioinformatics, cheminformatics, data marketing, hand writing recognition, speech recognition, pattern recognition and spam detection.

6.3 Supervised vs. Unsupervised Learning

In supervised learning the aim is to learn a mapping from the input to an output whose correct values are provided by a supervisor. In unsupervised learning there is no supervisor, there is only the input data (Alpaydin, 2004).

Any dataset used by machine learning algorithms is represented using the same set of features. The features may be continuous, categorical or binary. If the features are given with known labels (the corresponding correct outputs) then the learning is called supervised. If the instances are unlabeled then the learning is unsupervised (Kotsiantis, 2007).

6.4 Methods to Evaluate Performance

In order to validate the performance of a classification model, we need to analyze the models ability to correctly classify the data into "Default" or "Non Default". The performance of the classification model is evaluated on the test data. The methods commonly used to evaluate the performance of machine learning classifiers are (Miha Vuk, 2006):

- Receiver Operating Characteristic (ROC)Curves
- Lift Charts
- Calibration Plots
- Confusion Matrix
- Classification Ratios
- Kappa Coefficient

ROC curves, classification ratios and confusion matrices will be used for validation in this thesis.

6.4.1 Confusion Matrix

When dealing with two classes a confusion matrix is a method of assessing the performance of the classifier. The classifier is applied to the test data, and will assign a class ("Default" or "Non Default") to every point. Some of the classes will be incorrectly assigned when compared to the actual values. An incorrect classification would be if the classifier predicted a company to "Default" and the company is a "Non Default" company.

A confusion matrix is a method of summarising the accuracy of the algorithm, it can be seen in 6.1. A confusion matrix cross tabulates the actual (labeled) classes to the values predicted values from the classifier (Fielding, 2014). It shows a more detailed breakdown of correct and incorrect classifications for each class.

		Predicted	
		Negative	Positive
Actual	Negative	a	b
	Positive	c	d

Figure 6.1: Confusion Matrix

where:

- a is the number of correct predictions that are negative (predict "Non Default" correctly)
- b is the number of incorrect predictions that are positive (predict "Default" incorrectly)
- c is the number of incorrect of predictions that are negative (predict "Non Default" incorrectly)
- d is the number of correct predictions that are positive (predict "Default" incorrectly)

The rows of the above matrix correspond to the known class of the data, i.e. the labels in the data. The columns correspond to the predictions made by the model. The diagonal elements show the number of correct classifications made for each class, and the off-diagonal elements show the errors made.

The predict function and the table function are used to create a confusion matrix in R Studio (Karatzoglou, 2016). This can be seen in Appendix **??**.

6.4.2 Classification Ratios

The Accuracy Ratio(AC) is the proportion of the total number of predictions that were correct (Miha Vuk, 2006):

$$AC = \frac{a+d}{a+b+c+d} \tag{6.1}$$

The Recall or True Positive Rate (TP) is the proportion of positive cases ("Default") that were correctly identified:

$$TP = \frac{d}{c+d} \tag{6.2}$$

The False Positive Rate (FP) is the proportion of positive cases ("Default")that were incorrectly classified as negative ("Non Default"):

$$FP = \frac{b}{a+b} \tag{6.3}$$

The True Negative Rate (TN) is defined as the proportion of negatives cases ("Non Default") that were classified correctly:

$$TN = \frac{a}{a+b} \tag{6.4}$$

The False Negative Rate (FN) is the proportion of negative cases("Non Default") that were incorrectly classified as positive ("Default"):

$$FN = \frac{c}{c+d} \tag{6.5}$$

The Precision (P) is the proportion of the predicted positive cases ("Default") that were correct:

$$P = \frac{d}{b+d} \tag{6.6}$$

6.4.3 The Receiver Operating Characteristic (ROC) Curve

The Receiver Operator Curve (ROC) is a very useful tool for visualizing and evaluating classifiers. They have properties that make them especially useful for calculating classifier error analysis.

ROC Curves measure the ratio or trade-off between benefits and costs. A benefit is the $Sensitivity$ or True Positive Rate (TP) and a cost is $1-Specify$ or False Positive Rate FP. The Sensitivity or True Positive Rate is the probability of correctly classifying a positive class and the Specify is the probability of correctly classifying a negative class when a certain cut-off value or threshold is used Hsieh and Turnbull (1996). Each cut-off value or threshold may result in a different ($Sensitivity$,$1 - Specify$) pair or ($Sensitivity$,$1 - Specify$) point. For different cut-off or threshold values The $Sensitivity$ and the $1 - Specify$ are plotted in a two-dimensional graph in which the $Sensitivity$ is plotted on the $Y - axis$ and $1 - Specify$ rate is plotted on the $X-axis$ (Fawcett, 2006). An ROC Curve is a plot of all the ($Sensitivity$, $1-Specify$) operating points (Strickland, 2002), (Woods, 1997).

The point $(0, 1)$ on the ROC Curve is the perfect classifier, all positive cases and negative cases are classified correctly. It is $(0, 1)$ because the false positive rate is 0 (none), and the true positive rate is 1 (Miha Vuk, 2006).

The point $(0, 0)$ represents a classifier that predicts all cases to be negative, while the point $(1, 1)$ corresponds to a classifier that predicts every case to be positive.

A classifier is good if it has a high positive rate and a low negative rate, so in order to compare two classifiers we look at how close they are to the point $(0, 1)$ (closer to the left hand corner).

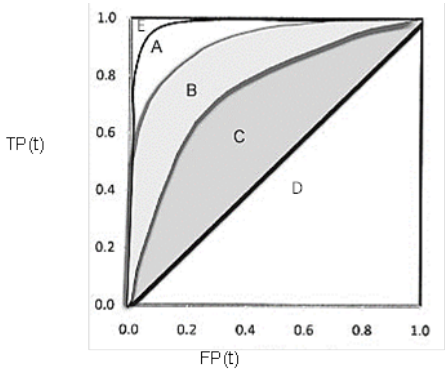


Figure 6.2: Five different ROC curves

The plot in Figure 6.2 is of five different ROC curves (Kumar and Indrayan, 2011). A is better than both B and C. E is the perfect classifier and D is a random classifier.

The area under the ROC curve is often used as a measure of quality of a probabilistic classifier. The random classifier D, has an area under the curve of 0.5 while the perfect classifier E, has an area of 1. The other classifiers (A, B, C) have an area between 0.5 and 1. In R the ROCR package is used to create ROC Curves (Fawcett, 2006).

6.4.4 Empirical ROC Curve

The $Sensitivity$ and $1-Specify$ are calculated for different cut-off or threshold values of the classifier. These discrete set of points can be thought of as the observed ROC Curve for observed threshold points. The points are then connected with a linear interpolation, leading to the Empirical ROC curve (Gonen, 2000). The Empirical ROC Curve is an increasing step function (Rajan, 2008) (Li and Ma, 2013). The Empirical ROC Curve is heavily-used because it makes no assumptions regarding the distribution of the individual predictors.

Figure 6.3 is a graph of the Empirical ROC curve each point on the curve corresponds to a specific cut-off or threshold. In this graph the points are connected linearly (Gonen, 2000).

6.4.5 Area under the ROC Curve

While the ROC curve contains most of the information about the accuracy of a continuous predictor, it is sometimes desirable to produce quantitative summary measures

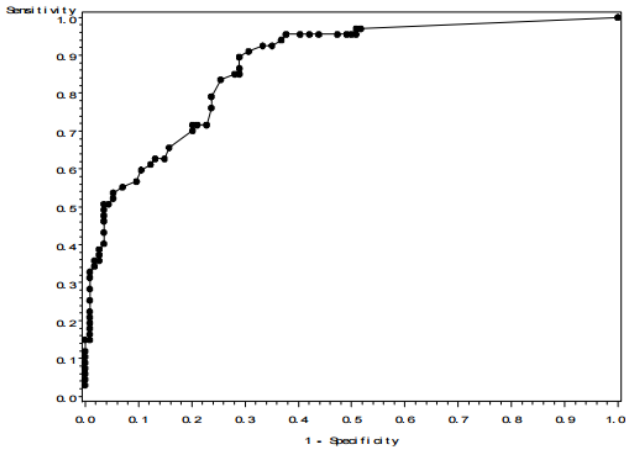


Figure 6.3: Empirical ROC Curve

of the ROC curve. The most commonly used such measure by far is the area under the ROC curve (AUC). The AUC reduces ROC performance to a single scalar value representing expected performance.

The AUC is a portion of the area of the unit square, its value will always be between 0 and 1. The random guessing produces the diagonal line between $(0, 0)$ and $(1, 1)$, which has an area of 0.5, so a classifier that has an AUC less than 0.5 represents poor performance. The AUC has an important statistical property: the AUC of a classifier is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance (Fawcett, 2006).

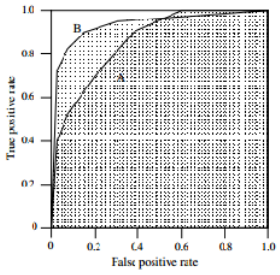


Figure 6.4: AUC ROC Curve

Figure 6.4 and Figure 6.5 shows the areas under two ROC curves A and B. The Classifier B has greater area and therefore better average performance. Figure 6.5 shows the area under the curve of Classifier A and Classifier B. Though the performance of the two classifiers is equal at the fixed point, Classifier A's performance becomes inferior to Classifier B's further from this point. This shows that it is possible for a high AUC classifier to perform in a specific region of ROC space than a low AUC

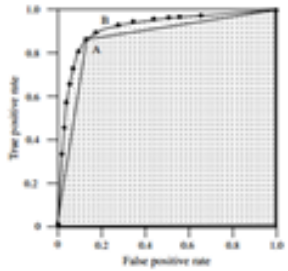


Figure 6.5: AUC ROC Curve

classifier (Fawcett, 2006).

6.4.6 Gini Co-efficient

A standardised variant of the AUC is also used widely in credit scoring models is the Gini Co-efficient. The Gini Co-efficient takes values between 0 (no difference between the score distributions of the two classes) and 1 (complete separation between the two distributions). The Gini Co-efficient can be expessed in terms of the AUC by equation 6.7.

$$G = 2AUC\text{-}1 \tag{6.7}$$

The maximum values that the AUC and Gini coefficient can take are both 1, corresponding to score distributions for which there exists a threshold which yields perfect separation between the sets of scores for the class 0 and class 1 training data. The minimum of the Gini co-efficient of 0 corresponds to identical score distributions (Hand, 2006).

6.4.7 Kappa Co-efficient

The kappa coefficient (K) measures inter-rater agreement for qualitative (categorical) items, correcting for expected chance agreement. It measures the Observed Accuracy with an Expected Accuracy (random chance). It is a measure of how well the classifier performed as compared to how well it would have performed simply by chance.

$$K = \frac{P(A) - P(E)}{1 - P(E)} \tag{6.8}$$

where $P(A)$ is the Observed Accuracy and $P(E)$ is the Expected Accuracy.

The Observed Accuracy is the number of instances that were classified correctly throughout the entire confusion matrix.

$$OA = \frac{b+d}{a+b+c+d} \tag{6.9}$$

The Expected Accuracy is the value is defined as the accuracy that any random classifier would be expected to achieve based on the confusion matrix.

$$OA = \frac{ActualFalse * PredictedFalse + ActualTrue * PredictedTrue}{Total * Total} \tag{6.10}$$

In terms of the confusion matrix it can be written as:

$$OA = \frac{(a+b)(a+c) + (c+d)(b+d))}{(a+b+c+d)(a+b+c+d)} \tag{6.11}$$

6.5 Summary

In both supervised and unsupervised classification the output (prediction) of the algorithm is based on the input data. The algorithm learns without being explicitly programmed. The input data is separated into training data and test data. The algorithm learns on the training data and is validated on the test data. In order to validate the performance of a classification algorithm, we need to analyse the algorithms ability to correctly predict or separate the classes. The performance of the classification model is evaluated on the test data.

Chapter 7

Neural Networks

7.1 Introduction

Multivariate analysis, structural form and reduced form models have been typically used for credit loss estimation. Different from statistical models, the Machine Learning Algorithms are non-parametric techniques for the prediction, so they overcome some constraints of the traditional statistical models (Aktan, 2011b). The disadvantages for each method are discussed in Chapter 4.

Neural networks are a type of machine learning algorithm. They are nonlinear models inspired from biological network of neurons found in the human central nervous system. They involve a cascade of simple nonlinear computations that when aggregated can implement robust and complex nonlinear functions. Neural networks can approximate most nonlinear functions, making them a quite powerful class of models. They have a wide range of uses including forecasting, classification and statistical pattern recognition. Research studies on using neural networks for default prediction started in 1990, and are still active now. Neural networks have generally outperformed the other existing methods (Atiya, 2001).

This thesis will focus on Neural Networks for classification, by supervised learning (perceptron and multilayer perceptron). In this Chapter, a general framework for understanding of the theory behind Neural Networks is presented. A detailed description of artificial neural networks can be found in W McCulloch (1943).

This chapter is organised as follows: The first section will describe the evolution of the neural networks. All of the important concepts used in the feed-forward neural network will be discussed in the first section. The next section will describe the

multilayer feed forward network, and the section after that will describe the training algorithm used.

7.2 General Concept of Neural Network

A neural network is a machine learning algorithm based on the structure and functions of biological neural networks. It is inspired by the way the human brain processes information. Neural networks learn from examples and behave like black boxes, the way they process information being inexplicit. In principle, the similarity between neural networks and the way of action of the human brain may be condensed in the following two aspects:

- Knowledge is acquired by the network through the learning (training) process;
- The intensities of the inter-neuron connections, known as (synaptic) weights, are, used to store acquired knowledge.

Figure 7.1 shows the different types of Neural Networks. There are at least three different types supervised, unsupervised and reinforced learning algorithms (Lippmann, 1988). The type of algorithm is chosen according to the problem we are trying to solve. Although the different types of neural networks are different in their principles they all have one thing in common; on the basis of learning data and learning rules artificial neural network is trying to achieve proper output response in accordance to input signals.

7.3 Evolution of Artificial Neural Networks

The evolution of the artificial neuron has progressed through several stages. The roots of which, are firmly grounded within neurological work done introduced by McCulloch and Pitts (1943). This section will introduce fundamental concepts of neural networks in particular the perceptron which represents the simplest form of a neural network

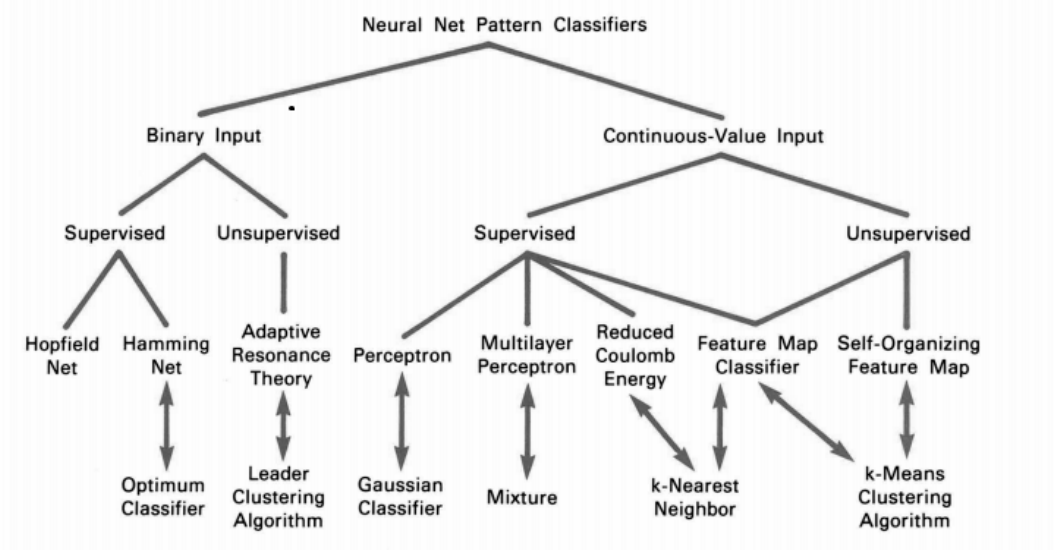


Figure 7.1: A summary of the different types of neural net classifiers corresponding to their classes

7.3.1 The McCulloch-Pitts Model of Neuron

Neural networks comprise of computational units known as processing elements, proposed by McCulloch and Pitts (1943). McCulloch and Pitts (1943) suggested the description of a neuron as a logical threshold element with two possible states. Their idea of a neuron as a logical operator was a fundamental contribution to the field. The theory for this section is from Fausett (1993) and Jones (2004).

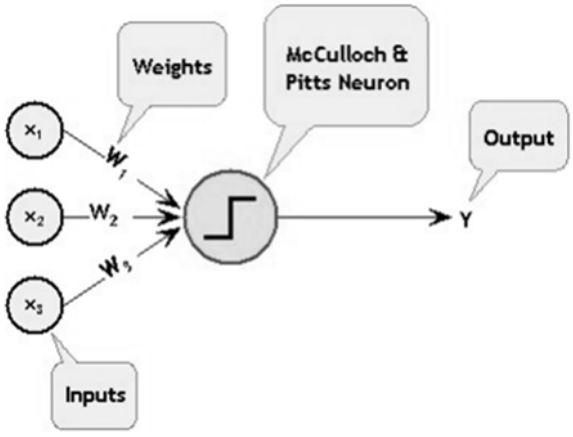


Figure 7.2: The McCulloch-Pitts Threshold Neuron

Figure 7.2 is a diagram of the McCulloch and Pitts Model (Jones, 2004). All input attributes x_i to the threshold neuron are combined into a single number and are connected by weighted paths, hence the output Z is a function of weighted inputs w_i.

$$Z = \sum w_i x_i - \mu \tag{7.1}$$

where:
w_i is the weight associated with the i^{th} input (attribute) x_i
μ is the bias term or intercept

According to McCulloch and Pitts (1943), the neuron does not respond to its inputs unless Z is greater than zero. If Z is greater than zero then the output from this neuron is set equal to 1. If Z is less than zero the output is zero. Thus neurons are binary, either active (1) or inactive (0). They can be described by the sigmoid (step) function (Vera Kurkova and Karny, 2001):

$$Y = sgn(Z) = \begin{cases} 1, & \text{if } Z > 0 \\ 0, & \text{if } Z \leq 0 \end{cases}$$

7.3.2 The Hebb-Model

The Hebb model was introduced by the psychologist Hebb (1949). It is commonly known as Hebbs Law. Hebbs Law is a learning rule that describes how the neuronal activities influence the connection between neurons. It determines how to alter the weights between model neurons.

Even though the Hebbs Law is an unsupervised learning technique and this papers focus is neural network supervised learning, Hebbs Law introduced influential concepts for neural networks, neural networks as a learning algorithm (through the weighting of inputs).

The change in the weights with an input vector x and an excitement state $Y(x)$ can be represented by:

$$\Delta w_i = \varepsilon Y(x) x_i \tag{7.2}$$

where:

w_i are the input weights
x_i is the input vector
$Y(x)$ is the excitement state as defined in Section 7.3.1

7.3.3 The Perceptron

The next major advance was the perceptron, introduced by Rosenblatt (1958). Rosenblatt extended the McCulloch and Pitts (1943) neuron by replacing this step function with a continuous function that maps Z to Y. The continuous function mapping Z to Y makes training easier. The Rossenblatt neuron is known as the perceptron.

The perceptron represents the simplest form of a neural network used for the classification of linearly separable patterns. Basically, it consists of a single artificial neuron with adjustable synaptic weights and bias. The perceptron built around a single artificial neuron is limited to performing pattern classification with only two classes. For the perceptron to function properly it is necessary that the two classes are linearly separable enough, otherwise the corresponding decision is beyond the computing capabilities of the perceptron (Leondes, 2003).

Perceptron Algorithm

The perceptron algorithm calculates the weighted sum of its inputs:

$$Z = \sum w_i x_i - \mu \tag{7.3}$$

where:
w_i is the weight associated with the i_{th} input (attribute) x_i
μ is the bias term
Z is know as the perceptron potential

Rosenblatts perceptron calculates its analog output from its potential. Unlike the McCulloch-Pitts neuron whose binary output is represented by the sigmoid function, the perceptrons output can be represented by many different functions. The output function $g(Z)$ is known as the activation function. The potential and the activation function can be seen in Figure 7.3 below (Jones, 2004).

There are different functions that can be used for the activation function $g(Z)$. The most common two are the identity function and a sigmoid function. Examples of the sigmoid function are hyperbolic tangent , arc tangent, squash activation function and

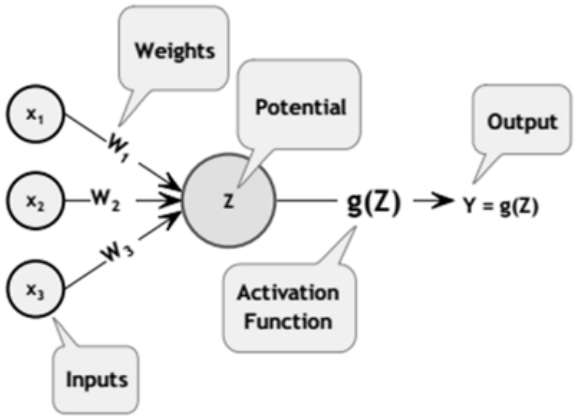


Figure 7.3: A Neural Net Perceptron

the logistic function. For neural networks for classification the logistic function is the most common. It is the activation function found in the Neural Network (nnet) R-Studio package and can be seen in Ripley (2016). This can be seen in Appendix ??.

The logistic function $g(x)$, maps the potential into the range 0 to 1:

$$g(x) = \frac{1}{1 + e^{-x}} \tag{7.4}$$

Since $0 < g(x) < 1$, the logistic function can be used to output probabilities.

7.4 Neural Network Architecture

A neural network consists of highly interconnected neurons such that each neurons output is connected through weights to the other neurons or back to itself. How the neural network processes data depends on the structure that organises the connections of the neurons in the network. The simplest neural network consists of a single neuron, while more complex networks will have two or more neurons. These neurons are arranged in layers and within the layer the neurons may be fully interconnected or not interconnected at all. The arrangement of the neurons in layers and how the neurons connect with each other is called the Neural Network Architecture. Neural Networks can be classified as single or multilayer. They can connect via a feedforward network or a recurrent network (Leondes, 2003).

We will focus on neural networks for classification, in particular binary classification,

either the company will default or it wont. A binary classification problem can be solved using a neural network with one logistic output. This output estimates the probability that the input data belong to one of the two choices, either default or non-default. A multilayer feed-forward network with a single hidden layer is used in the nnet R-Studio package, see Ripley (2016). This can be seen in Appendix **??**

7.5 Multilayer Feed-Forward Network

A Neural Network has a Feed-Forward type structure when the signal moves from input to output, passing through all the networks hidden layers. The flow of these computations is in one direction. The outputs of neurons are connected to the next layer and not to previous ones. These networks have the property that the outputs can be expressed as a function of inputs.

The simplest type of feed forward network is a Single-Layer Feed-Forward Network, there is an input layer of the source nodes, followed by the output layer of computing nodes. This can be seen in Figure 7.4 (Leondes, 2003).

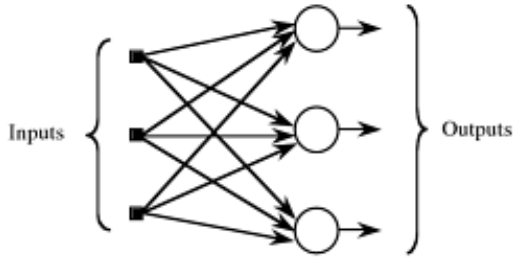


Figure 7.4: A Single Layer Feed Forward Network

When there are one or more layers this is known as a Multilayer Feed-Forward Network. The first layer is called the input layer, the last layer is called the output layer and the layers between are hidden layers. The hidden layers are different layers of neurons, their role being to act between the input layer and the output layer, so that the network performance is improved Leondes (2003). Figure 7.5 illustrates such a network with a single hidden layer. This is the same as the nnet R package Ripley (2016) which uses a single hidden layer.

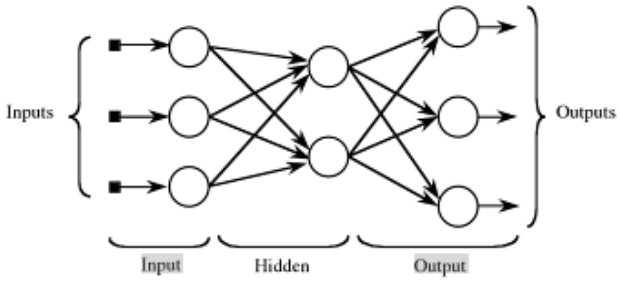


Figure 7.5: A Multi Layer Feed Forward Network

7.6 Multilayer Feed Forward Algorithm Description

Each perceptron in a particular layer is connected with all perceptrons in the next layer. The connection between the i^{th} and j^{th} perceptron is by the weight co-effient w_{ij}. This can be seen in Figure 7.6.

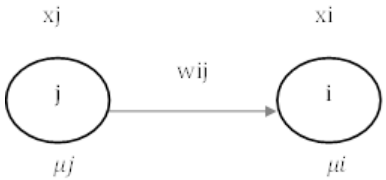


Figure 7.6: Illustration of the connection between two neurons i and j

The output perceptrons use activation function g_i to produce the outputs Y_i Svozil et al. (1997) where

$$Z_i = \sum w_i x_i - \mu_i \tag{7.5}$$

$$Y_i = g(Z_i) \tag{7.6}$$

Recall from Equation 7.4 that the activation function used is the logistic function

$$g(x) = \frac{1}{1 + e^{-x}} \tag{7.7}$$

7.7 Training

For supervised neural networks, the algorithm is being explicitly trained to produce the correct outputs for the given inputs. This is a process in which the labelled training data (target output) is presented to the network as an example, which is then compared to the output of the neural network algorithm (calculated output). The network is trained iteratively by adjusting the weights in the network so that the network output matches the labelled target. This iterative adjustment is called neural network training (Moonasar, 2007).

An Error Function is used to measure the difference between the target output and the calculated output. The Error Function is then minimised and the combination of weights which minimizes the error function is considered to be a solution of the learning problem.

7.7.1 Training Algorithm

The Error Function E is calculated as the difference between the target output and the calculated output.

$$E = \sum_{0} (Y_o - \hat{Y}_0)^2 \tag{7.8}$$

where:
o is the output of all the peceptrons
Y_o is the calculated output as a summation across all output perceptrons
$\hat{Y}_0$ the target output calculated as a summation across all output perceptrons

Recall from Equations 7.5 and 7.6 that Y_o is a function of the threshold coefficients μ_i and weights w_{ij}. The algorithm training is done by adjusting or training the threshold coefficients μ_i and weights w_{ij}.

Hence:

$$E = E(w_1, w_2, ..., w_p) \tag{7.9}$$

where:
E is the Error Function
$w_1, w_2, ..., w_p$ are the weights

so μ_i and weights w_{ij} are adjusted to minimise the sum of the squared differences, between the computed and required values. This is an iterative process.

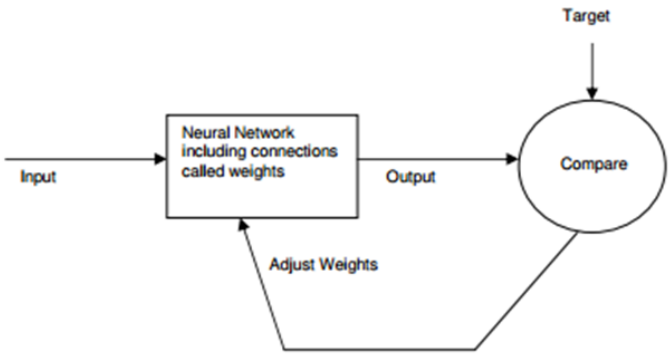


Figure 7.7: Training of the Neural Network

Figure 7.7 explains how the neural network is trained by adjusting the weights to meet the labelled target. For each input vector x we calculate the output vector $g(Z_i)$. The difference between the output vectors and the target vectors are the errors. The algorithim aims to find values for the weights w_i such that the sum of the errors squared is minimized (Moonasar, 2007) .

7.7.2 Back-propagation training algorithm

The Back-propagation algorithm is used to find the minimum of the Error Function E defined in Equation 7.8 in weight space. It achieves this by using the method of Gradient Descent. The combination of weights which minimizes the Error Function is considered to be a solution of the learning problem. Each weight is adjusted, according to its contribution to the error. This process occurs iteratively for each layer of the network, starting with the last set of weights, and working back towards the input layer (hence the name Back-propagation) (Rojas, 1996).

This method requires computation of the gradient of the error function at each iteration step, we must guarantee the continuity and differentiability of the error function

(Rojas, 1996).

E is a continuous and differentiable function furthermore it is a function of the weights w_i. For a continuous and differentiable function the Steepest Descent Algorithm can be used to find a local minimum of the function. The Gradient Descent will start with an initial guess of the solution, which is usually randomly chosen weights. The gradient is then calculated at that point. It will then step the solution in the negative direction of the gradient and we repeat the process. The algorithm will eventually converge where the gradient is zero (which correspond to a local minimum)(Rojas, 1996).

The gradient of the function will be calculated by the partial derivative of E w.r.t w_i:

$$\nabla E = \frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \frac{\partial E}{\partial w_3} ..... \frac{\partial E}{\partial w_p} \tag{7.10}$$

Each weight is updated using the increment

$$\Delta w_i = -\gamma \frac{\partial E}{\partial w_i} \tag{7.11}$$

where γ represents a learning constant

The minimum of the Error Function is where $\Delta E = 0$

7.8 Convergence

The best way to determine whether your network has reached the best set of weights for your training data is to validate the results obtained in Section 7.7 using the test set of data. The Neural Network needs to generalize its inputs, so that it can correctly classify queries not only for the training data, but also for other examples. The performance can be evaluated (on the test set) by performance measures mentioned in Chapter 6:

- ROC Curves
- Lift Charts
- Calibration Plots
- Confusion Matrix

The training time for any network is important. If the network is trained too long on the training data, it will overfit to the training data, which means that it will only classify examples correctly that are in the training data set (Li et al., 2012)

7.9 Summary

Neural networks are non linear mathematical models inspired by the biological neural networks. Neural Networks are especially useful for classification. Neural networks are trained, so that a certain input will lead to a target output. The training is preformed on the training data set which has labelled output examples. The neural network is trained by adjusting the threshold coefficients and the weights so that the sum of squares error between the target output and the network calculated output is minimised. This is done by using the Gradient Decent Algorithm. The best way to determine whether the algorithm has converged is by applying it to the test data and checking whether it classifies the examples correctly.

Chapter 8

Support Vector Machines

8.1 Introduction

Support vector machines (SVM) are the most recent development from the machine learning community. In machine learning, support vector machines (SVMs) are supervised learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. SVM takes a set of input data and predicts, for each given input, which of two possible classes comprises the input, making the SVM a non-probabilistic binary linear classifier.

Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. The output of an SVM model is a representation of the two categories, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite- dimensional space, which can be used for classification into the two different data classes.

Support vector machines were introduced by Cortes and Vapnik (1995). Cortes and Vapnik (1995) introduces the concept of the optimal separating hyperplane which is used to catagorise the data into separate classes. Below is a formulation of the support vector machine algorithm as proposed by them.

Below is a detailed description of the mathematical theory on support vector machines. The theory for this section is taken from Dikkers (2005) , Burges (1998), Berwick (2009) and Auria and Moro (2008). Vapnik (2000) Provides a detailed description of the optimal hyperplane for linear separable data as well as for non-

separable data.

This chapter is organised as follows, the case of a linear support vector machines, where the score function is still linear and parametric, will first be explained. This is in order to introduce the concepts of support vector machines in a simplified context. Following that the support vector machine will be made non-linear and non-parametric by the introduction of a kernel.

8.2 Support Vector Machines Basics

The Support Vector Machines (SVM) is a supervised machine learning algorithm for binary classification problems. SVMs classify a binary output, class 1 or class 2 according to its score value, which is a function of the input data. The function used to classify is neither linear nor parametric. SVMs were developed for binary classification but they can be extended to non binary cases.

The output of SVM is y_i, $i = 1, 2, 3, ...n$ where n is the number of input vectors. y_i is dependant on n input vectors x_i, $i = 1, 2, 3, ...n$ These input vectors are a set of labelled points known as the input space.

$$y_i = \begin{cases} 1, & \text{if } x_i \text{ is in class 1} \\ -1, & \text{if } x_i \text{ is in class 2} \end{cases}$$

In the case of using SVM for default prediction the problem is as follows: a company will either be classified as default or non-default according to its score value, which is a function of selected financial ratios. The input data is x_i, $i = 1n$ n being the number of companies and $x_i s$ are the financial ratios.

$$y_i = \begin{cases} 1, & \text{if } x_i \text{ default} \\ -1, & \text{if } x_i \text{ does not default} \end{cases}$$

The SVM will try to find the optimal hyperplane that separates the two classes (default of non default). A hyperplane in R^n is a subspace in n-1 of the form:

$$H = x : a_1 x_1 + a_2 x_2 + a_3 x_3 + ... + a_n x_n = \underline{b} \tag{8.1}$$

where: $\underline{a}$, $\underline{b} \in R_n$

A detailed description about hyperplanes is found in Stanley (2007) and in Schbel (1999). The optimal hyperplane is defined on the labelled training data and when unlabelled data is input into the algorithm the output will be either default or non-default.

8.3 Separable Data in a Two Dimensional Space

Consider this example in R^2, let x_1 and x_2 be two financial ratios for n different companies. Either a company will be classified as default or non-default based on the SVM score which is a linear function of the inputs x_1 and x_2.

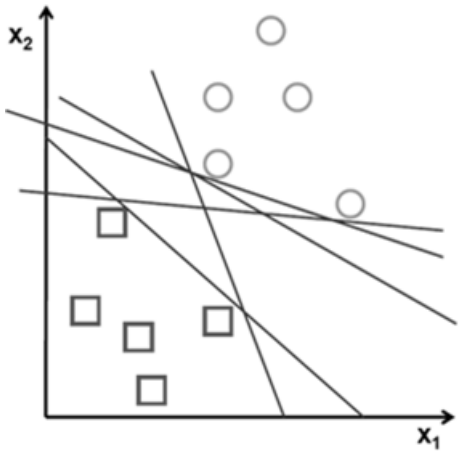


Figure 8.1: Different Linear Decision Function

Figure 8.1 is a graphical representation of the different possible ways of separating class 1 and class 2. Let the circles represent the class of default and the squares represent the class of non-default. The equation of the line separating the default from the non default companies is needed. Noting that in R^2 this is a decision line but in higher dimensions this will be a decision plane or hyperplane. There are many solutions for the equation of the decision line to separate the default from the non-default companies. SVM will find the optimal solution for the equation of the decision line, known as the optimal hyperplane.

The SVM defines the criterion to be looking for a decision surface that is maximally far away from any data point. The minimum distance of any point to the decision lines is referred to as the margin. The optimal hyperplane (in two dimensions) is the solution for the decision line that gives the maximum margin. A line is considered sub optimal if it passes too close to the points because it will be noise sensitive and

it will not generalize correctly. The support vectors are the points which fall within this margin.

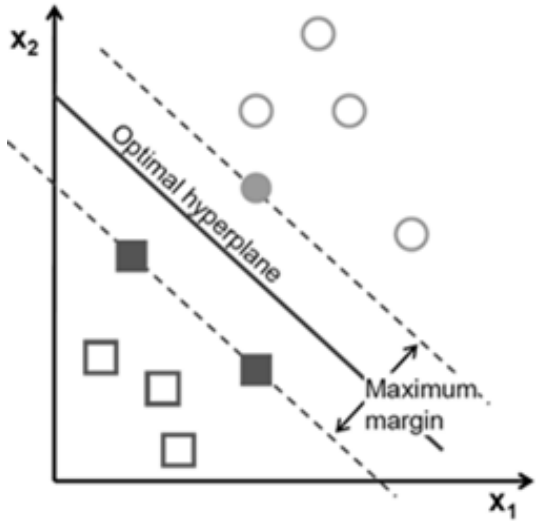


Figure 8.2: The Optimal Hyperplane

Figure 8.2 is a graphical representation of the optimal hyperplane. The dark points are points on the sub optimal lines are called Support Vectors.

8.4 Algebra

This section introduces the concept of the support vector machine algebra with the algebra behind linear support vector machines. It then describes the soft margin hyperplane and lagrangian multipliers.

8.4.1 Linear SVM

Let the data be separable, i.e there exists a hyperplane H that separates the data.

$$\begin{aligned} w \cdot x_i - b \geq \ when\, y_i = 1 \\ w \cdot x_i - b \leq \ when\, y_i = -1 \end{aligned} \tag{8.2}$$

Let $H1$ and $H2$ be the hyperplanes:

$$\begin{aligned} H1 &: w \cdot x_i - b = 1 \\ H2 &: w \cdot x_i - b = -1 \end{aligned} \tag{8.3}$$

Define:
d_+ = the shortest distance to the closest positive point
d_- = the shortest distance to the closest negative point
Then the margin between the hyperplanes $H1$ and $H2$ is $d_+ + d_-$.

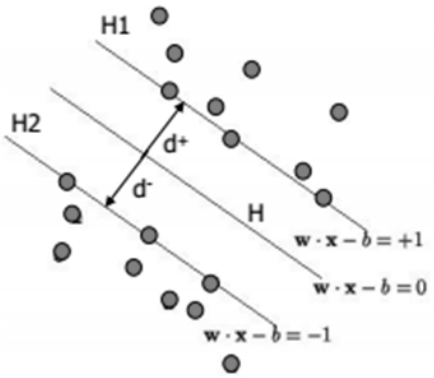


Figure 8.3: Illustration of a Margin

Figure 8.3 is a graphical representation of a margin. The distance between H and $H1$ is $\frac{|wx+b|}{||w||} = \frac{1}{||w||}$, so the distance between $H1$ and $H2$ is $\frac{2}{||w||}$.

We wish to find the optimal hyperplane, the hyperplane with the maximum distance $\frac{2}{||w||}$. Hence we want to minimise $||w||$. There is a condition that there must be no data points between $H1$ and $H2$. This condition is represented mathematically by:

$$\begin{aligned} x_i^T w + b \geq \ 1 \, when \, y_i = 1 \\ x_i^T w + b \leq \ -1 \, when \, y_i = -1 \end{aligned} \tag{8.4}$$

The classifier itself that identifies new patterns now becomes:

$$y = sign[x^T w + b] \tag{8.5}$$

Equation 8.4 can be rewritten as $y_i(x_i.w) \geq 1$. Also replace $||w||$ with $\frac{1}{2}||w||^2$

Hence the optimisation function is defined by:

$$Min\,\frac{1}{2}||w||^2 \tag{8.6}$$

8.4.2 The Soft Margin Hyperplane

In a non-perfectly separable case the margin is know as soft. The training data cannot be separated without error, this means there will be in sample classification errors. These in sample classification errors occur and also need to be separated. In order to deal with these errors introduce a slack variable ξ. Note that for no misclassification $\xi = 0$ and for a classification error ξ is positive. The misclassification error needs to be minimised.

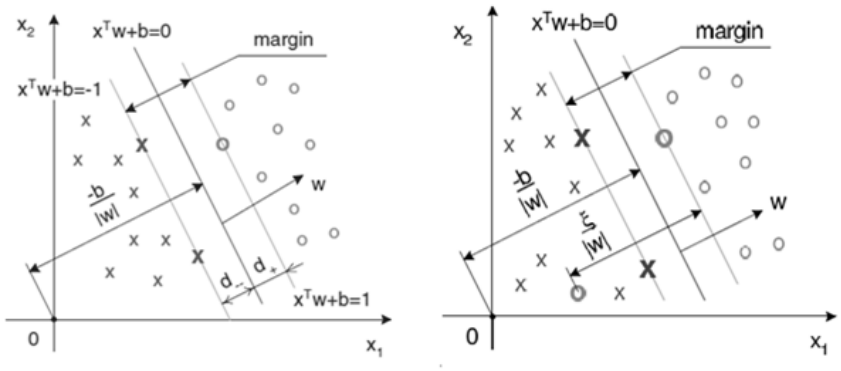


Figure 8.4: The Separating Hyperplane

Figure 8.4 shows the separating hyperplane $x^T w + b = 0$ with the margin in a linearly separable (left) and non-separable (right) case. Here crosses denote "Default" companies and zeros are the "Non Default" companies.

We can impose another constraint on 8.4 the condition that there must be no data points between $H1$ and $H2$ except for misclassification errors. This can be represented mathematically by:

$$\begin{aligned} x_i^T w + b \geq\ & 1 - \xi_i \, when \, y_i = 1 \\ x_i^T w + b \leq\ & -1 + \xi_i \, when \, y_i = -1 \end{aligned} \tag{8.7}$$

In summary we have 8.7 as $y_i(x_i^T w) \geq 1 - \xi_i$

Furthermore introduce the sum of the misclassification errors into the objective function C. The parameters C are called capacity. They are related to the width of the margin. The smaller the C, the bigger margins are possible. For a classical SVM C = constant.

Introducing the misclassification errors C into Equation 8.6 it becomes:

$$Min\,\frac{1}{2}||w||^2 + C\Sigma_i\xi_i \tag{8.8}$$

Thus the optimisation problem now becomes:

$$\begin{aligned} Min\,\frac{1}{2}||w||^2 + C\Sigma_i\xi_i \\ s.t\,y_i(x_i.w) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{aligned} \tag{8.9}$$

This is a constrained optimization problem. Solved by lagrangian multiplier method.

8.4.3 Lagrangian Multipliers

A detailed description of lagrangian multipliers is found in Hoffman and Bradley (1999), Himmelblau (1972) and Hoa (1984). A brief description of the Lagrangian problem is provided here.

The lagrange multiplier theorem lets us translate the original constrained optimization, known as the primal into an ordinary system of simultaneous equations (unconstrained optimization problem) at the cost of introducing an extra variable. The unconstrained optimization problem is known as the dual.

If we have a constrained optimization problem we have:

$$\text{Minimize } f(x) \text{ subject to } g(x) = 0 \tag{8.10}$$

There is a condition that f and g need to have continuous first partial derivatives.

We define the Lagrangian L as:

$$L(x, \lambda) = f(x) - \lambda g(x) \tag{8.11}$$

The method of lagrangian multipliers states that if $f(x_0, y_0)$ is a maximum of $f(x, y)$ for the original constrained problem, then there exists λ_0 such that (x_0, y_0, λ_0) is a stationary point for the lagrange function. i.e partial derivatives are 0. This can be represented mathematically by:

$$\nabla L = 0 \tag{8.12}$$

Lagrangian Formulation

The lagrangian formation of Equation (8.9) is defined by Hardle et al. (2007).

$$L_P = \frac{1}{2}||w||^2 + \Sigma_i \xi_i C_i - \Sigma_i \alpha_i \{\gamma_i (x_i^T w + b) - 1 + \xi_i\} - \Sigma_i \mu_i \xi_i \tag{8.13}$$

where L_P is the primal
$\alpha_i \geq 0$ are the lagrangian multipliers for the constraint in (8.9)
γ_i and μ_i are the lagrangian multipliers for the constraint in (8.9)

Solving the Lagrangian Equation for SVM

The primal lagrangian L_P is a quadratic optimization problem. The solution is given by the sadle point minimized with respect to w, b and ξ and maximised w.r.t α and u The details to solving the lagrangian is given by Dikkers (2005). We start with

$$\min_{w,b,\xi} \max_{\alpha,u} L_P(w, b, \xi, \alpha, u) \tag{8.14}$$

To solve Equation 8.14 we have to let

$$\begin{aligned} \frac{\partial L_P}{\partial w} = 0 \Rightarrow w = \sum_i \alpha_i \gamma_i x_i^T \\ \frac{\partial L_P}{\partial b} = 0 \Rightarrow sum_i \alpha_i \gamma_i = 0 \\ \frac{\partial L_P}{\partial \xi} = 0 \Rightarrow 0 \leq \alpha_i \leq \sum_i C_i \end{aligned} \tag{8.15}$$

The Karush Kuhn Tucker Conditions in Section 8.4.3 described below are used to solve Equation 8.13 by deriving the dual L_D of the primal L_P.

The dual lagrangian L_D is

$$L_D = \frac{1}{2} w(\alpha)^T w(\alpha) - \Sigma_i \alpha_i - \Sigma_i \delta_i \alpha_i + \Sigma_i \gamma_i (\alpha_i - C_i) - \beta \Sigma_i \alpha_i y_i \tag{8.16}$$

where $\alpha_i, \delta_i, \gamma_i$ and β are Lagrange multipliers for all i . The function $w(\alpha)$ is a scalar product in some Hilbert space. A description of Hilbert Spaces can be seen in Berberian (2000).

For a linear SVM:

$$w(\alpha)^T w(\alpha) = \Sigma_i \Sigma_j \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{8.17}$$

The classifier construction problem has simplified from the primal $L_P(w, b, \xi, \alpha, \mu)$ to the dual $L_D(\alpha)$. These α_i are referred to as the support values and give the relative weight of their corresponding support vector x_i.

Finally by solving the above equations we obtain the optimal decision function $D(x)$, the output of which will be default or non default for a particular company as a function of x_i, $i = 1n$ the financial ratios of the company. We get

$$D(x) = sign(\Sigma_j \alpha_j y_j (x_j^T x) + b) \tag{8.18}$$

Karush Kuhn Tucker Conditions(KTC)

A description of how the KTC condition relates to lagrangian multipliers is found in Himmelblau (1972). The Kuhn Tucker conditions state that at any local constrained optimum, no change in the variables can improve the value of the objective function. The KTC are closely related to lagrangian multipliers.

Let the problem be

$$\begin{aligned} & Min\, f(x) \\ \text{Subject to } & h_i(x) = b_i \; i = 1....m \\ \text{and } & g_j(x) \leq c_j \; j = 1....r \end{aligned} \tag{8.19}$$

Define lagrangian multipliers λ_i associated with j equalities u_j for the inequalities, and form the lagrangian function

$$L(x, \lambda, u) = f(x) + \Sigma_i \lambda_i [h_i(x) - b_i] + \Sigma_j u_j [g_j(x) - c_j] \tag{8.20}$$

Then if x^* is a local minimum of the problem in Equation 8.22, then there exists a vector of lagrangian multipliers λ^* and u^* such that x^* is a stationary point of the function $L(x, \lambda^*, u^*)$ i.e.

$$\nabla_x L(x^*, \lambda^*, u^*) = \nabla f(x^*) + \Sigma_i \lambda_i^* [\nabla h_i(x^*) - b_i] + \Sigma_j u_j^* [\nabla g_j(x^*) - c_j] \tag{8.21}$$

and complementary slackness holds for the inequalities such that

$$\begin{aligned} u_j^* &\geq 0 \\ u_j^* [g_j(x^*) - c_j] & \end{aligned} \tag{8.22}$$

8.5 Non-linear SVM

The above analysis was done for linear separable data. We will now look at the case where the data is nonlinear separable. Lin (1994) extended that the above can be extended to non-linear decision hyper surfaces for classification of nonlinearly separable data. This is done by a non-linear mapping of the input vectors to a high dimensional space.

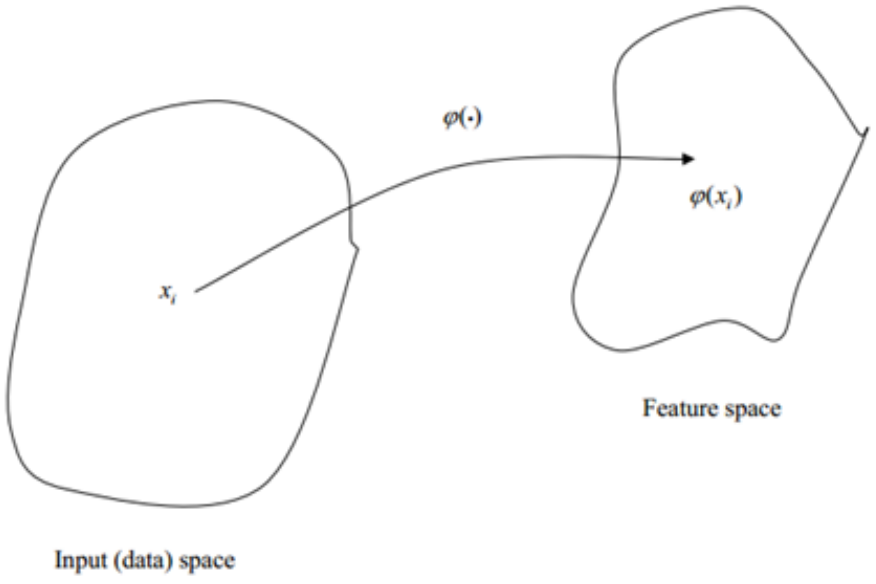


Figure 8.5: Mapping of non-linear SVM

Figure 8.6 is a non-linear mapping into a higher dimensional space. The mapping into the feature space is done by the function $\theta()$. Hence $x_i \longrightarrow \theta_(x_i)$. If we substitute $x \longrightarrow \theta_(x)$ into the Dual Problem (Equation 8.25) we get the $x's$ appearing as an inner product so the mapping will look like this:

$$x_i^T x_j \longrightarrow \theta_(x_i)^T \theta_(x_j) \tag{8.23}$$

8.6 The Kernel Function

Kernel functions can be used in many applications, the most common being pattern recognition as they provide a simple bridge from linearity to non-linearity for algorithms which can be expressed in terms of dot products. For further reading on kernel functions refer to Hofmann et al. (2008).

Kernel methods make use of kernel functions to map the data into higher dimensional spaces in order to structure the data better. Kernel methods uses the eigenvectors and eigenvalues of the data to draw conclusions from the directions of maximum variance to construct inner product kernels There are no constraints on the form of this mapping. Substituting a higher dimensional kernel in place of inner products is called the Kernel Trick. The Kernel Trick is based on Mecer theorem (Q Minh and Yao, 2006).

The Kernel Trick would be taking x, transforming it into some feature space $\theta(x)$, and replacing it with the kernel function such that

$$\theta(x_i^T)\theta(x_j) = K(x_i x_j) = K_{ij} \tag{8.24}$$

The reason for doing this is that we can compute K very cheaply without ever explicitly forming $\theta(x_i)$ or $\theta(x_j)$. This means a huge computational saving. The only reason for doing the kernel trick is to save computational expense in computing large basis expansions by directly computing kernel functions.

By the introduction of the kernel function SVMs are able to apply simple linear classifiers on data mapped into a feature space and to provide a method to compute a non-linear classification function without big computational effort as the complexity always remains only dependent on the dimension of the input space.

The concepts described above can be seen in Figure 8.6 where the Quadratic Kernel is used to map points from a 2 dimensional data space into a 3 dimensional data space.

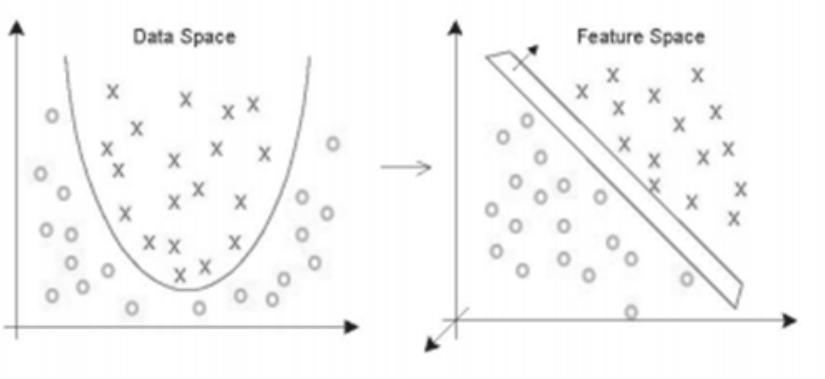


Figure 8.6: Mapping from a 2 Dimensional Data Space into a 3 Dimensional Feature Space

8.7 Non-Linear SVM Equation

Substitute $x_i^T x_j = \theta(x_i^T)\theta(x_j) = K(x_i, x_j)$ into Equation 8.25. The Dual Lagrangian of our optimisation problem then becomes

$$\begin{aligned} L_D &= \Sigma_i \alpha_i - \Sigma_i \Sigma_j \alpha_i \alpha_j y_i y_j \theta(x_i^T)\theta(x_j) \\ L_D &= \Sigma_i \alpha_i - \Sigma_i \Sigma_j \alpha_i \alpha_j y_i y_j K(x_i, x_j). \end{aligned} \tag{8.25}$$

8.8 Choice of the Kernel

The most common Kernel functions are (Karatzoglou and Meyer, 2006)

- Linear: $K(x, x') = (x, x')$
- Polynomial: $K(x, x') = (\gamma(x, x') + r)^d, \gamma > 0$, r constant
- Gaussian Radial Basis Function (RBF): $K(x, x') = e^{-\gamma\|x-x'\|^2}, \gamma > 0$
- Laplace Radial Basis Function (RBF): $K(x, x') = e^{-\gamma\|x-x'\|}, \gamma > 0$
- Sigmoid $K(x, x') = tanh(\gamma(x, x') + r)$, r constant

8.9 Summary

The support vector machines is a supervised machine learning algorithm for binary classification problems. SVMs classify a binary output, class 1 or class 2 according to its score value, which is a function of the input data. SVMs are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. The solution to the SVM problem is the optimal hyperplane (the decision line that gives the maximum margin).

Chapter 9

Advantages and Disadvantages of SVM and Neural Networks

9.1 Introduction

All classification techniques have advantages and disadvantages, which are more or less important according to both the data being analysed and the problem that needs to be solved. Both support vector machines and neural networks are a useful tool for default analysis because of the non-regularity in the data, which is not normally distributed nor does it have a known distribution. This chapter will explore the advantages and disadvantages of support vector machines and neural networks as well as providing a comparison of neural networks and support vector machines.

9.2 Advantages of Support Vector Machines

- By introducing the kernel, SVMs gain flexibility in the choice of the form of the threshold separating one class from another. The function does not need to be linear nor even have the same functional form for all data, since its function is non-parametric and operates locally.
- The kernel implicitly contains a non-linear transformation. The transformation occurs implicitly and human expertise judgement beforehand is not needed.
- Support vector machines provide a good out-of-sample generalization, therefore Support Vector Machines can be robust, even when the training sample has some bias (Auria and Moro, 2008).

9.3 Disadvantages of Support Vector Machines

- The input variables are still independent variables even though the polynomial kernel attempts to model the interaction among the variables
- SVMs function best on dense data that is, data with few to no missing values and data whose plotted points fall near each other
- There is no clear way on how to choose the optimal input feature subset for support vector machines (see Chapter 10).
- There is no clear way on how to set the best kernel parameters. The choice of the kernel will have a great effect on the accuracy of the support vector machine.
- There is a high computational cost when there is a large amount of features. This can be addressed by a feature selection algorithm (see Chapter 10).
- Support vector machines lack transparency of results. They do not deliver a parametric score function, since its dimension may be very high. The easiest way to interpret the results is graphically.
- The weights of the financial ratios are not constant. Thus the marginal contribution of each financial ratio to the score is variable (Auria and Moro, 2008).

9.4 Advantages of Neural Networks

- Neural networks can approximate complex non linear mappings. With enough nodes in the hidden layer it has been shown that Neural Networks can approximate any non linear function (Jain and de Silva, 1999).
- Neural networks (in particular back propagation) are easy to use with few parameters to adjust (Priddy and Keller, 2005).
- The algorithm is easy to implement.
- Neural networks are applicable to a wide range of problems.

9.5 Disadvantages of Neural Networks

- It is hard to know how many neurons and layers are necessary.
- Lack of transparency

- Learning can be slow.
- New learning will override old learning unless old patterns are repeated in the training process (Priddy and Keller, 2005).

9.6 Support Vector Machines vs Neural Networks

Theoretically support vector machines have many advantages over Neural Networks including:

- In training a neural network the sum of squares error between outputs and desired training outputs is minimised, therefore the class boundaries change as the initial weights change. When the training data is scare and linearly separable the generalization ability will deteriorate with a neural network but not a support vector machine. This is because a support vector machine is trained to maximise the margin (Abe, 2005).
- A significant advantage of support vector machines over neural networks is that neural networks suffer from multiple local minima. The solutions to SVMs are global and unique. Figure 9.1 shows a function with multiple local minima. A function has a local minimum point at $x*$ if $f(x*) \leq f(x)$ for all x in X within distance ε of $x*$ (Abe, 2005), (Olson and Dursun, 2008).

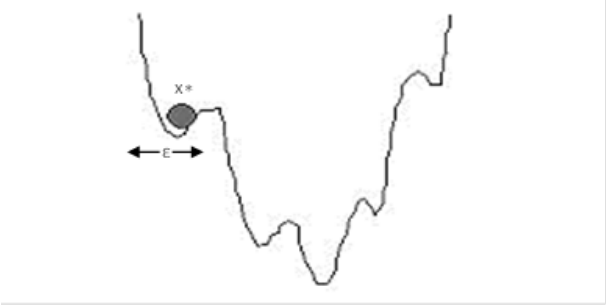


Figure 9.1: Multiple Local Minima

- The time to train a neural network is another significant disadvantage. In particular a back propagation network may require thousands of iterations through the training examples in order to adjust the weights correctly to find the optimal solution (Aktan, 2011a).
- Neural networks requires a larger sample size than SVM to train the model correctly (Wani, 2013).
- Support vector machines are less prone to over fitting than neural networks. Overfitting occurs when the model describes random error or noise instead of the underlying relationship. It will occur when the function is too closely fit to a limited set of data points (Chen and Odobez, 2013) (Aktan, 2011a).

9.7 Factorization

Factorization Machines (FM) are a new model class that combines the advantages of Support Vector Machines (SVM) with factorization models. Factorization modelling techniques are a class of widely successful models that attempt to find weighted low-rank approximations to the user-item matrix, where weights are used to hold out missing entries (Sammut, 2010).

FMs were introduced by (Rendel, 2010). They were designed to address some of the weakness of SVMs in Section 9.3. No training examples are required in the model parameters, making the models much more compact. Factorization machines perform extremely well on sparse data, including data of very high sparsity. As a trade-off, they do not perform well on dense data, so other algorithms are more suited to this class of data. SVMs function best on dense data that is, data with few to no missing values and data whose plotted points fall near each other.

Like SVMs, FMs are a general predictor working with any real valued feature vector. In contrast to SVMs, FMs model all interactions between variables using factorized parameters. Thus they are able to estimate interactions even in problems with huge sparsity (like recommender systems) where SVMs fail (Rendel, 2010).

Chapter 10

Feature Selection

10.1 Introduction

One of the challenges faced when using classification algorithms for default prediction, is which features or input variables to select. In this dissertation the input ratios are the financial ratios from a companys financial statements. There are more than a hundred variables generated by different combinations of items from the financial statements. Some of these input ratios may be irrelevant in predicting default and some of them may be redundant due to their high inter-correlation. support vector machines and neural networks do not offer the opportunity of an automated internal relevance detection and hence algorithms for feature selection play an important role.

Datasets with tens of thousands variables have become increasingly common in many real-world applications. When there are so many irrelevant and redundant features, the classification algorithms can suffer. The high-dimensional feature vectors can impose a high computational cost as well as the risk of over fitting when classification is performed. Furthermore reducing the dimensionality of the data by selecting a subset of the original variables can be beneficial in terms of storing and processing measurement.

Therefore it is necessary to reduce the dimensionality through ways like feature selection (Zhuo et al., 2008). Feature selection (also known as subset selection) is a process commonly used in machine learning, where a subset of features is selected from the available data for application of a learning algorithm. The feature subset must be the best combination of the original subsets that contributes most to the classification. So we prefer the model with the smallest possible number of parameters that adequately represent the data.

In the literature there are three general approaches to solve the feature selection problem: filter methods, wrapper methods and embedded methods (Guyon, 2003). Wrappers utilize the learning machine of interest as a black box to score subsets of variable according to their predictive power. Filters select subsets of variables as a pre-processing step, independently of the chosen predictor. Embedded methods perform variable selection in the process of training and are usually specific to given learning machines (Guyon and Elisseeff, 2003).

Filter approaches is recommended for fast data analysis. However, in order to better validate the results and to select fewer input variables Wrapper approaches is recommended. Wrapper approaches can choose the best input variables for building classifiers (Kumari and Swarnkar, 2011). They search for an optimal feature subset tailored to a particular algorithm and domain (Kohavi and John, 1997).

10.2 Advantages of Feature Selection

Advantages of feature selection are (Ladha and T.Deepa, 2011)

- Reducing the dimensionality of the feature space, to limit storage requirements and increase algorithm speed.
- It removes the redundant, irrelevant or noisy data.
- The immediate effects for data analysis tasks are speeding up the running time of the learning algorithms.
- Improving the data quality.
- Increasing the accuracy of the resulting model.
- Feature set reduction, to save resources in the next round of data collection or during utilization.
- Performance improvement, to gain in predictive accuracy.
- Data understanding, to gain knowledge about the process that generated the data or simply visualize the data.

10.3 Filter Methods

Filters select subsets of variables at independently of the prediction method. They carry out the feature selection process as a pre-processing step with no induction algorithm. The general characteristics of the training data are used to select features for example, distances between classes or statistical dependencies (Marono and Betanzos, 2007).

Examples of filter methods are principal component analysis or clustering analysis, chi-square, euclidean distance, i-test, information gain, correlation based feature selection, markov blanket filter and fast correlation based feature selection (Kumari and Swarnkar, 2011).

Illustrations of studies using Filter Methods for variable selection with reference to default prediction is Chen (2011) who used principle component analysis for variable selection and concluded that principal component analysis was able to use 80 percent fewer financial ratios and is still able to provide highly-accurate forecasts of financial bankruptcy.

10.4 Wrapper Methods

Wrapper Methods offer a simple and powerful way to address the problem of variable selection, regardless of the chosen machine learning algorithm. Wrapper methods use the classification method to score subsets of the variables. They utilize the classifier as a black box to score the subsets of features based on their predictive power. They assess subsets of variables according to their usefulness to a given predictor, they conduct a search for a good subset using the learning algorithm itself as part of the evaluation function. By using the learning machine as a black box, wrappers are remarkably universal and simple (Guyon and Elisseeff, 2003).

Examples are stepwise methods in linear regression. Sequential forward selection, sequential backward elimination, randomized hill climbing, genetic algorithms and partial swarm optimisation. In forward selection, variables are progressively incorporated into larger and larger subsets, whereas in backward elimination one starts with the set of all variables and progressively eliminates the least promising ones (Guyon and Elisseeff, 2003) (Kohavi and John, 1997).

One of the commonly used feature selection methods for small samples problems is recursive feature elimination (rfe) method or backward elimination. Rfe method utilizes the generalization capability embedded in support vector machines and is thus

suitable for small samples problems (Chen and Jeong, 2007)

Despite its good performance, RFE tends to discard "weak" features, which may provide a significant improvement of performance when combined with other features.

The genetic algorithm is a popular wrapper technique for feature selection with support vector machines. It selects subsets, achieving multicriteria optimization in terms of generalization accuracy and costs associated with the features (Yang, 1998). There have been many bankruptcy prediction studies on this topic. Mohamad et al. (2004) use the genetic algorithm evaluated by support vector machines, findings were that the genetic algorithm is good at getting high classification accuracy for training data of small or high dimension data. Frohlich and Chapelle (2003) also proposed a genetic algorithm based feature selection approach that for support vector machines. Huang et al. (2007) used the genetic algorithm approach to optimize the parameters and feature subset simultaneously, without degrading the support vector machine classification accuracy.

Other studies using wrapper based feature selection methods for default prediction with support vector machines where Hardle et al. (2007) made use of forward and backward selection applied to the accuracy ratio, the ratio of the areas between the ideal and the predictive model. Wang et al. (2013) applied particle swarm optimization algorithm is used to optimize parameters of the support vector machines. Results were that article partial swarm optimization with support vector machines has a distinct improvement in the aspect of accuracy rate as compared to support vector machines and neural networks with out partial swarm optimization. Hardle et al. (2011) used the forward selection procedure applied to the accuracy ratio in the support vector machine analysis.

The filter approach selects important features first and then the learning algorithm (for example SVM) is applied for classification. On the other hand, the wrapper approach either modifies SVM to choose important features as well as conducts training and testing or combines SVM with other optimization tools to perform feature selection. This can be illustrated in Figure 10.1 and Figure 10.2.

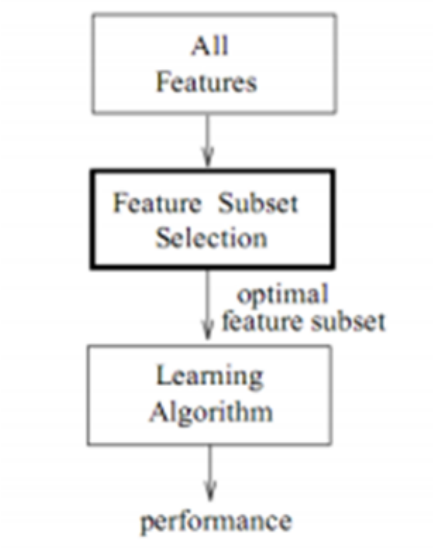


Figure 10.1: A filter model of feature selection

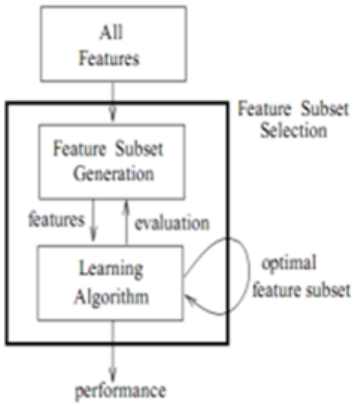


Figure 10.2: A wrapper model of feature selection

10.5 Embedded Methods

Another less popular method for feature selection are the embedded methods. Embedded methods incorporate variable selection as part of the training process and are usually specific to given learning machines (Guyon and Elisseeff, 2003).

10.6 Advantages and Disadvantages

10.6.1 Filter methods

Filter methods are fast, independent of the classifier (Kumari and Swarnkar, 2011). Other advantages include that they are easily scale to very high-dimensional datasets, they are computationally simple and need to be performed only once, and then the classifier can be evaluated. Disadvantages are that they do not interact with the classifier. They are often univariate or low-variate. This means that each feature is considered separately, thereby ignoring feature dependencies, which may lead to worse classification performance when compared to other types of feature selection techniques.

It is argued that, compared to wrappers, filters are faster. Still, recently proposed efficient embedded methods are competitive in that respect. Another argument is that some filters provide a generic selection of variables, not tuned for/by a given learning machine. Another advantage is the Filter Methods can be used as a preprocessing step to reduce space dimensionality and overcome overfitting (Guyon and Elisseeff, 2003).

10.6.2 Wrapper methods

Wrapper methods provide interaction between the subsets and classifier used. They therefore have the ability to take into account feature dependencies. The disadvantages are there is higher risk of overfitting than filter techniques and they are very computationally intensive.

Wrappers are often criticized because they seem to be a brute force method requiring massive amounts of computation, but it is not necessarily so. Efficient search strategies may be devised. Using such strategies does not necessarily mean sacrificing prediction performance. In fact, it appears to be the converse in some cases: coarse

search strategies may alleviate the problem of overfitting (Guyon and Elisseeff, 2003).

10.6.3 Embedded Methods

Embedded methods are less computationally intensive than wrapper methods, but they are less specific to a machine learning algorithm. They incorporate variable selection as part of the training process may be more efficient in several respects: they make better use of the available data by not needing to split the training data into a training and validation set; they reach a solution faster by avoiding retraining a predictor from scratch for every variable subset investigated (Guyon and Elisseeff, 2003).

10.7 Ratio Analysis

Ratio analysis is a form of financial statement analysis that is used to obtain a quick indication of a firm's financial performance in several key areas. The ratios can be categorized into liquidity ratios, profitability ratios and cashflow ratios. Below is a description of each category:

10.7.1 Liquidity Ratios

Liquidity ratios measure a firms ability to meet its short-term obligations. Investors often take a close look at liquidity ratios. If a company is consistently having trouble meeting its short-term debt it is at a higher risk of bankruptcy. Liquidity ratios are a good measure of whether a company will be able to comfortably continue as a going concern. Cash and other easily sellable liquid assets can be used to cover payables, short-term debt and other liabilities (Idris, 1998).

10.7.2 Profitability Ratios

Measures that indicate how well a firm is performing in terms of its ability to generate profit. They measure a companys ability to generate earnings relative to sales, assets and equity.

10.7.3 Cashflow Ratios

Cashflow ratios determine how much cash the company is generating from their sales and the amount of cash they have to cover obligations. Cash flow ratios are more reliable indicators of liquidity than balance sheet or income statement ratios such as the quick ratio or the current ratio. Sometimes a company that appear very profitable can actually be at a financial risk if they are generating little cash from these profits.

There have been many studies on cash flow and cash flow related variables for bankruptcy prediction. Jooste (2007) determined that bankrupt companies have lower cash flows than non-bankrupt companies furthermore findings were that income statement and balance sheet ratios were not enough to measure liquidity. A company can have positive liquidity ratios and increasing profits, yet have serious cash flow problems.

10.7.4 Solvency Ratios

Solvency ratios, also referred to as leverage ratios, indicate a company's financial health in the context of its debt obligations. They identify going concern issues and a firm's ability to pay its debt in the long term.

10.8 Description of Variables in this study

The purpose of this study as seen in Chapter 1 is to compare the performance of Machine Learning Techniques, in particular support vector machines and neural networks, to traditional statistical models in a south african context. This study uses a set of ratios selected to evaluate the financial performance of each company. The selection and the choice of ratios were based on three main conditions:

- The ratios have been frequently used in past studies
- The ratios have been shown to perform well in past studies
- The ratios represent liquidity ratios, profitability ratios, cash flow ratios and solvency ratios.

In order to select these variables a matrix of studies was created in Microsoft Excel. A simple count function is used to find the popular ratios used. This can be seen

in Figure 10.3. 23 financial ratios were chosen. These 23 ratios comprise of those representing liquidity ratios, profitability ratios, cash flow ratios and solvency ratios as can be seen in Figure 10.4.

Raio/ Paper	Altman (1968)	Razvan-Alexandru (2009):	Dikkers (2005):	Einarsson (2008):	Hardle et al. (2007)	Wang et al. (2013)	Aliakbari (2009)	Huang et al. (2004)	Min and Lee (2005)	Hardle et al. (2011)	Auria and Moro (2008)	Yap et al. (2012)	Mahdi (2013)	Wu et al. (2006)	Count
Net Income/Total Assets		1				1		1	1	1		1		1	7
Net Income/Sales		1				1	1	1	1	1	1		1	1	9
Operating Income/Total Assets		1					1			1				1	4
Operating Income/Sales		1					1	1	1	1	1			1	7
EBIT/TA	1	1					1			1		1		1	6
EBIT/Sales		1			1		1		1	1					5
Equity/Total Assets		1	1	1	1						1		1		6
Current Liabilities/Total Assets		1	1				1		1	1		1	1	1	8
Debt/Total Assets (Debt ratio)		1			1			1	1	1		1			6
EBIT/Interest Expenses (Interest Coverage Ratio)		1	1		1		1	1	1	1	1		1	1	10
Cash/Total Assets		1			1	1				1		1	1		6
Cash /curent liabilities		1			1					1		1	1		5
Cash Flow Ratio					1	1		1			1		1		5
Current Assets/Current Liabilities (Current Ratio)		1		1	1	1	1	1	1	1		1	1	1	11
Working Capital/Total Assets	1	1							1	1	1	1	1		7
Total Assets/Sales	1	1								1		1	1		5
Inventories/Sales		1			1		1				1				4
Accounts Receivable/Sales		1				1	1		1	1	1			1	7
Accounts payable / Sales		1					1			1				1	4
Gross Profit Margin			1		1		1	1							4
Total liabilities/shareholders equity (Debt to Equity)			1		1						1		1		4
ROA				1			1	1						1	4
Current assets -Inv) / Current Liab (Quick Ratio)			1		1	1		1	1	1			1		7

Figure 10.3: Variable Selection Matrix

Ratio	Group	Ratio Number
Net Income/Total Assets	Profitability	Ratio 1
Net Income/Sales	Profitability	Ratio 2
Operating Income/Total Assets	Profitability	Ratio 3
Operating Income/Sales	Profitability	Ratio 4
EBIT/Total Assets	Profitability	Ratio 5
EBIT/Sales	Profitability	Ratio 6
Equity/Total Assets	Solvency	Ratio 7
Current Liabilities/Total Assets	Solvency	Ratio 8
Debt/Total Assets	Solvency	Ratio 9
EBIT/Interest Expenses	Solvency	Ratio 10
Cash/Total Assets	Cashflow	Ratio 11
Cash /curent liabilities	Cashflow	Ratio 12
Free Cash Flow Ratio	Cashflow	Ratio 13
Operating Cash flow ratio	Cashflow	Ratio 14
Current Assets/Current Liabilities	Liquiidity	Ratio 15
Working Capital/Total Assets	Liquiidity	Ratio 16
Inventories/Sales	Profitability	Ratio 17
Accounts Receivable/Sales	Profitability	Ratio 18
Accounts payable / Sales	Profitability	Ratio 19
Gross Profit Margin	Profitability	Ratio 20
Debt/Equity ratio	Solvency	Ratio 21
ROA	Profitability	Ratio 22
Current assets – Inventory) / Current Liabilities	Liquiidity	Ratio 23

Figure 10.4: Final 23 Variables used in this study

A large number of authors have referred to the literature to select their final variables. This is the case for Deakin (1972) with Beaver (1966) model. This method of selecting predictors may be relevant when the aim is to look into the conditions of replicating existing models but such a strategy is not efficient for at least two reasons. First, the performance of a variable is not stable. Second, the predictive ability of one variable cannot be assessed in isolation, but in conjunction with others and with a specific modeling technique. A good variable or set of variables does not exist in itself; a good set of variables seems to be in part the result of the characteristics of the set itself and that of the fit between this set and the modeling method. Therefore choosing bankruptcy predictors *solely* for their popularity in the literature is not sufficient (du Jardin P, 2012).

du Jardin P (2012) also states that relying on statistics and selecting features on the basis of their scores in various statistical tests (variable ranking) has also proven to be insufficient.

Guyon (2003) outlines the limitation of variable ranking techniques and presents several situations in which the variable dependencies cannot be ignored. He points out that noise reduction and better class separation may be obtained by *adding* variables that are presumably redundant. He also stated that individual variables can have no separation power by themselves but taken *together*, the variables can provide good class separability. Guyon (2003) points out that even though perfectly correlated variables are truly redundant in the sense that no additional information is gained by adding them, very high variable correlation does not mean absence of variable complementarity. It was on this basis that a wrapper selection technique was used for variable selection rather than preforming statistical analysis for feature selection.

This study chooses a set of dominant ratios derived from a larger set of related ratios as a prior screening tool or pre processing step. There are hundreds of financial ratios that can be constructed from a set of financial statements and thus preforming an analysis of this type on all of them would be too exhaustive. These ratios were selected to evaluate liquidity, profitability, cash flow and solvency. After these 23 Ratios were selected a feature selection was used to choose the subset of variables in order to improve classification accuracy. A forward elimination wrapper method was used.

This is similar to the technique used by (Muller, 2008). Muller (2008) who first used the input variables from previous studies, followed by a feature selection technique for choosing the optimal input variables. Wu et al. (2006) also employed variables successful in bankruptcy prediction modelling as a screening tool to create a subset of variables after which a more formal evaluation was made.

10.9 Description of Input Ratios Used

1. Net Income/ Total Assets Also known as ROA (Return on Assets). It indicates how profitable a company is relative to its total assets.

2. Net Income/Sales Also known as Profit Margin. It is the number of Rands of after-tax profit a firm generates per Rand of sales.

3. Operating Income/Total Assets - Is the operating profit expressed as a percentage of the total assets, it indicates

4. Operating Income/Sales - Also known as the operating profit margin. It indicates how much profit a company makes after paying for production costs.

5. EBIT/Total Assets - Measures a company's earnings before interest and taxes (EBIT) against its total net assets. It is an indicator of how effectively a company is using its assets to generate earnings before contractual obligations must be paid.

6. EBIT/Sales- - Measures a company's earnings before interest and taxes (EBIT) against its sales. It indicates how much profit a company makes before contractual obligations must be paid.

7. Equity/ Total Assets - The amount of equity the company has relative to its total assets. It can be used to help determine how much shareholders would receive in the event of a liquidation.

8. Current Liabilities/Total Assets It measures the firm's ability to repay current debt by indicating the percentage of a company's total assets that are provided via current debt.

9. Total Liabilities/Total Assets Also known as the Debt Ratio. It measures the firm's ability to repay total debt by indicating the percentage of a company's total assets that are provided via total debt.

10. EBIT/Interest Expense Also known as the interest coverage ratio. It measures the companys ability to meet its interest payments. It is the company's earnings before interest and taxes (EBIT) over its interest payments.

11. Cash/Total Assets This measures the companys operating cash flows to its total assets. It can be seen as a companys cash return on total assets.

12. Cash /Current Liabilities - A measure of how well current liabilities are covered by the cash flow generated from a company's operations.

13. Free Cash Flow Ratio (Free Cash Flow/Net Debt) The free cashflow is calculated as operating cash flow minus capital expenditures. It represents the cash that a company is able to generate after laying out the money required to maintain it. The

14. Operating Cash Flow Ratio (Operating Cash Flow / Current Liabilities) - This ratio provides an indication of a company's ability to cover total debt with its cash flow from operations.

15. Current Assets/Current Liabilities Also known as the Current Ratio. It is a measure of the company's ability to pay back its short-term liabilities with its short-term assets.

16. Working Capital/Total Assets - Measures a companys ability to cover its short term financial obligations.

17. Inventories/Sales Measures a companys investment in inventory in relation to the amount of sales.

18. Accounts Receivable/Sales Measures your investment in accounts receivable in relation to your monthly sales amount. Accounts receivable is money which is owed to a company by a customer for products and services provided on credit. The accounts receivable to sales ratio helps you identify recent increases in accounts receivable.

19. Accounts payable / Sales - Measures how fast a company pays off its creditors (suppliers). Accounts payable is money owed by a business to its suppliers shown as a liability on a company's balance sheet.

20. Gross Profit Margin - Gross Profit/Sales. It measures how much out of every dollar of sales a company actually keeps in its gross profit.

21. Debt to Equity Ratio - is the ratio of total liabilities of a business to its shareholders' equity. It is a measure of a company's leverage.

22. ROA Net Income/ Total Assets. The return on assets (ROA) shows percentage how profitable a company's assets are in generating revenue.

23. Current assets Inventory) / Current Liabilities This ratio is known as the Quick Ratio. This ratio measures a companys ability to meet its short-term obligations with its most liquid assets.

10.10 Statistical Analysis of Financial Ratios in this study

10.10.1 Pairwise Comparison

The pairwise comparisons were started with a set of kernel density plots comparing the distributions of each of the 23 ratios by default status. Kernel density plots provide a representation of whether two distributions overlap substantially or not. Figure 10.5, 10.6, 10.7, 10.8, 10.9, 10.10, 10.11 and 10.12 are the Kernel density plots for the 23 variables. Red curves correspond to "Default" and green correspond to "Non Default".

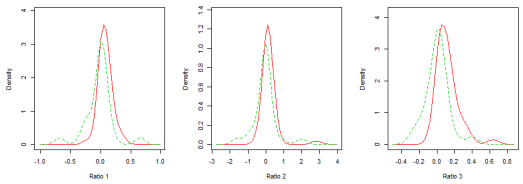


Figure 10.5: Kernel Density Plot

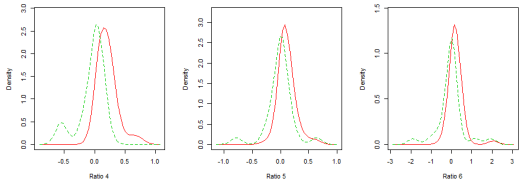


Figure 10.6: Kernel Density Plot

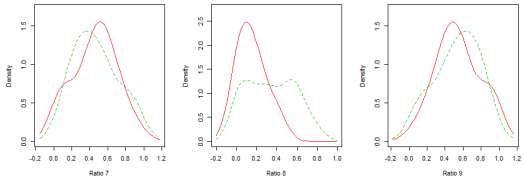


Figure 10.7: Kernel Density Plot

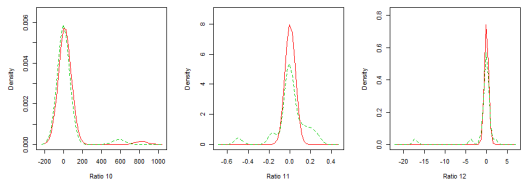


Figure 10.8: Kernel Density Plot

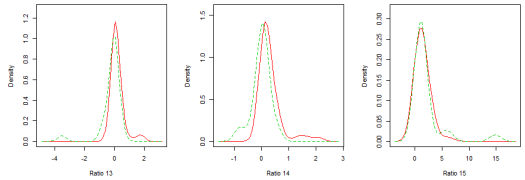


Figure 10.9: Kernel Density Plot

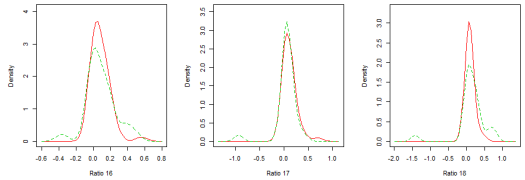


Figure 10.10: Kernel Density Plot

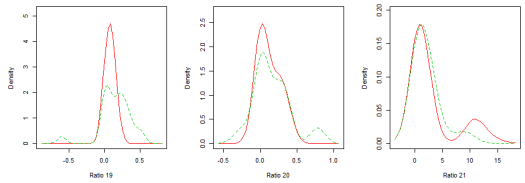


Figure 10.11: Kernel Density Plot

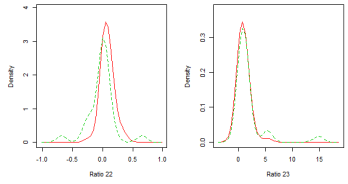


Figure 10.12: Kernel Density Plot

Largest visual differences in distributions (i.e. the smallest horizontal overlap) are observed for Ratios 2, 3, 4, 5, 8, 14 and 22.

A more formal test to compare two distributions for each ratio is the Mann-Whitney test for 2 independent samples. The null hypothesis is that true location shift equals zero, while the alternative hypothesis that it does not equal zero. P-values greater than 0.05 indicate no significant differences in distributions between the two groups (default and non-default) (Nachar, 2008). The results can be seen in Figure 10.13

Ratio	p-value	Ratio	p-value
1	0.002	13	0.002
2	0	14	0.001
3	0	15	0.55
4	0	16	0.617
5	0.001	17	0.535
6	0	18	0.254
7	0.995	19	0.022
8	0.002	20	0.589
9	0.995	21	0.995
10	0	22	0.002
11	0.751	23	0.157
12	0.835		

Figure 10.13: P-Values

The Ratios 7, 9, 11, 12, 15, 16, 17, 18, 20, 21, 23 have a P-value greater than 0.05. These ratios are unlikely to have a high discriminating power.

Figure 10.15, 10.15, 10.16 and 10.17 are the Box-and-Whisker Plots for Ratios 1, 2, 3, 4, 5, 6, 8, 10, 13, 14, 19, 22

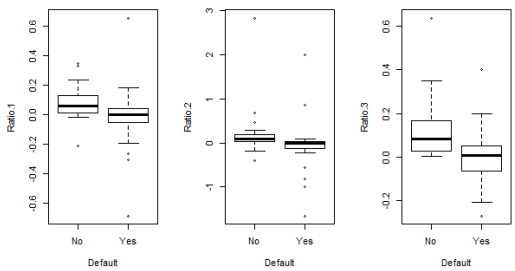


Figure 10.14: Box-and-Whisker Plot

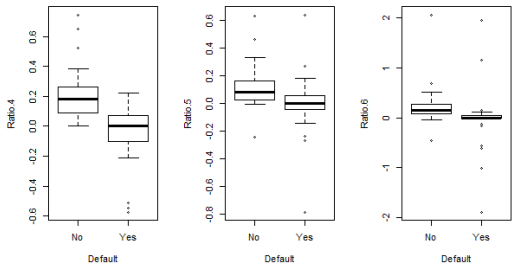


Figure 10.15: Box-and-Whisker Plot

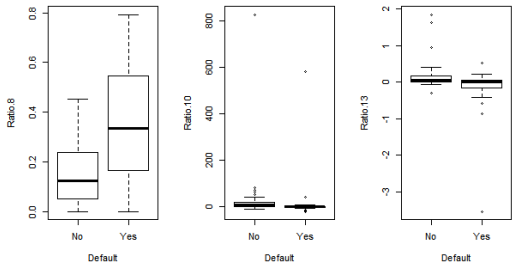


Figure 10.16: Box-and-Whisker Plot

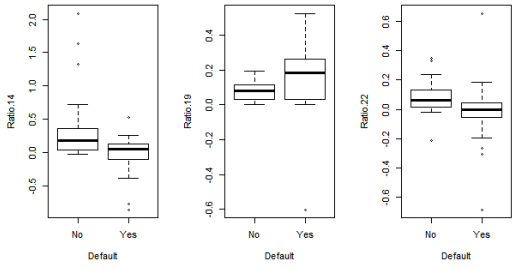


Figure 10.17: Box-and-Whisker Plot

10.10.2 Correlation Matrix

Figure 10.18 is the correlation matrix for the 23 financial ratios.

	Ratio 1	Ratio 2	Ratio 3	Ratio 4	Ratio 5	Ratio 6	Ratio 7	Ratio 8	Ratio 9	Ratio 10	Ratio 11	Ratio 12	Ratio 13	Ratio 14	Ratio 15	Ratio 16	Ratio 17	Ratio 18	Ratio 19	Ratio 20	Ratio 21	Ratio 22	Ratio 23
Ratio 1	1.00	0.71	0.61	0.43	0.96	0.78	0.46	-0.38	-0.46	0.18	-0.15	-0.10	-0.13	0.27	0.27	0.11	-0.06	-0.07	-0.15	0.22	-0.33	1.00	0.24
Ratio 2	0.71	1.00	0.17	0.41	0.62	0.97	0.40	-0.38	-0.40	0.01	-0.16	-0.23	-0.34	-0.09	0.33	-0.08	0.17	-0.08	-0.08	0.26	-0.20	0.71	0.33
Ratio 3	0.61	0.17	1.00	0.62	0.70	0.24	0.27	-0.15	-0.27	0.18	0.01	0.08	0.41	0.61	0.10	0.36	0.06	-0.01	-0.19	0.32	-0.29	0.61	0.04
Ratio 4	0.43	0.41	0.62	1.00	0.45	0.46	0.16	-0.43	-0.16	-0.03	0.05	0.04	0.25	0.27	0.06	0.08	0.10	-0.11	-0.26	0.39	-0.10	0.43	0.03
Ratio 5	0.96	0.62	0.70	0.45	1.00	0.71	0.43	-0.30	-0.43	0.15	-0.09	-0.07	0.00	0.39	0.26	0.18	-0.07	-0.02	-0.14	0.24	-0.34	0.96	0.23
Ratio 6	0.78	0.97	0.24	0.46	0.71	1.00	0.41	-0.42	-0.41	0.01	-0.21	-0.31	-0.33	-0.06	0.40	-0.06	0.08	-0.06	-0.08	0.30	-0.21	0.78	0.40
Ratio 7	0.46	0.40	0.27	0.16	0.43	0.41	1.00	-0.30	-1.00	0.37	-0.12	-0.27	-0.03	0.23	0.57	0.05	0.07	-0.14	-0.02	0.18	-0.80	0.46	0.53
Ratio 8	-0.38	-0.38	-0.15	-0.43	-0.30	-0.42	-0.30	1.00	0.30	-0.20	0.15	0.14	0.10	0.01	-0.17	0.30	0.19	0.31	0.38	0.17	-0.07	-0.38	-0.17
Ratio 9	-0.46	-0.40	-0.27	-0.16	-0.43	-0.41	-1.00	0.30	1.00	-0.37	0.12	0.27	0.03	-0.23	-0.57	-0.05	-0.07	0.14	0.02	-0.18	0.80	-0.46	-0.53
Ratio 10	0.18	0.01	0.18	-0.03	0.15	0.01	0.37	-0.20	-0.37	1.00	-0.06	-0.03	0.26	0.33	-0.02	-0.10	-0.07	-0.08	-0.14	-0.08	-0.15	0.18	0.00
Ratio 11	-0.15	-0.16	0.01	0.05	-0.09	-0.21	-0.12	0.15	0.12	-0.06	1.00	0.77	0.34	0.30	-0.54	0.00	-0.09	-0.08	-0.20	-0.28	-0.03	-0.15	-0.55
Ratio 12	-0.10	-0.23	0.08	0.04	-0.07	-0.31	-0.27	0.14	0.27	-0.03	0.77	1.00	0.34	0.36	-0.76	0.10	-0.01	-0.06	-0.16	-0.40	0.09	-0.10	-0.81
Ratio 13	-0.13	-0.34	0.41	0.25	0.00	-0.33	-0.03	0.10	0.03	0.26	0.34	0.34	1.00	0.80	-0.19	0.27	0.02	0.06	-0.10	0.03	-0.01	-0.13	-0.22
Ratio 14	0.27	-0.09	0.61	0.27	0.39	-0.06	0.23	0.01	-0.23	0.33	0.30	0.36	0.80	1.00	-0.05	0.34	0.04	0.05	-0.12	0.03	-0.21	0.27	-0.11
Ratio 15	0.27	0.33	0.10	0.06	0.26	0.40	0.57	-0.17	-0.57	-0.02	-0.54	-0.76	-0.19	-0.05	1.00	0.15	-0.14	-0.20	-0.08	0.42	-0.38	0.27	0.99
Ratio 16	0.11	-0.08	0.36	0.08	0.18	-0.06	0.05	0.30	-0.05	-0.10	0.00	0.10	0.27	0.34	0.15	1.00	0.19	0.32	-0.09	0.26	-0.12	0.11	0.09
Ratio 17	-0.06	0.17	0.06	0.10	-0.07	0.08	0.07	0.19	-0.07	-0.07	-0.09	-0.01	0.02	0.04	-0.14	0.19	1.00	0.54	0.54	0.09	-0.11	-0.06	-0.21
Ratio 18	-0.07	-0.08	-0.01	-0.11	-0.02	-0.06	-0.14	0.31	0.14	-0.08	-0.08	-0.06	0.06	0.05	-0.20	0.32	0.54	1.00	0.72	0.13	0.12	-0.07	-0.18
Ratio 19	-0.15	-0.08	-0.19	-0.26	-0.14	-0.08	-0.02	0.38	0.02	-0.14	-0.20	-0.16	-0.10	-0.12	-0.08	-0.09	0.54	0.72	1.00	0.09	-0.14	-0.15	-0.08
Ratio 20	0.22	0.26	0.32	0.39	0.24	0.30	0.18	0.17	-0.18	-0.08	-0.28	-0.40	0.03	0.03	0.42	0.26	0.09	0.13	0.09	1.00	-0.22	0.22	0.42
Ratio 21	-0.33	-0.20	-0.29	-0.10	-0.34	-0.21	-0.80	-0.07	0.80	-0.15	-0.03	0.09	-0.01	-0.21	-0.38	-0.12	-0.11	0.12	-0.14	-0.22	1.00	-0.33	-0.32
Ratio 22	1.00	0.71	0.61	0.43	0.96	0.78	0.46	-0.38	-0.46	0.18	-0.15	-0.10	-0.13	0.27	0.27	0.11	-0.06	-0.07	-0.15	0.22	-0.33	1.00	0.24
Ratio 23	0.24	0.33	0.04	0.03	0.23	0.40	0.53	-0.17	-0.53	0.00	-0.55	-0.81	-0.22	-0.11	0.99	0.09	-0.21	-0.18	-0.08	0.42	-0.32	0.24	1.00

Figure 10.18: Correlation Matrix for 23 financial variables

10.11 Conclusion

One of the challenges faced when using classification algorithms for default prediction, is which features or input variables to select. There are three general approaches to solve the feature selection problem: Filter Methods, Wrapper Methods and Embedded Methods. This study chooses a set of dominant ratios derived from a larger set of related ratios as a prior screening tool or pre processing step. Following that a wrapper method is used to select the most important features that increase variable performance and decrease computational time.

.

Chapter 11

Hyper-parameter Tuning

Machine learning predictive modeling algorithms are governed by hyper-parameters. Not only do ideal settings for the hyper-parameters dictate the performance of the training process, but more importantly they govern the quality of the resulting predictive models. Tuning hyper-parameter values is a critical aspect of the model training process and is considered a best practice for a successful machine learning application.

Illustrations of hyper-parameters are: the depth of a decision tree, number of trees in a forest, number of hidden layers and neurons in each layer in a neural network, and degree of regularization to prevent overfitting are a few examples of quantities that must be prescribed for these algorithms Koch et al. (2017).

11.1 Common Hyper-parameter Tuning Approaches

11.1.1 Grid Search

The Grid Search Method is a Brute Force Method for tuning. It is an approach to parameter tuning that will methodically build and evaluate a model for each combination of algorithm parameters specified in a grid. It allows to test different combinations of hyper-parameters and find one in which the accuracy of the model is improved. In the grid, each algorithm parameter can be specified as a vector of possible values. These vectors combine to define all the possible combinations to try.

11.1.2 Random Search

A simple yet surprisingly effective alternative to performing a grid search is to train and assess candidate models by using random combinations of hyper-parameter values.

11.1.3 Neural Network Tuning Process using Caret

When you train a neural network (using function *nnet*) using *Caret* the two hyper-parameters that need to be tuned are size and decay. Size is the number of units in hidden layer (*nnet* fit a single hidden layer neural network) and decay is the regularization parameter to avoid over-fitting.

11.1.4 Support Vector Machines Tuning using Caret

When you train the support vector machine (RBF-SVM) using Caret then you have two parameters to tune: C and gamma (the radius of RBF).

11.1.5 Hyper-parameter Tuning in R

The *Caret R* package provides both a Random Search Method and a Grid Search Method for searching parameters. In this study a Grid Search Method is used via the *Caret R* package Kuhn (2017). The parameters used to specify the tune grid are entered manually. In the grid, each algorithm parameter can be specified as a vector of possible values. These vectors combine to define all the possible combinations to try. This is done via the *expand.grid* and *tuneGrid* function. The argument *tuneGrid* can take a data frame with columns for each tuning parameter.

Chapter 12

Validation

Validation techniques are motivated by two fundamental problems in pattern recognition model selection and performance estimation. Once The model has been chosen performance is typically measured by one of the Methods to Evaluate Performance as discussed in Section 6.4.

Ideally if there was enough data we would set aside a validation set and use it to assess the performance of the model (Hastie et al., 2009). This is known as cross validation. Cross-Validation is a statistical method of evaluating and comparing learning algorithms by dividing data into two segments: one used to learn or train a model and the other used to validate the model (Refaeilzadeh et al., 2009).

The holdout method is the simplest kind of cross validation. The holdout method, sometimes called test sample estimation, partitions the data into two mutually exclusive subsets called a training set and a test set, or holdout set. It is common to designate 2/3 of the data as the training set and the remaining 1/3 as the test set. The training set is given to the inducer, and the induced classier is tested on the test set. The holdout estimate is a random number that depends on the division into a training set and a test set. In random subsampling the holdout method is repeated k times,and the estimated accuracy is derived by averaging the runs. The standard deviation can be estimated as the standard deviation of the accuracy estimations from each holdout run. If the training and test set are formed by a split of an original dataset, then an over-represented class in one subset will be a underrepresented in the other. Thus the evaluation may depend heavily on which data points end up in the training set and which end up in the test set, and thus the evaluation may be significantly different depending on how the division is made. In practice, the dataset size is always finite, and usually smaller than we would like it to be. The holdout method makes inefficient use of the data: a third of dataset is not used for training

the inducer (Kohavi, 1995).

K-fold cross validation is one way to improve over the holdout method. The data set is divided into k equally (or nearly equally) subsets or folds, and the holdout method is repeated k times. Each time, one of the k subsets or folds is used as the test set and the other $k-1$ subsets are put together to form a training set. The cross-validation estimate of accuracy is the average error across all k trials (Kohavi, 1995). The variance of the resulting estimate is reduced as k is increased. The disadvantage of this method is that the training algorithm has to be rerun from scratch k times, which means it takes k times as much computation to make an evaluation.

Leave-one-out cross-validation (LOOCV) is a special case of cross-validation where the number of folds k equals the number of instances N in the data set. Thus, the learning algorithm is applied N separate times, once for each instance, using all other instances as a training set and using the selected instance as a single-item test set (Efron, 1982). That means that N separate times, the function approximation is trained on all the data except for one point and a prediction is made for that point. Repeated k-fold cross validation does the same as above but more than once. For example, five repeats of 10-fold CV would give 50 total resamples that are averaged.

In this study there were only 67 observations so a cross validation was necessary to help protect against overfitting. Two method were used for cross validation a repeated k-fold cross validation and a LOOCV. These were implemented using the caret package Kuhn (2017).

For the repeated K-Fold cross validation a 30 repeated 5- Fold Cross Validation was preformed. Within each fold the Confusion Matrix, AUC, Precision True Negaitive Rate, False Posoive Rate, False Negative Rate and Gini were calculated as described in .. This was repeated 30 times. Once all those measures were calculated the mean was calculated for each measure and the standard deviation for the accuracy (or error) was calculated based on the 150 simulations. The ROC curve was calculated from the mean Confusion Matrix.

For the LOOCV the data set had 67 observations, hence, there were 67 folds. Each time the algorithm learned from 66 observations (training data) and made predication for the left out observation (testing data). Once there were all the 67 predictions from each fold, it was compared to the actual data. The confusion matrix , ROC Curves and different measures were calculated.

12.1 Feature selection and Cross Validation

Cross-validation (CV) is often being used for validation of machine learning models nevertheless, cases where it is incorrectly applied are not uncommon, especially when the predictive model building includes a feature selection stage.

Hastie et al. (2009) illustrates a recurring mistake when applying CV for model assessment with a feature selection

- Screen the predictors: find a subset of 'good' predictors that show fairly strong (univariate) correlation with the class labels
- Using just this subset of predictors, build a multivariate classifier
- Use cross-validation to estimate the unknown tuning parameters and to estimate the prediction error of the final model

This is incorrect as using Cross Validation to compute an error estimate for a classifier that has itself been tuned using Cross Validation gives a significantly biased estimate of the true error. This means that the decisions made to select the features were made on the entire training set, that in turn are passed onto the model. This may cause a mode a model that is enhanced by the selected features over other models being tested to get seemingly better results, when in fact it is biased result.

Proper use of CV for estimating true error of a classifier developed using a well defined algorithm requires that all steps of the algorithm, including classifier parameter tuning, be repeated in each Cross Validation loop. This means that feature selection is performed on the prepared fold right before the model is trained . A nested CV procedure provides an almost unbiased estimate of the true error. We show that using CV to compute an error estimate for a classifier that has itself been tuned using CV gives a significantly biased estimate of the true error. Proper use of CV for estimating true error of a classifier developed using a well defined algorithm requires that all steps of the algorithm, including classifier parameter tuning, be repeated in each CV loop. A nested CV procedure provides an almost unbiased estimate of the true error (Varma and Simon, 2006).

Figure 12.1 is the incorrect way to do cross validation with feature selection, the feature selection is outside the cross validation loop. Figure 12.2 is the correct way to do the cross validation with feature selection, the feature selection is within the cross validation loop.

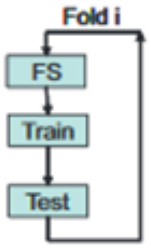


Figure 12.1: Incorrect way to preform Cross Validation with Feature Selection

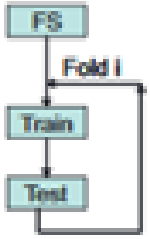


Figure 12.2: Correct way to preform Cross Validation with Feature Selection

In this study for both the LOOCV and the Repeated K-Fold cross validation the feature selection is preformed in the inner loop of the cross validation.

Hastie et al. (2009) show the correct way of preforming a feature selection are as follows:

- Find a subset of good predictors that show fairly strong (univariate) correlation with the class labels, using all of the samples except those in fold k
- Using just this subset of predictors, build a multivariate classifier, using all of the samples except those in fold k
- Use the classifier to predict the class labels for the samples in fold k

12.2 Grid Search Optimisation and Cross Validation

When preforming a grid search tuning process the grid search method is nested within the cross-validation loop to select the optimal parameters for each algorithm. If the tuning parameters are selected prior to the cross validation the error is no longer representative for the model and this results in over fitting. There within each fold of the Cross Validation the Confusion Matrix, AUC, Precision True Negaitive Rate, False Posoive Rate, False Negative Rate and Gini were calculated. Once all those measures were calculated the mean was calculated for each measure as described in Section 12.1.

Chapter 13

Methodology

The research design for this study is as follows: first a set of variables was chosen as a preprocessing step based on the criteria that the ratios have been frequently used in past studies, the ratios have been shown to perform well in past studies and the ratios represent liquidity ratios, profitability ratios, cash flow ratios and solvency ratios. These set of variables is to be used to asses the predictive power of support vector machines, neural networks, logit analysis and LDA analysis. A wrapper feature selection was applied to each algorithm on the set of input variables to improve each algorithm's performance and to remove redundant features. This type of feature selection is dependant on the algorithm it is evaluated on so each different algorithm will have a different set of features. There were only 67 observations so a cross validation was necessary to help protect against overfitting. The wrapper was preformed in the inner loop of the cross validation. Two cross validation methods were used, Leave-Out-One Cross validation and K-Fold Cross Validation and the results were compared using the methods to evaluate performance in Section 6.4. All analysis was preformed on 2y Prior to Default.

13.0.1 Input Variables

This study uses a set of ratios selected to evaluate the financial performance of each company. The selection and the choice of ratios were based on three main conditions as described in Chapter Section . In order to select these variables a matrix of studies was created in Microsoft Excel. 23 financial ratios were chosen as seen in Figure 10.4.

13.0.2 Description of Algorithms

The *caret* package contains functions to streamline the model training process for complex regression and classification problems. The package utilizes a number of R packages in its implementation Kuhn (2010). A list of the functions used in the modelling as well of the algorithms tuning parameters can be seen in Figure 13.1 below:

Model	Method Value	Package	Tuning Parameters
Support Vector Machines with Radial Basis Function Kernel	svmRadial	kernlab	sigma, C
Neural Network	nnet	nnet	size, decay
Linear Discriminant Analysis	lda	MASS	None
Generalized Linear Model	glm		None

Figure 13.1: Functions and Tuning Parameters used in this study

Support Vector Machine

Support Vector Machines with Radial Basis Function Kernel.

Neural Network

Neural Network the tuning parameter 'size' was held constant at a value of 5 (5 hidden layers).

13.0.3 Wrapper method chosen

A Recursive Feature Elimination (RFE) Method is used as the wrapper method as described in Section 13.0.3. It is done through the caret package in R. How it is implemented with Cross Validation in RStudio can be seen in Kuhn (2010).

Recursive Feature Elimination (RFE) includes two nested levels of cross-validation, in this study a 30 repeat 5 fold cross validation is preformed. At the first level, the training data is partitioned in 5 folds and RFE is applied 30 times. In each application, one of the folds is put aside for testing generalization performance while the other folds together form the training data for the RFE procedure. Features were selected based on the Accuracy Metric.

In each fold of the cross validation a recursive feature elimination (RFE) Method was preformed by using the function $rfeControl$. The final features output were calculated by doing a frequency count on achieving the best accuracy.

13.0.4 K-Fold Cross Validation

A 30 repeated 5- Fold Cross Validation was preformed by using the *trainControl* function, by selecting "repeatedcv". Within each fold the Confusion Matrix, AUC, Precision True Negative Rate, False Posoive Rate, False Negative Rate and Gini were calculated. This was repeated 30 times. Once all those measures were calculated the mean was calculated for each measure and the standard deviation for the accuracy (or error) was calculated based on the 150 simulations.

13.0.5 LOOCV

For the LOOCV the data set had 67 observations, hence, there were 67 folds. Each time the algorithm learned from 66 observations (training data) and made predication for the left out observation (testing data). Once there were all the 67 predictions from each fold, it was compared to the actual data. The confusion matrix , ROC Curves and different measures were calculated. The LOOCV was programmed by utilising a for loop for 67 runs.

13.0.6 Grid Search Method

For the Support Vector Machine the grid was created with a starting value of sigma to be 0.1 and C to be 10. For the Neural Network the size was set to a starting value of 20, and decay 0.001.

Chapter 14

Analysis and Results

14.1 Introduction

In this Chapter, the results of the data analysis, as described in the previous chapters, are discussed. The aim of this Chapter is to compare the performance of Support Vector Machines and Neural Networks, to traditional statistical models (Logit Analysis and Multivariate Discriminate Analysis) for Default Prediction when applied to South African Companies.

Recall from Section 1.3 that the historic data is a set of financial ratios constructed from the companys historic financial statements. The data sample consists of 23 financial ratios from 67 companies listed on the JSE. The 67 companies are grouped into "Default" or "Non Default". Due to the lack of data a cross validation was necessary to help protect against overfitting, see Chapter 12. Results were reported with and without a feature selection. All results are based on 2 years prior to default.

14.2 LOOCV

For the LOOCV the data set had 67 observations, hence, there were 67 folds. Each time the algorithm learned from 66 observations (training data) and made predication for the left out observation (testing data). Once there were all the 67 predictions from each fold, it was compared to the actual data. The confusion matrix , ROC Curves and different measures were calculated. The LOOCV was programmed by utilising a for loop for 67 runs.

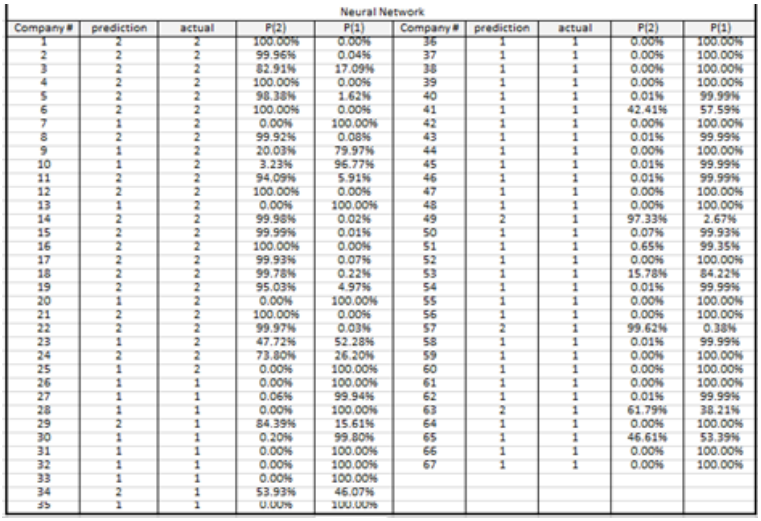
14.2.1 Algorithm Output

Figure 14.1, 14.2, 14.3 and 14.4 are the algorithm output and probabilities of default the set of Support Vector Machines, Neural Networks, Logit Analysis and LDA Analysis respectively.

Support Vector Machine

Company #	prediction	actual	P(2)	P(1)	Company #	prediction	actual	P(2)	P(1)
1	2	2	94.37%	5.63%	36	2	1	80.90%	19.10%
2	2	2	87.42%	12.58%	37	1	1	11.96%	88.04%
3	2	2	84.46%	15.54%	38	1	1	0.23%	99.77%
4	2	2	78.90%	21.10%	39	1	1	0.59%	99.41%
5	2	2	68.34%	31.66%	40	1	1	4.16%	95.84%
6	2	2	99.64%	0.36%	41	1	1	12.80%	87.20%
7	2	2	73.24%	26.76%	42	1	1	6.55%	93.45%
8	2	2	78.99%	21.01%	43	1	1	7.53%	92.47%
9	2	2	86.33%	13.67%	44	1	1	10.75%	89.25%
10	2	2	96.19%	3.81%	45	1	1	5.59%	94.41%
11	1	2	10.52%	89.48%	46	1	1	4.36%	95.64%
12	2	2	79.07%	20.93%	47	2	1	72.24%	27.76%
13	2	2	74.36%	25.64%	48	1	1	0.33%	99.67%
14	2	2	99.93%	0.07%	49	1	1	7.44%	92.56%
15	2	2	87.72%	12.28%	50	1	1	0.90%	99.10%
16	2	2	88.28%	11.72%	51	1	1	5.72%	94.28%
17	2	2	99.54%	0.46%	52	1	1	1.15%	98.85%
18	2	2	57.72%	42.28%	53	1	1	8.33%	91.67%
19	2	2	77.01%	22.99%	54	1	1	2.41%	97.59%
20	2	2	68.90%	31.10%	55	1	1	4.61%	95.39%
21	2	2	84.90%	15.10%	56	1	1	28.66%	71.34%
22	2	2	73.32%	26.68%	57	2	1	84.30%	15.70%
23	2	2	89.52%	10.48%	58	1	1	0.09%	99.91%
24	1	2	55.70%	44.30%	59	1	1	8.01%	91.99%
25	2	2	64.77%	35.23%	60	1	1	2.57%	97.43%
26	1	1	0.24%	99.76%	61	1	1	0.24%	99.76%
27	1	1	11.85%	88.15%	62	1	1	3.29%	96.71%
28	1	1	0.42%	99.58%	63	1	1	5.12%	94.88%
29	2	1	70.27%	29.73%	64	1	1	2.25%	97.75%
30	1	1	11.21%	88.79%	65	2	1	78.16%	21.84%
31	1	1	1.22%	98.78%	66	1	1	23.59%	76.41%
32	1	1	2.63%	97.37%	67	1	1	0.13%	99.87%
33	1	1	0.05%	99.95%					
34	1	1	46.95%	53.05%					
35	1	1	1.48%	98.52%					

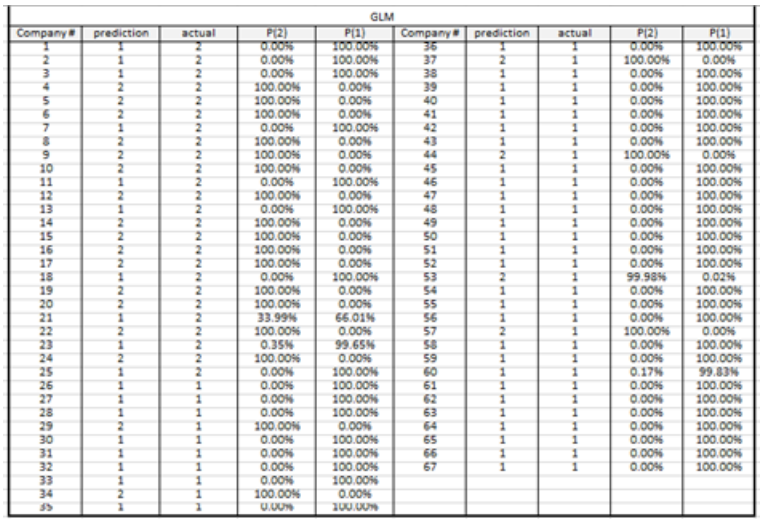
Figure 14.1: Algorithm Output and Probabilities of Default for Support Vector Machines



Neural Network

Company #	prediction	actual	P(2)	P(1)	Company #	prediction	actual	P(2)	P(1)
1	2	2	100.00%	0.00%	36	1	1	0.00%	100.00%
2	2	2	99.96%	0.04%	37	1	1	0.00%	100.00%
3	2	2	82.91%	17.09%	38	1	1	0.00%	100.00%
4	2	2	100.00%	0.00%	39	1	1	0.00%	100.00%
5	2	2	98.38%	1.62%	40	1	1	0.01%	99.99%
6	2	2	100.00%	0.00%	41	1	1	42.41%	57.59%
7	1	2	0.00%	100.00%	42	1	1	0.00%	100.00%
8	2	2	99.92%	0.08%	43	1	1	0.01%	99.99%
9	1	2	20.03%	79.97%	44	1	1	0.00%	100.00%
10	1	2	3.23%	96.77%	45	1	1	0.01%	99.99%
11	2	2	94.09%	5.91%	46	1	1	0.01%	99.99%
12	2	2	100.00%	0.00%	47	1	1	0.00%	100.00%
13	1	2	0.00%	100.00%	48	1	1	0.00%	100.00%
14	2	2	99.98%	0.02%	49	2	1	97.33%	2.67%
15	2	2	99.99%	0.01%	50	1	1	0.07%	99.93%
16	2	2	100.00%	0.00%	51	1	1	0.65%	99.35%
17	2	2	99.93%	0.07%	52	1	1	0.00%	100.00%
18	2	2	99.78%	0.22%	53	1	1	15.78%	84.22%
19	2	2	95.03%	4.97%	54	1	1	0.01%	99.99%
20	1	2	0.00%	100.00%	55	1	1	0.00%	100.00%
21	2	2	100.00%	0.00%	56	1	1	0.00%	100.00%
22	2	2	99.97%	0.03%	57	2	1	99.62%	0.38%
23	1	2	47.72%	52.28%	58	1	1	0.01%	99.99%
24	2	2	73.80%	26.20%	59	1	1	0.00%	100.00%
25	1	2	0.00%	100.00%	60	1	1	0.00%	100.00%
26	1	1	0.00%	100.00%	61	1	1	0.00%	100.00%
27	1	1	0.06%	99.94%	62	1	1	0.01%	99.99%
28	1	1	0.00%	100.00%	63	2	1	61.79%	38.21%
29	2	1	84.39%	15.61%	64	1	1	0.00%	100.00%
30	1	1	0.20%	99.80%	65	1	1	46.61%	53.39%
31	1	1	0.00%	100.00%	66	1	1	0.00%	100.00%
32	1	1	0.00%	100.00%	67	1	1	0.00%	100.00%
33	1	1	0.00%	100.00%					
34	2	1	53.93%	46.07%					
35	1	1	0.00%	100.00%					

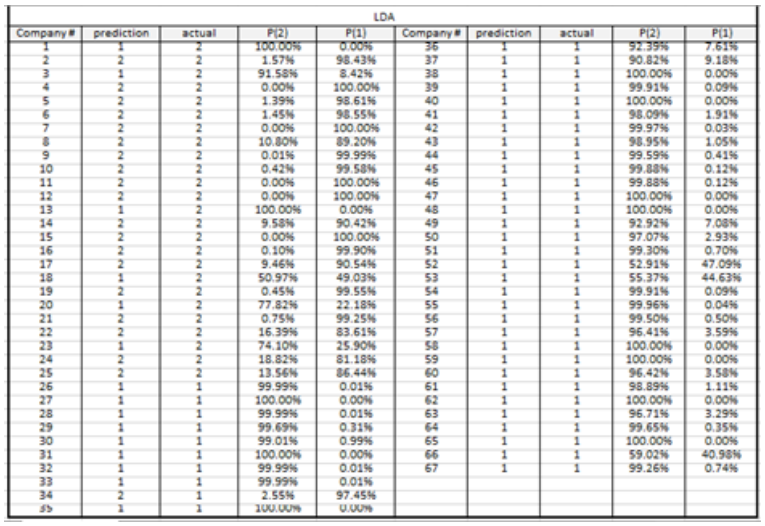
Figure 14.2: Algorithm Output and Probabilities of Default for Neural Networks



GLM

Company #	prediction	actual	P(2)	P(1)	Company #	prediction	actual	P(2)	P(1)
1	1	2	0.00%	100.00%	36	1	1	0.00%	100.00%
2	1	2	0.00%	100.00%	37	2	1	100.00%	0.00%
3	1	2	0.00%	100.00%	38	1	1	0.00%	100.00%
4	2	2	100.00%	0.00%	39	1	1	0.00%	100.00%
5	2	2	100.00%	0.00%	40	1	1	0.00%	100.00%
6	2	2	100.00%	0.00%	41	1	1	0.00%	100.00%
7	1	2	0.00%	100.00%	42	1	1	0.00%	100.00%
8	2	2	100.00%	0.00%	43	1	1	0.00%	100.00%
9	2	2	100.00%	0.00%	44	2	1	100.00%	0.00%
10	2	2	100.00%	0.00%	45	1	1	0.00%	100.00%
11	1	2	0.00%	100.00%	46	1	1	0.00%	100.00%
12	2	2	100.00%	0.00%	47	1	1	0.00%	100.00%
13	1	2	0.00%	100.00%	48	1	1	0.00%	100.00%
14	2	2	100.00%	0.00%	49	1	1	0.00%	100.00%
15	2	2	100.00%	0.00%	50	1	1	0.00%	100.00%
16	2	2	100.00%	0.00%	51	1	1	0.00%	100.00%
17	2	2	100.00%	0.00%	52	1	1	0.00%	100.00%
18	1	2	0.00%	100.00%	53	2	1	99.98%	0.02%
19	2	2	100.00%	0.00%	54	1	1	0.00%	100.00%
20	2	2	100.00%	0.00%	55	1	1	0.00%	100.00%
21	1	2	33.99%	66.01%	56	1	1	0.00%	100.00%
22	2	2	100.00%	0.00%	57	2	1	100.00%	0.00%
23	1	2	0.35%	99.65%	58	1	1	0.00%	100.00%
24	2	2	100.00%	0.00%	59	1	1	0.00%	100.00%
25	1	2	0.00%	100.00%	60	1	1	0.17%	99.83%
26	1	1	0.00%	100.00%	61	1	1	0.00%	100.00%
27	1	1	0.00%	100.00%	62	1	1	0.00%	100.00%
28	1	1	0.00%	100.00%	63	1	1	0.00%	100.00%
29	2	1	100.00%	0.00%	64	1	1	0.00%	100.00%
30	1	1	0.00%	100.00%	65	1	1	0.00%	100.00%
31	1	1	0.00%	100.00%	66	1	1	0.00%	100.00%
32	1	1	0.00%	100.00%	67	1	1	0.00%	100.00%
33	1	1	0.00%	100.00%					
34	2	1	100.00%	0.00%					
35	1	1	0.00%	100.00%					

Figure 14.3: Algorithm Output and Probabilities of Default for Logit Analysis



LDA									
Company #	prediction	actual	P(2)	P(1)	Company #	prediction	actual	P(2)	P(1)
1	1	2	100.00%	0.00%	36	1	1	92.39%	7.61%
2	2	2	1.57%	98.43%	37	1	1	90.82%	9.18%
3	1	2	91.58%	8.42%	38	1	1	100.00%	0.00%
4	2	2	0.00%	100.00%	39	1	1	99.91%	0.09%
5	2	2	1.39%	98.61%	40	1	1	100.00%	0.00%
6	2	2	1.45%	98.55%	41	1	1	98.09%	1.91%
7	2	2	0.00%	100.00%	42	1	1	99.97%	0.03%
8	2	2	10.80%	89.20%	43	1	1	98.95%	1.05%
9	2	2	0.01%	99.99%	44	1	1	99.59%	0.41%
10	2	2	0.42%	99.58%	45	1	1	99.88%	0.12%
11	2	2	0.00%	100.00%	46	1	1	99.88%	0.12%
12	2	2	0.00%	100.00%	47	1	1	100.00%	0.00%
13	1	2	100.00%	0.00%	48	1	1	100.00%	0.00%
14	2	2	9.58%	90.42%	49	1	1	92.92%	7.08%
15	2	2	0.00%	100.00%	50	1	1	97.07%	2.93%
16	2	2	0.10%	99.90%	51	1	1	99.30%	0.70%
17	2	2	9.46%	90.54%	52	1	1	52.91%	47.09%
18	1	2	50.97%	49.03%	53	1	1	55.37%	44.63%
19	2	2	0.45%	99.55%	54	1	1	99.91%	0.09%
20	1	2	77.82%	22.18%	55	1	1	99.96%	0.04%
21	2	2	0.75%	99.25%	56	1	1	99.50%	0.50%
22	2	2	16.39%	83.61%	57	1	1	96.41%	3.59%
23	1	2	74.10%	25.90%	58	1	1	100.00%	0.00%
24	2	2	18.82%	81.18%	59	1	1	100.00%	0.00%
25	2	2	13.56%	86.44%	60	1	1	96.42%	3.58%
26	1	1	99.99%	0.01%	61	1	1	98.89%	1.11%
27	1	1	100.00%	0.00%	62	1	1	100.00%	0.00%
28	1	1	99.99%	0.01%	63	1	1	96.71%	3.29%
29	1	1	99.69%	0.31%	64	1	1	99.65%	0.35%
30	1	1	99.01%	0.99%	65	1	1	100.00%	0.00%
31	1	1	100.00%	0.00%	66	1	1	59.02%	40.98%
32	1	1	99.99%	0.01%	67	1	1	99.26%	0.74%
33	1	1	99.99%	0.01%					
34	2	1	2.55%	97.45%					
35	1	1	100.00%	0.00%					

Figure 14.4: Algorithm Output and Probabilities of Default for LDA Analysis

14.2.2 LOOCV with Feature Selection

The feature selection was preformed at each fold of the LOOCV. Figure 14.5 is the set of confusion matrices evaluated with LOOCV and LOOCV with feature selection.

Figure 14.6 is the set of Classification Ratios evaluated with LOOCV and the LOOCV with feature selection.

Figure 14.7, 14.8, 14.9 and 14.10 are the ROC curves for the ROC curves for SVM, Neural Networks, Logit Analysis and LDA Analysis respectively.

The top reacurring features (Using a count function) were : SVM - Ratio 4, Ratio 3 and Ratio 6. Neural Networks - Ratio 8, Ratio 4 and Ratio 23. LDA - Ratio 4, Ratio 6 and Ratio 2 and Logit - Ratio 14, Ratio 3 and Ratio 8

SVM	Reference	
Prediction	No	Yes
No	37	2
Yes	5	23

NN	Reference	
Prediction	No	Yes
No	37	7
Yes	5	18

LDA	Reference	
Prediction	No	Yes
No	41	6
Yes	1	19

GLM	Reference	
Prediction	No	Yes
No	36	10
Yes	6	15

SVM FS	Reference	
Prediction	No	Yes
No	39	5
Yes	3	20

NN FS	Reference	
Prediction	No	Yes
No	38	8
Yes	4	17

LDA FS	Reference	
Prediction	No	Yes
No	38	8
Yes	4	17

GLM FS	Reference	
Prediction	No	Yes
No	39	4
Yes	3	21

Figure 14.5: Confusion Matrices evaluated with LOOCV and LOOCV Feature Selection

Algorithm	Accuracy	95% CI Accuracy	True Positive Rate	False Negative Rate	True Negative Rate	False Positive Rate	Precision	AUC	Gini	Kappa
SVM	0.896	(0.7965, 0.957)	0.881	0.119	0.949	0.051	0.821	0.900	0.801	0.782
SVM FS	0.881	(0.7782, 0.947)	0.929	0.071	0.886	0.114	0.870	0.864	0.729	0.740
NN	0.821	(0.708, 0.9039)	0.881	0.119	0.841	0.159	0.783	0.800	0.601	0.611
NN FS	0.821	(0.708, 0.9039)	0.905	0.095	0.826	0.174	0.810	0.792	0.585	0.604
LDA	0.896	(0.7965, 0.957)	0.976	0.024	0.872	0.128	0.950	0.868	0.736	0.767
LDA FS	0.821	(0.708, 0.9039)	0.905	0.095	0.826	0.174	0.810	0.792	0.585	0.604
GLM	0.761	(0.6414, 0.8565	0.857	0.143	0.783	0.217	0.714	0.729	0.457	0.472
GLM FS	0.896	(0.7965, 0.957)	0.929	0.071	0.907	0.093	0.875	0.884	0.769	0.770

Figure 14.6: Classification Ratios evaluated with LOOCV and LOOCV Feature Selection

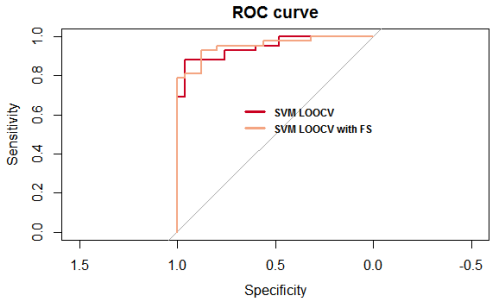


Figure 14.7: ROC Curves for SVM evaluated with LOOCV and LOOCV Feature Selection

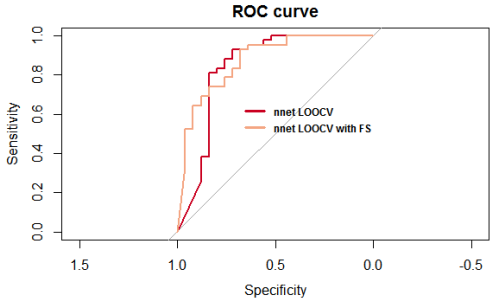


Figure 14.8: ROC Curves for Neural Networks evaluated with LOOCV and LOOCV Feature Selection

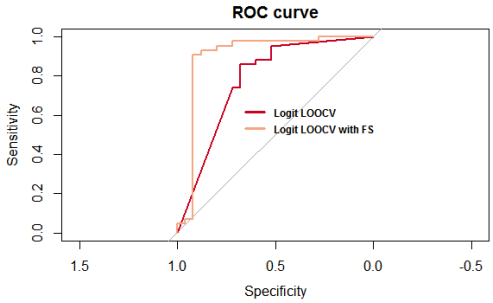


Figure 14.9: ROC Curves for Logit Analysis evaluated with LOOCV and LOOCV Feature Selection

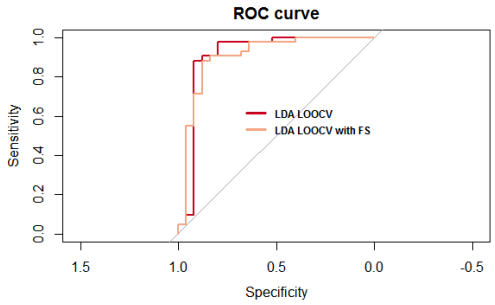


Figure 14.10: ROC Curves for LDA Analysis evaluated with LOOCV and LOOCV Feature Selection

14.3 K - Fold Cross Validation

Each Algorithm was run with Resampling: Cross-Validated (5 fold, repeated 30 times). There were 67 samples, 23 predictors and 2 classes: 'No', 'Yes'.

14.3.1 Confusion Matrix

Figure 14.11 is the confusion matrices for each algorithm. The top section is each algorithm‘s confusion matrix without the feature selection and the bottom is each the algorithm‘s confusion matrix with feature selection. The Confusion Matrices were calculated by averaging the Cross-Validated results (5 fold, repeated 30 times), each entry is the percentual average cell counts across resamples. The R function confusionMatrix uses the predicted classes and thus a 50% probability cutoff

SVM	Reference	
Prediction	No	Yes
No	62.7	35.8
Yes	0	1.5

NN	Reference	
Prediction	No	Yes
No	55.7	12.4
Yes	7	24.9

LDA	Reference	
Prediction	No	Yes
No	57.8	10.4
Yes	4.9	26.9

GLM	Reference	
Prediction	No	Yes
No	55.2	11.9
Yes	7.5	25.4

SVM FS	Reference	
Prediction	No	Yes
No	54.2	15.2
Yes	8.5	22.1

NN FS	Reference	
Prediction	No	Yes
No	56.4	11
Yes	6.3	26.3

LDA FS	Reference	
Prediction	No	Yes
No	57.5	9.5
Yes	5.2	27.9

GLM FS	Reference	
Prediction	No	Yes
No	54.3	11.6
Yes	8.4	25.7

Figure 14.11: Confusion Matrix

14.3.2 Classification Ratios

Figure 14.12 are the classification ratios for each algorithm. All results except the Accuracy Standard deviation and the Kappa Standard deviation were calculated by averaging the Cross-Validated results (5 fold, repeated 30 times).

Algorithm	Mean Accuracy	SD Accuracy	Accuracy 95% CI	Mean True Positive Rate	Mean False Negative	Mean True Negative Rate	Mean False Positive Rate	Mean Precision	Mean AUC	Mean Gini	Mean Kappa	SD Kappa
SVM	0.641	0.032	(0.649,0.634)	1.000	0.000	0.040	0.960	0.636	0.540	0.080	0.035	0.085
SVM FS	0.763	0.121	(0.792,0.734)	0.864	0.136	0.593	0.407	0.781	0.900	0.801	0.473	0.260
NN	0.807	0.105	(0.832,0.782)	0.889	0.111	0.668	0.332	0.818	0.857	0.714	0.616	0.188
NN FS	0.827	0.100	(0.851,0.803)	0.899	0.101	0.705	0.295	0.837	0.928	0.855	0.617	0.226
LDA	0.847	0.092	(0.869,0.825)	0.922	0.078	0.720	0.280	0.847	0.937	0.874	0.673	0.194
LDA FS	0.853	0.092	(0.875,0.831)	0.917	0.083	0.747	0.253	0.859	0.958	0.916	0.676	0.204
GLM	0.806	0.094	(0.829,0.783	0.881	0.119	0.680	0.320	0.822	0.918	0.836	0.567	0.228
GLM FS	0.810	0.103	(0.825,0.776)	0.867	0.133	0.688	0.312	0.824	0.876	0.752	0.563	0.230

Figure 14.12: Classification Ratios per Algorithm

14.3.3 Feature Selection

Figure 14.13 are the ratios from the feature selection for each algorithm. The final variables for the feature selection were based on a frequency count.

SVM

Variables	Accuracy	Kappa	Accuracy SD	Kappa SD
4	0.7628	0.47342	0.12082	0.25968
8	0.7558	0.45945	0.1019	0.23015
16	0.7225	0.32342	0.09809	0.27092
23	0.6395	0.04105	0.02923	0.08996

The top 4 variables (out of 4):
Ratio.4, Ratio.6, Ratio.2, Ratio.3

LDA

Variables	Accuracy	Kappa	Accuracy SD	Kappa SD
4	0.7573	0.4155	0.09836	0.2522
8	0.7393	0.3844	0.09153	0.237
16	0.8232	0.6099	0.10364	0.227
23	0.8533	0.6763	0.09224	0.2044

The top 5 variables (out of 23):
Ratio.4, Ratio.6, Ratio.2, Ratio.3, Ratio.10

Neural Network

Variables	Accuracy	Kappa	Accuracy SD	Kappa SD
4	0.7678	0.502	0.09397	0.2014
8	0.8036	0.572	0.09032	0.1997
16	0.827	0.6172	0.10009	0.2259
23	0.8265	0.6148	0.09218	0.2095

The top 5 variables (out of 16):
Ratio.8, Ratio.4, Ratio.10, Ratio.21, Ratio.23

GLM

Accuracy	Kappa	Accuracy SD	Kappa SD
0.8003	0.5625	0.1032	0.2296

During resampling, the top 5 selected variables (out of a possible 16):
Ratio.14 (100%), Ratio.3 (100%), Ratio.4 (100%), Ratio.8 (100%), Ratio.13 (99.3%)

Figure 14.13: Feature Selection

14.3.4 ROC Curves

Figure 14.14, 14.15, 14.16 and 14.17 are the ROC Curves for each algorithm.

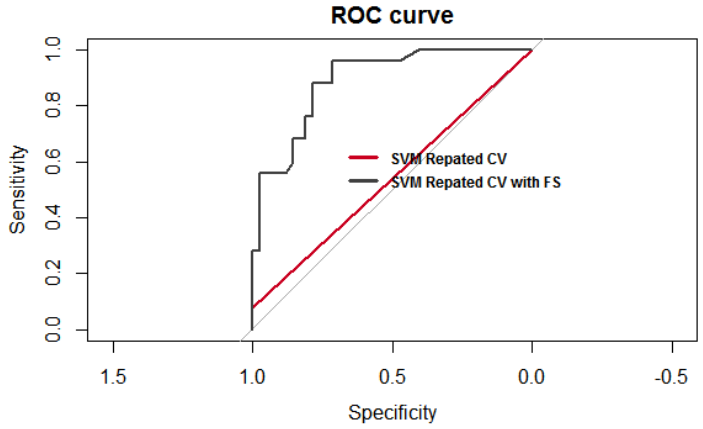


Figure 14.14: SVM ROC Curve

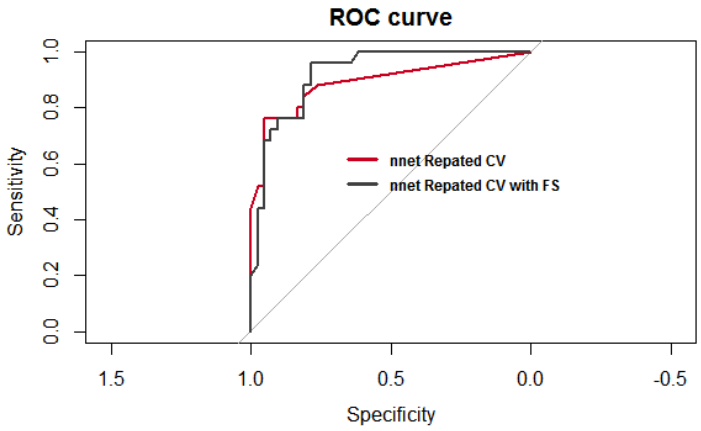


Figure 14.15: Neural Network ROC Curve

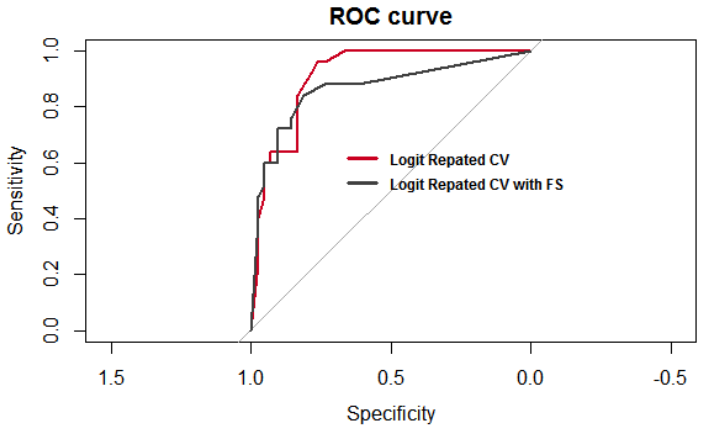


Figure 14.16: Logit ROC Curve

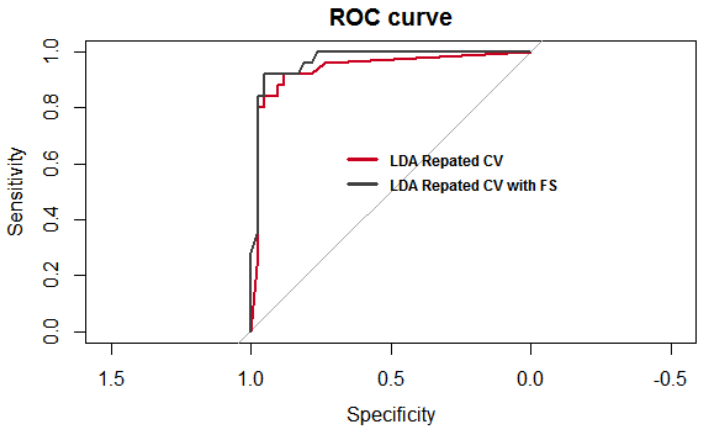


Figure 14.17: LDA ROC Curve

14.4 ROC Curve Comparison

Figure 14.18, 14.19, 14.20 and 14.21 are the comparison of the ROC curves run with a Cross Validation, LOOCV, Cross Validation with feature selection and LOOCV with feature selection

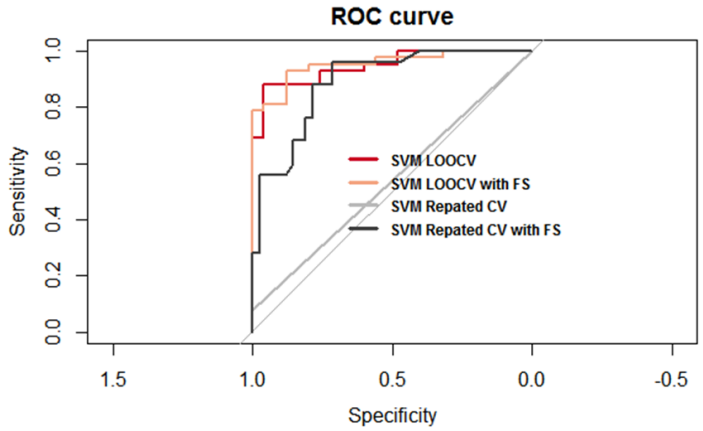


Figure 14.18: Comparison of ROC Curves for SVM

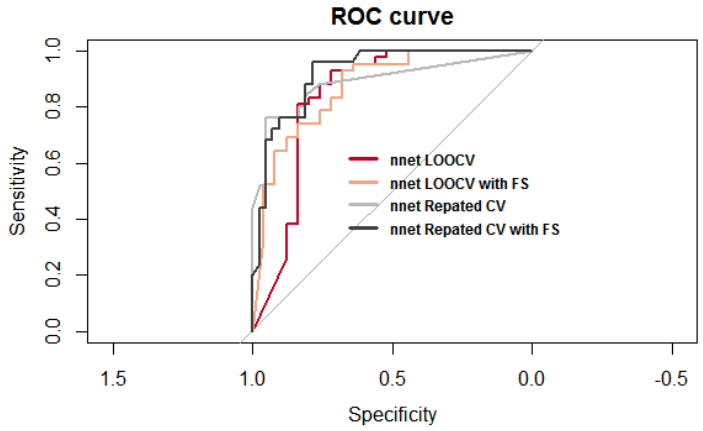


Figure 14.19: Comparison of ROC Curves for Neural Network

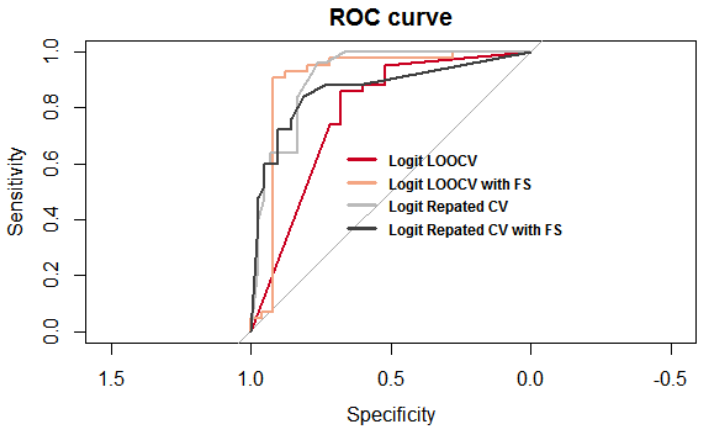


Figure 14.20: Comparison of ROC Curves for Logit Analysis

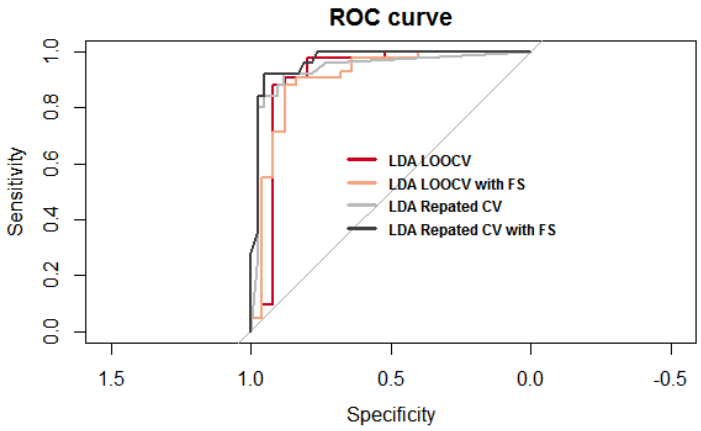


Figure 14.21: Comparison of ROC Curves for LDA Analysis

Chapter 15

Conclusion

This chapter explores the three hypotheses posed in Chapter 1 Section 1.1 with the results obtained in Chapter 14

$H1 : Support\ Vector\ Machines\ and\ Neural\ Networks\ predict\ default\ with\ a\ higher\ accuracy\ than\ LDA\ and\ LogitAnalysis$

$H2 : The\ data\ available\ is\ sufficient\ to\ predict\ default\ 2\ years\ prior\ to\ default$

$H3 :\ Feature\ selection\ increases\ the\ algorithm's\ accuracy$

Referring to Figure 14.6 when using a LOOCV SVM, Neural Networks, Logit Analysis and LDA analysis all predicted default within a certain accuracy. Thus we can *accept* the null hypothesis $H2$ that the data available is sufficient to predict default 2 years prior. The worst preforming algorithm was logit analysis and the best was SVM and LDA Analysis. It is with this result that we *reject* the hypothesis $H1$. LDA outperformed Neural Networks and offered the same performance as Support Vector Machines. When analysing the results after a feature selection was applied we saw that the only algorithm that accuracy increased from the feature selection was logit analysis. Thus we *reject* the hypothesis $H3$.

Referring to Figure 14.12 when using a 30 repeated 5 Fold-Cross-Validation SVM, Neural Networks, Logit Analysis and LDA analysis all predicted default within a certain accuracy. Thus we can *accept* the null hypothesis $H2$ that the data available is sufficient to predict default 2 years prior. The worst preforming algorithm was SVM. It is with this result that we *reject* the hypothesis $H1$. LDA outperformed both Neural Networks and SVM. When analysing the results after a feature selection was applied we saw that Feature Selection increased the performance of all the algorithms

thus *accept* the hypothesis $H3$.

In conclusion the data available to do the analysis is sufficient to predict default 2 years prior to default if a cross validation is preformed. When a LOOCV is used with a feature selection the feature selection did not improve accuracy for all the models. When a feature selection is preformed with a repeated K Fold Cross Validation the accuracy of the algorithms increases with the feature selection. Using both validation methods results are that LDA offered superior performance to Support Vector Machines and Neural Networks.

Bibliography

S. S. A Loigstaff. A simple approach to valuing risky fixed and floating rate debt. *Journal of Finance*, 50:789–819, 1995.

S. Abe. *Support Vector Machines for Pattern Classification*. Springer, 2005.

J. Abhishek and P. Jayesh. Predicting corporate bankruptcyusing financial ratios:an empirical analysis: Indian evidence from 2007-2010. *GFJM*, 4:1–14, June 2012.

M. Adya and F. Collopy. How effective are neural networks at forecasting and prediction? a review and evaluation. *Journal of Forecasting*, 17:481–495, 1998.

S. Aktan. Early warning system for bankruptcy: Bankruptcy prediction. Master's thesis, des Karlsruher Instituts fr Technologie (KIT), 2011a.

S. Aktan. Application of machine learning algorithms for business failure prediction. *Investment Management and Financial Innovations*, 8:52–65, 2011b.

M. Al-Osaimy. A neural networks system for predicting islamic banks performance. *JKAU: Econ. and Adm*, 11:33–46, 1998.

S. Aliakbari. Prediction of corporate bankruptcy for the uk firms in manufacturing industry. Master's thesis, Brunel University, 2009.

E. Alpaydin. *Introduction to Machine Learning*. MIT Press, first edition, 2004.

E. Altman. Financial ratios, discriminant analysis and the prediction of corporate bankrupcy. *International Conference on Business and Economics Research*, 4:589–609, September 1968a.

E. Altman. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *Journal of Finance*, 23:589–609, 1968b.

E. Altman. An emerging market credit scoring system for corporate bonds. *Emerging Markets Review*, 6:311–323, 2005.

E. Altman, G. Marco, and F. Varetto. Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the italian experience). *Journal of Banking and Finance*, 18:505–529, 1994.

E. Altman, M. Iwanicz-Drozdowska, E. Laitinen, and A. Suvas. Distressed firm and bankruptcy prediction in an international context: areview and empirical analysis of altmans z-score model. pages 1–48, 2014.

E. I. Altman and A. Saunders. Credit risk measurement: Developments over the last 20 years. *Journal of Banking and Finance*, 21:1721–1742, 1998.

J. Arron and M. Sandler. The use of neural networks in predicting company failure. *De Ratione*, 8:57–58, 1995.

A. Atiya. Bankruptcy prediction for credit risk using neural networks: A survey and new results. *IEEE Transactions on Neural Networks*, 12:929–935, 2001.

L. Auria and A. Moro. Support vector machines (svm) as a technique for solvency analysis. Discussion Paper 2008, DIW Berlin German Institute for Economic Research, 2008.

Basel. Credit risk modelling: Current practices and applications. *Basel Committee on Banking Supervision*, 1999.

Basel. The internal ratings approach. Technical report, Bank for International Settlements, 05 2001.

Basel. An explanatory note on the basel ii irb risk weight functions. Bank for International Settlements 18, Bank for International Settlements, 2005.

W. Beaver. Financial ratios as predictors of failure. *Journal of Accounting Research*, 4:71–111, 1966.

J. Bell. *Machine Learning: Hands-On for Developers and Technical Professionals*. Wiley, first edition, 2014.

J. Bellovary, D. Giacomino, and M. Akers. A review of bankruptcy prediction studies 1930-present. *Journal of Financial Education*, 33:1–32, 2007.

S. Berberian. *Introduction to Hilbert Space*. AMS Chelsea Publishing, 2000.

R. Berwick. An idiots guide to support vector machines (svms). UCF Course notes, 2009. URL `http://www.cs.ucf.edu/courses/cap6412/fall2009/papers/Berwick2003.pdf`.

S. T. Bharath and T. Shumway. Forecasting default with the kmv-merton model. Technical report, University of Michigan Business School, 2004.

F. Black and M. Scholes. The pricing of options and corporate liabilities. *The Journal of Political Economy*, 81:637–654, 1973.

C. Boltman. Logistic regression and it application to credit scoring. Master's thesis, University of Pretoria, 2009.

C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

E. Charalambakis and I. Garrett. On the prediction of financial distress in developed and emerging markets: Does the choice of accounting and market information matter? a comparison of uk and indian firms. *Review of Quantitative Finance and Accounting*, pages 1–28, 2015.

B. Chen and M. Chen. Applying particles swarm optimization for support vector machines on predicting company financial crisis. *International Conference on Business and Economics Research*, 1:301–305, 2011.

D. Chen and J. Odobez. Comparison of support vector machine and neural network for text texture clarification. 3:1151–1156, December 2013.

M. Chen. Bankruptcy prediction in firms with statistical and intelligent techniques and a comparison of evolutionary computation approaches. *Computers and Mathematics with Applications*, 62:4514–4524, 2011.

X. Chen and J. Jeong. Enhanced recursive feature elimination. *Machine Learning and Applications, 2007. ICMLA 2007. Sixth International Conference*, 4, 2007.

P. Cort and S. Radloff. A two stage model for the prediction of corporate research in south. *Investment Analysts Journal*, 38:9–19, 1993.

C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

A. Damodaran. *The Dark Side of Valuation: Valuing Young, Distressed, and Complex Businesses*. FT Press, second edition, 2009.

K. Daya. Financial ratios as predictors of corporate failure in south africa. Master's thesis, Rhodes University Unpublished MBA Research, 1977.

E. Deakin. A discriminant analysis of predictors of business failure. *Journal of Accounting Research*, 10:167–179, 1972.

H. J. Dikkers. Support vector machines in ordinal classification. Master's thesis, Delft University of Technology, 2005.

du Jardin P. The influence of variable selection methods on the accuracy of bankruptcyprediction models. *Bankers, Markets and Investors*, 116:20–39, 2012.

R. Edmister. An empirical test of financial ratio analysis for small business failure prediction. *The Journal of Financial and Quantitative Analysis*, 7:1477–1493, 1972.

B. Efron. The jackknife, the bootstrap and other resampling plans. *CBMS-NSF regional conference series in applied mathematics, Society for Industrial and Applied Mathematics (SIAM)*, 1982.

A. I. Einarsson. *Credit Risk Modeling*. PhD thesis, Technical University of Denmark, 2008.

A. Elizalde. Credit risk models i: Default correlation in intensity models. Master's thesis, Kings College London, 2003.

A. Elizalde. Credit risk models i:default correlation in intensity models. Technical report, Kings College London, 2006.

B. Engelmann and R. Rauhmeier. *The Basel II Risk Parameters*. Springer, second edition, 2004.

B. Engelmann and R. Rauhmeier. *The Basel II Risk Parameters: Estimation, Validation, Stress Testing to Loan Management*. Springer, second edition, 2008.

J. C. F Black. Valuing corporate securities: Some effects of bond indenture provisions. *Journal of Finance*, 31:351–367, 1976.

L. V. Fausett. *Fundamentals of Neural Networks: Architectures, Algorithms And Applications Paperback*. Pearson, first edition, 1993.

T. Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27:861–874, 2006.

A. Fielding. *Machine Learning Methods for Ecological Applications*. Springer, first edition, 2014.

R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

S. Focardi and F. Fabozzi. *The Mathematics of Financial Modeling and Investment Management*. Wiley Finance, 2004.

J. Frade. Credit risk modeling: Default probabilities. Master's thesis, Florida State University, 2008.

H. Frohlich and O. Chapelle. Feature selection for support vector machines by means of genetic algorithms. in proceedings of the 15th ieee international conference on tools with artificial intelligence. pages 142–148, 2003.

A. Gaughan. *Mergers, Acquisitions, and Corporate Restructurings*. Wiley, first edition, 2015.

V. Georgakopoulos. Current approaches to credit risk measurement. Master's thesis, National and Kapodistrian University of Athens, 2004.

R. Geske. The valuation of corporate liabilities as compound options. *Journal of Financial and Quantitative Analysis*, 12:541–552, 1977.

M. Gonen. Receiver operating characteristic (roc) curves. *Statistics and Data Analysis*, 24(1):210–310, 2000.

P. Gurny and M. Gurny. Comparison of credit scoring models on probability of default estimation for us banks. *IEEE 10th International Conference on Data Mining (ICDM)*, pages 163 – 181, 2013.

I. Guyon. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

M. C. H Liu, A Gegov. *Rule Based Systems for Big Data: A Machine Learning Approach*. Springer, first edition, 2016.

D. Hand. Measuring classifier performance: a coherent alternative to the area under the roc curve. *Mach Learn (2009)*, 77:103–223, 2006.

W. Hardle. Bankruptcy prediction with support vector machines: An application for german companies. Master's thesis, Universitat zu Berlin, 2010.

W. Hardle, R. Moro, and D. Schafer. Estimating probabilities of default with support vector machines. Discussion Paper Series 2: Banking and Financial Studies 18, Deutsche Forschungsgemeinschaft SFB 649 Economic Risk, 2007.

W. Hardle, R. Moro, L. Hoffmann, and S. Aliakbari. Forecasting corporate distress in the asian and pacific region. Discussion Paper Series 2: Banking and Financial Studies 18, Deutsche Forschungsgemeinschaft SFB 649 Economic Risk, 2011.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer US, first edition, 2009.

U. N. HDI. Human development index (hdi), June 2009. URL http://hdr.undp.org/en/composite/HDI/.

D. Hebb. *The organisation of Behaviour*. Wiley, first edition, 1949.

Himmelblau. *Applied Non Linear Programming*. Mac Graw Hill, seventh edition, 1972.

B. Hlahla. Assessing corporate financial distress in south africa. Master's thesis, University of Witwatersrand, 2010a.

B. Hlahla. Assessing corporate financial distress in south africa. Master's thesis, University of Witwatersrand, 2010b.

N. D. Hoa. The lagrange multiplier functions in the equation approach to constrained optimization. *Universitatis Iagellonicae Acta Mathematica Issue 24*, pages 99–117, 1984.

Hoffman and Bradley. *Calculus for Bussiness, Economics and the Social and Life Sciences*. Mac Graw Hill, seventh edition, 1999.

T. Hofmann, B. Scholkopf, and A. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36:1171–1220, 2008.

F. Hsieh and B. Turnbull. Nonparametric and semiparametric estimation of the receiver operating characteristic curve. *The Annals of Statistics*, 24(1):25–40, 1996.

C. Huang, M. Chen, and C. Wang. Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*, 33:847–856, 2007.

Z. Huang, H. Chen, C. Hsu, W. Chen, and S. Wu. Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems 37*, pages 543–558, 2004.

J. Hull and A. White. Valuing credit default swaps i no counterparty default risk. *The Journal of Derivatives*, 8:29–40, 2000.

J. Hull, I. Nelken, and A. White. Mertons model, credit risk, and volatility skews. Master's thesis, University of Toronto, 2003.

N. Idris. Financial ratios as the predictor of corporate distress in malaysia. Master's thesis, Wichita State University, 1998.

J. Jacobs. The application of failure prediction models on non-listed companies. Master's thesis, Tshwane University Of Technology, 2007.

L. Jain and C. de Silva. *Intelligent Adaptive Control: Industrial Applications*. CRC Press, 1999.

M. JeanBlanc and Y. Lecam. xxx website, 2008.

E. Jones. An Introduction to Neural Networks A White Paper. Technical report, Visual Numerics, 2004.

L. Jooste. An evaluation of the usefulness of cash flow ratios to predict financial distress. *Acta Commercii*, 7:1–13, 2007.

J. G. K Poston, W Harmon. A test of financial ratios as predictors of turnaround verses failure among financially distressed firms. *Journal of Applied Business Research*, 10:41–56, 1994.

A. Karatzoglou. Package kernlab. Technical report, CRAN, 03 2016.

A. Karatzoglou and D. Meyer. Support vector machines in r. *Journal of Statistical Software*, 15, 2006.

H. Kidane. Predicting financial distress in it and services companies in south africa. Master's thesis, University of Free State, 2004.

J. Kim, H. Weistroffer, and R. Redmond. Expert systems for bond rating: a comparative analysis of statistical, rule-based and neural network systems. *Expert Systems*, 10:167–172, 1993.

P. Koch, B. Wujek, O. Golovidov, and S. Gardner. Automated hyperparameter tuning for effective machine learning. *SAS Institute Inc*, 2017.

Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *International Joint Conference on Arti
cial Intelligence (IJCAI)*, 1995.

Kohavi and John. Wrappers for feature subset selection. *Artificial Intelligence*, 97: 273–324, 1997.

K. Koiov. Discriminant analysis as a tool for forecasting companys financial health. *Procedia - Social and Behavioral Science*, 110:1148–1157, 2014.

S. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, pages 249–268, 2007.

M. Kuhn. Variable selection using the caret package. Technical report, CRAN, 06 2010.

M. Kuhn. Package 'caret'. Technical report, CRAN, 09 2017.

R. Kumar and A. Indrayan. Receiver operating characteristic (roc) curve for medical researchers. *Indian Pediatr*, 48:277–287, 2011.

B. Kumari and T. Swarnkar. Filter versus wrapper feature subset selection in large dimensionality micro array: A review. *International Journal of Computer Science and Information Technologies*, 2:1048–1053, 2011.

L. Ladha and T.Deepa. Feature selection methods and algorithms. *International Journal on Computer Science and Engineering (IJCSE)*, 3:1787–1797, 2011.

H. Lehland. Presendential address: agency costs, risk management, and capital structure. *Journal of Finance*, 53:12131243, 1998.

C. T. Leondes. *Intelligent Systems: Technology and Applications, Volume 1: Implemantation Tequniques*. CRC Press, first edition, 2003.

J. Li and S. Ma. *Survival Analysis in Medicine and Genetics*. CRC Press, first edition, 2013.

J. Li, J. hang Cheng, J. yuan Shi, and F. Huang. *Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement*. Springer, second edition, 2012.

S. Lin. A two-stage logistic regression-ann model for the prediction of distress banks: Evidence from 11emerging countries. *African Journal of Business Management*, 4: 3149–3168, October 2010.

W. Lin. A case study on support vector machines versus artificial neural networks. Master's thesis, University of Pittsburgh, 1994.

R. P. Lippmann. Neural network classifiers for speech recognition. *The Lincoln Laboratory Journal*, 1:108–124, 1988.

A. Maeteletsa and J. Kruger. A two stage model for the prediction of corporate research in south. *Investment Analysts Journal*, 38:9–19, 1993.

S. Mahdi. Bankruptcy prediction by using support vector machines and genetic algorithms. *Studies in Business and Economics*, 8(1):104–114, April 2013.

A. Mamo. Applicability of altman (1968) model in predicting financial distress of commercial banks in kenya. Master's thesis, University of Nairobi, 2011.

N. Marono and A. Betanzos. Filter methods for feature selection: a comparative study. *Proceedings of the 8th international conference on Intelligent data engineering and automated learning*, pages 178–187, 2007.

W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

R. Merton. On the pricing of corporate debt: The risk structure of interest rates. *The Journal of Finance*, 29:449–470, 1974.

T. C. Miha Vuk. Roc curve, lift chart and calibration plot. *Metodoloski zvezki*, 3: 89–108, 2006.

J. Min and Y. Lee. Business failure prediction with support vector machines and neural networks: A comparative study. *Expert Systems with Applications*, 28:603–614, 2005.

J. Minussi, D. Soopramanien, and D. Worthington. Modelling and testing the assessment of risk of defaulting of companies in the context of the basel accord: A brazilian case study. *PANORAMA SOCIOECONMICO AO 24*, 33:76–85, 2006.

J. Minussi, D. Soopramanien, and D. Worthington. Statistical modelling to predict corporate default for brazilian companies in the context of basel ii using a new set of financial ratios. Technical report, Lancaster University Management School, 2007.

M. Mohamad, S. Deris, S. Yatim, and M. Othman. Feature selection method using genetic algorithm for the classifcation of small and high dimension data. *First International Symposium on Information and Communications Technologies*, 3:1–4, 2004.

V. Moonasar. Credit risk analysis using artificial intelligence: Evidence from a leading south african banking institution. Master's thesis, University of South Africa, 2007.

T. K. Mria Miankov, Katarna Koiov. Comparison of mertons model, black and cox model and kmv model. *8th International Scientific Conference*, pages 280–289, 1977.

G. Muller. Development of a model to predict financial distress of companies listed on the jse. Master's thesis, University of Stellenbosch, 2008.

N. Nachar. The mannwhitney u a test for assessing whether two independent samples come from the same distribution. *Tutorials in Quantitative Methods for Psychology*, 4:13–21, 2008.

S. Naidoo. *Cost of financial distress model for JSE Listed Companies*. PhD thesis, University of South Africa, 2006a.

S. Naidoo. *Cost of financial distress model for JSE Listed Companies*. PhD thesis, University of South Africa, 2006b.

J. Ohlson. Financial ratios and the probabilistic prediction of bankruptcy. *The Journal of Accounting Research*, 18:109–131, 1980.

D. Olson and D. Dursun. *Advanced Data Mining Techniques*. Springer, 2008.

B. C. on Banking Supervision. The standardised approach for measuring counterparty credit risk exposures. Technical report, Bank for International Settlements, March 2014.

B. C. on Banking Supervision. Regulatory treatment of accounting provisions interim approach and transitional arrangements. Technical report, Bank for International Settlements, October 2016a.

B. C. on Banking Supervision. Guidance on credit risk and accounting for expected credit losses. Technical report, Bank for International Settlements, December 2016b.

W. Pam. Discriminant analysis and the prediction of corporate bankruptcy in the banking sector of nigeria. *International Journal of Finance and Accounting*, 2: 319–325, 2013.

M. G. Petr Gurn. Comparison of credit scoring models on probability of default estimation for us banks. *Prague Economic Papers*, 2:163–181, 2013.

E. Pour. Why do companies delist voluntarily from the stock market? *Journal of Banking and Finance*, pages 4850–4860, 2013.

K. Priddy and P. Keller. *Artificial Neural Networks: An Introduction*. Spie Press, 2005.

S. Purohit, V. Mahadevan, and A. Kulkarni. Credit evaluation model of loan proposals for indian banks. *International Journal of Modeling and Optimization*, 2:529–534, August 2012.

P. N. Q Minh and Y. Yao. Mercers theorem, feature maps, and smoothing. *COLT'06 Proceedings of the 19th annual conference on Learning Theory*, pages 154–168, 2006.

J. K. R Poli and T. Blackwell. Particle swarm optimization an overview. *Swarm Intell*, 2007.

K. Rajan. *Regression Methods for Classification Accuracy in Diagnostic Studies with Ordinal Scale Outcomes*. PhD thesis, University of Washington, 2008.

D. Rama. An empirical evaluation of the altman (1968) failure prediction model on south african jse listed companies. Master's thesis, WITS: School of Accounting, 2012.

R. S. Rao. A review on detection and elimination of corporate financial distress. *International Journal of Trend in Research and Development*, 4:550–555, 2017.

S. Razvan-Alexandru. Estimating probabilities of default using support vector machines. Master's thesis, Humboldt University, 2009.

J. Reeves. A study on failure prediction models as enhancements to the credit evaluation procedure in south african corporate bank. Master's thesis, University of Natal, 2001.

Refaeilzadeh, Tang, and Liu. *Cross-Validation*. Springer US, first edition, 2009.

S. Rendel. Factorization machines. *IEEE 10th International Conference on Data Mining (ICDM)*, pages 995 – 1000, 2010.

B. Ripley. Package 'nnet'. Technical report, CRAN, 05 2016.

R. Rojas. *Neural Networks A Systematic Introduction*. Springer, first edition, 1996.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

C. Sammut. *Latent Factor Models and Matrix Factorizations*. Springer, first edition, 2010.

A. Schbel. *Locating Lines and Hyperplanes Theory and Algorithms*. Springer, first edition, 1999.

Senoto. Macroeconomic variables underlying synchronisation in probabilities of default in south african context senkoto masters in financial economics. Master's thesis, University of Johannesburg, 2012.

S. Sharma and M. Shebalkov. Application of neural network and simulation modeling to evaluate russian banks performance. *Journal of Applied Finance and Banking*, 3:19–37, 2013.

W. Sirirattanaphonkun and S. Pattarathammas. Default prediction for small-medium enterprises in emerging market evidence from thailand. *Seoul Journal of Business*, 18:26–53, 2012.

T. S. Sreedhar T Bharath. Forecasting default with the kmv-merton model. Master's thesis, University of Michigan, 2004.

R. Stanley. An introduction to hyperplane arrangements. *Geometric Combinatorics IAS Park City Mathematics Series*, 13:389–496, 2007.

P. Strebel and J. Andrews. A simple predictor of company failure, fact and opinion. Master's thesis, University of the Witwaterand, 1977.

R. N. Strickland. *Image-Processing Techniques for Tumor Detection*. Marcel Dekker, Inc, first edition, 2002.

S. Sundaresan. A review of mertons model of the firms capital structure with its wide applications. *The Annual Review of Financial Economics*, 5:1–21, 2013.

D. Svozil, V. Kvasnicka, and J. Pospichai. Neural network classifiers for speech recognition. *Chemometrics and Intelligent Laboratory Systems*, 39:43–62, 1997.

M. J. T Bielecki, S Crepey. Defaultable game options in a hazard process model. *Journal of Applied Mathematics and Stochastic Analysis*, 2009:1–33, 2009.

C. Triandafil and P. Brezeanu. Corporate financial analysis and localization criteria - emerging versus developed countries: Case study on it commercial companies. *Annals of the University of Petroani, Economics*, 10:341–348, 2010.

F. Tshitanga. Cost of financial distress model for jse listed companies. Master's thesis, University of Pretoria, 2010.

S. van der Ploeg. Bank default prediction models a comparison and an application to credit rating transitions. Master's thesis, ERASMUS UNIVERSITY ROTTERDAM, 2010.

V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, first edition, 2000.

S. Varma and R. Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 2006.

O. A. Vasicek. Credit Valuation. Technical report, KMV Corporation, 01 1984.

R. N. Vera Kurkova, Nigel C. Steele and M. Karny. *Artificial Neural Nets and Genetic Algorithms*. Springer, 2001.

M. Virag and T. Nyitrai. *Application Of Support Vector Machines On The Basis Of The First Hungarian Bankruptcy Model*. PhD thesis, Corvinus University of Budapest, Hungary, 2013.

W. P. W McCulloch. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

W. Wang, J. Cao, H. Lu, and J. Wang. A default discrimination method for manufacturing companies by improved pso-based ls-svm. *International Journal of Hybrid Information Technology*, 6:95–106, March 2013.

Y. Wang. Structural credit risk modeling: Merton and beyond. *Risk Management*, 16:31–33, 2009.

S. Wani. Comparative study of back propagation learning algorithms for neural networks. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3:1151–1156, December 2013.

C. Warburton. *The Development of International Monetary Policy*. Routeledge, first edition, 2017.

T. Ward and B. Foster. A note on selecting a response measure for financial distress. *Journal of Business Finance and Accounting*, 24:869–879, July 1987.

R. White. The pricing and risk management of credit default swaps, with a focus on the isda model. *OpenGamma Quantitative Research*, 16, 2014.

K. Woods. Generating roc curves for artificial neural networks. *IEEE Transactions on Medical Imaging*, 3:329–337, 1997.

C.-H. Wu, W.-C. Fang, and Y.-J. Goo. Variable selection method affects svm approach in bankruptcy prediction. *Advances in intelligent Systems Research*, 2006.

J. Yang. Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems and their Applications (Volume: 13, Issue: 2, Mar/Apr 1998)*, 2:44–49, 1998.

B. Yap, D. Yong, and W. Poon. How well do financial ratios and multiple discriminant analysis predict company failures in malaysia. *International Research Journal of Finance and Economics*, 54:167–175, 2010.

B. Yap, S. Munuswamy, and B. Mohammed. Evaluating company failure in malaysia using financial ratios and logistic regression. *Asian Journal of Finance and Accounting*, 4:330–342, 2012.

B. Yap, G. Clayton, and Z. Mohamed. Using financial ratios and managing financial risks in investing in grey zone companies: Evidence from malaysia. *International Journal of Finance and Accounting Studies*, 1:2–8, 2013.

G. Zhang, M. Hu, B. Patuwo, and D. Indro. Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. *European Journal of Operational Research*, 116:16–32, 1999.

L. Zhuo, J. Zheng, F. Wang, B. Ai, and J. Qian. A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 47:397–402, 2008.