UNIVERSITY OF NAIROBI

SCHOOL OF COMPUTING AND INFORMATICS

DECISION SUPPORT SYSTEM ON BAD DEBT RECOVERY IN THE DEPOSIT PROTECTION FUND BOARD - KENYA

By

Simon Nyahe Waithaka

P56/7424/2005

August 2011

Project submitted in partial fulfillment of the degree of Masters of Science in Information Systems

DECLARATION

THIS PROJECT IS MY ORIGINAL WORK AND HAS NOT BEEN SUBMITTED FOR A DEGREE IN ANY OTHER UNIVERSITY

Signed:

Simon N. Waithaka

P56/P/7424/2005

Date: 26/9/2011

THIS PROJECT HAS BEEN SUBMITTED FOR EXAMINATION WITH MY APPROVAL AS A UNIVERSITY SUPERVISOR

Signed:

Muchemi L.

Lecturer

School of Computing and Informatics

University of Nairobi

Date: ~~20~~ 6TH OCT 2011

ABSTRACT

The Deposit Protection Fund Board (DPFB), in particular Institutions in Liquidations division, is often plagued by a lengthy liquidation process, unpaid loans, collection agency fees and various legal charges. The unpaid loans and the recovery procedures contribute significantly to the rising cost and increasing length of the liquidation process. The DPFB does not have any decision support tools that can be use to guide on debt recovery for institutions in liquidation. They rely on manual methods such us response to demand notes, the existence of security and the general availability of documents on the loan to classify whether a loan is a potentially good or bad debt.

Institutions in liquidation have massive loans data, which have been utilised in this project to aid the learning process of data mining tools in evaluating whether a particular debtor is likely to pay their debts. These results are meant to act as an enhancement to the mentioned manual methods.

This research explored the effectiveness of various data mining tools in evaluating whether a debt is likely to be repaid. The research involved recognition of the manual methods DPFB uses in classifying debts, the IT measures taken by DPFB to enhance debt recovery and the level of success achieved so far.

Loans data was collected from 27 institutions' databases. Data was prepared by selecting suitable variables (predictive and target). The predictive variables were four, namely; amount at liquidation, contacts availability, debt type and customer type. The target variable was the indication on whether a debt is good or bad.

Seven data mining methods were selected based on guidance from literature review. Predictive modelling software – DTREG was employed to train and validate the data mining techniques. The performance of each of the model was analysed using confusion matrix and area under receiver operating curve. Results were obtained from both balanced and imbalanced data. Balanced data performed better than imbalanced. The results which are detailed in chapter four depict that neural networks tools generally gave high accuracy. These findings guided to the development of a DSS prototype based on neural networks. This model can be used to aid DPFB decision support on debt recovery.

A neural network based DSS software - Alyuda Forecaster XL whose details a highlighted in section 4.5 formed the Model Based Management System (MBMS) component of the developed DSS prototype. On testing the software using different number of debt records, it was noted its classification accuracy increased with higher number of records.

The Data Base Management System (DBMS) part of the DSS was based on SQL 2000 while the Dialog Generation and Management System (DGMS) was based on Visual Studio and Crystal Report software.

In conjunction with the manual methods currently used by DPFB to classify debts, the developed decision support system is expected to enhance the accuracy of classification that may eventually lead to curbing the lengthy liquidation period and reduce expenses incurred in the process.

ACKNOWLEDGEMENTS

wish to express my profound gratitude first and foremost to my God creator of heaven and gave me the insight, ability and strength to be able to conceptualize and perform this study. nks also goes to my supervisor in this project, Mr. Lawrence Muchemi, who is a lecturer at the Computing and Informatics, University of Nairobi for his kind and tireless guidance throughout h. In particular he helped me moderate my ideas and ensured that the study was on track and quired masters' level standards.

ly thanks are also extended to my dear wife Grace for her constant encouragement and support t the study. To my children Victor, Joy, Praises and Enoch whose comments and mere round me was a great encouragement.

o my workmates in DPFB who enhanced my understanding of the debt recovery processes the manual methods used in classifying debts.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

ACKNOWLEDGEMENTS iii

TABLE OF CONTENTS iv

LIST OF FIGURES vii

LIST OF TABLES viii

LIST OF TERMINOLOGY AND ABBREVIATIONS ix

CHAPTER ONE 1

INTRODUCTION 1

1.1 Background of the study 1

1.2 Statement of the problem 1

1.3 Purpose of the study 2

1.4 Objective of the study 2

1.5 Research questions 3

1.6 Significance of the study 3

1.7 Limitations of the study 4

1.8 Assumption of the study 4

1.9 Organization of the study 4

CHAPTER TWO 6

LITERATURE REVIEW 6

2.0 Introduction 6

2.1 Contributions to bad debt recovery by scholars 6

2.2 CHARACTERISTICS OF SAMPLED DATAMININIG TOOLS 9

2.2.1 K- Nearest Neighbor (K-NN) - (Case- or Memory-based Reasoning) 9

Radial Basis Function 17

2.2.6 K-Means Clustering 23

2.2.7 Logistic Regression 26

2.2.8 Knowledge discovery and Data Mining 28

2.3 Decision Support Systems 29

2.3.1 Introduction 29

2.3.2 Characteristics of Decision Support Systems 29

2.3.3 Examples of DSS Solutions and their Utilization 30

2.3.5 Architecture of decision support systems 33

CHAPTER THREE 36

RESEARCH AND METHODOLOGY 36

3.1 Introduction 36

3.3 Location of the study 37

3.4 Target population 37

3.5 Data Mining Methodology 38

3.6 Study variables 38

3.7 Data Analysis and Validation Techniques 39

3.7.1. Simple validation 39

3.7.2 Cross validation 39

3.7.3. Confusion Matrix 40

3.7.4 Area under ROC curve 40

3.7.5. Prototype Design and Development 41

CHAPTER FOUR 42

ANALYSIS OF RESULTS AND DISCUSSIONS 42

4.1 Project Data 42

4.1.0 Data Collection 42

4.1.1 Data Preparation 43

4.1.2 Selected Data Mining Tools 44

4.2 Training and Validation 44

4.2.1 Cross Validation 44

4.2.2 Confusion Matrix 44

4.2.3. Area under ROC curve 45

4.3 Results of Models Assessment Based on Confusion Matrix & ROC Charts 45

4.3.1 Decision Tree 45

4.3.2 Neural Networks Multilayer Perception with 4 layers (2 hidden) 47

4.3.3 Neural Networks Multilayer Perceptron with 3 layers(1 hidden) 50

4.3.4 Radial Basis Function (RBF) Neural Networks – Conceptually similar to K-

4.3.5 Probabilistic Neural Networks (PNN) and General Regression Neural Networks (GRNN) – ... 54

4.3.7 K-Mean Clustering 60

4.4 Design of the Prototype 63

4.4.1 Input Table Design 66

4.4.2 Output Design 66

4.4.5 Program Design 69

4.5.1 Testing of Alyuda Forecaster XL 70

4.5.2 Size of Data and Accuracy in Alyuda Forecaster XL 72

CHAPTER FIVE 73

5.0 CONCLUSIONS AND RECOMMENDATIONS 73

5.1 REVIEW OF OBJECTIVES OF THE STUDY 73

5.2 ACHIEVEMENTS OF THE STUDY 73

5.2.1 IT Measures Taken to Enhance Debt Recovery 73
5.2.2 Methods used to Classify Debts 73
base their classification of good and bad debts on: 73
5.2.3 Success Achieved in Debt Recovery 74
5.2.4 Determination of Appropriate Data Mining Tool 74
5.2.5 Development of DSS Prototype 74
5.3 CONCLUSIONS 75
5.3 RECOMMENDATIONS 76
5.4 SUGGESTIONS FOR FURTHER WORK 76
REFERENCES 77
APPENDICES 79
APPENDIX I: 4.6 A DECISION SUPPORT SYSTEM PROTOTYPE 79
APPENDIX II: SAMPLE DATA AND OUTPUT 97
APPENDIX III: PROTOTYPE SAMPLE CODE SAMPLE 99
APPENDIX IV: LETTER OF INTRODUCTION TO THE DIRECTOR – DPFB 102
APPENDIX V: REPLY LETTER FROM THE DIRECTOR - DPFB 103
APPENDIX VI: GUIDE TO USING DT-REG DEMONSTRATION SOFTWARE 104
.... 106

LIST OF FIGURES

Figure 2.1_Decision Tree Nodes 10
Figure 2. 3 RBF - K-Nearest Neighbor (k-NN) 16
Figure 2.4 Radial Basis Transfer Function 17
Figure 2.5 RBF function on Three dimensions 18
Figure 2. 6 PNN/GRNN - K-Nearest Neighbor (k-NN) 19
Figure 2.7 Radial Basis Transfer Function - PNN/GRNN 20
Figure 2.8 Radial Basis Transfer Function on Three Dimension 21
Figure 2.9 PNN/GRNN Network: 22
Figure 2.10 PNN/GRNN Network: 25
Figure 2. 11 Logistic Regression Model - Dose-Response Curve 27
Figure 2.12 Components of decision support system. Source: Sprague and Carlson 34
Figure 2.13Components of the Expected Prototype 35
Figure 3.1 DTREG Work Flow 37
Figure 4.1 DT - Imbalanced data - Area under ROC curve (AUC) 46
Figure 4.2 DT - Balanced data - Area under ROC curve (AUC) 47
Figure 4.3 NN MP 4/2 Balanced data - Area under ROC curve (AUC) = 0.88731 50
Figure 4.5 NN MP 3/1 Imbalanced data - Area under ROC curve (AUC) 51
Figure 4.6 NN MP 3/1 Balanced data - Area Under ROC curve (AUC) 52
Figure 4.7 RBF NN Imbalanced data - Area under ROC curve (AUC) 53
Figure 4.8 RBF NN Balanced data - Area under ROC curve (AUC) 54
Figure 4.9 PNN/GRNN - Imbalance Data - Area under ROC curve (AUC) 56
Figure 4.10 PNN/GRNN - Balance Data - Area under ROC curve (AUC) 57
Figure 4.11 LR - Imbalanced Data - Area under ROC curve (AUC) 59
Figure 4.12 LR - Balanced Data - Area under ROC curve (AUC) 60
Figure 4.13 Imbalanced data - Area under ROC curve (AUC) 61
Figure 4.14 K-MC - Balanced data - Area under ROC curve (AUC) 62
Figure 4.15 Loan Master Table 66
Figure 4.16 Project Data Table 67
Figure 4.17 Text File Output (Comma delimited) 68
Figure 4.18 Report Design 68
Figure 4.19 Data Preparation Process 69

LIST OF TABLES

Table 2.1 Data Mining Problems and Appropriate Modeling Techniques........ 8

Table 3. 1 Theoretical Confusion Matrix........ 40

Table 4.1 Confusion Matrix DT - Imbalanced data........ 45

Table 4.3 MP NN 4/2 - Imbalanced data........ 47

Table 4.4 NN MP 3/1 Balanced data........ 51

Table 4.6 RBF NN Imbalanced data........ 52

Table 4.7 RBF NN Balanced data........ 53

Tab le 4.8 PNN/GRNN - Imbalanced Data........ 55

Table 4.10 LR - Imbalanced data........ 57

Table 4.11 LR - Balanced data........ 58

Table 4.12 K-MC - Imbalanced data........ 60

Table 4.13 Tabulated Results of the Assessment........ 63

Table 4.14 K-MC - Balanced data........ 61

Table 4.15 Tabulated Results of the Assessment........ 63

Table 4.16 Confusion Matrix – Alyuda Software........ 70

Table 4.17 Tabulated Results – Different sizes of data........ 70

LIST OF TERMINOLOGY AND ABBREVIATIONS

1. DPFB - Deposit Protection Fund Board
2. IT - Information Technology
3. KNN - K- Nearest Neighbor
4. DT – Decision Tree
5. ANN - Artificial Neural Networks
6. RBF - Radial Basis Function
7. PNN/GRNN – Probabilistic Neural Network/General Regression Neural Networks
8. MLP - Multiple Level Perceptron
9. CRISP-DM - CRoss Industry Standard Process for Data Mining
10. HSDSS - Home Loan Packages Selection Decision Support System Using Financial Model
11. DSS – Decision Support System
12. HTML – HyperText Markup Language
13. CDC - Centers for Disease Control
14. NAMCS - National Ambulatory Medical Care Survey
15. DBMS - Data base management system
16. MBMS - Model - based management system
17. DGMS - Dialog generation and management system
18. AUC – Area Under Curve
19. ROC – Receiver Operating Curve
20. NNMP 3/1 – Multi Perceptron (3 Layers 1 hidden)
21. NNMP 4/2 – Multi Perceptron (4 Layers 2 hidden)
22. CRISP-DM (CRoss Industry Standard Process for Data Mining)
23. CRB - Credit Reference Bureau
24. LR – Logistic Regression
25. K-MC - K-Mean Clustering
26. CBK – Central Bank of Kenya

CHAPTER ONE
INTRODUCTION

1.1 Background of the study

Deposit Protection Fund Board (DPFB) was established in 1986 in pursuant to the Banking Act Cap. 488, Section 36. DPFB has a principal objective of contributing to the stability of the country's financial system and to protect the less-financially-sophisticated depositors from loss of their deposits in the event of banks failure. DPFB is mandated with the responsibility of insuring deposits, paying the guaranteed amounts and liquidation of the failed banks and other financial institutions. It is currently managing over 25 institutions in liquidation. Some of these institutions have been in liquidation for more than 15 years. This project focuses on the liquidation of failed banks and other financial institutions and the usage of various data mining tools to aid decision support in debt recovery.

This study is intended to find out a most effective data mining tool in evaluating whether a debt is likely to be repaid. This is important information to DPFB because an early detection of non performing debts would allow the DPFB institutions in liquidation to focus preliminary debt recovery efforts on the good customers and save on administrative expenses by either writing off the bad debts or turning them over immediately to a collection agency. It has been desirable to accomplish the liquidation process in the shortest time possible but without a concrete decision support model this can continually remain an illusion. One of components that would hasten this process is to be able to make decisions on the bad debts recovery based on some decision support tools which is the main focus of this study.

1.2 Statement of the problem

Banks have found out that to predict whether a particular customer is likely to repay their debt is an inherently complex and unstructured process (Jozef Zurada et al., 2005). Thus, due to one of its mandate, to liquidate fallen banks, a process that includes debt recovery, DPFB often becomes unwilling creditors to a multitude of borrowers.

Currently Deposit Protection Fund Board does not have any decision support model that can assist a liquidator in forecasting or predicting of some vital elements in liquidation. For example it would be important to be able to estimate by what period an institution is likely to become a non performer, the estimated loan balance after a period, the estimate deposit balance after some time, the estimated investment growth after a given period, the estimated number of claimants who will claim their deposits, estimation on amount to be claimed, the estimated number of depositors who will be fully paid just to name a few. Without such a guiding model the liquidation process may unnecessarily long and expensive. The limited parameters available to the DPFB about the customer-debtor may seem to have no apparent relationship to the likelihood that a customer-debtor will repay the bad debt. However, this research made use of knowledge discovery and data mining tools that are at our disposal. Knowledge discovery is defined as the process of identifying valid, novel, and potentially useful patterns, rules, relationships, rare events, correlations, and deviations in data (Fayyad et al., 1996). This process relies on well-established technologies, such as machine learning, pattern recognition, statistics, neural networks, fuzzy logic, evolutionary computing, database theory, artificial intelligence, and high performance computing to find relevant knowledge in very large databases. The knowledge discovery process is typically composed of the following phases: understanding the overall problem domain; obtaining a data set; cleaning, pre-processing, transforming, and reducing data; applying data mining tools; interpreting mined patterns; and consolidating and implementing discovered knowledge.

Data mining, which is an important phase in the knowledge discovery process, uses a number of analytical tools: discriminant analysis, neural networks, decision trees, fuzzy logic and sets, rough sets, genetic algorithms, association rules, and k-nearest neighbour (or memory-based reasoning) which are suitable for the tasks of classification, prediction, clustering, summarisation, aggregation, and optimization (Jozef Zurada et al.,2005). The two major tasks on this project were to focus on classification and prediction. These are the most common and perhaps the most straightforward data mining tasks. Classification consists of examining the features of a newly presented object and assigning it to one of a predefined set of two classes or outcomes. The outcome here was either i) Debt is recoverable or ii) Debt is not recoverable.

A data mining model employing one of the data mining tools was trained using pre-classified examples. The goal was to build a model that will be able to accurately classify new data based on the outcomes and the interrelation of many discrete variables. The variables identified in this project are: i) Debt amount at liquidation ii) Customer contacts availability (either available or not available) iii) Period of loan at liquidation iv) Type of loan (normal loan or others) v) Customer type (staff or non staff) vi) Initial Loan amount and the target variable is whether the debt is recoverable or otherwise.

This project examined and compared the effectiveness of seven data mining techniques; Decision Tree, NN MP with 4 layers (2 hidden), NN MP with 3 layers (1 hidden), RBF NN, PNN/GRNN, Logistic Regression and K-Mean Clustering in classifying whether a debt is recoverable or not.

1.3 Purpose of the study

The purpose of this study was to select seven data mining techniques and establish which among them most accurately assesses whether a debt is likely to be good or bad. The selected tool was then utilised to develop a DSS prototype that can be used to classify. The outcome is to aid in recommending a suitable tool that can aid DPFB in decision support during their debt recovery process.

1.4 Objective of the study

- To ascertain the IT measures taken to enhance the debt recovery process in DPFB.
- To identify the methods DPFB management uses to classify debts.

- To establish the level of success achieved in the debt recovery process in DPFB.
- To determine the appropriate data mining tools in evaluating whether a debt is likely to be repaid using data from institutions in liquidation in DPFB.
- Develop a decision support prototype based on the identified data mining tool.
- Based on the finding of the study to recommend the appropriate tool that can aid DPFB management in decision support in loan recovery.

1.5 Research questions

The study will be guided the following research question:

- What are the IT measures taken by DPFB to enhance the debt recovery process?
- What are the methods currently utilized by DPFB in classifying debts?
- What is the level of success achieved in the debt recovery process in DPFB?
- Which is an appropriate data mining tools in evaluating whether a debt is likely to be repaid in DPFB?

1.6 Significance of the study

- The findings of this study are expected to be significant in offering decision support to managers of institutions in liquidation in DPFB - Kenya.
- The result of this study is expected to reduce the time taken to wind up institutions in liquidation which is currently hampered by the debt recovery process.
- The findings are expected to aid in reducing the expenses incurred during loans recovery process in that an early detection of non performing debts would allow the DPFB institutions in liquidation to focus preliminary debt recovery efforts on the good customers and save on administrative expenses by either writing off the bad debts or turning them over immediately to a collection agency.
- This outcome of the study is expected to help the DPFB liquidation managers by assisting to re-evaluate their approach on debt recovery.
- The study will also contribute to research methodology that other scholars and researcher can adopt for future research. It will also form a base on which others can develop their studies.

- Although the study will concentrate on institutions in liquidation in DPFB – Kenya, the findings of the study will also be applicable to other Deposit Insurance Schemes in other countries.

1.7 Limitations of the study

- Debts defaults may sometimes be attributed to unforeseen events or be governed by factors that may be difficult or impossible to see in the attributes of the consumer (i.e. stability of marriage, general health, and job stability). This study is not able to capture those aspects.

- This research depends on the reliability of the data available on debt recovery from institutions undergoing liquidation in DPFB - Kenya.

- This study will rely on the co-operation of the respondents who are mainly DPFB staff. The researcher may not have control of the attitudes of the respondents which may affect the validity of the responses. This is due to the fact that respondents may at times give acceptable, but not honest answers.

- The information from the available literature might not be authenticated though the research will incorporate as many resources as possible.

1.8 Assumption of the study

The researcher will base the study on the following assumptions:

- All the respondents will be co-operative and that they will give reliable and honest responses.

- The literature used is authentic.

- Data mining tools are valid and reliable measures for dealing with bad debt recovery in institutions.

- The data collected from institution in liquidation is consistent and hence suitable for the study.

- The software used to classify the selected techniques is reliable.

- The DPFB will make use of the results obtained in this study.

1.9 Organization of the study

The study will be organized into five chapters. Chapter one, which is the introduction will place the context under the following subtopics; background of the study, statement of the problem, purpose of the study, objectives of the study, research questions, significance of the study, limitations of the study and the assumptions of the study. Chapter two will review the related literature. It will subdivide into various subheadings. Chapter three will comprise of research methodology which will also be divide into various subheadings. Chapter four will consist of presentation and analysis of data collected, research findings and discussion of the research findings. Chapter five will provide the summary of the findings, conclusions, recommendations and suggestions for further research.

CHAPTER TWO
LITERATURE REVIEW

2.0 Introduction

This chapter shall focus and evaluate the relevant information on bad debt recovery from the deposit protection fund board of Kenya. The literature shows evidence that cases where complexities among variables is dominant, data mining tools such as neural networks, decision trees, rough sets as well as the k-nearest neighbor habitually provide better classification accuracy rates than common statistical.

2.1 Contributions to bad debt recovery by scholars

The evaluation of credits granted to small Belgian firms using a decision tree-based inductive learning approach to refine the credit granting process based on the impact of Type I and Type II credit errors (classifying good loans as bad loans, classifying bad loans as good loans) was carried out by Tessmer (1998) to determine the performance of data mining tools. Tessmer argued that near misses have the ability to nudge the learning process towards a more accurate definition of the boundary between positive and negative examples and recommended the procedure called the dynamic updating process that relocates the boundary between Type I and Type II -errors to define a more informed credit granting decision. Barney et al. (1999) analyzed the performance of neural networks in distinguishing between farmers defaulting on farmers Home Administration Loans. Using an unbalanced data, Barney found that neural networks perform well in correctly classifying farmers into those who made timely payments and those who did not.

In his endeavor to evaluate the performance of data mining tools in business applications of data mining, Back et al. (1996) designed several neural network models to classify the financial performance of Finnish companies. Desai et al. (1996) explored the ability of neural networks in building credit scoring models in the credit union environment. They studied data samples containing several variables collected from three credit unions and showed that neural networks were particularly useful in detecting bad loans, whereas the use decision trees outperformed neural networks in the overall (bad and good loans) classification accuracy. Jo and Han (1997) used case-based reasoning (k-nearest neighbor) and neural networks for bankruptcy prediction.

In more recent papers, Jagielska et al. (1999) used the credit approval data set to investigate the performance of neural networks, decision tree and the k-nearest neighbor when applied to automated knowledge acquisition for classification problems. Jagielska concluded that the genetic approach compared more favorably with the neural and k-nearest neighbor approaches. Piramuthu (1999) analyzed the beneficial aspects of using both neural networks and decision trees for credit-risk evaluation decisions. Neural networks performed significantly better than decision trees in terms of classification accuracy, on both training as well as testing data. West (2000) investigated the credit scoring accuracy of five neural network architectures and compared them to traditional statistical methods. Using two real world data sets and testing the models using 10-fold cross-validation, the author found that among neural architectures the mixture-of-experts and radial basis function did best, whereas among the traditional methods regression analysis was the most accurate.

Thomas (2000) surveyed the techniques for forecasting financial risk of lending to consumers. Glorfeld and Hardgrave (2000) presented a comprehensive and systematic approach to developing an optimal architecture of a neural network model for evaluating the creditworthiness of commercial loan applications. The neural network developed using their architecture was capable of correctly classifying 75% of loan applicants and was superior to neural networks developed using simple heuristics. Yang et al. (2001) examined the application of neural networks to an early warning system for loan risk assessment. Finally, Zurada (2002) investigated data mining techniques for loan-granting decisions and predicted default rates on consumer loans.

VISA International uses neural networks to detect fraudulent credit card transactions which have provided savings estimated to be $40 million over a six month period. An English company has applied neural networks in direct marketing to identify the characteristics of people most likely to respond to a direct mailing campaign. The effort was worth 40,000 new customers, equivalent to $500,000 savings in mailing costs. American Express Co. has deployed neural networks in three projects. One involves a character recognition system, another is for direct mail prospects, and the third is for portfolio management trading support system (Berry, 1995). In a pilot study to detect fraud conducted at American Express, neural networks provided an improvement of 3% over the previously used decision trees models (Punch, 1994).

The bank industry has been focused on ways to reduce bad-debt balance for the last several years. In one of the earlier studies, Zollinger et al. (1991) identified a sample of clients classified as bad debt and charity cases from several banks on the globe. They built a neural network model and found that several institutional variables such as total debt and client variables such as age, gender, insurance status, employment status, and availability of contacts were significant factors in recovering bad debt for the institutions under liquidation. A similar study was performed by Buczko (1994) who analyzed data on bad debt for several institutions under liquidation. The study confirmed that the total debt incurred by a particular client become a major issue in credit section of the financial institution as the number of unemployed persons has increased and bank revenues have declined.

In an article, Veletsos (2003) described the predictive modeling software (IBM Intelligent Miner and DB2) used for bad-debt recovery implemented by the IBM Company for the financial institutions under liquidation at Orlando. The final study was completed in 2003 and included approximately 2,400 clients. The model is based on a variety of data variables, including credit factors, demographic information and previous organizational payment patterns. The model provides the client financial service department a list of clients sorted from the most likely to pay the debt to the least likely to pay the debt. In this study, which is a substantial extension of the approach discussed by Veletsos (2003), and Zurada and

Lonial (2004), we use a larger and unbalanced sample of clients and fewer variables to test the effectiveness of the three data mining models for recovering bad-debts.

Jozef Zurada (2005) compared the effectiveness of neural networks, decision trees, logistic regression, memory-based reasoning, and the ensemble model in evaluating whether a debt is likely to be repaid in healthcare company. His findings were that the neural network, logistic regression, and the combined model produced the best classification accuracy.

In an article, LIS - Rudjer Boskovic Institute (2001) states that, generally, goals of prediction and description tasks are achieved by applying one of the primary data mining methods. In the table below data mining problem types are related to appropriate modeling techniques.

Table 2.1 Data Mining Problems and Appropriate Modeling Techniques

Classification	Rule induction methods, Decision trees, Neural networks, K-nearest neighbors, Case based reasoning
Prediction	Regression analysis, Regression trees, Neural networks, K-nearest neighbors,
Dependency analysis	Correlation analysis, Regression analysis, Association rules, Bayesian networks, Inductive logic programming
Data description and summarization	Statistical techniques, OLAP
Segmentation or clustering	Clustering techniques, Neural networks, Visualization methods

2.2 CHARACTERISTICS OF SAMPLED DATAMININIG TOOLS

This section provides the basic features of the sampled tools without inclusion the mathematical descriptions of the algorithms underlying the operation of each tool.

2.2.1 K- Nearest Neighbor (K-NN) - (Case- or Memory-based Reasoning)

In pattern recognition, the k-nearest neighbors' algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space. k-NN is a type of instance based learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of its nearest neighbor.

It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. A common weighting scheme is to give each neighbor a weight of $1/d$, where d is the distance to the neighbor. This scheme is a generalization of linear interpolation. The neighbors are taken from a set of objects for which the correct classification is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. The *k*-nearest neighbor algorithm is sensitive to the local structure of the data. Nearest neighbor rules in effect compute the decision boundary in an implicit manner. It is also possible to compute the decision boundary itself explicitly, and to do so in an efficient manner so that the computational complexity is a function of the boundary complexity.

Some KNN advantages are described in follows: a) Simple to use; b) Robust to noisy training data, especially if the inverse square of weighted distance is used as the "distance" measure; and c) Effective if the training data is large.

Decision Trees (DT)

A decision tree is a logical model represented as a binary (two-way split) tree that shows how the value of a target variable can be predicted by using the values of a set of predictor variables. An example of a decision tree is shown below:

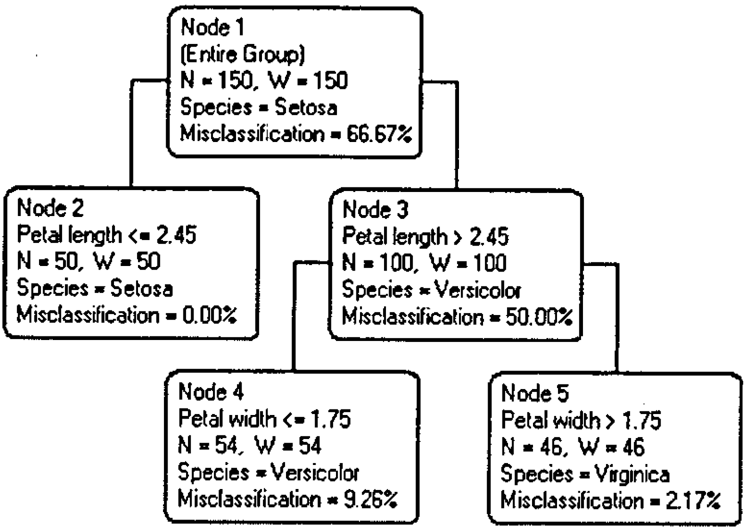


Figure 2.1

Decision Tree Nodes

The rectangular boxes shown in the tree are called "nodes". Each node represents a set of records (rows) from the original dataset. Nodes that have child nodes (nodes 1 and 3 in the tree above) are called "interior" nodes. Nodes that do not have child nodes (nodes 2, 4 and 5 in the tree above) are called "terminal" or "leaf" nodes. The topmost node (node 1 in the example) is called the "root" node. (Unlike a real tree, decision trees are drawn with their root at the top). The root node represents all the rows in the dataset.

In the top of the node box is the node number. Use the node number to find information about the node in the reports generated by DTREG. The "N = nn" line shows how many rows (cases) fall in the node. The "W = nn" line shows the sum of the weights of the rows in the node.

Splitting Nodes

A decision tree is constructed by a binary split that divides the rows in a node into two groups (child nodes). The same procedure is then used to split the child groups. This process is called recursive partitioning. The split is selected to construct a tree that can be used to predict the value of the target variable.

Decision trees are particularly useful for classification tasks. Like neural networks, decision trees learn from data. Using search heuristics, decision trees find explicit and understandable rules-like relationships among the input and output variables. Search heuristics use recursive partitioning algorithms to split the original data into finer and finer subsets, or clusters. The algorithms have to find the optimum number of splits and determine where to partition the data to maximize the information gain. The fewer the splits, the more explainable the output as there are fewer rules to understand.

Decision trees are built of nodes, branches and leaves that indicate the variables, conditions, and outcomes, respectively. The most predictive variable is placed at the top node of the tree. The algorithms make the clusters at the node gradually purer by progressively reducing disorder in the original data set. Disorder and impurity can be measured by the well-established measures of entropy and information gain borrowed from information theory. One of the most significant advantages of decision trees is the fact that knowledge can be extracted and represented in the form of classification rules. Each rule represents a unique path from the root to each leaf. In addition, at each node, one can measure the number of cases entering the node, the way those cases would be classified if these were leaf nodes, and the percentage of records classified correctly.

Some decision tree advantages includes: a) Easy to understand b) Map nicely to a set of business rules c) Applied to real problems d) Make no prior assumptions about the data e) Able to process both numerical and categorical data.

2.2.2 Artificial Neural Networks (ANN)

A Brief History of Neural Networks

The selection of the name —neural network was one of the great PR successes of the Twentieth Century. It certainly sounds more exciting than a technical description such as "A network of weighted, additive values with nonlinear transfer functions". However, despite the name, neural networks are far from "thinking machines" or "artificial brains". A typical artificial neural network might have a hundred neurons. In comparison, the human nervous system is believed to have about 3x1010 neurons. We are still light years from —Data on Star Trek.

The original "Perceptron" model was developed by Frank Rosenblatt in 1958. Rosenblatt's model consisted of three layers, (1) a "retina" that distributed inputs to the second layer, (2) "association units" that combine the inputs with weights and trigger a threshold step function which feeds to the output layer, (3) the output layer which combines the values. Unfortunately, the use of a step function in the neurons made the perceptions difficult or impossible to train. A critical analysis of perceptrons published in 1969 by Marvin Minsky and Seymore Papert pointed out a number of critical weaknesses of perceptrons, and, for a period of time, interest in perceptrons waned.

Interest in neural networks was revived in 1986 when David Rumelhart, Geoffrey Hinton and Ronald Williams published "Learning Internal Representations by Error Propagation". They proposed a multilayer neural network with nonlinear but differentiable transfer functions that avoided the pitfalls of the original perceptron's step functions. They also provided a reasonably effective training algorithm for neural networks.

Usage of Neural Networks

Neural networks are predictive models loosely based on the action of biological neurons. Artificial neural network models are used in a variety of applications, for example, nonlinear mapping, data reduction, pattern recognition, clustering, and classification. In this paper classification application of the tool will be focused. Artificial neural networks are one of disciplines of artificial intelligence which attempts to implement some of the powerful characteristics of the human brain on digital computers. Neural networks learn the nonlinear relationships, patterns, and trends in the data when the training data are presented to the network. Once trained, the artificial neural network models make high fidelity predictions for a fresh data set not seen by the network during training. The study shall apply a popular three-layer feed-forward neural network with back propagation. The network has three layers, input, hidden, and output layer.

Network inputs are fed to the input layer and the output layer and the output layer produces output. The middle layer is called "hidden layer" since it does not communicate with the environment directly. All three layers contain a number of neurons that are connected by means of connection weights. The number of neurons in the input layer equals the number of inputs to the network, while the number of neurons in the output layer corresponds to the number of system outputs. Each neuron contains two elements: the summation node and the sigmoid transfer function. The summation node calculates the product of each normalized input value and the weight value.

The problem of modeling consists of finding a proper set of connection weights using a suitable optimization algorithm so that the error between predicted and experimental outputs is minimized. Neural networks do not depend on the assumptions about the independence and distribution of residuals of input variables. On the other hand, a large volume of data is required for training, and the neural networks parameters provide little insight into the physics of the process and this constitutes the disadvantage of using neural networks.

Neural networks offer a number of advantages which includes: a) Requiring less formal statistical training b) Ability to implicitly detect complex nonlinear relationships between dependent and independent variables c) Ability to detect all possible interactions between predictor variables d) The availability of multiple training algorithms.

The Multilayer Perceptron Neural Network Model

The following diagram illustrates a perceptron network with three layers:

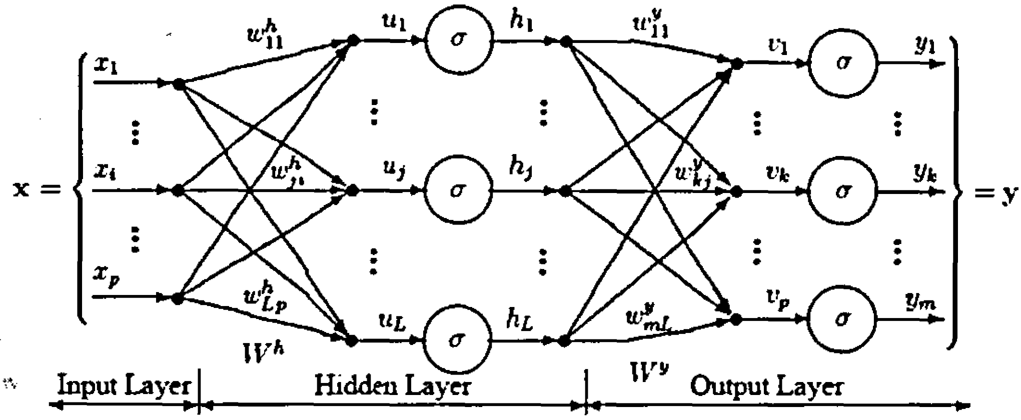


Figure 2.2 Multilayer Perceptron Neural Network Model

This network has an input layer (on the left) with three neurons, one hidden layer (in the middle) with three neurons and an output layer (on the right) with three neurons. There is one neuron in the input layer for each predictor variable (x1...xp). In the case of categorical variables, N-1 neurons are used to represent the N categories of the variable.

Input Layer

A vector of predictor variable values (x1...xp) is presented to the input layer. The input layer (or processing before the input layer) standardizes these values so that the range of each variable is -1 to 1. The input layer distributes the values to each of the neurons in the hidden layer. In addition to the predictor variables, there is a constant input of 1.0, called the bias that is fed to each of the hidden layers; the bias is multiplied by a weight and added to the sum going into the neuron.

Hidden Layer

Arriving at a neuron in the hidden layer, the value from each input neuron is multiplied by a weight (wji), and the resulting weighted values are added together producing a combined value uj. The weighted sum (uj) is fed into a transfer function, σ, which outputs a value hj. The outputs from the hidden layer are distributed to the output layer.

Output Layer

Arriving at a neuron in the output layer, the value from each hidden layer neuron is multiplied by a weight *(wkj)*, and the resulting weighted values are added together producing a combined value *vj*. The weighted sum *(vj)* is fed into a transfer function, σ, which outputs a value *yk*. The *y* values are the outputs of the network.

If a regression analysis is being performed with a continuous target variable, then there is a single neuron in the output layer, and it generates a single *y* value. For classification problems with categorical target variables, there are *N* neurons in the output layer producing *N* values, one for each of the *N* categories of the target variable.

Multilayer Perceptron Architecture

The network diagram shown above is a full-connected, three layers, feed forward, perceptron neural network. —Fully connected means that the output from each input and hidden neuron is distributed to all of the neurons in the following layer. —Feed forward means that the values only move from input to hidden to output layers; no values are fed back to earlier layers (a Recurrent Network allows values to be fed backward).

All neural networks have an input layer and an output layer, but the number of hidden layers may vary. Here is a diagram of a perceptron network with two hidden layers and four total layers:
When there is more than one hidden layer, the output from one hidden layer is fed into the next hidden layer and separate weights are applied to the sum going into each layer.

Training Multilayer Perceptron Networks

The goal of the training process is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible.
There are several issues involved in designing and training a multilayer perceptron network:

- Selecting how many hidden layers to use in the network.
- Deciding how many neurons to use in each hidden layer.
- Finding a globally optimal solution that avoids local minima.
- Converging to an optimal solution in a reasonable period of time.
- Validating the neural network to test for overfitting.

Selecting the Number of Hidden Layers

For nearly all problems, one hidden layer is sufficient. Two hidden layers are required for modeling data with discontinuities such as a saw tooth wave pattern. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to a local minima. There is no theoretical reason for using more than two hidden layers. DTREG can build models with one or two hidden layers. Three layer models with one hidden layer are recommended.

Deciding how many neurons to use in the hidden layers

One of the most important characteristics of a multilayer perceptron network is the number of neurons in the hidden layer(s). If an inadequate number of neurons are used, the network will be unable to model complex data, and the resulting fit will be poor.

If too many neurons are used, the training time may become excessively long, and, worse, the network may over fit the data. When overfitting occurs, the network will begin to model random noise in the data. The result is that the model fits the training data extremely well, but it generalizes poorly to new, unseen data. Validation must be used to test for this.

2.2.4 Radial Basis Function (RBF) Neural Networks (DTREG manual pg 253-256)

A Radial Basis Function (RBF) neural network has an input layer, a hidden layer and an output layer. The neurons in the hidden layer contain Gaussian transfer functions whose outputs are inversely proportional to the distance from the center of the neuron.

RBF networks are very similar to PNN/GRNN networks (see page 271). The main difference is that PNN/GRNN networks have one neuron for each point in the training file, whereas RBF networks have a variable number of neurons that is usually much less than the number of training points. For problems with small to medium size training sets, PNN/GRNN networks are usually more accurate than RBF networks, but PNN/GRNN networks are impractical for large training sets.

How RBF networks work

Although the implementation is very different, RBF neural networks are conceptually similar to *K-Nearest Neighbor* (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure:

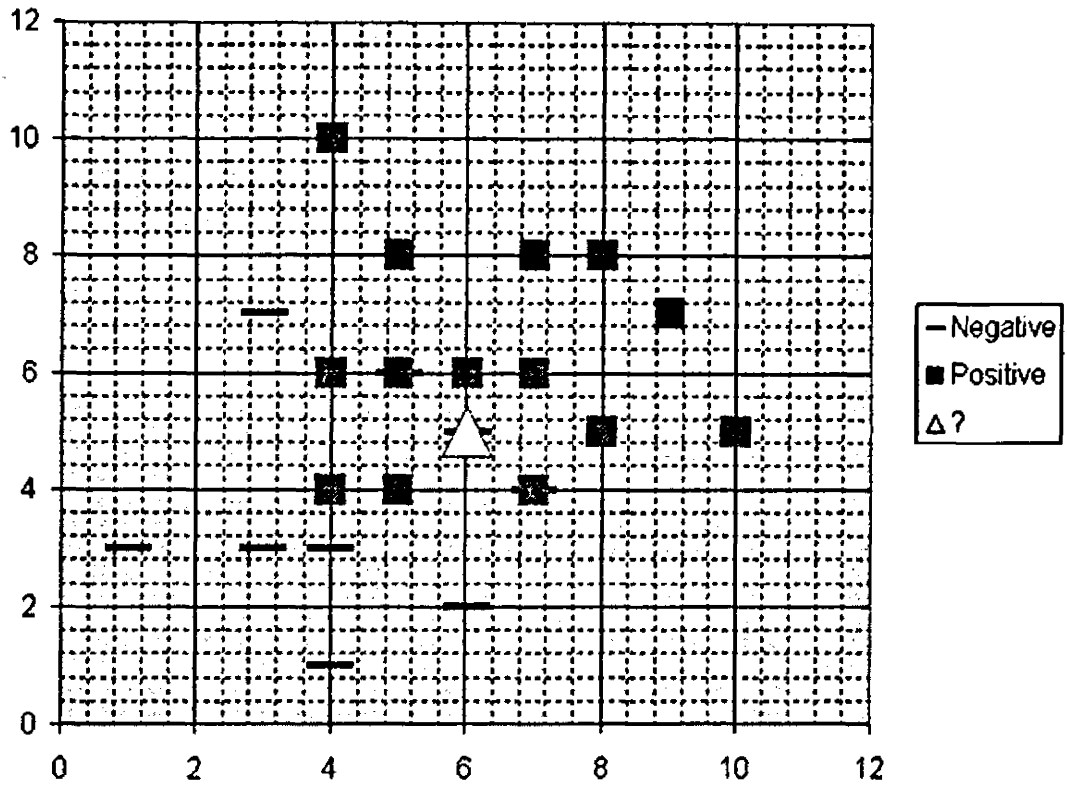


Figure 2. 3 RBF - K-Nearest Neighbor (k-NN)

Assume that each case in the training set has two predictor variables, x and y. The cases are plotted using their x,y coordinates as shown in the figure. Also assume that the target variable has two categories, positive which is denoted by a square and negative which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values x=6, y=5.1. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbor classification performed for this example depends on how many neighboring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

An RBF network positions one or more RBF neurons in the space described by the predictor variables (x,y in this example). This space has as many dimensions as there are predictor variables. The Euclidean distance is computed from the point being evaluated (e.g., the triangle in this figure) to the center of each neuron, and a radial basis function (RBF) (also called a kernel function) is applied to the distance to compute the weight (influence) for each neuron. The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (distance)

The further a neuron is from the point being evaluated, the less influence it has.

Radial Basis Function

Different types of radial basis functions could be used, but the most common is the Gaussian function:

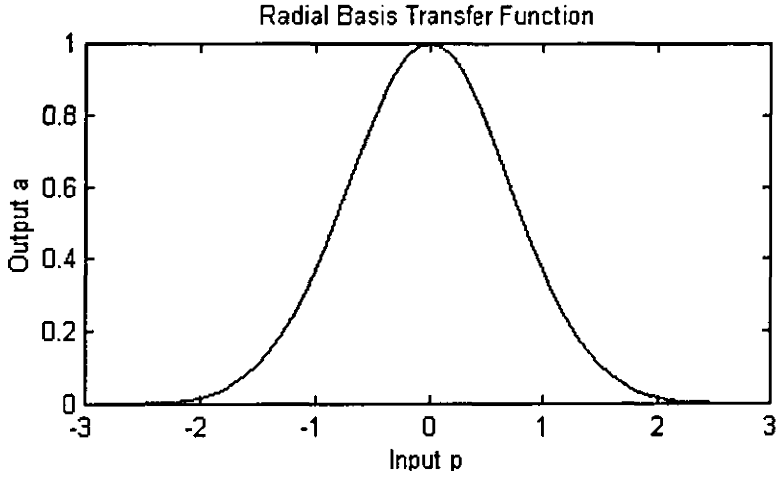


Figure 2.4 Radial Basis Transfer Function

If there is more than one predictor variable, then the RBF function has as many dimensions as there are variables. The following picture illustrates three neurons in a space with two predictor variables, X and Y. Z is the value coming out of the RBF functions:

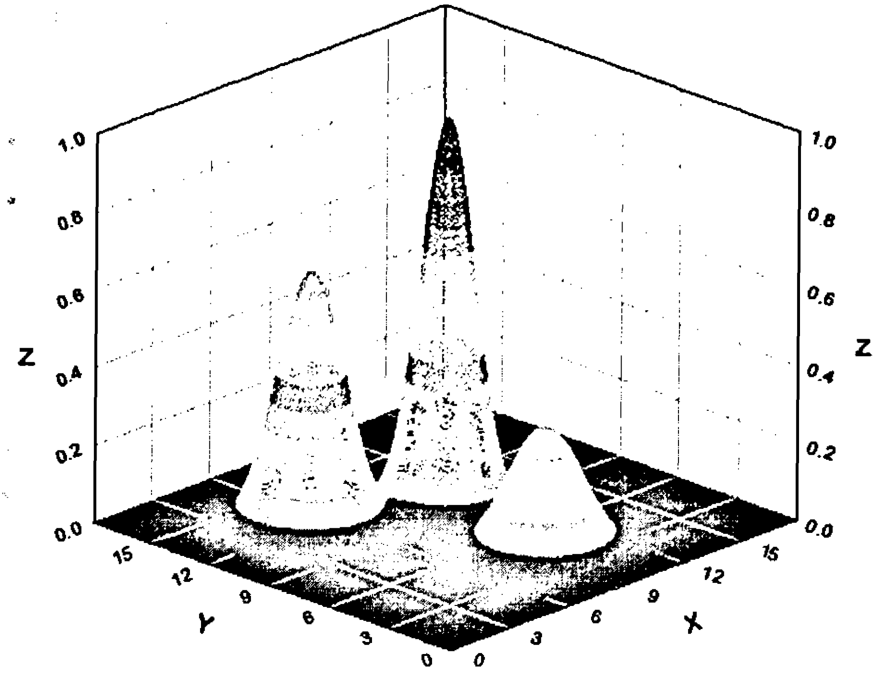


Figure 2.5 RBF function on three dimensions

The best predicted value for the new point is found by summing the output values of the RBF functions multiplied by weights computed for each neuron.

2.2.5 Probabilistic and General Regression Neural Networks

Probabilistic and General Regression Neural Networks have similar architectures, but there is a fundamental difference: Probabilistic networks perform classification where the target variable is categorical, whereas general regression neural networks perform regression where the target variable is continuous. If you select a PNN/GRNN network, DTREG will automatically select the correct type of network based on the type of target variable.

PNN and GRNN networks have advantages and disadvantages compared to multilayer perceptron networks:

- It is usually much faster to train a PNN/GRNN network than a MLP network.
- PNN/GRNN networks often are more accurate than MLP networks.
- PNN/GRNN networks are relatively insensitive to outliers (wild points).

- PNN networks generate accurate predicted target probability scores.
- PNN networks approach Bayes optimal classification.
- PNN/GRNN networks are slower than MLP networks at classifying new cases.
- PNN/GRNN networks require more memory space to store the model.
- PNN/GRNN networks are very similar to RBF networks. See page 253 for information about RBF networks.

How PNN/GRNN networks work

Although the implementation is very different, probabilistic neural networks are conceptually similar to K-Nearest Neighbor (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure:

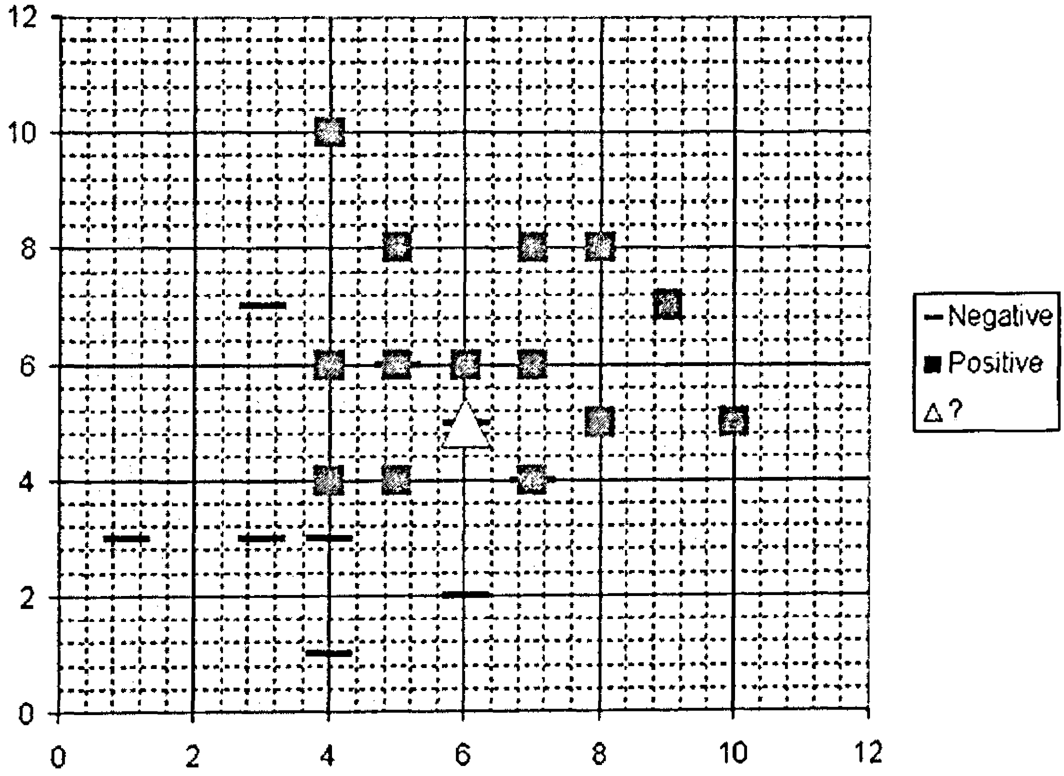


Figure 2. 6 PNN/GRNN - K-Nearest Neighbor (k-NN)

Assume that each case in the training set has two predictor variables, x and y. The cases are plotted using their x,y coordinates as shown in the figure. Also assume that the target variable has two categories, positive which is denoted by a square and negative which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values x=6, y=5.1. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbor classification performed for this example depends on how many neighboring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

A probabilistic neural network builds on this foundation and generalizes it to consider all of the other points. The distance is computed from the point being evaluated to each of the other points, and a radial basis function (RBF) (also called a kernel function) is applied to the distance to compute the weight (influence) for each point. The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (distance)

The further some other point is from the new point, the less influence it has.

Radial Basis Function

Different types of radial basis functions could be used, but the most common is the Gaussian function:

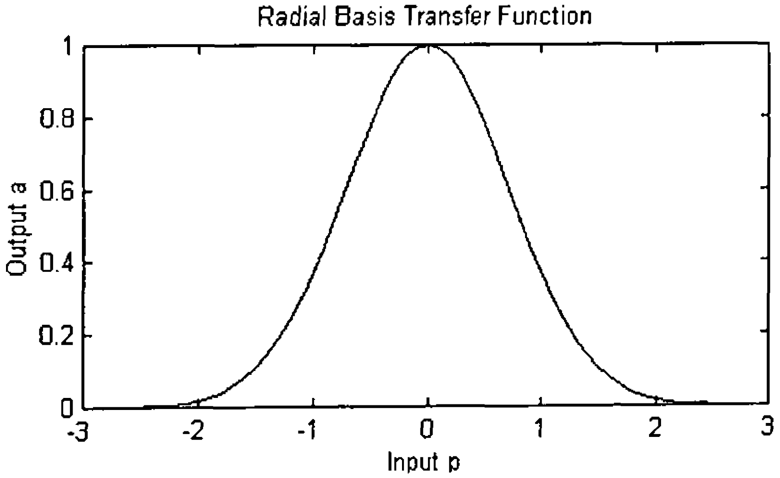


Figure 2.7 Radial Basis Transfer Function - PNN/GRNN

If there is more than one predictor variable, then the RBF function has as many dimensions as there are variables. Here is a RBF function for two variables:

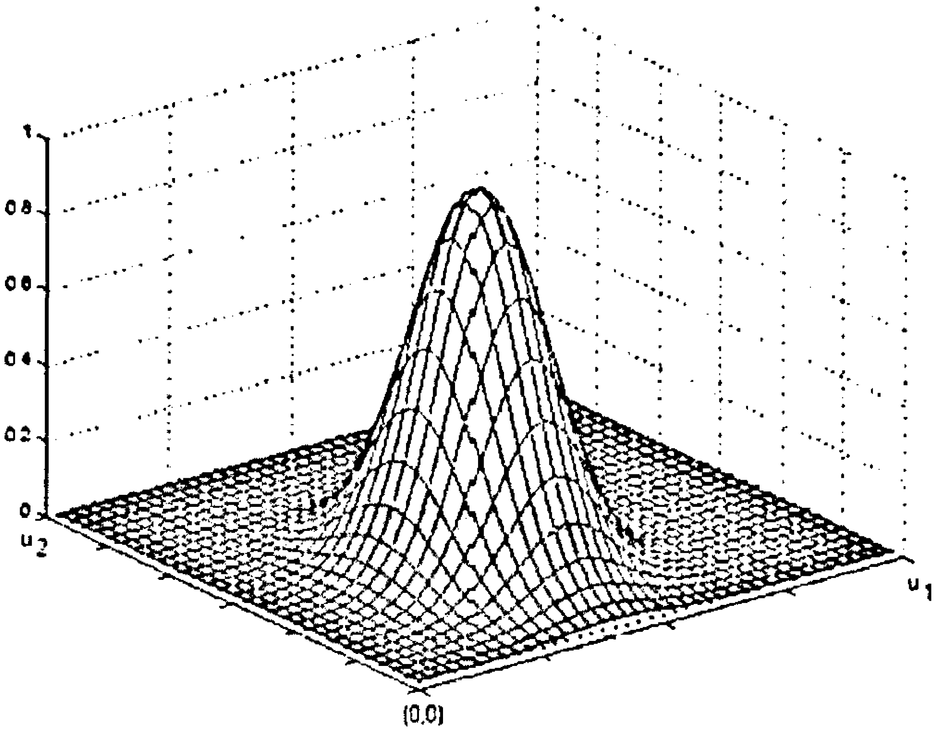


Figure 2.8 Radial Basis Transfer Function on Three Dimension

The best predicted value for the new point is found by summing the values of the other points weighted by the RBF function.

Architecture of a PNN/GRNN Network

In 1990, Donald F. Specht proposed a method to formulate the weighted-neighbor method described above in the form of a neural network. He called this a "Probabilistic Neural Network". Here is a diagram of a PNN/GRNN network:

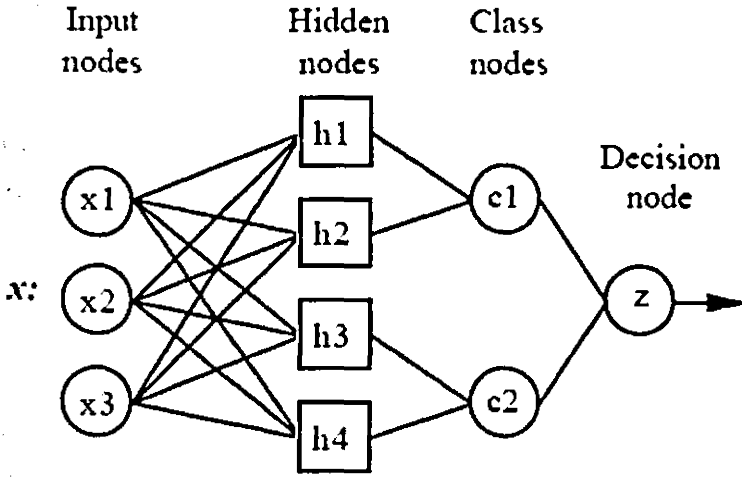


Figure 2.9 PNN/GRNN Network:

All PNN/GRNN networks have four layers:

Input layer – There is one neuron in the input layer for each predictor variable. In the case of categorical variables, N-1 neurons are used where N is the number of categories. The input neurons (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

Hidden layer – This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the x vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function using the sigma value(s). The resulting value is passed to the neurons in the pattern layer.

Pattern layer / Summation layer – The next layer in the network is different for PNN networks and for GRNN networks. For PNN networks there is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron's category. The pattern neurons add the values for the class they represent (hence, it is a weighted vote for that category).

For GRNN networks, there are only two neurons in the pattern layer. One neuron is the denominator summation unit the other is the numerator summation unit. The denominator summation unit adds up the weight values coming from each of the hidden neurons. The numerator summation unit adds up the weight values multiplied by the actual target value for each hidden neuron.

Decision layer – The decision layer is different for PNN and GRNN networks. For PNN networks, the decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

For GRNN networks, the decision layer divides the value accumulated in the numerator summation unit by the value in the denominator summation unit and uses the result as the predicted target value.

Removing unnecessary neurons

One of the disadvantages of PNN/GRNN models compared to multi-level feed forward networks is that PNN/GRNN models are large due to the fact that there is one neuron for each training row. This causes the model to run slower than multilayer perceptron networks when using scoring to predict values for new rows.

DTREG provides an option to cause it remove unnecessary neurons from the model after the model has been constructed. Removing unnecessary neurons has three benefits:

1. The size of the stored model is reduced.
2. The time required to apply the model during scoring is reduced.
3. Removing neurons often improves the accuracy of the model.

The process of removing unnecessary neurons is a slow (order N2), iterative process. Leave-one-out validation is used to measure the error of the model with each neuron removed. The neuron that causes the least increase in error (or possibly the largest reduction in error) is then removed from the model. The process is repeated with the remaining neurons until the stopping criterion is reached. For models with more than 1000 training rows, the neuron removal process may become impractically slow. If you have a multi-CPU computer, you can speed up the process by allowing DTREG to use multiple CPU's for the process. When unnecessary neurons are removed, the "Model Size section" of the analysis report shows how the error changes with different numbers of neurons. You can see a graphical chart of this by clicking Chart/Model size.

2.2.6 K-Means Clustering

Developed between 1975 and 1977 by J. A. Hartigan and M. A. Wong (Hartigan and Wong, 1979), K-Means clustering is one of the older predictive modeling methods. K-Means Clustering is a relatively fast modeling method, but it is also among the least accurate models that DTREG offers.

The basic idea of K-Means clustering is that clusters of items with the same target category are identified, and predictions for new data items are made by assuming they are of the same type as the nearest cluster center.

K-Means clustering is similar to two other more modern methods:

- **Radial Basis Function neural networks** (see page 2.2.4). An RBF network also identifies the centers of clusters, but RBF networks make predictions by considering the Gaussian-weighted distance to all other cluster centers rather than just the closest one.
- **Probabilistic Neural Networks** (see page 2.2.5). Each data point is treated as a separate cluster, and a prediction is made by computed the Gaussian-weighted distance to each point.

Usually, both RBF networks and PNN networks are more accurate than K-Means clustering models. PNN networks are among the most accurate of all methods, but they become impractically slow when there are more than about 10000 rows in the training data file. K-Means clustering is faster than RBF or PNN networks, and it can handle large training files. K-Means clustering can be used only for classification (i.e., with a categorical target variable), not for regression. The target variable may have two or more categories.

To understand K-Means clustering, consider a classification involving two target categories and two predictor variables. The following figure (Balakrishnama and Ganapathiraju) shows a plot of two categories of items. Category 1 points are marked by circles, and category 2 points are marked by asterisks. The approximate center of the category 1 point cluster is marked "C1", and the center of category 2 points is marked "C2".

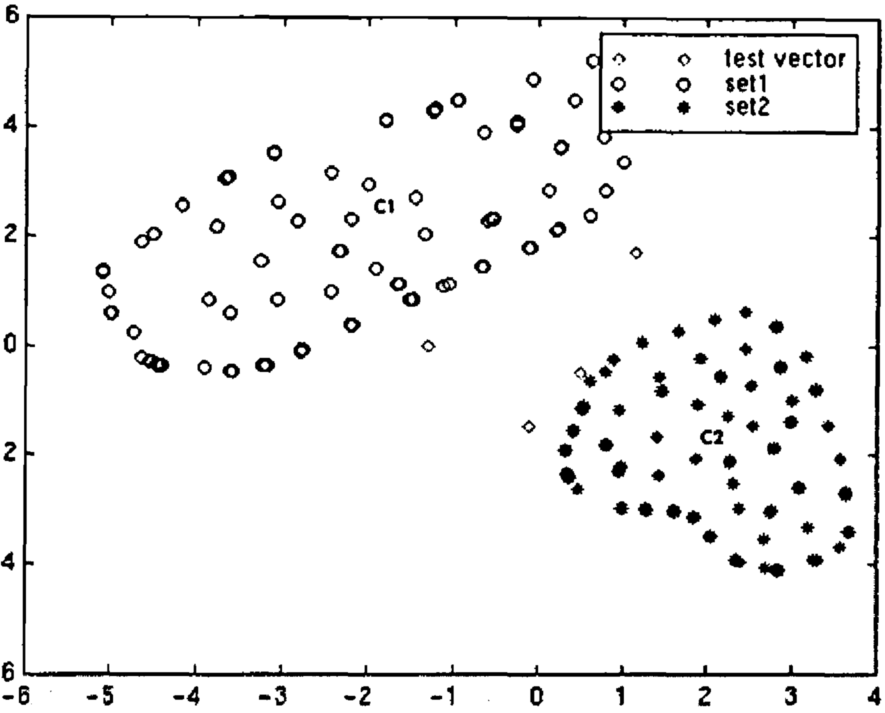


Figure 2.10 PNN/GRNN Network:

Four points with unknown categories are shown by diamonds. K-Means clustering predicts the categories for the unknown points by assigning them the category of the closest cluster center (C1 or C2). There are two issues in creating a K-Means clustering model:

1. Determine the optimal number of clusters to create.
2. Determine the center of each cluster.

Most K-Means clustering programs don't provide any systematic way to find out the optimal number of clusters, and it usually isn't as obvious as shown in the figure above. So the person trying to create a model must experiment and try guesses to see what works best. DTREG provides an automatic search function that creates models using a varying number of clusters tests each one and reports which is best. The model performance tests can be performed using cross-validation or holdout sampling. You can turn off the automatic search and specify a fixed number of clusters if you prefer.

Given the number of clusters, the second part of the problem is determining where to place the center of each cluster. Often, points are scattered and don't fall into easily recognizable groupings. Cluster center determination is done in two steps:

A. Determine starting positions for the clusters. This is performed in two steps:

1. Assign the first center to a random point.

2. Find the point furthest from any existing center and assign the next center to it. Repeat this until the specified number of cluster centers has been found.

B. Adjust the center positions until they are optimized. DTREG does this using a modified version of the Hartigan-Wong algorithm that is much more efficient than the original algorithm.

2.2.7 Logistic Regression

Logistic Regression is a type of predictive model that can be used when the target variable is a categorical variable with two categories – for example live/die, has disease/doesn't have disease, purchases product/doesn't purchase, wins race/doesn't win, etc. A logistic regression model does not involve decision trees and is more akin to nonlinear regression such as fitting a polynomial to a set of data values.

Logistic regression can be used only with two types of target variables:

1. A categorical target variable that has exactly two categories (i.e., a binary or dichotomous variable).

2. A continuous target variable that has values in the range 0.0 to 1.0 representing probability values or proportions.

As an example of logistic regression, consider a study whose goal is to model the response to a drug as a function of the dose of the drug administered. The target (dependent) variable, Response, has a value 1 if the patient is successfully treated by the drug and 0 if the treatment is not successful. Thus the general form of the model is:

Response = f(dose)

The input data for Response will have the value 1 if the drug is effective and 0 if the drug is not effective. The value of Response predicted by the model represents the probability of achieving an effective outcome, P(Response=1|Dose). Just as with all probability values, it is in the range 0.0 to 1.0. One obvious question is ―Why not simply use linear regression? In fact, many studies have done just that, but there are two significant problems:

1. There are no limits on the values predicted by a linear regression, so the predicted response might be less than 0 or greater than 1 – clearly nonsensical as a response probability.
2. The response usually is not a linear function of the dosage. If a minute amount of the drug is administered, no patients will respond. Doubling the dose to a larger but still minute amount will not yield any positive response. But as the dosage is increases a threshold will be reached where the drug begins to become effective. Incremental increases in the dosage above the threshold usually will elicit an increasingly positive effect. However, eventually a saturation level is reached, and beyond that point increasing the dosage does not increase the response.

The Dose-Response Curve

The logistic regression dose-response curve has an S (sigmoidal) shape such as shown here:

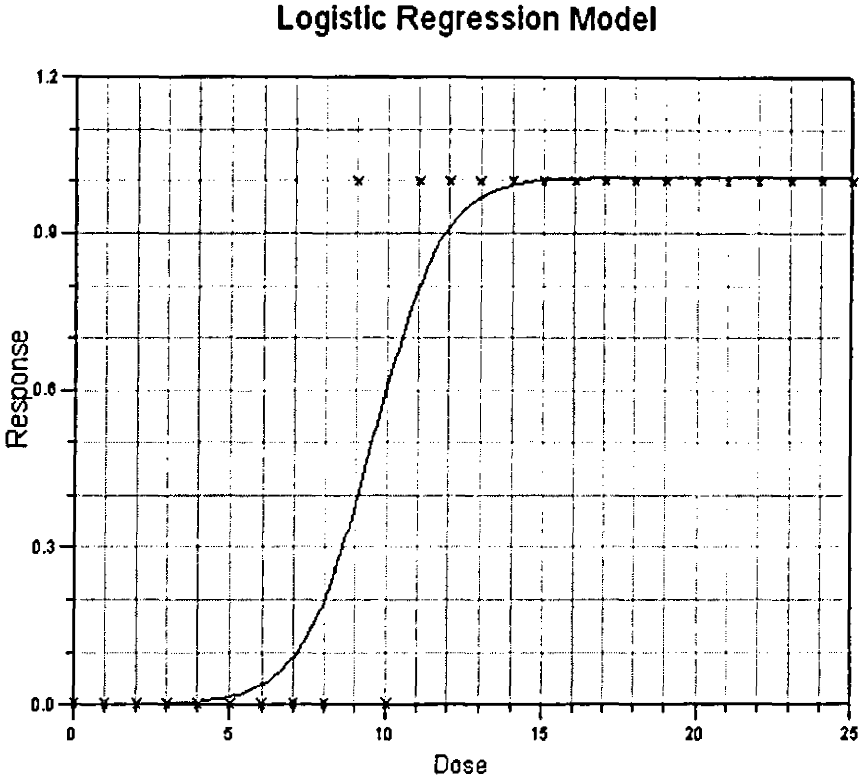


Figure 2. 11 Logistic Regression Model - Dose-Response Curve

Notice that all of the Response values are 0 or 1. The Dose varies from 0 to 25. Below a dose of 9 all of the Response values are 0. Above a dose of 10 all of the response values are 1.

The Logistic Model Formula

The logistic model formula computes the probability of the selected response as a function of the values of the predictor variables. If a predictor variable is categorical variable with two values, then one of the values is assigned the value 1 and the other is assigned the value 0. Note that DTREG allows you to use any value for categorical variables such as —Male and —Female, and it converts these symbolic names into 0/1 values. So you don't have to be concerned with recoding categorical values. If a predictor variable is a categorical variable with more than two categories, then a separate dummy variable is generated to represent each of the categories except for one which is excluded. The value of the dummy variable is 1 if the variable has that category, and the value is 0 if the variable has any other category;

hence, no more than one dummy variable will be 1. If the variable has the value of the excluded category, then all of the dummy variables generated for the variable are 0. DTREG automatically generates the dummy variables for categorical predictor variables; all you have to do is designate variables as being categorical. In summary, the logistic formula has each continuous predictor variable, each dichotomous predictor variable with a value of 0 or 1, and a dummy variable for every category of predictor variables with more than two categories less one category. The form of the logistic model formula is:

$P = 1/(1+exp(-(\beta_0 . \beta_1 X_1 . \beta_2 X_2 + ... + \beta_k X_k))))$

Where β0 is a constant and βi are coefficients of the predictor variables (or dummy variables in the case of multi-category predictor variables). The computed value, P, is a probability in the range 0 to 1. The exp() function is e raised to a power. You can exclude- the β0 constant by turning off the option "Include constant (intercept) term" on the logistic regression model property page.

2.2.8 Knowledge discovery and Data Mining

Knowledge discovery in databases (KDD) as defined by Fayyad et al, (1996) is the process of identifying valid, novel, potentially useful, and ultimately understandable patterns or models in data. **Data mining (DM)** is a step in the knowledge discovery process consisting of particular data mining algorithms that, under some acceptable computational efficiency limitations, find patterns or models in data. This is most cited among numerous definitions of data mining and knowledge discovery. However, for many, data mining is a synonym for knowledge discovery.

The data mining methodology will be based on ***CRISP-DM*** (*CR*oss *I*ndustry *S*tandard *P*rocess for *D*ata *M*ining) which is a methodology based on practical, real-world experience which was defined, by the consortium of companies which applied data mining from the days of its infancy as defined in an article by LIS - Rudjer Boskovic Institute (2001) . It entails the following steps:

Problem understanding

i) Data understanding

i) Data preparation

i) Modeling

i) Evaluation of results

i) Deployment of results

2.3 Decision Support Systems

2.3.1 Introduction

In the 1980s and 1990s, a new concept of information systems has evolved because managers increasingly need information to make decisions about how to organize and control resources effectively [Robert S. et al., 1998]. These systems are known as decision support systems. They are quite different from the information systems of the past. The decision making process has three major phases namely, the intelligence, design and choice. In the intelligence phase, the decision-maker searches for conditions calling for a decision. The decision-maker may be reacting to problems or may recognize opportunities. In the design phase, the decision-maker develops and analyses alternative courses of action by either searching for ready-made alternatives or developing custom-made solution. Lastly the decision-maker selects the best alternative in the choice phase.

There are two different types of decision problems. Problems are either structured or semi-structured depending on how familiar the decision-maker is with the existing state, desired state and transformation necessary to get from one state to another. This study will consider the semi-structured approach. In organizations, managerial decision problems are semi-structured because the decision environment is uncertain, complex and unstable.

Decision Support Systems are designed to support semi-structured decisions in situations in which the information is incomplete and where satisificing is a goal [Turbman E., 2001].

2.3.2 Characteristics of Decision Support Systems

Decision Support Systems are computer-based systems, which attempt to solve semi and non-structured problems by combining data and models. They make use of flexible user interfaces. The systems have become very necessary because of increase in complexity of business which has resulted in monitoring operations, changing objectives related to efficiency, profitability and markets, instability in economies and increase in domestic and foreign competition has also made the use of these systems very necessary. These systems provide a competitive edge over competitors who are not using the technology by assisting managers in precision decision processes. The DSS also improve effectiveness and efficiency in their work. They solve complex problems and allow quick responses to unexpected decisions through their analysis tools and can be used to evaluate outcomes of strategies under different conditions and configurations.

Figure 2.11 shows the components of a Decision Support System. It provides a basic understanding of the general structure of a DSS. The economic models will be formulated based on the data from MIS. The management information system (MIS) data will be generated from within the organization systems. The transaction processing system (TPS) data will be transactions that are generated within the organization. They include transactions such as the loan repayments entries.

The knowledge management subsystem (KMS) can support any other sub-systems or act as an independent component. It provides intelligence to augment the decision maker's own.

2.3.3 Examples of DSS Solutions and their Utilization

Leong, Jenney (2005) in their Masters thesis, University of Teknologi Malaysia, Faculty of Computer Science and Information System developed a prototype Home Loan Packages Selection Decision Support System Using Financial Model (HSDSS). It is a decision support system that uses mathematical model to allow user to explore the impact of available options. The optimal solution is obtained by using blind search with complete enumeration to check all the alternatives. This searching approach works together with weighted point system, so that the alternatives will have their weight of points after the searching is done. Based on the result of the ranking of the alternatives, HSDSS provides advices to the homebuyers on the matter of selecting suitable home loan packages. It is meant to speed up and simplify how homebuyers make decision in choosing home loan packages, in addition to improving the competitive advantage for real estate service providers. As a conclusion, this system is capable in solving the current problems associated with choosing best suit home loan packages.

Martin Blunn(2005) developed a decision support prototype to assist archeologists with Non-Laboratory Soil Analysis Techniques. Although the prototype constructed was incomplete, by focusing on developing key aspects of the implementation technology such as HTML web page generation, Java Servlets and XML file manipulation, the author was able to 'prove the concept' for the project and make substantial recommendations for its future completion.

"Promedas" is a prototype of a diagnostic Decision Support System based on a large causal probabilistic network, using recently developed computational techniques. It was developed at Foundation for Neural Networks Nijmegen, The Netherlands at the University Medical Centre Utrecht, The Netherlands and is based on a large causal probabilistic network. It uses recently developed computational techniques. It is meant to improve the quality and it is meant to improve efficiency of health care, while reducing its costs at the same time. The system intends to support diagnosis making in the setting of the outpatient clinic and for educational purposes. Its target-users are general internists, super specialists (i.e. cardiologists, rheumatologists), interns and residents, medical students and others working in the hospital environment. The system offers diagnostic advice. In active decision mode, it supports the diagnostic process by indicating the most useful next step in the diagnostic process.

Automation and Data Systems Division of Southwest Research Institute SwRI engineers have developed a novel medical software algorithm capable of automating the interpretation, simplification and noise reduction of diagnostic data for the clinician, thus requiring minimal human intervention to produce an accurate differential diagnosis. The software prototype serves as a testing and demonstration vehicle for the new software technology. The prototype is a rich, visual DSS developed using the Java programming language, R statistical package, MySQL database management system, and Swing graphical toolkit.

The diagnostic knowledge base for the software prototype was constructed using the Centers for Disease Control (CDC) National Ambulatory Medical Care Survey (NAMCS) dataset and relational database technology. The three major knowledge "focus" areas are:

- Diagnostic Data – Diagnoses, causes and cases (i.e., diagnostic records)

- Empirical Data – Model variables, principal components and regression equations

Monika Kastner, Jamy Li, Danielle Lottridge, Christine Marquez, David Newton and Sharon E Straus (2010) developed a functional prototype that may aid physicians in their clinical decision making in osteoporosis disease management at the point of care. The prototype incorporates all aspects of disease management (risk assessment, diagnosis, and treatment), and is multi-targeted to deliver clinical decision support for physicians and education for patients about osteoporosis.

Tony Austin, Steve Iliffe, Mark Leaning and Mike Modell (1996) developed a prototype for asthma management targeted at the primary care setting and based on the British Thoracic Society Guidelines. Two July 2005 press releases described decision support applications at Airbus and Hellmann Worldwide Logistics. Airbus expanded its use of an Applix TM1 solution (a data driven DSS solution) to approximately 100 controllers at 16 Engineering Competency Centers located in France, Germany, the UK, and Spain. Hellmann is an air and sea freight shipping company that serves customers from 341 cities in 134 countries. Hellmann selected BusinessObjects XI to provide real-time access to customer related information such as tracking and tracing statuses, invoicing, inventory, and KPI management. Buckman Labs standardized on Information Builders' software (a data driven DSS software) for global information integration (01/30/2006). It uses WebFOCUS to generate sales analysis reports. The company has annual sales of $429 million, produces 700 different products, and employs over 1,500 people working in more than 90 countries.

ABN AMRO selected Teradata Data Warehouse to build a platform for business decision support in Asia (02/15/2006). ABN AMRO is an international bank with more than 3,000 branches in more than 60 countries and territories. The data warehouse will support business development of ABN AMRO consumer businesses in Asia. Regional headquarters in Hong Kong will be able to view the region's total business as well as the performance of each individual country's business, and each country will have a view of its own data. The focus is on DSS for customer relationship management, customer revenue analysis, and monitoring credit risk metrics.

On March 21, 2006 Cognos announced Fresh Del Monte purchased the Cognos Performance Management solution (a data driven DSS). Fresh Del Monte is a leading global producer and distributor of fruit and vegetable products in Europe, Africa and the Middle East.

Obren Makov (2009) developed the Baby Gender Calculator. This is software that is used to determine periods with increased probability of conceiving a baby boy or a baby girl for a particular parent couple. Basis for this calculation are the birth date data of both parents. Baby Gender Calculator Program is very intuitive and simple to use, and its efficiency is easy to check using data on born children and their parents.

In the literature review, the study was not able to establish any literature that focused on decision support systems in Deposit insurance schemes.

Types of Decision Support Systems

There are a number of Decision Support Systems. These can be categorized into five types:

Communication-driven DSS

Most communications-driven DSSs are targeted at internal teams, including partners. Its purpose are to help conduct a meeting, or for users to collaborate. The most common technology used to deploy the DSS is a web or client server. Examples: chats and instant messaging software, online collaboration and net-meeting systems.

Data-driven DSS

Most data-driven Doss are targeted at managers, staff and also product/service suppliers. It is used to query a database or data warehouse to seek specific answers for specific purposes. It is deployed via a main frame system, client/server link, or via the web. Examples: computer-based databases that have a query system to check (including the incorporation of data to add value to existing databases.

Document-driven DSS

Document-driven DSSs are more common, targeted at a broad base of user groups. The purpose of such a DSS is to search web pages and find documents on a specific set of keywords or search terms. The usual technology used to set up such DSSs is via the web or a client/server system.

Knowledge-driven DSS:

Knowledge-driven DSSs or 'knowledgebase' are they are known, are a catch-all category covering a broad range of systems covering users within the organization setting it up, but may also include others interacting with the organization - for example, consumers of a business. It is essentially used to provide management advice or to choose products/services. The typical deployment technology used to set up such systems could be salient/server systems, the web, or software running on stand-alone PCs.

Model-driven DSS

Model-driven DSSs are complex systems that help analyse decisions or choose between different options. These are used by managers and staff members of a business, or people who interact with the organization, for a number of purposes depending on how the model is set up - scheduling, decision analyses etc. These DSSs can be deployed via software/hardware in stand-alone PCs, client/server systems, or the web.

This project focuses on data driven decision support.

2.3.5 Architecture of decision support systems

As shown in Figure 1.1, there are three fundamental components of DSSs:

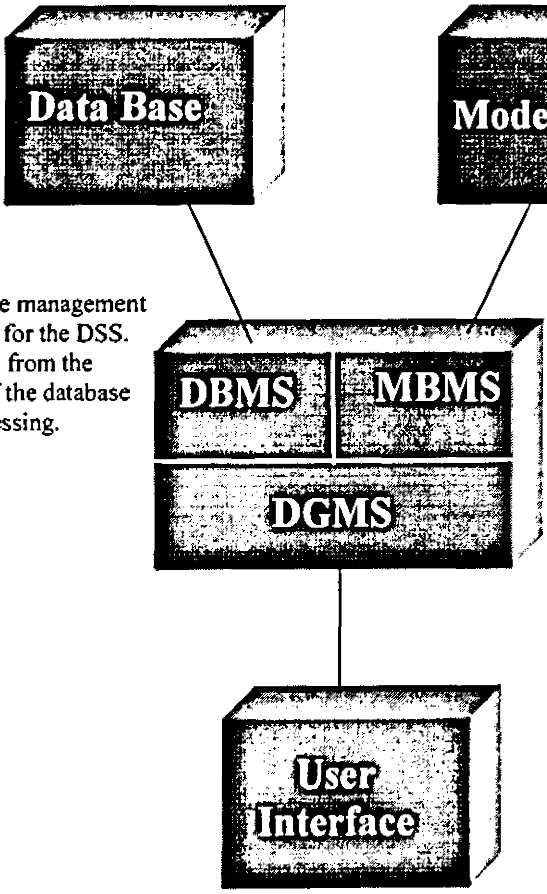
- Data base management system (DBMS). A DBMS serves as a data bank for the DSS. It stores large quantities of data that are relevant to the class of problems for which the DSS has been designed and provides logical data structures (as opposed to the physical data structures) with which the users interact. A DBMS separates the users from the physical aspects of the database structure and processing. It should also be capable of informing the user of the types of data that are available and how to gain access to them.
- Model - based management system (MBMS). The role of MBMS is analogous to that of a DBMS. Its primary function is providing independence between specific models that are used in a DSS from the applications that use them. The purpose of an MBMS is to transform data from the DBMS into information that is useful in decision making. Since many problems that the user of a DSS will cope with may be unstructured, the MBMS should also be capable of assisting the user in model building.
- Dialog generation and management system (DGMS). The main product of an interaction with a DSS is insight. As their users are often managers who are not computer-trained, DSSs need to be equipped with intuitive and easy-to-use interfaces. These interfaces aid in model. The primary responsibility of a DGMS is to enhance the ability of the system user to utilize and benefit from the DSS. The broader term of DGMS is user interface.

While a variety of DSSs exists, the above three components can be found in many DSS architectures and play a prominent role in their structure. Interaction among them is illustrated in Fig. 1.

Essentially, the user interacts with the DSS through the DGMS. This communicates with the DBMS and MBMS, which screen the user and the user interface from the physical details of the model base and database implementation.

General Components of a DSS

(1982)



DBMS – Data Base management system - data bank for the DSS. Separates the users from the physical aspects of the database structure and processing.

MBMS - Model based management system - Its purpose is to transform data from the DBMS into information that is useful in decision making.

DGMS - Dialog generation and management system. Enhance the ability of the system user to utilize and benefit from the DSS. Essentially user interface

Figure 2.12 Components of decision support system. Source: Sprague and Carlson

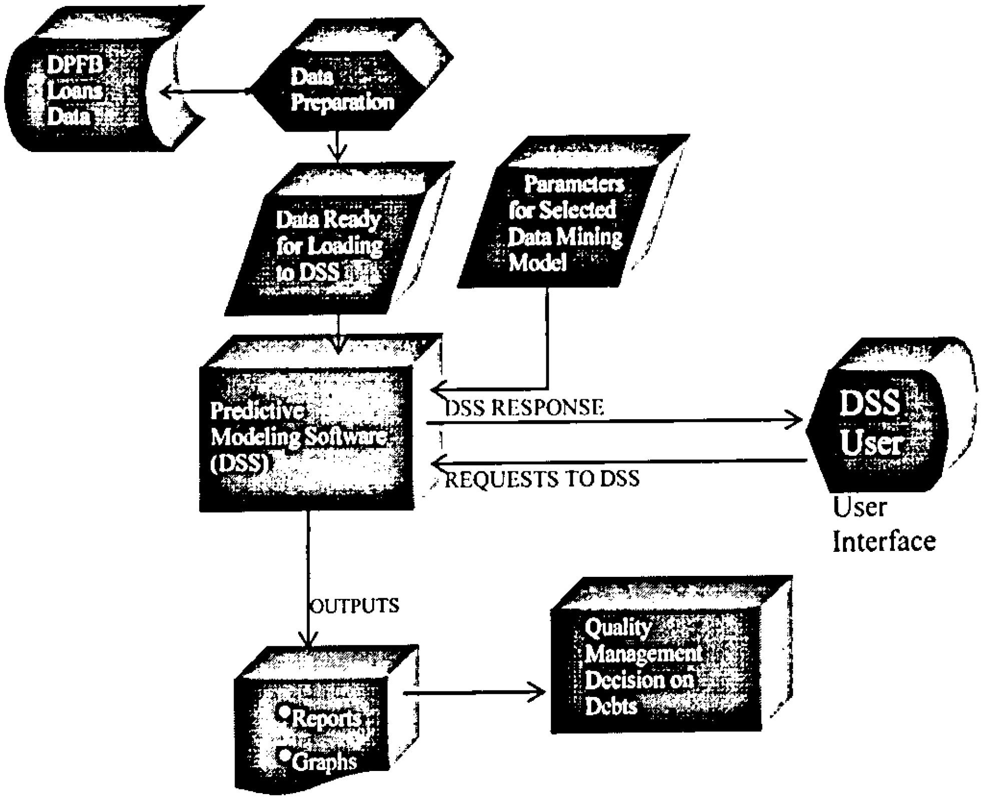

DPFB Loans Data
Data Preparation
Data Ready for Loading to DSS
Paramcters for Selected Data Mining Model
Predictive Modeling Software (DSS)
DSS RESPONSE
REQUESTS TO DSS
DSS User
User Interface
OUTPUTS
Reports
Graphs
Quality Management Decision on Debts

Figure 2.13Components of the Expected Prototype

CHAPTER THREE
RESEARCH AND METHODOLOGY

3.1 Introduction

This chapter will focus on the methods and procedures used in realizing the objectives of the study. A research design is used in guiding the researcher on getting solutions to the research questions. The main contribution of a research design includes that of helping the researcher collect, analyze and interpret the collected data. This chapter includes the research design, location of the study, target population, data mining methodology, study variables and the data analysis and validation techniques.

3.2 The Research Design

3.2.1 Project Components

The study comprised of survey, data collection, data preparation, training and validation of various data mining tools, selection of a suitable tool, design and development of a prototype using the selected tool.

3.2.2 Data Gathering and Preparations

The study involved established the IT facets in place meant to enhance debt recovery in institutions in liquidation. It assessed the level of success in the debt recovery process among the institutions in liquidation by sampling data and analyzing it. There verbal interviews involved questioning selected DPFB liquidation division staff. Observation and document reviews were also used for information gathering. The debt recovery data was solicited from the loans data belonging to 27 institutions in DPFB liquidation division. This data was then transformed ready for use in a DSS by identifying the target variable and predictor variables. Any records that lacked any necessary variables were left out. Variables that were not common in most of the data were also discarded.

3.2.3 Data Mining Tool Classifier (DTREG)

A suitable data mining classification tool was identified. The selected software was DTREG which is software For Predictive Modeling and Forecasting. DTREG accepts a dataset containing of number of rows with a column for each variable. One of the variables is the "target variable" whose value is to be modeled and predicted as a function of the "predictor variables". DTREG analyzes the data and generates a model showing how best to predict the values of the target variable based on values of the predictor variables. It generates reports showing the accuracy of the tools in the classifications. DTREG was used to train, validate and generate accuracy reports for various data mining tools. Cross validation method was used in the training and validation. Confusion matrix tool was used to tabulate results and the accuracy of each data mining tool was calculated. The Area Under ROC Curve (AUC) was also used as a further measure of accuracy. Based on the results an appropriate tool was selected and used to develop a prototype that can be used by DPFB in making decisions on debt recovery.

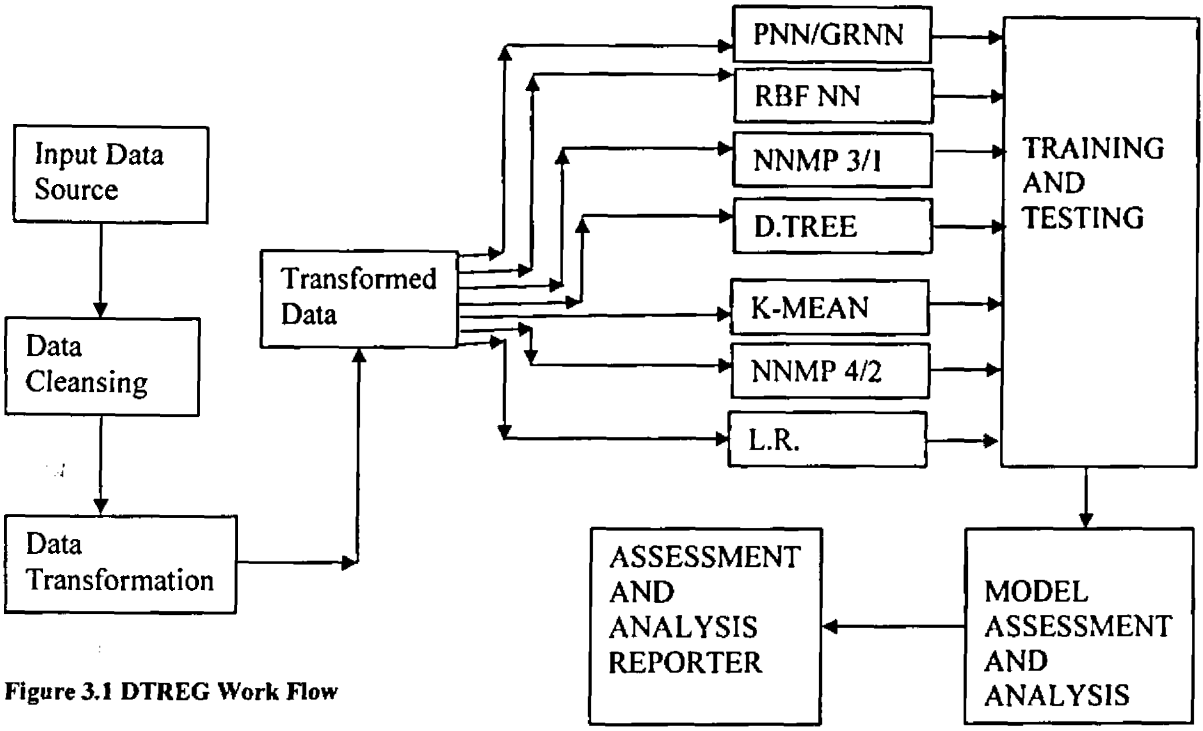


Figure 3.1 DTREG Work Flow

3.3 Location of the study

The study was conducted in the Liquidation division of the DPFB department, Central Bank of Kenya. The selection of the division was done purposively from the three divisions of the department namely; Finance & Administration, Legal and Liquidation. Liquidation division was selected since it handles the liquidation process of the institutions in liquidation. It is also well known to the researcher and was therefore easier to locate the relevant data and information. In addition he has been interacting regularly with the users in this division during the continuous maintenance of their liquidation system. The researcher is also conversant with some of the challenges encountered in the liquidation process. The researcher had the task finding out the measures in place to enhance debt recovery, the level of success achieved in the debt recovery process, appreciating the problems encountered, data preparation, selection of seven data mining tools, adapting a suitable data mining classification software, running the evaluations using the tools, analyzing the results, identified the accurate data mining tool and eventual prototype development.

3.4 Target population

According Brinker (1988) target population is defined as the whole all large population from which a sample is to be selected. The target population for this study consisted of employees in the in liquidation division of the DPFB department in Central Bank of Kenya. It targeted twelve employees comprising of two staff from each section within liquidation division. It also included the head of the liquidation division. The targeted data was the debt recovery records maintained in the loans master files of the institutions in liquidation. Data was collected from the 27 institution in liquidation in DPFB. The data size used was over 2,500 loans records. This quantity was dictated by the demo version of the DTREG otherwise there were over 18,000 loans records available in the target. The institutions use a liquidation system that has a loans maintenance module. The debt recovery data was extracted from the loan master SQL tables in this module.

3.5 Data Mining Methodology

The data mining methodology was based on CRISP-DM (CRoss Industry Standard Process for Data Mining) which is a methodology based on practical, real-world experience which was defined, by the consortium of companies which applied data mining from the days of its infancy as defined in an article by LIS - Rudjer Boskovic Institute (2001) . It entails the following steps:

- Problem understanding
- Data understanding
- Data preparation
- Modeling – training and testing
- Evaluation of results
- Prototype development
- Deployment of results

3.6 Study variables

The variables identified in this project were identified as:

Predictor Variables

i) Debt amount at liquidation.

ii) Customer contacts availability (either available (value 1) or not available (value 0).

iii) Type of loan (normal loan (value 1) or others (value 0).

iv) Customer type (staff (value 0)or non staff (value 1)

v) Initial Loan amount.

Target Variable

The target variable (variable to be predicted) was the prediction whether the debt is recoverable or not. For the training data which must have the target variable, an account with a balance of zero was classified as good (value 1). An account with a balance was classified a bad (value 0).

3.7 Data Analysis and Validation Techniques

3.7.1. Simple validation

The most basic testing method is called simple validation. This is done by setting aside a percentage of the database as a test database, and do not use it in any way in the model building and estimation. This percentage is typically between 5% and 33%. For all the future calculations to be correct, the division of the data into two groups must be random, so that the training and test data sets both reflect the data being modeled. After building the model on the main body of the data, the model is used to predict the classes or values of the test database. Dividing the number of incorrect classifications by the total number of instances gives an error rate. Dividing the number of correct classifications by the total number of instances gives an accuracy rate (i.e., accuracy = 1 – error). For a regression model, the goodness of fit or "r-squared" is usually used as an estimate of the accuracy. In building a single model, even this simple validation may need to be performed dozens of times.

For example, when using a neural net, sometimes each training pass through the net is tested against a test database. Training then stops when the accuracy rates on the test database no longer improve with additional iterations.

3.7.2 Cross validation

If you have only a modest amount of data (a few thousand rows) for building the model, you can't afford to set aside a percentage of it for simple validation. Cross validation is a method that lets you use all your data. The data is randomly divided into two equal sets in order to estimate the predictive accuracy of the model. First, a model is built on the first set and used to predict the outcomes in the second set and calculate an error rate. Then a model is built on the second set and used to predict the outcomes in the first set and again calculate an error rate.

Finally, a model is built using all the data. There are now two independent error estimates which can be averaged to give a better estimate of the true accuracy of the model built on all the data. Two Crows Corporation (1999).

This project mainly focused on the above type of cross validation.

Typically, the more general n-fold cross validation is used. In this method, the data is randomly divided into n disjoint groups. For example, suppose the data is divided into ten groups. The first group is set aside for testing and the other nine are lumped together for model building. The model built on the 90% group is then used to predict the group that was set aside. This process is repeated a total of 10 times as each group in turn is set aside, the model is built on the remaining 90% of the data, and then that model is used to predict the set-aside group. Finally, a model is built using all the data. The mean of the 10 independent error rate predictions is used as the error rate for this last model. This project focused on cross validation at the training, testing and validation stage.

3.7.3. Confusion Matrix

The classification results can often be summarized in a theoretical confusion matrix presented in Table 2 (Giudici, 2003; Kantardzic, 2003). This is the table that was used to tabulate results for the seven selected data mining tools.

Table 3.1 Theoretical Confusion Matrix

Actual Category	--------Predicted Category----------		
	Event (1)	Non-event(0)	Total
Event(1)	a	b	a+b
Non-event(0)	c	d	c+d
Total	a+c	b+d	a+b+c+d

The table classifies the observations on a test data set into four possible categories:
Cases predicted as events and effectively such (with absolute frequency equal to a)
Cases predicted as non-events and effectively events (with absolute frequency equal to b)
Cases predicted as events and effectively non-events (with absolute frequency equal to c)
Cases predicted as non-events and effectively such (with absolute frequency equal to d)

The computation of error rate is based on counting of errors in a testing process. These errors are defined as misclassification (wrongly classified examples). In a binary classification problem, where the target variable has only two classes: True (Event or 1) and False (Non-event or 0), two different related error rates are of interest. These are *false negative errors [it is expected to be T (Event), but it is classified* as F (Non-event) and false positive errors [it is expected to be F (Non-event), but it is classified as T (event)]. In Table 2, their absolute frequencies are denoted by b and c, respectively. Assuming that all errors are of equal importance, and error rate R is the number of errors E divided by the number of samples S in the testing set: (R=E/S=(b+c)/(a+b+c+d)). The accuracy of a model is a part of the testing data set that is classified correctly, and it is computed as one minus the error rate: A=1-R=(S-E)/S=(a+d)/(a+b+c+d). In many applications, it is not adequate to characterize the performance of a model by four single numbers a, b, c, and d that measure the correct classification and error rates. This project used confusion matrix for the analysis of the models.

3.7.4 Area under ROC curve

More accurate, complex and global measures are necessary to describe the quality of the model. These measures include receiver operating characteristic (ROC) charts (Giudici, 2003). These charts were used to assess more thoroughly the classification performance of the developed models. The ROC chart is of special interest because it allows one to analyze both false negative and false positive errors at the same time for different cut-off points. The chart measures the predictive accuracy of a model. It is based on the confusion matrix in Table 2. More precisely, the ROC chart is based on the following conditional probabilities:

Sensitivity a/(a+b) is the proportion of events predicted as such.

Specificity d/(c+d) is the proportion of non-events predicted as such.

False positives c/(c+d)=1-specificity is the proportion of non-events predicted as events (type II error). For example, in the bad debt recovery case, these are the test cases in which customers are wrongly predicted as payers.

False negatives b/(a+b)=1-sensitivity is the proportion of events predicted as non-events (type I error). In the bad debt recovery example, these are the cases of debts that are wrongly classified as bad debts.

The curve is obtained by graphing, for any fixed cut-off value, the sensitivity on the vertical axis and the false positives (1-specificity) on the horizontal axis. Each point on the curve corresponds to a particular cut-off. The ROC curve can also be used to select cut-off point, trading off sensitivity and specificity. A classification model can be tuned by setting an appropriate threshold value to operate at a desired value of the false positive rate. If one tries to decrease the false positive rate parameter of the model, however, it would increase the false negative rate and vice versa. In terms of model comparison, the ideal curve coincides with the vertical axis, so the best curve is the leftmost curve. The curve will always lie above the 45 line. The curve permits one to assess the performance of the model at various operating points (thresholds in a decision process using the available model) and the performance of the model as a whole (using as a parameter the area below the ROC curve). Furthermore, the area between the curve and the 45 line can also be calculated, and gives the Gini index of performance. The higher the area is, the better the model. The ROC curve is especially useful for a comparison of the performances of several models obtained using different data-mining methodologies. For more details, see (Giudici, 2003; Kantardzic, 2003).

3.7.5. Prototype Design and Development

This project had a design and development of a decision support prototype phase. The prototype has an interface for user input presented appropriate results to the user.

The architecture followed is as stipulated in the literature review under the sub head 2.3.5.

CHAPTER FOUR

ANALYSIS OF RESULTS AND DISCUSSIONS

4.1 Project Data

4.1.0 Data Collection

Data was collected from the 27 institutions in liquidation division of DPFB department Central Bank Kenya. The data source was from the loan tables in each of the institution's database. The data was scrutinized to ensure selection of viable data that had all the required variables. The data cleaning exercise involved removal of records with null value in required variables. Variables that were missing in most of the data were also discarded.

Decisions on Bad/Good Debts in DPFB

Currently DPFB does not have a Decision Support System to aid in classifying good and bad debts. They base their classification of good and bad debts on:

- Customer's response to demand notes:
  - No response signals a potential bad debt.
  - Response signals potential good debt.
- Study on loans documentations to ascertain validity and reliability:
  - Poor documentations signals potential bad debt.
  - Proper documentation signals potential good debt.
- Availability of security:
  - No security signals a potential bad debt.
  - Available security signals a potential good debt.

IT Measures on Debt Recovery Process

DPFB has an in house developed liquidation system:

- Within the system is a loans module that assists in:
  - Maintenance of loan information:
    - Takeover balances.

    - Repayments transactions.
    - Interest calculation when applicable.
    - Loan master details.
    - Accounts history.

- They have Registry system that assists in tracking loans file movement.
- They have a Court Case module that is meant to assist in tracking loan cases in court.
- In a recent development DPFB uses the liquidation system to produce data that is forwarded to the Credit Reference Bureau (CRB) for credit information sharing. The CRB component can stimulate and enhance the debtors' loan repayments due to the implications of having negative information at the CRB which may hinder any further access to credit.

Success Achieved so Far

From DPFB loans data the following was established: Taking sample of seven institutions and looking at their loans data, out of 6,356 loans only 1,531 have been recovered/ compromised. This is 24% success. This is fairly low success as this included the negotiated and compromised debts.

4.1.1 Data Preparation

The data preparation stage involved the following:

- Data selection: This was done to ensure that only data with all the required variables was used.
- Data cleaning: This was meant to discard any data with null values in required variables.
- Data transformation: This to prepare the data to suit the model as explained in the next bullet.
- SQL 2000 was used to prepare the data in the following manner:
    - A new table was created and data from the loans table was uploaded into it.
    - Account name was removed from the table for confidentiality purposes.
    - For loan type '201' which is the code used for normal loan, this was replaced with a '1'. Any other loan type was replaced with a '0'.
    - For contacts, all records with contacts were updated with a '1' while those without were given '0'.

- The liquidation amount was converted to its absolute value. This is because the data is stored as a negative value.
- Customer type non-staff were replaced with a '1' while customer type staff was replaced with a '0'.
- Any account with 0 balance was classified as good by giving it the value '1' while those with a balance were classified a bad by giving a value of '0'.

Eventually 2,588 loans records were completed ready for the training, validation and testing. There were over 18,000 records in the target sample but could not all be considered due to limitations in the training and testing software.

4.1.2 Selected Data Mining Tools

On the basis of the literature review, the following data mining tools were identified for the study:

1. Decision Tree
2. NN MP with 4 layers (2 hidden)
3. NN MP with 3 layers (1 hidden)
4. RBF NN
5. PNN/GRNN
6. Logistic Regression
7. K-Mean Clustering

4.2 Training and Validation

4.2.1 Cross Validation

The validation method used was cross validation. The data was randomly subdivided into two equal sets in order to estimate the predictive accuracy of the models. First, a model was built on the first set and used to predict the outcomes in the second set and calculate an error rate. A model was then built on the second set and used to predict the outcomes in the first set and again calculate an error rate.

Finally, a model was built using all the data. There were now two independent error estimates which can be averaged to give a better estimate of the true accuracy of the model built on all the data. Two Crows Corporation (1999).

4.2.2 Confusion Matrix

For classification problems, a confusion matrix is a very useful tool for understanding results. A confusion matrix (Table 2 section 3.14) shows the counts of the actual versus predicted class values. It shows not only how well the model predicts, but also presents the details needed to see exactly where things may have gone wrong. The columns show the predicted classes, and the rows show the actual classes. Therefore the diagonal shows all the correct predictions. The details on the confusion matrix are highlighted in section 3.14.

4.2.3. Area under ROC curve

Another important tool used is the Area under ROC curve. The chart measures the predictive accuracy of a model. It is based on the confusion matrix. The details on the confusion matrix are highlighted in section 3.15.

4.3 Results of Models Assessment Based on Confusion Matrix & ROC Charts

The following results are derived from:

1. Imbalanced data of 1458 good debts and 542 bad debts.
2. Balanced data of 249 good debts and 249 bad debts.

4.3.1 Decision Tree

Table 4.1 Confusion Matrix DT - Imbalanced data

Actual Category	---------Predicted Category----------	
	Bad Debt	**Good Debt**
Bad Debt	**342**	**200**
Good Debt	**110**	**1348**

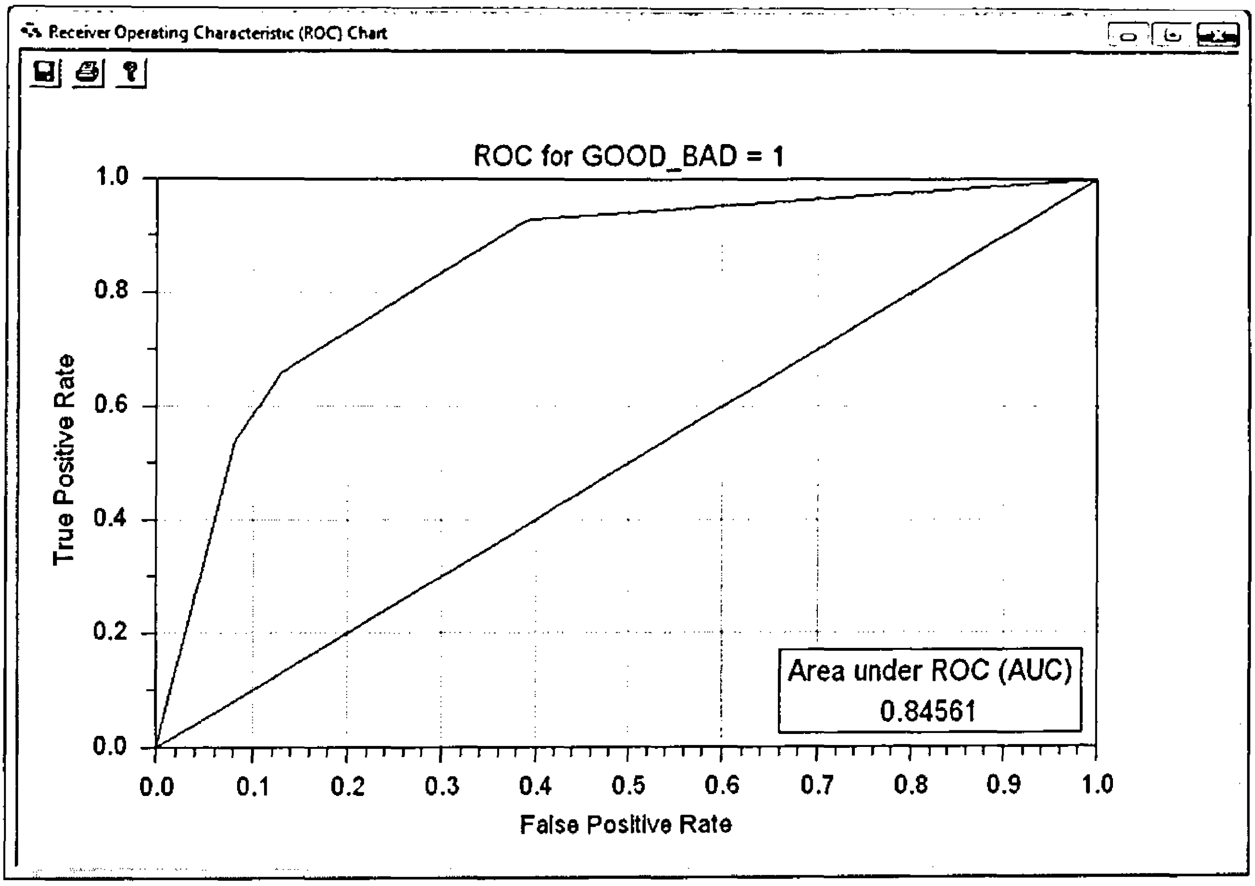
Using imbalanced data, the Decision Tree model classified 342 actual bad debts as bad and 1348 actual good debts as good. However it classified 200 actually bad debts as good debts and 110 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Given that TP – True Positives, TN – True Negatives, FP – False Positive and FN – False Negatives then: **Accuracy of model = (TP+TN)*100/(TP+TN+FP+FN) = (342+1348)/ (342+1348+110+200) =84.5%.**

Table 4.2 Confusion Matrix DT - Balanced data

Actual Category	---------Predicted Category----------	
	Bad Debt	**Good Debt**
Bad Debt	**214**	**35**
Good Debt	**49**	**200**

Using balanced data, the Decision Tree model classified 214 actual bad debts as bad and 200 actual good debts as good. However it classified 35 actually bad debts as good debts and 49 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy = 83.13%.**



Area under ROC (AUC)
0.84561

Figure 4.1 DT - Imbalanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.84561** as shown above.

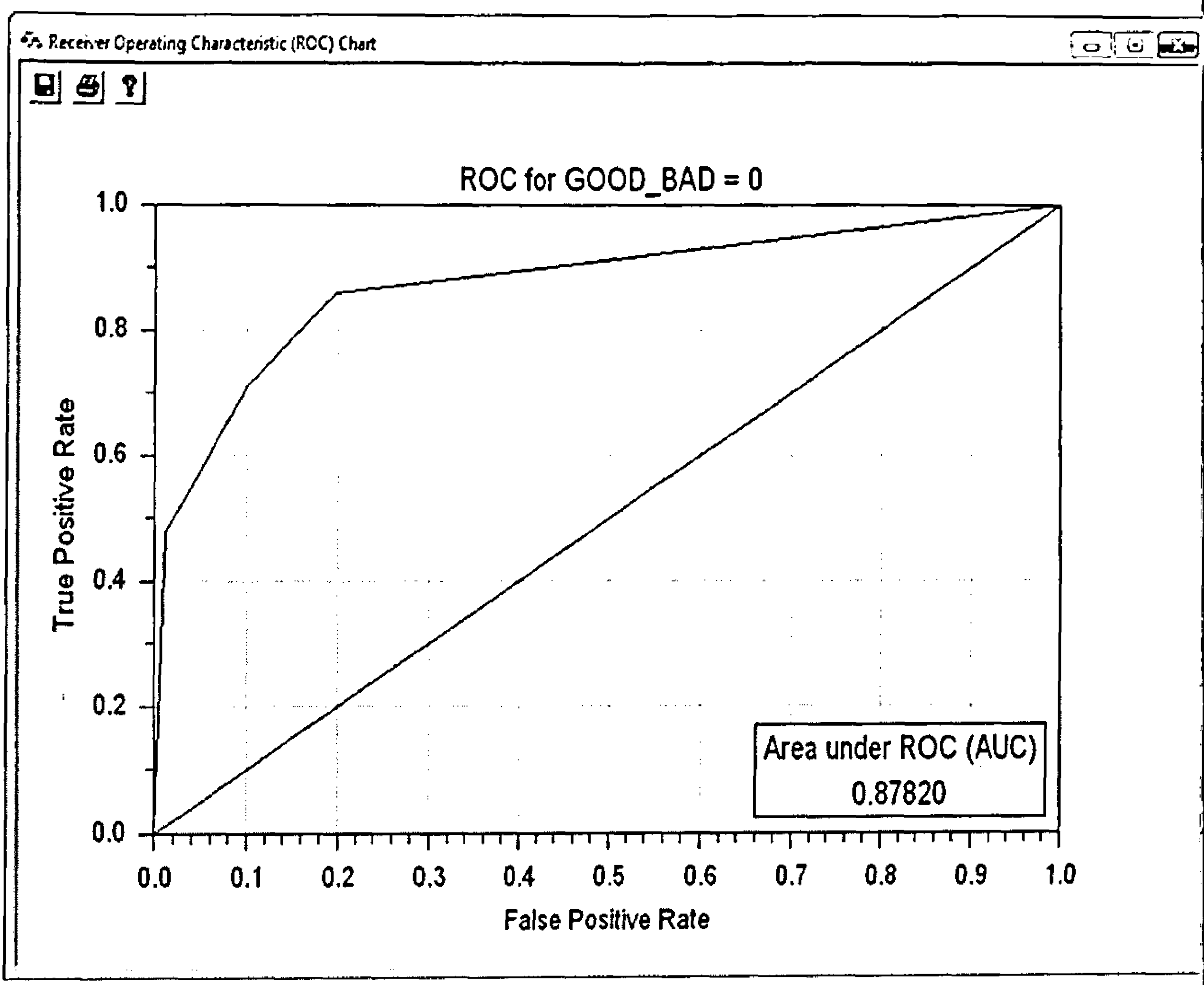


Figure 4.2 DT - Balanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for balanced data was found to be 0.87820 as shown above.

4.3.2 Neural Networks Multilayer Perception with 4 layers (2 hidden)

Table 4.3 MP NN 4/2 - Imbalanced data

Actual Category	**-------------------Predicted Category**	**---------------------**
	Bad Debt	**Good Debt**
Bad Debt	**234**	**308**
Good Debt	**138**	**1320**

Using imbalanced data, the Decision Tree model classified 234 actual bad debts as bad and 1320 actual good debts as good. However it classified 308 actually bad debts as good debts and 138 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy of model = (TP+TN)*100/ (TP+TN+FP+FN) = (234+ 1320)*100/ (234+1320+138+308) =155400/2000 = 77.7%

Table 4.4 MP NN 4/2 - Balanced data

Actual Category	**-------------------Predicted Category----------------------**	
	Bad Debt	**Good Debt**
Bad Debt	**166**	**83**
Good Debt	**30**	**219**

Using balanced data, the Decision Tree model classified 166 actual bad debts as bad and 219 actual good debts as good. However it classified 83 actually bad debts as good debts and 30 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy = 77.31%**

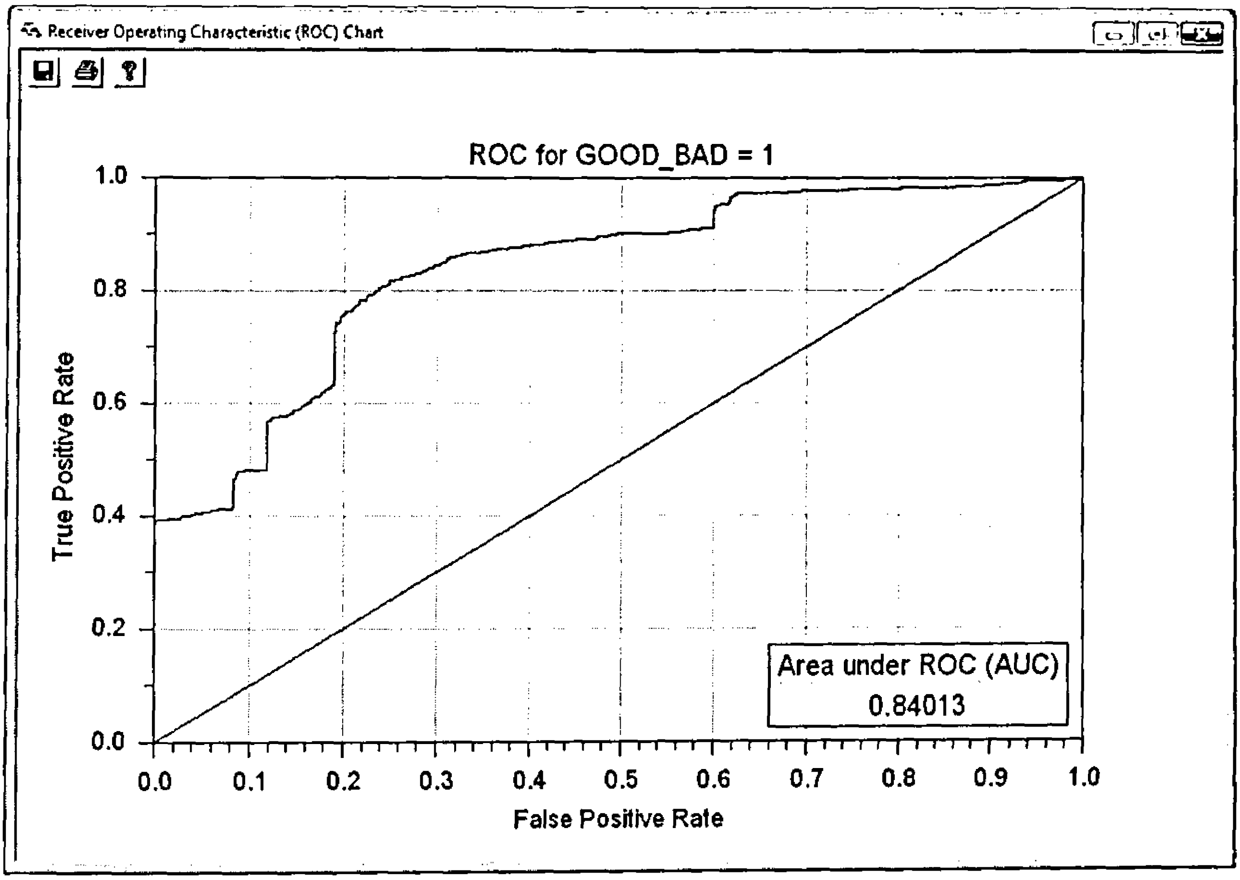


Figure 4.3 NN MP 4/2 Imbalanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.840133** as shown above.

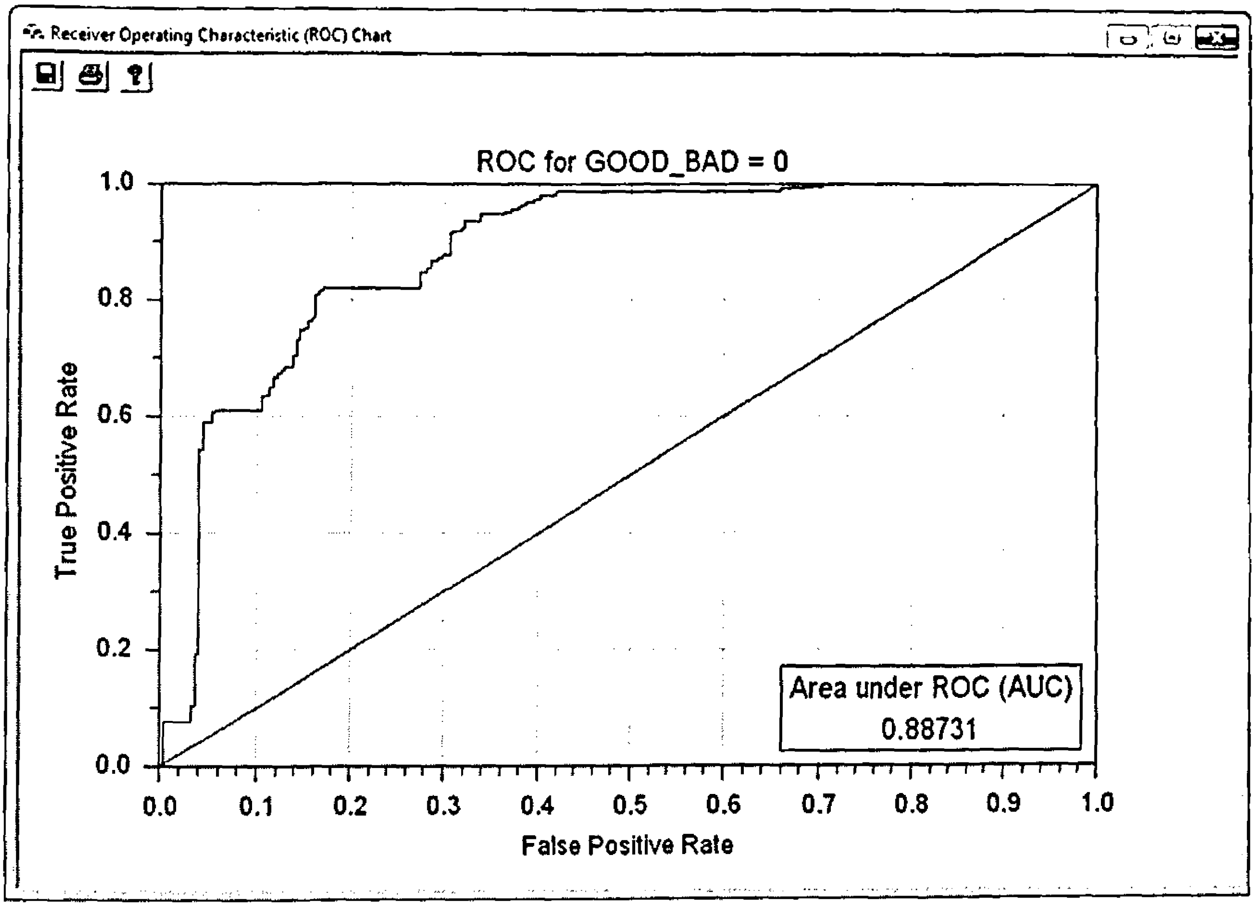


Figure 4.4 NN MP 4/2 Balanced data - Area under ROC curve (AUC) = 0.88731

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.88731** as shown above.

4.3.3 Neural Networks Multilayer Perceptron with 3 layers(1 hidden)

Table 4.5 MP NN 3/1 Imbalanced data

Actual Category	---------------Predicted Category------------------	
	Bad Debt	Good Debt
Bad Debt	295	247
Good Debt	84	1374

Using imbalanced data, the Decision Tree model classified 234 actual bad debts as bad and 1320 actual good debts as good. However it classified 308 actually bad debts as good debts and 138 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy = (TP+TN)*100/ (TP+TN+FP+FN) = (295+ 1374)*100/(295+1374+84+247) =166900/2000 = 83.45%

Table 4.6 NN MP 3/1 Balanced data

Actual Category	----------------Predicted Category----------------------	
	Bad Debt	Good Debt
Bad Debt	188	61
Good Debt	32	217

Using balanced data, the Decision Tree model classified 188 actual bad debts as bad and 217 actual good debts as good. However it classified 61 actually bad debts as good debts and 32 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy = 81.33%**

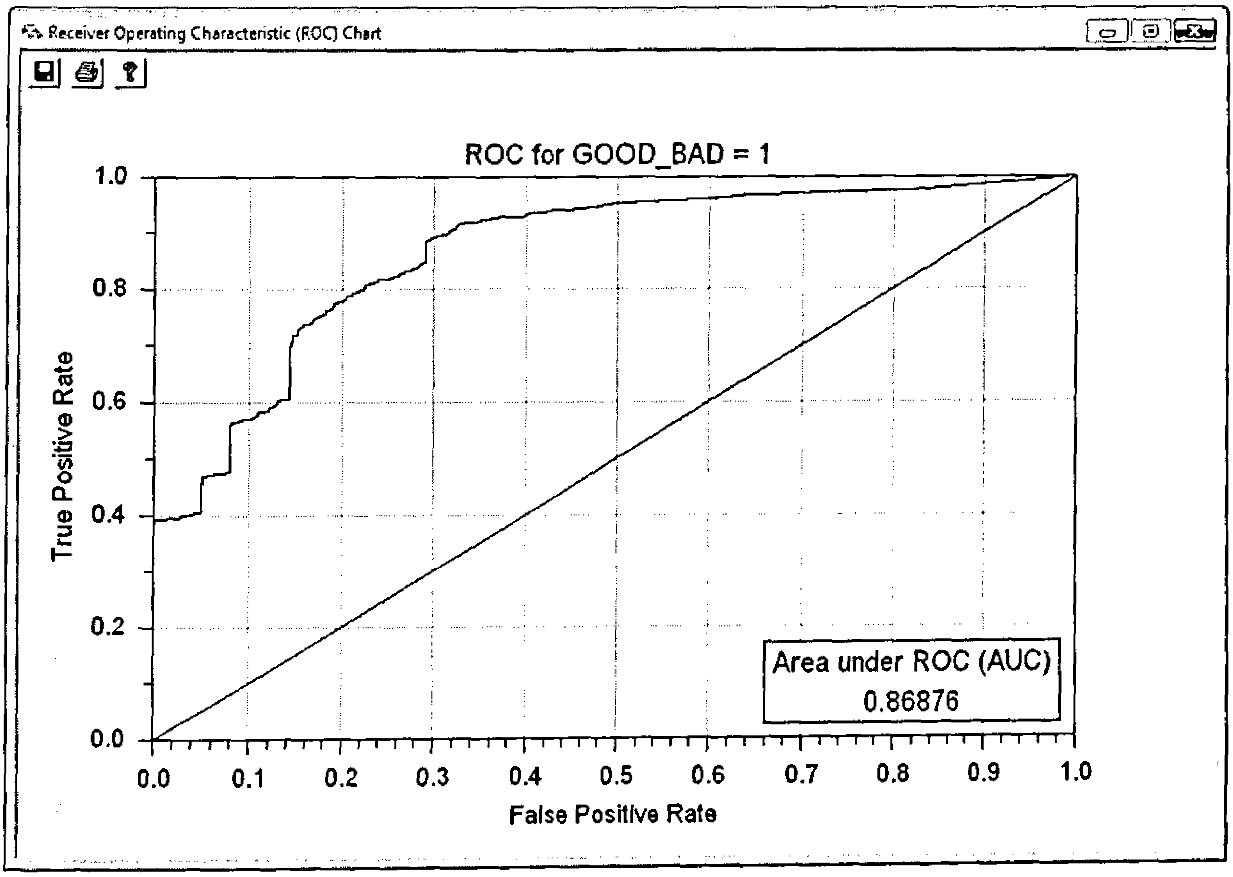


Figure 4.5 NN MP 3/1 Imbalanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be 0.868763 as shown above.

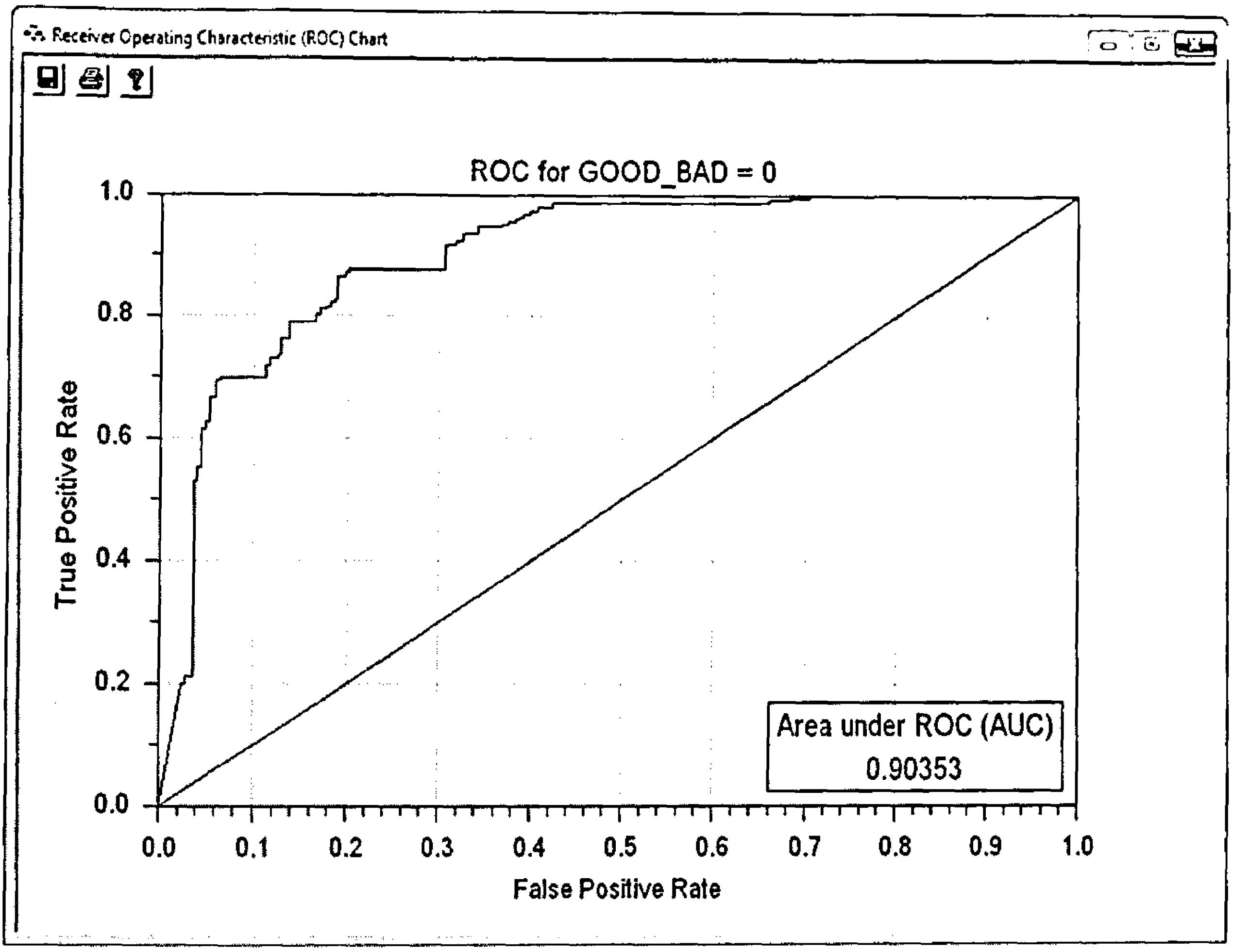


Figure 4.6 NN MP 3/1 Balanced data - Area Under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for balanced data was found to be **0.90353** as shown above.

4.3.4 Radial Basis Function (RBF) Neural Networks – Conceptually similar to K-Nearest Neighbor

Table 4.7 RBF NN Imbalanced data

Actual Category	**-----------------Predicted Category-----------------------**	
	Bad Debt	**Good Debt**
Bad Debt	**364**	**178**
Good Debt	**120**	**1338**

Using imbalanced data, the Decision Tree model classified 364 actual bad debts as bad and 1338 actual good debts as good. However it classified 178 actually bad debts as good debts and 120 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy = (364+1338)*100/(364+1338+120+178) = 170200/2000 =85.1%

Table 4.8 RBF NN Balanced data

Actual Category	---------------Predicted Category---------------------	
	Bad Debt	Good Debt
Bad Debt	217	32
Good Debt	41	208

Using balanced data, the Decision Tree model classified 217 actual bad debts as bad and 208 actual good debts as good. However it classified 32 actually bad debts as good debts and 41 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy = 85.34%**

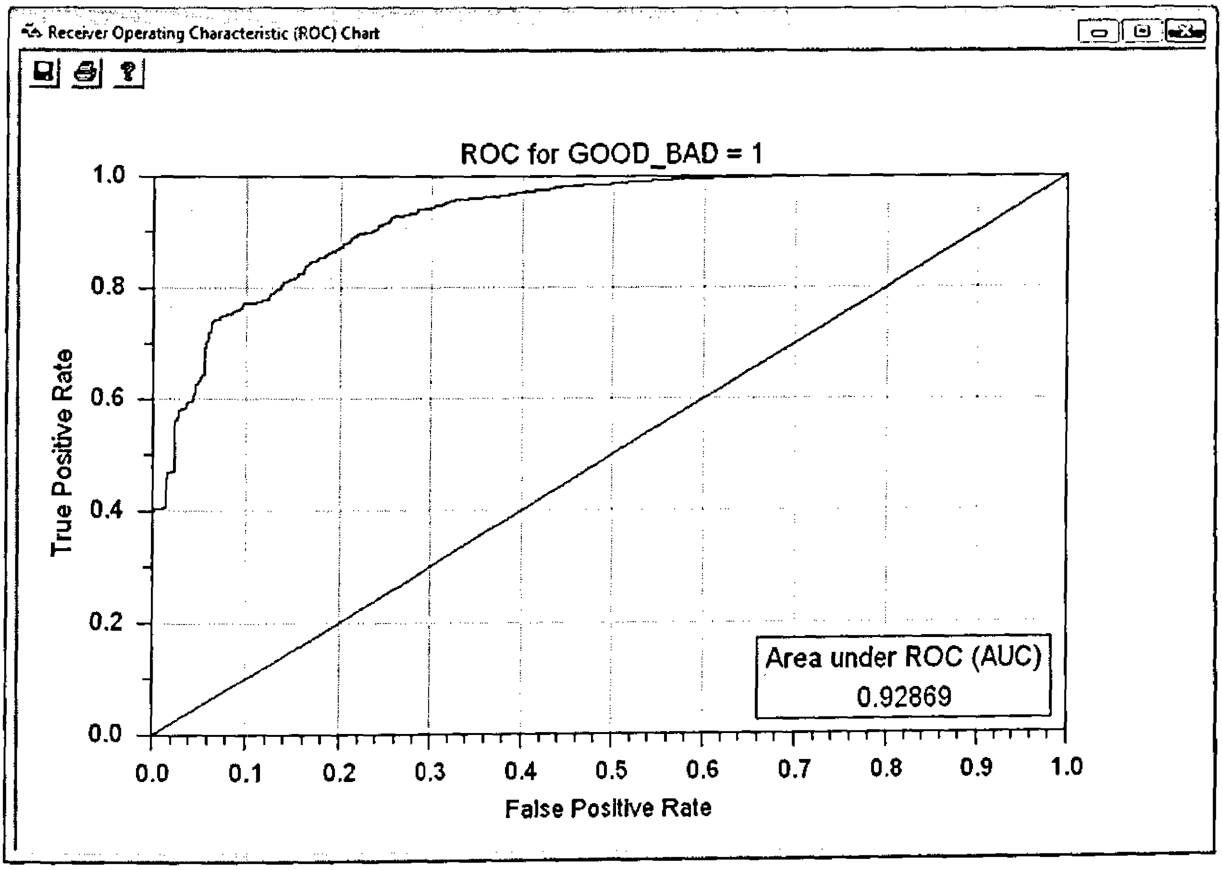


Figure 4.7 RBF NN Imbalanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.917201** as shown above.

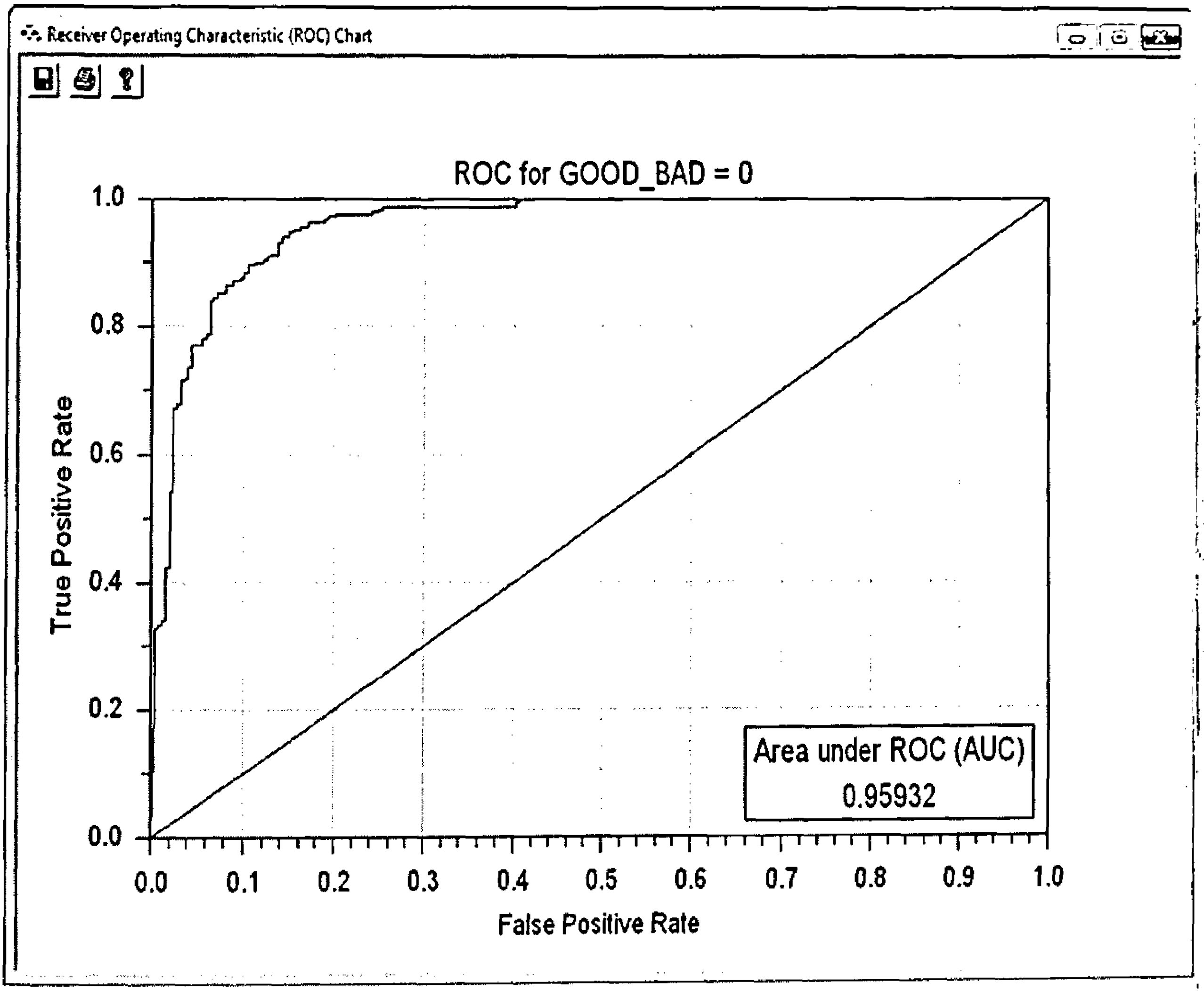


Figure 4.8 RBF NN Balanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for balanced data was found to be **0.95932** as shown above.

4.3.5 Probabilistic Neural Networks (PNN) and General Regression Neural Networks (GRNN) – PNN/GRNN Neural Network. Also conceptually similar to K-Nearest Neighbor

Tab le 4.9 PNN/GRNN - Imbalanced Data

Actual Category	----------------Predicted Category	
	Bad Debt	Good Debt
Bad Debt	377	165
Good Debt	77	1381

Using imbalanced data, the Decision Tree model classified 377 actual bad debts as bad and 1381 actual good debts as good. However it classified 165 actually bad debts as good debts and 77 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy = (377+1381)*100/(377+1381+165+77)=87.90%

Tab le 4.10 PNN/GRNN - Balanced Data

Actual Category	----------------Predicted Category	
	Bad Debt	Good Debt
Bad Debt	217	32
Good Debt	41	208

Using balanced data, the Decision Tree model classified 217 actual bad debts as bad and 208 actual good debts as good. However it classified 32 actually bad debts as good debts and 41 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy = 88.25%**

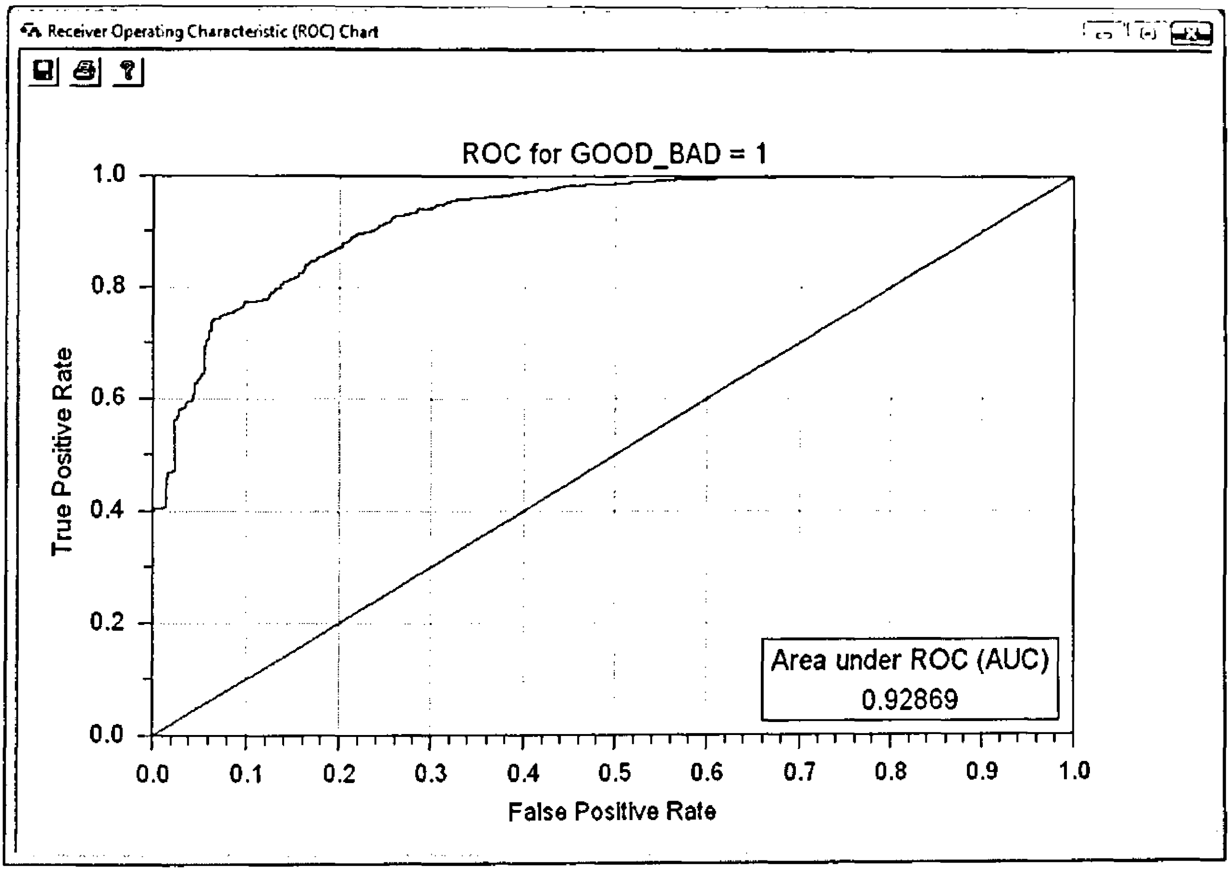


Figure 4.9 PNN/GRNN - Imbalance Data - Area under ROC curve (AUC)

The Area under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.928690** as shown above.

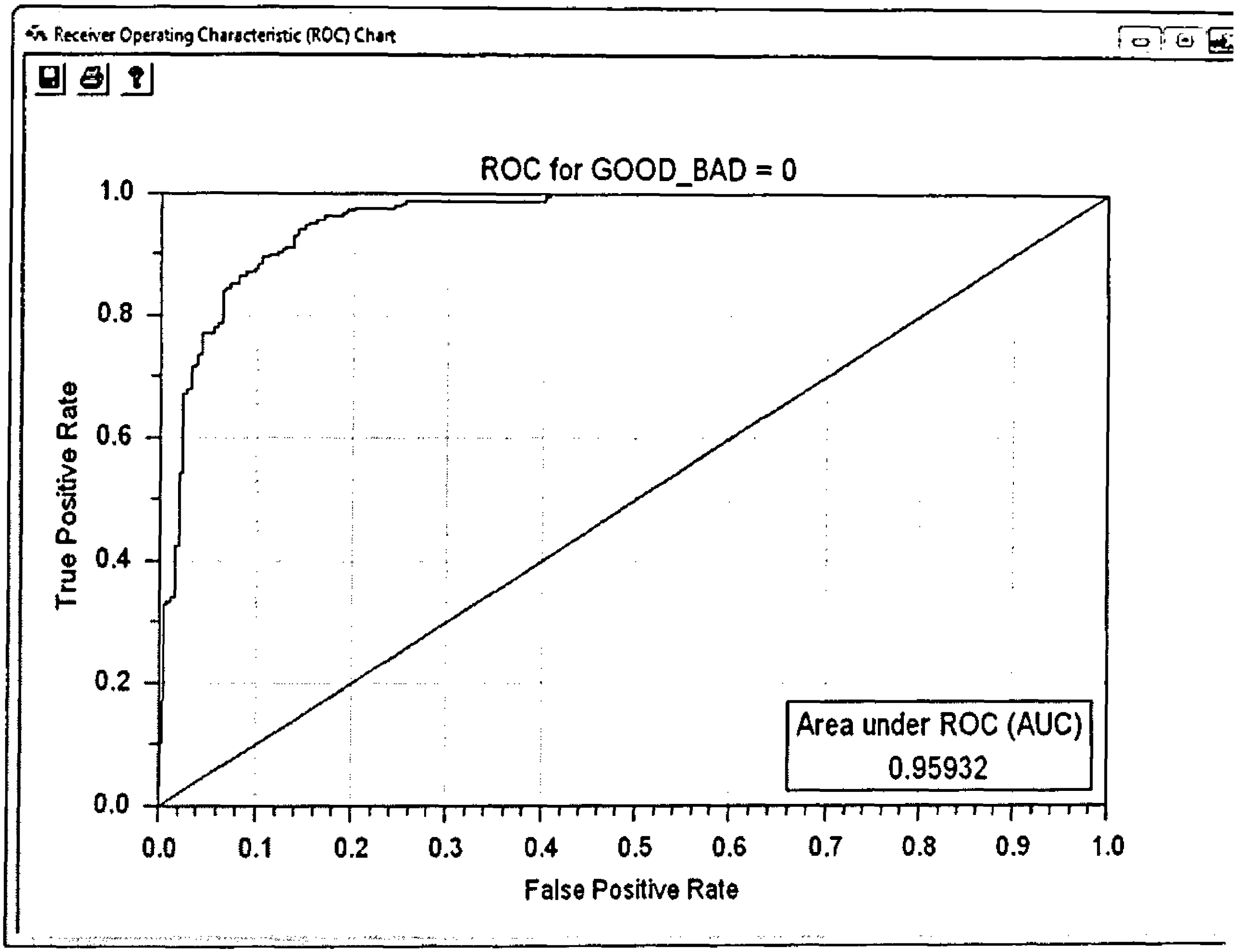


Figure 4.10 PNN/GRNN - Balance Data - Area under ROC curve (AUC)

Area under ROC curve (AUC) = 0.98963

4.3.6 Logistic Regression – a variation of linear regression

Used when the dependent (response) variable is a dichotomous variable (i.e. it takes only two values, which usually represent the occurrence or non-occurrence of some outcome event, usually coded as 0 or 1) and the independent (input) variables are continuous, categorical, or both.

Table 4.11 LR - Imbalanced data

Actual Category	**---------------------------Predicted Category---------------------**	
	Bad Debt	**Good Debt**
Bad Debt	**195**	**347**
Good Debt	**113**	**1339**

Using imbalanced data, the Decision Tree model classified 195 actual bad debts as bad and 1339 actual good debts as good. However it classified 347 actually bad debts as good debts and 113 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy = (195+1339)*100/(195+1339+113+347)= 76.70%

Table 4.12 LR - Balanced data

Actual Category	---------------------------Predicted Category-------------------	
	Bad Debt	**Good Debt**
Bad Debt	**151**	**91**
Good Debt	**28**	**221**

Using balanced data, the Decision Tree model classified 151 actual bad debts as bad and 221 actual good debts as good. However it classified 91 actually bad debts as good debts and 28 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy =76.1%**

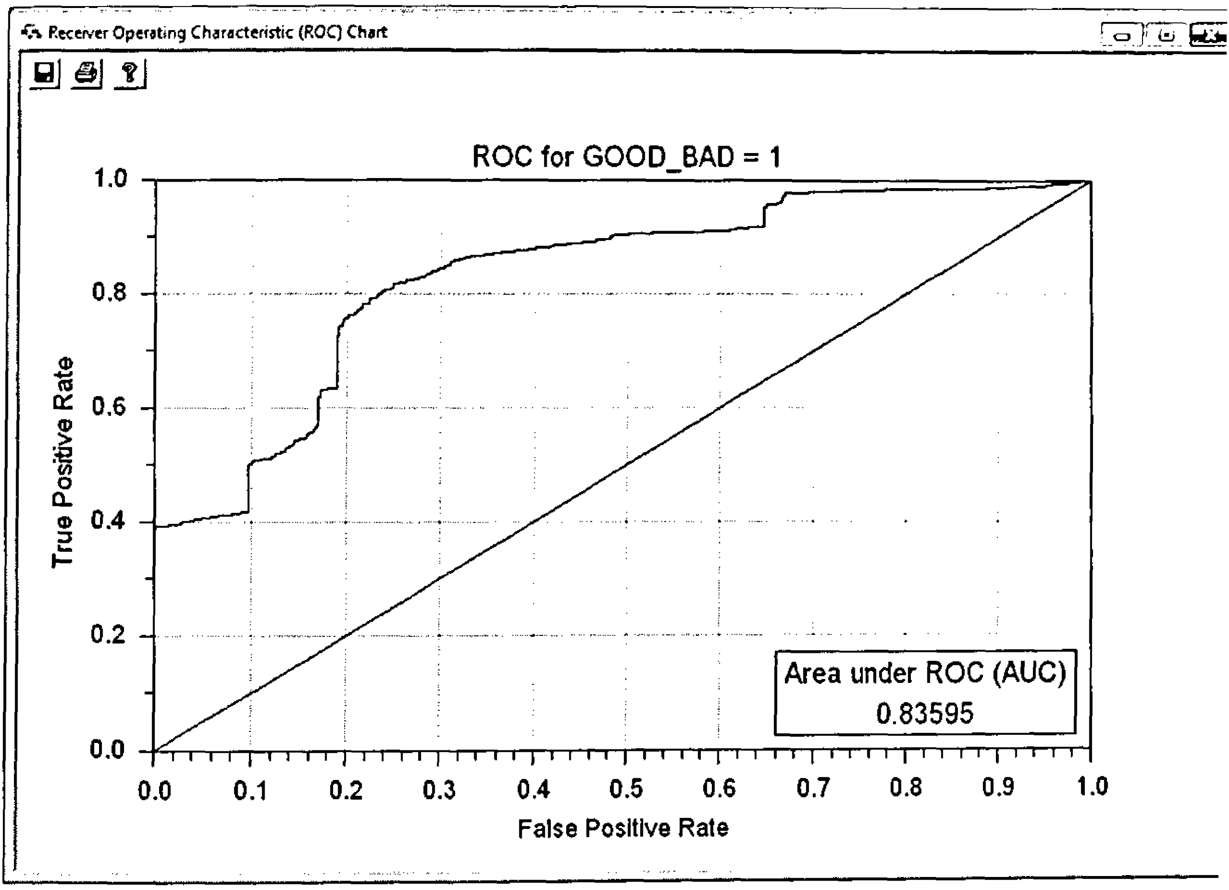


Figure 4.11 LR - Imbalanced Data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.835946** as shown above.

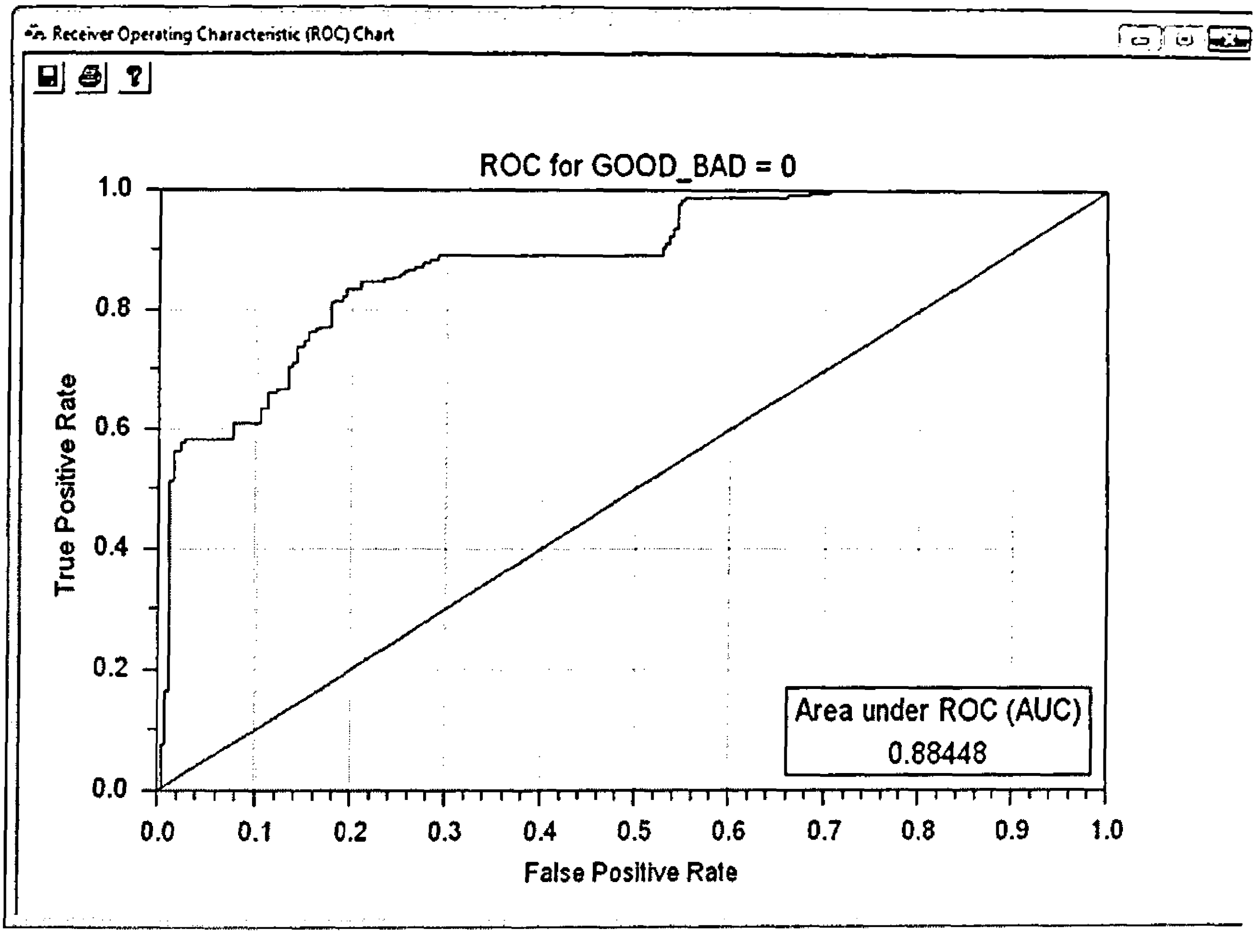


Figure 4.12 LR - Balanced Data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for balanced data was found to be **0.88448** as shown above.

4.3.7 K-Mean Clustering

The basic idea of K-Means clustering is that clusters of items with the same target category are identified, and predictions for new data items are made by assuming they are of the same type as the nearest cluster center.

Table 4.13 K-MC - Imbalanced data

Actual Category	----------------Predicted Category------------------	
	Bad Debt	Good Debt
Bad Debt	271	271
Good Debt	86	1372

Using imbalanced data, the Decision Tree model classified 271 actual bad debts as bad and 1372 actual good debts as good. However it classified 271 actually bad debts as good debts and 86 actually good debts as bad debts. Hence the accuracy is calculated as shown below:

Accuracy = (271+1372)*100/(271+1372+86+271)= 82.15%

Table 4.14 K-MC - Balanced data

Actual Category	------------------Predicted Category------------------	
	Bad Debt	Good Debt
Bad Debt	171	78
Good Debt	21	228

Using balanced data, the Decision Tree model classified 171 actual bad debts as bad and 228 actual good debts as good. However it classified 78 actually bad debts as good debts and 21 actually good debts as bad debts. Hence the accuracy was calculated and found to be: **Accuracy =80.02%**

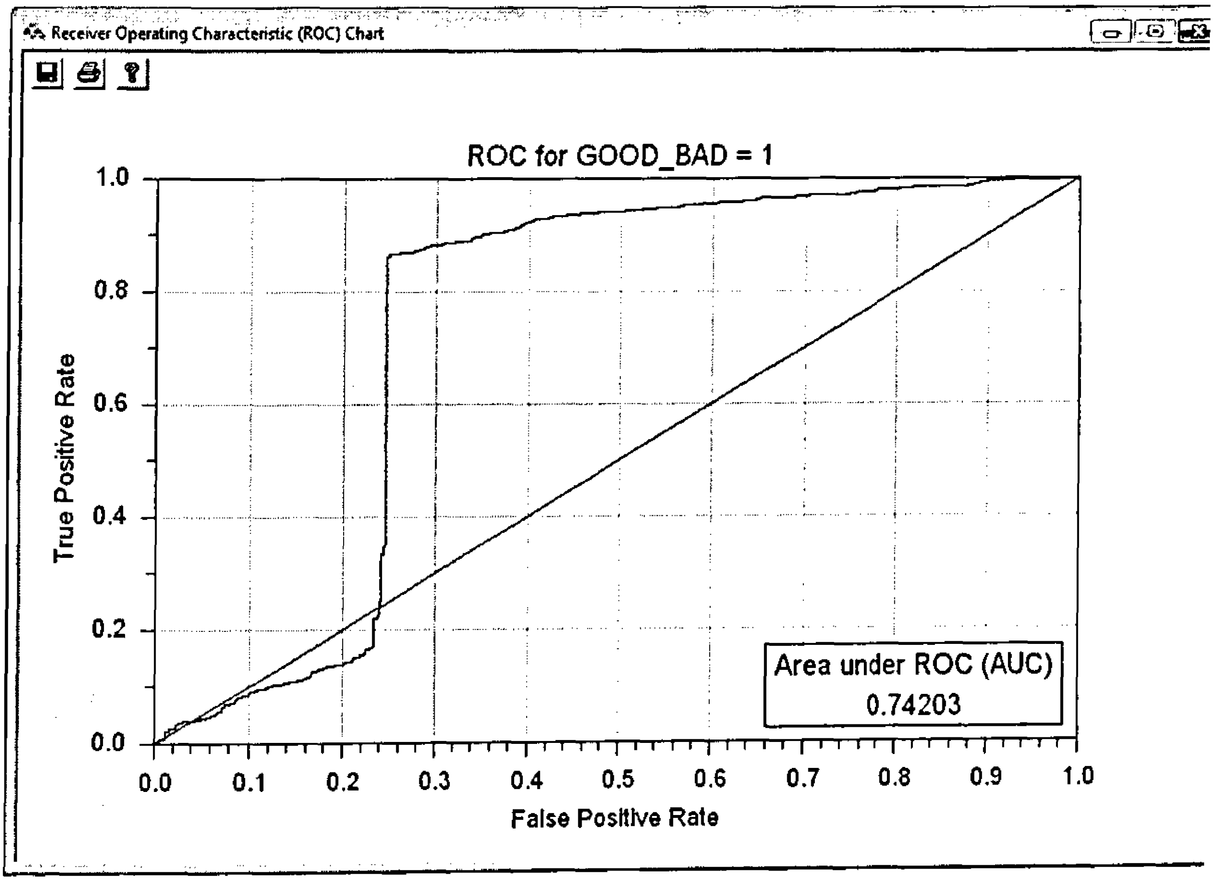
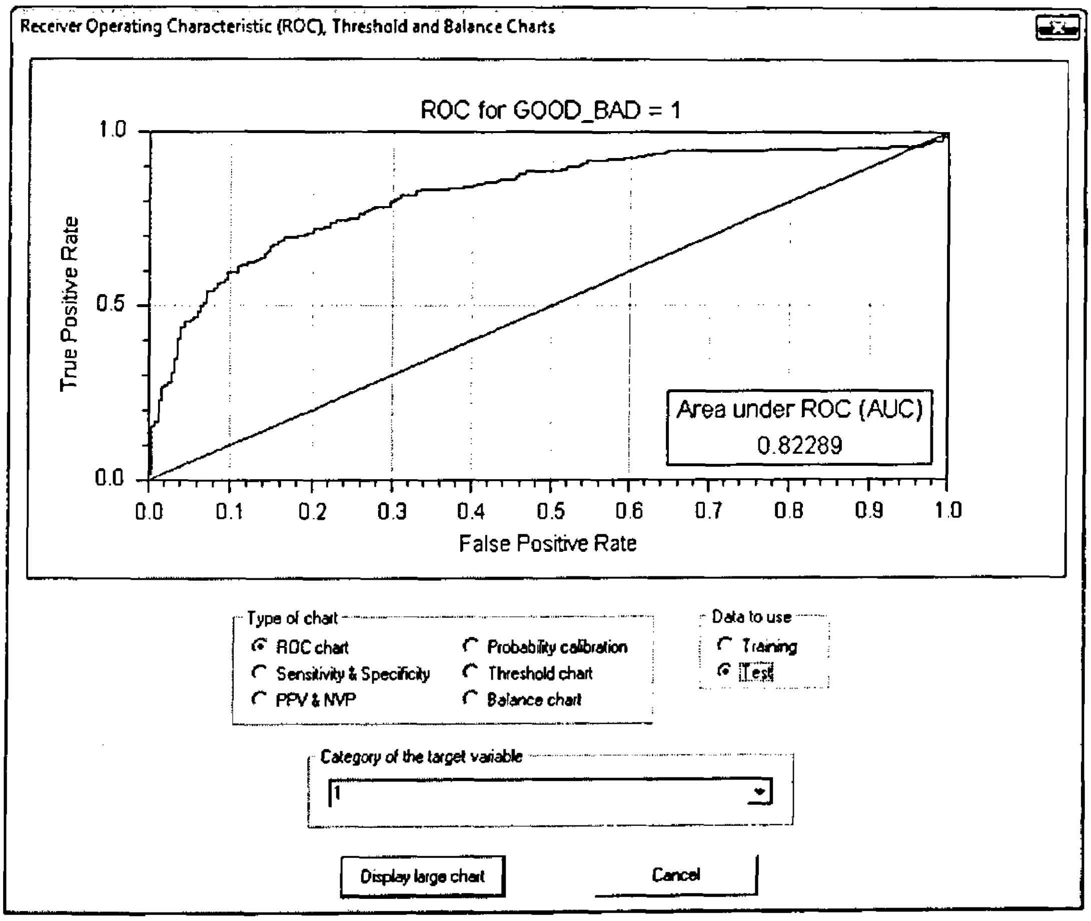


Figure 4.13 Imbalanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for imbalanced data was found to be **0.742033** as shown above.



Type of chart

● ROC chart ○ Probability calibration
○ Sensitivity & Specificity ○ Threshold chart
○ PPV & NVP ○ Balance chart

Data to use

○ Training
● Test

Category of the target variable: 1

Figure 4.14 K-MC - Balanced data - Area under ROC curve (AUC)

The Area Under the Receiver Operating Curve (AUC) for balanced data was found to be 0.82289 as shown above.

Table 4.15 Tabulated Results of the Assessment

	Decision Tree	NN MP with 4 layers(2 hidden)	NN MP with 3 layers(1 hidden)	RBF NN – KNN	PNN/GRNN	Logistic Regression	K-Mean Clustering
Accuracy (Imbalanced)	84.5%.	77.7%	83.45%	85.1%	87.90%	76.70%	82.15%
Accuracy (Balanced)	83.13%	77.31%	81.33%	85.34%	88.25%	76.1%	80.02%
AUC (Imbalanced)	0.84561	0.84013	0.86876	0.91720	0.92869	0.83595	0.74203
AUC (Balanced)	0.87820	0.88731	0.90353	0.95932	0.98963	0.88448	0.82289

Order of performance

Based on the results obtained above, the order of the performance of the data mining tools:

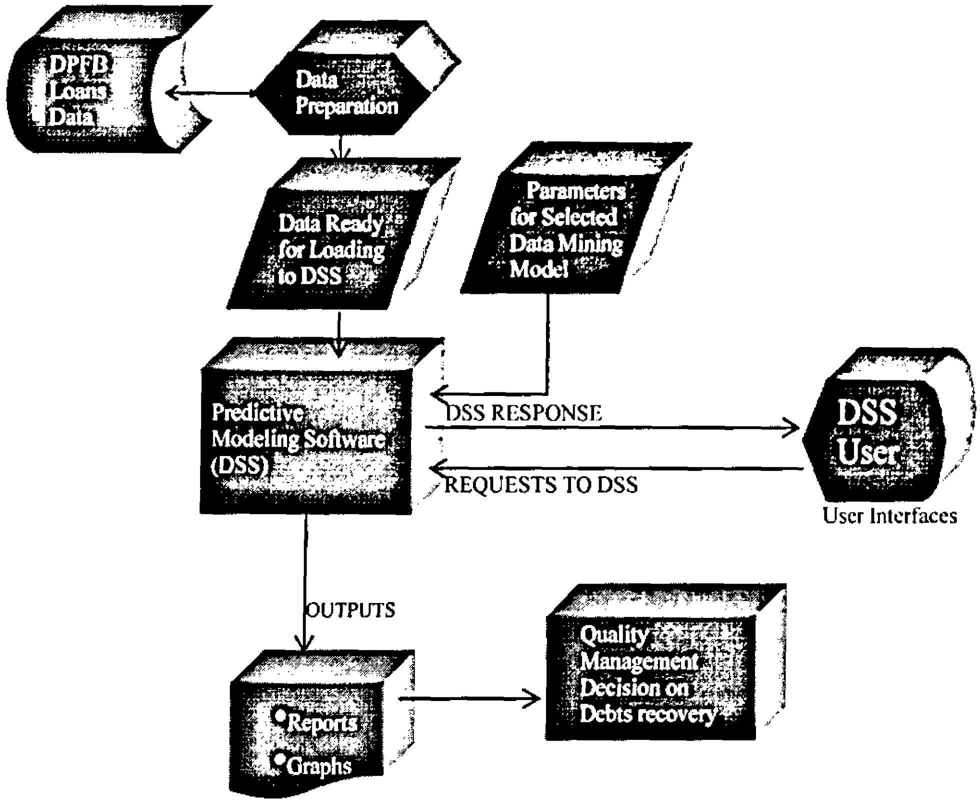
1. PNN/GRNN
2. RBF NN
3. NN MP with 3 layers (1 hidden)
4. Decision Tree
5. K-Mean Clustering
6. NN MP with 4 layers (2 hidden)
7. Logistic Regression

Neural Networks models performed better overall as compared to the others as shown in the above analysis hence it is recommended that the prototype be based on Neural Networks.

The difference in accuracy while using imbalanced and balanced data was minimal. However for DT, MPNN4/2, MPNN3/1, LR, K-mean the accuracy came down when using balanced data. While for RBF, PNN/GNN the accuracy went up with balanced data. For all of them the AUC went up with balanced data.

4.4 Design of the Prototype

The component of the prototype is as described in the literature review in the diagram below:


DPFB Loans Data
Data Preparation
Data Ready for Loading to DSS
Parameters for Selected Data Mining Model
Predictive Modeling Software (DSS)
DSS RESPONSE
REQUESTS TO DSS
DSS User
User Interfaces
OUTPUTS
Reports
Graphs
Quality Management Decision on Debts recovery

These components are elaborated below:

DPFB Loans Data – This is the loans data that is extracted from a loans table in the institutions' databases. The content of this data is as shown on figure I.

Data Preparation – This is the process of transforming the data ready for the decision support program. This process is meant to reflect the predictor and target variable as described in section 3.6. A sample of transformed data is as shown on figure I.9.

Data Ready for Loading to DSS – This is the transformed data that is ready to be processed by the DSS. A sample of transformed data is as shown on figure I.9.

Predictive modeling Software – This is software used to train and classify debts. This is the software described in section 4.5.

Parameters for Selected Data Mining Model – This mainly refer to selections done by the user on training data, output data and the classification data. This is mainly referring to selection done by user as indicated in the user guide of the prototype.

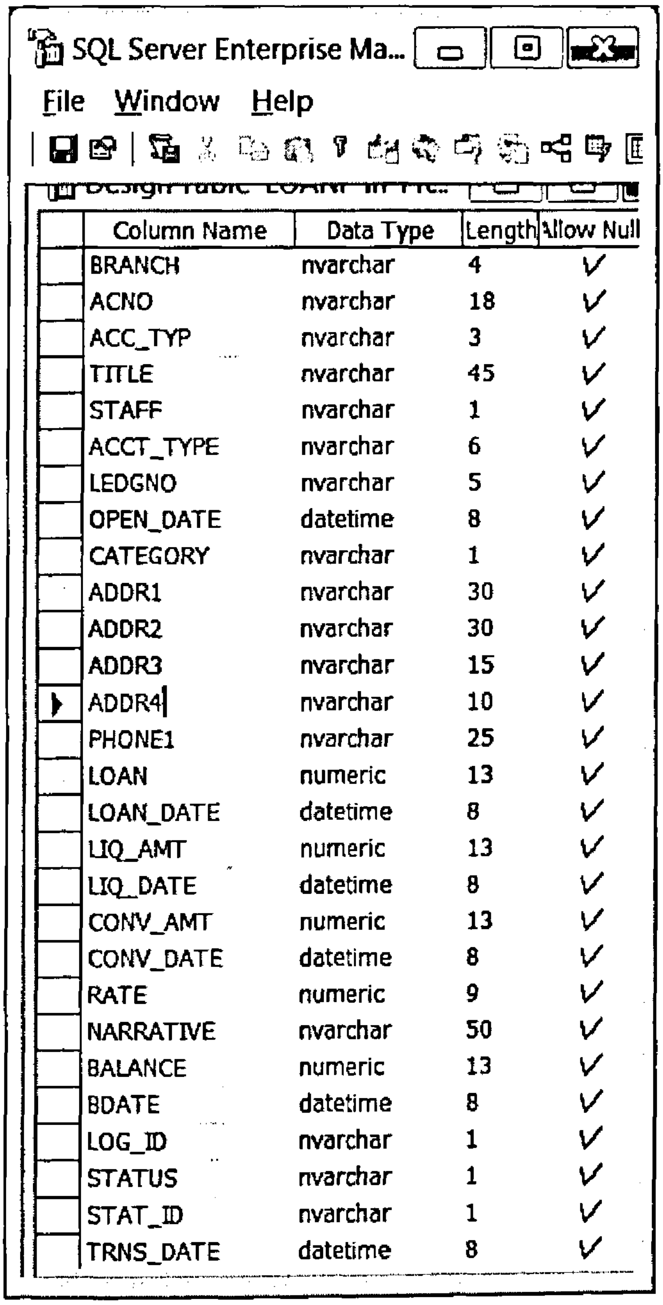
Reports – Reports generated after classification by the modeling software. Sample report is shown on figure I.29.

Quality Management Decision on Debts recovery– The results from this model is expected assist in making quality management decisions on debt recovery in DPFB.

User interfaces – This are the interfaces that the user uses to interact with the model. This interfaces are clearly portrayed in the user guide in Appendix I.

4.4.1 Input Table Design

The input table is LOANF located in the institution's database. The structure component is as follows:

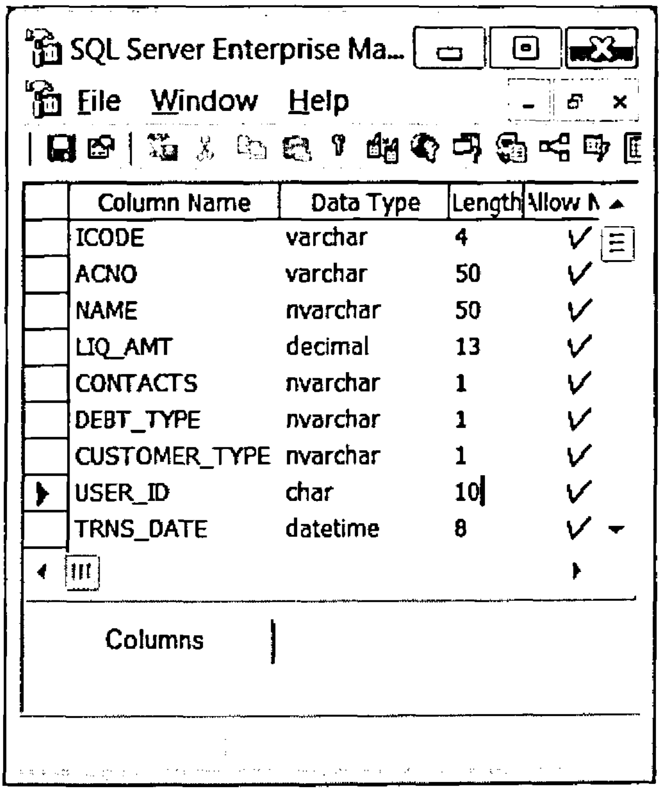


Column Name	Data Type	Length	Allow Null
BRANCH	nvarchar	4	✓
ACNO	nvarchar	18	✓
ACC_TYP	nvarchar	3	✓
TITLE	nvarchar	45	✓
STAFF	nvarchar	1	✓
ACCT_TYPE	nvarchar	6	✓
LEDGNO	nvarchar	5	✓
OPEN_DATE	datetime	8	✓
CATEGORY	nvarchar	1	✓
ADDR1	nvarchar	30	✓
ADDR2	nvarchar	30	✓
ADDR3	nvarchar	15	✓
ADDR4	nvarchar	10	✓
PHONE1	nvarchar	25	✓
LOAN	numeric	13	✓
LOAN_DATE	datetime	8	✓
LIQ_AMT	numeric	13	✓
LIQ_DATE	datetime	8	✓
CONV_AMT	numeric	13	✓
CONV_DATE	datetime	8	✓
RATE	numeric	9	✓
NARRATIVE	nvarchar	50	✓
BALANCE	numeric	13	✓
BDATE	datetime	8	✓
LOG_ID	nvarchar	1	✓
STATUS	nvarchar	1	✓
STAT_ID	nvarchar	1	✓
TRNS_DATE	datetime	8	✓

Figure 4.15 Loan Master Table

4.4.2 Output Design

The output table is PROJDATA structure is as shown below:



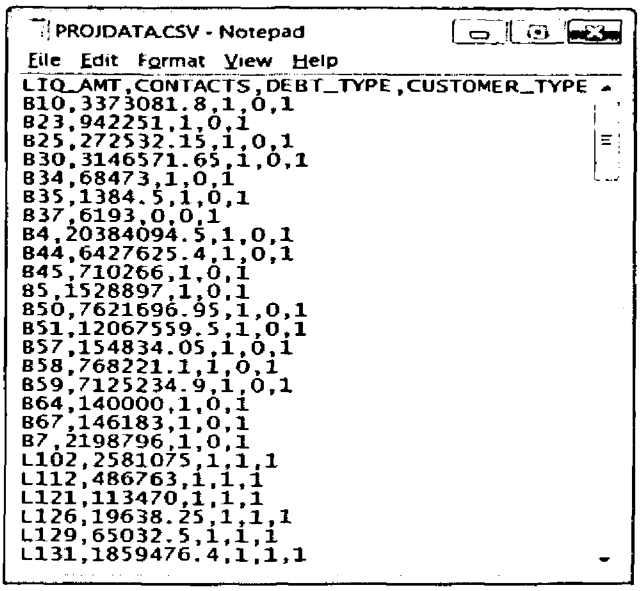
	Column Name	Data Type	Length	Allow N
	ICODE	varchar	4	✓
	ACNO	varchar	50	✓
	NAME	nvarchar	50	✓
	LIQ_AMT	decimal	13	✓
	CONTACTS	nvarchar	1	✓
	DEBT_TYPE	nvarchar	1	✓
	CUSTOMER_TYPE	nvarchar	1	✓
▶	USER_ID	char	10	✓
	TRNS_DATE	datetime	8	✓

Figure 4.16 Project Data Table

The other output file is a text file which is coma delimited whose format is as shown below:

LIQ_AMT,CONTACTS,DEBT_TYPE,CUSTOMER_TYPE
XXXXXX,9999999999.99,X,X,X

The out looks as follows:



```
LIQ_AMT,CONTACTS,DEBT_TYPE,CUSTOMER_TYPE
B10,3373081.8,1,0,1
B23,942251,1,0,1
B25,272532.15,1,0,1
B30,3146571.65,1,0,1
B34,68473,1,0,1
B35,1384.5,1,0,1
B37,6193,0,0,1
B4,20384094.5,1,0,1
B44,6427625.4,1,0,1
B45,710266,1,0,1
B5,1528897,1,0,1
B50,7621696.95,1,0,1
B51,12067559.5,1,0,1
B57,154834.05,1,0,1
B58,768221.1,1,0,1
B59,7125234.9,1,0,1
B64,140000,1,0,1
B67,146183,1,0,1
B7,2198796,1,0,1
L102,2581075,1,1,1
L112,486763,1,1,1
L121,113470,1,1,1
L126,19638.25,1,1,1
L129,65032.5,1,1,1
L131,1859476.4,1,1,1
```

Figure 4.17 Text File Output (Comma delimited)

Date Printed XX/XX/XXXX **LOANS CLASSIFICATION REPORT**

Institution Name: XXXXXXXXXXXXXXX

Institution Code	Account No.	Account Title	Amount at Liquidation	Loan Classification
XXXX	XXXXXXXX	XXXXXXXXXXXX	99999999.99	XXXXXXXXXXXX

Figure 4.18 Report Design

4.4.5 Program Design

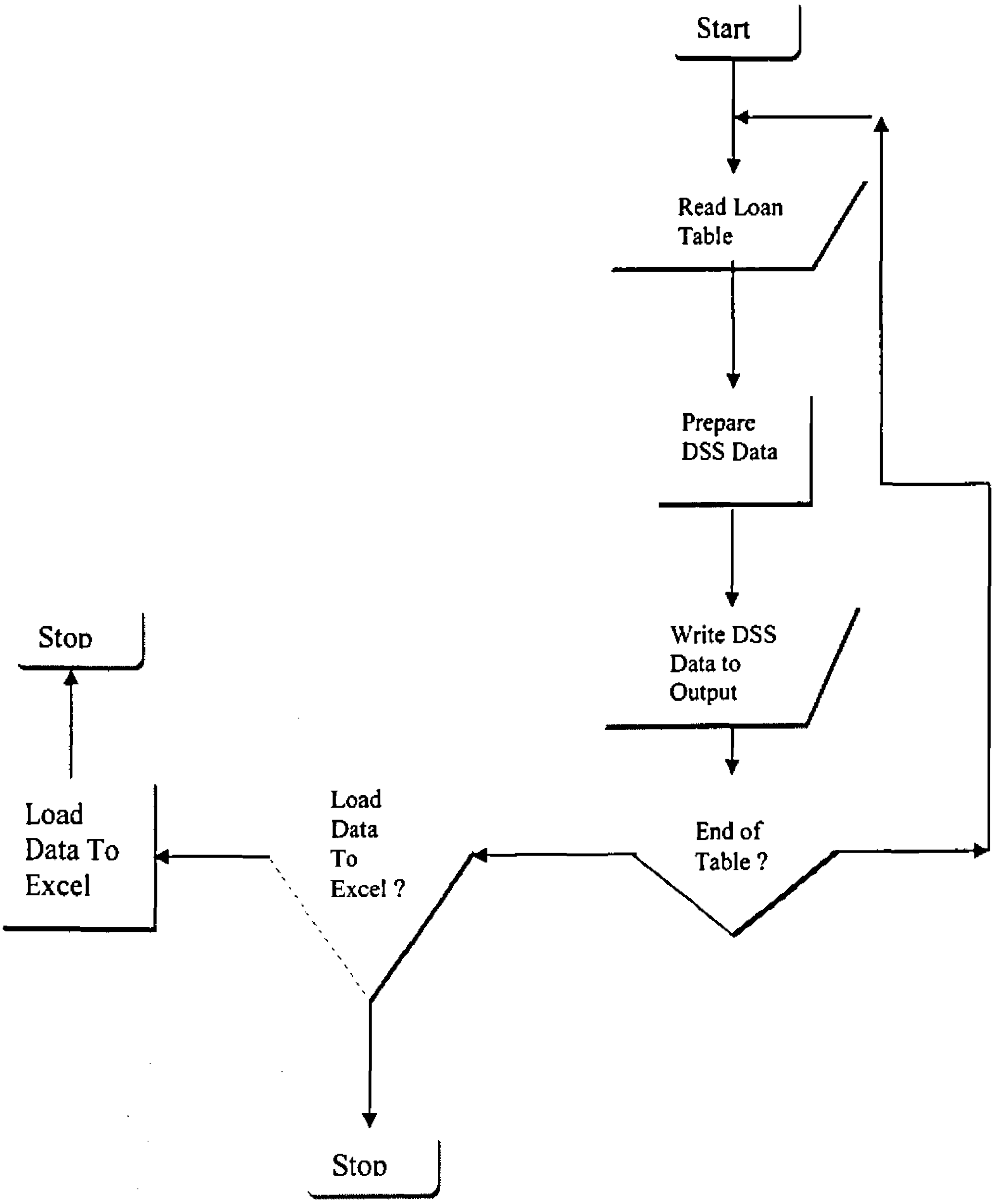


Figure 4.19 Data Preparation Process

Training Process

- Start forecasting module.
- Create new network.
- Select input range.
- Select target range.

- Run training to completion.
- Training will run to the end of the learning process.

Forecasting Process

- Start forecasting component.
- Select input range.
- Run the forecast.
- On completion output displayed on the forecasted column.

Reporting Process

- Start process
- Prepare data for to completion
- Display report
- Print report

4.5 Prototype Development

- **Tools**

The prototype was developed using various tools:

- Microsoft Visual Basic language version 6.0
- For the database (DBMS) SQL 2000 was used.
- The reporting tool was Cystal Report 8.5.

The DSS software that formed a part of the MBMS was Alyuda Forecaster XL. It uses algorithms developed by Alyuda's Research Group which tailor data to a neural network and select the most suitable model and prepare it to solve the problem. On the invocation of Alyuda Forecaster XL it's menu appears within Excel. This software uses Alyuda's proprietary algorithms of automatic data preprocessing and neural network preparation (174.123.20.5/download/alyuda-forecaster-xl-1-0-22147.html, www.2haveit.com/listdetai~id~13292.html). These algorithms tailor data to a neural network, algorithm to train network and select the most suitable neural network architecture and prepare the network for forecasting. Uses constructive select the network topology. This constructive algorithm is developed by Alyuda's Research Group and is capable of automatic selection and tuning of training parameters and network topology.

- DGMS – The Dialogue Generation and Management System which essentially is the user interface was developed using Microsoft Visual Basic language version 6.0.

4.5.1 Testing of Alyuda Forecaster XL

- 500 loan records were prepared for training and validation of Alyuda Forecaster XL.
- Training and validations were performed on the data using the software.

- The actual plus predicted data was loaded to SQL 2000 and analysed. The following results were noted:

Table 4.16 Confusion Matrix – Alyuda Software

Actual Category	-------------------------Predicted Category-----------------	
	Bad Debt	Good Debt
Bad Debt	198	52
Good Debt	7	243

Accuracy = (198+243)*100/(198+243+7+52)= 88.2%

4.5.2 Size of Data and Accuracy in Alyuda Forecaster XL

Different sizes of data were used in testing Alyuda Forecaster XL. The following results were observed:

- While using 269 records the accuracy was observed to be 70.26%.
- Using 515 records Alyuda obtained 74.95% accuracy.
- With 1203 records the accuracy rose to 79.14.

Table 4.17 Tabulated Results – Different sizes of data

Training and Testing Data	1203 New Records	525 New Records	269 New Records
88.2%	79.14%	74.95%	70.26%

Observations

- With low quantity of data low accuracy was encountered.
- The higher the number of debts records the higher the accuracy.

From the above the following can be deduced:

- There Alyuda Forecaster XL may not be suitable for small quantity of data.
- Hence small data will require to be combined with other data for higher accurate classification.

CHAPTER FIVE

5.0 CONCLUSIONS AND RECOMMENDATIONS

5.1 REVIEW OF OBJECTIVES OF THE STUDY

As recorded in chapter one the objectives of this study were:

- To ascertain the IT measures taken to enhance the debt recovery process in DPFB.
- To identify the methods DPFB management uses to classify debts.
- To establish the level of success achieved in the debt recovery process in DPFB.
- To determine the appropriate data mining tools in evaluating whether a debt is likely to be repaid using data from institutions in liquidation in DPFB.
- Based on the finding of the study to recommend the appropriate tool that can aid DPFB management in decision support in loan recovery.
- Development of a DPFB DSS Prototype

5.2 ACHIEVEMENTS OF THE STUDY

This study was able to achieve the following:

5.2.1 IT Measures Taken to Enhance Debt Recovery

The IT measures taken by DPFB to enhance the debt recovery process were identified as follows: DPFB has an in house developed liquidation system. Within the system is a loans module that assists in maintenance of loan information. This information includes: Takeover balances, Repayments transactions, Loan master details and Accounts history. They also have the following:

- A Registry system that assists in tracking loans files movement.
- A Court Case module that is meant to assist in tracking loan cases in court.
- In a recent development DPFB uses the liquidation system to produce data that is forwarded to the Credit Reference Bureau (CRB). Due to the implications of shared credit information, the CRB component can stimulate and enhance the debtors' response due to attempts to keep a clean record.

5.2.2 Methods used to Classify Debts

The methods that DPFB uses to classify good or bad debts were noted as follows:
Currently DPFB does not have a Decision Support System to aid in classifying good and bad debts. They base their classification of good and bad debts on:

- Customer's response to demand notes:
    - No response signals a potential bad debt.

    - Response signals potential good debt.
- Study on loans documentations to ascertain validity and reliability:
    - Poor documentations signals potential bad debt.
    - Proper documentation signals potential good debt.
- Availability of security:
    - No security signals a potential bad debt.
    - Available security signals a potential good debt.

5.2.3 Success Achieved in Debt Recovery

From DPFB loans data the following was established: Taking sample of seven institutions and looking at their loans data, out of 6,356 loans only 1,531 have been recovered/ compromised. This is 24% success. This is fairly low success as this included the negotiated and compromised debts.

5.2.4 Determination of Appropriate Data Mining Tool

On determination of the appropriate data mining tools in evaluating whether a debt is likely to be repaid using data from institutions in liquidation in DPFB, the following was achieved:

The study compared the effectiveness of the models: decision trees, MP NN (4/2), MP NN (3/1), RBF NN, PNN & GRNN, logistic regression and K-mean clustering in classifying debts in DPFB. The data analysis and evaluation of the performance of the various models was based on data collected from 27 institutions in DPFB.

Using Cross Validation the models were trained and validated using known cases i.e. debts with target variable.

More subtle and meaningful interpretation of the results was obtained by the use of confusion matrix and Area Under the ROC Chart i.e. Area Under Curve (AUC).

Using trial version of data mining software (DTREG) the study found that the performance of the mentioned models in classifying debts is in the following order starting with the most accurate: PNN/GRNN, RBF NN, NN MP 3/1, Decision Tree, K-Mean Clustering, NN MP 4/2 and Logistic Regression. Their percentage accuracies were recorded as 87.90%, 85.1%, and 84.5%, 83.45%, 82.15%, 77.7% and 76.70% respectively. The AUC was found to be 0.92869, 0.91720, 0.84561, 0.86876, 0.74203, 0.84013 and 0.83595 respectively.

The neural network models showed better performance as noted in the first three i.e. PNN/GRNN, RBF NN, NN MP 3/1.

5.2.5 Development of DSS Prototype

On development of a decision support prototype the following was achieved:

Forecasting and Classification software, Alyuda Forecaster XL, was used to aid in the Model Based Management System (MBMS) component of the DSS prototype. This software uses algorithms developed by Alyuda's Research Group which tailor data to a neural network and select the most suitable model and prepare it to solve the problem.

The Data Base Management (DBMS) component of the DSS prototype was based on SQL 2000.

The Dialogue Generation and Management System (DGMS) component was developed using Microsoft Visual Basic 6.0 with the reporting tool being Cystal Report 8.5.

The training of the prototype was done using already existing record in institutions' loan data with known target variable.

After the training the prototype was able to classify debts with unknown target variable (good debt or bad debt).

Out of the study it was notable that Neural Networks are suitable for classifying debt (into good or bad debt).

The use of a Neural Network based DSS in classifying debts has been realized from this study where debts were classified as good or bad based on specific variables. As shown in the study a high accuracy was achieved of at least 79.14%. This accuracy as revealed in the study would rise with higher number of records.

The use of this DSS is expected to boost the success in debt recovery which has been at 24% as depicted in the study.

5.3 CONCLUSIONS

As shown in the study the DSS prototype will produce a high accuracy with large data in classifying debts. This is significant when utilized by DPFB since it will potentially enable an early detection of performing and non performing debts. This will allow the DPFB institutions in liquidation to focus preliminary debt recovery efforts on the good customers and save on administrative expenses by either writing off the bad debts or turning them over immediately to a collection agency.

Using the system to aid in the early detection of performing and non-performing debts and application of the derived knowledge will prospectively reduce liquidation expenses and shorten the length of the liquidation process.

After comparing the performance of the selected tools, and establishing that Neural Networks model outperformed the other models, this study has ascertained that Neural Networks are suitable for decision support in debt recovery process and can be used for decision support during debt recovery process.

Report generated by the system developed in this study will provide the DPFB with a list of debts grouping the recoverable and unrecoverable loans hence ease of distinguishing the debts.

The study has showed that neural networks are superior to the other selected tools in predicting whether a debt is recoverable or not. Apart from being simple and fast in learning, a major advantage is that no assumptions need to be made about underlying function or model since the neural network is able to extract hidden information from the historical data.

5.3 RECOMMENDATIONS

In addition to the manual techniques used by DPFB to classify debts and in bid to enhance effectiveness in debt collection, it is recommended that DPFB first deploy and use this decision support system to enhance the classifications before it turns over potentially bad debt cases to a collection agency. This will fortify the manually obtained results.

As shown on table 4.5.2.1 Alyuda Forecaster XL gives better accuracy when classifying large amount of debts data as compared to few records. Hence for cases of institutions with few debt data, it is recommended that this data be combined with other data from other institutions in order to achieve higher accuracy in the classification. It is recommended that at least 1,500 records for any single classification run be used. This will ensure at least over 80% accuracy according to the finding in the study.

5.4 SUGGESTIONS FOR FURTHER WORK

- One of its limitations of the study was the few number of independent variables used for prediction. For future study, it will be good to include for analysis more input variables e.g. age of loan at liquidation, employment status, availability of security, availability of reliable documentation, gender, age of debtor, loan amount and any other relevant variable that can be availed or deduced. This will however call for DPFB to ensure that the mentioned variables are availed or captured at liquidation.
- It will also be interesting to explore whether debtors in full time employment are more likely to pay their debts than debtors in business whose income can be irregular. This will require a further variable in this light.
- The DSS that was developed utilized a windows environment that handled the user interface. A web based development would more robust. With improved telecommunications, faster and more powerful computers such a system can be improved to make use of the World Wide Web.
- Although the results obtained from this study could be generalized, the study strictly focused on institutions in liquidation in DPFB department of Central Bank of Kenya. This study can be extended to the Deposit Insurance Schemes (DISs) in Africa as their functions on debt recovery a quite similar to the one in DPFB - Kenya.
- Further this study can be extended to the Eastern and the Western World. It would be interesting to identify the variations between Africa, Eastern world and the Western world.

REFERENCES

Text Books and Articles

Afifi, A.A, and Clark, V., (1990), *Computer-Aided Multivariate Analysis,* Van Nostrand Reinhold Co., New York.

Anonymous, (1995), *"Visa Stamps on Fraud"*, International Journal of Retail and Distribution Management,

Back B., Laitinen, T., and Sere, K.,, (1996.) *"Neural Networks and Genetic Algorithms for Bankruptcy Predictions,"* Expert Systems with Applications, Vol. 11, No. 4, 407-413

Barney, D.K., Graves, O.F., and Johnson, J.D., (1999). *"The Farmers Home Administration and Farm Debt Failure Prediction,"* Journal of Accounting and Public Policy, Vol. 18, 99-139,

Berry, J., (1995.) *"A Potent New Tool for Selling: Database Marketing"*, Business Week, Iss. 3388, pg. 56,

Berry, M.J.A., and Linoff, G.S., (1997). *Data Mining Techniques for Marketing, Sales, and Customer Support,* John Wiley & Sons, Inc,

Desai, V.S., Crook, J.N., and Overstreet, G.A. Jr., (1996)."A *Comparison of Neural Networks and Linear Scoring Models in the Credit Union Environment,"* European Journal of Operation Research, Vol. 95, 24-37

Dhar, V., and Stein, R., (1997) *Seven Methods for Transforming Corporate Data Into Business Intelligence,* Prentice Hall,.

Fayyad, U.S., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R. (1996). (Eds.), *Advances in Knowledge Discovery and Data Mining,* AAAI Press / The MIT Press, Menlo Park, California, USA.

Giudici, P., (2003) *Applied Data Mining: Statistical Methods for Business and Industry,* John Wiley & Sons, Chichester, West Sussex, England

Glorfeld, L.W., and Hardgrave, B.C., (1996) *"An Improved Method for Developing Neural Networks: The Case of Evaluating Commercial Loan Credit Worthiness,"* Computer & Operations Research, Vol. 23, No. 10, 933-944.

Hagan, M.T., Demuth, H.B., and Beale, M., (1996) Neural Network Design, PWS Publishing Company.

Han, J., and Kamber, M., (2001) *Data Mining: Concepts and Techniques,* Morgan Kaufmann Publishers: San Francisco, CA,

Jain, B.A., and Nag, B.N., (1997) *"Performance Evaluation of Neural Network Models,"* Journal of Management Information Systems, Vol. 14, No. 2, 201-216.

Jo, H., and Han, I., (1997) *"Bankruptcy Prediction Using Case-Based Reasoning, Neural Networks, and Discriminant Analysis,"* Expert Systems with Applications, Vol. 13, No. 2, 97-108.

Jones, C. (1991), *"An introduction to graph-based modeling systems, part II: graph-grammars and the implementation"*, ORSA Journal on Computing, Vol. 3, pp. 180-206.

Kantardzic, M., (2003) *Data Mining: Concepts, Models, Methods, and Algorithms,* IEEE Press/Wiley.

Kumar, N., Krovi, R., and, Rajagopalan, B., (1997.) *"Financial Decision Support with Hybrid Genetic and Neural Based Modeling Tools,"* European Journal of Operation Research, Vol. 103, 339-349.

Lee, K.C., Han, I., and Kwon, Y., (1996) *"Hybrid Neural Network for Bankruptcy Predictions,"* Decision Support Systems, 18, 63-72.

Mitchell, T.M., (1997*) Machine Learning,* WCB/McGraw-Hill, Boston, Massachusetts.

Phillip H. Sherrod (2003-2010), *"DTREG Predictive Modeling Software"*

Piramuthu, S., (1999) *"Financial Credit-Risk Evaluation with Neural and Neurofuzzy Systems,"* European Journal of Operational Research, Vol. 112, 310-321,.

Punch, L., (1994.) *"When Big Brother Goes to Far",* Credit Card Management, Vol. 7, Iss. 7, pg. 22

Pyle, D., (2003) *Business Modeling and Data Mining,* Morgan Kaufman Publishers (An Imprint of Elsevier Science), San Francisco, California.

Tessmer, A.C., (1997*) "What to Learn from Near Misses: An Inductive learning Approach to Credit Risk Assessment,"* Decision Sciences, Vol. 28, No. 1, 105-120.

Thomas, L.C., (2000) *"A Survey of Credit and Behavioral Scoring: Forecasting Financial Risk of Lending to Consumers,"* International Journal of Forecasting, Vol. 16, 149-172.

West, D., (2000.) *"Neural Network Credit Scoring Models,"* Computers & Operations Research, Vol. 27, 1131-1152,

Zurada, J., and Zurada, M., (2002) *Data Mining Techniques in Predicting Default Rates on Customer Loans"*, Review of Business Information Systems, Vol. 6, No. 3, pp. 65-83

Websites

http://174.123.20.5/download/alyuda-forecaster-xl-1-0-22147.html

http://www.alyuda.com/neuralnetworks.htm by Alyuda Research, LLC (2001-2011)

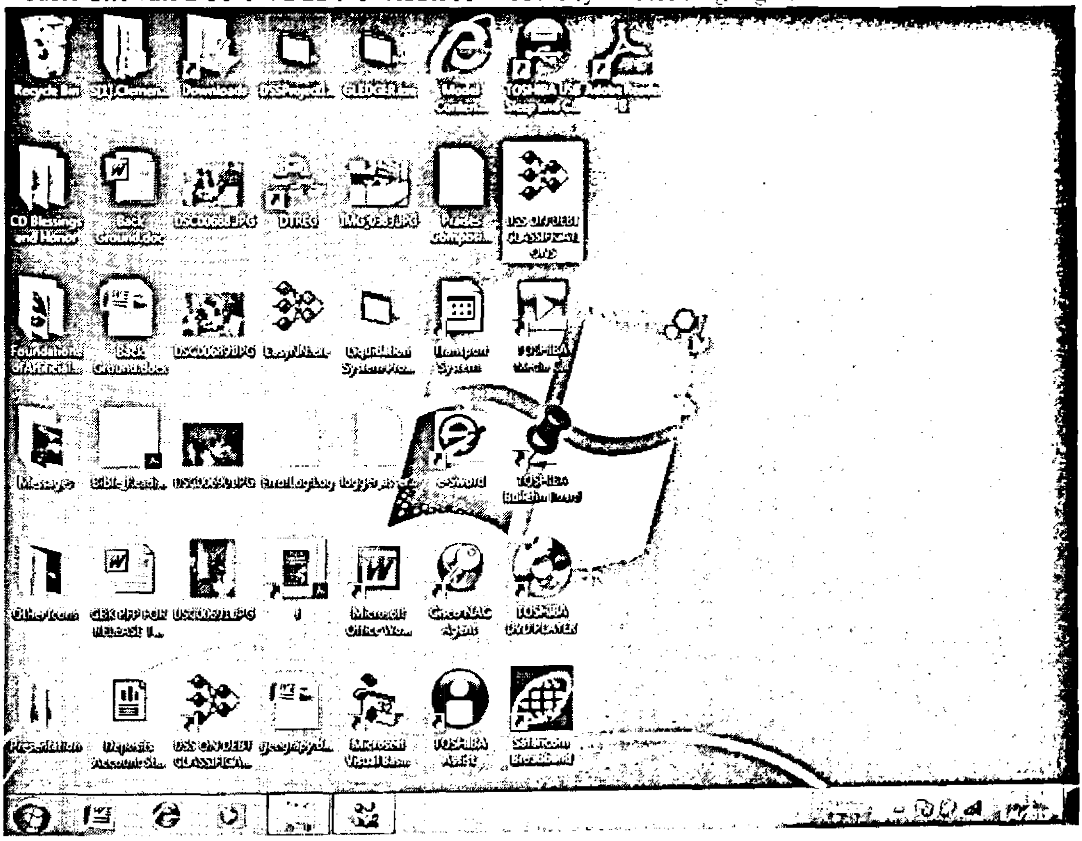
http://www.dtreg.com by Phillip H. Sherrod 2003-2010

http://www.dtreg.com/DownloadDemo.htm by Phillip H. Sherrod 2003-2010

http://fileforum.betanews.com/detail/Alyuda-Forecaster-XL/1050645806/1

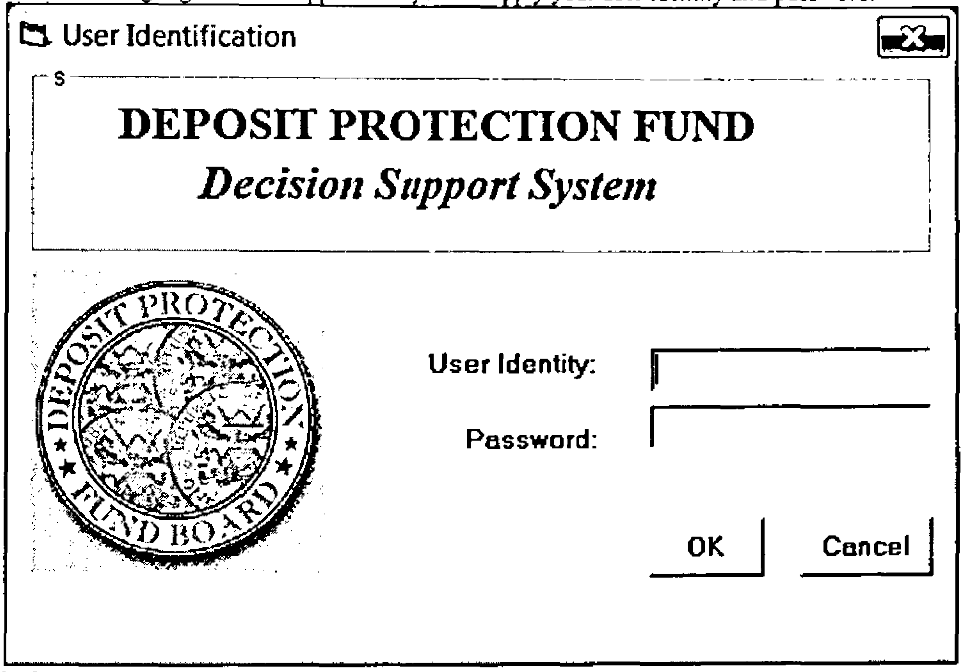
APPENDICES
APPENDIX I: 4.6 A DECISION SUPPORT SYSTEM PROTOTYPE USER MANUAL

1. Double Click the DSS ON DEBT CLASSIFICATIONS system icon highlighted below:



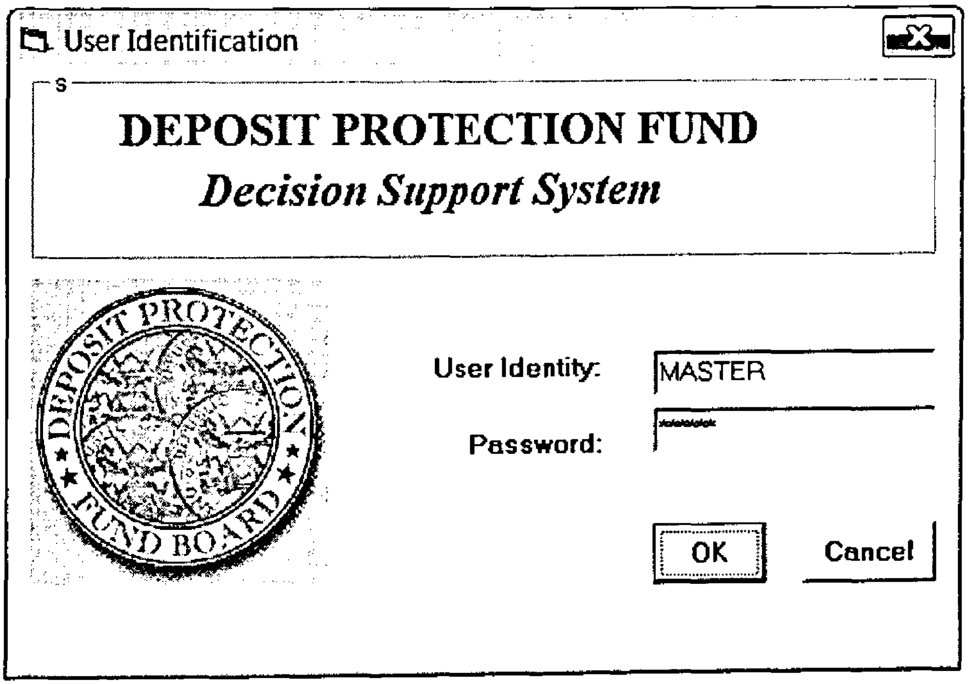
Shortcut to DSS

2. The following logon screen appears for you to supply your user identity and password:



Login Screen

3. Enter your user identity and password and click the OK button.

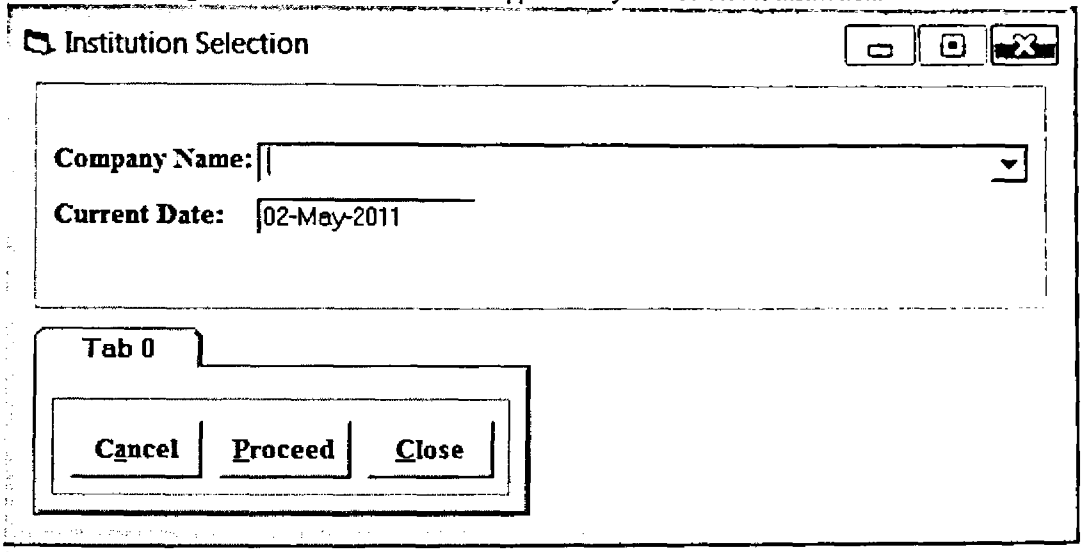


User Identity: MASTER

Password: ********

Login Screen with Credentials

4. The following institution selection window appears for you to select the institution:



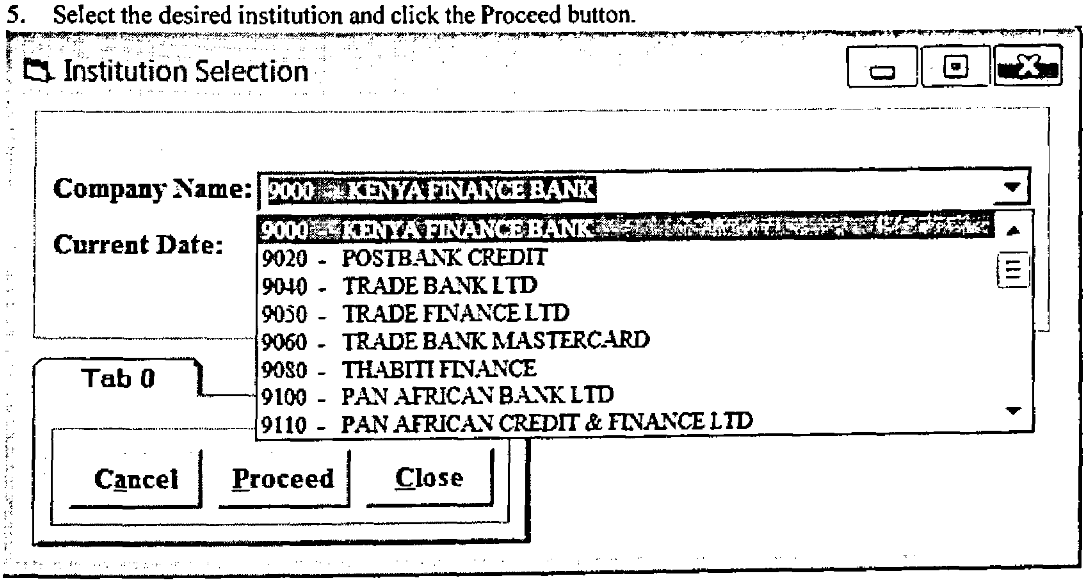
Institution Selection

Company Name:

Current Date: 02-May-2011

Institution Selection Dialogue

5. Select the desired institution and click the Proceed button.



Institution Selection

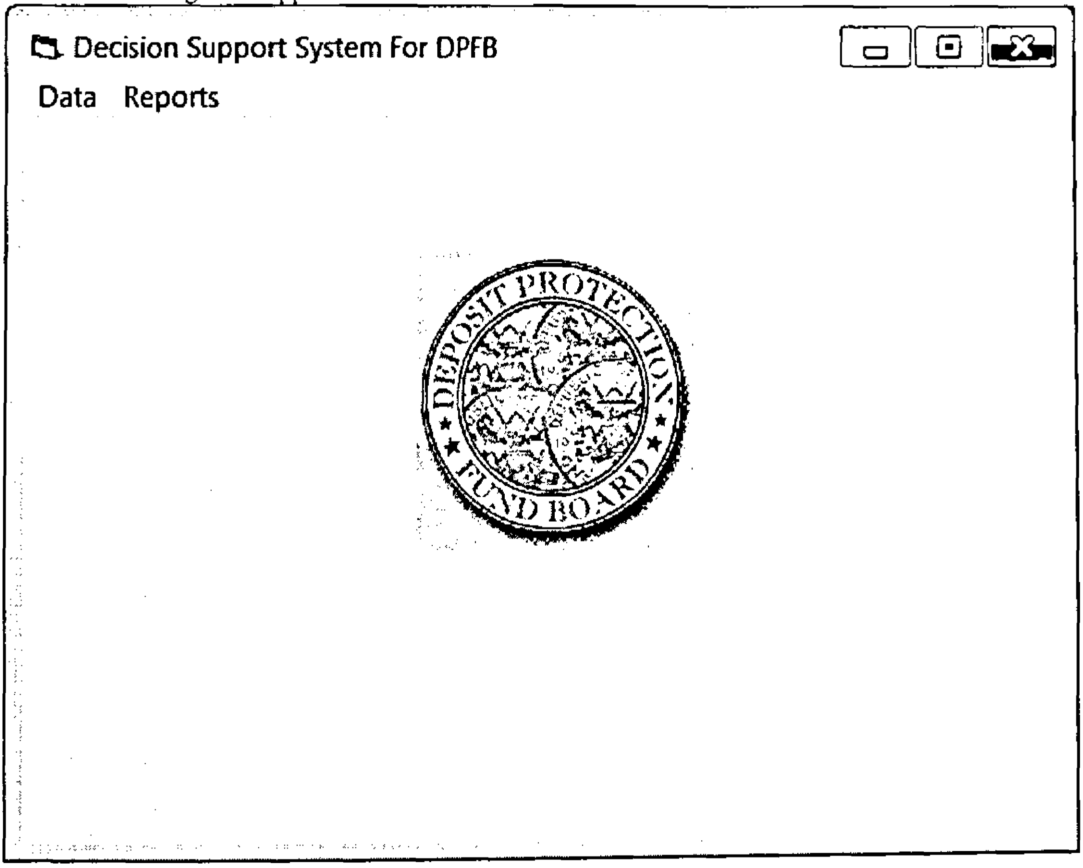
Company Name: 9000 - KENYA FINANCE BANK

Current Date:

9000 - KENYA FINANCE BANK
9020 - POSTBANK CREDIT
9040 - TRADE BANK LTD
9050 - TRADE FINANCE LTD
9060 - TRADE BANK MASTERCARD
9080 - THABITI FINANCE
9100 - PAN AFRICAN BANK LTD
9110 - PAN AFRICAN CREDIT & FINANCE LTD

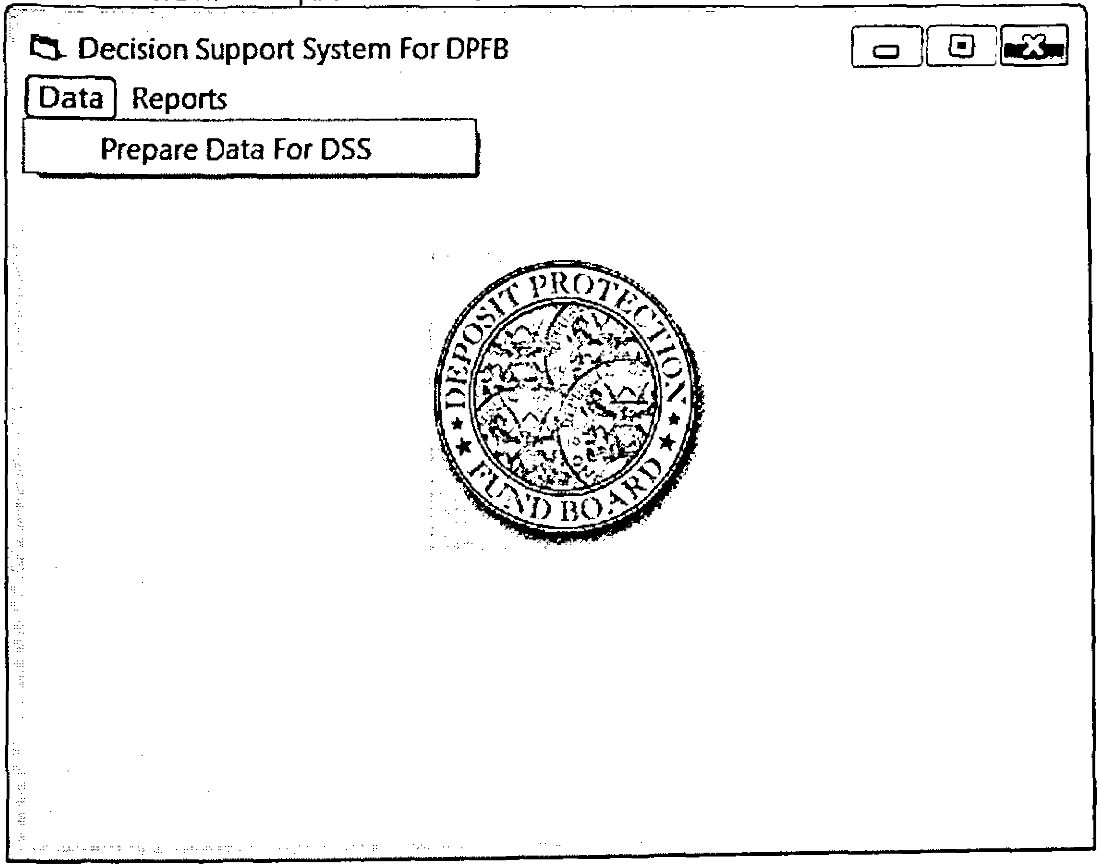
Institution Selection Dialogue with Institution List

6. The following menu appears:


Decision Support System For DPFB
Data
Reports
DEPOSIT PROTECTION FUND BOARD

Main Menu

7. Select Data ~> Prepare Data for DSS:

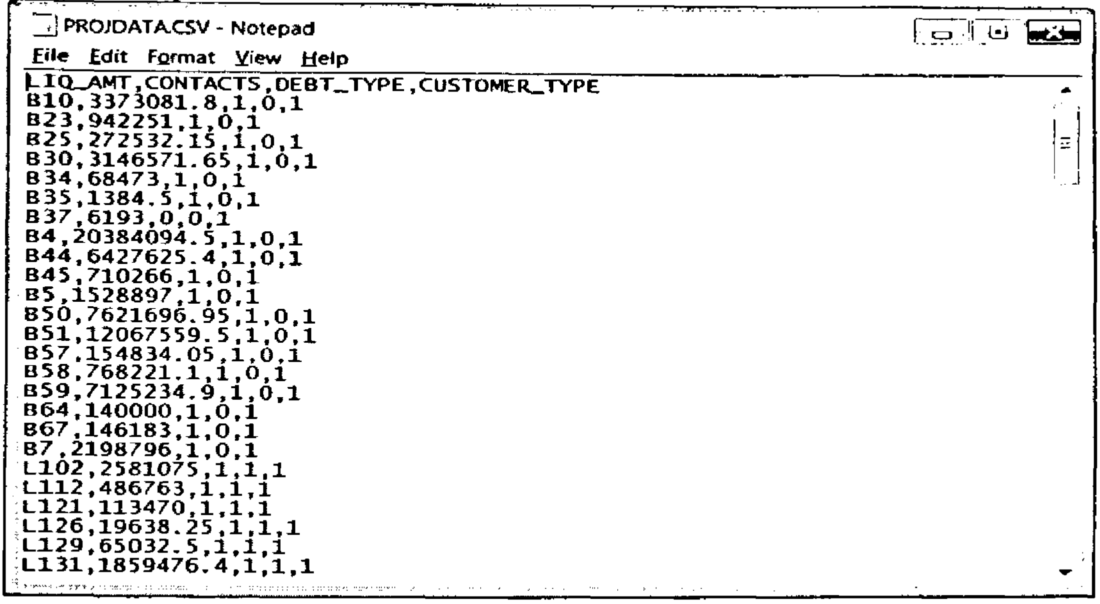

Decision Support System For DPFB
Data
Reports
Prepare Data For DSS
DEPOSIT PROTECTION FUND BOARD

Main Menu with Selection

8. The following steps will happen at this stage:

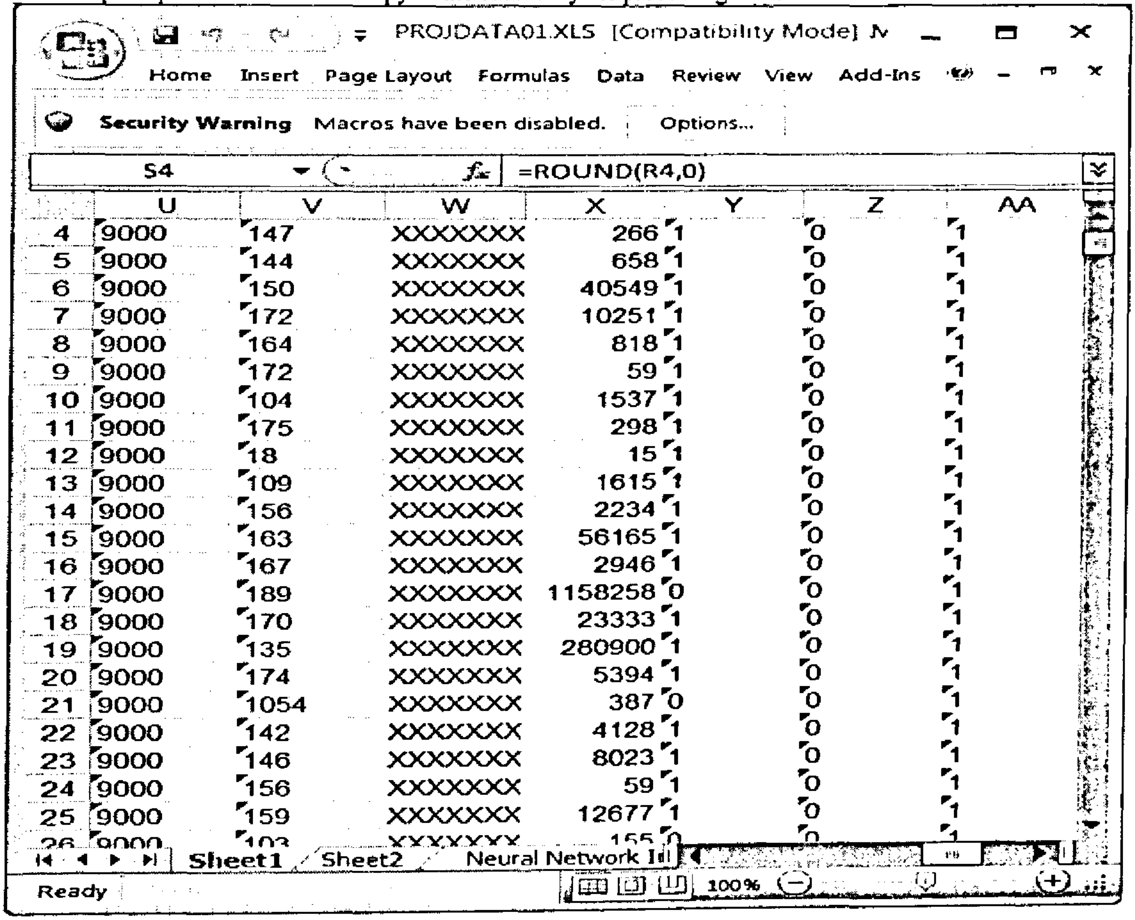
- The system access the raw data in the loans table and prepare the DSS data:
    - It prepares CSV data file that looks as follows:

CSV Data Output File



```
LIQ_AMT,CONTACTS,DEBT_TYPE,CUSTOMER_TYPE
B10,3373081.8,1,0,1
B23,942251,1,0,1
B25,272532.15,1,0,1
B30,3146571.65,1,0,1
B34,68473,1,0,1
B35,1384.5,1,0,1
B37,6193,0,0,1
B4,20384094.5,1,0,1
B44,6427625.4,1,0,1
B45,710266,1,0,1
B5,1528897,1,0,1
B50,7621696.95,1,0,1
B51,12067559.5,1,0,1
B57,154834.05,1,0,1
B58,768221.1,1,0,1
B59,7125234.9,1,0,1
B64,140000,1,0,1
B67,146183,1,0,1
B7,2198796,1,0,1
L102,2581075,1,1,1
L112,486763,1,1,1
L121,113470,1,1,1
L126,19638.25,1,1,1
L129,65032.5,1,1,1
L131,1859476.4,1,1,1
```

It also open up excel and loads a copy of the data ready for processing. This data looks as follows:

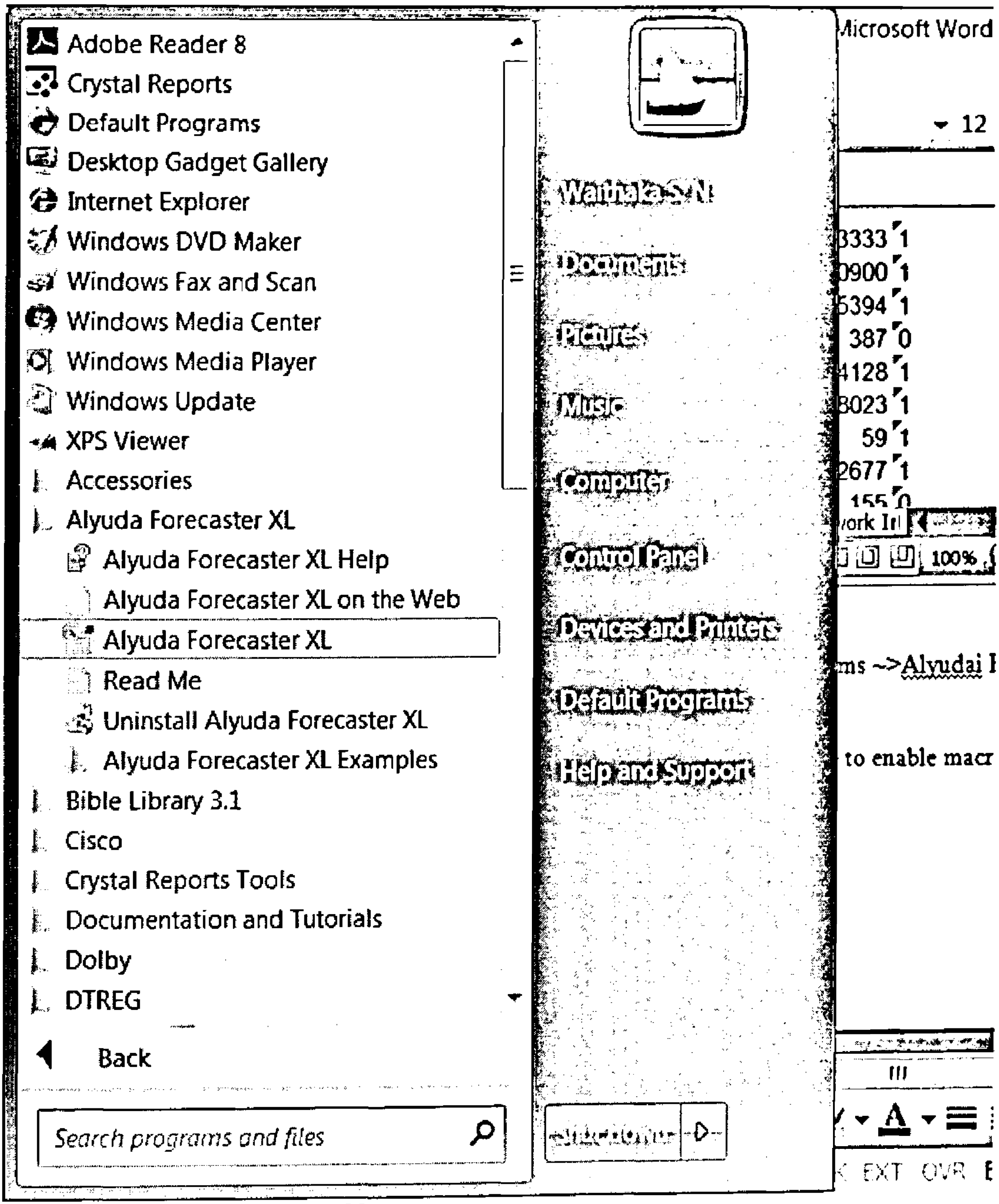


	U	V	W	X	Y	Z	AA
4	9000	147	XXXXXXX	266	1	0	1
5	9000	144	XXXXXXX	658	1	0	1
6	9000	150	XXXXXXX	40549	1	0	1
7	9000	172	XXXXXXX	10251	1	0	1
8	9000	164	XXXXXXX	818	1	0	1
9	9000	172	XXXXXXX	59	1	0	1
10	9000	104	XXXXXXX	1537	1	0	1
11	9000	175	XXXXXXX	298	1	0	1
12	9000	18	XXXXXXX	15	1	0	1
13	9000	109	XXXXXXX	1615	1	0	1
14	9000	156	XXXXXXX	2234	1	0	1
15	9000	163	XXXXXXX	56165	1	0	1
16	9000	167	XXXXXXX	2946	1	0	1
17	9000	189	XXXXXXX	1158258	0	0	1
18	9000	170	XXXXXXX	23333	1	0	1
19	9000	135	XXXXXXX	280900	1	0	1
20	9000	174	XXXXXXX	5394	1	0	1
21	9000	1054	XXXXXXX	387	0	0	1
22	9000	142	XXXXXXX	4128	1	0	1
23	9000	146	XXXXXXX	8023	1	0	1
24	9000	156	XXXXXXX	59	1	0	1
25	9000	159	XXXXXXX	12677	1	0	1
26	9000	103	XXXXXXX	155	0	0	1

Training Data in Excel

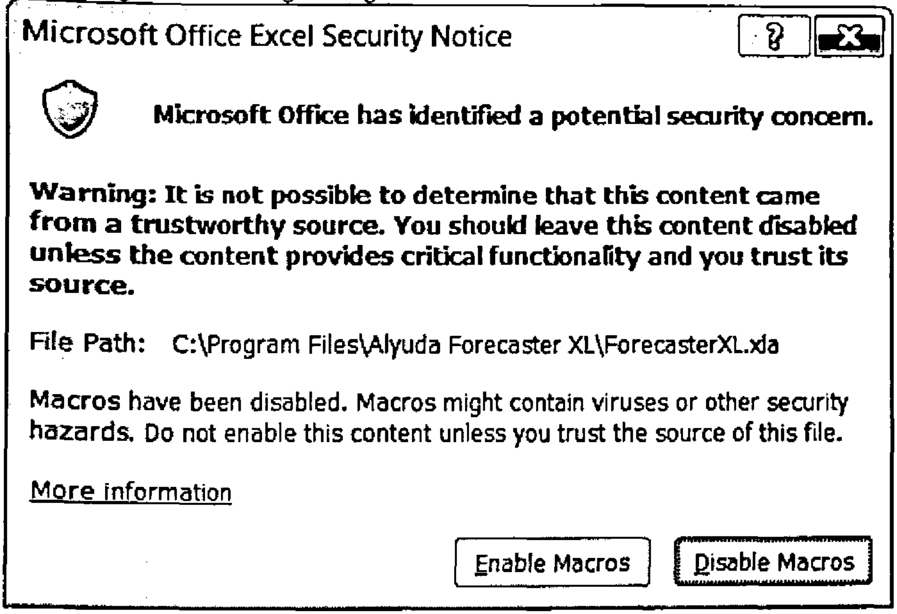
9. Load the Alyuda Forecaster XL by:

- Click the start button ~> All programs ~>Alyuda Forecaster XL ~> Alyuda Forecaster XL as shown below:



Selecting Alyuda Forecaster from Menu

You will get the following message to enable macros:



Microsoft Office Excel Security Notice

Microsoft Office has identified a potential security concern.

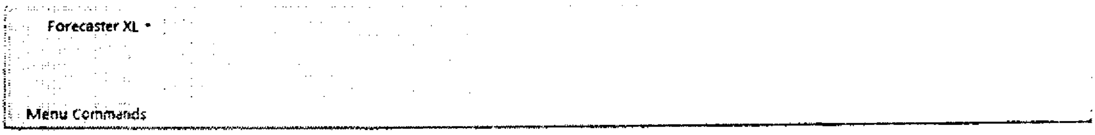
Warning: It is not possible to determine that this content came from a trustworthy source. You should leave this content disabled unless the content provides critical functionality and you trust its source.

File Path: C:\Program Files\Alyuda Forecaster XL\ForecasterXL.xla

Macros have been disabled. Macros might contain viruses or other security hazards. Do not enable this content unless you trust the source of this file.

Enable/ Disable Macros – MS Excel Security Notice

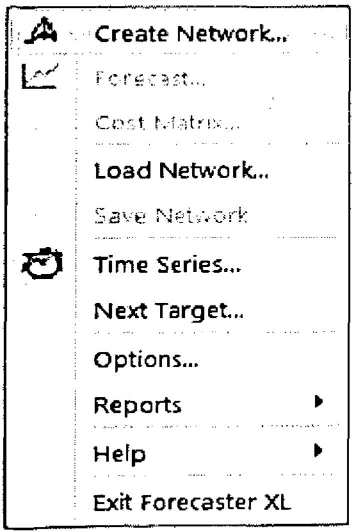
10. Click on enable macros.
11. Alyuda Forecaster XL appear as a part of the menu as shown below:



Appearance of Alyuda Forecaster XL in MS Excel

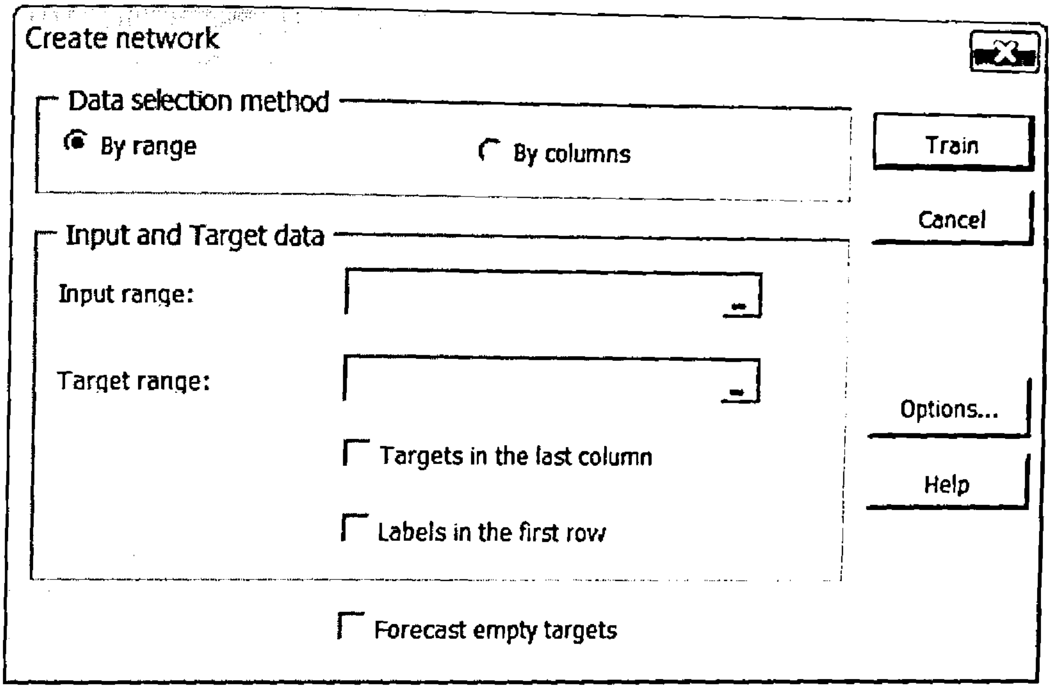
<u>Training</u>

12. Click on Alyuda Forecaster XL and select on Create New Network as shown below:



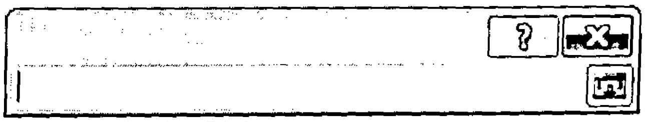
Alyuda Forecaster Menu

13. You get the following dialogue box:



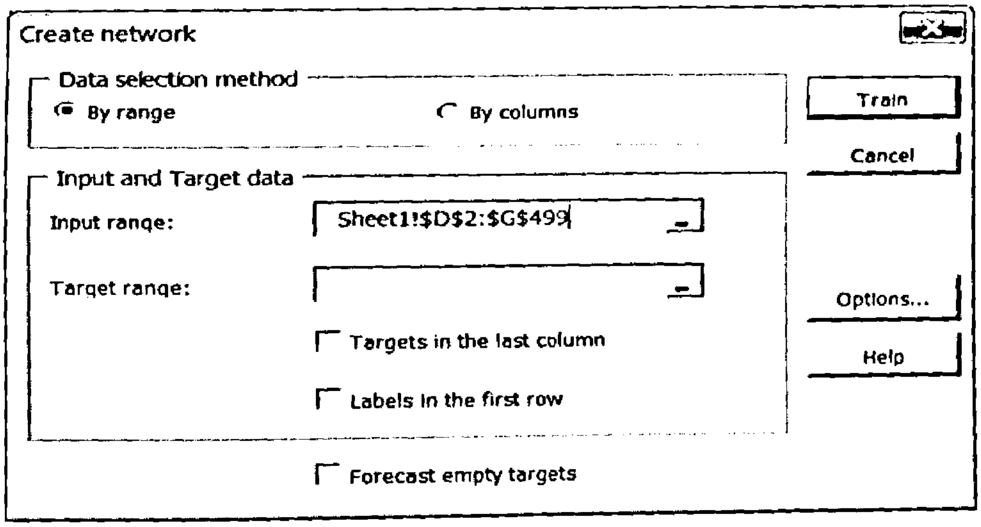
Input and Target Data Selection Dialogue

14. Click on the input range. The following dialogue appears awaiting you to select the input range on the Excel on the input set:



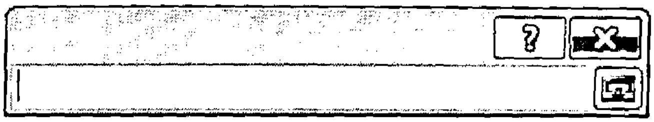
Input Range Selection Dialogue

15. Select the input range in Excel and click the button at right. The range appear as shown below:


Input range: Sheet1!$D$2:$G$499

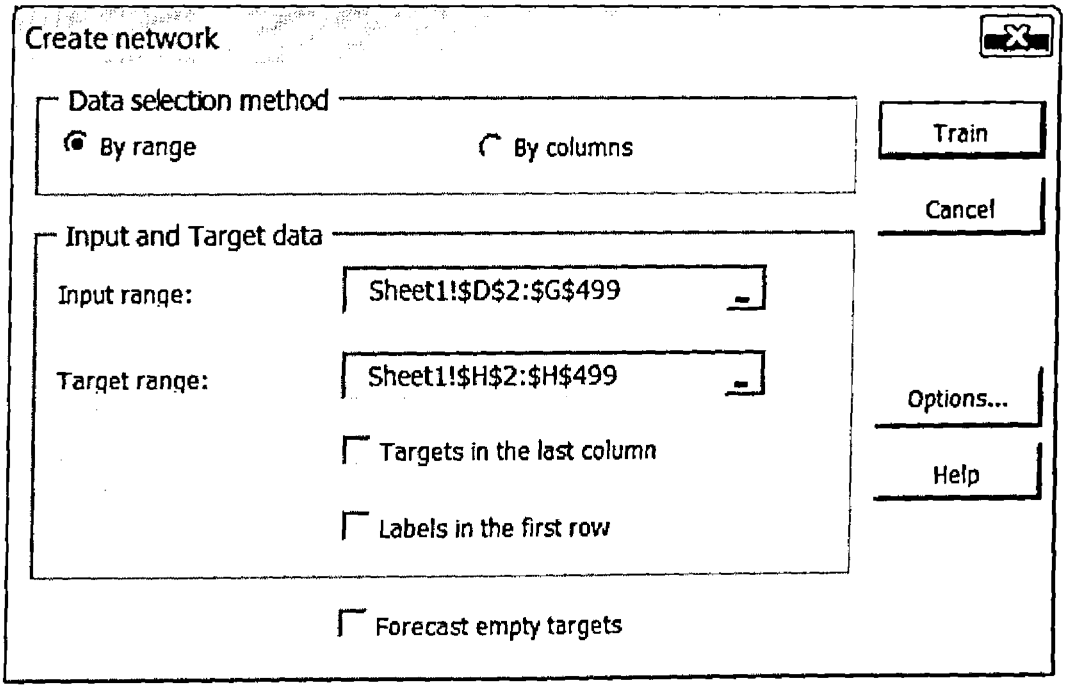
Selected Input Range

16. Click on the output range button. The following dialogue appear awaiting you to select the output range on the Excel on the training set:



Target Range Selection Dialogue

17. Select the output range in Excel and click the button at right. The range appear as shown below:


Create network

Data selection method

● By range ○ By columns

Input and Target data

Input range: Sheet1!$D$2:$G$499

Target range: Sheet1!$H$2:$H$499

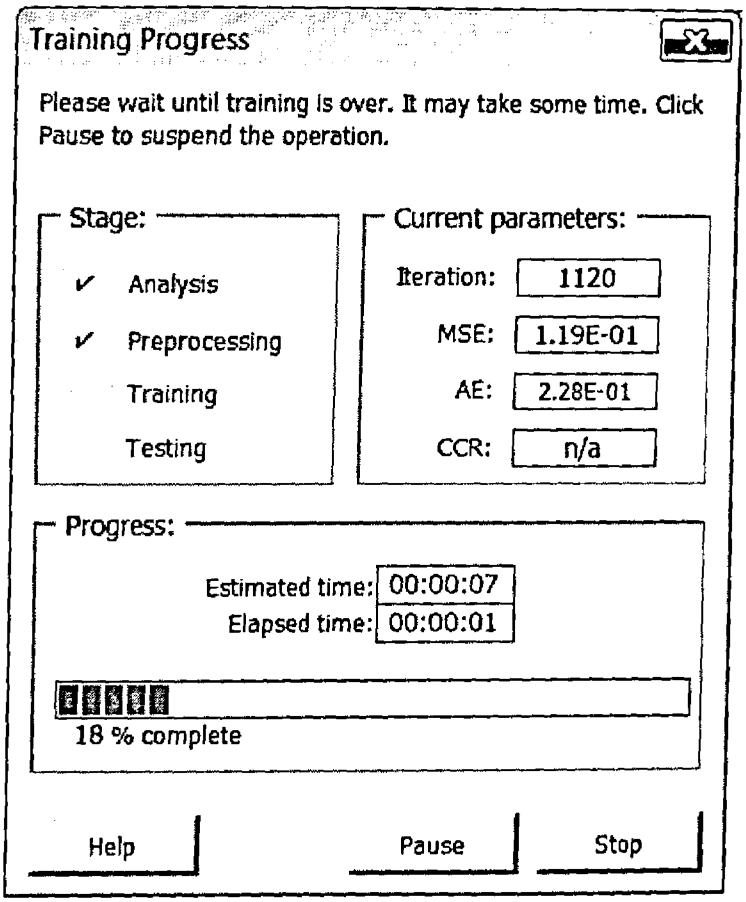
□ Targets in the last column

□ Labels in the first row

□ Forecast empty targets

Selected Input and Target Range

18. Click on the Train Button to train the model. The following dialogue appears showing training progress.



Training Progress

Please wait until training is over. It may take some time. Click Pause to suspend the operation.

Stage:

✓ Analysis

✓ Preprocessing

Training

Testing

Current parameters:

Iteration: 1120

MSE: 1.19E-01

AE: 2.28E-01

CCR: n/a

Progress:

Estimated time: 00:00:07

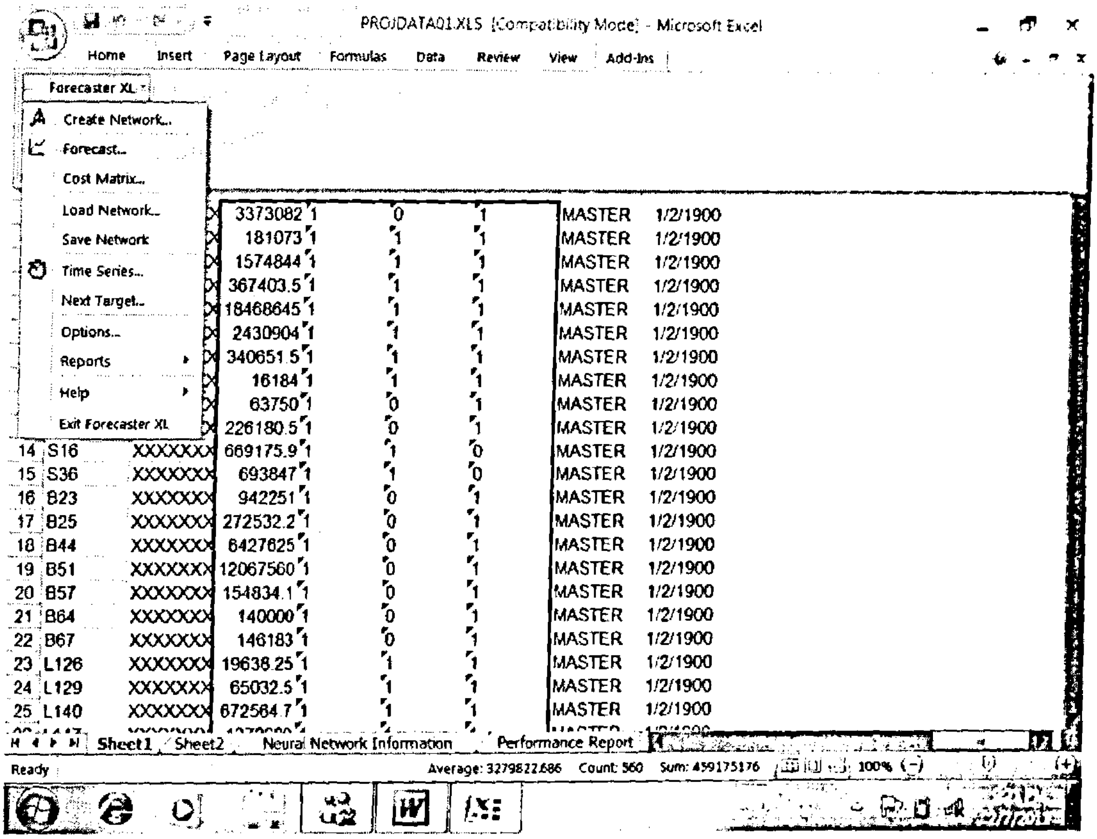
Elapsed time: 00:00:01

18 % complete

Training Progress Dialogue

Classification

19. Select Forecast on the Forecaster XL menu as follows:

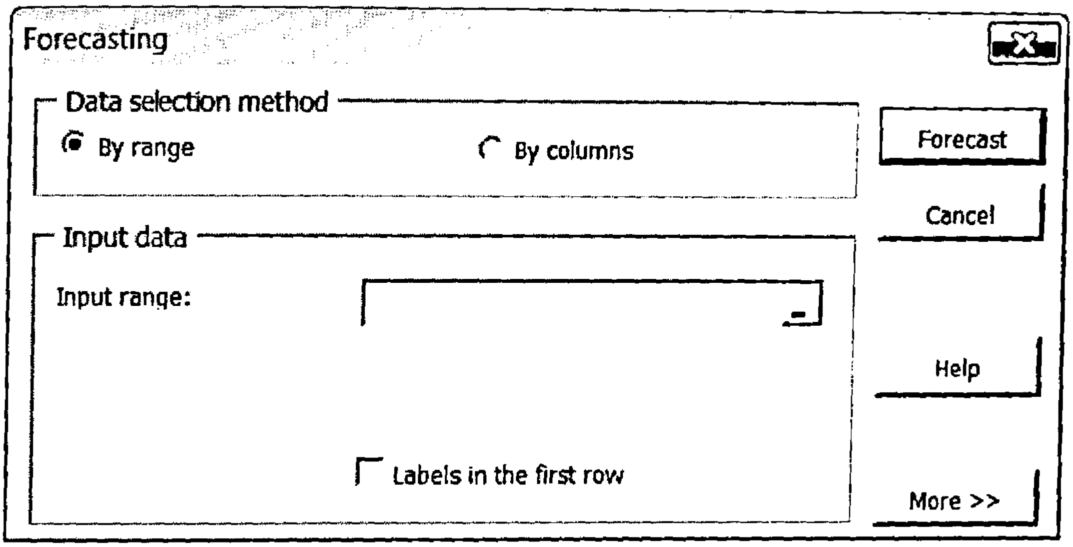


			3373082	1	0	1	MASTER	1/2/1900
			181073	1	1	1	MASTER	1/2/1900
			1574844	1	1	1	MASTER	1/2/1900
			367403.5	1	1	1	MASTER	1/2/1900
			18468645	1	1	1	MASTER	1/2/1900
			2430904	1	1	1	MASTER	1/2/1900
			340651.5	1	1	1	MASTER	1/2/1900
			16184	1	1	1	MASTER	1/2/1900
			63750	1	0	1	MASTER	1/2/1900
			226180.5	1	0	1	MASTER	1/2/1900
14	S16	XXXXXXX	669175.9	1	1	0	MASTER	1/2/1900
15	S36	XXXXXXX	693847	1	1	0	MASTER	1/2/1900
16	B23	XXXXXXX	942251	1	0	1	MASTER	1/2/1900
17	B25	XXXXXXX	272532.2	1	0	1	MASTER	1/2/1900
18	B44	XXXXXXX	6427625	1	0	1	MASTER	1/2/1900
19	B51	XXXXXXX	12067560	1	0	1	MASTER	1/2/1900
20	B57	XXXXXXX	154834.1	1	0	1	MASTER	1/2/1900
21	B64	XXXXXXX	140000	1	0	1	MASTER	1/2/1900
22	B67	XXXXXXX	146183	1	0	1	MASTER	1/2/1900
23	L126	XXXXXXX	19638.25	1	1	1	MASTER	1/2/1900
24	L129	XXXXXXX	65032.5	1	1	1	MASTER	1/2/1900
25	L140	XXXXXXX	672584.7	1	1	1	MASTER	1/2/1900

Average: 3279822.686 Count: 560 Sum: 459175176

Forecaster Selection from Alyuda Forecaster Menu

20. The following dialogue appears:



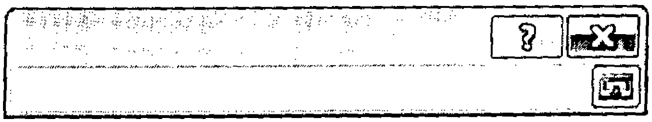
Data selection method: ● By range ○ By columns

Input range:

□ Labels in the first row

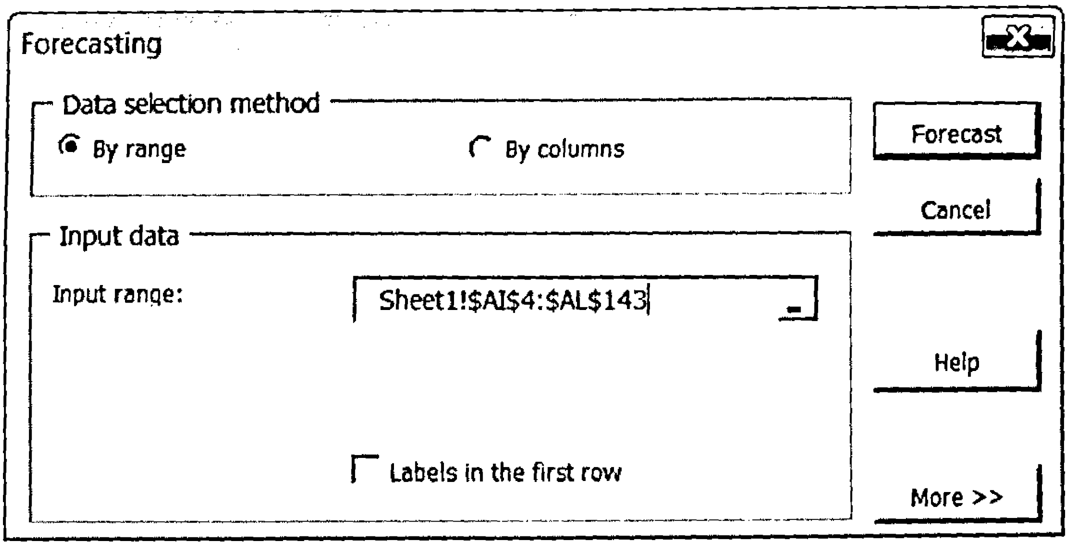
Forecasting Input Data Selection Dialogue

21. Select the range on the right. The following dialogue appears:



Forecasting Input Data Selection Dialogue

22. Select input range on the classification set and then click the button at the right. The following dialogue appears:



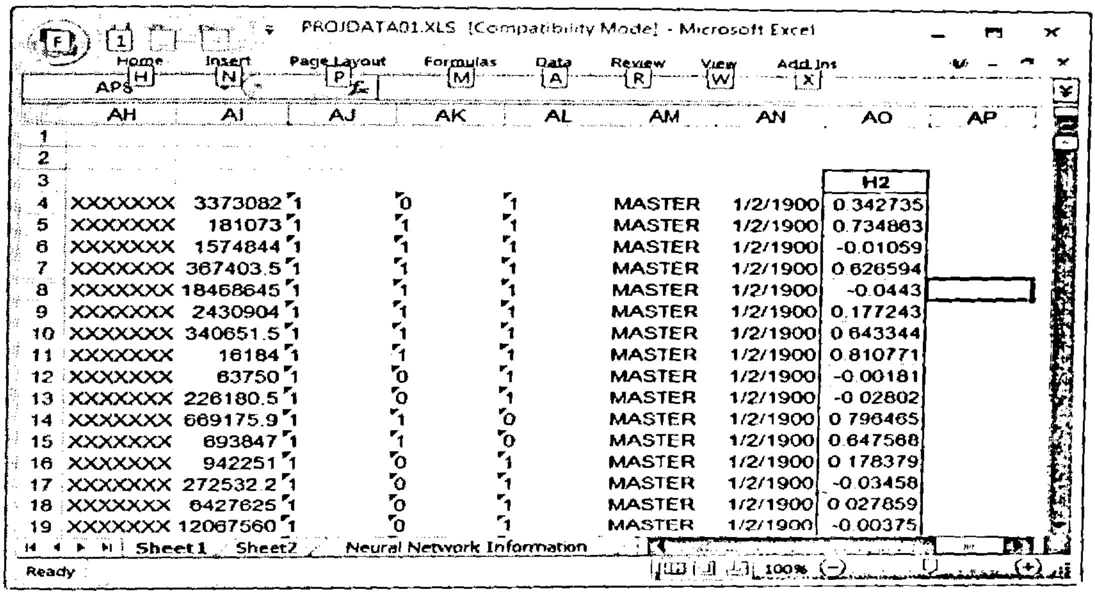
Data selection method: ● By range ○ By columns

Input range: Sheet1!$AI$4:$AL$143

□ Labels in the first row

Selected Input Range

23. Click forecast button to forecast. The Forecasted output appears on the right on a green column as shown below:

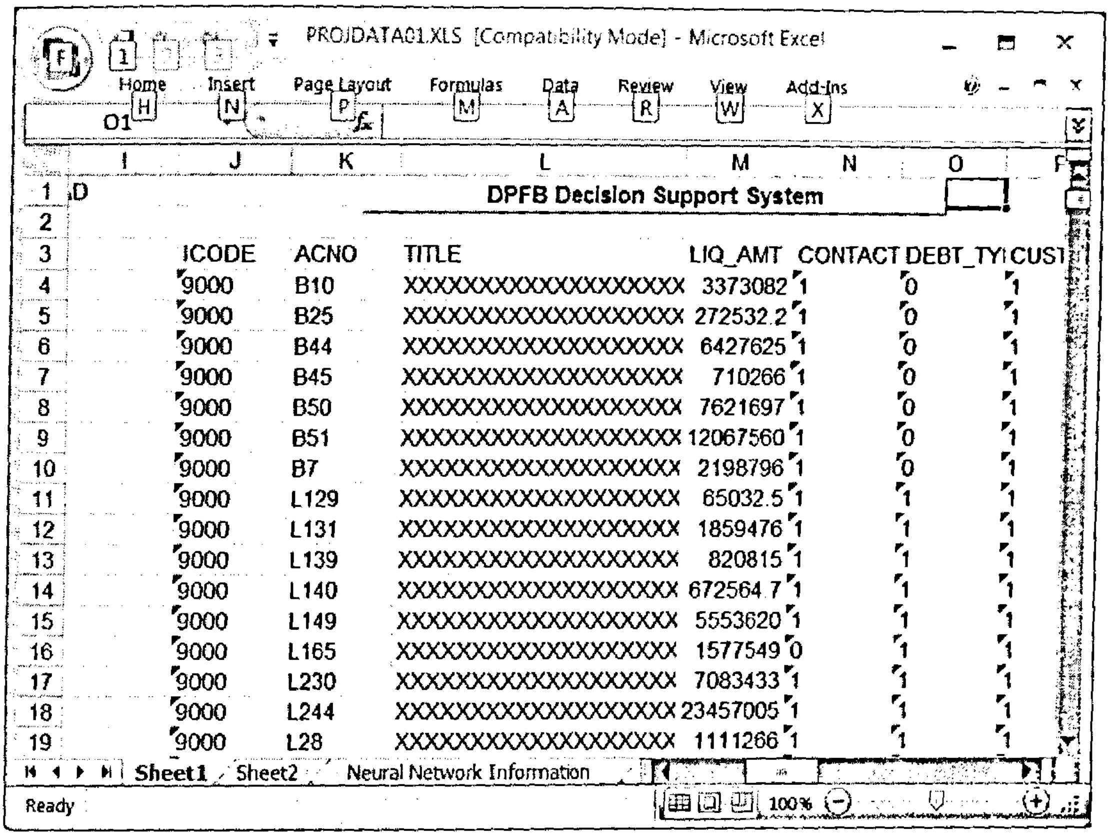


	AH	AI	AJ	AK	AL	AM	AN	AO	AP
1									
2									
3								H2	
4	XXXXXXX	3373082	1	0	1	MASTER	1/2/1900	0.342735	
5	XXXXXXX	181073	1	1	1	MASTER	1/2/1900	0.734863	
6	XXXXXXX	1574844	1	1	1	MASTER	1/2/1900	-0.01059	
7	XXXXXXX	367403.5	1	1	1	MASTER	1/2/1900	0.626594	
8	XXXXXXX	18468645	1	1	1	MASTER	1/2/1900	-0.0443	
9	XXXXXXX	2430904	1	1	1	MASTER	1/2/1900	0.177243	
10	XXXXXXX	340651.5	1	1	1	MASTER	1/2/1900	0.643344	
11	XXXXXXX	16184	1	1	1	MASTER	1/2/1900	0.810771	
12	XXXXXXX	63750	1	0	1	MASTER	1/2/1900	-0.00181	
13	XXXXXXX	226180.5	1	0	1	MASTER	1/2/1900	-0.02802	
14	XXXXXXX	669175.9	1	1	0	MASTER	1/2/1900	0.796465	
15	XXXXXXX	693847	1	1	0	MASTER	1/2/1900	0.647568	
16	XXXXXXX	942251	1	0	1	MASTER	1/2/1900	0.178379	
17	XXXXXXX	272532.2	1	0	1	MASTER	1/2/1900	-0.03458	
18	XXXXXXX	6427625	1	0	1	MASTER	1/2/1900	0.027859	
19	XXXXXXX	12067560	1	0	1	MASTER	1/2/1900	-0.00375	

Forecasted Output

<u>Report Production</u>

24. Click DPFB Decision Support button at the top of the Excel sheet that is shown below to login into the system:

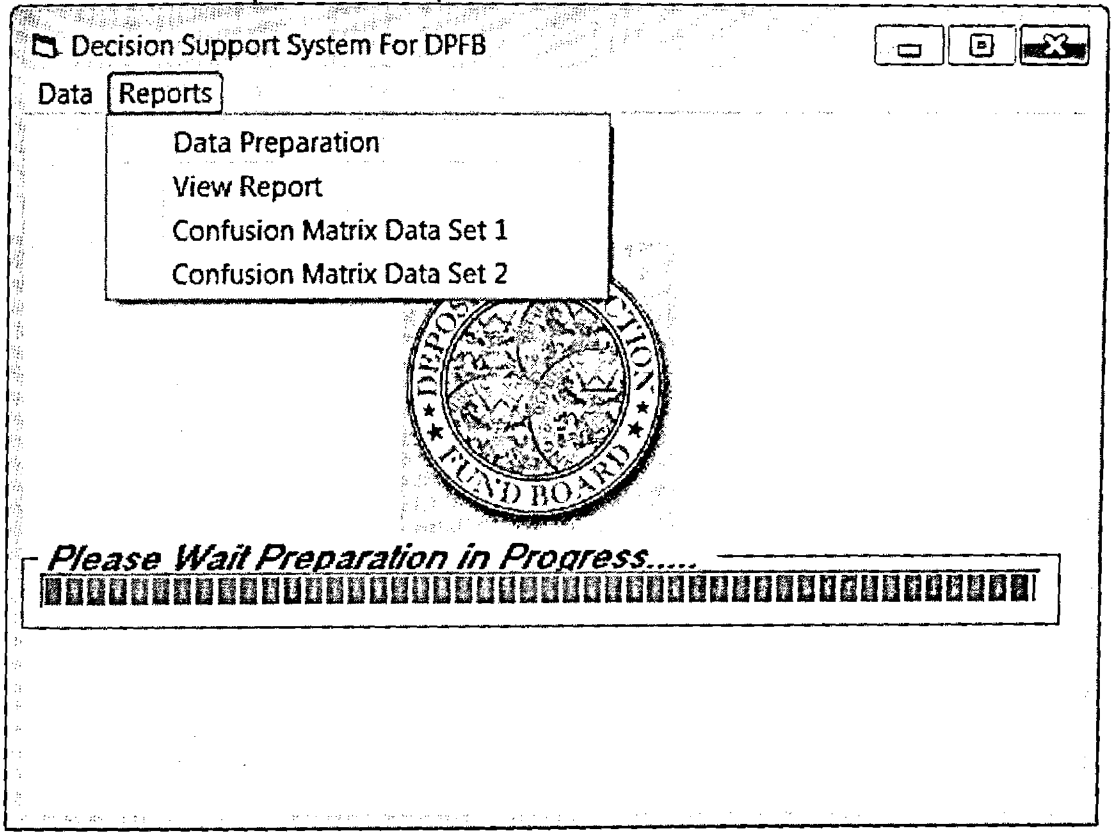

PROJDATA01.XLS [Compatibility Mode] - Microsoft Excel
Home
Insert
Page Layout
Formulas
Data
Review
View
Add-Ins
O1
Sheet1
Sheet2
Neural Network Information
Ready
100%

	I	J	K	L	M	N	O
1	D			DPFB Decision Support System			
2							
3		ICODE	ACNO	TITLE	LIQ_AMT	CONTACT	DEBT_TY	CUST
4		9000	B10	XXXXXXXXXXXXXXXXXXXX	3373082	1	0	1
5		9000	B25	XXXXXXXXXXXXXXXXXXXX	272532.2	1	0	1
6		9000	B44	XXXXXXXXXXXXXXXXXXXX	6427625	1	0	1
7		9000	B45	XXXXXXXXXXXXXXXXXXXX	710266	1	0	1
8		9000	B50	XXXXXXXXXXXXXXXXXXXX	7621697	1	0	1
9		9000	B51	XXXXXXXXXXXXXXXXXXXX	12067560	1	0	1
10		9000	B7	XXXXXXXXXXXXXXXXXXXX	2198796	1	0	1
11		9000	L129	XXXXXXXXXXXXXXXXXXXX	65032.5	1	1	1
12		9000	L131	XXXXXXXXXXXXXXXXXXXX	1859476	1	1	1
13		9000	L139	XXXXXXXXXXXXXXXXXXXX	820815	1	1	1
14		9000	L140	XXXXXXXXXXXXXXXXXXXX	672564.7	1	1	1
15		9000	L149	XXXXXXXXXXXXXXXXXXXX	5553620	1	1	1
16		9000	L165	XXXXXXXXXXXXXXXXXXXX	1577549	0	1	1
17		9000	L230	XXXXXXXXXXXXXXXXXXXX	7083433	1	1	1
18		9000	L244	XXXXXXXXXXXXXXXXXXXX	23457005	1	1	1
19		9000	L28	XXXXXXXXXXXXXXXXXXXX	1111266	1	1	1

Decision Support Link from Excel

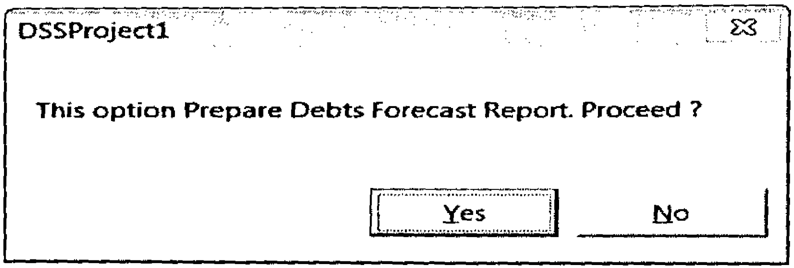
The login screen appears for your user identity and password.

25. Select the Data Preparation in the Reports menu as shown below:


Decision Support System For DPFB
Data
Reports
Data Preparation
View Report
Confusion Matrix Data Set 1
Confusion Matrix Data Set 2
DEPOSIT PROTECTION FUND BOARD
Please Wait Preparation in Progress.....

Data Preparation Menu

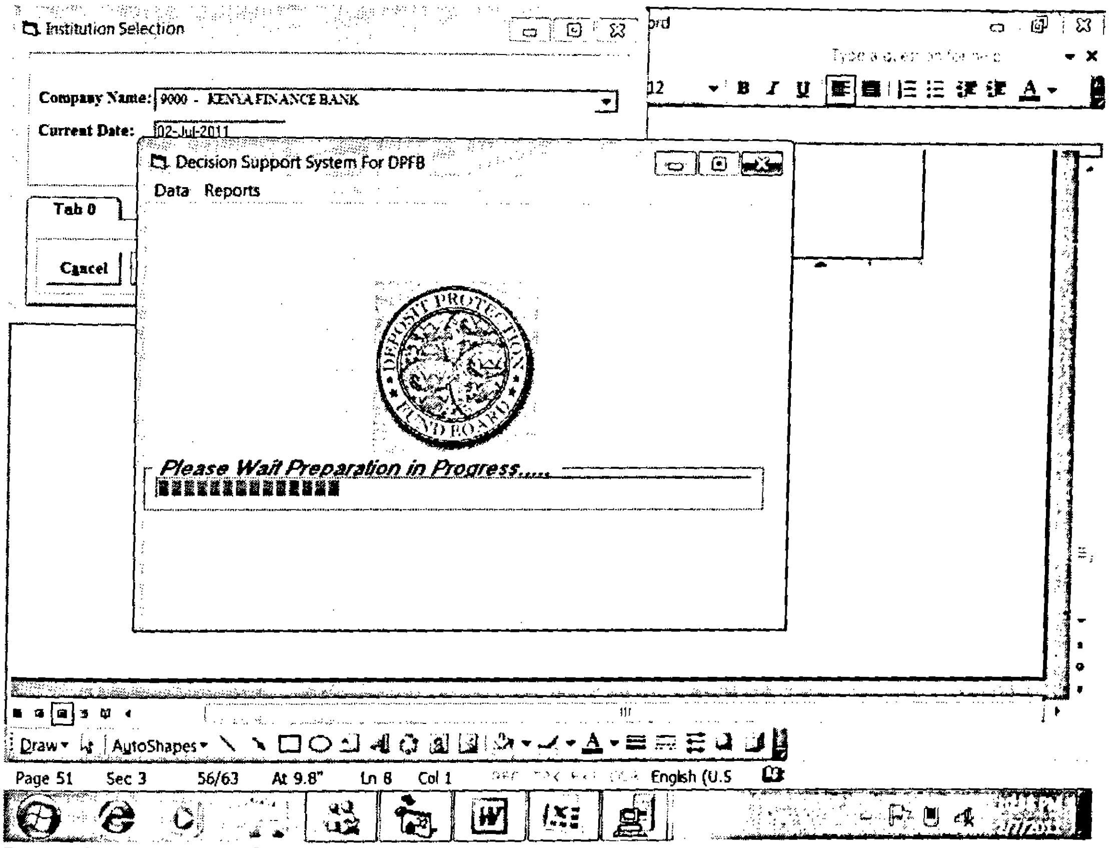
26. You get the following question to confirm whether to proceed:


DSSProject1

This option Prepare Debts Forecast Report. Proceed ?

Interactive Dialogue to Proceed in Report Preparation

The preparation will proceed as shown below:



Company Name: 9000 - KENYA FINANCE BANK

Current Date: 02-Jul-2011

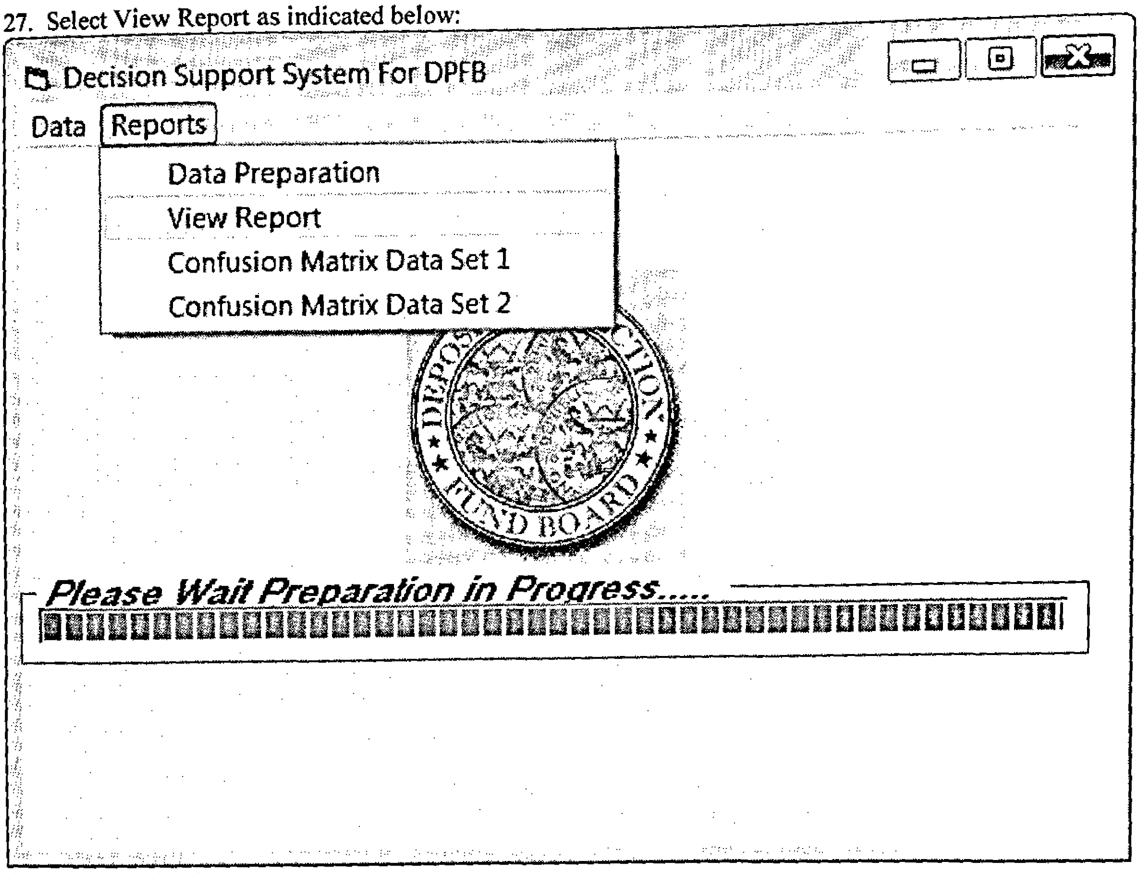
Please Wait Preparation in Progress.....

Report Preparation in Progress

At the end the following message will appear indicating completion of report data preparation:

Click OK.

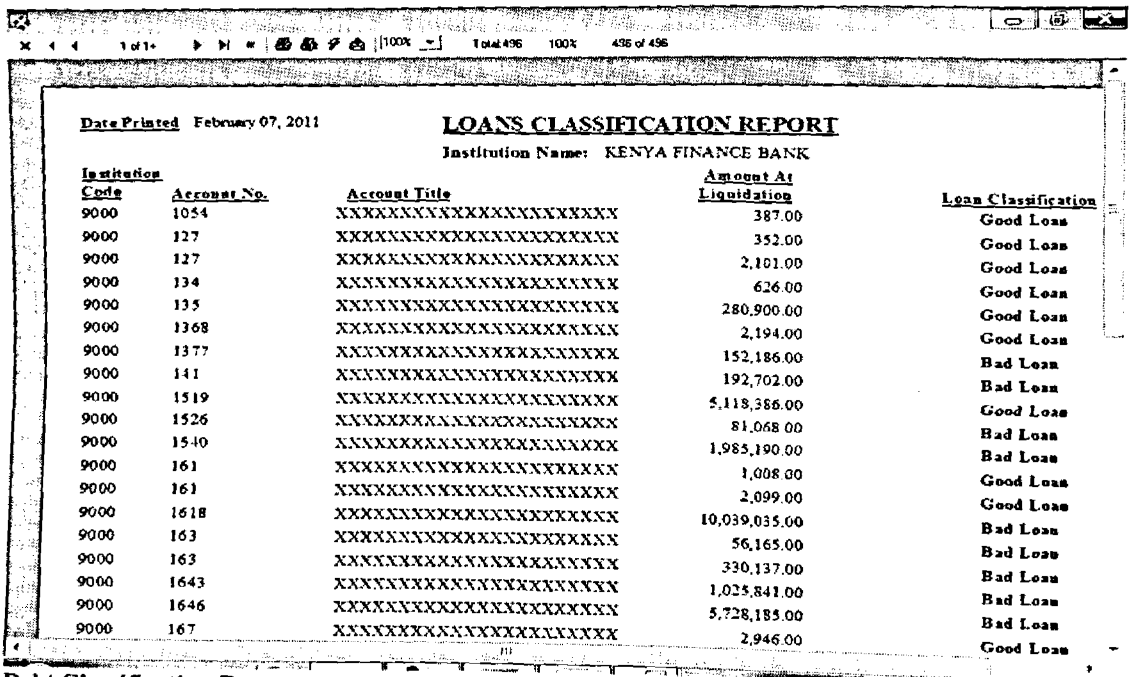
27. Select View Report as indicated below:


Decision Support System For DPFB
Data
Reports
Data Preparation
View Report
Confusion Matrix Data Set 1
Confusion Matrix Data Set 2
DEPOSIT PROTECTION FUND BOARD

Please Wait Preparation in Progress.....

Report View Option

28. The Decision Support report appears as shown below:



Date Printed February 07, 2011

LOANS CLASSIFICATION REPORT

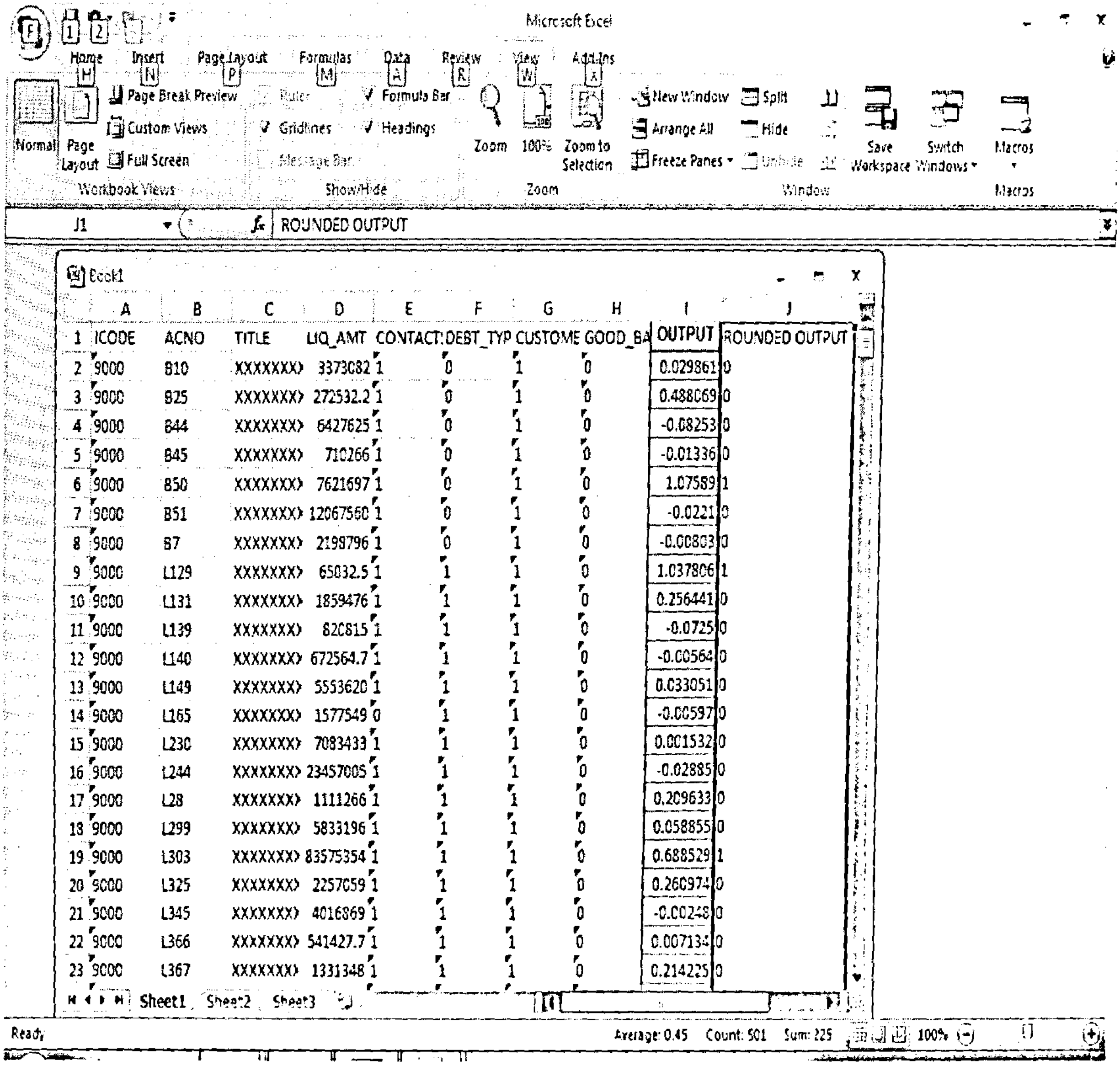
Institution Name: KENYA FINANCE BANK

Institution Code	Account No.	Account Title	Amount At Liquidation	Loan Classification
9000	1054	XXXXXXXXXXXXXXXXXXXXXXXX	387.00	Good Loan
9000	127	XXXXXXXXXXXXXXXXXXXXXXXX	352.00	Good Loan
9000	127	XXXXXXXXXXXXXXXXXXXXXXXX	2,101.00	Good Loan
9000	134	XXXXXXXXXXXXXXXXXXXXXXXX	626.00	Good Loan
9000	135	XXXXXXXXXXXXXXXXXXXXXXXX	280,900.00	Good Loan
9000	1368	XXXXXXXXXXXXXXXXXXXXXXXX	2,194.00	Good Loan
9000	1377	XXXXXXXXXXXXXXXXXXXXXXXX	152,186.00	Bad Loan
9000	141	XXXXXXXXXXXXXXXXXXXXXXXX	192,702.00	Bad Loan
9000	1519	XXXXXXXXXXXXXXXXXXXXXXXX	5,118,386.00	Good Loan
9000	1526	XXXXXXXXXXXXXXXXXXXXXXXX	81,068.00	Bad Loan
9000	1540	XXXXXXXXXXXXXXXXXXXXXXXX	1,985,190.00	Bad Loan
9000	161	XXXXXXXXXXXXXXXXXXXXXXXX	1,008.00	Good Loan
9000	161	XXXXXXXXXXXXXXXXXXXXXXXX	2,099.00	Good Loan
9000	1618	XXXXXXXXXXXXXXXXXXXXXXXX	10,039,035.00	Bad Loan
9000	163	XXXXXXXXXXXXXXXXXXXXXXXX	56,165.00	Bad Loan
9000	163	XXXXXXXXXXXXXXXXXXXXXXXX	330,137.00	Bad Loan
9000	1643	XXXXXXXXXXXXXXXXXXXXXXXX	1,025,841.00	Bad Loan
9000	1646	XXXXXXXXXXXXXXXXXXXXXXXX	5,728,185.00	Bad Loan
9000	167	XXXXXXXXXXXXXXXXXXXXXXXX	2,946.00	Good Loan

Debt Classification Report

APPENDIX II: SAMPLE DATA AND OUTPUT

Training and Validation Data

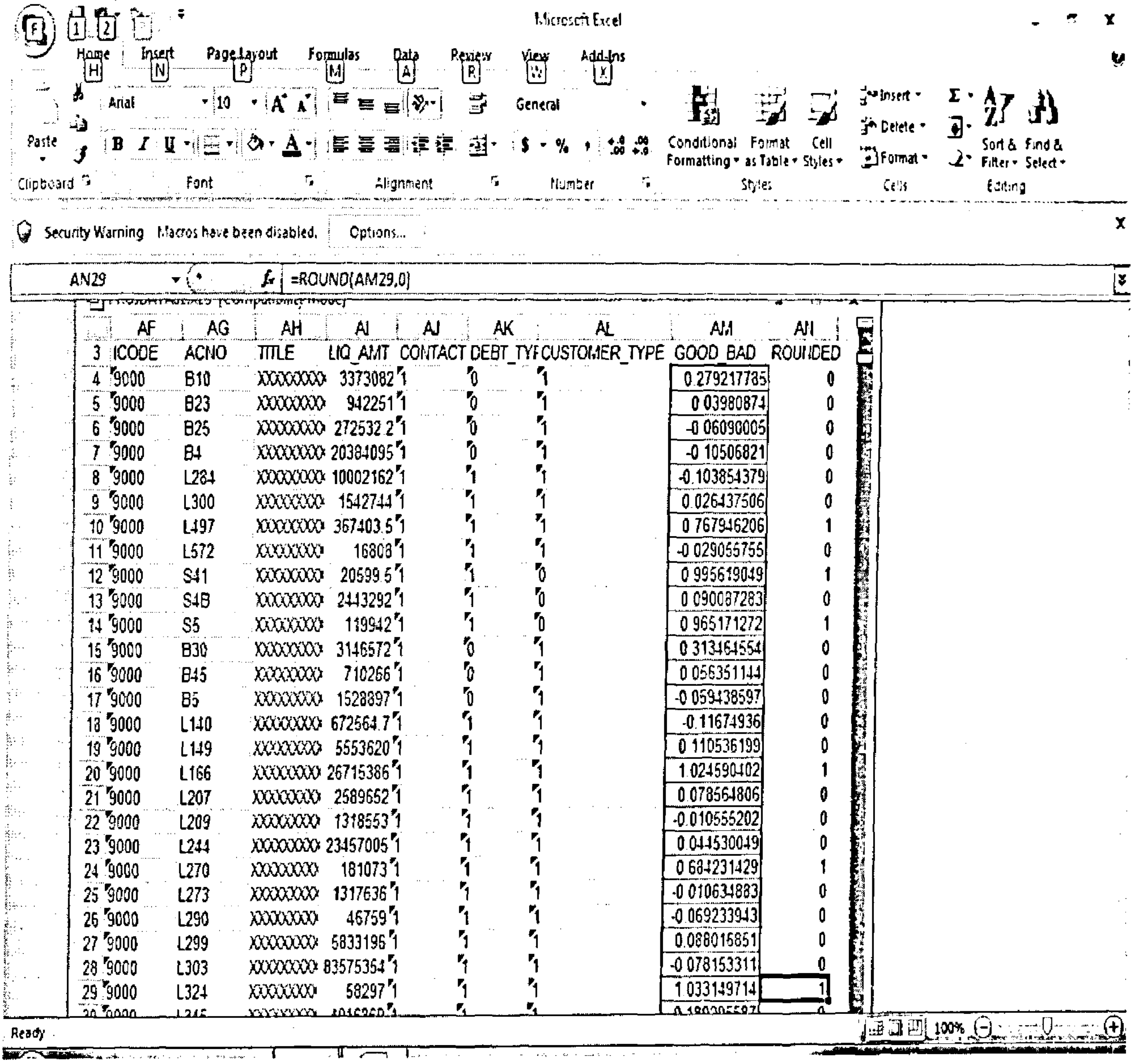


J1 ROUNDED OUTPUT

	A	B	C	D	E	F	G	H	I	J
1	ICODE	ACNO	TITLE	LIQ_AMT	CONTACT	DEBT_TYP	CUSTOME	GOOD_BA	OUTPUT	ROUNDED OUTPUT
2	9000	810	XXXXXXX>	3373082	1	0	1	0	0.029861	0
3	9000	825	XXXXXXX>	272532.2	1	0	1	0	0.488069	0
4	9000	844	XXXXXXX>	6427625	1	0	1	0	-0.08253	0
5	9000	845	XXXXXXX>	710266	1	0	1	0	-0.01336	0
6	9000	850	XXXXXXX>	7621697	1	0	1	0	1.07589	1
7	9000	B51	XXXXXXX>	12067560	1	0	1	0	-0.0221	0
8	9000	B7	XXXXXXX>	2198796	1	0	1	0	-0.00803	0
9	9000	L129	XXXXXXX>	65032.5	1	1	1	0	1.037806	1
10	9000	L131	XXXXXXX>	1859476	1	1	1	0	0.256441	0
11	9000	L139	XXXXXXX>	820815	1	1	1	0	-0.0725	0
12	9000	L140	XXXXXXX>	672564.7	1	1	1	0	-0.00564	0
13	9000	L149	XXXXXXX>	5553620	1	1	1	0	0.033051	0
14	9000	L165	XXXXXXX>	1577549	0	1	1	0	-0.00597	0
15	9000	L230	XXXXXXX>	7083433	1	1	1	0	0.001532	0
16	9000	L244	XXXXXXX>	23457005	1	1	1	0	-0.02885	0
17	9000	L28	XXXXXXX>	1111266	1	1	1	0	0.209633	0
18	9000	L299	XXXXXXX>	5833196	1	1	1	0	0.058855	0
19	9000	L303	XXXXXXX>	83575354	1	1	1	0	0.688529	1
20	9000	L325	XXXXXXX>	2257059	1	1	1	0	0.260974	0
21	9000	L345	XXXXXXX>	4016869	1	1	1	0	-0.00248	0
22	9000	L366	XXXXXXX>	541427.7	1	1	1	0	0.007134	0
23	9000	L367	XXXXXXX>	1331348	1	1	1	0	0.214225	0

Ready Average: 0.45 Count: 501 Sum: 225 100%

Classified Data



AN29 =ROUND(AM29,0)

	AF	AG	AH	AI	AJ	AK	AL	AM	AN
3	ICODE	ACNO	TITLE	LIQ_AMT	CONTACT	DEBT_TYF	CUSTOMER_TYPE	GOOD_BAD	ROUNDED
4	9000	B10	XXXXXXXX	3373082	1	0	1	0.279217785	0
5	9000	B23	XXXXXXXX	942251	1	0	1	0.03980874	0
6	9000	B25	XXXXXXXX	272532.2	1	0	1	-0.06090005	0
7	9000	B4	XXXXXXXX	20384095	1	0	1	-0.10506821	0
8	9000	L284	XXXXXXXX	10002162	1	1	1	-0.103854379	0
9	9000	L300	XXXXXXXX	1542744	1	1	1	0.026437506	0
10	9000	L497	XXXXXXXX	367403.5	1	1	1	0.767946206	1
11	9000	L572	XXXXXXXX	16808	1	1	1	-0.029055755	0
12	9000	S41	XXXXXXXX	20599.5	1	1	0	0.995619049	1
13	9000	S4B	XXXXXXXX	2443292	1	1	0	0.090087283	0
14	9000	S5	XXXXXXXX	119942	1	1	0	0.965171272	1
15	9000	B30	XXXXXXXX	3146572	1	0	1	0.313464554	0
16	9000	B45	XXXXXXXX	710266	1	0	1	0.056351144	0
17	9000	B5	XXXXXXXX	1528897	1	0	1	-0.059438597	0
18	9000	L140	XXXXXXXX	672564.7	1	1	1	-0.11674936	0
19	9000	L149	XXXXXXXX	5553620	1	1	1	0.110536199	0
20	9000	L166	XXXXXXXX	26715386	1	1	1	1.024590402	1
21	9000	L207	XXXXXXXX	2589652	1	1	1	0.078564806	0
22	9000	L209	XXXXXXXX	1318553	1	1	1	-0.010555202	0
23	9000	L244	XXXXXXXX	23457005	1	1	1	0.044530049	0
24	9000	L270	XXXXXXXX	181073	1	1	1	0.684231429	1
25	9000	L273	XXXXXXXX	1317636	1	1	1	-0.010634883	0
26	9000	L290	XXXXXXXX	46759	1	1	1	-0.069233943	0
27	9000	L299	XXXXXXXX	5833196	1	1	1	0.088016851	0
28	9000	L303	XXXXXXXX	83575354	1	1	1	-0.078153311	0
29	9000	L324	XXXXXXXX	58297	1	1	1	1.033149714	1

APPENDIX III: PROTOTYPE SAMPLE CODE SAMPLE

DATA PREPARATION FOR DSS PROCEDURE

```
Private Sub DSSDataPrep_Click()
   Frame1.Visible = True
   Frame1.Refresh
   ProgressBar1.Visible = True
   ProgressBar1.Value = 0
   Close #1
   sf = "C:\PROJECTDATA\PROJDATA.csv"
   Open sf For Output As #1
   OUTSTRING = "ICODE" + "ACNO" + "TITLE" + "LIQ_AMT" + "," + "CONTACTS" + "," +
"DEBT_TYPE" + "," + "CUSTOMER_TYPE"
   Print #1, OUTSTRING
   cndep.Execute "Exec  PRCDELEPROJDATA"
   If rinst.State = 1 Then rinst.Close
   rinst.Open "PROJDATA", cndep, adOpenStatic, adLockOptimistic, adCmdTable
   If rloan.State = 1 Then rloan.Close
   rloan.Open "LOANF", cndep, adOpenStatic, adLockOptimistic, adCmdTable
   rloan.MoveFirst
   Frame1.Caption = "LOADING DATA TO TEXT AND SQL TABLE"
   Do While Not rloan.EOF
      If rloan!ACC_TYP = "201" Then
        ACTYP = "1"
      Else
        ACTYP = "0"
      End If
      If rloan!Category = "N" Then
        custyp = "1"
      Else
        custyp = "0"
      End If
      If IsNull(addr1) Then
        CONTACTS = "0"
      Else
        CONTACTS = "1"
      End If
      liqamt = Abs(rloan!conv_amt)
      cndep.Execute "Exec  PrcaddProjdata '" & ISCOD & "','" & rloan!ACNO & "','" & rloan!Title & "','"
& liqamt & "','" & CONTACTS & "','" & ACTYP & "','" & custyp & "', '" & USERID1 & "', '" & DATE1
& "'"
      OUTSTRING = ISCOD + "," + rloan!ACNO + "," + rloan!Title + "," + Trim(Str(liqamt)) + "," +
CONTACTS + "," + ACTYP + "," + custyp
      Print #1, OUTSTRING
      rloan.MoveNext
      If ProgressBar1.Value < 99 Then
        ProgressBar1.Value = ProgressBar1.Value + 0.5
      Else
        ProgressBar1.Value = 0
      End If
   Loop
   ProgressBar1.Value = 100
   MsgBox "Data Loaded To SQL AND TEXT COMPLETED"
   MSG = "PROCEED TO OPEN EXCEL AND LOAD DATA FROM SQL TABLE ?"
```

```
  If MsgBox(MSG, vbYesNo) = vbYes Then
  Else
      MsgBox "DATA NOT LOADED TO EXCEL"
      Exit Sub
  End If
  Frame1.Caption = "OPENING EXCEL AND LOADING DATA FROM SQL"
  mStaffFile1 = "C:\DSS PROJECT\DATA"
  Set xlAPP = CreateObject("excel.application")
  xlAPP.WindowState = 3
  xlAPP.Visible = True
  xlAPP.Workbooks.Open FileName:=mStaffFile1 & "\PROJDATA01.xls"
  'xlApp.Workbooks.Open FileName:=mStaffFile1 & "\Upload incremental Transactions" + "'& 
Mid$(Textline, 41, 4) &'" + "'& Mid$(Textline, 45, 4) &'" + ".xls"
  xlAPP.Visible = True
'  rtpConnectString = "DSN=BAFILSYSDSN;UID=SA;PWD=;DSQ= " & dbase
  If rtran.State = 1 Then rtran.Close
  rtran.Open "select icode, acno, name, liq_amt, contacts, debt_type, customer_type from PROJDATA ",
cnsdep, adOpenKeyset, adLockOptimistic
  '(icode,acno,title,liq_amt,contacts,actyp,custyp)
  xlAPP.Range("AF4").CopyFromRecordset rtran
  MsgBox "Data Preparation Completed Successfully. TEXT Data Location\File Name are: " & sf
  MsgBox "DATA LOADED TO EXCEL"
End Sub
```

PROCEDURE FOR PREPARARATION

```
Private Sub OutPutMnu_Click()
  On Error GoTo SheetErr
  Dim fso As New FileSystemObject
  DATE1 = Format(Date, "DD/MMM/YYYY")
  Frame1.Visible = True
  Frame1.Refresh
  ProgressBar1.Visible = True
  ProgressBar1.Value = 0
  MSG = "This option Prepare Debts Forecast Report. Proceed ?"
  If MsgBox(MSG, vbYesNo) = vbYes Then
  Else
      MsgBox "Preparation Not done"
      Exit Sub
  End If
  Set m_objFSO = New Scripting.FileSystemObject
  Set objFolder = m_objFSO.GetFolder("C:\DSS PROJECT\DATA\")
  cnt = 0
  cndep.Execute "Exec  PRCDELEDSSRPTABLE"
  For Each objfile In objFolder.Files
    cnt = cnt + 4
    Dim ActiveWorkBook As Excel.Application
    Set ActiveWorkBook = New Excel.Application
    ActiveWorkBook.Workbooks.Open FileName:="C:\DSS PROJECT\DATA\" & objfile.Name
      J = 4
      Do While ActiveWorkBook.Sheets(1).Cells(J, 32) <> ""
        cndep.Execute "Exec  PrcaddDssReport '" & ActiveWorkBook.Sheets(1).Cells(J, 32) & "','" &
ActiveWorkBook.Sheets(1).Cells(J, 33) & "','" & ActiveWorkBook.Sheets(1).Cells(J, 34) & "','" &
ActiveWorkBook.Sheets(1).Cells(J, 35) & "','" & Round(Abs(ActiveWorkBook.Sheets(1).Cells(J, 40)), 0) &
"', '" & USERID1 & "', '" & DATE1 & "'"
```

```
          I = I + 1
          J = J + 1
          If ProgressBar1.Value < 99 Then
            ProgressBar1.Value = ProgressBar1.Value + 0.5
          Else
            ProgressBar1.Value = 0
          End If
        Loop
      Next objfile
        ActiveWorkBook.ActiveWorkBook.Saved = True
        ActiveWorkBook.Workbooks.Close '(False)
        Set ActiveWorkBook = Nothing
        ProgressBar1.Value = 100
        MsgBox "Report Data Preparation Complete"
        Exit Sub
SheetErr:

    MsgBox Err.Description
    Screen.MousePointer = vbDefault
    MsgBox "Please There is a Problem with the worksheet " & objfile.Name
End Sub
```

PROCEDURE FOR VIEWING REPORT

```
Private Sub ViewRepMnu_Click()
    With DSSMainFrm.CrystalReport1
        .Reset
        .WindowState = crptMaximized
        .WindowShowCancelBtn = True
        .WindowShowCloseBtn = True
        .WindowShowExportBtn = True
        .WindowShowPrintSetupBtn = True
        .WindowShowPrintBtn = True
        .WindowShowRefreshBtn = True
        .WindowShowProgressCtls = True
        .WindowShowZoomCtl = True
        .DiscardSavedData = True
        .Connect = rtpConnectString
        .ReportFileName = App.Path & "\Decision Support Report.rpt"
        .Action = 1
    End With
End Sub
```

APPENDIX IV: LETTER OF INTRODUCTION TO THE DIRECTOR – DPFB

Simon Nyahe Waithaka,
University of Nairobi,
School of Computing and Informatics,
P.O Box,
Nairobi,
2010.

Dear sir/Madam,

RE: DECISION SUPPORT SYSTEM ON BAD DEBT RECOVERY IN THE DEPOSIT PROTECTION FUND BOARD - KENYA

I am currently pursuing a Masters of Science degree in Information Systems degree at the University of Nairobi. I am conducting a research on the above topic. It is my humble request that you permit me to conduct this research. The research will observe banker / customer confidentiality. To ensure this, no customer names will feature in the collected data.

I take this opportunity to thank you in advance as I anticipate your kind consideration on this matter.

Yours truly,

Simon N. Waithaka
June 07, 2006

DEPOSIT PROTECTION FUND BOARD

3RD FLOOR, CBK HQ. BUILDING, HAILE SELASSIE AVENUE, NAIROBI, KENYA
P.O. BOX 45983-00100, TEL: 217400/1/2/3/4, FAX: 211122

14th June 2006

Simon N. Waithaka

Thro'

Ag. Director
IMS Department

Forwarded
15/06/2006

Dear Sir,

RE: PROJECT FOR MSC COURSE IN INFORMATION SYSTEMS

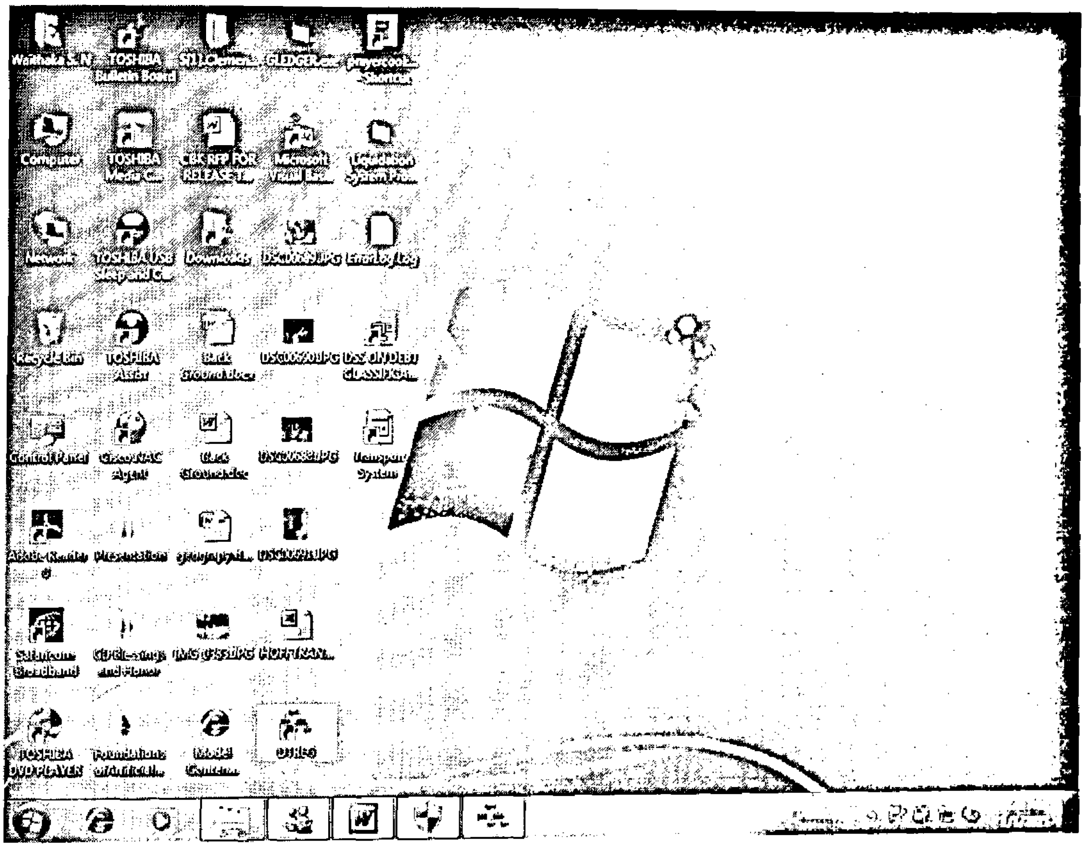
This has reference to your letter on the above subject. We confirm that there is no objection to your request to identify an area of study in DPFB for your project. The research should however observe the banker/customer confidentiality.

Yours faithfully,

K. CHELOTI
DIRECTOR

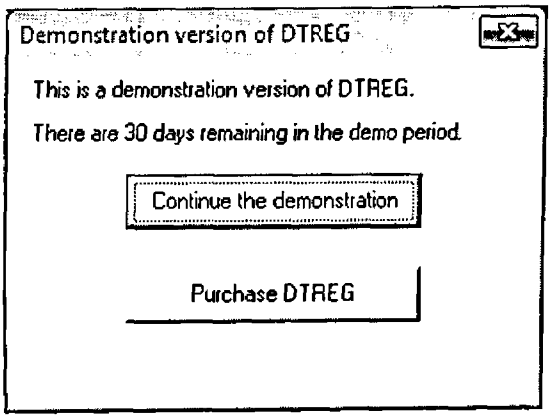
APPENDIX VI: GUIDE TO USING DT-REG DEMONSTRATION SOFTWARE

Click the DTREG icon on the desk top



DTREG Icon

The following dialogue appears indicating that this is a demonstration version of DTREG:



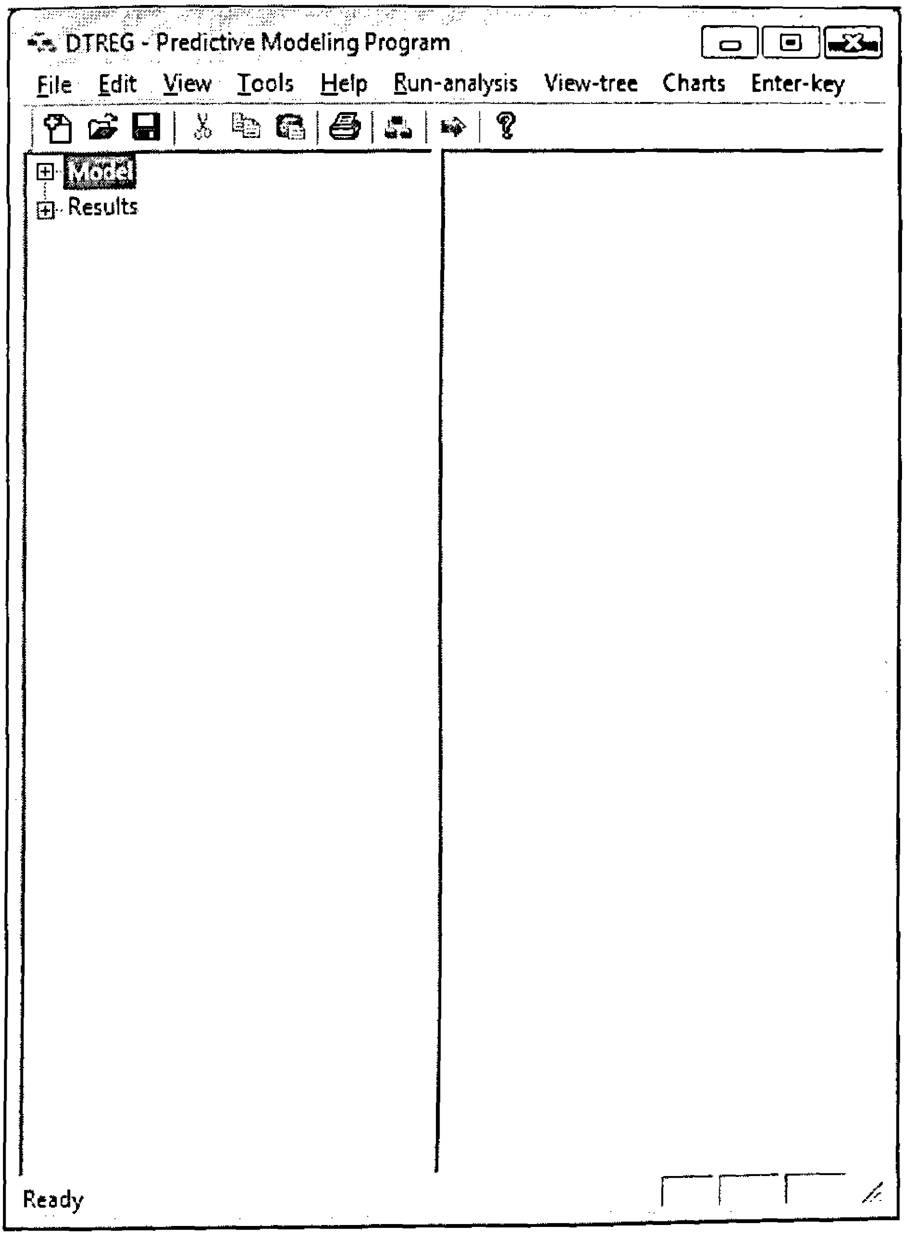
Demonstration version of DTREG

This is a demonstration version of DTREG.

There are 30 days remaining in the demo period.

Proceed Dialogue

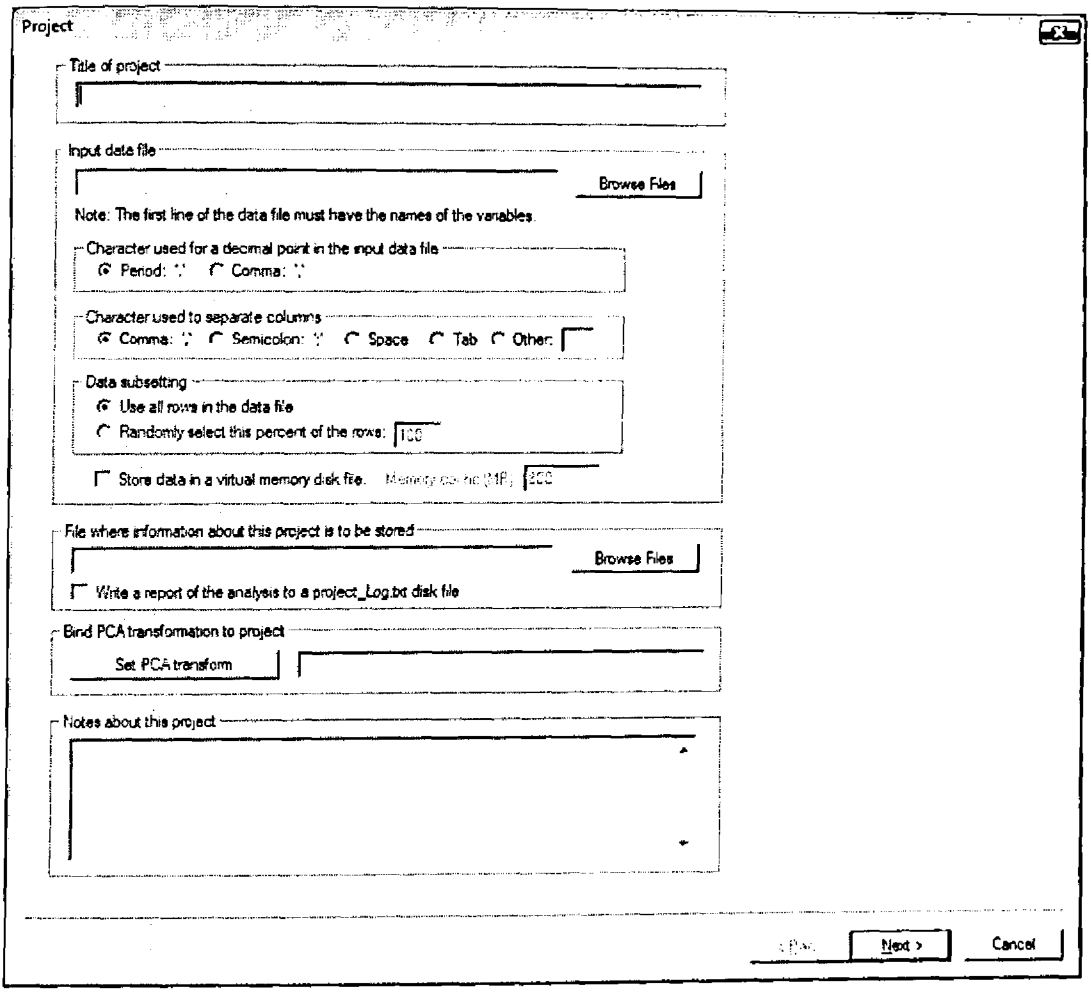
Click Continue the demonstration version to get the following window:


DTREG - Predictive Modeling Program
File Edit View Tools Help Run-analysis View-tree Charts Enter-key
Model
Results
Ready

DTREG Modeling Dialogue

Click create new project icon on upper left of the dialogue.

You are prompted to enter the project name and locate the data file as follows:



Project

Title of project

Input data file

Browse Files

Note: The first line of the data file must have the names of the variables.

Character used for a decimal point in the input data file

● Period: '.' ○ Comma: ','

Character used to separate columns

● Comma: ',' ○ Semicolon: ';' ○ Space ○ Tab ○ Other:

Data subsetting

● Use all rows in the data file

○ Randomly select this percent of the rows: 100

□ Store data in a virtual memory disk file. Memory cache (MB): 200

File where information about this project is to be stored

Browse Files

□ Write a report of the analysis to a project_Log.txt disk file

Bind PCA transformation to project

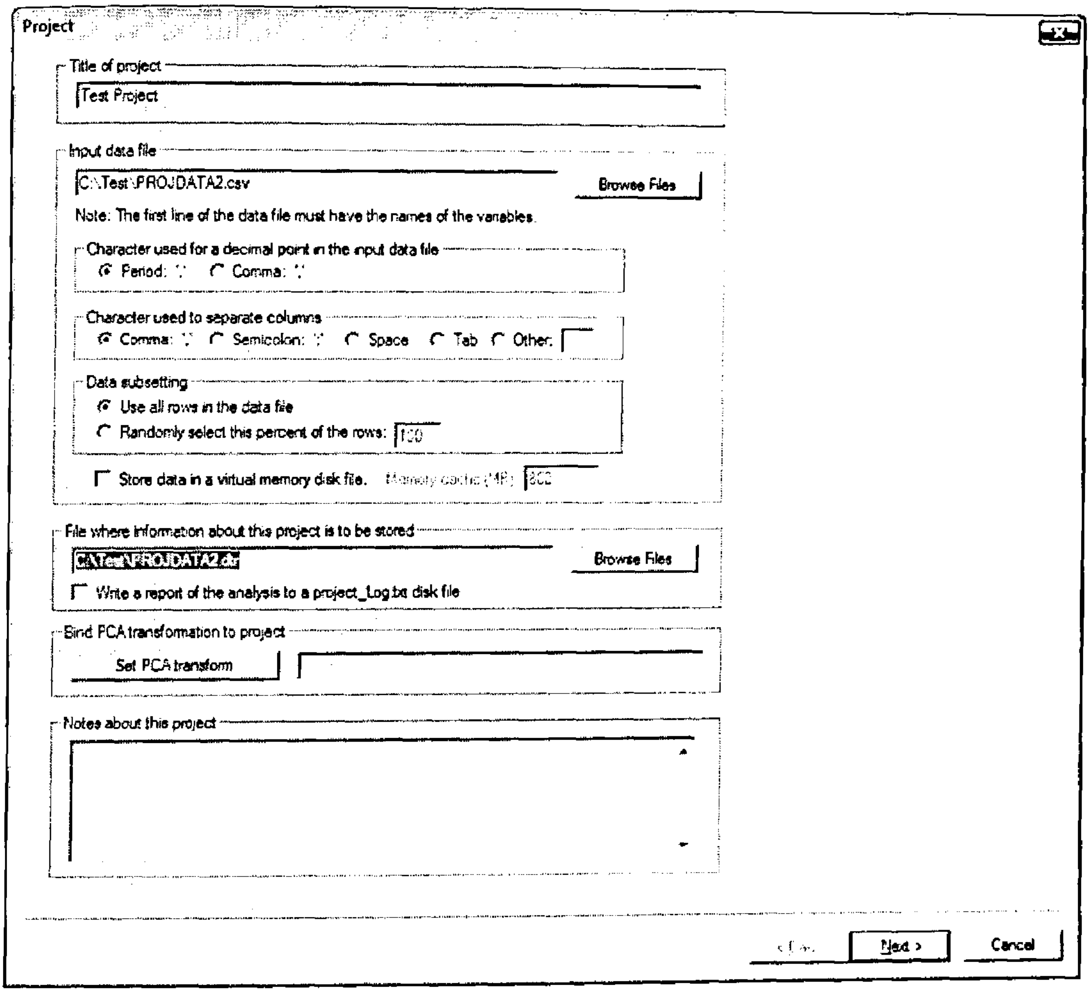
Set PCA transform

Notes about this project

< Prev Next > Cancel

Project Title and Input Data Dialogue

Enter project title and browse for data file. The system automatically populates the file where information about the project is to be stored. The dialogue now looks as follows:



Project

Title of project: Test Project

Input data file: C:\Test\PROJDATA2.csv

Note: The first line of the data file must have the names of the variables.

Character used for a decimal point in the input data file: ● Period: '.' ○ Comma: ','

Character used to separate columns: ● Comma: ',' ○ Semicolon: ';' ○ Space ○ Tab ○ Other:

Data subsetting:
● Use all rows in the data file
○ Randomly select this percent of the rows: 100

□ Store data in a virtual memory disk file. Memory cache (MB): 800

File where information about this project is to be stored: C:\Test\PROJDATA2.dtr

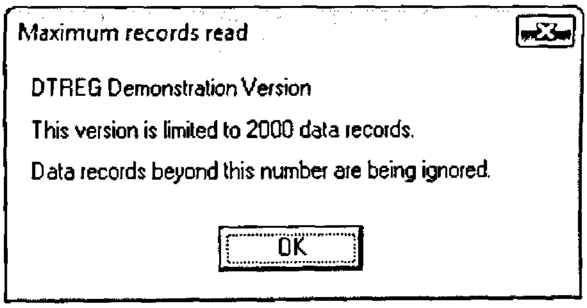
□ Write a report of the analysis to a project_Log.txt disk file

Bind PCA transformation to project

Notes about this project

4 Populated Project Title and Input Data Dialogue

Click next button. You get the following message on the limitation of the demonstration version i.e. only limited to 2000 data records:



Maximum records read

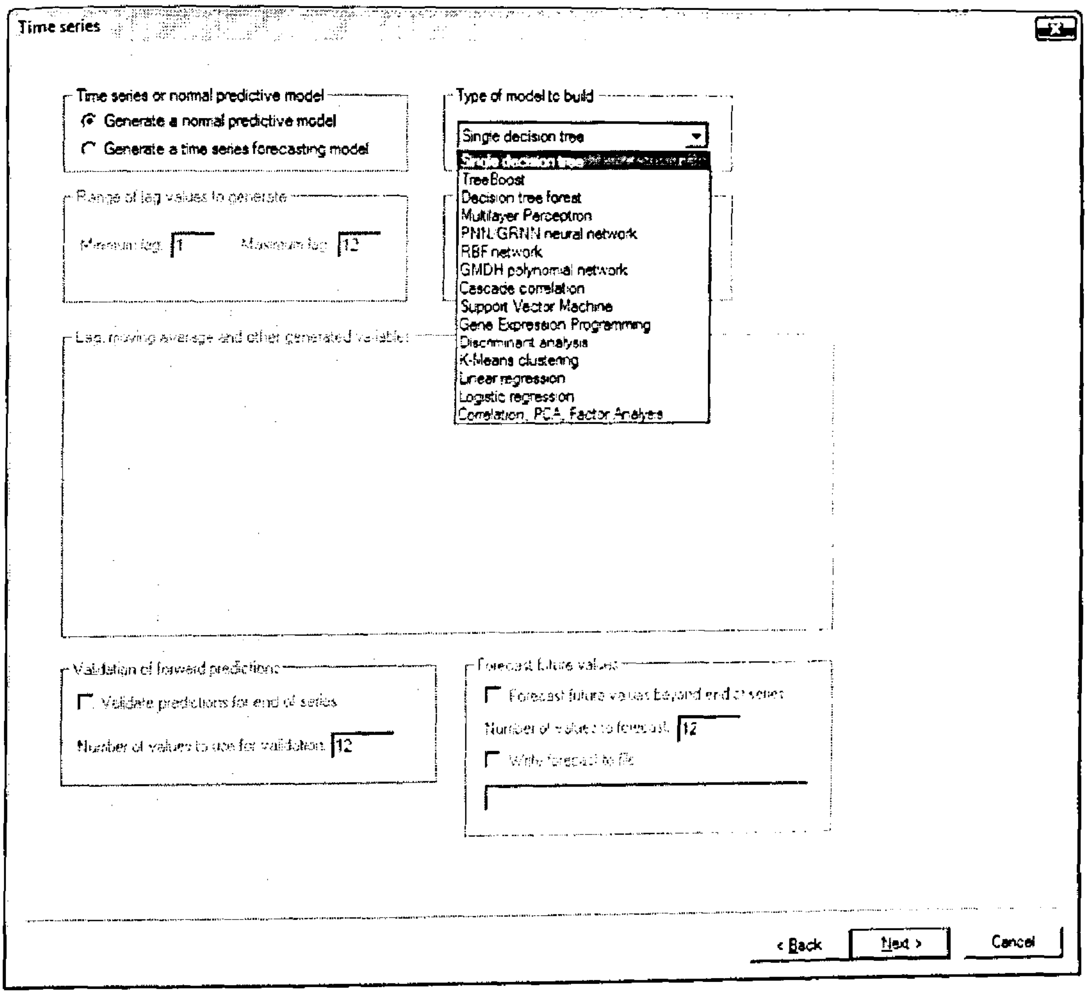
DTREG Demonstration Version

This version is limited to 2000 data records.

Data records beyond this number are being ignored.

Dialogue on DTREG Demo Version Limitations

Click OK to get the following dialogue for you to select the type of model to build:



Time series

Time series or normal predictive model

● Generate a normal predictive model

○ Generate a time series forecasting model

Type of model to build: Single decision tree

Range of lag values to generate

Minimum lag: 1   Maximum lag: 12

Lag, moving average and other generated variables

Validation of forward predictions

□ Validate predictions for end of series

Number of values to use for validation: 12

Forecast future values

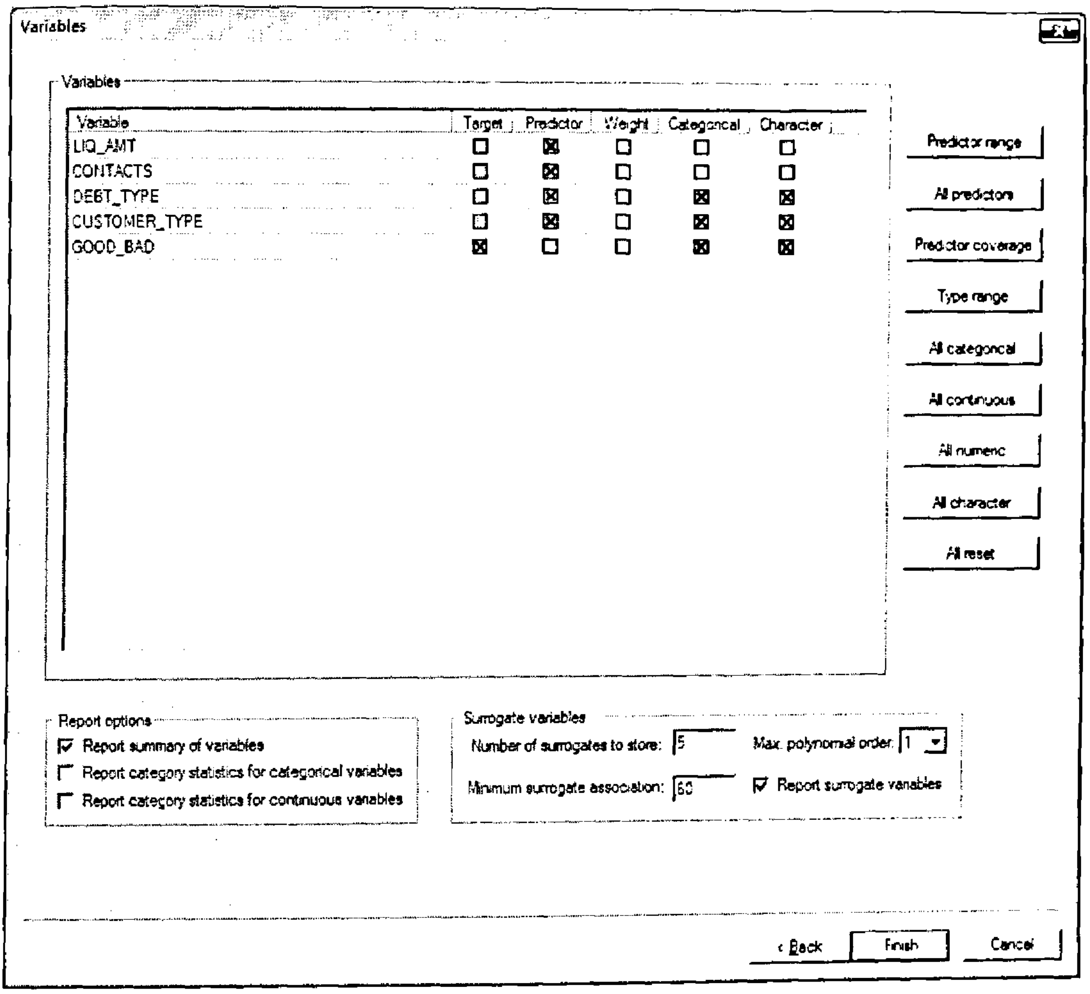
□ Forecast future values beyond end of series

Number of values to forecast: 12

□ Write forecast to file

Selection of Model Type

For this guide we build a model for PNN/GRNN neural networks. Select that and click next to get the following dialogue on the model variables and check the boxes as shown below:



Variables

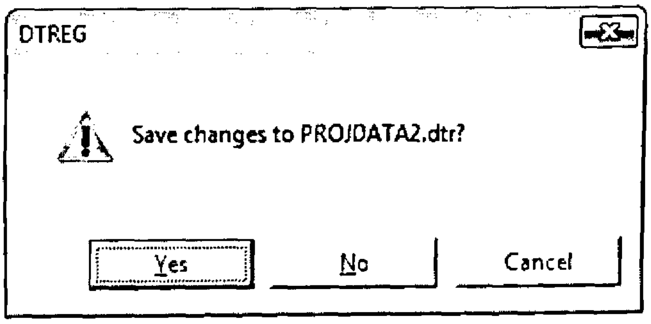
Variable	Target	Predictor	Weight	Categorical	Character
LIQ_AMT	☐	☒	☐	☐	☐
CONTACTS	☐	☒	☐	☐	☐
DEBT_TYPE	☐	☒	☐	☒	☒
CUSTOMER_TYPE	☐	☒	☐	☒	☒
GOOD_BAD	☒	☐	☐	☒	☒

Report options
☑ Report summary of variables
☐ Report category statistics for categorical variables
☐ Report category statistics for continuous variables

Surrogate variables
Number of surrogates to store: 5
Max. polynomial order: 1
Minimum surrogate association: 60
☑ Report surrogate variables

Selection of Model Variables

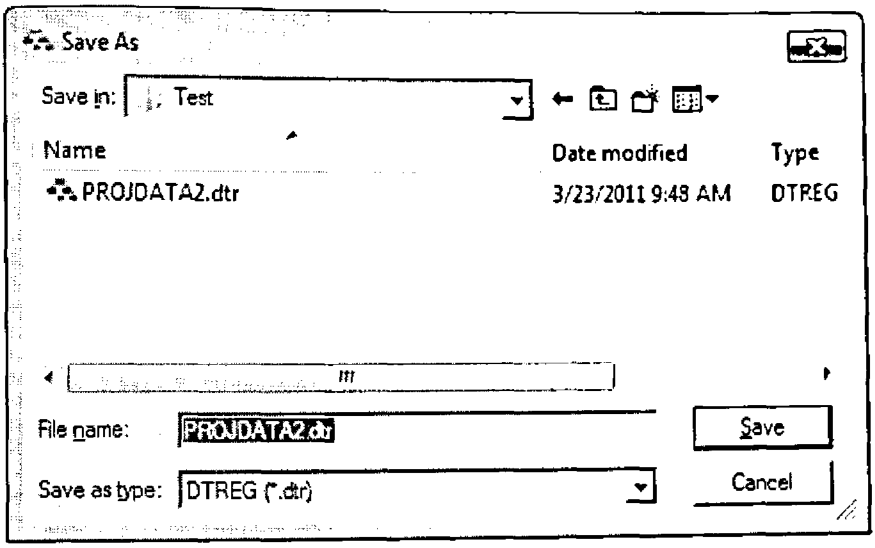
Click finish to get the following dialogue on whether to save the data:



Save changes to PROJDATA2.dtr?

Save Changes Dialogue

Click Yes. The system display the following dialogue for you to click Save button:



Save in: Test

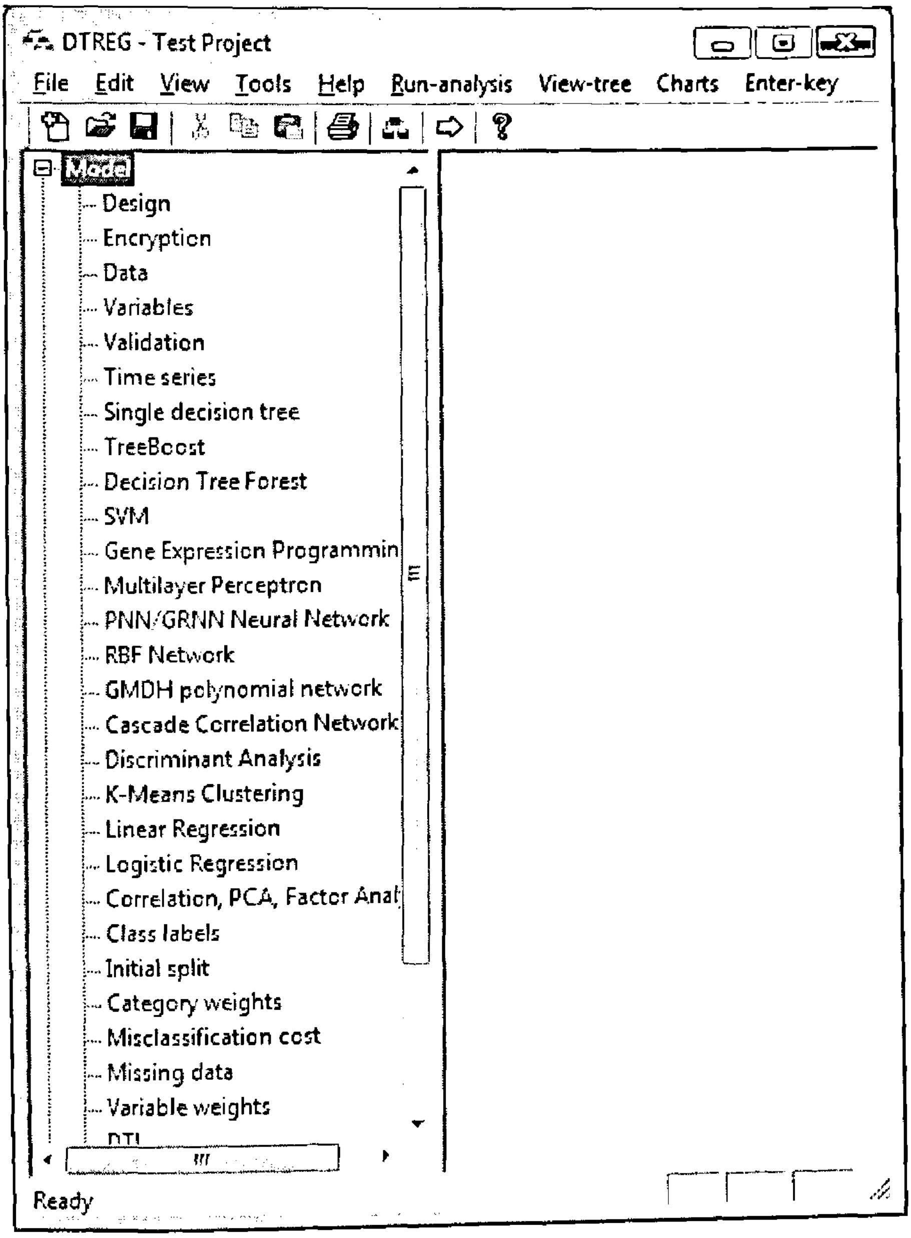
Name	Date modified	Type
PROJDATA2.dtr	3/23/2011 9:48 AM	DTREG

File name: PROJDATA2.dtr

Save as type: DTREG (*.dtr)

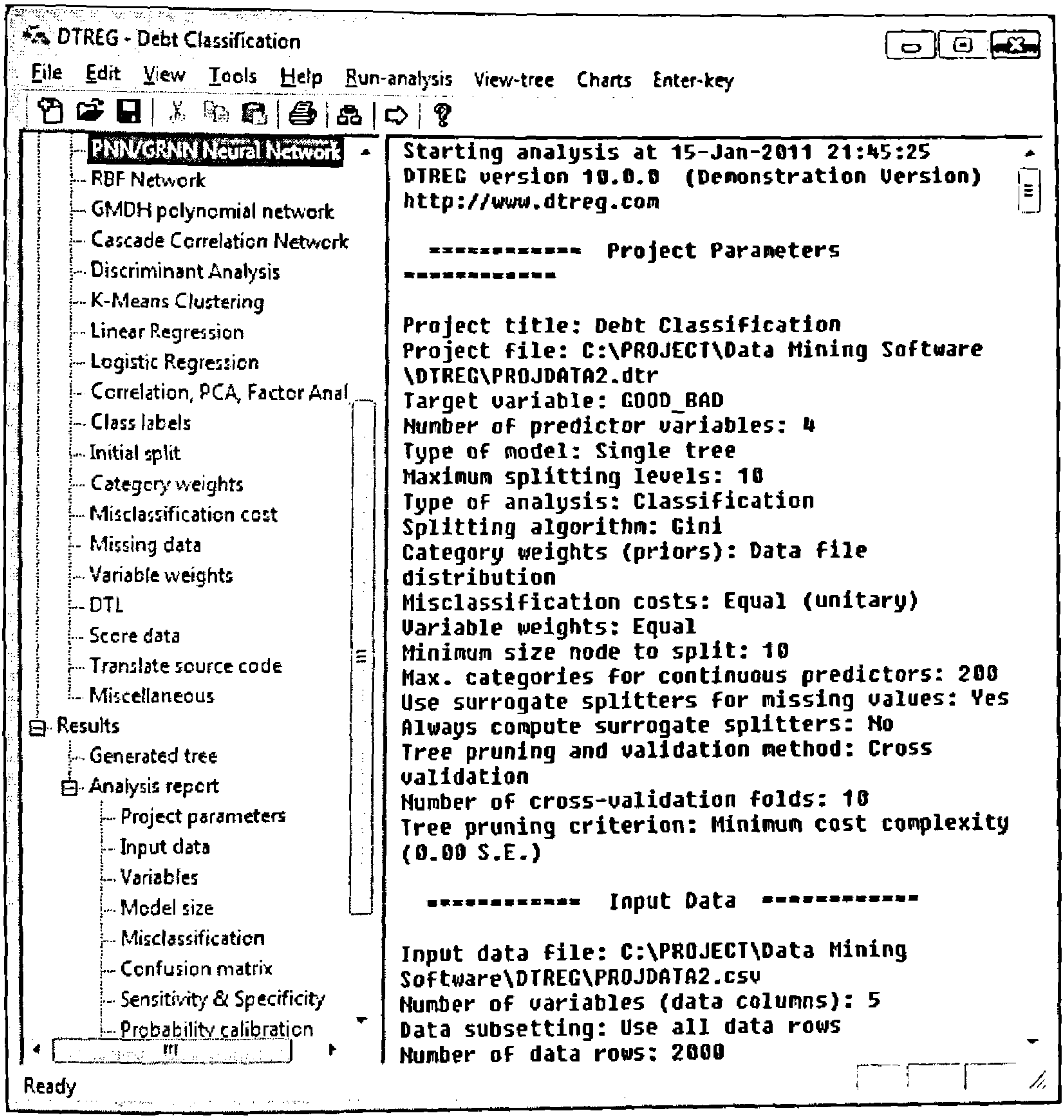
File Name Dialogue

Click the run analysis arrow i.e. the yellow arrow pointing to the right:



Running the Model

The analysis will run and display the results to the right panel as shown below:



```
Starting analysis at 15-Jan-2011 21:45:25
DTREG version 10.0.0  (Demonstration Version)
http://www.dtreg.com

  ============  Project Parameters
============

Project title: Debt Classification
Project file: C:\PROJECT\Data Mining Software
\DTREG\PROJDATA2.dtr
Target variable: GOOD_BAD
Number of predictor variables: 4
Type of model: Single tree
Maximum splitting levels: 10
Type of analysis: Classification
Splitting algorithm: Gini
Category weights (priors): Data file
distribution
Misclassification costs: Equal (unitary)
Variable weights: Equal
Minimum size node to split: 10
Max. categories for continuous predictors: 200
Use surrogate splitters for missing values: Yes
Always compute surrogate splitters: No
Tree pruning and validation method: Cross
validation
Number of cross-validation folds: 10
Tree pruning criterion: Minimum cost complexity
(0.00 S.E.)

  ============  Input Data  ============

Input data file: C:\PROJECT\Data Mining
Software\DTREG\PROJDATA2.csv
Number of variables (data columns): 5
Data subsetting: Use all data rows
Number of data rows: 2000
```

Running the Model Dialogue on the Right

The results are obtained on the right panel.