Department of Computer and Information Sciences

APPLYING BUSINESS ANALYTICS IN PRACTICE TO A BANK TELEMARKETING DATASET

OLUWASEUN ESTHER OLUWABUSOLA

(201582168)

SEPTEMBER, 2015.

DECLARATION

This dissertation is submitted in part fulfilment of the requirements for the degree of MSc of the University of Strathclyde.

I declare that this dissertation embodies the results of my own work and that it has been composed by myself. Following normal academic conventions, I have made due acknowledgement to the work of others.

I declare that I have sought, and received, ethics approval via the Departmental Ethics Committee as appropriate to my research.

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to provide copies of the dissertation, at cost, to those who may in the future request a copy of the dissertation for private study or research.

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to place a copy of the dissertation in a publicly available archive.

Yes [✓] No [ ]

I declare that the word count for this dissertation (excluding title page, declaration, abstract, acknowledgements, table of contents, list of illustrations, references and appendices is 18808 words.

I confirm that I wish this to be assessed as a Type 1 2 (3) 4 5

Signature:

Date:

ABSTRACT

The business analytics approach to doing business involves the use of an organization's transactional data to gain knowledge on how business processes can be improved through the use of data mining techniques which are aimed at identifying interesting patterns that can be adopted by an organization to make more data-driven decisions.

As part of the process of driving the relevance of Business Analytics and Data Mining approach in businesses, this project focused on applying several data mining techniques to help identify patterns on how data gathered from a Portuguese bank telemarketing activity can be used to determine the likelihood of customers subscribing to term deposits. The dataset used contains 20 input attributes regarding information about the bank telemarketing campaigns conducted by a Portuguese bank and a target variable was used to predict if a customer would be subscribing to a term deposit.

When dealing with real world dataset such as that being used in this research, there is usually the problem of class imbalance where the occurrence of one class is more predominant than the other. To prevent the underperformance of the learning algorithms due to the imbalance class problem in this dataset, the research focused on adopting the data preprocessing method, the feature selection method and the use of ensembles to ensure that the classification algorithms chosen reach their optimum predictive performance.

The data mining task was carried out by using the Weka Machine learning tool to analyse the dataset. During the course of this research, it was discovered that the process of combining the feature selection task with the use of the ensembles learning method was the best approach to improving the predictive capabilities of the learning algorithms used. Association rules were also discovered in relation to analysing what factors contribute to the most likely reasons why customers would subscribe to the bank's term deposit.

ACKNOWLEDGEMENT

I wold like to thank God for the grace to reach the successful completion of this project.

I would also like to thank my supervisor Dmitri Roussinov for his guidance and support from the beginning to the end of the project.

Finally, I would also like to thank my parents for their emotional and financial support throughout the course of my study.

Oluwaseun Esther Oluwabusola

TABLE Of CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

1. Introduction

This chapter provides an overview of what the research entails. The concept of Business Analytics and data mining was discussed by providing industry based examples where successes have been recorded by adopting the Business Analytics approach to doing business. The research questions, problem, context, objectives were discussed to help provide better insight into what the research would be focusing on.

1.1. Background

Over the years, more businesses have evolved quickly from making decisions based on mere intuition to making data-driven decisions. Organizations are consistently looking for how best to remain relevant in business by obtaining value from the data being gathered from their daily business activities. Most successful Organizations have realised the importance of harnessing the potentials in the data being archived.

The concept of data mining has emerged due to the amount of transactional data being recorded on a daily basis. The need for a data analysis tool that can provide organizations with the ability to automatically discover hidden information from large datasets and also the ability to convert such information to applicable knowledge has become of huge necessity (Han, Kamber and Pei, 2011). Data mining may be defined as the process of discovering interesting and meaningful patterns from data and deriving knowledge from the discovered patterns (Han, Kamber and Pei, 2011) which results in some form of advantage (Witten and Frank 2005).

Major industrial sectors such as healthcare, banking, retail, intelligence and telecommunications are currently taking advantage of the data mining tools and technique to improve their business processes (Eze, Adeoye and Ikemelu, 2014). For instance, in the banking sector, data mining is being used to develop models for risk analysis, Customer Relationship Management (CRM), direct marketing and prevention of credit card fraud. In the healthcare, data mining has gained popularity for prevention of medical insurance fraud and abuse (Sokol

et al., 2001), prediction of trends in behaviour and health condition of patients (Milovic and Milovic, 2012).

The knowledge of data mining and business analytics is needed for proper collection and analysis of data (Ledolter, 2013). The application of the knowledge derived from data mining and business analytics can help an organization to make strategic data-driven business decisions which would lead to an improvement in the organization's competitive advantage.

Business analytics involves the use of data to gain insight into an organization's business process (Respício, 2010). The analysis of such data helps to improve an organization's knowledge on how best to improve its business process and gain competitive advantage.

The process of applying business analytics as part of an organizational business practice has proven to help organizations improve revenue, save cost, time and efforts by focusing on vital business areas. Large organizations such as Capital one, T-Mobile, Netflix, eBay and Amazon are currently gaining huge benefits from the application of data mining and business analytics to core areas of their departments. More organizations are gradually embracing the business analytics approach to doing business. For instance, amazon uses different data gathered from previous buying behaviours of customers to predict what items they might be considering to buy. This is achieved through the use of data gathered from their previous order, shopping cart, items viewed and wish list. This information is used to make suggestions to customers on other items that are made available.

Business performance can be improved through the process of analysing the data gathered from an organization's business activities. The business intelligence field has provided organizations with better insights on how their business processes can be improved for further business growth.

Watson and Wixom (2007) describes business intelligence as comprising of two major stages; the input and output stage. The input stage refers to the data gathering and data warehousing stage while the output stage involves the utilization of the knowledge acquired through the analysis of the data gathered.

For any organization to remain relevant in today's competitive business world, a proper understanding and analysis of customer behaviour is key (Davenport and Harris, 2007). This helps to ensure that decisions made by an organization are profitable both to the organization and its customer.

1.2. Research Context

This research was aimed at applying the concept of business analytics to a bank telemarketing dataset. The dataset contains some information about the banking operations carried out by a Portuguese retail bank from 2008 to 2013. The data was made available by Moro, Laureano and Cortez (2014) through the popular University of California at Irvine (UCI) machine learning repository.

In the banking sector, transactions are usually being recorded on a daily basis and the ability to maintain a good relationship with customer is very important. Customers need to be made aware of new products and services being offered by the bank. This can be achieved through direct marketing or mass marketing (Ling and Li, 1998). In direct marketing, offers are being made to customers on a one-to-one basis**.** Mass marketing on the other hand is a form of marketing which is carried out with the aim of reaching out to a large population to create awareness about the products and services being offered by a company. These offers are usually generic and aren't particularly tailored to appeal to a particular individual (Thomas, 2007).

Direct marketing has been known to yield more positive results than mass marketing because it provides organizations with better opportunities to interact with both current and prospective customers (Elsalamony, 2013). Examples of direct marketing include telemarketing, direct mail, direct response Television etc. (Roberts and Berger, 1999).

Telemarketing is a form of direct marketing, which involves the use of telecommunication mediums to create awareness to new and existing customers about products and services offered by a company (Bencin, 1992). McCausland (2000) describes the ability to identify and

proper solutions to prospective customers as one of the means of ensuring the success of telemarketing.

The dataset contains information about direct marketing campaigns carried out by the Portuguese bank with the aim of getting customers to subscribe to a term deposit. Term deposit refers to a deposit held by a financial institution for a fixed period of time which is usually made known to the customer. During this period, the customer has no access to the deposit held by the bank. Requesting for the deposit before the agreed maturity period would attract some of penalty which is usually made known to the customer at the beginning of the agreement.

The dataset also contains record of both the inbound and outbound calls made by the bank. Outbound calls are usually made to customers during marketing campaigns, with the aim of having them subscribe to a term deposit (Moro, Laureano and Rita, 2014). The bank also took advantage of the inbound calls made by the customers. For inbound calls placed by customers to the bank, the customer is also being made aware of the offer to subscribe to the bank's term deposit.

The dataset consists of 41188 instances and 21 attributes. The 21st attribute represents the class which is used to predict if a client is going to subscribe to a term deposit. The class attribute consists of only two values represented with yes or no. The yes class represents the group of customers who subscribed to a term deposit while the No class refers to the group of customers who did not subscribe to a term deposit.

The ability to identify customers with a better like hood of subscribing to a term deposit is important because more targeted marketing campaigns would be aimed at potential customers in order to help reduce time and resources spent on such campaigns.

1.3. Statement of the Problem

Despite the increased awareness of the usefulness of data mining, several factors could lead to underutilization of the available data, part of which could be linked to presence of inconsistency in dataset resulting in overfitting or under fitting by the learning algorithms. In data mining, most often we have cases of classes being unevenly distributed.

The majority class refers to the class where the occurrence of a particular case is predominant while the minority class refers to the class where we have very few occurrence of a particular case. In this scenario, most algorithms tend to work more in favour of the majority class by having more accurate predictions while the minority class is usually being misclassified or assumed to be non-existent within the data sample. In the next chapter, we would be talking more about the minority and majority class.

1.4. Research Objectives

For this project, the focus was on working towards choosing the best method that produces a classifier with better predictive capability which can be used to determine if customers are going to subscribe to a term deposit or not. The tests conducted were carried out sequentially, analysing how the various changes made to the dataset affects the overall predictive accuracy of the models that were utilized.

Correlation analysis was carried out to help identify accurate rules that could be adopted by banks to identify customers with a high likelihood of subscribing to a term deposit. The identified rules could help to discover what group of people within a class are more likely to subscribe. This could be in relation to their age, marital status, gender, educational status etc. This also includes finding out if personal efforts put by banks into their campaign process determines how well customers would subscribe to this service. For instance, the number of times each customer was contacted via phone call, the result of previous campaign exercises carried out etc. All of these attributes were explored to discover closely related patterns that would be useful in improving marketing campaign which would lead to better services.

1.5. Research questions

How well does the data pre-processing method, the ensembles method and the feature selection method help to handle class samples?

How well does each technique independently handle class accuracy predictions?

Which of these three approaches tend to work better in improving the predictive accuracy of our models?

What difference would it make if these methods are combined together?

What factors contribute to the reasons why different algorithms perform better than the other?

1.6. Research Methods

For this research, the machine learning tool known as The Waikato Environment for Knowledge Analysis (Weka) was used to carry out analysis of the dataset. Weka consists of a collection of several machine learning algorithms used for data mining activities. It contains in-built tools that can be used for data pre-processing, classification, clustering, association rules, attribute selections and data visualization. This software was used to carry out all of the necessary tests, analysis and visualization. The analysis carried out was based on the 21 attributes made available in this dataset. A basic understanding of the specific domain being considered, a proper understanding of the dataset being analysed and the ability to manipulate the dataset is of great essence (Holmes, Donkin and Witten, 1994).

The conclusions arrived at were dependent on the results generated while carrying out different manipulation on the dataset using Weka. The process of evaluating the accuracy of the classifier for an imbalanced dataset cannot be entirely dependent on the number of correctly classified instances, other approaches such as the ROC value, precision and recall, cost-sensitive measurement must be used to ensure that the right conclusions are made (Chawla, 2005).

CHAPTER TWO

2. Literature Review

Large volumes of transactions are usually carried out in banks and recorded on a daily basis, utilising these data to improve business processes could pose a challenge. Data mining aims at providing organizations with the opportunity to analyse and interact with large volume of datasets to discover relevant patterns that can be used to improve business processes. Most data mining task involves making predictions based on two class value, predicting whether or not certain events would occur based on the analysis of different attributes (Ledolter, 2013).

In the business sector today, the ability to interpret and analyse data is of great essence in targeting customers that are most likely to invest in new products and services being offered. Customer retention is of great importance; the ability to make new customers and retain existing customers can be achieved through data mining by identifying patterns relating to customer needs based on previous customer behaviour (Chitra and Subashini, 2013).

The process of creating products and services which meets customer satisfaction is key in improving an organization's competitiveness. The ability to offer products and services which meets customer's needs and satisfaction requires a proper understanding and ability to identify their individual behavioural characteristics and individual preferences (Zatari, 2014). The business analytics approach to doing business is an emerging area of knowledge which would be required for businesses to thrive through in this new era. Business analysis involves several data analysis methods (Shmueli, Patel and Bruce, 2016) which can be used to identify patterns and relationships existing within an organisation's data. The knowledge derived from this analysis can be used to make informed and target-based decisions.

Most often, the word Business analytics, Business intelligence and data mining are usually used interchangeable whereas a certain level of distinction occurs in the theoretical and practical application of each term.

2.1. Differences Between Business Analytics, Data mining and Business Intelligence

Data mining is a term used to describe several data analysis techniques which can be used to identify interesting patterns within an organization's large data resource (Lee, 2013). The data to be mined can be gotten from data warehouses, databases, web and other information archives (Han, Pei and Kamber, 2011). The data mining task includes techniques such as association rule learning, classification, regression analysis and clustering.

Business analytics involves the use of the knowledge derived from the data mining results to make more informed business decisions. The next level of Business analytics is referred to as Business Intelligence. Business analytics now includes both Business Intelligence and data mining processes (Han, Pei and Kamber, 2011).

Business intelligence involves both data visualization and reporting to help understand reasons for past and current occurrence (Han, Pei and Kamber, 2011). It is a combination of operational data and analytical tools which would be used to represent complex business related information in an easily understandable state to the decision makers in an organization which could help increase its competitive advantage (Negash, S., 2004).

Due to the exponential increase in the volume of transactional data being recorded on a daily basis, the application of business analytics and data mining to several business divisions has gained more recognition (Lee, 2013). More businesses are gradually beginning to discover the potentials and significance of applying data mining for organizational business growth.

2.2. Real World Applications of Data Mining

Over the years, the Business Intelligence field has rapidly evolved and several development and innovations has occurred which has provided better insight into how data can effectively be managed and utilized to drive change and help organization's focus on improving their business needs and requirements to suit the evolving market and consumer trends. In 2010, IBM introduced a new version of its Business Analytics (BA) portfolio known as IBM Cognos 10 Business Intelligence which was focused on improving client's ability to utilize relevant data gathered to improve their decision making process through the use of adequate analysis and

reporting techniques tailored to meet both their recent and future business intelligence requirements (Rouse, 2010). Capital one has stood out amongst its other credit card lending companies based on its use of Business Analytics to study the behaviour of their customers (Davenport and Harris, 2007). Capital one makes specialised offers to customers based on results generated from the analysis of its individual customer spending patterns as opposed to just generic offers made to all customers.

In 1995, Tesco which is currently recognized as the largest retailer in the UK introduced its loyalty card scheme which enables them to keep tracks of customer's shopping patterns (Wright and Sparks, 1999) when a customer purchases an item. This provided them with the understanding of the type of products to suggest to what class and people and how previous buying behaviours could be used to determine future choices based on attributes relating to a product. This loyalty card scheme has been greatly attributed to the company's success which is believed to be the company's most significant competitive advantage (Rigby, 2006). The key to staying relevant in today's highly competitive business world is the use of Business Intelligence and Data mining techniques to drive business growth (Moro, Laureano and Cortez, 2011), there is a high level of focus on customer retention and service improvement (Witten and Frank, 2005).

2.3. Imbalance Class Problem

Data mining is associated with large volume of data ranging from thousands of transactional records to several millions of records which could consists of several thousands of attributes and data types. Large datasets usually occur more within sectors where several transactions are made on a regular basis. Examples of such sectors include telecommunications, transportation, retail, marketing, banking etc. (Leventhal, 2010). For such large transactions, the absence of errors and imbalance in the class prediction is almost impossible. Datasets collated from real world tasks are usually imbalanced due to the presence of a class being more predominant than the other. The class imbalance problem occurs due to the nature of most application domains and can't be totally controlled like noise and outliers which occur due to mechanical errors and deficiencies which occur from the data generation process (Sowah et al., 2016). This class

imbalance problem has led to the reduction of the generalization of the results generated from machine learning algorithms (Kim, 2007).

Class imbalance occurs when there is an underrepresentation of a particular class to be predicted. This occurs when we have more instances of a particular class in a training set which largely outnumbers other classes within the dataset. When a high level of imbalance occurs within a dataset there tends to be issues with creating an effective classification model which can be used to accurately make class predictions (Seiffert et al., 2010). In class imbalance, the class instances belong to either the majority or the minority class. The minority class refers to the class within the dataset which have very few instances represented while the majority class refers to the class whose instances outnumbers others with several occurrences. Data mining algorithms usually tend to favour the majority class due to the presence of more instances which leads to some instances of the minority class being incorrectly identified. A classifier which is biased towards the majority class could record a high accuracy for its overall performance but record a very poor performance for the minority instances (Wasikowski and Chen, 2010). In most typical examples, the minority class usually carry the highest cost of misclassification (Seiffert et al., 2010) whose impact cannot be neglected due to gravity of the misclassification of such instances. For instance, in the healthcare, we usually tend to have more negative instances of an ailment as opposed to positive instances when compared within a larger population. In the case of a rare medical condition, the process of incorrectly classifying a sick person as fit would have more severe effect as opposed to classification a healthy person as sick. For medical datasets with high risk patients, the cost of misclassifying the minority class (high risk patient) is higher than the cost of the misclassifying the majority class when dealing with most imbalanced datasets (Rahman and Davis, 2013). The effect of such misclassification could result in the death of such patient if not detected.

Several attempts have been made by different researches from various fields to deal with imbalanced datasets in domains such as fraudulent telephone calls (Fawcett & Provost, 1996), telecommunications management (Ezawa, Singh, & Norton,1996), Credit risk assessment

(Huang, Hung and Jiau, 2006), network intrusion detection (Cieslak, Chawla & Striegel, 2006) and detection of oil spills in satellite images (Kubat, Holte & Matwin, 1998).

2.3.1 Solving The Class Imbalance Problem

Several methods have been introduced by different researches to solve the imbalance class problem. These are techniques are usually broadly classified into the sampling method, algorithmic method and the feature selection method. The most commonly used method for solving class imbalance is the data sampling method.

2.3.1.1. Data Sampling Method

The data sampling method consists of two major categories. This is recognized as undersampling and oversampling of the dataset. Oversampling is the process of increasing the minority class instances while undersampling is the process of reducing the majority class instances. Seiffert et al. (2010) identified random resampling as the easiest approach for resampling a dataset. Random undersampling is the process of removing already existing instances of the majority class within the dataset while Random Oversampling is the process of duplicating the minority class instances. Both the Random oversampling and undersampling can be done until a certain desired level of balance is attained (Seiffert et al., 2010). Japkowicz (2000) also identified the use of the oversampling and the undersampling technique as an effective measure to address the class imbalance problem, thereby leading to more accurate predictions. Despite the relevance attached to achieving an adequate level of balance within the dataset, both data sampling techniques both have their advantages and disadvantages. The process of randomly removing some instances within the dataset could result in the loss of certain important instances. The disadvantage of Random oversampling includes overfitting of the training data and also additional computation task for large dataset due to the process of adding more instances to the minority class (Chawla, Japkowicz and Kotcz, 2004).

2.3.1.2. Algorithm Method

Several methods have been introduced to tackle the class imbalance problem using the algorithm based approach but the goal of each diverse means has been to work towards optimizing the performance of each classifier being used (Wasikowski and Chen, 2010). The algorithm based approach to dealing with class imbalance works on the algorithms itself as

opposed to the manipulation of the dataset done in the sampling methods. To address the class imbalance problem using this method requires a proper understanding of the classifier's learning algorithm that would be used and also the corresponding application domain being examined (Sun et al., 2007). A proper understanding of why the algorithm didn't perform well on the dataset while in its imbalanced state is also required (Sun et al., 2007). This solution works by adapting the classifier's learning algorithm to be bias towards the minority class so that better result can be achieved to reduce the effect of misclassification of the positive instances. Several algorithms can be combined together to obtain a better predictive accuracy. This can be achieved through the use of ensemble learning.

2.3.1.2.1. Ensemble Learning

Ensemble learning is the process of combining several machine learning algorithms by making optimum use of each algorithm's uniqueness to improve upon the predictive capability of the model. Ensemble methods may help to overcome the three major problems faced by individual classifiers which prevent them from making accurate predictions. These problems are identified as the statistical problem, the computational problem and the representational problem (Dietterich, 2002). A classifier which has the representational problem is said to have high bias. A classifier with high bias tends to have most of its classes incorrectly predicted. A classifier with the statistical problem is said to have high variance. This problem occurs when the learning algorithm searches through a space of hypotheses which is too large from the training dataset (Dietterich, 2002). In this case, most classifiers tend to overfit the model but usually have a high risk of incorrectly predicting the classes when used on a different training set. The computational problem occurs when there is no assurance that the learning algorithm being used would find the best hypothesis that would work best with the training data. An example of such algorithm includes decision tree and neural network (Dietterich, 2002). Syarif et al. (2012) identified bagging, boosting and stacking as the three commonly used ensemble classifier techniques.

Ensemble Classifier Techniques

i) Bagging: In The bagging technique, the original training set is broken down into several subset s and each subset can have several base learners which can be used to make predictions. An aggregate of the output of each classifier's prediction is used to make the final prediction for the entire dataset (Hido, Kashima and Takahashi, 2009). Bagging has been identified as one of the easiest but most effective ensemble technique for addressing unstable classification problems (Syarif et al., 2012). For bagging to be effective, Breiman (1996) noted that instability in the learner is a requirement due to the fact that for a variance reduction process to be initiated, the learning algorithm must already possess high variance. An unstable model such as decision tree algorithm would best be used rather than a stable model such as the logistic regression for bagging to be effective (Breiman, 1996.)

ii) Boosting: Boosting is an ensemble method for boosting weak classifiers. This is achieved through the process of utilizing the individual uniqueness of the weak classifiers and effectively combining them to construct a strong classifier (Syarif et al., 2012). The AdaBoost algorithm is one of the most widely used technique for combining and converting sets of weak classifiers into a strong classifier. Boosting schemes which are similar to the AdaBoost algorithm undergo series of iteration by using a sampling by replacement approach. The first round of sampling is done by placing equal level of probabilities for each sample. After the first round of sampling, each sample which was accurately predicted gets a lower probability value which reduces its chances of being selected in the next iteration while samples which are wrongly classified are allocated a higher probability value to increase its chances of being selected in the next iteration (Wasikowski and Chen, 2010)

iii) Stacking: This technique uses a different approach from bagging and boosting. Stacking can be divided into two levels. The first level is referred to as the base learner (level-0) and the second level is referred to as the stacking model learner (level-1) classifier (Syarif et al., 2012). The output generated from each of the base leaners is combined to form a new dataset. This new dataset is then used by the

stacking model learner to generate the final prediction. For instance, the output from three or four different classifiers can be combined together to act as an input for the stacking model learner which eventually creates the final output prediction for the dataset. The function of the stacking model learner is to find out how best to combine the outputs from the base learner to generate a more accurate result (Graczyk et al., 2010).

2.3.1.3. Feature Selection Method

The use of sampling techniques and algorithm based approach might not be sufficient in addressing class imbalance problem due to the challenge of high dimensionality in the dataset which usually accompanies an imbalanced dataset (Weiss and Provost, 2003). To address this problem, a feature selection approach would be needed to help in selecting a subset of features which would be helpful in utilizing the model to reach its optimum performance (Tiwari, 2014). The feature selection process is a key requirement when dealing with a dataset with high dimensionality. For dataset with high dimensionality, the use of filter is adopted to independently score each feature based on a rule (Longadge and Dongre, 2013).

2.3.2. Related Works on handling class imbalance

Chawla et al. (2002) addressed the class imbalance problem by using a combination of the method of under-sampling the majority class and oversampling the minority class. C4.5, Ripper and Naive Bayes classifier was used to carry out this test. They came to the conclusion that applying both Synthetic Minority Oversampling Technique SMOTE and under-sampling produces better results than just applying the oversampling technique to the dataset.

Chawla et al. (2003) carried out series of test to address class imbalance on four datasets. They proposed an approach which involves the integration of the SMOTE within the standard boosting procedure. The SMOTE algorithm was used to ensure that more accurate prediction is made for the minority class and then boosting was used to ensure that accuracy measurement is not chosen over the actual content of the entire dataset. SMOTE was introduced at each round of boosting to ensure that the learner adapts more to the minority class instances so that the True positive rates can be improved. The results generated from this approach showed that

the SMOTEBOOST algorithm was better in making more accurate predictions for the minority class than the use of a single classifier, AdaBoost, AdaCost, and the first SMOTE then Boost method.

A combination of Random Undersampling and Boost has been discovered to be more effective and faster than the use of the combination of SMOTE algorithm and Boosting. Seiffert et al. (2010) adopted a technique similar to that used by (Chawla, Lazarevic, Hall, and Bowyer, 2003). They developed a hybrid system which comprises of sampling and boosting. This algorithm was referred to as RUSBoost. Their approach involves random undersampling of the majority class and then introducing the boosting process. They compared RUSBoost to SMOTEBoost used by Chawla et al. (2003) and the RUSBoost technique was discovered to perform as well as or even better than SMOTEBoost whilst also introducing a simpler faster, and effective approach than SMOTEBoost. Both RUSBoost and SMOTEBoost are known to introduce the sampling techniques into the AdaBoost Algorithm (Seiffert et al., 2010). The experimental results generated by Galar et al. (2012) also shows that the positive synergy existing between RUS and ensemble techniques often leads to better results when handling imbalance dataset. The proper combination of bagging techniques and data pre-processing techniques also proves to be powerful when dealing with imbalance class.

2.4. Previous Works On the Bank Telemarketing Dataset

Previous work carried out by researchers on the bank telemarketing dataset only focused on making class predictions through the use of different classifying algorithms. None of the researcher who worked on this dataset adopted any approach to addressing the imbalanced nature of the dataset.

Abbas (2015) focused on the use of the Decision Tree Algorithm and the Rough Set Theory. He focused on identifying rules that would be relevant in identifying the probability of a customer subscribing to a term deposit. For the Rough Set Theory approach, only three attributes were used in making the class predictions. The Age, Balance and Duration attributes were the most

relevant attributes in analysing the bank marketing dataset. For the decision tree technique adopted by him, accuracy selection was based on the gain ratio generated for all the attributes.

Al-Shayea (2013) examined the use of neural networks in making class predictions for the bank marketing dataset. The result generated showed that neural network has the ability to learn patterns which can be used to effectively determine if customers would subscribe to a term deposit. Moro, Laureano and Cortez (2011) conducted their research on the dataset by using the rminer library. The decision tree, Support vector Machines (SVM) and Naïve Bayes classifiers were used. Call duration and month of contact were identified as the most important features used by the classifiers for making predictions.

CHAPTER 3

3. RESEARCH METHODOLOGY

This chapters provides a detailed study on the method adopted for carrying out the classification and association rule mining task using the Weka machine learning algorithm. It provides a detailed study on how the data-preprocessing steps were used to improve the quality of the bank telemarketing dataset. For the classification task, the sampling technique, the ensembles method and the feature selection method was used to determine the approach for making the class predictions. The association rule mining task was also conducted by using the ensemble based methods and the filter selection method to help discover if there are interesting rules that might be relevant for the bank marketing campaign.

3.1. Research Types

3.1.1. Quantitative vs Qualitative Research Method

The quantitative approach to research involves the collection, analysis and interpretation of results generated through the numerical and statistical analysis of data. The results generated in quantitative research are usually gotten through the use of statistical software (Johnson, and Onwuegbuzie, 2004) whose results are presented in a numerical form which can be compared and illustrated by using different visualisation methods such as graphs, tables, chart etc. The research findings from a quantitative research can be generalised when the data is based on random samples of sufficient size (Johnson and Onwuegbuzie, 2004).

The qualitative approach to research also involves the collection, analysis and interpretation of results generated from a data but more from a social science perspective. This involves the collection of open-ended data through the process of conducting interviews and surveys (Creswell, 2013.). The type of data usually collected, analysed and interpreted here includes factual, behavioural and opinion related data. The conclusions being made here may not be applicable in a different setting due to changes in the various responses of diverse participants (Johnson and Onwuegbuzie, 2004). Types of qualitative research include case study and grounded theory (Creswell, 2013).

3.2. Research Design

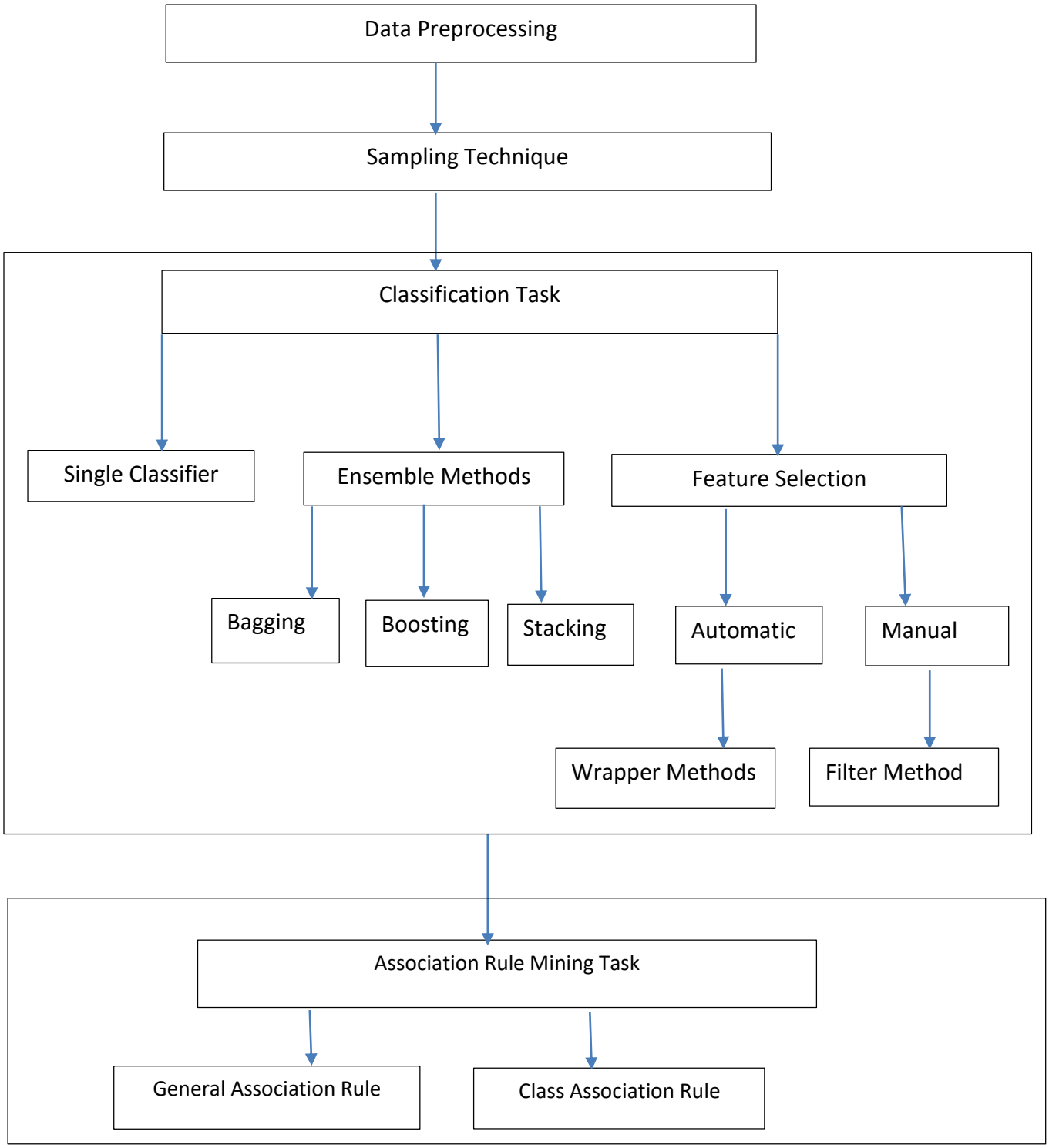


Figure 3.0: Research Design 1

3.3. Data Types

The qualitative and quantitative approach to research both involve the collection and analysis of data involving various attributes to be examined and studied. The various types of data which could be encountered while carrying a research has been generally classified as belonging to one or more of the two categories below:

1) **Categorical variables**

A variable is said to be categorical if it consists of limited number of values which are usually non-overlapping (Agresti and Kateri, 2011). A categorical data type consists of three possible measurement scales; this is identified as ordinal, nominal and binary. An ordinal datatype refers to that which contains an order of occurrence and can be ranked based on their order of precedence. Example includes age, height, size etc. A nominal data type refers to categories which contain no specific order of occurrence. This includes gender, eye colour, and race etc.

A variable is said to be binary if it consists of only two possible allowed values. Examples include gender, yes or no.

2) **Numerical variables**

The numerical data type consists of both continuous and discrete variables. Continuous variables contain no finite values while a discrete variable refers to a variable with a finite amount of permitted values. When a continuous variable is represented using categories, it is referred to as categorization or discretization of the continuous variable (Powers and Xie, 2000)

3.4. Dataset Collection

The bank telemarketing dataset was downloaded from the University of Irvine (UCI) database. UCI is a machine learning repository which makes machine learning compatible dataset available to the public for test and research purposes. The bank telemarketing dataset folder on UCI's website contains four different bank telemarketing datasets gotten from a Portuguese bank. The dataset labelled as "bank-additional-full.csv "which contains all of the example was

used for this research. This dataset contains a total number of 41188 instances where we have 4640 instances belonging to the Yes category and 36548 belonging to the No category.

3.5. Dataset Description

The dataset contains information about the direct marketing campaign conducted by a Portuguese bank. The marketing campaign carried out in this dataset was achieved through the use of phone calls only. The dataset consists of 20 input attributes and a class attribute. The task here is to predict if a customer would subscribe to the bank's term deposit. This was represented using "yes" and "no" as the output variable where "no" represents customers who did not subscribe to a term deposit while "yes" represents customers who subscribed to the term deposit. Table 3.0 shows a list of the attributes in the dataset and their corresponding meaning in this context. The information held by each of attributes are identified as belonging to one of three categories. Attribute 1 to 7 contains information about the bank's client data, attribute 8 to 11 contains information about last contact made for the current marketing campaign attribute 12 to 15 contains information relating to the previous marketing campaign conducted and attribute 16 to 20 contains information relating to social and economic related variables.

Table 3.0: List of the Independent variables and their description

Attribute index	Attribute Name	Attribute Description	Data Type	Values
1	Age	Age of the customer	Numerical	Numerical
2	Job	Job type of the customer	Categorical	Admin, blue-collar, entrepreneur, management, retired, self-employed, services, student, technician, unemployed, unknown

3	Marital	Marital status	Categorical	divorced, (divorced /widowed)married, single, unknown
4	Education	Educational level	Categorical	basic.4y,basic.6y,basic.9y, high school, illiterate, professional course, university degree, unknown
5	Default	Has a credit default status?	Categorical	No, yes, unknown
6	Housing	Has a housing loan status?	Categorical	No, yes, unknown
7	Loan	Has a personal loan status?	Categorical	No, yes, unknown
8	Contact	Contact communication medium	Categorical	Cellular, telephone
9	Month	Last recorded contact month	Categorical	Jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec
10	Day_of_week	Last contact day of the week	Categorical	Mon, tue, wed, thu, fri
11	Duration	Last call duration recorded in seconds	Numerical	Numerical
12	Campaign	Number of contacts made to the customer during the campaign	Numerical	Numerical

13	Pdays	Number of days passed before the next contact day during a campaign	Numerical	Numerical
14	Previous	Number of contacts made to the customer prior to the campaign period	Numerical	Numerical
15	Poutcome	Outcome of the previous campaign	Categorical	Failure, success, non-existent
16	Emp.var.rate	Quarterly indicator of the employment variation index	Numerical	Numerical
17	Cons.price.idx	Monthly indicator of the consumer price index	Numerical	Numerical
18	Cons.conf.idx	Monthly indicator of the consumer confidence index	Numerical	Numerical
19	Euribor3m	Daily indicator of the euribor 3-month rate	Numerical	Numerical
20	Nr.employed	Quaterly indicator of the number of employees	Numerical	Numerical

Table 3.1: Description of the independent variable

Attribute index	Attribute Name	Attribute description	Data type	Values
21	Class	Has the client subscribed to a term deposit?	Categorical	Yes, No

3.6. Research Method And Tools

The quantitative research method was adopted for this research. Williams (2010) identifies the quantitative approach to research as the process of collecting and analysing data with numerical properties through the use of relevant mathematical models which varies depending on the subject being studied. It can also involve the use of already existing statistical data which can also be analysed (Mujis, 2010).

The quantitative approach to research involves the analysis of both the independent and dependent variable. The dependent variable in this case is what the study is focused on which involves the use of Machine Learning Algorithm to predict whether or not a customer would subscribe to a term deposit. The independent variables that would be used in this research are those available in the dataset analysed. This includes age, type of job, marital status, educational status, credit status, housing loan status, personal loan status, communication type, last contact month, last contact day of the week, last call duration, frequency of contacts to the clients during the campaign, interval between the previous campaign day and contact day, results of previous marketing campaign, Employment variation rate, Consumer price index, consumer confidence index , euribor 3 month rate and number of employees within the organization. The independent variables consist of both the numerical and categorical data types while the dependent variable is of the categorical data type. The choice of the research method was dependent on the research problem and objectives.

3.6.1. Machine Learning

Machine learning refers to the process of using statistical concept in developing mathematical models which are capable of carrying out descriptive and predictive modelling (Alpaydin, 2014). The model developed may be predictive due to its ability to make predictions of future occurrences based on some set of training and test data being used while in descriptive modelling, the system is able to provide insight through the knowledge gained from the supplied data. The machine learning process can be divided into supervised and unsupervised learning:

1) Supervised Learning: This works based on the concept of learning from the training set and using the knowledge derived from this to make predictions on some set of newly supplied input.

2) Unsupervised Learning: This uses the concept of learning from observation and discovery (Pujari, 2001). It works through the process of identifying patterns and relationships existing between the supplied data.

The supervised and unsupervised machine learning methods are adopted for this research. The supervised learning adopted here is the use of classification models to make predictions based on the input data supplied. The unsupervised method adopted is the use of association rule mining to identify patterns and relationships existing between the various factors contributing to the reasons why customers are likely to subscribe to a term deposit based on the rules discovered. The Weka machine learning workbench was used to carry out the data mining task.

Weka is a data mining system used for executing data mining tasks through the implementation of machine learning algorithms. It was developed by the University of Waikato in New Zealand. Weka can be used to implement machine learning tasks such as classification, regression, association rule mining, attribute selection and clustering. The Weka workbench presents a flexible and easy means of carrying out analysis of a dataset through the use of its various machine learning algorithms and the data visualization tool. The Weka workbench provides access to four different sections in its Graphical User Interface (GUI) which can be used to carry out diverse data mining tasks. The interfaces are identified below:

1. Explorer: The explorer window can be used to carry out different data mining and visualization task which provides a means of exploring the dataset.
2. Experimenter: This window provides an interface for carrying out the evaluation of machine learning algorithms.
3. Knowledge Flow: The knowledge flow window provides the opportunity of accessing the machine learning components through the means of a data flow connection which is achieved by the individual connection of the Weka components.

4. Simple Command Line Interface: provides an interface for executing the Machine learning tests through the process of typing in commands.

Figure 3.1: The Weka GUI Chooser. 1

3.6.1.1. Data Exploration

The explorer tab is the main Graphical User Interface which was used to carry out various data mining tasks. The data mining tasks which can be carried out on this interface are identified as classification, regression, association rule mining, attribute selection, filtering and clustering

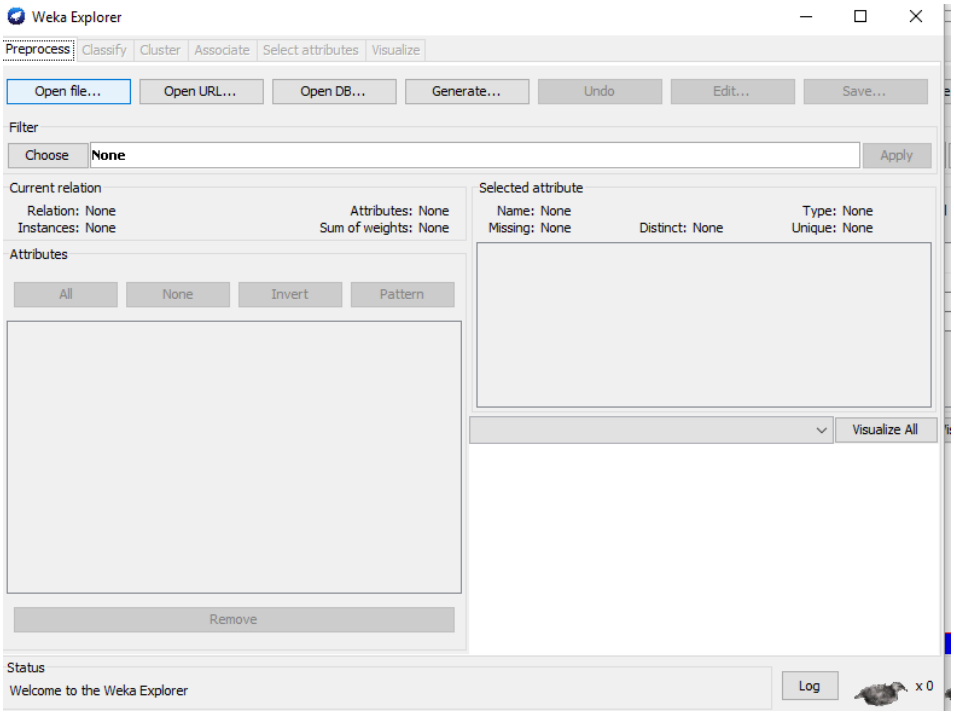


Figure 3.2: The Weka explorer interface.

3.7. Dataset Preprocessing

The data preprocessing step is a very important step in data mining which must be carried out before the application of a data mining technique. The quality of the results or discoveries made from the analysis of a dataset is largely dependent on the quality of the data being used. To work towards generating conclusions that are reusable and adaptable, adequate data pre-processing steps must be carried out. This involves carrying out tasks such as data cleaning, data integration, data transformation and data reduction (García, Luengo and Herrera, 2015).

In today's real world setting, the dataset being utilized are usually being gotten from various sources which makes databases highly susceptible to noisy, inconsistent and missing values (Han, Pei and Kamber, 2011). The process of ensuring that high quality results are being generated through data mining involves proper data preprocessing steps. The data preprocessing steps includes data cleaning, data integration, data transformation and data reduction (García, Luengo and Herrera, 2015). Data cleaning involves the identification or removal of noise and outliers, filling in missing values and removal of duplicate data. Data integration involves the combination of several sources of data into one. Data transformation involves the conversion of data into a suitable form for a data mining task, example of such conversion includes normalization and discretization. Data reduction involves the selection and removal of attributes in a dataset

3.7.1 File Conversion

The dataset was originally gotten from the UCI database in the Comma-separated Values (CSV) format. Weka can only launch dataset which are stored in the Attribute-Relation File Format (ARFF). To convert the dataset into ARFF, the file was launched using the notepad editor and the file structure was changed by editing the first lines which contains the attribute names and the header structure comprising of the @relation, @attribute and @data tag. The @relation tag is used to store the name of the dataset, the @attribute tag is used to store the attribute names and possible data values while the @data tag is used to store the values for each of the instances contained in the dataset. After this conversion process, the file was saved using the. arff extension.

Figure 3.3 below shows the file in its initial csv format while Figure 3.4 below shows the file its converted state to an Arff format.

```
"age";"job";"marital";"education";"default";"balance";"housing";"loan";"contact";"day";
"month";"duration";"campaign";"pdays";"previous";"poutcome";"y"
58;"management";"married";"tertiary";"no";2143;"yes";"no";"unknown";5;"may";261;1;-1;0;"unknown";"no"
44;"technician";"single";"secondary";"no";29;"yes";"no";"unknown";5;"may";151;1;-1;0;"unknown";"no"
33;"entrepreneur";"married";"secondary";"no";2;"yes";"yes";"unknown";5;"may";76;1;-1;0;"unknown";"no"
47;"blue-collar";"married";"unknown";"no";1506;"yes";"no";"unknown";5;"may";92;1;-1;0;"unknown";"no"
33;"unknown";"single";"unknown";"no";1;"no";"no";"unknown";5;"may";198;1;-1;0;"unknown";"no"
35;"management";"married";"tertiary";"no";231;"yes";"no";"unknown";5;"may";139;1;-1;0;"unknown";"no"
28;"management";"single";"tertiary";"no";447;"yes";"yes";"unknown";5;"may";217;1;-1;0;"unknown";"no"
42;"entrepreneur";"divorced";"tertiary";"yes";2;"yes";"no";"unknown";5;"may";380;1;-1;0;"unknown";"no"
58;"retired";"married";"primary";"no";121;"yes";"no";"unknown";5;"may";50;1;-1;0;"unknown";"no"
43;"technician";"single";"secondary";"no";593;"yes";"no";"unknown";5;"may";55;1;-1;0;"unknown";"no"
41;"admin.";"divorced";"secondary";"no";270;"yes";"no";"unknown";5;"may";222;1;-1;0;"unknown";"no"
29;"admin.";"single";"secondary";"no";390;"yes";"no";"unknown";5;"may";137;1;-1;0;"unknown";"no"
53;"technician";"married";"secondary";"no";6;"yes";"no";"unknown";5;"may";517;1;-1;0;"unknown";"no"
58;"technician";"married";"unknown";"no";71;"yes";"no";"unknown";5;"may";71;1;-1;0;"unknown";"no"
57;"services";"married";"secondary";"no";162;"yes";"no";"unknown";5;"may";174;1;-1;0;"unknown";"no"
51;"retired";"married";"primary";"no";229;"yes";"no";"unknown";5;"may";353;1;-1;0;"unknown";"no"
45;"admin.";"single";"unknown";"no";13;"yes";"no";"unknown";5;"may";98;1;-1;0;"unknown";"no"
57;"blue-collar";"married";"primary";"no";52;"yes";"no";"unknown";5;"may";38;1;-1;0;"unknown";"no"
```

Figure 3.3: Presentation of the dataset in the CSV Format

```
@relation 'bank marketing'

@attribute age numeric
@attribute job {admin.,blue-collar,entrepreneur,housemaid,management,retired,self-employed,services,student,technician,unemployed,unknown }
@attribute marital {divorced,married,single,unknown}
@attribute education {basic.4y,basic.6y,basic.9y,high.school,illiterate,professional.course,university.degree,unknown }
@attribute default {no,yes,unknown }
@attribute housing {no,yes,unknown}
@attribute loan {no,yes,unknown}
@attribute contact {cellular,telephone}
@attribute month {jan,feb,mar,apr,may,jun,jul,aug,sep,oct,nov,dec}
@attribute day_of_week {mon,tue,wed,thu,fri}
@attribute duration numeric
@attribute campaign numeric
@attribute pdays numeric
@attribute previous numeric
@attribute poutcome {failure,nonexistent,success }
@attribute emp.var.rate numeric
@attribute cons.price.idx numeric
@attribute cons.conf.idx numeric
@attribute euribor3m numeric
@attribute nr.employed numeric
@attribute class {yes,no}
@data

56,housemaid,married,basic.4y,no,no,no,telephone,may,mon,261,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
57,services,married,high.school,unknown,no,no,telephone,may,mon,149,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
37,services,married,high.school,no,yes,no,telephone,may,mon,226,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
40,admin.,married,basic.6y,no,no,no,telephone,may,mon,151,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
56,services,married,high.school,no,no,yes,telephone,may,mon,307,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
45,services,married,basic.9y,unknown,no,no,telephone,may,mon,198,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
59,admin.,married,professional.course,no,no,no,telephone,may,mon,139,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
41,blue-collar,married,unknown,unknown,no,no,telephone,may,mon,217,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
24,technician,single,professional.course,no,yes,no,telephone,may,mon,380,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
25,services,single,high.school,no,yes,no,telephone,may,mon,50,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
41,blue-collar,married,unknown,unknown,no,no,telephone,may,mon,55,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
25,services,single,high.school,no,yes,no,telephone,may,mon,222,1,999,0,nonexistent,1.1,93.994,-36.4,4.857,5191,no
```

Figure 3.4: Conversion of the dataset to the Arff format

3.7.2. Loading the dataset in Weka

After converting the dataset to arff, the dataset was loaded into Weka through the Weka explorer Interface by clicking on the open file icon on the preprocess tab and locating the dataset saved on the local file system

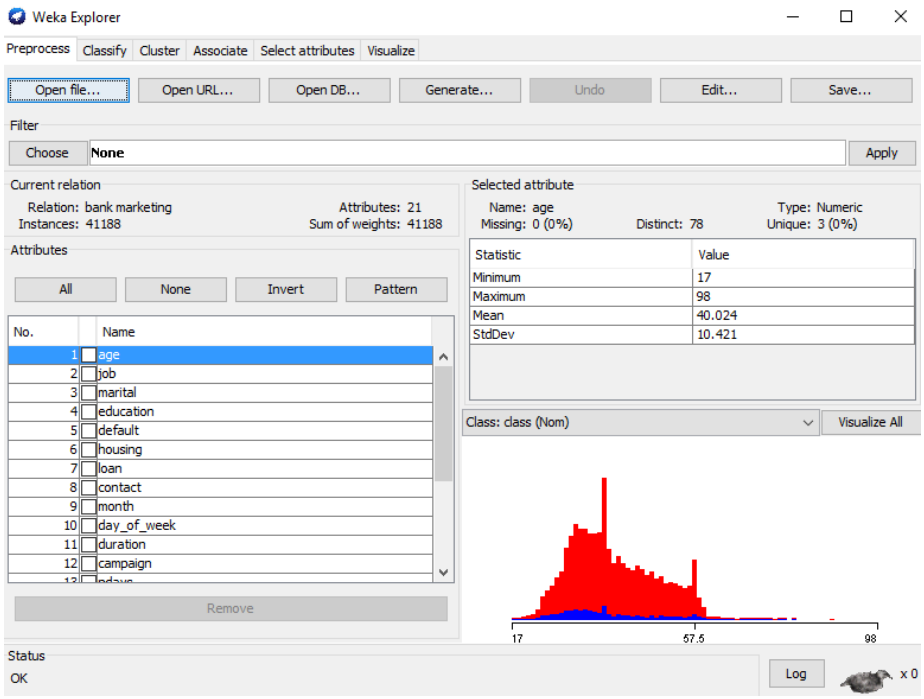


Figure 3.5: Loading the dataset in Weka

3.7.3. Data Cleaning

Data cleaning is an important step in data preprocessing due to its ability to help improve the quality of the dataset for a more reliable output. The presence of impurities in real-world data application has brought about the development of several methods to eradicate this problem to help improve the accuracy and usability of existing data (Müller and Freytag, 2005). The data cleaning process involves the detection or removal of outliers, smoothing noisy data, filling in missing values and resolving inconsistency within a dataset (Han, Pei and Kamber, 2011).

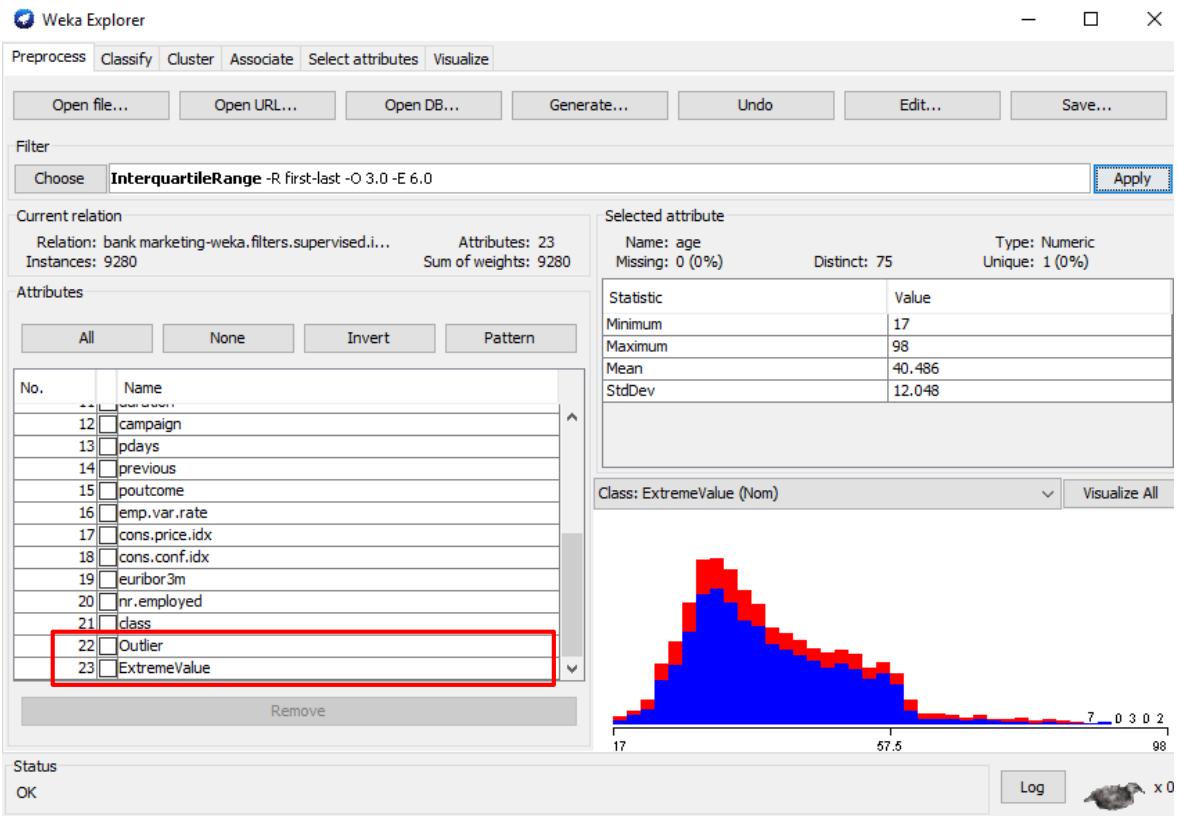
3.7.3.1 Detection of noisy, missing and inconsistent data

Real world data undergoes different processes which makes them highly susceptible to noise. It is impossible to have a real-world dataset which is entirely perfect (Zhu and Wu, 2004). The presence of noise in a dataset can lead to a reduction in the performance of learning algorithms

which could result in misclassification and longer computational time for the classifier (Zhu and Wu, 2004). Outliers are data with different behaviour from the rest of the samples which acts as though they were generated from a different probability distribution (Zhu and Wu, 2004). The presence of outliers in a dataset can either be due to variability in the attribute value which is correctly represented or could be as a result of an incorrectly recorded value. The process of achieving high quality data for the learning task can be achieved through some data cleaning process which could involve the removal or correction of noisy instances and prediction of missing attribute values.

The dataset was inspected to check for the presence of missing values. All of the attributes were checked to ensure that no missing values exist. All of the instances were found to contain no missing values

The outliers and extreme values were detected by using the interquartile range filter under Weka's unsupervised attribute tab. The application of this filter resulted in the addition of two extra attributes to the already existing 21 attributes. The 22nd attribute was labelled as an outlier while the 23rd attribute was labelled as an extreme value. The 22nd and 23rd attributes were examined to check why those instances were being classified as outliers and extreme value. After proper examination of some of the instances identified as having an outlier, it was discovered that the duration attribute was responsible for the occurrence of this classification due to some of its instances containing very high value when compared with others. The duration attribute contains information relating to the call duration of the marketing campaigns directed at each individual in all of the instances. This attribute was considered important to the dataset and so the instances containing such high values were retained for the purpose of carrying out an effective analysis to help discover if the amount of time spent on calls affects the outcome of the campaign.



1Figure 3.6: Identification of the outliers and extreme values

```
46,admin.,single,university.degree,no,yes,no,cellular,nov,tue,1166,3,999,1,failure,-1.1,94.767,-50.8,1.046,4963.6,no,yes,yes
34,technician,married,unknown,no,no,no,cellular,nov,tue,985,3,999,0,nonexistent,-1.1,94.767,-50.8,1.046,4963.6,yes,yes,no
36,blue-collar,single,basic.6y,no,no,no,cellular,nov,tue,1556,4,999,0,nonexistent,-1.1,94.767,-50.8,1.046,4963.6,yes,yes,no
```

Figure 3.7: Detection of instances containing outliers and extreme values

3.7.4. Data Transformation

The data transformation process involves the conversion of the data into an appropriate mining form. Different data mining task could involve the use of the data in several formats. For instance, the association mining task requires the transformation of continuous values into categorized values. Han, Pei and Kamber (2011) identified six strategies for data transformation; these are identified as smoothing, attribute construction, aggregation, normalization, discretization and concept hierarchy generalization.

3.7.4.1. Data Discretization

Discretization is a necessary step in data mining when handling certain machine learning task such as association rule mining, induction rules and Bayesian networks which require the use of discrete values as opposed to continuous values before any of its tasks can be executed (Maslove, Podchiyska and Lowe, 2013). Certain machine learning algorithms such as Support Vector Machines and Random Forest are known to work well on discretised data due to their robustness for high dimensionality data (Lustgarten et. al., 2008). Other machine learning algorithms such as Naïve Bayes also work well in its discretised state due their sensitivity to the dimensionality of dataset

Discretization refers to the process of converting continuous attributes data type into a discrete form, it also acts as a form of variable selection approach (Lustgarten, et al., 2008). The discretization of a numeric attribute can lead to the improvement of a classifier's performance.

Discretization method can either be classified as supervised or unsupervised discretization. The supervised discretization approach makes use of the class variable in the discretization process while the unsupervised approach does not. The unsupervised method consists of the Equal-Width binning method which divides the continuous variables into a specified number of intervals and the Equal-Frequency binning method which divides the continuous variables into a specified fraction of instances per interval (Lustgarten et al., 2011).

For this research, the unsupervised approach to discretization was adopted by using the equal frequency binning method. This was achieved by accessing this option from Weka's unsupervised attribute filter by changing the "UseEqualFrequecy" option to true. The dataset was only used in its discretised discretized state during the association rule mining task due to the fact that association rule mining only permits the use of categorical values.

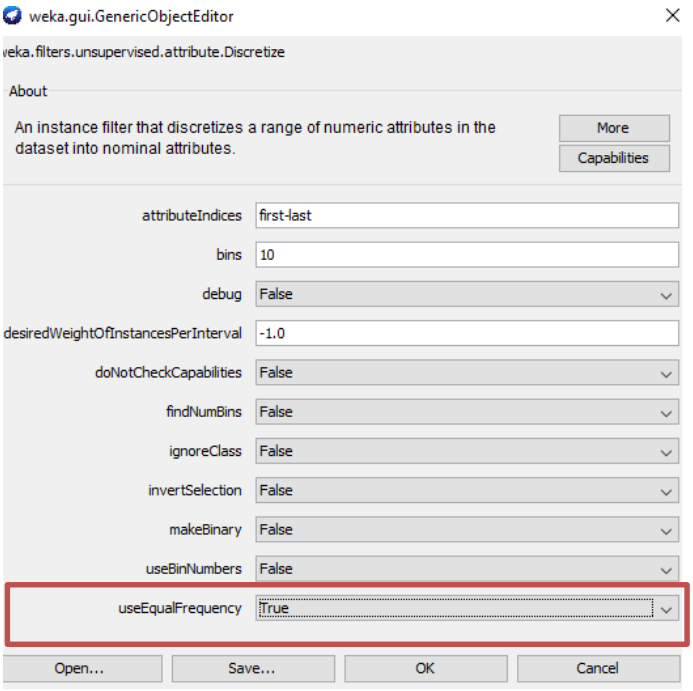


Figure 3.8: Equal frequency discretization

3.8. Data Balancing Technique

In data mining, the process of balancing the class could be an important step when dealing with a dataset with a high amount of imbalance depending on the focus of the data mining task. For this a dataset, the class attribute comprises of two possible outcomes. The data for the Yes class is highly skewed when compared to that of the No class. The Yes class contains 4640 while the No class contains 36548 instances. The two most common data resampling technique to address the class imbalance is the oversampling and undersampling technique. The oversampling technique is used to increase the minority class instances while the undersampling technique is used to reduce the majority class instances.

To ensure that the result generated from the outcome of the data mining task is an unbiased representation of both occurrences, the dataset was balanced by using the random undersampling method. The random undersampling method was chosen over the oversampling approach due to the fact that the size of the dataset would be very large if an oversampling technique is applied to the already existing majority class which contains 35648 instances. The undersampling technique has been known to lead to the loss of potentially viable and useful

information but this approach was still being selected due to the use of computationally intensive machine learning algorithms such as the Support Vector Machine and the Neural Network which is not very suitable for the classification task of large dataset due to their intensive computational complexity (Cervantes et al., 2008). Drummond and Holte (2003) suggested that the use of the oversampling method could be unwarranted due to its computational cost especially when the undersampling approach also has the tendency to produce similar good performance.

3.8.1. Undersampling the dataset

The dataset was undersampled by using the supervised instance filter called Spread Subsample. Spread Subsample is a method which works through the process of random elimination of the instances from the majority class which is achieved through a spread value specified by a user which is used to determine the balance ratio to be achieved between the majority and minority class (Loyola-González et al., 2013). This filter was used to specify the rate at which the distribution between the majority and the minority class can be spread. The distribution spread option was set to 1.0 so that the majority class instances would be undersampled by using a 1:1 ratio with the minority class. The majority class instances were reduced to the same number of instances as the minority class.

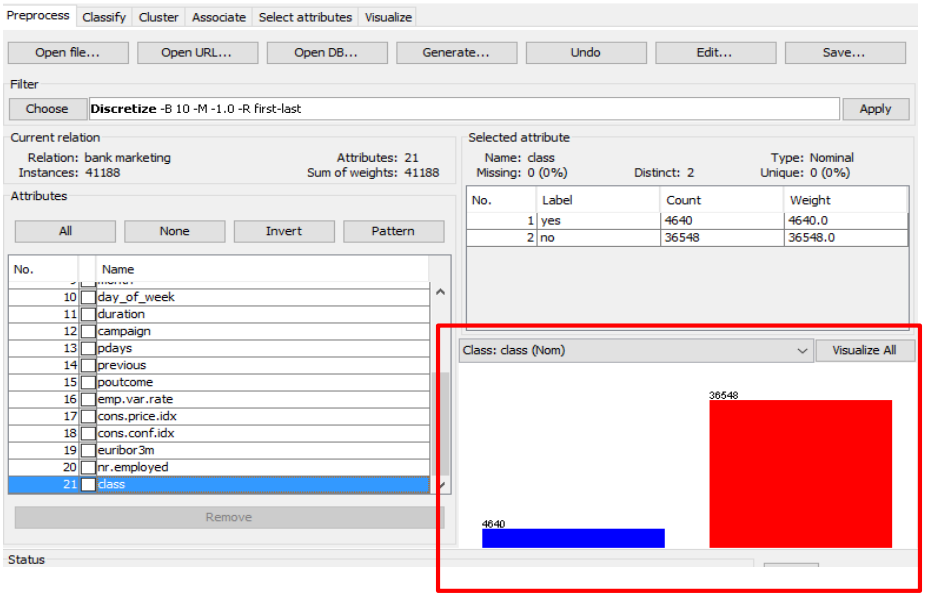


Figure 3.9: Visualization of the class distribution before undersampling

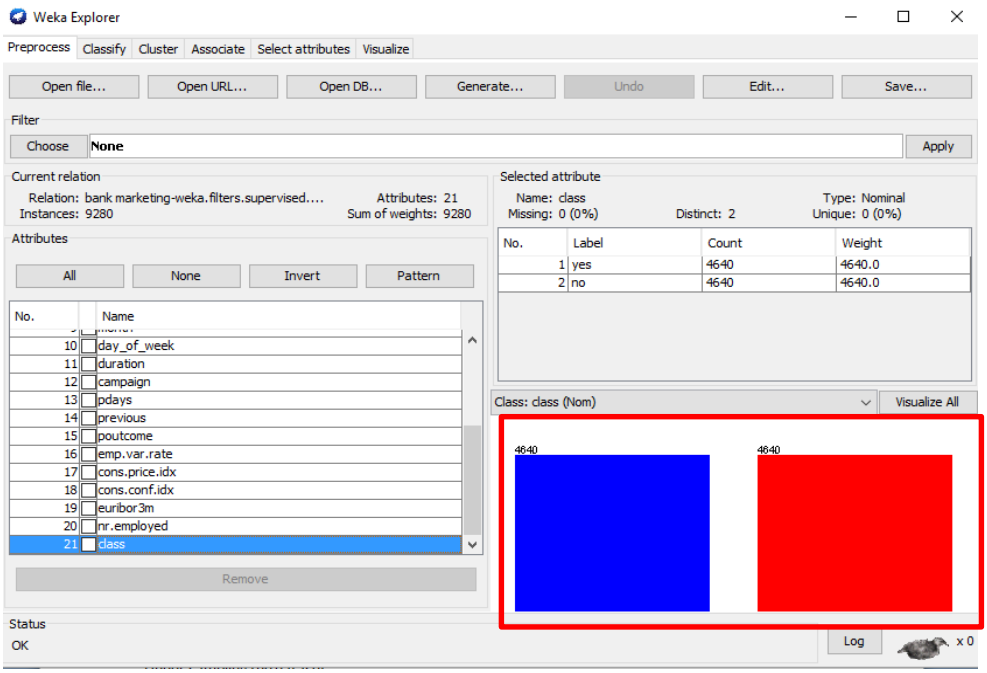


Figure 3.10: Visualization of the class distribution after undersampling

3.9. Classification

Classification is a supervised learning task whose aim is to predict a target output. The classification process adopted for this research was carried out in three phases. The first phase represents the use of the classification algorithms on the undersampled dataset, the second phase involves the use of the bagging, boosting and stacking approach to classify the dataset while the third phase involves the use of the feature selection approach to classify the dataset. A classification algorithm can handle different attributes of different datatype depending on its design capabilities. For instance, classification algorithms such as Naïve Bayes and J48 can handle continuous and categorical input values. A proper understanding of the capability of the intended algorithm is key to understanding how best to achieve optimum accuracy. In order to generate a classification tool which is applicable to various domains, an existing understanding of the response of the classification algorithms to different datasets is essential (Panda and Patra, 2008). The patterns discovered during the classification task can be used to understand how the algorithm would classify new instances.

To achieve the aim of comparing the results achieved through the sampling technique, algorithm based technique and the feature selection technique to handling class accuracy predictions, six classification algorithms were used all through the process of carrying out this analysis. These are identified below with a brief introduction into their individual concepts and approach;

1) Naïve Bayes: The Nave Bayes classifier is an easy to implement statistical method for classification which operates based on the assumption of the class conditional independence which assumes that the effect of the values contained in an attribute operates independently with no regards to the presence of any other attributes in the class. It has the ability to handle both categorical and continuous attributes. Naïve Bayes has been used by various researchers in the area of direct marketing (Karim and Rahman, 2013), credit scoring (Antonakis and Sfakianakis, 2009), network intrusion detection (Panda and Patra, 2007) and medical diagnosis (Kazmierska and Malicki, 2008). The effectiveness of the utilisation of this classifier has been recently outperformed by other classifiers such as the boosting-based classifier and the Support Vector machine (Alabau, et al., 2006).
2) Random Forest: Random Forest is an ensemble of unpruned classification or regression trees developed through bootstrap sampling of the training data using the random selection of input variables to determine the split point (Khalilia, Chakraborty and Popescu, 2011). It operates on the principles of making predictions based on voting or aggregate of the ensemble's prediction. It can be used as a classification or regression tool and it has the ability to handle high dimensional data and can use a large number of tress in the ensemble. Random Forest has the effective ability to handle missing values and also estimate the relevance of the variables to be used in the classification process (Breiman, 2001) whilst ensuring that the accuracy of its predictions is sustained.
3) J48: The J48 classifier is a C4.5 decision tree classifier which has the ability to handle both continuous and discrete values as the independent variables while the dependent variable must be categorical. J48 ignores missing values when building a tree based on its ability to predict the value for those items through the use of information gotten

from the attribute values for other items within the dataset (Patil and Sherekar, 2013). It uses the concept of information gain by setting a threshold value and selecting attributes which have values equal to or greater than the set threshold. After the construction of the tree, the algorithm carries out a pruning process on the tree where attributes that do not help in reaching the leaf nodes are removed (Kaur and Chhabra, A, 2014). The j48 classifier is among the most powerful decision tree classifiers, it is also popular due to its ease of construction which does not require any domain expertise or parameter setting and is a very useful tool for exploratory knowledge analysis (Rajput, 2011).

4) LibSVM: This is a library for Support Vector Machines (SVM); it is one of the mostly used SVM software that supports the use of various SVM formulations to carry out classification, regression and distribution estimate (Chang and Lin, 2011). It supports vector regression, support vector classification and one-class support vector machine. The SVM algorithm uses the hyperplane to correctly identify the appropriate training instances and also creates a maximum margin between the support vectors (Sevakula and Verma, 2012). It uses the hyperplane to segment the different class labels to ensure that a large margin exists between those classes so that more accurate predictions can be made.
5) Sequential Minimal Optimization (SMO): SMO is a Support Vector Machine (SVM) learning algorithm which is suitable for solving the training problem encountered by SVM. The process of training a SVM requires solving very large Quadratic Optimization(QP) problems which are usually computationally intensive (Platt, 1998). SMO provides a means of solving this large Quadratic Programming(QP) optimization problem by breaking it down into series of smaller QP problems (Platt, 1998). Most SVM in real world applications are routinely trained by using the SMO due to its fast, simple and easily adaptable means of scaling to large problems (Sun et al., 2010).
6) Multilayer Perceptron: The Multilayer Perceptron Neural Networks are the most commonly used feedforward neural networks, it has been termed the universal approximate and also has the ability to provide optimal solutions to classification

problems (Chaudhuri and Bhattacharya, 2000). The MMLPNN consists of the input, output and the hidden layer. Data flows in a unidirectional manner from the hidden layer which processes it and sends over the input information to the output layer (Orhan, Hekim and Ozer 2011. A sufficient amount of neurons should be contained in the hidden layer so that the problem of generalisation and overfitting can be avoided due to insufficient or excessive number of neurons in the hidden layer (Orhan, Hekim and Ozer, 2011). The model is very powerful and can be effectively applied to diverse number of real-world application such as medical diagnosis (Kumar, 2012.), pattern recognition (Chaudhuri and Bhattacharya, 2000) and signal classification (Orhan, Hekim and Ozer, 2011).

Classifier Test Options

The classifiers chosen are usually tested based on four modes of evaluation. These are identified as use training set, supplied test set, cross-validation and percentage split. The 10 fold cross-validation test option was used for the evaluation of the models used.

A brief description of each of the available test option is given below:

1) Use training set: The classifier is trained on certain number of instances and the same instances learnt by the classifier are also tested. This means that the classifier is being evaluated on the same training instances. This method is considered less efficient because it is considered cheating because testing is being done on the training set.

2)Use supplied test set: For this method, the performance of the classifier is evaluated based on its ability to predict the new set of instances being loaded from a different file (Kirkby, Frank and Reutemann, 2007). The data loaded through the preprocess tab is used as the training data while a new file is being supplied to be used as a test set.

3) Cross-validation (CV): The performance of the classifier is evaluated by using the specified cross-validation folds. For instance, for a tenfold cross-validation, the dataset is split into 10

folds and 9 of these divisions are used as the training set while the remaining 1 fold is used as the test set. This process is repeated for all of the folds. This process is repeated for each of the folds by leaving out the last fold and using it as a test set. This method is very popular for algorithm selection due to its simplicity and universality of the data splitting heuristics (Arlot, and Celisse, 2010)

4) Percentage split: This method is used to specify what percentage of the dataset is used for the training and test purpose. The training and testing of the classifier is carried out just once as opposed to the use of cross-validation which carries out the split process for the number of folds specified. This split is usually done using 66% for training and 34% for testing (Salvithal and Kulkarni, 2013)

3.9.1. Classification task 1: Single Classifier Based Method

The process of classifying new data points using the machine learning concept requires the use of a classification algorithm. This is usually achieved by selecting a classifier to carry out the training and test process. The dataset was resampled by using the undersampling technique and then carrying out a 10-fold cross validation for each of the selected classifiers. The performance of single classifiers has recently been outperformed by using an ensemble of classifiers which has shown to improve the generalization performance of several machine learning algorithms (Oza and Tumer, 2001).

3.9.2. Classification task 2: Algorithmic Method

The bagging, boosting and stacking approach are ensembles method which are used to improve the accuracy of different classifiers through a combination of several machine learning algorithms. Ensemble methods are learning models that help in improving the performance of classifiers through the process of combining the output of different classifiers predictions and then taking a weighted vote of their individual predictions. The use of ensembles methods has the capability of converting the performance of a weak classifier into a good one. It also helps to reduce variance and also perform regularization (Daumé III, 2012). A lot of research has been carried out in supervised learning task to help determine how best to construct a good ensembles of classifiers. The main discovery in relation to this diverse research has been the

identification of the fact that in most cases, an ensemble of classifiers usually produces better results than the individual classifiers that makes up the ensemble (Dietterich, 2000). To achieve optimal classification through the use of ensembles, it is best to combine classifiers with high variability in their individual classification results. Unfortunately, the inductive biases of most algorithms tend to be highly correlated and this results in them being susceptible to similar types of error (Daumé III, 2012). Since ensembles methods use the concept of majority voting to carry out the final classification process, it is best to combine classifiers that are not correlated. For instance, if classifier 1 makes an incorrect prediction but classifier 2 and 3 could classify the classes correctly, the majority voting would be applied and cause the instances to be classified based on the output of classifier 2 and 3 but if the classification algorithms combined are correlated, they all tend to make similar mistakes and cause the final classification to wrong based on their individual outputs.

3.9.2.1. Bagging

The dataset was classified using the bagging technique. Six base classifiers were used to test the performance of the bagging method. Bagging uses the concept of bootstrap resampling where the training set is sampled with replacement. Bagging uses a single classifier to train several datasets. This is achieved by splitting the original dataset into different training set and each of the training set is trained by using the same learner for each partition and the outputs of the different models are combined together to generate a final model. The bagging technique works well for unstable algorithms such as decision tree, neural networks and rule based algorithms where a slight change in the training data could result in a major change in the output while stable learning algorithms such as linear regression, nearest neighbour and linear threshold algorithms are generally regarded as being stable (Dietterich, 2000).

The combination of algorithms chosen for the comparison of the bagging results consists of both the stable and unstable learning algorithm. This was done in order to study how well all of the six algorithms chosen independently handle class predictions. The J48, Multilayer Perceptron, and Random Forest classifier are all unstable algorithms while Naïve Bayes, LibSVM and SMO classifier are considered as stable learners.

3.9.2.2. Boosting

The AdaboostM1 classifier was used to boost the performance of the base classifiers used to perform the tests. Daumé III (2012) describes boosting to be more of a framework rather than an algorithm. He describes it as a framework which converts the performance of a weak classifier into that of a strong classifier. Adaboost can be used on different classification algorithms but ideally, researchers suggest that it should be used on weak classifiers to achieve optimum performance (Rangel, Lozano and García 2005)

3.9.2.3. Stacking

The stacking method was utilized by carrying out series of test using a combination of several classifiers. The test was done by carrying out three different iterations. The first set of test involves the combination of two classifiers where the classifier used as the base and Meta learner was inter-switched after each iteration to find out which algorithm works best as the Meta classifier. The same process was adopted for the second and third set of test involving a combination of three and four classifiers respectively.

3.9.3. Classification task 3: Feature Selection

The feature selection task was carried out through the use of the manual and automatic feature selection technique. Feature selection helps to reduce the dimensionality of the dataset through an attribute selection process which helps to reduce dimensionality before passing on the instances to the classifiers. The effectiveness of a data mining task can be reduced when irrelevant features are used while carrying out the various data mining processes. The process of removing certain irrelevant and redundant features within a dataset can help create better predictors. There several benefits of carrying out a variable and feature selection task, this includes providing better understanding of the data through data visualization, reduction in training time and also reduction in measurement and storage requirement (Guyon, and Elisseeff, 2003).

3.9.3.1. Manual Feature Selection

The manual feature selection process was carried out by splitting the dataset into three partitions based on the selection of certain features. The first partition contains information regarding the bank's client, the second partition contains information regarding the contacts made by the bank to the customers

while the third partition contains information regarding the social and economic variables present in the dataset. This method was adopted to help discover what factors contribute to reasons why customers would subscribe to a term deposit and also to help develop and improve the predictive accuracy of the models used.

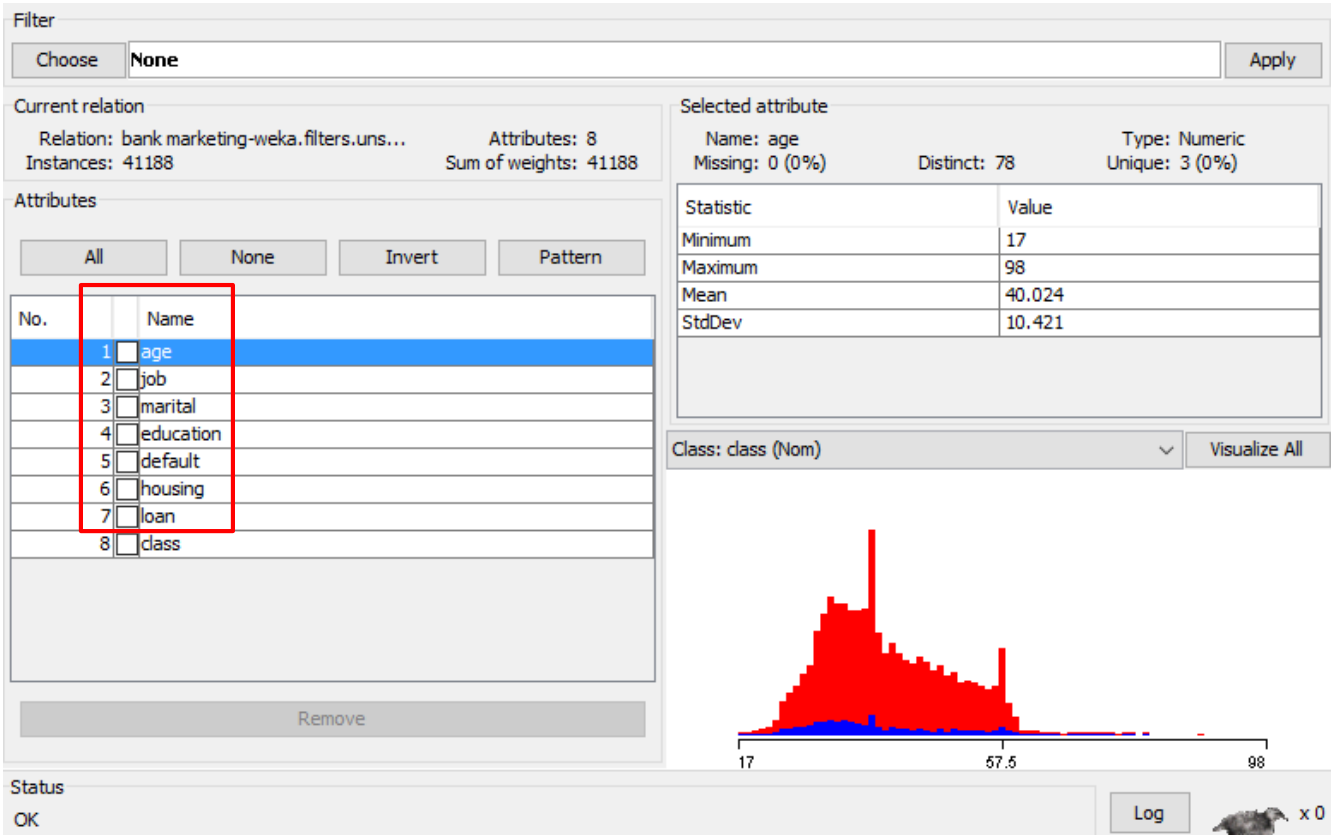


Figure 3.11: Feature selection using demographics related attributes

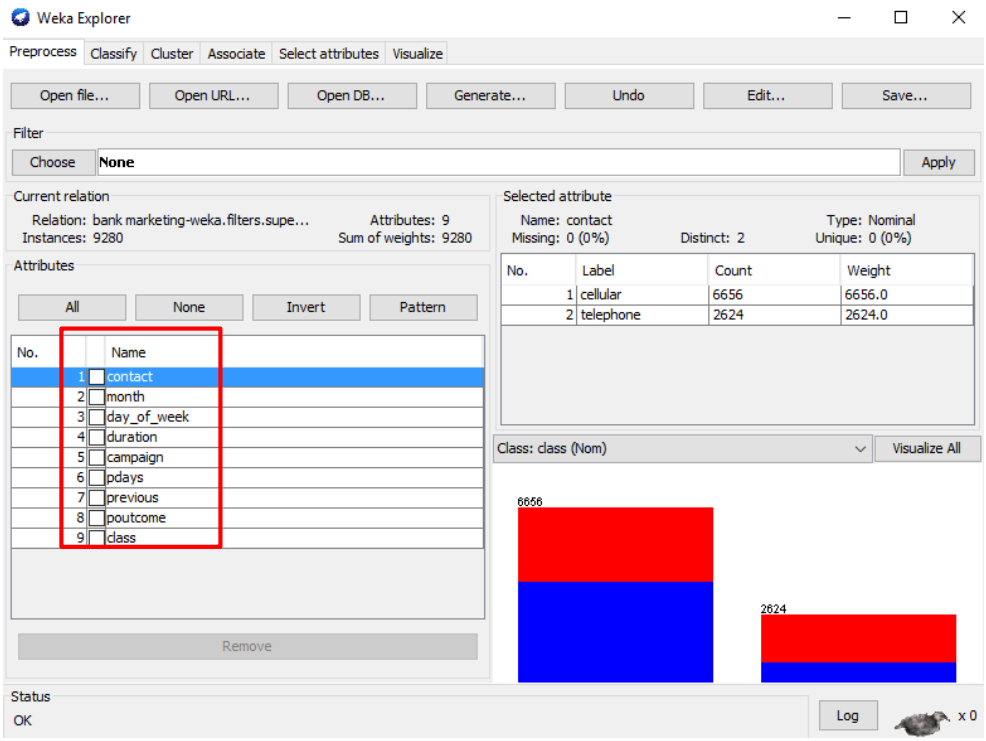


Figure 3.12: Feature selection using bank campaign history related attributes

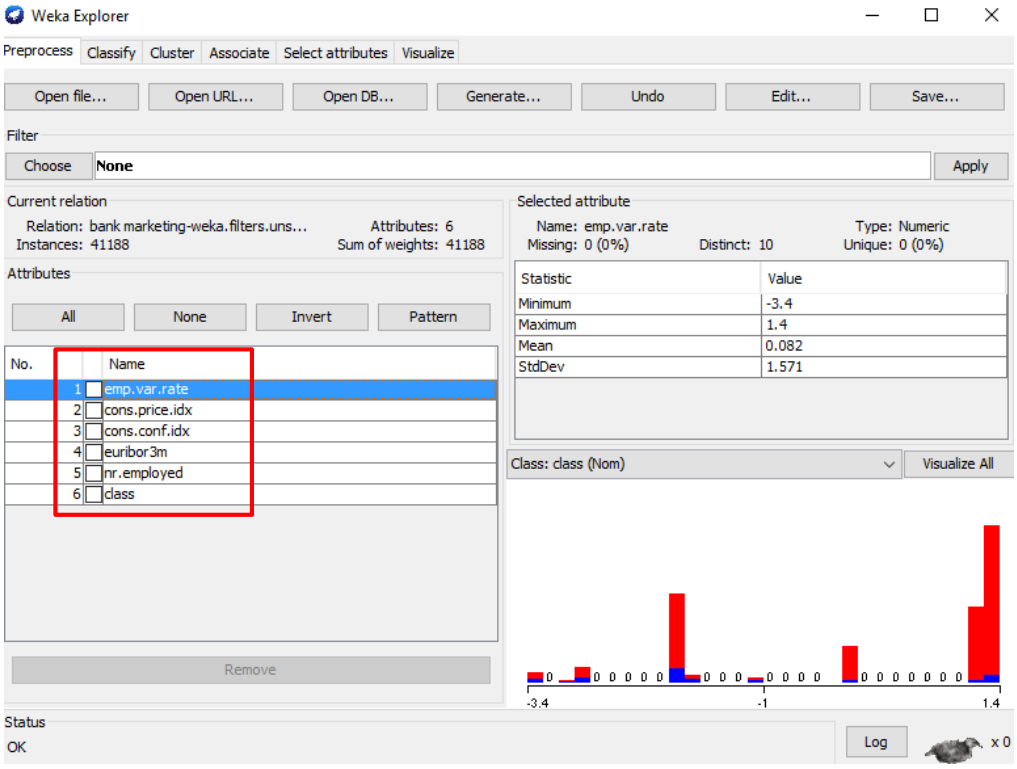


Figure 3.13: Feature selection using social and economic related attributes

3.9.3.2. Automatic Feature Selection

The automatic feature selection method can be divided into wrapper, filters and embedded methods (Maldonado and Weber, 2011). For the wrapper method, the feature selection algorithm is used alongside an induction algorithm (Kohavi and John 1997). The feature selection algorithm carries out the feature evaluation and selection process while making use of an induction algorithm to evaluate the subsets of features (Kohavi and John, 1997). The final subsets of features with the highest evaluation is chosen and then passed on to the induction algorithm. The filter method uses a subset of features as a pre-processing step without taking the predictor into consideration while carrying out the feature selection process (Guyon and Elisseeff, 2003). This method uses the statistical properties of the features to select a subset of features based on a rank score, this is always done before applying any classification algorithm (Maldonado and Weber, 2011). The filter approach has been argued to be faster than the wrapper method and also can also be used as pre-processing step to help reduce space dimensionality and overfitting (Guyon and Elisseeff, 2003). Wrapper methods are computationally intensive but tend to produce more accurate results than the filter method (Maldonado and Weber, 2011). The Embedded methods introduce the feature selection process as part of the training process (Guyon and Elisseeff, 2003). It determines what subset of feature is most important while building the model.

3.9.3.2.1. Wrapper Based Methods

For the wrapper based feature selection task, the classifier, evaluator and the search method option was selected in Weka. The classifier used as the base learner for the classification task was also the same classifier used to estimate the accuracy of the feature subsets. For instance, in figure 3.14 below, Naïve Bayes is used as the feature selection algorithm and it is wrapped around the Naïve Bayes classification algorithm. The wrapper subset evaluator was used as wrapping evaluator for the wrapping algorithm. The Best first search method was selected over the greedy stepwise due its ability to utilize both the forward and backward search propagation technique which provides the opportunity to determine at what point the search should be terminated after a defined number of non-improving nodes as opposed to the greedy stepwise option which uses just either the backward or forward approach to search through a space of

attribute subset. According to a research carried out by Kohavi and John (1997), the best first method is known to perform better than the greedy stepwise method.

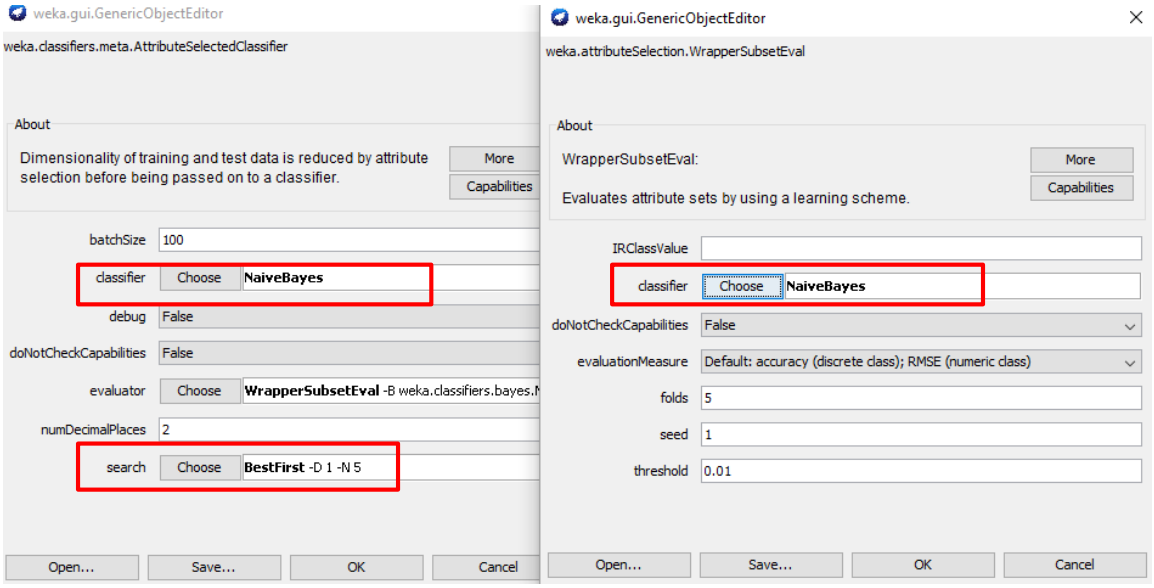


Figure 3.14: Wrapper feature selection method

3.9.3.2.2. Filter Method

For the filter based feature selection task, the classifier, evaluator and the search method options were also selected in Weka. The Gain Ratio Attribute evaluator was used to carry out the evaluation on the relevance of an attribute. The ranker search method was used as the search method to rank attributes based on their individual evaluation scores calculated by the evaluator. The number of attributes to select was set to 8. This helps to define the number of attributes the Ranker would pass on to the base classifier for the classification of the class instances.

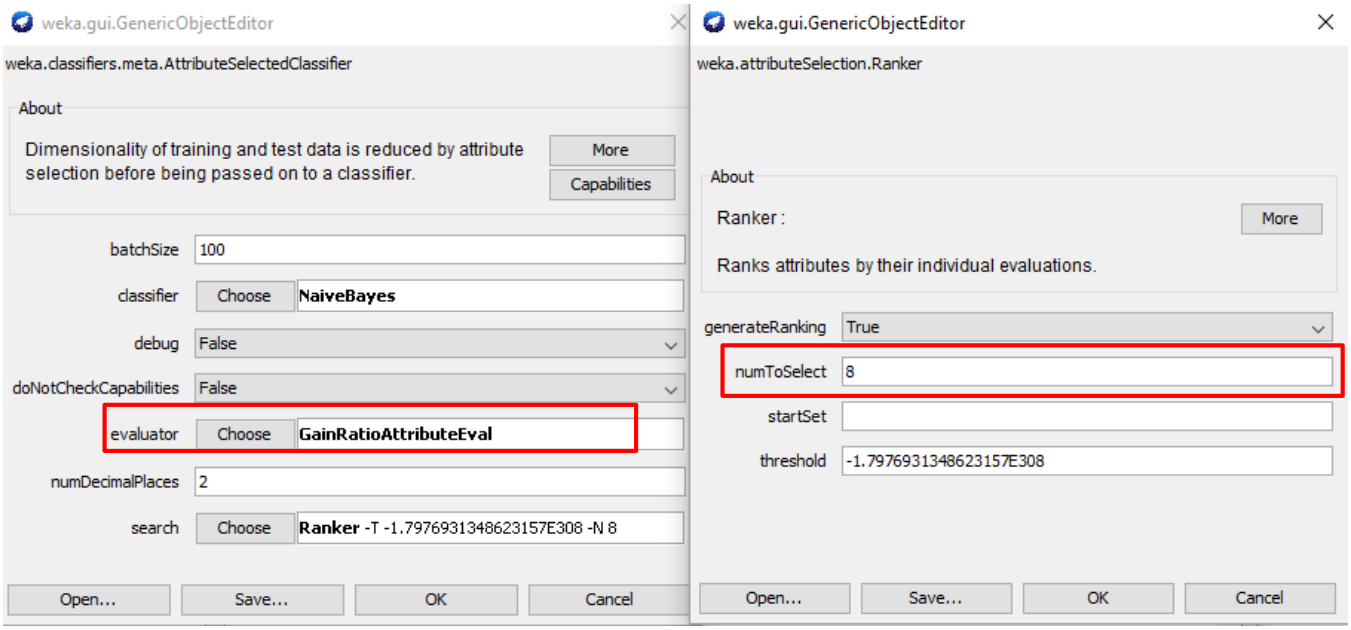


Figure 3.15: Feature selection using the gain ratio evaluation measure

3.9.4. Classification Task 4: Combining Feature Selection and Ensembles

The feature selection task helps to handle the problem of high dimensionality in dataset while the ensembles method helps to address the problem of having weak classifiers by producing stronger classification result through a combination of several classification algorithms. The feature selection task and the use of ensembles can be combined to generate better results. Previous researches have combined the feature selection task with boosting (Gao, Khoshgoftaar and Wald, 2014) and bagging (Johansson et al., 2010) for better performance optimization. In this research, the feature selection task was carried within each of the ensembles method (bagging, boosting and stacking). The filter feature selection method was embedded in each of the ensembles methods adopted. For instance, when implementing the feature selection task in bagging, only the attributes selected based on the specified ranking is used on each bootstrap as opposed to the use of all the attributes present in the dataset. The gain ratio evaluation measure and the ranker search method was used in the filter feature selection.

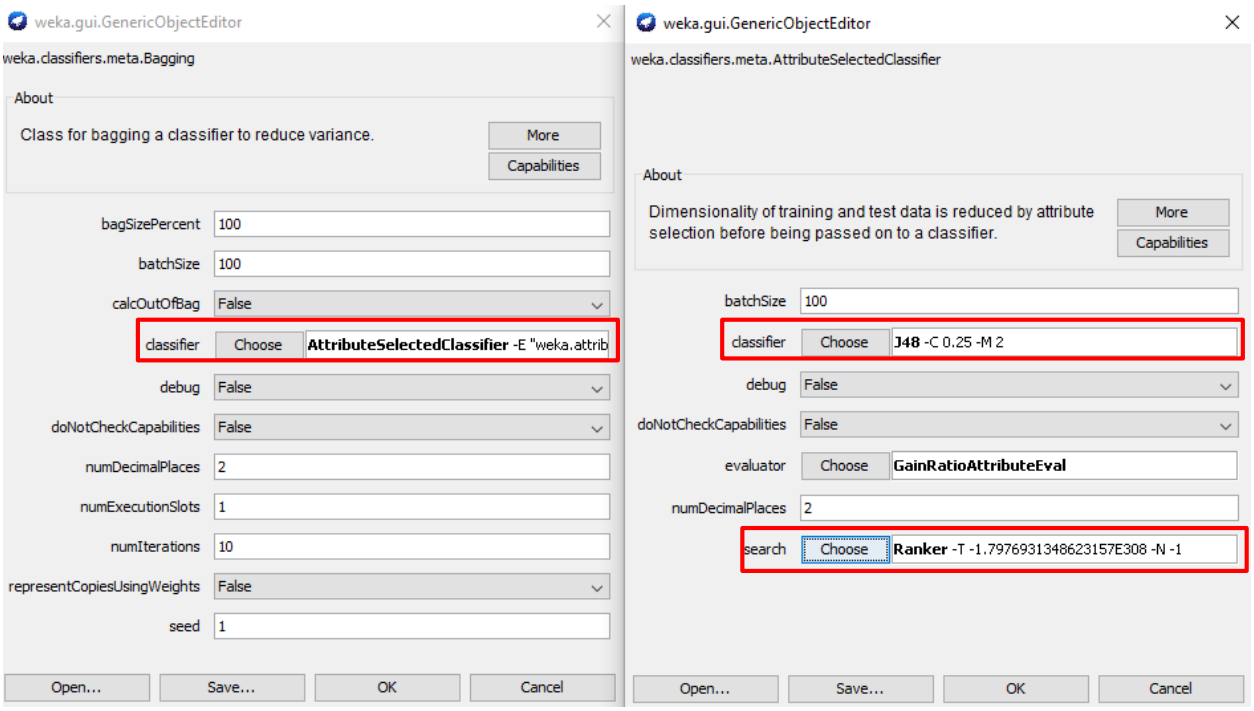


Figure 3.15: Combining Bagging and feature selection

3.9.5. Performance Evaluation Measures

Most performance evaluators in use today focus on the ability of the classifier to correctly identify instances (Sokolova, Japkowicz and Szpakowicz, 2006). The use of percentage accuracy as a measure for evaluation could be problematic due to the fact that real-world applications do not have equal cost of misclassification. The use of classifier accuracy ignores the misclassification cost differences and assumes that all class predictions carry equal weight. For instance, for this dataset, the misclassification cost for the incorrectly classifying customers who did not subscribe to a term deposit is lower than the cost of incorrectly classifying customers who actually did subscribe to the term deposit. For such cases where the cost of misclassification varies, the use of other evaluation measure which considers the true positive, false negative, true negative and false positive rate is important in order to have a representative view of both classes. The use of Precision, Recall, ROC and the confusion matrix as a means of performance evaluation helps to put all of these categories into consideration. These evaluation measures are considered below:

1) Confusion Matrix: The classification made by a classification system can be visually represented using the confusion Matrix. The confusion matrix shows the number of correctly and incorrectly classified instances for each of the class labels. The confusion matrix comprises of four sections, the true positive, false positive, false negative and true negative. The true positive represents the number of positive instances which were correctly classified as positive, the false negative represents the number of positive instances which were wrongly classified as negative, the true negative represents the number of negative instances which were correctly classified as negative while the false positive represent the number of negative instances which were wrongly classified as positive. In this dataset, the true positive instances represent customer who were correctly classified as subscribing to the bank's term deposit while the true negative represents customer who actually did not subscribe to a term deposit.

Table 3.1: The Confusion Matrix

	Predicted value	Predicted value
Actual value	True Positive	False Negative
Actual value	False Positive	True Negative

2) Precision: This is used to evaluative the predictive power of a classifier. It represents the ability of the classifier to correctly classify instances. It represents the number of correctly classified positive instances divided by the total number of correctly and incorrectly classified positive instances in the dataset.

3) Recall: Recall refers to the ability of the classifier to remember all the positive instances. It represents the number of correctly classified positive instances divided by the overall number of positive instances in the dataset.

4)Percentage Accuracy: this is used to evaluate the overall performance of a classifier. It provides a value which represents the performance of the classifier measured in percentage. It shows the percentage value for which the class labels were accurately classified.

5) ROC: The Receiver Operator Characteristics (ROC) is a good performance evaluator when dealing with dataset with an imbalance and unequal classification error cost (Fawcett, T, 2006). The ROC graph is a useful tool when comparing the performance of various classification algorithms, it shows the relationship between the specificity and sensitivity of the classifier. Sensitivity refers to the ability of the classification algorithm to accurately identify and classify the true positive instances while specificity refers to the ability of the classification algorithm to accurately identify and classify the true negative instances. It also provides a method for visualizing, classifying and selecting classifiers based on their performance (Fawcett, T, 2006).

3.9.2. Association Rule Mining

The goal of association mining is to discover pattern of occurrence among itemset. The association rule algorithm helps to discover rules that satisfy certain specified constraints. The most common association rule constraint is Support (Zheng, Kohavi and Mason, 2001). Support refers to the number of instances within a dataset which satisfies an item set consisting of both the Left hand side and the right hand side (Zheng, Kohavi and Mason, 2001). In association rule mining, the confidence metric is also used to specify constraints for an association rule. Confidence refers to the ratio of the support of a rule to the support of the left hand side of an itemset. Zheng, Kohavi and Mason (2001) identified two major steps taken by association algorithms to generate a rule; the first involves the selection of frequent itemset while the second step involves the construction of the association rules using these identified frequent itemset.

The Apriori algorithm is one of the most popular methods used for mining frequent itemset in a transactional database (Lin, Lee and Hsueh, 20102). The algorithm has the ability to generate both the frequent itemset and the association rules (Zheng, Kohavi and Mason, 2001). The algorithm first generates a set of frequent itemset which satisfies a minimum support constraint and then constructs rules which utilizes these itemset based on a specified minimum confidence level.

The Apriori algorithm was used for the association rule mining of the dataset. The minimum support and confidence level was set up on Weka's interface. The class association rule mining

was carried out to help develop association rules that are specific to the class labels as opposed to the use of a general classification rule mining which just identifies general rules from an itemset. This was done to help discover specific rules guiding reasons why customer would subscribe to a term deposit and the likely factors contributing to the reasons why others customers might not subscribe. The default minimum support and confidence level on Weka's Aprirori interface was used. A minimum confidence metric of 0.9, an upper bound minimum support of 1.0 and a lower bound minimum support of 0.1 was used.

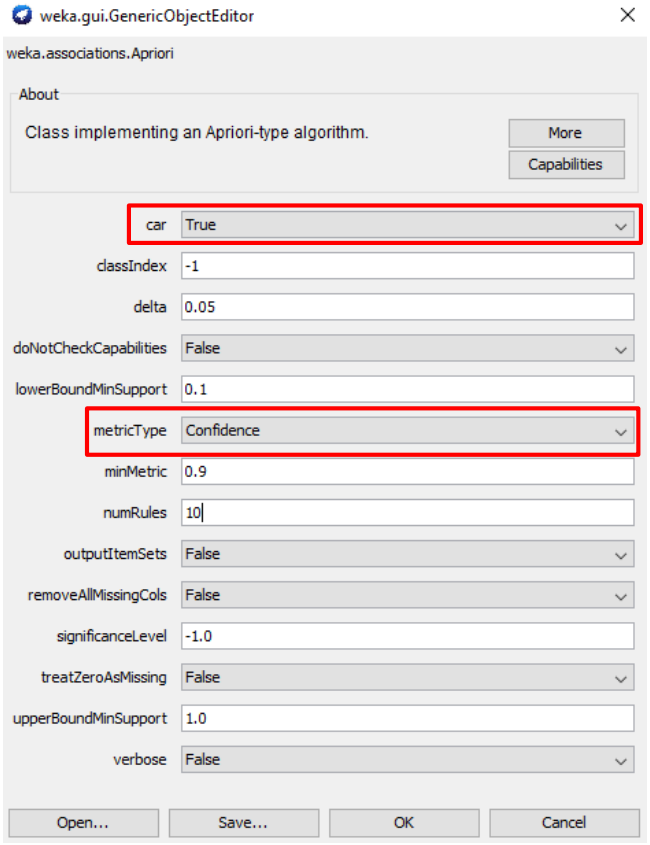


Figure 3.16: Class association rule mining using the Aprirori algorithm

CHAPTER 4

4. RESULT AND ANALYSIS

This chapter focuses on the results generated through the analysis of the dataset. It provides insight into how the research questions were being answered and also how the research objectives were achieved. It explains how the research objective of identifying the best method of producing a classifier with a higher predictive capability could help in improving the likelihood of customers subscribing to a term deposit. The use of the data preprocessing methods, the algorithmic method and the feature selection method was adopted to address the research questions. The results for this section was presented in the form of tables. This chapter also covers the results of the predictions produced through the use of the selected classification algorithms and the rules derived through the use of association rule mining. A combination of the classification and association rule learning aspect of the study was considered as being essential so that models which are capable of effectively making a considerable high level of prediction was achieved as well as to help discover patterns and rules which could help to determine what group of customers are likely to subscribe to a term deposit.

The result and analysis of this section was divided into seven different scenarios comprising of the major methods adopted to analysing and interpreting the result. These scenarios are identified below.

- Scenario 1: Applying the single classifier method for the classification task.
- Scenario 2: Analysing the effect of applying the algorithmic method (Bagging, Boosting and Stacking)
- Scenario 3: Comparing the result of each of the ensembles to the single classifier.
- Scenario 4: Comparing the results of the ensembles methods to each other.
- Scenario 5: Analysing the result of the manual and automatic feature selection task.
- Scenario 6: Combining the algorithmic method with the feature section task
- Scenario 7: Interpretation of the interesting association rules

4.1. Single Classifier Method

The dataset used here has already been undersampled by using the Spread Subsample filter to achieve an equal amount of class distribution. The 10-fold cross validation test option was used in which the dataset was partition into 10 subsamples and each for each of the iterations, one of the subsample is retained for testing purpose while the remaining 9 subsamples act as the training set. The result in table 4.1 below shows the result achieved applying the selected classifiers on the already pre-processed dataset

Table 4.1: Single classifier method

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.920	0.871	0.942	0.904	0.860	0.873
Precision	0.892	0.796	0.886	0.834	0.864	0.875
Recall	0.890	0.795	0.884	0.833	0.860	0.873
Percentage accuracy (%)	88.97	79.49	88.35	83.28	86.02	87.33

From table 4.1 above, the Random Forest classifier currently has the best performance with an ROC of 0.942 with the J48 and Multilayer perceptron also performing relatively well. The SVM classifiers currently have the least performance. The random forest classifier is known to perform well on large datasets due to its ability to grow several unpruned trees and aggregate their predictions by selecting a subset of features to determine the split point.

4.2. Ensembles Based Method

The use of the algorithm method encompasses the use of the ensemble methods to improve the accuracy of the single classifiers used. The results generated for the bagging, boosting and stacking technique was compared to the result of the single classifiers shown in table 4.1 above which indicates the result gotten for each individual classifier in its default state.

Table 4.2: Bagging Result

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.941	0.867	0.938	0.927	0.893	0.894
Precision	0.894	0.788	0.887	0.867	0.861	0.875
Recall	0.890	0.787	0.883	0.866	0.856	0.874
Percentage accuracy (%)	89.032	78.65	88.27	86.56	85.58	87.36

Table 4.3: Boosting Result

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.927	0.916	0.942	0.871	0.899	0.912
Precision	0.868	0.852	0.867	0.836	0.838	0.876
Recall	0.867	0.852	0.866	0.836	0.838	0.875
Percentage accuracy (%)	86.71	85.22	86.58	83.59	83.80	87.46

4.2.1. Comparing the effect of applying the Bagging technique

The result shown in Table 4.1 and Table 4.2 indicates that bagging increased the ROC value of the classifier's slightly but there was a decrease in the ROC of Naïve Bayes classifier from 0.871 to 0.867. The decrease in the performance of the Naïve Bayes classifier can be attributed to the fact that the Naïve Bayes classifier is a stable algorithm and the bagging method has been known to work best on unstable algorithms such as the decision tree and neural networks. Bagging a stable classifier can lead to a reduction in the performance of the learning algorithm (Deng, Jin and Zhong, 2005). The bagging method works best with unstable algorithms due to the fact that bagging uses the bootstrap aggregating technique and the ability of the learner to respond to changes in the training set is a key requirement for the bagging to be effective. For the J48 classifier, an increase in performance was recorded due to the fact that the process of bagging a J48 classifier leads to similar result as a Random forest classifier because a RF classifier contains bagged decision trees. The Random forest classifier is already a form of bagging containing several bagged trees therefore applying another bagging algorithm would most likely not lead to an improvement in its performance as observed in the result in table 4.1 and 4.2. Bagging helps to reduce variance which in turn leads to better prediction performance of the learning algorithm.

```
Correctly Classified Instances        8262               89.0302 %
Incorrectly Classified Instances      1018               10.9698 %
Kappa statistic                          0.7806
Mean absolute error                      0.1622
Root mean squared error                  0.2894
Relative absolute error                 32.4483 %
Root relative squared error             57.88   %
Total Number of Instances             9280

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.936     0.155      0.858     0.936     0.895      0.941    yes
                 0.845     0.064      0.929     0.845     0.885      0.941    no
Weighted Avg.    0.89      0.11       0.894     0.89      0.89       0.941

=== Confusion Matrix ===

   a    b   <-- classified as
 4341  299 |    a = yes
  719 3921 |    b = no
```

Figure 4.0: Confusion Matrix of the bagged J48 classifer

Figure 4.0 shows the confusion matrix of the bagged j48 classifier where the number of actual and predicted values of the classification task are recorded. The confusion matrix shows that out of 4640 instances belonging to the Yes class, the algorithm was able to correctly predict 4341 of those intances while 299 instances were misclassified. For the No Class, the algorithm was able to accurately predict 3921 instances while 719 instances were misclassified.

4.2.2. Comparing the effect of applying the Boosting technique

Comparing the result of table 4.1 to that of table 4.3 above shows that applying the AdaBoost algorithm to the Support vector machines classifiers brought about a significant amount of increase in their ROC. Rangel, Lozano and García 2005), (Wickramaratna, Holden and Buxton 2001) argue that using the SVM classifiers with the boosting method appears to be counterproductive due to the fact that the AdaBoost is used to create a strong classifier by using a combination of a set of weak classifiers and the SVM is considered a strong classifier which is usually difficult to train therefore applying the boosting technique would lead to a reduction in the classifier's performance. For instance, Rangel, Lozano and García (2005) proposed that the use of a weakened SVM with the Adaboost would lead to better performance than the single SVM trained in this manner but from the result obtained in table 4.3, applying the Adaboost with the LibSVM and the SMO still lead to a significant improvement in performance. Boosting the Naïve Bayes classifier led to a significant amount of increase in its performance when compared to the single Naïve Bayes classifier result in table 4.1 with the ROC increasing from 0.871 to 0.916. The improvement in the performance of the Naïve Bayes classifier can be attributed to the fact that the Naïve Bayes is an example of a model with high bias but low variance and the process of applying Adaboost helps to produce better results by placing more weights on the incorrectly classified instances to ensure that the chances of them being predicted correctly is increased. Boosting works well for models with high bias but low variance because models with high bias tend to have higher number of misclassifications due to the fact that the models tend to focus more on making the target class easier to learn. Therefore, the boosting process helps to assign more weights to these incorrectly classified instances so that the performance of the base model is improved. Boosting a Naïve Bayes classifier leads to excellent generalization and learning ability of the classifier (Elkan, 1997).

4.2.3. Comparing The Results of Bagging and Boosting

The boosting method was considered more effective in improving the ROC of the classifiers, most especially based on the result of the LibSVM, SMO and the naïve Bayes classifier. The bagging technique produced a better ROC for the Multilayer Perceptron when compared with that of boosting, the bagging result shows an ROC OF 0.927 for MLP while Boosting produced a lesser result of 0.871.

4.2.4. Result and analysis of the effect of applying the stacking technique

When deciding on what approach to use in selecting the candidate algorithm, the best method to use is by testing the classifier on the problem to be worked on and then the classifiers that performs best is selected (Seewald, 2002.). In other to have a better understanding of the how the process of combining multiple classifier can improve our predictive accuracy, the classifiers used for this task were not restricted to just the best performing classifiers as indicated by the results shown in table 4.1 but all of the six selected classifiers used in this project were involved in the stacking task. The stacking method was implemented by carrying series of test using different combination techniques. The 2 classifier, 3 classifiers and 4 classifier test was conducted. For the 2 and classifier test, the classifiers used was restricted to Naïve Bayes, Random Forest and the j48 classifier. At each point of the test process, the positions of the meta and the base leaner was interchanged. For instance, the first test used the Naïve Bayes as the Base learner while the J48 was used as the meta learner. This was interchanged in the second test carried. For the 4 classifier stacking, strong classifiers such as the MLP, LibSVM and SMO was introduced as the meta learners. The result of the stacking test is shown in Table 4.4 to Table 4.6 below

Table 4.4: 2 classifier stacking result

Base Learners	Meta Learner	ROC	Precision	Recall
Naïve Bayes	J48	0.802	0.796	0.794
J48	Naïve Bayes	0.922	0.892	0.889
J48	Random Forest	0.916	0.863	0.863
Random Forest	J48	0.888	0.884	0.880
Naïve Bayes	Random Forest	0.818	0.723	0.723

Table 4.5: 3 classifier stacking result

Base Learners	Meta Learner	ROC	Precision	Recall
Naïve Bayes + J48	Random Forest	0.916	0.858	0.857
Naïve Bayes + Random Forest	J48	0.893	0.883	0.880
Random Forest + J48	Naïve Bayes	0.944	0.892	0.891

Table 4.6: 4 classifier stacking result

Base Learners	Meta Learner	ROC	Precision	Recall
Random Forest + Naïve Bayes + J48	SMO	0.890	0.893	0.890
Random Forest + SMO + J48	Naïve Bayes	0.944	0.889	0.887
Random Forest + Naïve Bayes + J48	LibSVM	0.890	0.893	0.890
Random Forest + Naïve Bayes + J48	Multilayer Perceptron	0.941	0.894	0.890

From the result shown in table 4.4 containing a single base learner and a meta learner, the ROC value obtained when using the naïve Bayes classifier as the base learner recorded a lower ROC, precision and recall value when compared to the result of using better performing classifiers such as the j48 and Random Forest classifier. The reason for this performance can be attributed to the fact that the Naïve Bayes classifier on its own as shown in table 4.1 only performed slightly well therefore the process of using a meta learner such as the j48 would not bring about an improvement in its performance since the j48 classifier takes in the output predictions made by the Naïve Bayes classifier to carry out its own final prediction.

Further test shows that combining other classifiers improved the performance of the classifiers. For instance, in Table 4.4 where Naïve Bayes was used as a base learner and random Forest as a meta learner an ROC of 0.802 was recorded but when the j48 classifier was added as the second base learner, the performance increased to 0.912 as shown in table 4.5. The improvement in the performance of the classifiers was noticed for the 2 and 3 classifier stacking approach. When the SVM classifiers were introduced as a meta learner in table 4.6, there was a reduction in the performance of the classifiers. When adopting the stacking technique, the algorithm uses the output of the base learners to generate the training data for the meta learner. Therefore, the combiner algorithm chosen as the meta learner should be able to handle the final predictions made using the combined predictions of the base learner. The result gotten for this section indicates that the wrong choice of the meta leaner could lead to the underutilization of the benefits of adopting the stacking approach. For instance, in table 4.6 where SMO was used as a meta learner, ROC of 0.890 was recorded but when SMO was included as part of the base learners and introducing the j48 algorithm instead as a meta learner, a better ROC performance of 0.944 was recorded.

4.3. Result and analysis of the feature selection process

The feature selection task was conducted in two phases, the manual and automatic feature selection.

4.3.1. Result of the effect of using the manual feature selection process

The manual feature selection was done due to the fact that the automatic feature selection method chooses what attributes to include in the classification task through the use of a statistical method. The manual feature selection was used to help have control over what attributes are used for the classification task. Table 4.7 to 4.9 below shows the result gotten by splitting the dataset into three partitions where each partition contains variables belonging to similar domain. For instance, partition 1 contains information about the customer, the second partition contains information about the bank's marketing campaign and the third partition contains economic and social variable. The classification results of each of the partition is shown in Figure 4.1 to 4.3.

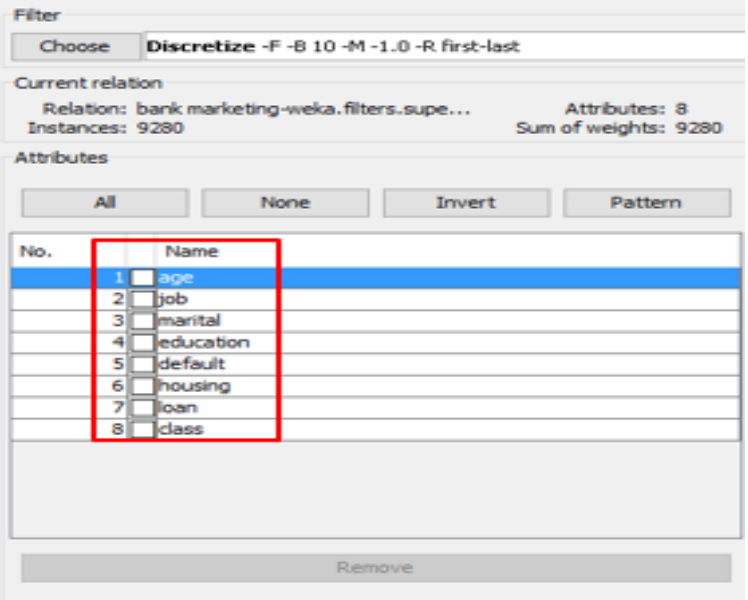


Figure 4.1: Conducting feature selection task using demographics related attributes

Table 4.7: Result of feature selection using demographics related attributes

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.646	0.657	0.612	0.628	0.602	0.586
Precision	0.613	0.607	0.583	0.591	0.602	0.586
Recall	0.613	0.607	0.583	0.582	0.602	0.586
Percentage of correctly classified instances (%)	61.27	60.69	58.33	58.23	60.18	58.60

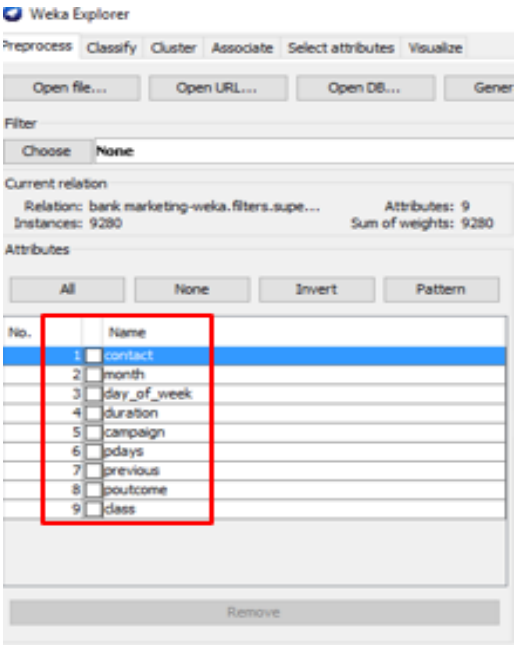


Figure 4.2: Conducting feature selection task using bank campaign history related attributes

Table 4.8: Result of feature selection using bank campaign history related attributes

Performance Measurement	**Classification Algorithms**					
	J48	**Naïve Bayes**	**Random Forest**	**Multilayer Perceptron**	**LibSVM**	**SMO**
ROC	0.904	0.868	0.900	0.915	0.787	0.835
Precision	0.850	0.776	0.821	0.844	0.788	0.835
Recall	0.850	0.736	0.821	0.843	0.787	0.835
Percentage of correctly classified instances (%)	84.99	73.62	82.10	84.31	78.73	83.49

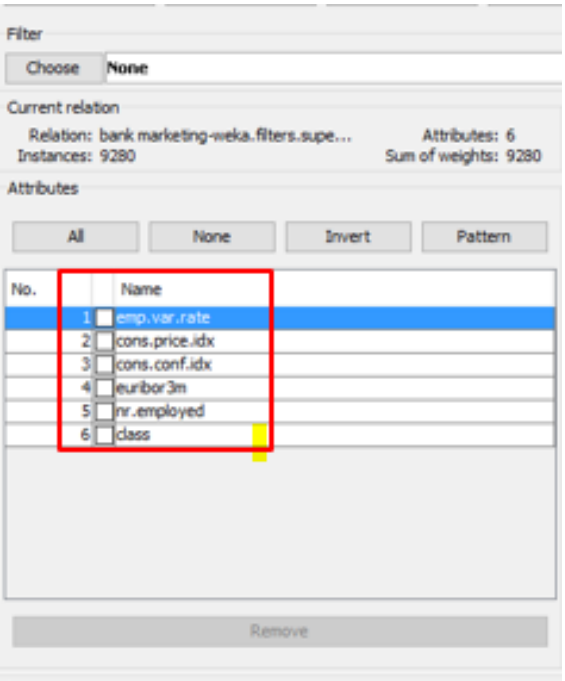


Figure 4.3: Conducting feature selection task using social and economic related attributes

Table 4.9: Result of feature selection using social and economic related attributes

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.771	0.748	0.793	0.779	0.738	0.719
Precision	0.755	0.719	0.756	0.744	0.756	0.719
Recall	0.746	0.719	0.739	0.734	0.738	0.719
Percentage of correctly classified instances (%)	74.57	71.85	73.88	73.43	73.84	71.86

4.3.2. Analysis of the effect of using the manual feature selection process

From the results shown in table 4.7 to table 4.9, the use of demographics related data as shown in table 4.7 was not good at classifying the instances. All of the results recorded for the classifiers had low ROC, Precision, recall and percentage accuracy. This indicates that the demographics related attributes cannot be used as a major determining factor to identify customers subscribing who are most likely to subscribe to a term deposit. The second table as shown in table 4.8 above shows that attributes containing information regarding the personal effort put by the bank into marketing campaign provides the best means of classifying the instances. This was indicated by the performance of the classifiers with their ROC ranging from 0.787 to 0.915. The third partition containing economic and social related attributes also performed relatively well when compared with the performance for the demographics related attributes. The performance of all of the classifiers in the second partition outperformed those in partition 1 and 3 when compared with the results gotten for the same classifiers. The Multilayer Perceptron classifier had highest overall accuracy of 84.31, ROC of 0.915, precision 0.843 and recall of 0.844 with the J48 classifier having the second position with an accuracy of 84.99%, ROC of 0.904, Precision of 0.850 and Recall of 0.850.

4.3.3. Result of the effect of using the automatic feature selection process

The automatic feature selection task was executed by using the Wrapper and the filter method. For the filter method, the gain ratio evaluation measure was used to determine what subsets of attributes would be used by the classifier. For the filter method, the ranker search method was used to select the best eight attributes by evaluating the worth of each of the attributes by using the measurement of gain ratio. The evaluation of the performance of the filter and wrapper based feature selection was restricted to the use of just the J48, Naïve Bayes and Random Forest classifier. The SVM classifiers and the Multilayer perceptron were exempted from this task due to their individual computational intensiveness when combined with the wrapper methods which on its own is also highly computational demanding. The wrapper method is more computationally intensive than the filter method because it trains and evaluates the classification model for each subset of features (Saeys, 2004). The wrapper based method uses the prediction made by the classification algorithm to determine the subset of

features that would be used in the final classification task whereas the filter method works independent of the performance of the classification algorithm in determining the feature subsets. To allow for proper comparison between both methods, the researcher resorted to using the same classification algorithms for both tasks thereby eliminating the SVM classifiers and the neural network classifier. The result gotten for both feature selection methods are shown in table 4.10 and 4.11

Table 4.10: Feature selection result using the wrapper method

Performance Measurement	Classification Algorithms		
	J48	Naïve Bayes	Random Forest
ROC	0.927	0.900	0.930
Precision	0.894	0.854	0.884
Recall	0.891	0.851	0.8811
Percentage of correctly classified instances (%)	89.07	85.12	87.88

```
Ranked attributes:
 0.1449   13 pdays
 0.0816   15 poutcome
 0.0757   20 nr.employed
 0.0754   11 duration
 0.0702   16 emp.var.rate
 0.0585   19 euribor3m
 0.0575   18 cons.conf.idx
 0.0548   14 previous
```

Figure 4.4: 8 ranked attributes using gain ratio evaluation

Table 4.11: Feature selection result using the filter method

Performance Measurement	Classification Algorithms		
	J48	Naïve Bayes	Random Forest
ROC	0.930	0.886	0.935
Precision	0.893	0.792	0.873
Recall	0.888	0.791	0.872
Percentage of correctly classified instances (%)	88.84	79.08	87.21

4.3.4. Analysis of the effect of using the automatic feature selection process

The wrapper based and filter method used had similar results with the performance of the Random Forest classifier. Figure 4.4 above shows the ranking of the eight attributes selected for the classification task by measuring the gain ratio of each of the attributes with respect to the class

4.4. Combining Feature selection and Algorithmic Method

To further work towards improving the results gotten from the use of a single classifier, the feature selection process and the use of the ensembles method, the feature selection task was used in combination with the ensembles method. The results derived from this task is shown in table 4.12 to table 4.14.

Table 4.12: Result of combining feature selection and boosting

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.931	0.919	0.931	0.927	0.894	0.914
Precision	0.871	0.851	0.881	0.878	0.836	0.851
Recall	0.871	0.851	0.880	0.875	0.835	0.851
Percentage accuracy (%)	87.07	85.14	87.96	87.47	83.55	85.05

Table 4.13: Result of combining feature selection and bagging

Performance Measurement	Classification Algorithms					
	J48	Naïve Bayes	Random Forest	Multilayer Perceptron	LibSVM	SMO
ROC	0.943	0.885	0.940	0.943	0.902	0.863
Precision	0.888	0.790	0.881	0.883	0.860	0.850
Recall	0.886	0.989	0.880	0.880	0.856	0.849
Percentage accuracy (%)	88.57	78.87	88.01	88.01	85.63	84.94

Table 4.14: 4 classifier stacking result

Base Learners	Meta Learner	ROC	Precision	Recall
Random Forest + Naïve Bayes + J48	SMO	0.889	0.894	0.889
Random Forest + SMO + J48	Naïve Bayes	0.942	0.887	0.885
Random Forest + Naïve Bayes + J48	LibSVM	0.890	0.896	0.890
Random Forest + Naïve Bayes + J48	Multilayer Perceptron	0.941	0.894	0.889

4.4.1. Analysing the effect of combining feature selection task with the ensembles method

The results derived from the combination of the feature selection and ensembles methods used in this research shows that this method helps to improve the existing results gotten when the feature selection task and the use of ensembles were performed in isolation. The process of removing redundant or irrelevant attributes within the dataset before performing the task of combining different classifiers together brought about an improvement in the performance of the algorithms. The results gotten in table 4.12 shows that using the feature selection approach within the Adaboost algorithm is an effective way to improve the classification performance of the models used. A significant amount of improvement was observed for the Multilayer

Perceptron (MLP) when introducing the feature selection task into the boosting process. The result shows an increase in the percentage accuracy and ROC of the MLP after applying the feature selection method with the boosting process. This is shown in table 4.3 and table 4.12.

4.6. Association Rule Mining

The association rule mining was conducted by using the Aprirori algorithm which allows the metric type and minimum metric size to be selected. For class association rule mining, the only metric type permitted when using Weka is the confidence metric.

No rules were found at a confidence value greater than 0.8 but for a minimum confidence of 0.8, most of the rules generated were related to the classification of the NO class which represents customers who didn't subscribe to the bank's deposit. Since more focus should be on customers who would subscribe to a term deposit. A further test was carried out by increasing the number of rules to 100 so that discovery can be made for association rules where the Yes class is being represented. The first classification of the Yes class came up at a confidence value of 0.64.

```
Best rules found:

 1. emp.var.rate=(0.5-1.25] 1216 ==> class=no 976    conf:(0.8)
 2. cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
 3. contact=telephone emp.var.rate=(0.5-1.25] 1216 ==> class=no 976    conf:(0.8)
 4. contact=telephone cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
 5. month=may emp.var.rate=(0.5-1.25] 1216 ==> class=no 976    conf:(0.8)
 6. month=may cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
 7. month=may cons.conf.idx=(-37.9--36.25] 1216 ==> class=no 976    conf:(0.8)
 8. month=may nr.employed=5138.7-5193 1216 ==> class=no 976    conf:(0.8)
 9. pdays=514_max emp.var.rate=(0.5-1.25] 1216 ==> class=no 976    conf:(0.8)
10. pdays=514_max cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
```

Figure 4.5: Best Rules found for No class in the discovered association rules

```
 95. month=may pdays=514_max 2416 ==> class=no 1650    conf:(0.68)
 96. month=may 2545 ==> class=no 1659    conf:(0.65)
 97. default=no contact=cellular campaign=0_1.5 2831 ==> class=yes 1807    conf:(0.64)
 98. default=no loan=no contact=cellular campaign=0_1.5 2348 ==> class=yes 1495    conf:(0.64)
 99. default=no housing=yes loan=no contact=cellular 2642 ==> class=yes 1650    conf:(0.62)
100. default=no housing=yes contact=cellular 3178 ==> class=yes 1976    conf:(0.62)
101. education=university.degree contact=cellular 2352 ==> class=yes 1438    conf:(0.61)
102. default=no loan=no contact=cellular 4812 ==> class=yes 2941    conf:(0.61)
103. default=no contact=cellular 5840 ==> class=yes 3562    conf:(0.61)
104. contact=cellular campaign=0_1.5 3196 ==> class=yes 1948    conf:(0.61)
105. loan=no contact=cellular campaign=0_1.5 2655 ==> class=yes 1614    conf:(0.61)
106. emp.var.rate=(-2.35--1.75] 2429 ==> class=yes 1461    conf:(0.6)
107. marital=married previous=0_1 4285 ==> class=no 2560    conf:(0.6)
108. marital=married poutcome=nonexistent 4285 ==> class=no 2560    conf:(0.6)
109. marital=married pdays=514_max previous=0_1 4285 ==> class=no 2560    conf:(0.6)
110. marital=married pdays=514_max poutcome=nonexistent 4285 ==> class=no 2560    conf:(0.6)
111. marital=married previous=0_1 poutcome=nonexistent 4285 ==> class=no 2560    conf:(0.6)
112. marital=married pdays=514_max previous=0_1 poutcome=nonexistent 4285 ==> class=no 2560    conf:(0.6)
113. marital=married loan=no previous=0_1 3546 ==> class=no 2111    conf:(0.6)
114. marital=married loan=no poutcome=nonexistent 3546 ==> class=no 2111    conf:(0.6)
115. marital=married loan=no pdays=514_max previous=0_1 3546 ==> class=no 2111    conf:(0.6)
116. marital=married loan=no pdays=514_max poutcome=nonexistent 3546 ==> class=no 2111    conf:(0.6)
117. marital=married loan=no previous=0_1 poutcome=nonexistent 3546 ==> class=no 2111    conf:(0.6)
118. marital=married loan=no pdays=514_max previous=0_1 poutcome=nonexistent 3546 ==> class=no 2111    conf:(0.6)
119. default=no housing=no contact=cellular 2529 ==> class=yes 1502    conf:(0.59)
120. housing=yes loan=no contact=cellular 3013 ==> class=yes 1781    conf:(0.59)
```

Figure 4.6: Identification of the Yes class in the discovered association rules

To help discover better rules with a high level of support and confidence, the dataset was divided into three partitions.

The first partition consists of attributes relating to the information of the customer. This includes age, job, marital status, education, credit default status, housing loan status and personal loan status.

The second partition contains data relating to contact made and campaign related information. This include the contact type, the last contact month of the year, the last contact day of the week, the duration of last contact call, the number of times a customer was contact during the campaign, the interval between the number of days before the customer was contacted from previous campaign, the number of contacts made before the campaign exercise, and the outcome of the last marketing campaign.

The third partition contains information relating to social and economic based attributes such as the employment variation rate, the consumer price index, consumer confidence index, euribor rate and the number of employees.

The association rules generated from each of the partition shows that rules with a higher confidence are generated when dealing with information relating to the contact data and the information regarding the result of the efforts put into the bank's marketing campaigns. The highest confidence generated for any of the rules in the first partition was 0.56, while the highest confidence generated for the rules in the second partition and third partition was observed to be 0.8. This further helps to confirm the performance of the classifiers in the manual feature selection task which shows that the use of demographics related attributes are not the best options to use when identifying customers with better likelihood of subscribing to the term deposit. Figure 4.7 to Figure 4.9 below shows the support and confidence for the 10 best rules derived for the three partitions.

```
Best rules found:

 1. default=no housing=yes loan=no 3396 ==> class=yes 1909    conf:(0.56)
 2. default=no housing=yes 4087 ==> class=yes 2282    conf:(0.56)
 3. default=no loan=no 6439 ==> class=yes 3478    conf:(0.54)
 4. default=no 7784 ==> class=yes 4197    conf:(0.54)
 5. marital=married 5379 ==> class=no 2847    conf:(0.53)
 6. marital=married loan=no 4445 ==> class=no 2347    conf:(0.53)
 7. housing=yes loan=no 4016 ==> class=yes 2098    conf:(0.52)
 8. housing=no 4236 ==> class=no 2210    conf:(0.52)
 9. housing=no loan=no 3655 ==> class=no 1903    conf:(0.52)
10. housing=yes 4831 ==> class=yes 2507    conf:(0.52)
```

Figure 4.7: Association rules discovered when using the demographics related attributes

```
Best rules found:

 1. contact=telephone month=may pdays=514_max 1303 ==> class=no 1041    conf:(0.8)
 2. contact=telephone month=may previous=0_1 1285 ==> class=no 1023    conf:(0.8)
 3. contact=telephone month=may poutcome=nonexistent 1285 ==> class=no 1023    conf:(0.8)
 4. contact=telephone month=may pdays=514_max previous=0_1 1285 ==> class=no 1023    conf:(0.8)
 5. contact=telephone month=may pdays=514_max poutcome=nonexistent 1285 ==> class=no 1023    conf:(0.8)
 6. contact=telephone month=may previous=0_1 poutcome=nonexistent 1285 ==> class=no 1023    conf:(0.8)
 7. contact=telephone month=may pdays=514_max previous=0_1 poutcome=nonexistent 1285 ==> class=no 1023    conf:(0.8)
 8. contact=telephone month=may 1314 ==> class=no 1042    conf:(0.79)
 9. contact=telephone previous=0_1 2479 ==> class=no 1792    conf:(0.72)
10. contact=telephone poutcome=nonexistent 2479 ==> class=no 1792    conf:(0.72)
11. contact=telephone pdays=514_max previous=0_1 2479 ==> class=no 1792    conf:(0.72)
12. contact=telephone pdays=514_max poutcome=nonexistent 2479 ==> class=no 1792    conf:(0.72)
13. contact=telephone previous=0_1 poutcome=nonexistent 2479 ==> class=no 1792    conf:(0.72)
14. contact=telephone pdays=514_max previous=0_1 poutcome=nonexistent 2479 ==> class=no 1792    conf:(0.72)
15. contact=telephone pdays=514_max 2546 ==> class=no 1830    conf:(0.72)
16. contact=telephone 2624 ==> class=no 1837    conf:(0.7)
```

Figure 4.8: Association rules discovered when using the previous campaign history attributes

```
Best rules found:

 1. emp.var.rate=(0.5-1.25] 1216 ==> class=no 976    conf:(0.8)
 2. cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
 3. emp.var.rate=(0.5-1.25] cons.price.idx=(93.956-94.0105] 1216 ==> class=no 976    conf:(0.8)
 4. emp.var.rate=(0.5-1.25] cons.conf.idx=(-37.9--36.25] 1216 ==> class=no 976    conf:(0.8)
 5. emp.var.rate=(0.5-1.25] nr.employed=5138.7-5193 1216 ==> class=no 976    conf:(0.8)
 6. cons.price.idx=(93.956-94.0105] cons.conf.idx=(-37.9--36.25] 1216 ==> class=no 976    conf:(0.8)
 7. cons.price.idx=(93.956-94.0105] nr.employed=5138.7-5193 1216 ==> class=no 976    conf:(0.8)
 8. cons.conf.idx=(-37.9--36.25] nr.employed=5138.7-5193 1216 ==> class=no 976    conf:(0.8)
 9. emp.var.rate=(0.5-1.25] cons.price.idx=(93.956-94.0105] cons.conf.idx=(-37.9--36.25] 1216 ==> class=no 976    conf:(0.8)
10. emp.var.rate=(0.5-1.25] cons.price.idx=(93.956-94.0105] nr.employed=5138.7-5193 1216 ==> class=no 976    conf:(0.8)
```

Figure 4.9: Association rules discovered when using the social and economic attributes

4.6.1. Interestingness Of The Association Rule

The association rules generated by using the Aprirori algorithm were evaluated using the confidence and support evaluation metric. The measure of interestingness of a rule was determined by examining the evaluation metric and the generated association rule. A rule was considered interesting if it has a high level of support and confidence and also has the potential to help improve the bank telemarketing process in making more targeted campaigns. A few of the rules generated were considered slightly interesting and are presented in the table 4.14. It was discovered that an interesting relationship exists between customer loan status, contact method (cellular or telephone) and the number of contacts made to a customer during a campaign period. The association rule shows that customers with no credit loan status who were contacted using a cellular communication type and who were not contacted or contacted just once during a campaign period recorded a higher probability of subscribing to the bank's term deposit.

The association rule also shows that customers who are educated to the university degree level and owns a mobile phone stand better chances of subscribing to a term deposit as opposed to a less educated and fixed telephone line user.

Some interesting association rules were also discovered for customers who did not subscribe to a term deposit. It was observed that the economic and social variables were also a determining factor as to whether a customer would subscribe to a term deposit. It was observed that for

cases where the consumer price index was greater than 94 and the customer is a telephone user, the customers did not subscribe to a term deposit. In the concept of economics, an increase in the Consumer Price Index(CPI) represents an increase in the prices of goods and services. An increase in CPI indicates the presence of inflation and a decrease in the consumer price index indicates that there is the presence of deflation in the costs of goods and services. In our research context, a decrease in the CPI increases the probability of customers subscribing to a term deposit. This occurrence can be attributed to the fact that when inflation occurs people are most likely to spend more money on items that used to cost less. This therefore leads to a reduction in the consumer's purchasing power and the bank therefore stands lower chances of getting customers to subscribe to a term deposit. This indicates that the economic condition is an important factor contributing to the likely reasons why a customer would choose to subscribe to a term deposit.

It was also observed from the association rule that when there is a decline in employment rate customers usually tend to subscribe to a term deposit. We could argue that an increase in unemployment rate in the economy could result in people subscribing to a fixed term deposit due to the fear of uncertainties about the future which could lead to improvement in the saving habits of customers. We could also argue that since Portugal is also a country which practices the policy of paying unemployment benefit then people who are unemployed are most likely still able to afford to save some amount of money as opposed to a country where unemployed citizens are not given any form of entitlement.

Table 4.14: Identification of Interesting Association Rules

Rules	Support	Confidence
Default = no contact= cellular campaign =0_1==> **class=yes**	2831==>1807	64%
Default = no loan= no contact= cellular campaign = 0_1==> **class=yes**	2348==>1495	64%
Education=university. degree contact = cellular ==> class= yes	2352 ==>1438	61%
Emp.var.rate= -2.35 _ -1.75 ==> **class=yes**	2493==>1461	60%
Marital= married previous 0_1==> **class=no**	4285==>2560	60%
Emp.var.rate= 0.5 -1.5 ==> **class=no**	1216 ==>976	80%
Cons.price. index = 94_max ==> **class= no**	1216==> 976	80%
Contact = telephone emp.var. rate= 0.5 -1.5 ==> **class=no**	1216==> 976	80%
Contact= telephone Cons.price. index = 94_max ==> **class= no**	1216==> 976	80%
Month = may nr. employed = 5138_5193==> **class=no**	1216==>976	80%

CHAPTER 5

5. CONCLUSION AND RECOMMENDATION

The purpose of this study has been to apply the concept of business analytics in practice to a bank telemarketing dataset. This was achieved by adopting the use of the Weka's machine learning platform to examine how the concept of business analytics can help improve business processes. During this process, the research questions and objectives were met by analysing the dataset. It was identified that the presence of imbalance in the size of the dataset could lead to the underperformance of the learning algorithms therefore the research question was centred around how the data Preprocessing method, the ensembles method and the feature selection method can help to improve the performance of the various selected classifiers. A combination of the ensembles method and the feature selection method was also explored in a quest to getting the most of the performance of the learning algorithms. During this process, several findings were discovered and these are summarized in the points below

5.1. Key Findings

- It was discovered that the use of the ensemble methods is an effective way to improve the performance of the classifiers.
- During the feature selection tasks, it was discovered that the use of demographics related attributes is not a sufficient approach to determine customers with a higher likelihood of subscribing to a term deposit.
- Boosting the Naïve Bayes classifier lead to a significant increase in its performance.
- It is not just enough to select classifiers for stacking. The decision on what classifier is used as the base learner or meta learner could lead to an increase or decrease in the performance process of the stacking method.
- The process of combining the feature selection task with the ensembles approach is considered a more effective approach than just individually applying the feature selection task or the ensembles method.
- The use of previous banking campaign history and social and economic related attributes were the best at determining the predictions made by the classifiers. This was

furthered verified by the use of the wrapper methods where all of the eight selected and ranked attributes were related to one of these categories.

- Customers are most likely to subscribe to term deposit when the condition of the economy is more favourable while there is a decline in the prices of goods and services.
- Customers with no previous history of having a default in credit (failure to meet repayment conditions) status stand better chances of subscribing to the bank's term deposit.

5.2. Limitations

When analysing the performance of a classifier, it is imperative to note that no classifier is universally acceptable as being the best at handling all forms of data mining tasks. Different algorithms respond to certain types of attributes differently and the best performing classifier for a particular dataset could be worst when dealing with an entirely different application domain. Therefore, the possibility of the generalization of the findings of this research to other application domain cannot be evaluated.

While adopting the sampling technique, the researcher only carried out the tests using the undersampling technique which led to loss of information which could have been vital in improving the performance of the classifier. The undersampling method was adopted due to the already large size of the dataset which lead to inability to carry out an effective comparison of the oversampling and undersampling methods. At the initial stage of the implementation process, the researcher attempted to carry out similar tasks done in the undersampling process to the oversampling process but this was terminated due to its computational intensiveness which led to some of the tests done in Weka running for as long 9 hours with still no results generated. The process of comparing the undersampling technique to the oversampling technique would have been useful in helping to better evaluate the sensitivity of the algorithms to changes in the dataset.

5.3. Relating to previous research

Based on most of the literatures examined in chapter 2 and 3, the research further agrees that the use of ensembles are highly effective methods for achieving better classifier performance through the combination of weak classifiers. The research agrees with Dietterich, T.G., (2000) who said that ensembles are able to out-perform single classifiers method.

As indicated by researchers such as (Gao, Khoshgoftaar and Wald, 2014) and (Johansson et al., 2010), we were further able to determine that the process of combining feature selection task with ensembles is a better way of improving the classifiers prediction in which certain feature subsets are chosen by an algorithm to undergo the final class predictions made by the ensemble learners.

We were also able to evaluate that the application of the bagging technique works best with unstable algorithms such as decision tree, RF and Multilayer Perceptron. This behaviour as identified by various researchers can be attributed to the fact that the bagging method works bests with algorithms that can easily respond to changes in the dataset when combining various predictions (Syarif et al., 2012), (Breiman ,1996). As part of this research, we discovered that Boosting the SVM classifiers could be an effective method for improving the classifier's performance contrary to the belief that boosting a strong classifier such as the SVM classifier would lead to a reduction in performance of the classifier. For instance, Wickramaratna, Holden and Buxton (2001) argue that using Adaboost with a strong classifier would lead to performance degradation due to the notion that Adaboost forces a strong learner to focus on hard to classify instances and they suggest weakening the classifier to prevent performance degradation.

5.4. Learning outcomes

- A better understanding on how to evaluate the performance of classifiers using different performance measures.
- Improved understanding on how to combine several machine learning algorithms
- A better understanding on how to interpret and extract interesting association rules

- Improved knowledge on how data mining techniques that can be used to improve productivity in a work environment by proffering solutions through the measurement and discovery of patterns from data.

5.5. Recommendation

The use of other sampling techniques such as the Synthetic Minority Oversampling Technique (SMOTE), Random Oversampling Examples (ROSE) and Random Oversampling Technique (ROS) can be adopted to address the class imbalance problem. A combination of both the undersampling and oversampling method can also be explored for better performance.

The use of other feature selection techniques such as Information Gain, Correlation-based feature selection and Chi-Squared for the filter based methods and also the use of other wrapper evaluation methods such as the genetic algorithms and sequential feature selection algorithms.

Weakening the Support Vector Machine(SVM) classifiers when carrying out boosting to help discover how this can improve the SVM boosting process.

More tests should be done by using parameter tuning so that better performance can be achieved for algorithms by changing the default setting of the algorithms used.

REFERENCES

Abbas, S. (2015) 'Deposit subscribe Prediction using Data Mining Techniques based Real Marketing Dataset'.

Agresti, A. & Kateri, M. (2011) Categorical data analysis. Springer.

Al-Shayea, Q.K. (2013) 'Evaluating Marketing Campaigns of Banking Using Neural Networks'. Proceedings of the World Congress on Engineering, 2013.

Alabau, V et al. (2006) 'The naive Bayes model, generalisations and applications'pp.162.

Alpaydin, E. (2014) Introduction to machine learning. MIT press.

Arlot, S. & Celisse, A. (2010) 'A survey of cross-validation procedures for model selection'. Statistics surveys, pp.40-79.

Bencin, R.L. (1992) 'Build a Telemarketing Blueprint for Success!'. Agency Sales, 22 (8), pp.38.

Breiman, L. (1996) 'Bagging predictors'. Machine learning, 24 (2), pp.123-140.

BV, E.C. (2000) Working in Portugal. Available at: http://www.expatica.com/pt/employment/Unemployment-benefit-in-Portugal_105301.htm (Accessed: 29 August 2016).

BV, E.C. (2000) Unemployment benefit in Portugal. Available at: http://www.expatica.com/pt/employment/Unemployment-benefit-in-Portugal_105301.html (Accessed: 29 August 2016).

Cervantes, J., Li, X., Yu, W. & Li, K. (2008) 'Support vector machine classification for large data sets via minimum enclosing ball clustering'. Neurocomputing, 71 (4), pp.611-619.

Chawla, N.V., Bowyer, K.W., Hall, L.O. & Kegelmeyer, W.P. (2002) 'SMOTE: synthetic minority over-sampling technique'. Journal of artificial intelligence research, pp.321-357.

Chawla, N.V., Lazarevic, A., Hall, L.O. & Bowyer, K.W. (2003) 'SMOTEBoost: Improving prediction of the minority class in boosting'. European Conference on Principles of Data Mining and Knowledge Discovery. Springer, pp.107-119.

Chitra, S.B. (2013) 'Data Mining Techniques and its Applications in Banking Sector'. *International Journal of Emerging Technology and Advanced Engineering*, 3 (8), pp.219-226.

Cieslak, D.A., Chawla, N.V. & Striegel, A. (2006) 'Combating imbalance in network intrusion datasets'. pp.732-737.

Creswell, J.W. (2013) Research design: Qualitative, quantitative, and mixed methods approaches. Sage publications.

Daumé III, H. (2012) 'A course in Machine Learning'. chapter, 5 pp.69.

Davenport, T.H. & Harris, J.G. (2007) Competing on analytics: The new science of winning. Harvard Business Press.

Dietterich, T.G. (2000)'Ensemble methods in machine learning'. International workshop on multiple classifier systems. Springer, pp.1-15.

Dietterich, T.G. (2002) 'Ensemble learning'. The handbook of brain theory and neural networks, 2 pp.110-125.

Drummond, C. & Holte, R.C. (2003). 'C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling'. Workshop on learning from imbalanced datasets II. Citeseer.

Elkan, C. (1997) Boosting and naive Bayesian learning.

Elsalamony, H.A. & Elsayad, A.M. (2013) 'Bank Direct Marketing Based on Neural Network and C5. 0 Models'. International Journal of Engineering and Advanced Technology (IJEAT), 2 (6).

Ezawa, K.J., Singh, M. & Norton, S.W. (1996) 'Learning goal oriented Bayesian networks for telecommunications risk management'. ICML. pp.139-147.

Eze, U.F., Adeoye, O.S. & Ikemelu, C.R.-k. (2014) 'Industry Wide Applications of Data Mining'. International Journal of Advanced Studies in Computers, Science and Engineering, 3 (2), pp.28-37.

Fawcett, T. (2006) 'An introduction to ROC analysis'. Pattern recognition letters, 27 (8), pp.861-874.

Fawcett, T. & Provost, F.J. (1996) 'Combining Data Mining and Machine Learning for Effective User Profiling'. KDD. pp.8-13.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. & Herrera, F. (2012) 'A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches'. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42 (4), pp.463-484.

Gao, K., Khoshgoftaar, T.M. & Wald, R. (2014). 'Combining Feature Selection and Ensemble Learning for Software Quality Estimation'. *FLAIRS Conference*, 2014.

García, S., Luengo, J. & Herrera, F. (2015) Data preprocessing in data mining. Springer.

Graczyk, M., Lasota, T., Trawiński, B. & Trawiński, K. (Year) Published. 'Comparison of bagging, boosting and stacking ensembles applied to real estate appraisal'. Asian Conference on Intelligent Information and Database Systems, 2010. Springer, pp.340-350.

Guyon, I. & Elisseeff, A. (2003a) 'An introduction to variable and feature selection'. Journal of machine learning research, 3 (Mar), pp.1157-1182.

Han, J., Kamber, M. & Pei, J. (2011) Data mining: concepts and techniques. Elsevier.

Hido, S., Kashima, H. & Takahashi, Y. (2009) 'Roughly balanced bagging for imbalanced data'. Statistical Analysis and Data Mining, 2 (5-6), pp.412-426.

Huang, Y.-M., Hung, C.-M. & Jiau, H.C. (2006) 'Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem'. Nonlinear Analysis: Real World Applications, 7 (4), pp.720-747.

Johansson, U., Sönströd, C., Norinder, U., Boström, H. & Löfström, T. (2010) '*Using Feature Selection with Bagging and Rule Extraction in Drug Discovery*'. *Advances in Intelligent Decision Technologies.* Springer, pp. 413-422.

Johnson, R.B. & Onwuegbuzie, A.J. (2004) 'Mixed methods research: A research paradigm whose time has come'. Educational researcher, 33 (7), pp.14-26.

Karim, M. & Rahman, R.M. (2013) 'Decision tree and naive bayes algorithm for classification and generation of actionable knowledge for direct marketing'.

Kaur, G. & Chhabra, A. (2014) 'Improved J48 classification algorithm for the prediction of diabetes'. International Journal of Computer Applications, 98 (22).

Kazmierska, J. & Malicki, J. (2008) 'Application of the Naïve Bayesian Classifier to optimize treatment decisions'. Radiotherapy and Oncology, 86 (2), pp.211-216.

Khalilia, M., Chakraborty, S. & Popescu, M. (2011) 'Predicting disease risks from highly imbalanced data using random forest'. BMC medical informatics and decision making, 11 (1), pp.1.

Kirkby, R., Frank, E. & Reutemann, P. (2007) 'WEKA Explorer User Guide for Version 3-5-6'.

Kohavi, R. & John, G.H. (1997) 'Wrappers for feature subset selection'. Artificial intelligence, 97 (1), pp.273-324.

Kubat, M., Holte, R.C. & Matwin, S. (1998) 'Machine learning for the detection of oil spills in satellite radar images'. Machine learning, 30 (2-3), pp.195-215.

LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S. & Kruschwitz, N. (2011) 'Big Data, Analytics and the Path From Insights to Value'. MIT Sloan Management Review, 52 (2), pp.21-32.

Ledolter, J. (2013) Data mining and business analytics with R. John Wiley & Sons.

Lee, P.M. (2013) 'Use of data mining in business analytics to support business competitiveness'. Review of Business Information Systems (RBIS), 17 (2), pp.53-58.

Leventhal, B. (2010) 'An introduction to data mining and other techniques for advanced analytics'. Journal of Direct, Data and Digital Marketing Practice, 12 (2), pp.137-153.

Lin, M.Y., Lee, P.Y. & Hsueh, S.C. (2012) 'Apriori-based frequent itemset mining algorithms on MapReduce'. Proceedings of the 6th international conference on ubiquitous information management and communication. ACM, pp.76.

Ling, C.X. & Li, C. (Year) Published. 'Data Mining for Direct Marketing: Problems and Solutions'. KDD, 1998. pp.73-79.

Longadge, R. & Dongre, S. (2013) 'Class imbalance problem in data mining review'. arXiv preprint arXiv:1305.1707.

Loyola-González, O., García-Borroto, M., Medina-Pérez, M.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A. & De Ita, G. 'An empirical study of oversampling and undersampling methods for lcmine an emerging pattern based classifier'. Mexican Conference on Pattern Recognition. Springer, pp.264-273.

Lustgarten, J.L., Gopalakrishnan, V., Grover, H. & Visweswaran, S. (2008) 'Improving classification performance with discretization on biomedical datasets'. AMIA annual symposium proceedings, 2008. American Medical Informatics Association, pp.445.

Lustgarten, J.L., Visweswaran, S., Gopalakrishnan, V. & Cooper, G.F. (2011) 'Application of an efficient Bayesian discretization method to biomedical data'. BMC bioinformatics, 12 (1), pp.1.

Maldonado, S. & Weber, R. (2011). 'Embedded feature selection for support vector machines: state-of-the-art and future challenges'. Iberoamerican Congress on Pattern Recognition. Springer, pp.304-311.

Maslove, D.M., Podchiyska, T. & Lowe, H.J. (2013) 'Discretization of continuous features in clinical datasets'. Journal of the American Medical Informatics Association, 20 (3), pp.544-553.

McCausland, R. (2000) 'VARs pick up on telemarketing'. Accounting Technology, 16 (11), pp.56-60.

Milovic, B. & Milovic, M. (2012) 'Prediction And Decision Making In Health Care Using Data Mining'. Kuwait Chapter of the Arabian Journal of Business and Management Review, 1 (12), pp.126-136.

Moro, S., Cortez, P. & Rita, P. (2014) 'A data-driven approach to predict the success of bank telemarketing'. Decision Support Systems, 62 pp.22.

Moro, S., Laureano, R. & Cortez, P. (2011) 'Using data mining for bank direct marketing: An application of the crisp-dm methodology'. Proceedings of European Simulation and Modelling Conference-ESM'2011. Eurosis, pp.117-121.

Müller, H. & Freytag, J.C. (2005) Problems, methods, and challenges in comprehensive data cleansing. Professoren des Inst. Für Informatik.

Negash, S. (2004) 'Business intelligence'.

Orhan, U., Hekim, M. & Ozer, M. (2011) 'EEG signals classification using the K-means clustering and a multilayer perceptron neural network model'. Expert Systems with Applications, 38 (10), pp.13475-13481.

Oza, N.C. & Tumer, K. (2001). 'Input decimation ensembles: Decorrelation throvugh dimensionality reduction'. International Workshop on Multiple Classifier Systems. Springer, pp.238-247.

Panda, M. & Patra, M.R. (2008). 'A comparative study of data mining algorithms for network intrusion detection'. 2008 First International Conference on Emerging Trends in Engineering and Technology. IEEE, pp.504-507.

Pettinger, T. (2016) *Savings ratio UK*. Available at: http://www.economicshelp.org/blog/848/economics/savings-ratio-uk/ (Accessed: 29 August 2016).

Platt, J. (1998) 'Sequential minimal optimization: A fast algorithm for training support vector machines'.

Powers, D.A. & Xie, Y. (2000) Statistical methods for categorical data analysis. Academic Press New York.

Pujari, A.K. (2001) Data mining techniques. Universities press.

Rajput, A., Aharwal, R.P., Dubey, M., Saxena, S. & Raghuvanshi, M. (2011) 'J48 and JRIP rules for e-governance data'. International Journal of Computer Science and Security (IJCSS), 5 (2), pp.201.

Rangel, P., Lozano, F. & García, E. (2005) Published. 'Boosting of support vector machines with application to editing'. ICMLA, 2005.

Respício, A. (2010) Bridging the Socio-technical Gap in Decision Support Systems: Challenges for the Next Decade. IOS Press.

Rigby, E. (2006) 'Eyes in the till Every time a Clubcard owner goes shopping at Tesco, more information is added to what is probably the biggest collection of up-to- date personal data in the UK. So should we be worrying about a Big Brother at the checkout - or celebrating an astounding achievement in sales and market analysis?'. Financial Times, 16.

Roberts, M.L. & Berger, P.D. (1999) Direct marketing management. Prentice Hall International (UK).

Rouse, M. (2010) 'Cognos'. [Online] Available from: http://searchcio.techtarget.com/definition/Cognos [Last Accessed: 23 March 2016].

Saeys, Y., Degroeve, S., Aeyels, D., Rouzé, P. & Van de Peer, Y. (2004) 'Feature selection for splice site prediction: a new method using EDA-based feature ranking'. *BMC bioinformatics,* 5 (1), pp.64

Salvithal, N.N. & Kulkarni, R. 'Evaluating Performance of Data Mining Classification Algorithm in Weka'.

Seewald, A.K. (2002) 'Meta-learning for stacked classification'. *audiology*,*24*(226), p.69.

Seiffert, C., Khoshgoftaar, T.M., Van Hulse, J. & Napolitano, A. (2010) 'RUSBoost: A hybrid approach to alleviating class imbalance'. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 40 (1), pp.185-197.

Sevakula, R.K. & Verma, N.K. (2012). 'Support vector machine for large databases as classifier'. International Conference on Swarm, Evolutionary, and Memetic Computing. Springer, pp.303-313.

Shmueli, G., Patel, N.R. & Bruce, P.C. (2016) Data Mining for Business Analytics: Concepts, Techniques, and Applications in XLMiner. John Wiley & Sons.

Sokol, L., Garcia, B., Rodriguez, J., West, M. & Johnson, K. (2001) 'Using data mining to find fraud in HCFA health care claims'. Topics in Health Information Management, 22 (1), pp.1-13.

Sokolova, M., Japkowicz, N. & Szpakowicz, S. (2006). 'Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation'. Australasian Joint Conference on Artificial Intelligence. Springer, pp.1015-1021.

Sowah, R.A., Agebure, M.A., Mills, G.A., Koumadi, K.M. & Fiawoo, S.Y. (2016) 'New Cluster Undersampling Technique for Class Imbalance Learning'. International Journal of Machine Learning and Computing, 6 (3), pp.205.

Sun, Z., Ampornpunt, N., Varma, M. & Vishwanathan, S. (2010) Published. 'Multiple kernel learning and the SMO algorithm'. Advances in neural information processing systems. pp.2361-2369.

Svetnik, V., Liaw, A., Tong, C., Culberson, J.C., Sheridan, R.P. & Feuston, B.P. (2003) 'Random forest: a classification and regression tool for compound classification and QSAR modeling'. Journal of chemical information and computer sciences, 43 (6), pp.1947-1958.

Syarif, I., Zaluska, E., Prugel-Bennett, A. & Wills, G. (2012) Published. 'Application of bagging, boosting and stacking to intrusion detection'. International Workshop on Machine Learning and Data Mining in Pattern Recognition. Springer, pp.593-602.

Thomas, A.R. (2007) 'The end of mass marketing: or, why all successful marketing is now direct marketing'. Direct Marketing, 1 (1), pp.6-16.

Tiwari, D. (2014) 'Handling Class Imbalance Problem Using Feature Selection'. International Journal of Advanced Research in Computer Science & Technology, pp.516-520.

Watson, H.J. & Wixom, B.H. (2007) 'The current state of business intelligence'. Computer, 40 (9), pp.96-99.

Weiss, G.M. & Provost, F. (2003) 'Learning when training data are costly: the effect of class distribution on tree induction'. Journal of Artificial Intelligence Research, 19 pp.315-354.

Witten, I.H. & Frank, E. (2005) Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.

Zatari, T (2006). 'Data mining in marketing'.

Zheng, Z., Kohavi, R. & Mason, L. (2001) 'Real world performance of association rule algorithms'. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp.401-406.

Zhu, X. & Wu, X. (2004) 'Class noise vs. attribute noise: A quantitative study'. Artificial Intelligence Review, 22 (3), pp.177-210.