Predicting grade progression within the Limpopo Education System

A DISSERTATION SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF SCIENCE

AT THE UNIVERSITY OF CAPE TOWN

IN PARTIAL FULFILMENT

OF THE REQUIREMENTS

OF THE DEGREE

OF

Masters in Philosophy (Information Technology)

By

Frans Ramphele

September 2018

Supervised by

Assoc. Prof. Sonia Berman

Plagiarism Declaration

I have used the required convention for citation and referencing. Each contribution and quotation in this assignment from the work(s) of other people has been acknowledged, cited and referenced accordingly.

I acknowledge that copying someone else's assignment or essay, or part of it, is wrong, and declare that this is my own work.

Signature Signed by candidate

Acknowledgement

Firstly, I would like to express my special thanks to my supervisor (Assoc. Prof. Sonia Berman) for her expertise and guidance throughout the process of writing this dissertation. Without you, I could not have reached this current level of academic success.

Secondly, I would like to thank everyone who contributed to this project; and to Victor Huni for encouraging me to complete my dissertation. It was indeed a great experience!!

Abstract

One way to improve education in South Africa is to ensure that additional support and resourcing are provided to schools and learners that are most in need of help. To this end, education officials need to understand the factors affecting learning and the schools most in need of appropriate interventions. Several theories, models and methods have been developed to attempt to address the challenges faced in the education sector. Educational Data Mining (EDM) is one which has gained prominence in addressing these challenges. EDM is a field of data mining using mathematical and machine learning models to improve learners' performance, education administration, and policy formulation.

This study explored the literature and related methodologies used within the EDM context and constructed a solution to improve learner support and planning in the Limpopo primary and secondary schools education system. The data utilized included socio-economic environment, demographic information as well as learner's performance sourced from the Education Management Information Systems database of the Limpopo Department of Education (LDoE). Feature selection methods; Information Gain, Correlation and Asymmetrical Uncertainty were combined to determine factors that affect learning. Three machine learning classifiers, AdaboostM1 (Decision Stump), HoeffdingTree and NaïveBayes, were used to predict learners' grade progression. These were compared using several evaluation metrics and HoeffdingTree outperformed AdaboostM1 (Decision Stump) and NaïveBayes. When the final HoeffdingTree model was applied to the test datasets, the performance was exceptionally good. It is hoped that the implementation of this model will assist the LDoE in its role of supporting learning and planning of resource allocation.

Acronyms

ADABOOST	: Adaptive Boosting
AUC	: Area Under Curve
BKT-BF	: Bayesian Knowledge Trace – Brute Force
BKT-CGS	: Bayesian Knowledge Trace – Contextual Guess and Slip
BKT-EM	: Bayesian Knowledge Trace – Expectation Maximization
BKT-PPS	: Bayesian Knowledge Trace – Prior Per Student
CFAR	: Correct on First Attempt Rate
CGPA	: Cumulative Grade Point Average
CLI	: Command Line Interface
CRISP-DM	: Cross-Industry Standard Process - Data Mining
DBSCAN	: Density Based Spatial Clustering of Application with Noise
EDM	: Educational Data Mining
EMIS	: Education Management Information System
FN	: False Negative
FP	: False Positive
GPA	: Grade Point Average
GPI	: Gender Parity Index
GUI	: Graphic User Interface
JVM	: Java Virtual Machine
KDD	: Knowledge Discovery in Database
LDoE	: Limpopo Department of Education
LER	: Learner Enrolment Ratio
LTSM	: Learner Teacher Support Material
LURITS	: Learner Unit Record Information Tracking System
MAS	: Multi-Agent System
NP	: Not Promoted
P	: Promoted
PCC	: Percentage of Correct Classification
PFA	: Performance Factor Analysis
PRC	: Precision-Recall Curve
RASE	: Root of Average Squared Error
RMSE	: Root Mean Square Error
ROC	: Receiver Operating Characteristics
SA-SAMS	: South African – School Administration Management System
SEMMA	: Sample, Explore, Modify, Model, Assess
SMOTE	: Synthetic Minority Over-Sampling Technique
SLIQ	: Supervised Learning In Quest
SPRINT	: Scalable Parallelizable Induction of Decision Trees
SQL	: Structured Query Language
TP	: True Positive
UDMP	: Unified Data Mining Process
UDMT	: Unified Data Mining Theory
WEKA	: Waikato Environment Knowledge Analysis

List of Figures

List of Tables

Appendices

Table of Contents

Chapter 1: Introduction

1.1 Problem Statement

The Limpopo Education system consists of 3854 public ordinary schools with almost 1 700 000 learners in the schools. Usually, the department makes forward planning in terms of provisioning of educators to schools and procurement of the Learner Teacher Support Materials (LTSM) for the learners. The two activities take more than 70% of the equitable budget allocated to the department and require accurate grade enrolment statistics to optimize resource allocation. At present, it is very challenging to determine the correct grade statistics for the coming year as the current methods have flaws and therefore lead to a range of failures on planning and resource allocation.

On the other hand, the efficiency of every education system is measured by its ability to produce learners that are able to compete globally. One problem that often hinders the progress of learners within the Limpopo educational context is the inability of the education professionals to promptly identify the learners at risk of failing, and the variables that hamper the learning environment. The inability of the Limpopo Department of Education (LDoE) to detect these learners well in advance means that learners that are performing badly cannot be supported and this creates more problems and affects the efficiency of the education system.

1.2 Purpose of the Research Study

The purpose of the research study is to develop a machine learning model that will predict learner progression to the next grade. The following questions are to be answered by the research study. These questions guide the literature review, methodology, evaluation and reporting of findings.

Research Question 1: What are the main factors that affect learner progression within the Limpopo Education Environment?

Predicting learners' performance and progression is a complex exercise. It involves identification of multiple factors that collaborate to determine how the learner performs. Hijazi et al [1] identified socio-economic, psychological and environmental factors as critical determinants of learners' performance [1]. He further suggested that biographical attributes like gender and race could also influence learners' performance. Hijazi et al [1] assertions are supported by Siyepu [2] when he stated that the performance of learners in South African schooling context is affected by

"*issues of poverty, resources and infrastructure of schools, low teacher qualification, and poor learning cultures in schools*"

Research Question 2: Which classifier between NaïveBayes, HoeffdingTree and AdaboostM1 (Decision Stump) will provide a better prediction accuracy of learner progression within the Limpopo Education Environment?

There are several machine learning algorithms that can be used to solve a classification problem and accurately predict changes in the dependent variable; which in our case is the learner progressing to the next grade or not. However, each algorithm depends on a variety of factors to return accurate results for the analysis [3]. NaïveBayes, HoeffdingTree, and AdaboostM1 (Decision Stump) are the common classifiers used in predicting learners' performance; and they have interesting qualities that among others include simplicity in use. Both the HoeffdingTree and Decision Stump are decision trees and do have an integrated mechanism to deal with redundant attributes or model overfitting. In addition, the three classifiers selected have empirical evidence of being applied successfully to predict learners' performance and are computationally efficient [20, 24-28, 30, 40-43].

1.3 Importance of the Research Study

Planning for resource allocation is a key competency of the LDoE provincial structure. The LDoE provincial structure has an obligation to, among others, direct interventions that are crucial to enable teaching and learning in schools. The results of the study will, therefore, assist the provincial LDoE:

(a) To understand factors that influence learners' performance
(b) To identify learners that are at risk of failing a grade
(c) To enable early interventions to support learners at risk
(d) To enable proper planning in terms of resource allocations
(e) To help minimize learner dropout rates
(f) To sustain learner engagement in learning
(g) To identify which schools need intervention and resourcing

1.4 Research Methodology

Information Technology and Computer Science have evolved over time and have created more opportunities to assist humanity to understand relationships among data variables within big data. This topic relates to the application of data mining techniques and processes to identify hidden patterns in the data.

The LDoE collects operational data from schools every quarter and hosts it in their centralised database. The data collected includes basic school-level data that, among others, includes learner data, teacher data, and school inventory data to mention a few. The study takes advantage of the available data and uses existing technology in data mining and machine learning to explore and find factors that contribute to learners' performance within the Limpopo Education context.

The classification problem for the study is to predict learner progression probabilities early, so as to enable proper intervention and support for the learners at risk. Having said that, data attributes for the study were extracted from the province's Education Management Information System (EMIS) data warehouse and among others, includes learner biographical information, learners' performance and attendance information of both learners and educators.

The first step in the study involved identification of various factors that influence learners' performance and grade progression within the Limpopo Education system. The author used domain knowledge as an educator and pedagogic literature [1, 2] to find a set of attributes considered crucial to learners' performance. The attributes selected were further subjected to feature selection methods using a combination of Sequential Forward Stepwise (SPS), Information Gain (Info-Gain), and Correlation and Asymmetrical uncertainty attribute evaluation methods to eliminate variables statistically negligible to the study.

The second step was to identify and compare three supervised machine learning algorithms that have the potential to understand the underlying structure of the data and be able to predict the learners' performance based on the factors identified in step one above. The literature was used to inform the selection of the algorithms. NaïveBayes, HoeffdingTree and AdaboostM1 (Decision Stump) were selected as the choices for the study. Two experimental feature sets were created. The first feature set (**Set A**) excluded seven attributes with low predictive power based on the exploratory study conducted using filter-based feature selection algorithms. The second experimental feature set (**Set B**) only composed of six attributes with high predictive power based on the exploratory study conducted using forward stepwise feature selection technique. The three classifiers were ran against the two feature sets and the results were compared using different model evaluation techniques in order to select the best feature set and the classifier to build the final model. Waikato Environment Knowledge Analysis (WEKA) data mining tool was used to conduct the experiment and develop the final classification model. [44, 45]

1.5 Experimentation and Evaluation

WEKA experimenter was used to set up and execute the experiments discussed in section 1.4. The results of the experiment were analyzed to assess the classifier performance and generalisability; and to choose the best feature subset considering among others, the model prediction accuracy, computational efficiency, hardware requirements and time for data preparation. The algorithm and feature subset with high classification performance and computational efficiency were used to develop the final model for the research. The model was further tested against the hold-out balanced and unbalanced data sets, to observe how it behaves with unseen data.

1.6 Research Ethical Consideration

Limpopo's EMIS databases, host atomic level information for learners and educators that is highly classified. The information belongs to the government and the use of it is governed by the Protection of Personal Information Act (PAIA) 2013. The act provides inter alia, minimum conditions to handle personal information and code of conduct governing the access and usage of such information.

- Section 57 subsection 1 states that prior authorization is required before using the classified information
- Section 15 subsection 3 requires that the information should be used only for the purpose authorized for and that the information should not be published in identifiable form

Prior approval was granted by the Head of LDoE to utilize data from EMIS only for the purpose of this research. The Department also placed a condition that the researcher should share the copy of the final research report with the Department.

In accordance with the ethical code of the University of Cape Town, an adequate level of confidentiality of the research data was ensured throughout and anonymity of information given priority.

1.7 Chapter Layout

This section provides an overview of what will be discussed in different chapters of the research report.

Chapter 2: Background Literature Review

This chapter provides a general background to the research, possible theoretical frameworks and procedural constructs and concepts critical to guide the research study. The chapter also provides an overview of similar studies conducted, successes

and challenges met and discusses critical concepts and how they were used in different research similar to this study.

Chapter 3: Data Understanding and Preparation

This chapter outlines methods used to assemble, transform, clean and consolidate data as well as uploading it to the WEKA environment for pre-processing and classification.

Chapter 4: Feature Selection

This chapter discusses the exploratory studies conducted to enable feature selection.

Chapter 5: Experiment Design and Execution

This chapter outlines the environment used to conduct the experiment, provides a view of how data was organised; how the experiment was set up and executed, and the challenges met.

Chapter 6: Results and Findings

This chapter provides a comparative view of how different models performed in predicting learner progressions, including observations and findings in relation to the research questions. It describes how the best model for the study was selected, developed and trained to solve the research problem.

Chapter 7: Conclusion

This chapter summarises the findings of the experiment and provides a view as to the extent to which the research aims have been met. The chapter also proposes future work required to improve the research.

Chapter 2: Background and Literature Review

2.1 Introduction

This chapter reviews an existing literature that examines the impact of data mining on the prediction of learners' performance. The chapter also explores common processes or techniques used to guide data mining activities. Furthermore, the chapter interrogates various models used to predict learners' performance, the methodology used and how the final models were tested for reliability and generalisation. And lastly, in the context of the existing theory of data mining, a conceptual framework will be produced to guide the remaining chapters of the study.

2.2 Machine Learning and Training Methodologies

Machine learning is a sub-discipline of artificial intelligence that is concerned with developing systems that can learn with experience and time [4]. Alpaydın [5] asserts that machine learning uses statistics in building mathematical models that will assist in making inferences from training data or past experiences. He went further to say, the model can either be predictive which is about making future inferences/predictions or descriptive which is about learning and gaining knowledge using the current data inputs. Nilsson [6] and Donalek [7] identified two methods of learning in artificial intelligence:

(a) Supervised Learning

- A set of structured data with the known outcome is provided as input to the system during training.
- Has the capability to learn from this and generalize to new data.
- Construction of proper training, validation, and testing datasets are very important for the algorithms to create accurate models.

(b) Unsupervised Learning

- Unstructured data is provided as input to the system during training.
- The expected output is unknown and there is no way of evaluating the final solution.
- Unsupervised learning mostly uses clustering of similar items.

Ayodele [8] further identified three additional types of learning methods, which are: semi-supervised, reinforcement learning, transduction and learning to learn. However, these are not relevant to the LDoE problem, and are not discussed further.

2.3 Data Mining

Data mining is defined as a process of discovering hidden knowledge from large databases using mathematical models and /or machine learning algorithms. The data mining process is sometimes known as Knowledge Discovery in Databases (KDD) [9]. The knowledge discovery process involves collection of data, cleaning, transformation, integration, analysis, and presentation of the data. The tools used to prepare, process and disseminate the data during the knowledge discovery process have some level of intelligence and can discover patterns and deep knowledge of the data through learning [9].

2.4 Educational Data Mining

Pechenizkiy et al [17] uncovers a specialized field of data mining in the context of education known as Educational Data Mining (EDM). EDM is defined as,

"*an emerging discipline, concerned with developing methods for exploring the unique and increasingly large-scale data that come from educational settings and using those methods to better understand students, and the settings which they learn in*" [17]

The field utilizes artificial intelligence and education domain knowledge to mine educational data and discover hidden knowledge that will help improve teaching and learning as well as education administration and policy formulation. Gautam et al [18] observed the following goals of EDM from the literature:

(a) Predicting students' future learning behavior: This involves creating models that impersonate students' learning behavior and use those models to predict their future development patterns and related challenges.

(b) Discovering or improving domain models: This involves experimenting with factors influencing pedagogy (teaching) and devising intelligent machine learning models to optimize instructional sequences seeking to support the learning styles.

(c) Studying the effects of educational support: This involves using machine learning and mathematical models to assess the impact of educational support for learning and teaching.

(d) Advancing scientific knowledge about learning and learners: This involves using machine learning and mathematical models to advance knowledge in the education domain.

EDM goes beyond conventional data analysis process. It uses classical data mining processes and analysis concepts like classification, clustering, regression, visualisation, etc. The logical process of educational data mining is different to data analysis. The primary goal of EDM is to generate hypotheses and secondary to that, extract the knowledge out of the data. Thus, the EDM focus is more on generating questions from the data rather than the answers [18]. The insights gained by the data mining process will be verified by conventional data analysis techniques.

2.5 Existing Research in EDM

The review of the literature shows a wealth of research available to support this study. An example of a study relating to this research was that of Erkan [3] who developed a model to predict learners at risk of performing poorly. In his study, he used a combination of instance-based learning classifier, Decision Tree and NaïveBayes machine learning techniques. The study was divided into two phases, which are training and testing. During the training, he used the time-varying attributes like an actual performance of the learner and attendance over time as opposed to time-invariant attributes like gender and race. He then used three decision schemes to combine the results from the three techniques to decide if a learner will fail. The outcome of the study was that "*combining results of different machine learning algorithms may produce a better classification than a single technique*".

In another study by Kotsiantis et al [4], five machine learning techniques (Decision Trees, Bayesian Nets, Perceptron-based Learning, Instance-Based Learning and Rule-learning) were used to predict learners' performance in distance learning systems. The study was also divided into two phases, training and testing. In this study, both the time variable (performance data on assignments) and time invariable attributes (biographical information) of the learner were used. All five machine learning techniques were trained using the data and it was found that NaïveBayes algorithm is the most accurate in predicting the learners' performance.

The methodology used by Kotsiantis et al [4] did not report on how the learner attributes used in the study were selected and this could lead to the researcher paying more attention to attributes that have a low impact on the study, and produce results that are less reliable. It is important to emphasize that the study conducted by [4] only focuses on distance learning (Higher Education) and the methodology used can produce different results if applied in the basic education environment, as the variables that may affect learners' performance are different.

2.6 Data Attribute Assembly

According to Guyon [19], building a feature representation provides an opportunity for the data miner to incorporate domain knowledge of the classification or clustering problem. In a study conducted by [20-24] to predict student performance, Cumulative Grade Point Average (CGPA) was used as the main attribute for their research. For the research and conception purpose, the CGPA is the overall GPA (Grade Point Average), which includes dividing the number of quality points a learner earned by a possible amount of points in the course grade. The CGPA provides a good measurement of learner aptitude and knowledge comprehension abilities.

Adding to the usage CGPA, the research conducted by [22-27] have furthermore used learner demographic information (e.g. gender, learner disability, nationality, location, learner age, financial background, family composition etc.) among the attributes selected for their study. The literature correctly argues that demographic information provides a better understanding of environmental and socio-economic factors affecting a learner and their ability to learn [1, 2].

The study conducted by Shahiria et al [30] reveals that some of the research predicting learners' performance use psychometric tests to identify student interest, study behavior, engage time, and family support in order to evaluate student's achievement. The study correctly argues that the psychometric method is rarely applied in predicting student performance because it focuses much on a collection of qualitative data that is difficult to find from the respondents [30].

It must be noted that the role of the human in the knowledge discovery process cannot be substituted. The use of domain knowledge can go very far to assist a researcher to constrain the attribute search space and enhance the data mining process.

2.7 Feature Selection

Feature selection is the process of selecting features/attributes with high predictive outcome. This process is at the center of a successful model development. The idea behind feature selection is to remove redundant and irrelevant features in the data, to improve classification accuracy without compromising the underlying structure of the data [19].

Ramaswami et al [31] confirm assertions by Guyon [19] that feature selection has been proven to be effective in enhancing learning efficiency, increasing predictive accuracy and reducing the complexity of the prediction model.

2.7.1 Feature Selection Methods

According to Ramaswami et al [31], algorithms for feature selection fall into two broad categories which are shown in Table 2.1 below:

Table 2.1: Feature selection algorithms

Filters (Use set of rules (Heuristics) to explore relationships in the data)	**Wrappers** (Use the learning algorithms to evaluate the usefulness of features)
Correlation-based attribute evaluation Chi-Square attribute evaluation Gain-Ratio attribute evaluation Information Gain attribute evaluation Relief attribute evaluation Symmetrical Uncertainty Attribute evaluation	Wrappers use learning algorithms like Decision Tree (J48) NaïveBayes AdaboostM1 etc.

The search strategy forms an important part of the attribute evaluator and defines how the evaluator should search the data in the process of evaluating the impact of each attribute to the predictive class. The search strategy (Greedy Stepwise, Ranking, Best First etc.) must be selected together with attribute evaluator to do feature selection [19, 31].

Attribute selection is the least documented phase of the data mining process. Most of the researchers [24, 26-28, 30-32] conducted a study among others, to predict learners' performance, learner dropouts and retention and did not describe the feature selection in their data mining process. Considering the benefits of the feature selection process as outlined by [19, 31], a poor or missing feature selection phase might turn the predicted outcome to be a chance process that lacks scientific and epistemological grounds for proving the validity and generalisation of the results.

In an article by [20], correlation-based attribute evaluation with ranker search strategy was used to establish a subset of features for a classification exercise. The selected subset of features was put through three different classification algorithms and the Square Root of Average Squared Error (RASE) metric was used to assess the accuracy and validity of the attribute subset.

Ramesh et al [25] conducted a similar study predicting student performance using a statistical and data mining approach. A combination of feature selection algorithms; Chi-Squared, Info-Gain, One Rule (OneR), Symmetrical Uncertainty, and Relief with ranker search method was used in their research. Ranker is a search method that

navigates different combinations of attributes based on a set of heuristics and lists the results in ranked order. The results returned by the rankings of the five attribute evaluators were then added and averaged to determine the ranking of the attributes; ten out of 27 features were selected for the classification exercise. The Percentage of Correct Classification (PCC) was used to assess the validity of the attribute subset.

Tair et al [23] approached feature selection differently using association rules to assess the extent of the relationship between antecedent (independent variable) and consequent (dependent variable). The lift value of greater than 1 shows a positive correlation between antecedent and consequent, and as such the antecedent was included in the feature set

Reference is made to a study conducted by John et al [53] to assess the impact of irrelevant features in the data mining process. In this study, it has been proved that the feature selection is significant to improve the classification accuracy of the mining tasks. A specific reference was made to the algorithms like Iterative Dichotomiser 3 (ID3), Classification and Regression Tree (CART) and C.45 which failed to ignore irrelevant features in the data supplied to them. The findings were that, if the identified irrelevant features were ignored, the classification accuracy of the research study would have been improved

Dougherty et al [47] argue that using cross-validation methods for feature selection is risky due to the possible high variance among the classifiers generated during the process. He further asserts that testing of the generated feature subsets on the classifiers can provide a better test for the optimal subsets to be selected for the study.

Kumar et al [36] expressed a similar view and confirm that literature does not prescribe any procedure on how the feature selection process should be conducted. The only test for accuracy and validity of the feature subset selected is through the accuracy, sensitivity, and reliability of the classification model [36].

2.7.2 Feature Selection Methods Selected for the Experiment

According to Marono et al [69], the wrapper methods typically provide an optimized feature selection, but are very expensive in terms of the processing requirements. The embedded method usually incorporates feature selection as part of the model training process and feature relevance is obtained logically from the goal of the learning model. On the other hand, filter methods are computationally less expensive and provide better generalisation because they act independent of the induction algorithm used [69]. For the purpose of the research experiment; Correlation Attribute,

Info-Gain and Symmetric Uncertainty feature selection methods will be used in the study due to their generalisation capabilities as well as considering the available data processing platform. Refer to table 3.1 for the hardware specification available for the research experiment**.**

The following are the mathematical description of how the three selected feature selection methods determine the relevant features [70]:

(a) Correlation Attribute

It is univariate method which works only with numeric data. It uses Pearson's correlation to determine the linear relationship between the variables. The nominal values are changed into numeric using weighted averages.The formula for correlation attribute is calculated as follows

$$\mathrm{r} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \qquad \text{2.7.2 (a1)}$$

where σ_x and σ_y are the standard deviation of x_i and y_i and $\bar{x}$ and $\bar{y}$ are the means.There Pearson's correlation coefficient is computed by taking the covariance of two variables and dividing by the product of their standard deviations.

(b) Information Gain

It is an entropy-based feature selection method that determines feature relevance between the attribute and class label using entropy. Information Gain for a feature X and the class label Y is calculated using the formula.

$$H(X,Y) = H(X) - \mathrm{H}(\mathrm{X}|\mathrm{Y}) \qquad \text{2.7.2 (b1)}$$

Where

- H(X) is the entropy of X
- $\mathrm{H}(\mathrm{X}|\mathrm{Y})$ is the entropy of Y after observing X.

The entropy of X is calculated as follows:

$$\mathrm{H}(\mathrm{X}) = -\sum_{i} P(x_i)\, log_2\,(P(x_i)) \qquad \text{2.7.2 (b2)}$$

The entropy of X after observing Y is calculated as follows:

$$\mathrm{H}(\mathrm{X},\mathrm{Y}) = -\sum_{i} P(y_j) \sum_{i} P(x_i \mid y_j)\, log_2(P(x_i \mid y_j)) \qquad \text{2.7.2 (b3)}$$

This method calculates the Information Gain of each attributes individually and considers the one with high Information as relevant. The main drawback of the algorithm is that it selects the feature with high Info-Gain which may or may not be more informative.

(c) Symmetric Uncertainty (SU)

It is a variant of Information Gain which was developed to overcome the drawbacks of Information Gain by dividing the sum of the entropies of X and Y as follows:

$$\mathrm{SU} = 2 * \frac{\mathrm{InfoGain}(\mathrm{x},\mathrm{y})}{\mathrm{H}(\mathrm{x}) + \mathrm{H}(\mathrm{y})} \qquad \text{2.7.2 (c1)}$$

Where

- SU=1, when the knowledge X can predicts Y.
- SU=0, when X and Y are uncorrelated.

2.8 Classification Methods

The goal of classification in data mining is to accurately predict the target class for each case in the data. This is supported by Joazeiro et al [37] when he states that the objective of the prediction methods is to deduce a numerical or nominal attribute as functional output/dependent variable given a single or set of independent variables. Classifiers and Regressors are the common prediction methods found in classical data mining projects. Examples of classification algorithms include decision trees, decision rules, ensemble learners, Bayesian functions and lazy learners. Classification involves two activities, which are learning and prediction. Joazeiro et al [37] assert that cross-validation is a common method used to train and test the classification methods. For the purpose of this study, the classification methods discussed will be limited to Ensemble learners, Decision trees, and Bayes classifiers.

2.8.1 Decision Trees

Shahiria et al [30] have drawn attention to the fact that decision trees are the most common classification technique used for prediction. In the study conducted by [20, 24-28, 30] to predict among others, student performance and student retention, decision trees (J48, C4.5, REPTree) were among other algorithms tested against the data. Shahiria et al [30] correctly argue that most researchers include decision trees in their studies due to their simplicity and less effort required for data preparation. Petri [32] has expressed a similar view and states that decision trees can adapt to various data structures. Furthermore, due to their non-parametric processing capability, they are not sensitive to outliers.

Irrespective of the resilient nature of the decision trees as expressed in the literature [30, 32] above, in all studies [20, 24-28, 30] reviewed, decision trees were never a choice for building the final classifier that is able to perform well and be able to generalize when new data is applied to the classifier.

Petri [32] reveals that decision trees suffer the curse of dimensionality. Thus, they tend to perform well if few relevant attributes are used, and poorly if many complex interactions are included in the data. He further states that their greedy characteristic leads to over-sensitivity to irrelevant attributes and data noise during training. Examples of decision trees include Decision stump, J48, HoeffdingTree, Random Forest, RepTree etc.

According to Petri [32], there are options to deal with the limitations of the decision trees discussed above. One of the options is to partition the data and use algorithms like Supervised Learning In Quest (SLIQ) or Scalable Parallelizable Induction of Tress (SPRINT), which are able to address the memory restrictions of the decision tree and later combine the results to form a single decision tree classifier. Another option is to partition the data and train the decision tree incrementally. An example of incremental learning decision trees is HoeffdingTree [75].

2.8.2 Bayes Classifiers

NaïveBayes is a family of conditional probabilistic algorithms that implement Bayes theory. The Bayes theory is the fundamental statistical approach to the problem of pattern classification and has strong assumptions that features are independent of one another [38]. Examples of Bayes classifiers include Bayesian Network and Simple NaïveBayes

The literature states that Bayes classifiers can be trained very efficiently in a supervised learning environment and with a small training dataset to estimate the parameters necessary for the classification problem [38, 39].

Simple NaïveBayes algorithms are among the most common classifiers used in the EDM field to predict student learning behaviors. Their decoupling nature of the class conditional feature distribution makes them powerful in alleviating problems related to the curse of dimensionality when subjected to a large set of features in a dataset [38]. Although it has been pointed out by [38, 39] that the far-reaching feature independence assumption is not always applicable in reality, they still have several properties that make them very useful in practice and as a choice for many researchers.

A study conducted by Zhang et al [26] to improve the accuracy of students' final grade using optimal equal width binning and Synthetic Minority Over-Sampling Technique (SMOTE), Artificial Neural Network (ANN) and NaïveBayes models yielded almost the same classification accuracy of 75% when applied to the data. NaïveBayes was selected as a choice for the study due to its simplicity and computational efficiency.

In addition, the work of [25-27, 30] included a NaïveBayes classifier among the choice of algorithms used for their study. NaïveBayes outperformed other algorithms in [26, 27] and was outperformed in [25, 30] respectively. Shahiria et al [30], pointed out that NaïveBayes performed better than the other classifiers when additional attributes were added to the attribute space. Shahiria et al [30] has expressed a similar view that NaïveBayes is more robust and intelligent in dealing with challenges related to model overfitting.

2.8.3 Ensemble Learners

According to Dietterich [33], *"ensemble methods are learning algorithms that construct a set of classifiers and then classify new data points by taking a weighted vote of their predictions".* Zhi-Hua [34] has expressed a similar view and described ensemble learning as a process of training multiple learners to solve a single problem. The idea around ensemble methods is to improve the classification accuracy and generalisation ability of the weak/base learners that are better than random guessing. The base learners are usually decision trees due to their simplicity in architecture and computational efficiency [34]. The most common ensemble learning algorithms are bagging and boosting.

Bagging: bagging draws random sampling sets from the training data called bootstraps (random sampling with replacement) and each sample is used to train an ensemble of generated weak learners in order to promote model variance. It then averages the prediction (for regression problems) or uses majority voting (for classification problems) from the ensemble classifiers to improve the base classification accuracy [34]. Below is the processing logic of the bagging ensemble method:

Step 1: Creates ***n*** number of bootstraps (sampling with replacement)
Step 2: Train base learners ***b*** on each bootstrapped ***n***
Step 3: Average the prediction accuracy or use majority voting of ***b*** on each

An example of bagging method is implemented by the Random Forest algorithm which uses a combination of the random decision tree and bagging method to achieve high classification accuracy.

Boosting: boosting use all the data to train ensemble learners sequentially and iteratively. The instances that were misclassified by the previous learners are given more weight and served to the next learner for focused processing. To elaborate:

Step 1: Draw a first random subset of training samples ***t1*** (without replacement) from the training dataset ***t*** to train a first base learner ***b1***
Step 2: Draw a second random subset of training sample ***t2*** (without replacement) from the training dataset ***t*** and add 50 percent of samples ***t1*** (misclassified) to train a second base learner ***b2***
Step 3: Draw a third sample ***t3*** from the training set ***t*** on which ***b1*** and ***b2*** differ to train a third base learner ***b3***
Step 4: Lastly, combine all the three base learners (***b1, b2, b3***) via majority voting to define the accuracy level.

An example of the boosting algorithms is Adaptive Boosting (AdaboostM1), AdaboostM2 and Logistic Boosting (LogitBoost)

The study conducted by [40-43] compared classical data mining algorithms with ensemble methods to predict student performance in both online and offline learning environments. In [40], Decision Tree with AdaBoost, Random Forest, K-Nearest Neighbors (KNN), Logistic Regression, NaïveBayes and Stochastic Gradient Descent were used to predict student performance in an online math learning environment. The algorithms were applied to three different attribute subsets representing varying learning objectives. AdaBoost on Decision Tree, Logistic Regression and Random

Forest performed best. Stapel et al [40] constructed an ensemble classifier from the three algorithms using a soft voting strategy and applied it to the three attribute subsets. An increase in the classification accuracy was reported. It was also observed that in one set of attributes, the false positives increased and the researcher [40] arguably associates the problem with the voting strategy used for constructing the ensemble. The researcher recommended a stacked generalisation methodology to deal with the shortcoming of the simple weighted ensemble implemented via soft voting.

Stacked generalisation is a different type of ensemble construction logic that introduces a second stage classifier to combine multiple base learners. Although the method has been proven to work in theory, it is less widely used than bagging and boosting as discussed above [34].

Gayathri and Shet [41] used J48, NaïveBayes, Decision table and Bagged decision table among his choices of algorithms to predict learners' performance. Although a Bagged decision table was not a choice for addressing this problem, it increased the classification accuracy of the decision table from 40% to 82%. In their research, J48 was selected as a choice for the study with the prediction accuracy of 85% [41].

A reference to the study conducted by [42] to predict student post-test scores for an Intelligent Tutoring System, Bayesian Knowledge Tracing (BKT) variants (*Less Data (BKT-Less Data), Expectation Maximization (BKT-EM), Brute Force (BKT-BF), Prior Per Student (BKT-PPS), Contextual Guess and Slip (BKT-CGS)*), Correct on First Attempt (CFAR), Performance Factor Analysis (PFA), Tabling and Ensembles (*Stepwise, Stepwise with Averaging, Random Forest, Uniform Averaging, Linear Regression, Logistic Regression*) were used. The researcher hypothesized that the ensembles will perform better than other methods, considering the empirical evidence and literature around the concept. The results of the research were compared using correlation factors and Root Mean Square Error (RMSE) metrics, and the hypothesis was proved wrong. The researcher attributes this negative outcome to insufficient data used in the study, and concludes that ensembles perform optimally with larger datasets.

2.8.4 Classifiers Selected for the Experiment

Expanding from section 2.8.1 to 2.8.3, the most common used classifiers in similar research studies includes Decision Trees [20, 24-28, 30], NaïveBayes [25-27, 30], and ensemble classifiers [40-43]. Simple NaïveBayes, HoeffdingTree and

AdaboostM1 (Decision Stump) were selected for the experiment due to their frequency of use in similar studies, and simplicity in their built architecture and computational efficiency.

The following provides the mathematical description of how the three selected classifiers process information in accordance with their predictions [70]:

AdaboostM1

AdaboostM1 is a boosting ensemble classifier that trains base learners sequentially. For every learner[1] with index *t*, AdaboostM1 computes the weighted classification error using the following formula [51, 75]:

$$\varepsilon_t = \sum_{n=1}^{n} d_n^{(t)} I(y_n \neq h_t(x_n)) \qquad \text{2.8.4 (a1)}$$

Where;

- $\boldsymbol{x_n}$ is a vector of predictor values for observation $\boldsymbol{n}$.
- $\boldsymbol{y_n}$ is the true class label.
- $\boldsymbol{h_t}$ is the prediction of learner with index $\boldsymbol{t}$.
- **I** is the indicator function.
- $\boldsymbol{d_n^{(t)}}$ is the weight of observation $\boldsymbol{n}$ at step $\boldsymbol{t}$.

AdaBoostM1 then increases weights for observations misclassified by learner *t* and reduces weights for observations correctly classified by learner t. The next learner $\boldsymbol{t+1}$ is then trained on the data with updated weights $\boldsymbol{d_n^{(t+1)}}$
After training completes, AdaBoostM1 computes prediction for new data using

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x) \qquad \text{2.8.4 (a2)}$$

where

$$\alpha_t = \frac{1}{2}\log\frac{1-\varepsilon_t}{\varepsilon_t}$$

[1] The term *learner* in this context refers to a classifier, not a pupil/data instance.

are the weights of the weak hypotheses in the ensemble. Training by AdaBoostM1 can be viewed as stagewise minimization of the exponential loss

$$\sum_{n=1}^{n} w_n \exp(-y_n f(x_n)) \qquad \text{2.8.4 (a3)}$$

where

- $y_n \in \{-1,+1\}$ is the true class label.
- w_n are observation weights normalized to add up to 1.
- $f(x_n) \in (-\infty,+\infty)$ is the predicted classification score.

NaïveBayes

A Bayesian classifier uses Bayes theorem to predict the classification outcome. The following formula is used for prediction [38, 75];

$$p(c_j|d) = \frac{p(d|c_j)\, p(c_j)}{p(d)} \qquad \text{2.8.4 (b1)}$$

• $p(cj|d)$= probability of instance ***d*** being in class c_j (**Posterior probability**)

• $p(d|c_j)$= probability of generating instance ***d*** given class c_j. (**Likelihood***)*

• $p(c_j)$ = probability of occurrence of class c_j. (**Prior probability**)

• $p(d)$= probability of instance ***d*** occurring (**Evidence**)

To simplify the interpretation of the formula, we can express it as follows;

$$posterior\ probability = \frac{\text{Likelihood} * \text{priorProbability}}{\text{evidence}} \qquad \text{2.8.4 (b2)}$$

The outcome (posterior probability) is highly dependent on the prior probability if the amount of data used to train the classifier is small. When a large training set is used, the impact of the prior probability is lower.

The Simple NaïveBayes usually becomes a choice for classification projects because it is fast and computationally efficient. As already indicated in chapter 3, it's not sensitive to irrelevant features and can handle real and discrete data as well as streamed data

HoeffdingTree

The A HoeffdingTree-Very Fast Decision Tree (Hoeffding-VFDT) is an incremental, anytime decision tree induction algorithm that is capable of learning from massive data streams, assuming that the distribution generating examples does not change over time. HoeffdingTree exploit the fact that a small sample can often be enough to choose an optimal splitting attribute. This idea is supported mathematically by the Hoeffding bound, which quantifies the number of observations (in our case, examples) needed to estimate some statistics within a prescribed precision (in our case, the goodness of an attribute).

The classification problem is generally defined as follows. *A* set of **N** training examples of the form ***(x,y)*** is given, where ***y*** is a discrete class label and ***x*** is a vector of ***d*** attributes, each of which may be symbolic or numeric. The goal is to produce from these examples a model ***y = f(x)*** that will predict the classes ***y*** of future examples *x* with high accuracy [52]

Central to the processing ability of the HoeffdingTree is the Hoeffding Bound. This gives a certain level of confidence about the best attribute to split the node. The Hoeffding bound states that, with confidence level $\mathbf{1}-\boldsymbol{\delta}$, the true meaning of variable ***r*** is at least, $\mathbf{1}-\epsilon$ where ϵ can be calculated as shown in equation 2.8.4(c1) below [52, 56]

$$\epsilon = \sqrt{\frac{R^2 \ln(\frac{1}{\delta})}{2N}} \qquad \text{2.8.4 (c1)}$$

where;

N; independent observations
R; Bounded range

HoeffdingTree uses the Info-Gain as rules to find the upper and lower bounds with high confidence. The upper bound and lower bound are calculated using

Upper Bound

$$G(A,T)^{+} = \sum_{v \in A} P\ (T,A,v) + \sqrt{\frac{R^2 \ln(\frac{1}{\delta})}{2N}\ H\ (\mathrm{Sel}\ (T,A,v))^{+}} \qquad \text{2.8.4 (c2)}$$

Lower Bound

$$G(A,T)^{-} = \sum_{v \in A} P\,(T,A,v) + \sqrt{\frac{R^2\ \ln(\frac{1}{\delta})}{2N}\ H\ (\text{Sel}\ (T,A,v))^{-}} \qquad \text{2.8.4 (c3)}$$

Where;

A; an attribute in the T set of training samples

$P\ (T,A,v)$; a fragment of training samples in set *T that holds the value v from attribute A*

$\text{Sel}\ (T,A,v)$; selects all the training samples having value v for attribute A from set T.

2.9 Data Mining Techniques

Data mining processes or techniques are central to guide the professional data mining tasks. They provide a logical and empirically tested approach to data mining. Jackson [10] identified three data mining techniques that are commonly used by data miners:

2.9.1 KDD (Knowledge Discovery Process)

The traditional KDD process was developed in 1996 by Fayyad [11]. The process was developed to guide the data mining activities. Below is the graphical illustration of the KDD model and the description of each phase [11]

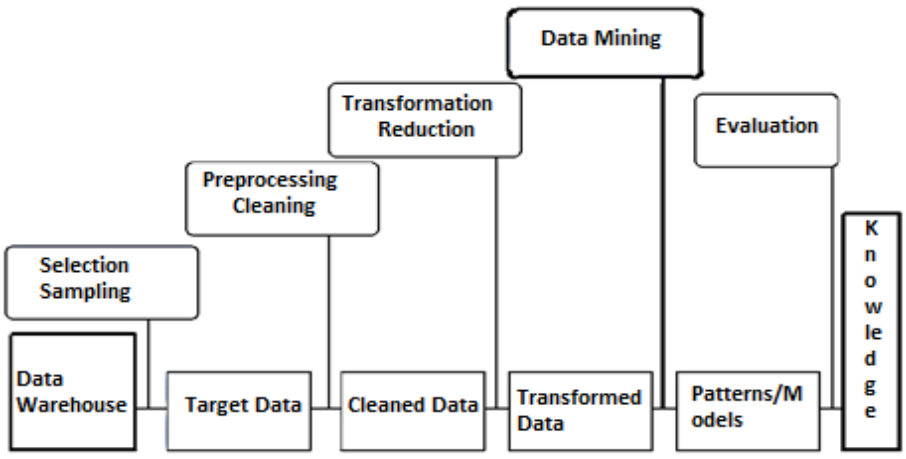


Figure 2.1: KDD process model [11]

The KDD model consists of five core phases [11]:

(a) Selection and Sampling: This phase involves understanding the application domain and the goal of the data mining process. This is used to create a target dataset that will represent the data mining population. The phase will include activities like problem definition, data collection, attributes composition and sampling.

(b) Pre-processing and Cleaning: This phase involves cleaning of the target dataset. The phase will include activities like data cleaning, normalisation, handling of missing data and outliers. It is also important to learn and understand the business data at this stage to ensure a clinical and successful data cleaning process.

(c) Transformation and Reduction: This phase involves the transformation of the data in the format acceptable to the data mining method(s) to be used. At this stage, one needs to select and match a goal of the data mining to the method(s) used. This will include a selection of the data mining methods, analyse data and identify attributes which affect the class attribute and finally, produce a set of data that is ready to be fed into the data mining function.

(d) Data Mining and Pattern definition: This phase is the kernel of the data mining process. It involves activities like choosing the algorithms to be used for mining, feeding data to the mining algorithms and running the algorithms against the data. The output of this process will be a trained data mining model that will solve the problem for new, unseen data.

(e) Evaluation and Knowledge Discovery: This phase involves interpretation of the mined patterns in the data and makes them understandable to the user. The activities in this phase will include data visualisation, Interpretation, testing and validation, summarization and acting on the discovered knowledge through documentation or reporting.

2.9.2 SEMMA

SEMMA (SAMPLE, EXPLORE, MODIFY, MODEL, ASSESS) is a data mining technique developed by SAS Institute to assist data miners to organize their data mining tasks; to explore statistical and data visualisation techniques and develop and test predictive models. Below is the graphical illustration of the SEMMA model and the description of each phase [10]

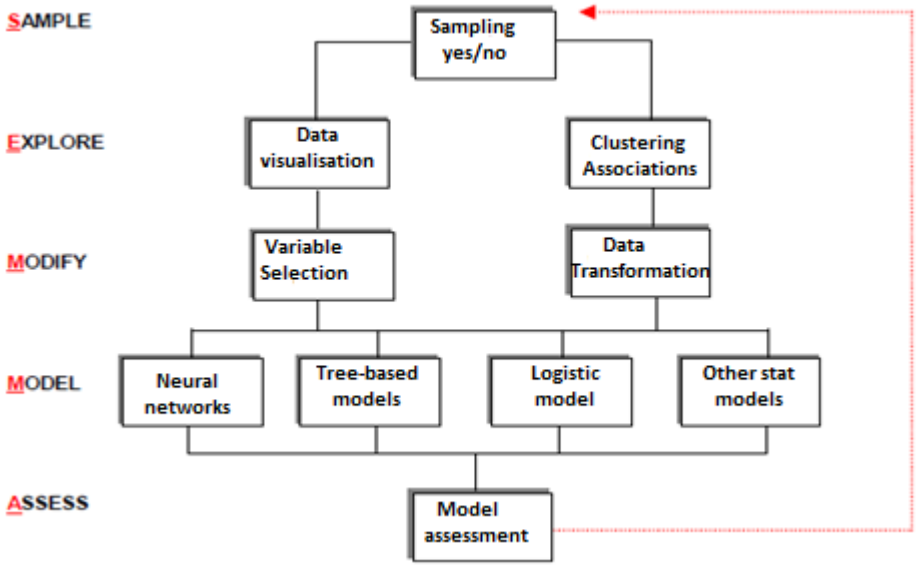


Figure 2.2: SEMMA data mining technique [10]

(a) Sample (*Optional*): Use scientific modelling techniques to create a sample that represents the entire population. SAS Institute encourages the creation of the sample to reduce costs associated with processing time and infrastructure.

(b) Explore: Explore data to identify existing patterns in the data. This will assist to identify data anomalies or data trends. Clustering and data visualisation can help to facilitate data exploration.

(c) Modify: Based on the results of the data exploration and patterns or data anomalies identified, the data attributes are modified to fit the intention of the mining process.

(d) Model: This is the stage where one uses data mining algorithms to develop a model that will predict the desired outcome.

(e) Assess: The assessment stage involves testing the developed model in a different context. As such, the test data is used with different instances of the same data attributes used during the model development, in order to assess the reliability of the model.

2.9.3 CRISP-DM

Cross-Industry standard process for data mining (CRISP-DM)

This data mining technique was conceived in 1996. The initial work to develop the model started in 1997 under a European Union project funded by ESPRIT. CRISP-DM process evolved to be the technology neutral industry standard for the data mining

process. Below is the graphical illustration of the CRISP-DM model and the description of each phase [10].

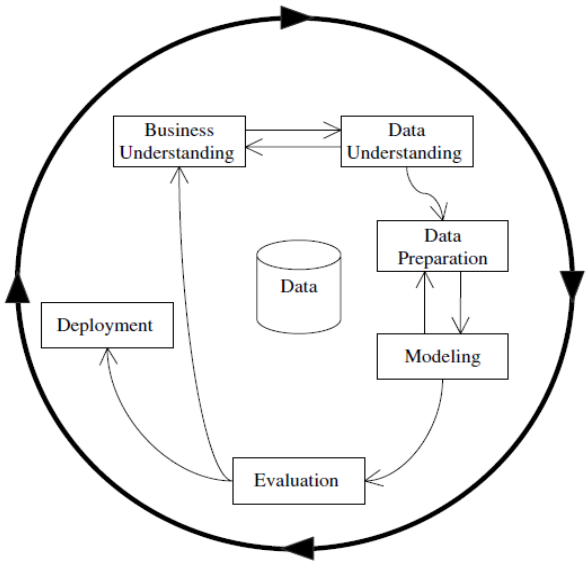


Figure 2.3: The CRISP-DM process model [10]

(a) Business Understanding: This involves a clear understanding of the business requirements of the data mining process. This stage will result in a clear problem definition.

(b) Data Understanding: This phase involves data collection and further understanding the structure, quality and related challenges of the data.

(c) Data Preparation: This involves preparation of the final data to be fed into the model through a set of data pre-processing activities e.g., data normalisation, attributes selection, sampling etc.

(d) Modelling: This phase involves subjecting final data against different data mining techniques. The output of this phase will be a developed model.

(e) Evaluation: The model will be tested thoroughly for accuracy to see if it addresses the business requirement or problem definition. The model should also be able to work with different data instances. The output of this phase will be a well-tested and reliable model to serve the business requirement.

(f) Deployment: The tested model can now be deployed. All the documentation required for the end user to run the model will be created in this phase.

2.10 Comparison of KDD, SEMMA, and CRISP-DM and Related Challenges

Azevedo [12] conducted a study to compare the implementation of KDD, SEMMA, and CRISP-DM. The study confirmed that SEMMA and CRISP-DM are comparable data mining approaches and they both implement the traditional KDD.

There are currently insufficient theories governing the research and techniques in the field of data mining [13]. Most of the data mining processes in use both in the industry and the education sector are, ordinarily, the common single-step data mining process. Khan et al [13] went further to state that the single-step data mining process is designed to handle discrete data mining tasks such as clustering, classification, visualisation, regression and association individually. They further assert that data mining tools based on single-step data mining processes are failing to produce deeper knowledge, and propose a multi-agent system for composition of multiple mining tasks.

There is a universal acceptance among data mining researchers that the discovery of deeper knowledge from data is not a single-step process, but a multi-step and unified process [13-15].

2.11 Unified Data Mining Theory

To mitigate limitations associated with single-step data mining processes as well as standardizing research in the field of data mining, the Unified Data Mining Theory (UDMT) was developed by Khan et al [13-14].

The theory supports the idea of multi-step and unification of data mining processes. It sees data mining activities (clustering, classification, and visualisation) as unified by means of composite functions that are dependent on one another towards knowledge discovery. The illustration below depicts the understanding of UDTM model from the Khan et al [13] perspective.

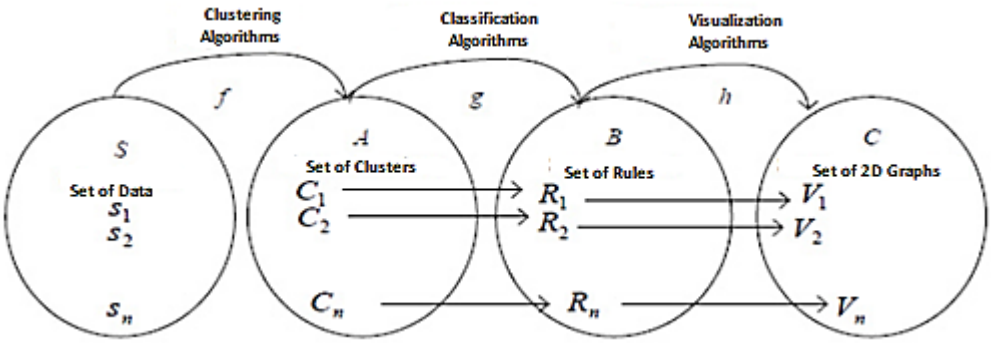


Figure 2.4: Unified Data Mining Theory [13]

The model identified three key data mining functions that need to work together to achieve the user goal of data mining. Those functions are clustering, classification, and visualisation.

The proposed unified theoretical framework is based on the following assumptions which are also called the steps for knowledge extraction from a dataset [13-16]:

Step 1: Create partitions of the dataset.
Step 2: Create the clusters of each partition **(Clustering)**
Step 3: Construct the decision rules for each cluster **(Classification)**
Step 4: Plot the 2D or 3D graphs of each rule or classifier **(Visualisation)**

Khan et al [13] argue that,

"*The foundation of the proposed UDMT is that without clustering, there is no classification, without classification there is no visualization and hence without visualization, there is no knowledge*"

This emphasizes the interdependency of the functions as well as the order of how the functions should be performed. The first function (Clustering) feeds to the second (Classification) and consequently, the product of classification feeds to visualisation.

2.12 Critical Review of the Literature

The literature views data mining as a process of knowledge discovery. Expanding from the literature in section 2.9 to 2.11 in relation to data mining techniques, we have noted a fact that some efforts have been taken within the domain knowledge to define standards for the data mining processes [11-16]. This was done solely to improve justification and rationality of belief in the knowledge being developed for integration in the EDM domain.

In section 2.10, it was noted that the data mining techniques can be mapped to the traditional KDD process. The methodology (KDD) promotes a systematic approach to data mining were feature selection is integral to the process. However, it has been observed that this phase -*feature selection*- is the most ignored in the literature.

Adding to the discussion around the application of methodology, it has been observed that different researchers use a variety of methods to evaluate the accuracy and generalisation of the classification models. The most used metrics are the scalar quantities like Percentage of Correct Classification (PCC), Area Under Curve (AUC)

and Recall metrics. Hlaváč [48] argues that the scalar quantities like PCC, AUC and Recall alone do not provide enough information to assess the performance of the classifier. He went further and recommended the use of Receiver Operating Characteristics (ROC) curves and Precision-Recall curves to assess the cost of misclassification and possible trade-off between True Positive Rate (TPR) and True Negative Rate (TNR).

Finding enough data to conduct research that can later be generalized has also proved to be difficult for many researchers in the education context. Most of the literature reviewed had relatively little data available at their disposal to conduct their research. This has a tendency of limiting the researchers to the choice of algorithms to use for their research studies. An example of such studies was one conducted by Pardos et al [42] to predict student post-test scores for an intelligent tutoring system using a combination of ensembles and other classical algorithms. The study failed to support the hypothesis that ensemble techniques perform better, and the researcher relates this to insufficient data used in the study. In addition, insufficient data makes it very difficult to generalize the research findings. Loannidis [55] has drawn attention to the fact that a *"research finding is less likely to be true when the studies conducted in a field are smaller; when effect sizes are smaller; when there is a greater number and lesser pre-selection of tested relationships"*

In terms of the applicability of research in various reviewed literature, most of the researchers focused more on predicting learners' performance in higher learning institutions (Universities and Colleges). There was very little research done to predict learners' performance and grade progression in basic education (Grades1-12). The models developed for higher education or even basic education in other countries also cannot be expected to work in Limpopo due to the varying complexity around education policy imperatives and contextual sensitivity of the classification algorithms.

Several methodologists have pointed out that the high rate of non-replication is as a result of researchers claiming conclusive research findings solely on the basis of a single study assessed by formal statistical significance [55]. It is important that, the research report become descriptive in terms of the details of the methodology used and applicability of the research to encourage replication and to assess the reliability of the findings. Laws [54] correctly argues that until research can be reproduced by other researchers and arrive at the same findings, one cannot be confident that the findings will hold true in the long run.

2.13 Research Conceptual Framework

The work of Solanki [15] confirms Khan et al [14] argument that many tools in the market are single-step data mining tools and are not able to implement the UDMT. Solanki [15] conducted a study of three data mining tools. WEKA, Tanagra and Konstanz Information Miner (KNIME) were compared in their ability to implement UDMT. WEKA was found to be better in terms of implementing different data mining algorithms, followed by KNIME and lastly Tanagra. The study also found that all the tools under experiment (WEKA, Tanagra, and KNIME) lacked the following:

- Automatic selection of the appropriate algorithm for clustering, classification and visualisation
- The correct application of algorithm as function. That is, automatically taking the results of the previous algorithm to the next algorithm in the composite functions phase.

The UDMT promises a robust and reliable approach to data mining as compared to the single-step data mining tools. Considering the literature by Solanki [15] in relation to the inability of the tools to implement the UDMT, this study adopts the CRISP-DM technique as a classification methodology for the research. Figure 2.5 below associates the CRISP-DM method to various chapters in the thesis to create conceptual understanding of how the methodology was used to guide the research process.

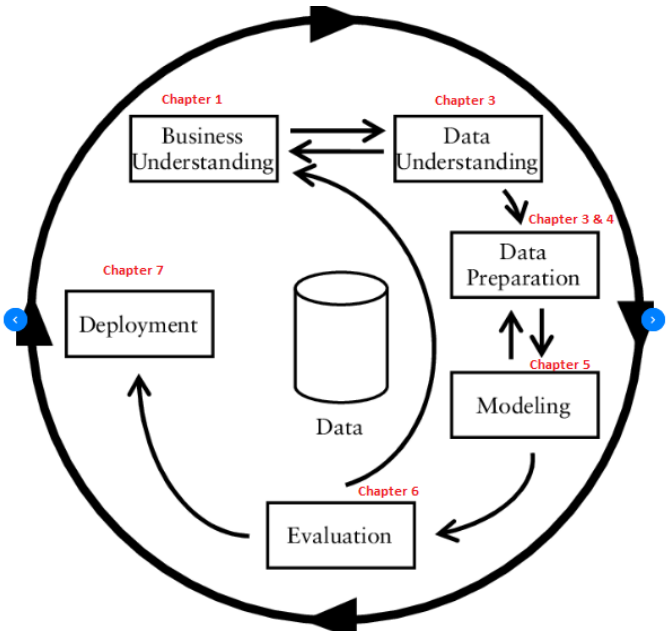


Figure 2.5: CRISP-DM life cycle

Refer to section 2.9.3 for a detailed discussion on the different phases of the CRISP-DM life cycle. The CRISP-DM will serve as a conceptual framework that will provide a blueprint to guide the composition of research literature, research methods and constructs and ultimately, guides in terms of answering the research questions.

2.14 Summary

This chapter introduced EDM and its implications in the education context. The literature related to the study was reviewed and synthesized to provide an argument for the choices made in the research. The chapter also provided the theoretical background of the research study, which embraces a process-centric approach to data mining. It discussed the common data mining techniques like KDD, SEMMA, CRISP-DM and UDMT in data mining, their similarities, and limitations. CRISP-DM was adopted to guide or provide a conceptual framework and structure for the research.

Chapter 3: Data Understanding and Preparation

3.1 Introduction

This chapter provides a discussion on EMIS as a source of data; in order to provide an appreciation of the data available for the research. Furthermore, the chapter will provide a detailed review of how the data was pre-processed and the tools used. The chapter will outline the process used to prepare data for the data mining process. In addition, the chapter will briefly discuss the Structured Query Language (SQL) run against the database to extract, transform, cleanse and consolidate data up until is loaded into WEKA (*data mining tool used in the study*).

3.2 Understanding EMIS Data Source

3.2.1 What Is EMIS?

The EMIS centralized database contains basic school-level data. In recent years, EMIS has transformed to host operational data of all the schools with unit level information about learners, educators, and schools. The EMIS's scope includes LURITS (Learner Unit Record Information Tracking System), South African School Administration and Management System (SA-SAMS), Business Intelligence, Geographic Information System, Data Quality Audits and more recently the Data Driven Districts Dashboard (DDD).

3.2.2 SA-SAMS as operational system for EMIS

SA-SAMS is a fully integrated electronic data management system for schools that collects a variety of operational data from schools through a number of modules. This data can be used for surveys, quarterly school reports and reporting for other educational programmes. SA-SAMS is continuously updated with new policies in order to assist schools with their data administration, management, and reporting.

3.2.3 EMIS System Description

The operational information about schools from the school administration systems is collected every quarter by EMIS personnel in the circuits, districts and provincial level. The EMIS officers from the circuits will collect information of schools under strict guidelines of the districts EMIS section and provincial EMIS unit. The collected databases will then be submitted to the District accompanied by the required documentation. The databases will then be quality assured and submitted to the provincial EMIS unit. The provincial EMIS unit will further quality assure the

databases, and accept or reject the databases based on the collection requirements. There is a range of quality assurance tools that are used during quality assurance and processing of the information, some are provided by National Department and some are developed in-house.

3.2.4 Description of WEKA 3.8.1

This research study uses WEKA to pre-process data and conduct experiments and related analysis. WEKA is a free and open source data mining tool developed by University of Waikato. The tool has been developed using Java and works conveniently as a standalone or can be called from the Java programming environment [44].

WEKA has a collection of machine learning algorithms for data mining tasks which, among others, includes tools for data pre-processing, classification, regression, clustering, association rules, and visualisation. It is also well-suited for developing new machine learning schemes [44].

User Interface

The WEKA main user interface is the **explorer**. However, the same functionality in the explorer can still be accessed through the component-based **Knowledge Flow** interface, **Command line** interface and the **Experimenter** Interface. The illustration below shows different panels that can be accessed from the WEKA tool:

Figure 3.1: WEKA panel window

(a) Explorer: The *Explorer* interface consists of different panels, providing access to the main components of the workbench [45].

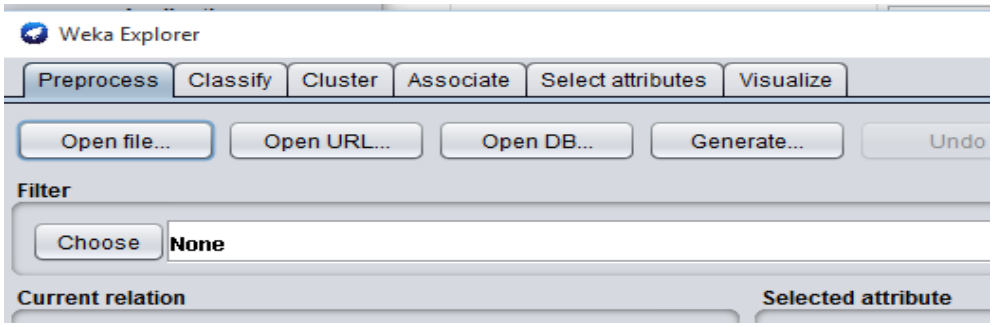


Figure 3.2: WEKA explorer window

- **Pre-Process** panel enables a user to import data from an external database into WEKA using data filtering algorithms. The pre-processing panel can further be used to access different data filtering algorithms to clean the data, remove and replace instances and attributes, transform the data to the required quality, save the data in **".arff"** format required by WEKA, etc.

- **Classify** panel provides a window to access different classification and regression algorithms and to run them against the dataset. It provides means to evaluate the accuracy of the algorithms as well as visualizing results of the model to make further observations on the data. Classification algorithms include Logistic Regression, NaïveBayes, Decision Tree, k-Nearest Neighbors and Support Vector Machines (SVM)

- **Associate** panel provides access to association rules that aim to identify interrelationships between data attribute. Examples of association rules includes apriori, filtered associator and FPgrowth

- **Cluster** panel provides access to clustering techniques like k- means, EM(Expectation Maximization), COBWEB(*hierarchical conceptual cluster*) , Canopy (*unsupervised pre-clustering algorithm*) , hierarchical clusterer, filtered clusterer, farthest first and the Density Based Spatial Clustering of Application with Noise (DBSCAN)

- **Select Attribute** panel provides algorithms to help identify the most predictive attributes in the data. Example of feature selection algorithms include Correlation, Gain-Ratio and Info-Gain etc. The attributes must be selected with a search algorithm like ranker, best first, and greedy stepwise

- Lastly, is the **Visualize** panel, which provides capabilities to visualize the data and with tools to help the user to evaluate the data

(b) Experimenter: It provides an environment for exploring and experimenting with different machine learning algorithms on datasets. You are still able to access all of the explorer functions from the experiment environment

(c) Knowledge flow: It provides an environment to automate the knowledge discovery process. Within the knowledge flow design canvas, you are able to put together different components and create a complex knowledge discovery process. It also supports incremental learning.

(d) Simple CLI: It provides a simple command line interface (CLI) that allows direct execution of WEKA commands**.**

3.3 Data Preparation and Experiment Execution Environment

Table 3.1 and 3.2 below illustrate the hardware and software specification used to set up the environment to prepare the data and conduct the experiments reported here:

Hardware

Table 3.1: Hardware used for the experiment

	Specification
System Model:	HP ProBook 450 G3
System Type:	x64-based PC
Processor:	Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz, 2400 Mhz, 2 Core(s), 4
Physical Memory (RAM)	4.00 GB
Total Physical Memory:	3.90 GB
Available Physical Memory:	909 MB
Total Virtual Memory:	8.15 GB
Available Virtual Memory	1.26 GB
Hard Drive	500 GB

Software

Table 3.2: Software used for experiment setup

Software	**Versions**	**Purpose**
Operating System	Windows 10	Enables operation of the computer
Data Mining Tool	WEKA 3.8.1	Developing, training and testing data mining classifiers for predictive purposes
Office Package	MS Office 2013	Documenting research process
MS SQL server	MS SQL server R2	Hosting data required for the research

3.4 Data Assembly

Expanding from the literature [22-27] discussed in chapter 2, learner biographical information, learners' performance, attendance information (of both learners and educators) and data attributes of the school (to acknowledge the context in which learning is happening) were used in the research study. The data was sourced from the SA-SAMS warehouse which is built on the Microsoft SQL server platform. The database consists of more than 400 tables (fact tables and related dimension tables) with millions of records on performance information about schools, learners, and educators for over four years.

In order to enable machine learning algorithms to understand the learners' performance patterns and be able to make a correct prediction, learners' performance data for two consecutive years were selected. This is supported by the study conducted by [20-24] in which they used CGPA among other attributes in their research. The CGPA is usually used at the College or University level to provide a better measurement of learner aptitude and knowledge comprehension abilities. It acknowledges the current and historical performance of the learner in a particular course. In this study, CGPA will be represented by the following attributes listed in table 3.1 below.

Table 3.3: Learners' performance attributes.

Attribute	Description	Data Year
PY = Previous Year (2015 data) CY = Current Year (2016 data)		
PY_LPDecision_Term1	Previous year term 1 learner promotion decision. *This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"*	2015
PY_LPDecision_Term2	Previous year term 2 learner promotion decision*. This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"*	2015
PY_LPDecision_Term3	Previous year term 3 learner promotion decision *This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"*	2015
PY_LPDecision_Term4	Previous year term 4 learner promotion decision. *This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"*	2015
PY_PQuality_Term4	Previous year term 4 promotion quality. *This is a numeric attribute; the marks of a learner in each of the subjects enrolled for are calculated out of 100 and averaged*	2015
CY_GradeYears	Number of years a learner is in the same grade. *This is a numeric attribute. The learner promotion policy prescribes that a learner can fail a grade only once in a phase. However, there are some exceptions in the data were a learner has failed more than once*	2016
CY_ProgressedStatus	Learner Progression Status. *The learner progression policy was passed in 2013 on a regulation gazette no 9886 of 28 December 2031. The policy was accompanied by clear guidelines on how to progress a learner. This is a nominal attribute and can either be **(True)** for a "conditional pass" or **(False)** for "not a conditional pass".*	2016
CY_LPDecision_Term1	Current year term 1 learner promotion decision *This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"*	2016
CY_PQuality_Term1	Current year term 1 promotion quality*. This is a numeric attribute; the marks of a learner in each of the subjects enrolled for are calculated out of 100 and averaged*	2016
CY_LPDecision_Term4	Current year term 4 promotion decision. *This is a nominal attribute and can either be **(P)** for "pass" or **(NP)** for "not pass"* **(Predictive Class)**	2016

In view of the discussions above, 2015 Term 1-4 and 2016 Term 1 learners' performance data were selected to provide a semblance of CGPA. It is safe to mention that the 2015 and 2016 were the most recent complete sets of data available at the time the research was undertaken. It is again important to note that the predictive class "CY_LPDecision_Term4" data were used during training and testing of the final model. The data instances in the predictive class are represented by "??" symbol when evaluating the accuracy of the model. **The study thus aimed at**

predicting student performance at the end of 2016 based on data at the end of the first term of that year, together with data from the previous year.

Expanding from the literature in section 2.6 , it is crucial that we include learner demographic information to enable better understanding of environmental and socio-economic factors affecting a learner and the ability to learn [1, 2]. In view of the latter, current year (2016) demographic information as listed in Table 3.4 below were also selected.

Four derived attributes were added, and were calculated as follows:

GPI (Gender Parity Index)
Purpose: The GPI measures progress towards gender parity in education. It also reflects the level of women's empowerment in society.
Category: Participation
Interpretation: the desired answer is 1, and this will imply that male and female learner enrolment are equal. If the answer is more than 1, it means we have more females than males; and less than one means we have more males than females
Formula

$$\frac{y}{x} \; where \; x = male \; enrolment, y = female \; enrolment$$

LER (Learner Educator Ratio)
Purpose: To measure levels of human resource input
Category: Internal Efficiency
Interpretation: The policy pre-scripts require 35 learners to 1 educator (35:1) for the secondary and **40:1** for primary
Formula

$$\frac{x}{y} \; where \; x = total \; school \; enrolment, y = number \; of \; educators \; at \; the \; school$$

Learner Absenteeism Rate
Purpose: To measure levels of learner class attendance
Category: Internal Efficiency
Interpretation: The desired answer is 0.The higher the answer, the more we have problems around learner class attendance. Current year (2016) Term 1 learner absenteeism has been used. Both the learner absentee rate and school absentee rate should be calculated using the formula below
Formula

$$\frac{x}{y} \times \frac{100}{z} where \; x = total \; days \; of \; learner \; absences,$$
$$y = total \; school \; enrolment, z = number \; of \; days \; per \; term$$

Educator Absenteeism Rate was calculated in an analogous way.

Table 3.4: Additional Learner demographic Information

Attribute	Description	Data Year
PY = Previous Year (2015 data); CY = Current Year (2016 data)		
CY_GPI	Current year Gender Parity Index. D*erived numeric attribute. Details in section 4.4*	2016
CY_LER	Current year Learner Educator Ratio. D*erived numeric attribute. Details in section 4.4*	2016
CY_SLAbsentee_Rate	Current year School Absentee Rate for learners. D*erived numeric attribute. Details in section 4.4*	2016
CY_SEAbsentee_Rate	Current year School Absentee Rate for educators. D*erived numeric attribute. Details in section 4.4*	2016
SQuintile	Poverty Indicator of the school. *Nominal value between 1 (poorest) and 5 (least poor) 1=Very Poor; 2=Fairly Poor; 3=Moderately Poor; 4 =Poor; 5=least Poor*	2016
SDistrict	The district a school is attached to. *Nominal value 1=Polokwane,2=Lebowakgomo,3=Riba cross,4=Sekhukhune,5=Waterberg,6=Mogalakwena,7=Tzaneen,8=Mopani,9=Vhembe,10=Tshipise-sagole*	2016
LCitizenship	Citizenship. *Nominal value.1 =Citizen; 2 = Immigrant*	2016
LGender	Gender. *nominal attribute and can either be 1 =male; 2 = female, 3=Unspecified*	2016
LHomeLanguage	The language spoken by a learner at home. *This is a nominal attribute and can take the following values 1=Afrikaans,2=English,3=IsiNdebele,4=SiSwati,5=IsiXhosa,6=IsiZulu,7=SeSotho,8=SePedi,9=SeTswana,10=TshiVenda,11=XiTsonga,12=Sign Language,13=Other*	2016
LRace	Population group. *Nominal. Can take the following values 1= African Black;2=Asian Indian; 3=Coloured;4=White;5=Other*	2016
LPhase	The phase of a grade a learner is in. *Nominal. Can take the following values* 1= Foundation, 2=Intermediate, 3=Senior,4=FET	2016
CY_Grade	Current Grade. *Nominal. Can take values between 1 and 12*	2016
CY_LAbsenteeRate_Term1	Current Year term 1 learner absentee rate. D*erived numeric attribute. Details in section 4.4*	2016

Description of groupings

The attributes are measuring the internal efficiency of the schools in terms of management of school policies like attendance, human resource input to teaching and learning, transformation agenda on gender equality and participation and have been grouped accordingly. Table 3.5 below summarises data fields extracted from SA-SAMS for the experiment and the order they appear on WEKA. The table consists of 23 fields; 19 were directly extracted from SA-SAMS warehouse and fields 1 to 4 have been derived/calculated.

Table 3.5: Summary of Data attributes extracted

Attribute		Attribute type	Entity	Attribute Category					
				Biographical	Efficiency	Location	Participation	Performance	Poverty
0	Seq		Learner						
1	CY_GPI	Numeric	Schools				X		
2	CY_LER	Numeric	School		X				
3	CY_SLAbsentee_Rate	Numeric	School		X				
4	CY_SEAbsentee_Rate	Numeric	School		X				
5	Squintile	Nominal	School						X
6	Sdistrict	Nominal	School			X			
7	Lcitizenship	Nominal	Learner	X					
8	Lgender	Nominal	Learner	X					
9	LhomeLanguage	Nominal	Learner	X					
10	Lrace	Nominal	Learner	X					
11	Lphase	Nominal	Learner	X					
12	CY_Grade	Nominal	Learner	X					
13	CY_GradeYears	Numeric	Learner					X	
14	CY_ProgressedStatus	Nominal	Learner					X	
15	PY_LPDecision_Term1	Nominal	Learner					X	
16	PY_LPDecision_Term2	Nominal	Learner					X	
17	PY_LPDecision_Term3	Nominal	Learner					X	
18	PY_LPDecision_Term4	Nominal	Learner					X	
19	PY_PQuality_Term4	Numeric	Learner					X	
20	CY_LAbsenteeRate_Term1	Numeric	Learner		X				
21	CY_LPDecision_Term1	Nominal	Learner					X	
22	CY_PQuality_Term1	Numeric	Learner					X	
23	CY_LPDecision_Term4	Nominal	Learner					X	

3.5 Data Cleansing

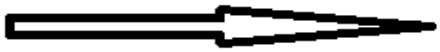
Data cleaning refers to a process of identifying or detecting data anomalies; correcting inaccurate data by modifying, replacing missing fields or removing incomplete sets of data.

A range of data anomalies were observed in the data and a set of SQL statements were run to cleanse the data. Below is an example of an SQL statement written to correct, for example, the “citizenship” field.

```
/****** Script for SelectTopNRows command from SSMS  ******/
/*Step 1 : Displaying list values under citizenship attribute */
select citizenship from research.dbo.Final_Transformed_Data
group by citizenship
/*Step 2 : Cleaning list values under citizenship attribute*/
Update research.dbo.Final_Transformed_Data set [citizenship] =
(Case
    when citizenship = 'SADC' then '2' /* Immigrant */
    when citizenship = 'SA' then '1' /* Citizen */
    when citizenship = 'Immigrant' then '2'
    when citizenship = 'SA Citizen' then '1'
    when citizenship = 'Ghana' then '2'
    when citizenship = 'Congo' then '2'
    when citizenship = 'Pakistan' then '2'
    when citizenship = 'SA Citizenship' then '1'
end)
```

Results | Messages

	citizenship
1	SADC
2	SA
3	Immigrant
4	INDIA
5	SA Citizen
6	ZIMBABWE
7	SA Citizen
8	Ghana
9	CONGO
10	PAKISTAN
11	SA Citizenship



Cleaned list Values

Results | Messages

	citizenship
1	2
2	NULL
3	1

Figure 3.3: Cleaning of citizenship field

You will observe that South African citizenship has been written differently in the data as “SA”, “SA Citizenship” and “SA Citizen”. The SQL statement (labelled step2) modifies the data to either 1=Citizen or 2= Immigrant. The same process was used to clean the nominal attributes like race, gender, phase, grade, progressedstatus etc. Refer to Annexure A for a set of SQL statements used to clean all the nominal attributes.

3.6 Data Aggregation and Integration

After the data was cleaned in different tables, the SQL statements were run against the data to create a flat file structure where each learner record is mapped to a single row. Refer to Annexure A **(***A.4, A.5, and A.6***)** for a range of SQL statements used to aggregate and integrate the data. Figure 3.4 below illustrates the final dataset after aggregation and consolidation was conducted.

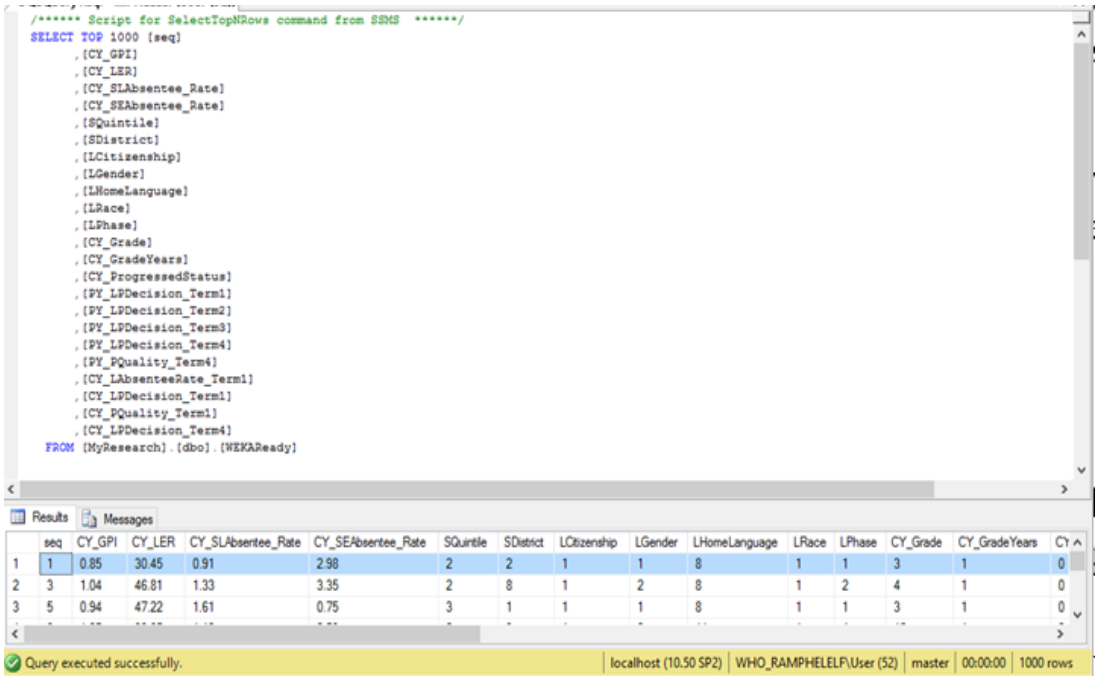


Figure 3.4: Final dataset ready for WEKA

3.7 Importing Data from SQL Server into WEKA

The jdbc driver bridge (JDBC Driver 6.0) was downloaded and configured to provide a connection interface between WEKA and MS SQL server. The heap-size for JVM (Java Visual Machine) was increased to "**maxheap=2048m"** so it is able to handle processing requirements of WEKA.

The connection string ***"jdbc:sqlserver:///localhost;databaseName=Research***"was used to connect WEKA to the "Research" table in the SQL server hosting the prepared data for the study.

The SQL statement ***"SELECT * FROM [Research].[dbo].[WekaReady]"*** was used to load data into WEKA. Figure 3.5 below shows an explorer window with the data already imported into WEKA for further processing.

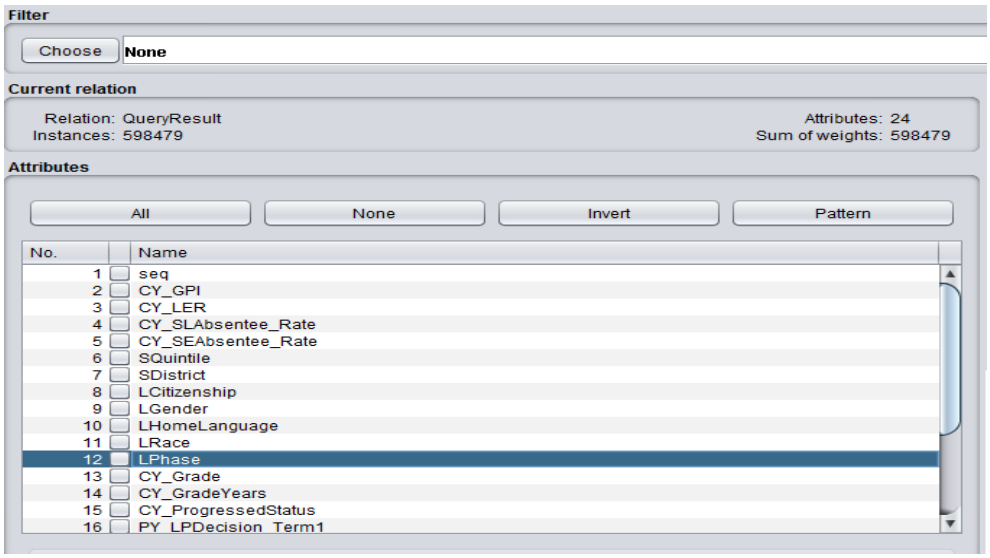


Figure 3.5: WEKA data pre-processing panel

A total of 23 (*seq not counted*) attributes and 598479 records of individual learners were loaded into WEKA for further processing.

3.8 Assessment of Original Data

3.8.1 Class separation and data distribution

Reference to Ghasemi et al [63], normality test on the data is important, particularly when parametric tests like t-test, analysis of variance, correlation, standard deviation, etc. are going to be used to make reliable conclusions about the reality presented by the data.

Figure 3.6 provides visualisation of normality plots of different variables from the raw data. It can be observed that all the variables are either skewed to the left or the right of their median except for “CY_GPI” (Gender Parity) which shows some Gaussian/Normal distribution except for the extended kurtosis that could have been caused by the noise in the data. It can also be observed that “CY_SEAbsentee_Rate” has values that extend beyond 100% in terms of the maximum values for that attribute which is wrong. It is safe to indicate that it is as a result of data anomaly from one school (about 40 learners) and that provides an inconsistent view.

The literature [63] by Ghasem went further to say, with a large enough sample size of more than 30%, the violation of normality assumptions is insignificant and therefore, parametric procedures can still be used even when the data is not normally distributed. It is safe to indicate that the sample used is about 35% of the entire

population, and therefore we are safe to use parametric tests to further guide the feature selection process.

Figure 3.6 and 3.7 shows a data distribution plot and scatter plot matrix respectively and provides a better visualisation of the class separation problem inherent in the data .Variables with clear class separation are expected to provide a better prediction impact. It can be observed from both figure 3.6 and 3.7 that grade years and promotion quality are expected to provide a better prediction impact than the rest of the variables.

The varied nature of the data will require rescaling of the attributes in the data to ensure the same treatment during processing as well as to uncover the underlying structure of the data.

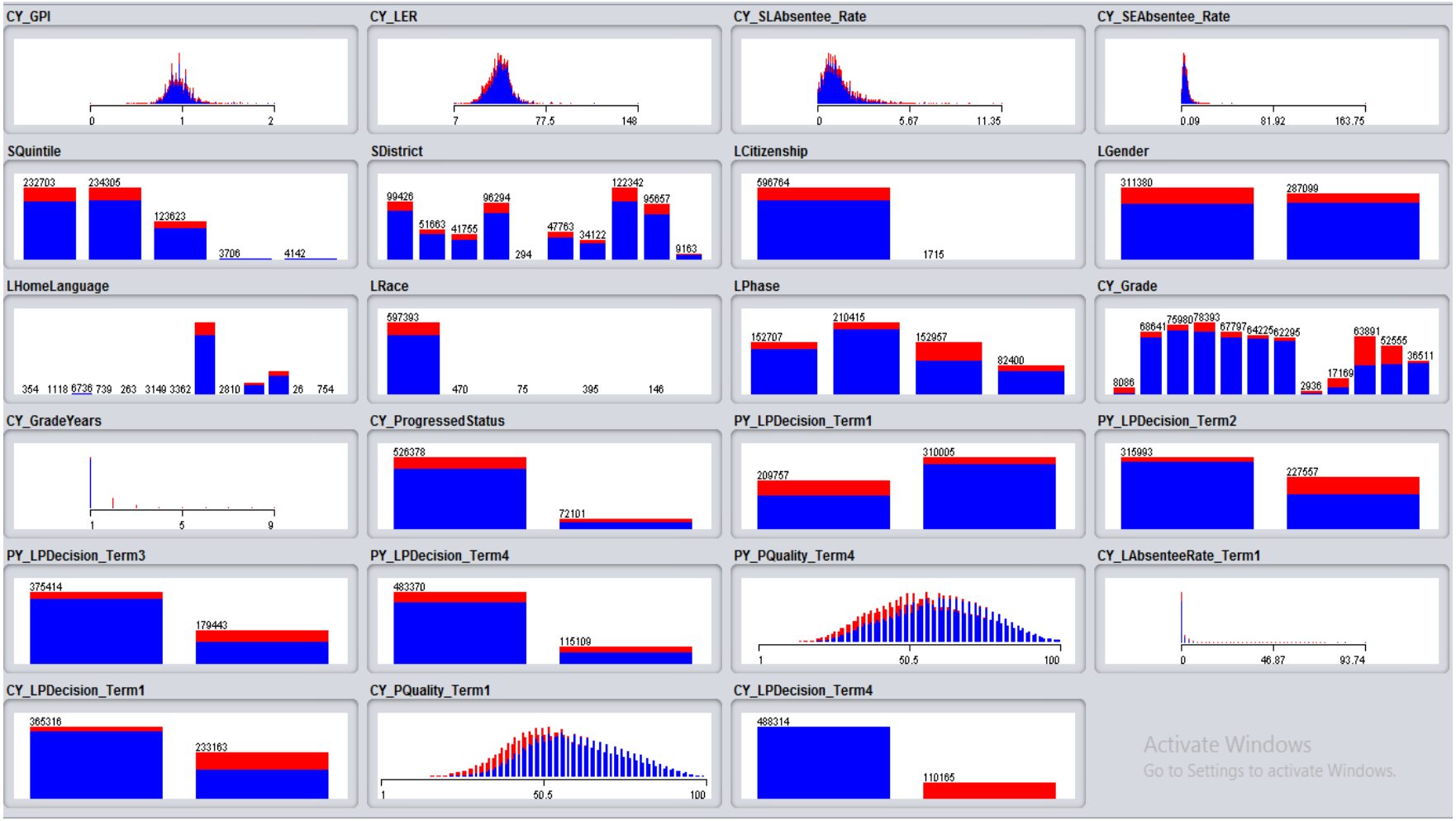


Figure 3.6: Data Distribution

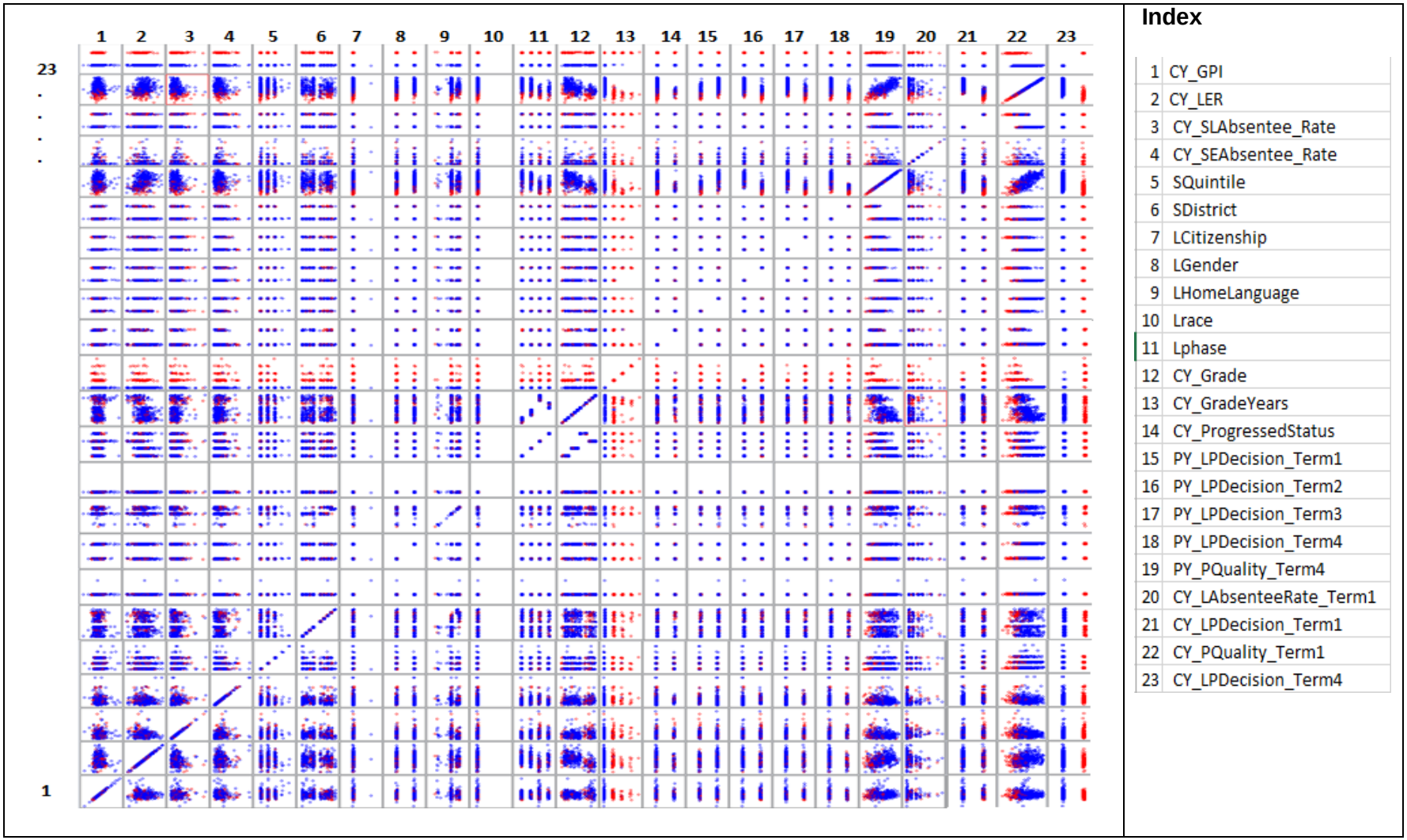


Figure 3.7: Scatter Plot Matrix

3.8.2 Correlation and collinearity

According to Shong-Chok [67], Pearson's correlation coefficient is sensitive to skewed data distributions and outliers. Shong-Chok [67] suggested the use of non-parametric correlation functions like Spearman's correlation when dealing with data that is not normally distributed and we are not sure of linearity of relationships among the variables.

As discussed in section 3.8.1 in respect of the raw data lacking a normal distribution, this motivates the author to use the Spearman's correlation function which is non-parametric, to assess the correlation and collinearity of the attributes. The following guide is commonly used to determine the effect or strength of correlation between the variables [67]:

- *0.00 to 0.19 "very weak"*
- *0.20 to 0.39 "weak"*
- *0.40 to 0.59 "moderate"*
- *0.60 to 0.79 "strong"*
- *0.80 to 1.0 "very strong*

It must be noted that the correlation is an absolute value where the sign only shows the direction of the relationship.

Considering the discussions and guidelines above, it can be observed from figure 3.8 and 3.9 that both "grade years" and "promotion quality" have a notable relationship with the predictive class. Using guidelines above, the relationship of the two attributes can be described as strong and moderate respectively

Multicollinearity arises when there is high correlations between predictor variables in a manner that, one predictor variable can be used to predict another. Reference to Schreiber and Jackson [64], multicollinearity creates redundant information in the data and has a potential to skew the results of the classification model. He further asserts that,

"In the incidence of multicollinearity, it is difficult to come up with reliable estimates of individual coefficients for the predictor variables in a model which results in incorrect conclusions about the relationship between outcome and predictor variables"

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	CY_GPI	1.00	-0.04	0.00	0.06	0.09	-0.02	0.00	0.06	0.01	0.01	0.15	0.15	0.02	0.03	0.00	0.03	-0.02	0.02	-0.07	0.00	-0.01	-0.04	0.02
2	CY_LER	-0.04	1.00	-0.18	0.02	0.07	0.00	0.02	0.01	0.01	-0.01	-0.46	-0.45	-0.18	-0.16	-0.12	-0.19	-0.09	-0.22	0.37	-0.07	-0.12	0.30	-0.19
3	CY_SLAbsentee_Rate	0.00	-0.18	1.00	0.17	0.02	-0.12	0.00	-0.01	-0.18	-0.01	0.23	0.23	0.11	0.11	0.10	0.16	0.10	0.15	-0.23	0.24	0.11	-0.19	0.13
4	CY_SEAbsentee_Rate	0.06	0.02	0.17	1.00	-0.08	-0.15	0.00	0.00	-0.15	-0.01	0.04	0.04	0.03	0.02	0.01	0.02	0.01	0.03	-0.05	0.04	0.04	-0.05	0.03
5	SQuintile	0.09	0.07	0.02	-0.08	1.00	-0.01	0.02	0.01	0.09	0.05	0.03	0.03	-0.03	0.03	-0.01	-0.01	-0.02	-0.02	0.02	0.00	-0.03	0.03	-0.03
6	SDistrict	-0.02	0.00	-0.12	-0.15	-0.01	1.00	0.01	0.00	0.66	-0.02	-0.02	-0.03	0.03	0.04	-0.01	-0.04	-0.03	0.03	-0.01	-0.03	-0.01	0.00	0.02
7	LCitizenship	0.00	0.02	0.00	0.00	0.02	0.01	1.00	0.00	0.05	0.15	-0.02	-0.02	-0.01	-0.01	-0.01	-0.01	-0.01	-0.01	0.02	0.00	-0.01	0.02	-0.01
8	LGender	0.06	0.01	-0.01	0.00	0.01	0.00	0.00	1.00	0.00	0.00	0.01	0.02	-0.11	-0.07	-0.09	-0.10	-0.12	-0.10	0.17	-0.04	-0.16	0.18	-0.11
9	LHomeLanguage	0.01	0.01	-0.18	-0.15	0.09	0.66	0.05	0.00	1.00	-0.02	-0.03	-0.03	0.00	0.03	-0.03	-0.05	-0.02	0.01	0.00	-0.04	-0.02	0.01	0.01
10	LRace	0.01	-0.01	-0.01	-0.01	0.05	-0.02	0.15	0.00	-0.02	1.00	-0.02	-0.02	-0.01	-0.01	-0.01	-0.01	-0.01	-0.01	0.02	0.00	-0.01	0.02	-0.01
11	LPhase	0.15	-0.46	0.23	0.04	0.03	-0.02	-0.02	0.01	-0.03	-0.02	1.00	0.97	0.19	0.24	0.19	0.30	0.13	0.26	-0.58	0.05	0.21	-0.49	0.21
12	CY_Grade	0.15	-0.45	0.23	0.04	0.03	-0.03	-0.02	0.02	-0.03	-0.02	0.97	1.00	0.15	0.23	0.18	0.30	0.13	0.25	-0.59	0.05	0.18	-0.49	0.17
13	CY_GradeYears	0.02	-0.18	0.11	0.03	-0.03	0.03	-0.01	-0.11	0.00	-0.01	0.19	0.15	1.00	0.12	0.17	0.22	0.20	0.19	-0.38	0.10	0.37	-0.47	0.90
14	CY_ProgressedStatus	0.03	-0.16	0.11	0.02	0.03	0.04	-0.01	-0.07	0.03	-0.01	0.24	0.23	0.12	1.00	0.17	0.22	0.19	0.15	-0.35	0.06	0.26	-0.34	0.13
15	PY_LPDecision_Term1	0.00	-0.12	0.10	0.01	-0.01	-0.01	-0.01	-0.09	-0.03	-0.01	0.19	0.18	0.17	0.17	1.00	0.56	0.44	0.25	-0.41	0.05	0.27	-0.32	0.18
16	PY_LPDecision_Term2	0.03	-0.19	0.16	0.02	-0.01	-0.04	-0.01	-0.10	-0.05	-0.01	0.30	0.30	0.22	0.22	0.56	1.00	0.53	0.36	-0.55	0.08	0.30	-0.41	0.24
17	PY_LPDecision_Term3	-0.02	-0.09	0.10	0.01	-0.02	-0.03	-0.01	-0.12	-0.02	-0.01	0.13	0.13	0.20	0.19	0.44	0.53	1.00	0.34	-0.45	0.07	0.30	-0.34	0.21
18	PY_LPDecision_Term4	0.02	-0.22	0.15	0.03	-0.02	0.03	-0.01	-0.10	0.01	-0.01	0.26	0.25	0.19	0.15	0.25	0.36	0.34	1.00	-0.60	0.10	0.18	-0.29	0.18
19	PY_PQuality_Term4	-0.07	0.37	-0.23	-0.05	0.02	-0.01	0.02	0.17	0.00	0.02	-0.58	-0.59	-0.38	-0.35	-0.41	-0.55	-0.45	-0.60	1.00	-0.12	-0.45	0.73	-0.40
20	CY_LAbsenteeRate_Ter	0.00	-0.07	0.24	0.04	0.00	-0.03	0.00	-0.04	-0.04	0.00	0.05	0.05	0.10	0.06	0.05	0.08	0.07	0.10	-0.12	1.00	0.08	-0.12	0.10
21	CY_LPDecision_Term1	-0.01	-0.12	0.11	0.04	-0.03	-0.01	-0.01	-0.16	-0.02	-0.01	0.21	0.18	0.37	0.26	0.27	0.30	0.30	0.18	-0.45	0.08	1.00	-0.69	0.40
22	CY_PQuality_Term1	-0.04	0.30	-0.19	-0.05	0.03	0.00	0.02	0.18	0.01	0.02	-0.49	-0.49	-0.47	-0.34	-0.32	-0.41	-0.34	-0.29	0.73	-0.12	-0.69	1.00	-0.51
23	CY_LPDecision_Term4	0.02	-0.19	0.13	0.03	-0.03	0.02	-0.01	-0.11	0.01	-0.01	0.21	0.17	0.90	0.13	0.18	0.24	0.21	0.18	-0.40	0.10	0.40	-0.51	1.00

Figure 3.8: Spearman Correlation Matrix-Raw Data

Notes to guide interpretation of the correlation matrix

- The distribution of each attribute is shown on the diagonal.
- On each cross-section is the correlation value the two attributes involved
- The correlation of 0.50 and more is considered moderate to strong respectively in determining the strength of the relationship between the two variable (Refer to section 4.9.2 for Evans (1996) strength assessment guidelines)

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	CY_GPI		0.000	0.051	0.000	0.000	0.000	0.073	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.083	0.000	0.000	0.000	0.000	0.173	0.000	0.000	0.000
2	CY_LER	0.000		0.000	0.000	0.000	0.065	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	CY_SLAbsentee_Rate	0.051	0.000		0.000	0.000	0.000	0.657	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
4	CY_SEAbsentee_Rate	0.000	0.000	0.000		0.000	0.000	0.027	0.015	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	SQuintile	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	SDistrict	0.000	0.065	0.000	0.000	0.000		0.000	0.166	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.338	0.000
7	LCitizenship	0.073	0.000	0.657	0.027	0.000	0.000		0.585	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.104	0.000	0.000	0.000
8	LGender	0.000	0.000	0.000	0.015	0.000	0.166	0.585		0.088	0.717	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
9	LHomeLanguage	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.088		0.000	0.000	0.000	0.005	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
10	LRace	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.717	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.014	0.000	0.000	0.000
11	LPhase	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
12	CY_Grade	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
13	CY_GradeYears	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.005	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
14	CY_ProgressedStatus	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
15	PY_LPDecision_Term1	0.083	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
16	PY_LPDecision_Term2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000	0.000
17	PY_LPDecision_Term3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000	0.000
18	PY_LPDecision_Term4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000	0.000
19	PY_PQuality_Term4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000	0.000
20	CY_LAbsenteeRate_Term1	0.173	0.000	0.000	0.000	0.000	0.000	0.104	0.000	0.000	0.014	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000	0.000
21	CY_LPDecision_Term1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000	0.000
22	CY_PQuality_Term1	0.000	0.000	0.000	0.000	0.000	0.338	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		0.000
23	CY_LPDecision_Term4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	

Figure 3.9: Correlation Significance levels-Raw Data

Notes to guide interpretation of the significance level matrix

- The distribution of each attribute is shown on the diagonal.
- On each cross-section is the significance value of the two attributes involved
- The significance level is between 0 and 1 where 0 means statistically significance; and 1 mean no significance. Any value more than 0.05 is considered not significant.

In view of the above discussions and the fact that the study also intends to understand clearly which variables are important to predict performance, it is important to analyse collinearity among the variables to ensure that we are able to eliminate any confusion that can lead us to incorrect conclusions. Refer to figure 3.8 and 3.9 which illustrate correlation matrix and significance levels of all the attributes respectively; the following can be observed.

- Phase and Grade have a high collinearity. This is because a separate group of grades forms each phase. It can further be observed that phase contributes more to predictive class than the grade
- There is a high collinearity between promotion quality and promotion decision. This is true because the lower the promotion quality, the higher the risk of failing
- District and home-language have high collinearity and this is because the districts are also demarcated based on different ethnic groups

It is safe to indicate that the strength of the correlation does not automatically imply there is a causal relationship between the variables. However, the assertions made in the observations above in terms of causality are based on common knowledge about attributes.

3.8.3 Standard Deviation

Standard deviation is a statistical method that calculates the amount of dispersion of features from the average. According to Yousefpour [62], the standard deviation can provide a good estimate of the impact of an attribute on a predictive class. A higher standard deviation is associated with a higher impact of the feature on a predictive class [62]. Table 3.6 below shows the standard deviation, mean, variance and average deviation of the raw data

Table 3.6: Relative Standards deviation of numeric attributes

		Mean	Std.Dev	Variance	Min	Max	Ave.Dev
1	CY_GPI	0.96	0.12	0.02	0.00	2.00	0.09
2	CY_LER	41.80	8.33	69.40	7.00	148.00	6.34
22	CY_PQuality_Term1	56.76	15.74	247.69	1.00	100.00	12.86
19	PY_PQuality_Term4	57.37	17.00	288.86	1.00	100.00	14.03
13	CY_GradeYears	1.23	0.55	0.30	1.00	9.00	0.38
17	PY_LPDecision_Term3	1.23	0.57	0.32	0.00	2.00	0.46
4	CY_SEAbsentee_Rate	4.33	3.17	10.07	0.09	100.00	2.08
3	CY_SLAbsentee_Rate	1.32	1.04	1.09	0.00	11.35	0.72
20	CY_LAbsenteeRate_Term1	0.78	2.39	5.73	0.00	93.74	1.27

Based on Yousefpour [62] assertions, we can safely deduce that *CY_LER, CY_PQuality_Term1, PY_PQuality _Term4* are good predictor variables.

3.8.4 Data Rescaling

Expanding from section 3.8.1 above, the assessment of the original data revealed that the raw data need to be rescaled particularly on the continuous numeric variables. This will assist in ensuring that the noise in the data is reduced; and that all the attributes in the data will get equal treatment during the processing. According to Ruoming et al [68], many data mining systems cannot correctly handle continuous attributes. He further affirms that the problem can be resolved by replacing continuous attributes with discretised intervals [68] and this will improve quality of discovered knowledge and also lessen the running time of data mining activities.

In view of the discussion above, the raw data was discretised using unsupervised filter "*weka.filters.unsupervised._attribute.Discretize"* and converted into bins/subranges using equal frequency binning. Three bins were used each time, as the number of bins being one greater than the number of classes is generally best. Annexure E shows the final discretised data structure. It can be observed from the data structure that the numeric attributes like CY_GPI, CY_LER, CY_SLAbsentee_Rate, CY_SEAbsentee_Rate, CY_GradeYears and CY_LAbsenteeRate_Term 1 has been discretised while other attributes remain unchanged. This will serve as the initial data in the feature selection process described in the next chapter.

3.9 Summary

This chapter discussed the role of EMIS within the Limpopo Education system and the application landscape they use to collect, process and disseminate information for planning and resource provisioning purposes. The chapter further discussed WEKA, the data mining tool used for the research as well as different functions and features available in the tool. The chapter explored the data pre-processing activities as well as different statistical parametric tests conducted in order to understand the data and inform further decisions around the features to be used in the study. In addition, different data scaling methods were applied to the raw data to improve classification accuracy.

Chapter 4: Feature Selection

4.1 Introduction

This chapter outlines the exploratory study conducted on feature selection.

4.2 Feature Selection

Feature Selection is an important step in data pre-processing aiming at simplifying the model for better prediction accuracy as well as reducing the required computational power. The literature by Jovic et al [46] further asserts that the feature selection methods are context responsive and do not respond the same with similar data. It is therefore important to select the best feature selection algorithms that understand the data at disposal and the goal of data mining process [46].

Two exploratory studies were conducted to enable the selection of the best feature set for the study. The discussion and the results of the two studies are provided in section 4.2.1 and 4.2.2:

4.2.1 First Exploratory Study (Feature Selection Filters)

In the first study, Correlation attribute, Info-Gain and Symmetrical Uncertainty attribute evaluation were applied to the data produced in chapter 3 (*discretised dataset*).

Table 4.1 shows the result of the three filter based feature selection methods applied to the discretised data. If we are to interpret the results of correlation attribute evaluation based on the guildelines provided by Shong-Chok [67] on section 3.8.2, only the “GradeYears” and “CY_LPDecision_Term1” will be selected since the strength of their relationship with the predictive class is strong. However, it must be mentioned that the results of the correlation attribute evaluation in table 4.1 differ with the correlation matrix in figure 3.8. The results of correlation feature evaluation filter favoured “Promotion Decisions” which is course grained as (“N” & “P”) over “Promotion Quality” which takes categorical values between 1 and 7. The only explanation for that is that the correlation attribute evaluation uses Pearson’s correlation function while the correlation matrix in figure 3.8 was calculated on the Spearman’s correlation function. Pearson’s correlation processes data based on the assumption of a normal distribution of the data. If these conditions fail, the validity of the Person’s correlation can be questioned. However, and as discussed in section 3.8 in chapter 3, the population size used promised validity even when the data is not

normally distributed [35]. These results are unexpected. It must be noted that Spearman correlation is non-parametric and is not based on any assumptions about the data.

Table 4.1: Attribute evaluation results

		Ranking Position			Weight		
	Attribute	**Correlation**	**Info-Gain**	**Symmetric**	**Correlation**	**Info-Gain**	**Symmetric**
12	CY_GradeYears	1	1	1	0.8625	0.5073	0.6738
21	CY_PQuality_Term1	2	3	6	0.1910	0.2157	0.1306
18	PY_PQuality_Term4	3	5	9	0.1385	0.1279	0.0756
20	CY_LPDecision_Term1	4	2	2	0.3982	0.1147	0.1387
15	PY_LPDecision_Term2	5	4	3	0.2943	0.0647	0.0775
16	PY_LPDecision_Term3	6	6	4	0.2701	0.0511	0.0640
11	LPhase	7	8	8	0.1441	0.0476	0.0363
14	PY_LPDecision_Term1	8	7	5	0.2429	0.0421	0.0507
2	CY_LER	9	10	10	0.1290	0.0269	0.0237
17	PY_LPDecision_Term4	10	9	7	0.1843	0.0219	0.0314
3	CY_SLAbsentee_Rate	11	13	14	0.0846	0.0113	0.0099
13	CY_ProgressedStatus	12	11	11	0.1259	0.0101	0.0166
8	LGender	13	12	12	0.1119	0.0092	0.0108
19	CY_LAbsenteeRate_Term1	14	14	13	0.0919	0.0077	0.0098
5	SQuintile	15	15	17	0.0150	0.0016	0.0014
6	SDistrict	16	16	18	0.0132	0.0016	0.0009
1	CY_GPI	17	17	15	0.0215	0.0009	0.0008
4	CY_SEAbsentee_Rate	18	18	16	0.0211	0.0007	0.0006
9	LHomeLanguage	19	20	21	0.0102	0.0004	0.0003
10	LRace	20	19	20	0.0111	0.0001	0.0004
7	LCitizenship	21	21	19	0.0116	0.0001	0.0003

In respect of both the Info-Gain and Symmetric Uncertainty, the results look more similar. Refer to figure 4.1.

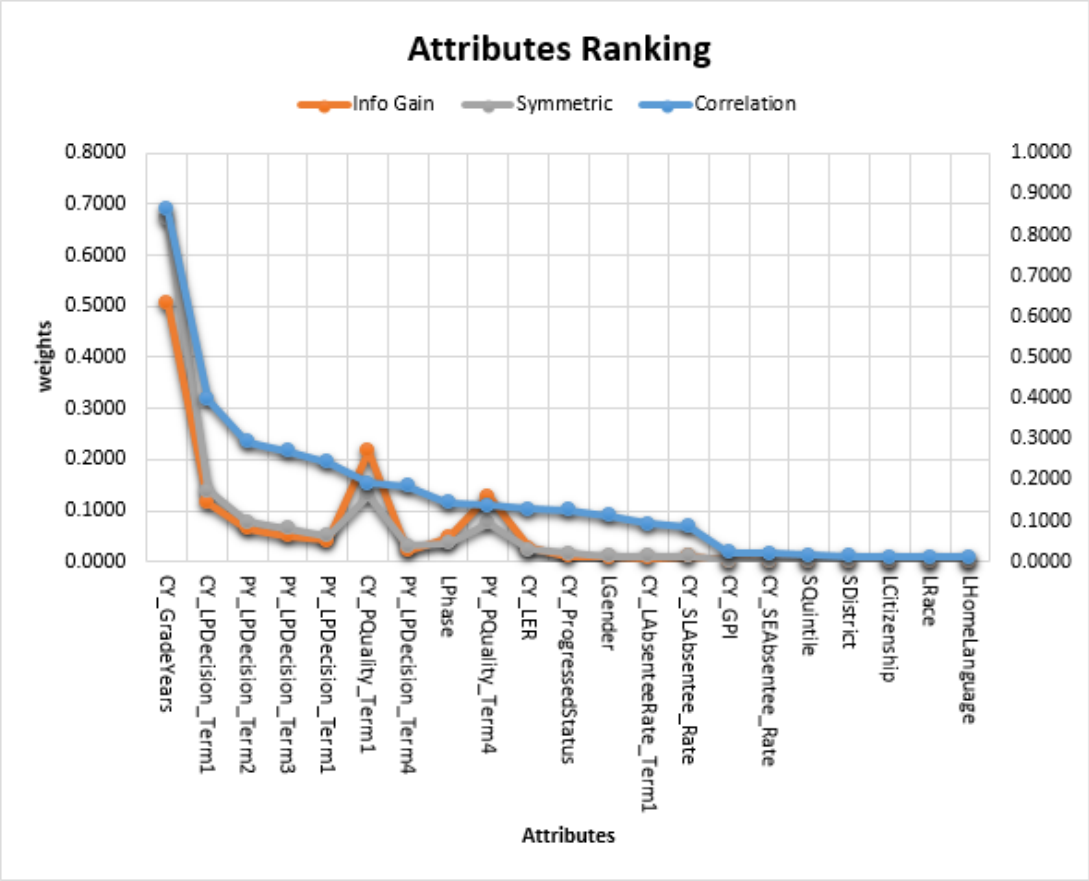
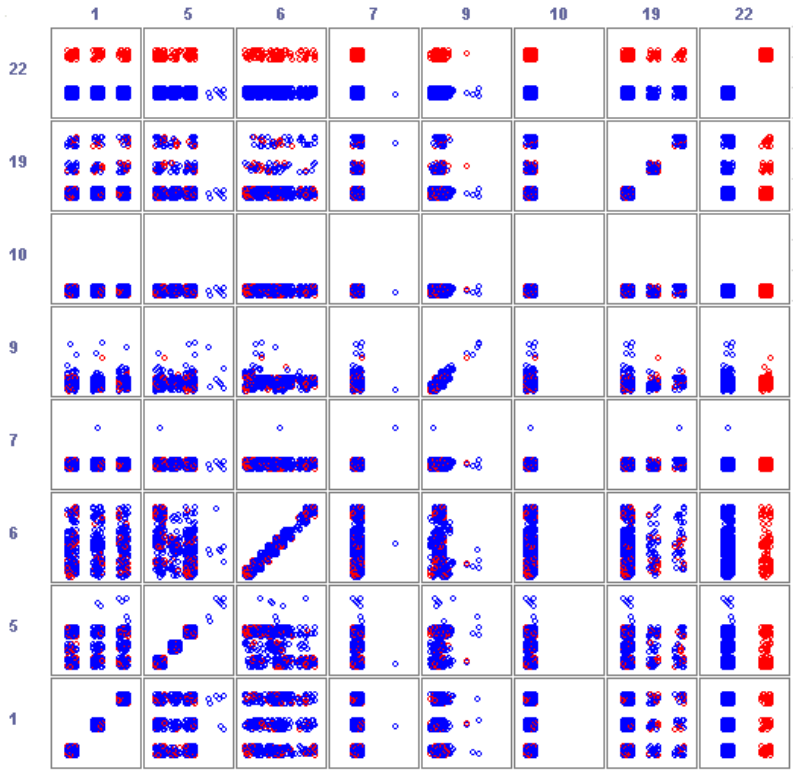


Figure 4.1: Filter based feature selection weights

It can be observed that both Symmetric Uncertainty and Info-Gain plots have the same pattern even though the weights of attributes are the same. The reason for this is because both of the algorithms are using entropy to calculate information gain. The literature is silent in terms of providing guidelines on how to choose the best Info-Gain threshold for the best contributing attributes. However, as discussed above, Info-Gain can only take a value between 0 and 1, where 0 = no relationship and 1 = available relationship between the variables. In view of that, we can make deductions that "grade years" has a strong relationship with the predictive class in both Symmetric Uncertainty and Information Gain.

It is interesting that the three feature selection methods returned the seven attributes *CY_GPI, SQuintile, SDistrict, LHomeLanguage, LRace,* CY_SEAbsentee_Rate *and LCitizenship* as the bottom 7 with lowest impact on the predictive class, as did Spearman's correlation shown in figure 3.8.

Furthermore, to observe the relationship of the seven attributes against the predictive class, the scatter plot matrix in figure 4.2 was generated from WEKA.



	Attributes
1	CY_GPI
5	SQuintile
6	SDistrict
7	LCitizenship
9	LHomeLanguage
10	Lrace
19	CY_LAbsenteeRate_Term1
22	CY_LPDecision_Term4

Figure 4.2: Scatter Plot Matrix (Least Contributing Attributes)

It can be observed from the scatter plot that most of the learners are of same Race and Citizenship (South Africans). Further to that, all the seven attributes (***CY_GPI, SQuintile, SDistrict, LHomeLanguage, LRace, CY_SEAbsentee_Rate and LCitizenship)*** do not show any clustering/class separation of the two predictive class values (P and NP, shown as red and blue respectively) and therefore, it makes sense why they were found to have a low predictive power during the feature selection experiment.

The first study thus investigated a selected feature set **(Set A)** made by removing the seven attributes from the dataset. Experiments were performed to assess the classification accuracy of the model without the seven attributes.

4.2.2 Second Exploratory Study (Forward Stepwise Feature Evaluation)

The second exploratory study used a manual forward stepwise technique to evaluate the impact of individual attribute against the predictive class. The method requires the addition of one variable at a time. At each stage, the added variable is tested for its impact on increasing the performance of the model. The most significant or principal

variables are then added to the final model to solve the classification problem at hand [19].

The inclusion order of the attributes in the forward stepwise space is usually guided by the predictive power of the attribute. Different methods, including correlation coefficient, standard deviation, etc. can be used to guide the attribute succession order in forward stepwise space. In view of the above, Correlation, Info-Gain and Symmetrical Uncertainty attribute evaluation and Spearman Correlation attributes ranking in terms of predictive power were used to determine the order of variable inclusion into the forward stepwise technique. Table 4.2 shows the predictive power ranking of the variables against the feature selection methods listed above as well as the Spearman correlation coefficient of the raw data.

Table 4.2: Attributes Predictive Power Ranking

		Attributes Rank			
	Attribute	**Info-Gain**	**Symmetric**	**Correlation**	**Spearman**
12	CY_GradeYears	1	1	1	1
21	CY_PQuality_Term1	2	3	6	2
18	PY_PQuality_Term4	3	5	9	3
20	CY_LPDecision_Term1	4	2	2	4
15	PY_LPDecision_Term2	5	4	3	5
16	PY_LPDecision_Term3	6	6	4	7
11	LPhase	7	8	8	6
14	PY_LPDecision_Term1	8	7	5	10
2	CY_LER	9	10	10	8
17	PY_LPDecision_Term4	10	9	7	9
3	CY_SLAbsentee_Rate	11	13	14	12
13	CY_ProgressedStatus	12	11	11	13
8	LGender	13	12	12	14
19	CY_LAbsenteeRate_Term1	14	14	13	15
5	SQuintile	15	15	17	17
6	SDistrict	16	16	18	19
1	CY_GPI	17	17	15	18
4	CY_SEAbsentee_Rate	18	18	16	16
9	LHomeLanguage	19	20	21	22
10	LRace	20	19	20	21
7	LCitizenship	21	21	19	20
Total Mappings		**13.00**	**14.00**	**9.00**	**8**

The predictive power rankings that are similar in both the filter based attribute selection method and the correlation coefficient are highlighted. It can be observed that Symmetrical attribute evaluation ranking matched other filters and the correlation coefficient best. The ranking of Symmetrical attribute was then used to conduct forward stepwise feature evaluation and the results in respect of the PCC are shown in table 4.3.

Table 4.3: Results of the forward Stepwise (PCC)

Feature Set	Attributes Indices **22 = Predictive Class** *(CY_LPDecision_Term4)*	HoeffdingTree	NaïveBayes	Adaboost M1(DS)
1	22,12	94.6	94.6	94.6
2	22,12,21	94.6	94.5	94.6
3	22,12,21,20	94.6	93.3	94.6
4	22,12,21,20,18	94.6	91.0	94.6
5	22,12,21,20,18,15	94.6	90.2	94.6
6	22,12,21,20,18,15,16	94.6	89.8	94.6
7	22,12,21,20,18,15,16,14	94.6	88.9	94.6
8	22,12,21,20,18,15,16,14,11	94.7	88.5	94.6
9	22,12,21,20,18,15,16,14,11,17	94.7	88.2	94.6
10	22,12,21,20,18,15,16,14,11,17,2	94.7	88.0	94.6
11	22,12,21,20,18,15,16,14,11,17,2,13	95.1	87.8	94.6
12	22,12,21,20,18,15,16,14,11,17,2,13,8	95.1	87.9	94.6
13	22,12,21,20,18,15,16,14,11,17,2,13,8,3	95.1	87.8	94.6
14	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19	95.1	87.8	94.6
15	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5	95.1	87.8	94.6
16	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1	95.1	87.8	94.6
17	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1,6	95.2	87.8	94.6
18	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1,6,4	95.2	87.8	94.6
19	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1,6,4,10	95.2	87.8	94.6
20	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1,6,4,10,9	95.2	87.8	94.6
21	22,12,21,20,18,15,16,14,11,17,2,13,8,3,19,5,1,6,4,10,9,7	95.2	87.8	94.6

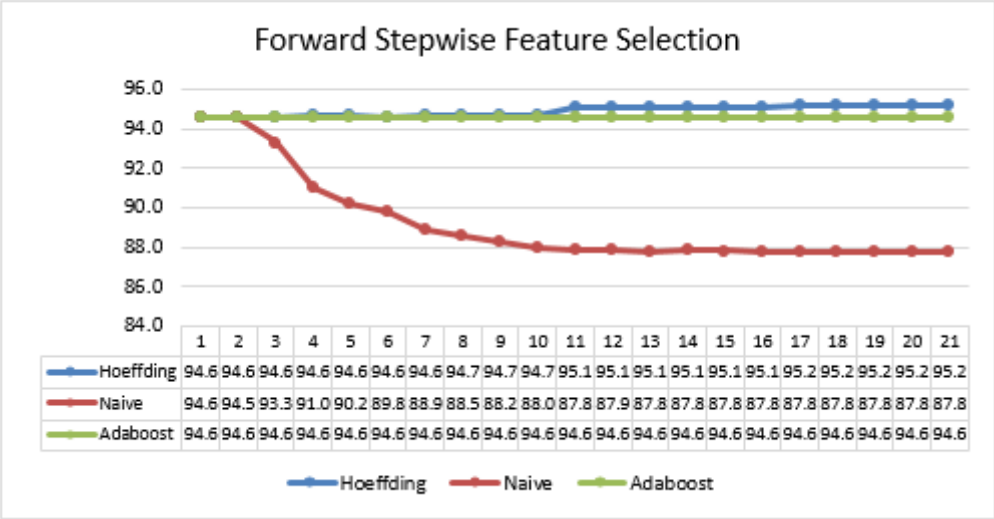
Figure 4.3 shows the graph visualising the PCC of the three classifiers produced by the forward stepwise feature selection method. Refer to table 4.2 for the names of the attributes corresponding to the indices. One important observation is that,

(a) AdaboostM1 (Decision Stump) used only “number of years in a grade” for its predictions and ignored all other attributes subsequently added into the feature space; since the proportion who repeat a grade is high and the likelihood of their being made to repeat it a second time is very low, this essentially means that AdaboostM1 is unable to detect those who will fail a grade for the first time

(b) NaiveBayes used each attribute added to the “number of years in a grade” but with reduced classification performance and lastly,

(c) HoeffdingTree increased its performance with every new attributes added to its feature space

Figure 4.3 shows the plotted results of the forward stepwise feature selection exploratory study.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Hoeffding	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.7	94.7	94.7	95.1	95.1	95.1	95.1	95.1	95.1	95.2	95.2	95.2	95.2	95.2
Naive	94.6	94.5	93.3	91.0	90.2	89.8	88.9	88.5	88.2	88.0	87.8	87.9	87.8	87.8	87.8	87.8	87.8	87.8	87.8	87.8	87.8
Adaboost	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6	94.6

Figure 4.3: Forward Stepwise Feature Selection Results

It can be observed that the eleven attributes provide a better classification accuracy using HoeffdingTree classifier. The eleven attributes are as follows – in the order of priority:

- CY_GradeYears
- CY_PQuality_Term1
- CY_LPDecision_Term1
- PY_PQuality_Term4
- PY_LPDecision_Term2
- PY_LPDecision_Term3
- PY_LPDecision_Term1
- LPhase
- PY_LPDecision_Term4
- CY_LER
- CY_ProgressedStatus

The main disadvantage of Sequential Forward Stepwise (SFS) is the fact that it is unable to remove features that become obsolete after the addition of other features. In view of the latter, the collinearity of the eleven attributes was further assessed. With reference to figure 3.8, there is a high collinearity between “promotion quality” and “promotion decision”. In addition, as observed again in figure 3.8, the promotion quality has better predictive power than the promotion decision. Therefore, the promotion decision becomes redundant and can be removed without losing the prediction accuracy of the model. The latter suggest that, CY_GradeYears, CY_PQuality_Term1, PY_PQuality_Term4, LPhase, CY_LER and

CY_ProgressedStatus remain. In the second study, these six constituted the feature set **(Set B)** for further experimentation.

4.3 Feature Sets for the Experiment

In view of the discussion in section 4.2, two feature sets were created. The first feature set (**Set A**) excludes the seven attributes with low predictive power factor produced in the first exploratory study. The second feature set **(Set B**) consists of the six attributes with high predictive power as suggested by the second exploratory study. Table 4.4 shows the final feature sets for the experiment. According to Kumar et al [36], the only test for accuracy and validity of the feature subset is through the accuracy, sensitivity, and reliability of the classification model. In view of that, the two feature sets will be explored in more detail in the next section to determine the best one to build the final model. Chapter 6 will present the results of the experiment on the two feature sets.

Table 4.4: Feature sets for the experiment

Feature Set A (excludes 7 Attributes)	Feature Set B (Only 6 Attributes)
Number of Features = 14 + 1 (class) CY_LER CY_SLAbsentee_Rate LGender LPhase CY_GradeYears CY_ProgressedStatus PY_LPDecision_Term1 PY_LPDecision_Term2 PY_LPDecision_Term3 PY_LPDecision_Term4 PY_PQuality_Term4 CY_LAbsenteeRate_Term1 CY_LPDecision_Term1 CY_PQuality_Term1 CY_LPDecision_Term4(*Predictive Class*)	**Number of Features = 6 + 1 (class)** CY_LER LPhase CY_GradeYears CY_ProgressedStatus PY_PQuality_Term4 CY_PQuality_Term1 CY_LPDecision_Term4 (*Predictive Class*)

4.4 Summary

This chapter explored different methods to extract best feature sets for the study. The filter based feature selection methods; Correlation, Info-Gain and Symmetric Uncertainty were used and all returned seven attributes (***CY_GPI, SQuintile, SDistrict, LHomeLanguage, LRace, CY_SEAbsentee_Rate and LCitizenship)*** with low predictive power. The feature set excluding the seven attributes was created and named feature **Set A**. The second exploratory study used forward stepwise with

Hoeffding Tree, Naïve Bayes and Adaboost (Decision Stump) as the induction classifiers. HoeffdingTree provided a better PCC with eleven attributes while NaïveBayes performance decreased and Adaboost (Decision Stump) remained unchanged respectively. The eleven attributes with high predictive power were further subjected to collinearity assessment and this resulted in further removal of the redundant attributes; and remained with six attributes (***CY_LER, LPhase, CY_GradeYears, CY_ProgressedStatus, PY_PQuality_Term4 and CY_PQuality_Term1***) that were used to compose the second feature set named **Set B**. The two feature sets formed the basis of this research and further experiments were conducted to assess their predictive power and generalisation ability.

Chapter 5: Experiment Design and Execution

5.1 Introduction

This chapter provides a discussion on how the experiment was designed and executed as well as the challenges met during the experiment.

5.2 Organization of Experimental Data

5.2.1 Determination of the Sample Size

The data available for the research had imbalance class as class *(P)* is 63% more than *class (NP).* According to Longadge et al [65], imbalance class data pose a challenge for many algorithms as they focus more on the classification of the major class while ignoring minority class. Longadge et al [65] suggests that this imbalance class problem can be resolved either through algorithmic approach, data pre-processing or feature selection approach

This study took an approach of data pre-processing to address possible challenges around the imbalance data. First, an arbitrary sample of 20% (119694 out of 598479) was extracted from the discretised dataset produced in chapter 3 using WEKA filter *(weka.filters.supervised.instance.resample)* with a bias of 1.0 to enable extraction of equal number of instances of each class in the sample. The dataset was named **Training Set** and consisted of 59847 instances of class (P) and 59847 instances of class (NP) respectively.

It is important to note that the sample size must be selected appropriately to avoid model overfitting and underfitting which might subsequently affect the accuracy of the final classification model. In view of that, the concept of learning curves was applied to the Training_Set to determine the sample size more appropriate to build the model.

The filtered classifier using J48 as induction algorithm was used to create the learning curves with varying percentage of data from feature **Set A** and **Set B** respectively. J48 decision tree has been used successfully by different researchers as a base classifier in both feature selection and solving classification problems [20, 24-28, 30]. In addition, J48 has been empirically proven to generate well behaving learning curves [66]. Figure 5.1 and 5.2 shows the learning curves where PCC and RMSE were plotted against number of instances.

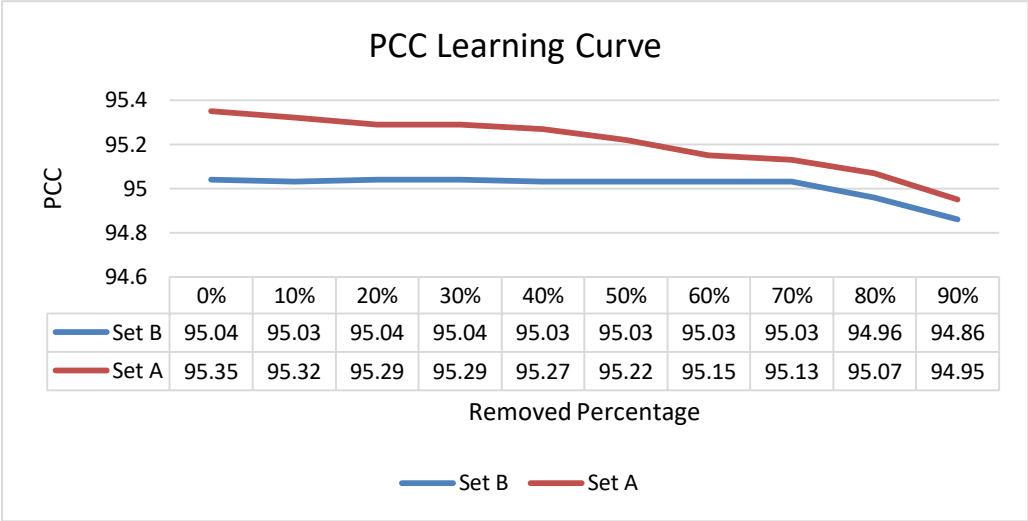


Figure 5.1 Learning Curve: PCC

The PCC in figure 5.1 does not change significantly between 10% to 100% usage of the data. However, there is a noticeable undesirable change in PCC in both sets of data when less than 30% of data is used.

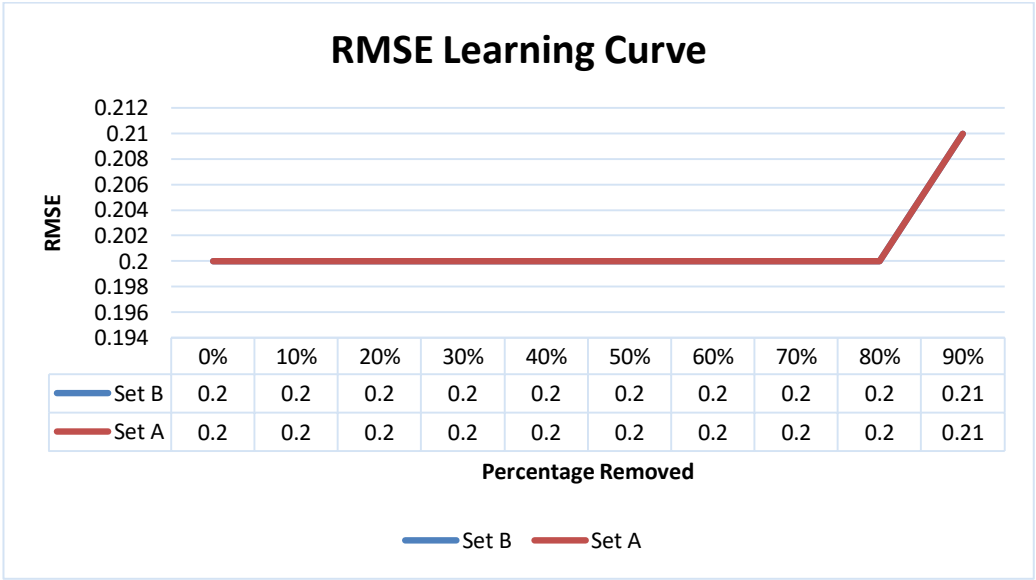


Figure 5.2 Learning Curve: RMSE (*blue line is underneath the red line***)**

The RMSE is a measure of the differences between values predicted by a model against the observed values. In principle, a small RMSE shows a better model. It can be observed from figure 5.2 that less than 20% of the sample size decreases the reliability of the model. In addition, the RMSE plot in figure 5.2 shows that 20% of the data and above produces same RMSE. Based on the above observations, *30%*

(35908) of the sample size from Training_Set will be sufficient for building the models. This implies that the *70% (83787)* will be used for testing the model during training. Table 5.1 summarises the split

Table 5.1: Dataset for developing and performing initial test on the classifiers

Label	# Records	#NP	#P	Purpose
Training Set	**119694**	**59847**	**59847**	
30%	35908	17954	17954	Building the model
70%	83786	41893	41893	Initial Testing of the model (**Test_Set1**)

5.2.2 Additional Datasets for the Study

A separate 15% (*71817 records out of the remaining 478783 records*) was extracted using a filter (*weka.filters.supervised.instance.resample)* with a bias of 1.0 and *"invert selection"* parameter active to extract balanced class data different to the Training_Set in section 5.2.1. The data was labelled **"Test_Set2"** and used for testing the final model.

Finally, the entire 2015/2016 dataset with 598479 records was used to validate the performance of the final model and was labelled **Test_Set3**. It is safe to indicate that this data includes the "Training_Set" and "Test_Set2" which the model already knows. However, this step was also important to be able to observe how the model will behave with the unbalanced data which will be the reality during the implementation of the model.

Lastly, 2016/2017 data was sourced and labelled Test_Set4. The data includes 1251795 learner records of 2016/2017. Table 5.2 summarises the baseline data used for the experiment.

Table 5.2: Baseline data for the experiment

Label	Purpose	Year	# Records	#NP	#P
Category: Training and Evaluation Aim : To build and test the preliminary models which will be compared with the intention to select the best one desired for the study					
Training_Set	Training (30%)	2015/2016	35908	17954	17954
	Testing (70%)	2015/2016	83786	41893	41893
	Total		**119694**	**59847**	**59847**
Category: Test Data sets Aim: To test the final selected model for accuracy, reliability and generalization capability					
Test_Set2	Testing	2015/2016	71816	35908	35908
Test_Set3	Testing	2015/2016	598479	110165	488314
Test_Set4	Testing	2016/2017	1251795	320737	931058

The three classifiers were run on a meta filtered classifier which is a class that runs a classifier on data that has been passed through a filter. Like the classifier, the structure of the filter is based exclusively on the training data and test instances will be processed by the filter without changing their structure. Table 5.3 show filters defined to process the same data using the two different feature sets (**Set A and B**) required for the study.

Table 5.3: Filters to split baseline data into two experimental data setups

Filter	Feature Set
"weka.filters.unsupervised.attribute.Remove -R 1,5,6,7,9,10,17\" The filters remove the following attributes out of the processing space CY_GPI, SQuintile, SDistrict, LHomeLanguage, LRace, CY_SEAbsentee_Rate and LCitizenship	Set A
"weka.filters.unsupervised.attribute.Remove -V -R 2,11,12,13,18,21,22\" The filter only includes the following attributes into processing space: CY_LER, LPhase, CY_GradeYears, CY_ProgressedStatus, PY_PQuality_Term4, CY_PQuality_Term1, CY_LPDecision_Term4 (*Predictive Class*)	Set B

In summary, the data has been organised into two experimental setups, A and B. Experiment A used 14 attributes (*Feature **Set A***) and experiment B used 6 attributes (*Feature **Set B***). The two experiments will be compared to decide on the best classifier and feature set for the study.

5.3 Experiment Execution

Step 1: Evaluate Classifier Performance with Feature Set A (Exp A)

- The WEKA Experimenter was used to design the experiment using "**Training_Set**" data. The three classifiers (NaïveBayes, HoeffdingTree, and AdaboostM1 (Decision Stump) were configured to exclude the attributes *CY_GPI, SQuintile, SDistrict, LCitizenship, LHomeLanguage, CY_SEAbsentee_Rate and LRace* using a WEKA filtered classifier for feature **Set A**. Refer to table 5.3 for more information on the arbitrary filter used
- The experimenter was configured to train on 30% and test with 70% of data from the Training_Set
- The experimenter was set to repeat the experiment ten times and save the results for further analysis
- The knowledge flow was designed and configured accordingly using the similar parameters used in the experimenter. This exercise was done to produce the ROC

and AUC data and related graphs to assess the performance of the model on feature **Set A**. The results of the knowledge flow were also saved

Step 2: Evaluate Classifier Performance with Feature Set B (Exp B)

- The same experimental procedure in step 1 was repeated now with the feature **Set B** filter that only includes CY_LER, LPhase, CY_GradeYears, CY_ProgressedStatus, PY_PQuality_Term4, CY_PQuality_Term1, CY_LPDecision_Term4 (*predictive class*) from the Training_Set Both the experimenter and knowledge flow were run and the results were saved for further analysis.

Step 3: Compare Experiment A and B

- The results of Experiment A and B were analyzed and compared using different model evaluation techniques discussed in more detail in chapter 6.
- The outcome of step 3 was a decision in terms of which classifier and feature set are best for the study.

Step 4: Develop the Final Model

- The best classifier and attribute set found in step 3 was used to develop the final model for the study
- The model was built on 30% data and initially tested with the 70% remaining data from the Training_Set
- The initial testing results for the final model were saved for further analysis

Step 5: Test the Final Model

- The final model produced in step 4 was further tested on additional datasets (Test_Set2, Test_Set3 and Test_Set4) described in section 5.3.4.
- The results were saved for further analysis.

5.4 Experiment Challenges

A range of issues was experienced during the experiment execution. The topmost challenge was how WEKA handled memory during processing, from attribute selection through to model creation (classification). WEKA requires all the data to be loaded in-memory for its processing. With the amount of data exposed to us and the limited data processing capabilities the computer used for the research provided, particularly the available physical memory (4 GB), on several occasions, WEKA ran out of processing memory and crashed the application. The author did everything

possible to avail more memory to the application using different techniques. For example:

- Increasing the priority of the application from normal to real time.
- Closing user background processes that consume some part of the available physical memory.
- Increasing the heap-size of the applications that use Java Visual Machine (JVM) like WEKA.

Another observation made was that working with big data is time-consuming. WEKA struggles to close the threads after the process completes and does not provide the performance results of the classifiers. At times, the processes ceased intermittently and stopped processing. Despite all this, the WEKA tool provides a range of functions that are helpful in cleaning, transforming and mining underlying patterns in the data. The experiment was successfully concluded despite all the challenges.

5.5 Summary

Chapter 5 provided an overview of the research environment. The chapter also provided an overview of how the data has been organized as well as how the experiment was executed. The results of the experiment will be discussed in more detail in chapter 6.

Chapter 6: Results and Findings

6.1 Introduction

This chapter discusses the results of the experiment conducted in chapter 5. HoeffdingTree, NaïveBayes, and AdaboostM1 (Decision Stump)) were trained on the same data and the results will be compared to select the best classifier and attribute set to base the final model. Lastly, the research questions will be revisited and an assessment made to see if they have been addressed.

6.2 Guidelines for Interpreting the Classifier Performance

The experiment was repeated ten times on the same data for each classifier and the average was used to derive the confusion matrix illustrated in figure 6.1.

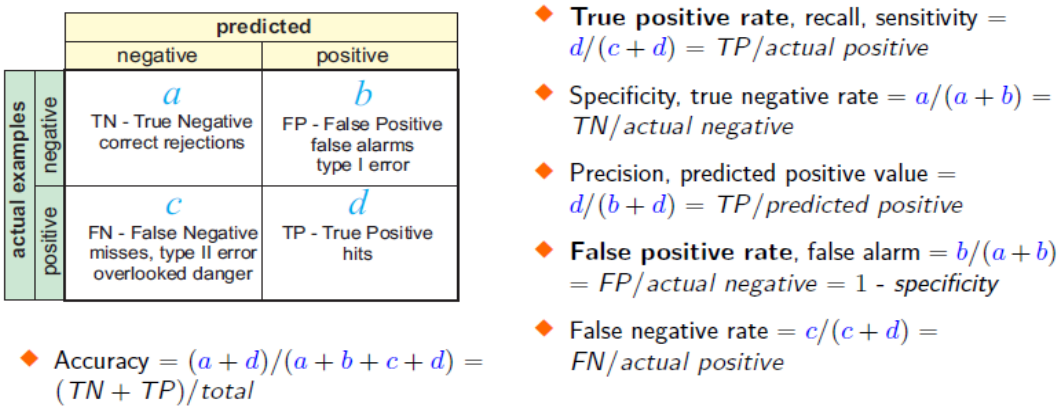


Figure 6.1: Confusion Matrix [48]

The confusion matrix provides an initial measure of accuracy and reliability of the models [48]. According to Hlaváč [48], classification accuracy, classifier sensitivity, specificity, and precision are among other classifier performance indicators that can be calculated from the confusion matrix. However, Hlaváč [48] further states that these scalar quantities do not provide enough information to assess the performance of the classifier and its reliability and sustenance after the implementation. In view of the latter, Hlaváč [48] suggested, the scalar quantities must be supported by visualizing True Positive (TP) -y axis- against False Positive (FP) –x axis- to obtain a curve called the Receiver Operating Characteristics (ROC) which will assist one to assess misclassification costs and class ratios as well as the conditions under which the classifiers outperform one another. Thus, the ROC can be used to assess the trade-off between sensitivity and specificity of the classifier or an ability of the classifier to distinguish between the classes

Figure 6.2 below is taken from [57] where this is used to guide the interpretation of the ROC curve, stating *"a perfect classifier will score in the top left-hand corner were False Positive Rate (FPR) =0, True Positive Rate (TPR) =1 A worst case classifier will score in the bottom right-hand corner where FPR=1, TPR=0. A random classifier would be expected to score somewhere along the positive diagonal (TPR=FPR) since the model will throw up positive and negative examples at the same rate" [57].*

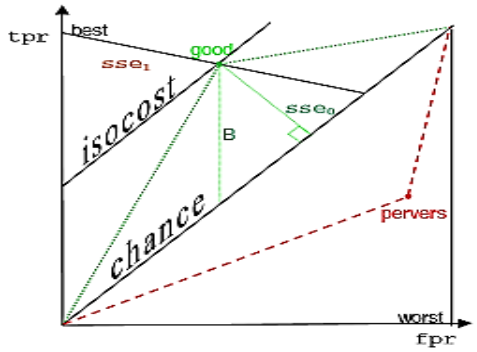


Figure 6.2: ROC Interpreter guide [57]

The main diagonal represents chance, with parallel isocost lines representing equal cost-performance. Points above the diagonal represent performance better than chance, those below worse than chance.

In addition, it will be in the nature of the LDoE learners' performance dataset that we will always find imbalance distributions of classes (P > NP). Eyras [49] and Saito et al [58] state that in the event of the imbalanced class distribution dataset, Precision Recall (or PR) Curves are very useful and can add another dimension of quality and provide a more reliable estimation of accuracy of a classifier. Like the ROC curves, the optimal area under the curve (or AUC) is equal to one.

Another important metric that can be generated from the confusion matrix is a Matthews's correlation coefficient (MCC). This is a measure of the quality of binary classifications and more useful, particularly when dealing with imbalance class data, than the PCC.

$$MCC = \frac{\mathrm{TP\ x\ TN} - \mathrm{FP\ x\ FN}}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

The MCC can take any value between -1 and 1; where -1 = perfect prediction; 0 = not better than random prediction and 1= perfect prediction [74]

In view of the discussion above, the following metrics will be used to conduct comparative performance evaluation of the models:

- **PCC** (*Percentage of Correct Classification*)
- **Type II errors** (*Overlooked Danger: pupil who will fail classified as passing*)
- **AUC-ROC** (*Area Under Curve –Receiver Operating Characteristics*)
- **AUC-PR** (*Area Under Curve –Precision Recall)*
- **MCC** (*Matthew Correlation Coefficient*)
- **ROC curve** (*Receiver Operating Characteristics Curve)*
- **PR curve** (*Precision Recall Curve*)

6.3 Guidelines for Interpreting Feature Selection Performance

Expanding from section 4.3 in chapter 4, [36] states that: *"The overall classification rule, including the feature selection algorithm, should be tested on a model that the experimenter believes is somewhat representative of the population of the data. The test should use the number of potential features, the feature set size, the sample size, and the error estimator for the experiment."*

This [36] further states that, *"without such a performance characterization, one lacks the epistemological ground on which to draw conclusions from the analysis, since the scientific meaning of the analysis depends on the mathematical properties of methods used in the analysis"*

As guided by the literature [36]; and further supported by [47], the author used the performance and generalisation ability of the classifier as a means to test the best attribute subset between **Set A** and **Set B** produced in chapter 4 .

The high dimensionality of the feature sets always creates complexity in the model and results in overfitting. It is therefore critical that few attributes are used as long as they help achieve the intention of classification goal. This is supported by a commonly quoted maxim or principle called Occam's razor when it says "it is vain to do by more what can be done by fewer". In the context of this study, this simply suggests that the model complexity should also form part of the evaluation.

6.4 Comparative Analysis of the Experiments

6.4.1 Preliminary Analysis

Table 6.1 illustrates the model evaluation metrics generated after executing steps 1 to 3 of the experimental procedures outlined in chapter 5. It can be observed that HoeffdingTree has performed very well in terms of the PCC, MCC, AUC-ROC and

AUC-PR in both of the feature sets, However, in terms of dealing with, the type II error, it was outperformed by AdaBoostM1 (Decision Stump).

Table 6.1: Model Evaluation Scalar Metrics

	Set B (6 Attributes)			Set A (15 Attributes)		
	HoeffdingTree	NaïveBayes	AdaBoostM1	HoeffdingTree	NaïveBayes	AdaBoostM1
Percent Correct	94.8	90.92	94.56	95.12	87.89	94.56
False Negative Rate	0.02	0.12	0.01	0.02	0.17	0.01
Matthews Correlation	0.90	0.82	0.89	0.90	0.76	0.89
Area Under ROC	0.98	0.98	0.98	0.98	0.96	0.98
Area Under PRC	0.98	0.98	0.97	0.98	0.96	0.97

For ease of analysis, figure 6.3 provides visualisation of the model evaluation metrics in table 6.1. The table is showing best results in green; 2nd best in yellow.

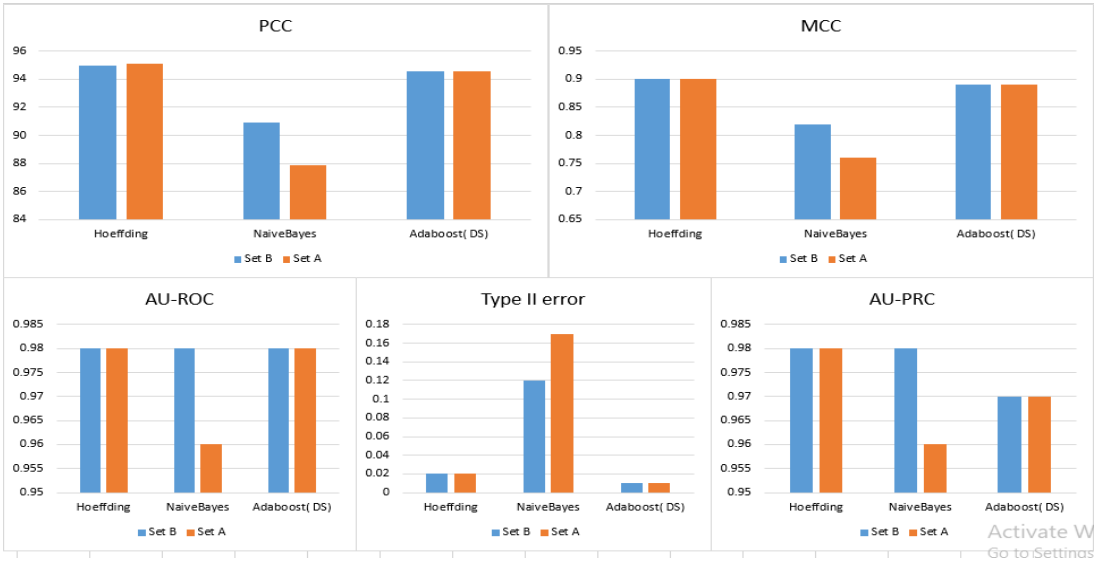


Figure 6.3: Visualisation of Model Evaluation Metrics

In terms of the model performance from the metrics in figure 6.3, HoeffdingTree has performed better, followed by AdaBoostM1 (Decision Stump) and lastly, NaïveBayes in both of the feature sets. It's important to note that HoeffdingTree has shown slight improvement in PCC with feature **Set A**, but still maintained the same quality and generalisation ability in respect of the MCC, AUC-ROC and AUC-PR. AdaBoostM1 (Decision Stump) came in the second position in both of feature sets and has maintained all its quality measures. Lastly, NaïveBayes is in the last position in both of the feature sets, but has suffered with feature **Set A** (15 attributes) where its prediction accuracy, model quality and generalisation ability decreased

6.4.2 Area under ROC curve

According to [48,49], it is important to confirm the AUC-ROC scalar quantities before choosing a classifier due to the trade-off between False Negative (FN) and False Positive (FP).The AUC can sometimes be high as a result of a single class, contributing more to the classification cost and results in an unreliable choice of predictive model.In view of the latter, the ROC curve of both classes and feature sets of all the classifiers were plotted and are shown in figure 6.4 [57].

The following principles were used to make deductions from the ROC curves:

- If ROC curves for different classifiers do not intersect, it implies that one classifier dominates the other
- If ROC curves for different classifiers intersect, one classifier is better for some cost ratios, and other is better for other cost ratios

It can be observed that with feature **Set A**, both HoeffdingTree and AdaboostM1 (Decision Stump) are approaching the top left-hand corner where FPR=0, TPR=1 while NaïveBayes classifier is slightly nearer the bottom right-hand corner where FPR=1, TPR=0 in both classes. These suggest that both HoeffdingTree and AdaboostM1(Decision Stump) in their order of their performance are better than NaïveBayes in terms of their generalisation ability and model stability. It can further be observed from the ROC curve that class (P) converges to TPR=1 faster than class (NP). These observations imply that the model is slightly more skilled in predicting class (P) than class (NP).

With regard to feature **Set B** , it can be observed that NaïveBayes is now converging faster to TPR=1 as compared to feature **Set A**. This implies that it has gained more prediction power and generalisation ability as suggested by the scalar metrics in section 6.4.1. In addition, HoeffdingTree slightly changed between the two feature sets while AdaboostM1 remained unchanged.

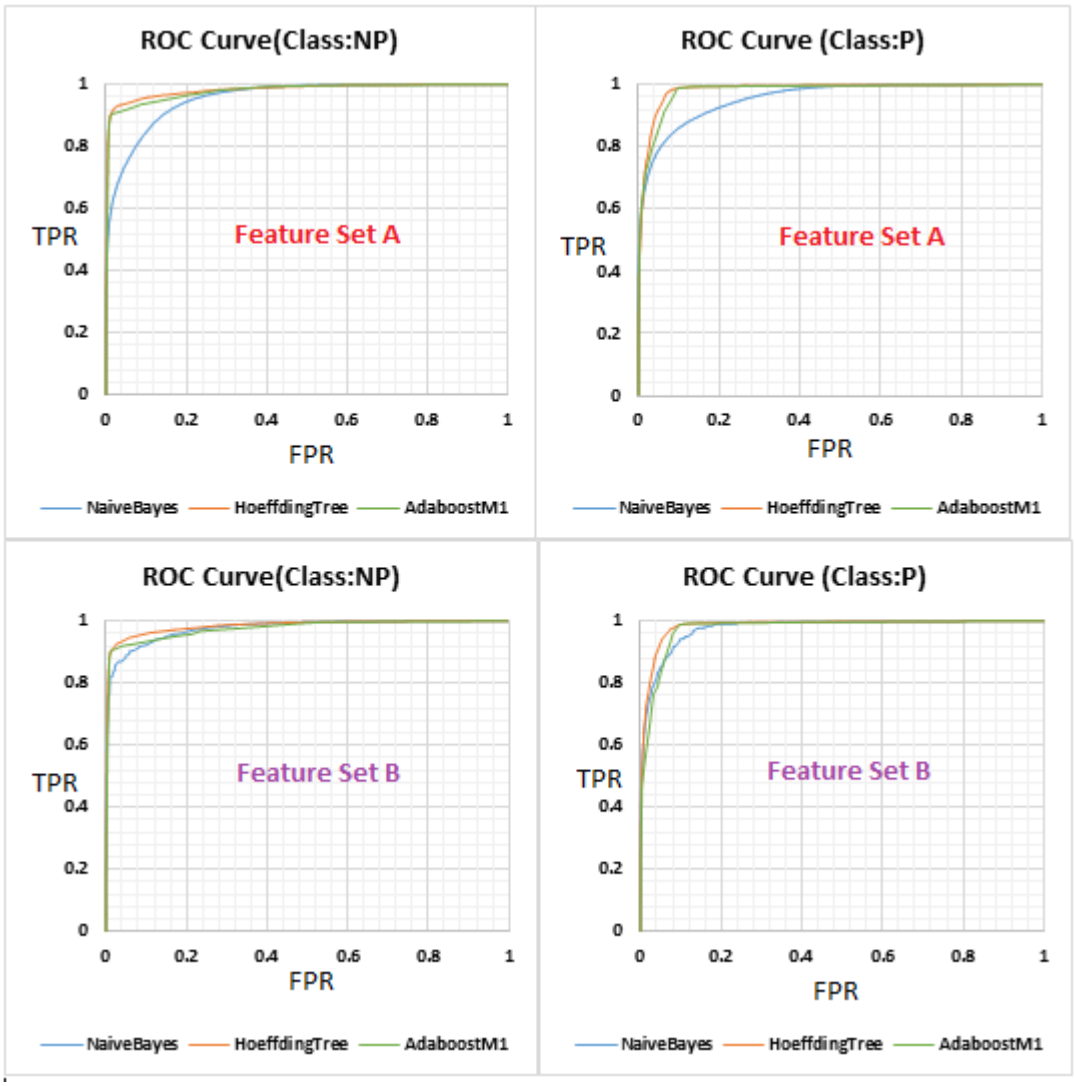


Figure 6.4: Comparative view of Experiment A and B - ROC curve

All the classifiers showed some skill on the classification problem and are therefore better than the random classifier, since they are all above the chance line where FPR=TPR. The observation made on the ROC supports and confirms the analysis and discussions in section 6.2.1.

HoeffdingTree and Decision Stump have a built-in protection against irrelevant features and some intelligence to compute the best feature subset (*embedded feature selection capability*) [61]. This explains the reason why AdaboostM1 (Decision Stump) and HoeffdingTree‘s ROC looks similar when subjected to the two feature sets.

NaïveBayes is vulnerable to correlated features and can benefit more from feature selection [61]. This explains the reduction in classification performance of NaïveBayes when subjected to feature **Set A**.

6.4.3 Area under PR curve

Saito and Rehmsmeier [58] compared the reliability of ROC and PR curves in evaluating the performance of the model on the imbalanced datasets; and found that PR curve plots provide a more reliable performance measures than the ROC. To support the finding, the researchers [58] were quoted as follows:

"We show here that the visual interpretability of ROC plots in the context of imbalanced datasets can be deceptive with respect to conclusions about the reliability of classification performance, owing to an intuitive but wrong interpretation of specificity. PRC plots, on the other hand, can provide the viewer with an accurate prediction of future classification performance due to the fact that they evaluate the fraction of true positives among positive predictions"

The finding is confirmed by [59]:

"If you are doing (conflict) research with sparse binary data and are interested in whatever reason in model fit, (1) your models don't do as well as ROC might lead one to believe, and (2) consider precision-recall curves as an addition or alternative"

In view of the above, it is important to further confirm the performance of the classifier through PR plots. Figures 6.5 shows the PR curve for both class and feature sets (**Set A and B**) respectively.

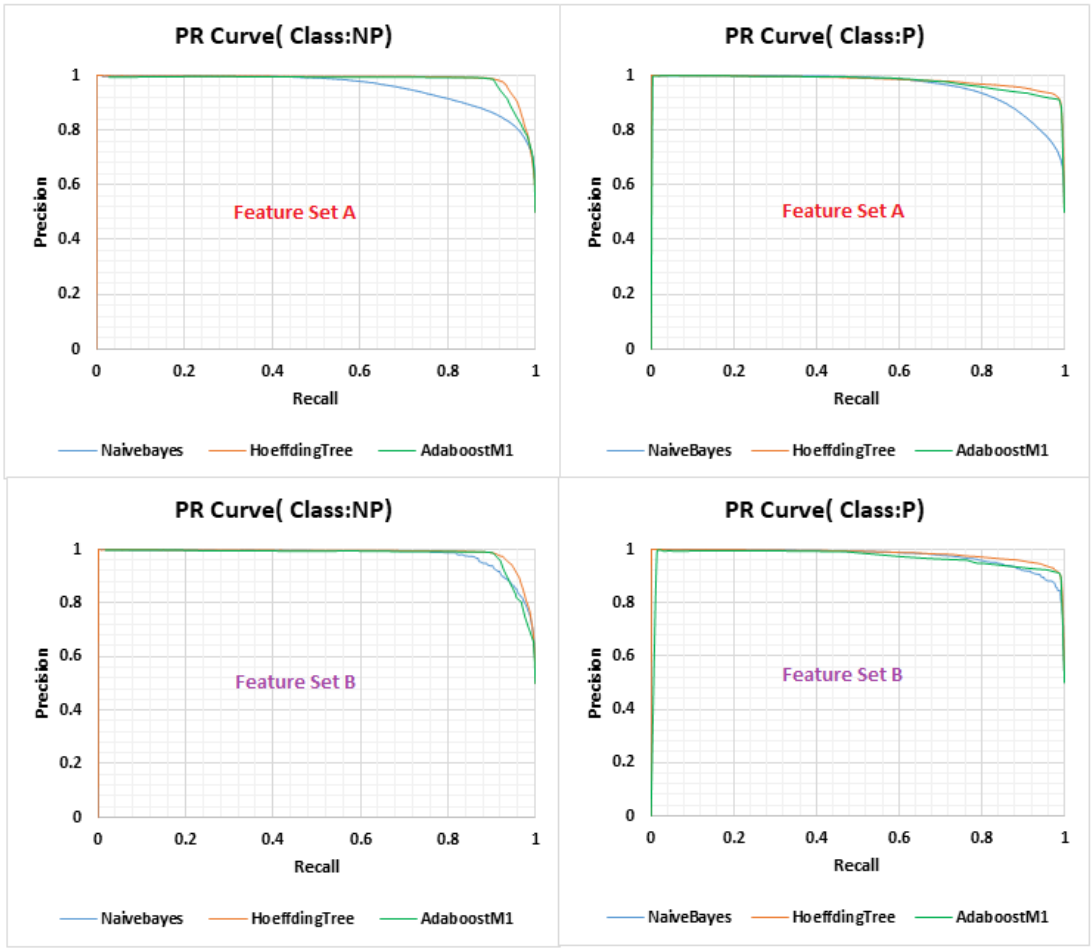


Figure 6.5: Comparative view of Experiment A and B PR curve

According to Ekelund [59], *"The perfect test will have a PRC that passes through the upper right corner (corresponding to 100 % precision and 100 % recall). Generally, you can say that the closer a PRC is to the upper right corner, the better the test is."*

In view of the above, one can deduce from figure 6.5 that HoeffdingTree outperformed all the classifiers, followed by AdaboostM1 (Decision Stump) and lastly, NaïveBayes with feature **Set A**. In addition, HoeffdingTree still outperformed all the classifiers in in feature **Set B** while both AdaboostM1 (Decision Stump) and NaïveBayes competes for the cost ratios through their intersections and makes it very difficult to decide the best classifier particularly using the PR plot. However, evaluation metrics in section 6.2.1 favours AdaboostM1 and therefore, NaïveBayes will be on the last position.

6.5 Research Findings

Section 6.4 provided analysis and motivation for the author to be able to select a preferred classifier and attribute subset for the research. The first research question was to assess the contribution of attributes towards learners' performance and progression:

Research Question 1: What are the main factors that affect learners' performance within the Limpopo Education Environment?

Citing from the literature by Guyon [19] and Ramaswami et al [31] as discussed in chapter 3 of this report, the goal of feature selection is to improve learning efficiency, increasing predictive accuracy and reducing the complexity of the prediction model that will lead to the curse of dimensionality.

From the analysis, feature **Set A** (15 Attributes) provides a better classification accuracy and model stability for both HoeffdingTree and AdaboostM1 (Decision Stump) compared to feature **Set B** (6 Attributes) which only favoured NaïveBayes and slightly affected the PCC of HoeffdingTree while AdaboostM1 remained unchanged.

The findings of the analysis can be supported by a study conducted by Post et al [61] to assess the impact of the feature selection on different classifiers which, among others included NaïveBayes, AdaBoost (Decision Stump), and HoeffdingTree.

The study made observations that,

- most of the decision trees (HoeffdingTree included & Decision Stump) have a built-in protection against irrelevant features and some intelligence to compute the best feature subset (*embedded feature selection capability*). This explains the reason why AdaboostM1 remained unchanged even when more features are added as well as why HoeffdingTree benefited.
- NaïveBayes is vulnerable to correlated features and can benefit more from feature selection. This explains the reduction in classification performance of NaïveBayes when subjected to feature **Set A**.

The findings provide a framework to explain the observations on the ROC and PR curves in figure 6.4 and 6.5 respectively. The second research aim was to find the best classifier between HoeffdingTree, NaïveBayes, and AdaboostM1 (Decision Stump):

Research Question 2: Which classifier between NaïveBayes, HoeffdingTree, and AdaboostM1 (Decision Stump) will provide a better prediction accuracy of learner progression within the Limpopo Education Environment?

Section 6.4 analysed the performance of NaïveBayes, HoeffdingTree, and AdaboostM1 (Decision Stump). Based on the guidelines in section 6.2, we conclude that **HoeffdingTree** outperformed AdaboostM1 (Decision Stump) and NaïveBayes in all of the feature sets. In addition, the feature set choice had little impact on classification accuracy and generalisation ability of the HoeffdingTree and none on AdaboostM1 (Decision Stump).

It is safe to indicate that both the HoeffdingTree and Decision Stump are decision trees. However the Decision Stump has been boosted using AdaboostM1 to improve its performance.

Expanding from the literature on decision trees in section 2.8.1 in chapter 2 of this report, Shahiria et al [30] have drawn attention to the fact that, decision trees are the most common classification technique to predict learners' performance due to their simplicity and reduced effort in data preparation. However, in other work [20, 24-28, 30] similar to the study, decision trees were outperformed. The results are different in this research, the two decision tree algorithms differed minimally in their performance, and were better than Naïve Bayes. This is so because more work was done to assess the relevancy of the attributes under feature selection and the feature space was also reduced as the outcome of the exercise.

Petri [32] reveals that decision trees tend to perform well if few relevant attributes are used. He further states that their greedy characteristic leads to over-sensitivity to irrelevant attributes and data noise during training. The latter provides more reasons to choose feature **Set B** as the most suitable for the final model and the reason are solely based on

- The limitations of the decision trees based on the literature [32] as explained above.
- Occam's razor. In the context of this study, this simply suggests and as declared in section 6.3 that the model complexity will also form part of evaluation in which the simpler version will be the selected choice.

- The fifteen attributes(feature **Set A**) will require more processing power in terms of hardware and software, more time for data collection and preparation; with little benefits in terms of prediction accuracy and generalisation ability.

In summarising the findings, HoeffdingTree and feature **Set B** (with six attributes) are the best for the study and were used to develop the final model. Returning to research question 1, the following attributes affect learners' performance in Limpopo

Table 6.2: Attributes with predictive power to learners' performance

Contributing Attributes	Contributing Attributes	Non-Contributing Attributes
Feature Set B: CY_LER LPhase CY_GradeYears CY_ProgressedStatus PY_PQuality_Term4 CY_PQuality_Term1	**Redundant** CY_Grade PY_LPDecision_Term1 PY_LPDecision_Term2 PY_LPDecision_Term3 PY_LPDecision_Term4 CY_LPDecision_Term1 **Notes** CY_Grade is collinear to LPhase All PY_LPdecision collinear with PY_PQuality_Term4 CY_LPDecision_Term1 collinear with CY_PQuality_Term1	**Excluded by Exploratory Study 1** SQuintile SDistrict LCitizenship LHomeLanguage LRace CY_LAbsenteeRate_Term1 CY_GPI **Excluded by Exploratory Study 2** CY_SLAbsentee_Rate CY_SEAbsentee_Rate LGender

6.6 Development and Validation of Final Model

The final model was developed using Hoeffding and feature **Set B** with six attributes from the balanced class data used to evaluate the performance of the classifiers. The model was built using 30% of the Training_Set and was initially tested with 70% of the Training_Set (Test_1). The final model was further tested with the 3 hold-out sets Test_Set2 (balanced, 2015/16), Test_Set3 (all 2015/16 data) and Test_Set4 (all 2016/17 data). Table 6.3 and 6.4 show the performance results of the model.

Table 6.3: Testing results with all data (1)

	Test_Set1	Test_Set2	Test_Set3	Test_Set4
Total Instances	83786	71816	598479	1251795
Correctly Classified	79426	68296	578749	1006723
Incorrectly Classified	4360	3520	19730	245072
PCC	94.8 %	95.1 %	96.7%	80.1 %
Kappa Statistics	0.8959	0.902	0.8915	0.4919
Mean Absolute Error	0.0726	0.0775	0.0782	0.22
RMSE	0.2026	0.2015	0.1847	0.4035

Table 6.4: Testing results with all data (2)

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Test_Set1	0.975	0.079	0.925	0.975	0.949	0.897	0.984	0.981	P
	0.921	0.025	0.974	0.921	0.947	0.897	0.984	0.984	NP
Test_Set2	0.976	0.074	0.929	0.976	0.952	0.903	0.984	0.982	P
	0.926	0.024	0.975	0.926	0.950	0.903	0.984	0.985	NP
Test_Set3	0.976	0.075	0.983	0.976	0.980	0.892	0.985	0.996	P
	0.925	0.024	0.899	0.925	0.912	0.892	0.985	0.954	NP
Test_Set4	0.863	0.365	0.873	0.863	0.868	0.492	0.866	0.945	P
	0.635	0.137	0.614	0.635	0.624	0.492	0.866	0.665	NP

The final model performed exceptionally well with testing set Test_1, Test_2 and Test_3 which contained 2015/16 data, i.e. data from the same year as that on which the model was trained. With the full set of 2016/17 data, the model has a reduction in terms of its ability to predict the smaller class “NP”. However, the model still shows some knowledge in resolving the “NP” class. This deduction can be supported by the Precision Recall Curve (PRC) = 0.67 for class “NP” which is better than the random guess. It must be noted that the ROC Area and MCC for Test_4 are the same for both classes. These are quality metrics that indicate the robustness and prediction accuracy of the model. The actual and predicted failure rate of all the schools participated in the study for both 2016 and 2017 was calculated.Table 6.5 summarises actual and predicted failure rate at the provincial level:

Table 6.5: Failure Rate

2016 Failure Rate		2017 Failure Rate	
Actual	**Predicted**	**Actual**	**Predicted**
20%	21%	24%	25%

6.7 Summary

Chapter 6 analyzed the results of the experiment described in chapter 5. All the research questions were answered. HoeffdingTree with feature **Set B** (6 attributes) were identified as the most reliable for the study. The final model was developed using HoeffdingTree and feature **Set B** and performed well in all its predictions except for predicting failures in a previously-unseen year of data – in which case, while not performing as well, was better than a chance predictor, with an AUC-ROC of 0.866 and a PRC of 0.665 respectively.

Chapter 7: Conclusion

7.1 Mini-thesis summary

The study produced a model to predict learner progression prospects to the next grade at the end of the academic year. To achieve that, the study investigated the factors that affect learner progression within the Limpopo educational context; and as well, compared the predictive ability of the three data mining techniques in the context of the research against different sets of attributes. Such a model can assist in better allocation of resources to the schools in the Limpopo province of South Africa, and also help to identify early, those schools and learners needing additional interventions to improve their chances of success in learning and teaching. The study responded to the following research questions:

7.1.1 Research Question 1

What are the main factors that affect learners' performance within the Limpopo Education Environment?

The author identified 22 features - expected to have high predictive power- from among over 400 relations in the LDoE database based on the author's domain proficiency and literature on pedagogy. The study then evaluated these features and showed that:

- Ten of these attributes could be omitted from classifier training with negligible effect on classifier accuracy,
- Eleven of the features has predictive ability.

The ten attributes with negligible effect were Learners' race, citizenship, district, home Language, the school's absentee rate for both the Learners and Educators, individual's Learner absentee rate, schools' Quintile, individual Learner gender and school's GPI. From this we can deduce that GPI (the extent to which a school has a male: female learner balance) or even the individual gender orientation of a learner does not impact upon failure rate. Even more interestingly, the national practice of distinguishing schools as belonging to one of five quintiles (that indicate their resourcing level), is not a useful one in Limpopo, even though quintiles 1 to 4 are all common in the province.

Of the 11 features with predictive ability, six of the attributes were used to build the final model and the other five were ignored because of their redundancy. The six

features used in the final model were; Learner enrolment in a school, number of years a learner is in a grade, a phase a learner is enrolling, current year learner progression status, previous year fourth term promotion quality as well as current year first term promotion quality.

The previous year Term 1 to Term 4 promotion decisions were found to be collinear with Term 4 promotion quality while current year Term 1 promotion decision is collinear with current year Term 1 promotion quality and therefore, all the five promotion decisions were excluded from the study in favour of promotion quality.

7.1.2 Research Question 2

Which classifier between NaïveBayes, HoeffdingTree, and AdaboostM1 (Decision Stump) will provide a better prediction accuracy of learner progression within the Limpopo Education Environment?

Of the three classifiers evaluated, decision trees were found to perform better than NaïveBayes, with Hoeffding Tree slightly more accurate than AdaboostM1 (Decision Stump). One important observation is that,

(a) AdaboostM1 (Decision Stump) used only “number of years in a grade” for its predictions and ignored all other attributes subsequently added into the feature space; since the proportion who repeat a grade is high and the likelihood of their being made to repeat it a second time is very low, this essentially means that AdaboostM1 is unable to detect those who will fail a grade for the first time

(b) NaiveBayes used each attribute added to the “number of years in a grade” but with reduced classification performance and lastly,

(c) HoeffdingTree increased its performance with every new attribute added into its feature space.

The performance of HoeffdingTree (*Decision Tree*) can be traced back to the literature. Petri [32] asserts that decision trees can easily adapt to various data structures due to their non-parametric processing capability; and they tend to perform well if few relevant attributes are used. Hence it performed better with the six attributes from **SET B** and this also provides a perspective in the reliability of the feature selection process conducted in the study.

7.1.3 Effects of Changing Learning Environment

It is evident in the research that 2016/17 test data has resulted in the reduced classification accuracy as well as reduced predictive ability of the smaller class. This was as a result of the changed learning cultures, particularly in respect of the efforts both learners and educators took to enable learning and teaching respectively. It must be understood that the model intends to identify learners at risk of failing and not to decide on that. Thus, if more efforts are taken to assist learners at risk, they should ultimately pass and this doesn't suggest that the model was wrong in making initial predictions.

7.2 Limitations of the Study

The initial approach of this research was to use the Unified Data Mining (UDMT) to guide the study. The theory has a convincing process and is widely supported [13-16]. However, there was a range of challenges the researcher met in terms of the implementation of the theory using WEKA. The theory emphasizes automation of data mining process from the ingress point of Unified Data Mining Process (UDMP), where clustering, classification and visualisation tools will work together to select appropriate features and algorithms based on the goal of the data mining and the type of data fed into the UDMP process. Solanki H [15] found that all the tools investigated (WEKA, Tanagra, and KNIME) lacked:

- Automatic selection of the appropriate algorithm for clustering, classification, and visualisation.
- The correct application of algorithm as a function. Thus, automatically taking the results of the previous algorithm results to serve the next algorithm in the composite functions phase is lacking.

In view of these challenges, the research study approach had to be changed to adopt the CRISP-DM process [10].

Quality of data was another limiting factor, observed during the preparation of data. Not all of the learner data attributes were available per school and this could have been caused by learner migration, learner dropouts and capturing of data at the school-level. At face value, the selected algorithm showed some level of flexibility, but the results can be different if the problem around completeness of data becomes severe. In addition, there was a lot of data cleansing that had to be done to ensure consistency in the data attributes and improve the modelling efficiency. If the model

is put into production without the proper cleaning of the data, the integrity of the predictions can be questionable.

7.3 Prospects of Future Work

Although the supervised resampling with no replacement was used to balance the class distribution in the data during training, classifier performance evaluation and building of the final classifier, the model seems to be best in predicting class P (pass) as compared to NP (fail). Even though the design and processing technique used by the selected classifier (HoeffdingTree) showed some degree of intelligence and knowledge in addressing the problem as well as the ability to generalize with minimal issues [50], the problem is still not completely resolved. These open doors for researchers to dig deeper into the data mining process used and explore different strategies that could be used to root out the problem completely. It is also important that attributes not currently collected as part of the school data, such as school curriculum coverage and educator subject competency profiles should be included in future to enable improved predictions.

Many separate utilities were written in the course of this work in order to consolidate, clean, and normalise data for input to the model; a user-centred tool to guide the user through this in a semi-automated process would facilitate deploying the model in practice.

References

[1] S. T. Hijazi and R. Naqvi, “Factors Affecting Students’ Performance: A Case of Private Colleges,” *Bangladesh e-Journal of Sociology,* vol. 3, no. 1, pp. 1-9, January 2006.

[2] S. Siyepu, “The zone of proximal development in the learning of mathematics,” *South African Journal of Education,* vol. 33, no. 2, p. 2, 2013.

[3] E. Erkan, “Identifying At-Risk Students Using Machine Learning Techniques: A Case Study With IS 100,” *International Journal of Machine Learning and Computing,* vol. 2, no. 4, pp. 476-479, 2012.

[4] Kotsiantis, S.B; Pierrakeas, C. J; Zaharakis, I.D; Pintelas, P.E, “Efficiency of Machine Learning Techniques In Predicting Students’ Performance In Distance Learning Systems,” University of Patras,Department of Mathematics, Greece, 2003.

[5] E. Alpaydin, “Introduction to Machine Learning,” London, Massachusetts Institute of Technology, 2010, pp. 2-3.

[6] J. Nilsson, “Introduction to Machine Learning,” Stanford, Robotics Laboratory,Stanford University, 1998, pp. 5-6.

[7] C. Donalek, “Supervised and Unsupervised Learning,” April 2011. [Online]. Available: http://www.astro.caltech.edu. [Accessed 28 February 2017].

[8] T. Ayodele, “Types of Machine Learning Algorithms,” Intech Open Science, University of Portsmouth,United Kingdom, 2010.

[9] K. Thearling, “An Introduction to Data Mining: Discovering Hidden Value in your Data Warehouse,” 2012. [Online]. Available: http://www.thearling.com.

[10] J. Jackson, “Data Mining: A Conceptual Overview,” *Communications of the Associations for Information Systems,* vol. 8, pp. 267-296, 2002.

[11] F. Usama and R. Uthurusamy, “Data Mining to Knowledge Discovery in Databases,” *Communication of the ACM,* vol. 39, no. 11, pp. 24-26, 1996.

[12] A. Azevedo and M. Santos, “KDD, SEMMA AND CRISP-DM: A parallel overview,” in *ResearchGate, IADIS European Conference Data Mining*, 2008.

[13] D. Khan, N. Mohamudally and D. Babajee, “Unified Theoretical Framework for Data Mining,” *Procedia Computer Science,* vol. 17, pp. 104-113, 2013.

[14] D. Khan, N. Mohamudally and D. Babajee, “Towards the Formulation of a Unified Data Mining Theory, Implemented by Means of Multiagent Systems (MASs),” Intech Open Science, 2012.

[15] H. Solanki, “Comparative Study of Data Mining Tools and Analysis with Unified Data Mining Theory,” *International Journal of Computer Applications,* vol. 75, no. 16, pp. 23-28, 2013.

[16] S. Rasheeduddin, “The theoretical framework of data mining and its techniques,” *International Journal of Social Science & Interdisciplinary Research,* vol. 2, no. 1, pp. 81-85, January 2013.

[17] M. Pechenizkiy, K. Koedinger, M. Feng, “Educational Data Mining: Home,” International Educational Data Mining Society , July 2011. [Online]. Available: http://www.educationaldatamining.org.

[18] R. Gautam and D. Pahuja, “A Review on Educational Data Mining,” *International Journal of Science and Research (IJSR),* vol. 3, no. 11, pp. 2929-2932, 2014.

[19] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *Journal of Machine Learning Research ,* vol. 3, pp. 1157-1182, 2003.

[20] Z. Ibrahim and D. Rusli, “Predicting Students Academic Performance: Comparing Artificial Neural Network, Decision Tree and Linear Regression,” ResearchGate,21st Annual SAS Malaysia Forum, Malaysia,Kuala Lumpur , 2007.

[21] D. Angeline, “Association rule generation for student performance analysis using apriori algorithm,” *Computer Science Engineering & its Applications ,* vol. 1, no. 1, pp. 12-16, March 2013.

[22] M. Quadri and N. Kalyankar, “Drop Out Feature Of Student Data For Academic Performance Using Decision Tree Techniques,” *Global Journal of Computer Science and Technology ,* vol. 10, no. 2, pp. 2-5, April 2010.

[23] M. Tair and A. El-Halees, “Mining Educational Data To Improve Students Performance: A Case Study,” *International Journal of Information and Communication Technology Research,* vol. 2, no. 2, pp. 140-145, February 2012.

[24] S. Jishan, R. Rashu, N. Haque and R. Rahman, “Improving Accuracy Of Students Final Grade Prediction Model Using Optimal Equal Width Binning And Synthetic Minority Over-Sampling Technique,” *Springer, Decision Analytics,* vol. 2, no. 1, pp. 1-25, 2015.

[25] V. Ramesh, P. Parkavi and K. Ramar, “Predicting Student Performance: A Statistical And Data Mining Approach,” *International Journal of Computer Applications,* vol. 63, no. 8, pp. 35-39, February 2013.

[26] Y. Zhang, S. Oussena, T. Clark and H. Kim, “Use Data Mining To Improve Student Retention In Higher Education – A Case Study,” in *12th International Conference on Enterprise Information Systems*, Madeira, Portugal,, 2010.

[27] A. Amjad, “Educational Data Mining & Students’ Performance Prediction,” *International Journal of Advanced Computer Science and Application,* vol. 7, no. 5, pp. 212-219, 2016.

[28] S. Sembiring, M. Zarlis, D. Hartama and S. E. Ramliana, “Prediction of Student Academic Performance by an Application Of Data Mining Techniques,” in *International Conference on Management and Artificial Intelligence*, Bali, Indonesia , 2011.

[29] F. Mohammed-Ali, B. Emhemed and Z. Suliman, “Predicting Performance Of Classification Algorithms,” *International Journal of Computer Engineering and Technology (IJCET),* vol. 6, no. 2, pp. 19-28, 2015.

[30] A. Shahiria, W. Husaina and N. Rashida, “A Review on Predicting Student’s Performance Using Data Mining Techniques,” *Procedia Computer Science,* vol. 72, pp. 414-422, 2015.

[31] M. Ramaswami and R. Bhaskaran, “A Study on Feature Selection Techniques in Educational Data Mining,” *Journal of Computing,* vol. 1, no. 1, pp. 7-11, December 2009.

[32] C. Petri, “Decision Trees,” 2010. [Online]. Available: http://www.cs.ubbcluj.ro. [Accessed 12 June 2017].

[33] T. Dietterich, “Ensemble Methods in Machine Learning,” 2000. [Online]. Available: http://web.engr.oregonstate.edu. [Accessed 14 June 2017].

[34] Z. Zhi-Hua, Ensemble Methods Foundations and Algorithms, London, New York: Chapman and hall, 2012.

[35] M. Sewel, “Ensemble Learning,” August 2008. [Online]. Available: https://pdfs.semanticscholar.org. [Accessed 15 June 2017].

[36] M. Kumar, S. Chidambaram and K. Srinivasagan, “Optimization Technique for Feature Selection and Classification Using Support Vector Machine,” in *International Conference on Computational Intelligence in Data Mining,volume 1*, Springer, New Delhi, 2015.

[37] R. Joazeiro and P. Inventado, “Educational Data Mining and Learning Analytics,” 2014. [Online]. Available: https://www.semanticscholar.org. [Accessed 22 June 2017].

[38] E. Keogh, “NaïveBayes Classifiers,” 2006. [Online]. Available: http://www.cs.ucr.edu. [Accessed 22 June 2017].

[39] R. Nisbet, J. Elder and G. Miner, Handbook of Statistical Analysis and Data Mining Applications, Canada: Elsevier Science Direct, 2009.

[40] V. Stapel, Z. Zheng and N. Pinkwart, “An Ensemble Method to Predict Student Performance in an Online Math Learning Environment,” in *Proceedings of the 9th International Conference on Educational Data Mining*, Raleigh, NC USA , 2016.

[41] S. Gayathri and S. Shet, “Approach for Predicting Student Performance Using Ensemble Model Method,” *International Journal of Innovative Research in Computer and Communication Engineering ,* vol. 2, no. 5, pp. 161-168, 2014.

[42] Z. Pardos, S. Gowda, R. Baker and N. Heffernan, “Ensembling Predictions of Student Post-Test Scores for an Intelligent Tutoring System,” 2011. [Online]. Available: http://www.upenn.edu. [Accessed 05 July 2017].

[43] E. Amrieh, T. Hamtini and I. Aljarah, “Mining Educational Data to Predict Student's academic Performance using Ensemble Methods,” *International Journal of Database Theory and Application ,* vol. 1, no. 8, pp. 119-136 , 2016.

[44] R. Garner, “WEKA: The Waikato Environment for Knowledge Analysis,” 1995. [Online]. Available: http://www.cs.waikato.ac.nz. [Accessed 10 July 2017].

[45] S. Slater, S. Joksimovic, V. Kovanovic, R. Baker and D. Gasevic, “Tools for educational data mining: a review,” 2012. [Online]. Available: http://www.upenn.edu. [Accessed 13 July 2017].

[46] A. Jović, K. Brkić and N. Bogunović, “A review of feature selection methods with applications,” [Online]. Available: https://pdfs.semanticscholar.org.

[47] E. Dougherty, H. Jianping and S. Chao, “Performance of Feature Selection Methods,” 20 February 2009. [Online]. Available: https://www.ncbi.nlm.nih.gov. [Accessed 27 July 2017].

[48] V. Hlaváč, “Classifier performance evaluation,” [Online]. Available: http://people.ciirc.cvut.cz. [Accessed 30 July 2017].

[49] E. Eyras, “Model Accuracy Measures,” 2017. [Online]. Available: http://comprna.upf.edu.

[50] S. Perveena, M. Shahbaza, A. Guergachib and K. Keshavjee, “Performance Analysis of Data Mining Classification Techniques to Predict Diabetes,” *Procedia Computer Science ,* vol. 82, pp. 115-121, 2016.

[51] R. Schapire, “Explaning Adaboost,” 09 October 2013. [Online]. Available: http://rob.schapire.net. [Accessed 30 07 2017].

[52] P. Domingos and G. Hulten, “Mining high-speed data streams,” 10 August 2000. [Online]. Available: https://homes.cs.washington.edu. [Accessed 30 July 2017].

[53] G. John, R. Kohavi and K. Pfleger, “Irrelevant Features and the Subset Selection Problem,” in *Machine Learning Proceedings 1994,Eleventh International Conference*, New Brunswick, 1998.

[54] K. Laws, “Psychology, replication & beyond,” December 2016. [Online]. Available: https://www.researchgate.net. [Accessed 30 July 2017].

[55] J. Loannidis, “Why Most Published Research Findings Are False,” *PLoS Medicine,* vol. 2, no. 8, pp. 696-701, August 2005.

[56] R. Latif, H. Abbas, S. Latif and A. Masood, “EVFDT: An Enhanced Very Fast Decision Tree Algorithm for Detecting Distributed Denial of Service Attack in Cloud-Assisted Wireless Body Area Network,” 9 August 2015. [Online]. Available: https://www.hindawi.com. [Accessed 30 July 2017].

[57] D. Powers, “Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness & Correlation,” *Journal of Machine Learning Technologies ,* vol. 2, no. 1, pp. 37-63 , 27 February 2011.

[58] T. Saito and M. Rehmsmeier, “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets,” *Plos One,* vol. 10, no. 3, pp. 1-21, 16 January 2015.

[59] A. Berger, “Precision Recall Curves,” 15 April 2016,. [Online]. Available: https://papers.ssrn.com. [Accessed 05 December 2017].

[60] S. Ekelund, “Precision-recall curves – what are they and how are they used?,” April 2017. [Online]. Available: https://acutecaretesting.org. [Accessed 05 December 2017].

[61] M. Post, P. Van der Putten and J. Van Rijn, “Does Feature Selection Improve Classification? A Large Scale Experiment in OpenML,” 09 December 2016. [Online]. Available: http://liacs.leidenuniv.nl. [Accessed 05 December 2017].

[62] A. Yousefpour, R. Ibrahim, H.-N. Abdull-Hamed and S. Hajmohammadi, “Feature Reduction Using Standard Deviation with Different Subsets Selection in Sentiment Analysis,” in *Asian Conference on Intelligent Information and Database Systems* , Switzerland, 2014.

[63] A. Ghasemi and S. Zahediasl, “Normality Tests for Statistical Analysis: A Guide for Non-Statisticians,” *International Journal of Endocrinology and Metabolism ,* vol. 10, no. 2, pp. 486-48, January 2012.

[64] N. Schreiber and M. Jackson, “Multicollinearity: What Is It, Why Should We Care, and How Can It Be Controlled?,” 2017. [Online]. Available: https://analytics.ncsu.edu. [Accessed 14 July 2018].

[65] R. Longadge, S. Dongre and L. Malik, “Class Imbalance Problem in Data Mining: Review,” *International Journal of Computer Science and Network ,* vol. 2, no. 1, pp. 1-6, February 2013.

[66] M. Weiss and A. Battistin, “Generating Well-Behaved Learning Curves: An Empirical Study,” 28 December 2014. [Online]. Available: https://pdfs.semanticscholar.org. [Accessed 14 July 2018].

[67] N. Shong-Chok, “Pearson’s versus spearman’s and Kendall’s correlation coefficients for continuous data,” 26 May 2010. [Online]. Available: http://d-scholarship.pitt.edu. [Accessed 14 July 2017].

[68] J. Ruoming, Y. Breitbart and C. Mouh, “Data discretization unification,” Kent State Universit, Kent, 2007.

[69] S. Marono, A. Betanzos and M. T. Sanroman, “Filter methods for feature selection. A comparative study,” University of A Corun˜a, Department of Computer Science, Spain,Corun˜a, 2007.

[70] K. Mani and P. Kalpana, “A Review on Filter Based Feature Selection,” *International Journal of Innovative Research in Computer and Communication Engineering ,* vol. 4, no. 5, pp. 9146 -9156, May 2016.

[71] R. Porkodi, “Comparison of filter based feature selection algorithms: an overview,” *International Journal of Innovative Research in Technology and Science,* vol. 2, no. 2, pp. 108-113, 2014.

[72] J. Novakovic, P. Strbac and D. Bulatović, “Toward optimal feature selection using ranking methods and classification algorithms,” *Yugoslav Journal of Operations Research,* vol. 21, no. 2, pp. 119-135, 2011.

[73] R. J. Brooks and M. Tobias, “Choosing the Best Model: Level of Detail, Complexity, and Model Performance,” *Elsevier Science (Pergamon),Math Computing and Modelling,* vol. 24, no. 4, pp. 1-14, May 1996.

[74] Y. Liu, J. Cheng, C. Yan, X. Wu and F. Chen, “Research on the Matthews Correlation Coefficients Metrics of Personalized Recommendation Algorithm Evaluation,” *International Journal of Hybrid Information Technology ,* vol. 8, no. 1, pp. 163-172 , 2015.

[75] MathsWorks, “Mathworks Documentation,” Mathworks, 2018. [Online]. Available: https://www.mathworks.com. [Accessed 13 July 2017].

Appendices

Appendix A (Examples of SQL statements for Cleaning, Aggregating and Integrating data)

A.1 Cleaning and Encoding Citizenship Attribute

The SQL statement (labelled step 1) displays all possible values captured under citizenship attribute. The values are then encoded to <1= *Citizen and 2= Immigrant*> using SQL statement (labelled step 2)

```
/****** Script for SelectTopNRows command from SSMS  ******/
/*Step 1 : Displaying list values under citizenship attribute */
select citizenship from research.dbo.Final_Transformed_Data
group by citizenship
/*Step 2 : Cleaning list values under citizenship attribute*/
Update research.dbo.Final_Transformed_Data set [citizenship] =
(Case
    when citizenship = 'SADC' then '2' /* Immigrant */
    when citizenship = 'SA' then '1' /* Citizen */
    when citizenship = 'Immigrant' then '2'
    when citizenship = 'SA Citizen' then '1'
    when citizenship = 'Ghana' then '2'
    when citizenship = 'Congo' then '2'
    when citizenship = 'Pakistan' then '2'
    when citizenship = 'SA Citizenship' then '1'
end)
```

Results | Messages

	citizenship
1	SADC
2	SA
3	Immigrant
4	INDIA
5	SA Citizen
6	ZIMBABWE
7	SA Citizen
8	Ghana
9	CONGO
10	PAKISTAN
11	SA Citizenship

Cleaned list Values

Results | Messages

	citizenship
1	2
2	NULL
3	1

A.2 Cleaning and Encoding Gender Attribute

The SQL statement (labelled step 1) displays all possible values captured under gender attribute. The values are then encoded to <1= *Male, 2= Female and 3= Unspecified*> using SQL statement (labelled step 2)

```
/****** Script for SelectTopNRows command from SSMS  ******/
/*Step 1 : Displaying list values under gender attribute */
select gender from research.dbo.Final_Transformed_Data
group by gender
/*Step 2 : Cleaning list values under gender attribute*/
Update research.dbo.Final_Transformed_Data set [gender] =
(Case
    when gender = 'Male' then '1' /* Male */
    when gender = 'Female' then '2' /* Female */
    when gender = 'Unspecified' then '3'
end)
```

Results | Messages

	gender
1	Male
2	Female
3	Unspecified

Cleaned list values

Results | Messages

	gender
1	2
2	3
3	1

A.3 Cleaning and Encoding Race Attribute

The SQL statement (labelled step 1) displays all possible values captured under race attribute. SQL statement (labelled step) deletes all other data anomalies that the author was unable to map them to any race. The values are then encoded to <1= *African Black, 2= Asian/Indian, 3=Coloured, 4= White and 5 =Other* > using SQL statement (labelled step 2)

```
/****** Script for SelectTopNRows command from SSMS  ******/
/*Step 1 : Displaying list values under race attribute */
select race from research.dbo.Final_Transformed_Data group by race
/*Step 2 : Cleaning list values under race attribute*/
Delete from research.dbo.Final_Transformed_Data  /* Delete all unknown values */
where race in('b6','0104020286084','1107140895089','b2','b4','b5','b1')
Update research.dbo.Final_Transformed_Data set [race] = /* Converting nominal list values to  numeric */
(Case
    when race = 'African/Black' then '1' /* African/Black */
    when race = 'Asian/Indian' then '2' /* Asian/Indian */
    when race = 'Coloured' then '3' /* Coloured */
    when race = 'White' then '4' /* White */
    when race = 'Other' then '5' /* Other */
end)
```

Results | Messages

	race
1	B6
2	0104020286084
3	1107140895089
4	B2
5	Coloured
6	B4
7	1001050490082
8	Asian/Indian
9	B5
10	B1
11	White
12	African/Black
13	Other

Cleaned list values

Results | Messages

	race
1	2
2	3
3	4
4	5
5	1

A.4 Cleaning Promotions Attribute

The SQL statement below checks for different data anomalies or variations of values captured under promotion descriptor. All other promotion descriptors not making sense were deleted and were only left with “P” for promotion and “NP” for not promoted. The process was repeated for different academic terms and years included in the study (2015 and 2016).The information for other years not included in the study were deleted.

```
/****** Script for SelectTopNRows command from SSMS ******/

/* Cleaning Promotion Table

check for variations of promotion descriptors*/
SELECT reportcode, count(reportcode) as PromotionDescriptor
FROM [ProgressionQuality].[dbo].[Promotion] group by reportcode

/*Delete rows with different promotion descriptors than P and NP*/
delete from [ProgressionQuality].[dbo].[Promotion] where not reportcode in(
'P','NP')

/*Remove rows with null values*/
delete from [ProgressionQuality].[dbo].[Promotion] where reportcode is null

/*check for variations of learneraverage outliers*/
SELECT [LearnerAverage], count(LearnerAverage) as OutlierCount
FROM [ProgressionQuality].[dbo].[Promotion] group by LearnerAverage order by
LearnerAverage desc
```

```
/*Delete for learneraverage outliers*/
Delete
FROM [ProgressionQuality].[dbo].[Promotion] where Learneraverage between 101
and 1000

SELECT datayear
FROM [ProgressionQuality].[dbo].[Promotion] group by datayear

Delete
FROM [ProgressionQuality].[dbo].[Promotion] where datayear between 2009 and
2014

SELECT term
FROM [ProgressionQuality].[dbo].[Promotion] group by term

Delete
FROM [ProgressionQuality].[dbo].[Promotion] where term=5
```

A.5 Example of Pivoting Learner Averages

The SQL statement below pivot the learner average pass (promotion *quality as described in the study*). Tables called Ltransform_3 was created and was later integrated into a flat table structure to have a single record for every learner. The process was repeated with promotion decisions

```
SELECT pvt.[EmisCode],pvt.[LearnerID],
 pvt.[1] as LAverage2015_Term4,
 pvt.[2] as LAverage2016_Term2,
 pvt.[3] as LAverage2016_Term3,
 pvt.[4] as LAverage2016_Term4 into LTransform_3
FROM ( SELECT [EmisCode],[LearnerID],[Term],[LearnerAverage]
FROM dbo.Promotion where datayear between 2015 and 2016 ) AS t
PIVOT
 ( avg([LearnerAverage])
 FOR [Term] IN
 ([1],[2],[3],[4])
 ) AS pvt;
```

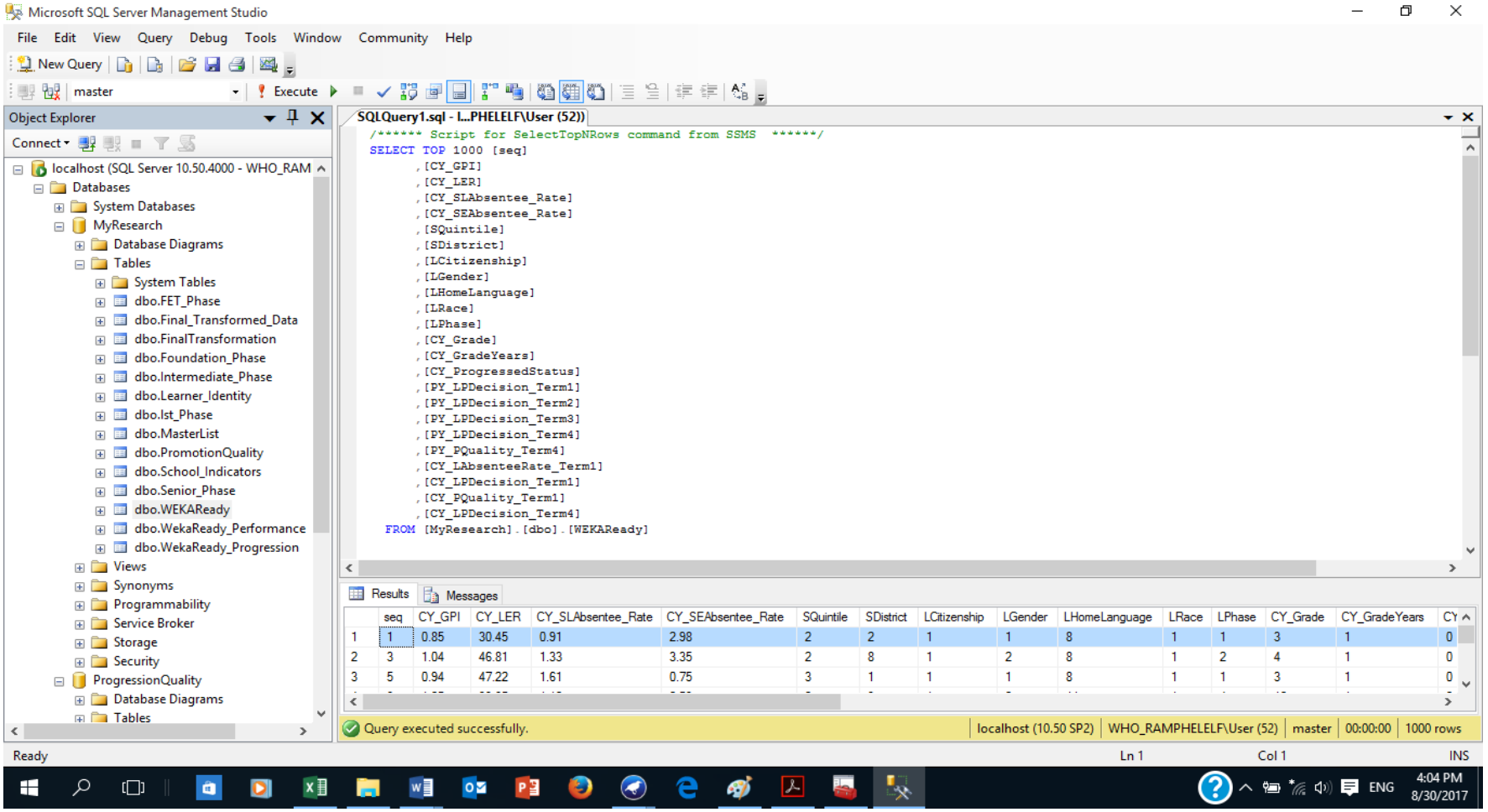
A.6 Integration of Data Tables

The SQL statement below consolidate all individual tables using EMIS number and Learner Accession number as a composite key. This will ensure that every learner is connected to his record accordingly. A table called WEKAFinal was created with about **598479 learner records** and their related information ready for uploading into WEKA

```
SELECT dbo.Learner_Identity.seq, dbo.WEKAReady.CY_GPI, dbo.WEKAReady.CY_LER,
dbo.WEKAReady.CY_SLAbsentee_Rate, dbo.WEKAReady.CY_SEAbsentee_Rate,
 dbo.WEKAReady.SQuintile, dbo.WEKAReady.SDistrict,
dbo.WEKAReady.LCitizenship, dbo.WEKAReady.LGender,
dbo.WEKAReady.LHomeLanguage, dbo.WEKAReady.LRace,
 dbo.WEKAReady.CY_Grade, dbo.WEKAReady.CY_GradeYears,
dbo.WEKAReady.CY_ProgressedStatus, dbo.WEKAReady.PY_LPDecision_Term1,
dbo.WEKAReady.PY_LPDecision_Term2,
 dbo.WEKAReady.PY_LPDecision_Term3, dbo.WEKAReady.PY_LPDecision_Term4,
dbo.WEKAReady.CY_LAbsenteeRate_Term1, dbo.WEKAReady.CY_LPDecision_Term1,
 dbo.WEKAReady.CY_LPDecision_Term4, dbo.lst_Phase.Phase,
dbo.PromotionQuality.LAverage2015_Term4,
dbo.PromotionQuality.LAverage2016_Term1 into WEKAFinal
FROM dbo.PromotionQuality INNER JOIN
 dbo.WEKAReady INNER JOIN
 dbo.Learner_Identity ON dbo.WEKAReady.seq = dbo.Learner_Identity.seq INNER
JOIN
 dbo.lst_Phase ON dbo.WEKAReady.CY_Grade = dbo.lst_Phase.Grade ON
dbo.PromotionQuality.EmisCode = dbo.Learner_Identity.EMIScode AND
 dbo.PromotionQuality.LearnerID = dbo.Learner_Identity.ID
GROUP BY dbo.Learner_Identity.seq, dbo.WEKAReady.seq, dbo.WEKAReady.CY_GPI,
dbo.WEKAReady.CY_LER, dbo.WEKAReady.CY_SLAbsentee_Rate,
dbo.WEKAReady.CY_SEAbsentee_Rate,
 dbo.WEKAReady.SQuintile, dbo.WEKAReady.SDistrict,
dbo.WEKAReady.LCitizenship, dbo.WEKAReady.LGender,
dbo.WEKAReady.LHomeLanguage, dbo.WEKAReady.LRace,
 dbo.WEKAReady.CY_Grade, dbo.WEKAReady.CY_GradeYears,
dbo.WEKAReady.PY_LPDecision_Term1, dbo.WEKAReady.PY_LPDecision_Term2,
dbo.WEKAReady.PY_LPDecision_Term3,
 dbo.WEKAReady.PY_LPDecision_Term4, dbo.WEKAReady.CY_LAbsenteeRate_Term1,
dbo.WEKAReady.CY_LPDecision_Term1, dbo.WEKAReady.CY_LPDecision_Term4,
dbo.lst_Phase.Phase,
 dbo.PromotionQuality.LAverage2015_Term4,
dbo.PromotionQuality.LAverage2016_Term1, dbo.WEKAReady.CY_ProgressedStatus
```

A.7 Final Dataset

This is how the final table looked like after data has been cleaned, aggregated and consolidated into a single table. Some of the data fields were renamed in the process to enable better comprehension and simplicity of the research

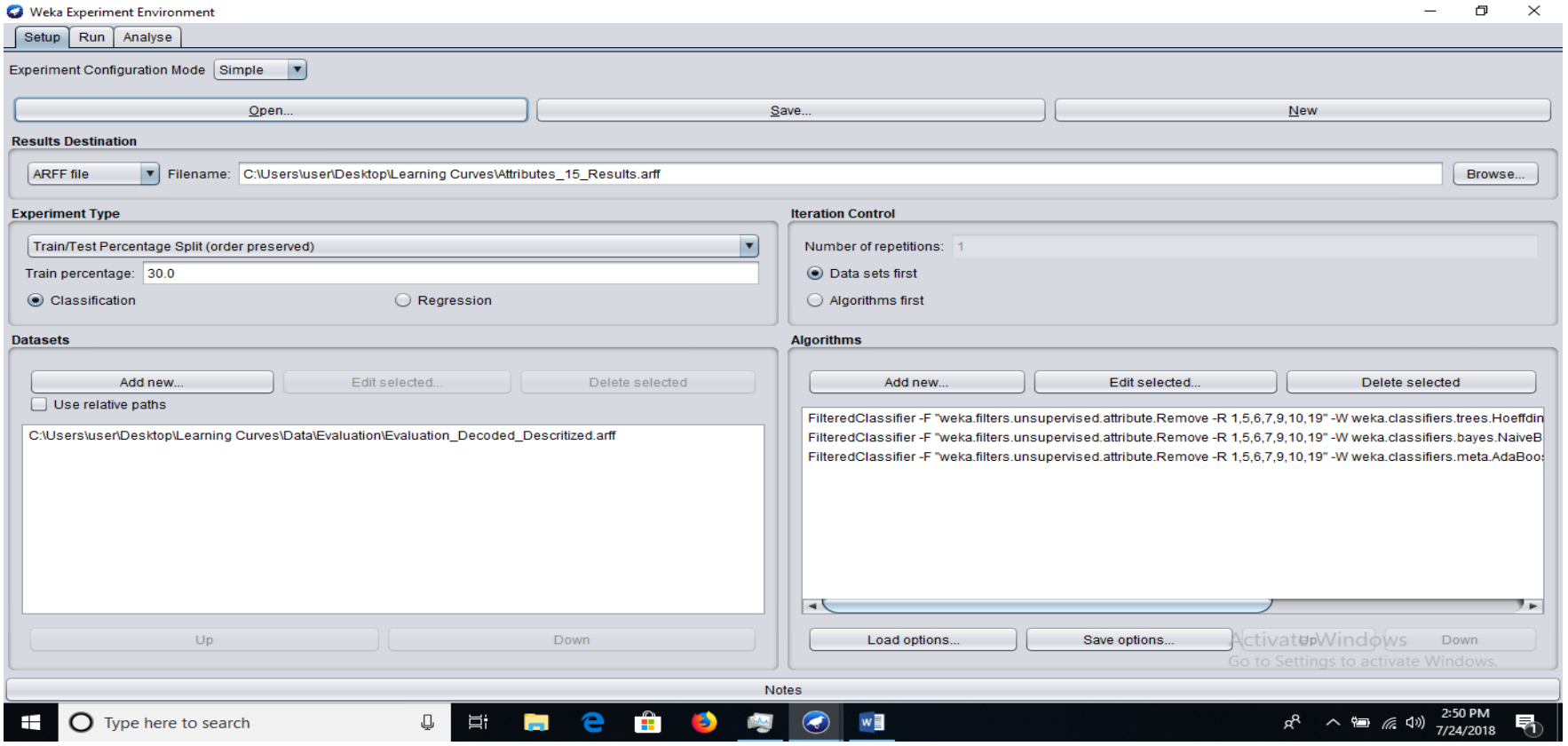


Appendix B (experimental setup A and B)

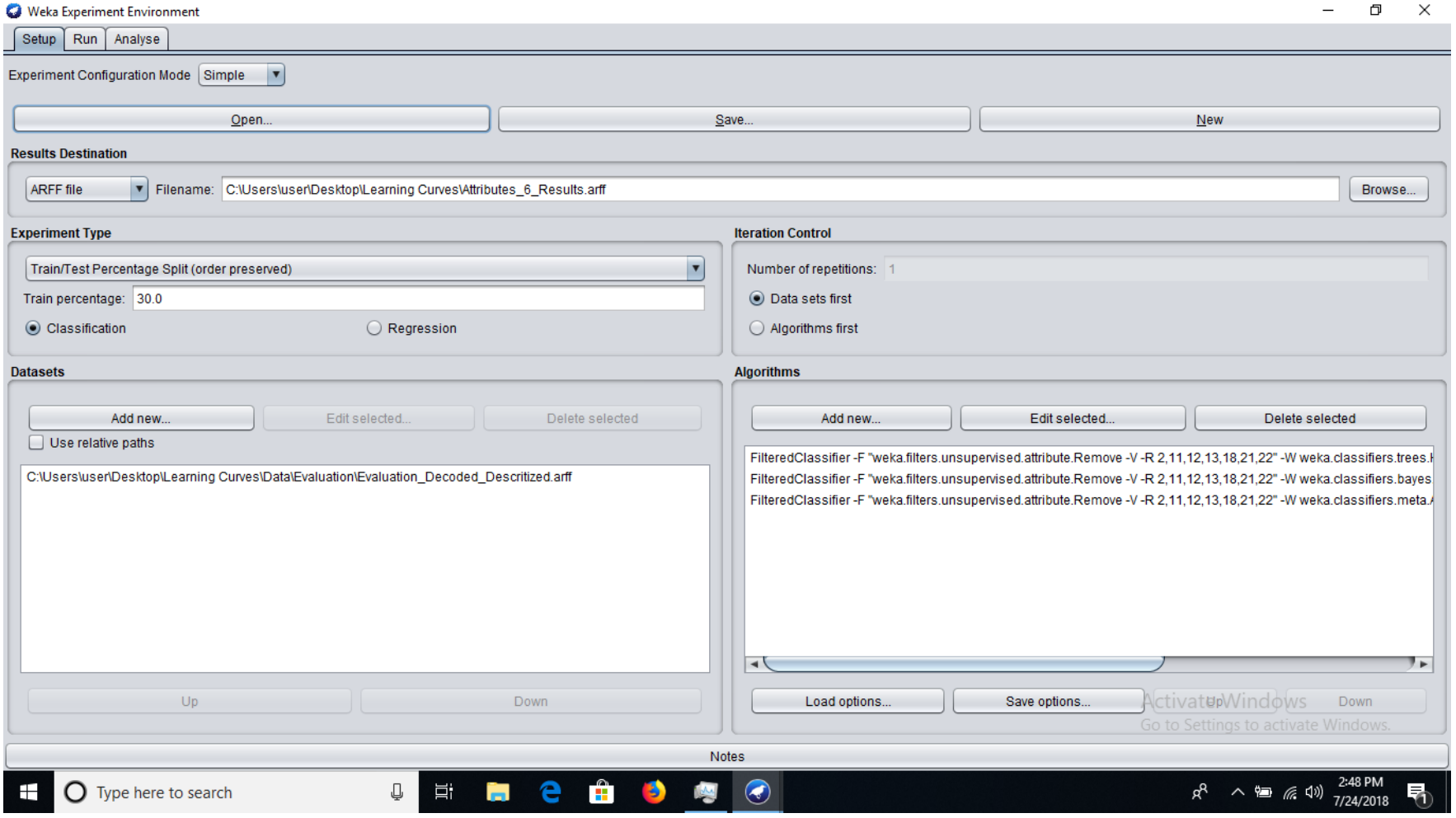
B.1 Example of Experiment A Setup

This is the WEKA environment for designing and executing data mining experiments. The three classifiers (HoeffdingTree, NaïveBayes and AdaBoostM1 (Decision stump) were added and run against the “Training Dataset” in both experiment A and B using different filters. The results of the experiment were then saved. The results can be made available on request

Experimental Setup A (14 Attributes)

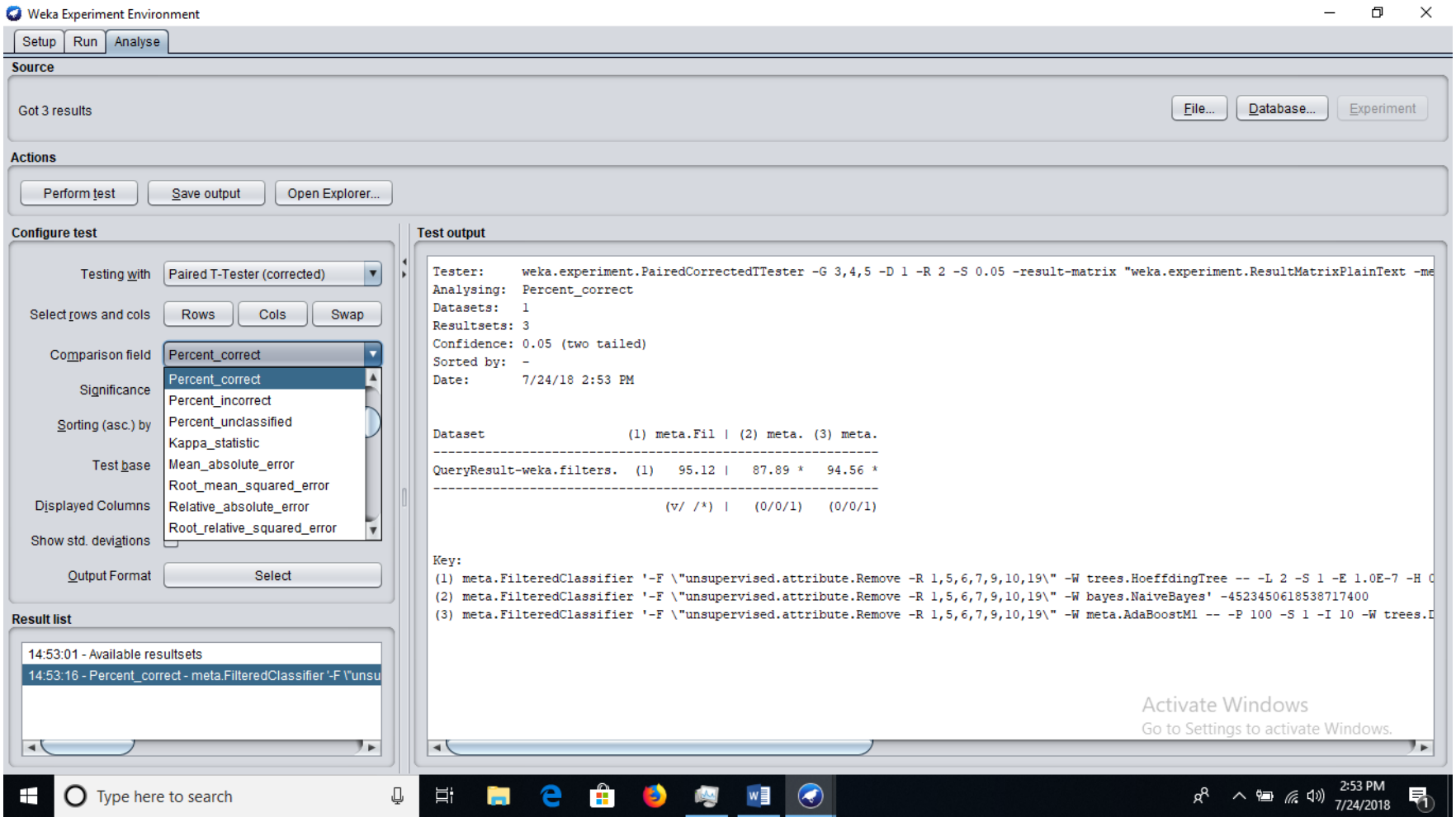


Experiment B setup (6 Attributes)



B.2 Experimenter Analysis Environment

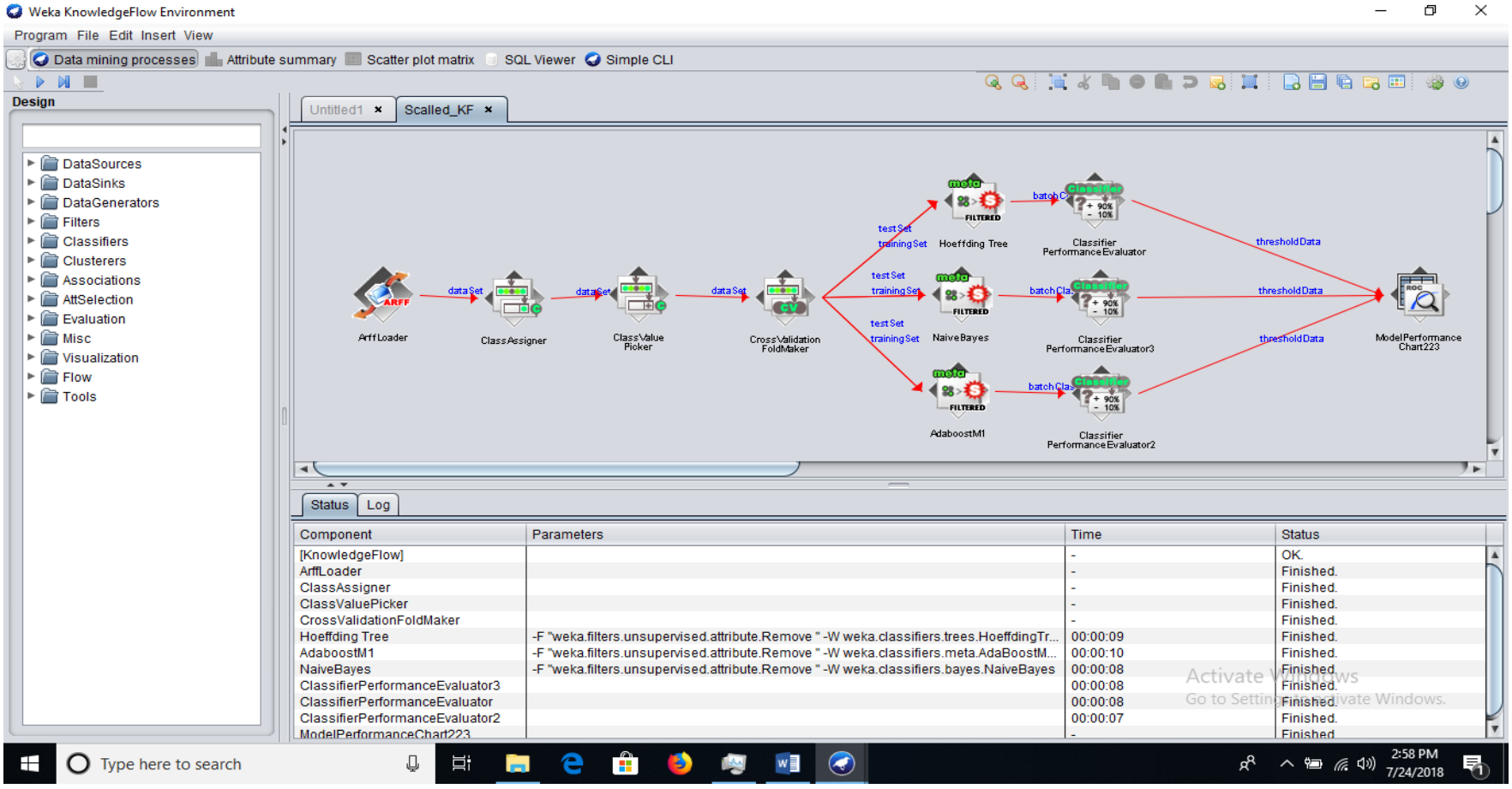
This is the environment where you load the experiment results and performs different analysis to assess the performance of the classifiers against the dataset provided.



Appendix C (knowledge flow)

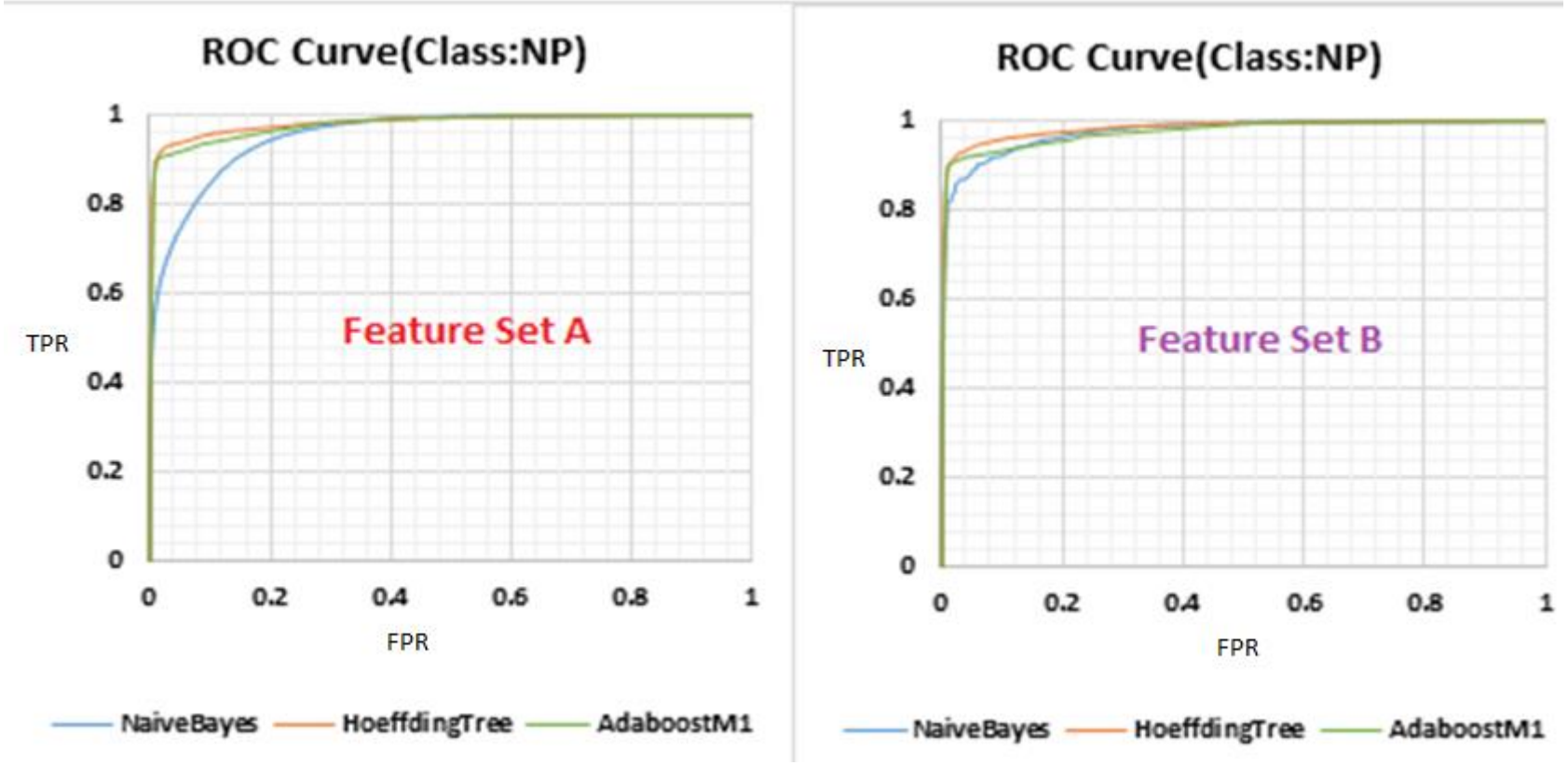
C.1 Design of Knowledge Flow to Analyse ROC and PRC

The knowledge flow was designed to automatically run the knowledge discovery process. This process was designed the same way as how the experimenter was setup in B.1 . The knowledge flow was done solely to be able to generate the ROC curves and Precision-Recall curves to further explore different classification behavior and generalisation possibilities



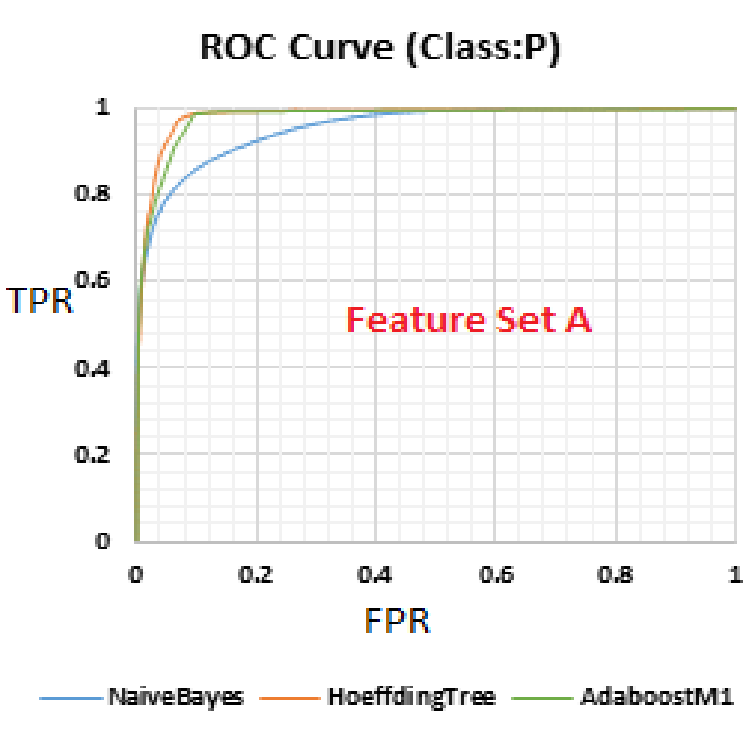
C.2 Experiment A and B ROC: Class NP

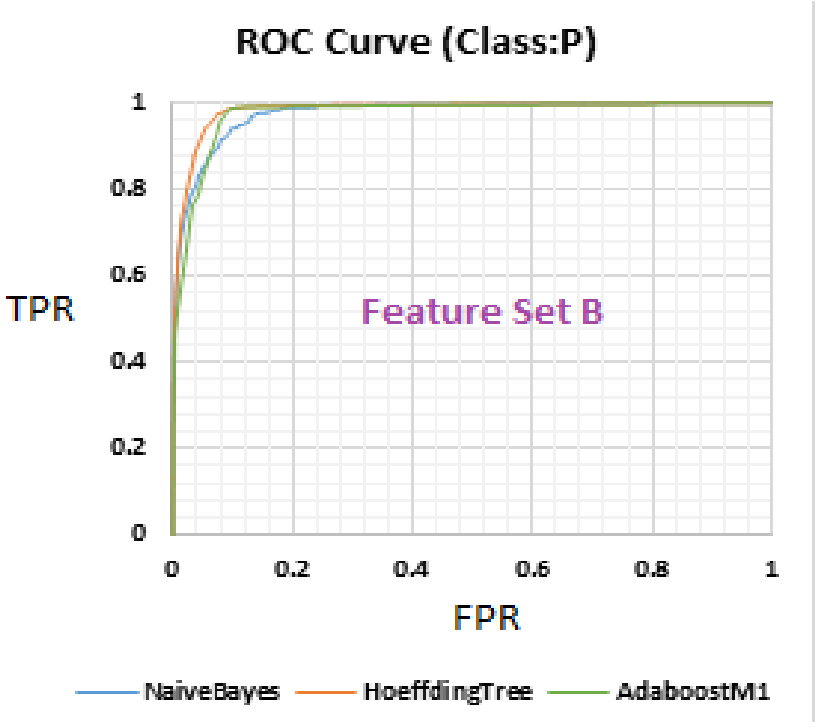
This is ROC for class NP: generated by the knowledge flow in C.1. The ROC curve below can help to assess decision bias of class NP



C.3 Experiment A and B ROC: Class P

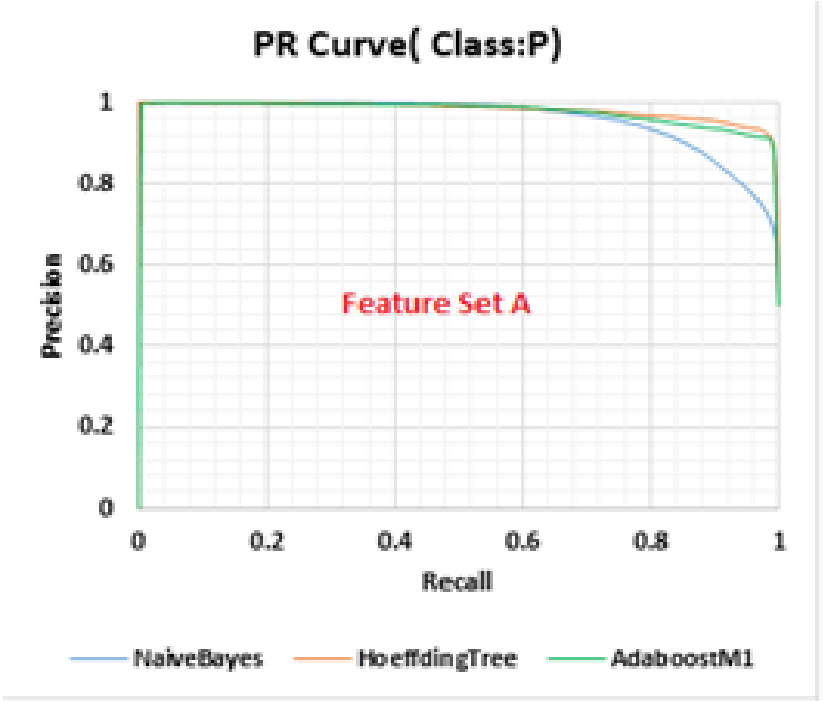
This is ROC for class P: generated by the knowledge flow in C.1 above. The ROC curve below can help to assess decision bias of class P

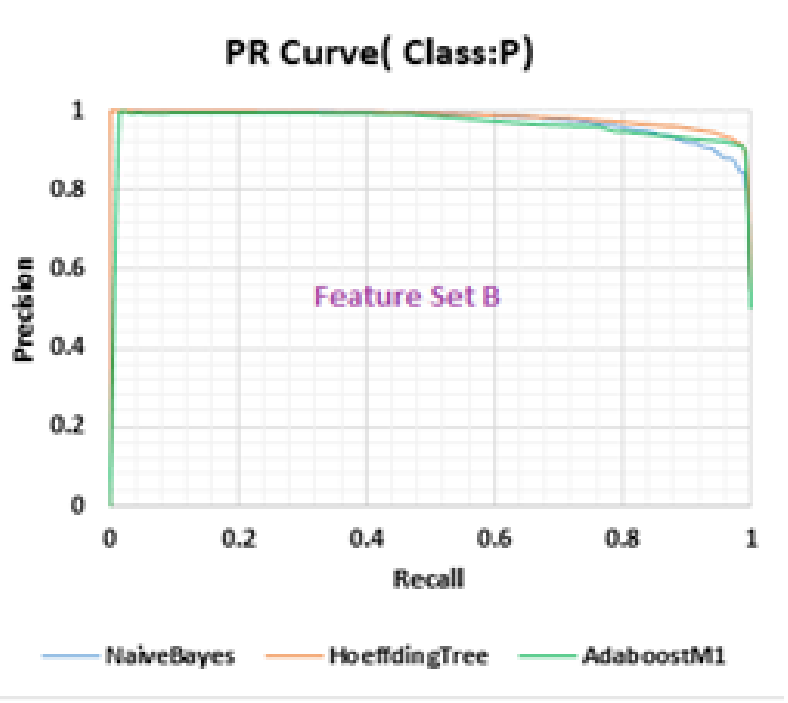




C.4 Experiment A & B PRC: Class P

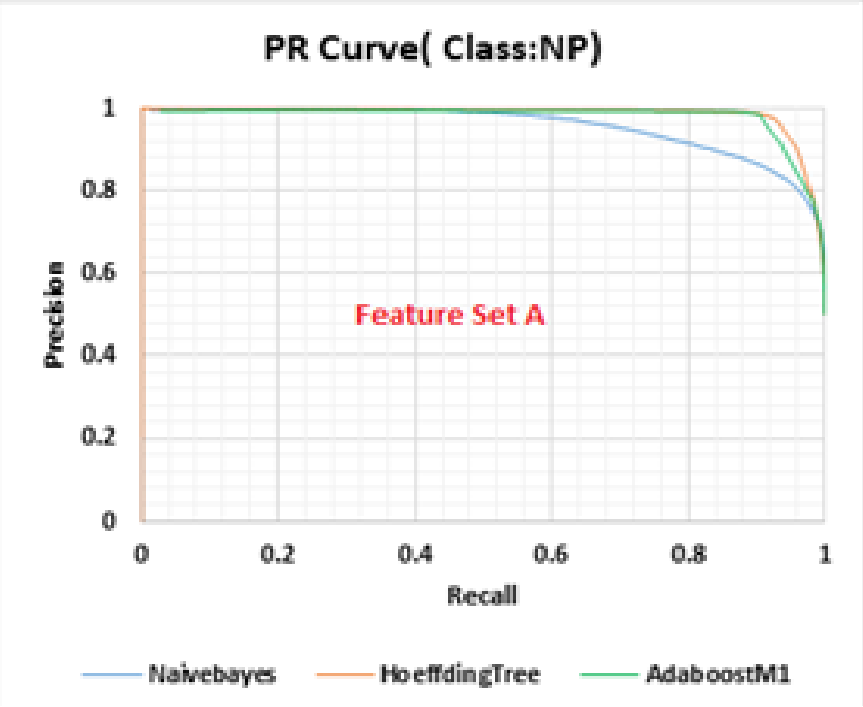
This is a Precision-Recall Curve for class P: generated by the knowledge flow in C.1. The Precision-Recall Curve below helps to assess the sensitivity of model against class P

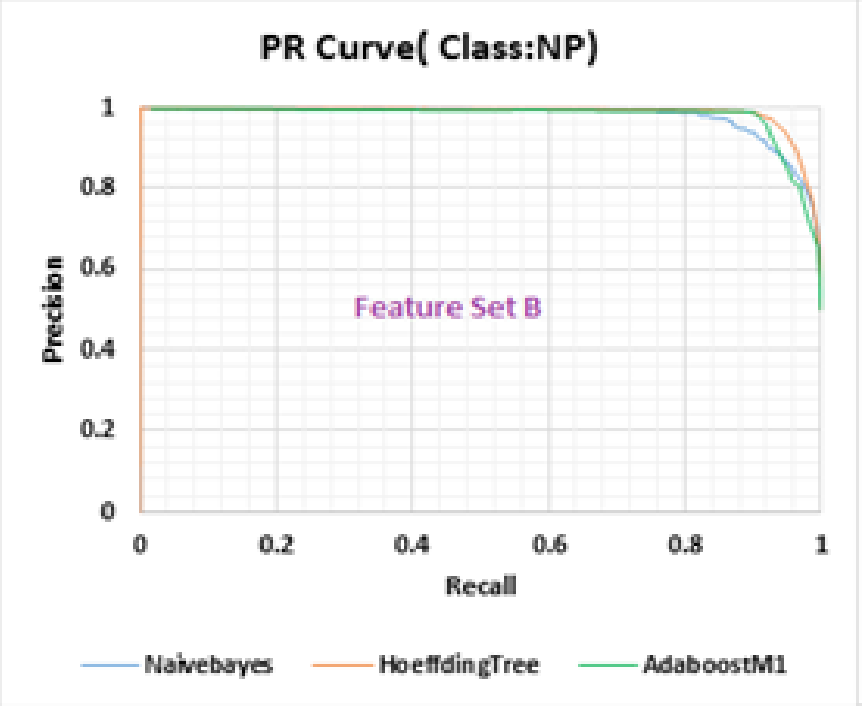




C.5 Experiment A PRC: Class NP

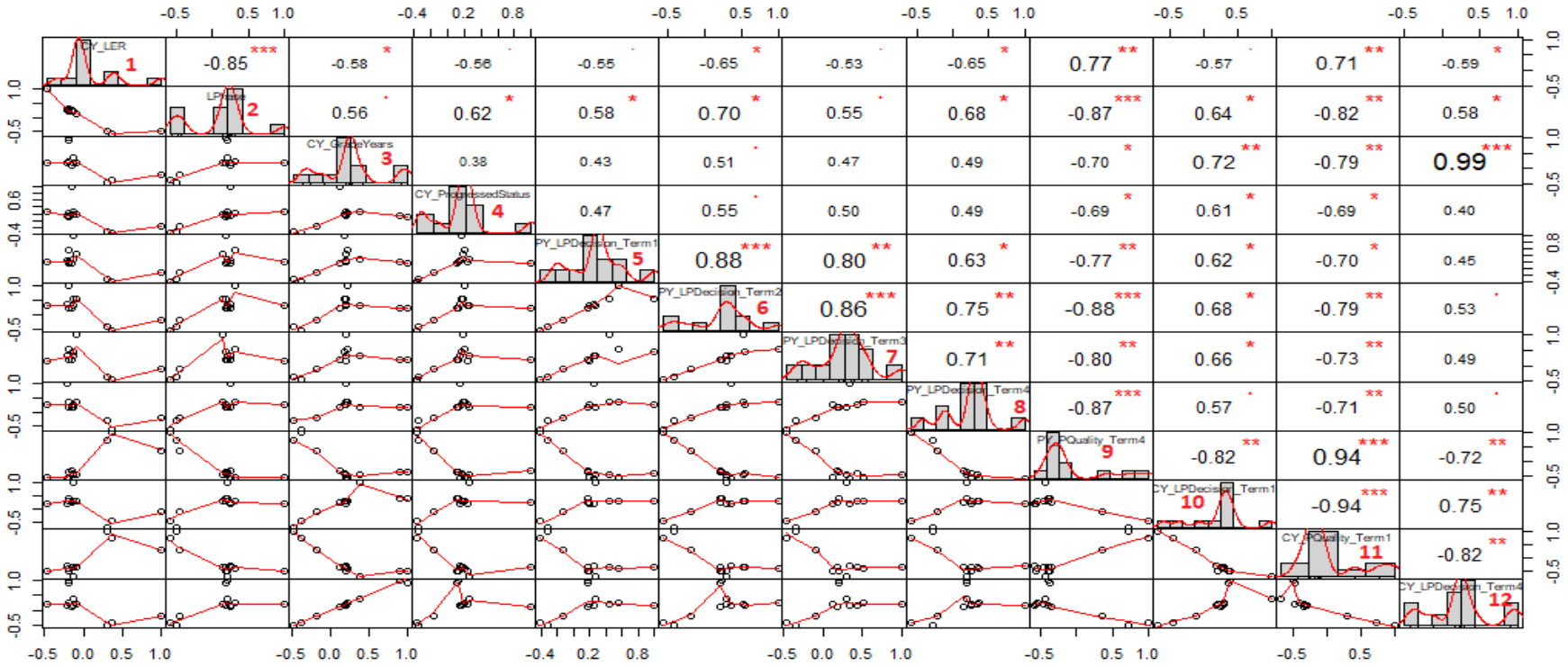
This is a Precision-Recall Curve for class NP: generated by the knowledge flow in C.1. The Precision-Recall Curve below helps to assess the sensitivity of model against class NP





Annexure D (correlation analysis)

D.1 Correlation plot (11 Attributes from the 2nd exploratory study)



Notes to guide interpretation of the graph

- The distribution of each attribute is shown on the diagonal.
- The bottom of the diagonal shows bivariate scatter plots of attributes on each cross-section
- The top of the diagonal shows the value of the correlation and the significance level as stars where the p-values (0, 0.001, 0.01, 0.05, 0.1, 1) are represented as symbols (“***”, “**”, “*”, “.”, " “) respectively

Annexure E (Discretize Data Structure)

E.1 Descritise Data Structure

```
@relation 'All-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Discretize-F-B3-M-1.0-R1,2,3,4,13,20-weka.filters.unsupervised.attribute.Remove-R12'

@attribute CY_GPI {'\'(-inf-0.905]\'','\'(0.905-0.995]\'','\'(0.995-inf)\''}
@attribute CY_LER {'\'(-inf-38.625]\'','\'(38.625-45.255]\'','\'(45.255-inf)\''}
@attribute CY_SLAbsentee_Rate {'\'(-inf-0.795]\'','\'(0.795-1.415]\'','\'(1.415-inf)\''}
@attribute CY_SEAbsentee_Rate {'\'(-inf-2.815]\'','\'(2.815-4.915]\'','\'(4.915-inf)\''}
@attribute SQuintile {Very-Poor,Moderately-Poor,Fairly-Poor,Least-Poor,Poor}
@attribute SDistrict {Vhembe,Sekhukhune,Mogalakwena,Polokwane,Mopani,Lebowakgomo,Tshipise-Sagole,Tzaneen,Riba-Cross,Waterberg}
@attribute LCitizenship {Citizen,'Immigrant '}
@attribute LGender {Male,Female}
@attribute LHomeLanguage {XiTsonga,SePedi,TshiVenda,IsiNdebele,IsiZulu,English,SeTswana,SeSotho,SiSwati,Afrikaans,'Other ',IsiXhosa,Sign-Language}
@attribute LRace {African-Black,Asian-Indian,White,Coloured,'Other '}
@attribute LPhase {Senior,Foundation,Intermediate,FET}
@attribute CY_GradeYears {'\'(-inf-1.5]\'','\'(1.5-2.5]\'','\'(2.5-inf)\''}
@attribute CY_ProgressedStatus {TRUE,FALSE}
@attribute PY_LPDecision_Term1 {NP,P}
@attribute PY_LPDecision_Term2 {NP,P}
@attribute PY_LPDecision_Term3 {NP,P}
@attribute PY_LPDecision_Term4 {P,NP}
@attribute PY_PQuality_Term4 {1,2,3,4,5,6,7}
@attribute CY_LAbsenteeRate_Term1 {'\'(-inf-1.04]\'','\'(1.04-3.125]\'','\'(3.125-inf)\''}
@attribute CY_LPDecision_Term1 {NP,P}
@attribute CY_PQuality_Term1 {1,2,3,4,5,6,7}
@attribute CY_LPDecision_Term4 {P,NP}

@data
'\'(-inf-0.905]\'','\'(-inf-38.625]\'','\'(0.795-1.415]\'','\'(4.915-inf)\'',Very-Poor,Vhembe,Citizen,Male,XiTsonga,African-Black,Senior,'\'(-inf-1.5]\'',TRUE,NP,NP,NP,P,2,'\'(-inf-
'\'(0.995-inf)\'','\'(45.255-inf)\'','\'(0.795-1.415]\'','\'(4.915-inf)\'',Very-Poor,Sekhukhune,Citizen,Female,SePedi,African-Black,Foundation,'\'(-inf-1.5]\'',FALSE,P,P,P,P,4,'\'(-
'\'(-inf-0.905]\'','\'(45.255-inf)\'','\'(0.795-1.415]\'','\'(4.915-inf)\'',Very-Poor,Mogalakwena,Citizen,Male,SePedi,African-Black,Foundation,'\'(-inf-1.5]\'',FALSE,P,P,P,P,6,'\'(-
'\'(0.905-0.995]\'','\'(38.625-45.255]\'','\'(0.795-1.415]\'','\'(2.815-4.915]\'',Moderately-Poor,Polokwane,Citizen,Male,SePedi,African-Black,Intermediate,'\'(-inf-1.5]\'',FALSE,P,F
'\'(-inf-0.905]\'','\'(45.255-inf)\'','\'(1.415-inf)\'','\'(2.815-4.915]\'',Moderately-Poor,Mopani,Citizen,Male,SePedi,African-Black,Foundation,'\'(-inf-1.5]\'',FALSE,P,P,P,P,5,'\'(
'\'(0.905-0.995]\'','\'(-inf-38.625]\'','\'(-inf-0.795]\'','\'(4.915-inf)\'',Very-Poor,Vhembe,Citizen,Female,TshiVenda,African-Black,Intermediate,'\'(-inf-1.5]\'',FALSE,NP,P,P,P,4,'
'\'(0.995-inf)\'','\'(38.625-45.255]\'','\'(0.795-1.415]\'','\'(2.815-4.915]\'',Very-Poor,Lebowakgomo,Citizen,Male,SePedi,African-Black,Foundation,'\'(-inf-1.5]\'',FALSE,NP,NP,NP,P,
```

Annexure F (Experimental Results)

F.1 Experimental Results

	Set B			Set A			Additional Info
	HoeffdingTree	**NaïveBayes**	**AdaBoostM1**	**HoeffdingTree**	**NaïveBayes**	**AdaBoostM1**	
Number_of_training_instances	35908	35908	35908	35908	35908	35908	
Number_of_testing_instances	83786	83786	83786	83786	83786	83786	
Number_correct	79569	76181	79229	79700	73638	79229	
Number_incorrect	4217	7605	4557	4086	10148	4557	
Number_unclassified	0	0	0	0	0	0	
Confusion Matrix							
Num_true_negatives	38499	39263	37892	38606	38796	37892	Specificity
Num_false_positives	3348	2584	3955	3241	3051	3955	Type 1 error
Num_false_negatives	869	5021	602	845	7097	602	Type 2 error
Num_true_positives	41070	36918	41337	41094	34842	41337	Sensitivity/Recall
Calculations Derived from Confusion Matrix							
Percent_correct	94.97	90.92	94.56	95.12	87.89	94.56	Ideal=100%
IR_precision	0.92	0.93	0.91	0.93	0.92	0.91	Ideal=100%
IR_recall	0.98	0.88	0.99	0.98	0.83	0.99	Sensitivity (Ideal=100%)
True_negative_rate	0.92	0.94	0.91	0.92	0.93	0.91	Specificity (Ideal=100%)
False_positive_rate	0.08	0.06	0.09	0.08	0.07	0.09	Type 1 error
False_negative_rate	0.02	0.12	0.01	0.02	0.17	0.01	Type 2 error (Ideal=0%)
Additional Performance Evaluation Metrics							
Kappa_statistic	0.90	0.82	0.89	0.90	0.76	0.89	Ideal =1.00
Root_mean_squared_error	0.20	0.25	0.22	0.20	0.31	0.21	Ideal =0.00
Matthews_correlation	0.90	0.82	0.89	0.90	0.76	0.89	Ideal =1.00
Area_under_ROC	0.98	0.98	0.98	0.98	0.96	0.98	Ideal =1.00
Area_under_PRC	0.98	0.98	0.97	0.98	0.96	0.97	Ideal =1.00