A project report on

Default Prediction

Submitted by
ASHOK SUTHAR (17MCMT17)
DHILBER M (17MCMI15)

Students of
M.TECH IN COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF COMPUTER & INFORMATION SCIENCE
UNIVERSITY OF HYDERABAD
2017-2019

Submitted to
Dr. V. RAVI
Professor
INSTITUTE FOR DEVELOPMENT AND RESEARCH IN BANKING TECHNOLOGY (IDRBT)

Default Prediction Using Loan Data

Abstract

Default prediction asks you to determine whether a loan will default. Rather than just distinguishing between good or bad counterparties in a binary way, we seek to anticipate and incorporate the prediction capabilities to be able to know which of the existing customers may end up as defaulters and help in avoiding the business losses.
Here, a loan default prediction model is constructed using different training and testing algorithms such as MLP, Decision Trees, SVM, Logistic Regression among others, to train and test the prediction model. This paper has two objectives. First, it illustrates the use of various data mining techniques to construct and test default prediction models. Second, it illustrates the combination of default prediction models to give a superior final model.

Introduction

Any customer or person can come under financial difficulties and it usually does not happen at once. In most cases there are several indicators which can be used to predict the outcome; Such as calls to the customer services, enquiries about the products, a different browsing pattern on the web or mobile app, late or no payments. A historic data set plays a very important role in the prediction of future outcomes or possibilities. By tracing such patterns and behaviours it is possible to prevent, or sometimes at least provide a better service for the customer as well as reduced risks for the bank.
The data set used in this paper is taken from the UCI Machine Learning Repository [Yeh, I. C., & Lien, C. H. (2009)]. This research was aimed at the case of customer's default payments in Taiwan and compares the predictive accuracy of probability of default among six data mining methods. With the real probability of default as the response variable (Y), and the predictive probability of default as the independent variable (X), the simple linear regression result (Y = A + BX) shows that the forecasting model produced by artificial neural network has the highest coefficient of determination; its regression intercept (A) is close to zero, and regression coefficient (B) to one. Therefore, among the six data mining techniques, artificial neural network is the only one that can accurately estimate the real probability of default.

This research employed a binary variable, default payment (Yes = 1, No = 0), as the response variable. This study reviewed the literature and used the following 23 variables as explanatory variables:

X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
X2: Gender
X3: Education
X4: Marital status
X5: Age
X6 - X11: History of past payment.
X12-X17: Amount of bill statement (NT dollar).
X18-X23: Amount of previous payment (NT dollar).

Among the 30000 observations, 23364 (or 77.88%) are good credit risk and 6636 (22.12%) are bad credit risk. The 23 attributes available for constructing credit scoring models including demographic characteristics (e.g., gender and age) and loan details (e.g., loan amount and payment delay etc).
We want to develop prediction model that can be used to determine if a new applicant(instance) is a defaulter or not, based on values for one or more of the predictor variables.

Problem statement

Default prediction can be formally defined as a statistical (or quantitative) method that is used to predict the probability that a loan applicant or existing borrower will default or become fail to pay credit amount. This helps to determine whether loan should be granted to a borrower or not. Default prediction can also be defined as a systematic method for evaluating risk of a customer defaulting, that provides a consistent analysis of the factors that have been determined to cause or affect the level of risk.

The objective of default prediction is to help loan providers. quantify and manage the financial risk involved in providing loan so that they can make better lending decisions quickly and more objectively.

Literature Survey

Angelini et al. [1] used a feed-forward neural network with classical topology and a feed-forward neural network with ad hoc connections, justifying their use of neural network that it is one of the best methods to design a prediction model. In their experiments, data of 76 small businesses from a bank in Italy were used. The conclusions reached that both methods produced efficient models that can correctly predict default with low error.

Tsai et al. [2] produced loan default prediction model using advanced Data Envelopment Analysis Discriminant Analysis (DEA-DA), the statistics-oriented discriminant analysis (DA), logistic regression (LR), and the neural networks (NN). A comparison was done between all these methods, using the accuracy percentage and found that DEA-DA and NN produced the best prediction models.

Akkoc [3], used a three stage hybrid Adaptive Neuro-Fuzzy Inference model, which is combination of statistics and Neuro-Fuzzy. A 10-fold cross was used for validation and a comparison with traditional models show that the produced model is much better.
Credit risk or loan default is considered part of CRM (customer relationship management).

Jafarpour and Garvandani [4] showed the importance of the use of CRM system in banks, taking Iranian banks as an example. The suggested banking CRM model is based on relation between banks and customers dimensions through different relationship channels, which cause improvement in loyalty, life cycle and lifetime value of a customer. A Table and formula is designed that banks can use them to find their customers who can change to higher-level customers and can invest on them to change such customers to more loyal and profiTable customers. However it was not explained well how this Table and formula were designed and which technique was used.

Rani and Loshma [5] presented a framework of an evolving information system based on knowledge from data mining, and has discussed the framework by focusing *on*knowledge of classification. Their main focus was to research customer classification and prediction in Customer Relation Management concerned with data mining based on Back propagation technique. However back propagation can be time demanding but the use of multicore computers can solve the problem.

Ngai et al. [6] identified eighty seven articles related to application of data mining techniques in CRM, and published between 2000 and 2006. The majority of the reviewed articles relate to customer retention. The classification model is the most commonly applied model in CRM for predicting future customer behaviors. They also stated that neural networks were used in a wide range of CRM domains. However this study has some limitations, it surveyed articles published between 2000 and 2006.

Hsu and Hung [7] illustrated that support vector machine (SVM) is suiTable for the bank credit rating classifications. Furthermore, if the data samples increases and applied normal correlation significant test, or adopt other feature selection approach, the SVM predicting accuracy may increase, to make it more effective in rating issues such as the bank credit rating. As for multiple discriminate analysis (MDA), although it has the lowest training errors, it is likely to result in over-fitting, which caused the testing accuracy not accepTable. General if well chosen, feature selection approach improve the model accuracy.

This real world dataset, has been successfully used for prediction modelling and testing systems in many previous works, and is taken from [11] [12] [13] [14] [15] [16] [17] [18] [19] [20] [21].

Advantages of using Default Prediction

- Limit Losses

Default prediction utilizes machine learning technology to determine probability of loan defaulters. Hence allowing for banks to predict and prevent their losses in case of a customer is predicted to be a future defaulter, by guiding him in a better way, or by coming up with strategies to reduce such risks by finding out the reasons for defaulting.

- Better Pricing Rates can be implemented

With more credit available the cost of credit for borrowers has decreased. Default prediction allows for assessing risk factors and allowing for a more suitable pricing rate for the loans given. The customers which has least probability of defaulting can be given slightly lower interest rates to attract them more and increase profits.

- Faster Loan Decisions

Technology that utilizes predictive models for defaulters allows lenders to make instant credit decisions. This is notable in virtually all areas where a consumer seeks credit, from a retail store to an auto dealership to buying a home. In the personal loan and mortgage lending industry, applications can be approved in hours rather than weeks for borrowers who has the least probability of defaulting.

- Opportunities to Improve Predictive model

With data flowing constantly, the predictive models can be updated to show the changes in customer behaviours. Customers who were thought to be defaulters may again come in the loyal customer category and hence better decisions for them can be made.

- Increased Credit Availability

The use of default prediction model gives lenders a much better understanding of risk than previously, giving them the confidence to offer loan to more people, more securely. Lenders who use these models can approve more loans, because model gives them more information upon which they can base their decisions to give loans. In addition, lenders can tailor a range of loans to different risk levels and offer a whole range of credit options.

Tools Used

- Python 3
- Weka

Techniques Used

- Support Vector Machine
- Decision Tree
- Logistic Regression
- Multilayer Perceptron
- K-Nearest Neighbour
- Principal Component Analysis

- Ensembling

Variable Description

Var. #	Variable Name	Variable Type
1.	Amount of the given credit (NT dollar)	Numerical
2.	Gender	Nominal
3.	Education	Nominal
4.	Marital status	Nominal
5.	Age	Numerical
6.	the repayment status in September, 2005	Nominal
7.	the repayment status in August, 2005	Nominal
8.	the repayment status in July, 2005	Nominal
9.	the repayment status in June, 2005	Nominal
10.	the repayment status in May, 2005	Nominal
11.	the repayment status in April, 2005	Nominal
12.	amount of bill statement in September, 2005	Numerical
13.	amount of bill statement in August, 2005	Numerical
14.	amount of bill statement in July, 2005	Numerical
15.	amount of bill statement in June, 2005	Numerical
16.	amount of bill statement in May, 2005	Numerical
17.	amount of bill statement in April, 2005	Numerical
18.	amount paid in September, 2005	Numerical
19.	amount paid in August, 2005	Numerical
20.	amount paid in July, 2005	Numerical
21.	amount paid in June, 2005	Numerical
22.	amount paid in May, 2005	Numerical
23.	amount paid in April, 2005	Numerical
24.	Class	Nominal

X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
X2: Gender (1 = male; 2 = female).
X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
X4: Marital status (1 = married; 2 = single; 3 = others).
X5: Age (year).
X6 - X11: History of past payment.
We tracked the past monthly payment records (from April to September, 2005) as follows:
X6 = the repayment status in September, 2005;
X7 = the repayment status in August, 2005;

. . .;
X11 = the repayment status in April, 2005.
The measurement scale for the repayment status is: -1 = pay duly;
1 = payment delay for one month;
2 = payment delay for two months; . . .;
8 = payment delay for eight months;
9 = payment delay for nine months and above.
X12-X17: Amount of bill statement (NT dollar).
X12 = amount of bill statement in September, 2005;
X13 = amount of bill statement in August, 2005; . . .;
X17 = amount of bill statement in April, 2005.
X18-X23: Amount of previous payment (NT dollar).
X18 = amount paid in September, 2005;
X19 = amount paid in August, 2005;
. . .;
X23 = amount paid in April, 2005.

Model Description

Logistic Regression

Logistic regression is a technique borrowed by machine learning from the field of statistics.It is the go-to method for binary classification problems (problems with two class values). The prediction is based on the use of one or several predictors (numerical and categorical). A linear regression is not appropriate for predicting the value of a binary variable for two reasons:

- A linear regression will predict values outside the acceptable range (e.g. predicting probabilities outside the range 0 to 1).
- Since the dichotomous(with two values) experiments can only have one of two possible values for each experiment, the residuals will not be normally distributed about the predicted line.

On the other hand, a logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression is similar to a linear regression, but the curve is constructed using the natural logarithm of the “odds” of the target variable, rather than the probability. Moreover, the predictors do not have to be normally distributed or have equal variance in each group.

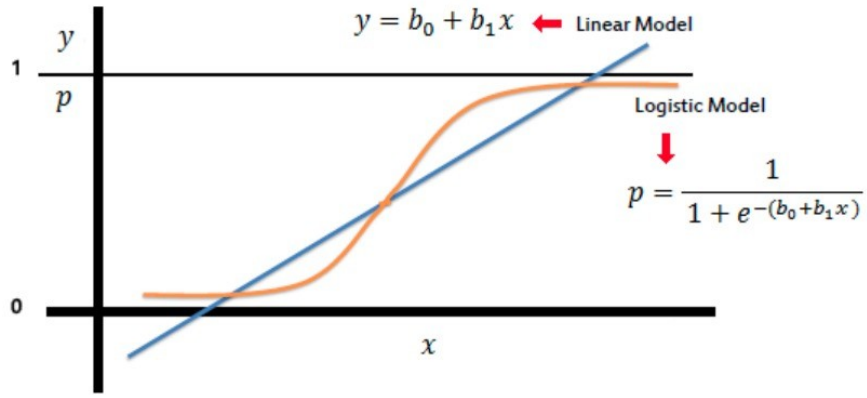

Fig-1 Loogistic Regression

In the logistic regression the constant (b_0) moves the curve left and right and the slope (b_1)defines the steepness of the curve.

Decision Tree

A decision tree is a graph that uses a branching method to illustrate every possible outcome of a decision. Decision trees can be drawn by hand or created with a graphics program or specialized software. Informally, decision trees are useful for focusing discussion when a group must make a decision. Programmatically, they can be used to assign monetary/time or other values to possible outcomes so that decisions can be automated. Decision Tree Software is used in data minig to simplify complex strategic challenges and evaluate the cost-

effectiveness of research and business decisions. Variables in a decision tree are usually represented by circles.

Advantages of Decision Tree

- **Decision trees require relatively little effort from users for data preparation**
- **Nonlinear relationships between parameters do not affect tree performance**
- **The best feature of using trees for analytics - easy to interpret and explain.**
- **Decision trees implicitly perform variable screening or feature selection**

Support Vector Machine

A Support Vector Machine (SVM) performs classification by finding the hyperplane that maximizes the margin between the two classes. The vectors (cases) that define the hyperplane are the support vectors.

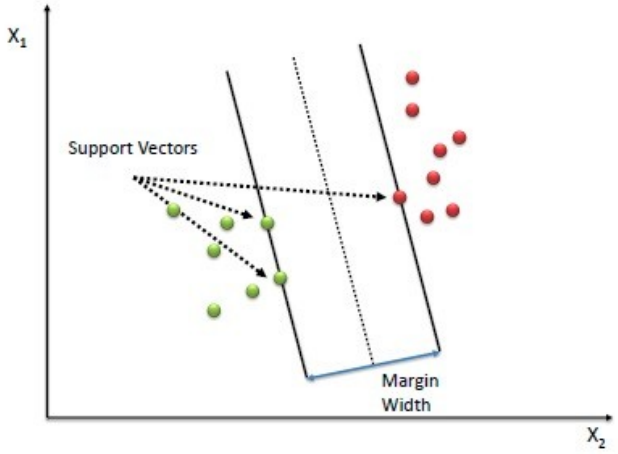

Fig-2 Support Vector Machine

The beauty of SVM is that if the data is linearly separable, there is a unique global minimum value. An ideal SVM analysis should produce a hyperplane that completely separates the vectors (cases) into two non-overlapping classes. However, perfect separation may not be possible, or it may result in a model with so many cases that the model does not classify correctly. In this situation SVM finds the hyperplane that maximizes the margin and minimizes the misclassifications.

K Nearest Neighbors – Classification

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique.

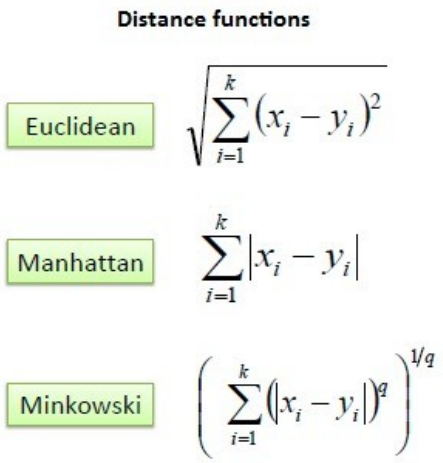

Fig-3 Distance functions

Algorithm - A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor.

It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

Multi Layer Perceptron

A Multi Layer Perceptron(MLP) or Artificial Neural Network(ANN) is a system that is based on the biological neural network, such as the brain. The brain has approximately 100 billion neurons, which communicate through electro-chemical signals. The neurons are connected through junctions called synapses. Each neuron receives thousands of connections with other neurons, constantly receiving incoming signals to reach the cell body. If the resulting sum of the signals surpasses a certain threshold, a response is sent through the axon. The ANN attempts to recreate the computational mirror of the biological neural network, although it is not comparable since the number and complexity of neurons and the used in a biological neural network is many times more than those in an artificial neutral network.

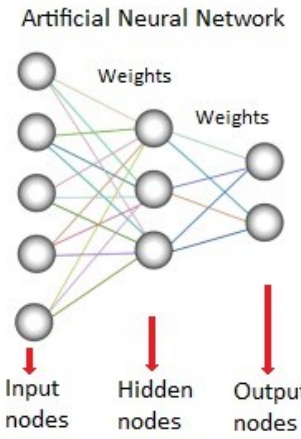

Fig-4 Artificial Neural Network

An ANN is comprised of a network of artificial neurons (also known as "nodes"). These nodes are connected to each other, and the strength of their connections to one another

is assigned a value based on their strength: inhibition (maximum being -1.0) or excitation (maximum being +1.0). If the value of the connection is high, then it indicates that there is a strong connection. Within each node's design, a transfer function is built in. There are three types of neurons in an ANN, **input nodes**, **hidden nodes**, and **output nodes**.

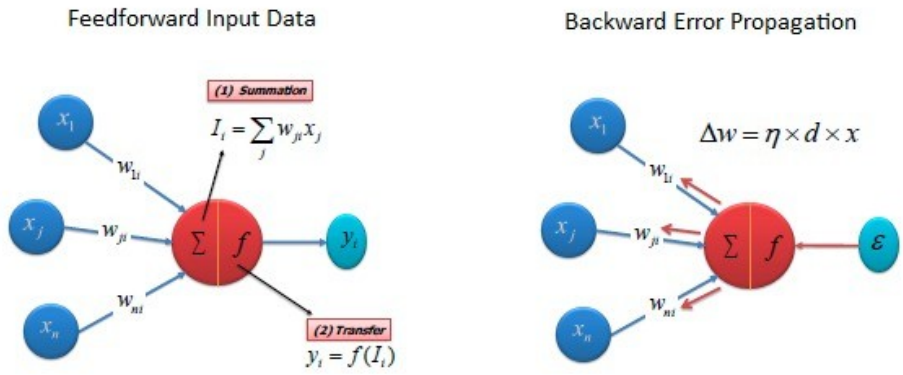

Fig-5 Forward and Backword Propagation

The input nodes take in information, in the form which can be numerically expressed. The information is presented as activation values, where each node is given a number, the higher the number, the greater the activation. This information is then passed throughout the network. Based on the connection strengths (**weights**), inhibition or excitation, and transfer functions, the activation value is passed from node to node. Each of the nodes sums the activation values it receives; it then modifies the value based on its transfer function. The activation flows through the network, through hidden layers, until it reaches the output nodes. The output nodes then reflect the input in a meaningful way to the outside world. The difference between predicted value and actual value (error) will be propagated backward by apportioning them to each node's weights according to the amount of this error the node is responsible for.

Principal Component Analysis

Principal component analysis (PCA) involves a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called *principal components*. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible.

Objectives of principal component analysis

- To discover or to reduce the dimensionality of the data set.
- To identify new meaningful underlying variables.

Ensembling

Ensemble modeling is a powerful way to improve the performance of your model. Ensemble is the art of combining diverse set of learners (individual models) together to improvise on the stability and predictive power of the model. In the above example, the way we combine all the predictions together will be termed as Ensemble Learning.

The three most popular methods for combining the predictions from different models are:

- **Bagging**. Building multiple models (typically of the same type) from different subsamples of the training dataset.
- **Boosting**. Building multiple models (typically of the same type) each of which learns to fix the prediction errors of a prior model in the chain.
- **Stacking**. Building multiple models (typically of differing types) and supervisor model that learns how to best combine the predictions of the primary models.

Data Mining Methodology – CRISP-DM

CRISP-DM stands for cross-industry process for data mining. The CRISP-DM methodology provides a structured approach to planning a data mining project. It is a robust and well-proven methodology. We do not claim any ownership over it. We did not invent it. We are however evangelists of its powerful practicality, its flexibility and its usefulness when using analytics to solve thorny business issues. It is the golden thread than runs through almost every client engagement.

This model is an idealised sequence of events. In practice many of the tasks can be performed in a different order and it will often be necessary to backtrack to previous tasks and repeat certain actions.

- **Business understanding-** This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition, and a preliminary plan designed to achieve the objectives.

- **Data understanding-** The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.

- **Data preparation-** The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include table, record, and attribute selection as well as transformation and cleaning of data for modeling tools.

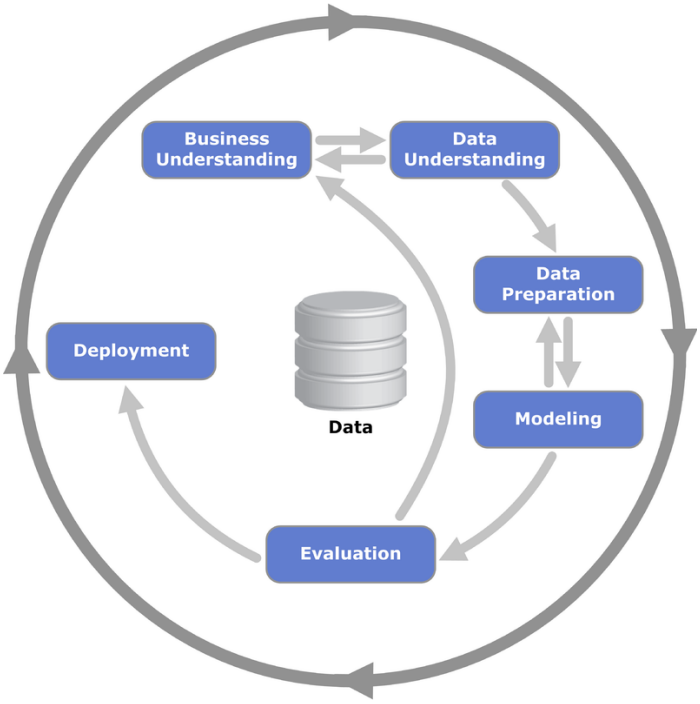

Fig-6 CRISP-DM

- **Modeling-** In this phase, various modeling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements on the form of data. Therefore, stepping back to the data preparation phase is often needed.
- **Evaluation-** At this stage in the project you have built a model (or models) that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to more thoroughly evaluate the model, and review the steps executed to construct the model, to be certain it properly achieves the business objectives. A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.
- **Deployment-** Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that is useful to the customer. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data scoring (e.g. segment allocation) or data mining process. In many cases it will be the customer, not the data analyst, who will carry out the deployment steps. Even if the analyst deploys the model it is important for the customer to understand up front the actions which will need to be carried out in order to actually make use of the created models.

Results

(1)

*With Feature Selection
*Hold-out(70-30)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	78.33	46.5	88.3
Logistic Regression	77.66	45.3	86.9
MLP	76.86	44.1	85.0
K-NN	72.0	59.3	84.5

(2)

After applying Pricipal Component Analysis
*Principal Components=20
*With Feature Selection
*Hold-out(70-30)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	76.3	51.1	87.3
Logistic Regression	75.0	46.5	83.6
K-NN	69.66	55.8	79.9
MLP	76.3	44.18	84.5

(3)

*With Feature Selection
*hold_out(70-30)
*Non-normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	77.66	45.3	86.9
Logistic Regression	77.33	43.02	85.51
MLP	77.00	84.1	40.6

K-NN	74.33	54.6	85.9

(4)

After applying Pricipal Component Analysis
*Principal Components=20
*With Feature Selection
*hold_out(70-30)
*Non-normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	74.00	40.6	79.9
Logistic Regression	77.33	43.02	85.5
K-NN	74.66	55.8	86.9
MLP	77.00	46.5	86.4

(5)

*With Feature Selection
*K-Fold cross validation(K=10)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	76.5	43.3	90.7
Logistic Regression	77.0	46.3	90.1
MLP	76.9	49.0	88.8
K-NN	70.8	38.6	84.5

(6)

After applying Pricipal Component Analysis
*Principal Components=20
*With Feature Selection
*K-Fold cross validation(K=10)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	76.3	41.0	91.4
Logistic Regression	76.5	47.6	88.8
K-NN	70.6	38.3	84.4

MLP	77.5	49.0	89.71

(7)

*With Feature Selection
*K-fold cross validation(k=10)
*Non-normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	75.80	45.6	88.71
Logistic Regression	76.40	46.6	89.1
MLP	76.70	52.3	87.1
K-NN	72.30	41.6	85.4

(8)

After applying Pricipal Component Analysis
*Principal Components=20
*With Feature Selection
*K-fold cross validation(k=10)
*Non-normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	76.40	45.6	89.5
Logistic Regression	76.50	47.6	88.8
K-NN	72.10	41.3	85.2
MLP	75.00	49.0	86.1

(9)

*Without Feature Selection
*Hold-out(70-30)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	77.3	52.3	87.3
Logistic Regression	78.6	56.9	87.3
MLP	76.33	56.9	84.1
K-NN	69.33	39.5	81.3

(10)
After applying Pricipal Component Analysis
*Principal Components=20
*Without Feature Selection
*Hold-out(70-30)
*Normalised data

Model	Accuracy	Specificity	Sensitivity
SVM	77.66	55.8	86.4
Logistic Regression	78.33	58.1	86.4
K-NN	68.33	38.3	80.3
MLP	76.33	58.1	83.6

(11)
*Without Feature Selection
*Hold-out(70-30)
*Non-normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	77.33	53.4	86.9
Logistic Regression	77.33	56.9	85.5
MLP	78.3	56.9	84.1
K-NN	70.0	30.2	85.9

(12)
After applying Pricipal Component Analysis
*Principal Components=20
*Without Feature Selection
*Hold-out(70-30)
*Non-normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	73.0	24.4	92.5
Logistic Regression	77.33	56.9	85.5
K-NN	68.66	30.2	84.1
MLP	76.33	58.1	85.9

(13)
*Without Feature Selection
*K-fold cross validation(K=10)
*Normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	76.5	43.6	90.5
Logistic Regression	76.1	44.6	89.5
MLP	77.3	51.3	88.4
K-NN	70.8	38.3	84.7

(14)
After applying Pricipal Component Analysis
*Principal Components=20
*Without Feature Selection
*K-fold cross validation(K=10)
*Normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	75.5	39.6	90.8
Logistic Regression	75.8	45.3	88.8
K-NN	70.3	37.3	84.4
MLP	75.4	49	56

(15)
*Without Feature Selection
*K-Fold cross validation(K=10)
*Not-normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	76.5	48.3	88.5
Logistic Regression	76.2	45.0	89.5
MLP	76.0	52.3	86.1

K-NN	68.4	30.3	84.7

(16)

After applying Pricipal Component Analysis
*Principal Components=20
*Without Feature Selection
*K-Fold cross validation(K=10)
*Not-normalised Data

Model	Accuracy	Specificity	Sensitivity
SVM	75.5	43.6	89.1
Logistic Regression	76.2	46.3	89.0
K-NN	68.7	30.3	85.14
MLP	77.3	56.3	86.2

(17)

Decision Tree

FSS	Test-Training Method	Accuracy	Specificity	Sensitivity
Yes	Hold-Out	75.6	39.2	88.6
Yes	Cross-Validation	72.4	44.3	84.4
No	Yes	73.6	36.7	88.6
No	Cross-Validation	70.7	39	84.2