CRANFIELD UNIVERSITY

SWATI JADHAV

DATA MINING IN COMPUTATIONAL FINANCE

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING

PhD Thesis

PhD

Academic Year: 2013 - 2017

Supervisors: Dr. Karl Jenkins and Dr. Hongmei He

December 2017

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING

PhD Thesis

Academic Year 2013 - 2017

SWATI JADHAV

DATA MINING IN COMPUTATIONAL FINANCE

Supervisors: Dr. Karl Jenkins and Dr. Hongmei He

December 2017

This thesis is submitted in partial fulfilment of the requirements for the degree of PhD

ABSTRACT

Computational finance is a relatively new discipline whose birth can be traced back to early 1950s. Its major objective is to develop and study practical models focusing on techniques that apply directly to financial analyses. The large number of decisions and computationally intensive problems involved in this discipline make data mining and machine learning models an integral part to improve, automate, and expand the current processes.

One of the objectives of this research is to present a state-of-the-art of the data mining and machine learning techniques applied in the core areas of computational finance. Next, detailed analysis of public and private finance datasets is performed in an attempt to find interesting facts from data and draw conclusions regarding the usefulness of features within the datasets.

Credit risk evaluation is one of the crucial modern concerns in this field. Credit scoring is essentially a classification problem where models are built using the information about past applicants to categorise new applicants as 'creditworthy' or 'non-creditworthy'. We appraise the performance of a few classical machine learning algorithms for the problem of credit scoring.

Typically, credit scoring databases are large and characterised by redundant and irrelevant features, making the classification task more computationally-demanding. Feature selection is the process of selecting an optimal subset of relevant features. We propose an improved information-gain directed wrapper feature selection method using genetic algorithms and successfully evaluate its effectiveness against baseline and generic wrapper methods using three benchmark datasets.

One of the tasks of financial analysts is to estimate a company's worth. In the last piece of work, this study predicts the growth rate for earnings of companies using three machine learning techniques. We employed the technique of lagged features, which allowed varying amounts of recent history to be brought into the prediction task, and transformed the time series forecasting problem into a

supervised learning problem. This work was applied on a private time series dataset.

Keywords:

Computational finance, Data mining, Data analysis, Machine learning, Credit scoring, Information gain, Wrapper, Feature selection, Earnings Per Share

Dedicated to Pappa and Baba

ACKNOWLEDGEMENTS

It is my pleasure to thank the people whose contribution and support made this work possible and improved the quality of my research.

I would like to express my sincere gratitude and thanks to my supervisor Dr. Karl Jenkins for the support he provided and the freedom he granted me to explore my potential and pursue my research goals.

My heartfelt gratitude and appreciation to Dr. Hongmei He for co-supervising my PhD work and for her constant stream of constructive feedback and solution-oriented approach. She was always available for advice and support, and I had the opportunity to be part of the many interesting and stimulating discussions we had.

I take this opportunity to express my deepest gratitude to the Doctoral committee members for critically reviewing my work from time to time and directing me with their precious review to improve this work.

On a personal note, a big thank you to my husband for his constant support and encouragement, thank you for putting up with me and believing in me more than I believed in myself. And my children, who saw less of me during this period, and making sure I am not stressed all the time!

Finally, thank you to my mother and my mother-in-law for being so proud of me and for their constant questionnaire about the progress of my work! It kept me on track during the whole process!

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

ANFIS	Adaptive Neural Fuzzy Inference Systems
ANN	Artificial Neural Network
ARCH	Auto-Regressive Conditional Heteroskedasticity
ARIMA	Auto-Regressive Integrated Moving Average
AUC	Area Under Curve
CART	Classification And Regression Tree
DM	Data Mining
DT	Decision Tree
EPS	Earnings Per Share
FPR	False Positive Rate
FS	Feature selection
GA	Genetic Algorithm
GAW	Genetic Algorithm Wrapper
IG	Information Gain
IGDFS	Information Gain Directed Feature Selection algorithm
KNN	K-Nearest Neighbour
LDA	Linear discriminant analysis
LR	Linear Regression
MARS	Multivariate Adaptive Regression Splines
ML	Machine Learning
MLP	MultiLayer Perceptron
NB	Naïve Bayes
NN	Neural Network
PCA	Principal Component Analysis
PE	Price-Earnings ratio
PSO	Particle Swarm Optimization
QP	Quadratic Programming
RBF	Radial Basis Function
RF	Random Forest
RMSE	Root Mean Square Error
ROC	Receiver Operating Characteristics

SOM	Self-Organizing Maps
SVM	Support Vector Machine
SVR	Support Vector regression
TPR	True Positive Rate
WNN	Wavelet Neural Network

1 INTRODUCTION

This chapter introduces the research topic. The motivations behind the research are discussed along with a brief description of its significance. The aims and objectives of the thesis are then specified. Finally, the chapter concludes with a high-level summary of the organisation of this thesis.

1.1 Background and Motivation

To position the contribution of this thesis, we begin with a high-level overview of Data mining (DM), Computational finance, Credit scoring, Feature selection and the application of prediction techniques of Machine Learning (ML) in finance. In this section, the research motivations, and the research questions that are addressed in this thesis are articulated. The reader is offered a clear and thorough discussion of the research problems in preparation for forthcoming chapters.

In 2006, Clive Humby declared at the Association of National Advertisers' Senior marketer's summit that Data is the new oil and this might not be far from the truth. Data is just like crude, it is valuable only when refined and broken down to create valuable entities to make it profitable. Tables of data containing numbers are not helpful on their own, but the knowledge abstracted from data is indispensable to companies, research, governments and other interest groups.

Data mining, the process of extracting information from data has become an important research topic over the last decades, confirmed by numerous publications and conferences in the field. It is the process of applying machine learning techniques to "databases". Data mining is a cross-disciplinary field focusing on discovering properties of data. Machine Learning is a sub-field of data science that focuses on designing algorithms that train on a dataset to make predictions. The information revealed by data mining processes is limitless. If used correctly, the knowledge from data could transform business and drive the creation of business value. The real power of data lies in the application of

analytical tools to extract useful knowledge and quantify the factors that potentially help explain the underlying process.

Machine learning algorithms are designed to learn from large amounts of historical data and then make a forecast. Credit scoring for loans from retail banks is an example where machine learning techniques along with other data-mining algorithms perform better.

Computational finance is a branch of applied computer science that deals with problems of practical interest in finance [1]. It emphasises practical numerical methods rather than mathematical proofs and focuses on techniques that apply directly to financial analyses [2]. Its main objective is the development, analysis and implementation of solutions for the computational problems arising in the finance industry. Many financial models require extensive computations for their analysis. Efficient numerical methods have thus become an important aspect of computational finance.

For a researcher, computational finance is essentially analysis, critical evaluation, and application of methods of computation to practical finance problems. Some of the key areas it addresses are exploratory data analysis including visualisation, model building and evaluation, finding the best model to explore correlation among variables, computation and management of market and credit risk and time series prediction.

Since decades, the process of financial decision-making has been automated [3]. Recent focus is on continuously assessing the current methods and finding new methods in order to gain an advantage, increase returns, and customise products. Advances in computing hardware and algorithms give rise to unprecedented opportunities in computational finance research. We are in a much better position to explore ways to forecast and to gain better understanding of the market [4].

The two major research problems in accounting and finance domain are bankruptcy prediction and credit scoring [5]. Since the 1990s, machine-learning techniques have been studied extensively as tools for prediction and modelling

in the field of finance. The literature review chapter in this thesis reviews 179 related scientific articles during the period from 2010 to 2015, focusing on development of state-of-the-art data mining and machine learning techniques in the subareas of finance such as credit rating, loan prediction, money laundering, stocks prediction and time series prediction.

Credit scoring, which is a process of determining if a credit applicant is creditworthy or not, has many benefits for lenders and borrowers equally. The models built for credit scoring aim to provide an objective analysis of a consumer's creditworthiness. Due to the availability of models, credit providers can afford to focus only on the information related to credit risk. This eliminates the personal subjectivity of a human in decision making and reduces the need for human intervention on credit evaluation and the cost of delivering credit [6]-[7]. But the most important benefit of credit scoring is that it helps increase the speed and consistency of the loan application process and allows the automation of the lending process [7]. With the help of credit scores, financial institutions are able to make faster, better, and higher quality decisions [8].

When the classes to which the applicants belong to are known a priori, the problem of credit scoring is called classification. Historically, several traditional statistical techniques have been used in the construction of credit scoring models. Some of the data mining and machine learning techniques that have been previously used, but rather infrequently, to construct credit scoring models include genetic algorithm, k-nearest neighbour, linear programming, and expert systems, and more recently, the decision tree, neural networks. These methods are important data mining techniques for predictive modelling.

In this work, we aim to investigate three classical ML classifiers for the problem of credit scoring, and test them on three benchmark datasets.

In most real-world credit scoring applications, the data is highly dimensional since credit institutions want to capture as much information about credit applicants as is possible to help make the credit approval decision. Applied finance researchers and practitioners remain concerned with prediction accuracy when building credit

modelling systems [9]. Feature selection is a term commonly used in machine learning to describe a set of methods to reduce a dataset to a convenient size for processing and investigation. It is a significant step in machine learning and data mining since an optimum feature subset can give better results than using full feature-set. In finance industry, this process involves the choice of appropriate features based on their relevance to the study when building a credit model [10]. When the dimensionality of the dataset affects the credit models, finding the optimum subset of predictive features becomes an important problem, which is addressed in this work.

The motivation for feature selection is:

- A credit scoring model approximates the underlying function between the input and the output, hence selecting the input features with highest effect on the output is reasonable and important. This will keep the size of the model small and reduce the computational cost.
- One of the requirements of the training phase of a supervised ML classifier is having labelled dataset. Large amounts of training data are a requirement in order to achieve satisfactory generalization from the models. With high dimensionality, producing labelled training examples becomes very expensive, especially if it is done manually [11]Also, training time becomes unreasonably long. As the size of the model grows, the solution size affects the memory requirements during both training and testing phases especially for large real-time problems.

In order to find the best subset of features to be evaluated, the exhaustive search is the ideal approach. However, since such method is computationally expensive and time consuming, three main classes of feature selection are identified in the literature: Filter, wrapper and hybrid feature selection methods. Usually, filter methods choose the best features by using some informative measure. Various filtering methods and their modifications are proposed in the literature leading to the ‘selection trouble’ when dealing with a specific feature selection task [12].

Filter feature selection does not take into account the properties of the classifier, as it performs usually statistical tests on variables. The wrapper technique takes into consideration the classifier proprieties. Using a single classifier in the wrapper evaluation process may influence the final selection result because each particular classifier has its own specificity and nature [12]. A possible solution could be to use an array of classifiers and compare their outcomes.

This work proposes an improved feature selection algorithm using information gain and genetic algorithm wrapper. The performance is evaluated empirically against baseline ML methods and generic wrapper methods of feature selection. Three techniques from machine learning are used in the wrapper technique namely, Support vector machines, Naïve Bayes and k-nearest neighbours.

A company's financial analysis involves evaluation of financial ratios rather than just amounts. Ratios enables one to examine the relationships between seemingly unrelated financial items and thus gain useful information about condition of a company. Financial ratios provide a wealth of information that cannot be obtained anywhere else. Evaluation of stocks of a company to buy or sell is an important decision to be made by the investors of a company. Nowadays, when huge amounts of data are made available with the advent of technology, this decision does not become any easier without the help of some help from analytics.

Earnings Per Share (EPS) is considered as one of the most important profitability metrics of a company. It represents the returns delivered by the company for each outstanding share of common stock. It is a major indicator for investors to purchase stocks. Price-Earnings (PE) ratio is obtained by dividing the stock price by EPS. The EPS used here can be current or future earnings. EPS over past quarters as well as "forward" forecasted quarters are most frequently used in the calculation of PE ratio of a company. Comparison of a stock's current PE with those of its competitors or with its own average multiple over three to ten years gives useful information about hopeful future profits, investment in the company and also if a possible bargain has happened. Investment into a stock depends on

the current PE ratio: Is it too high or low compared with the PE ratio of the stock's peers, industry or aggregate market?

In the last piece of work, a private dataset provided by FactSet Research Systems Inc, London [13] is used for an application involving analyses with financial ratios. We perform a detailed analysis of the data and propose three regression models to predict EPS: (a) Statistical Regression Model using Linear Regression; (b) Neural network regression using Multilayer Perceptron and (c) Neural network regression using Radial Basis Function. For construction of these models, quarterly EPS data are employed.

1.2 Objectives of the thesis

This work focuses on following objectives in the area of data mining in computational finance:

1. To perform a detailed data analysis for three benchmark public credit dataset and one real dataset. Raw data are simple facts and figures which are unprocessed. When it is organised, structured and interpreted, it becomes 'Information'. We aim to discover the 'Information' from the 'Data' to make it meaningful and useful to the user of the data.
2. To assist the research community by providing a comprehensive literature review of the state-of-the-art in the field of application of ML techniques in computational finance. We consider five sub-areas of computational finance for the literature review.
3. To develop an improved feature selection algorithm using wrapper techniques for credit scoring application with a high-dimensional dataset. We employ information gain, wrapper with genetic algorithm, three machine learning techniques for this study.
4. To develop a strategy to predict Earnings-per-share ratios for companies traded in stock market using regression techniques. We employ three regression techniques for this work.

1.3 Structure of the thesis

This thesis is divided into eight chapters, and the references.

Chapter 1 gives an introduction and an overview of the various topics that are addressed in this thesis. It also discusses the research motivation, research objectives.

Chapter 2 presents the research methodology developed to achieve the research aims and objectives. This chapter covers the research framework, research design, and various phases of the methodology.

Chapter 3 begins by the discussion of data mining as a field and its relevance in the field of computational finance. This is followed by a comprehensive review of the literature in the areas of Credit rating, Loan prediction, Money laundering, Stocks prediction and Time series. A view on the developments made in the recent years in the field of computational finance is presented. The essential background and theoretical knowledge gained while researching in these areas is presented in this chapter.

Chapter 5 presents application and evaluation of a few classical machine learning techniques to the problem of credit scoring. The methods applied are discussed first followed by the experimental discussion and comparison of the techniques.

Chapter 6 starts with a systematic analysis of existing work done on feature selection, the methodology of the work, an overview of the techniques employed. The problem of feature selection using genetic algorithm as a wrapper is discussed. The new algorithm for feature selection is proposed. This chapter ends with the experimental description and results evaluation.

Chapter 7 describes the work in the field of computational finance to predict the earnings per share for industry using regression techniques. We discuss the techniques employed, problem formulation, followed by the experimental description, results and evaluation.

The concluding remarks, with discussions on the original elements of the thesis are presented in chapter 8. The findings of this research are discussed in detail. This chapter concludes with the contributions of this PhD thesis, research limitations and directions for future work.

1.4 Publications

Following research articles were published as part of this PhD work:

1) Jadhav Swati, Hongmei He, and Karl Jenkins. "Prediction of Earnings per Share for Industry." Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), 2015 7th International Joint Conference on. Vol. 1. IEEE, 2015.

2) Jadhav Swati, Hongmei He, and Karl W. Jenkins. "An Academic Review: Applications of Data Mining Techniques in Finance Industry." International Journal of Soft Computing and Artificial Intelligence 4.1 (2017): 79-95.

3) Jadhav Swati, Hongmei He, and Karl Jenkins. “Information Gain Directed Genetic Algorithm Wrapper Feature selection for Credit Rating, submitted to Applied Soft Computing in July 2017.

1.5 Summary

This chapter has covered the introduction to the study. The motivation behind the work and objectives set forth in the research were detailed. The chapter also presented the structure and organisation of the thesis.

2 RESEARCH METHODOLOGY

2.1 Introduction

This chapter discusses formulation of the research design and methodology adopted to achieve the goals of the study. Scope of the research is discussed followed by an overview of the research framework. A breakdown of the framework shows how the processes are linked and presents the outputs expected from each stage of the research. The research design describes the problems addressed in this study.

As the machine learning algorithms become more sophisticated and advanced, they are applied in a wider range of fields. Compared with traditional analysis approach taken in finance industry, machine learning allows computers to take a more free-form approach, aiming to identify predictable patterns in data without being given specific guidance about what underlying relationships may look like. This is a significant departure from traditional financial analysis, and the industry has started to see its potential to add value.

Quantitative research has typically been predicated on the discovery of linear relationships between input data (such as historical price movements, interest rates or company earnings) and future movements in asset prices. Machine learning algorithms are good at finding out more subtle, non-linear, relationships within data.

This study attempts to provide answers to following questions in a structured way:

(a) What are the current developments in computational finance? - Answering this question will help us understand where room for improvement is. The third chapter summarises the current state of work and knowledge in the computational finance domain based on the peer-reviewed literature with a special focus on credit rating, loan prediction, money laundering, stocks prediction and time series prediction.

(b) How ‘data analysis’, an important phase in ‘Knowledge Discovery in Databases’ could be applied to three public and especially one private datasets to gain understanding of the datasets?

(c) Which machine learning algorithms are effective for credit scoring and how do they perform in credit risk prediction? – Due to the huge number of available models, selecting a few models to investigate in more detail becomes important. This question will be answered by evaluating model performance on data. The fourth chapter is focused on the applications of a few classical machine learning algorithms for the credit scoring problem: how to prepare the data set, how to measure and improve the performance, which parameters are important for the performance of a model, how to evaluate different models and choose the best one among all possibilities.

(d) How the problem of credit scoring can benefit from utilising the predictive power of individual features? – Individual features contribute towards the decision-making process of credit scoring. When the predictive power of individual features is low and there are a lot of features, feature selection is a promising approach. The technique is suitable for selecting a subset from a large number of features and help improve the prediction accuracy. Numerous studies are conducted in this field and we aim to develop a new technique and compare it with existing techniques.

(e) Can the analysis of the special events of the market such as earnings per share announcements predict stock market movement? - Machine learning techniques can provide reasonable market movement predictions, which could provide a valued reference for investors. The sixth chapter presents the application of machine-learning techniques to predict the earnings per share of some chosen companies being traded in the stock market.

2.2 Research Framework

Research framework depicts the implementation steps followed throughout the research. It is used as a guide for conducting focused study in the scope of the

research. Figure below shows an operational framework that will be followed in this study.

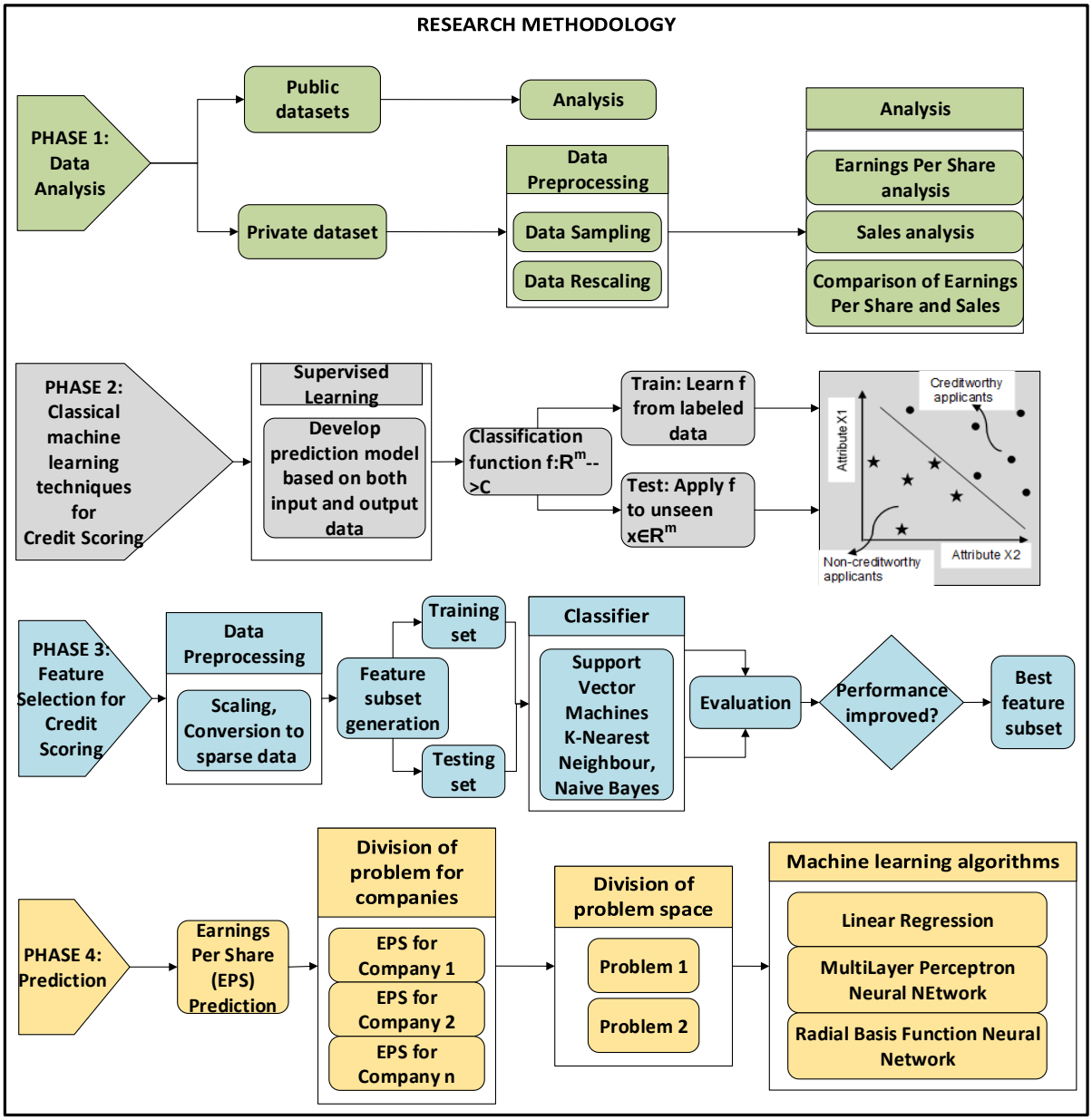

Figure 2-1: A schematic research framework of the study

The framework supports a three-phase approach. Phase 1 answers question b from previous section. Phase-2 answers question (c). Phase-3 is based on feature selection (question d). Phase-4 is aimed to evaluate three machine learning models for the problem of prediction of stock prices using EPS (question

e). These phases are depicted in the figure above and discussed in the next section.

2.3 Research Design

The research is conducted through four main phases. The following subsections describe each phase briefly.

2.3.1 Data Analysis (Phase-1)

Two types of datasets are analysed in the first phase of this work:

- Three public credit scoring datasets;
- One real industry dataset which is estimates of Earnings Per Share and Sales for companies being traded in the stock market.

For the industry dataset, the data is first transformed before analysing the correlation between variables. The results of this phase help analyse the datasets in detail. The public datasets are used in phase 2 and 3 and private dataset is used in phase 4 of the research.

2.3.2 Problem 1 (Phase-2)

Investigate a few classical Machine Learning techniques for the problem of Credit Scoring:

The typical business process followed by finance institutions for the provision of a loan service is: accept loan applications, evaluate the credit risk, make the decision on the granting of the loans, and supervise the repayment of principles and interests. This problem could be approached by applying machine learning by building models during the loan origination process.

Various statistical and computational methods can be used to construct objective models to differentiate the loan applicants as creditworthy or not, and consequentially to estimate the probability of their default. Machine learning refers to a set of algorithms specifically designed to tackle computationally intensive pattern-recognition problems in extremely large datasets. These techniques are ideally suited for consumer credit-risk analytics because of the

large sample sizes and the complexity of the possible relationships among consumer transactions and characteristics [14].

With problem solving at its centre, the machine learning landscape can be broadly divided into two areas: supervised machine learning and unsupervised machine learning. Supervised machine learning contains a set of methods for discovering the relationship between data when the problem has a defined target variable. For example, if the problem is to predict if a loan will default, the method used will be supervised, and the target will have a value of either yes or no based on whether the loan defaulted. This problem can be further defined as classification because the target in this example is categorical. If the target is a continuous variable (e.g. pounds lost on a defaulted loan) then the problem falls into the regression category.

In the origination process, the research population consists of all the applicants who want to apply for loans. By using the historical data of application records, the model could be trained to judge whether a new applicant is sufficiently reliable to be granted the loan if the characteristic indicators of the applicant have been provided, such as their income, marital status, age etc.

This challenge of credit scoring is well suited to be formulated as a supervised machine learning problem, where a learner algorithm is presented with input/output pairs from past data, in which the input data represent pre-identified attributes to be used to determine the output value.

In the supervising process, the trained model from previous stage is used to judge whether or not a new customer has a large probability of defaulting. This automated process offered by machine learning models is more time efficient and accurate compared with the traditional ways.

However, there are many machine learning algorithms available right now, and the question as to which one is the best has no definite answer since the performance of the algorithms is data and problem-sensitive. The general way to find an appropriate model for a single specific data set, or a type of data set, is to apply to it some widely-used and well-proven algorithms.

According to Vera et al. [15], “despite the intense study of credit scoring, there is no consensus on the most appropriate classification technique to use.” Some conflicts may occur when comparing the findings of different studies [16]. However, most methods applied in credit scoring have similar levels of performance [17]. For banks and financial institutions motivational reasons for preferring a certain method are the interpretability and the transparency of the method [18]. According to Vera et al. [15], “two aspects of methods for credit scoring are very important: that is the predictive performance, as well as the insights or interpretations that are revealed by the model.” Not all the ideas prove to be useful in practical credit scoring. Analysts are always seeking ways to extend the scope and improve the accuracy of credit scoring [19]. In an attempt to contribute in the area of credit scoring, the three public datasets are used and we choose three classical machine learning techniques of SVM, KNN and Naïve Bayes for this work. Chapter 4 investigates these methods in detail.

2.3.3 Problem 2 (Phase-3)

Propose a new method for feature selection (FS) based on Information gain, Genetic algorithm wrapper with three different classifiers in credit scoring tasks:

An important goal of the credit risk prediction is constructing the best classification model for a particular data set. There are a lot of irrelevant and redundant features in financial data in general and credit data in particular [20]. This makes the data noisy and unreliable, hence the accuracy of classification can be reduced leading to bad decisions. Feature selection reduces the dimension and the computational complexity of the problem and saves the cost of measuring non-selected features. Fewer features allow a credit analyst to focus on collecting only relevant and essential variables.

We employ FS to improve the accuracy of credit scoring models. The importance of features is accurately evaluated by the integrated information gain. Wrapper feature selection methods measure the goodness of the feature subset with the help of embedded machine learning algorithm. To measure the performance of the learning algorithm, learning accuracy measure is used.

We have introduced a new feature selection approach based on feature scoring. The proposed method first determines optimised model parameters. Next, each feature is ranked according to the metric of information gain. This method is independent of the classifier algorithms. The top k ranked features are propagated through the wrapper method of FS. The method tries to find a feature subset as small as possible while classifier hypothesis has high accuracy. We used genetic algorithm based meta-heuristic optimisation algorithm to improve the accuracy of classifier hypothesis. Chapter 6 details this work.

2.3.4 Problem 3 (Phase-4)

Employ machine learning models to predict the Earnings Per Share of market firms with estimates data:

One way to forecast the market movement is by analysing the special events of the market such as earnings announcements. Earnings announcement for a company is an official public statement of the company's profitability for a specific time period, typically a quarter or a year. Each company has its specific earnings announcement dates. Stock price of a company is affected by the earnings announcement event. Equity analysts usually predict the earnings per share (EPS) prior to the announcement date.

This study employs the techniques of Linear Regression, Multilayer Perceptron and RBF neural networks for prediction of EPS. Chapter 7 details the study.

2.4 Summary

This chapter detailed the methodology followed in this research work. First, we described the problems faced by the finance industry. Next, the research framework of the work was introduced, followed by the actual research design and the techniques investigated.

The research framework comprises three phases. In first phase, the credit scoring problem is investigated using few classical machine learning algorithms. Second phase of the work proposes a new method for feature selection on datasets based on genetic algorithm wrapper with three different classifiers in credit

scoring tasks. In the third phase, a few machine learning models are employed to predict the Earnings Per Share of market firms with estimates data.

3 LITERATURE REVIEW: APPLICATIONS OF DATA MINING TECHNIQUES IN FINANCE INDUSTRY

In this chapter, a review of the literature topics related to this PhD thesis will be given. This chapter reviews the current applications of Data Mining (DM) techniques to various finance tasks. We begin by looking at the current applications of DM techniques in five subdomains of finance area. The ideas already present in the literature will be explored with the aim to highlight potential knowledge gaps, which could be filled with further research.

3.1 Introduction

DM, also known as Knowledge Discovery, is to extract interesting nontrivial, implicit, previously unknown and potentially useful information or patterns from data in large databases [21]. Computational Finance is the application of computational techniques to finance. Financial mathematics, stochastic, numerical mathematics and scientific computing are combined to solve the real problems in finance industry. Computational Finance is important for the business world when it comes to corporate strategic planning by giving insights into what could happen in the future if a strategy is implemented, and predicting the risks associated with financial instruments.

The financial industry is a data-driven business, since the data generated is reliable and of high quality [22]. Big Data has become the asset of banks, as they are resources for analysing credit quality, monitor fraud and reduce customer churn. The DM techniques allow access to the right information at the right time and are used by the finance industry in various areas such as fraud detection, intelligent forecasting, credit rating, loan management, customer profiling, money laundering, marketing and prediction of price movements to name a few.

With passing time, various prediction techniques have been developed, and many of them have been employed in finance industry, especially after 1960s, following a triggered corporate financial distress [23]-[24]. Many computational finance problems can be mapped to DM problems, thus corresponding DM

technologies can be applied to solve these problems. The techniques for association, classification, clustering, regression problems in DM, which have been investigated extensively in the area of computational finance, included Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), Bayesian Classifier, Decision Trees (DTs), and Genetic Algorithms(GAs).

Research on DM in finance and the application of its outcomes is a relatively new research field. The aim of the present study is to provide a state-of-the-art review about current research efforts on applying DM in finance and accounting. This review introduces the reader to specific topics concerning research objectives and methods employed. In particular this study tries to address the following questions:

- What are the specific financial application areas to which DM methods have been extensively applied?
- What DM methods have been applied and to what extent. Do these methods outperform previous more traditional methods?
- Over what kind of data do the methods operate? Are sample sizes sufficiently large? What are the applied feature selection methods?
- What are the relative performance metrics considerations?

We review classic business issues in finance industry, such as Credit rating, Load prediction, Money laundering, Stocks prediction and Time series. Final section concludes the survey.

The next section describes in brief the data mining techniques most used in finance domain.

3.2 Data mining techniques

3.2.1 ANN

Neural networks were first proposed in 1944 by Warren McCullough and Walter Pitts. ANNs are a collection of interconnected processing nodes, imitating the

structure and functionality of a human neuron, which convert an input vector into some output. These neurons work together to develop their own solutions to problem. Each neuron accepts an input, applies a (often nonlinear) function to the input and then passes the output on to the next layer. In a most form of ANN, called as feed-forward ANNs, a neuron feeds its output to all the units on the next layer, without any feedback to the previous layer. The processing capacity of the network is stored in the linking weights, which are obtained by a process of learning from a set of training patterns. In the training process the neural network is presented with example data and then the network's internal weights are adjusted repeatedly to reduce the difference between the expected and actual output of the perceptron until the desired neural network response is obtained.

3.2.2 Decision Trees

Decision tree uses a tree-like model of decisions on the dataset to build classification or regression models. It splits the data into smaller and smaller subsets while an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes. The topmost node is the root node which corresponds to the best predictor. A decision node has two or more branches and a leaf node represents a classification or decision. Decision trees can work with both categorical and numerical data.

3.2.3 Regression techniques

Linear regression methods investigate the relationship between a response (dependent, output or target) variable and one or more predictors (independent or input) variables. It is most useful in forecasting, time series modelling and determining cause-effect relationships between variables. Type of regression depends on the number of independent variables, type of dependent variable and shape of the regression line. Linear Regression establishes a relationship between dependent variable and one or more independent variables using a best fit straight regression line. In multi-variable linear regression, a model is created for the relationship between multiple independent variables and a dependent

variable. The model remains linear as the output is a linear combination of the input variables.

In logistic regression, the outcome variable is dichotomous (a 0/1 outcome). The simple logistic regression model can be extended to two or more independent variables. It is widely used for classification problems. Other types of regression are Polynomial (for non-linearly separable data), Stepwise, Ridge (when high collinearity among the variables), Lasso (same as ridge except with a regularization term as an absolute value), ElasticNet (hybrid of Lasso and Ridge) regression.

3.2.4 Hybrid techniques

Hybrid data mining techniques are developed by integrating multiple more pre-existing techniques. The ability of individual technique is different, hence combining them may increase the performance of the resultant model while removing the weakness of a single model. Thus, the user may benefit from the advantages of two or more techniques. Such integrated models are difficult to formulate and implement than simple methods.

The data mining techniques of support vector machine, k- nearest neighbour and Naïve Bayes are explained in chapter 5.

3.3 Credit Rating

The word ‘credit’ means ‘buy now and pay later’; the word ‘scoring’ refers to ‘the use of a numerical tool to rank order cases according to some real or perceived quality in order to discriminate between them, and ensure objective and consistent decisions [25]. The process of modelling creditworthiness and used by financial institutions is referred to as ‘credit scoring’.

Credit evaluation is one of the vital processes in banks’ credit management decisions. The process performs collection, analysis and classification of different credit elements and variables, which determines the credit decisions.

In finance industry, credit rating is used to check credit worthiness of a person, an authority, corporations, non-profit organisations or even governments. Credit rating is expressed as a letter grade (such as 'AAA' and 'AA' for high credit quality and 'A' and 'BBB' for medium credit quality etc.). Whereas credit rating tells about creditworthiness of a business, corporation or government, credit score is expressed in numerical form and often used for individuals. Credit rating carries out credit evaluation, and assigns a score to credit report which represents credit worth of that entity. A credit score between 700 and 850 is considered as good. Figure 3-1 shows the credit scoring distribution for the general American public [26]. Nevertheless, both credit rating and score are used by creditors to assess a borrower's prospect of repaying a debt.

DM techniques are used to build credit scoring models, to help banks make decision of accepting or rejecting a client's credit. Business success of banking industry depends strongly on the evaluation of credit risk of potential debtors and it is an important part of financial risk management.

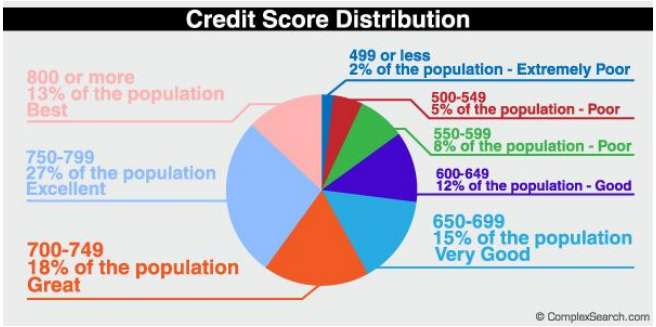

Figure 3-1: Credit Scoring Distribution

Hybrid methods, which combine different models to provide better performance than that can be gained by individual models, are becoming focus of lot of research work in finance industry. An integrated approach using sampling to calculate F-score and SVM was proposed in [27].This is one of the few studies focusing on reducing the computational time required for credit scoring. A novel hybrid approach for rectifying the data imbalance problem was proposed by employing k-Reverse Nearest Neighbourhood and One Class support vector machine (OCSVM) in tandem in [28]. Decision Tree (DT), Support Vector

Machine (SVM), Logistic Regression (LR), Probabilistic Neural Network (PNN), Group Method of Data Handling (GMDH), Multi-Layer Perceptron (MLP) were used for testing. In a similar study for credit scoring [29], four approaches, including conventional statistical Linear discriminant analysis (LDA) and Decision tree, were combined with support vector machine classifier for feature selection to retain information. It was concluded that the hybrid credit scoring approach is mostly robust and effective in finding optimal subsets and is a promising method to the fields of DM. The use of linear and quadratic discriminant analysis, logistic regression, multilayer perceptron, SVM, classification trees and ensemble methods [30] reduced misclassification up to 13.7%, compared with other classic models such as Linear and quadratic discriminant analysis, Logistic regression and Bagging. He et al. [31] used domain-driven multiple constraint-level programming-based classification outperforming the SVM, decision tree, and neural network in terms of sensitivity and Kolmogorov-Smirnov value while maintaining the trade-off between time efficiency and acceptable accuracy and specificity. Whereas Kim et al. [32] used ordinal pairwise partitioning, cooperating with multi-class support vector machine for the credit rating, and it outperformed the multi-class support vector machine and other DM techniques, such as Multiple Discriminant Analysis(MDA), Multinomial Logistic Regression(LOGIT), Case-based Reasoning(CBR) and ANN. Similarly, Huang [33] proposed a Gaussian-process-based multi-class classifier (GPC), which outperformed conventional multi-class classifiers and SVMs. Li et al. [34] proposed a vector machine based infinite decision agent ensemble learning system, in which, soft margin boosting was used to overcome overfitting, and the perceptron kernel was used to simulate infinite subagents. Other research in SVM-based hybrid methods can be found in [35]-[36]. Koutanaei et al. [37] developed a hybrid DM model of feature selection and ensemble learning classification algorithms on the basis of three stages namely: Data gathering and pre-processing, followed by Employment of four Feature Selection algorithms of principal component analysis (PCA), genetic algorithm (GA), information gain ratio and relief attribute evaluation function. In the third stage, the classification results showed that the artificial neural network (ANN) adaptive boosting (AdaBoost) method had higher

classification accuracy in credit scoring. Number of classifiers were compared in a study [38] such as logit/probit and LDA to fully nonlinear classifiers, including NNs, SVMs and more recent statistical learning techniques such as generalised boosting, AdaBoost and Random Forests (RFs). Study concluded that simpler classifiers such as Regression and LDA can be viable alternatives to more sophisticated approaches, particularly if interpretability is an important objective of the modelling exercise.

Several statistics-based DM techniques, such as SVMs, decision trees, neural networks, and k-nearest neighbours are used to construct credit scoring models.

SVMs have been used widely for credit scoring [39]-[40]. Predictive models for credit card fraud detection are in active use in practice [41]. Research has shown that SVM is one of the most effective tools in credit risk evaluation. However, the performance of SVM is sensitive to the algorithms for the quadratic programming, to the parameters setting in its learning machines, and to the importance of different classes. Danenas et al. [42] used SVM based classifiers, and Yu et al. [43] claimed the weighted least squares SVM classifier achieved promising results. Similarly, to help assess the creditworthiness of loan applicants, Trustorff et al. [44] showed that the theoretical advantages of SVM classifiers can be used to improve the accuracy and the reliability of prediction of probabilities-of-default classification. Recently, for credit risk evaluation on larger databases, Danenas et al. [45] presented a technique for optimal linear SVM classifier selection based on particle swarm optimisation technique. Also, comparison with Logistic Regression and RBF networks was carried out which showed the proposed technique giving comparable results to the classic techniques.

Zhou et al. [46] used several SVM ensemble models to reduce inductive bias towards samples and parameter settings, shown by single SVM machines. In a similar study, Ghodselahi [47] used ensemble model of ten Support Vector Machine classifiers to improve the accuracy of classification for credit granting decisions. A hybrid ensemble model for credit risk combining both clustering and classification was designed. Ten SVM classifiers were the members of the ensemble model.

ANNs have been criticised for their ‘black box’ approach and interpretative difficulties but they are a very flexible family of models and are another well-known technology for credit scoring. A lot of efforts have been made onto the applications of neural networks in credit risk evaluation, and the back-propagation learning algorithm is an efficient learning algorithm for training ANNs in the automatic processing of credit applications. Wah et al. [48] investigated three credit scoring models, logistic regression (LR) model, classification and regression tree (CART) model and neural network (ANN) model, to discriminate rejected and accepted credit card applicants of a bank. Results showed that the Neural Network model had a slightly higher validation predictive accuracy rate. Later, [49] proposed a hybrid system with genetic algorithm and artificial neural networks to find optimum feature subset to enhance the classification accuracy for retail credit risk assessment. Other research on ANN can be found in [50] and [51]. ANN and CART decision trees have shown that the forecast accuracy of credit rating process could be increased up to 96.5% [52].

For credit scoring, Marcano-Cedeno et al. [53] developed an ANN training algorithm, inspired by the neurons' biological property of meta-plasticity, which can be efficient when few patterns of a class are available, or when information inherent to low probability events is crucial. In situations where poor financial information was available, Falavigna [54] proposed a simulation model for assigning rating judgements to financial firms. It was the first tool able to forecast the default event two years before the bankruptcy.

Artificial neural networks, especially, MLPs were used [55] for credit scoring in microfinance industry, and it is shown that MLPs credit scoring can get higher accuracy in performance and lower misclassification costs than the classic Linear Discriminant analysis, Quadratic Discriminant analysis and Logistic Regression models. Self-Organising Maps (SOM), another variant of ANN which is a clustering and unsupervised method is a common way for labelling the clusters and is called as Voted method. This method labels each cluster based on the majority class in it. The study [56] compared the capabilities of SOM that is labelled by Voted method and SOM that is labelled by a feedforward Neural

Network in forecasting of credit classes. The comparison performed well in a commonly used benchmark, the Australian dataset.

In an effort to develop credit risk estimation models and to evaluate an influence of input data reduction on credit risk models accuracy, Mileris and Boguslauskas [57] used ANNs and Logistic Regression(LR). The highest classification accuracy was shown by LR model followed by ANN. The Discriminant Analysis technique was the third accurate in classifying companies with credit risk.

The challenges of constructing credit scoring models lie in the availability of data and sample selection issues, and classification methods such as scorecards and decision trees are relatively easier to deploy in practical applications [58]. Also dual strategy ensemble trees can reduce the influence of the noise data and the redundant attributes of data to obtain the relative higher classification accuracy [59]. In cases of large class imbalance, the C4.5 decision tree algorithm, quadratic discriminant analysis and k-nearest neighbours perform significantly worse than Linear Discriminant Analysis and Gradient Boosting [60]. Bahnsen et al. [61] proposed an example-dependent cost-sensitive decision tree algorithm, by incorporating the different example-dependent costs into a new cost-based impurity measure and a new cost-based pruning criterion. Further the proposed method built significantly smaller trees in only a fifth of the time with a superior performance measured by cost savings. This could lead to a method with more business-oriented results creating simpler models that are easier to analyse. Zakrzewska [62] investigated a possibility of connecting unsupervised and supervised techniques for credit risk evaluation by building different rules for different groups of customers. Each credit applicant was assigned to the most similar group of clients from the training data set and credit risk is evaluated by applying the rules proper for this group. Results obtained with this technique of clustering and decision tree on the real credit risk data sets showed higher precisions and simplicity of rules obtained for each cluster than for rules connected with the whole data set. DT was again the subject of study in [63]. An ensemble approach based on merged decision trees, the correlated-adjusted

decision forest (CADF) was introduced to produce both accurate and comprehensible credit risk models.

For credit rating, Tsai and Chen [64] developed four different hybrid models based on classification and clustering techniques providing highest prediction accuracy and lowest error rates in terms of credit rating, where a new technique, Grey DM was introduced, based on Grey System's concept, Analytic Hierarchy Process technology and classical DM technologies.

Rough sets have received less research attention in the area of credit scoring. [65] used rough sets with SVM to create a credit scoring classifier, outperforming linear discriminant analysis, logistic regression and neural networks. Integration of rough set, fuzzy set and probability theories was proposed in [66] for classifying credit risks. A basic parameter, representing the likelihood that a loan will not be repaid and will fall into default, was the inspiration behind this study. Chuang et al [67] developed a two stage hybrid model based on artificial neural networks and rough set theory for credit scoring. Some other research work can be found in [68]. Recently Shen and Tzeng [69] proposed an integrated hybrid soft computing model to resolve the financial prediction problem by adopting a dominance-based rough set approach to solve the financial performance prediction problem. Multiple criteria decision making and the influential weights of DANP (DEMATEL-based ANP) were used for further processing of core attributes along with the data from 2008 to 2011 from central bank of Taiwan for obtaining decision rules and forming an evaluation model. In the results, the proposed model showed that the top-ranking bank outperformed the other four banks.

Integrated approaches have been the favourite of researchers because they take advantage of different DM techniques. Also in the cases where the training data and expert rules are insufficient and/or corrupted, mixed or hybrid approaches are required [70]. ANNs, SVMs and rough sets also have been proven to work well for the problem of credit rating. Bayesian networks and Fuzzy Apriori Genetic algorithms were explored in [71] and [38] respectively.

Ubiquitous DM(UDM) classifier, which works in three steps such as: (1) constructing different models using datasets, (2) inducing rules from these models, and (3) consolidating these rules was used to predict credit ratings [72]. This study combined a number of classification models into one such that the performance of the consolidated model is better that of the original individual classification models in their classification accuracy and efficiency. Also, the model was benchmarked against logistic regression (LR), Bayesian style frequency matrix, multilayer perceptron (MLP), classification tree methods (C5.0), and neural network rule extraction algorithms. Empirical results indicated that UDM outperformed all these single classifier models.

There exist numerous techniques and methods, but most of them depend on a particular data set or attribute set in question. In order to better apply these techniques and methods, getting insight of problems will help improve the performance of decision making or classification.

Figure 3-2 shows the distribution of surveyed techniques in the field of Credit Rating. Hybrid methods where usually multiple techniques are combined in stages, are most widely used techniques for predicting credit scores. The hybrid methods included combining SVM, ANN, DT, GA, Clustering, Regression, Rough sets and KNN. Many classic techniques were studied in parallel to compare the performance. Such studies have been distributed in all the categories that were used. Modern classification techniques, despite being criticised as 'Black-box' techniques and being computationally expensive, often imply SVM and ANN. Many variants of SVMs, such as Least Square SVM, Multi-agent ensemble and Kernel-based, have been developed. MLP and RBF Neural Networks were often used. Obviously, complex non-linear techniques of ANNs and SVMs play significant roles for building credit scoring models in this period, totally accounting for 33% of studies. A traditional classification technique of regression is studied equally alongside SVM and ANN. Hybrid approaches account for 30%. Moreover, most hybrid approaches are the extension of SVMs and ANNs. Decision trees were studied in conjunction with SVM, Regression, ANN and Genetic Algorithms. A number of other techniques such as Gaussian process, Multiple Criteria Linear

Programming-Optimisation technique, Bayesian Network and Fuzzy logic have seen fewer studies.

Figure 3-3 depicts the number of wins in the usage of machine learning techniques for the purpose of prediction of credit ratings. Many studies reviewed in this work have used multiple techniques for comparison purposes. Such studies are classified under all the techniques used. The winning technique is usually one, in some cases two giving comparable performance. Hence the number of wins is lesser that the number of techniques found in Distribution. In Figure 3-3, the hybrid methods have emerged as winning techniques more often compared to other techniques.

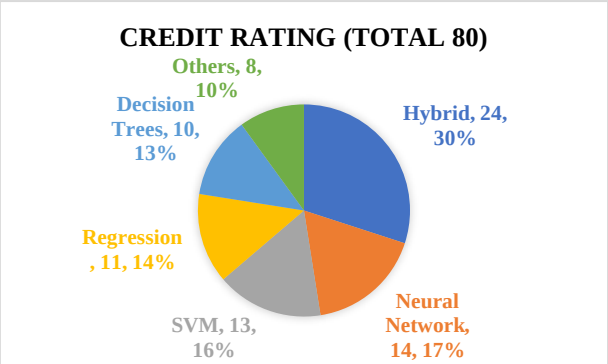

Figure 3-2: Distribution of surveyed techniques applied for Credit Rating

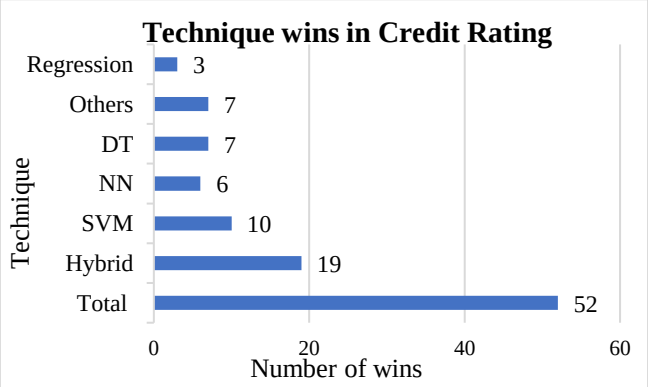

Figure 3-3: Numbers of wins for the DM techniques applied for Credit Rating

In the area of Credit Rating, hybrid computational intelligence techniques have shown their superiorities over single techniques as can be seen from the number of wins above. By integrating the standard basic techniques into hybrid machine

learning solutions, various intelligent searches and reasoning techniques have been developed to solve advanced and complex problems based on different domain knowledge.

Few studies focus on NN, SVM and DT, still fewer on other machine learning techniques along with regression. Hybrid methods and SVMs have been proved as consistent winners. Some classifiers systematically outperform others based on how they handle the linearly separable data, the random noise in the data and the outliers and the non-linear relationships in the data. SVMs have been the most popular individual techniques. Many studies developing hybrid classification strategies have made use of SVMs. SVMs can separate linearly separable data and are almost immune to outliers as well as can fit non-linear data with great performance due to the kernel trick (see section 5.4.2). SVMs use a risk function consisting of the empirical error and a regularised term derived from the structural risk minimisation principle [73]. SVMs chooses a hyperplane with greatest margin separating the classes. The greater the size of this "buffer zone" the lower the risk of a wrong classification for an unseen sample. Thus, a SVM does structural risk minimisation by choosing the separator with the greatest margin where the margin can be seen as a measure of complexity. SVMs like neural networks do not suffer from constraints of statistical distributions. With SVMs overfitting is unlikely to occur and they often produce very accurate classifiers [74]. Also, they do not require huge training samples and have little possibility of overfitting. In addition, the solution of SVM may be the global optimum. Therefore, overfitting of the results is unlikely to occur with SVM. However, determination of the parameters of a kernel function, and the penalty cost hinders accurate prediction results when using SVM.

3.4 Loan Default Prediction

Banks and financial institutions rely on loan default prediction in credit risk management (Kou et al. 2014). They use DM models to predict loan default before they decide to grant a loan with the goal of reducing defaults.

Chen and Chen [75] used the homogeneity between different city districts, the magnitude of the heterogeneity, and a prior distribution for the heterogeneity to formulate Loss Given a Default, which indicates the credit risk for a given default. Prediction of loan default has been studied widely by researchers. Reddy et al. [76] used attribute relevance analysis to predict loan default. This method excluded irrelevant attributes, thus reducing number of units for neural network model. The evaluation method of financing credit capacity was proposed in [77] combined with Kirkpatrick model and fuzzy neural network algorithm.

To predict loan risk well ahead of time using an imbalanced and large dataset, Srinivasan et al. [78] coupled Partial Least Squares Regression model and Variable Influence on Projection scores to select the most important variables. This made the model less complex and computationally efficient, particularly for high risk loan records. RFs were used on large imbalanced data to predict loan defaults in [79]. The original RF algorithm was improved by allocating weights to decision trees in the forest during tree aggregation for prediction. The weights were easily calculated based on out-of-bag errors in training. To predict the performance of online peer-to-peer lending and classify the risk of loan into four categories, DT, ANNs and SVM were used in [80]. This study used RF for feature selection in the modelling phase and showed that the term of loan, annual income, the amount of loan, debt-to-income ratio, credit grade and revolving line utilisation play an important role in loan defaults. But the prediction performance of SVM, Classification and Regression Tree (CART) and MLP were almost equal. But the study [81] found that DT enhanced with resampling techniques like AdaBoost performed better at enhancing the capabilities in classification than in prediction. Decision Trees are part of study in many studies in 2015 such as [82] for comparison between C4.5 and ID3 where C4.5 reached highest performance with data partition of 90%-10%; DT along with Fuzzy set theory where no preprocessing or sampling was done for imbalanced datasets problem, outperforming C4.5 DT [83]; DT along with ANN and SVM with ANNs showing more accuracy than other classifiers [84]; comparative study of RF, SVM, LR and KNN for identifying good borrowers in social lending with winner being RFs [85];

RFs giving better results than ID3 and C4.5 [86]; Fuzzy apriori combined with PCA creating a compact rule base and better results than the single fuzzy apriori model and other combined feature selection methods [87]; DT to build up a model to predict prospective business sectors in retail banking [88].

Cao et al. [89] developed a new model, particle swarm optimisation combined with cost sensitive SVM to deal with the problem of unbalanced data classification and asymmetry misclassification cost in loan default discrimination problem. An extended tuning method for cost-sensitive regression and forecasting was suggested in [90] which was applied to loan charge-off forecasting on a real-world banking data. Logistic Regression used in [91] to estimate the probability of default for the customer credit in Vietnamese bank. Lasso Logistic Regression Ensemble was used in [92].

Many techniques from machine learning have received less attention such as KNN, Hidden Markov model, Loss Given Default, Association Rule mining, Wavelet ANN. In a study of dynamically monitoring loan service [93], the authors applied Hidden Markov Model to show that more accurate monitoring can be achieved by segmenting the defaulted data and training them separately. Chandra et al [94] presented a novel, hybrid soft computing system based on integration of the sample-weighting SVM and wavelet neural network to predict failure of banks. Support vectors along with their corresponding actual output labels were used to train the wavelet neural network (WNN). Further, Garson's algorithm for feature selection is adapted using WNN. Thus, the new hybrid, WNN-Support Vector Wavelet Neural Network accomplishes horizontal and vertical reduction in the dataset as support vectors reduce the pattern space dimension and the WNN-based feature selection reduces the feature space dimension. To identify characteristic patterns of prospective lenders, Aribowo and Cahyana [95] used Association Rule Mining Classifier using Weighted Itemset Tidset tree. Another hybrid model [96] using ANN and SOMs outperformed the traditional Discriminant analysis and Logistic Regression, SVM and Random Forest classification models to predict bankruptcy. Hybrid models [97] made use of Association rule mining and process mining for fraud detection and the study

in [98] used Particle Swarm Optimisation and SVMs for bankruptcy prediction. Loan default prediction forms part of Credit risk analysis for the business of a financial institution. DM increases understanding by showing which factors should be included and which factors most affect specific outcomes [99].

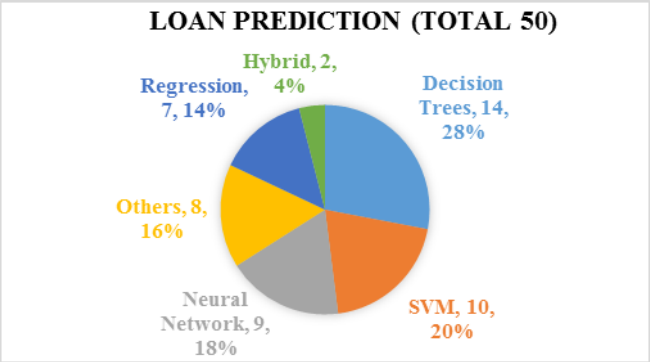

Figure 3-4: Distribution of surveyed techniques applied for Loan Default Prediction

Figure 3-4 represents the use of DM techniques applied for the problem of loan default prediction from 2010 to 2015. Researchers have explored the effectiveness of Decision trees such as ID3, CART and C4.5 extensively. Hybrid techniques are still an emerging trend. The hybrid methods, KNN and Markov model have been investigated more. There is a lot of emphasis on complex, nonlinear supervised algorithms such as SVM, ANNs as well as Regression. The winners are shown in Figure 3-5. Decision trees and Neural Networks give better performance than other methods of SVM, Regression and Hybrid models, Markov model, Fuzzy set theory, KNN and Association Rule Mining.

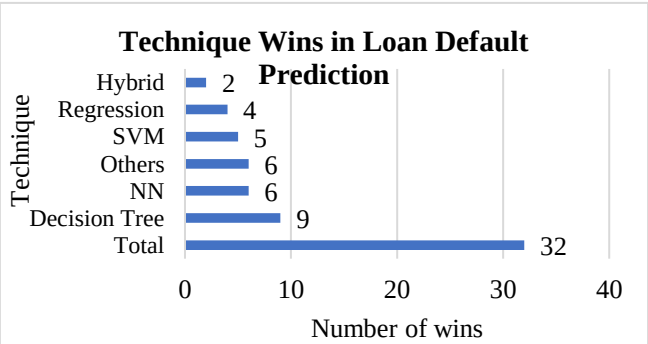

Figure 3-5: Numbers of wins for the DM techniques applied for Loan Default Prediction

As seen in above figure, DTs have been a successful technique in loan default prediction. They are among the tools useful for the task of classification. Feature selection (discussed in chapter 6) is an important aspect of financial analytics. Decision trees implicitly perform variable screening or feature selection. When a decision tree is fit to a training dataset, the top few nodes on which the tree is split are essentially the most important variables within the dataset and feature selection is completed automatically. To overcome scale differences between parameters - for example if we have a dataset which measures revenue in millions and loan age in years, say; this will require some form of normalisation or scaling before we can fit a regression model and interpret the coefficients. DTs do not require variable transformations such as normalisation because the tree structure remains the same with or without the transformation. Missing values will not prevent splitting the data for building trees. Decision trees are also not sensitive to outliers since the splitting happens based on proportion of samples within the split ranges and not on absolute values. DTs do not require any assumptions of linearity in the data. Thus, we can use them in scenarios where we know the parameters are nonlinearly related. The best feature of using trees for analytics - easy to interpret and explain to executives. Decision trees are very intuitive and easy to explain.

3.5 Money Laundering

Money laundering is the process whereby "dirty money" produced through criminal activity is transformed into "clean money", the criminal origin of which is difficult to trace. Financial Institutions are the most affected services for money laundering purposes [100]. Fraud is an extremely serious problem for credit-card companies. Visa and MasterCard lost over $700 million in 1995 from fraud. The cost of global payment card fraud grew by 19% last year to reach $14 billion. The cost of U.S. payment card fraud grew by 29% to $7.1 billion. In the rest of the world, card fraud grew by 11% to $6.8 billion [101]. Some of the well-known examples to support the importance of DM technology in financial institutions are: U. S. Treasury Department, Mellon Bank USA, Capital One Financial Group, American Express, MetLife Inc., Bank of America (USA) [102]. A system

developed by the Financial Crimes Enforcement Network of the U.S. Treasury Department called ‘FAIS’ (FinCEN Artificial Intelligence System) detects potential money laundering activities from a large number of big cash transactions [102]. Mellon Bank has used the data on existing credit card customers to characterise their behaviour to predict what they will do next, to predict which customers will stop using credit card in the next few months. Using DM techniques, Capital One Financial Group tries to help market and sell the most appropriate financial product to 150 million potential prospects residing in its data warehouse. American Express uses data warehousing and DM to cut spending and loan application screening. Metlife Inc. uses the “information - extraction” approach in which the input text is skimmed for specific information relevant to the particular application. Bank of America identified savings of $4.8 million in 2 years (a 400% return on investment) from use of a credit risk management system based on statistical and DM analytics. They have also developed profiles of most valuable accounts in order to identify opportunities to sell them additional services. Recently, to retain deposits, they have used ‘Knowledge Extraction Engines’ analytic framework in identifying clients likely to move assets and then creating offers conducive to retention [102]. As banking and payments have moved onto mobile and online channels, the opportunities for fraud have expanded. It is found that 50% of fraud is undetected until after the money has been lost. Patterns in data can be examined to identify chances of fraud occurrence, and prevention is possible.

Every financial institution is taking the responsibility of developing policies and procedures to fight money laundering. Money laundering regulation is seeing several transformations, thus financial institutions require establishing a well-defined plan against money laundering within their organisations.

A seminal study [103] focused on financial fraud and money laundering, included reviews of classification methods such as SVM, Classification Trees and Ensemble Learning, Classification Rules and Rule Ensembles, Neural Networks, Bayesian Belief Networks, Hidden Markov Models.

In an interesting study [104] using Neural Networks in fraud detection, the auditors could use ANNs as complementary to other techniques at the planning stage of their audit to predict if a particular audit client was likely to have been victimised by a fraudster. Lot of studies have combined use of two or more classic techniques. DM techniques, such as clustering, neural networks, genetic algorithms, were applied for the cause of anti-money laundering detection in [105], and heuristics. Also Liu et al. [106] built a core decision tree with clustering algorithm to identify abnormal transaction. Dreżewski et al. [107] developed a system along with data importer and analysing algorithms, such as clustering and frequent-patterns-mining algorithms. In a study to evaluate different clustering algorithms, Cai et al. [108] showed that density-based clustering does not suit financial dataset. Normalised centroid-based clustering with higher DI or lower DBI gives the best number of clusters to help understanding financial data classification. K- means clustering method with multi-level feed forward network in [109] stressed that published financial statement data contains falsification indicators and ANNs provided highest accuracy. Clustering based anti-money-laundering system in [110] showed that the definition of client profiles that are more tailored to the system's goal, with a database of a greater time span (1 year) and a more thorough exploration of the types of attributes available for the used algorithms, produced better results.

Neural networks and regression models have been used for fraud detection since the dot com bubble burst that caused the 2000 stock market crash. Ravisankar et al. [111] used Neural Network, SVM and Logistic regression to detect fraud in the financial statement of big companies. Whereas Perols [112] used Neural Networks, SVMs, logistic regression and C4.5 to compare the performance of popular statistical and machine learning models, Zhou and Kapoor [113] used Neural Networks, Bayesian Networks, regression, decision tree to detect financial statement fraud. Wei et al. [114] presented an algorithm to mine contrast patterns along with neural network and decision forest to distinguish fraudulent behaviour from genuine behaviour.

Among the ANN and SVM models, kernel-based SVM methods are found to be most robust and accurate. To address the non-stationary problem in financial forecasting, Qin et al. [115] presented a novel non-linear combination of multiple kernel learning model, called Gompertz model, for time series. This model showed good results compared to original multiple kernel learning and single SVM, but with heavy computation burden. In the early warning system model for financial risk detection [116], a plenty of quantitative dependent variables, including 31 risk profiles, 15 risk indicators, 2 early warning signals and 4 financial road maps were considered in decision tree, using the Chi-square Automatic Interaction Detector algorithm to recognise those companies that need improvement. To meet the demands of the dynamic nature of business operations, Sun et al. [117] constructed a new dynamic financial distress prediction model by integrating financial indicator selection (a sequential floating forward selection method), principal component analysis and back-propagation neural network, optimised by genetic algorithm. This model performed better than static models. In a study of credit loan fraud detection, Choi et al. [118] used individual level utility of each customer instead of the mean-level utility for classification using Decision Tree, Bayesian network and their Bagging to predict the probability of each customer being a fraud. Grammar-based Multi-objective Genetic Programming with Statistical Selection Learning which applies the concepts of multi-objective optimisation, token competition, and ensemble learning for evolving classification rules, in [119] to identify fraudulent information was able to obtain better performance in classifying fraudulent firms than LR, ANN, SVM, Bayesian networks and DTs.

Since money laundering is a non-linear problem and is a noisy process, because no distinct boundaries between legitimate and fraud accounts exist, neural networks offer the best dynamic solutions which are capable of evolution over time in this problem domain [120]. Figure 3-6 also shows the fact that the common classification technique of ANN remains the most common learning models in the area of money laundering and fraud detection. The category shown as ‘Other’ covers techniques such as genetic programming, Bayesian networks and kernel

learning and these techniques are studied comparably with the individual baseline models of ANN and DT.

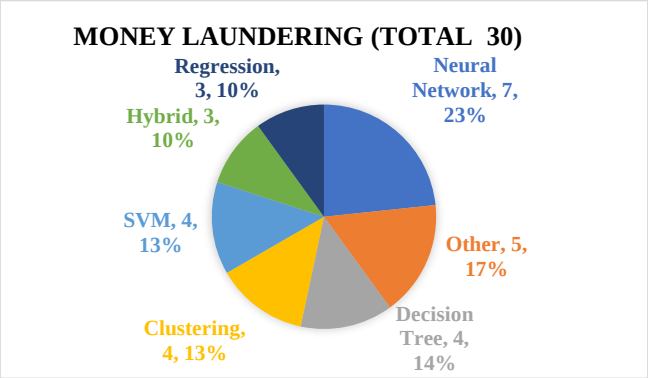

Figure 3-6: Distribution of surveyed techniques applied for Money Laundering

Figure 3-7 below iterates the fact that ANNs and Other techniques discussed above have proved their accuracy in the field of detection of money laundering.

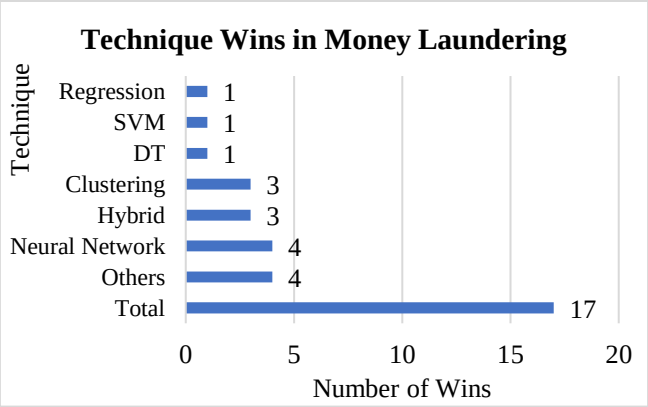

Figure 3-7: Numbers of Wins for the DM techniques applied for Money Laundering

3.6 Stocks Prediction

Stock prediction with the help of DM techniques is vastly investigated by researchers in past decade. This application can be used by financial markets to make qualitative decisions. Prediction of stock market is not an easy task because of the fluctuations of the stock market. Many methodologies and models have been developed to predict the probability of profit-making in the stock market. This area has received the most attention from researchers.

In the past, stock price prediction models have been developed based on statistical or regression analysis tools. Some classic nonlinear models, such as Auto-regressive, Vector Autoregressive [121], Auto-regressive moving average (ARMA), auto-regressive integrated moving average (ARIMA), auto-regressive conditional heteroskedasticity (ARCH), and generalised ARCH (GARCH) [122], have been used to predict the values and trends of the stock market.

ANNs are a popular tool for financial forecasting because they are capable of dealing with very complex patterns. ANNs with high number of hidden layers and units can learn regularities having arbitrary complexity. ANNs have been investigated for studying stock markets in [123] and [124]. Some recent work on ANN can be found in [125] for stock price forecasting, [126] for stochastic time effective ANNs in forecasting stock market indexes, [127] for ANN compared with Auto-regression in forecasting problem as well as in [128]-ANN and [129]- ANN for forecasting of stock market. As fuzzy technology is a good approach to representing uncertainty of objects or concepts, recently, there were many studies focused on fuzzy techniques, combining with other DM techniques to improve the robustness of system for stocks prediction. For example, Fuzzy ANN was explored in [130] and [131] to improve the stock market forecasting capability of the system. Recently a four layer ‘bat-neural network multi-agent system’ architecture for dealing with the distributed nature of stock prediction problem was proposed [132]. This multi-agent approach to create autonomous and independent subtasks to design an accurate prediction model used preprocessing methods in a parallel way such as data normalisation, time lag selection and feature selection. Comparison with genetic algorithm neural network and generalised regression neural network showed best mean absolute percentage error statistics for the proposed model. Some other techniques used in this field are Fuzzy set theory [133] for prediction of stock index prices.

More recently, hybrid DM technologies have been widely used to develop prediction models for stock price/index forecasting. Genetic Algorithm with ANN in wrapper is investigated by [134]. Araújo and Ferreira [135] worked on Morphological-Rank-Linear (MRL) filter combined with a Modified Genetic

Algorithm (MGA). Cheng et al [136] used rough sets theory and GA, Huang [137] used GA and support vector regression; [138] used GA, decision tree, SVM; Qiu et al. [139] used C-fuzzy decision trees and k-nearest neighbours. Hajek [140] demonstrated that the behaviour of stock price's movement can be effectively predicted using prototype generation classifiers. These methods are based on building new artificial prototypes from the training data set and improved the performance of nearest neighbour based classification. Chen and Chen [141] proposed a hybrid fuzzy time series model based on the granular computing approach for stock price forecasting. It regulated the interval lengths during the iteration process using the entropy-based discretisation method. The proposed model also used the binning-based partition to determine reasonable interval lengths by partitioning the universe of discourse and related linguistic values of each datum to change through repeated iterations.

Evolutionary or Genetic Algorithms (GA) have been used routinely to generate useful solutions in a variety of search and optimisation problems. GAs are inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Since GAs can rapidly locate good solutions even for difficult search spaces, they have been applied in Stock market and other finance fields widely. Huang et al [142] used fuzzy-based GAs. Hsu [143] used self-organising map and genetic programming. Associative classification is more accurate than a traditional classification approach but this method is not good at handling numerical data and its relationships; which leads to an ongoing research problem of how to build associative classifiers from numerical data. Chien and Chen [144] proposed a highly competitive GA-based algorithm to build an associative classifier able to discover trading rules from these numerical indicators. Multi-gene Symbolic Regression genetic programming [145] evolving linear combinations of non-linear functions of the input variables was compared with traditional multiple linear regression model. This prediction model for the S&P 500 showed more robust results especially in the validation/testing case. Decision Forests again emerged as winners in the study in [146] followed by Support Vector Machines, Kernel Factory, AdaBoost, Neural Networks, k-

Nearest Neighbours and Logistic Regression. This study stressed the usage of ensemble methods in the field of stock price prediction.

Multivariate adaptive regression splines (MARS) is a multivariate, nonlinear, nonparametric regression approach. Because MARS itself has excellent variable selection capabilities, the difference between the degrees of significance for different variables can be analysed, thus providing users with convenient data interpretation and higher user value [147]. Zarandi et al. [148] used MARS for stock price forecasting along with SVR and Adaptive Neural Fuzzy Inference Systems (ANFIS) on four different datasets. This technique was found to be more accurate than the other techniques in predicting all datasets. Recently, MARS along with fuzzy C-means has been researched more in other fields such as bankruptcy prediction [149] and [150].

Recently, SVM-based approaches have been investigated for the problems of stock price prediction. The following work can be found in this area:

Ding [151] applied SVR and compared with Ordinary Least Squares Regression, Back Propagation ANN, Radial Basis Function Networks (RBFN); Wen et al. [152] used SVM with box theory; Luo and Chen [153] integrated piecewise linear representation and weighted SVM. Kazem et al. [154] used SVR with chaos-based firefly algorithm. Yeh et al. [155] used multiple-kernel SVR approach. SVMs and traditional technical trading rules, such as Relative Strength Index (RSI) and the Moving Average Convergence Divergence were studied in [156]. These rules were inputs to SVMs to determine the best situations to buy or sell the market. SVM itself can be modified with Fishers feature selection, Volume Weighted-SVM, input vector delays and technical indicators in combination with walk-forward optimisation procedure successfully for the purpose of predicting short-term trends on the stock market [157]. But still, Least Squares Support Vector Machines (LS-SVMs) is a popular choice for classification in this area [158].

The sole use of a statistical model or a machine-learning method cannot adequately model all situations. Generally, combination of different models that

use different sources of information is more effective. Hybrid prediction models that combine some of these methods such as self-organising maps (SOM), hidden Markov models, SVR, particle swarm optimisation (PSO), Regression and simulated annealing, have also been investigated for improving prediction accuracy and can be found in [159] and [160]. Liao and Chou [161] applied association rules and clustering to investigate the co-movement in the Taiwan and China (Hong Kong) stock markets. Chitrakar and Chuanhe [162] combined clustering algorithm *k*-Medoids with SVM and produced better performance in terms of Accuracy, Detection Rate and False Alarm Rate, compared to the combination of *k*-Medoids algorithm with Naïve Bayes classification. A two stage fusion approach involving SVR in the first stage and ANN, Random Forest (RF) and SVR in the second stage was proposed in [163] to address the problem of predicting future values of stock market indices. Experiments with single stage and two stage fusion prediction models showed that two stage hybrid models performed better than the single stage prediction models. The performance improvement is significant in case when ANN and RF are hybridised with SVR and moderate when SVR was hybridised with itself. The benefits of two stage prediction models over single stage prediction models become evident as the predictions are made for more number of days in advance. The best overall prediction performance was achieved by SVR–ANN model.

Hajek [140] applied several Prototype Generation Classifiers to predict the trend of the NASDAQ Composite index, and demonstrated that prototype generation classifiers were more accurate than SVMs and neural networks considering the buy or sell hit ratio of correctly predicted trend directions. Xiong et al. [164] proposed a swarm-based intelligent metaheuristic called firefly algorithm and multi-output SVR as a promising alternative for interval-valued financial time series forecasting problems and statistically outperforming in terms of the forecast accuracy.

Many application-oriented studies are being conducted. Tsai and Hsiao [165] applied multiple feature selection methods such as union, intersection and multi-intersection approaches to identify more representative variables for better

prediction. Some researchers mined textual and contextual information from financial reports to predict stock price movement [166], [167]. Researches [168], [169] used sentiment classification and similarities in Candlestick charts for the task of stock prediction. Bollen et al. [170] analysed the text content of daily Twitter feeds by mood tracking tools along with use of Self-Organising Fuzzy neural network and showed good results of predictions by considering specific public mood dimensions.

Use of statistics and other forecasting methods to predict stock prices was the focus of some research work. For example, [171] proposed a stock price prediction model able to extract data from time series data, news and comments on the news and to predict the stock price. They tested their model on numerical data only, yet missing text contents. Their results showed they were able to outperform other prediction methods such as SVR, Technical Analysis, Sentiment Analysis and Numerical Dynamics. The tone of the financial documents is significantly correlated with historical financial ratios such as profitability, liquidity, debt ratios, and stock price return. Hajek et al. [172] used sentiment information hidden in corporate annual reports successfully to predict short-run stock price returns with the application of several neural networks and ε-support vector regression models which performed better than linear regression models.

The nature of the stock market prediction problem requires the intelligent combination of several computing techniques rather than using them exclusively. More efforts have gone into developing hybrid methods, involving algorithms from basic DM functions such as Association, Classification, Clustering and Regression along with methods from Statistics. These are closely followed by SVMs, ANNs and Regression techniques. Modern finance requires efficient ways to summarise and visualise the stock market data, and extract information from sentiments of market reports. The overall applicability of these methodologies and models still remains to be improved. Figure 3-8 below shows the contribution of different techniques in the field of stocks prediction.

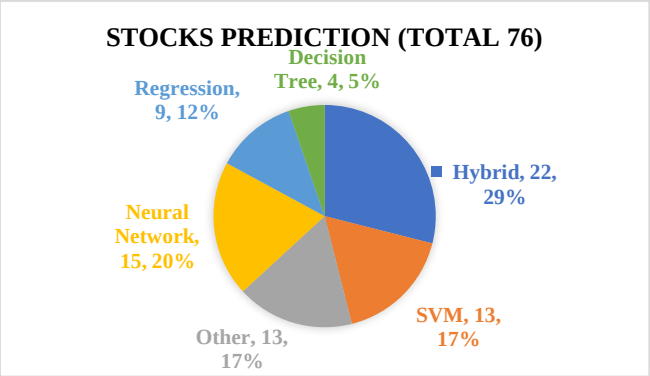

Figure 3-8: Distribution of surveyed techniques applied for Stocks Prediction

The two stage methods of supervised learning, such as ANN, SVM, Regression and Decision Tree, which consist of first training and then predicting, have been used extensively in the area of stock market prediction. But the methods of hybrid machine learning have shown to be more accurate in prediction than single baseline methods. SVM has been receiving increasing interest in the areas ranging from pattern recognition where it was originally applied, to stocks prediction due to its remarkable generalisation performance. This is proved from Figure 3-8 and Figure 3-9. The techniques of Genetic Algorithm, the econometrics model of Random Walk, Bayesian network, Naïve Bayes, Nearest neighbour, Multiple Kernel learning, and Clustering are accounted as 'Other' techniques here, which follow the usage of SVM.

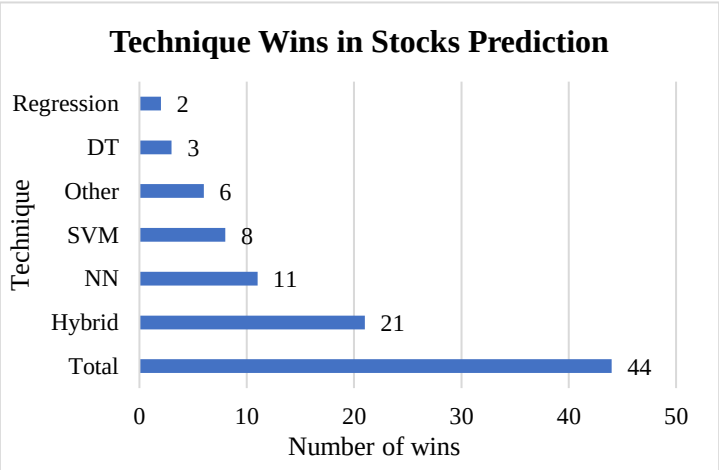

Figure 3-9: Number of wins for the DM techniques applied for Stocks Prediction

Irrespective of what combinations were used in hybrid learning studies, they have shown superiority over single models in the area of stock prediction.

3.7 Financial time series forecasting

Time Series is a collection of organised data obtained from sequential measurements over a period of time. Time series DM tries to extract all meaningful patterns from the shape of data. The goal of time series analysis is to predict the near future data based on past data. The relationship between the future value and previously observed values of variables is modelled in order to forecast the future values of a time series. Prediction of time series values is a distinct and extensive research area.

Modern time series forecasting involves many factors that are complexly correlated with each other, and hence it involves working with noisy, random, nonstationary and chaotic data. Time series forecasting is significant in the field of finance since knowledge of the dynamic relationships among economic variables is essential for the investors to make right decisions at right times to maximise their financial profit and there is a strong dependency between future and past in finance market. According to [173], time series analysis may have one or more of the following objectives.

- Analysis and interpretation- find and interpret model to describe the time dependence in the data.
- Forecasting or prediction- given a sample from the series, forecast the next value, or the next few values.
- Control- adjust various control parameters to make the series fit closer to a target.
- Adjustment- in a linear model the errors could form a time series of correlated observations; adjust estimated variances to allow for this.
- Queries- moving averages, aggregates over time, year to year comparisons [174].

- Forecasting- forecasting of stock prices, sales and financial risk, estimate of credit worthiness, estimates of future financial outcomes for a company or country.

The research on time series prediction has been carried out by researchers from various communities such as machine learning, DM, computer science, artificial intelligence, econometrics and statistics. This section concentrates on the work carried out in financial time series forecasting.

One of the most important static models for prediction, SVR has been applied in the prediction of financial time series with many characteristics of large sample sizes, noise, non-stationary, non-linearity, associated risk. Jiang and He [175] introduced local grey support vector regression and the use of grey relational grade as weighting function to adapt each test point in the time series locally and flexibly improved the performance of SVR. Classification is one of the main tasks in DM. In the time series problem domain, special consideration is usually given due to the nature of the data. Sugimura and Matsumoto [176] proposed a system that acquires feature patterns and developed a classifier for time series data without using background knowledge given by a user. SVMs, which are good at better generalisation of the training data, are studied in forecasting of financial time series aided by preprocessing and can be found in [177]. The key issue of deciding the parameters of the predicting model when using SVM was handled in this study by selecting the PSO method as the optimal tool to build a classifier, namely PSOSVM.

Xiong et al. [178] proposed a swarm intelligence based ‘fully complex-valued radial basis function neural networks’ by using discrete particle swarm optimisation and PSO for joint optimisation of the structure and parameters of the model for interval time series forecasting. The proposed method improved prediction performance and statistically outperformed some well-established contenders like ARIMA and interval valued methods like Holt’s linear trend method and MLP (Multi-Layer Perceptron applied to Interval-Valued Data) in

terms of accuracy measure. Work in neural networks has concentrated on forecasting future values of the time series using current values.

Heuristic approaches, based on ANN or Evolutionary Computation, have been shown to obtain some really good results in time series modelling and forecasting. Neural networks, offering flexible nonlinear modelling capability, can be adaptively generated through training with the features extracted from the data. Recently, ANNs have been used extensively in time series forecasting. Sometimes, classification problems can be transferred to optimisation problems to select best set of features to achieve the best classification performance. For example, feature selection was studied by [179] to present a novel CARTMAP neural network based on Adaptive Resonance Theory that incorporated automatic, intuitive, transparent and parsimonious feature selection with fast learning. Systematic ANN modelling processes and strategies for TSF were developed in [180]. Other ANN studies are: [181] and [182] for time series prediction.

Saigal and Mehrotra [183] applied DM techniques to financial time series data for calculating currency exchange rates of US dollars to Indian Rupees. It comprised performance comparison of regression, vector autoregressive model and ANN on time series data in terms of the forecasting errors in accuracy generated by the models while predicting the currency exchange rates. The analysis was done using four Models: multiple regression in excel, multiple linear regression of dedicated time series analysis in Weka, vector autoregressive model in R and neural network model using Neural Works Predict.

Another hybrid model Multiple Kernel Learning and Genetic Algorithm for Forecasting Short-Term Foreign Exchange Rates in [184] combined multiple kernel learning for regression and a genetic algorithm (GA) to construct the trading rules. More study of hybrid models is done in [185] for financial forecasting. Regression, Fuzzy set theory, Ant Colony Optimisation [186] for time series forecasting; Fuzzy set theory, Clustering [187] for predicting real-world time series; Regression in [188] for prediction of stocks, and in [189] for time series modelling.

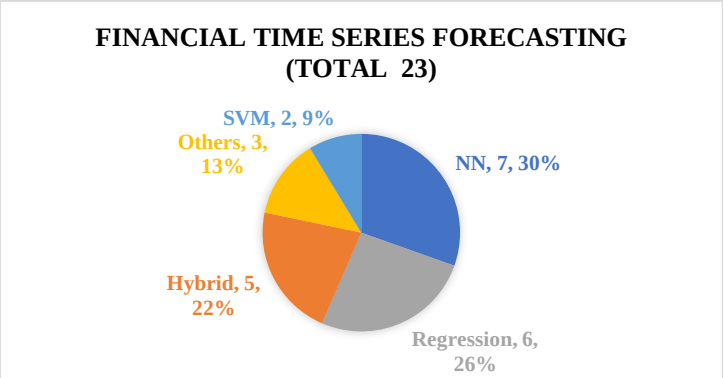

Figure 3-10: Distribution of surveyed techniques applied for Time series Prediction

Time series data has such features that require more involvement by the data miner for preparation process than for non-series data [190]. Hence existing research in the field of time series prediction is inadequate and the problem of mining of sequence and time series data is considered as one of the challenging problems in the field of DM. Dimensionality, representation, lack of well-established approaches in time series, handling of multivariate time series are some of the issues that still need to put much attention.

Figure 3-10 above shows that despite being time consuming in setting up, ANNs provide great classification and forecasting functionality. They are powerful with non-linear data, and hence are popular technique for time series forecasting. Regression and its variants having good explanation ability are widely used by researchers for time series prediction. Hybrid machine learning models combine strengths of multiple knowledge representation model types and are researched quite often along with Regression. The category of 'Other' techniques, consisting of Clustering and Fuzzy set theory, has been investigated in a few studies but SVM has received less attention in financial time series prediction in the surveyed period since they are better suited for classification tasks.

Similar trend is seen in Figure 3-11 for the techniques emerging as winners in the area of time series prediction. ANNs prove to be the most winning techniques in predicting financial time series. Although ANNs can model both linear and non-linear structures of a time series, to handle both equally well, hybrid methods have proved successful and they closely follow usage of ANNs. Due to its

simplicity and interpretability, regression is still popular in time series forecasting and has emerged winner in five studies along with hybrid techniques. Techniques such as clustering and fuzzy set theory have proved better at three studies in this period.

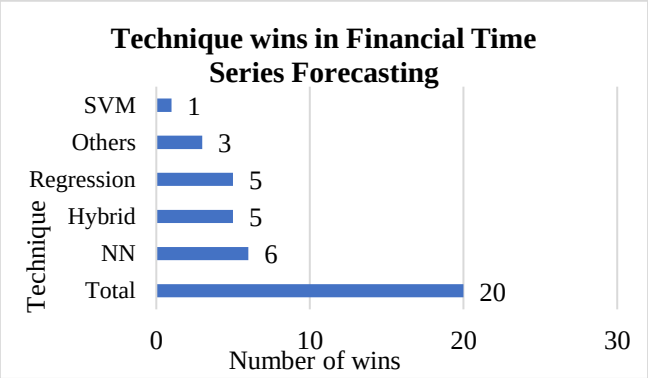

Figure 3-11: Number of wins for the DM techniques applied for Financial Time series Forecasting

3.8 Summary

Financial data are being generated rapidly along with technological advances. DM techniques have been used to discover unknown patterns and predict future in financial markets and they have been investigated over a long period. Availability of vast data and potential significant benefits of solving finance problems have motivated extensive research in this field. This comprehensive review of the existing literature has proved the existence of room to develop more mature and efficient models to help the finance industry.

This work categorised the applications of DM applications in finance industry in respects of Credit Rating, Loan Prediction, Money Laundering, Stocks Prediction and Time Series Prediction from 2010 to 2015. Many DM techniques, including Regression, Neural Networks, Support Vector Machines, Decision Trees, Hybrid methods and some 'Other' techniques, e.g. Genetic Algorithms, have been applied in these areas. Different technologies have different performance for different finance problems. There is no doubt that DM techniques play significant roles in finance industry.

The applications of DM techniques in the fields of forecasting, pricing and prediction are related to various fields, such as Statistics, Econometrics, Artificial Intelligence and Machine learning. Regression, Clustering, Neural Networks, SVMs, Genetic Algorithms are the advanced techniques being popularly investigated in Computational Finance.

A lot of investment decisions in the area of finance industry are supported by DM techniques, which help finance industry to understand the data and gain competitive advantage from the data. Hybrid models tend to evolve more and more. The synergy derived by machine learning, fuzzy logic, clustering and genetic algorithms have proved successful for hybrid methods. Nevertheless, the process should not ignore the importance of data quality, as the uncertainty of data requires the robustness of the mining technology.

The key findings of this study are:

(i) No single algorithm or technique works best across all types of datasets, problems. The choice is governed by the important aspects of dataset being used, the problem area, research objective, data preprocessing techniques involved, performance evaluation criteria, security, privacy and data integrity issues.

(ii) Hybrid models provide better and more accurate results, and hence are used more in the area of Credit Rating and Stock-market prediction. This proves the fact that unless subject to sufficiently rigorous tests entailed by hybrid techniques, a lot of the research could prove to be false.

(iii) Most investigation is for Credit Rating, followed by Stocks Prediction, Loan Prediction, Time Series Prediction and lastly Money Laundering.

(iv) What differentiates finance market is the world of unstructured data that is emerging as large source of actionable insights.

(v) The review finds that Stock prediction and Credit rating have received most attention of researchers, compared to Loan prediction, Money Laundering and Time Series prediction. Due to the dynamics, uncertainty and variety of data,

nonlinear mapping techniques have been deeply studied than linear techniques. Also, it has been proved that hybrid methods are more accurate in prediction, closely followed by Neural Network technique.

A concept worth mentioning alongside neural networks is the technique of deep learning. The recent usage of deep neural networks, also called as deep learning has led to remarkable results in finance sector. Deep learning is a subset of machine learning referring to a method that uses the base concept of ANNs. It allows the creation of complex neural network system to solve complex prediction and classification problems in the real world. It can extract insights from massive, unconnected datasets to unravel the complicated challenges facing financial institutions that have been reviewed in this literature survey [191]. The techniques have the potential to solve a wide spectrum of longstanding problems in finance industry. e.g.: Using deep neural networks, Clinc has developed a conversational AI platform for financial institutions that is like a more advanced “Siri for your bank account” or a “Google Now for your finances”. It supports the same natural language flow that would be available with a personal banker [192]. The important use cases from finance for the use of deep learning and artificial intelligence are risk management, customer understanding, anti-money laundering solutions.

To conclude, this survey could provide a clue of applications of DM techniques for finance industry, and a summary of methodologies for researchers in this area. Especially, it could provide a good vision of DM Techniques in computational finance for beginners who want to work in the field of computational finance.

4 DATA ANALYSIS

4.1 Introduction

A fundamental prerequisite to machine learning is data analysis. The aim of this chapter is to present, visualise, analyse, describe and interpret the datasets in a systematic manner in order to bring order, structure and meaning to the data.

Three benchmark credit scoring datasets and one private dataset is employed in this PhD work. These datasets are used in chapter 5 to investigate a few classical machine learning algorithms for the problem of credit scoring and in chapter 6 to develop a new feature selection strategy.

This chapter identifies trends and relations in accordance with the research aims as per the research methodology from chapter 2.

4.2 The Credit-scoring public datasets

Table 4-1: Class distribution of the datasets used in the study

Dataset	N	n	N_n	N_p
German Credit	1000	20	700	300
Australian Credit	690	14	307	383
Taiwan Credit	30000	24	23364	6636

In the table above,

N = number of total samples present in the dataset;

n = number of features in the dataset;

N_n = number of good credit samples (i.e. corresponding applicant is creditworthy);

N_p = number of bad credit samples (i.e. corresponding applicant is not creditworthy).

4.2.1 German credit dataset

The German Credit data set comes from the Statlog data set repository [193]. It is a historical pre-loan data with observations for 1000 past credit applicants on 20 variables. First variable is the good or bad credit attribute. The original data set consisted of a combination of symbolic and numerical attributes, but we used

the version consisting of only numerical valued attributes. The data set consists of 1000 records of 20 attributes each (7 numerical and 13 categorical) and a binary outcome. The 20 attributes available for constructing credit scoring models include demographic characteristics (e.g., gender and age) and credit details (e.g., credit history and credit amount). The applicants are rated as 'good credit' or 'bad credit'. Among the 1,000 observations, the two target classes are distributed as: 700 samples (70%) for 'good credit' class and 300 samples (30%) for 'bad credit' class.

Table 4-2: Description of the German credit dataset

Feature number	**Feature name**
1	*Good or bad credit - binary - predictor variable*
2	Status of existing checking account
3	Duration of credit in months
4	Credit history: 5 categories such as no credit taken/ critical account etc.
5	Purpose of credit: 11 categories such as car/business etc.
6	Credit amount
7	Savings account/bonds: 4 categories between 100 DM and 1000 DM
8	Present employment since
9	Instalment rate in percentage of disposable income
10	Personal status: 5 categories such as divorced/single/married and gender
11	Other debtors / guarantors: 3 categories such as none/co-applicant/guarantor
12	Present residence since
13	Property: 4 categories such as real estate/ building society/car or other/ unknown
14	Age of the applicant in years
15	Other instalment plans: 3 categories such as bank/stores/none
16	Housing: 3 categories such as if applicant rents/owns/has a free housing

17	Number of existing credits at this bank
18	Job: 4 categories such as if the applicant is unemployed/ unskilled/skilled etc.
19	Number of people being liable to provide maintenance for
20	Telephone: has/does not have
21	Foreign worker: yes/no

4.2.2 Australian credit dataset

The Australian Credit data set contains data for credit card applications from 690 individuals and comes from the Statlog data set repository [193] (part of the UCI data repository). All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data. Both attributes and classes have been encoded. This dataset is interesting because it shows a good mix of attributes. There are 6 continuous and 8 categorical attributes. The two target classes are quite evenly distributed with 307 examples (roughly 44.5%) for class 1(creditworthy) and 383 examples (≈ 55.5%) for class 2 (non-creditworthy).

In the table below, the last attribute is the target variable good or bad credit (creditworthy or not).

Table 4-3: Description of the Australian credit dataset

Feature number	**Feature name**	**Feature number**	**Feature name**
1	A_1	9	A_9
2	A_2	10	A_{10}
3	A_3	11	A_{11}
4	A_4	12	A_{12}
5	A_5	13	A_{13}
6	A_6	14	A_{14}
7	A_7	*15*	*A_{15} - binary - predictor variable*
8	A_8		

This dataset is anonymous, i.e. meanings of variables is not known.

4.2.3 Taiwan credit dataset

The Taiwan Credit dataset contains data about customers' default payment in Taiwan [193]. This is the largest dataset used in this study. The two target classes have 23364 cases (77.88%) of 'good credit' and 6636 cases (22.12%) of 'bad credit'.

Table 4-4: Description of the Taiwan credit dataset

Feature number	Feature name	Feature number	Feature name
X_1	Amount of the given credit (New Taiwan dollar)	X_{12}	Amount of bill statement in September, 2005
X_2	Gender (1=male; 2=female)	X_{13}	Amount of bill statement in August, 2005
X_3	Education (1=graduate school; 2=university; 3=high school; 4=others)	X_{14}	Amount of bill statement in July, 2005
X_4	Marital status (1 married; 2=single; 3=others)	X_{15}	Amount of bill statement in June, 2005
X_5	Age (year)	X_{16}	Amount of bill statement in May, 2005
X_6-X_{11}	History of past payment. Past monthly payment records (from April to September, 2005); The measurement scale for the repayment status is: -1=pay duly; 1=payment delay for one month; 2=payment delay for two months; . . .; 8=payment delay for eight months; 9=payment delay for nine months and above.	X_{17}	Amount of bill statement in April, 2005
X_6	Repayment status in September, 2005	X_{18}-X_{23}	Amount of previous payment (New Taiwan dollar)
X_7	Repayment status in August, 2005	X_{18}	Amount paid in September, 2005
X_8	Repayment status in July, 2005	X_{19}	Amount paid in August, 2005
X_9	Repayment status in June, 2005	X_{20}	Amount paid in July, 2005

X_{10}	Repayment status in May, 2005	X_{21}	Amount paid in June, 2005
X_{11}	Repayment status in April, 2005	X_{22}	Amount paid in May, 2005
X_{12}-X_{17}	Amount of bill statement (NT dollar)	X_{23}	Amount paid in April, 2005
		X_{24}	*Default next month - binary - predictor variable*

4.2.4 Data Analysis and Visualisation

Following is the list of objectives for this section:

- Analyse the chosen datasets to verify the correlation of variables;
- Analyse the datasets to determine which variables are more effective in determining the class of applicants.

The procedure is explained in detail for the German credit dataset and followed similarly for the other two datasets.

4.2.4.1 German credit dataset

Scatter plots are useful in performing exploratory data analysis since they help determine the frequency of values for each attribute in the dataset, which directly means the importance of the attributes.

Scatter plot matrix is a matrix of pair-wise scatter plots of variables. Its usage is to determine whether the variables in the dataset are correlated and also the type of correlation i.e. positive or negative. It shows pair-wise relationship and distributions in the data.

Correlation is a measure of the strength of linear relationship between two variables. A strong correlation between the two variables indicates high linear association between the variables. This association may be positive or negative. Two variables that are uncorrelated are not necessarily independent, because they might have a nonlinear relationship.

This work used Pearson's coefficient formula:

$$r = \frac{\sum_{i=1}^{n}(Y_{act}-\bar{Y}_{act})(Y_{est}-\bar{Y}_{est})}{\sqrt{\sum_{i=1}^{n}(Y_{act}-\bar{Y}_{act})^2(Y_{est}-\bar{Y}_{est})^2}}$$

(4-1)

Where: n is the sample size, Y_{act} is the real observed value, $\bar{Y}_{act}$ is the average of real observed value, Y_{est} is the predicted value, $\bar{Y}_{est}$ is the average of predicted value from the model.

The figure below shows the matrix of pair-wise correlations for this dataset using a colour gradient. Such heatmap maps the correlation by a colour-scale, ranging from blue to red. Dark blue relates to the negative extreme value of the metric, i.e. -1 for Pearson correlation, and dark red refers to the positive extreme value, i.e. 1 for Pearson correlation.

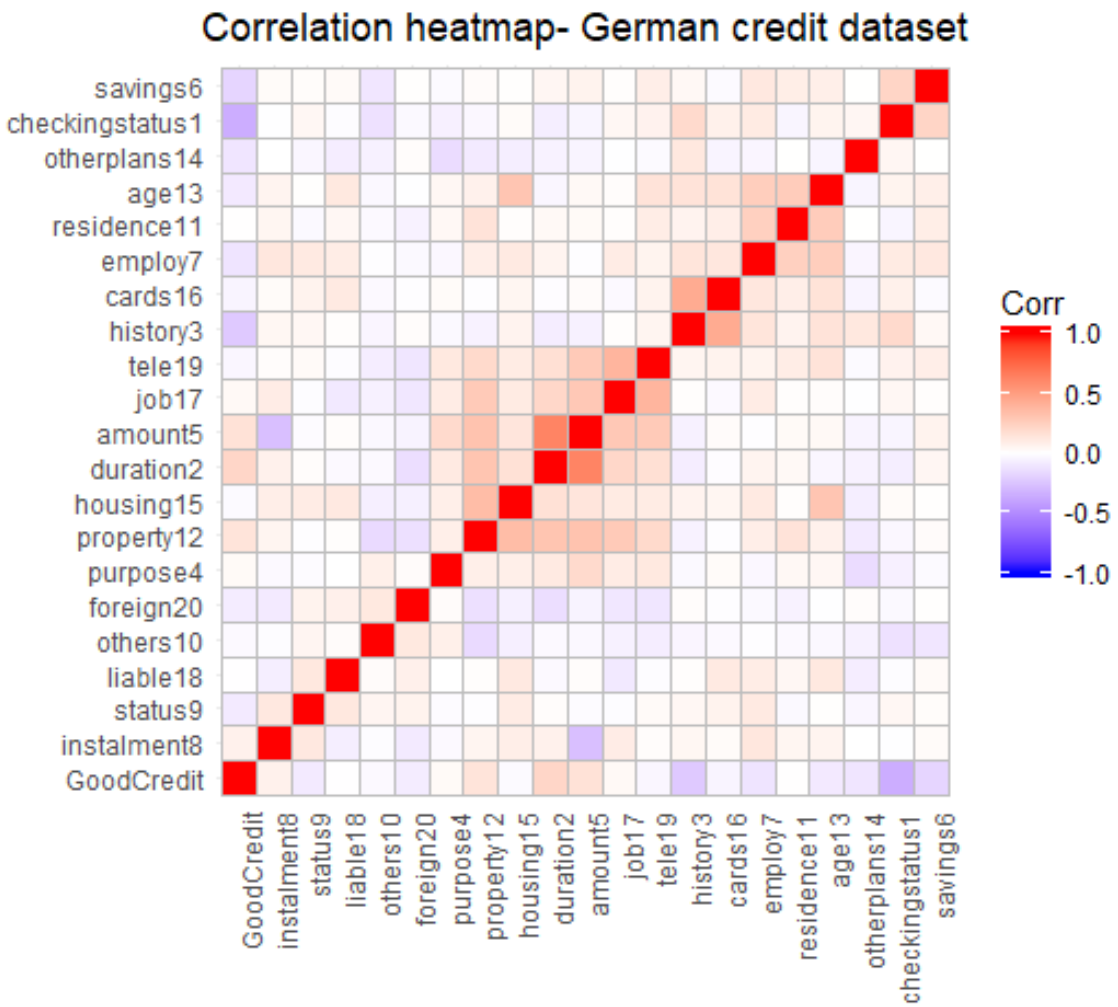

Figure 4-1: Correlation heatmap - German Credit dataset

The variables ‘Duration’, ‘Credit Amount’, ‘History, ‘Number of Cards’, ‘Job’ and ‘Telephone’ show some level of positive correlation, indicated by shades of red (exclude the diagonal). Positive correlation between pairs of variables indicate that as one variable increases (or decreases) in value, the second variable also increases (or decreases) in value.

In the following figure, we show the actual values of pairwise Pearson correlation coefficients of the variables in this dataset.

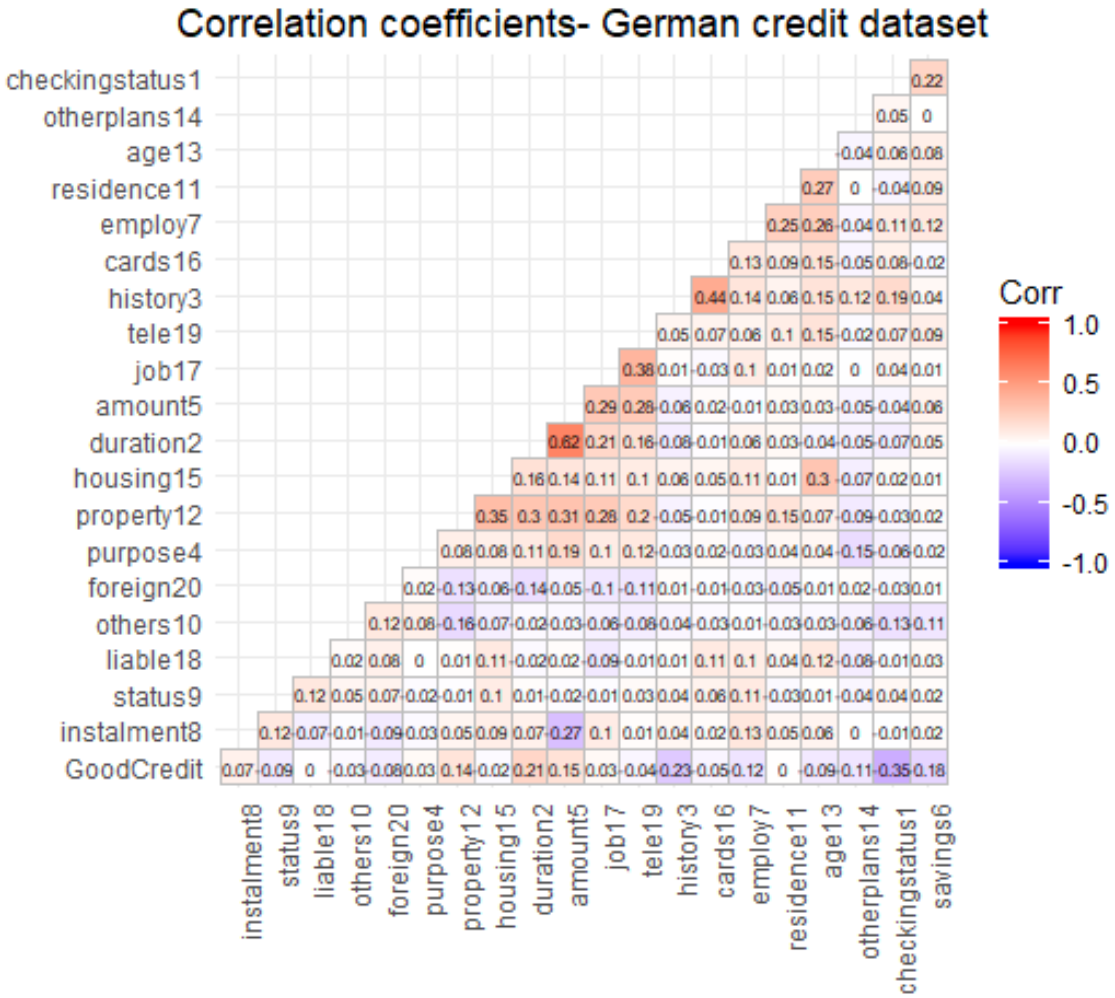

Figure 4-2: Pearson correlation coefficients - German credit dataset

The figure above shows good positive correlation between many pairs of variables; darker the shade of red, stronger the positive relation and darker the

shade of blue, stronger the negative correlation. E.g. a good positive correlation: 0.62 between ‘Duration’ and ‘Credit Amount’, ‘History’ and ‘Number of Cards’ (0.44), ‘Job and ‘Telephone’ (0.38). E. g. Good negative correlation: ‘Good Credit’ and ‘Checking status’ (-0.35), ‘Instalment’ and ‘Amount’ (-0.27).

e.g. we can conclude that as Duration increases (or decreases), the Credit amount in the dataset increases (or decreases).

Below, we will analyse some of these variables in detail.

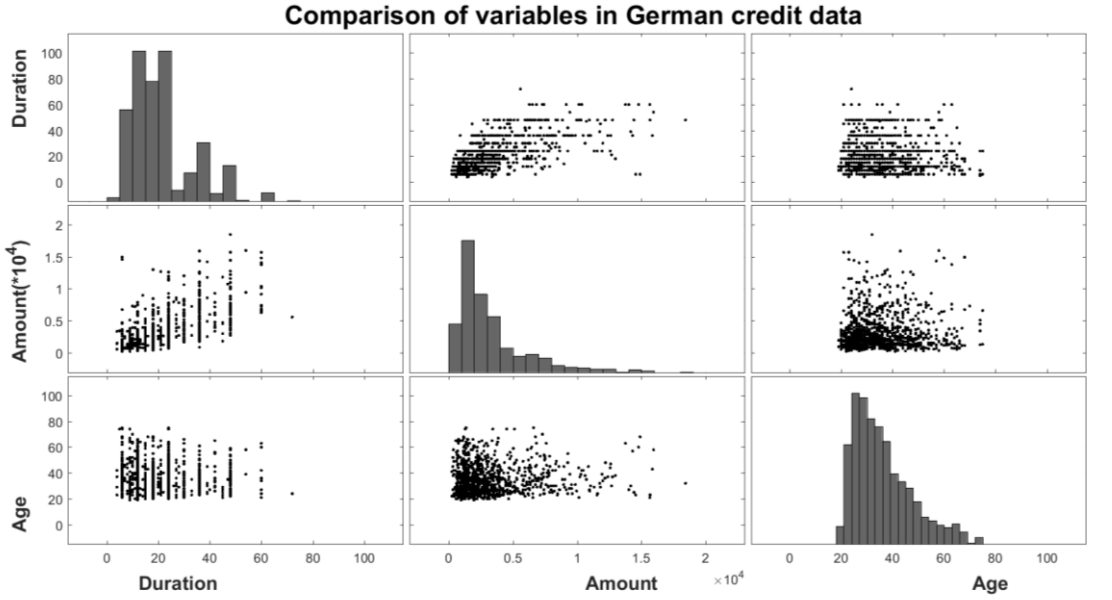

Figure 4-3: The correlated variables in German credit dataset

Figure 4-3 shows an array of the bivariate scatterplots for three variables (‘Duration of credit in months’, ‘Amount of credit’ and ‘Age of the applicant in years’) along with a univariate histogram for each of them in the major diagonal. The distributions appear to be positively-skewed-normal, i.e. many values for these variables tend towards zero. Observing the non-diagonal cells, we can say that these variables show some positive correlation among them and they will be useful in predicting the class of credit clients.

To summarise these variables, the observations are:

- Mean value i.e. a well-defined peak centre of the distribution, cannot be used as a typical value for these variables. Mode could be the choice of value which will appear towards left of the distribution.
- When the data was collected, these variables may have lower bound (e.g. to apply for credit, Age could be zero) and no upper bound on them (e.g. no ceiling on the Age when applying for credit).

For the classification problem of credit scoring, the separation of applicants in the underlying classes could be observed as shown in the figure below. The points in each scatterplot are color-coded by the two classes: Class 0 (dots) is creditworthy, Class1 (+) is non-creditworthy. Such a plot is useful in visualising which variables could be useful in separating and predicting the class of a client.

Figure 4-4: The correlated variables and the separability of classes in German credit dataset

We can see that none of the three variables are very good at separating the classes since the data points are scattered throughout the plot and no clear separation between the classes is seen.

Next, some of the variables are analysed in detail. The figure below shows the correlation between ‘Amount’ and ‘Age’. The clear majority of the points are clustered in the lower left corner of the plot. The graph is expanded due to outliers

in both dimensions (likely due to in part heteroskedasticity that comes with smaller credit amounts for the lower age people). Higher-age applicants have applied for larger credit amounts.

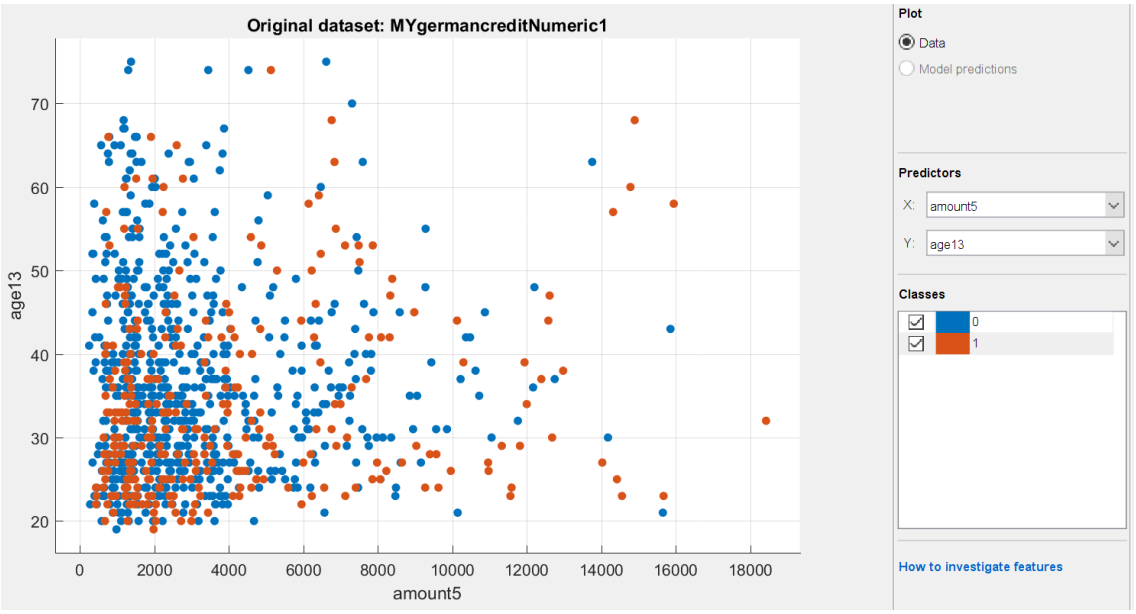

Figure 4-5: Scatterplot showing separability of classes using attributes 'Amount' and 'Age' of German credit data

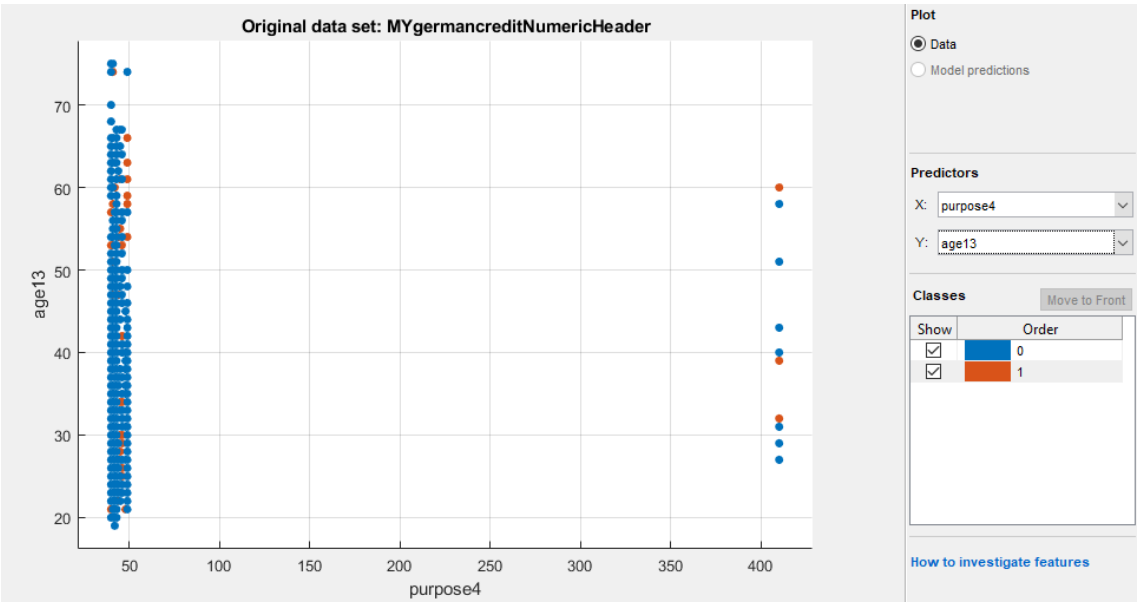

Figure 4-6: Scatterplot showing how 'Purpose' and 'Age' are correlated

Above figure shows that ‘Purpose’ and ‘Age’ could be useful in separating the credit applicants into two classes as the diagram shows class 0 (blue) and 1 (red) applicants clearly separated when using these two variables. Also, the variable ‘Purpose’ takes only few distinct values around 50 and some above 400 (‘other’ category for purpose of credit).

Multivariate or high-dimensional numerical datasets are hard to visualise because we can visualise at most 3 dimensions. Parallel coordinates plot attempts to plot such dataset systems in a different manner. Since plotting more than 3 orthogonal axes is impossible, parallel coordinate schemes plot all the axes parallel to each other in a plane without destroying too much of the geometric structure. The plot contains one line for each observation in the dataset. The colour of each line indicates one of the two classes e.g. credit worthy or non-creditworthy.

They are ideal for comparing many variables together and seeing the relationships between them. Each variable (column here) is given its own axis and all the axes are placed in parallel to each other. Values are plotted as series of lines connected across each axis. This means that each line is collection of points placed on each axis, that have all been connected together. Re-ordering the axes can help in discovering patterns or correlations across variables.

We can observe from the figure below that ‘Checking Status’ and ‘History’ have fewer distinct values as compared with ‘Duration’ and ‘Amount’. It also shows clusters of data: A lot of data points between 1 standard deviation and 2 standard deviation are clustered together around ‘Age’.

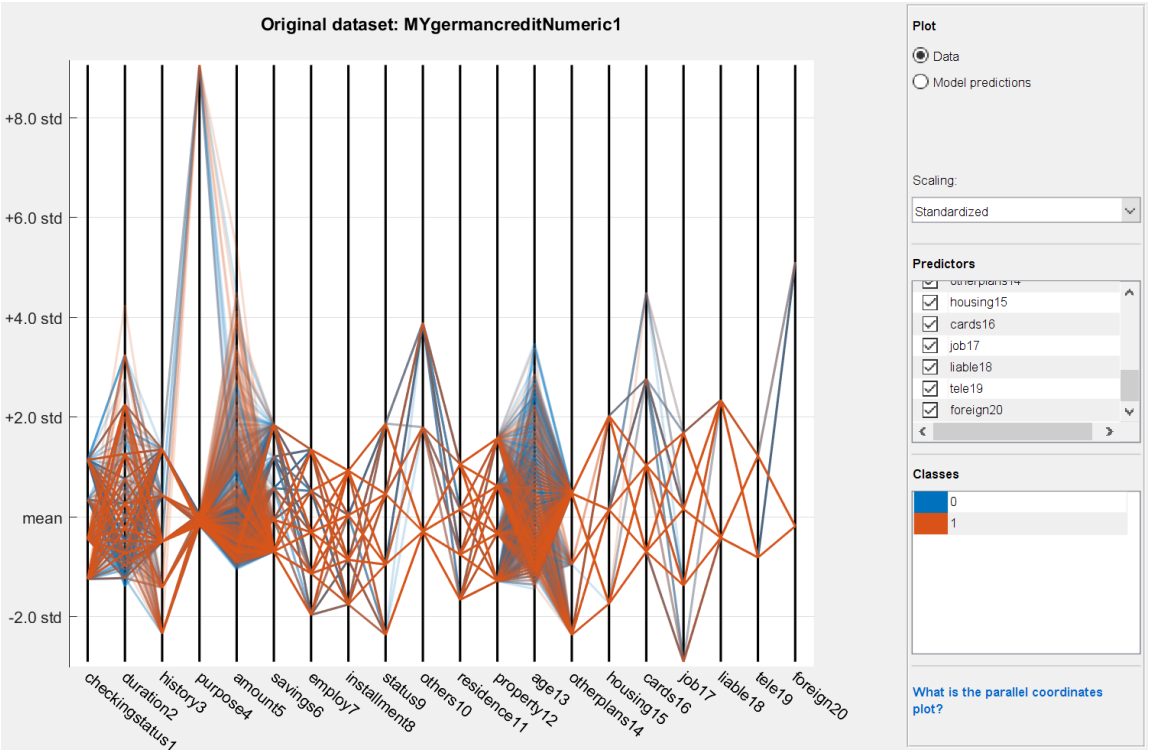

Figure 4-7: Parallel coordinates plot showing relation between all the variables in German credit data

4.2.4.2 Australian credit dataset

The figure below shows scatter plot matrix of all variables in this dataset. It can be used to determine whether the variables are correlated and also the type of correlation i.e. positive or negative. It shows pair-wise relationship (non-diagonal cells) and distributions (diagonal cells) of variables in the data.

The figure below shows the matrix of pair-wise correlations for this dataset using the same colour gradient as detailed in previous section.

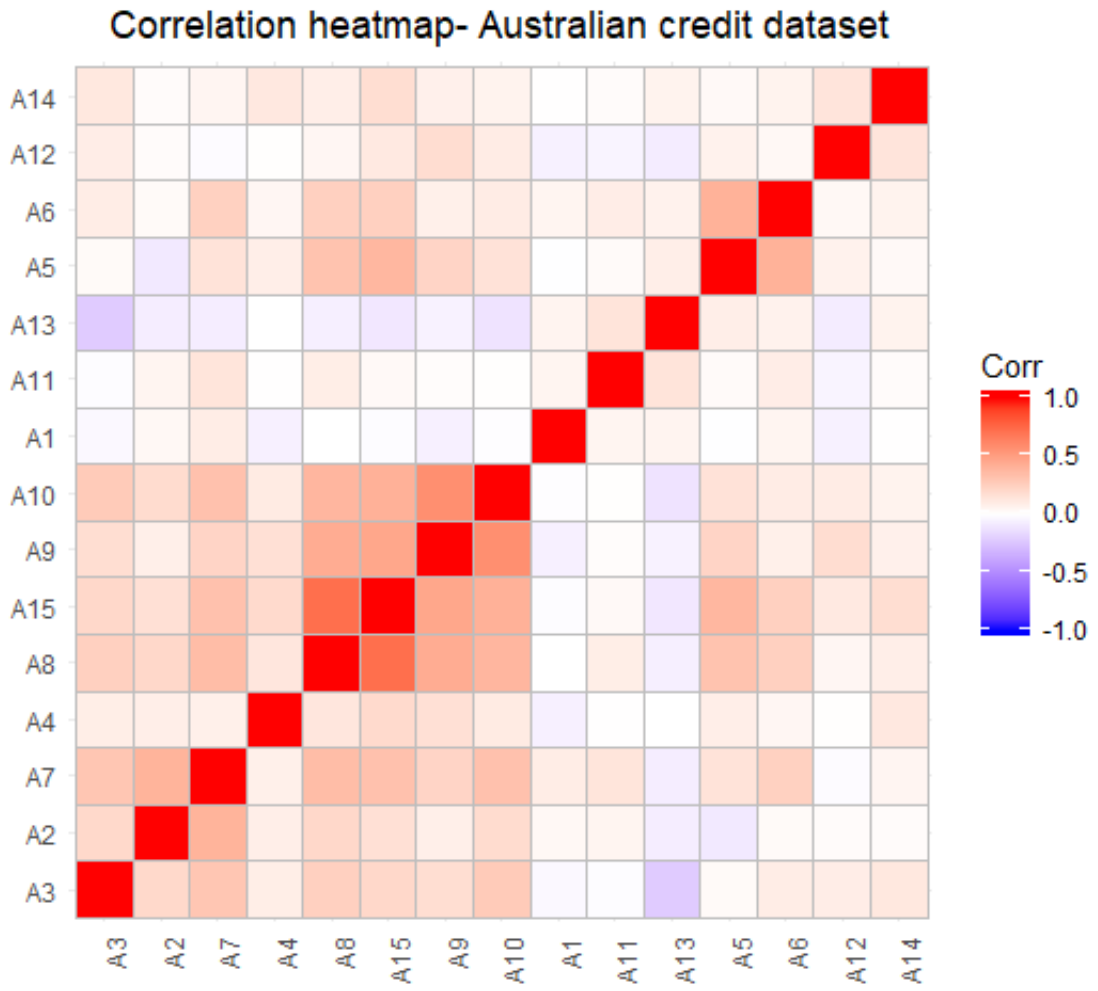

Figure 4-8: Correlation heatmap - Australian Credit dataset

The variables indicated by deeper shades of red such as A6, A5, A10, A9, A15 show good positive correlation.

In the following figure, we show the actual values of pairwise Pearson correlation coefficients of the variables in this dataset.

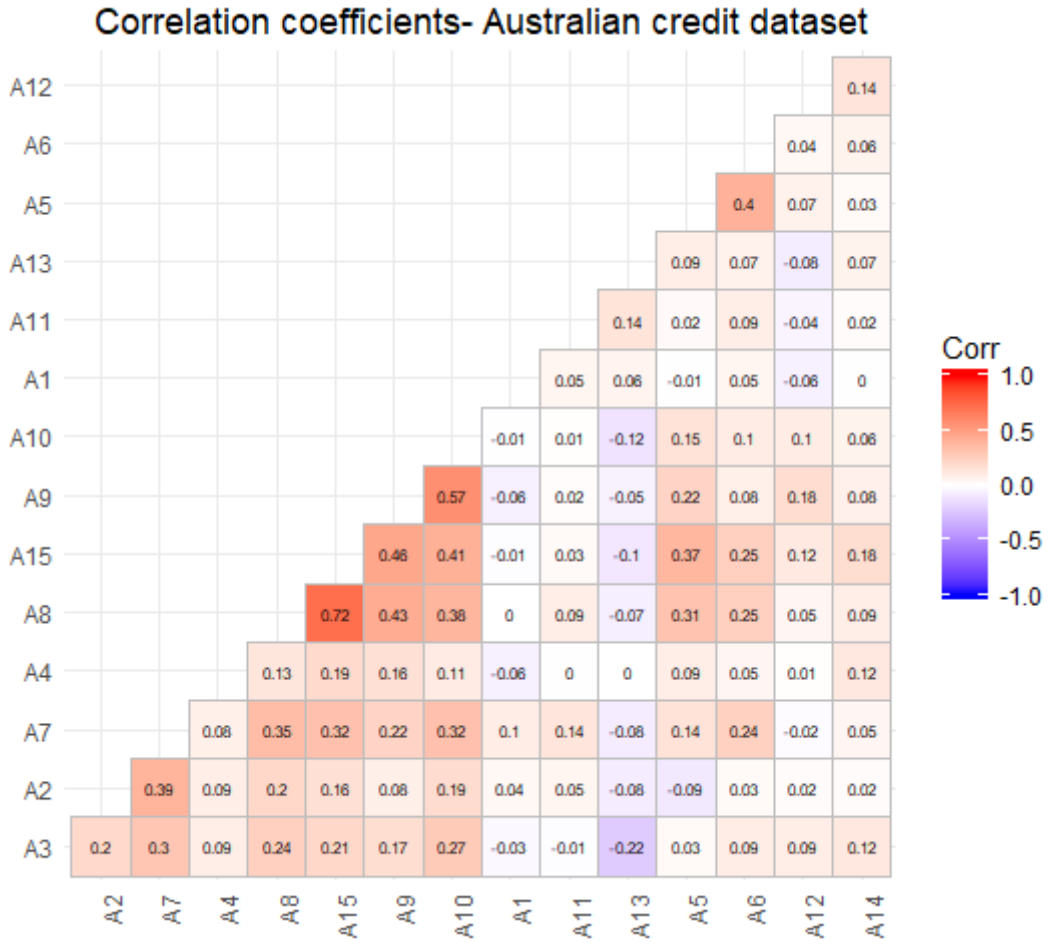

Figure 4-9: Pearson correlation coefficients - Australian credit dataset

We analyse that this dataset shows good positive correlation between many pairs of variables in dictated by darker shades of red and the corresponding correlation coefficient. Variable A8 and A15 show highest positive correlation of 0.72. A3 and A13 exhibit highest negative correlation of -0.22 (darker blue).

Below, we will analyse some of these variables in detail.

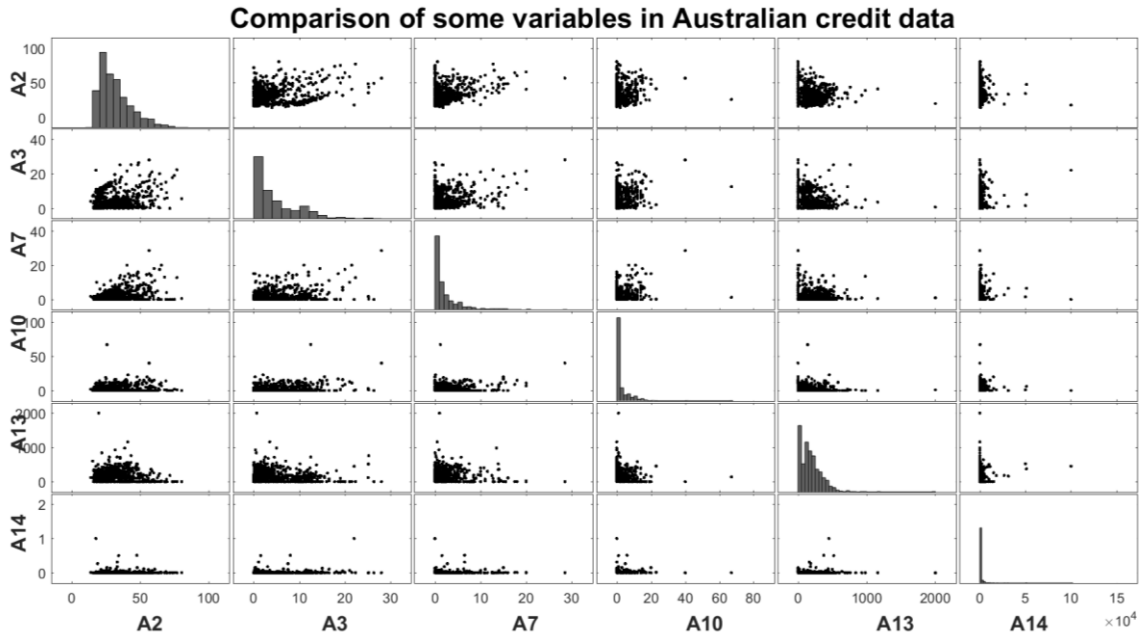

Figure 4-10: The correlated variables in Australian credit dataset

- The distribution of variables is skewed normal;
- There is no strong correlation between the pair of variables. However, the pattern created for all the variables A2-A14 above by the markers slant upward from the lower left corner towards upper right corner, fanning out from the origin. This upward slope conveys positive correlation.

Now we plot the same data showing class distribution. The two classes are shown by dots (class -1: bad credit) and crosses (class 1: good credit)

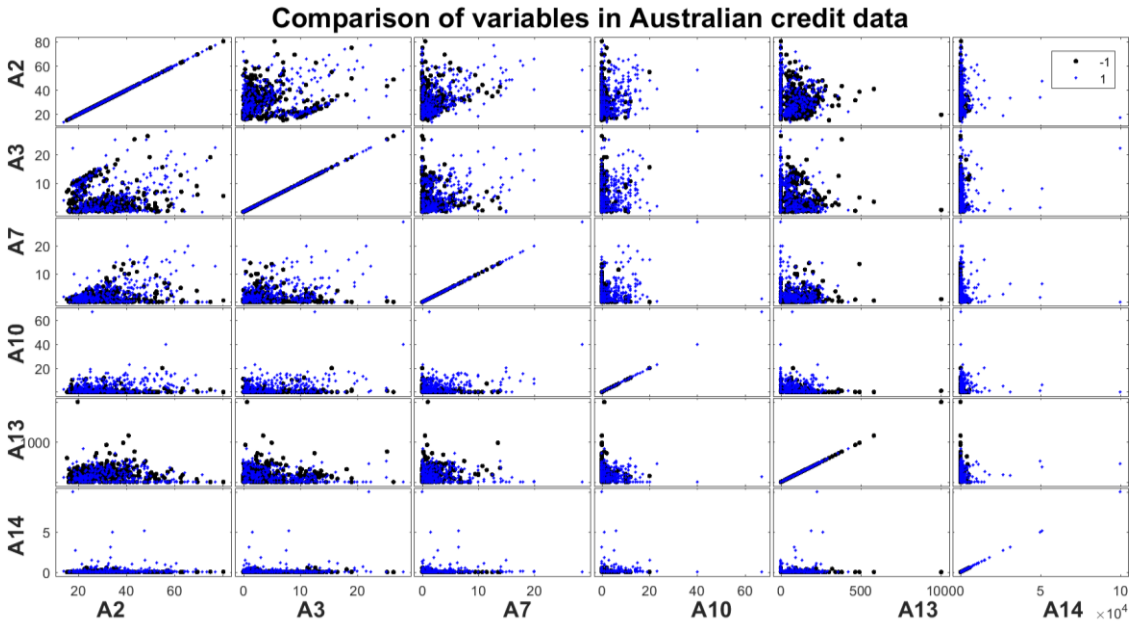

Figure 4-11: The correlated variables and the separability of classes in Australian credit dataset

- To investigate which variables from the Australian credit dataset are useful for predicting the response, various combinations of predictor variables are visualised on x and y axes above.
- Although none of the variables clearly separate the classes, it is observed that attributes A14, A13 as well as A14, A7 show some separability between the dots and crosses and could be useful in separating the two classes.

This could be seen in the two figures below.

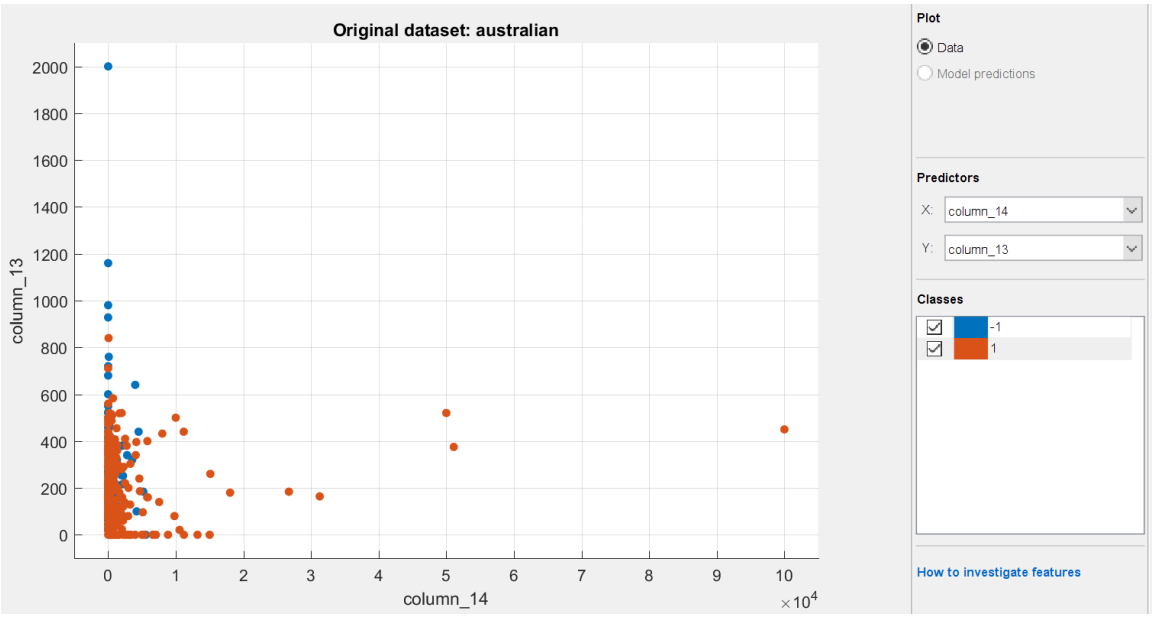

Figure 4-12: Scatterplot showing separability of classes using attributes 14 and 13 of Australian credit data

Figure 4-12 and Figure 4-13 show some level of positive correlation between the shown variables as well as separability between them w.r.t. the binary classes.

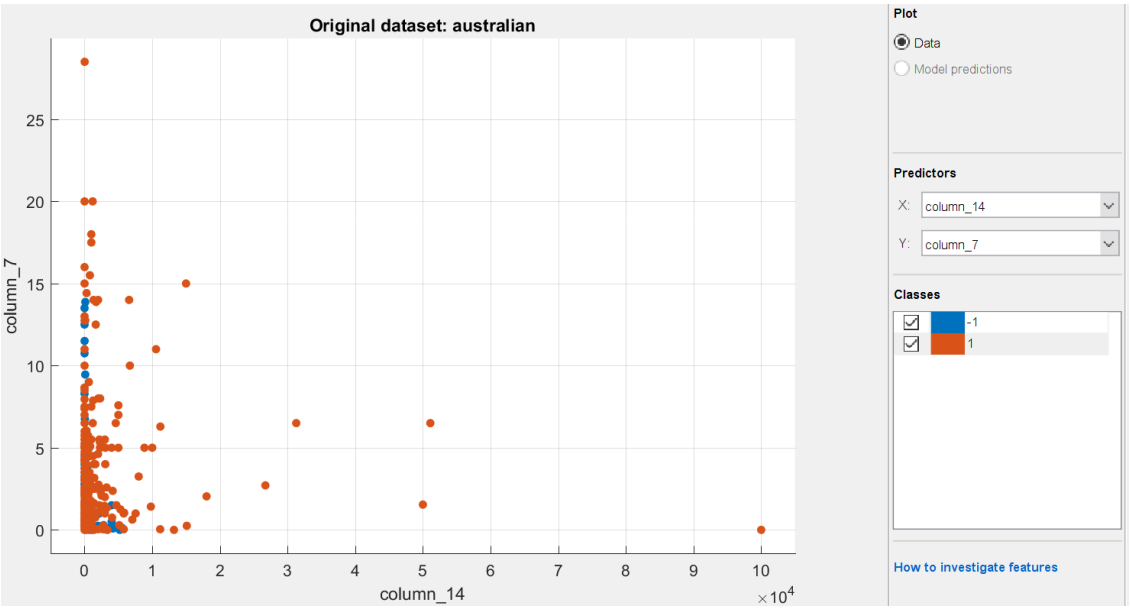

Figure 4-13: Scatterplot showing separability of classes using variables 14 and 7 of Australian credit data

Above two figures are an indication that variables 7, 13, 14 could be used to separate the credit applicants into two classes.

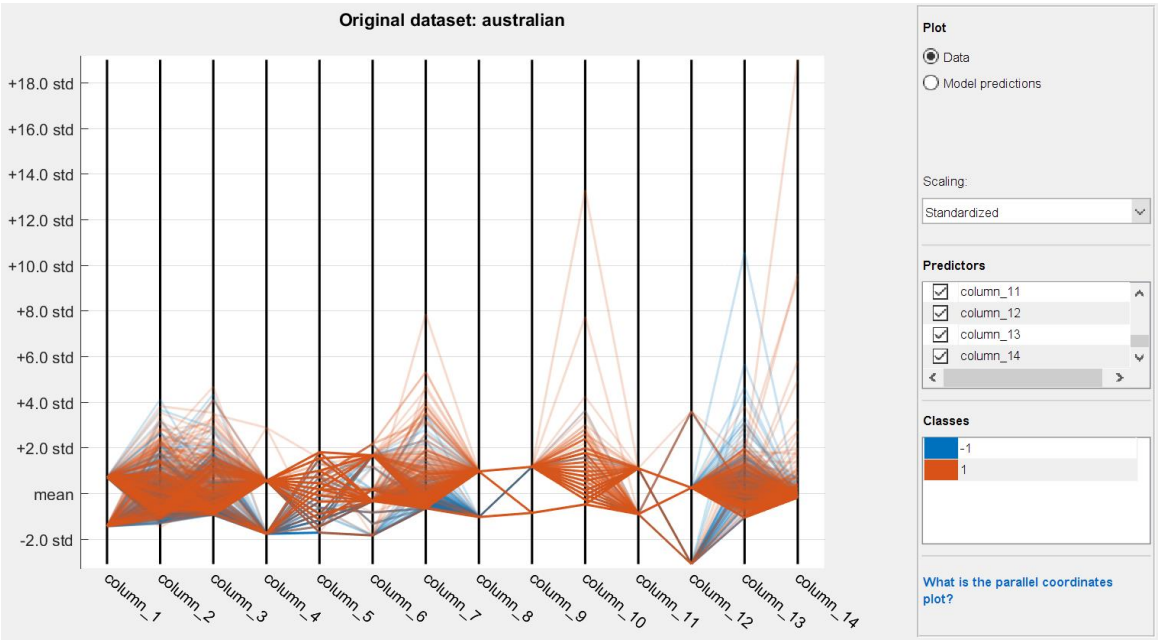

Figure 4-14: Parallel coordinates plot showing relation between all the variables in Australian credit data

This multivariate dataset could be analysed from the figure above. We can see that columns 1, 4, 8, 9, 11, 12 have fewer distinct values as compared with other variables.

4.2.4.3 Taiwan credit dataset

The figure below shows scatter plot matrix of all variables in this dataset. We can determine whether the variables are correlated and also the type of correlation i.e. positive or negative. It shows pair-wise relationship (non-diagonal cells) and distributions (diagonal cells) of variables in the data.

The figure below shows the matrix of pair-wise correlations for this dataset using a colour gradient. The heatmap shows the correlations with deeper colours indicating higher positive (red) or negative (blue) correlations. Many variable pairs exhibit dark shades of red and blue. This dataset exhibits a strong linearity.

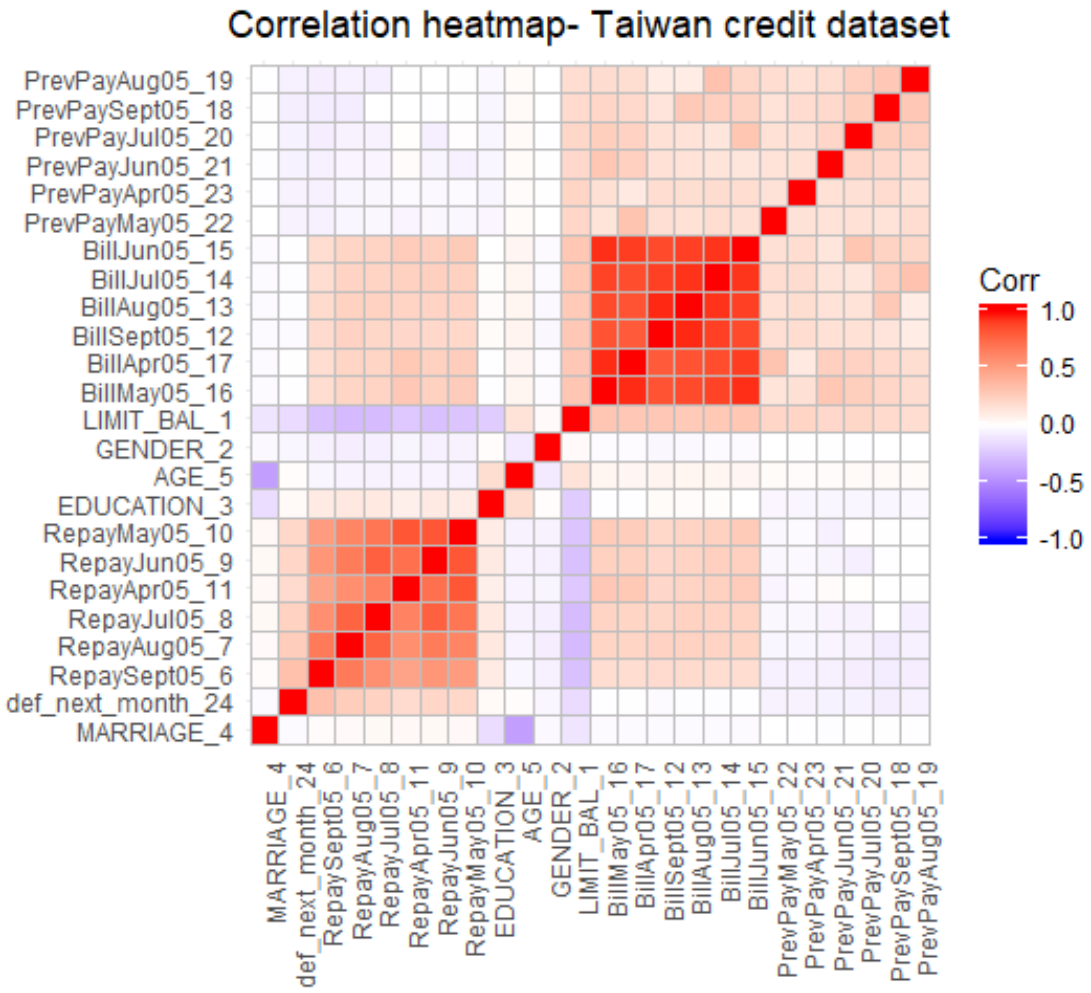

Figure 4-15: Correlation heatmap - Taiwan Credit dataset

Figure 4-16 also depicts the values of correlation coefficients. Many variables in this dataset are strongly correlated. The dataset comprises Repayment-status and Amount-of-bill-statement in 5 months period. There is a strong correlation among them. We conclude that this dataset is strongly linear.

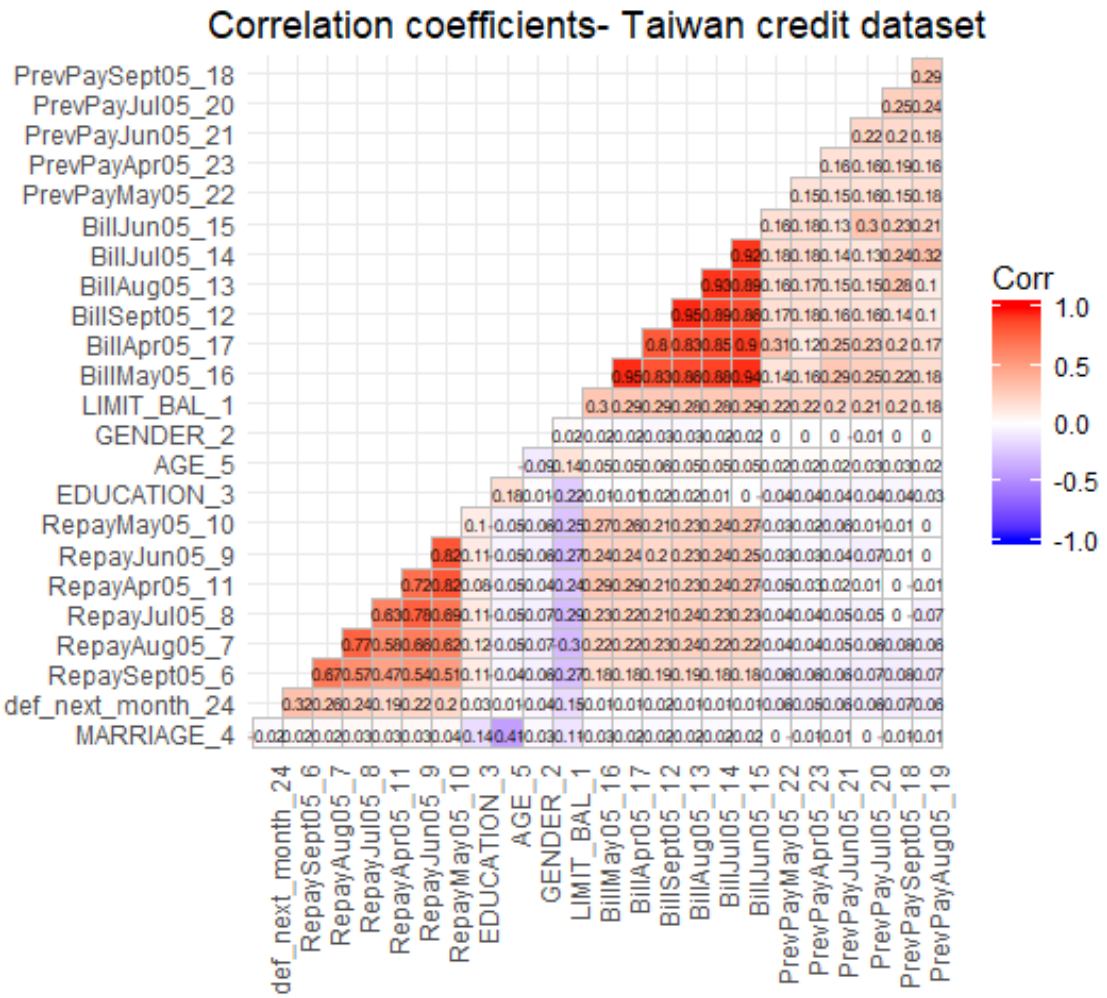

Figure 4-16: Pearson correlation coefficients - Taiwan credit dataset

Figure 4-17 shows some of the correlated variables and their distribution within the dataset.

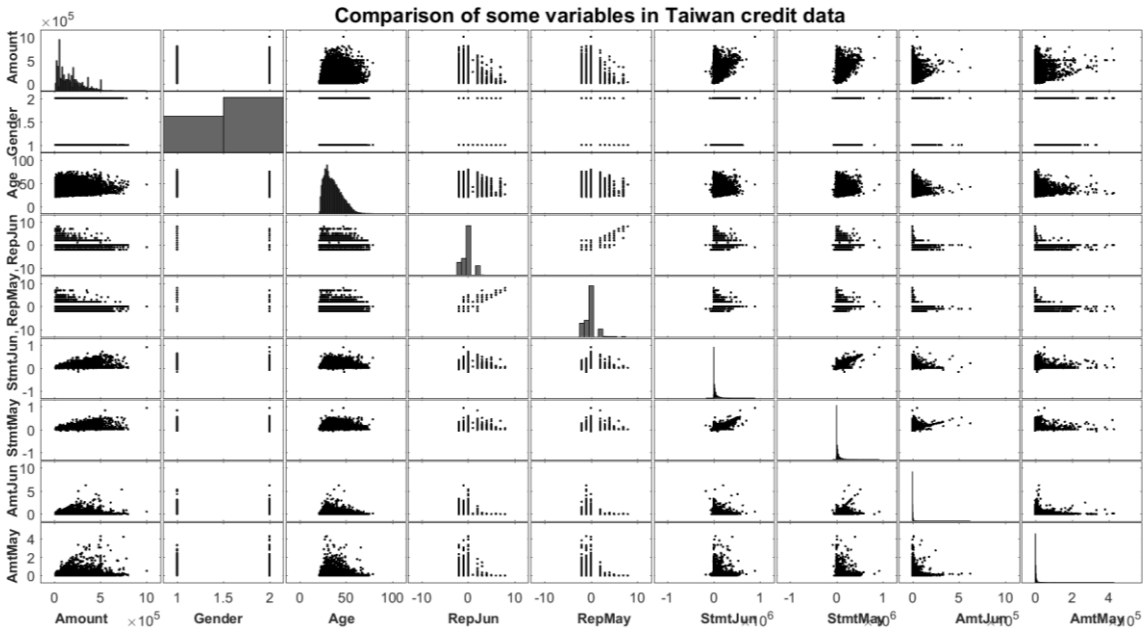

Figure 4-17: The correlated variables in Taiwan credit dataset

God positive correlation can be observed between few pairs of variables from the Taiwan credit dataset.

We will plot these variables to determine their usefulness towards prediction of the class of the applicant. We will use these same attributes as predictors for classification models in next section.

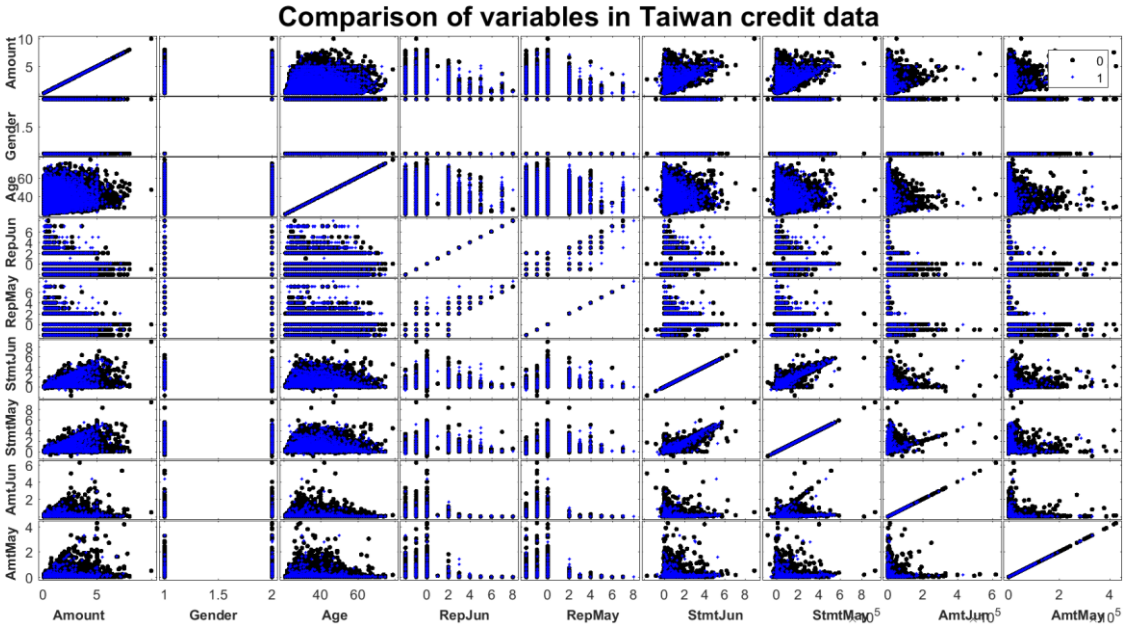

Figure 4-18: The correlated variables and the separability of classes in Taiwan credit dataset

From Figure 4-18, there are quite a few variables which could separate the two classes very well shown by two different colour levels.

Let us look at some of these critical variables individually. The figure below shows relation between ‘Age of the credit client’ and the ‘Bill Statement he received in May 2005’.

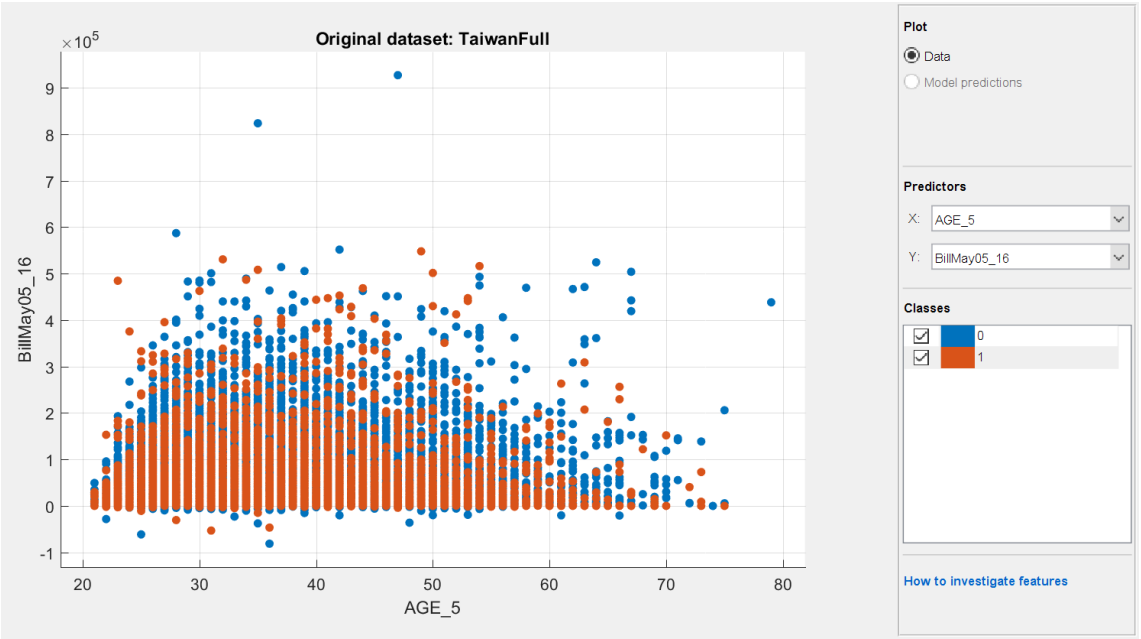

Figure 4-19: Scatterplot showing separability of classes using attributes 'Age' and 'Bill Statement in May 2005' of Taiwan credit dataset

Amount of bill statement increases with age from 20 to 45, but decreases afterwards. This was observed for all the months in the year 2005. This could mean that middle age clients have obtained larger credits and hence the bill statements too are large, whereas young and senior clients have smaller credits to their accounts. We can also spot some outliers in the data, which may need special attention.

Similarly, 'Age' and the 'Payment made in May 2005' has relationship between them as can be seen below.

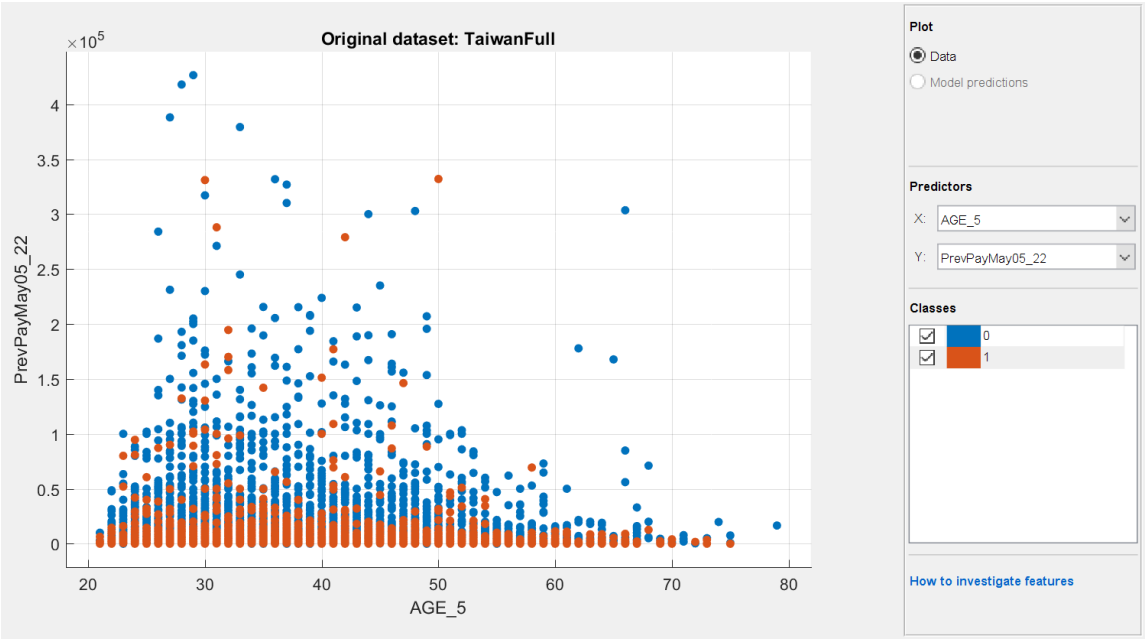

Figure 4-20: Scatterplot showing separability of classes using variables 'Age' and 'Previous Payment in May 2005' of Taiwan credit dataset

Figure 4-20 indicates that the amount paid in September 2005 increases as the 'Age' increases from 20 to 45, but it decreases from age 45 to 80. This is similar to the trend observed for amount of bill statement. This could be a good indicator of the payment capability of clients at different ages. This trend is observed for all the months in year 2005.

Similar trend is observed in the figure below for Age (x-axis) vs. Amount of the given credit (y-axis). Interestingly there are two outliers at age 47 (with amount of credit given equal to 1000000 New Taiwan Dollar) and age 79 (with amount of credit given equal to 440000 New Taiwan Dollar) as shown by blue dots.

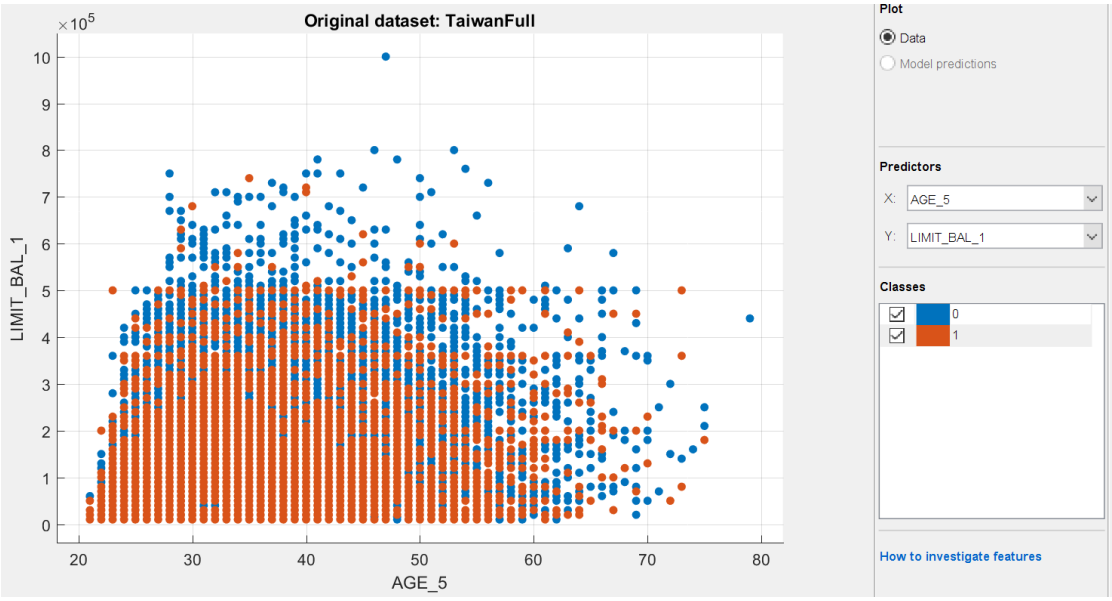

Figure 4-21:Scatterplot showing separability of classes using attributes 'Age' and 'Amount of the given credit' of Taiwan credit dataset

The Bill statement received in a month and the Payment made in that month shows some positive relation as shown below.

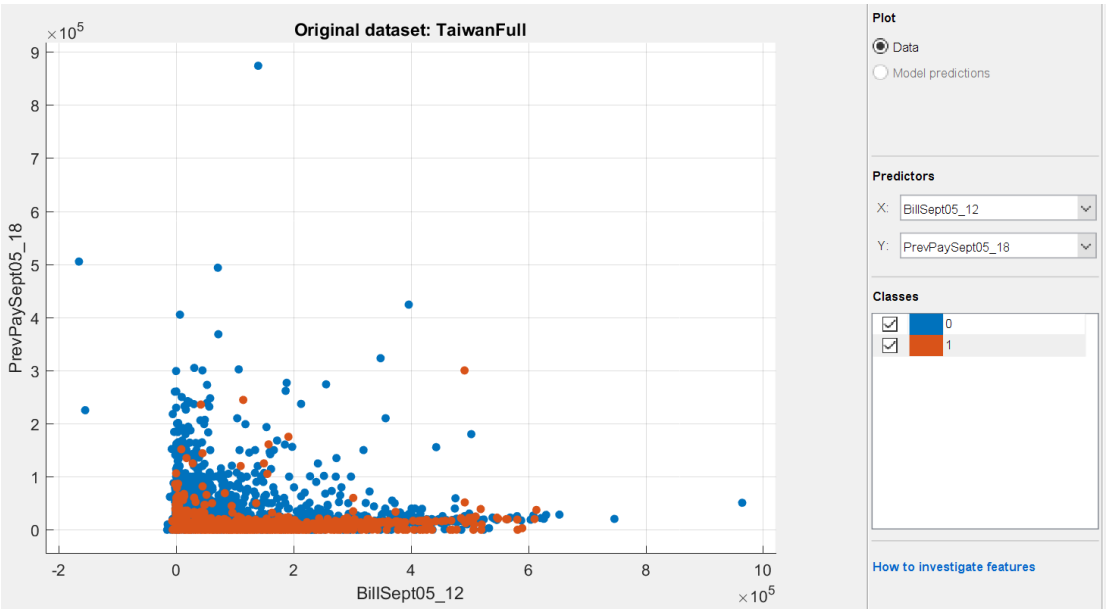

Figure 4-22: Scatterplot showing separability of classes using attributes 'Amount of bill statement in September 2005' and 'Amount paid in September 2005' of Taiwan credit dataset

From the figure above, barring few due to negative values on x-axis, mostly heteroskedastic relation is seen between ‘Amount of bill statement in September 2005’ on x and ‘Amount paid in September 2005’ (y-axis) values. This means that the clients paid larger amounts when they received larger bills in that period. This trend was observed for the rest of the bill statements and payments made in the year 2005.

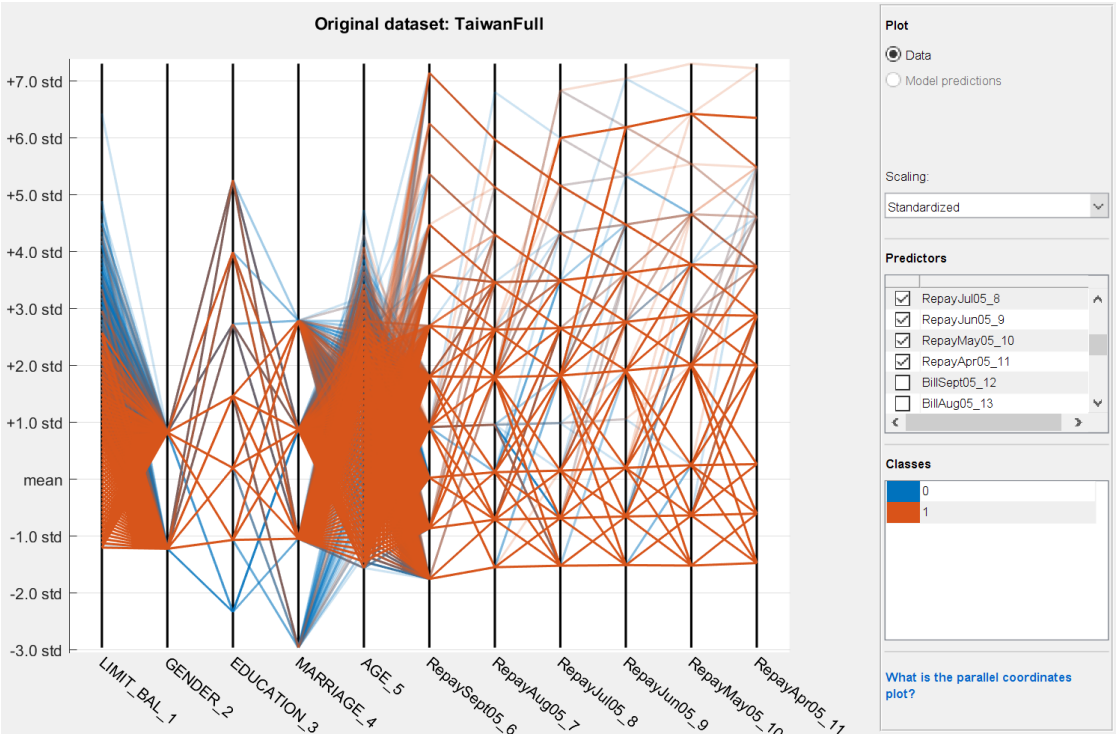

Figure 4-23: Parallel coordinates plot showing relationships between features in Taiwan credit data

From the figure above, ‘Limiting Balance’ (i.e. Amount of the given credit) and ‘Gender’ are the features that separate the classes best, hence these could be useful predictors for predicting the classes of clients.

4.3 The Industry dataset

The real dataset used in this study is provided by FactSet Research Systems Inc. [13] The dataset contains data for companies being traded in stock market. The nature of the data is Estimates, each value is an estimate made by "the market" on what the value of a specific piece of information about a given company will

be at a time in the future. The values get more accurate and closer to the real value that the company ends up reporting as time passes, thus the last value for a given data item could be considered to be the actual value the company reported. The data is either Earnings Per Share (EPS) or Sales. Further, for each of these categories, the data is estimated either for quarters or years.

Table 4-5: Description of the industry dataset

Feature number	Feature name
1	Company identifier
2	The "fact" that this row contains data for: EPS/Sales
3	If the data is Quarterly/Annual
4	Fiscal period for which the data value is relevant
5	Date when the value was created
6	Actual EPS/Sales value

Basic EPS of a company is the earnings available to common shareholders divided by the weighted average number of common shares outstanding. This measure is used to determine the strength of a company's stock. Each quarter, a company releases its EPS. Before that, a stock trader will have a look at the estimates. Many investors believe that stock prices react favourably or negatively to differences between analyst EPS estimates and actual performance. If the actual EPS beats the estimates, the company is performing well, else not and the stock will drop. Also by looking at the past EPS, stock traders decide if the company's stock is worth buying or not. Also, sales of a company are closely related to the EPS. The remaining of this chapter analyses the EPS, Sales and compares both for each of the six companies.

4.3.1 Data Preprocessing

This dataset needs to be preprocessed in order to make it ready to analyse.

4.3.1.1 Data sampling

The private dataset used in this study is a big dataset. Large amounts of data afford simple models much more power since it leads to better decisions. More

data can definitely make predictive models more accurate and more general. On the other hand, more data can be time-consuming to use. More data better represents the population but at the same time the data needs to be just enough for the computation to happen in a reasonable amount of time.

To see the justification statistically, consider a sample $X = \{x_1, x_2, \ldots, x_n\}$ of size n. Estimate of the mean of the population μ could be computed with the estimator $\mu' = \frac{1}{n} . \sum_{i=1}^{n} x_i$. The standard error of this estimator is inversely proportional to n:

$$Standard\ error = \frac{Standard\ Deviation}{\sqrt{n}} \quad \textbf{(4-2)}$$

More n implies lower standard error, hence better model. Hence, for analysis, we select a subset of the data about the six companies with largest available number of observations.

4.3.1.2 Data Rescaling

The range of values for the EPS and Sales data for a company varies widely. We normalise the range of these variables so that each feature contributes approximately proportionately when analysing the data.

We rescale the features to normalise in the range [0, 1]. The formula is given as [194]:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \quad \textbf{(4-3)}$$

where x is an original value and x' is the normalised value.

4.3.2 Analysis of the EPS data

Earnings Per Share(EPS) is the most widely used ratio and communicates how much profit is generated on a per-share basis. EPS is the profit a company earns from a single stock available in the market. It reveals a lot about the financial health of a company. Increasing EPS is a very good sign for a company. A stock price has a tendency to decline if a company fails to meet analyst estimates for EPS. Hence, EPS useful while choosing stocks from a bunch of stocks. Investors

could use the plots to keep a close look at the EPS of the companies in its quarterly or yearly results.

In this section, we perform EPS analysis for the six chosen companies with identifiers 1290, 3180, 11217, 14324, 18965 and 29642.

- How the EPS for each company varies over a period;
- How the average quarterly EPS varies over a period;
- How the average yearly EPS varies over a period.

4.3.2.1 Company 1290

We observe how the company's performance fluctuated over a period in the following plot.

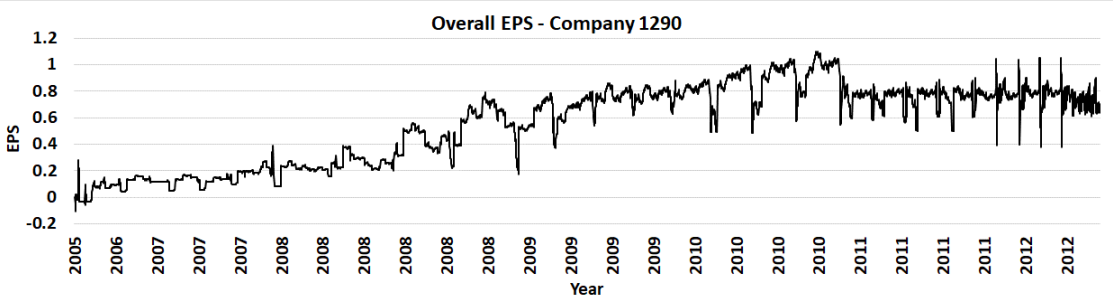

Figure 4-24: Company 1290 - Overall EPS

The figure above shows sample number on X axis. We can see that this company's initial EPS estimate are negative. The reason could be that the historical growth rate in earnings for this company were negative (meaning company was at loss) and hence the estimates for expected growth too were low. Many companies have negative earnings at initial growth stage which impact their retained earnings. Overall, the company's EPS estimates show peak at the end of year 2010. From year 2011 onwards, it fluctuates, but around the same mean.

It would be interesting to see how the actual EPS vs. the estimated EPS differ. Since the dataset does not have actual EPS data, and has the Sales data for the same companies, in section 4.3.4, we compare these two.

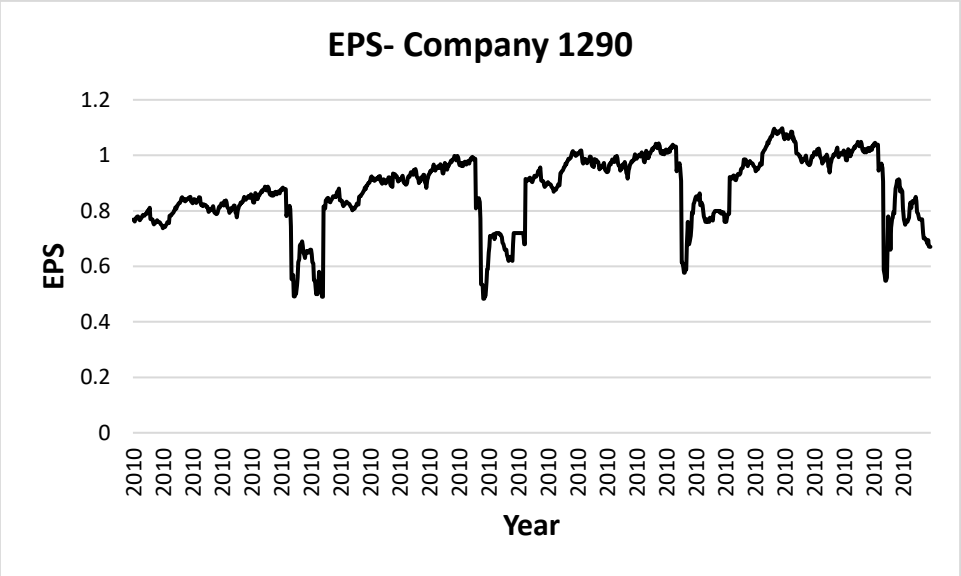

Figure 4-25: A closer look at EPS in year 2010 for company 1290

E.g., if we look at the EPS values in year 2010 more closely, Figure 4-25 shows that a pattern has started to emerge for estimated EPS. We observe that the estimates have forecast steady growth (upward trend), but also the growth is trimmed after each time period, hence the downward dips. This revision in the estimates of EPS could be because the analysts found out that the company may not perform well and hence revised the estimated EPS for the company.

The stock market is forward looking. That is, stock prices are established based on the expectations that prospective investors have for the future earnings power of the company [195]. It may happen that even when a company announces increased earnings, the company's stock price falls, or vice-versa. What this means is that the actual earnings did not turn out as the market expected. Thus, expectations of the investors play an important role in determining if a stock's price increases or decreases when actual earnings are published. Stock prices adjust as the expectations change or are proven wrong. Estimated EPS represents these expectations and current price reflects the estimates.

The quarterly details for company 1290 could be observed in the figure below.

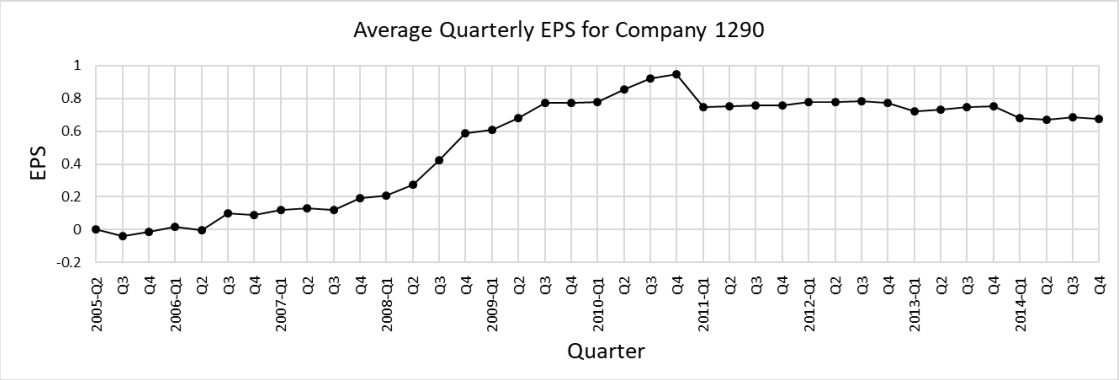

Figure 4-26: Company 1290 - Average quarterly EPS

The figure above could be used to observe the quarterly details. The average quarterly EPS for this company peaks in quarter 4 of the year 2010.

Following figure allows to see the yearly trend of EPS for this company, without the minute details as in figure above.

EPS is the resultant profit after all taxes and depreciation over a period of time - which could be yearly or quarterly- that is receivable for a shareholder holding a single share of that company. Quarterly EPS estimates are useful for investors as well as the company to evaluate it whenever required. The investors can use the quarterly values to extrapolate the yearly EPS for the company instead of having to wait for year-end EPS.

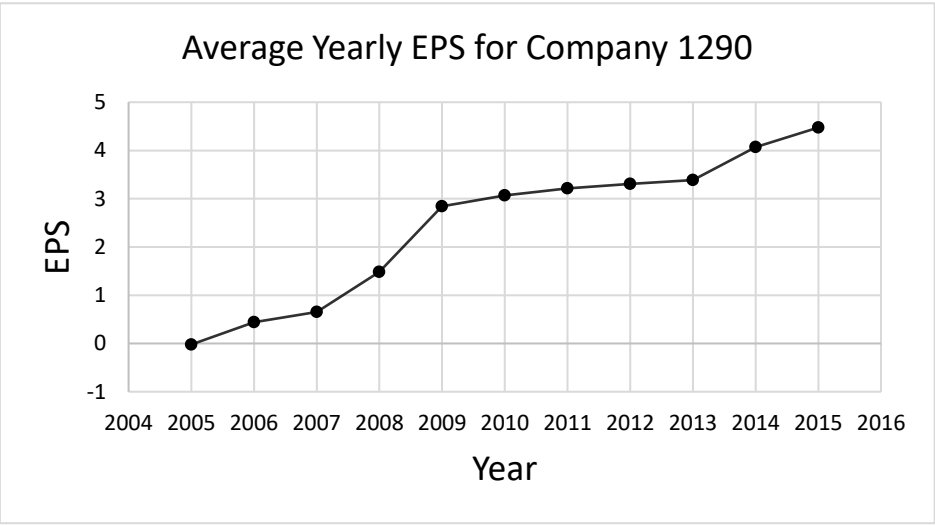

Figure 4-27: Company 1290 - Average yearly EPS

From Figure 4-27 above, this company's average yearly EPS clearly shows upward trend.

4.3.2.2 Company 3180

For this company, the estimates show very short bursts in pattern. This indicates that the estimates are adjusted quite frequently.

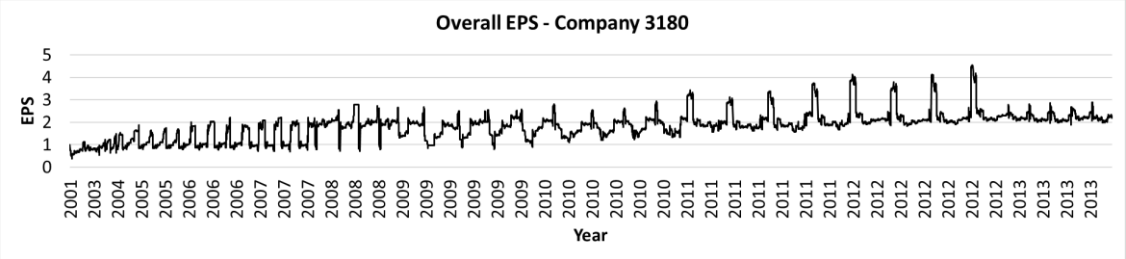

Figure 4-28: Company 3180 - Overall EPS

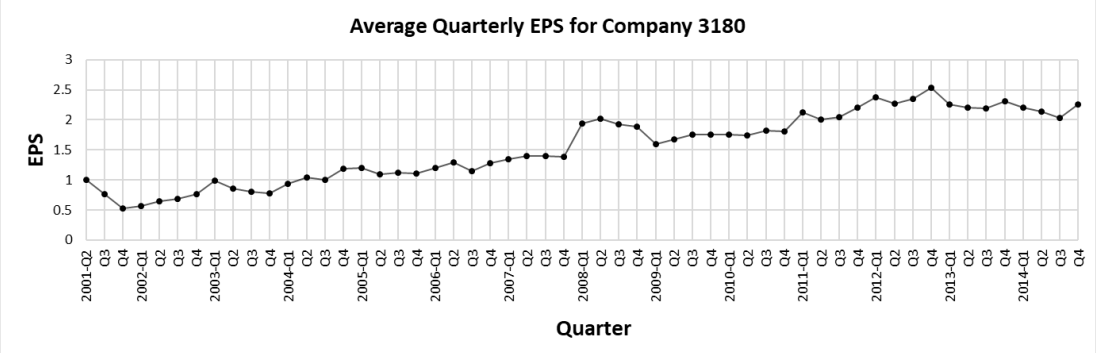

Figure 4-29: Company 3180 - Average quarterly EPS

From Figure 4-29, for this company, the estimated EPS peaked at quarter 4 of 2012.

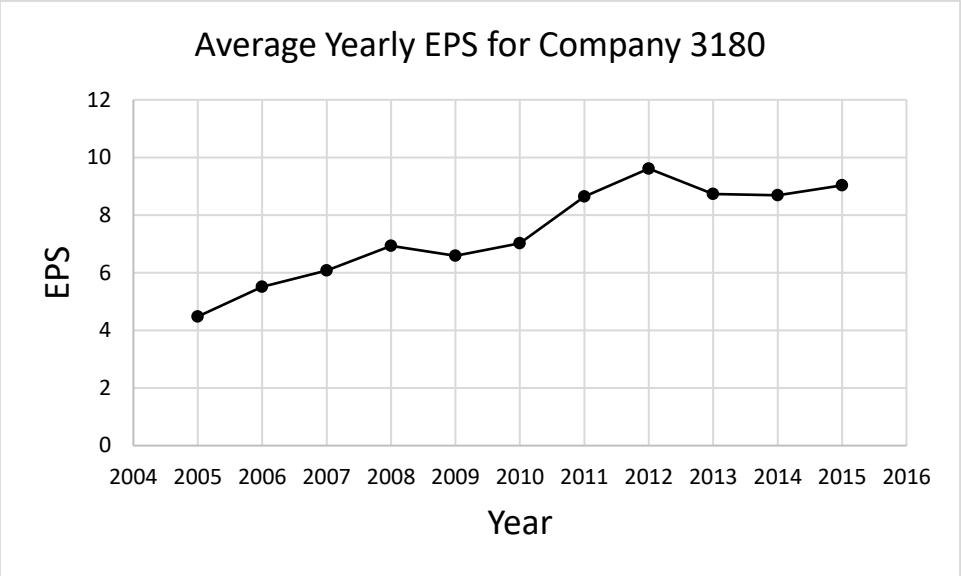

Figure 4-30: Company 3180 - Average yearly EPS

From Figure 4-30, for this company, the estimated EPS peaked twice in the recorded period: 2008 and 2012.

4.3.2.3 Company 11217

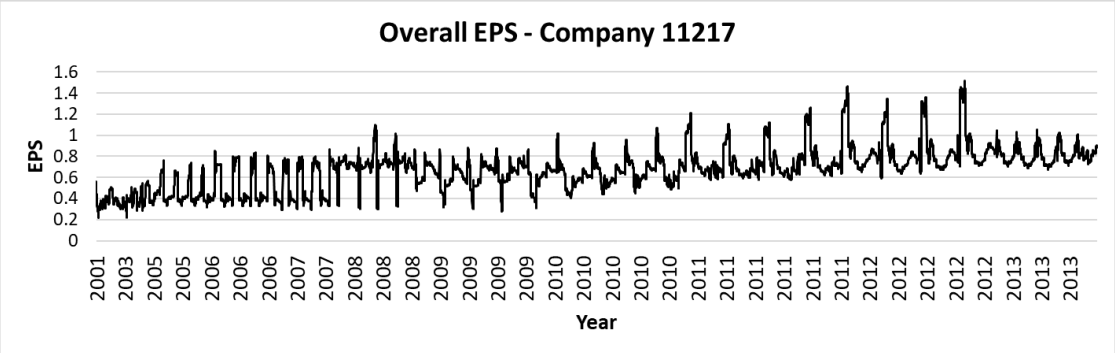

Figure 4-31: Company 11217 - Overall EPS

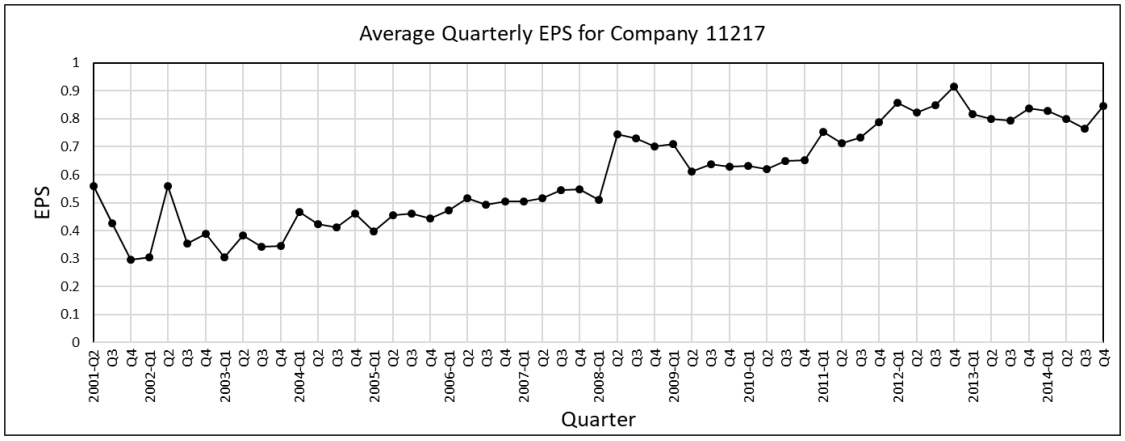

Figure 4-32: Company 11217 - Average quarterly EPS

From the figure above, in the initial period, this company's EPS estimates dipped and also, in later periods, there are good number of big fluctuations. But the overall trend is upward as shown in following figure.

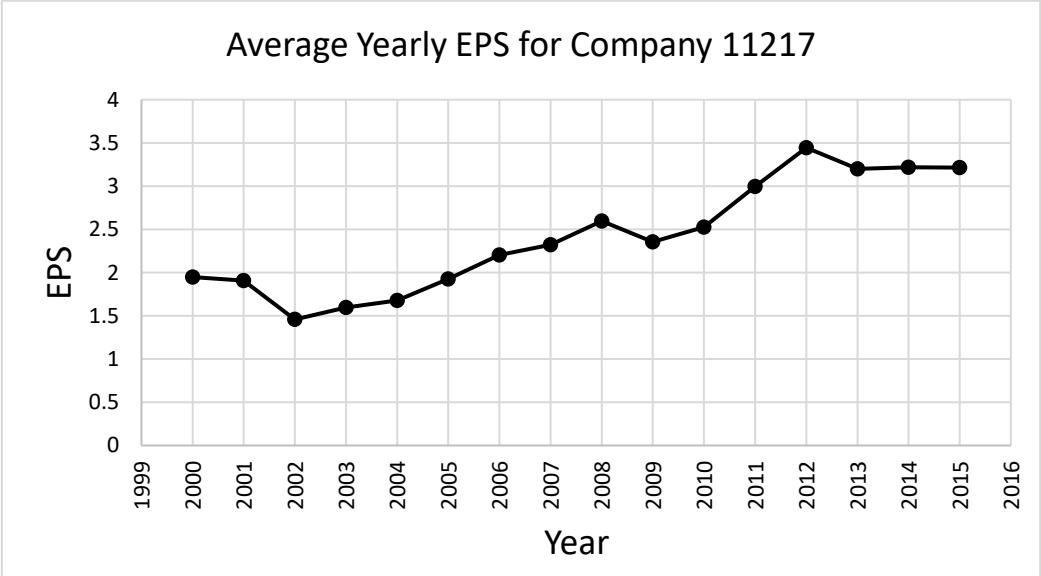

Figure 4-33: Company 11217 - Average yearly EPS

4.3.2.4 Company 14324

For this company, the figure below indicates the adjustments done in the estimates at the end of each year, i.e. big drops and subsequent high estimates in the EPS.

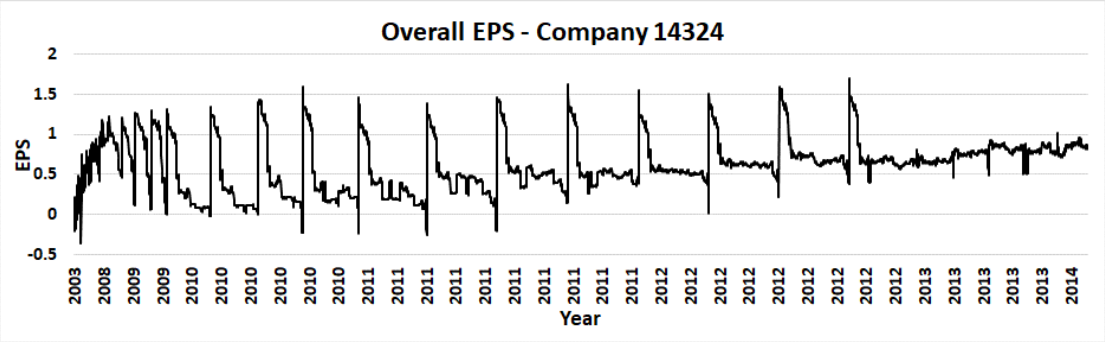

Figure 4-34: Company 14324 - Overall EPS

But the quarterly estimates in the figure below show the biggest drop in the first quarter of 2010, which also affects its overall trend (Figure 4-36).

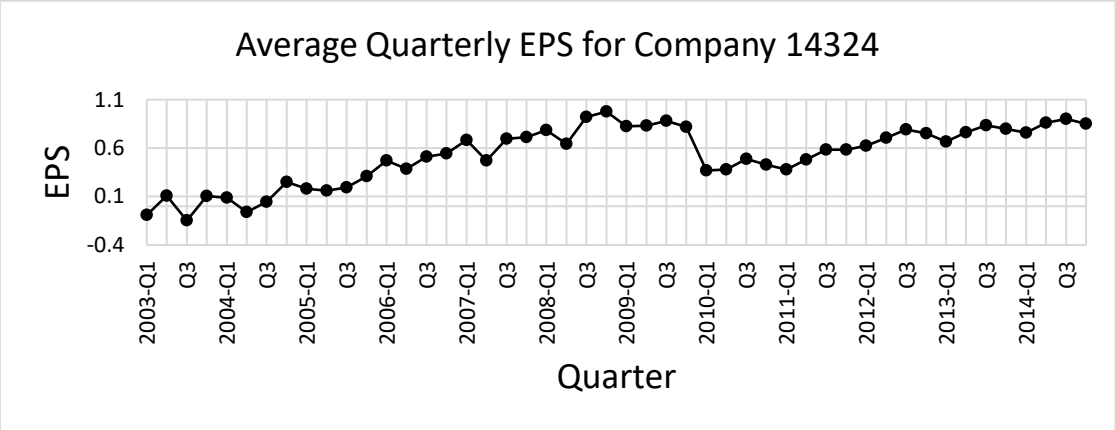

Figure 4-35: Company 14324 - Average quarterly EPS

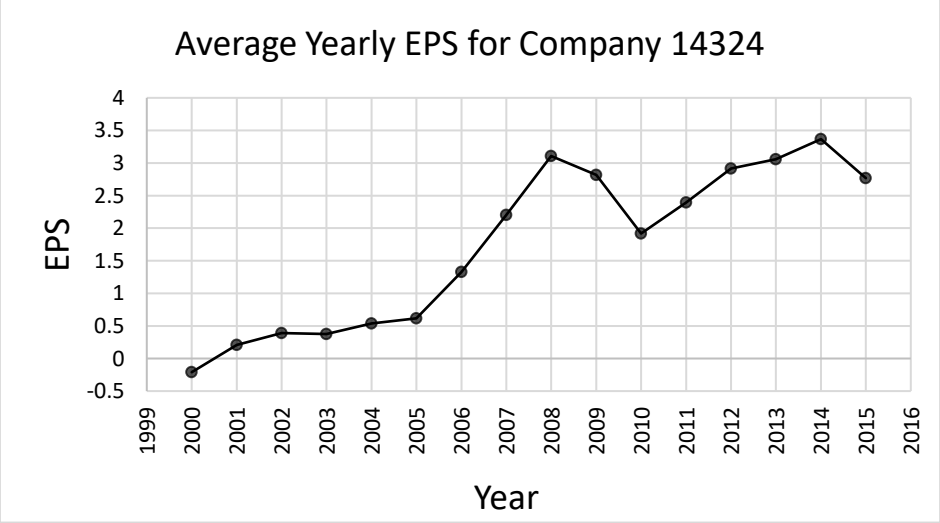

Figure 4-36: Company 14324 - Average yearly EPS

Similarly, we observe for the remaining companies in next two subsections.

4.3.2.5 Company 18965

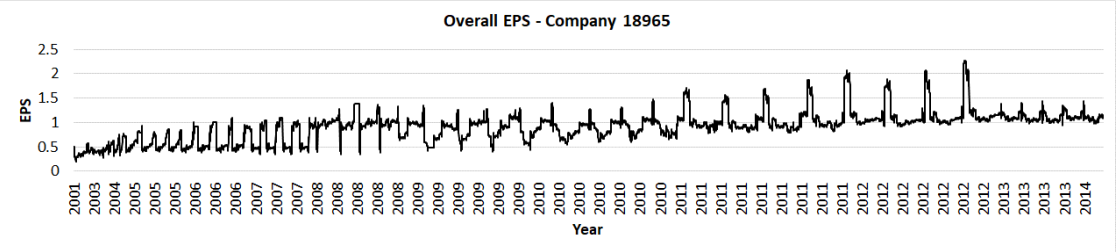

Figure 4-37: Company 18965 - Overall EPS

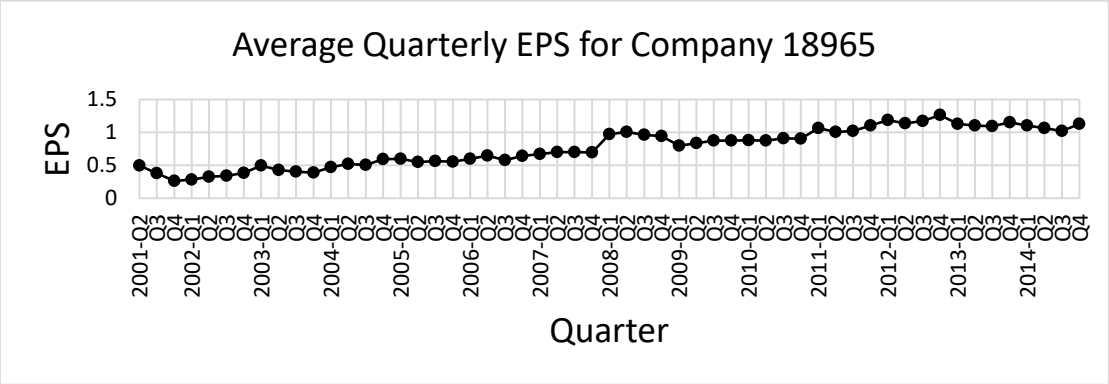

Figure 4-38: Company 18965 - Average quarterly EPS

From the figure above, this company's EPS estimates decrease in the beginning. There is a big growth in EPS in the period from fourth quarter of 2007 until first quarter of 2009. The quarterly analysis helps identify the peaks and valleys in the EPS.

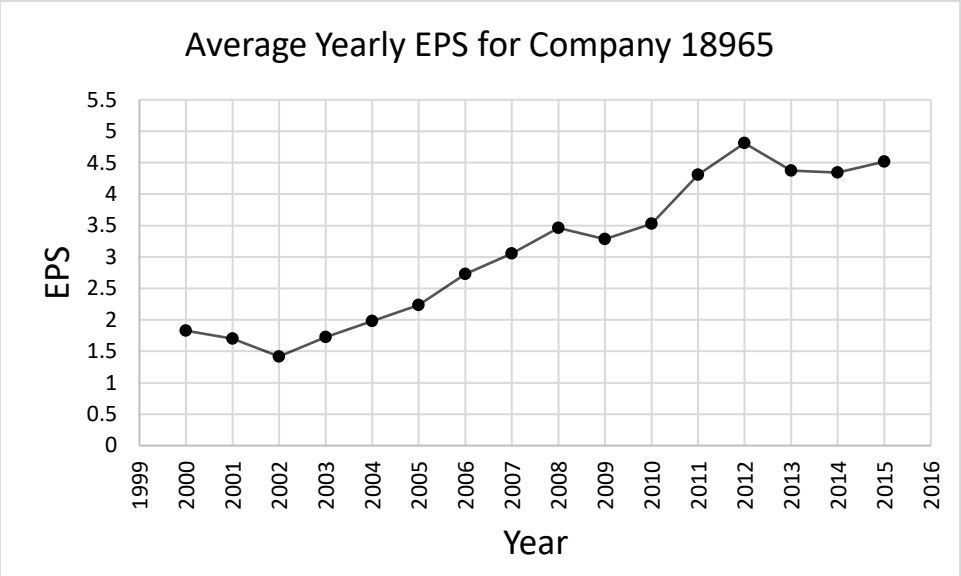

Figure 4-39: Company 18965 - Average yearly EPS

The yearly EPS plot for this company helps identify the overall EPS trend. This company has highest EPS in the year 2012 which is consistent with the findings of Figure 4-38.

4.3.2.6 Company 29642

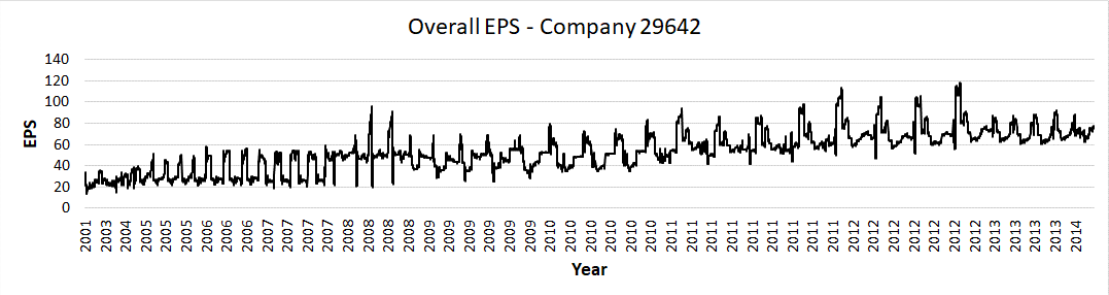

Figure 4-40: Company 29642 - Overall EPS

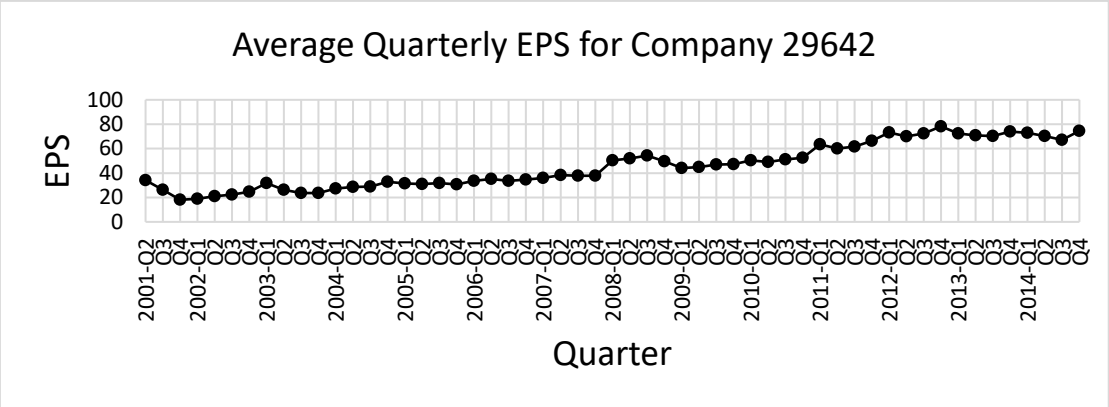

Figure 4-41: Company 29642 - Average quarterly EPS

The EPS of this company has steadily grown from Q4 of 2001 until Q4 of 2007 on average, but a sudden increase is seen until Q3 of 2008.

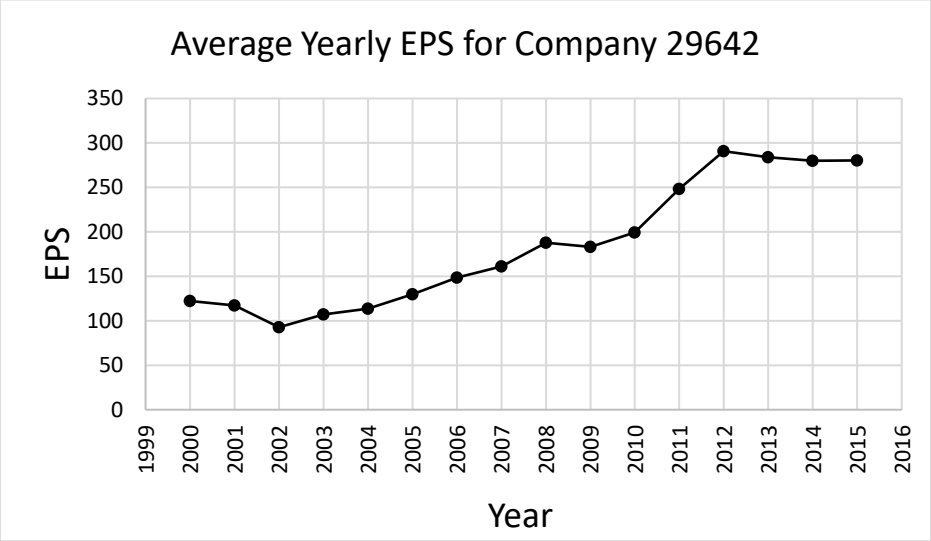

Figure 4-42: Company 29642 - Average yearly EPS

Figure 4-42 shows that the lowest EPS was in year 2002.

4.3.3 Analysis of Sales data

In this section, we perform Sales analysis for the six chosen companies with identifiers 1290, 3180, 11217, 14324, 18965 and 29642.

- How the Sales for each company varies over a period;
- How the average quarterly Sales varies over a period;
- How the average yearly Sales varies over a period.

4.3.3.1 Company 1290

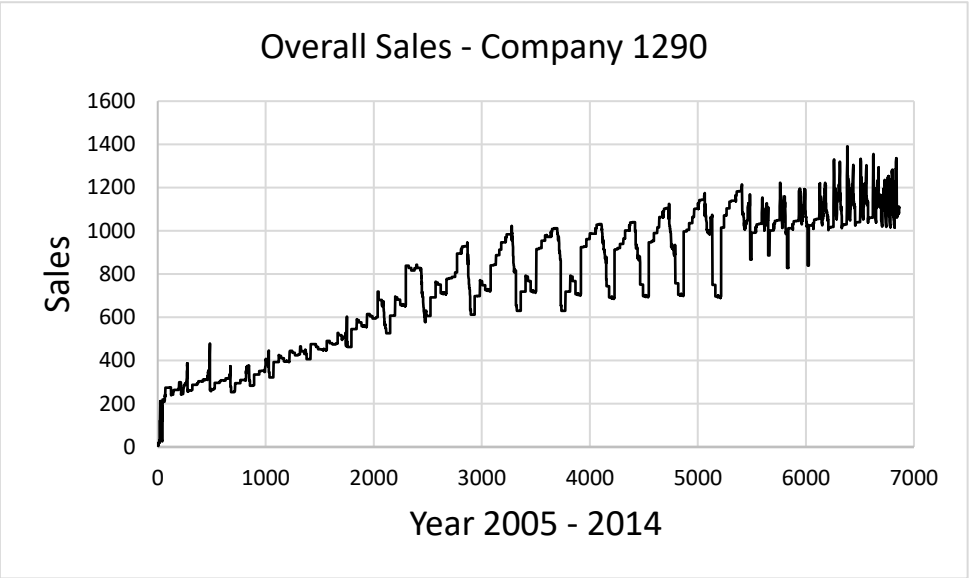

Figure 4-43: Company 1290 - Overall quarterly Sales

For this company, unlike EPS estimates, the sales estimates are smooth line as seen below.

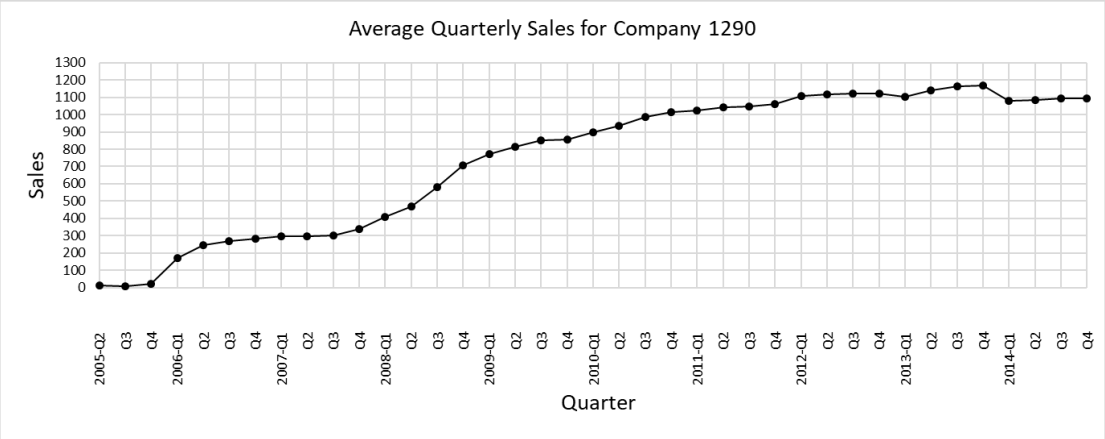

Figure 4-44: Company 1290 - Average quarterly Sales

Figure 4-45: Company 1290 - Average yearly Sales

From the figure above, the sales estimates have a very good smooth trend upward.

4.3.3.2 Company 3180

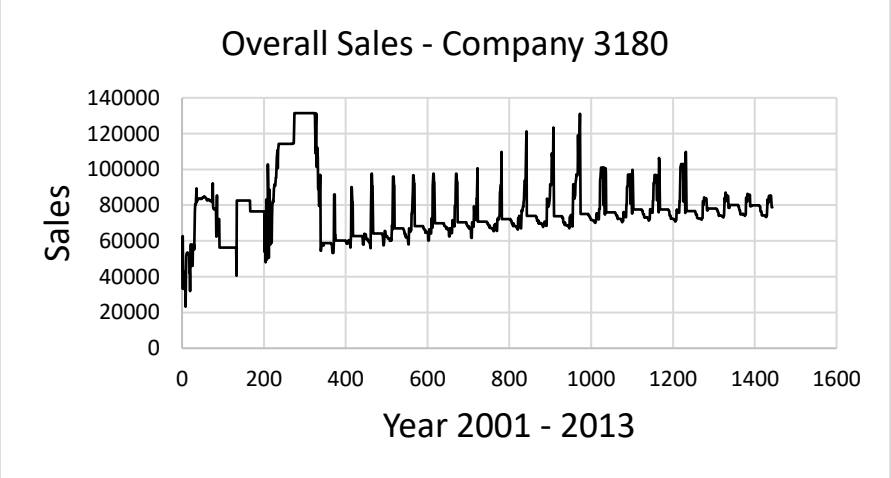

Figure 4-46: Company 3180 - Overall quarterly Sales

For this company, the overall Sales estimates show some constant values

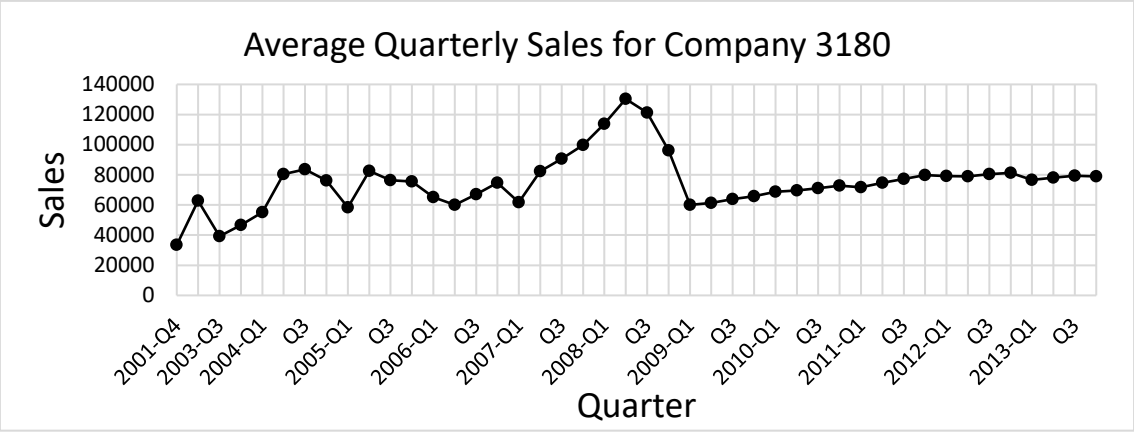

Figure 4-47: Company 3180 - Average quarterly Sales

The figure above indicates that the quarterly Sales is plateaued between year 2009 and 2013.

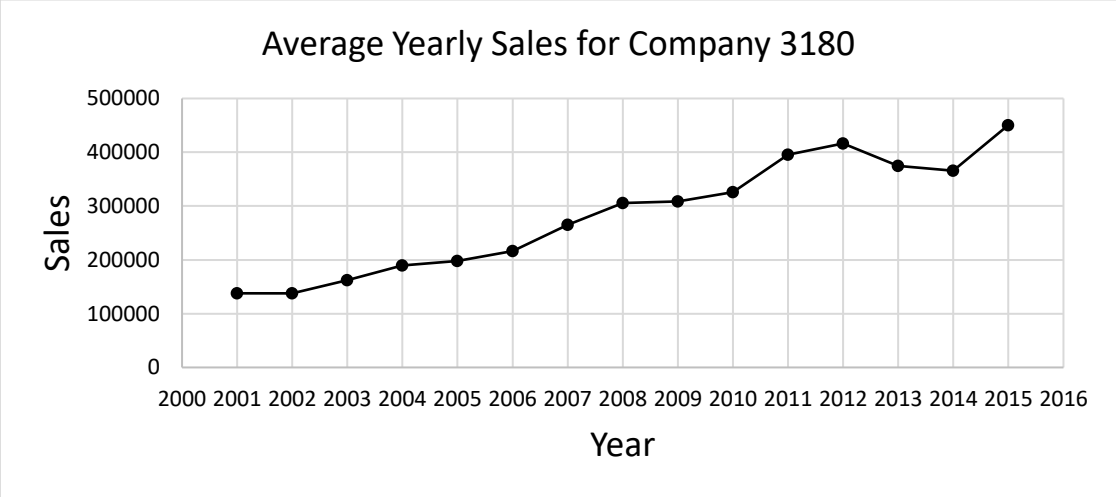

Figure 4-48: Company 3180 - Average yearly Sales

Above, if compared with other companies, this company's EPS growth is quite steady without many sudden peaks or valleys.

4.3.3.3 Company 11217

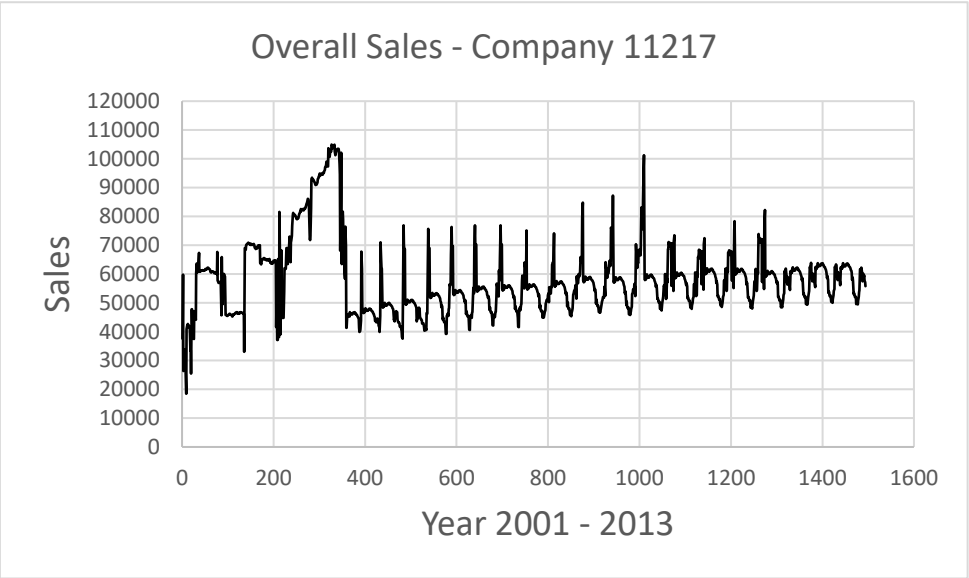

Figure 4-49: Company 11217 - Overall quarterly Sales

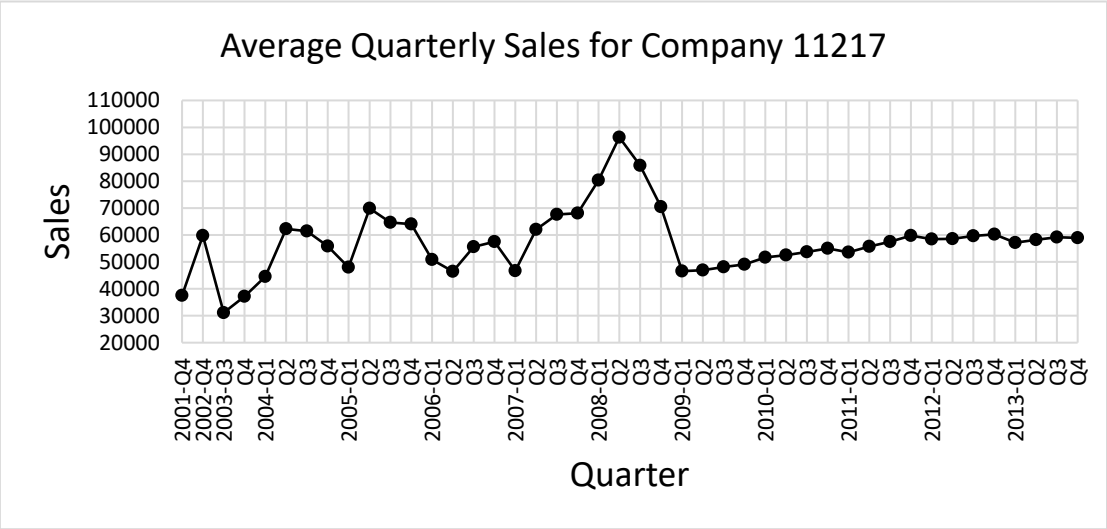

Figure 4-50: Company 11217 - Average quarterly Sales

The figure above indicates that the quarterly Sales is plateaued between year 2009 and 2013.

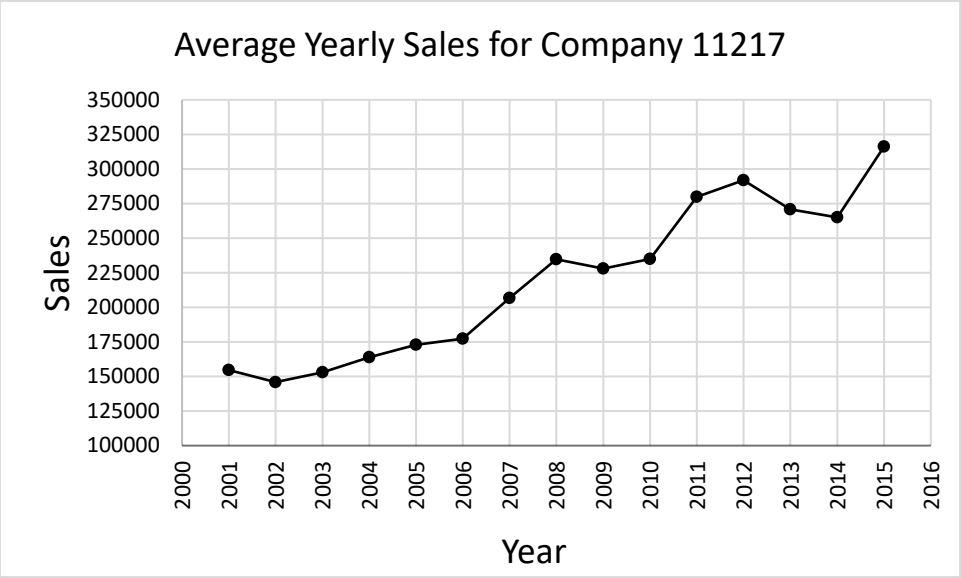

Figure 4-51: Company 11217 - Average yearly Sales

4.3.3.4 Company 14324

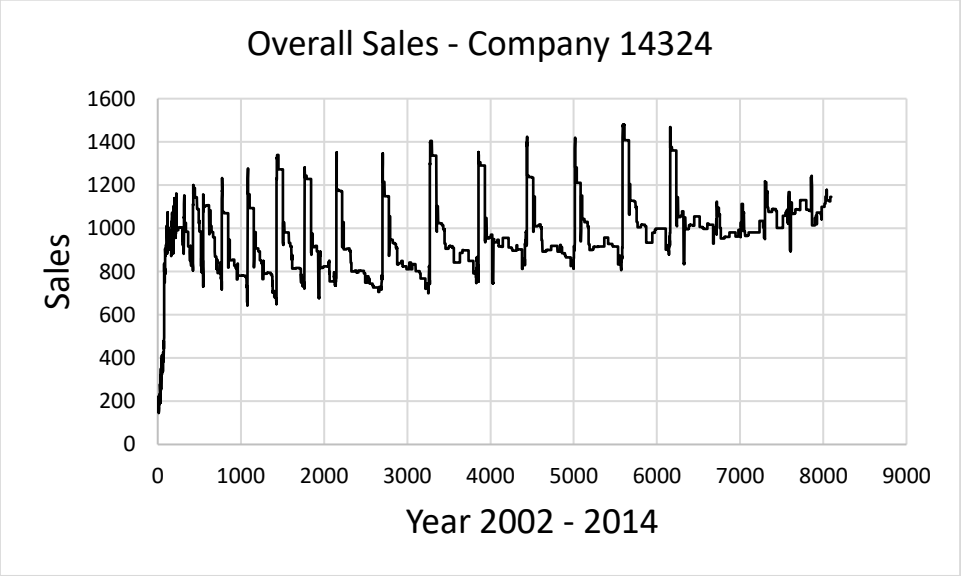

Figure 4-52: Company 14324 - Overall quarterly Sales

Above, we observe that the sales estimate jumps abruptly in the initial period.

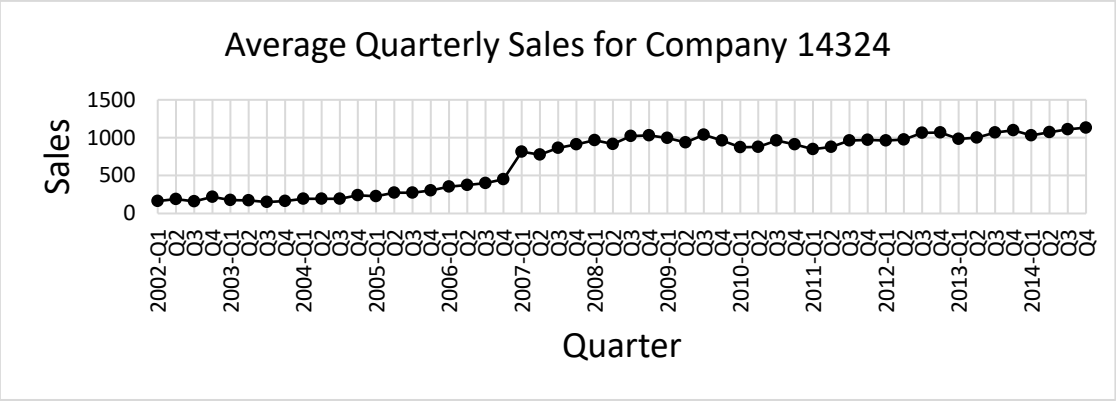

Figure 4-53: Company 14324 - Average quarterly Sales

The figure above indicates that the quarterly Sales is plateaued between year 2002 and 2007.

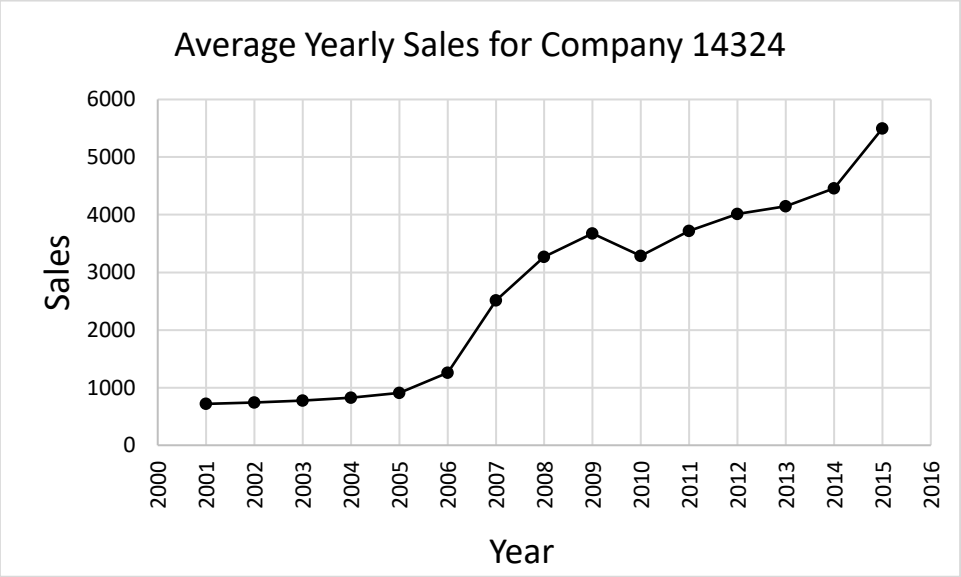

Figure 4-54: Company 14324 - Average yearly Sales

4.3.3.5 Company 18965

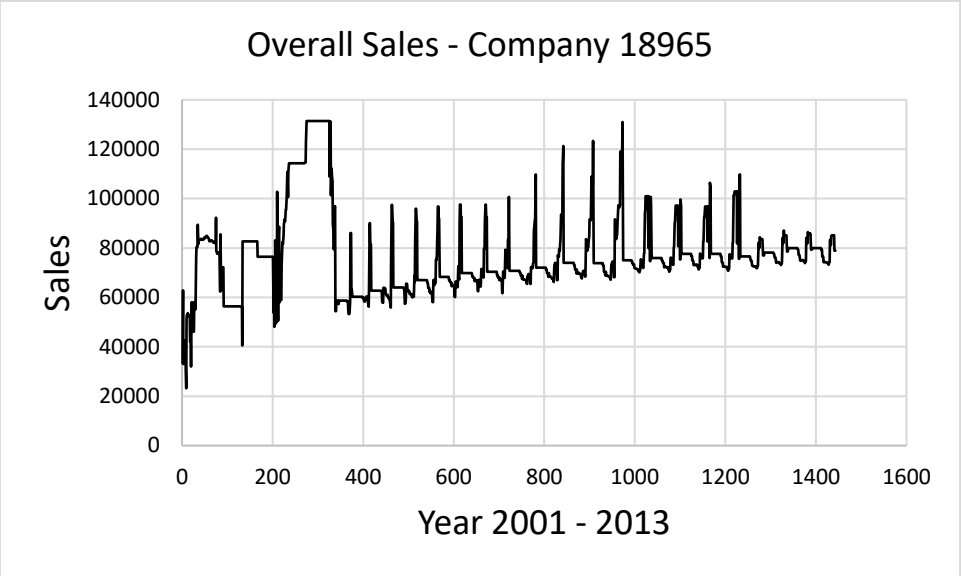

Figure 4-55: Company 18965 - Overall quarterly Sales

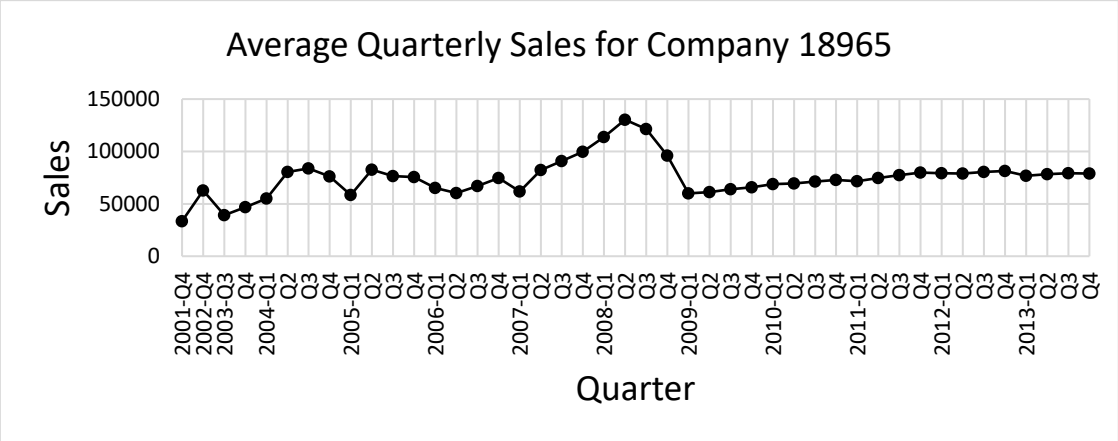

Figure 4-56: Company 18965 - Average quarterly Sales

The figure above indicates that the quarterly Sales is plateaued between year 2009 and 2013.

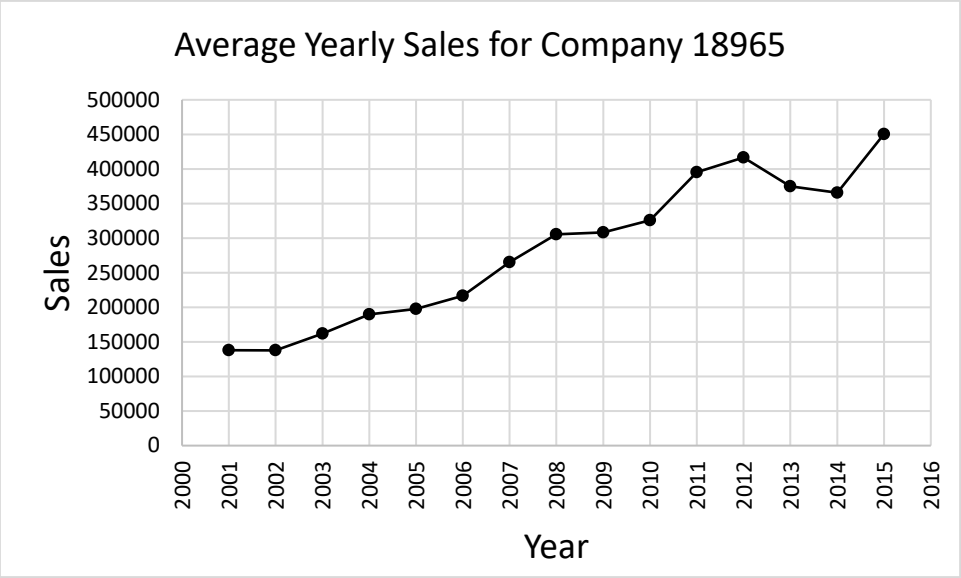

Figure 4-57: Company 18965 - Average yearly Sales

4.3.3.6 Company 29642

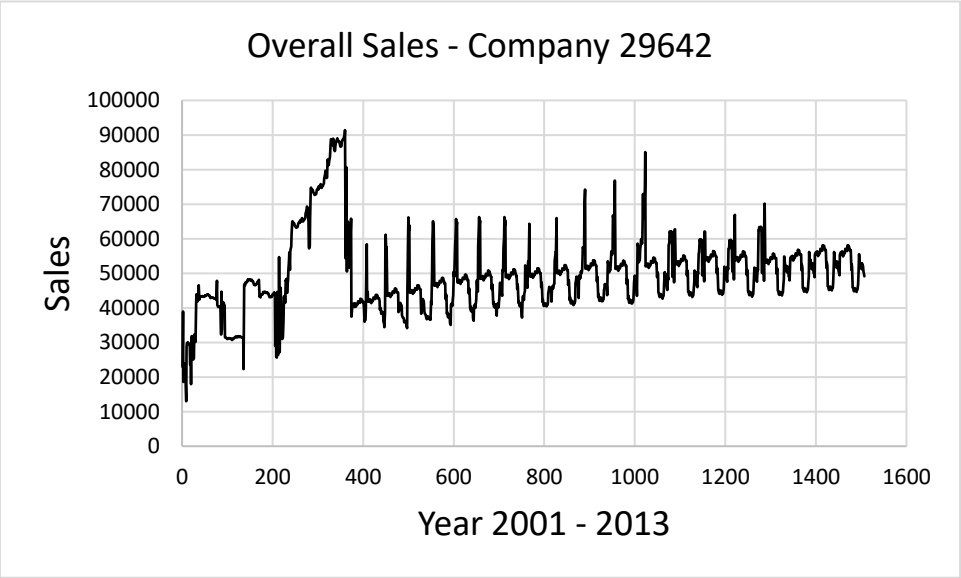

Figure 4-58: Company 29642 - Overall quarterly Sales

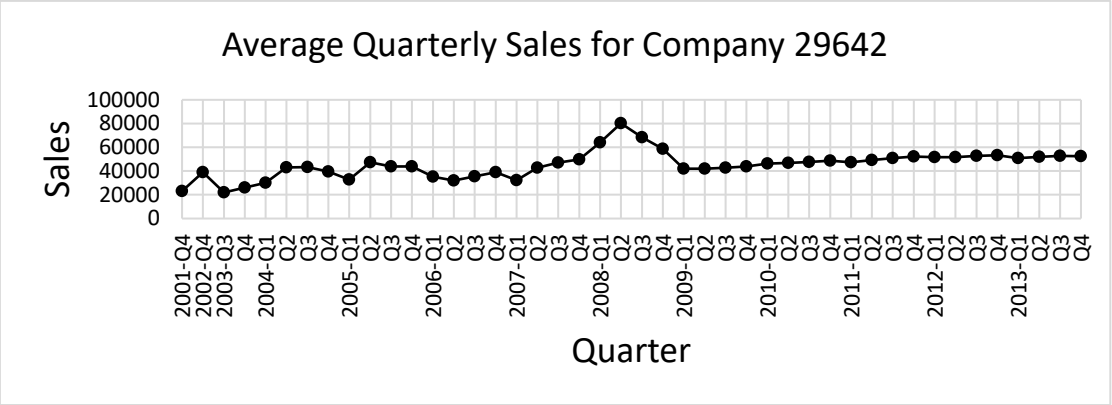

Figure 4-59: Company 29642 - Average quarterly Sales

The figure above indicates that the quarterly Sales is plateaued between year 2009 and 2013.

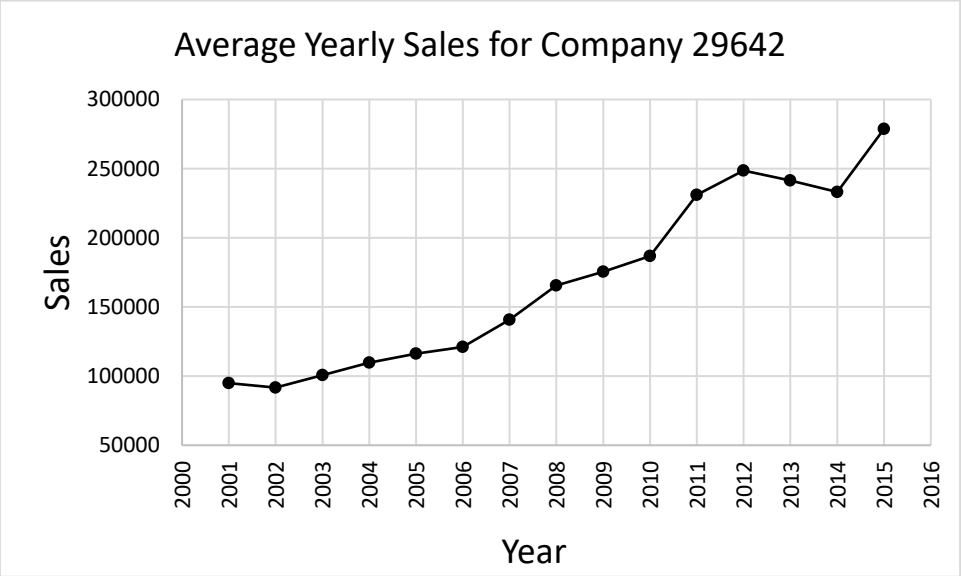

Figure 4-60: Company 29642 - Average yearly Sales

4.3.4 Comparison of EPS and Sales

The estimated EPS values of a company are determined based on the company's guidance and other factors. If a company beats the projected earnings, its stock price will usually go up. Else, its stock price will most likely decline. The financial analysts observe past EPS to decide if the company's stock is worth buying or not. Hence, a company's sales are closely related to the EPS.

In this section, we analyse the correlation between the Quarterly EPS and Sales estimates of the chosen companies.

4.3.4.1 Company 1290

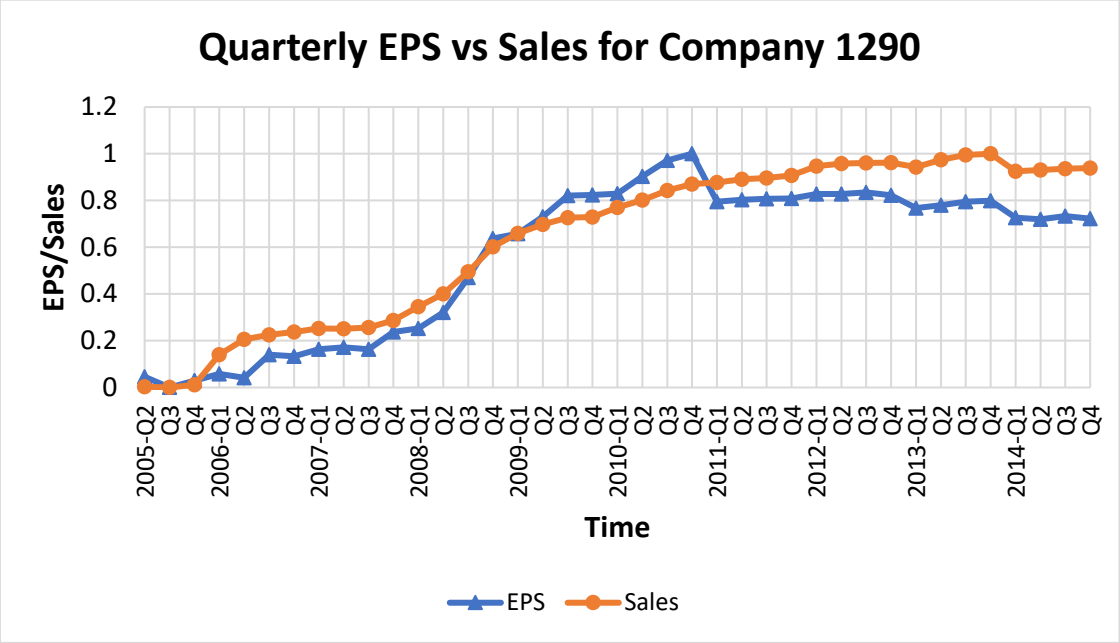

Figure 4-61: Company 1290 - Quarterly EPS vs Sales

A company with strong EPS might see the market price of its stock rise. This higher stock price might create a positive impression of the company's products in the market, resulting in increased sales. Inversely, poor EPS might reduce the stock prices resulting in lower consumer confidence, fewer sales and ultimately lower EPS. Thus, the correlation between EPS and sales could be useful in determining the performance of a company. Hence, we also observe the correlation between the EPS and Sales values.

As shown below, the correlation between the quarterly EPS and Sales values for company 1290 is quite high:

	EPS
Sales	0.9531

Figure 4-61 shows this correlation for company number 1290 graphically.

Similarly, we compare the yearly EPS and Sales values for this company below.

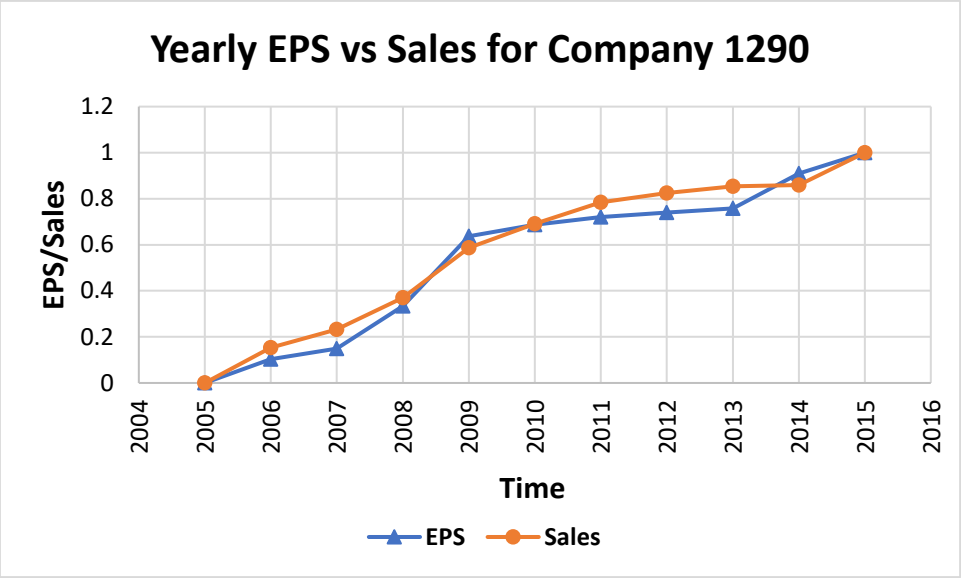

Figure 4-62: Company 1290 - Yearly EPS vs Sales

We can see that the correlation is not always very strong, in fact in the period between 2008 and 2009, it is negative.

The correlation between the yearly EPS and Sales values for company 1290 is:

	EPS
Sales	0.9884

Similarly, in the next subsections, we compare remaining companies.

4.3.4.2 Company 3180

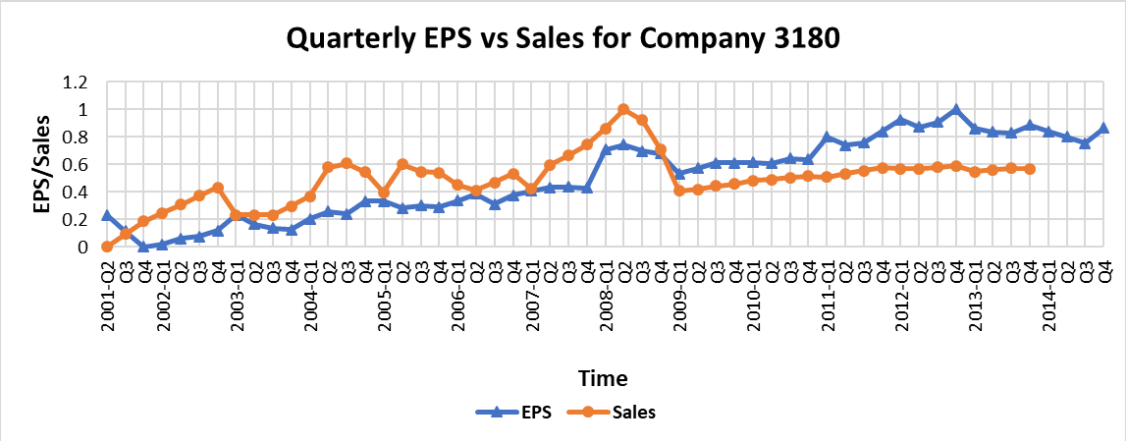

Figure 4-63: Company 3180 - Quarterly EPS vs Sales

The correlation between the quarterly EPS and Sales values for company 3180 is:

	EPS
Sales	0.60

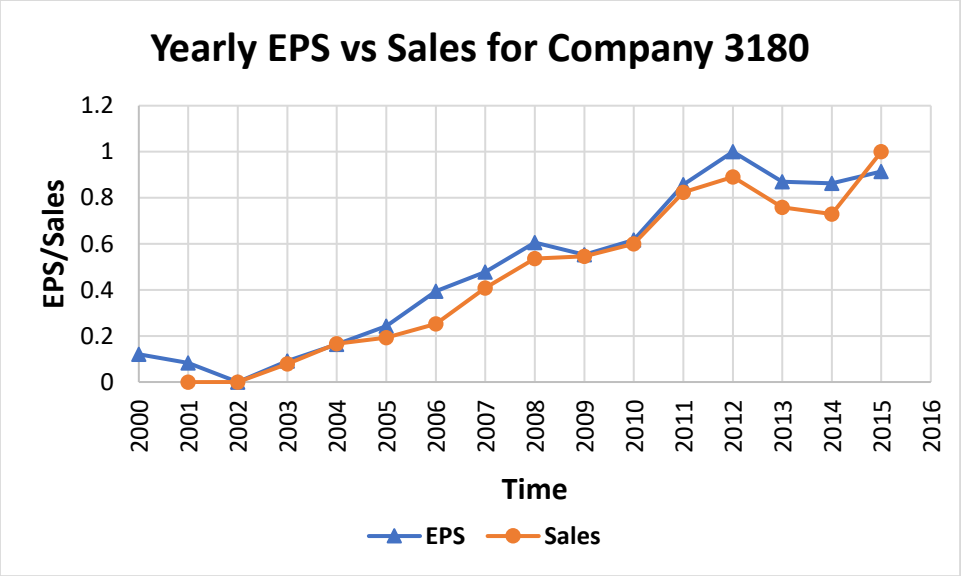

Figure 4-64: Company 3180 - Yearly EPS vs Sales

The correlation between the yearly EPS and Sales values for company 3180 is:

	EPS
Sales	0.9836

4.3.4.3 Company 11217

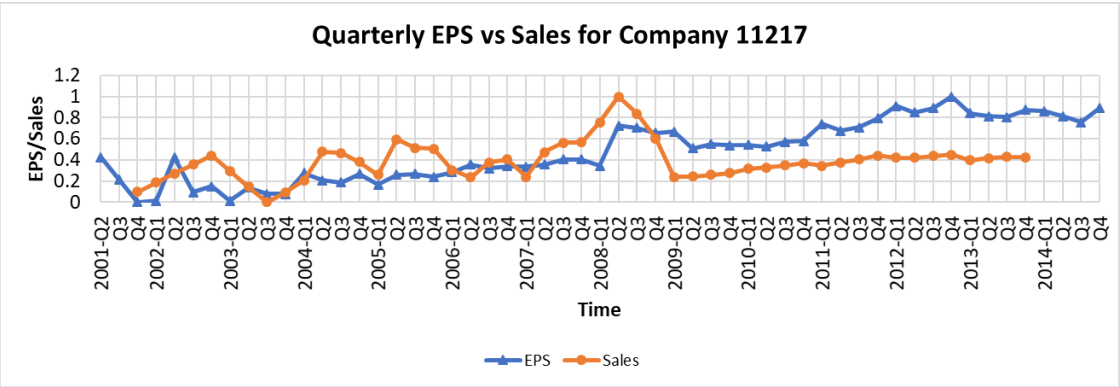

Figure 4-65: Company 11217- Quarterly EPS vs Sales

The correlation between the quarterly EPS and Sales values for company 11217 is:

	EPS
Sales	0.3678

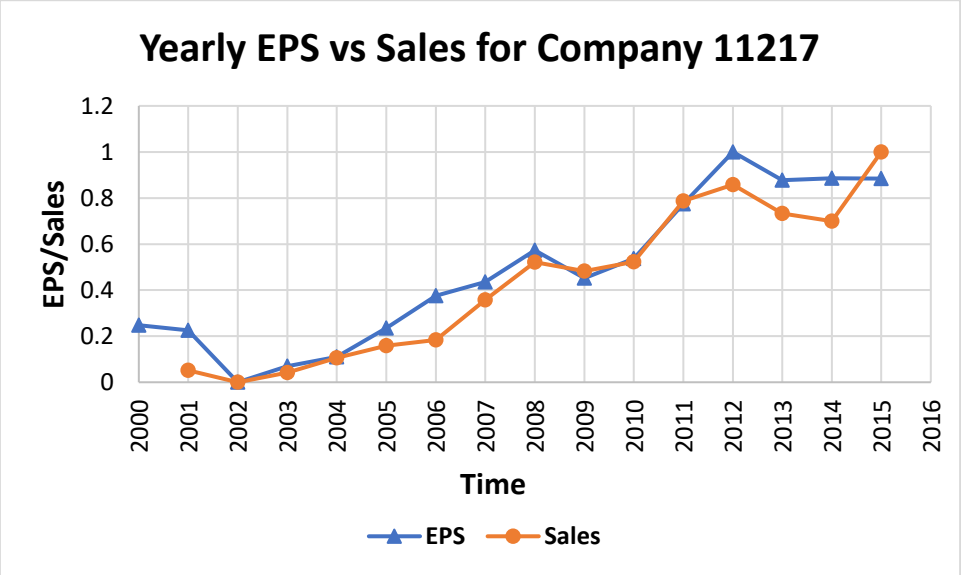

Figure 4-66: Company 11217 - Yearly EPS vs Sales

The correlation between the yearly EPS and Sales values for company 11217 is:

	EPS
Sales	0.9628

4.3.4.4 Company 14324

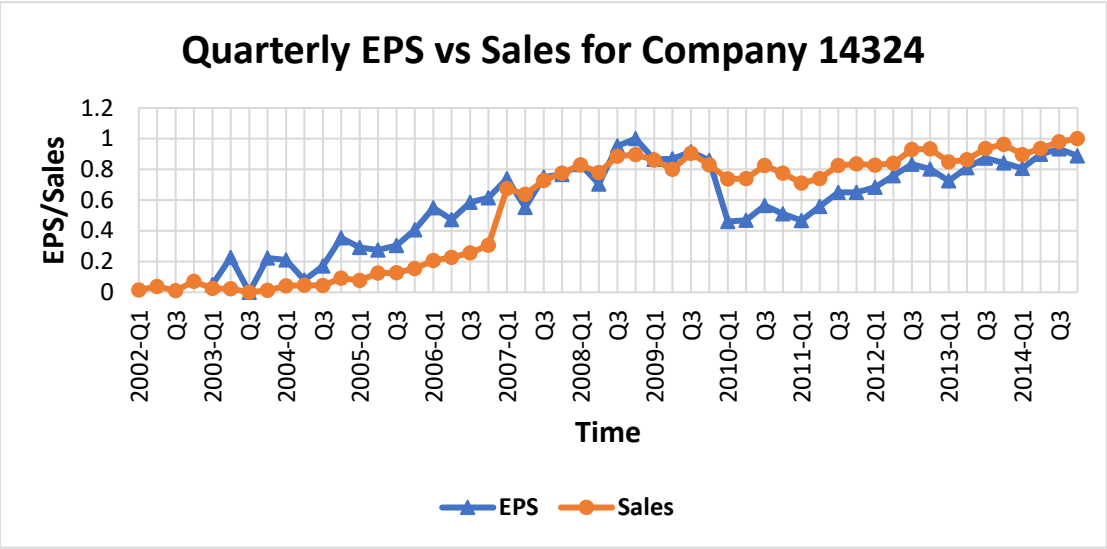

Figure 4-67: Company 14324 - Quarterly EPS vs Sales

The correlation between the quarterly EPS and Sales values for company 14324 is:

	EPS
Sales	0.8920

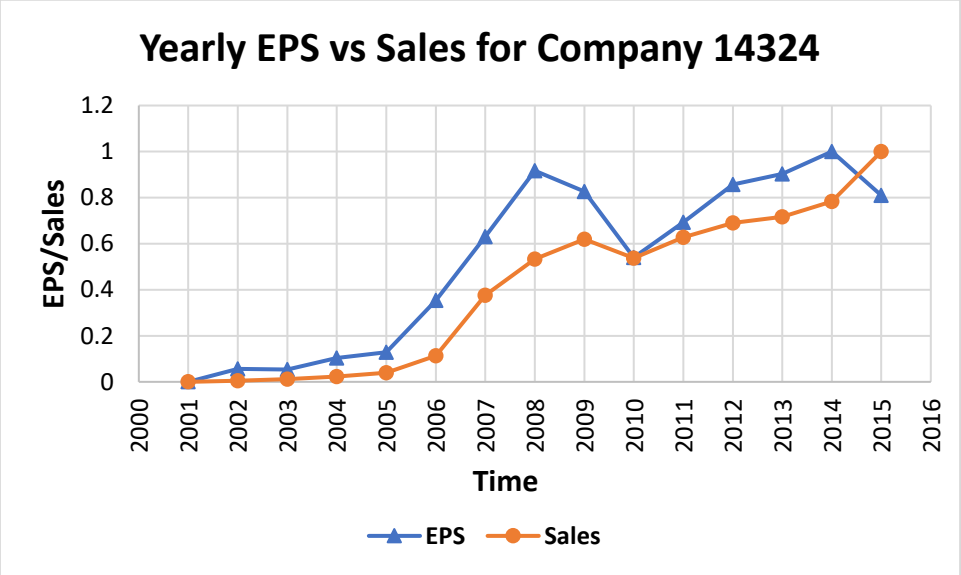

Figure 4-68: Company 14324 - Yearly EPS vs Sales

The correlation between the yearly EPS and Sales values for company 14324 is:

	EPS
Sales	0.9283

4.3.4.5 Company 18965

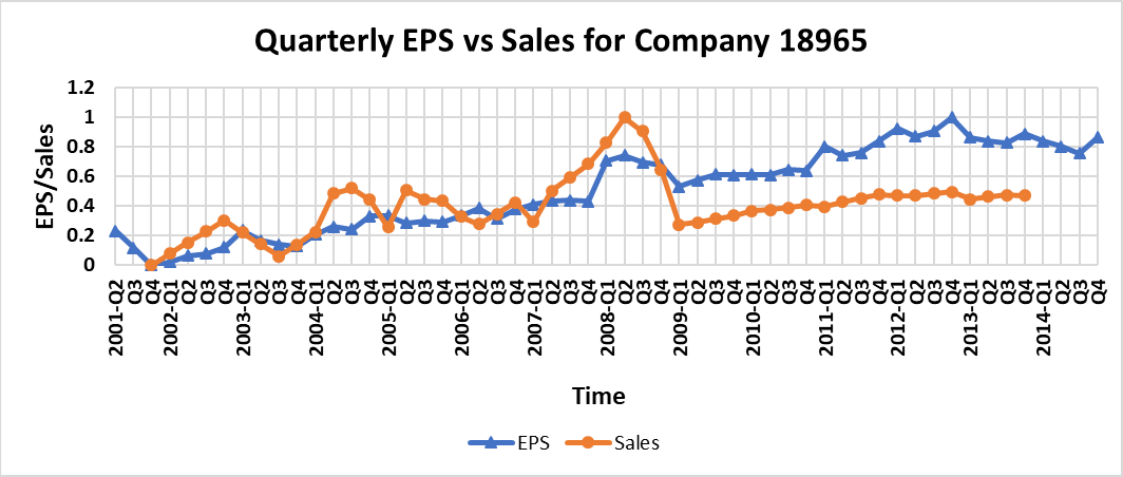

Figure 4-69: Company 18965 - Quarterly EPS vs Sales

The correlation between the quarterly EPS and Sales values for company 18965 is:

	EPS
Sales	0.5749

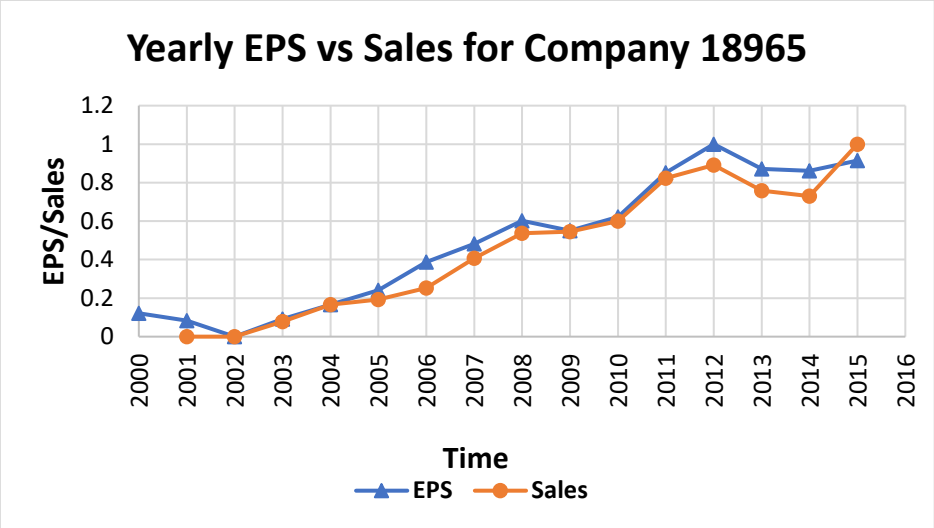

Figure 4-70: Company 18965 - Yearly EPS vs Sales

The correlation between the yearly EPS and Sales values for company 18965 is:

	EPS
Sales	0.9839

4.3.4.6 Company 29642

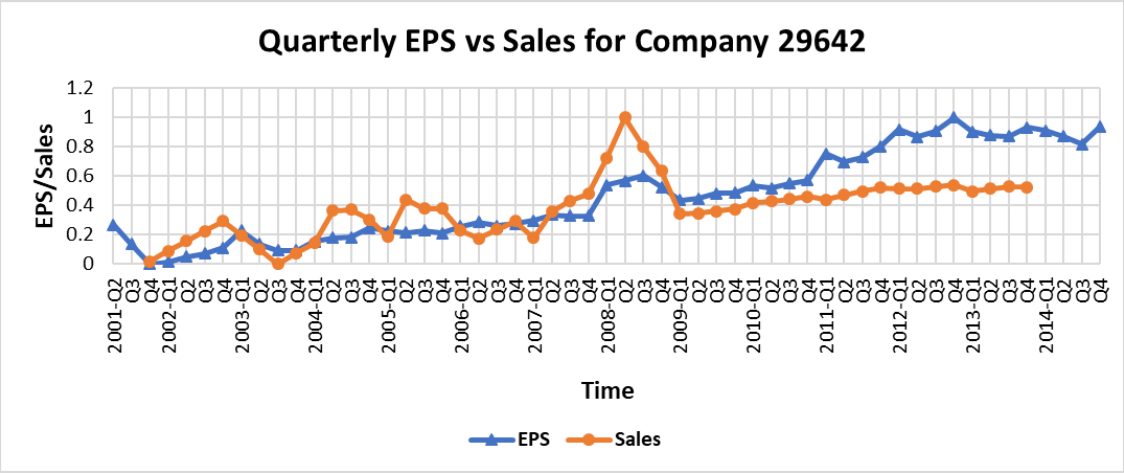

Figure 4-71: Company 29642 - Quarterly EPS vs Sales

The correlation between the quarterly EPS and Sales values for company 29642 is:

	EPS
Sales	0.7041

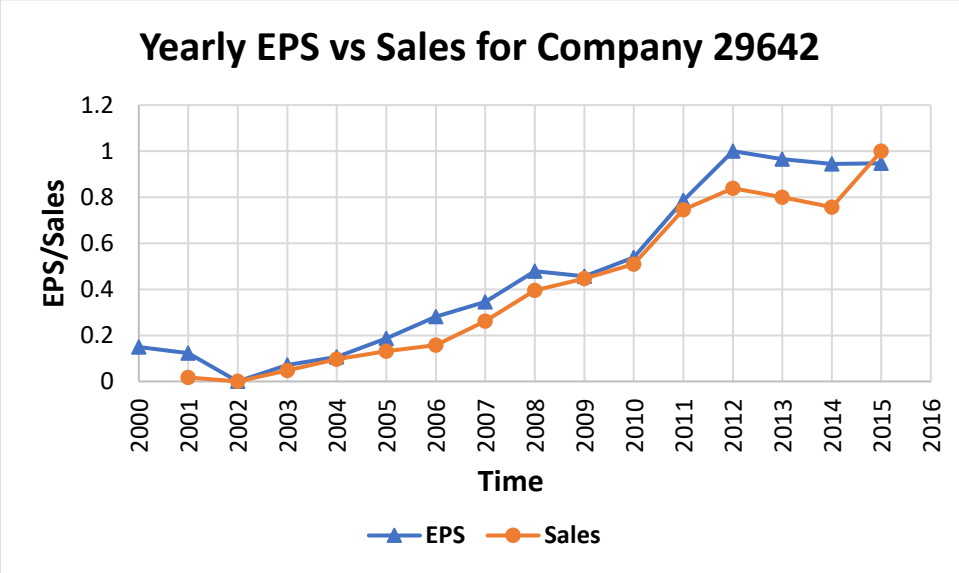

Figure 4-72: Company 29642 - Yearly EPS vs Sales

The correlation between the yearly EPS and Sales values for company 29642 is:

	EPS
Sales	0.9820

All the six companies exhibit a good correlation between Sales estimates and EPS estimates and this is observed from all the plots in this subsection.

4.4 Findings and Conclusions

In this chapter, experiments are conducted for a detailed analysis of the chosen datasets. The relationships among the variables and the usefulness of variables towards predicting the classes is investigated.

First, three public credit scoring datasets are analysed and the findings are:

- For the German credit data, only three variables show some level of correlation among them: ‘Duration of credit’, ‘Amount of credit’, ‘Age of applicant’. This is due to the fact that most of the other variables are originally categorical ones and are converted to numeric for data analysis and data mining purposes. The ‘Credit amount’ and ‘Age’ as well as ‘’Credit Amount’ and ‘Duration’ were found to be highly positively correlated. Out of all these variables, none are very good at separating the credit applicants into ‘good credit’ or ‘bad credit’ class.
- For the Australian credit dataset, few variables are found to be highly correlated with each other, some of them being good at separating the classes.
- For the Taiwan credit data, many attributes show good positive correlation among them and also good class separability.

Next, we performed data analysis for a real industry dataset comprising estimates of Earnings per share and Sales data. The observations are:

- Localised patterns are observed for the EPS estimates;
- Initial negative EPS estimates are observed for company 1290, 14324. The companies could be operating at a loss. A negative EPS (also called net loss

per share) tells exactly how much money the company lost per share of outstanding stock.

- Overall yearly EPS for all the six companies shows upward trend;
- Smoother Sales estimates than the EPS estimates: For the company 1290;
- Plateaued Sales between years 2009 and 2013: For companies 3180, 11217,18965 and 29642;
- Plateaued Sales between years 2002 and 2007: For company 14324;
- The correlation between EPS and Sales for all the companies is summarised below which lists the correlation coefficient R:

Table 4-6: Correlation coefficients between EPS and Sales for all the companies

	Quarterly EPS and Sales	**Yearly EPS and Sales**
Company 1290	***0.9531***	***0.9884***
Company 3180	0.6	0.9836
Company 11217	0.3678	0.9628
Company 14324	0.892	0.9283
Company 18965	0.5749	0.9839
Company 29642	0.7041	0.9820

- We observe a strong correlation between yearly EPS and Sales. Company 1290 exhibits good correlation between quarterly EPS and Sales followed by company 14324. All other companies exhibit poor correlation for quarterly data.
- The human brain is proven to process visual representation of data far more easily than any other form. It ensures faster comprehension of relationships than cluttered reports or spreadsheets, more true in financial data. The analyses carried out in this chapter are useful to the research community as well as the end users of the private dataset. Use of heatmaps for multidimensional datasets provides new ways to interpret the data. The investigation of relationships among variables in public datasets allows understanding of the strengths and direction of a relationship. In case of the EPS and Sales data, the individual analysis of

the EPS and Sales of companies will empower the decision makers to filter the information from the very large dataset, to get real value from the dataset and identify connections among various pieces of data.

- The results of data analysis are useful if fed back to the origins of data generation. It allows for a guided sampling of data to avoid repetitive data by diversifying the methods of data collection. This steadily intensifies understanding of the problem as a whole.

5 A FEW CLASSICAL MACHINE LEARNING TECHNIQUES FOR THE PROBLEM OF CREDIT SCORING

5.1 Introduction

From large financial institutions to smaller banks, machine learning is being applied to various tasks with promising results e.g. risk management, compliance, financial crime, fraud detection and cybersecurity, credit underwriting to name a few. However, we rarely see public information of 'successful' machine-learning methods as applied to financial markets. Every research study undertaken provides a 'component' piece upon which an entire system may be built. This chapter provides an independent assessment of some of the classical machine learning algorithms for the problem of credit scoring.

5.2 Research motivation and contribution

Although the traditional method of credit scores is still useful, the industry is seeing rapid shift towards alternative models. A number of emerging companies use proprietary "machine-learning" algorithms get better insights into the data available for each consumer [196]. The machine-learning tools used by the finance industry are closely guarded trade secrets, making it impossible to offer a comprehensive picture of the industry. There is no single methodology to design and solve the unstructured problem of measuring creditworthiness. The credit scorer may also be interested in constantly improving upon the existing model. There exist many articles as to how machine learning could change the credit scoring industry but not many research studies dedicated to comparing various ML techniques in this domain.

The work by Ben-David & Frank [197] made a comparison between machine learning models and a credit scoring expert system, whose results revealed that while some of the machine learning models' accuracies are better than those expert system model, most of them are not. The extensive literature review presented in chapter 3 also indicates that, although work has been conducted in

the area of credit scoring, a potential exists for more detailed work as gaps still exist e.g. scope for improvement in existing classical machine learning models. Whenever a model is trained, we learn our personal preferences for selecting systems. With machine learning, we can test thousands of models and will certainly find something that works for a business case. Between the choice of target, features and meta-features, there are almost unlimited possible models to choose from and each learning algorithm is different. Compared with model based strategies, not many successful machine learning systems exist [198]. Also, the size of the dataset dictates choice of model complexity. ML trains a model from patterns in the data, exploring a space of possible models defined by parameters. As a rule, models should be kept as simple as possible [199].

In this chapter, we address following objectives:

1. Undertake evaluation of a few classical machine learning techniques for the problem of credit scoring. Three chosen machine learning techniques are applied on three public credit scoring datasets to compare their results in order to answer the question: From an initial set of possible models, which is the most appropriate model to fit our datasets?
2. We test the performance of the classifiers against a much larger database of credit card customers (Taiwan credit database) which has not been considered in the literature so far. This dataset restricts to those clients whose past monthly payment records and amount paid by them are restricted in the period from April to September 2005. Hand [200] points out that for many classification problems, the population distributions are nonstationary, in that the class distributions shift over time. This is particularly true of credit data with applicants' behaviour changing over time due to changing economic conditions or changes to product and marketing practices. For this reason, a clearer model can be developed if based on data taken from a narrow time period within which there is likely to be less variability in these circumstances [201].
3. The effect of varying the hyperparameters on the accuracy of the SVM and KNN classifiers is analysed. An essential step in order to get the best possible

fit for the dataset is the tuning of parameters. We aim to provide an evidence of how hyperparameters could affect the performance of classical machine learning techniques for the credit scoring problem. First, we compare the performance of the classifiers when using the default parameters set by the software MATLAB. The second analysis concerns the variation of single parameters of the classifiers, while maintaining other parameters in the default value.

4. We aim to appraise the most appropriate techniques for credit scoring. The motivations for this particular research topic are to determine the predictive power of commonly used techniques such as Gaussian SVM, KNN and Naïve Bayes and compare their classification performance. The aim is twofold: to better inform the industry and to improve the usability of potential techniques.

5.3 The problem of credit scoring

Financial institutes use scoring models to lower credit risk in credit appraisals, and in the granting and supervision of credit. Large amount of consumers' credit data is collected by the credit department of the institute. Often, the intuitive experience of people dictates the evaluation of the consumer's credit. However, the credit classification models help evaluate the applicant's credit score accurately. Credit scoring models based on classical statistical theories have been used. However, handling large amounts of data input is not a very strong point of these models affecting the accuracy of prediction and model generalisations. Many financial institutes use machine learning (ML) models in credit scoring to achieve a more accurate credit scoring from large amounts of data.

According to a large number of empirical studies, machine learning techniques – along with other data-mining algorithms based on computational innovation and transformation – seem to perform better when fitting data and forecasting. However, deciding on one model is a difficult task as the performance of the algorithms is data-sensitive and the performance needs to be tested on different test sets relating to the problem. The general way to find an appropriate model

for a single specific data set, or a type of data set, is to apply it to some widely-used and well-proven algorithms and then compare the performance.

There are some studies done in this area which consider multiple algorithms along with methods of feature selection, however not many studies have investigated the baseline classical ML methods against the three most widely used public credit datasets.

The benefits of credit scoring involve reducing the credit analysis cost, enabling faster credit decisions, closer monitoring of existing accounts and prioritising credit collections [202]. Even a fraction of improvement in the prediction of credit scoring accuracy could potentially translate into noteworthy future savings. Hence, a majority of previous studies focused on increasing the accuracy of credit decisions. Automatic credit models help creditors and bankers make decisions regarding loans, development of markets, assessment of creditworthiness and detection of fraud. When an applicant applies for credit, creditors construct the credit classification rules and scoring models based on the historical data of the previous applicants. With sizeable loan portfolios, even a slight improvement in credit scoring accuracy can reduce the creditors' risk and translate considerably into future savings [203].

5.4 Methodologies

To fulfil the objectives listed in section 1.2, we consider techniques of Gaussian Support Vector Machines, k-nearest neighbours and Naïve Bayes to compare the performance on three public credit datasets. The details of the datasets can be found in chapter 4. The measures of accuracy, ROC analysis and confusion matrix analysis are employed to assess the performance across various classifiers. The following subsections describe these techniques in the light of credit scoring problem.

To analyse the execution time and accuracy performance of the classifiers, we conduct experiments over the three datasets for credit classification for twelve classical machine learning techniques. The SVM approach can be extended to a

non-linear surface by using a kernel trick. The kernel functions used are: Polynomial SVM, Gaussian SVM.

Following table lists all the techniques applied to the chosen databases in this section [204], [205].

Table 5-1: Techniques used for classification

[K=Kernel; x_i, x_j = vectors of features in the input space; σ= Kernel scale; P= Number of predictors; N= Number of neighbours]

Serial No	**Technique**	**Function**
1	Linear SVM	$K\left(x_i, x_j\right) = x_i - x_j$
2	Polynomial- Quadratic SVM	$K\left(x_i, x_j\right) = \left(x_i - x_j\right)^2$
3	Polynomial- Cubic SVM	$K\left(x_i, x_j\right) = \left(x_i - x_j\right)^3$
Gaussian:		$K\left(x_i, x_j\right) = exp\left(-\left\|x_i, x_j\right\|^2 \Big/ 2\sigma^2\right)$
4	Fine Gaussian SVM	$\sigma = \frac{\sqrt{P}}{4}$
5	Medium Gaussian SVM	$\sigma = \sqrt{P}$
6	Coarse Gaussian SVM	$\sigma = \sqrt{P} * 4$
7	Fine KNN	Finely detailed distinctions between classes; N = 1
8	Medium KNN	Medium distinctions between classes; N= 10
9	Coarse KNN	Coarse distinctions between classes; N= 100

10	Cosine KNN	Medium distinctions between classes; cosine distance metric; N = 10
11	Cubic KNN	Medium distinctions between classes; cubic distance metric; N= 10
12	Weighted KNN	Medium distinctions between classes; distance weight; N= 10

5.4.1 SVM- The Linear case

The Support Vector Machine (SVM) classification method was introduced in 1992 by Boser, Guyon, and Vapnik [206]. The SVM classifier is widely used in many disciplines due to its high accuracy, ability to deal with high-dimensional data, and flexibility in modelling diverse sources of data [207].

Unlike most of the traditional learning machines that adopt the Empirical Risk Minimisation Principle, SVMs implement the Structural Risk Minimisation Principle [207], which seeks to minimise an upper bound of the generalisation error rather than minimise the training error. This will result in better generalisation than conventional techniques. For non-linear data, SVMs work better for binary classification, notwithstanding SVM takes a little longer but learns more robust frontiers because of the ability to maximise margin.

When training an SVM, several decisions should be made regarding preprocessing of the data, choice of kernel, and parameter values of the SVM. Uninformed choices may result in severely reduced performance [208]. SVMs have become very popular in solving difficult classification problems in a variety of application domains. The reasons are based on two key properties:

1. The solutions provided by SVMs to classification problems are very good at generalisation even if the data is high-dimensional and training sample set is small. This is attributed to the regularisation since the regularisation term C, the separation margin, is implicitly able to capture important dimensions from the high dimensional data and trains sparse classifiers. While SVMs are useful

to reduce the "curse of dimensionality" problem by reducing the risk of over-fitting the training data, feature selection is required according to the nature of the problem. This thesis addresses the feature selection problem in chapter 6.

2. SVMs are able to find non-linear solutions to difficult and non-linearly separable problems efficiently using the "kernel trick". Non-linear decision boundaries can be generated using linear classifiers. Also, kernel functions allows the application of a classifier to data with no obvious fixed-dimensional vector space [209]. SVM is a hyperplane classifier based on drawing separating lines to classify objects of different classes. A binary classification problem is called linearly separable, if there exists at least one hyper-plane such that all the points of one class fall on one side and all points of the other class fall on the other side of the plane.

When separating the credit applicants as creditworthy or not, two cases need consideration: Linearly separable and non-linearly separable ones.

The input to SVMs is a set of training pair samples, representing the instances (x_i, y_i) for $i = 1, \dots, n$ where $y_i \in \{+1, -1\}$ are labels of the instances. The output is a set of weights w (or w_i), one for each feature, whose linear combination predicts the value of y. Margin of Separation is the separation between the hyperplane and the closest data point (“difficult points”) for a given weight vector w and bias b. SVM finds an optimal hyperplane that maximises the margin of separation by optimisation. This reduces the nonzero number of weights to correspond to the important features deciding the hyperplane, i.e. support vectors. Support vectors are critical and they change the position of the dividing hyperplane if removed.

Consider the case of classification of an example function $f(x) = y = \{+1, -1\}$.

The instances are $\{x_i, y_i\}$ where x_i is a point in instance space R^n made up of n attributes; y_i are class values for classification of x_i.

To find a linear separator, this problem could be viewed as a constraint satisfaction problem where all the data satisfy following constraints:

$$\vec{x_i}.\vec{w} + b \geq +1 \quad if\ y_i = +1 \tag{5-1}$$

$$\vec{x_i}.\vec{w} + b \leq -1 \quad if\ y_i = -1 \tag{5-2}$$

here, w is the normal to the hyperplane, $\frac{|b|}{\|w\|}$ is the perpendicular distance from the hyperplane to the origin, and $\|w\|$ is the Euclidean norm of w.

We can combine these two constraints into following:

$$y_i(\ \vec{x_i}.\vec{w} + b\ ) \geq 1, \qquad (\forall_i) \tag{5-3}$$

When two classes are linearly separable, an optimum separating hyperplane can be found by minimising the squared norm of the separating hyperplane. The minimisation can be set up as a convex quadratic programming (QP) problem.

The points for which above equation is true lie on the hyperplanes *H1* and *H2* which constitute the boundaries of the widest margin. The width of the margin ρ can be calculated as:

$$\rho = \frac{|1-b|}{\|\vec{w}\|} - \frac{|-1-b|}{\|\vec{w}\|} = \frac{2}{\|\vec{w}\|} \tag{5-4}$$

i.e. equal width each side. We can calculate the optimal margin classifier separating hyperplane by solving the primal optimisation problem:

$$\min_{\vec{w}\in\mathcal{H}}\ \tau(\vec{w}) = \frac{1}{2}\ \|\vec{w}\|^2$$

subject to

$$y_i(\ \vec{x_i}.\vec{w} + b\ ) \geq 1, \qquad (\forall_i) \tag{5-5}$$

Above constrained optimisation problem can be solved by using Lagrange multipliers. It makes the constraints easier to handle and the training data is used as a dot product between vectors. The new minimisation problem is,

$$\min_{\vec{w},b} L(\vec{w}, b, \alpha) \equiv \frac{1}{2} \|\vec{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (x_i \vec{w} + b) + \sum_{i=1}^{l} \alpha_i \quad \textbf{(5-6)}$$

with Lagrange multipliers $\boldsymbol{\alpha_i \geq 0}$ for each constraint in equation **(5-5)**. The equation **(5-6)** is minimised with respect to $\boldsymbol{w}$ and $\boldsymbol{b}$ with the requirement that the derivatives of $\boldsymbol{L(\vec{w}, b, \alpha)}$ with respect to all the $\boldsymbol{\alpha}$ vanish.

5.4.2 SVM: The Non-linear case

When the dataset is not easily classified by linear classifier, the original data could be mapped on to a higher-dimensional space where the training set is separable. Then the linear classifier could be used in the higher dimensional space. This is called as the 'kernel trick'. The two commonly used families of kernels are polynomial kernels and radial basis functions.

The degree of the polynomial kernel controls the flexibility of the resulting classifier. The dimensionality of polynomial kernel function is dependent on its degree parameter. The lowest degree polynomial is the linear kernel (or no kernel at all), which is not sufficient to classify non-linear relationship between features. Higher degree kernel can map patterns into higher dimensional space and therefore yield a higher dimensional hyperplane. The polynomial kernel of degree 1 leads to a linear separation. Higher-degree polynomial kernels allow a more flexible decision boundary. When degree rises, the kernel is flexible enough to discriminate between two classes with a sizeable margin. Very high-degree polynomial yields a decision boundary with greater curvature.

When the data is not linearly separable, an objective function that trades off misclassifications against minimising $\|\vec{w}\|^2$ from (5-6) to find an optimal compromise can be set up which is a "slack" variable $\xi_i \geq 0$. It is added for each training example with the requirement:

$$\vec{x_i} . \vec{w} + b \geq 1 - \xi_i \quad if\ y_i = +1 \quad \textbf{(5-7)}$$

$$\vec{x_i} . \vec{w} + b \leq -1 + \xi_i \quad if\ y_i = -1 \quad \textbf{(5-8)}$$

The function to be optimised is given by:

$$J = \|\vec{w}\|^2 + C(\sum_{i=1}^{n} \xi_i^k) \quad \textbf{(5-9)}$$

Here, the constant C is a user specified parameter. k is set to 1 in SVMs. The solution is:

$$\vec{w} = \sum_i \alpha_i \, y_i \, x_i \quad \textbf{(5-10)}$$

The α_is are Lagrange multiplier coefficients.

5.4.3 Gaussian SVM

In this study, Gaussian SVM is used to classify the credit datasets consisting of two classes, namely creditworthy and non-creditworthy. Gaussian tends to converge rapidly and performs well with relatively small dataset [8].

The problem of classification in Credit Scoring is realised as mapping of input features set into the decision variable (taking value as creditworthy or not creditworthy), represented as $y=f(x)$, where y is the decision variable and x is the feature vector. Identifying creditworthy applicants from not creditworthy ones is not a linearly separable problem. Non-linear machines which map the data to higher dimensions can be used to find a SVM hyperplane minimising the number of errors for the training set.

Gaussian-kernel SVM is powerful for non-linear binary classification problems, and repetitions of the assessment is easier with the Gaussian-kernel SVM in MATLAB [210]. The Gaussian-kernel SVM maps the problem space to higher dimension, making the data linearly separable and using the linear SVM to solve the problem in the new higher dimension space. A Gaussian-kernel SVM performs well in high dimensional spaces, even if the number of dimensions is greater than the number of samples [210]. Assume $\Phi(x)$ is a feature map (which can be very high dimensional) where x is mapped to the kernel function $(x_i, x_j) = \Phi(x_i)^T \, \Phi(x_j)$. The kernel SVM is expressed as:

$$f(x) = \left(\sum_{i=1}^{N} \alpha_i y_i \, k(x_i, x_j) + b \right) \quad \textbf{(5-11)}$$

where α_is are dual variables (the Lagrange multiplier for each training example i) and $k(x_i, x_j)$ is the kernel function defined as the inner product between two feature vectors, performing the nonlinear mapping into feature space.

Correspondingly, learning to maximise:

$$\sum \alpha_i - \frac{1}{2} \left(\sum_{jk} \alpha_j \, \alpha_k \, y_j \, y_k \, k(x_i, x_j) \right) \quad \textbf{(5-12)}$$

this is the SVM optimisation problem, where

$$y_i \in \{+1, -1\} \, for \; all \; i = 1 \; to \; n \quad \textbf{(5-13)}$$

Subject to constraints $C \geq \alpha_i \geq 0, \; for \; all \; i = 1 \; to \; n \; \forall_i$ and $\sum \alpha_i \, y_i = 0$

where C, which is now the upper bound on α_i, is a penalty parameter and is determined by the user. SVMs are able to deal with high dimensional data through the use of this regularisation parameter C.

The kernel function k can be used to implement non-linear models of the data.

The (Gaussian) Radial Basis Function kernel is commonly used as the kernel of a SVM.

$$k(x, x') = exp \left(\frac{-\|x - x'\|^2}{2\sigma^2} \right) \quad \textbf{(5-14)}$$

The Gaussian-kernel SVM:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \, exp \left(\frac{-\|x - x'\|^2}{2\sigma^2} \right) + b \quad \textbf{(5-15)}$$

The radial basis function kernel has an additional kernel parameter γ i.e. kernel bandwidth to be optimised, where $\gamma = \frac{1}{2\sigma^2}$. As γ increases the fit becomes more and more non-linear.

What is the effect of the width parameter of the Gaussian kernel (σ) for a fixed value of the soft-margin constant?

The kernel scale parameter $\sigma>0$ controls the width of the Gaussian and hence the flexibility of the resulting classifier. The Gaussian kernel is zero if the squared distance $\|x - x'\|^2$ is much larger than σ. The discriminant function (Equation 11) is thus a sum of Gaussian "bumps" centred around each support vector. When σ is large, a given data point x has a nonzero kernel value relative to any example in the set of support vectors. Therefore, the whole set of support vectors affects the value of the discriminant function at x, leading to almost smooth linear decision boundary. As we decrease σ, the kernel becomes more local and the flexibility of the decision boundary increases leading to greater curvature of the decision surface [209]. Thus, very small values of σ lead to overfitting.

5.4.4 KNN

The KNN algorithm, first proposed by Fix and Hodges [211] is a nonparametric technique for classifying objects based on closest training examples in the feature space. A test point is classified based on voting of the nearest k neighbours (i.e. when nearest neighbours are searched, the label of test point depends on the majority votes of k nearest neighbour labels). The term 'nearest' can be measured in terms of distance function of various forms.

This is instance-based learning where the function is only approximated locally and all computation is delayed until classification, i.e. there is no model building process, hence they are also called as lazy learners. With little or no prior knowledge about the distribution of the data, KNN is the method of choice for classification. The entire training set is retained during learning and a class represented by the majority label of the query's k-nearest neighbours in the training set is assigned to it.

KNN is based on the principle that instances within a dataset will generally exist in close proximity to other instances that have similar properties [212]. From the training set, objects are classified by a majority vote of their k nearest neighbours. k *is* a small positive number and the correct classification of the neighbours is known a priori. The distance to the neighbours of an object to be classified is determined by a distance metric, e.g. Euclidean distance or Manhattan distance.

The high degree of local sensitivity makes KNN highly susceptible to noise in the training data – thus, the value of k strongly influences the performance of the KNN algorithm.

5.4.4.1 Distance Metric

KNN predicts the class of a new point based on the outcome of the k neighbours closest to the point. The closeness is measured by a distance measure. Distance measure find distance between a new data point and existing training dataset. The choices to measure this distance are Euclidean, Manhattan, City-block, Chebychev, Hamming, Minkowsky etc. The most widely used measure Euclidean distance.

Let A and B be two points in a feature space and let A and B be represented by feature vectors $A = (x_1, x_2, \ldots, x_m)$ and $B = (y_1, y_2, \ldots, y_m)$, where m is the dimensionality of the feature space. To calculate the distance between A and B, the normalised Euclidean metric is generally used by:

$$dist\ (A, B) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \qquad \textbf{(5-16)}$$

Few of the main advantages of KNN are its simplicity, effectiveness, intuitiveness and impressive classification performance in many application areas. It is robust to noisy training data and is effective if the training data is large [213], however it can have poor run-time performance when the training set is large. KNN is sensitive to irrelevant or redundant features because all features contribute to the similarity and thus to the classification. By careful feature selection or feature

weighting, this can be avoided. Computation cost of KNN is quite high because of computation of distance of each new point to all training samples.

5.4.4.2 Choosing the value of K

Choosing a proper k value is important to the performance of KNN classifier. If k value is too large, the classifier may misclassify the test instance because its list of nearest neighbours may include data points that are located far away from its neighbourhood. We may end up smoothing things out too much and eliminating some important details in the distribution. On the other hand, if k value is too small, then the classifier may be susceptible to overfitting because of noise in the training data set. K should be large enough to avoid overfitting, but small enough to avoid oversimplifying the distribution.

5.4.5 Naïve Bayes

The Naive Bayes algorithm is a fast and highly scalable model building and scoring technique. The actual technique is a probabilistic learning method using Bayes theorem, with the “naive” assumption of independence between each pair of features. This allows each distribution to be independently estimated as a one-dimensional distribution. The NB theorem calculates a posterior probability for each class by counting the frequency of combinations of values in the historical data. Thus, the probability of a customer being good credit or bad credit given the probability of such an event that has already occurred, could be found.

Let $U = \{X, C\}$ be the finite set of customer credit information, $X = \{x_1, \dots, x_n\}$ be the set of credit characteristics variables such as age, gender, annual income, and $C = \{c_1, c_2\}$ be the class variable with two classes denoting whether a credit client is a good or bad credit applicant. According to Bayesian theorem, the probability of an instance $X = \{x_1, \dots, x_n\}$ belonging to c_j, can be written as:

$$P(c_j|x) = \frac{P(x|c_j)\ P(c_j)}{P(x)} \quad \textbf{(5-17)}$$

where

$P(c_j|x)$ is the posterior probability of class c_j (target) given predictor x;

$P(c_j)$ is the prior probability of class c_j;

$P(x|c_j)$ is the likelihood which is the probability of predictor x given class c_j; [NB assumes each predictor x is only correlated with class c_j and non-correlated with any other values of predictors from X;]

$P(x)$ is the prior probability of predictor x and is a constant.

The assumption of conditional independence of variables given the class label simplifies the estimation of the class-conditional probabilities from the training data.

NB is robust against the number of predictors and size of data and require a small amount of training data to estimate the necessary parameters for accurate classification. It performs well in case of categorical input variables compared to numerical variables. For numerical variable, normal distribution is assumed. A limitation of Naive Bayes is the same assumption of independent predictors. In real life, getting a set of predictors, which are completely independent is a rare possibility.

5.4.6 Performance Assessment Methods

The most commonly used measure of classifier performance is accuracy: the percent of correct classifications predicted. This measure is easy to understand and makes comparison of the performance of different classifiers trivial. But it does not lend well to observe the performance for each class, especially for those datasets where the classes are not balanced.

The accuracy is the number of correctly predicted observations divided by the total number of observations, and can be calculated with the confusion matrices by:

Accuracy = (TP+TN) / (TP+FN +TN +FP)

where:

- TP: True Positives; an applicant is creditworthy and is classified as creditworthy;
- TN: True Negatives; an applicant is not creditworthy and is classified as not;
- FP: False Positives; an applicant is wrongly detected as being creditworthy;
- FN: False Negatives; an applicant is wrongly detected as being not creditworthy.

For a highly-unbalanced problem, we do not want to overfit to a single class, and the receiver operating characteristic (ROC) is a good performance measure. ROC is a plot showing the trade-off between the rates of correct predictions of creditworthy applicants with the rate of incorrect predictions of creditworthy applicants. The value of Area Under the Curve (AUC) of ROC ranges from 0.5 to 1.00, and the values above 0.80 can be viewed a good discrimination between the two categories of the target variable.

5.4.7 Validation

k-fold cross-validation technique is used to validate the models for assessing how the results generalise to an independent new dataset and to estimate prediction error. The training data were randomly split into k mutually exclusive subsets (folds) of equal size and then an SVM classifier was trained on each subset of data respectively. Each time one of the subsets is left out for training and used to obtain an estimate of the classification accuracy. After the training and accuracy computation is performed k times, an average of the resultant accuracies gives a prediction of the classification accuracy. Cross validation uses all observations in the available data for testing, all the test sets are independent and the reliability of the results is improved.

5.5 Experiments

The aim of this study is credit scoring, which essentially is classification of the credit applicants into two classes as creditworthy or not-creditworthy. The process of credit scoring includes collecting, analysing and classifying different

credit elements and variables to assess the credit decisions into a dataset. We perform following experimental steps to achieve the objective:

- To apply a few classical techniques of classification on the datasets;
- To investigate the effect of varying hyperparameters of the chosen classification models on the results of classification;
- To investigate effect of varying hyperparameters of classifiers;
- To investigate various performance measures for the classification problem.

5.5.1 Experimental Setup

The experiments are run on three benchmarked UCI data sets, which are described in chapter 4. A ten-fold cross validation is performed on the datasets while constructing the model to validate the model for assessing the generalisability. The true/false positive rates and the true/false negative rates were measured and the aggregated classification results over the folds were computed from these results. As part of pre-processing, the categorical/symbolic variables in original dataset are transformed to numerical.

Experiments for the classification models of Linear SVM, KNN and Naïve Bayes are conducted using MATLAB R2017a on Windows 10 operating system on a CPU with an i5 processor with a speed of 3.33 GHz and 16 GB RAM.

5.5.2 Results- Accuracy Performance of the classifiers

First twelve classical machine learning techniques are applied on the three datasets. The experiments conducted show the following accuracy performance.

Figure 5-1 shows the accuracy statistics for all the techniques for the German credit dataset.

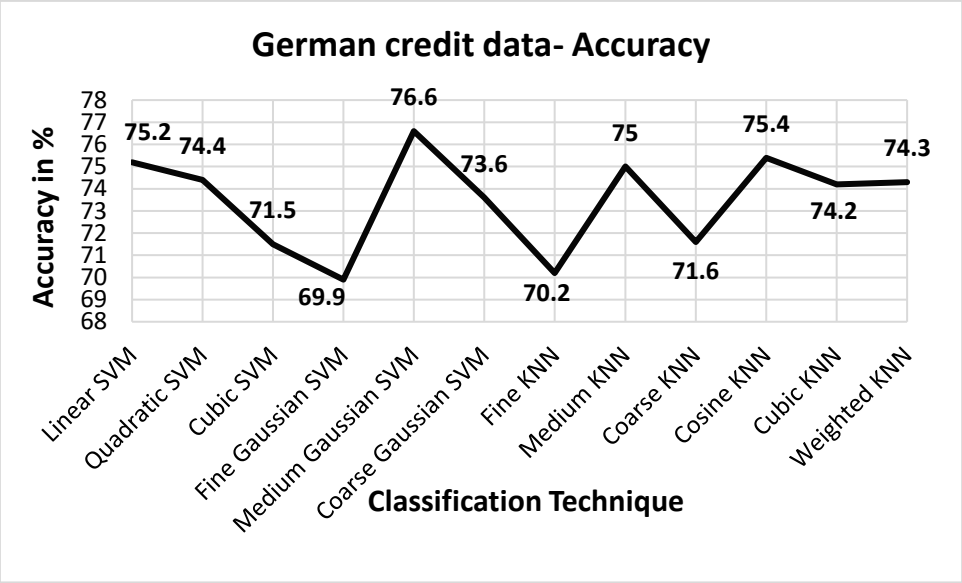

Figure 5-1: Accuracy statistics for the classifiers for German credit dataset

From above figure, for the German credit data:

- SVM- The highest accuracy (76.6%) is yielded by Medium Gaussian SVM
- KNN- The highest accuracy (75.4%) is yielded by Cosine KNN

Figure 5-2 shows the accuracy statistics for all the techniques for the Australian credit dataset.

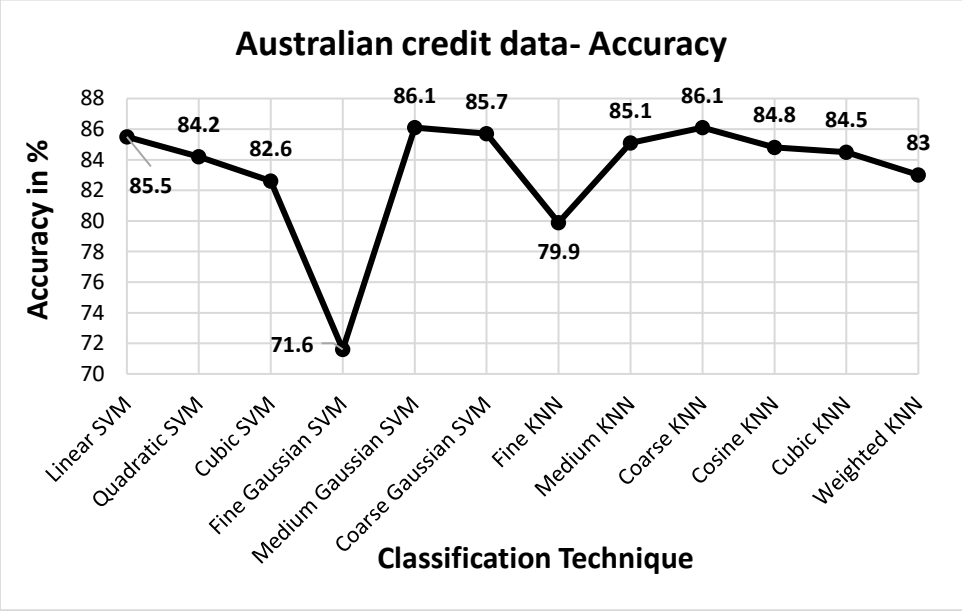

Figure 5-2: Accuracy statistics for the classifiers for Australian credit dataset

From above figure, for the Australian credit data:

- SVM- The highest accuracy (86.1%) is yielded by Medium Gaussian SVM
- KNN- The highest accuracy (86.1%) is yielded by Coarse KNN

Figure 5-3 shows the accuracy statistics for all the techniques for the Taiwan credit dataset.

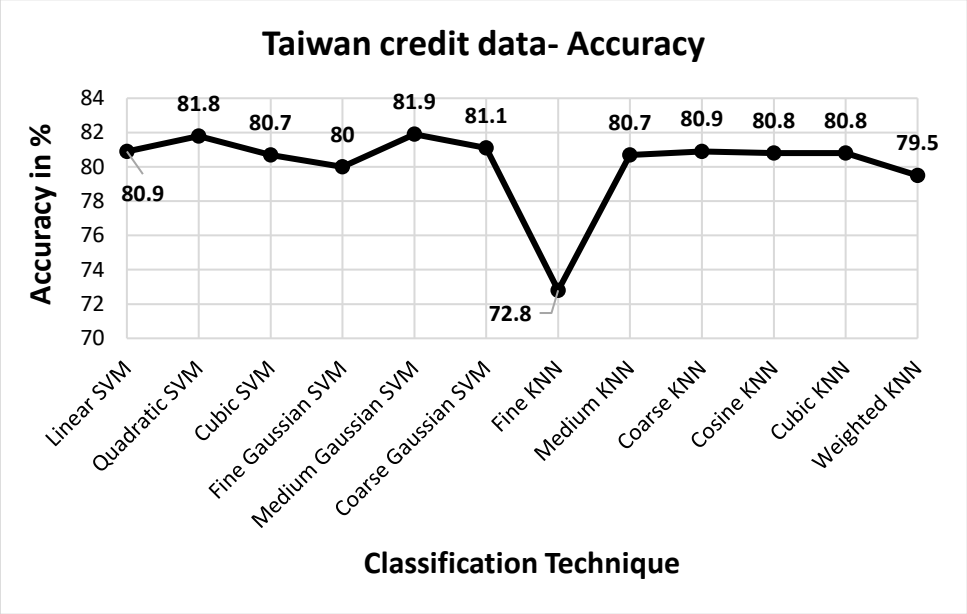

Figure 5-3: Accuracy statistics for the classifiers for Taiwan credit dataset

From above figure, for the Taiwan credit data:

- SVM- The highest accuracy (81.9%) is yielded by Medium Gaussian SVM
- KNN- The highest accuracy (80.9%) is yielded by Coarse KNN

Thus, overall, for all the three datasets, medium gaussian SVM performs best among all twelve models. Hence, we will explore this technique in detail to analyse various performance measures (section 5.5.3) and also effect of varying C and Gamma (γ) values (section 5.5.4, 5.5.5) for this technique. Next, we will explore the corresponding KNN technique with highest accuracy in detail.

5.5.3 Results- Classification Models

As identified in previous section, Medium gaussian SVM yields best classification results over all the three datasets in classifying the applicants as creditworthy or

not. In this section, we will analyse the performance of medium Gaussian, KNN and Naïve Bayes classifiers in detail. The findings of data analysis from chapter 4 will be used to identify the variables contributing most towards the classification of credit applicants. We also analyse the ROC and confusion matrix results of classification for all the datasets.

5.5.3.1 Gaussian SVM classifier for German credit dataset

After analysing the data in previous section, we create Classification models for German credit dataset in this section.

In chapter 4- Data analysis, 'Purpose' and 'Age' were identified to be able to separate the two classes well for this dataset. Following figure shows the Model predictions with these two columns as predictors. The figure shows both correctly identified points with filled circles and incorrectly identified ones with crosses for both the blue (class 0) and red (class 1) target classes.

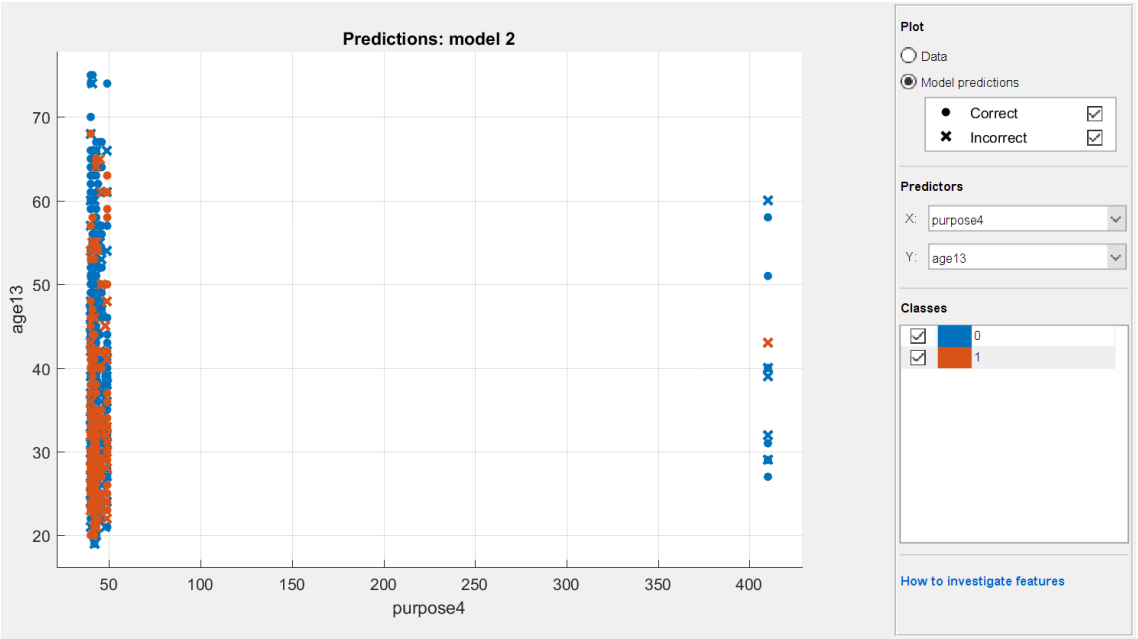

Figure 5-4: Model predictions for Gaussian SVM on German credit data using 'Purpose' and 'Age' as predictors

We also observe only correctly identified instances and incorrectly identified ones in following two figures for the same predictors mentioned above.

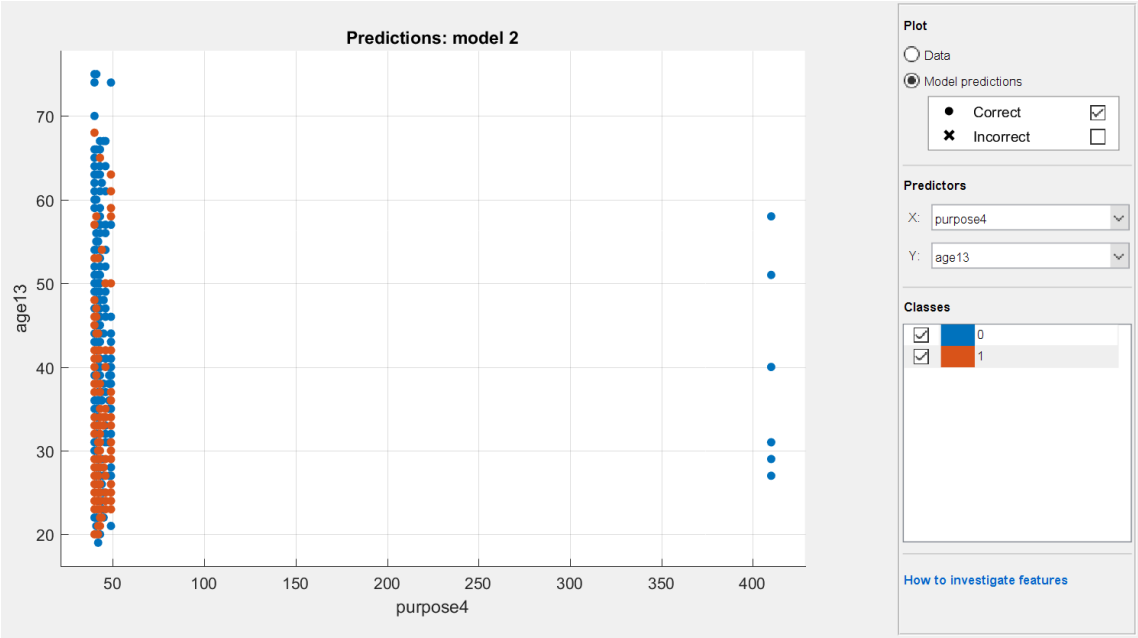

Figure 5-5: Correct points identified by Gaussian SVM on German credit data using columns 'Purpose' and 'Age' as predictors

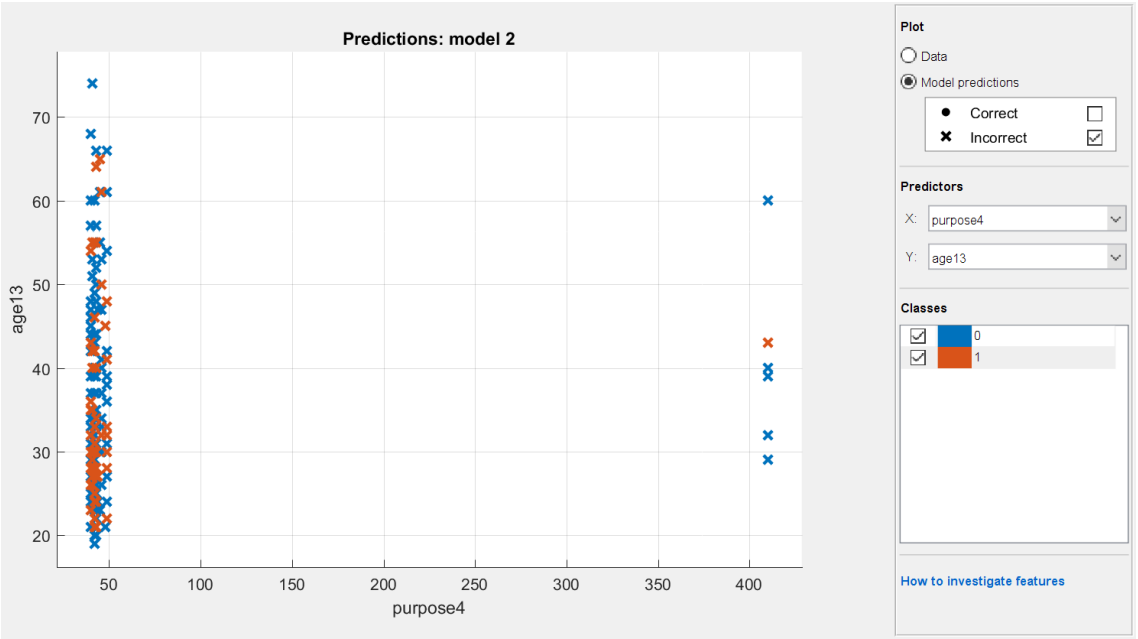

Figure 5-6: Incorrect points identified by Gaussian SVM on German credit data using columns 'Purpose' and 'Age' as predictors

The result of Gaussian SVM classifier can be evaluated with the help of ROC. ROC curves are an indication of the predictive power of the model and show true and false positive rates. The marker on the plot shows the performance of the

classifier which is the values of the false positive rate (FPR) and the true positive rate (TPR). The area under the ROC curve (AUC) is a measure of how well a parameter can distinguish between two classes (creditworthy/non-creditworthy). Large AUC values indicate better classifier performance.

Predictive Power	Area under ROC
Acceptable	>70%
Good	>80%
Very Good	>85%

A perfect result with no misclassified points is a right angle to the top left of the plot. A poor result that is no better than random is a line at 45 degrees.

The ROC curve for above classifier is shown below:

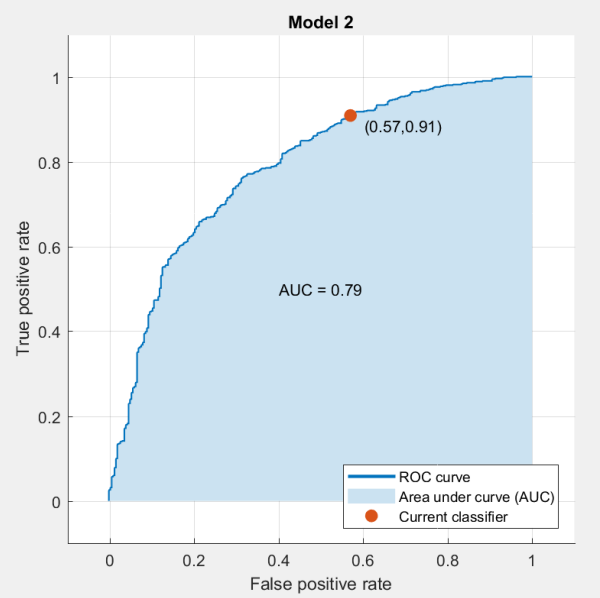

Figure 5-7: ROC curve for the Gaussian SVM model on German credit data

e.g. from above figure, a false positive rate (FPR) of 0.57 indicates that the current classifier assigns 57 of the observations incorrectly to the positive class. A true positive rate of 0.91 indicates that the current classifier assigns 91% of the observations correctly to the positive class.

Another way of performance analysis of a classifier is with the help of Confusion matrix:

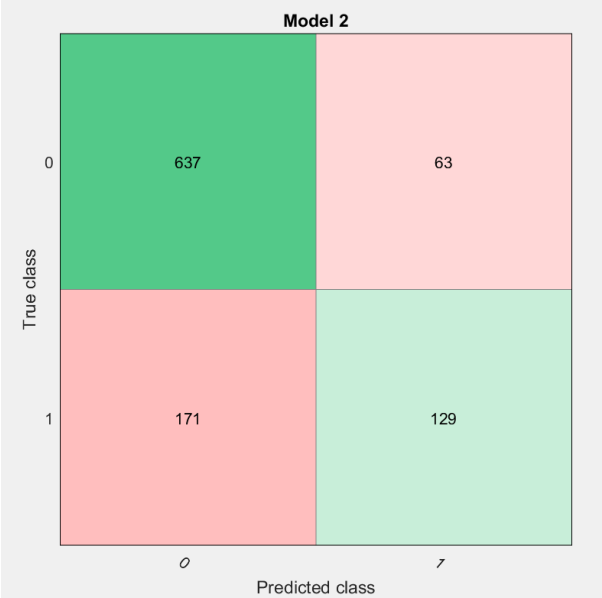

Figure 5-8: Confusion Matrix for number of observations for Gaussian SVM model on German credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

Confusion matrix shows how the classifier performed in each class. It helps identify the areas where the classifier has performed poorly. The rows show the true class, and the columns show the predicted class. The major diagonal cells show where the true class and predicted class match. If these cells are green, the classifier has performed well and classified observations of this true class correctly.

In this dataset, class 0 is the 'Creditworthy' and class 1 is 'Non-creditworthy' applicant. From the confusion matrix above, the classification table is:

	Classified as	
Test data	**Creditworthy**	**Non-creditworthy**
Creditworthy	637	63
Non-creditworthy	171	129

In 63 cases, true class of the applicant was 'Creditworthy' but the model misclassified as 'Non-creditworthy'. For 171 cases, the true class was 'Non-creditworthy' but the model misclassified as Creditworthy'.

Hence,

Overall Accuracy= (637+129)/1000=76.6%

Credit scoring models consider the characteristics of good as well as bad payers. Also, the model could handle very large dataset and come up with unbiased results.

If a decision of granting a credit is to be made based on judgemental methods which consider the characteristics of only those who were granted the loan, the decision would generally be biased towards awareness of bad payers only. In above example, the decision maker has to decide what to value the most, the good applicants classified as bad or vice-versa.

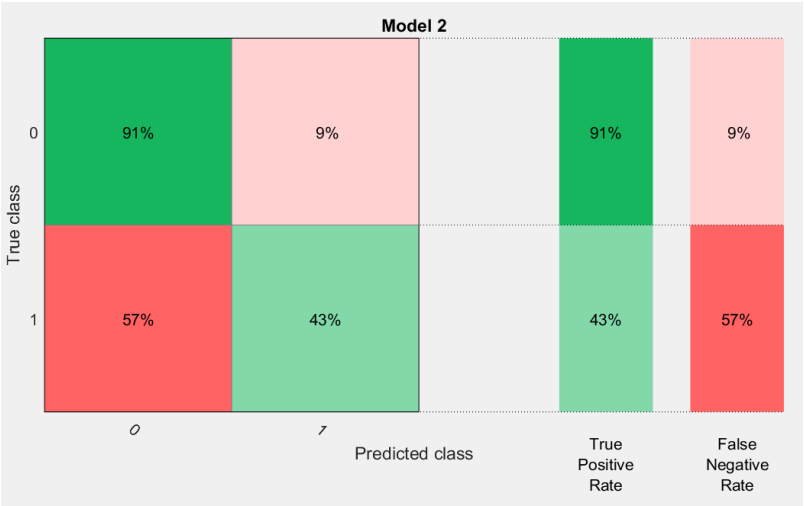

Figure 5-9: Confusion matrix showing True positive rate and False negative rate for Gaussian SVM model on German credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

Figure 5-9 shows same data in terms of percentage. We conclude that this model is good at predicting the ‘good credit’ class (91% accuracy), but not very good at predicting the ‘bad credit’ class (43% accuracy).

This information could help a decision maker choose the best model for his/her goal. If false positives in ‘good credit’ are very important to the classification problem in hand, then choose the best model at predicting this class. If false positives in ‘good credit’ are not very important, and models with fewer predictors

do better in other class, then choose a model to trade-off some overall accuracy to exclude some predictors and make future data collection easier.

A parallel coordinates plot is shown below to visualise the high dimensional data; 1 polyline in the plot is 1 data point with vertices on parallel axis with no order to vertical axes (see section 4.2.4.1).

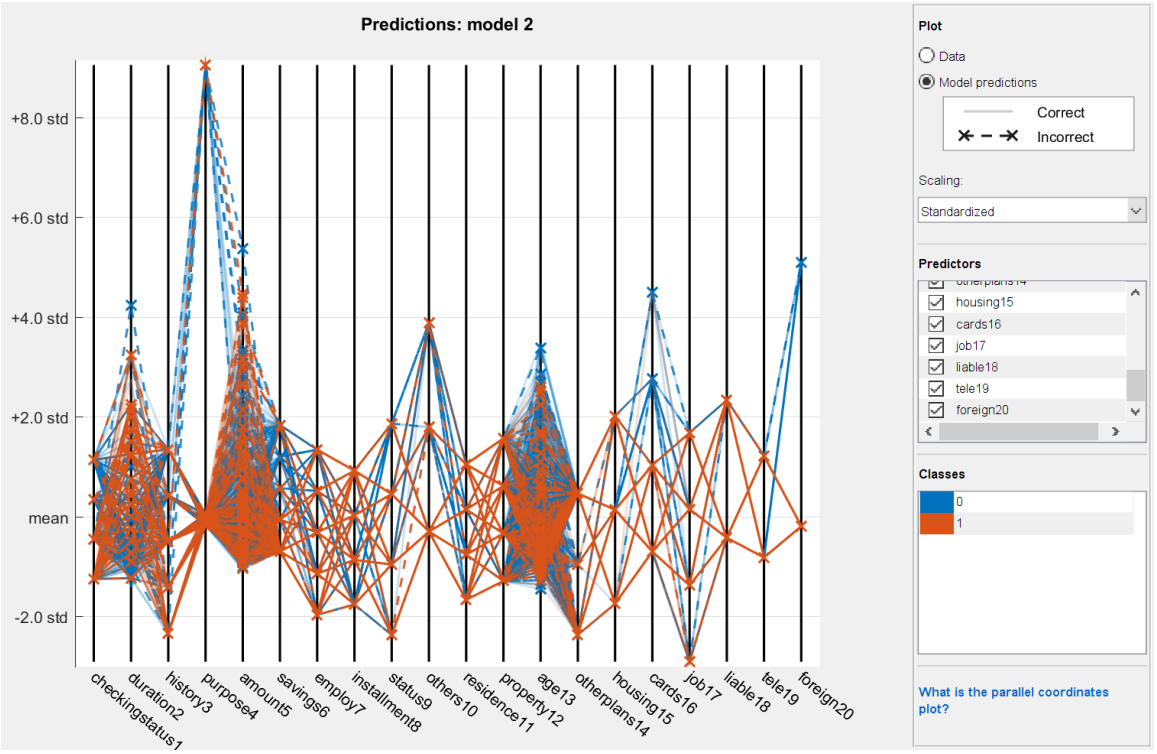

Figure 5-10: Parallel coordinates plot for Gaussian SVM model on German credit data

5.5.3.2 KNN classifier for German credit dataset

This section describes the KNN classifier performance for German credit dataset.

The two diagrams below show the incorrect and correct data points identified using ‘Purpose and ‘Age’ as predictors for KNN classifier.

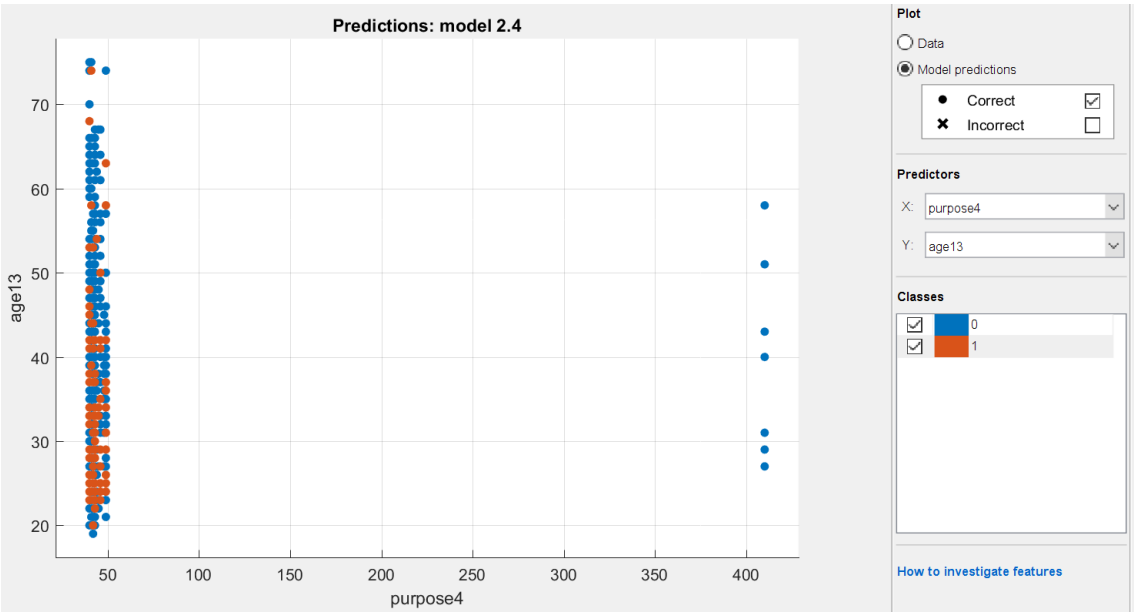

Figure 5-11: Correct points identified by KNN on German credit data using 'Purpose' and 'Age' as predictors

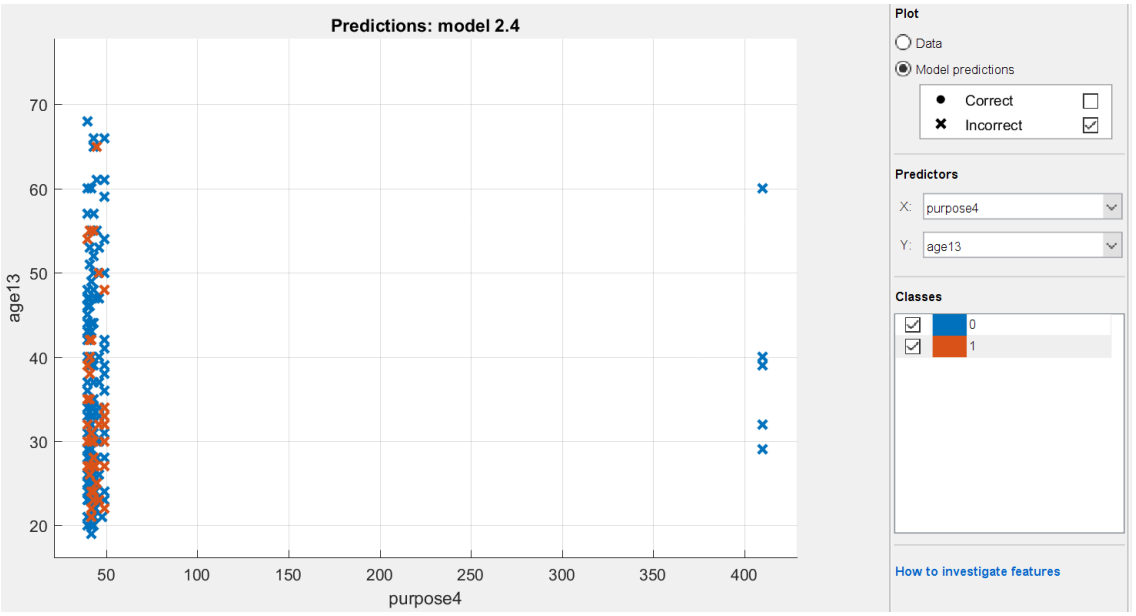

Figure 5-12: Incorrect points identified by KNN on German credit data using 'Purpose' and 'Age' as predictors

The ROC plot shown below for above KNN model shows AUC= 75%.

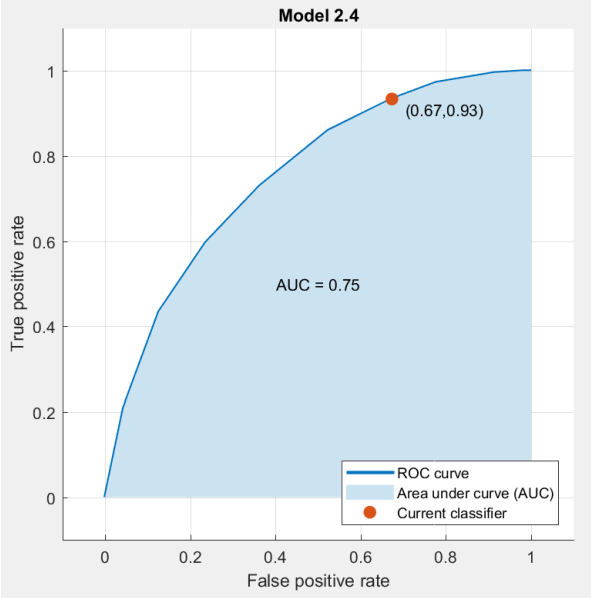

Figure 5-13: ROC curve for KNN classifier on German credit data

The Confusion matrix below shows 654 observations from class 0 (class 'good credit') as correctly classified and 98 from class 1 (class 'bad credit') as correctly classified.

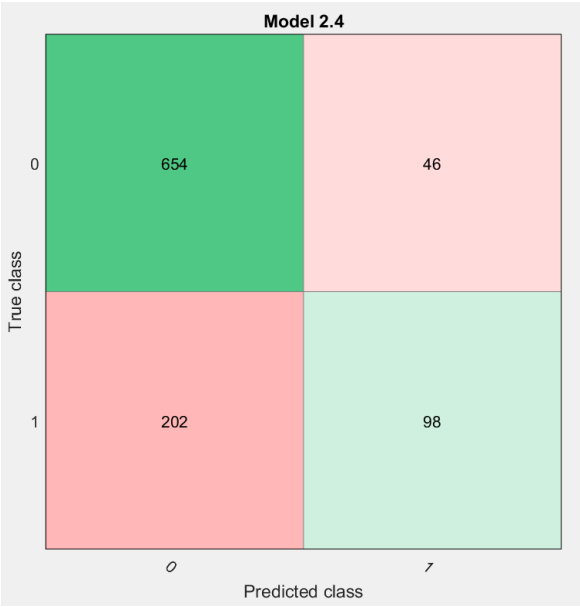

Figure 5-14: Confusion matrix for number of observations for KNN classifier on German credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

Thus, the overall accuracy for this model is (654+98)/ (654+46+202+98) = 75.2%.

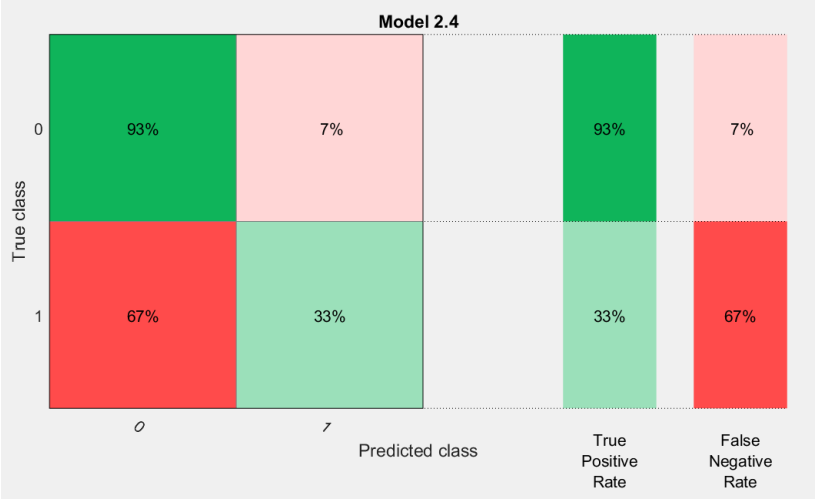

Figure 5-15: Confusion matrix for True positive class and False negative class for KNN classifier on German credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

Above variant of confusion matrix shows performance of the model in terms of percentage.

Parallel coordinates plot for model predictions for KNN classifier:

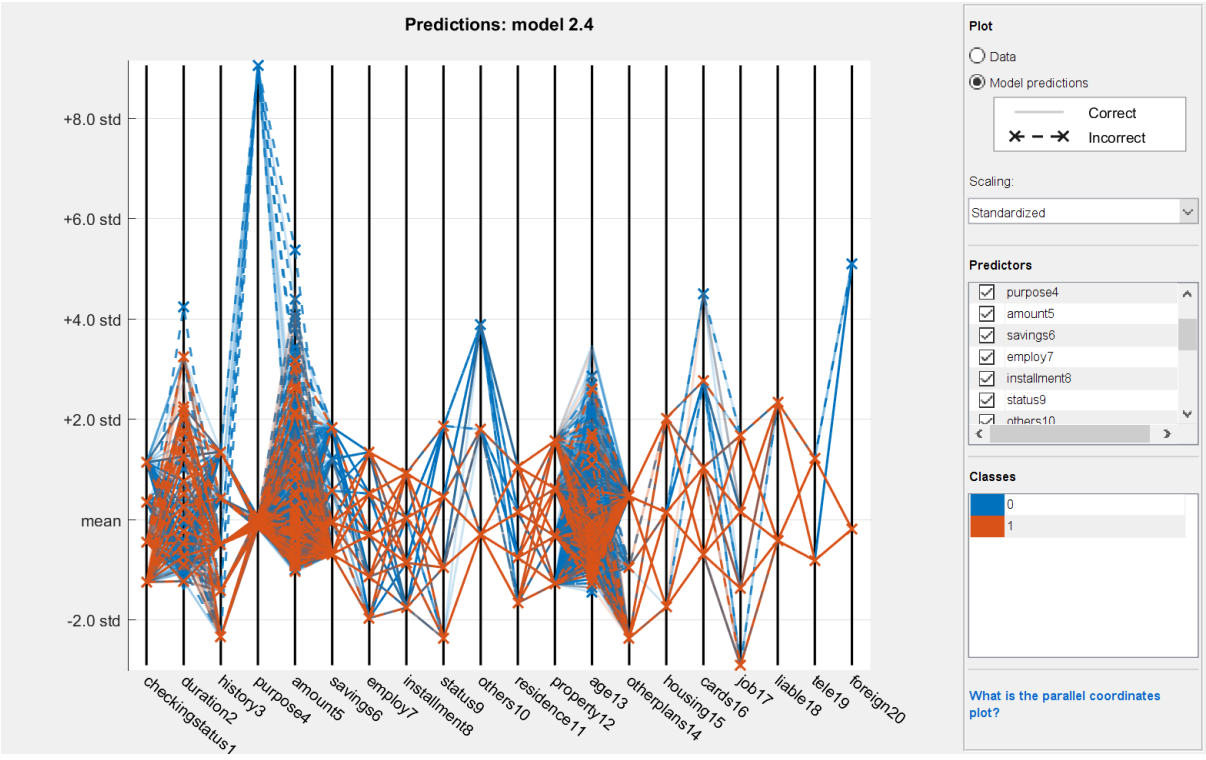

Figure 5-16: Parallel coordinates plot for KNN classifier on German credit data

5.5.3.3 Naive Bayes classifier for German credit dataset

The Naïve Bayes classifier yields accuracy equal to 73.6% and AUC as 78.58%.

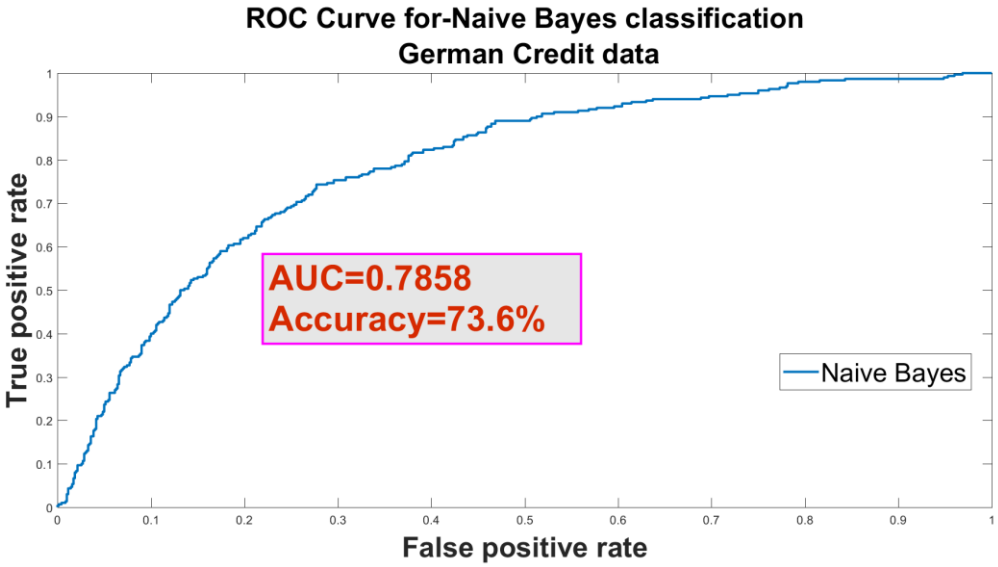

Figure 5-17: ROC curve for Naïve Bayes classifier on German credit data

Thus, we can see that the ROC curve statistics for German credit dataset for the Gaussian SVM, KNN and Naïve Bayes models are 78%, 75%, 78.58% respectively. The Naïve Bayes classification model performs marginally better than Gaussian SVM, whereas KNN's performance is the least among all.

In next three sections, we will apply the three classification models to Australian credit dataset.

5.5.3.4 Gaussian SVM classifier for Australian credit dataset

The Australian credit dataset does not identify the attributes with names.

Figure 5-18 shows that attributes 14 and 13 are good at predicting both classes (-1 for 'bad credit', 1 for 'good credit') for Gaussian SVM.

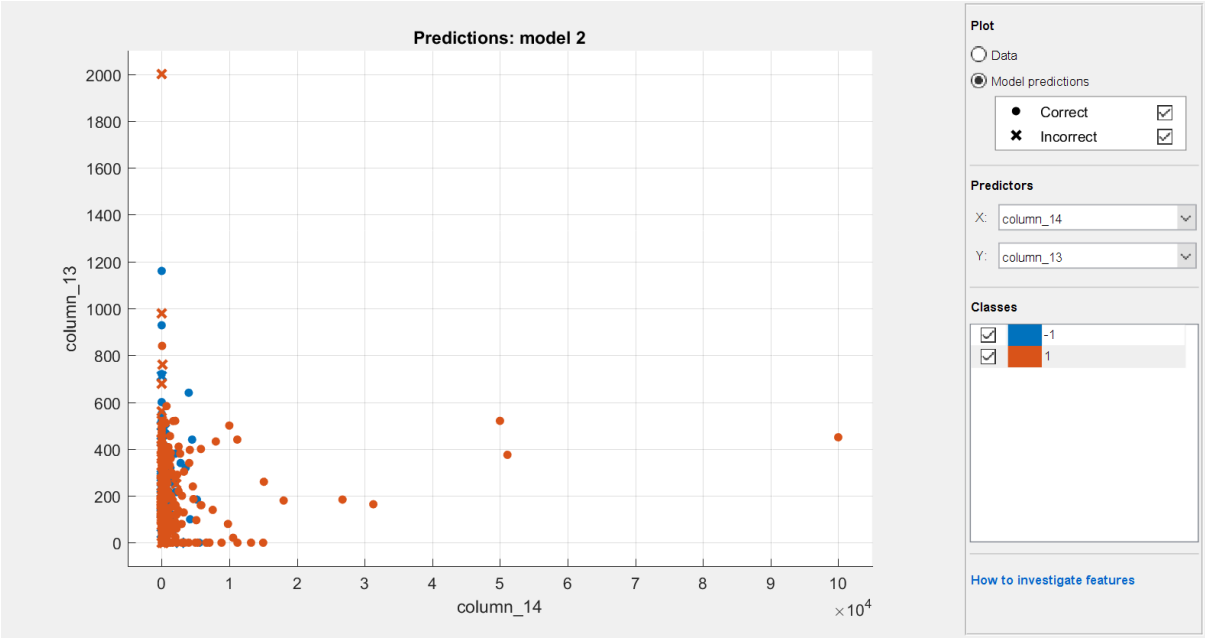

Figure 5-18: Model predictions by Gaussian SVM on Australian credit data using columns 14 and 13 as predictors

The two figures below show the correctly and incorrectly identified points by this model, which indicate that this model's performance is good. This will be verified later by ROC curve and confusion matrix.

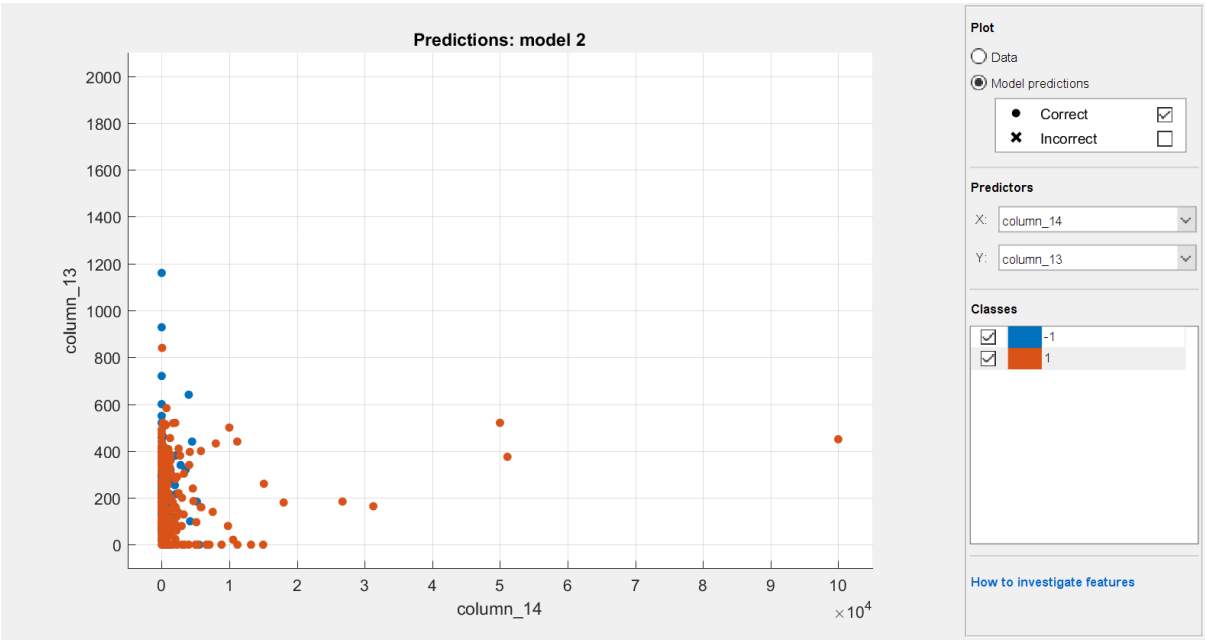

Figure 5-19: Only correct points identified by Gaussian SVM on Australian credit data using columns 14 and 13 as predictors

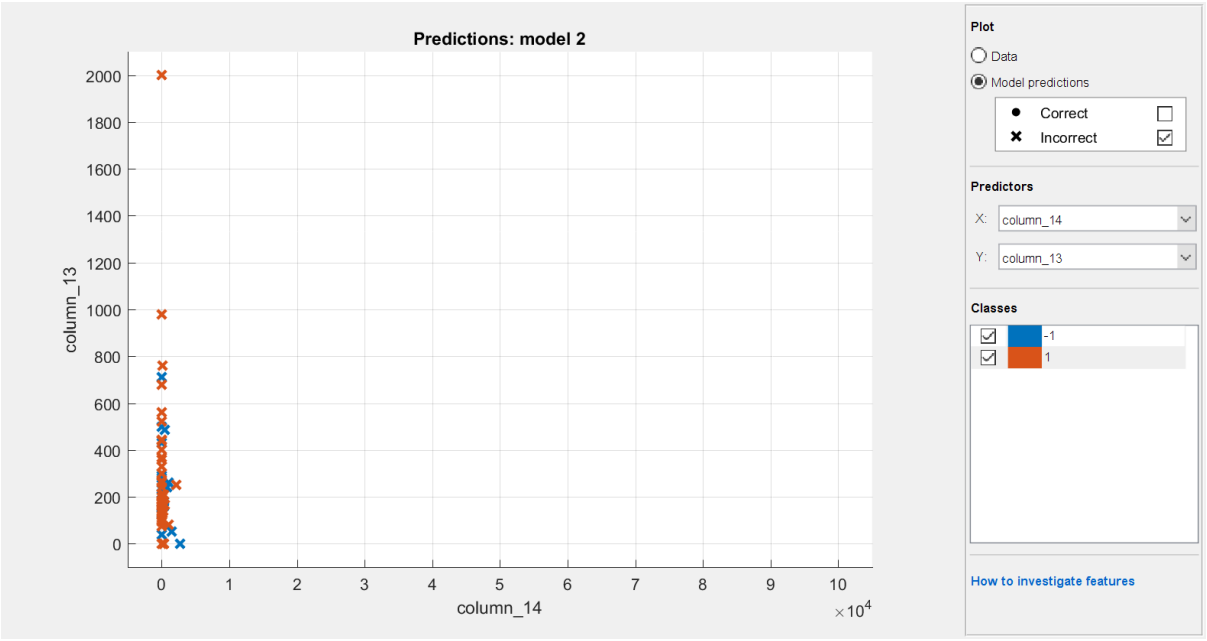

Figure 5-20: Only incorrect points identified by Gaussian SVM on Australian credit data using columns 14 and 13 as predictors

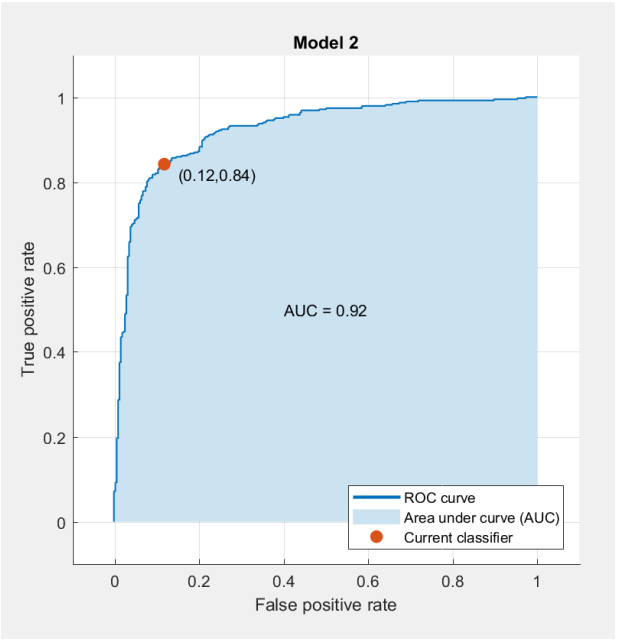

Figure 5-21: ROC curve for Gaussian SVM classifier on Australian credit data

This curve shows good area under curve and hence good performance of this model.

Let us observe the performance of this model using the confusion matrix metric.

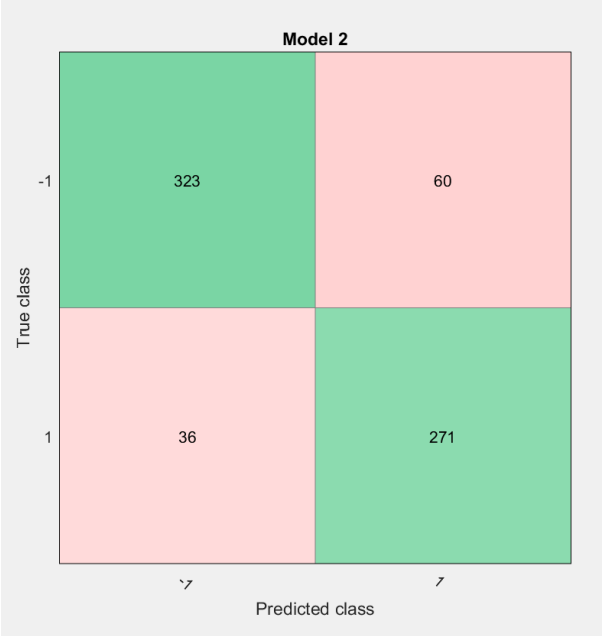

Figure 5-22: Confusion matrix for Gaussian SVM classifier on Australian credit data (Class 1: Creditworthy, Class -1: Non-creditworthy).

Thus, overall accuracy of this model= (323+271)/ (323+271+60+36) =86.08%.

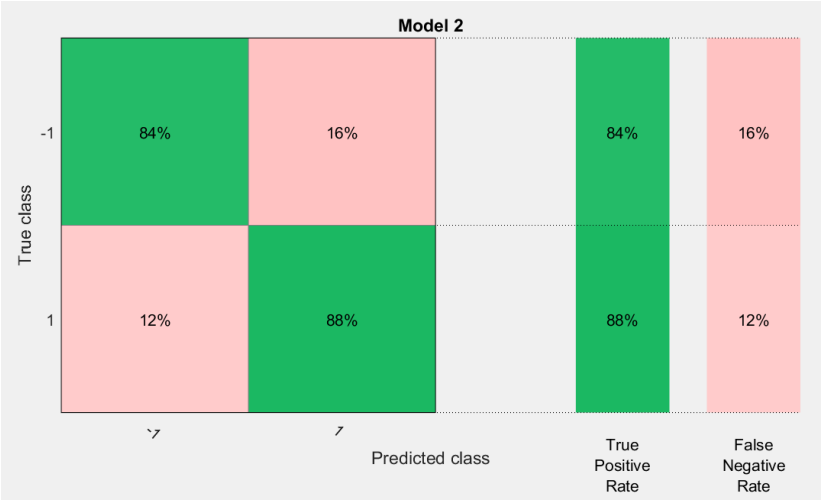

Figure 5-23: Confusion matrix for Gaussian SVM classifier on Australian credit data (Class 1: Creditworthy, Class -1: Non-creditworthy).

From above figure, the 'good credit' class (class 1) is predicted with 88% accuracy, whereas the 'bad credit' class (class -1) is predicted with 84% accuracy. Thus, this classification model gives good performance.

16% ‘good credit’ clients are incorrectly classified as being ‘bad credit’. Similarly, 12% ‘bad credit’ clients are misclassified as ‘good credit’. Thus 16% is the False Negative Rate for incorrectly classified points in this class, shown in the red cell in the False Negative Rate column. Again, this information is used by the decision maker based on the need if the model as is mentioned for German credit dataset in previous section.

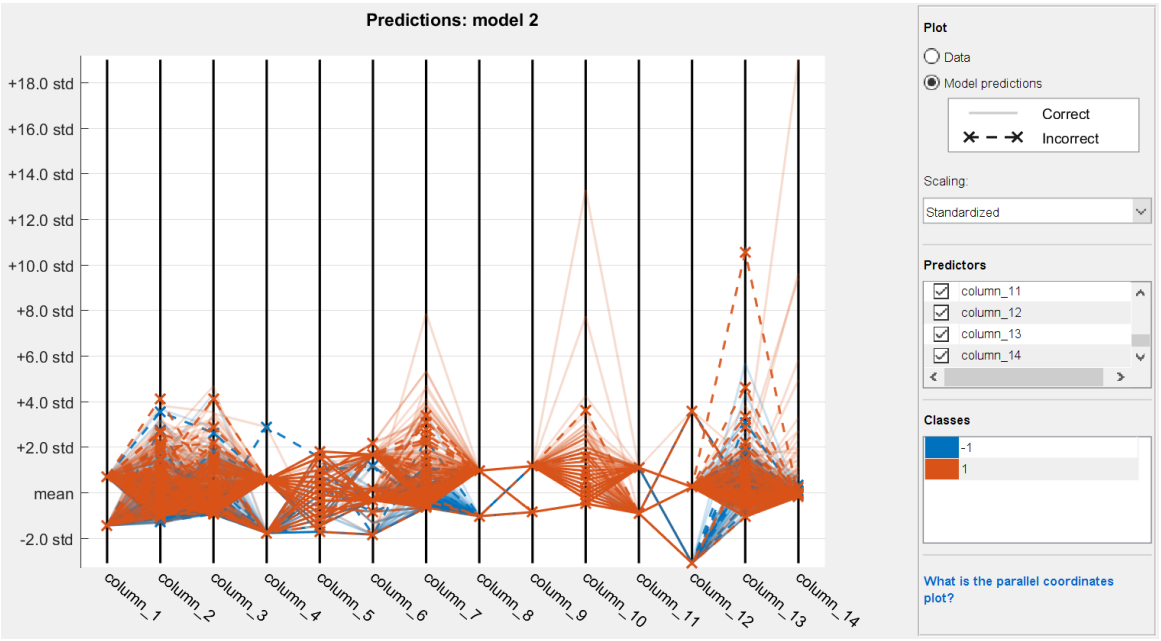

Figure 5-24: Parallel coordinates plot for Gaussian SVM classifier on Australian credit data

Above figure displays the performance of this classifier towards prediction with respect to each of the variable in the dataset. Column 13 has misclassified quite a few points of class1 shown by dashed red lines.

5.5.3.5 KNN classifier for Australian credit dataset

Here, first we observe KNN classifier’s predictions in terms of correctly and incorrectly identified observations for both the classes (-1 for ‘bad credit’ and 1 for ‘good credit’)

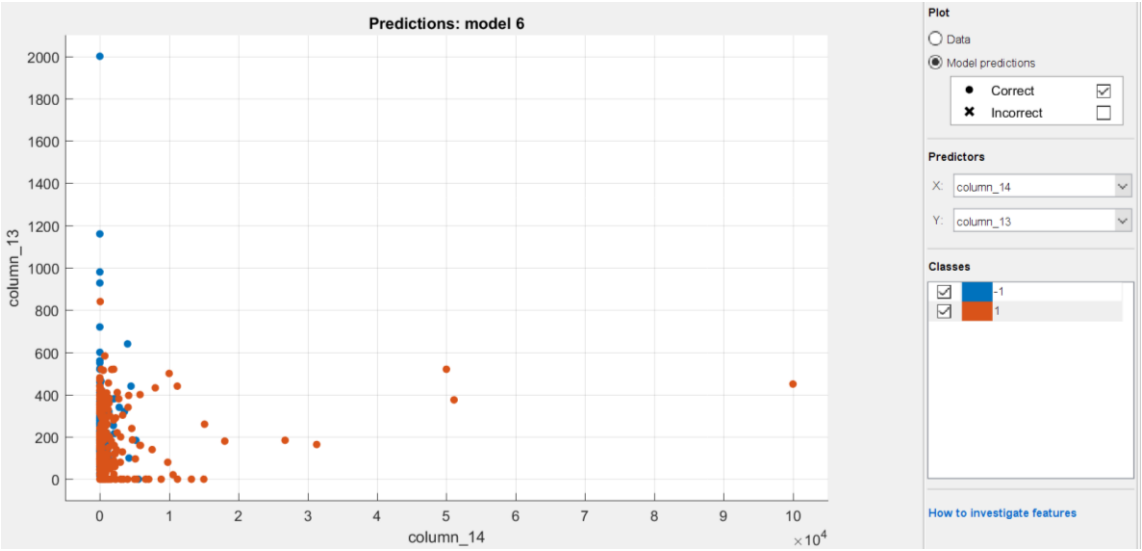

Figure 5-25: Correctly identified points by KNN on Australian credit data using columns 14 and 13 as predictors

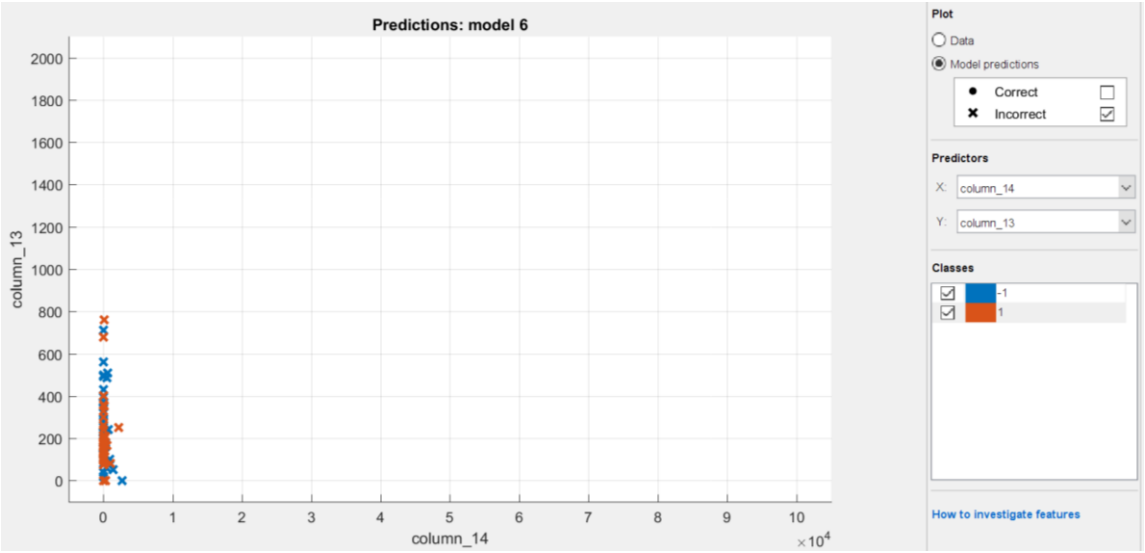

Figure 5-26: Incorrectly identified points by KNN on Australian credit data using columns 14 and 13 as predictors

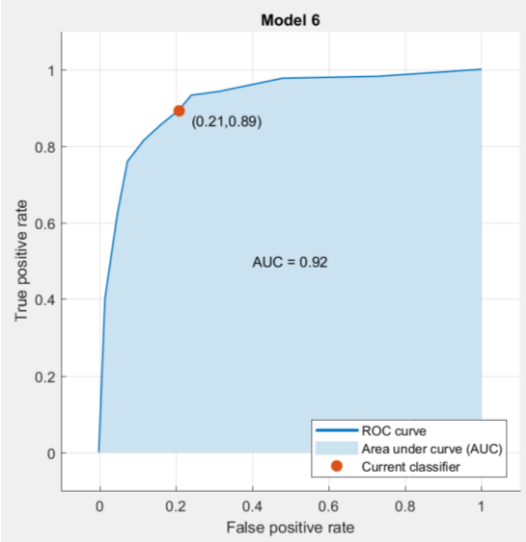

Figure 5-27: ROC curve for KNN classifier on Australian credit data

This ROC curve indicates good performance of this model since the are-under-curve is 92%.

We see the confusion matrix statistics below for the same model.

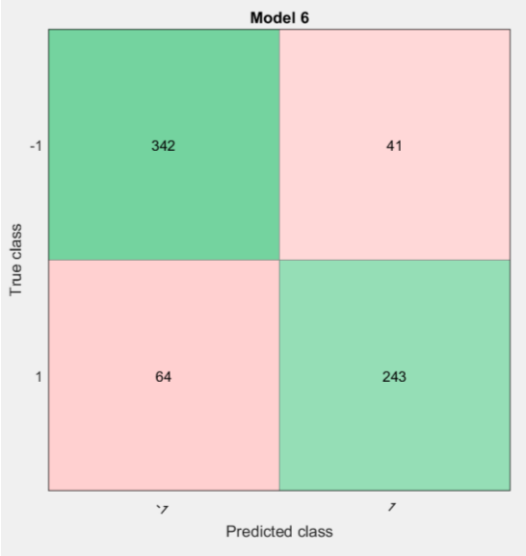

Figure 5-28: Confusion matrix for KNN classifier on Australian credit data (Class 1: Creditworthy, Class -1: Non-creditworthy).

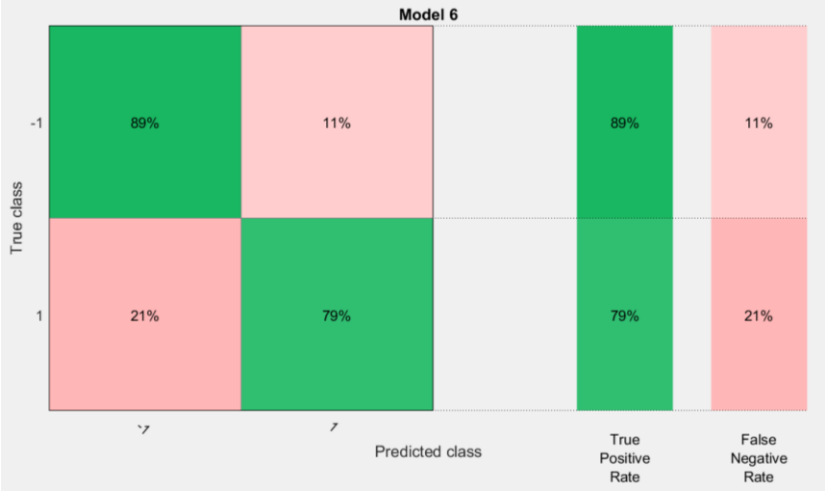

Figure 5-29: Confusion matrix for KNN classifier on Australian credit data (Class 1: Creditworthy, Class -1: Non-creditworthy).

The model is able to predict 342 cases of bad credit and 243 cases of good credit correctly which translates to 89% and 79% respectively.

5.5.3.6 Naïve Bayes for Australian credit dataset

The Naïve Bayes classifier's performance is 80.28% in terms of accuracy.

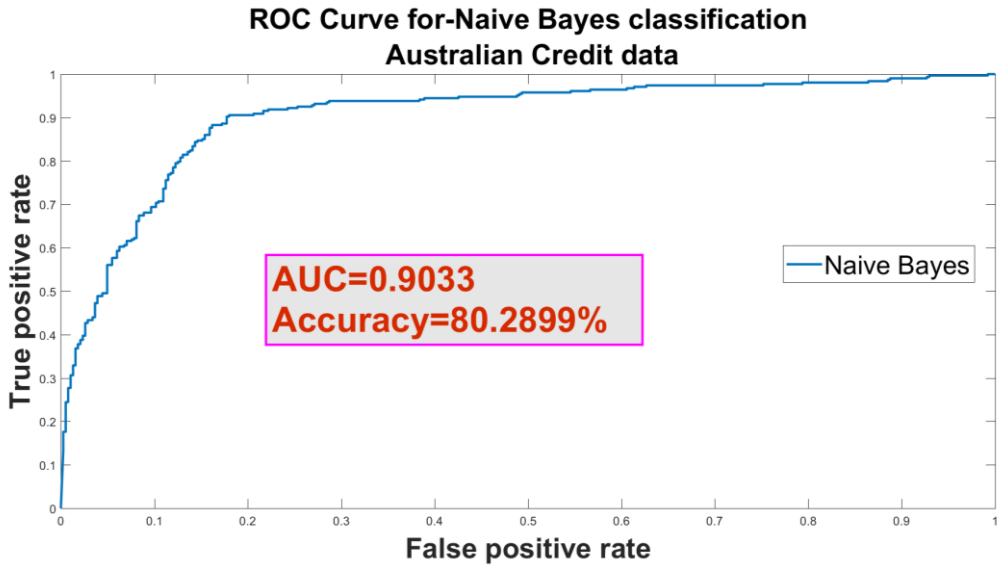

Figure 5-30: ROC curve for Naïve Bayes classifier on Australian credit data

Thus, the Gaussian SVM, KNN and Naïve Bayes classifier yield the accuracy of: 92%, 92% and 80.28% respectively. The SVM and KNN give better performance than NB.

5.5.3.7 Gaussian SVM for Taiwan credit dataset

We apply the three classification models on Taiwan credit dataset.

Below we see a pair of example features and how they are performing in predicting the class of clients.

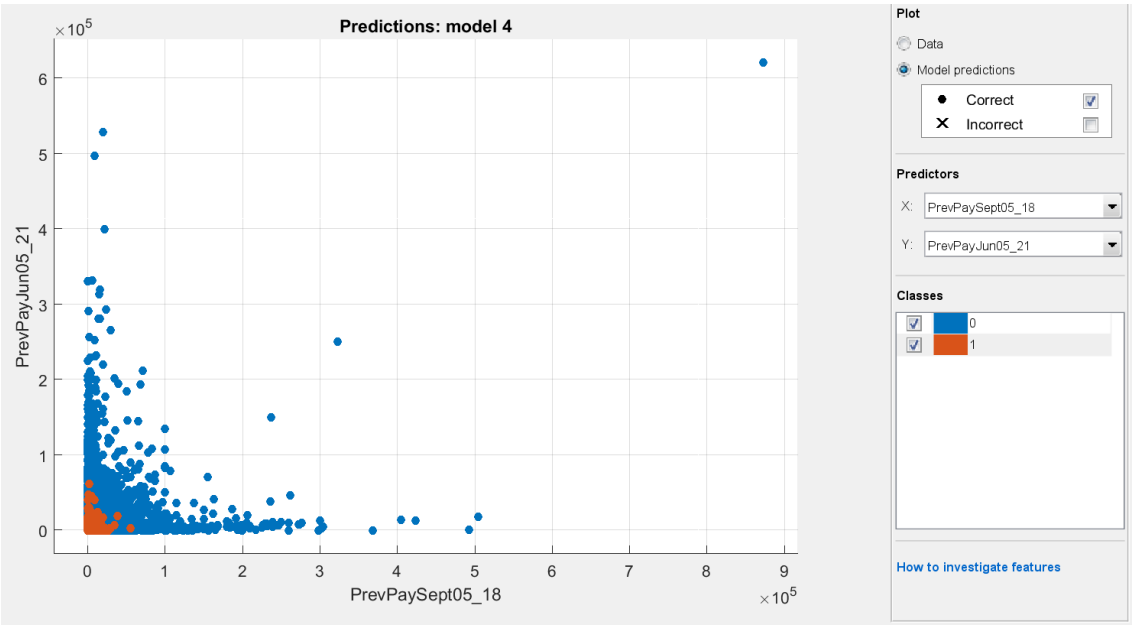

Figure 5-31: Model predictions for Gaussian SVM classifier for Taiwan credit dataset

Above figure shows that correct points are identified by Linear SVM on Taiwan credit data using columns 18 and 21 as predictors.

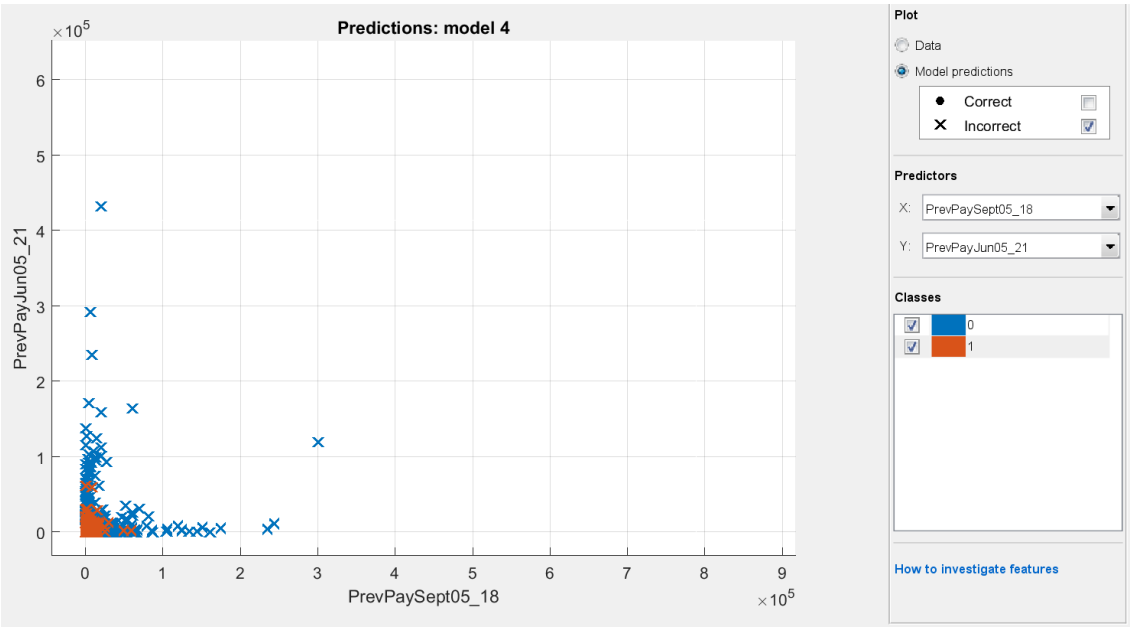

Figure 5-32: Incorrect points identified by Gaussian SVM on Taiwan credit data using columns 18 and 21 as predictors

The ROC curve and confusion matrix below shows the performance of this model.

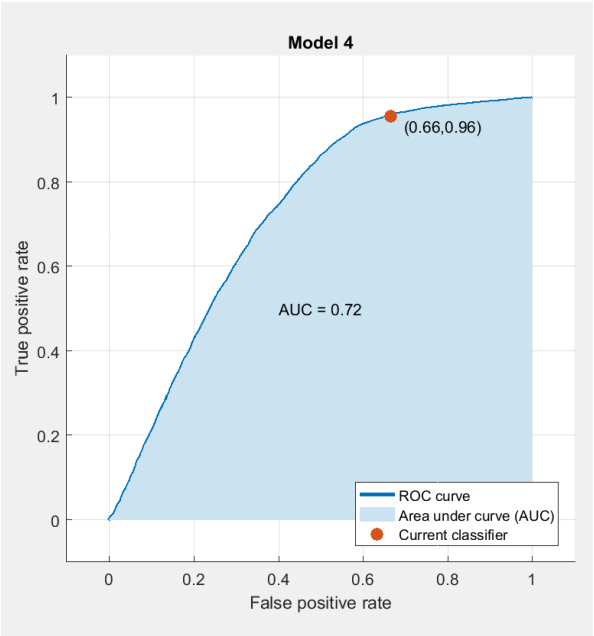

Figure 5-33: ROC curve for Gaussian SVM classifier on Taiwan credit data

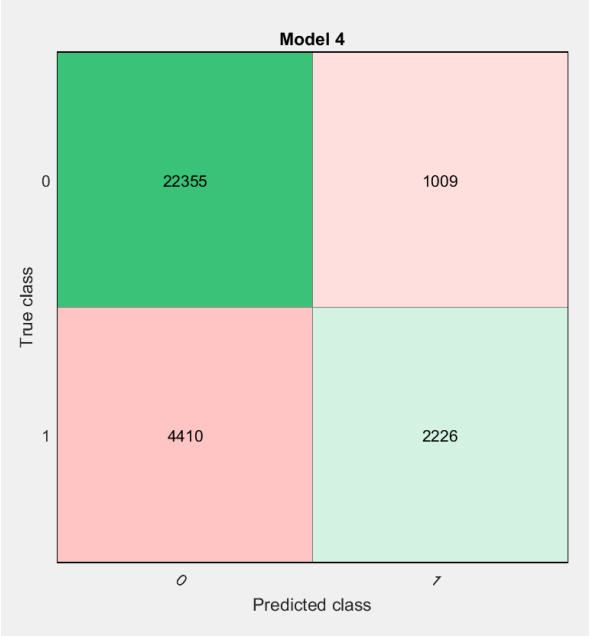

Figure 5-34: Confusion matrix for Gaussian SVM classifier on Taiwan credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

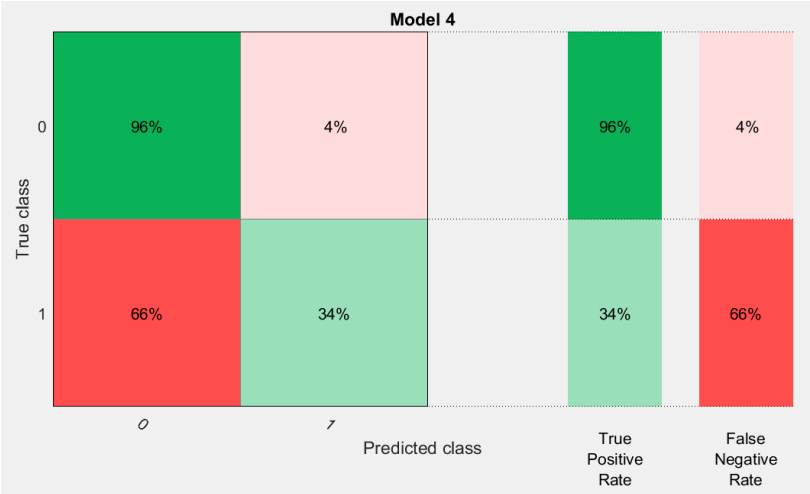

Figure 5-35: Confusion matrix for Gaussian SVM classifier on Taiwan credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

The ‘good credit’ class (class 0) is predicted 96 times out of 100 times, whereas the ‘bad credit’ class (class 1) is predicted 34 times out of 100 times in above model.

5.5.3.8 KNN classifier for Taiwan credit dataset

We will see how KNN classifier identifies the class correctly and incorrectly using a pair of attributes.

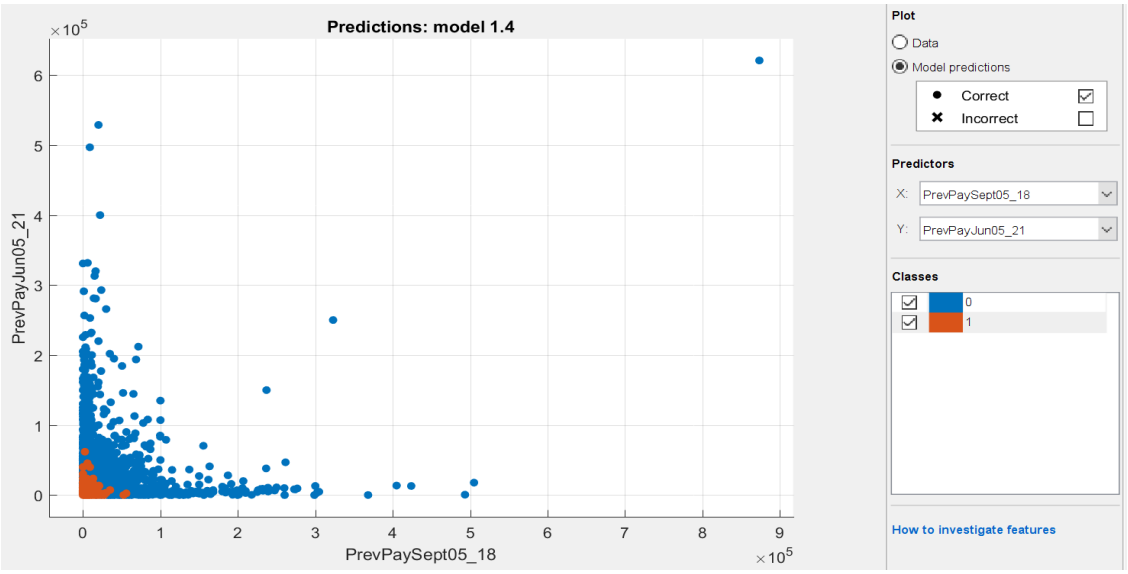

Figure 5-36: Correct points identified by KNN classifier for Taiwan credit dataset using 'Payment done in September 2005' and 'Payment done in June 2005' as predictors

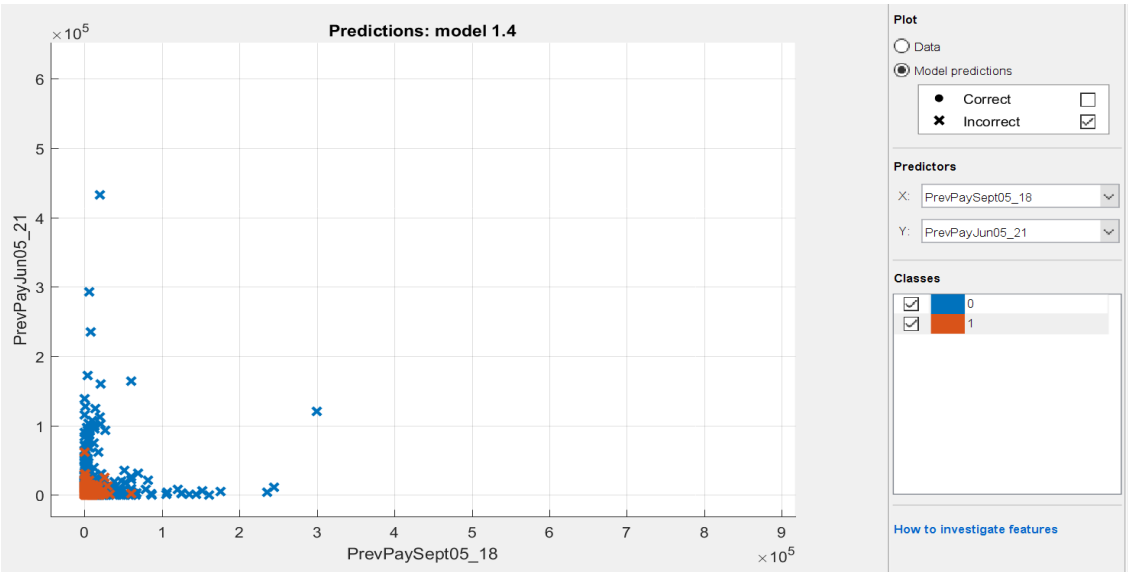

Figure 5-37: Incorrect points identified by KNN classifier for Taiwan credit dataset using 'Payment done in September 2005' and 'Payment done in June 2005' as predictors

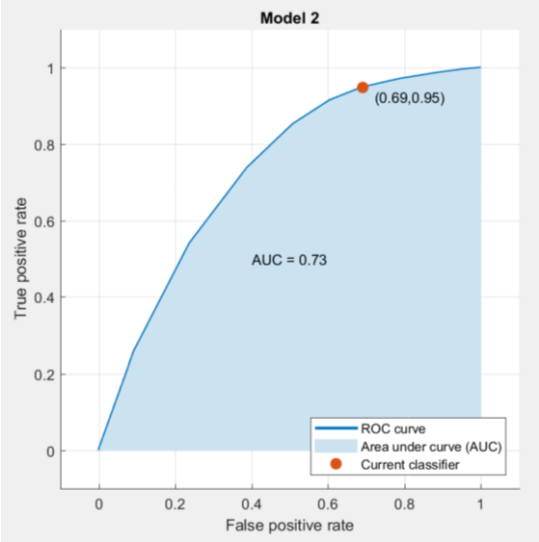

Figure 5-38: ROC curve for KNN classifier on Taiwan credit data

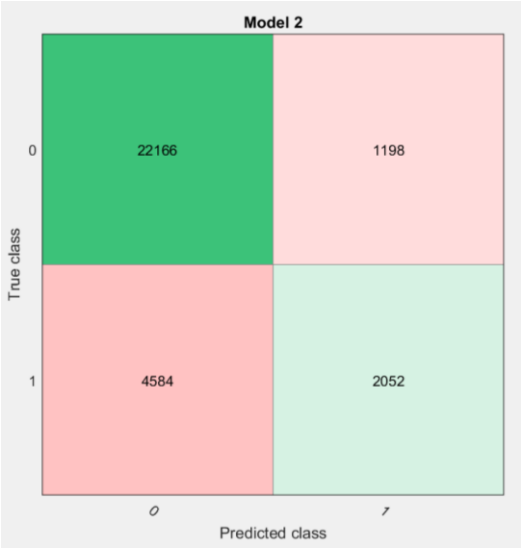

Figure 5-39: Confusion matrix for KNN classifier on Taiwan credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

The overall accuracy of this model is:

(22166+2052) / (22166+2052+1198+4584) = 80.72%

This model yields 95% accuracy in predicting class 0 (good credit) and 31% in predicting class 1(bad credit), as seen below.

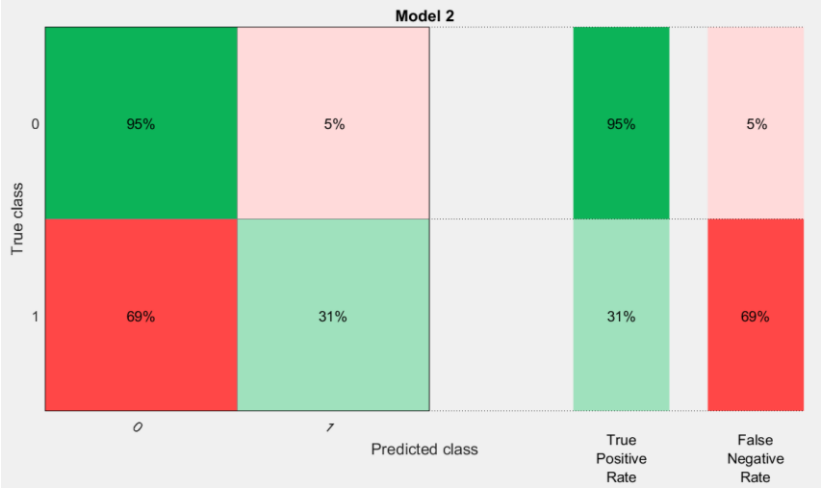

Figure 5-40: Confusion matrix percentage for KNN classifier on Taiwan credit data (Class 0: Creditworthy, Class 1: Non-creditworthy).

5.5.3.9 Naïve Bayes for Taiwan credit dataset: Accuracy= 71.03%

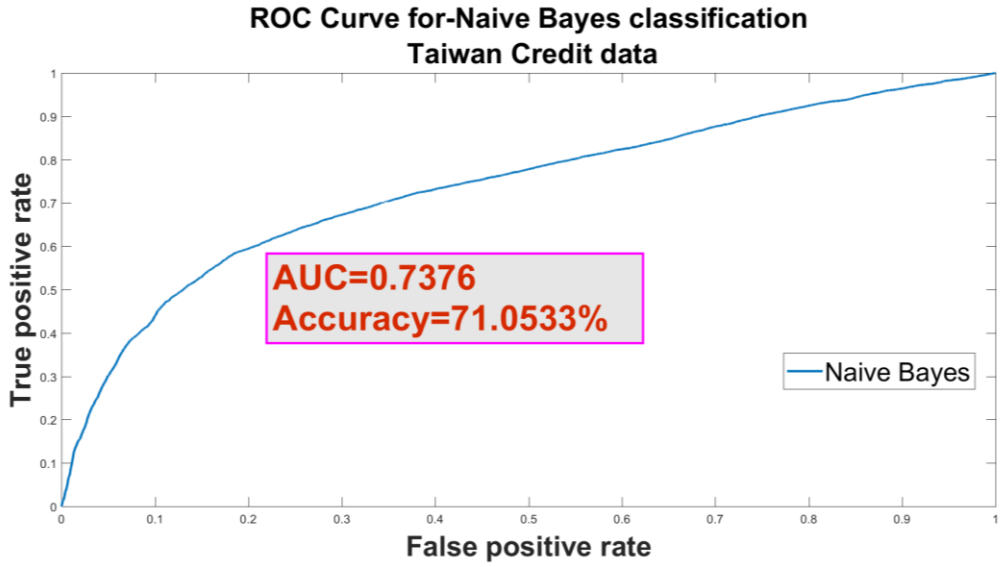

Figure 5-41: ROC curve for Naïve Bayes classifier on Taiwan credit data

5.5.4 Results- Effect of varying Cost of Penalty on the accuracy of Gaussian SVM classifier

We identified in section 5.5.2 that medium gaussian SVM performs best over all the datasets. In this section, we investigate the performance of this classifier for different values of cost of penalty (C) for the three datasets.

The C parameter trade-off between the training error and the flatness of the decision surface. A low C ensures a smooth decision surface, while a high C aims to classify all training examples correctly by giving the model freedom to select more samples as support vectors. Very high value of C may result in loss of generalisation properties of the classifier, because it will try to fit as best as possible all the training points.

The method followed is:

- Apply all three types of Gaussian SVM: Fine, Medium and Coarse with default settings;
    - The default settings are: Kernel Function = Gaussian, C=1
        - For Fine Gaussian, γ = sqrt (P) /4 (where P is number of predictors in the dataset);
        - For Medium Gaussian, γ = sqrt (P);
        - For Coarse Gaussian, γ = sqrt (P)*4;
- Select the one which yields best accuracy as a pre-set one and vary values of C. Choose the C which gives highest accuracy; See next three figures for all the three datasets.
- Use the value of C from previous step and vary value of γ (section 5.5.5).

For Gaussian SVM, following figures show the effect of varying values of parameter C on accuracy of the classification for the datasets.

The figure below indicates that the accuracy of the model for Gaussian SVM increases as the value of C increases from 0.001 to 1, but it plateaus after reaching 75.9% for the German credit dataset.

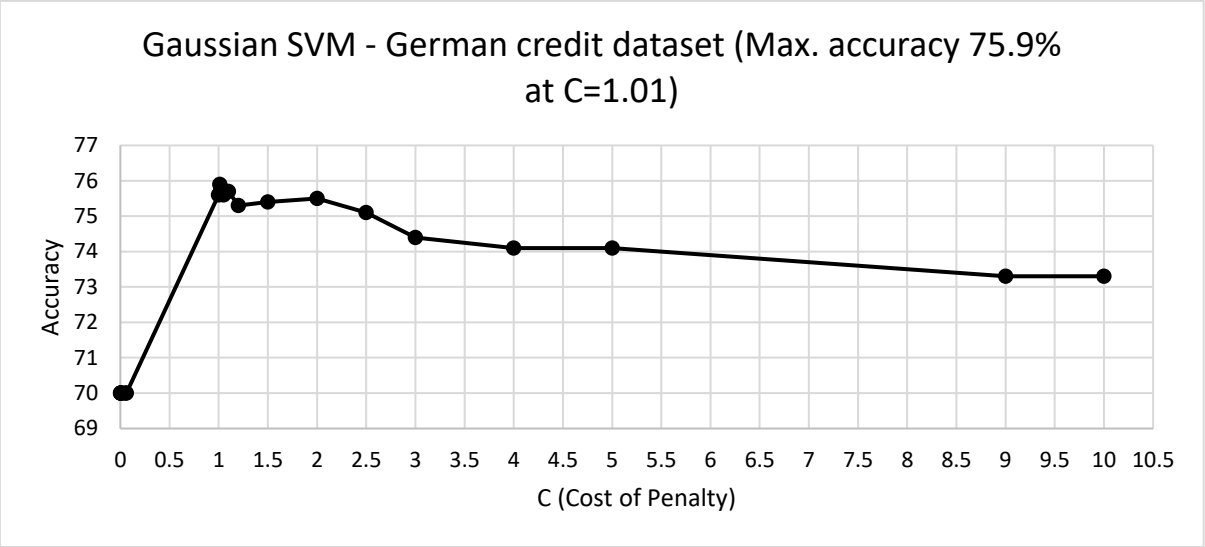

Figure 5-42: Effect of varying C value on accuracy of Gaussian SVM for German credit dataset

From figure below, for the Australian credit data, accuracy peaks (86.4%) at C =0.09.

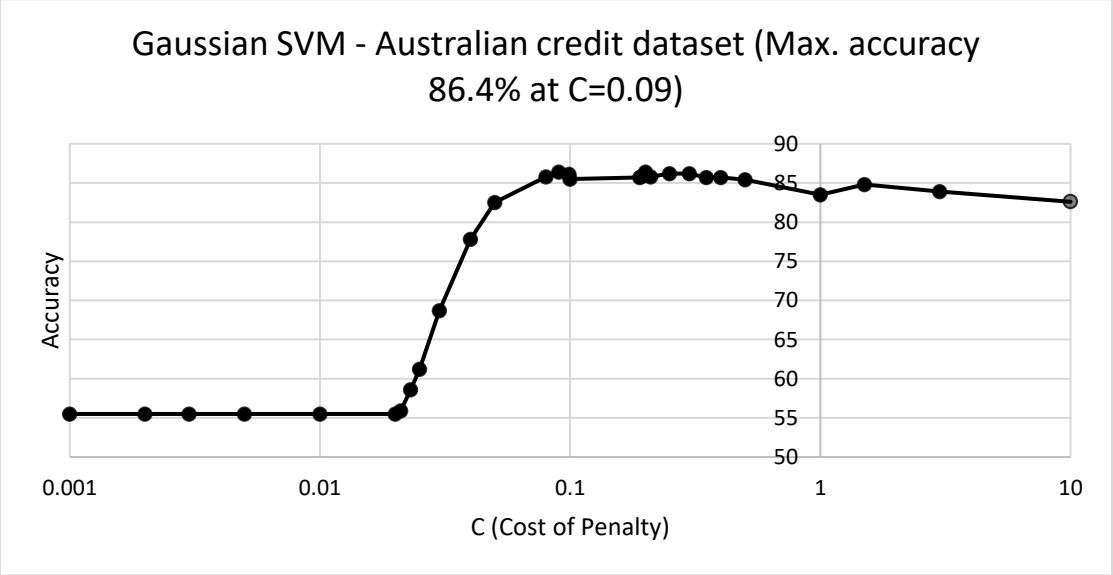

Figure 5-43: Effect of varying C value on accuracy of Gaussian SVM for Australian credit dataset

From figure below, for the Taiwan credit data, accuracy peaks (81.7%) at C=0.98 and then it remains the same.

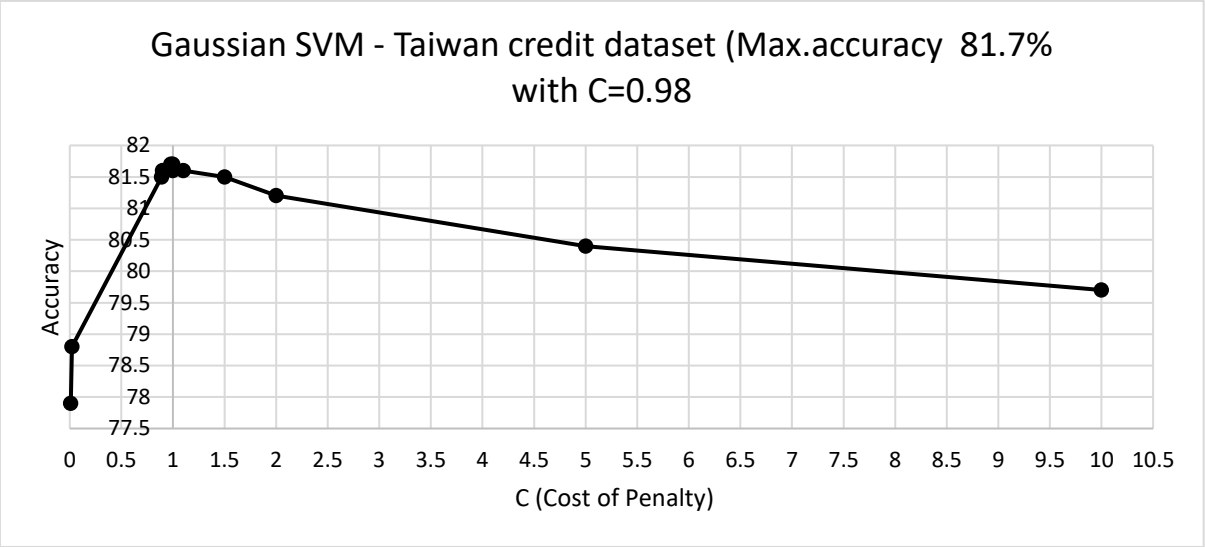

Figure 5-44: Effect of varying C value on accuracy of Gaussian SVM for Taiwan credit dataset

5.5.5 Results- Effect of varying Gamma (γ) on the accuracy of Gaussian SVM classifier

When γ is very small, the model is too constrained and cannot capture the complexity or decision boundaries. The region of influence of any selected support vector would include the whole training set. The resulting model will behave similarly to a linear model with a set of hyperplanes that separate the centres of high density of any pair of two classes. If γ is too large, the radius around a point only includes the support vector itself and no amount of regularisation with C is able to prevent overfitting [214].

For Gaussian SVM, following figures show the effect of varying values of parameter γ on accuracy of the classification for all the datasets.

From figure below, for the German credit dataset, maximum accuracy of 76.3% is obtained at γ=5 and then it plateaus until 6 before it starts reducing.

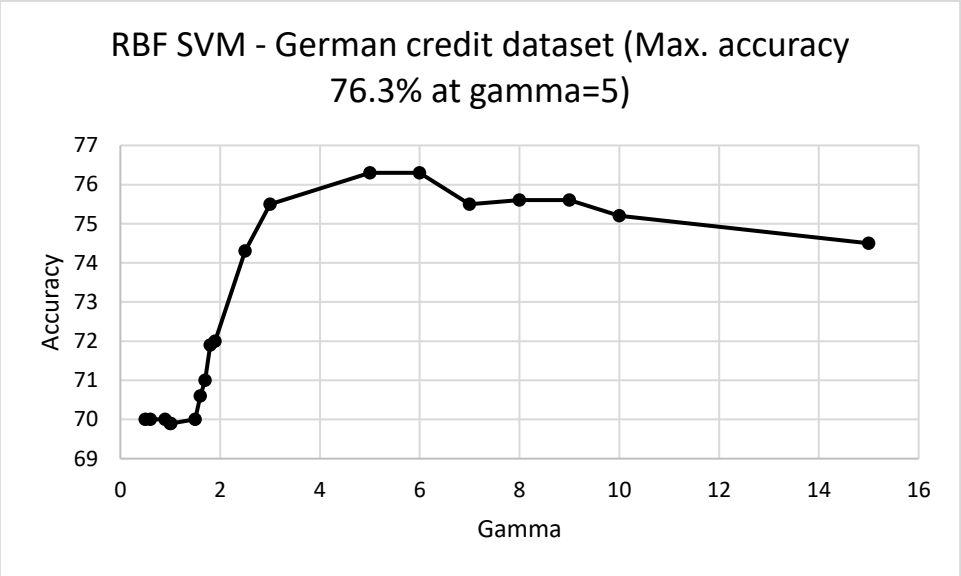

Figure 5-45: Effect of varying (γ) values on accuracy of Gaussian SVM for German credit dataset

From figure below, for the Australian credit dataset, maximum accuracy of 86.7% is obtained at γ=5 to 8.5.

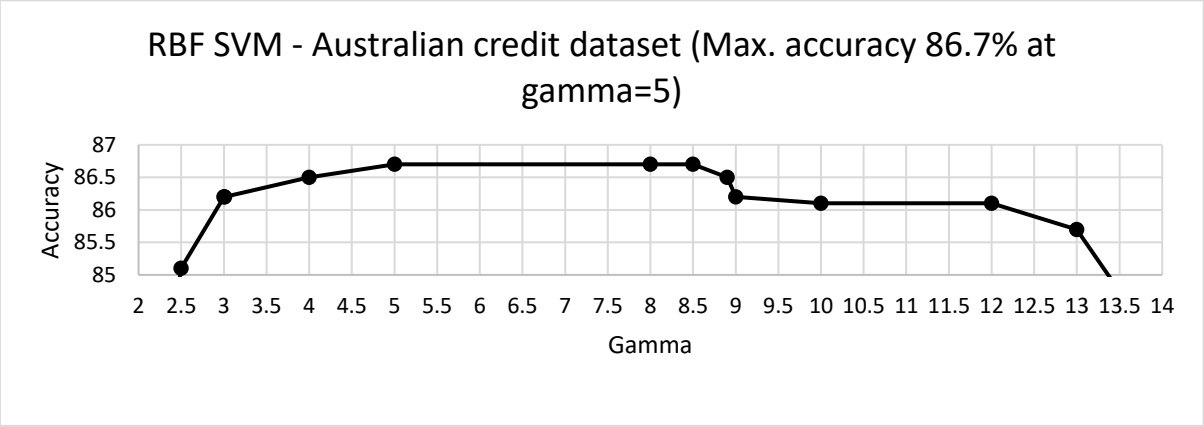

Figure 5-46: Effect of varying gamma (γ) values on accuracy of Gaussian SVM for Australian credit dataset

From figure below, for the Taiwan credit dataset, maximum accuracy of 82% is obtained at γ=5 and then it plateaus before reducing after γ=6.7.

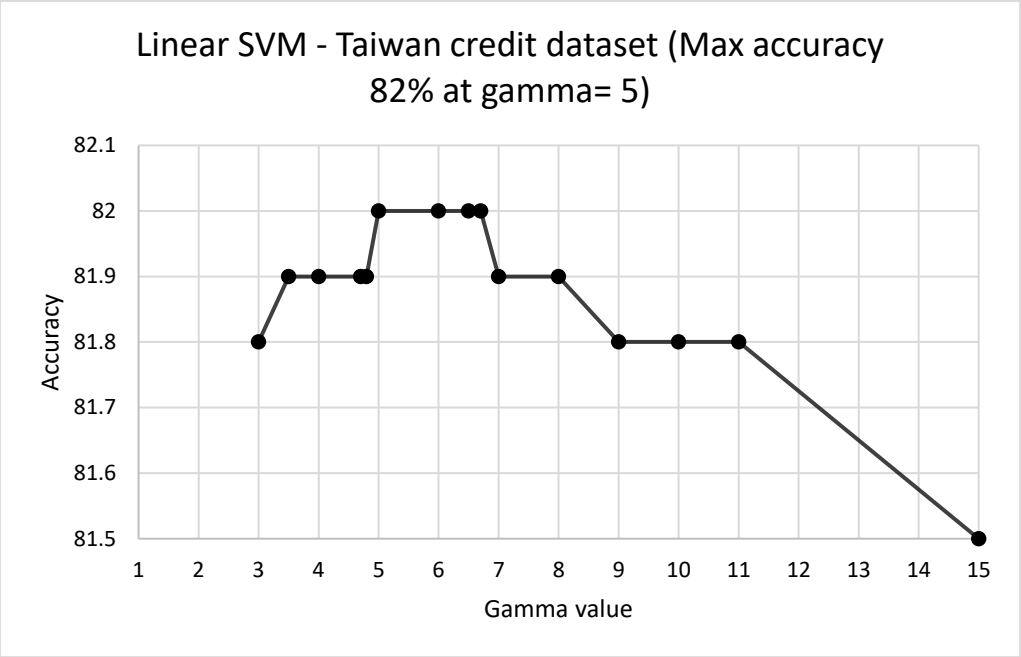

Figure 5-47: Effect of varying gamma (γ) values on accuracy of Gaussian SVM for Taiwan credit dataset

From all the figures above, it is seen that, as value of gamma increases, the accuracy increases but it flattens or even reduces after a certain point.

5.5.6 Results- Execution time Performance of the classifiers

We conducted experiments over the three datasets for credit classification to compare execution times for twelve classical machine learning techniques. The results are presented below.

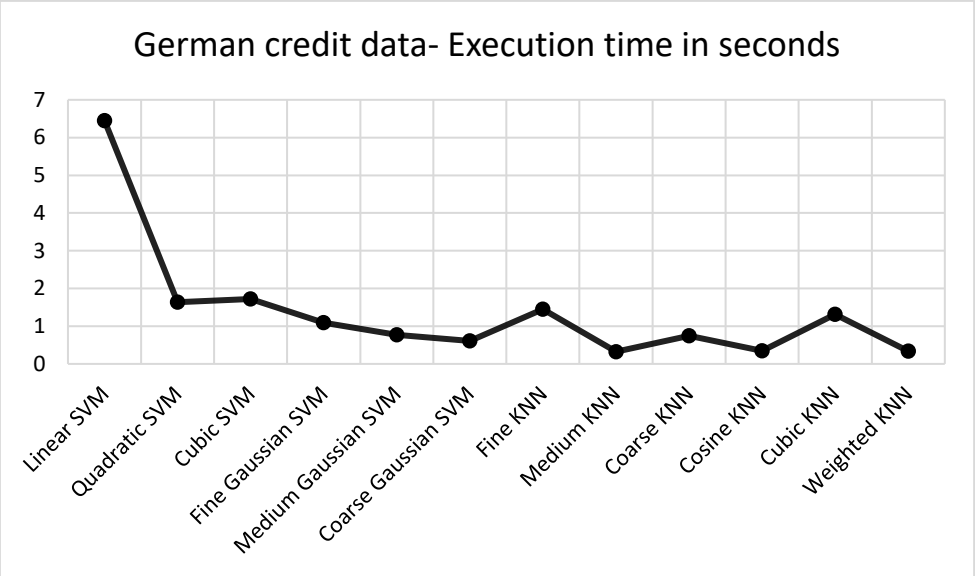

Figure 5-48: Execution time statistics for the classifiers for German credit dataset

For German credit dataset, Linear SVM takes longest with 6.4502 seconds and Medium KNN is fastest with 0.32478 seconds.

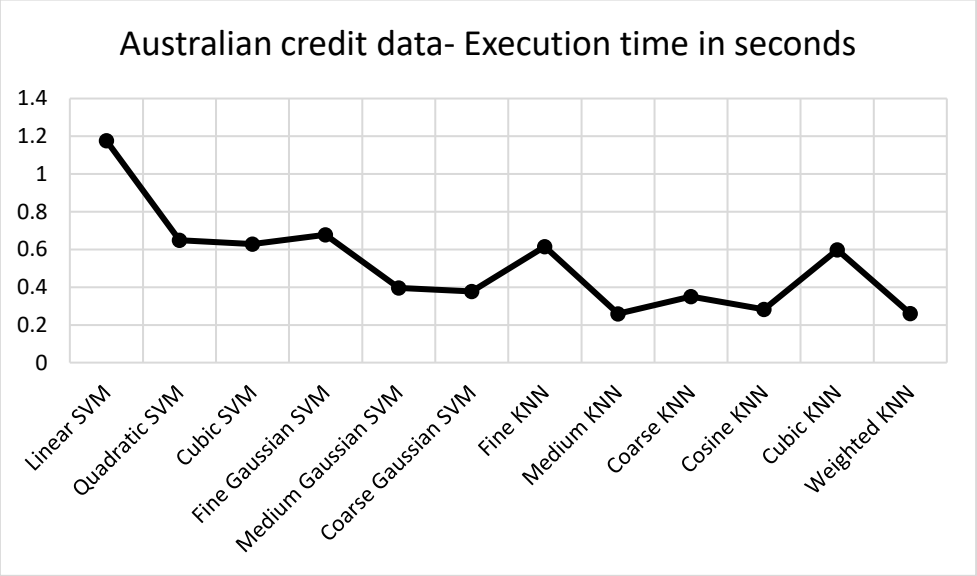

Figure 5-49: Execution time statistics for the classifiers for Australian credit dataset

For Australian credit dataset, Linear SVM takes longest with 1.1767 seconds and Medium KNN is fastest with 0.25937 seconds.

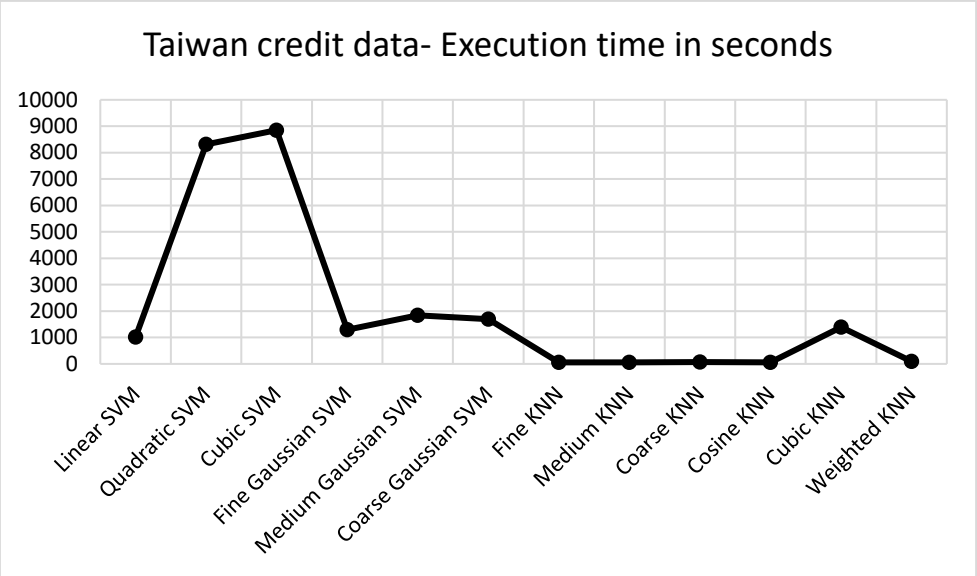

Figure 5-50: Execution time statistics for the classifiers for Taiwan credit dataset

As observed in chapter 4, the Taiwan credit dataset shows high linearity. In above figure, we observe that Linear SVM is one of the fastest algorithms on this dataset (as opposed to previous two datasets), which is consistent with the findings from chapter 4. Cubic SVM takes longest with 8850.7 seconds and Cosine KNN is fastest with 58.893 seconds.

5.6 Evaluation and Conclusion

A credit scoring model is a tool that is typically used in the decision-making process of accepting or rejecting a loan. Credit scoring problem is considered as an application of binary classification, i.e. a two–group classification problem distinguishing between good and bad applicants for credit based on information about the borrower from application forms and other relevant data. We model the credit scoring problem as the classification problem.

This chapter focused on:

- Determining accuracy of classification over twelve ML techniques for the three publicly available benchmark credit datasets to find the best technique;
- Application of a few chosen classical ML techniques from previous step to the problem of credit scoring to study the detailed performance;

- Investigation of the effect of varying the cost of penalty and gamma on accuracy of classification for Gaussian SVM classifier;
- Analysis of the execution times for the classifiers.

It was observed that medium Gaussian SVM technique followed by KNN yields best accuracy for all the three datasets.

The detailed comparison of the performance of these ML models is done and it is shown that these models could be useful tools for developing binary classifiers for the problem of credit scoring.

1. Three classical machine learning classification techniques were applied on the three datasets. The findings are:

Table 5-2: Results

Classifier	Performance measure	German credit data	Australian credit data	Taiwan credit data
Medium Gaussian SVM	Accuracy	***76.6%***	***86.1%***	***81.9%***
	AUC	79%	92%	72%
	Prediction accuracy	91%= good class, 43%= bad class	88%= good class, 84%= bad class	96%= good class, 34% = bad class
KNN	Accuracy	75.4%	***86.1%***	80.9%
	AUC	75%	92%	73%
	Prediction accuracy	93% = good class, 33% = bad class	79%= good class, 89% = bad class	95%= good class, 31% = bad class
NB	Accuracy	73.6%	80.29%	71.05%
	AUC	78.58%	90.33%	73.76%

In the experiments, the highest accuracy is produced by Medium Gaussian SVM for all the three datasets (and also KNN in case of Australian credit dataset) shown in bold italics in the table above. Large kernel tends to produce a smoother border between classes and a narrower kernel produces a more intricate border.

When kernel scale is set to high value as in coarse gaussian or to a low value as in fine gaussian, the results are moderate. Thus, for all the three datasets, the kernel scale parameter of medium value (which is equal to $\sqrt{Number\ of\ predictors}$) yields better results.

2. We investigated the effect of varying the cost of penalty on accuracy of classification for medium Gaussian SVM classifier.
    - For the German credit data, the best C is 1.01 with highest accuracy=75.9%;
    - For Australian credit data, the best C is 0.09 with highest accuracy=86.4%;
    - For Taiwan credit data, the best C is 0.98 with highest accuracy=81.7%.
3. We observed effect of varying the value of γ on accuracy of classification for medium Gaussian SVM classifier.

- For the German credit data, the best γ is in the range of 5 to 6 with highest accuracy=76.3%;
- For Australian credit data, the best γ is in the range of 5 to 8.5 with highest accuracy=86.7%;
- For Taiwan credit data, the best γ is in the range of 5 to 6.7 with highest accuracy=82%.

Selecting the best values for C and γ that could maximise the accuracy of the model is demonstrated in detail in chapter 5.

4. Next, the execution times for the classifiers were observed.

- Linear SVM takes longest for German and Australian credit datasets;
- Cubic SVM takes longest execution time for the largest Taiwan credit dataset. A kernel SVM requires on the order of n^2 computations for training and $n*d$ computations for classification, where n is number of training samples and d is the dimensionality of the dataset. A linear SVM requires on the order of $n*d$ computation for training (times the number of training iterations) and on the order of d computations for classification. Hence, for a large dataset such as the Taiwan credit dataset, the cubic kernel SVM is expensive for training and very expensive for classification in terms of time.

In real-time situations, often, accuracy is not enough to analyse the performance of a technique and deploy it. Sometimes a technique with lower accuracy may be selected based on the dataset characteristics, importance given to false positives and false negatives since these parameters incur costs in real applications. Acceptable tolerances in such scenarios decide the measure and technique to choose. Different measures allow for trade-offs that need to be chosen wisely.

6 FEATURE SELECTION WITH GENETIC ALGORITHM WRAPPER FOR CREDIT SCORING

6.1 Introduction

Quantitative credit scorecards and models are built for financial institutions as an efficient and effective means of assessing a loan applicant's credit risk. Credit approval decisions based on these quantitative credit scorecards have been shown to increase accuracy and consistency of credit scoring models. The effectiveness is due to the fact that such decisions are based on statistical models derived from data rather than the human judgement. Typically, credit scoring datasets are large and characterised by redundant and irrelevant features. To make the classification process computationally efficient, optimal features selection is essential. Feature selection creates more accurate and faster predictive models.

Feature subset selection removes redundant and irrelevant features from the dataset, thus improving the classification accuracy and reducing the computational cost [2,3]. The advantage of feature selection is that the information about the importance of a feature is not lost [217].

In this study, we apply Information Gain [218] and Genetic Algorithm based Wrapper technique to select features, and apply Support Vector Machines (SVM) [6,7], k-nearest neighbour (KNN) and Naïve Bayes as the classification algorithms in wrapper for credit rating.

The work is organised as follows: In Section 6.3, we provide a state of the art in classical wrapper algorithms such as Genetic Algorithms and Particle Swarm Optimisation, the machine learning models used as wrappers for feature selection for credit scoring and the challenges of feature selection along with the gaps identified. Section 6.4 discusses the techniques of Information Gain, KNN, Naïve Bayes, SVM and the performance measures employed in this study. A genetic algorithm with three different wrapper algorithms is developed in Section 6.5. Finally, Section 6.6 concludes the work with discussions about the findings.

6.2 Motivation

Financial credit scoring is one of the most crucial processes in the finance industry sector to be able to assess the credit-worthiness of enterprises and individuals. Various classification and prediction methods based on statistical and machine learning techniques have been employed for this task. “Curse of Dimensionality” is still challenge in machine learning techniques. Research has been carried out on Feature selection (FS) using genetic algorithm as wrapper to improve the performance of credit scoring models, however no overall best method exists which could be used in credit scoring problems. We aim to address the problem of credit scoring as a classification and feature subset selection problem. This work proposes a novel Information Gain Directed Feature selection algorithm (IGDFS), which performs the ranking of features based on information gain, propagates the top k features through the GA wrapper (GAW) algorithm using three classical machine learning algorithms of SVM, KNN and Naïve Bayes for credit scoring.

6.3 Existing Work

Feature selection techniques have emerged as crucial in the applications where the input space affects the classification algorithm’s performance. The process of feature selection searches through the space of all feature subsets while calculating evaluation measure to score the feature subsets. Since an exhaustive search is computationally too expensive, heuristic search techniques such as Genetic algorithm (GA) [220] and PSO [221] have been favoured for feature selection.

In the wrapper-based feature selection approach [222], a feature selection algorithm acts as a wrapper around a classification algorithm. The accuracy of this algorithm is used to find a subset of features. The wrapper approach especially is useful to solve the problems for which a fitness function cannot be easily expressed with an exact mathematical equation. The induction algorithm is used as a black box in feature selection process i.e. no knowledge of this

algorithm is needed [222]. The accuracy of this algorithm is used as evaluation measure to select the features.

A GA in wrapper (GAW) has been widely applied to feature selection in data mining [223]. A parallel genetic algorithm wrapper is proposed by Soufan et al. [224]. SVM classifier is the technique most popularly used in wrapper approach [13- 16]. When using SVM in a GA wrapper, both the feature subset and SVM parameters need to be optimised, simultaneously. In the literature, a few variants of GA+SVM algorithm have been proposed for different purposes. For example, a GA+SVM was used for the classification of hyper spectral images [227]. GA is used as a pre-processing step for SVM in a study by Verbiest et al. [228]. A Genetic Algorithm based Wrapper feature selection Hybrid Prediction Model (GWHPM) is proposed in a study by Anirudha et al. [229]. This study used k-means clustering technique to remove the outliers from the dataset, then an optimal set of features were obtained by using Genetic Algorithm based Wrapper feature selection; and then the selected feature set was used to build the classifier models of Decision Tree, Naive Bayes, k-nearest neighbour and Support Vector Machine.

Other machine learning models used in a wrapper approach include: C4.5 Decision trees [20, 21]; the model tree algorithm M5 [232]; Fuzzy Apriori Classifier [233]; Neural Network [234]; Bayesian Network classifier [235]. A hybrid genetic algorithm for feature selection wrapper based on mutual information is proposed by Huang et al. [236].

Particle Swarm Optimisation is another evolutionary computing method investigated for feature selection. Daamouche et al. [237] proposed usage of PSO for selecting the most informative features of classification. Lin et al. [238] simultaneously determined the parameters and a subset of features, without reducing SVM classification accuracy using PSO for parameter determination and feature selection of the SVM, and obtained similar result to GA + SVM.

The recent rapid growth in credit industry has made huge amounts of data available. Credit scoring datasets often are high dimensional which causes high

complexity, intensive computations and instability or lack of predictive accuracy for most classification models [232]. Feature selection is therefore necessary to reduce the burden of computing and to improv the prediction accuracy of the credit scoring models [239,240].

Somol et al. [241] studied filter and wrapper-based feature selection for credit scoring. Frohlich & Chapelle [242] proposed a GA-based feature selection approach using the theoretical bounds on the generalisation error for SVMs instead of performing cross-validation to estimate the generalisation error of a given feature subset, which is computationally attractive. Huang et al. [203] proposed three strategies to build the hybrid SVM-based credit scoring models to evaluate the applicant's credit score using the features of applicant. Not many studies were found at the time of this study dealing with parameters optimisation for the SVM classifier or focused on building a credit scoring model based on SVM model; and the authors proposed a hybrid GA-SVM strategy to perform feature selection task and model parameters optimisation. The SVM classifier achieved an identical classification accuracy with relatively few input features, compared with neural networks, genetic programming, and decision tree classifiers. Liang et al. [243] deployed three filters using GA and PSO, embedded with six different prediction models, such as linear SVM, RBF-SVM, KNN, Naive Bayes, Classification and regression tree (CART), and MLP in wrapper on bankruptcy and credit scoring classification problems. They concluded that there exists no best combination of the feature selection method over the four datasets used in the study. Waad et al. [244] also used Logistic Regression, Naïve Bayes, MLP, Random Forest trees in wrapper on three credit datasets and showed that feature subsets selected by fusion methods are either superior to or at least as adequate as those selected by individual methods.

Various supervised wrapper methods have been studied for feature selection due to the classification accuracy entailed by the underlying algorithm although it comes at a cost of flexibility and scalability.

Li [245] applied KNN whereas Chen & Li [39] proposed wrapper strategy of feature selection, including LDA, Rough set theory, Decision tree, F-score and

SVM classification model in credit scoring. Koutanaei et al. [37] successfully applied ensemble classifier for feature selection in credit scoring; Hamadani et al. [246] proposed hybrid models using Rough sets for feature selection during the data pre-processing phase and two hybrid sequences, Naïve Bayes networks and genetic algorithm to classify customers into good and bad credit risk groups. When executed on German, Australian and Japanese credit scoring datasets the results demonstrated that this approach gave superior performance in terms of classification accuracy and achieved higher overall classification rate as compared to several other previous studies which included ANN, SVM, DT. Wang et al. [247] combined the rough set and scatter search meta heuristic for feature selection for credit scoring. Hajek & Michalak [248] suggested wrapper approach by using multilayer perceptron, RBF neural network, SVM, Naïve Bayes, Random Forest, Linear Discriminant Classifier and Nearest Mean Classifier feature selection for credit rating prediction. Khanbabaei & Alborzi [231] used clustering as preprocessing step, genetic algorithm for feature selection and Decision tree for scoring of the credit customers. Oreski & Oreski [249] presented a hybrid GA with ANN to identify an optimum feature subset to increase the classification accuracy and scalability for credit risk assessment. Kozeny [250] developed a GA with weighted bitmask as alternatives of polynomial fitness functions to estimate parameter range for building credit scoring models. Sang et al. [251] integrated parallel Random Forest method and feature selection methods such as filter (t-test, LDA, LR), wrapper (GA, PSO) in credit scoring model.

The computing complexity of a machine learning algorithm is directly affected by problem space. Because of rapid advances in computing and information technologies, combining different types of techniques has become the norm in many of today's real applications. There is a growing tendency of using hybrid methods for complex problems.

It is often a requirement in certain applications to be able to interpret the predictive power of each feature in the dataset. In such cases, a feature selection method that returns a score such as Information Gain is more useful than methods that return only a ranking or a subset of features, where the importance of features is

not accounted for. The choice of feature selection method largely depends on the problem, the type of data (numerical or discrete, complexity, etc.) and future use of the model. Which methods are most useful for feature subsetting is an open debate.

To fill the gap identified above, we will address the issue of feature selection in the domain of credit scoring by proposing an Information gain directed feature selection method by incorporating the GA wrapper with machine learning techniques of SVM, KNN and Naïve Bayes.

6.4 Methodology

To classify the credit applicants, this work first ranks the features in order of importance to decision making/classification by measuring the information gain. The results are incorporated in the information directed wrapper feature selection method using genetic algorithm. Three classic machine learning models are used for the credit scoring embedded in the wrapper of GA, as a black box of fitness evaluation and these are SVM, KNN and NB.

The SVM hyperparameter selection is done by the method of grid search. The hyperparameter selection for k-nearest neighbour method (KNN) is done with cross-validation based on Euclidean distance calculations. KNN calculates a decision boundary (i.e. boundaries for more than 2 classes) and uses it to classify new points. The K in KNN is a hyperparameter that need to be selected to get the best possible fit for the dataset. K controls the shape of the decision boundary. The best K is the one corresponding to the lowest error rate in cross validation. If test set is being used for hyperparameter setting, it may lead to overfitting.

6.4.1 Information Gain of features

Financial institutions are primarily interested in determining which consumers are most likely to default on loans, i.e. credit-worthiness of applicants. However, they are also interested in knowing which characteristics of a consumer are most likely to affect their likelihood to default [201]. For example, are young credit applicant less likely to default than older ones? This information allows credit modellers to determine the key predictors and customise the model to include this discovery.

To build a good model, the analyst will iteratively continue searching for such predictors that also exhibit useful patterns with respect to the target, and contribute favourably to the total power of the scorecard [252].

There are many ways of scoring the features such as information entropy, correlation, Chi squared test, Gini Index. Any of these evaluation classes gives a score for each feature so that they can be ranked. Entropy is one way to measure diversity. Impurity of information can be measured by information entropy which quantifies the uncertainty associated with predicting the value of a random variable.

Let y be a discrete random variable with two possible outcomes. The binary entropy function E, expressed in logarithmic base 2, i.e. Shannon unit:

$$E(y) = -p(+)log_2 p(+) - p(-)log_2(p(-)) \quad \textbf{(6-1)}$$

where, (+,–) are the classes, $p(+)$ is the probability of some samples $y \in (+)$, and $p(-)$ is the probability of $y \in (-)$. We first use Entropy as a measure to quantify the uncertainty in each feature involved in decision making. The conditional entropy of two events X and Y, when X has value x is:

$$E(Y|X) = \sum_{x \in X} p(x)E(Y|X = x) \quad \textbf{(6-2)}$$

$$= -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x)log_2 p(y|x)$$

$$= -\sum_{x \in X} \sum_{y \in Y} p(x,y)log_2 p(y|x)$$

Note: $\lim_{x \to 0} x \ log_2(x) = 0$

The smaller the degree of impurity, the more skewed the class distribution. Entropy and misclassification error are highest when class distribution is uniform. The minimum value of entropy is attained when all the samples belong to the same class.

Information Gain (IG) is widely used on high dimensional data to measure the effectiveness of features in classifying the training data. It is the expected amount of information, i.e. if the training data is split on the feature values, information gain is the measurement of the expected reduction in entropy after the split. The more an attribute can reduce entropy in the training data, the better the attribute in classifying the data.

Namely, the information gain (IG) is:

$$IG(x) = E(y) - E(y|x) \quad \textbf{(6-3)}$$

Higher information gain means better discriminative power for classification. Information gain (IG) is a good measure to determine the relevance of feature for classification.

The importance of features towards decision making in the model is done by evaluating them with the information gain measurement. Not all the attributes contribute equally to the decision making. Hence the attributes can be sorted in the order of their contribution in decision making by listing the features in decreasing order of information gain scores.

6.4.2 K-Nearest Neighbour (KNN) Algorithm

KNN algorithm is one of the simplest classification algorithms with highly competitive results, easy to interpret output, good calculation time and predictive power. It is one of the most effective nonparametric methods and easy to implement since only parameter K (the number of nearest neighbours) needs to be tuned. The number K of nearest neighbours is key to the performance of the classification process. KNN takes as input closest training samples and classifies a new object from the testing samples based on the minimum Euclidean distance without building a model.

$$d(X, Z) = \sqrt{\sum_{i=1}^{n} (Z_i - X_i)^2} \quad \textbf{(6-4)}$$

where, X and Z are *n*-dimensional vectors in the feature space.

If an object is close to the *k* nearest neighbours, then it is assigned class membership of most common *k* neighbours. The main task of KNN is to search the nearest neighbours for each sample. The parameter *k* needs tuning for each dataset for enhancing the classification accuracies. To choose the parameter *k* we use 10-Fold-cross validation to validate KNN for various quantities of neighbours near rule-of-thumb-values. Cross validation leads to the highest classification generalisability. If employing KNN with different values of *k* on a dataset, we obtain different accuracy at each round. The optimum *k* achieving the best accuracy, is used in the feature selection.

6.4.3 Naïve Bayes

The Naïve Bayes(NB) classifier uses Bayes' Theorem, which calculates a probability of class label C_j by counting the frequency of 'attribute value - class' combinations in the historical data.

As stated by Twala [9], "This classifier (NB) applies the Bayes rule to calculate the probability of class label C_i given all attributes A_j and predicts the class with the highest posterior probability". The probability of a class value C_i given an instance X for n observations is given by following equation.

$$p(C_i|X) = \prod_{j=1}^{n} p(A_j|C_i).p(C_i) \quad \textbf{(6-5)}$$

Let *D* be a training set of samples and corresponding class labels. Each sample is represented by an $n-D$ attribute vector X. X includes *n* independent attributes $(x_1, x_2, ..., x_n)$. If there are m class labels such as $C_1, C_2, ..., C_m$, then classification is to derive the maximal posteriori, $P(C_i|X)$:

$$P(C_i|X) = \frac{P(X|C_i).\ P(C_i)}{P(X)} \quad \textbf{(6-6)}$$

$P(X)$ is constant for all classes; hence $P(C_i|X)$ can be represented with Eq. (7) which needs to be maximised.

$$P(C_i|X) = P(X|C_i).P(C_i) \quad \textbf{(6-7)}$$

Naïve Bayes algorithm assumes that the attributes are conditionally independent, i.e. $P(X)$ is constant. Hence, the class assignments of the test samples are given by following two equations:

$$p(X|C_i) = \prod_{k=1}^{n} p(X_k|C_i) \quad \textbf{(6-8)}$$

$$C = \ argmax\ \{p(X \mid C_i\ ).p(C_i\ )\} \quad \textbf{(6-9)}$$

If for a new sample, the posterior probability $P(C_2/X)$ is the highest for all the k classes, then this sample belongs to class C_2 according to the NB classifier.

6.4.4 SVM classifier

SVM, a popular binary classifier is used in the wrapper algorithm as a fitness evaluator since it is able to deal with wider solution space in the early stages of feature space search [253]. The optimal hyperplane with the smallest number of support vectors has a wider adaptability and the highest classification performance [254]. SVMs do not suffer from local minima, offer good generalisation performance to new objects, and a representation that depends on few parameters [255]. This method, however, does not directly determine the importance of the features used [256].

6.4.4.1 RBF-kernel SVM for feature selection

The problem of classification in Credit Scoring is realised as mapping of input features set into the decision variable (taking value as creditworthy or not creditworthy), represented as $y=f(F)$, where y is the decision variable and F is the feature vector. Identifying creditworthy applicants from not creditworthy ones is not a linearly separable problem. Non-linear machines which map the data to higher dimensions can be used to find a SVM hyperplane minimising the number of errors for the training set.

RBF-kernel SVM, equivalent to a specific three-layer feed-forward neural network, is powerful for non-linear binary classification problems, and repetitions of the assessment with the RBF-kernel SVM are easier in MATLAB [210]. The RBF-kernel SVM maps the problem space to higher dimension, making the data linearly separable and using the linear SVM to solve the problem in the new higher dimension space. A RBF-kernel SVM performs well in high dimensional spaces, even if the number of dimensions is greater than the number of samples [210]. Assume $\Phi(F)$ is a feature map (which can be very high dimensional) where F is mapped to the kernel function $\left(F_j, F_i\right) = \Phi(F_j)^T \Phi(F_i)$. The kernel SVM is expressed as:

$$f(F) = \left(\sum_{i=1}^{N} \alpha_i y_i \, k\left(F_j, F_i\right) + b\right) \quad \textbf{(6-10)}$$

where α_is are dual variables (the Lagrange multiplier for each training example i) and $k\left(F_j, F_i\right)$ is the kernel function defined as the inner product between two feature vectors, performing the nonlinear mapping into feature space.

Correspondingly, learning to maximise:

$$\sum \alpha_i - \frac{1}{2}\left(\sum_{jk} \alpha_j \, \alpha_k \, y_j \, y_k \, k\left(F_j, F_i\right)\right) \quad \textbf{(6-11)}$$

this is the SVM optimisation problem, where $y_i \in \{+1, -1\} for\ all\ i = 1\ to\ n$

Subject to constraints $C \geq \alpha_i \geq 0,\ for\ all\ i = 1\ to\ n\ \forall_i$ and $\sum \alpha_i \, y_i = 0$

where *C*, which is now the upper bound on α_i, is a penalty parameter and is determined by the user. SVMs are able to deal with high dimensional data through the use of this regularisation parameter C.

The kernel function k can be used to implement non-linear models of the data.

The (Gaussian) Radial-based function (RBF) kernel (equation (6-12) is commonly used as the kernel of a SVM.

$$k(F, \tilde{F}) = exp\left(\frac{-\|F - \tilde{F}\|^2}{2\sigma^2}\right) \quad \textbf{(6-12)}$$

The RBF-kernel SVM:

$$f(F) = \sum_{i=1}^{N} \alpha_i y_i \, exp\left(\frac{-\|F - \tilde{F}\|^2}{2\sigma^2}\right) + b \quad \textbf{(6-13)}$$

The radial basis function kernel has an additional kernel parameter γ i.e. kernel bandwidth to be optimised, where $\gamma = \frac{1}{2\sigma^2}$. As γ increases the fit becomes more and more non-linear.

6.4.5 Performance Assessment Methods

We use the most commonly used measure of classifier performance: accuracy, i.e. the percent of correct classifications predicted (See section 5.4.6).

6.4.6 Validation

k-fold cross-validation technique is used to validate the models for assessing how the results generalise to an independent new dataset and to estimate prediction error. This study used k = 10 (See section 5.4.7).

6.5 Feature selection with Genetic Algorithm Wrapper

Feature selection extracts a subset of the original attributes reducing the attribute space of a feature set. Among the various ways of searching, exhaustive search is too expensive and a non-exhaustive, heuristic search techniques are often used [257].

6.5.1 Wrapper Approach

Figure 6-1 [222] illustrates the wrapper approach, where, the feature subset is selected by using a classification algorithm, i.e., selection is performed by using the algorithm as a black box without the requirement of knowledge of the algorithm, and the results produced by the classifier will be evaluated with classification accuracy or other performance measures.

In the feature selection process, the classifier is run on the selected features of dataset, partitioned into training sets and validation sets. The feature subset with highest accuracy is chosen as the final subset.

For each feature subset taken into consideration, the wrapper method trains the classifier and evaluates the feature subset by estimating the generalisation performance i.e. the accuracy of the machine trained with this feature subset on the original data. The search space is full feature space with *n* dimensions, where *n* is the number of full features. Hence, a n bit string can be used to represent the selected status of n features. Namely, each bit indicates whether a feature is selected (1) or unselected (0).

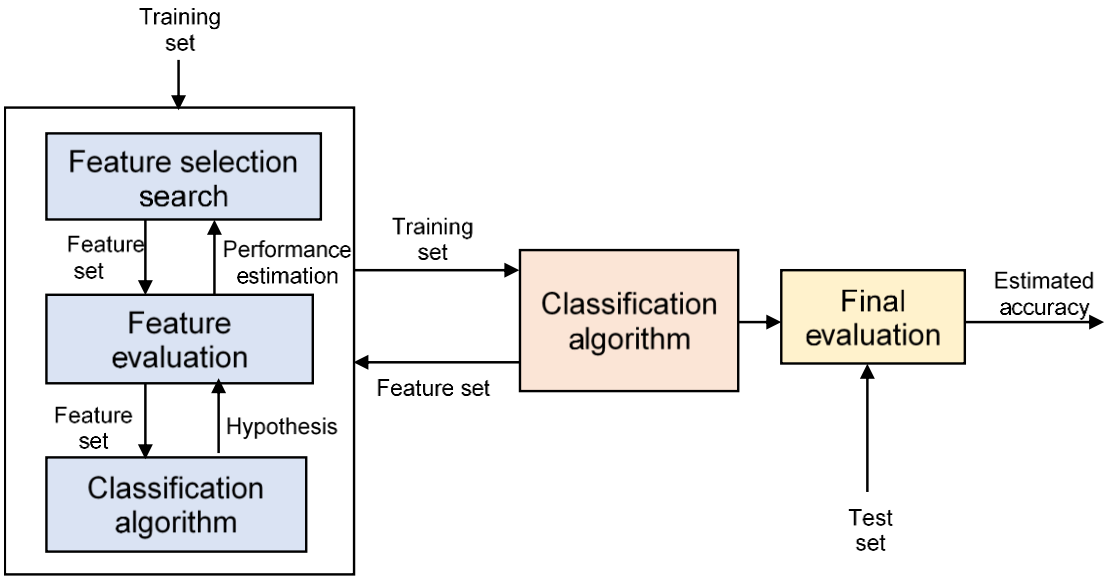

Figure 6-1: The Framework of Wrapper Approach for Feature Selection.

6.5.2 The Improved Genetic Algorithm Wrapper

Genetic algorithm is an adaptive heuristic search algorithm that mimics the evolution process of genetics. A population of competing solutions is maintained and evolved over time by selection, crossover, and mutation to converge to the best solution. Parallel search is performed on the solution space to find an optimal solution without getting stuck in a local optimum. In addition, the robustness to its search space size and the underlying multivariate distribution assumptions have made it a promising method for feature selection over a high-dimension space

[253]. GA scale admirably well to any dimension and any data with minimal knowledge of the data itself.

To apply this algorithm to solve the credit scoring problem, two essential issues need to be solved: fitness function and classifier choice. The classifier should be able to handle very high dimension feature space given a limited sample space. Support vector machines (SVM) are an exception [258]. SVMs are capable of treating a specific data, avoid overfitting and dimensionality curse and offer nonlinear modelling. In this study, we first apply the information gain to rank the features of dataset, then propagate the top n features through the wrapper-GA process of feature selection using SVM, KNN and NB as underlying classifiers.

Generally, the requirements for a search for an optimal solution in the whole feature space are: a state space, an initial state, a termination condition, and a search engine [222]. The size of search space is 2^n-1, where n is the total number of features. As each feature has two possible states: "1" or "0", an n bit string will have 2^n possible combinations. Assume τ features, which are not important to decision making in terms of the values of their information gains, be removed. The length of a binary string becomes n-τ. Even in the reduced search space $(2^{n-\tau})$, a brute-force search for a large space of $2^{n-\tau}$ is still infeasible. Of course, such space reduction is worthy for GA wrapper search.

The ingredients of a Genetic Algorithm are:

(1) Chromosome: GA maintains a diverse population $x_{1...n} = \langle x_1,..., x_n \rangle$ of n individuals x_i, the candidate solutions. The fitness of these individuals is evaluated by calculating an objective function $F(x_i)$ that is to be optimised for a given problem. These individual solutions are represented as 'chromosomes,' which cover the entire range of possible solutions (the search space). In this work, the population is maintained to propagate the top-ranking feature in the next generation.

In this study, binary coding system is used to represent the chromosomes. The bit strings representing the genotype (abstract representation) should be transformed into phenotype (physical make-up) by converting binary string into feature indices representation. The number *n* of bits represents the number of features. If the *i-th* bit is 1, then the feature x_i is selected and if it is 0, feature x_i is not selected.

(2) Selection operator: Selection is the process of evaluating the fitness of the individuals and selecting them for reproduction. There are several ways to perform selection. Some commonly implemented methods are Roulette-Wheel Selection, Tournament Selection, Elitist Selection, Rank Selection, Hierarchical Selection. This work has used Tournament selection to select sufficiently good individuals for mating.

(3) Crossover operator: Crossover operator creates two offspring from the two selected parent chromosomes by exchanging part of their genomes. Crossover is the process of extracting the best genes from parents and reassemble them into potentially superior offspring. The simplest form of crossover is known as one-point crossover. Other types are Two-Point Crossover, Uniform crossover. This work has used single point crossover. In this work, the crossover function is customised to propagate the top ranked feature.

(4) Mutation operator: Mutation maintains genetic diversity of population from one generation of chromosomes to the next and increases the prospect of the algorithm to generate more fit individuals. Using a small mutation probability, at each position in the string, a character at this position is changed randomly. Mutation of bit strings flips the bits at random positions with a small probability. This work has used uniform mutation. In the proposed algorithm, when the parents are selected to mutate, the topmost feature is propagated through.

(5) Elitism: Elitism guarantees that the best fit members are passed on to the next generation. The best individual or a set percentage of fittest members survives to the next generation. Small elitism compared to the population size, yields a good balance between diversity and non-overfitting situation. High elite count causes

the fittest individuals to dominate the population making the search less effective. This work guarantees that 2 elite offspring survive to the next generation.

(6) Diversity: An important factor that influences the performance of the genetic search is the diversity of the population. Diversity ensure that the solution space is adequately explored, especially in the earlier stages of the optimisation process. Very little diversity results into the GA converging prematurely. The initial range of the population and the amount of mutation affect the diversity of the population. Here tournament selection and uniform mutation are used in the evolutionary process of GA.

(7) Termination criteria: Three possible termination criteria could be used for the GA: A satisfying solution has been obtained, a predefined maximum number of generations has been reached, the population has converged to a certain level of genetic variation [259]. The convergence of the algorithm depends on the mutation probability: a very high mutation rate prevents the search from converging, whereas a very low rate results in premature convergence of the search. The termination criteria for this work is maximum number of generations = 40 to 60.

(8) Blackbox with fitness function: A fitness function evaluates how good each individual in the population is. The fitness $f(x_i)$ of each individual x_i is evaluated in each generation against the optimisation criterion. To create the next generation, the fittest individuals obtained are allowed to reproduce using the set crossover and mutation rate. In this study, SVM, KNN and NB are used as the induction algorithms in the black box of evaluation.

Hence, the fitness of each individual is:

$$f(x_i) = f(g(x_i)) \quad \textbf{(6-14)}$$

where $g(x_i)$ is the accuracy of classifier SVM, KNN, NB.

The three GA wrapper techniques with the SVM, KNN and NB are denoted as GA-SVM, GA-KNN, and GA-NB, respectively.

Algorithm 1 in Figure 6-2 provides the operational steps of the proposed method of Information Gain Directed Feature Selection, where Algorithm 2 are SVM, KNN and NB classifiers.

Algorithm 1 Information Gain Directed Feature Selection

1: Measure Information Gain of individual features from the dataset
2: Rank the features in the dataset according to their importance: $F = (f1 > f2 > f3, ...)$
3: **Input:** Top N feature set Fr and class label C
4: **Output:** S
5: $S \leftarrow null$
6: **procedure** GA
7: **Input:** *PopSize Ps, GenSize, GenomeLength N, ProbMutation Pm*
8: **Output:** *The Best individual in all generations*
9: **Initialize:** *Population: Ps*N*
10: *Retain f1 from Fr*
11: *Ps ← random binary chromosomes*
12: **for** each chromosome **do**
13: Compute fitness according to Algorithm2;
14: **end for**
15: **repeat**
16: Select parents *p1, p2* from population based on the fitness;
17: **for** all new children **do**
18: retain *f1* from *Fr;*
19: Crossover *p1, p2;*
20: Mutate each gene in new child chromosome with probability *Pm;*
21: **end for**
22: Evaluate fitness of new individuals according to Algorithm2
23: Replace least-fit population with new best individuals
24: **until** $StoppingCriteria$
25: **end procedure**

Figure 6-2: The IGDFS Algorithm

6.5.3 Description of the Experiment

6.5.3.1 Experimental setup

In this work, three publicly available credit datasets are used to test the performance of the proposed information gain directed feature selection (IGDFS) algorithm. These are the benchmark datasets frequently employed in the literature to compare performance of different classification methods. Table 6-1 describes these datasets. To ensure validity of the model to make predictions on new data, *k*-fold cross validation method is implemented.

Our implementation of algorithms was carried out on Intel Pentium IV CPU running at 1.6 GHz and 256 MB RAM, in MATLAB 2016 mathematical development environment and the LIBSVM toolbox developed by Chang & Lin [260].

For the proposed Information Gain Directed GA-based feature selection approach, the parameters for the SVM classifier were obtained using the Grid Search algorithm. The grid search algorithm is a widely-used method in the literature to find the best hyper parameters (i.e. model selection). The goal is to identify good penalty parameter C and the kernel parameter γ so that the classifier can accurately predict unknown data [261].

6.5.4 The Datasets

Table 6-1: Characteristics of all the datasets

Dataset	**N**	**n**	**N_n**	**N_p**
German Credit	1000	20	700	300
Australian Credit	690	14	307	383
Taiwan Credit	30000	24	23364	6636

In the table above,

N = number of total samples present in the dataset,

n = number of features in the dataset,

N_n = number of good credit samples,

N_p = number of bad credit samples.

The details of these datasets and the analysis is described in chapter 4.

6.5.5 Attribute normalisation

Attribute normalisation avoids the dominance of attributes in greater numeric ranges over those in smaller numeric ranges. Kernel values are calculated by inner products of feature vectors where greater-numeric-range attributes might cause numerical problems and normalisation avoids these numerical difficulties

[261]. We performed linear normalisation on each attribute to the range [−1, +1] using following formula:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \quad \textbf{(6-15)}$$

where x' is the scaled value of feature x, x is the original value of feature x, $min(x)$ is the minimal value of feature x and $max(x)$ is the maximal value of feature x.

6.5.6 Data preprocessing for LIBSVM

The data needs to be preprocessed since LIBSVM requires data be in sparse format, i.e. only the non-zero data are stored. Hence, the index specifies the column of the feature/attribute. We need to know how many classification classes will be used (for binary classification, it is 2) and the feature space. Classification class is true/false or 0,1. Feature space is a space for multidimensional data. Each feature must have its own ID (index) and its value.

Thus, the format of training and testing data is:

<label> <index1>:<value1> <index2>:<value2> … <indexN>:<valueN>

Each row contains an instance in the dataset and is ended by a '\n' character. For classification, *<label>* is an integer indicating the class label and is the target value of the training data. The pair *<index>:<value>* gives a feature value: *<index>* is an integer starting from 1 and *<value>* is a real number. Indices are in ascending order.

E.g. 1:53.7 means that feature 1 has value 53.7.

Index serves as a way to distinguish between the features.

E.g. 1 1:0.7 2:0.31 3:0.17 translates to:

Assign to class 1, the point (0.7,0.31,0.17)

6.5.7 SVM parameters selection

C is the cost of classification and γ is the kernel parameter for a nonlinear support vector machine (SVM) with a Gaussian radial basis function kernel.

The general procedure in developing an SVM is to optimise both C and γ for a dataset. The problem of optimising these parameter values is called model selection, and the selection results strongly influence the performance of the classifier. Accuracy is used to evaluate the performance of a model on the datasets. To achieve good performance, some preliminary experiments were conducted to determine the optimal model parameters using exhaustive grid search approach [261] in finding the best C and γ for each dataset.

Both C and γ are scale parameters, so the grid is on a logarithmic scale. Doubling/halving C and γ on adjacent grid points is a tried and tested process since a complete grid-search may be time-consuming. If too fine a grid is used, we may end up over-fitting the model selection criterion, so a fairly coarse grid turns out to be good for generalisation as well as computational expense. We employed exponentially growing sequences of C and γ to identify best parameters [261]. A coarse grid is used first to identify promising region on the grid and then a finer grid search is conducted on that region to obtain a better cross-validation rate.

The steps for grid search are as follows:

Step 1: Set up a grid in decision space of (C, γ) with $log_2 C \in \{-5,...15\}$ and $log_2 \gamma \in \{-15, ...,3\}$.

Step 2: Train SVM on each pair (C, γ) in the search space, with 10-fold cross validation on the training set.

Step 3: Experiment with various pairs of (C, γ) values and choose the parameter (C, γ) that leads to the highest accuracy in cross validation.

Step 4: Use these best parameters to create a predictive model.

6.5.8 KNN parameter selection

The optimal parameter K (number of neighbours) for KNN is the one that corresponds to the lowest test error rate. We want to choose the tuning parameters that best generalise the data and which leads to the highest classification generalisability. In a better approach, the test error rate is estimated by holding out a subset of the training set from the fitting process [262], [263]. We used k-fold cross validation as performance testing algorithm along with KNN. Various quantities of K were used as near rule-of-thumb-values. On each dataset, we employed KNN with different values for K and obtained different accuracy for each K. The K, which leads to achieving the best accuracy, is the optimum K.

6.5.9 Genetic Algorithm parameters

The general approach in determining the appropriate parameter set of genetic algorithm for a given dataset is to conduct a number of trials of different combinations and choose the best combination that produces good results for the particular problem [264]. In this study, the parameters of GA are selected based on some related works, such as [243], [265]. We tried different values of the population size (20–100), mutation rate (0.001–0.3), and number of generations (20-100) to compare and obtain the best parameter combination.

The settings used for the GA system are summarised in Table 6-2.

Table 6-2: The main GA parameters

Parameter	**Value**
Objective function	Fitness value = Average accuracy
Population Size	50-70
Number of generations	20-50
Parent Selection	Tournament selection
Tournament Size	2
Crossover Type	Single point
Mutation Rate	0.1
Mutation Type	Uniform mutation

Stop Condition	Maximum number of generations

6.5.10 Experimental Results and Discussion

6.5.10.1 Information Gain based Ranking

The three tables below show the information gain ranking for the features of all three datasets. The ranking directly reflects the contribution of the features towards classification. Considering these rankings, we devised the information gain directed feature selection (IGDFS) algorithm. From table below, the frequency of the feature 'Credit amount' is the most informative among all features and 'Number of people being liable to provide maintenance for' is the least informative in case of the German credit dataset.

Table 6-3: Information Gain (IG) order of features for the German Credit Dataset

IG Rank No.	Feature name	IG Rank No.	Feature name
1	Credit amount	11	Other instalment plans
2	Status of existing checking account	12	Personal status and sex
3	Duration in months	13	Foreign worker
4	Age in years	14	Other debtors / guarantors
5	Credit history	15	Instalment rate in percentage of disposable income
6	Savings account/bonds	16	Number of existing credits at this bank
7	Purpose	17	Job
8	Property	18	Telephone
9	Present employment since	19	Present residence since
10	Housing	20	Number of people being liable to provide maintenance for

Table 6-4: Information Gain (IG) order of features for the Australian Credit Dataset

IG Rank No.	Feature name	IG Rank No.	Feature name
1	X_2	8	X_9

2	X_{14}	9	X_5
3	X_8	10	X_6
4	X_3	11	X_4
5	X_{13}	12	X_{12}
6	X_7	13	X_{11}
7	X_{10}	14	X_1

Table 6-4 shows the ranking of features for Australian Credit dataset. This dataset does not name the features but identifies them with the labels *X1, X2,....X14*. As per the information gain ranking, feature *X2* is the most informative and *X1* is the least informative.

Table 6-5: Information Gain (IG) order of features for the Taiwan Credit Dataset

IG Rank No.	Feature name	IG Rank No.	Feature name
1	BILL_AMT_1	13	PAY_0
2	BILL_AMT_2	14	PAY_2
3	BILL_AMT_3	15	PAY_3
4	BILL_AMT_4	16	PAY_4
5	BILL_AMT_5	17	PAY_5
6	BILL_AMT_6	18	PAY_6
7	PAY_AMT_1	19	SEX
8	PAY_AMT_2	20	EDUCATION
9	PAY_AMT_3	21	MARRIAGE
10	PAY_AMT_6	22	LIMIT_BAL
11	PAY_AMT_4	23	AGE
12	PAY_AMT_5		

Table 6-5 shows the ranking of features for Taiwan Credit dataset. As per the information gain ranking, the feature BILL_AMT_1(Amount of bill statement in September 2005 (NT dollar)) is the most informative and AGE is the least informative.

6.5.10.2 Parameter selection for SVM by Grid-Search method

A grid search was employed to search the SVM parameter space using a logarithmic scale. We first performed a coarse search with $C = \{2^{-5}, 2^{-5}+\Delta C, 2^{-5}+2\Delta C, ..., 2^{15}\}$ and $\gamma = \{2^{-15}, 2^{-15}+\Delta\gamma, 2^{-15}+2\Delta\gamma, ..., 2^{3}\}$, where $\Delta_{C\,\text{coarse}} = \Delta_{\gamma\,\text{coarse}} = 2$. The promising region obtained on the grid is then refined with a finer search (step size $\Delta_{C\,\text{fine}} = \Delta_{\gamma\,\text{fine}}$ = 0.0625. The prediction accuracy (10-fold) had a maximum peak at (C, γ) = (2.1810, 0.0423) for German credit dataset. Thus, the optimal values of C and γ for this dataset are 2.1810 and 0.0423, respectively Figure 6-3.

Figure 6-3 to Figure 6-5 below show the contour plot of grid search results for optimum values of SVM parameters C and γ for all the three datasets. The two parameters are shown in logarithmic axes x and y in the graphs, the lines indicating the area where the deeper grid search was performed. The colours of the lines indicate the graphical bounds of the searched space in the graph. The parameter values obtained are used for training RBF-SVM.

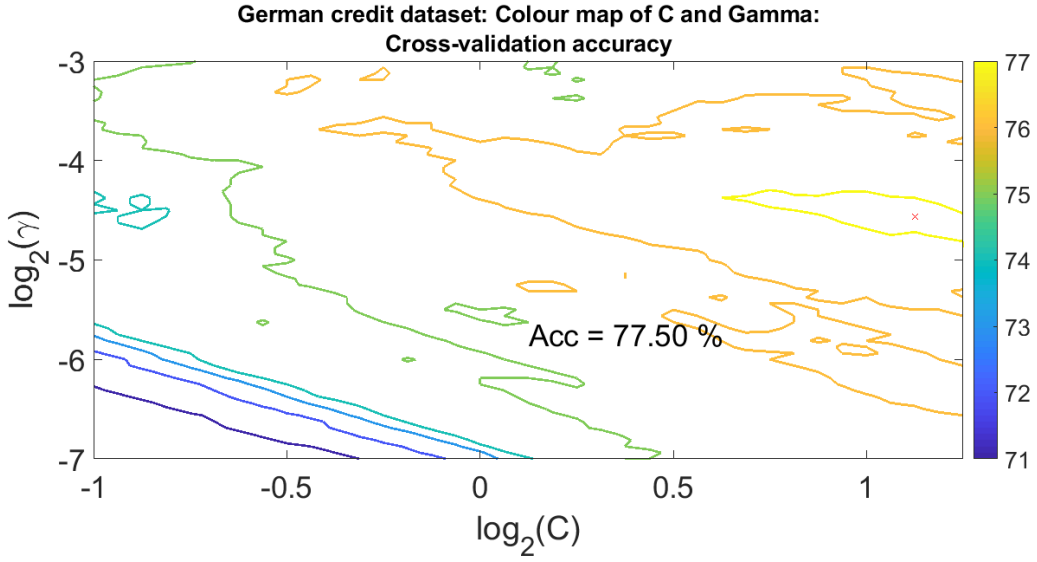

Figure 6-3: Result of grid search for optimised parameter values for German credit dataset. The model peaks at Accuracy=77.50%; *(C=2.1810, γ=0.0423)*

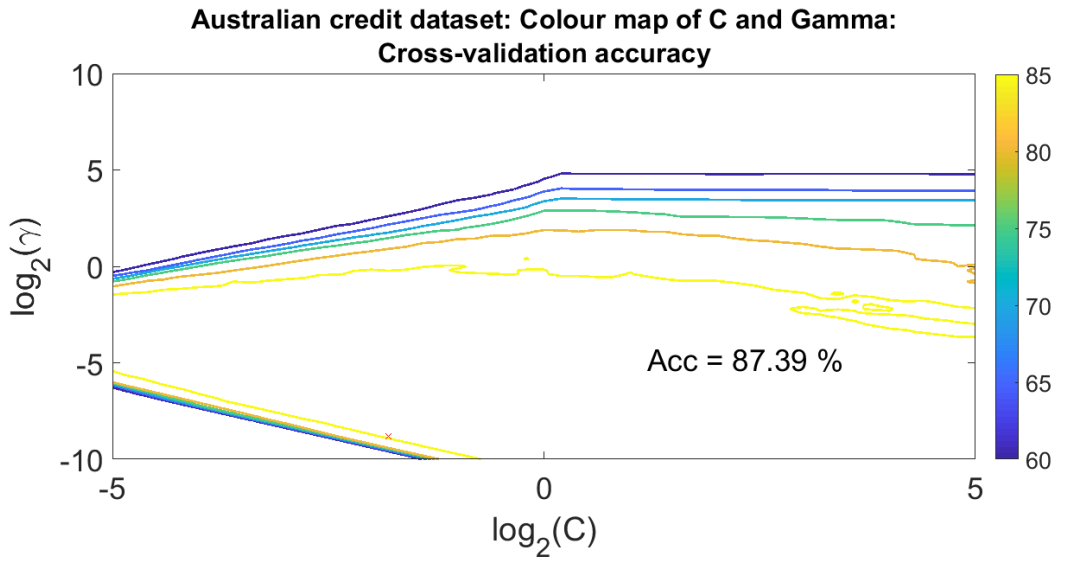

Figure 6-4: Result of grid search for optimised parameter values for Australian credit dataset. The model peaks at Accuracy=87.39%; ***(C=0.2872, γ=0.0022)***

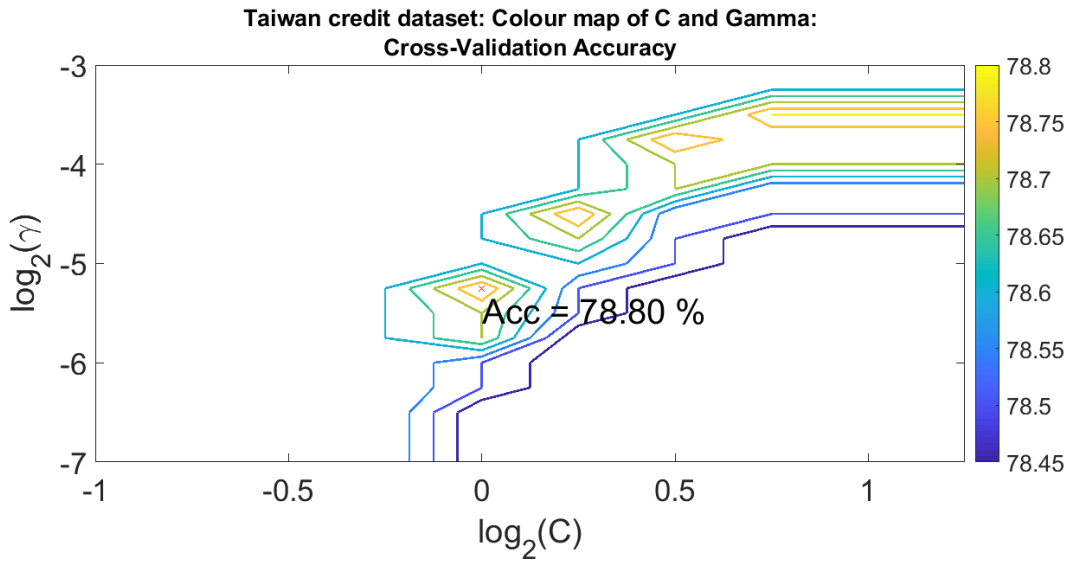

Figure 6-5: Result of grid search for optimised parameter values for Taiwan credit dataset. The model peaks at Accuracy=78.80%; ***(C=1, γ=0.0263)***

Above grid search shows how the SVM classifier is optimised by cross-validation using accuracy score. There are no rules of thumb for grid search parameter optimisation. The parameters are found at the best accuracy score of 77.5%, 87.39% and 78.80% for the German credit, Australian credit and Taiwan credit

datasets respectively. The parameter values obtained are used for the experiments in next sections.

6.5.10.3 Performance of best solutions

To strengthen the significance of feature selection, we first ran experiments on baseline classifiers with all features before applying Genetic Algorithm Wrapper (GAW) and then Information gain directed feature selection (IGDFS) using the three classical classifiers (see Table 6-6).

In GAW, Genetic algorithm acts as a wrapper technique with performance of three classical machine learning algorithms used to obtain the best fitness function. In the IGDFS algorithm, the top-ranking features obtained from information gain ranking are propagated through the wrapper process as shown in Algorithm 1 in previous section.

The results of 10-fold cross validation on GAW and IGDFS for all the datasets are shown in Table 6-6. The best average classification results are printed in bold italics. The GAW and IGDFS algorithms have performed better than the baseline classifier algorithms. Thus, feature selection improves the performance of classification compared to baseline methods. Compared with GAW, IGDFS yields improved accuracy in most of the classifiers except KNN (German credit data) and NB (Taiwan credit data).

Table 6-6: Accuracy of classifiers (Best performance in bold italics)

Method		German Credit data	Australian Credit data	Taiwan Credit data
SVM	Baseline	76.40	85.70	81.90
	GAW	80.40	89.01	81.20
	IGDFS	***82.80***	***90.75***	***82.57***
KNN	Baseline	75.20	85.70	80.80
	GAW	***75.80***	85.65	80.98
	IGDFS	70.20	***86.75***	***81.17***
NB	Baseline	73.70	80.43	71.36
	GAW	76.80	86.79	***82.02***
	IGDFS	***77.30***	***87.971***	81.98

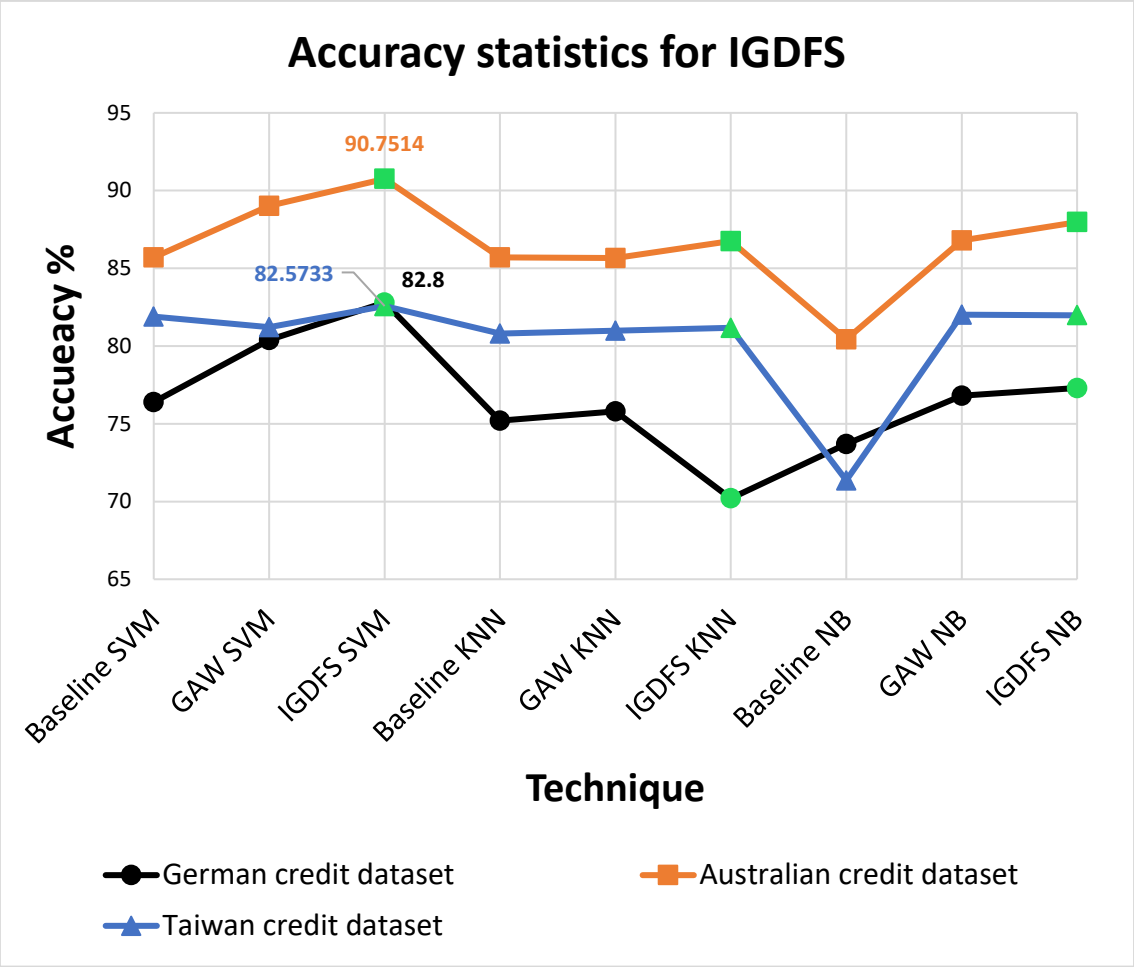

Figure 6-6: Accuracy statistics for the IGDFS algorithm

The plot above shows that the proposed method outperforms all other methods for all three datasets.

6.5.10.4 ROC curves for the best solutions

ROC curves allow for a detailed analysis of the differences. Figure 6-7 shows the ROC curves obtained with IGDFS for the three classifier algorithms on the German credit dataset.

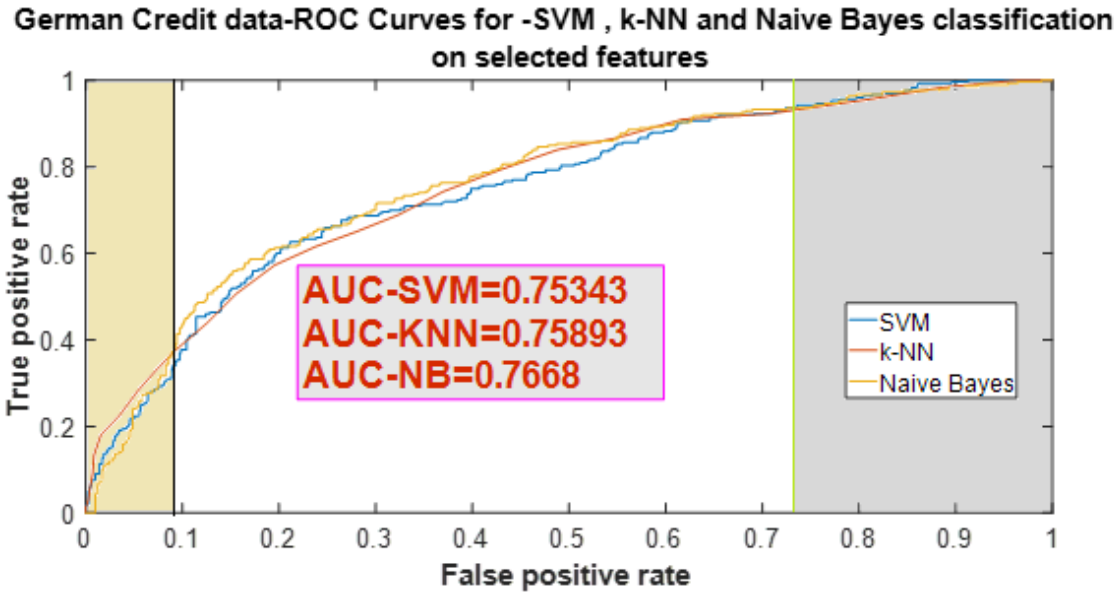

Figure 6-7: ROC results of the IGDFS algorithm on German credit dataset

Figure 6-7 shows that the three classifiers in the wrapper of the GA with IGDFS obtained almost the same performance for this dataset. Curves close to the perfect ROC curve (close to the top left corner) have a better performance level than the ones closer to the baseline. Comparisons of all the classifiers shows that the ROC curves are crossing each other. FPR (=1-Specificity) defines how many samples are classified as bad even if they were good credit. For smaller false positive rates, i.e. for early retrieval area shown with light yellow colour (a region with high specificity values in the ROC space- FPR between 0 and 0.1), IGDFS+KNN classifier (red curve) seems to perform better; i.e. it may be appropriate where we are interested in classifying the positive cases better, (say when the dataset is imbalanced). For middle FPR (between 0.1 and 0.75), IGDFS+NB (yellow curve) gives good results. As the FPR increases beyond 0.75, IGDFS+SVM (blue curve) performs best. Hence this model could be appropriate if we are interested in classifying the negative cases better. This area is shown with grey colour.

While using ROC measure, one effective approach to avoid the potential issues with imbalanced datasets is using the early retrieval area, which is a region with high specificity values in the ROC space. Investigation of this area is useful to analyse the performance with fewer false positives (or small false positive rate).

Figure 6-8 shows the performance of all three classifiers on Australian credit dataset. IGDFS + NB, which has the largest area under ROC curve, performs best in classifying the credit applicants in Australian Credit dataset. Next best performance is shown by IGDFS+KNN, followed by IGDFS+SVM.

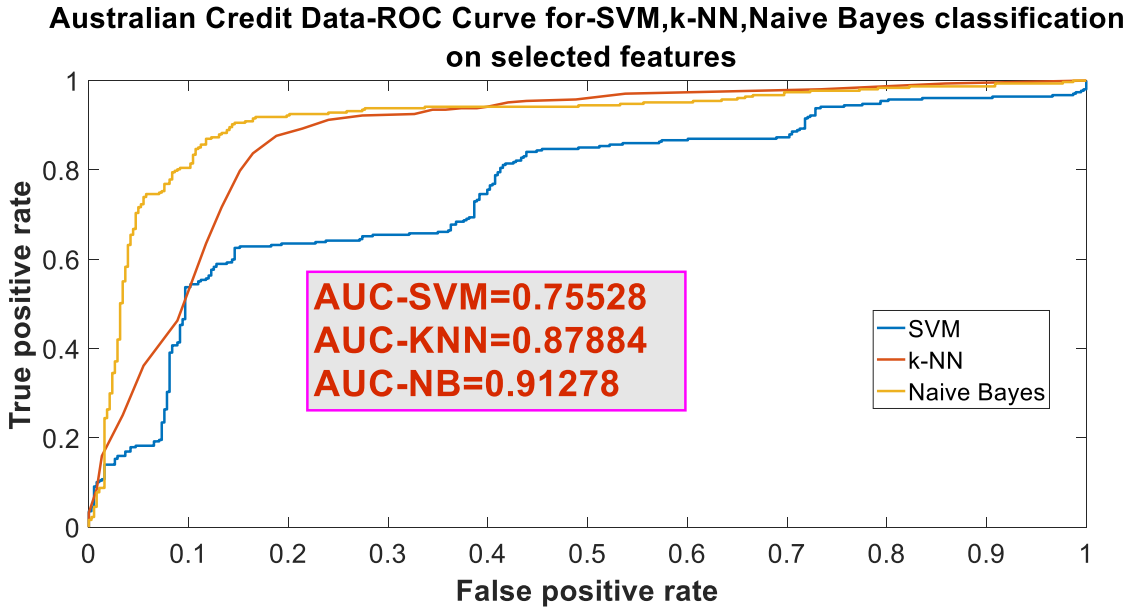

Figure 6-8: ROC results of the IGDFS algorithm on Australian credit dataset

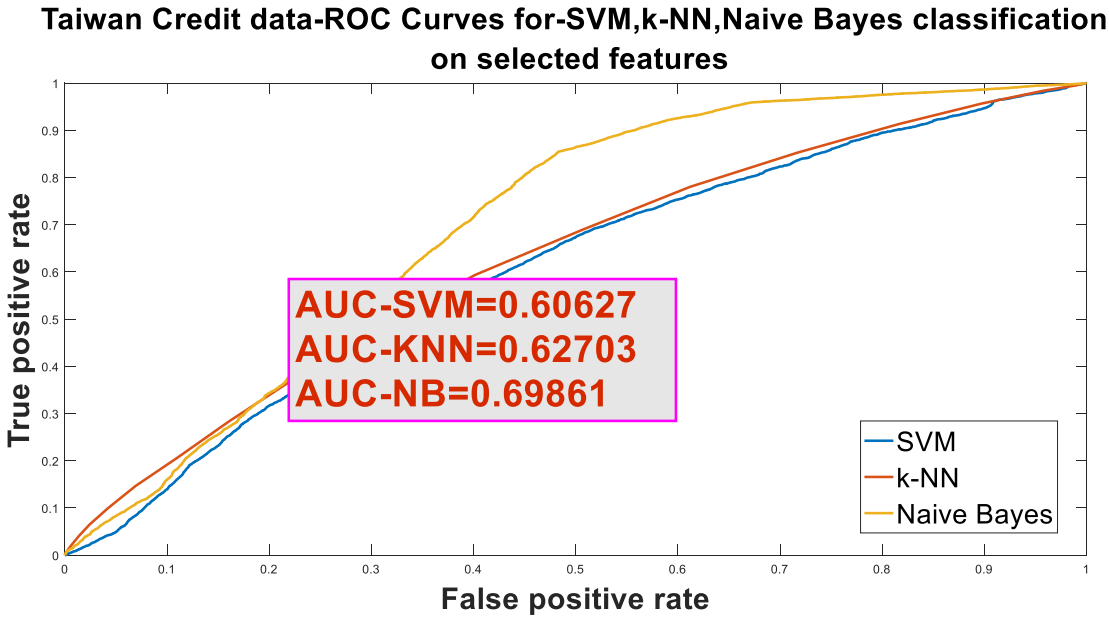

Figure 6-9: ROC results of the IGDFS algorithm on Taiwan credit dataset

For the Taiwan credit dataset IGDFS+NB gives best results followed by IGDFS+KNN, followed by IGDFS+SVM (Figure 6-9).

Analysing the performance from Figure 6-7 to Figure 6-9, the classifier and IGDFS combination giving best ROC performance for all three datasets is IGDFS+NB.

6.5.10.5 Comparison of GAW and IGDFS for all datasets

In this section, the performance of the proposed algorithm, Information gain directed feature selection (IGDFS) is compared with the Genetic Algorithm Wrapper algorithm (GAW) based feature selection method and the baseline classifiers in terms of prediction accuracy made by three different classifiers RBF SVM, KNN, and Naïve Bayes (See Table 6-6). The findings are:

- GAW+SVM performed well compared with baseline SVM for all the datasets. This implies that feature selection may have a positive impact on the performance of RBF SVM for the datasets. This is an improvement in the performance compared with the work done by Liang et al. [243]. But the best accuracy results are obtained by the improved IGDFS algorithm, where we identified important features and propagated them throughout the whole wrapper process.
- GAW+KNN performed slightly better than baseline KNN over the German and Taiwan credit dataset, but not for Australian credit dataset. The proposed IGDFS with KNN has performed best in Australian and Taiwan credit dataset but not the German credit dataset.
- GAW+NB significantly outperformed the baseline NB in all three datasets. This finding was consistent with similar work by Chen et al. [266], who found that NB classifier was highly sensitive to feature selection and the work done by Liang et al [243]. The IGDFS again has proved to be the best method and it gives the highest prediction accuracy for NB in all the datasets.

For the German credit data with 7 numerical and 13 categorical features:

- There is much less variation in the classification accuracy of IGDFS for all three classifiers;

- Wrapper methods (GAW) clearly outperform baseline methods;
- These GAW methods have shown very high-performance improvement when used with SVM and NB as underlying classifier and acceptable classification accuracy improvement on the German dataset with KNN.

For the Australian credit data (6 numerical and 8 categorical features):

- There is a lot of variation in the classification accuracy of IGDFS for all three classifiers;
- Wrapper methods (GAW) outperform baseline methods except for the KNN method;
- IGDFS performs best for all the three classifiers.

For the Taiwan credit data (with 16 numeric and 8 categorical features):

- There is not much variation in the results for all the three techniques;
- IGDFS performs better than GAW and baseline for all three classifiers.

6.6 Conclusions

Credit scoring and classification are significant problems in computational finance. In this work, we developed an elegant feature selection algorithm, called IGDFS, for credit scoring application based on Information Gain and Wrapper technique using three different classical machine learning algorithms: Support Vector Machines, k-Nearest Neighbour and Naïve Bayes. The average prediction results by IGDFS, Genetic Algorithm Wrapper and Baseline models are compared.

The intuition behind this work is that not all features are equally important and retaining the top contributing feature into the final selected subset may improve the results of classification, as those features that are not important to decision making could affect the performance of decision making.

Observing all the datasets investigated in this study, the classification accuracy achieved with different feature selection strategies is highly sensitive to the type

of data, total number of samples and the number of positive and negative samples in the dataset.

Among the three machine learning algorithms investigated, the SVM baseline, GAW and IGDFS accuracies are better than KNN and NB baseline, GAW and IGDFS. This provides an evidence of that SVMs may indeed suffer in high dimensional spaces where many features are irrelevant and feature selection may result in significant improvement in their performance [267].

GAW+KNN and IGDFS+KNN cannot improve the accuracy of classification on the selected feature sets for all the datasets, compared to the performance of baseline KNN only on the full features, even for German credit dataset, the accuracy obtained by IGDFS dropped. This might be because KNN is sensitive to the local structure of the data, and the data structure is decided by Euclidean distance. When we remove some features that have low information gain to the decision making, the reduction of features could affect the structure. Namely, the information gain of features could produce conflict with the original data structure for KNN.

Wrapper feature selection is a costly method due to its comprehensive search on the feature space. To reduce its computational cost, we used an Information Gain Directed Feature selection strategy in the first phase of the proposed method. This step removes features with low information gain, so that the wrapper method is carried out on a smaller space, and the time complexity is reduced. This can be seen by the results on all three credit datasets used in the study. We can conclude that there is a potential for improvement in the models' performances if the feature selection method is chosen carefully.

In future studies, the results with other combinations of parameters for genetic algorithms could be studied. The performance of IGDFS algorithm could be compared with other high dimensional datasets. Because of the nature of the credit scoring problem and its real application domain, computational time has a great significance for credit scoring model [268]. Reducing the cost of credit analysis and aiding faster credit evaluation are among top objectives of credit

scoring models. The computational complexity of the proposed algorithm, both in training and at runtime needs to be assessed to make it robust. Also, this work could be extended for other evolutionary algorithms such as Particle swarm optimisation and with other wrapper algorithms.

When deploying any application in real scenarios, cost vs. performance comparison needs consideration. Estimating the performance of a technique when high-performing resources are available is different from when working with a typical desktop. But the improvement obtained by the IGDFS algorithm is an indication that this technique has enough potential to explore in this direction.

7 PREDICTION OF EARNINGS PER SHARE FOR INDUSTRY

7.1 Introduction

The industry dataset available used in this PhD work is analysed in chapter 4. The dataset contains information about estimated Earnings Per Share (EPS) and sales information for a number of companies being traded on stock market. This chapter discusses the study undertaken to predict EPS for some of the companies from the dataset.

Traditionally, financial market analysis has been relying on past experience, knowledge and intuition. This makes sense when one has vast experience to back the decisions. A more rational approach using automation process which reduces the likelihood of errors, has been growing steadily because of the availability of large finance data. There is a growing evidence of research in the fields of data mining and machine learning and their applications to computational finance industry.

In a mature finance industry, a company that takes the dominant position in the industry earns greater profits because of better ways of handling its economic scale and market power [269]. Evaluation of stocks of a company to buy or sell is an important decision to be made by the investors of a company. Nowadays, when huge amounts of data are made available with the advent of technology, this decision does not become any easier without the help of some model. Thus, determining the best model directly affects the investment decisions for a company.

Earnings Per Share (EPS) is considered as one of the most important profitability metrics of a company. It represents the returns delivered by the company for each outstanding share of common stock. It is a major indicator for investors to purchase stocks. Price Earnings (PE) ratio is obtained by dividing the stock price by EPS. The EPS used here can be current or future earnings. EPS over past quarters as well as “forward” forecasted quarters are most frequently used in the calculation of PE ratio of a company. Comparison of a stock’s current PE with

those of its competitors or with its own average multiple over three to ten years gives useful information about hopeful future profits, investment in the company and also if a possible bargain has happened. Investment into a stock depends on the current PE ratio: Is it too high or low compared with the PE ratio of the stock's peers, industry or aggregate market?

The study also proposes three regression models to predict EPS: (a) Statistical Regression Model using Linear Regression (LR); (b) Neural network (NN) regression using Multilayer Perceptron (MLP) and (c) Neural network regression using Radial Basis Function (RBF). For construction of these models, quarterly EPS data are employed. The experimental results indicate that LR and MLP models outperform the RBF models, except for the high nonlinear data, where MLP gives better performance.

The rest of the chapter is organised as follows. Next section reviews the literature on usage of EPS for stock price forecasting. Then, the methodology of the research along with the applied methods is introduced. The dataset and experimental set up of the work is described next. The experimental results of forecasting performance across the LR and NNs are compared in next section followed by conclusions of the work.

7.2 Related Work

Prediction of EPS forms the basis for stock price forecasting. Forecasting is a function approximation problem involving choosing a model and fitting its parameters to the data. This problem is complex because of stock price changes in time being highly nonlinear. Many artificial intelligence, soft computing and machine learning methods have been used wherein neural networks and regression show good results since they are robust against noise, can model nonlinear relationships and give good generalisation performance.

Data mining and regression have long been researched upon to solve various problems. There are three types of Regression models such as Linear, Polynomial and Logistic Regression.

Regression modelling has many applications wherein the output is continuous such as in trend analysis, business planning, marketing, financial forecasting, time series prediction, biomedical and drug response modelling, and environmental modelling [270].

Artificial neural networks (ANNs) are one of the most common supervised data mining techniques used by the industry for forecasting.

MLPs have been employed for prediction of stock prices and indexes on various stock markets, see: [124], [271], [272]. Similarly, RBF neural networks were the topic of choice for same purpose in: [273], [274], [275]. Use of RBFs along with various other data mining techniques can be found in [276], [277], [278].

Other research regarding forecasting and prediction in the area of finance focuses on stock market, bankruptcy, fraud, credit scoring and business failures. Bankruptcy prediction attempts to predict bankruptcy and financial distress of public firms. It is one of the vast areas of finance research. Creditors and investors have always given importance to the evaluation of credit worthiness of firms.

A lot of them consider ANNs as the main technique of forecasting [84], [179], [111], [279], [280], [281], [51], [282], [283], [284], [285]. The learning and predicting potential of the adaptive neuro-fuzzy inference system (ANFIS) model, a variant of ANN is used for stock market returns prediction in [286], [287], [288].

Classic economic model of regression is used to predict stock trends in [289]. To obtain n-day ahead volatility forecasts, the implied volatility may be parameterised within an ARCH model [290]. Similarly, Regression along with neural network was applied in [291] and along with support vector machines was investigated in [154]. Other research work using Regression can be found in [292], [293], [294].

The observation is that models based on neural networks are suitable for stock market related forecasting. They are efficient at producing better results for trading systems with higher forecasting accuracy. The literature demonstrates

that soft computing techniques have natural connection with classical statistics methods and have been used alongside conventional models. However, difficulties arise when defining the structure of the model (the hidden layers, number of neurons etc.). While determining the structure of the model trial and error procedures are still employed.

A company's stock price is mainly affected by EPS since the stocks vary according to EPS ratio. Researchers have investigated several methods to construct models taking help of EPS: [295] suggested that firms disclose more frequently when experiencing favourable earnings results and that earnings forecasts are, usually associated with positive returns. Financial distress prediction was the topic of study in [296] wherein EPS was used as one of the inputs to neural networks.

A study involving financial ratios included EPS among others and showed that application of ensemble methods with diverse models have good predictive capacity and have good applications in the area of forecasting. PE ratio has been used in the research work to select stocks using neural network [297]. Few researchers have taken into consideration the EPS ratio as part of dataset. [298], [299] used it for stock price forecasting and [300] used it for financial crisis prediction.

In [301], a method of SVM was proposed with financial statement analysis for prediction of stocks using EPS as one of the finance parameters. EPS was used as a financial variable for financial crisis prediction in [302]. SVM and ANN models including PE ratio as one of the basic financial indicator give meaningful performance results for the stock selection [303]. Many stock prediction, stock selection, financial crisis prediction and fraud detection studies have used EPS as part of the study: [304], [305], [306], [307], [308], [309].

Actual EPS forecasting was the topic of research in few studies. In an interesting study of Markov process model to forecast subsequent quarterly EPS values, the authors applied time independent transition probability matrices to predict EPS of IT companies [310].

EPS forecasting using machine learning techniques is still a new area. Neural networks seem obvious choice to model nonlinear data, but need the decision about parameters, architecture and speed.

When a real problem needs to be solved, the goal is to find an approach as easy as possible with the performance as good as possible. Therefore, we select three models of LR, MLP and RBF for the EPS problem, and compare the suitability for the real data.

7.3 Data for the Experiments

This study uses a real industry dataset provided by FactSet Research Systems Inc. [13]. The dataset contains finance data for companies being traded in the stock market. The nature of the data is estimates, each value is an estimate made by "the market" on what the value of a specific piece of information about a given company will be at a time in the future. The data is described and analysed in detail in chapter 4.

7.3.1 Linearity Analysis

As discussed in chapter 4, quarterly EPS data for six companies is chosen for this study. The data is analysed for linearity using the method of correlation coefficients and the findings are shown in Table 7-1.

Table 7-1: Correlation coefficients for the chosen companies

Company 1	Company 2	Company 3	Company 4	Company 5	Company 6
0.642909	0.655228	0.655228	0.796691	0.281004	0.860497

The values indicate that the data for the company 6 is highly linear and that for the company 5 is highly nonlinear.

7.3.2 Linear Regression

Regression process starts with a dataset where the target values are known and other attributes might be the predictors in predicting value of the target. While building the regression model, the algorithm which is a relationship between

predictors and target estimates the target as a function of the predictors for each observation in the dataset. This model then can be applied to a dataset not seen by the model previously to determine target values.

Least squares regression, a standard approach to regression, finds a best-fitting line that minimises the mean squared difference between the observed values and the fitted values.

The simplest regression model is the linear regression model, which represents the linear relations between independent (also called as x-variables or predictors) and dependent variables (y-variables, response variables or goal variables), as shown in formula (7-1).

$$f(x) = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + \cdots \quad \textbf{(7-1)}$$

7.3.3 MLP Architecture- Feedforward Neural Networks

Artificial Neural Networks (ANN) are universal approximators, and they are very popular with regression applications where they obtain a close relation to a continuous objective function. As they are data-driven, if a good training dataset is available, they provide good forecasting results. ANN comprises a set of neural perceptrons. A Perceptron is a simplest neural network. It is a linear classifier, using sigmoid function as the activation function. A perceptron can be described with the following functions:

$$u = \sum_{i=1}^{N} W_i(p)x_i^n(p) \quad \textbf{(7-2)}$$

$$v = f(u) = \frac{1}{1+e^{-u}} \quad \textbf{(7-3)}$$

where u are activations, N is the total number of nodes in input layer, x_i is input vector, W_i is the weight vector connecting the neuron of the output layer for the pattern p, v is the output of hidden units which is a nonlinear sigmoid activation function.

Multilayer Perceptron (MLP) [311] extends the concept of perceptron by adding one or more hidden layers of neurons. Neural network is usually used to extract patterns from complex data, as adaptive learning makes it easy to model complex data, and they do not assume about underlying probability density functions or any information regarding the modelling sample under consideration. Therefore, we investigate the multiple layer perceptron regression for the real predictive problem, and use classic backpropagation algorithm to train the neural network.

The linearity analysis results from section 7.3.1 show that the data for company 5 is highly nonlinear. The MLP technique is chosen to investigate results on this data.

7.3.4 RBF Network Architecture

RBF network is one of the most popular neural networks and is a main competitor for MLP networks. RBFs are faster to train than MLPs of a similar size, as RBF is a feed forward neural network with a single hidden layer. But the number of hidden layer neurons required for RBF neural networks grows exponentially with the number of inputs. A unique feature of this network is the process that is performed in the hidden layer. Input layer sends the input value to each of the nodes in the hidden layer. Each node in the hidden layer (neurons) is characterised by a transfer function G. Usually the transfer function uses radial basis functions (e.g. Gaussian functions in formula (7-4)) as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters (See formula (7-5)).

$$f(x) = a\ exp\left(-\frac{((x-b)^2)}{2c^2}\right) \quad \textbf{(7-4)}$$

$$GW = b \quad \textbf{(7-5)}$$

where G is the transfer function, W is the weight vector, linking the hidden layer to the output layer and b is the output. To avoid overtraining of the network, 10-fold cross validation method is used. This method splits the data into 10 parts of equal size. In each of the 10 iterations, one part is used as testing set and

remaining as training sets. At the end of 10 runs, overall performance is the average of all runs' results.

7.4 Problem formulation

With time series, the variable to be predicted needs to be chosen and feature engineering applied to construct all of the inputs that will be used to make predictions for future time steps. Before applying a machine learning technique to a time series, the problems have to be re-framed as supervised learning problems. The data has to be transformed from one sequence of values to pairs of input and output sequences of values. We employ the technique of lag features which are the classical way to transform the time series forecasting problem into a supervised learning problem.

When fitting time series with regression models, one way is to use lagged versions of the variables in the regression model. This allows varying amounts of recent history to be brought into the forecast. Lagging allows the regression model to be able to predict what will happen in period t based on knowledge of what happened up to period t-n.

Essentially, a prediction problem learns a mapping or function, $y = f(x)$, where x is the input and y is the continuous output to model the relationship between x and y.

We formulate the EPS prediction problems into two subproblems given by (7-6) and (7-7) as:

For a given time series x, determine the function f such that:

$$y = x_{t+5} = f\,(x_{t-4}, x_{t-3}, x_{t-2}, x_{t-1}) \qquad \textbf{(7-6)}$$

The EPS numeric data is organised in matrix as follows:

In Table 7-2, x_1-x_4 are inputs of the model and y is the target.

Table 7-2: Problem 1

x_1	x_2	x_3	x_4	y

1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9

We expand the window width to include lagged features:

$$y = x_{t+6} = f\ (\ x_{t-4}, x_{t-3}, x_{t-2}, x_{t-1}) \quad \textbf{(7-7)}$$

In Table 7-3, x_1-x_4 are the inputs of the model and y is the sixth value as the target.

Table 7-3: Problem 2

x_1	x_2	x_3	x_4	y
1	2	3	4	6
2	3	4	5	7
3	4	5	6	8
4	5	6	7	9
5	6	7	8	10

7.4.1 Experimental setup

1. The purpose of the work is to find the best model for the quarterly EPS prediction problem for companies traded on the stock market.
2. Six companies with highest number of data points for quarterly EPS values are chosen for experiments.
3. The experiments are performed to compare the performance of three models LR, MLP and RBF.
4. All the experiments are run using 10-fold cross validation.
5. WEKA is used as the experimental platform.

7.4.2 Software

For this study, we used WEKA (Waikato Environment for Knowledge Analysis) software that is developed at the University of Waikato in New Zealand. WEKA is a very powerful data mining and machine learning workbench. It is an open

Source application, developed in Java and allows easy creation of classifiers. The philosophy behind WEKA is to move away from supporting a computer science or machine learning researcher, and towards supporting the end user of machine learning. WEKA is useful if one wants to explore the data or build a model quickly. It is most useful if one just wants to run some standard out-of-the-box classifiers against some data. It has a collection of good visualisation tools, diverse classification, regression and clustering algorithms, and good feature selection support.

7.4.3 Performance Evaluation

In this study, Correlation Coefficient (r) and the root mean square error (RMSE) are used for the evaluation of performance of the models.

A correlation coefficient equal to zero indicates no relationship between the variables; i.e. if one variable changes, the other may or may not change. A correlation of +1.00 or -1.00 indicates that the variables involved are perfectly associated positively or negatively. A higher correlation coefficient indicates better fitting to the data. It can be calculated with formula (7-8).

$$r = \frac{\sum_{i=1}^{n}(Y_{act} - \bar{Y}_{act})(Y_{est} - \bar{Y}_{est})}{\sqrt{\sum_{i=1}^{n}(Y_{act} - \bar{Y}_{act})^2\,(Y_{est} - \bar{Y}_{est})^2}} \quad \textbf{(7-8)}$$

RMSE gives the measure of the difference between values predicted by the model and the real values. The lower RMSE indicates the higher accuracy. It can be calculated using the formula (7-9).

$$RMSE = \frac{\sqrt{\sum_{i=1}^{n}(Y_{est} - Y_{act})^2}}{n} \quad \textbf{(7-9)}$$

Where: n is the sample size, Y_{act} is the real observed value, $\overline{Y}_{act}$ is the average of real observed value, Y_{est} is the predicted value, $\overline{Y}_{est}$ is the average of predicted value from the model.

7.5 Results and Evaluation

The performance of all the models built by the three algorithms for Correlation Coefficient r and RMSE in WEKA is shown in figures below.

7.5.1 Problem 1

In Problem 1, we predict fifth value using previous four values. Figure 7-1 illustrates the Correlation coefficient obtained with the three models for Problem 1. From this figure, we observe that the three models obtained similar performance for all the companies except for Company 5, for which MLP is slightly better than LR and RBF. Also, the Correlation Coefficient for Company 5 is lowest among all the six companies. It means the data for Company 5 is weakly linear. This aligns with the findings in section 7.3.1. Hence MLP obtained better performance for Problem 1.

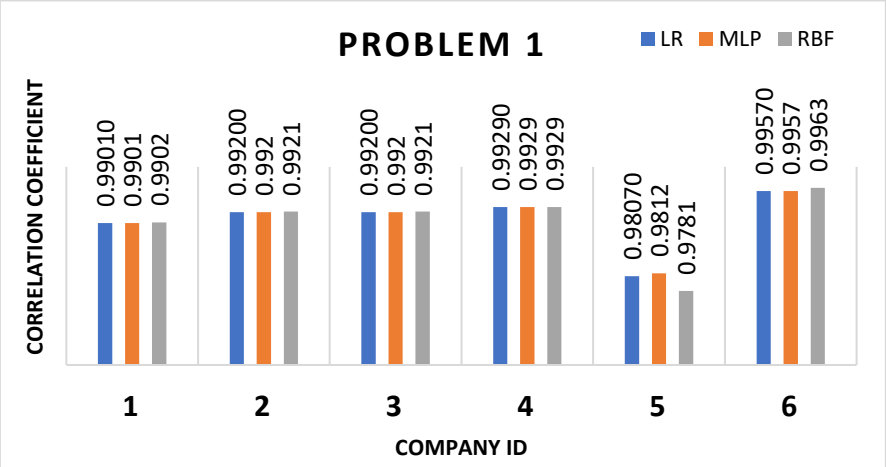

Figure 7-1: Correlation Coefficient obtained with the three models for Problem 1

Figure 7-2 shows the RMSE obtained with the three models for Problem 1. Obviously, it can be seen that RMSE for Company 4 is highest among all the companies. Although for all companies the three models obtained similar RMSE, for Company 5, MLP obtained the lowest RMSE compared with other two models for Problem 1. This is consistent with the Correlation Coefficient for Company 5 in Figure 7-1 .

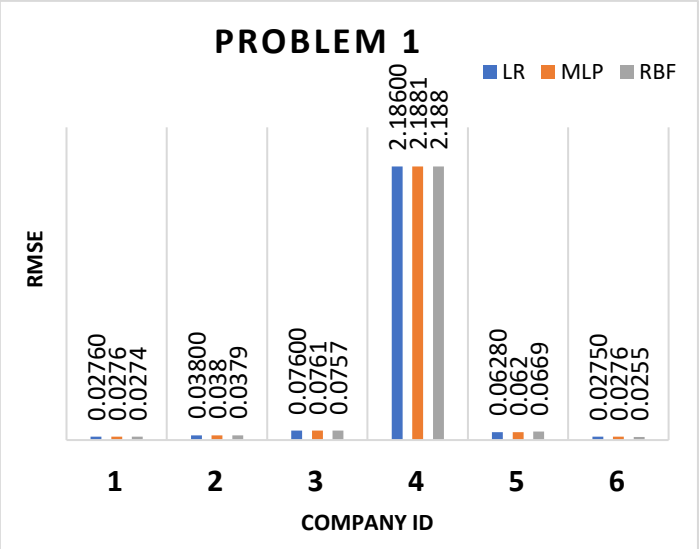

Figure 7-2: RMSE obtained with the three models for Problem 1

7.5.2 Problem 2

Figure 7-3 illustrates the Coefficient of Correlation of six companies for Problem 2. From this figure, the performance in Correlation Coefficient for all the six companies for Problem 2 is similar to the performance for Problem 1. But all values for Problem 2 are lower than that of Problem 1. Company 5 still got the lowest Correlation Coefficient among all the companies. LR and MLP obtained better performance than RBF.

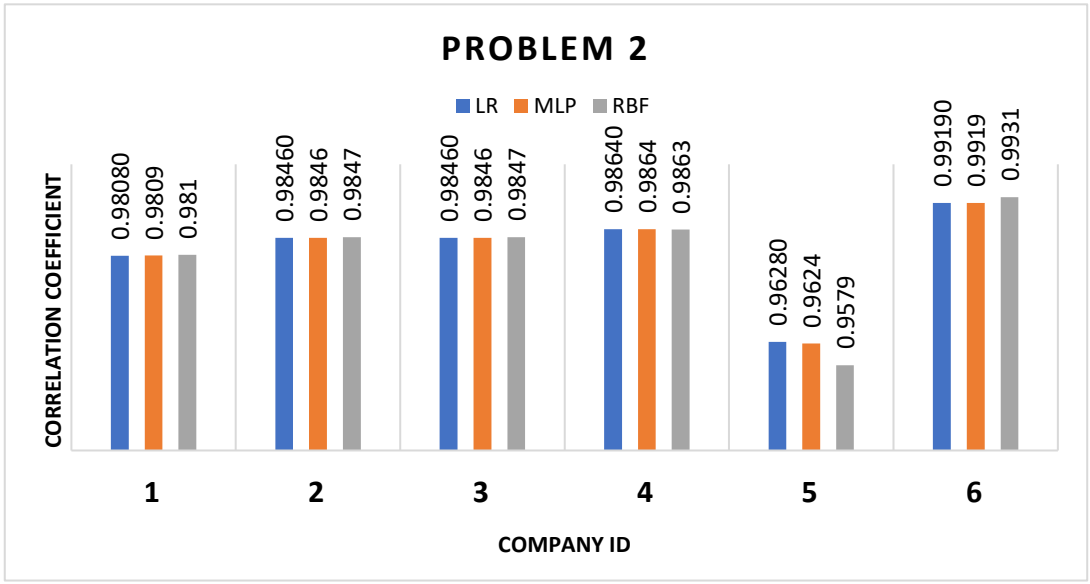

Figure 7-3: Correlation Coefficient obtained with the three models for Problem 2

From Figure 7-4, we observe that the RMSE of six companies for Problem 2 are similar to that for Problem 1. But the RMSE values of all companies for Problem 2 are larger than for Problem 1. For each company, the order of three models' performance in RMSE in Figure 7-4 is the same as the order of three models' performance in Correlation Coefficient in Figure 7-3 for Problem 2. The Correlation Coefficient is consistent to the RMSE assessment.

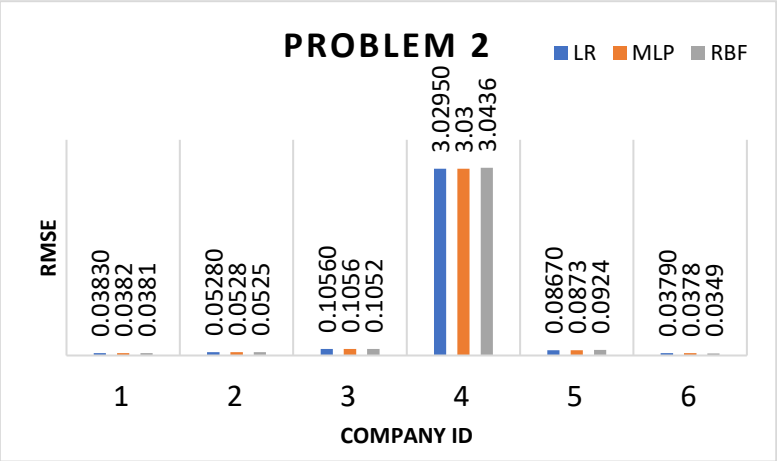

Figure 7-4: RMSE obtained with the three models for Problem 2

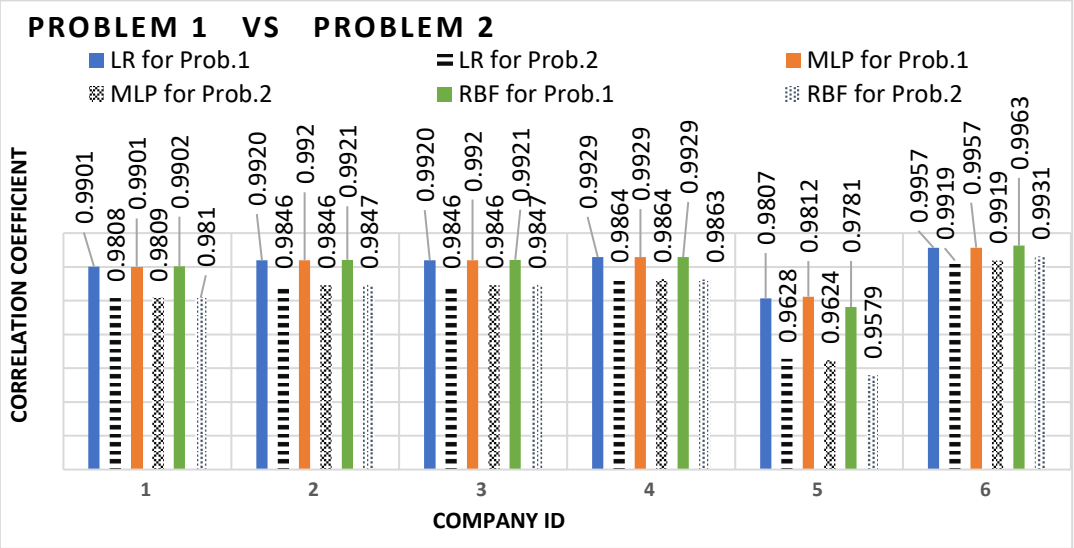

Figure 7-5: Correlation Coefficient for all the three models for six companies in both the problems

In summary, Figure 7-5 illustrates all the Correlation Coefficients of six companies for Problems 1 and 2. It can be seen that the performance of the three models for

Problem 1 is better than that for Problem 2. For company 5, which has high non-linearity, the MLP obtained the best performance.

7.6 Conclusions

The aim of this chapter was to predict the EPS for companies traded on the stock market using a private real dataset.

Before investing in a company, the key element all investors look after is earnings of the company, i.e. how much the company is making in profits. Future estimated earnings are a key factor as the future prospects of the company's business and potential growth opportunities are the direct determinants of its stock price. The observations of this study could be used by investors and traders alike to make investment decisions.

We employed three models (LR, MLP and RBF) to predict the change in the EPS of market firms with historical data. The experiments were carried out by running the three models on the data of six companies. We use the Correlation Coefficient and RMSE to assess the performance of the three models on the data of the six companies.

The experimental results show that:

- MLP obtained best performance for high non-linear data.
- The performance in Correlation Coefficient is consistent to the performance in RMSE for the three models.
- The performance of the three models for Problem 1 (where the lag window is smaller) is better than their performance for Problem 2 (where the lag window is wider).
- The conclusion is that we need to use different models for different data.

The range of the data used in this work is 13 years from year 2001 to year 2014. It future, the impact of volatility of the finance market on the results obtained in this work could be investigated.

8 CONCLUSION AND FUTURE WORK

8.1 Introduction

This chapter briefly summarise key contributions of this work and discuss several future work directions.

This study involves systematic work achieved via series of steps, which is used as guideline throughout the research, in order to accomplish the objectives.

In this PhD dissertation, we developed the research methodology to address four areas: Elaborate data analysis of two types of finance datasets, public and private; appraising a few classical machine learning techniques for the problem of credit scoring; development of a new algorithm for the problem of feature selection in the area of credit scoring; prediction of finance ratio of EPS for a few companies.

This study started with the aim of finding the state-of-the-art of computational finance industry. This was followed by addressing the four areas as outlined in the research methodology. In what follows, we will provide a chapter-by-chapter overview of the thesis and indicate the major contributions. Furthermore, we will also outline some issues for further research.

The chapter is divided into five sections: Section 8.1 introduces the chapter aim; Section 8.2 discusses the research findings of this study; Section 8.3 summaries the contributions and achievements of the research; Section 8.4 highlights the research limitations, finally, section 8.5 summarises recommendations for future work.

8.2 Research findings of the study

This section details the major findings of the study. The key findings are discussed based on the order of work followed in the work and experimental results accompanied by the author's interpretation and opinions.

8.2.1 Computational Finance and Data Mining: The state of the art

A comprehensive literature review of more than 200 research publications was undertaken in chapter 3 to survey the data mining and machine learning techniques applied to the finance industry with a focus on the publications published in the year 2010 to 2015. This study categorised 179 research articles into five application areas in finance industry. The key findings are:

1. The most investigated finance area in the decreasing order are: Credit Rating, followed by Stocks Prediction, Loan Default Prediction, Time Series Forecasting and lastly Money Laundering.
2. The review indicated that stock prediction and credit rating have received most attention of researchers, compared to loan default prediction, money laundering and time series forecasting.
3. Due to the dynamics, uncertainty and variety of data in the field of finance, nonlinear mapping techniques have been studied more extensively than linear techniques. Also, it has been proved that hybrid methods are more accurate in prediction, closely followed by neural network technique.
4. There is no single algorithm that is better than all the others across all types of datasets and problems. The choice is governed by the important aspects of dataset being used including the importance of features in the dataset, the problem area, research objective and performance evaluation criteria.
5. Hybrid models provide better and more accurate results, and hence are used more in the area of credit scoring and stock-market prediction. This necessitates sufficiently rigorous tests entailed by hybrid techniques to strengthen the findings. Hybrid methods are followed by SVM and Neural network close to each other for the problem of credit scoring. In chapter 5 and 6, SVM along with other techniques are used for the problem of credit scoring. Neural networks are explored in chapter 7.
6. The comprehensive literature review of the state of the art in finance industry has indicated that the existing systems have room for improvement to help the industry to achieve effective and efficient performance.

8.2.2 Data analysis

A systematic data analysis was performed in chapter 4 for the public and private datasets to investigate the relationships among variables. This involved data transformations and quantifying the correlations between the key variables in private dataset. The public credit scoring datasets have the credit scoring information for the credit applicants such as their age, income etc. and their credit status as ‘creditworthy’ or ‘non-creditworthy’. The private dataset is the estimates for Earnings Per Share (EPS) and Sales for companies being traded in the stock market. The key revelations of this analysis are:

1. For the three public datasets, the variables exhibiting highest level of correlation were identified.
2. It was observed that for the German and Australian credit dataset, very few variables show higher degree of correlation whereas for the Taiwan credit dataset, higher number of variable show good correlation. These were the key findings for the classification techniques are applied for the same datasets in chapter 5.
3. For the private dataset, six companies with highest size of data were chosen. The observation was that some of the companies showed initial negative estimated EPS values. The reason could be that the companies were operating at a loss. EPS estimates were found to be ‘bumpy’ whereas Sales estimates were ‘smooth’. Few companies show plateaued sales performance in a period of time. These findings are crucial in analysing a company’s performance as they indicate how each company was performing as well as it could be an indicator of the whole market e.g. upcoming recession.
4. Two of the companies showed good correlation between EPS and Sales values, whereas rest of them did not. This result is an indication that only two companies followed the EPS trend in Sales too. Such results could be used to better the procedure used to estimate the per share values in future.
5. Data analysis

8.2.3 Investigation of the credit scoring problem

The goal of credit scoring is to distinguish between good and bad payers by using a model which is built based on the repayment behaviour of set of applicants from the past. This problem statement essentially reduces to a binary classification problem. Many techniques have been suggested to tackle this. In chapter 3, we started with providing an overview of the state of the art in this field. In chapter 5, we then conducted a benchmarking study in order to validate the performance of a few of the classification techniques discussed on three publicly available credit data sets. The differences in classifier performance were quantified after investigating the credit scoring problem applied to public datasets using few classical machine learning algorithms. The choice of classifiers in this work run counter to the conventional / widely studied methods such as logistic regression for credit scoring. However, the findings of this study do not imply that the methods chosen will always yield best performance. However, the classical ML techniques are worth exploring in research community.

The results of the experiments showed that:

1. First, accuracy statistics for twelve machine learning techniques was determined for the three datasets. It was concluded that the medium Gaussian SVM classifier consistently yielded a very good performance on all data sets. The coarse gaussian and fine gaussian SVM produced moderate results.
2. The chosen three techniques are studied in detail for AUC and confusion matrix performance for each of the two classes.
3. Another objective of this work was to analyse the effect of varying the kernel parameters cost of penalty and gamma for Gaussian kernel SVM. For the German, Australian and Taiwan credit dataset, the best C values are 1.01, 0.09, 0.98 respectively. The best gamma values were 5 for all the three datasets. This experiment did not make use of any advanced parameter tuning method.
4. The execution time statistics for the three datasets showed that the linear SVM requires longest execution time for German and Australian credit datasets, but shortest for the Taiwan credit dataset. The reason behind this is

that the first two datasets show high non-linearity and the third one is highly linear as found in chapter 4, i.e. the correlation assessment of datasets from chapter 4. This result proved that the linearity of data affects the execution time performance of a dataset.

5. We concluded that machine learning techniques may be adopted to develop intelligent systems for credit scoring.
6. Often accuracy is not enough to analyse the performance of a technique and deploy it in real time. The characteristics of the dataset, importance given to false positives and false negatives and a number of other parameters relevant to the problem in hand decide the choice of a model since these parameters incur costs in real applications. Different measures along with accuracy statistics allow for trade-offs that need to be chosen wisely.

8.2.4 Information Gain Directed Feature Selection for the problem of credit scoring

Practitioners and researchers have developed a variety of traditional statistical models and data mining tools for credit scoring. In most real-world credit scoring applications, the data is highly dimensional since credit institutions want to capture as much information about credit applicants as is possible to help make the credit approval decision. Information theory has been used since long to determine worth of a feature in a dataset. Irrespective of the research carried out on Feature selection (FS) using genetic algorithm as wrapper to improve the performance of credit scoring models, no overall best method exists which could be used in all credit scoring problems. The problem of credit scoring is addressed as a classification and feature subset selection problem in chapter 6.

First, experiments were conducted with Baseline and GA wrapper (GAW) with the SVM, KNN and NB techniques. Then a new algorithm is proposed based on the information gain directed feature selection (IGDFS) using genetic algorithm wrapper techniques applied to credit scoring problem.

In the proposed IGDFS algorithm, the accuracy of SVM, KNN and Naïve Bayes classifiers determined the fitness function of the genetic algorithm. Grid search

method was adopted for parameter selection of RBF SVM kernel. First, the information gain ranking of the features in the dataset is determined. The proposed Information Gain Directed Feature selection algorithm uses top n features with highest information gain as input to the GA algorithm. The population is maintained to propagate the top-ranking feature in the next generation. The crossover function is customised to propagate the top ranked feature. When the parents are selected to mutate, the topmost feature is propagated through.

The results of proposed IGDFS algorithm are compared with the Baseline and GAW techniques. The findings of this work are:

1. Comparing Baseline, GAW and IGDFS: GAW and IGDFS perform better than the Baseline models. We conclude that feature selection improved the performance of classification. IGDFS showed best performance in 7 out of 9 cases.
2. GAW+SVM performed well compared with Baseline SVM for all the datasets. This implies that feature selection may have a positive impact on the performance of RBF SVM for the datasets. These findings are an improvement in the performance compared with the work done by Liang et al. [243].
3. GAW+NB significantly outperformed the baseline NB in all three datasets. This finding was consistent with similar work by Chen et al. [266], who found that NB classifier was highly sensitive to feature selection and the work done by Liang et al [243].
4. Analysing performance of IGDFS: Best ROC results are yielded by the Naïve Bayes technique for all the datasets.
5. Analysing which technique worked best on the three datasets: The proposed IGDFS performed best compared with Baseline and GAW for all the datasets. For the German credit dataset, not huge improvement is seen but even a slightest improvement could give a positive direction for further research.

6. We concluded that information gain guiding the first stage of feature selection can reduce the computing complexity of GA wrapper and improve results of feature selection.

What remains to be investigated is how the IGDFS algorithm will perform in high-computing environment.

8.2.5 Prediction of Earnings Per Share for companies

Investors and stock traders alike are eager to be able to predict what the stock markets will do in future. In chapter 7, machine learning models are employed to predict the Earnings Per Share (EPS) of market firms with historical data. The dataset is time series with estimates of EPS values. The main hypothesis for this work was that by applying machine learning algorithms and training it on the past data, it is possible to predict the Earnings per Share over certain fixed amount of time. Three regression models are proposed to predict EPS: (1) Statistical Regression Model using Linear Regression (LR) (2) Neural network (NN) regression using Multilayer Perceptron (MLP) and (3) Neural network regression using Radial Basis Function (RBF). Six companies with largest data size were chosen. The problem domain was divided into two by considering different time lagged window sizes for the time series prediction. The Correlation Coefficient and RMSE measures were used to assess the performance of the three models.

The findings of this work are:

1. The linearity analysis showed some interesting results, especially company number 5 has highly nonlinear data. This led to the choice of MLP technique for prediction of EPS.
2. The experimental results indicate that LR and MLP models outperform the RBF models, except for the high nonlinear data.
3. MLP obtained best performance for high non-linear data.
4. The performance of Correlation Coefficient is consistent with the performance of RMSE for the three models.

5. The performance of the three models for Problem 1 (where the time lag window is smaller) is better than their performance for Problem 2 (where the time lag window is wider).
6. The chosen methods performed well yielding better accuracy score. This means that the techniques proposed could significantly help the investors in decision making by recommending the stocks that are probable to perform well. The ML techniques employed indicate not very huge differences in the performance measures. However, since not many studies are dedicated to prediction of EPS, this is state of the art performance for this problem employing machine learning techniques.

8.3 Contributions and Achievements

The main contributions of this PhD research are:

- A comprehensive literature review of 179 research publications was undertaken to survey the data mining and machine learning techniques applied to the finance industry from the year 2010 to 2015.The review indicated that stock prediction and credit rating have received most attention of researchers, compared to loan default prediction, money laundering and financial time series forecasting. Due to the dynamics, uncertainty and variety of data, nonlinear mapping techniques have been deeply studied than linear techniques. Also, it has been proved that hybrid methods are more accurate in prediction, closely followed by neural network technique.
- This survey could provide a clue of applications of DM techniques for finance industry, and a summary of methodologies for researchers in this area. Especially, it could provide a good vision of DM Techniques in computational finance for beginners who wish to work in the field of computational finance.
- Detailed data analysis along with visual presentation was conducted on three benchmark public credit datasets and one private dataset, which is a first in the literature of the benchmarked public datasets. The findings of the public datasets could be useful for future researchers to better understand the correlations among variables in these datasets. The findings of private dataset could potentially be used by the dataset providers to better adapt the

procedure for estimating the EPS and Sales of companies. These results could also be used to carry out further research in prediction of EPS

- The aim of chapter 5 was to apply a few classical ML techniques on the benchmarked datasets. Even if these datasets are very popular in the research community, not many studies have applied classical machine learning techniques to all of these datasets, especially the Taiwan credit data which is the largest among all three datasets. Hence the analysis of this dataset in chapter 4 and consequent application of ML techniques to particularly this dataset is the novelty of this work.
- Experiments were conducted to: get an insight into the response of the datasets to varying the values of hyperparameters C and Gamma on the Gaussian kernel SVM accuracy; to determine best execution times for the chosen machine learning techniques.
- Design of a novel procedure for solving the problem of 'Curse of Dimensionality' in credit scoring datasets. Development of a new algorithm called IGDFS for selecting the features from datasets which yields highest accuracy by incorporating the top ranked features indicated by Information Gain in the initial phase of feature Selection.
- Achieving the optimal solutions in most of the cases: The IGDFS finds the optimal feature subset from the datasets. This could mean that if these reduced feature subsets are used in future credit scoring process, it may result in less computation power and time, thus improving the credit scoring process.
- In this thesis, based on detailed results and analyses, we have presented a number of arguments as to why this work is an important contribution to the credit scoring community. The purpose of this section is to gather these arguments, at the end of the thesis, in order to provide the reader with a clear and precise sense of how this work contributes to credit scoring research.
- A credit scoring model is just one of the factors used in evaluating a credit application. Assessment by a credit expert remains the decisive factor in the evaluation of a loan. A new method based on information gain directed feature selection using genetic algorithm wrapper techniques applied to credit scoring problem. Advanced parameter tuning methods were adopted.

8.4 Research Limitations

During this research, the researcher successfully fulfilled the research aim and objectives. This PhD work has some limitations, but these limitations can present opportunities and recommendations for future studies.

- One of the major problems for applying ML algorithms in credit risk prediction is the unavailability and scarcity of credit data. The web enabled online commerce, which meant that lots of credit related data could be collected by corporations by employing the software for tracking web browsing. But, most financial institutions do not share their data with other organisations since such data are subject to security and privacy restrictions.
- Since there are no appropriate multiple-comparison procedures available for credit scoring, this work could not be validated rigorously.
- Another question is of the organisational acceptance of advanced classifiers (or the lack thereof) by finance industry. Such research work definitely needs validation in real-time environment.
- In chapter 5, the aim was to apply a few classical ML techniques on the benchmarked dataset. Even if these datasets are very popular in the research community, not many single studies exist where classical machine learning techniques are applied to all of these datasets, especially the Taiwan credit data which is the largest among all three public datasets used in this work. Comparison of the work carried out here against other research work could not be done in detail.

8.5 Future work

Several existing directions exist for future research.

- The findings of the extensive literature review in the domain of computational finance suggest that more structured reviews could be adopted e.g. in different sub-areas of the main five categories identified in the literature review.
- One area which deserves future study is determination of the cost of computation for machine learning classifiers in the area of computational

finance. e.g., in general, we expect the IGDFS technique will need more computations than the standard baseline and GAW methods. Experiments could be executed on powerful machines with advanced parallel computing capability. An interesting topic would be to assess the computing time needed to construct the different models and their computational complexity.

- Finance industry is a best use case for application of machine learning techniques. All the pieces of work from this study need to be executed with large real-time datasets from industry. Unfortunately, this work could not be validated in real-time settings.
- One of the datasets, the German credit dataset is an imbalanced dataset with 70% individuals being classified as ‘good credit’. More work could be done to balance the datasets and then apply the ML techniques and results could be compared.
- The private dataset contains data for 44000 companies. This study has utilised data for six companies. Further work could be done on other companies’ data. This could help draw conclusions about the whole dataset.
- The experiments were carried out in desktop environment. Better resources, especially hardware could be employed for the experiments. The aim in this study was to demonstrate usage and development of machine learning techniques for the problems in finance domain which could provide a direction for future studies.

9 REFERENCES

[1] R. Seydel, R. Seydel, Tools for Computational Finance, Springer-Verlag, Berlin/Heidelberg, 2006. doi:10.1007/3-540-27926-1.

[2] M.J. Miranda, P.L. Fackler, Applied computational economics and finance, MIT press, 2004.

[3] C.B. Frey, M.A. Osborne, The future of employment: How susceptible are jobs to computerisation?, Technol. Forecast. Soc. Chang. 114 (2017) 254–280. doi:10.1016/j.techfore.2016.08.019.

[4] E. Tsang, S. Martinez-Jaramillo, Computational finance, IEEE Comput. Intell. Soc. Newsl. 3 (2004).

[5] Wei-Yang Lin, Ya-Han Hu, Chih-Fong Tsai, Machine Learning in Financial Crisis Prediction: A Survey, IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev. 42 (2012) 421–436. doi:10.1109/TSMCC.2011.2170420.

[6] C. Wendel, M. Harvey, Credit Scoring: Best Practices and Approaches, Commer. Lend. Rev. 18 (2003).

[7] T. Diana, Credit risk analysis and credit scoring--now and in the future, Bus. Credit. 107 (2005) 12–16.

[8] M. Banasiak, G.K.-B.C.-N. YORK-, undefined 2000, Predictive collection score technology, Elibrary.ru. (n.d.).

[9] B. Twala, Multiple classifier application to credit risk assessment, Expert Syst. Appl. 37 (2010) 3326–3336. doi:10.1016/j.eswa.2009.10.018.

[10] G. Fernandez, Data mining using SAS applications, Chapman & Hall/CRC, 2003.

[11] C. Dima, M. Hebert, A. Stentz, Enabling Learning From Large Datasets: Applying Active Learning to Mobile Robotics, (n.d.).

[12] O. Wu, H. Zuo, M. Zhu, W. Hu, J. Gao, H. Wang, Rank Aggregation based Text Feature Selection, (n.d.).

[13] FactSet, Financial Research | Investment Analytics Tools — FactSet Research Systems, (2017). https://www.factset.com/ (accessed November 22, 2017).

[14] A.E. Khandani, A.J. Kim, A.W. Lo, Consumer credit-risk models via machine-learning algorithms, J. Bank. Financ. 34 (2010) 2767–2787.

[15] V.L. Miguéis, D.F. Benoit, D. Van den Poel, Enhanced decision support in credit scoring using Bayesian binary quantile regression, J. Oper. Res. Soc. 64 (2013) 1374–1383. doi:10.1057/jors.2012.116.

[16] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, J. Oper. Res. Soc. 54 (2003) 627–635. doi:10.1057/palgrave.jors.2601545.

[17] L. Thomas, J. Crook, D. Edelman, Credit scoring and its applications, SIAM, 2017.

[18] D. Martens, T. Van Gestel, M. De Backer, R. Haesen, J. Vanthienen, B. Baesens, Credit rating prediction using Ant Colony Optimization, J. Oper. Res. Soc. 61 (2010) 561–573. doi:10.1057/jors.2008.164.

[19] L.C. Thomas, D.B. Edelman, J.N. Crook, Credit scoring and its applications, Society for Industrial and Applied Mathematics, 2002.

[20] V.-S. Ha, H.-N. Nguyen, FRFE: Fast Recursive Feature Elimination for Credit Scoring, in: 2016: pp. 133–142. doi:10.1007/978-3-319-46909-6_13.

[21] J. Han, M. Kamber, J. Pei, Data mining: concepts and techniques, Elsevier, 2011.

[22] S. Sumathi, S.N. Sivanandam, Introduction to data mining and its applications, Springer, 2006.

[23] M.A. Aziz, H.A. Dar, Predicting Corporate Financial Distress: Whither do We Stand?, Dep. Econ. Loughbrgh. Univ. (2004).

[24] N. V Rao, G. Atmanathan, M. Shankar, S. Ramesh, Analysis of bankruptcy prediction models and their effectiveness: An Indian perspective, Gt. Lakes Her. 7 (2013).

[25] R. Anderson, The Credit Scoring Toolkit: Theory and Practice for Retail Credit Risk Management and Decision Automation: Theory and Practice for Retail Credit Risk Management and Decision Automation, Oxford University Press, 2007.

[26] Complexsearch, What is a Good Credit Score: 2016 Range, Credit Score Scale & Chart [Complete Guide], (2016). http://www.complexsearch.com/what-is-a-good-credit-score/.

[27] A.B. Hens, M.K. Tiwari, Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method, Expert Syst. Appl. 39 (2012) 6774–6781. doi:10.1016/j.eswa.2011.12.057.

[28] G.G. Sundarkumar, V. Ravi, A novel hybrid undersampling method for mining unbalanced datasets in banking and insurance, Eng. Appl. Artif. Intell. 37 (2015) 368–377.

[29] F.-L. Chen, F.-C. Li, Combination of feature selection approaches with SVM in credit scoring, Expert Syst. Appl. 37 (2010) 4902–4909. doi:10.1016/j.eswa.2009.12.025.

[30] M.-D. Cubiles-De-La-Vega, A. Blanco-Oliver, R. Pino-Mejías, J. Lara-Rubio, Improving the management of microfinance institutions by using credit scoring models based on Statistical Learning techniques, Expert Syst. Appl. 40 (2013) 6910–6917. doi:10.1016/j.eswa.2013.06.031.

[31] J. He, Y. Zhang, Y. Shi, G. Huang, Domain-Driven Classification Based on Multiple Criteria and Multiple Constraint-Level Programming for Intelligent

Credit Scoring, Knowl. Data Eng. IEEE Trans. 22 (2010) 826–838. doi:10.1109/TKDE.2010.43.

[32] K.J. Kim, H. Ahn, A corporate credit rating model using multi-class support vector machines with an ordinal pairwise partitioning approach, Comput. Oper. Res. 39 (2012) 1800–1811. doi:10.1016/j.cor.2011.06.023.

[33] S.C. Huang, Using Gaussian process based kernel classifiers for credit rating forecasting, Expert Syst. Appl. 38 (2011) 8607–8611. doi:10.1016/j.eswa.2011.01.064.

[34] S. Li, I.W. Tsang, N.S. Chaudhari, Relevance vector machine based infinite decision agent ensemble learning for credit risk analysis, Expert Syst. Appl. 39 (2012) 4947–4953. doi:10.1016/j.eswa.2011.10.022.

[35] P. Hájek, V. Olej, Credit rating modelling by kernel-based approaches with supervised and semi-supervised learning, Neural Comput. Appl. 20 (2011) 761–773.

[36] X. Zhou, W. Jiang, Y. Shi, Y. Tian, Credit risk evaluation with kernel-based affine subspace nearest points learning method, Expert Syst. Appl. 38 (2011) 4272. doi:http://dx.doi.org/10.1016/j.eswa.2010.09.095".

[37] F.N. Koutanaei, H. Sajedi, M. Khanbabaei, A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring, J. Retail. Consum. Serv. 27 (2015) 11–23. doi:10.1016/j.jretconser.2015.07.003.

[38] S. Jones, D. Johnstone, R. Wilson, An empirical evaluation of the performance of binary classifiers in the prediction of credit ratings changes, J. Bank. Financ. 56 (2015) 72–85.

[39] F.L. Chen, F.C. Li, Combination of feature selection approaches with SVM in credit scoring, Expert Syst. Appl. 37 (2010) 4902–4909. doi:10.1016/j.eswa.2009.12.025.

[40] G. Wang, J. Ma, A hybrid ensemble approach for enterprise credit risk

assessment based on Support Vector Machine, Expert Syst. Appl. 39 (2012) 5325–5331.

[41] S. Bhattacharyya, S. Jha, K. Tharakunnel, J.C. Westland, Data mining for credit card fraud: A comparative study, Decis. Support Syst. 50 (2011) 602. doi:http://dx.doi.org/10.1016/j.dss.2010.08.008".

[42] P. Danenas, G. Garsva, S. Gudas, Credit risk evaluation model development using support vector based classifiers, Procedia Comput. Sci. 4 (2011) 1699–1707.

[43] L. Yu, X. Yao, S. Wang, K.K. Lai, Credit risk evaluation using a weighted least squares SVM classifier with design of experiment for parameter selection, Expert Syst. Appl. 38 (2011) 15392–15399. doi:10.1016/j.eswa.2011.06.023.

[44] J.-H. Trustorff, P.M. Konrad, J. Leker, Credit risk prediction using support vector machines, Rev. Quant. Financ. Account. 36 (2011) 565–581.

[45] P. Danenas, G. Garsva, Selection of support vector machines based classifiers for credit risk domain, Expert Syst. Appl. 42 (2015) 3194–3204.

[46] L. Zhou, K.K. Lai, L. Yu, Least squares support vector machines ensemble models for credit scoring, Expert Syst. Appl. 37 (2010) 127–133. doi:10.1016/j.eswa.2009.05.024.

[47] A. Ghodselahi, A hybrid support vector machine ensemble model for credit scoring, Int. J. Comput. Appl. 17 (2011) 1–5.

[48] Y.B. Wah, I.R. Ibrahim, Using data mining predictive models to classify credit card applicants, in: Proc.- 6th Intl.Conference Adv. Inf. Manag. Serv. IMS2010, with ICMIA2010 - 2nd Int. Conf. Data Min. Intell. Inf. Technol. Appl., 2010: pp. 394–398.

[49] S. Oreski, D. Oreski, G. Oreski, Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment, Expert Syst. Appl. 39 (2012) 12605.

doi:http://dx.doi.org/10.1016/j.eswa.2012.05.023".

[50] S.H. Ha, R. Krishnan, Predicting repayment of the credit card debt, Comput. Oper. Res. 39 (2012) 765–773. doi:10.1016/j.cor.2010.10.032.

[51] N.-C. Hsieh, L.-P. Hung, A data driven ensemble classifier for credit scoring analysis, Expert Syst. Appl. 37 (2010) 534. doi:http://dx.doi.org/10.1016/j.eswa.2009.05.059".

[52] S.C. Chen, M.Y. Huang, Constructing credit auditing and control & management model with data mining technique, Expert Syst. Appl. 38 (2011) 5359–5365. doi:10.1016/j.eswa.2010.10.020.

[53] A. Marcano-Cedeno, A. Marin-De-La-Barcena, J. Jimenez-Trillo, J.A. Pinuela, D. Andina, Artificial metaplasticity neural network applied to credit scoring, Int. J. Neural Syst. 21 (2011) 311–317.

[54] G. Falavigna, Financial ratings with scarce information: A neural network approach, Expert Syst. Appl. 39 (2012) 1784. doi:http://dx.doi.org/10.1016/j.eswa.2011.08.074".

[55] A. Blanco, R. Pino-MejÃ-as, J. Lara, S. Rayo, Credit scoring models for the microfinance industry using neural networks: Evidence from Peru, Expert Syst. Appl. 40 (2013) 356. doi:http://dx.doi.org/10.1016/j.eswa.2012.07.051".

[56] A. AghaeiRad, B. Ribeiro, Credit Prediction Using Transfer of Learning via Self-Organizing Maps to Neural Networks, in: Eng. Appl. Neural Networks, Springer, 2015: pp. 358–365.

[57] R. Mileris, V. Boguslauskas, Data Reduction Influence on the Accuracy of Credit Risk Estimation Models, Eng. Econ. 66 (2015).

[58] B.W. Yap, S.H. Ong, N.H.M. Husain, Using data mining to improve assessment of credit worthiness via credit scoring models, Expert Syst. Appl. 38 (2011) 13274–13283. doi:10.1016/j.eswa.2011.04.147.

[59] G. Wang, J. Ma, L. Huang, K. Xu, Two credit scoring models based on dual strategy ensemble trees, Knowledge-Based Syst. 26 (2012) 61–68. doi:10.1016/j.knosys.2011.06.020.

[60] I. Brown, C. Mues, An experimental comparison of classification algorithms for imbalanced credit scoring data sets, Expert Syst. Appl. 39 (2012) 3446–3453. doi:10.1016/j.eswa.2011.09.033.

[61] A.C. Bahnsen, D. Aouada, B. Ottersten, Example-dependent cost-sensitive decision trees, Expert Syst. Appl. 42 (2015) 6609–6619.

[62] D. Zakrzewska, On integrating unsupervised and supervised classification for credit risk evaluation, Inf. Technol. Control. 36 (2015).

[63] R. Florez-Lopez, J.M. Ramon-Jeronimo, Enhancing accuracy and interpretability of ensemble strategies in credit risk assessment. A correlated-adjusted decision forest proposal, Expert Syst. Appl. 42 (2015) 5737–5753.

[64] C.-F. Tsai, M.-L. Chen, Credit rating by hybrid machine learning techniques, Appl. Soft Comput. 10 (2010) 374–380.

[65] Y. Ping, L. Yongheng, Neighborhood rough set and SVM based hybrid credit scoring classifier, Expert Syst. Appl. 38 (2011) 11300–11304.

[66] A. Capotorti, E. Barbanera, Credit scoring analysis using a fuzzy probabilistic rough set model, Comput. Stat. Data Anal. 56 (2012) 981. doi:http://dx.doi.org/10.1016/j.csda.2011.06.036".

[67] C.-L. Chuang, S.-T. Huang, A hybrid neural network approach for credit scoring, Expert Syst. 28 (2011) 185–196. doi:10.1111/j.1468-0394.2010.00565.x.

[68] C.-C. Yeh, F. Lin, C.-Y. Hsu, A hybrid KMV model, random forests and rough set theory approach for credit rating, Knowledge-Based Syst. 33 (2012) 166–172.

[69] K.-Y. Shen, G.-H. Tzeng, A decision rule-based soft computing model for supporting financial performance improvement of the banking industry, Soft Comput. 19 (2015) 859–874.

[70] B. Kovalerchuk, E. Vityaev, Data mining in finance: advances in relational and hybrid methods, Springer Science & Business Media, 2000.

[71] C.K. Leong, Credit risk scoring with bayesian network models, Comput. Econ. (2015) 1–24.

[72] J.K. Bae, J. Kim, A Personal Credit Rating Prediction Model Using Data Mining in Smart Ubiquitous Environments, Int. J. Distrib. Sens. Networks. 2015 (2015).

[73] H. Ahn, K. Lee, K. Kim, Global Optimization of Support Vector Machines Using Genetic Algorithms for Bankruptcy Prediction, in: Springer, Berlin, Heidelberg, 2006: pp. 420–429. doi:10.1007/11893295_47.

[74] S. Aktan, Application of machine learning algorithms for business failure prediction, Invest. Manag. Financ. Innov. 8 (2011) 52–65.

[75] T.-H. Chen, C.-W. Chen, Application of data mining to the spatial heterogeneity of foreclosed mortgages, Expert Syst. Appl. 37 (2010) 993–997.

[76] M. V Jagannatha Reddy, B. Kavitha, Neural Networks for Prediction of Loan Default Using Attribute Relevance Analysis, in: Signal Acquis. Process. 2010. ICSAP ’10. Int. Conf., 2010: pp. 274–277. doi:10.1109/ICSAP.2010.10.

[77] Y. Zhang, A Novel FNN Algorithm and Its Application in FCC Evaluation Based on Kirkpatrick Model, in: Knowl. Discov. Data Mining, 2010. WKDD ’10. Third Int. Conf., 2010: pp. 447–450. doi:10.1109/WKDD.2010.33.

[78] B. V Srinivasan, N. Gnanasambandam, S. Zhao, R. Minhas, Domain-Specific Adaptation of a Partial Least Squares Regression Model for Loan Defaults Prediction, in: Data Min. Work. (ICDMW), 2011 IEEE 11th Int.

Conf., 2011: pp. 474–479. doi:10.1109/ICDMW.2011.69.

[79] L. Zhou, H. Wang, Loan default prediction on large imbalanced data using random forests, TELKOMNIKA Indones. J. Electr. Eng. 10 (2012) 1519–1525.

[80] Y. Jin, Y. Zhu, A Data-Driven Approach to Predict Default Risk of Loan for Online Peer-to-Peer (P2P) Lending, in: Commun. Syst. Netw. Technol. (CSNT), 2015 Fifth Int. Conf., 2015: pp. 609–613.

[81] R.S. Oetama, Enhancing Decision Tree Performance in Credit Risk Classification and Prediction, Ultim. J. Tek. Inform. 7 (2015) 50.

[82] R.K. Amin, Y. Sibaroni, others, Implementation of decision tree using C4. 5 algorithm in decision making of loan application by debtor (Case study: Bank pasar of Yogyakarta Special Region), in: Inf. Commun. Technol. (ICoICT), 2015 3rd Int. Conf., 2015: pp. 75–80.

[83] J.A. Sanz, D. Bernardo, F. Herrera, H. Bustince, H. Hagras, A compact evolutionary interval-valued fuzzy rule-based classification system for the modeling and prediction of real-world financial applications with imbalanced data, Fuzzy Syst. IEEE Trans. 23 (2015) 973–990.

[84] R. Geng, I. Bose, X. Chen, Prediction of financial distress: An empirical study of listed Chinese companies using data mining, Eur. J. Oper. Res. 241 (2015) 236–247.

[85] M. Malekipirbazari, V. Aksakalli, Risk assessment in social lending via random forests, Expert Syst. Appl. 42 (2015) 4621–4631.

[86] S. Sathyadevan, R.R. Nair, Comparative Analysis of Decision Tree Algorithms: ID3, C4. 5 and Random Forest, in: Comput. Intell. Data Mining-Volume 1, Springer, 2015: pp. 549–562.

[87] S. Sadatrasoul, M. Gholamian, K. Shahanaghi, Combination of feature selection and optimized fuzzy apriori rules: The case of credit scoring, Int. Arab J. Inf. Technol. 12 (2015) 138–145.

[88] M. Islam, M. Habib, others, A data mining approach to predict prospective business sectors for lending in retail banking using decision tree, arXiv Prepr. arXiv1504.02018. (2015).

[89] J. Cao, H. Lu, W. Wang, J. Wang, A loan default discrimination model using cost-sensitive support vector machine improved by PSO, Inf. Technol. Manag. 14 (2013) 193–204. doi:10.1007/s10799-013-0161-1.

[90] H. Zhao, A.P. Sinha, G. Bansal, An extended tuning method for cost-sensitive regression and forecasting, Decis. Support Syst. 51 (2011) 372–383.

[91] T. Duong, V. Tran, Q. Ho, A Proposed Credit Scoring Model for Loan Default Arobability: a Vietnamese bank case, in: Int. Conf. Qual. Quant. Econ. Res. (QQE). Proc., 2015: p. 52.

[92] H. Wang, Q. Xu, L. Zhou, Large Unbalanced Credit Scoring Using Lasso-Logistic Regression Ensemble, PLoS One. 10 (2015) e0117844.

[93] H. Lee, N. Gnanasambandam, R. Minhas, S. Zhao, Dynamic Loan Service Monitoring Using Segmented Hidden Markov Models, in: Data Min. Work. (ICDMW), 2011 IEEE 11th Int. Conf., 2011: pp. 749–754. doi:10.1109/ICDMW.2011.71.

[94] D.K. Chandra, V. Ravi, P. Ravisankar, Support vector machine and wavelet neural network hybrid: application to bankruptcy prediction in banks, Int. J. Data Mining, Model. Manag. 2 (2010) 1–21.

[95] A.S. Aribowo, N.H. Cahyana, Feasibility study for banking loan using association rule mining classifier, Int. J. Adv. Intell. Informatics. 1 (2015) 41–47.

[96] F.J.L. Iturriaga, I.P. Sanz, Bankruptcy visualization and prediction using neural networks: A study of US commercial banks, Expert Syst. Appl. 42 (2015) 2857–2869.

[97] R. Sarno, R.D. Dewandono, T. Ahmad, M.F. Naufal, F. Sinaga, Hybrid

association rule learning and process mining for fraud detection, IAENG Int. J. Comput. Sci. 42 (2015) 59–72.

[98] Y. Lu, N. Zeng, X. Liu, S. Yi, A new hybrid algorithm for bankruptcy prediction using switching particle swarm optimization and support vector machines, Discret. Dyn. Nat. Soc. 501 (2015) 294930.

[99] R. Gerritsen, Assessing loan risks: a data mining case study, IT Prof. 1 (1999) 16–21.

[100] F. Typologies, T. Reports, Money Laundering and Terrorist Financing Trends in FINTRAC Cases Disclosed between 2007 and 2011 FINTRAC Typologies and Trends Reports – April 2012, (2012).

[101] J. Heggestuen, The US Sees More Money Lost To Credit Card Fraud Than The Rest Of The World Combined, (2014).

[102] M. Kantardzic, Data mining: concepts, models, methods, and algorithms, John Wiley & Sons, 2011.

[103] A. Sudjianto, S. Nair, M. Yuan, A. Zhang, D. Kern, F. Cela-Díaz, Statistical Methods for Fighting Financial Crimes, Technometrics. 52 (2010) 5–19. doi:10.1198/TECH.2010.07032.

[104] M. Krambia-Kapardis, C. Christodoulou, M. Agathocleous, Neural networks: the panacea in fraud detection?, Manag. Audit. J. 25 (2010) 659–678. doi:10.1108/02686901011061342.

[105] N.A. Le Khac, M. Kechadi, Application of Data Mining for Anti-money Laundering Detection: A Case Study, in: Data Min. Work. (ICDMW), 2010 IEEE Int. Conf., 2010: pp. 577–584. doi:10.1109/ICDMW.2010.66.

[106] R. Liu, X. Qian, S. Mao, S. Zhu, Research on anti-money laundering based on core decision tree algorithm, in: Control Decis. Conf. (CCDC), 2011 Chinese, 2011: pp. 4322–4325. doi:10.1109/CCDC.2011.5968986.

[107] R. Dreżewski, J. Sepielak, W. Filipkowski, System supporting money

laundering detection, Digit. Investig. 9 (2012) 8–21. doi:http://dx.doi.org/10.1016/j.diin.2012.04.003.

[108] F. Cai, N.-A. Le-Khac, M.-T. Kechadi, Clustering approaches for financial data analysis: a survey, in: Proc. Int. Conf. Data Min., 2012: p. 1.

[109] K.K. Tangod, G.H. Kulkarni, Detection of Financial Statement Fraud using Data Mining Technique and Performance Analysis, Int. J. Adv. Res. Comput. Commun. Eng. 4 (2015) 549–555.

[110] C. Alexandre, J. Balsa, Client Profiling for an Anti-Money Laundering System, arXiv Prepr. arXiv1510.00878. (2015).

[111] P. Ravisankar, V. Ravi, G.R. Rao, I. Bose, Detection of financial statement fraud and feature selection using data mining techniques, Decis. Support Syst. 50 (2011) 491–500.

[112] J. Perols, Financial statement fraud detection: An analysis of statistical and machine learning algorithms, Audit. A J. Pract. Theory. 30 (2011) 19–50.

[113] W. Zhou, G. Kapoor, Detecting evolutionary financial statement fraud, Decis. Support Syst. 50 (2011) 570–575.

[114] W. Wei, J. Li, L. Cao, Y. Ou, J. Chen, Effective detection of sophisticated online banking fraud on extremely imbalanced data, World Wide Web. 16 (2013) 449–475.

[115] H. Qin, D. Dou, Y. Fang, Financial Forecasting with Gompertz Multiple Kernel Learning, in: Data Min. (ICDM), 2010 IEEE 10th Int. Conf., 2010: pp. 983–988. doi:10.1109/ICDM.2010.68.

[116] A.S. Koyuncugil, N. Ozgulbas, Financial early warning system model and data mining application for risk detection, Expert Syst. Appl. 39 (2012) 6238–6253. doi:10.1016/j.eswa.2011.12.021.

[117] J. Sun, K.-Y. He, H. Li, SFFS-PC-NN optimized by genetic algorithm for dynamic prediction of financial distress with longitudinal data streams,

Knowledge-Based Syst. 24 (2011) 1013–1023.

[118] K. Choi, G. Kim, Y. Suh, Classification model for detecting and managing credit loan fraud based on individual-level utility concept, ACM SIGMIS Database. 44 (2013) 49–67.

[119] H. Li, M.-L. Wong, Financial fraud detection by using Grammar-based multi-objective genetic programming with ensemble learning, in: Evol. Comput. (CEC), 2015 IEEE Congr., 2015: pp. 1113–1120.

[120] P.J.G. Lisboa, A. Vellido, B. Edisbury, Business Applications of Neural Networks: The State-of-the-Art of Real-World Applications, World scientific, 2000.

[121] S. Kumar, S. Managi, A. Matsuda, Stock prices of clean energy firms, oil and carbon markets: A vector autoregressive analysis, Energy Econ. 34 (2012) 215–226.

[122] H. Zhao, Dynamic relationship between exchange rate and stock price: Evidence from China, Res. Int. Bus. Financ. 24 (2010) 103–112.

[123] A.A. Adebiyi, C.K. Ayo, M.O. Adebiyi, S.O. Otokiti, Stock Price Prediction using Neural Network with Hybridized Market Indicators, J. Emerg. Trends Comput. Inf. Sci. 3 (2012) 1–9.

[124] M.M. Mostafa, Forecasting stock exchange movements using neural networks: Empirical evidence from Kuwait, Expert Syst. Appl. 37 (2010) 6302–6309.

[125] L.A. Laboissiere, R.A.S. Fernandes, G.G. Lage, Maximum and minimum stock price forecasting of Brazilian power distribution companies based on artificial neural networks, Appl. Soft Comput. 35 (2015) 66–74.

[126] J. Wang, J. Wang, Forecasting stock market indexes using principle component analysis and stochastic time effective neural networks, Neurocomputing. 156 (2015) 68–78.

[127] A.B. Kock, T. Teräsvirta, Forecasting macroeconomic variables using neural network models and three automated model selection techniques, Econom. Rev. 4938 (2015) 1–27. doi:10.1080/07474938.2015.1035163.

[128] T.D. Chaudhuri, I. Ghosh, Forecasting Volatility in Indian Stock Market using Artificial Neural Network with Multiple Inputs and Outputs, Int. J. Comput. Appl. 120 (2015).

[129] Z. Chengzhao, P. Heiping, Z. Ke, Comparison of Back Propagation Neural Networks and EMD-Based Neural Networks in Forecasting the Three Major Asian Stock Markets, J. Appl. Sci. 15 (2015) 90.

[130] A. Esfahanipour, W. Aghamiri, Adapted Neuro-Fuzzy Inference System on indirect approach TSK fuzzy rule base for stock market analysis, Expert Syst. Appl. 37 (2010) 4742–4748. doi:10.1016/j.eswa.2009.11.020.

[131] C.-F. Liu, C.-Y. Yeh, S.-J. Lee, Application of type-2 neuro-fuzzy modeling in stock price prediction, Appl. Soft Comput. 12 (2012) 1348–1358.

[132] R. Hafezi, J. Shahrabi, E. Hadavandi, A bat-neural network multi-agent system (BNNMAS) for stock price prediction: Case study of DAX stock price, Appl. Soft Comput. 29 (2015) 196–210.

[133] B. Sun, H. Guo, H.R. Karimi, Y. Ge, S. Xiong, Prediction of stock index futures prices based on fuzzy sets and multivariate fuzzy time series, Neurocomputing. 151 (2015) 1528–1536.

[134] B.B. Nair, S.G. Sai, A.N. Naveen, A. Lakshmi, G.S. Venkatesh, V.P. Mohandas, A GA-artificial neural network hybrid system for financial time series forecasting, in: Inf. Technol. Mob. Commun., Springer, 2011: pp. 499–506.

[135] R. de A. Araújo, T.A.E. Ferreira, A morphological-rank-linear evolutionary method for stock market prediction, Inf. Sci. (Ny). 237 (2013) 3–17.

[136] C.-H. Cheng, T.-L. Chen, L.-Y. Wei, A hybrid model based on rough sets theory and genetic algorithms for stock price forecasting, Inf. Sci. (Ny). 180

(2010) 1610–1629.

[137] C.-F. Huang, A hybrid stock selection model using genetic algorithms and support vector regression, Appl. Soft Comput. 12 (2012) 807. doi:http://dx.doi.org/10.1016/j.asoc.2011.10.009".

[138] B.B. Nair, V.P. Mohandas, N.R. Sakthivel, A genetic algorithm optimized decision tree-SVM based stock market trend prediction system, Int. J. Comput. Sci. Eng. 2 (2010) 2981–2988.

[139] W. Qiu, X. Liu, L. Wang, Forecasting shanghai composite index based on fuzzy time series and improved C-fuzzy decision trees, Expert Syst. Appl. 39 (2012) 7680–7689.

[140] P. Hajek, Forecasting Stock Market Trend using Prototype Generation Classifiers, WSEAS Trans. Syst. 11 (2012) 671–680.

[141] M.-Y. Chen, B.-T. Chen, A hybrid fuzzy time series model based on granular computing for stock price forecasting, Inf. Sci. (Ny). 294 (2015) 227–241.

[142] C.-F. Huang, B.R. Chang, D.-W. Cheng, C.-H. Chang, Feature selection and parameter optimization of a fuzzy-based stock selection model using genetic algorithms, Int. J. Fuzzy Syst. 14 (2012) 65–75.

[143] C.-M. Hsu, A hybrid procedure for stock price prediction by integrating self-organizing map and genetic programming, Expert Syst. Appl. 38 (2011) 14026–14036.

[144] Y.-W. Chang Chien, Y.-L. Chen, Mining associative classification rules with stock trading data–A GA-based method, Knowledge-Based Syst. 23 (2010) 605–614.

[145] A. Sheta, S.E.M. Ahmed, H. Faris, Evolving stock market prediction models using multi-gene symbolic regression genetic programming, Artif. Intell. Mach. Learn. AIML. 15 (2015) 11–20.

[146] M. Ballings, D. den Poel, N. Hespeels, R. Gryp, Evaluating multiple classifiers for stock price direction prediction, Expert Syst. Appl. 42 (2015) 7046–7056.

[147] C.-J. Lu, Sales forecasting of computer products based on variable selection scheme and support vector regression, Neurocomputing. 128 (2014) 491–499.

[148] M.H. Zarandi, M. Zarinbal, N. Ghanbari, I.B. Turksen, A new fuzzy functions model tuned by hybridizing imperialist competitive algorithm and simulated annealing. Application: Stock price prediction, Inf. Sci. (Ny). 222 (2013) 213–228.

[149] J. De Andrés, P. Lorca, F.J. de Cos Juez, F. Sánchez-Lasheras, Bankruptcy forecasting: A hybrid approach using Fuzzy c-means clustering and Multivariate Adaptive Regression Splines (MARS), Expert Syst. Appl. 38 (2011) 1866–1875.

[150] A. Martin, V. Gayathri, G. Saranya, P. Gayathri, P. Venkatesan, A hybrid model for bankruptcy Prediction using genetic Algorithm, fuzzy c-means and mars, arXiv Prepr. arXiv1103.2110. (2011).

[151] Z. Ding, Application of support vector machine regression in stock price forecasting, in: Business, Econ. Financ. Sci. Manag., Springer, 2012: pp. 359–365.

[152] Q. Wen, Z. Yang, Y. Song, P. Jia, Automatic stock decision support system based on box theory and SVM algorithm, Expert Syst. Appl. 37 (2010) 1015–1022.

[153] L. Luo, X. Chen, Integrating piecewise linear representation and weighted support vector machine for stock trading signal prediction, Appl. Soft Comput. 13 (2013) 806–816.

[154] A. Kazem, E. Sharifi, F.K. Hussain, M. Saberi, O.K. Hussain, Support vector regression with chaos-based firefly algorithm for stock market price

forecasting, Appl. Soft Comput. 13 (2013) 947–958.

[155] C.-Y. Yeh, C.-W. Huang, S.-J. Lee, A multiple-kernel support vector regression approach for stock market price forecasting, Expert Syst. Appl. 38 (2011) 2177–2186.

[156] C.L. Dunis, R. Rosillo, D. de la Fuente, R. Pino, Forecasting IBEX-35 moves using support vector machines, Neural Comput. Appl. 23 (2013) 229–236.

[157] K. Zbikowski, Using volume weighted support vector machines with walk forward testing and feature selection for the purpose of creating stock trading strategy, Expert Syst. Appl. 42 (2015) 1797–1805.

[158] I. Marković, M. Stojanović, M. Božić, J. Stanković, Stock market trend prediction based on the LS-SVM model update algorithm, in: ICT Innov. 2014, Springer, 2015: pp. 105–114.

[159] J.-J. Wang, J.-Z. Wang, Z.-G. Zhang, S.-P. Guo, Stock index forecasting based on a hybrid model, Omega. 40 (2012) 758–766.

[160] C.-F. Huang, C.-H. Chang, B.R. Chang, D.-W. Cheng, A study of a hybrid evolutionary fuzzy model for stock selection, in: Fuzzy Syst. (FUZZ), 2011 IEEE Int. Conf., 2011: pp. 210–217. doi:10.1109/FUZZY.2011.6007661.

[161] S.-H. Liao, S.-Y. Chou, Data mining investigation of co-movements on the Taiwan and China stock markets for future investment portfolio, Expert Syst. Appl. 40 (2013) 1542. doi:http://dx.doi.org/10.1016/j.eswa.2012.08.075".

[162] R. Chitrakar, H. Chuanhe, Anomaly detection using Support Vector Machine classification with k-Medoids clustering, in: Internet (AH-ICI), 2012 Third Asian Himalayas Int. Conf., IEEE, 2012: pp. 1–5.

[163] J. Patel, S. Shah, P. Thakkar, K. Kotecha, Predicting stock market index using fusion of machine learning techniques, Expert Syst. Appl. 42 (2015) 2162–2172.

[164] T. Xiong, Y. Bao, Z. Hu, Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting, Knowledge-Based Syst. 55 (2014) 87–100.

[165] C.-F. Tsai, Y.-C. Hsiao, Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches, Decis. Support Syst. 50 (2010) 258–269.

[166] S.W.K. Chan, J. Franklin, A text-based decision support system for financial sequence prediction, Decis. Support Syst. 52 (2011) 189–198.

[167] M. Hagenau, M. Liebmann, M. Hedwig, D. Neumann, Automated news reading: Stock price prediction based on financial news using context-specific features, in: Syst. Sci. (HICSS), 2012 45th Hawaii Int. Conf., IEEE, 2012: pp. 1040–1049.

[168] L.-C. Yu, J.-L. Wu, P.-C. Chang, H.-S. Chu, Using a contextual entropy model to expand emotion words and their intensity for the sentiment classification of stock market news, Knowledge-Based Syst. 41 (2013) 89–97.

[169] C.-F. Tsai, Z.-Y. Quan, Stock Prediction by Searching for Similarities in Candlestick Charts, ACM Trans. Manag. Inf. Syst. 5 (2014) 9.

[170] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market, J. Comput. Sci. 2 (2011) 1–8.

[171] S. Deng, T. Mitsubuchi, K. Shioda, T. Shimada, A. Sakurai, Combining technical analysis with sentiment analysis for stock price prediction, in: Dependable, Auton. Secur. Comput. (DASC), 2011 IEEE Ninth Int. Conf., IEEE, 2011: pp. 800–807.

[172] P. Hajek, V. Olej, R. Myskova, Forecasting Stock Prices using Sentiment Information in Annual Reports-A Neural Network and Support Vector Regression Approach, WSEAS Trans. Syst. (in Press. 2013). (2013).

[173] G. REINERT, Time Series, (2002).

http://www.stats.ox.ac.uk/~reinert/time/notesht10short.pdf.

[174] D. Shasha, Time series in finance: the array database approach, ACM SIGM0D, Abril. (1999).

[175] H. Jiang, W. He, Grey relational grade in local support vector regression for financial time series prediction, Expert Syst. Appl. 39 (2012) 2256. doi:http://dx.doi.org/10.1016/j.eswa.2011.07.100".

[176] H. Sugimura, K. Matsumoto, Classification system for time series data based on feature pattern extraction, in: Syst. Man, Cybern. (SMC), 2011 IEEE Int. Conf., 2011: pp. 1340–1345. doi:10.1109/ICSMC.2011.6083844.

[177] G. Zhiqiang, W. Huaiqing, L. Quan, Financial time series forecasting using LPP and SVM optimized by PSO, Soft Comput. 17 (2013) 805–818.

[178] T. Xiong, Y. Bao, Z. Hu, R. Chiong, Forecasting interval time series using a fully complex-valued RBF neural network with DPSO and PSO algorithms, Inf. Sci. (Ny). 305 (2015) 77–92.

[179] C. Wong, M. Versace, CARTMAP: a neural network method for automated feature selection in financial time series forecasting, Neural Comput. Appl. 21 (2012) 969–977.

[180] W. Yan, Toward Automatic Time-Series Forecasting Using Neural Networks, Neural Networks Learn. Syst. IEEE Trans. 23 (2012) 1028–1039. doi:10.1109/TNNLS.2012.2198074.

[181] M. Khashei, M. Bijari, Fuzzy artificial neural network (p, d, q) model for incomplete financial time series forecasting, J. Intell. Fuzzy Syst. 26 (2014) 831–845.

[182] H. Niu, J. Wang, Financial time series prediction by a random data-time effective RBF neural network, Soft Comput. 18 (2014) 497–508.

[183] S. Saigal, D. Mehrotra, Performance comparison of time series data using predictive data mining techniques, Adv. Inf. Min. 4 (2012) 57–66.

[184] S. Deng, K. Yoshiyama, T. Mitsubuchi, A. Sakurai, Hybrid method of multiple kernel learning and genetic algorithm for forecasting short-term foreign exchange rates, Comput. Econ. 45 (2015) 49–89.

[185] D. Huang, X. Wang, J. Fang, S. Liu, R. Dou, A hybrid model based on neural networks for financial time series, in: Artif. Intell. (MICAI), 2013 12th Mex. Int. Conf., 2013: pp. 97–102.

[186] Q. Cai, D. Zhang, W. Zheng, S.C.H. Leung, A new fuzzy time series forecasting model combined with ant colony optimization and auto-regression, Knowledge-Based Syst. 74 (2015) 61–68.

[187] W. Wang, X. Liu, Fuzzy forecasting based on automatic clustering and axiomatic fuzzy set classification, Inf. Sci. (Ny). 294 (2015) 78–94.

[188] A. Dutta, PREDICTION OF STOCK PERFORMANCE IN INDIAN STOCK MARKET USING LOGISTIC REGRESSION, Int. J. Bus. Inf. 7 (2015).

[189] Y. Bai, B. Wan, X. Zong, W. Rao, A Modified ARIMA Model Based on Extreme Value for Time Series Modelling, (2015).

[190] S. Bhattacharyya, P. Dutta, S. Chakraborty, Hybrid Soft Computing Approaches, (2016).

[191] Intel, Deep Learning Delivers Advanced Analytics for Financial Services Firms, (2017). https://www.intel.com/content/dam/www/public/us/en/documents/solution-briefs/deep-learning-delivers-advanced-analytics-solution-brief.pdf (accessed April 19, 2018).

[192] A. Chowdhry, How Artificial Intelligence Is Going To Affect The Financial Industry In 2018, (2018). https://www.forbes.com/sites/amitchowdhry/2018/02/26/clinc-artificial-intelligence/#35c1dce481f9 (accessed April 19, 2018).

[193] M. Lichman, {UCI} Machine Learning Repository, (2013).

[194] D.T. Larose, Discovering Knowledge in Data: An Introduction to Data Mining, John Wiley & Sons, 2014. doi:10.1002/9781118874059.

[195] AAII: The American Association of Individual Investors, Earnings Estimates and Their Impact on Stock Prices, (2018). http://www.aaii.com/investing-basics/article/earnings-estimates-and-their-impact-on-stock-prices (accessed April 28, 2018).

[196] M. Hurley, J. Adebayo, CREDIT SCORING IN THE ERA OF BIG DATA, Yale J. Law Technol. 18 (2016).

[197] A. Ben-David, E. Frank, Accuracy of machine learning models versus hand crafted" expert systems – A credit scoring case study, (n.d.). doi:10.1016/j.eswa.2008.06.071.

[198] The Financial Hacker, Build Better Strategies! Part 4: Machine Learning – The Financial Hacker, (2016). http://www.financial-hacker.com/build-better-strategies-part-4-machine-learning/ (accessed October 29, 2017).

[199] K. Veeramachaneni, Why You're Not Getting Value from Your Data Science, Harv. Bus. Rev. (2016). https://hbr.org/2016/12/why-youre-not-getting-value-from-your-data-science (accessed October 29, 2017).

[200] D.J. Hand, Classifier Technology and the Illusion of Progress, Stat. Sci. 21 (2006) 1–14. doi:10.1214/088342306000000060.

[201] T. Bellotti, J. Crook, Support vector machines for credit scoring and discovery of significant features, Expert Syst. Appl. (2009).

[202] J. Brill, The importance of credit scoring models in improving cash flow and collections, Bus. Credit. 100 (1998) 16–17.

[203] C.-L. Huang, M.-C. Chen, C.-J. Wang, Credit scoring with a data mining approach based on support vector machines, Expert Syst. Appl. 33 (2007) 847–856. doi:10.1016/j.eswa.2006.07.007.

[204] S.R. Gunn, Support Vector Machines for Classification and Regression,

1998.

[205] MathWorks, Choose Classifier Options - MATLAB & Simulink - MathWorks United Kingdom, (2017). https://uk.mathworks.com/help/stats/choose-a-classifier.html (accessed November 12, 2017).

[206] B.E. Boser, I.M. Guyon, V.N. Vapnik, A Training Algorithm for Optimal Margin Classifiers, in: Proc. Fifth Annu. Work. Comput. Learn. Theory, ACM, 1992: pp. 144–152.

[207] B. Schölkopf, K. Tsuda, J.-P. Vert, Kernel methods in computational biology, MIT Press, 2004.

[208] C.-W. Hsu, C.-C. Chang, C.-J. Lin, A Practical Guide to Support Vector Classification, (2003).

[209] A. Ben-Hur, J. Weston, A User's Guide to Support Vector Machines, (n.d.).

[210] H. He, A. Tiwari, J. Mehnen, T. Watson, C. Maple, Y. Jin, B. Gabrys, Incremental information gain analysis of input attribute impact on RBF-kernel SVM spam detection, in: 2016 IEEE Congr. Evol. Comput. CEC 2016, IEEE, 2016: pp. 1022–1029. doi:10.1109/CEC.2016.7743901.

[211] E. Fix, J.J.L. Hodges, Discriminatory Analysis - Nonparametric Discrimination: Consistency Properties, (1951).

[212] G.A. Vouros, T. Panayiotopoulos, Methods and applications of artificial intelligence : third Hellenic conference on AI, SETN 2004 : Samos, Greece, May 5-8 2004 : proceedings, Springer, 2004.

[213] D. Huang, D. Wunsch, D. Levine, K. Jo, Aspects of Artificial Intelligence: Fourth International Conference on Intelligent Computing, ICIC 2008 Shanghai, China, September 15-18, 2008, Proceedings, 2008.

[214] Scikit-learn Developers, RBF SVM parameters — scikit-learn 0.19.1 documentation, (2017).

http://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html (accessed December 2, 2017).

[215] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artif. Intell. 97 (1997) 245–271. doi:10.1016/S0004-3702(97)00063-5.

[216] D. Koller, M. Sahami, Toward optimal feature selection, Stanford InfoLab. (1996).

[217] A. Janecek, W. Gansterer, M. Demel, G. Ecker, On the relationship between feature selection and classification accuracy, New Challenges Featur. Sel. Data Min. Knowl. Discov. (2008) 90–105.

[218] T. Mitchell, Machine learning. 1997, Burr Ridge, McGraw Hill. (1997).

[219] C. Cortes, V. Vapnik, Soft margin classifier, US Pat. 5,640,492. (1997).

[220] M. Mitchell, An introduction to genetic algorithms, MIT press, 1998.

[221] J. Kennedy, Particle swarm optimization, Encycl. Mach. Learn. (2011) 760–766.

[222] R. Kohavi, G.H. John, The Wrapper Approach, in: Featur. Extr. Constr. Sel., Springer US, Boston, MA, 1998: pp. 33–50. doi:10.1007/978-1-4615-5725-8_3.

[223] L. Jourdan, C. Dhaenens, E. Talbi, A genetic algorithm for feature selection in data-mining for genetics, Proc. 4th Metaheuristics Int. Conf. (2001) 29–34.

[224] O. Soufan, D. Kleftogiannis, P. Kalnis, V.B. Bajic, DWFS: A Wrapper Feature Selection Tool Based on a Parallel Genetic Algorithm, PLoS One. 10 (2015) e0117988. doi:10.1371/journal.pone.0117988.

[225] H. Li, J. Sun, Predicting business failure using support vector machines with straightforward wrapper: A re-sampling study, Expert Syst. Appl. 38 (2011) 12747–12756.

[226] S. Maldonado, J. Pérez, C. Bravo, Cost-based feature selection for Support Vector Machines: An application in credit scoring, Eur. J. Oper. Res. 261 (2017) 656–665. doi:10.1016/j.ejor.2017.02.037.

[227] L. Zhuo, J. Zheng, X. Li, F. Wang, B. Ai, J. Qian, A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine, in: Geoinformatics 2008 Jt. Conf. GIS Built Environ. Classif. Remote Sens. Images, International Society for Optics and Photonics, 2008: p. 71471J–71471J. doi:10.1117/12.813256.

[228] N. Verbiest, J. Derrac, C. Cornelis, S. García, F. Herrera, Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis, Appl. Soft Comput. 38 (2016) 10–22. doi:10.1016/j.asoc.2015.09.006.

[229] R.C. Anirudha, R. Kannan, N. Patil, Genetic algorithm based wrapper feature selection on hybrid prediction model for analysis of high dimensional data, in: 2014 9th Int. Conf. Ind. Inf. Syst., IEEE, 2014: pp. 1–6. doi:10.1109/ICIINFS.2014.7036522.

[230] H. Sabzevari, M. Soleymani, E. Noorbakhsh, A comparison between statistical and data mining methods for credit scoring in case of limited available data, Proc. Credit Scoring Conf. UK. (2007) 1–8.

[231] M. Khanbabaei, M. Alborzi, The use of genetic algorithm, clustering and feature selection techniques in construction of decision tree models for credit scoring, Int. J. Manag. Inf. Technol. 5 (2013) 13–31.

[232] Y. Liu, M. Schumann, Data mining feature selection for credit scoring models, J. Oper. Res. Soc. 56 (2005) 1099–1108. doi:10.1057/palgrave.jors.2601976.

[233] S. Sadatrasoul, M. Gholamian, Combination of feature selection and optimized fuzzy apriori rules: the case of credit scoring., Int. Arab J. Inf. Technol. 12 (2015) 138–145.

[234] R. Allami, A. Stranieri, A genetic algorithm-neural network wrapper approach for bundle branch block detection, Comput. Cardiol. Conf. (2016) 461–464.

[235] A. Özçift, A. Gülten, Genetic algorithm wrapped Bayesian network feature selection applied to differential diagnosis of erythemato-squamous diseases, Digit. Signal Process. 23 (2013) 230–237. doi:10.1016/j.dsp.2012.07.008.

[236] J. Huang, Y. Cai, X. Xu, A hybrid genetic algorithm for feature selection wrapper based on mutual information, Pattern Recognit. Lett. 28 (2007) 1825–1844. doi:10.1016/j.patrec.2007.05.011.

[237] A. Daamouche, F. Melgani, N. Alajlan, Swarm optimization of structuring elements for VHR image classification, IEEE Geosci. Remote Sens. Lett. 10 (2013) 1334–1338.

[238] S. Lin, K. Ying, S. Chen, Z. Lee, Particle swarm optimization for parameter determination and feature selection of support vector machines, Expert Syst. Appl. 35 (2008) 1817–1824.

[239] H. Liu, H. Motoda, Feature selection for knowledge discovery and data mining (Vol. 454), Springer Science & Business Media, 2012.

[240] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[241] P. Somol, B. Baesens, P. Pudil, Filter-versus wrapper-based feature selection for credit scoring, Int. J. Intell. Syst. 20 (2005) 985–999.

[242] H. Frohlich, O. Chapelle, Feature selection for support vector machines by means of genetic algorithm, in: 15th Ieee Int. Conf. Tools with Artif. Intell., 2003: pp. 142–148.

[243] D. Liang, C.F. Tsai, H.T. Wu, The effect of feature selection on financial distress prediction, Knowledge-Based Syst. 73 (2014) 289–297. doi:10.1016/j.knosys.2014.10.010.

[244] B. Waad, B.M. Ghazi, L. Mohamed, A three-stage feature selection using quadratic programming for credit scoring, Appl. Artif. Intell. 27 (2013) 721–742.

[245] F. Li, The hybrid credit scoring strategies based on knn classifier, in: Fuzzy Syst. Knowl. Discov. 2009. FSKD'09. Sixth Int. Conf., IEEE, 2009: pp. 330–334.

[246] A.Z. Hamadani, A. Shalbafzadeh, T. Rezvan, A. Moghadam, An Integrated Genetic-Based Model of Naive Bayes Networks for Credit Scoring, Int. J. Artif. Intell. Appl. 4 (2013) 85–103. doi:10.5121/ijaia.2013.4107.

[247] J. Wang, A.-R. Hedar, S. Wang, J. Ma, Rough set and scatter search metaheuristic based feature selection for credit scoring, Expert Syst. Appl. 39 (2012) 6123–6128. doi:10.1016/j.eswa.2011.11.011.

[248] P. Hajek, K. Michalak, Feature selection in corporate credit rating prediction, (2013). doi:10.1016/j.knosys.2013.07.008.

[249] S. Oreski, G. Oreski, Genetic algorithm-based heuristic for feature selection in credit risk assessment, Expert Syst. Appl. 41 (2014) 2052–2064. doi:10.1016/j.eswa.2013.09.004.

[250] V. Kozeny, Genetic algorithms for credit scoring: Alternative fitness function performance comparison, (2014). doi:10.1016/j.eswa.2014.11.028.

[251] H. Van Sang, N. Nam, N. Nhan, A novel credit scoring prediction model based on Feature Selection approach and parallel random forest, Indian J. Sci. (2016).

[252] Fico, Using Segmented Models for Better Decisions (white paper), (2014).

[253] L. Li, W. Jiang, X. Li, K.L. Moser, Z. Guo, L. Du, Q. Wang, E.J. Topol, Q. Wang, S. Rao, A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset, Genomics. 85 (2005) 16–23. doi:10.1016/j.ygeno.2004.09.007.

[254] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, M. Ares, D. Haussler, Knowledge-based analysis of microarray gene expression data by using support vector machines, Knowledge-Based Anal. Microarray Gene Expr. Data by Using Support Vector Mach. 97 (2000) 262–267.

[255] N. Cristianini, J. Shawe-Taylor, An introduction to support vector machines, Cambridge University Press, Cambridge, UK, 2000.

[256] S. Maldonado, R. Weber, A wrapper method for feature selection using Support Vector Machines, Inf. Sci. (Ny). 179 (2009) 2208–2217. doi:10.1016/j.ins.2009.02.014.

[257] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artif. Intell. 97 (1998) 273–324. doi:http://dx.doi.org/10.1016/S0004-3702(97)00043-X.

[258] L. Wang, Y. Jin, Fuzzy Systems and Knowledge Discovery: Second International Conference, FSKD 2005, Changsha, China, August 27-29, 2005, Proceedings, Springer Science & Business Media, 2005.

[259] M. Lankhorst, Genetic algorithms in data analysis, [University Library Groningen][Host], 1996.

[260] C.-C. Chang, C.-J. Lin, LIBSVM: A Library for Support Vector Machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 27.

[261] C.-W. Hsu, C.-C. Chang, C.-J. Lin, others, A practical guide to support vector classification, (2003).

[262] A. Statnikov, C. Aliferis, I. Tsamardinos, D. Hardin, A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis, Bioinformatics. 21 (2004) 631–643.

[263] F. Pedregosa, G. Varoquaux, A. Gramfort, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[264] I. Kucukkoc, A. Karaoglan, R. Yaman, Using response surface design to

determine the optimal parameters of genetic algorithm and a case study, Int. J. Prod. Res. 51 (2013) 5039–5054.

[265] M. Srinivas, L.M. Patnaik, Genetic Algorithms: A Survey, Computer (Long. Beach. Calif). 27 (1994) 17–26. doi:10.1109/2.294849.

[266] J. Chen, H. Huang, S. Tian, Y. Qu, Feature selection for text classification with Naive Bayes, Expert Syst. Appl. 36 (2009) 5432–5435. doi:10.1016/j.eswa.2008.06.054.

[267] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, V. Vapnik, B. BioInformaticscom, Feature Selection for SVMs, Adv. Neural Inf. Process. Syst. (2001) 668–674.

[268] A.B. Hens, M.K. Tiwari, Computational time reduction for credit scoring : An integrated approach based on support vector machine and stratified sampling method, Expert Syst. with Appl. 39 (2012) 6774–6781. doi:10.1016/j.eswa.2011.12.057.

[269] T. San Ong, Y.N. Yichen, B.H. Teh, Can High Price Earnings Ratio Act As An Indicator Of The Coming Bear Market In The Malaysia?, Int. J. Bus. Soc. Sci. 1 (2010).

[270] P.S. Sajja, R. Akerkar, Intelligent technologies for Web applications, CRC Press, 2012.

[271] H. Ince, T.B. Trafalis, Short term forecasting with support vector machines and application to stock price prediction, Int. J. Gen. Syst. 37 (2008) 677–687.

[272] E. Guresen, G. Kayakutlu, T.U. Daim, Using artificial neural network models in stock market index prediction, Expert Syst. Appl. 38 (2011) 10389–10397.

[273] W. Shen, X. Guo, C. Wu, D. Wu, Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm, Knowledge-Based Syst. 24 (2011) 378–385.

[274] K. Chen, H.-Y. Lin, T. Huang, The prediction of Taiwan 10-year government bond yield, WSEAS Trans. Syst. 8 (2009) 1051–1060.

[275] X.-B. Yan, Z. Wang, S.-H. Yu, Y.-J. Li, Time series forecasting with RBF neural network, in: Mach. Learn. Cybern. 2005. Proc. 2005 Int. Conf., IEEE, 2005: pp. 4680–4683.

[276] Z. Guo, H. Wang, J. Yang, D.J. Miller, A Stock Market Forecasting Model Combining Two-Directional Two-Dimensional Principal Component Analysis and Radial Basis Function Neural Network, (2015).

[277] G. Sermpinis, K. Theofilatos, A. Karathanasopoulos, E.F. Georgopoulos, C. Dunis, Forecasting foreign exchange rates with adaptive neural networks using radial-basis functions and particle swarm optimization, Eur. J. Oper. Res. 225 (2013) 528–540.

[278] Y. Kara, M.A. Boyacioglu, Ö.K. Baykan, Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange, Expert Syst. Appl. 38 (2011) 5311–5319.

[279] V. Pacelli, V. Bevilacqua, M. Azzollini, An artificial neural network model to forecast exchange rates, J. Intell. Learn. Syst. Appl. 3 (2011) 57.

[280] P. Du Jardin, E. Séverin, Predicting corporate bankruptcy using a self-organizing map: An empirical study to improve the forecasting horizon of a financial failure model, Decis. Support Syst. 51 (2011) 701–711.

[281] P. Ravisankar, V. Ravi, Financial distress prediction in banks using Group Method of Data Handling neural network, counter propagation neural network and fuzzy ARTMAP, Knowledge-Based Syst. 23 (2010) 823–831.

[282] V. Esichaikul, P. Srithongnopawong, Using relative movement to support ANN-based stock forecasting in Thai stock market, Int. J. Electron. Financ. 4 (2010) 84–98.

[283] J.-Z. Wang, J.-J. Wang, Z.-G. Zhang, S.-P. Guo, Forecasting stock indices

with back propagation neural network, Expert Syst. Appl. 38 (2011) 14346–14355.

[284] F.A. De Oliveira, L.E. Zárate, M. de Azevedo Reis, C.N. Nobre, The use of artificial neural networks in the analysis and prediction of stock prices, in: Syst. Man, Cybern. (SMC), 2011 IEEE Int. Conf., IEEE, 2011: pp. 2151–2155.

[285] K.S. Vaisla, A.K. Bhatt, An analysis of the performance of artificial neural network technique for stock market forecasting, Int. J. Comput. Sci. Eng. 2 (2010) 2104–2109.

[286] A. Bagheri, H.M. Peyhani, M. Akbari, Financial forecasting using ANFIS networks with quantum-behaved particle swarm optimization, Expert Syst. Appl. 41 (2014) 6235–6250.

[287] M.-Y. Chen, A hybrid ANFIS model for business failure prediction utilizing particle swarm optimization and subtractive clustering, Inf. Sci. (Ny). 220 (2013) 180–195.

[288] M.A. Boyacioglu, D. Avci, An adaptive network-based fuzzy inference system (ANFIS) for the prediction of stock market return: the case of the Istanbul stock exchange, Expert Syst. Appl. 37 (2010) 7908–7912.

[289] S.A.S. Olaniyi, K.S. Adewole, R.G. Jimoh, Stock trend prediction using regression analysis–a data mining approach, ARPN J. Syst. Softw. 1 (2011) 154–157.

[290] B.J. Blair, S.-H. Poon, S.J. Taylor, Forecasting S&P 100 volatility: the incremental information content of implied volatilities and high-frequency index returns, in: Handb. Quant. Financ. Risk Manag., Springer, 2010: pp. 1333–1344.

[291] S. Saigal, D. Mehrotra, Performance comparison of time series data using predictive data mining techniques, Adv. Inf. Min. 4 (2012) 57–66.

[292] C. Serrano-Cinca, B. GutiéRrez-Nieto, Partial least square discriminant

analysis for bankruptcy prediction, Decis. Support Syst. 54 (2013) 1245–1255.

[293] W.-T. Pan, A new fruit fly optimization algorithm: taking the financial distress model as an example, Knowledge-Based Syst. 26 (2012) 69–74.

[294] H. Öğüt, M.M. Doğanay, N.B. Ceylan, R. Aktaş, Prediction of bank financial strength ratings: The case of Turkey, Econ. Model. 29 (2012) 632–640.

[295] J.M. Patell, Corporate forecasts of earnings per share and stock price behavior: Empirical test, J. Account. Res. (1976) 246–276.

[296] W.-S. Chen, Y.-K. Du, Using neural networks and data mining techniques for the financial distress prediction model, Expert Syst. Appl. 36 (2009) 4075–4086.

[297] T.-S. Quah, DJIA stock selection assisted by neural network, Expert Syst. Appl. 35 (2008) 50–58.

[298] G. Khirbat, R. Gupta, S. Singh, Optimal Neural Network Architecture for Stock Market Forecasting, in: Commun. Syst. Netw. Technol. (CSNT), 2013 Int. Conf., IEEE, 2013: pp. 557–561.

[299] R.K. Lai, C.-Y. Fan, W.-H. Huang, P.-C. Chang, Evolving and clustering fuzzy decision tree for financial time series data forecasting, Expert Syst. Appl. 36 (2009) 3761–3773.

[300] N.H. Pan, M.L. Lee, C.W. Chang, Construction Financial Crisis Warning Model Using Data Mining, in: Adv. Mater. Res., Trans Tech Publ, 2011: pp. 684–688.

[301] S. Han, R.-C. Chen, Using svm with financial statement analysis for prediction of stocks, Commun. IIMA. 7 (2007) 63.

[302] X. Song, Y. Ding, J. Huang, Y. Ge, Feature selection for support vector machine in financial crisis prediction: a case study in China, Expert Syst. 27 (2010) 299–310.

[303] M. Timor, H. Dincer, S. Emir, Performance comparison of artificial neural network (ANN) and support vector machines (SVM) models for the stock selection problem: An application on the Istanbul Stock Exchange (ISE)-30 index in Turkey, (2012).

[304] X.Y. Qiu, On building predictive models with company annual reports, (2007).

[305] Y.-X. Jiang, H. Wang, Q.-F. Xie, Classification model of companies' financial performance based on integrated support vector machine, in: Manag. Sci. Eng. 2009. ICMSE 2009. Int. Conf., IEEE, 2009: pp. 1322–1328.

[306] J.T.S. Quah, W.D. Ng, Utilizing computational intelligence for DJIA stock selection, in: Neural Networks, 2007. IJCNN 2007. Int. Jt. Conf., IEEE, 2007: pp. 956–961.

[307] H.-B. Li, M.-L. Wong, Knowledge discovering in corporate securities fraud by using grammar based genetic programming, J. Comput. Commun. 2 (2014) 148.

[308] J. Arefin, R.M. Rahman, Testing different forms of efficiency for Dhaka Stock Exchange, Int. J. Financ. Serv. Manag. 5 (2011) 1–20.

[309] K. Rezaie, V.M. Dalfard, L. Hatami-Shirkouhi, S. Nazari-Shirkouhi, Efficiency appraisal and ranking of decision-making units using data envelopment analysis in fuzzy environment: a case study of Tehran stock exchange, Neural Comput. Appl. 23 (2013) 1–17.

[310] M.P. Rajakumar, V. Shanthi, Forecasting earnings per share for companies in it sector using Markov process model, J. Theor. Appl. Inf. Technol. 59 (2014) 332–341.

[311] WEKA, MultilayerPerceptron, (n.d.). http://weka.sourceforge.net/doc.dev/weka/classifiers/functions/MultilayerPerceptron.html (accessed April 29, 2018).