LUT School of Business and Management

Master's thesis, business administration

Strategic Finance and Business Analytics

Default prediction modeling of Swedish SMEs with machine learning

Author: Jori Kaipio

1st Examiner: Sheraz Ahmed

2nd Examiner: Mikael Collan

ABSTRACT

Author: Jori Kaipio

Title: Default prediction modeling of Swedish SMEs with machine learning

School: School of Business and Management

Master's program: Strategic Finance and Business Analytics

Year: 2020

Master's Thesis: 41 pages, 7 tables, 7 figures

Examiners: Associate Professor Sheraz Ahmed

Professor Mikael Collan

Keywords: default prediction, machine learning, supervised learning, small and medium-sized enterprises

The purpose of the study is to evaluate and compare machine learning models against logistic regression in default prediction of Swedish based small and medium-sized enterprises. Machine learning models are modern approach for classification problems and have proved significant performance compared to statistical models in previous studies. The study consists of literature review on default prediction and an empirical analysis where the default prediction models are built using selected machine learning algorithms. The models selected to the study were logistic regression, Support Vector Machines, bagged decision trees and AdaBoost decision trees. Using equal samples of defaulted and non-defaulted Swedish SMEs, this study showed that the machine learning models slightly outperformed the logistic regression in terms of overall efficiency. The best performing models in this study are found to be AdaBoost decision tree and Support Vector Machine. The findings of this study conclude that Machine Learning models can perform better than the logistic regression model in default prediction of small and medium-sized companies.

TIIVISTELMÄ

Tekijä: Jori Kaipio

Aihe: Ruotsalaisten pk-yritysten konkurssiriskin mallintaminen koneoppimismallien avulla

Tiedekunta: School of Business and Management

Pääaine: Strategic Finance and Business Analytics

Vuosi: 2020

Pro Gradu: 41 sivua, 7 taulukkoa, 7 kuvaajaa

Tarkastajat: Apulaisprofessori Sheraz Ahmed

Professori Mikael Collan

Hakusanat: Luottoriskin ennustaminen, koneoppiminen, ohjattu oppiminen, pienet ja keskisuuret yritykset

Tutkimuksen tarkoituksena on arvioida ja vertailla koneoppimismallien ja logistisen regression ennustuskyvykkyyttä ruotsalaisten pienten ja keskisuurten yritysten luottoriskin mallintamisessa. Koneoppimismallit ovat moderni lähestymistapa luokitteluongelmiin ja aikaisemmissa tutkimuksissa luottoriskin mallintamisessa on löydetty merkittäviä suorituskykyeroja tilastollisiin malleihin verrattuna. Tutkimus sisältää kirjallisuuskatsauksen luottoriskin ennustamisesta sekä empiirisen osuuden, jossa luottoriskin ennustusmallit luodaan. Tutkimukseen valitut mallit ovat logistinen regressio, tukivektorikone, AdaBoost -tehostettu päätöspuu ja Satunnainen metsä bootstrap-aggregoitu päätöspuu. Käyttämällä tasapainoista otosta terveitä ja maksukyvyttömiä yrityksiä tutkimustulokset osoittavat, että koneoppimismallit pystyvät ennustamaan yritysten luottoriskiä hieman logistista regressiota tarkemmin, kun näiden mallien ennustamiskykyä vertaillaan kokonaisuudessaan. AdaBoost-tehostettu päätöspuu ja tukivektorikone olivat parhaat mallit luottoriskin ennustamiseen tässä tutkimuksessa. Tutkimustulokset osoittavat, että koneoppimismalleilla pystytään ennustamaan pienten ja keskisuurten yritysten konkurssiriskiä tarkemmin kuin logistisen regression avulla.

Table of contents

List of figures

List of tables

Abbreviations

AN	Actual Negative
ANN	Artificial Neural Network
AP	Actual Positive
AUC	Area under the curve
CM	Confusion matrix
CV	Cross-validation
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
KNN	k-Nearest Neighbors
MDA	Multiple discriminant analysis
ML	Machine learning
NN	Neural Network
PD	Probability of Default
RF	Random Forest
ROC	Receiver Operating Characteristic curve
SD	Standard deviation
SME	Small and medium-sized enterprise
SVM	Support Vector Machine

TN	True Negative
TP	True Positive

1 Introduction

The probability of an enterprise to default is interesting subject for vast number of interest groups. Examples of these interest groups are credit institutions, governments, individual investors, central banks. As the subject is important for so many, a lot of academic research have been conducted on the area. Pioneer research in multivariate default prediction was Altman's (1968) study, where he used discriminant analysis with financial ratios to evaluate the default risk of enterprises. He generated a Z-score model which is still in use and relevant method for default probability evaluation. However, as the amount of data collected has increased hand in hand with the computing power, new methods for default prediction have emerged. Modern machine learning (ML) methods have proved to be efficient in default prediction purposes and even outperformed the statistical methods (logistic regression and multiple discriminant analysis) which have encouraged more and more research to focus on these newer models.

According to McKinsey (2020) many banks and credit institutions are cautious in utilizing ML methods for credit risk evaluation and using these methods in lower-risk applications (e.g. marketing analysis) because ML methods are harder to interpret and they are not yet so widely used by the industry. Also, the regulators have not specifically instructed the credit institutions to use ML methods in credit evaluation. Previous studies have showed that ML models are effective in predicting future financial distress of companies. Barboza et al. (2017) compared ML models to logistic regression (LR) and found that the ML models outperformed the LR and Multiple discriminant analysis (MDA) models with US company data. McKinsey (2020) also states that fully utilizing ML and artificial intelligence in banking industry in risk management, service tailoring and better decision making could generate over $250 billion of more value in banking industry alone. This should encourage to perform more studies with ML models so that the businesses could improve their credit risk evaluation methods which ultimately should lead to lower possibility of facing future financial crises. It is also certain that the regulators must provide better instructions for financial industry organizations for the use of ML algorithms in their daily business operations.

1.1 Background and motivation of study

Small and medium-sized enterprises (SMEs) are backbone of most economies in the world which makes default prediction of these companies very interesting subject. According to European Commission (2020) in Sweden, SMEs generate 61 % of the total GDP of the country and provide jobs for 65 % of the total employees nationally and the numbers have been growing in recent years. The EU-level average for GDP created by SMEs is 56,4 % and the amount of provided jobs is 66,6 % which indicates that SMEs in Sweden are more crucial for the economy and they are operating more efficiently when compared to EU-averages as they create more value with less employees than averagely in other EU countries. Swedish entrepreneurial organization Förtagarna (2013) states in their report that SMEs have created 80 % of the new jobs in the Swedish economy since 1990. Based on OECD (2019) report on SME financing, Swedish SMEs must rely mostly on bank-based financing as the financial markets in Sweden do not provide alternative financing methods. This is typical in Nordic countries whereas in US or UK the funding sources are not limited to bank-financing for SMEs. Default data from Statistics Sweden (2020) indicates that around 7400 companies fall bankrupt (annual average from 2009-2019) in Sweden which is around 1 % of the active companies.

SMEs are small operators in the financial markets and they do not have publicly listed debt or credit ratings from the big agencies which leads to a situation where the public information of SMEs is limited to their financial statements and demographic information. This emphasizes the meaning of historical financial data in evaluating the quality of the companies. As stated in the introduction, default prediction of companies has been popular topic on studies in financial literature. However, most of the research have been focusing on large corporations instead of SMEs as the information on larger companies is more available. That is why the prevailing general models suits the profile of large corporations better than SMEs. Behr and Weinblat (2017) and Altman and Sabato (2007) states that models built for specifically SMEs for one country at a time lead to better performing models.

This research focuses on predicting default risk of Swedish SMEs using regression models (LR) and machine learning (ML) models. The objective is to compare the performance of LR and ML models that have been found the most effective ones in the previous literature in default prediction studies. The models that are built in this study are LR, Support Vector Machine (SVM), Random Forest (RF) bagged decision tree and AdaBoost decision tree. The models are built using financial data of Swedish SMEs. Following that, the prediction power and the accuracy of the models are evaluated and compared to each other to find the best suitable model for predicting the default risk of Swedish SMEs. Figure 1 demonstrates the focus of the study in Venn diagram.

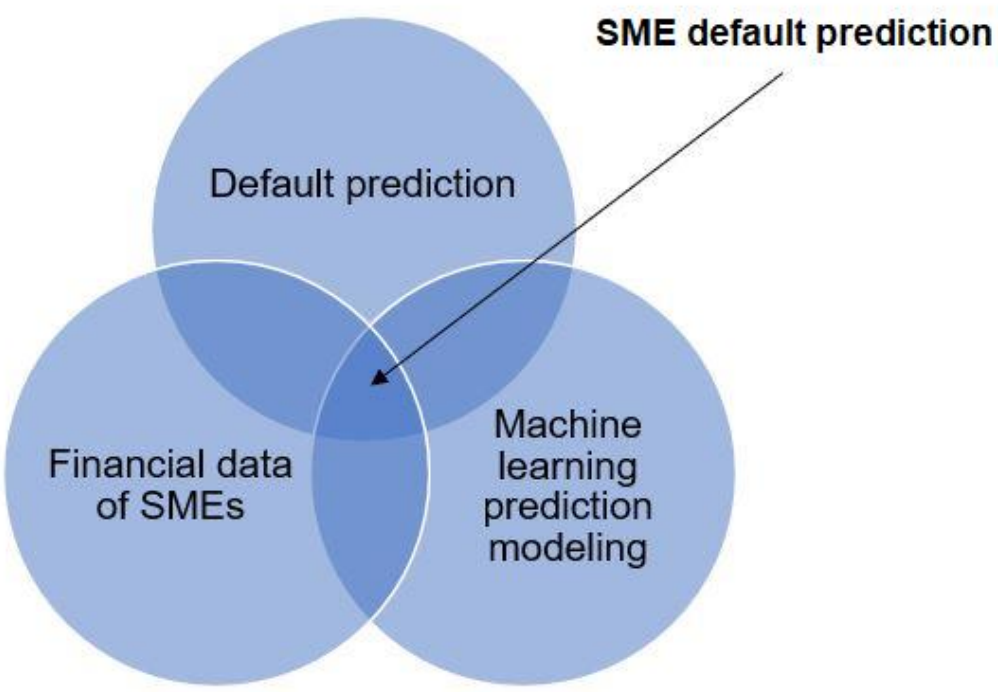

Figure 1. Focus of the study

1.2 Research objectives and research questions

The main goal of the study is to find suitable machine learning models for predicting default risk of small and medium-sized enterprises based in Sweden using their historical financial data and compare the performance of the models against logistic regression model which is the most used model in the literature for corporation bankruptcy

prediction based on literature review by Shi and Li (2019). To fulfill the objective, a holistic review of existing literature on using machine learning models in bankruptcy prediction is necessary to find and narrow down the relevant machine learning models. After the relevant models have been found from the literature, financial and industry data from Swedish SMEs are used to fit the models and the performance of the models are evaluated. Based on the objectives of the study, the research questions are as follows:

1. Which variables and what models should be used for small and medium-sized enterprise default prediction?
    a. What variables should be used for predicting a future default of an enterprise?
    b. What models have been used in default prediction of enterprises in previous studies?
    c. What machine learning models should be used for predicting default and how they are evaluated?
2. What model is the most suitable for predicting future financial distress of Swedish SME?
    a. How the different methods compare to each other with the financial data of Swedish SMEs
    b. Can machine learning models outperform the prevailing logistic regression models?

1.3 Research structure

The research consists of six chapters. Chapter two focuses on previous literature on default prediction of enterprises and explains what financial ratios should be used for default prediction. Chapter two also describes the prevailing default prediction models and algorithms used in previous literature and provides answers to the first set of research questions. The third chapter introduces the selected machine learning models for the study and their performance evaluation methods which is necessary for understanding the methodological side of the thesis. Fourth chapter explains the data and data processing conducted for the initial dataset. Chapter four also describes the descriptive statistics of the sample and its sub-samples and describes which why the

selected ML models got selected for the study and how the evaluation of their performance is conducted. Chapter five describes how the models were developed with MATLAB software and presents the performance results of the models with the training and testing data. Fifth chapter summarizes the results of the models and answers the second set of research questions. The final chapter number six contains conclusions and limitations of the research and discusses about future research on the subject.

2 Literature review

This section introduces the key concepts based on previous literature in default prediction. Default prediction modeling and its history is briefly introduced at first following a review on which independent variables have been used for default prediction of enterprises in previous studies. The last part of the chapter focuses on different methods and algorithms used for default prediction problems and some results of the prior research of SME default prediction studies are introduced. In the end of the chapter the first set of research questions is answered. The literature has been gathered mostly from LUT-university sources (LUT Primo) and in addition to that some open source materials have been utilized.

2.1 Introduction to enterprise default modeling

Default modeling has a long history tracing back hundreds of years and it has evolved hand in hand with evolution of financial markets. The fundamental aspects of risks in lending to enterprises have not changed over time even if the lending instruments and methods have evolved significantly. In the simplest case lender trusts the borrower and exchanges liquid assets (usually cash) for a documented promise to get payments in the future. The price of a loan is dependent on the creditworthiness of the borrower and market variables which are mostly fixed and are not related to the borrower. As it is extremely important for the lender to be able to price the loan depending on the borrower's probability to pay the loan back full, credit risk evaluation is needed. Lending business includes various other risk types (e.g. market risk, interest risk) but they are not introduced here as the research focuses on credit risk and more detailed in default prediction. Adnan Aziz and Dar (2006) divides tailored default prediction models in three classes: univariate and multivariate statistical models, machine learning models, and theoretical models. This research focuses on machine learning models.

Default risk in this context means a situation where the borrower is unable to meet the obligations of the loan or other financial instrument. Probability of default (PD) is a calculated value which explains the possibility for the counterparty to default in each time. Probability of default is usually interpreted from financial ratios or other historical financial data of the counterparty. In this research the counterparties are Swedish small

and medium-sized enterprises. In history, default prediction modeling was mostly done with single financial ratios or some other simple inputs but nowadays as the gathering of the data has increased enormously and computing power is extremely cheap compared to the history, the models can be built with more data and more complexity. Currently the models are often built with multiple financial ratios of the company by using regression modelling or machine learning methods. (Callaghan et al., 2015, 9-30).

Credit rating industry has formed for delivering the information on credit risk of enterprises and other financial assets. There are three big operators Moody's, Fitch and Standard & Poor's which are global and numerous amounts of agencies which operate on a smaller scale (White, 2010). The local agency in Finland is Suomen Asiakastieto. These agencies provide accurate information which is based on long-term data and knowledge but the agencies might be slow to react on their credit ratings and they are not available for smaller enterprises as it is expensive for companies to have the credit score from big operators. Previous research has also shown that tailored models have more predicting power on defaults than credit scores. (Callaghan et al., 2015, 9-30).

2.2 Variables used for default prediction

Horrigan (1968) states that the use of financial ratios to predict enterprise default have been used from late 19th century. The most used ratio that was used at first was current ratio ($\frac{Current\ assets}{Current\ liabilities}$) which is still used for important liquidity indicator of an enterprise. Beaver (1966) was the pioneer of using multiple financial ratios as predictors for default risk. Beaver (1966) used 30 different ratios and calculated a threshold value for each to predict the future financial distress for the enterprise. The use of enterprise balance sheet financial ratios in credit analysis has increased as the time has passed and they are still the foundation of credit risk analysis of businesses and individuals (Callaghan et al., 2015, 23-25).

Financial ratios of a company can be divided into four or five groups based on previous literature which are liquidity, profitability, activity, coverage, and leverage (Altman and Sabato 2007, Feldman and Libman 2007, Yoshino and Hesary 2015, Barboza et al., 2017). Some authors do not divide the variables in leverage and coverage in different

categories but the main difference between variables in leverage category and coverage category are that leverage ratios refer to the ability of a company to meet long-term debt obligations and coverage to short-term debt obligations. Table 1 visualizes examples of financial ratios in the introduced categories. Many default prediction studies of SMEs have been conducted by this categorical approach and using financial ratios from each category (Alman and Sabato 2007, Barboza et al., 2017, Yoshino and Hesary 2015, Ciampi and Gordini 2013).

Table 1. Example financial ratio variables in categories

Financial ratio group	Example financial ratio variable
Liquidity	Current ratio
	Quick ratio
	Cash+accounts receivable/current liabilities
Activity	Asset turnover
	Receivable turnover
	Inventory turnover
Profitability	Profit margin
	Return on assets
	Return on equity
Coverage	Interest cover
	Asset cover
Leverage	Solvency ratio
	Debt to assets
	Debt to equity

In addition to financial ratio variables used for default prediction, models can be extended by adding other types of variables. These variables include firm-specific qualitative variables such as age of an enterprise, industry type or number of employees or macroeconomic variables such as prevailing interest rates in the area. Some research suggests that adding these variables might enhance the predicting power (Grunert et al., 2005) but some available researches focused on smaller enterprises found out that adding these variables did not enhance the predicting power with SMEs (Káčer et al., 2019)

2.3 Multivariate models for default prediction

Introducing multivariate statistical methods for default prediction in the late 1960s changed the credit risk modeling and built the baseline which still stands even if the models have evolved and the amount of data available has grown (Callaghan et al., 2015, 24-30). Edward I Altman (1968) was the first one who created a multivariate prediction model instead of using single financial ratios independently. Altman's key driver was to unite the two prevailing corporate credit risk evaluation methods at that time: financial ratio analysis and statistical techniques. The financial ratios utilized for forecasting were similar with the findings of Beaver (1966) earlier but returned significantly better prediction results when the model used multiple variables at a time. Altman used multiple discriminant analysis (MDA) as the statistical method which derives the best possible linear combination of the chosen independent variables to classify the observations. Altman (1968) created a model which produced a Z-score for each observation that predicted if an enterprise is likely to face financial distress in a future or not.

Altman's Z-score model is still used for default prediction problems despite it is over 50 years old as it seems to be easily interpreted and it gives sufficient results. Obviously new and more sophisticated statistical models and machine learning models have been introduced (logistic regression, decision trees and neural networks to name a few). Besides that, new models have been introduced which utilizes real-time market data which enables the models' to predict default risk daily but these models are not introduced here because the research focuses on default risk prediction from historical financial data of a corporation. (Callaghan et al., 2015, 24-30)

do Prado et al. (2016) conducted a bibliometric study on bankruptcy prediction studies between 1968-2014 to get an overview of used multivariate methods and multivariate algorithms on the research field. They found out that the so-called traditional models (multiple discriminant analysis and logistic regression) remain still in use even though the more modern ML methods have conquered some of the space. These traditional models are used for baseline in many studies that compare the prediction performance of multiple models (usually traditional model versus machine learning models). Hybrid models which combines traditional models and machine are also getting attention. An

example of a hybrid model was research by Li et al. (2016) where they evaluated credit risk of SMEs by combining logistic regression and Artificial Neural network.

Logistic regression seems to be the most used and most efficient method when using traditional statistical methods for small and medium-sized enterprise default prediction. Altman and Sabato (2017) built a logistic regression model for US based SMEs to compare its performance with Altman's prior Z-score model and included an MDA model in the research. The results showed that the logistic regression model performed significantly better than both, the MDA model built with SME data and the Z-score model. Based on the results, they suggested that it is reasonable to build a tailored model for SME default prediction rather than use so called general models. Cultrera and Brédart (2016) investigated SMEs in Belgium and found out that the logistic regression model for bankruptcy prediction performed well in their sample. Sirirattanaphonkun and Pattarathammas (2012) compared the performance of logit model and MDA model on Thai SME data and found out that the logit model outperformed the MDA model (logit model had an accuracy of 85.5 % and MDA 81 % for out-of-sample predictions).

2.4 Machine learning methods for default prediction

The popularity and usage of machine learning models has been increasing in financial industry in recent years which has led to research on default prediction models with machine learning models also. ML models are especially efficient in prediction problems as they can find patterns in the data more efficiently than statistical models as they can handle non-linear relationships in the data. In real world applications, banks face some issues with using machine learning models as some of the algorithms cannot be interpreted by human and they are extremely complex to audit from outside by the financial industry regulators. (Van Liebergen, 2017)

Martin et al. (2019) conducted a literature review on using machine learning models in bank risk management. They found out that ML models are used in evaluating almost all risk types in banking industry but the popularity of evaluating credit risk is the highest among the banking risk classes. Support Vector Machines, Neural Networks and en-

sembled decision trees seems to be the most used methods for default prediction purposes of enterprises, and they have also outperformed other statistical and machine learning models on many occasions in previous research.

Barboza et al. (2017) compared the performance of several ML algorithms with enterprise financial data from the US. They base their research on Altman's (1968) initial paper on default prediction and use similar variables. The purpose of the paper was to evaluate different default prediction models (statistical and ML models) with a sample of 13.000 enterprises. It was found that the ensembled decision trees (boosted trees, bagged trees, and Random Forests) had the best prediction power in all tests. SVM models outperformed neural networks and statistical models by wide gap and had the second-best prediction power.

Support Vector Machine and its variants seems to be the most used machine learning model for bankruptcy prediction in prevailing literature and many of the studies have had encouraging results. For example, Ribeiro et al. (2012) had found that enhanced SVM model had relatively good prediction power with French enterprise data and Kim and Sohn (2010) found out that SVM model outperformed logistic regression and neural network models with Korean SME data.

Ensembled decision trees are not that often applied in research for bankruptcy prediction purposes when compared to SVMs or NNs, but it has shown some interesting results besides the research that Barboza et al. (2017). Behr and Weinblat (2017) studied a large set of enterprises in seven countries and found out that RFs provided promising results in all of the countries but highlighted that models are more efficient if they are fitted for one country at a time. Yeh et al. (2014) conducted going-concern study on Taiwanese companies and found out that RF hybrid model had a good performance on predicting enterprises going concern.

Heo and Yang (2014) compared ML models with default prediction of Korean construction companies and found out that AdaBoost decision tree outperformed SVM, ANN and normal decision tree models. Kim and Upneja (2014) conducted a financial distress prediction study with U.S. restaurants with AdaBoost and stated that the models was able to perform well on classification of the companies.

2.5 Summary of literature review

The aim of the literature review was to review existing literature on default prediction of SMEs and answer the first set of research questions. The research questions which were investigated in the chapter were:

> *Which variables and what models should be used for small and medium-sized enterprise default prediction?*
>
> > *What variables should be used for predicting a future default of an enterprise?*
> >
> > *What models have been used in default prediction of enterprises in previous studies?*
> >
> > *What machine learning models should be used for predicting default and how they are evaluated?*

The variables used in default prediction seems to be mostly financial ratios derived for balance sheet of an enterprise. In many studies the financial ratios have been divided into 5 categories which all describe a different part of performance or healthiness of an enterprise. The categories are following liquidity, profitability, activity, leverage, and coverage. Example variables in these categories are introduced in table 1 at chapter 2.2. Macroeconomic variables (interest rates, inflation etc.) and firm-specific qualitative variables (age of company, number of employees etc.) are used in addition to financial ratios in some studies but almost all of the studies are based on financial ratios.

Multivariate default prediction models were introduced for default prediction problems in the 1960s after the seminal research paper from Altman (1968). These models are mostly based on different financial ratios and can outperform univariate models which had been used before significantly. Most used models in default prediction in the history have been the statistical models from which logistic regression is the most used model nowadays but multiple discriminant analysis is still used in some studies. In recent years, more modern machine learning models have gained space from statistical models as they have showed better predicting performance in studies than statistical models.

Various ML models have been used in default prediction studies. Chapter 2.4. introduces the most used models based on Martin et al. (2019) who conducted a literature review on usage of machine learning models in banking industry. The most used models have been SVMs, ANNs and ensembled decision trees (boosted trees, bagged trees, and Random Forests) which have also shown the best prediction performance among models. All these three model types mentioned have a great deal of dimensions inside the models (e.g. hyperparameters, number of nodes, number of decision trees etc.) but it seems that by optimizing these models for the prediction purposes they are able to recognize patterns and adapt the most information on the data available. Based on the comparison studies of ML models in default prediction, the best performing models seem to be SVMs and ensembled decision trees (Barboza et al., 2017, Heo and Yang, 2014).

3 Machine learning classification models

This chapter answers briefly what machine learning is and for what it can be used for. The relevant ML models for the study are introduced and how the performance of the models is evaluated.

The field of machine learning has grown rapidly in last decade. As technology has evolved, it has become possible to gather and store data from almost all imaginable devices and places all around the world. We all have become producers and users of data. We want to see the reviews of a movie before watching one, we are checking the average temperatures of given location before booking a holiday, our behavior is monitored by cookies when using websites to get specialized offers and services. At the same time when the ability to gather and store huge amounts of data has inflated, the computing power of computers has also skyrocketed which has led to a situation where we have a huge amount of data to analyze and the computing power for the task. (Alpaydin, 2014, 1-4).

Joshi (2020, 9-20) describes that machine learning model is a program that can predict or learn to produce a behavior that it is not explicitly programmed to do. Machine learning models consists of three following features: it consumes data, quantifies the error or the distance between the performance of the model to the ideal performance and adjusts the model with that information to be able to perform better in the following iterations.

Machine learning algorithms can be ultimately divided in three groups: supervised, unsupervised and reinforcement learning algorithms. A good example of a supervised learning task is a classification, where a model is built with a relevant dataset (training data) of which we know the labels of the data and the output. After training a model, it can be used to classify a dataset of similar items. This kind of classification can be used for example to group enterprises in high credit risk group and low credit risk group. In unsupervised learning the labels of the data are not known. Unsupervised learning methods are usually used in clustering problems, where the model tries to cluster the dataset based on the attributes of observations. A real-life clustering problem could be a situation where company wants to divide their customers to groups

based on the customers buying history. A reinforcement model is a hybrid of these two previous models. (Joshi, 2020, 9-20). This research focuses on classification models as the objective is to classify Swedish SMEs into two groups: non-defaulting companies and defaulting companies.

3.1 Supervised machine learning methods for classification

This chapter introduces the most common machine learning models used for classification problems using supervised learning. Some of the introduced models can also be used for other tasks than classification but that will not be the focus of this chapter.

3.1.1 Logistic regression

Logistic regression is a commonly used statistical technique when the outcome of the variables is categorical (multiple categories or binary). Logistic regression uses maximum likelihood estimation which adjust the coefficients of the model until a certain criterion is fulfilled and the model is accepted. After that, the model can classify the observations into categories with the found threshold. Logistic regression can be described as follows:

$$log_{\beta} \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n$$

Where p is the probability of an event, β are regression coefficients for selected independent variables. The idea is to solve a p from the equation which gives the probability to belong to a predicted class. (Osborne, 2015, 19-44)

3.1.2 Support vector machines

Support vector machine (SVM) is a supervised machine learning method which is widely used in classification problems in various industries and it has showed good performance in prediction problems (Boyle, 2011). Kim and Sohn (2010) found SVM performing well for default prediction of Korean enterprises. SVM was introduced by Cortes and Vapnik (1995) and was initially introduced for classification problems with two groups.

In binary classification problem SVM tries to find an optimal hyperplane that will create a maximum separation between two classes with minimal number of used data points to separate the two classes in the dataset. The observations are classified into categories regarding to which side they are regarding the hyperplane and the goal of the function is to maximize the distance between the class boundaries using support vectors. SVM models can handle nonlinear data with using nonlinear kernel functions such as gaussian or quadratic. SVM models have hyperparameters which can be optimized such as the kernel function and penalty which can be optimized during the model development phase. Figure 2 shows a linear support vector machine where the solid line is the hyperplane and dotted lines are the support vectors that represent the boundaries between the two classes marked with red and blue dots. (Joshi, 2020, 65-71).

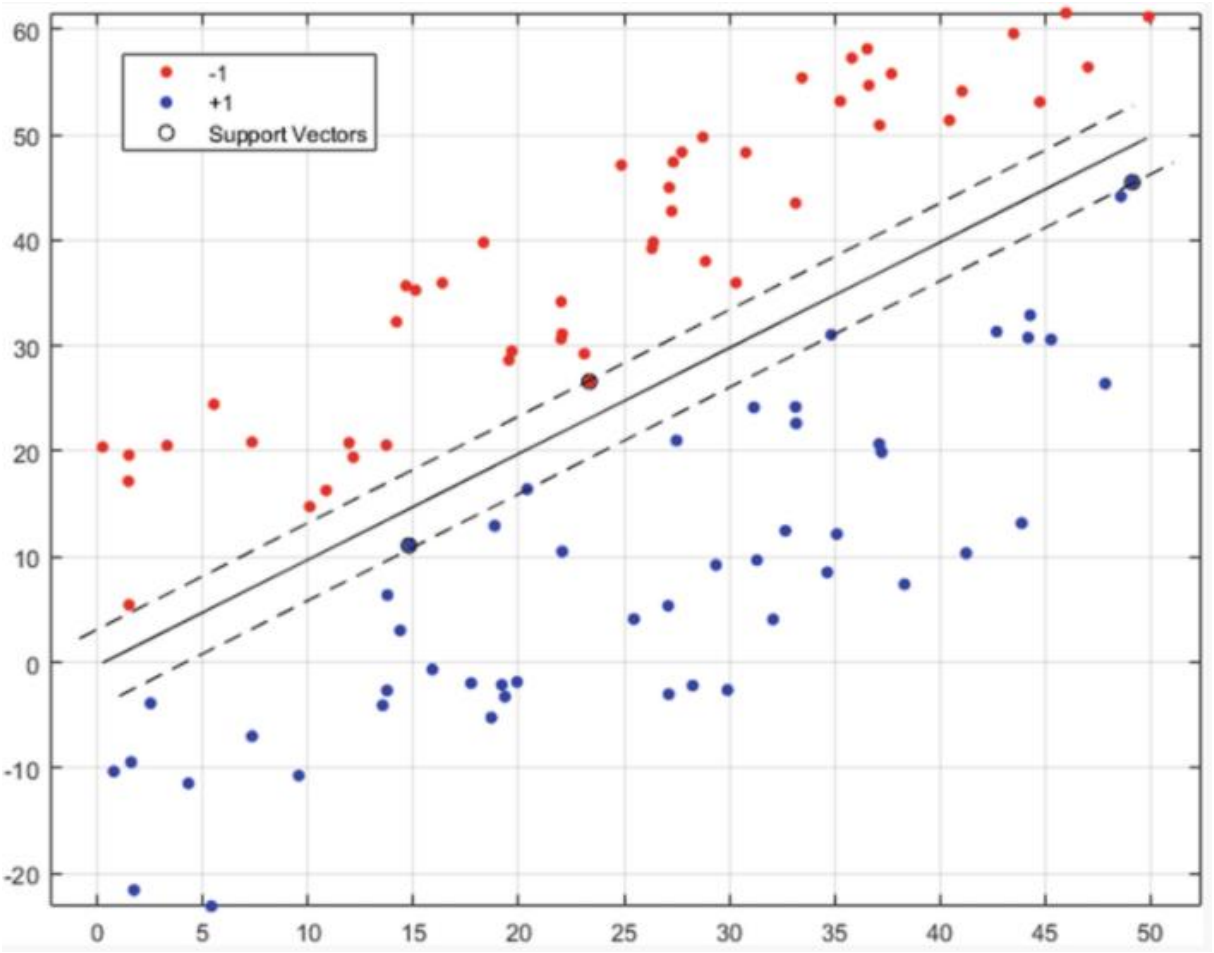

Figure 2. Linear binary SVM. (Joshi 2020)

3.1.3 Decision trees

Decision trees use unique approach to machine learning compared to e.g. SVMs and Artificial Neural Networks (ANN) and can be used for classification or regression problems. The output of the model is based on a hierarchical decision-making process which is quite like human behavior in real-life decision-making situations. Decision trees can also interpret non-numerical data which is typically impossible for other ML methods. (Joshi 2020, 53-63).

In the simplest form, decision tree uses a binary decision-making tree to find the ultimate output of the model by creating a threshold of each node of the model to categorize the observations (Alpaydin, 2014, 213-220). There are also more complex decision tree models, ensembled decision trees (random forests, boosted and bagged trees) where multiple decision trees are built, and the data and predicting power of numerous trees is adapted into a one model which can improve the performance of the model significantly. Boosting decision trees means that individual trees are trained in sequence where the models learn from mistakes of the previous models. In bagging, individual models are trained similarly by a random subset of the sample and then aggregating the outputs of the decision trees. (Joshi, 2020, 53-63).

Random Forest is an ensembled decision tree method introduced by Breiman (2001). RF is based on bagging algorithm for decision trees which means that the algorithm builds a set of decision trees with a random sub-sample of the initial sample and aggregates the information of all the trees. Bagged trees perform well with outliers because single tree does not affect the whole model that much. In addition to that, RF model also select a random subset of features for the model for every iteration which helps to avoid overfitting if some of the features in the model have significantly more predicting power than other features. The output of a RF model is the one which gets most votes from individual trees built for the model.

AdaBoost decision tree is a boosted ensemble decision tree model which utilizes the AdaBoost algorithm introduced by Freund and Schapire (1996). Boosted decision tree uses different approach than in bagging. In boosted decision tree, the first decision tree is built based on a random sample of the data, but the next trees are built based

on the outcome of the first tree. Thus, the trees cannot be built parallelly and require more computing power than bagged trees. AdaBoost algorithm serves as the performance enhancing algorithm in boosted decision tree by trying to minimize the misclassification rate of the model. AdaBoost decision tree gives a voting weight for each tree built for the model based on their misclassification rate with the training data. The lower the misclassification rate is, the more weight the tree gets for voting the final output of the model. (Joshi, 2020, 53-63).

3.2 Evaluation and validation of the models

Model evaluation is interesting when evaluating a performance of a single model as well as when comparing the performance of several models. Model performance can be measured in two different aspects: precision and speed. Precision evaluates how accurately the model can do what it is planned for and speed evaluates how fast the computations are. Usually the evaluation of the model's performance is done by evaluating the precision because models are rarely so complex that there is insufficient amount of computational power. (Kubat, 2017, 211-229). This research focuses only on evaluating the precision of the models.

3.2.1 Confusion matrix

Confusion matrix (CM) is simple, effective, and illustrative system for evaluating classification prediction performance of ML models. Confusion matrix suits well for evaluating a model which divides observations into binary classes where the model is predicting if an observation belongs to a class A or not. Confusion matrix can also be built for multi-class classification models if needed. Binary classification Confusion matrix illustrates the predicted results in two-by-two matrix where there are four options:

- True Positives (TP): positive prediction, true value positive
- False Positives (FP): positive prediction, true value negative
- False Negatives (FN): negative prediction, true value positive
- True Negatives (TN): negative prediction, true value negative

The diagonal values of confusion matrix (TP and TN) illustrates the accuracy of the model as they represent the situation when the model predicts right and the non-diagonal values of the confusion matrix (FP and FN) illustrates the number of misclassified observations with the data.

Figure 3 shows example of a confusion matrix.

	Predicted no	Predicted yes	
Actual no	TN= 30	FP= 20	50
Actual yes	FN= 10	TP= 40	50
	40	60	N=100

Figure 3. Confusion matrix example

For model evaluation purposes, several performance ratios can be calculated:

- Accuracy (how often the model is correct): $\frac{TN+TP}{N}$
- Misclassification rate (how often the model is incorrect): $\frac{FP+FN}{N}$
- Sensitivity (how often the model predicts true when the result is true): $\frac{TP}{TP+FN}$
- Specificity (how often the model predicts no when the result is no): $\frac{TN}{TN+FP}$
- False Negative Rate (FNR) (how often the model predicts false wrong): $\frac{FN}{FN+TN}$
- False Positive Rate (FPR) (how often the model predicts true wrong): $\frac{FP}{FP+TP}$

The metrics used should be selected in line with the data and the goals of the predictions. In a situation where it is crucial to capture all the true positives, sensitivity might be more important ratio than accuracy (Japkowicz and Shah, 2011, 94-105).

3.2.2 Receiver Operating Characteristic curve

Receiver Operating Characteristic curve (ROC) illustrates the relation between True Positives and False Positives predicted by the model. ROC curve is drawn between points (0,0) and (1,1) where in point (0,0) there is no correct classifications neither false positives. In point (1,1) model always predicts positive result for the classification. Figure 4 shows and example of few ROC curves by Kotu and Deshpande (2014).

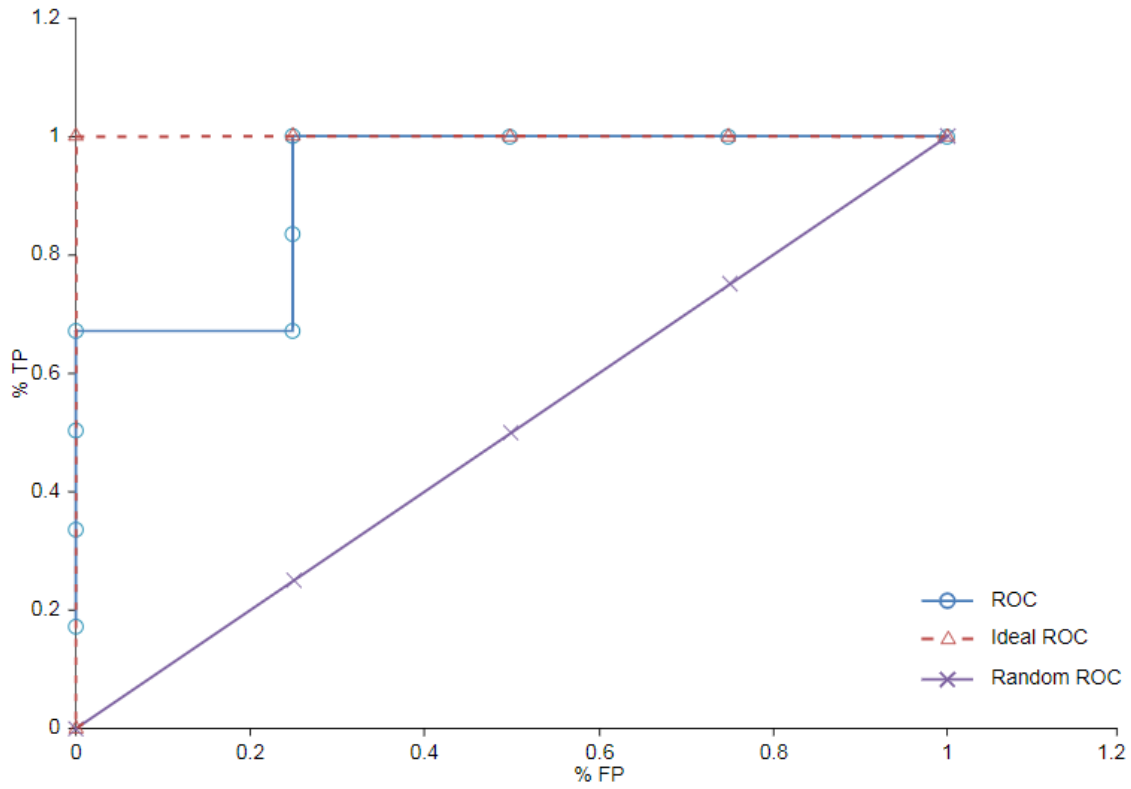

Figure 4. Example ROC curves. (Kotu and Deshpande 2014)

ROC curve is usually interpreted with area under the curve (AUC) which assesses the models' prediction performance. When the model predicts the classes always right, the value of AUC is 1. AUC value of 0,5 is a random threshold which means that if the models' AUC is 0,5 the model performs like random guessing which means that whenever the AUC is > 0,5 it has some predicting power. (Kotu and Deshpande, 2014, 261-264).

3.2.3 Training and testing set

The validation of machine learning models is an important task and is used to measure if the model can be generalized by using another set of data than the model was trained with to validate the results. In a situation where the training data is used to validate the model, the model will be overfit for the data and gives biased results. That is why the original dataset is divided into training set (for building the models) and test set (for validating the performance). Usually the training set is larger (60-80 % of the original sample) than the test set but there is no simple rules or generalized thresholds for the

set sizes. The key point in dividing the sets is that both should represent the variance in the whole sample. Also, the bigger the holdout for the test set is, the more information is left out from the training of the model. (Kohavi, 1995).

3.2.4 Validation and hyperparameter optimization

Cross-validation is a method where the training dataset is divided into k number of folds and the model is trained k times. The training set is built in a way that the training set is randomly split in k-number of folds with equal size of data and the model is trained and tested k times. The cross-validation is used to evaluate the changes in the model between the folds and to avoid overfitting for the training data. If cross-validation is used, the ultimate evaluation on the prediction performance should still be done with the test set. (Kohavi, 1995).

Hyperparameter optimization is almost always present when building machine learning models. Optimization is an important task because it usually boosts the prediction performance of the model when compared to a model without hyperparameter optimization. Different models have their own hyperparameters, e.g. hyperparameters of SVM are kernel function and scale and box constraints and hyperparameters of decision trees are number of learners and number of splits. Hyperparameter optimization is usually conducted using Bayesian optimization which uses probabilistic surrogate model and an acquisition function in making the choice of which spot to assess next (Hutter et al., 2019, 1-33).

4 Data and methodology

This chapter introduces the data and model selection and development for evaluating default risk of Swedish SMEs for 1-year period. I introduce the descriptive statistics of the dataset and discuss how the data has been preprocessed and which assumptions and decisions I have made during the data processing and model selection phases based on previous literature and goals of the study. Figure 5 presents the model building process from data collection to the model evaluation (model building and evaluation is introduced in chapter 5).

Figure 5. Process of building and evaluating the models

4.1 Data collection

The data for the study was collected from Bureau van Dijk's (2020) Amadeus database which contains financial information of public and private companies in Europe. Dataset consists of Sweden-based small and medium-sized enterprises from years 2016-2019. SME definition is derived from OECD (2020) which describes SME in Europe as an enterprise which has less than 251 employees, and the annual turnover is below 50 million euros. The sample was collected by first searching for the defaulted SMEs with non-missing data. The search criterion of defaulted enterprises included inactive bankrupt and liquidation status companies and active enterprise which has default on payments or insolvency proceedings currently. The amount of defaulted companies in the sample after cleaning and preprocessing procedures was 901, initially 930. After that I randomly selected same amount (901) of non-defaulted companies from the same timeframe which makes the sample balanced. The initial amount of non-defaulted companies from this timeframe was around 60 000. Balanced sample means that the sample contains same amount of defaulted and non-defaulted companies. This approach has been used by e.g. Altman (1968), Barboza et al. (2017), and Ciampi and Gordini (2013) in prior default prediction studies. The dataset includes financial

information from the last available year for the defaulted companies and one-year information per observation from the whole sample. The companies in the sample are from five different industries which are: manufacturing, construction, wholesale and retail, transportation and storage and accommodation and food service activities. Table 2 below shows the number of enterprises in the sample by industry. The observations are quite evenly distributed by industries between defaulted and non-defaulted groups.

Table 2. Enterprises in the sample by industry

Industry	**Construction**	**Wholesale and retail**	**Accomodation and food services**	**Manufacturing**	**Transportation and storage**	**Sum**
Defaulted	288	226	169	114	104	901
Non-defaulted	233	312	99	161	96	901

4.2 Feature selection

The dependent variable (Y) of the research is default which is a binary variable (when the enterprise is defaulted it gets the value of 1 and when it is not defaulted it gets value 0). Feature selection of independent variables was conducted by following relevant literature on default prediction because the dataset of defaulted SMEs included quite restricted number of variables to explore. Many of the previous studies on the subject have followed the Altman's (1968) seminal research which proposed that the financial ratios should be divided in 5 categories and select a single ratio from each category. I chose to lean on the approach of Altman (1968) and Barboza et al. (2017) by using variables that have been significant in other studies as the predicting independent variables. The variables in the categories are current ratio (X1, liquidity), profit margin (X2, profitability), asset based solvency ratio (X3, leverage), interest cover (X4, coverage), and asset turnover (X5, activity): The variables in categories and their formulas are presented in Table 3.

Table 3. Independent variables by categories and their formulas

Category	Variable	Formula
Liquidity	Current ratio	$\frac{Current\ assets}{Current\ liabilities}$
Profitability	Profit margin (%)	$\frac{Profit\ before\ tax}{Operating\ revenue} * 100$
Leverage	Asset based solvency ratio (%)	$\frac{Shareholders\ funds}{Total\ assets} * 100$
Coverage	Interest cover	$\frac{Operating\ profit}{Interest\ paid}$
Activity	Asset turnover	$\frac{Operating\ revenue}{Total\ assets}$

4.3 Data preprocessing and cleaning

As stated before, the data was collected from Amadeus database provided by Bureau van Dijk (2020). The initial sample had 930 defaulted companies from which 29 were removed due to missing values in the independent variables. All the removed companies had more than one missing value in the independent variables and the whole company was removed from the sample, not just the missing variables leaving the sample to include 901 defaulted companies. Non-defaulted companies did not have missing values in the independent variables. I was able to leave out the observations with missing values because the sample was still extensive enough as most of the default prediction studies have been conducted with smaller samples. The dependent variable of the model was not initially in the dataset, therefore I had to add it to each observation based on the information if the enterprise had defaulted or not. The dependent variables selected were available straight from the data source except asset turnover, which I added to the dataset by conducting the calculation based on the formula.

No transformations, e.g. normalization or standardization of the independent variables were done although it might affect the predicting power the model. Similar approach

was used with ML default prediction study by Barboza et al. (2017). I ended up with this decision because of two main reasons, that the model would be as easy as possible to use and understand by users or other interest groups and that the model could be used efficiently with out-of-sample predictions in the future with a data from other country or timeframe.

4.4 Splitting the dataset into training and testing sets

As introduced in chapter 3.2.3. datasets for machine learning models are usually divided in a way that around 60-80 % of the data are used for training the models and the rest of the sample is used to test how the model is performing with the rest of the data. Surprisingly, quite many studies on default prediction has decided to use only a relatively small training sets which have been 20-50 % of the sample, e.g. Barboza et al. (2017) and Ciampi and Gordini (2013). I decided to use more common method on machine learning applications and divide the sample to the training and test set in 70-30% respectively. I selected to divide the sample as described because I wanted to follow the general two rules of thumb that Joshi (2020, 169-176) introduced which are following: the model should get as much data as possible to be able to capture all the dimensions in the data and the test set should include sufficient variation and heterogeneity for the testing results to be robust. The data was split randomly into the training and testing set. The descriptive statistics of each set are introduced in the next chapter 4.5. The training set consists of 630 defaulted and 630 non-defaulted companies and the test set has 271 defaulted and 271 non-defaulted companies.

4.5 Descriptive statistics

Table 4 introduces descriptive statistics of the whole sample and different subsets of the sample (non-defaulted companies, defaulted companies, training set and test set). Table includes information on minimum values, maximum values, average values (mean), median values and standard deviation (SD) of the sample and subsets. As seen in the table that two of the variables which describes how much debt a company has and how great the interest expenses are comparing to earnings (X3 and X4) have large deviation on all of the sets which means that the values of these variables vary very much around their mean which might affect the predicting power of the variables.

The descriptive statistics on training and test set seems to be quite similar which tells us that both subsets should have enough information on the dataset and captures the dimensions of the sample in a similar way.

All the variables except asset turnover seem to have lower medians and means in defaulted than non-defaulted companies (Table 4). This is expected because lower values indicate that the companies that have defaulted have had worse financial ratios than the non-defaulted companies. It is somewhat surprising that defaulted companies have bigger scores in asset turnover ratio (also the SD is higher which can describe this in some extent) because bigger turnover ratio means that the company is able to create more revenue with its assets.

Table 4. Descriptive data of the whole sample and different subsets.

Whole sample (N=1802)					
	X1 (current ratio)	X2 (profit margin %)	X3 (asset based solvency ratio %)	X4 (interest cover)	X5 (asset turnover)
Min	0.00	-99.80	-99.03	-99.00	0.10
Max	67.16	48.93	94.74	987.00	31.47
Mean	1.65	-0.18	22.91	46.74	3.37
Median	1.24	1.18	22.46	3.47	2.73
SD	2.13	13.04	31.23	141.68	2.70
Non-Defaulted companies (N=901)					
Variable	X1 (current ratio)	X2 (profit margin %)	X3 (asset based solvency ratio %)	X4 (interest cover)	X5 (asset turnover)
Min	0.10	-79.17	-60.44	-94.80	0.10
Max	67.16	42.18	91.24	987.00	19.35
Mean	1.98	3.73	37.00	80.17	2.90
Median	1.51	3.39	36.26	12.38	2.40
SD	2.59	8.58	23.10	172.41	2.06
Defaulted companies (N=901)					
Variable	X1 (current ratio)	X2 (profit margin %)	X3 (asset based solvency ratio %)	X4 (interest cover)	X5 (asset turnover)
Min	0.00	-99.80	-99.03	-99.00	0.11
Max	27.51	48.93	94.74	909.00	31.47
Mean	1.32	-4.09	8.82	13.30	3.84
Median	1.06	-0.85	10.56	-0.29	3.05
SD	1.45	15.37	31.94	90.60	3.14
Training set (N=1260)					
Variable	X1 (current ratio)	X2 (profit margin %)	X3 (asset based solvency ratio %)	X4 (interest cover)	X5 (asset turnover)
Min	0.01	-88.81	-99.03	-99.00	0.10
Max	67.16	48.93	94.74	945.50	31.47
Mean	1.69	0.07	23.81	48.36	3.38
Median	1.26	1.27	23.85	3.73	2.70
SD	2.38	12.37	31.02	143.27	2.76
Test set (N=542)					
Variable	X1 (current ratio)	X2 (profit margin %)	X3 (asset based solvency ratio %)	X4 (interest cover)	X5 (asset turnover)
Min	0.00	-99.80	-98.18	-98.50	0.16
Max	16.67	48.45	91.24	987.00	19.35
Mean	1.54	-0.76	20.80	42.96	3.36
Median	1.18	0.93	19.28	2.78	2.79
SD	1.35	14.47	31.62	137.97	2.54

Tables 5 and 6 introduces the correlation matrices of independent variables (current ratio (X1, liquidity), profit margin (X2, profitability), asset based solvency ratio (X3, leverage), interest cover (X4, coverage), and asset turnover (X5, activity)) of the defaulted and non-defaulted companies, respectively. The correlations between the variables

are relatively low in both groups and do not suggest that variables should be removed from the dataset. The correlation coefficients are quite similar in both groups except the correlation between the variables X2 and X5 where there is positive correlation in defaulted companies and negative correlation in non-defaulted companies. This is interesting because it suggests that increased activity enhances the profitability in defaulted companies and weakens the profitability in non-defaulted companies.

Table 5. Correlation matrix of the variables in defaulted companies.

	X1	X2	X3	X4	X5
X1	1				
X2	0.138	1			
X3	0.333	0.331	1		
X4	0.144	0.355	0.285	1	
X5	-0.098	0.120	-0.207	-0.041	1

Table 6. Correlation matrix of the variables in non-defaulted companies.

	X1	X2	X3	X4	X5
X1	1				
X2	0.051	1			
X3	0.264	0.378	1		
X4	0.112	0.392	0.329	1	
X5	-0.110	-0.125	-0.229	-0.025	1

4.6 Model selection

The model selection was done based on previous literature introduced in previous chapters 2 and 3. I selected four models of which logistic regression represents the most used statistical model based on the literature and other three are ML models. The models selected for the empirical study are:

- Logistic regression (LR)
- Support Vector Machine (SVM)
- Random Forest bagged ensemble decision tree
- AdaBoost boosted ensemble decision trees

LR was chosen for the study because it seems to be the most used statistical model in the industry currently and has shown some prediction power in previous studies. Logistic regression is also easy to interpret and use and is widely used in classification problems in other industries and areas also. LR serves as a baseline for comparing the performance of the “traditional” and newer ML models.

SVM was picked for the study because it has shown successful results in previous default prediction studies and it is one of the most used ML model in this area. SVM also provides a unique approach to this study compared to the other models as it is the only distance-based model.

Ensembled decision tree models in general got selected because they seem to have ability to perform well on credit risk problems. For example, Barboza et al. (2017) found out that ensembled decision trees performed the best in their research comparing different ML models for US-based companies default prediction study. Other studies where ensembled decision trees performed well were introduced in the literature review. Tree-based models are not that commonly used in previous literature on default prediction than SVMs and Artificial Neural Networks. Decision trees provide an interesting and different approach and operating logic than other ML models selected to the study. Random Forest bagged decision tree was selected to the study as it has shown good performance in previous studies. The Random Forest model used in the study follows algorithm introduced by Breiman (2001). Boosted ensembled decision trees got selected to the study mainly for similar reasons than bagged decision trees. They have proved to provide robust prediction results in previous studies and are not that commonly used in history when compared to e.g. Artificial Neural Networks. The selected method was AdaBoost boosted decision tree which utilized AdaBoost function for optimizing the trees introduced by Freund and Schapire (1996).

4.7 Model evaluation and hyperparameter optimization

The model evaluation in three phases. When the model is developed, it is validated by observing if changes in the model enhance the performance of the predictions by using 5-fold cross-validation for all the models. At the model development phase also the hyperparameter optimization is conducted as they can have significant effect on the

model's prediction performance (Hutter et al., 2019, 3-33). Bayesian optimization will be used for hyperparameter optimization as it is the state-of-the-art optimization method for machine learning models (Hutter et al., 2019, 3-33). Secondly the models are evaluated with the training data for the optimized models to find out how the models perform with the data it was trained with. At the last stage all the models are evaluated by comparing their predictive power with the test data which was held out from the initial sample and was not used for training the models.

Literature proposes that confusion matrices and ROC AUC measures should be used for evaluating the performance of classification models. The confusion matrices and ratios obtained from the confusion matrix are used accompanied with ROC AUC curves for evaluating the performance of a single model and when ranking the prediction performance between the models in this study. The confusion matrix ratios used will be accuracy, sensitivity, specificity, false negative rate (FNR) and false positive rate (FPR). The two of the most important metrics from these are sensitivity and FNR as it is more harmful to predict unhealthy enterprise to a healthy class than to classify healthy company as unhealthy. This is because a revenue from single loan does not usually cover the expenses from borrower's default in other loan.

5 Development of the models and results

The development of the models and the performance results with the training data and testing data are covered in this chapter. The models are built using MATLAB's Classification Learner app (MATLAB 2020a) which is used for building the models with cross-validation and optimizing the hyperparameters of the models using Bayesian optimization. All the models were 5-fold cross-validated at the model development phase to avoid overfitting of the models to the training data. ML models have different hyperparameters which can be optimized to enhance the prediction power of the models. Hyperparameter optimization is conducted for SVM, AdaBoost decision tree and Random Forest bagged decision tree. MATLAB's classification learner application uses minimum classification error with the training data in hyperparameter optimization for finding the best performing hyperparameters for the model (MATLAB 2020b). After the models have been built and the hyperparameters have been optimized, the models are exported from Classification Learner and the predictions with the testing data are conducted in MATLAB software. The hand-made MATLAB codes are provided in the appendix section. The development of the models, hyperparameter optimization and selected confusion matrix metrics with training data are introduced at first and then the models are evaluated based on their prediction performance on the testing set. The confusion matrix metrics and their formulas are introduced in chapter 3.2.1. and the ROC AUC is introduced in chapter 3.2.2. The research question number 2 and its sub-questions are answered at the end of this chapter. The models were built and evaluated with MATLAB software.

5.1 Logistic regression

Logistic regression model was built with default settings of MATLAB Classification Learner without hyperparameter optimization as it is not available in the app. Classification Learner uses MATLAB's "fitglm" function. The model was validated with 5-fold cross-validation in the model development phase. Model development took around one second. LR model performed followingly with the training data (in-sample performance):

Accuracy	73.7 %
Sensitivity	75.2 %
Specificity	72.1 %
False Negative Rate	25.6 %
ROC AUC	0.80

The model performed well better than random guessing with the training data and does not show signs of overfitting as the accuracy is quite far from perfect. After the model was trained, it was exported to MATLAB workspace to use the model for predicting observations with testing data. MATLAB code for testing the model is introduced in Appendix 1. The prediction performance with testing data is introduced with the results of other models in chapters 5.5. and 5.6.

5.2 Support Vector Machines

SVM model development was done by utilizing MATLAB Classification Learner app using 5-fold cross-validation and hyperparameter optimization. Classification Learner uses “fitcsvm” function for building optimizable SVM model. Hyperparameter optimization was conducted with Classification Learner and the optimized hyperparameters were kernel function, box constraint level and kernel scale. Kernel function was optimized from gaussian, linear, quadratic, and cubic and box constraint level and kernel scale were optimized in range [0.001,1000]. The optimization was applied with Bayesian optimization using 100 iterations and it took around 45 minutes to run. The most accurate model found was with Gaussian kernel using kernel scale 277 with box constraint 128. This model had good accuracy it predicted non-defaulted companies well but had a high False Negative Rate (27.3 %) and low sensitivity (70.2 %) compared to a SVM model with linear kernel with automatic kernel scale and box constraint of 1. For that reason, the selected SVM model for the study was a linear SVM with default settings of MATLAB. SVM model performed followingly (in-sample performance):

Accuracy	73.9 %
Sensitivity	77.6 %
Specificity	70.2 %
False Negative Rate	24.2 %
ROC AUC	0.80

SVM model shows good performance with the training data but the performance metrics do not show signs of overfitting. After training the model, it was exported to MATLAB workspace to use the model for predicting observations with testing data. MATLAB code for testing the model is introduced in Appendix 2. The prediction performance with testing data is introduced with the results of other models in chapters 5.5. and 5.6.

5.3 AdaBoost boosted decision tree

Classification Learner app of MATLAB was used to build AdaBoost decision tree model. Classification Learner uses “fitcensemble” for building optimizable AdaBoost decision tree and used the AdaBoost boosting function introduced by Freund and Schapire (1996). AdaBoost boosted decision tree model was built using 5-fold cross-validation and hyperparameter optimization. Optimized hyperparameters were number of splits in range [1,1259], number of learners in range [10,500] and learning rate in range [0.001,1]. The optimization was done with Bayesian optimization with 100 iterations and the model development lasted around 3 minutes. Optimization found the best performing model with following hyperparameters: 7 number of splits, 10 number of learners and a learning rate of 0.1. Optimized AdaBoost decision tree showed following in-sample performance metrics:

Accuracy	74.4 %
Sensitivity	74.6 %
Specificity	74.1 %

False Negative Rate	25.5 %
ROC AUC	0.80

AdaBoost decision tree performed well using training data and do not show signs of overfitting. The built model was then exported to MATLAB workspace to use the model for predicting observations with testing data. MATLAB code for testing the model is introduced in Appendix 3. The prediction performance with testing data is introduced with the results of other models in chapters 5.5. and 5.6.

5.4 Random Forest bagged decision tree

MATLAB's classification learner was also used for building the Random Forest bagged decision tree and used the Random Forest algorithm Breiman (2001) introduced and uses "fitcensemble" function of MATLAB. The model was built by using 5-fold cross-validation and hyperparameter optimization. The optimized hyperparameters were number of splits in range [1,1259] and number of learners in range [10,500]. Optimization method was Bayesian with 100 iterations and the model building took around 2 minutes. The optimized hyperparameters were 6 splits and 64 learners. Bagged decision tree performed with the training data followingly:

Accuracy	74.6 %
Sensitivity	73.3 %
Specificity	75.9 %
False Negative Rate	26.0 %
ROC AUC	0.81

Bagged decision tree seems to fit the training data well and can classify the data adequately. Cross-validated model does not seem to have overfitting issues with the training data. The developed model was exported to MATLAB workspace after the model training to use the model for predicting observations with testing data. MATLAB code for testing the model is introduced in Appendix 4. The prediction performance with testing data is introduced with the results of other models in chapters 5.5. and 5.6.

5.5 Confusion matrices of the models with test data

The confusion matrices of the models (logistic regression, support vector machine, bagged tree and AdaBoost tree and evaluation statistics accuracy, misclassification, sensitivity, specificity, false negative rate and false positive rate derived from the matrices are presented in figure 6. The formulas for the evaluation statistics are introduced in chapter 3.2.1. Because the sample was balanced which means that the number of defaulted and non-defaulted enterprises was the same for the testing we can interpret that all of the models have prediction power as the accuracy of all models is above 70 %. Accuracy-wise the best performing model is AdaBoost boosted tree with classification accuracy of 72.0 % and the worst model based on accuracy is logistic regression. If we compare the sensitivity metric, SVM is the best performing model with sensitivity of 79.0 % and the worst models are bagged tree and logistic regression with sensitivity of 76.8 %. Bagged tree is the best performing model if the models are compared based on specificity and the worst model is SVM. Interpreting False Negative Rate and False Positive Rate is opposite compared to the other metrics because lowest value is the best. AdaBoost decision tree performs best when comparing false negative rates and logistic regression is again the worst performing model. Random Forest performs the best based on FPR and SVM is the worst performing model among these four. All statistics considered; the performance of the models is satisfactory as they all perform a lot better than random guessing. The performance differences between the models are very small and it is not possible to point out clear winner among the models based on these metrics.

Confusion matrix Logistic regression

	Predicted no default	Predicted default	
Actual no default	TN= 176	FP= 95	271
Actual default	FN= 63	TP= 208	271
	239	303	N=542

Accuracy	70.8 %
Misclassification	29.2 %
Sensitivity	76.8 %
Specificity	64.9 %
False Negative Rate	26.4 %
False Positive Rate	31.4 %

Confusion matrix SVM

	Predicted no default	Predicted default	
Actual no default	TN= 172	FP= 99	271
Actual default	FN= 57	TP= 214	271
	229	313	N=542

Accuracy	71.2 %
Misclassification	28.8 %
Sensitivity	79.0 %
Specificity	63.5 %
False Negative Rate	24.9 %
False Positive Rate	31.6 %

Confusion matrix RF bagged tree

	Predicted no default	Predicted default	
Actual no default	TN= 180	FP= 91	271
Actual default	FN= 63	TP= 208	271
	243	299	N=542

Accuracy	71.6 %
Misclassification	28.4 %
Sensitivity	76.8 %
Specificity	66.4 %
False Negative Rate	25.9 %
False Positive Rate	30.4 %

Confusion matrix Adaboost boosted tree

	Predicted no default	Predicted default	
Actual no default	TN= 177	FP= 94	271
Actual default	FN= 58	TP= 213	271
	235	307	N=542

Accuracy	72.0 %
Misclassification	28.0 %
Sensitivity	78.6 %
Specificity	65.3 %
False Negative Rate	24.7 %
False Positive Rate	30.6 %

Figure 6. Confusion matrices of the models with test data.

5.6 Receiver Operating Characteristic curves with test data

The ROC curves of the models are presented in figure 6. All the curves are well above the random guessing threshold (AUC 0,5) which goes diagonally from the down-left corner to the top-right corner. The difference between the models in the ROC curves is quite small, the best performing model based on AUC is bagged tree with an AUC value of 0.79 and the worst performing models are SVM and LR with AUC value of 0.77. The AUC value of AdaBoost decision tree was 0.78. Barboza et al. (2017) found AUC values of around 0.9 with these models which indicates that their models were able to predict better.

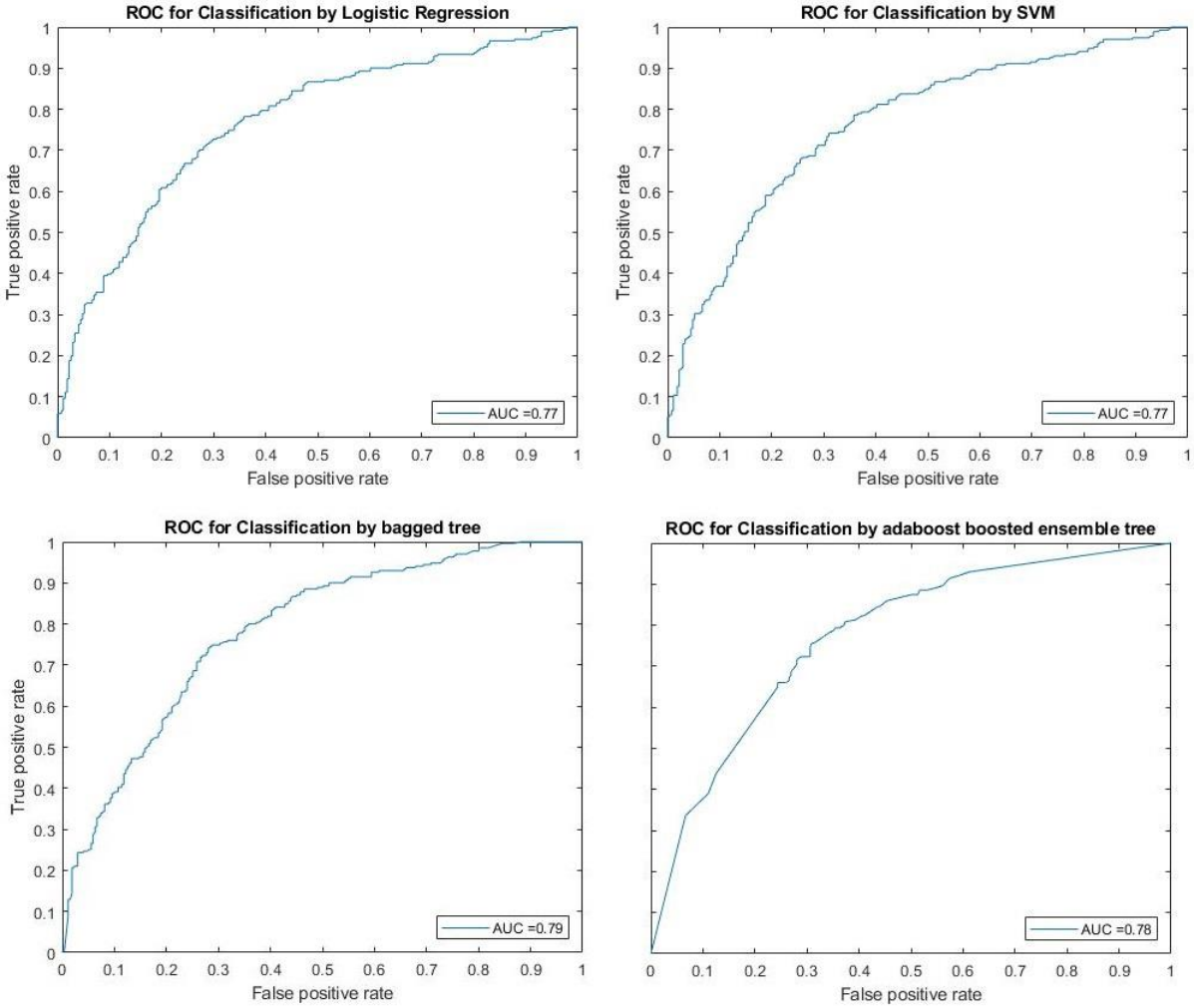

Figure 7. ROC curves of the models with test data.

5.7 Summary of results

The objective of the empirical part of was to fulfill the main objective of the study and to be able to answer the second research question and its sub-questions. The second set of research questions can be found below:

What model is the most suitable for predicting future financial distress of Swedish SME?

How the different methods compare to each other with the financial data of Swedish SMEs?

Can machine learning algorithms outperform the prevailing logistic regression?

The summary of evaluation metrics of the models and the ranking can be found in table 7 which shows the evaluation metrics of the models with the testing data (out of sample performance). The best performing score in each evaluation category is highlighted. AdaBoost decision tree performs best in FNR and Accuracy, SVM performs the best in sensitivity and Bagged decision tree in ROC AUC. The performance differences between the models are relatively small when compared to study Barboza et al. (2017) conducted when they evaluated different ML models with US-based companies. However, it is still possible to point out the two best performing models based on the most important evaluation metrics. The best performing models in this study are AdaBoost decision tree and SVM as they both score high on the most important metrics sensitivity and FNR. RF bagged decision tree performs a bit better than logistic regression when comparing the evaluation categories so based on this results ML models can outperform logistic regression.

Table 7. Evaluation metrics of the models

	Models			
	LR	SVM	RF bagged tree	AdaBoost tree
Sensitivity	76.8 %	**79.0 %**	76.8 %	78.6 %
Accuracy	70.8 %	71.2 %	71.6 %	**72.0 %**
ROC AUC	76.8 %	76.6 %	**78.5 %**	77.5 %
False Negative Rate	26.4 %	24.9 %	25.9 %	**24.7 %**

6 Conclusions

This chapter concludes the study and evaluates how the goals and objectives of the study were fulfilled. The main objective of the study was to find machine learning models for classifying Swedish small and medium-sized enterprises into a non-default and default classes. Three ML models were selected for the study which were Support Vector Machines, bagged decision tree and AdaBoost decision tree and their prediction performance was compared to logistic regression. The empirical results obtained showed surprisingly small differences in predictive power of the models when compared to prior studies on default prediction, but the ranking of the models was similar than in previous studies as ML models outperformed LR model.

6.1 Discussion on results

The results of the empirical study states clearly that it is possible to predict future default risk of Swedish SMEs using historical financial ratios of the companies which is in line with previous literature on subject. It was also found that the ML models performed better than LR when comparing the prediction evaluation metrics Although the results showed predicting power, accuracy of the models and other performance statistics have been better in prior studies, especially with ensemble decision trees and support vector machines. Barboza et al. (2017) were able to have over 80 % accuracies with SVMs and ensemble decision with out-of-sample predictions. In this study, the accuracy statistics of the models were in range 70-72 % in out-of-sample predictions. With accuracy rations around 70 %, I could state that it would not be reasonable to use this kind of model for automatic business decision-making for e.g. credit institutions but use these kinds of models for baseline of riskiness for customers and enhance the predictions with expert evaluations. The model was built for binary classification in this case, but e.g. credit score in range [1,100] could provide more information for the users of the model in business context.

Although the results between the models in the study did not differ as much as in some previous studies, results show that the ML models are able to perform better with predicting default risk of Swedish SMEs. LR lost in all evaluation metrics for all the ML models except in ROC AUC where it was the third out of four. This is an important

observation because logistic regression is still the most used method in the industry. It is not justified to stop using LR for prediction modeling based on these results, but ML models should be used at least for comparing or enhancing the prevailing models. The model development phase was quick to conduct for all the models except SVM, which took significantly longer time in hyperparameter optimization phase. The hyperparameter optimization for tree-based methods took around one minute to run with 100 iterations when the SVM hyperparameter optimization took around 45 minutes. This can limit the optimization of SVM models when using a home computer.

6.2 Limitations of this study

The limitations of this study originate mainly from the objectives and data available for the study. As the objective was to find suitable ML models and compare their prediction performance in general level with Swedish SMEs, it was not possible to go into every detail of each model. I chose to approach the problem in this way as I did not find any previous research on comparing the performance of ML models in default prediction of SMEs and there has been limited amount of studies focused on evaluating credit risk of Swedish companies. As the objectives of the study were on relatively high level, all the details of the models could not be covered which leads to limitations in the study. The most important limitations of the study in my opinion are introduced in next paragraphs.

The feature selection of the study was done by researchers' selection on variables based on previous studies which might have affected the prediction performance. As stated in previous studies, the methodological feature selection improves the prediction performance of the models. The main reason for not conducting a methodological feature selection was that the financial information of defaulted companies was quite limited which led to a small number of variables in the initial dataset. Also, using the financial ratios in different categories have proved to be efficient in prior studies.

The second limitation is also linked to the data used for the study. As the dataset contained only 1-year data of the enterprises, averages or any other time-based variables were not available for this research (e.g. growth rates of some ratios etc.). Also, quantitative, or macroeconomic variables were not used for building the models in this study

which have shown capability to improve performance in some studies. I decided to use similar approach to Altman (1968), Barboza et al. (2017) and Yoshino and Hesary (2015) who based their default prediction studies on financial ratios only.

6.3 Future research

As the objectives of this research were in general level and the focus of the study was to compare different methods, there is various ways to extend the knowledge in this subject in future research. I developed some ideas for future research during the process based on the limitations of this study and the current literature in the subject.

The evaluation performance could be enhanced by using more variables in different categories and conducting a methodological feature selection for the variables. Some researchers have had success when adding macroeconomic variables (e.g. interest rates), firm-specific quantitative variables (e.g. no of employees) or firm-specific growth rates (e.g. change in sales in last 3 years). Selecting one of the ML models and adding these variables into the study could enhance the model's prediction performance. One interesting direction for future research could be also so-called hybrid models which are not yet that much inspected in default prediction problems. Hybrid modelling means that the final model is built by using several (two or more) algorithms for the model development. A possible hybrid model could be one where unsupervised machine learning model self-organizing map would be used for clustering the sample to add new variables based on the clustering results. Then a supervised method could be used with the added clustering variables and financial ratio variables.

References

Adnan Aziz, M., & Dar, H.A. (2006). Predicting corporate bankruptcy: where we stand? Corporate Governance, Vol. 6 No. 1, 18-33.

Alpaydin, E. (2014). Introduction to Machine Learning (2014). Vol Third edition. The MIT Press.

Altman, E. I & Sabato, G. (2007). Modelling Credit Risk for SMEs (2007): Evidence from the U.S. Market. Abacus, volume 43 No. 3.

Altman, E. I (1968). Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy, Journal of Finance, Vol. 23, No. 4.

Barboza, F., Kimura, H., & Altman, E. (2017). Machine learning models and bankruptcy prediction. Expert Systems with Applications, 83, 405-417.

Beaver, W.H. (1966). Financial Ratios as Predictors of Failure. Journal of Accounting Research, 4, 71-111.

Behr, A., & Weinblat, J. (2017). Default Patterns in Seven EU Countries: A Random Forest Approach. International Journal of the Economics of Business, 24(2), 181-222.

Boyle, B., (2011). Support Vector Machines Data Analysis, Machine Learning and Applications. New York: Nova Science Publishers.

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

Bureau van Dijk (2020). Amadeus - Analyse major databases from European sources.

Callaghan, J., Murphy, A., & Qian, H. (2015). Third International Conference on Credit Analysis and Risk Management. Cambridge Scholars Publishing.

Ciampi, F., & Gordini, N. (2013). Small Enterprise Default Prediction Modeling through Artificial Neural Networks: An Empirical Analysis of Italian Small Enterprises. Journal of Small Business Management, 51(1), 23-45.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297.

Cultrera, L., & Brédart, X. (2016). Bankruptcy prediction: the case of Belgian SMEs. Review of Accounting and Finance, 15(1), 101-119.

do Prado, J., de Castro Alcântara, V., de Melo Carvalho, F., Vieira, K., Machado, L., & Tonelli, D. (2016). Multivariate analysis of credit risk and bankruptcy research data: a bibliometric study involving different knowledge fields (1968–2014). Scientometrics, 106(3), 1007-1029.

European Commission (2020). 2019 SBA Fact Sheet Sweden. https://ec.europa.eu/docsroom/documents/38662/attachments/28/translations/en/renditions/native

Feldman, M., & Libman, A. (2007). Crash Course in Accounting and Financial Statement Analysis: Vol. 2nd ed. Wiley.

Förtagarna (2013). Välfärdsskaparna – en analys av småföretagens betydelse för kommunernas skatteintäkter: www.foretagarna.se/Opinion/Rapporter/2013/Valfardsskaparna/

Freund Y., & Schapire R., (1996). Experiments with a New Boosting Algorithm. Machine Learning: Proceedings of the Thirteenth International Conference.

Grunert, J., Norden L., & Weber, M. (2005). The Role of Non-Financial Factors in Internal Credit Ratings. Journal of banking & finance 29.2, 509-531.

Heo, J., & Yang, J., (2014). AdaBoost based bankruptcy forecasting of Korean construction companies. Applied Soft Computing, 24, 494-499.

Horrigan, J. A Short History of Financial Ratio Analysis (1968). The Accounting review 43.2, 284-294.

Hutter, F., Kotthoff, L., & Vanschoren, J. (2019). Automated Machine Learning Methods, Systems, Challenges. Springer International Publishing.

Japkowicz, N., & Shah, M. (2011). Evaluating learning algorithms a classification perspective. Cambridge University Press.

Joshi, A. (2020). Machine learning and artificial intelligence (1st ed. 2020.). Springer.

Káčer, M., Ochotnický, P. & Alexy M. (2019). The Altman's Revised Z'-Score Model, Non-Financial Information and Macroeconomic Variables: Case of Slovak SMEs. Ekonomický časopis 67.4, 335-366.

Kim, H. S., & Sohn, S. Y. (2010). Support vector machines for default prediction of SMEs based on technology credit. European Journal of Operational Research, 201(3), 838-846.

Kim, S., & Upneja, A., (2014). Predicting restaurant financial distress using decision tree and AdaBoosted decision tree models. Economic Modelling, 36, 354-362.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Ijcai, 14(2), 1137-1143.

Kotu, V., & Deshpande, B. (2014). Predictive Analytics and Data Mining: Concepts and Practice with RapidMiner. Morgan Kaufmann.

Kubat M. (2017). Performance Evaluation. In: An Introduction to Machine Learning. Springer, Cham.

Li, K., Niskanen, J., Kolehmainen, M., & Niskanen, M. (2016). Financial innovation: Credit default hybrid model for SME lending. Expert Systems with Applications, 61, 343-355.

Martin, L., Sharma, S. & Maddulety, K. (2019). Machine Learning in Banking Risk Management: A Literature Review. Risks. 2019; 7(1):29.

MATLAB 2020a (2020). Classification Learner. https://se.mathworks.com/help/stats/classificationlearner-app.html

MATLAB 2020b (2020). Hyperparameter Optimization in Classification Learner App. https://se.mathworks.com/help/stats/hyperparameter-optimization-in-classification-learner-app.html

McKinsey (2020). Derisking Machine Learning and Artificial Intelligence. https://www.mckinsey.com/business-functions/risk/our-insights/derisking-machine-learning-and-artificial-intelligence

OECD (2019). Financing SMEs and Entrepreneurs 2019: An OECD Scoreboard, OECD Publishing, Paris.

OECD (2020). (SMALL AND MEDIUM-SIZED ENTERPRISES (SMES). https://stats.oecd.org/glossary/detail.asp?ID=3123

Osborne, J. (2015). Best Practices in Logistic Regression. Los Angeles: SAGE.

Ribeiro, B., Silva, C., Chen, N., Vieira, A., & Carvalho das Neves, J. (2012). Enhanced default risk models with SVM. Expert Systems with Applications, 39(11), 10140-10152.

Shi, Y., & Li, X. (2019). An overview of bankruptcy prediction models for corporate firms: A Systematic literature review. Intangible Capital, 15(2), 114–127.

Sirirattanaphonkun, W., & Pattarathammas, S. (2012). Default Prediction for Small-Medium Enterprises in Emerging Market: Evidence from Thailand. Seoul Journal of Business, 18(2), 25-54.

Statistics Sweden (2020). Anställda drabbade av konkurser efter region, näringsgren SNI 2007 och företagsform. Månad 2009M01 - 2020M09. https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__NV__NV1401/KonkurserAnst07/

Van Liebergen, B. (2017). Machine Learning: A Revolution in Risk Management and Compliance? Journal of Financial Transformation 45, 60-67.

White, L. (2010). Markets: The Credit Rating Agencies. The Journal of economic perspectives 24.2, 211-226.

Yeh, C., Chi, D., & Lin, Y., (2014). Going-concern prediction using hybrid random forests and rough set approach. Information Sciences, 254, 98-110.

Yoshino, N., & Hesary, F. (2015). Analysis of Credit Ratings for Small and Medium-Sized Enterprises: Evidence from Asia. Asian Development Review; Manila Vol. 32, Iss. 2, (Sep 2015), 18-37.

Appendix

Appendix 1. Matlab code for logistic regression

```
clc
clear all
close all

format bank

%Loading the sampledata

Trainset= readtable('sweden_train_set.xlsx');
Testset= readtable('sweden_test_set.xlsx');

Xtrain=Trainset(:,3:7);

Ytrain=Trainset(:,8);

Xtest=Testset(:,3:7);

%Fit a LR model with 5-fold cross-validation and bayesian optimization.
%Alle the confusion matrix and roc auc statistics for the model development
%are obtained from the classification learner application

Ytest=Testset(:,8);
Ydoubletest=table2array(Ytest);
Ydoubletrain=table2array(Ytrain);


%Loading the fitted model from the folder
load 'trainedLRmodel.mat'
%Making the predictions by using the test set x variables
Yfit=trainedModel.predictFcn(Xtest);
%Extracting the glm model trom trainedLRmodel
lrmdl=trainedModel.GeneralizedLinearModel;
%Using probability estimates from the logistic regression model as scores
scores = predict(lrmdl,Xtest);
%create confusion matrix
CLR= confusionmat(Ydoubletest,Yfit);
%create confusion chart
confusionchart(CLR)

%Create ROC curve
[X,Y,T,AUC] = perfcurve(Ydoubletest,scores,'1');
AUC=round(AUC,3,'significant')
plot(X,Y)
xlabel('False positive rate')
ylabel('True positive rate')
title('ROC for Classification by Logistic Regression')
legend(strcat('AUC = ', num2str(AUC)),'Location','southeast')
```

Appendix 2. Matlab code for SVM

```
clc
clear all
close all

%Loading the sampledata

Trainset= readtable('sweden_train_set.xlsx');
Testset= readtable('sweden_test_set.xlsx');

Xtrain=Trainset(:,3:7);

Ytrain=Trainset(:,8);

Xtest=Testset(:,3:7);

%Fit a SVM model with 5-fold cross-validation and bayesian optimization.
%Alle the confusion matrix and roc auc statistics for the model development
%are obtained from the classification learner application

Ytest=Testset(:,8);
Ydoubletest=table2array(Ytest);
Ydoubletrain=table2array(Ytrain);

%Load the fitted and optimized SVM model from the folder

load 'svmlinear.mat'
%Making the predictions by using the test set x variables
Yfit=SVMlinear.predictFcn(Xtest);

SVMmdl=SVMlinear.ClassificationSVM;
%Using probability estimates from the SVM model as scores
[label,score] = predict(SVMmdl,Xtest);
%create confusion matrix
CLR= confusionmat(Ydoubletest,Yfit);
%create confusion chart
confusionchart(CLR)

%Create ROC curve
[X,Y,T,AUC] = perfcurve(Ydoubletest,score(:,2),'1');
AUC=round(AUC,3,'significant')
plot(X,Y)
xlabel('False positive rate')
ylabel('True positive rate')
title('ROC for Classification by SVM')
legend(strcat('AUC = ', num2str(AUC)),'Location','southeast')
```

Appendix 3. Matlab code for AdaBoost decision tree

```
clc
clear all
close all
```

```
%Loading the sampledata

Trainset= readtable('sweden_train_set.xlsx');
Testset= readtable('sweden_test_set.xlsx');

Xtrain=Trainset(:,3:7);

Ytrain=Trainset(:,8);

Xtest=Testset(:,3:7);

%Fit a AdaBoost decision tree model with 5-fold cross-validation and bayes-
ian optimization.
%Alle the confusion matrix and roc auc statistics for the model development
%are obtained from the classification learner application

Ytest=Testset(:,8);
Ydoubletest=table2array(Ytest);
Ydoubletrain=table2array(Ytrain);

%Load the fitted and optimized Random Forest model from the folder

load 'adaboost.mat'
%Making the predictions by using the test set x variables
Yfit=adaboost.predictFcn(Xtest);

adaboostmdl=adaboost.ClassificationEnsemble;
%Using probability estimates from the SVM model as scores
[label,score] = predict(adaboostmdl,Xtest);
%create confusion matrix
CLR= confusionmat(Ydoubletest,Yfit);
%create confusion chart
confusionchart(CLR)

%Create ROC curve
[X,Y,T,AUC] = perfcurve(Ydoubletest,score(:,2),'1');
AUC=round(AUC,3,'significant')
plot(X,Y)
xlabel('False positive rate')
ylabel('True positive rate')
title('ROC for Classification by adaboost boosted ensemble tree')
legend(strcat('AUC = ', num2str(AUC)),'Location','southeast')
```

Appendix 4. Matlab code for Random Forest bagged decision tree

```
clc
clear all
close all

%Loading the sampledata

Trainset= readtable('sweden_train_set.xlsx');
```

```
Testset= readtable('sweden_test_set.xlsx');

Xtrain=Trainset(:,3:7);

Ytrain=Trainset(:,8);

Xtest=Testset(:,3:7);

%Fit a bagged decision tree (Random Forest) model with 5-fold cross-valida-
tion and bayesian optimization.
%Alle the confusion matrix and roc auc statistics for the model development
%are obtained from the classification learner application
Ytest=Testset(:,8);
Ydoubletest=table2array(Ytest);
Ydoubletrain=table2array(Ytrain);

%Load the fitted and optimized Random Forest model from the folder

load 'baggedtree.mat'
%Making the predictions by using the test set x variables
Yfit=baggedtree.predictFcn(Xtest);

BAGGEDmdl=baggedtree.ClassificationEnsemble;
%Using probability estimates from the SVM model as scores
[label,score] = predict(BAGGEDmdl,Xtest);
%create confusion matrix
CLR= confusionmat(Ydoubletest,Yfit);
%create confusion chart
confusionchart(CLR)

%Create ROC curve
[X,Y,T,AUC] = perfcurve(Ydoubletest,score(:,2),'1');
AUC=round(AUC,3,'significant')
plot(X,Y)
xlabel('False positive rate')
ylabel('True positive rate')
title('ROC for Classification by bagged tree')
legend(strcat('AUC = ', num2str(AUC)),'Location','southeast')
```