ISSN (Print) : 2319-8613
ISSN (Online) : 0975-4024

Credit Risk Analysis and Prediction Modelling of Bank Loans Using R

Sudhamathy G. [#1]

[#1] Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women University, Coimbatore – 641 043, India.
[1] sudhamathy25@gmail.com

Abstract—Nowadays there are many risks related to bank loans, especially for the banks so as to reduce their capital loss. The analysis of risks and assessment of default becomes crucial thereafter. Banks hold huge volumes of customer behaviour related data from which they are unable to arrive at a judgement if an applicant can be defaulter or not. Data Mining is a promising area of data analysis which aims to extract useful knowledge from tremendous amount of complex data sets. In this paper we aim to design a model and prototype the same using a data set available in the UCI repository. The model is a decision tree based classification model that uses the functions available in the R Package. Prior to building the model, the dataset is pre-processed, reduced and made ready to provide efficient predictions. The final model is used for prediction with the test dataset and the experimental results prove the efficiency of the built model.

Keyword-Credit Risk, Data Mining, Decision Tree, Prediction, R

I. INTRODUCTION

Credit Risk assessment is a crucial issue faced by Banks nowadays which helps them to evaluate if a loan applicant can be a defaulter at a later stage so that they can go ahead and grant the loan or not. This helps the banks to minimize the possible losses and can increase the volume of credits. The result of this credit risk assessment will be the prediction of Probability of Default (PD) of an applicant. Hence, it becomes important to build a model that will consider the various aspects of the applicant and produces an assessment of the Probability of Default of the applicant. This parameter PD, help the bank to make decision if they can offer the loan to the applicant or not. In such scenario the data being analysed is huge and complex and using data mining techniques to obtain the result is the most suitable option provided its efficient analytical methodology that finds useful knowledge. There are many such work has been done previously, but they have not explored the use of the features available in R package. R Package is an excellent statistical and data mining tool that can handle any volume of structured as well as unstructured data and provide the results in a fast manner and presents the results in both text and graphical manners. This enables the decision maker to make better predictions and analysis of the findings. The aim of this work is to propose a data mining framework using R for predicting PD for the new loan applicants of a Bank. The data used for analysis contains many inconsistencies like missing values, outliers and inconsistencies and they have to be handled before being used to build the model. Only few of the customer parameters really contribute to the prediction of the defaulter. So, those parameters or features need to be identified before a model is applied. To classify if the applicant is a defaulter or not, the best data mining approach is the classification modelling using Decision Tree. The above said steps are integrated into a single model and prediction is done based on this model. Similar works have been discussed in the "Related Work" Section and the gap in exploring using R has been highlighted. The "Methodology" Section explores the approach that has been followed using text as well as block diagrams. The "Results and Discussions" Section explores the coding and the resultant model applied in this work. It is also important to note that the metrics derived out of this model proves the high accuracy and efficiency of the built model.

II. RELATED WORK

In [1] the author introduces an effective prediction model for predicting the credible customers who have applied for bank loan. Decision Tree is applied to predict the attributes relevant for credibility. This prototype model can be used to sanction the loan request of the customers or not. The model proposed in [2] has been built using data from banking sector to predict the status of loans. This model uses three classification algorithms namely j48, bayesNet and naiveBayes. The model is implemented and verified using Weka. The best algorithm j48 was selected based on accuracy. An improved Risk prediction clustering Algorithm that is Multi-dimensional is implemented in [3] to determine bad loan applicants. In this work, the Primary and Secondary Levels of Risk assessments are used and to avoid redundancy, Association Rule is integrated. The proposed method predicts with better accuracy and consumes less time than previous methods.

In [4] a decision tree model was used as a classifier and for feature selection genetic algorithm is used. The model was tested using Weka. The work in [5] proposes two credit scoring models using data mining techniques to support loan decisions for the Jordanian commercial banks. Considering the rate of accuracy, the results

indicate that the logistic regression model performed better than the radial basis function model. The work in [6] builds several non-parametric credit scoring models. These are based on the multilayer perceptron approach. The work benchmarks their performance against other models which applies the traditional linear discriminant analysis, logistic regression and quadratic discriminant analysis techniques. The results show that the neural network model outperforms the other three techniques.

The work in [7] compares support vector machine based credit-scoring models that were built using Broad and Narrow default definitions. It was shown that models built from Broad definition default can outperform models developed from Narrow default definition. Bank loan default risk analysis, Type of scoring and different data mining techniques like Decision Tree, Random forest, Boosting, Bayes classification, Bagging algorithm and other techniques used in financial data analysis were studied in [8]. The aim of the study in [10] is to introduce a discrete survival model to study the risk of default and to provide the experimental evidence using the Italian banking system. The work in [11] checks the applicability of the integrated model on a sample dataset taken from Banks in India. The model is a combination based on the techniques of Logistic Regression, Radial Basis Neural Network, Multilayer Perceptron Model, Decision tree and Support Vector Machine. It also compares the effectiveness of these techniques for credit Scoring.

The purpose of the work in [12] is to estimate the Label of Credit customers via Fuzzy Expert System. The class of customers has been found by the Fuzzy Expert System and then by the Data Mining Algorithms. This is done using the Clementine software. The work in [14] explores the predicted behaviour of five classifiers in terms of credit risk prediction accuracy, and how such accuracy could be improved. The results of the credit datasets are compared with the performance of each individual classifier based on accuracy. The work in [15] proposed ensemble classifier is constructed by incorporating several data mining techniques, that involves optimal associate binning, discretize continuous values, neural network, support vector machine, and Bayesian network are used. The data driven nature of the proposed system distinguishes it from existing credit scoring systems. A novel credit scoring model is proposed in [16] that gets an aggregation of classifiers. The vertical bagging decision trees model, has been tested using the credit databases in the UCI Machine Learning Repository. The analysis results show the performance is outstanding based on accuracy.

III. METHODOLOGY

Credit risk evaluation has become more important nowadays for Banks to issue loans for their customers based on their credibility. For this the internal rating based approach is the most sought by the banks that need approval by the bank manager. The most accurate and highly used credit scoring measure is the Probability of Default called the PD. Defaulter is the one who is unlikely to repay the loan amount or will have overdue of loan payment by more than 90 days. Hence determining the PD is the crucial step for credit scoring of the customers seeking bank loan.

Hence in this paper we present a data mining framework for PD estimation from a given set of data using the data mining techniques available in R Package. The data used to implement and test this model is taken from the UCI Repository. The German credit scoring dataset with 1000 records and 21 attributes is used for this purpose. The numeric format of the data is loaded into the R Software and a set of data preparation steps are executed before the same is used to build the classification model. The dataset that we have selected does not have any missing data. But, in real time there is possibility that the dataset has many missing or imputed data which needs to be replaced with valid data generated by making use of the available complete data. The k nearest neighbours algorithm is used for this purpose to perform multiple imputation. This is implemented using the knnImputation() function of package **DMwR**. The numeric features are normalized before this step.

The dataset has many attributes that define the credibility of the customers seeking for several types of loan. The values for these attributes can have outliers that do not fit into the regular range of data. Hence, it is required to remove the outliers before the dataset is used for further modelling. The outlier detection for quantitative features is done using the function levels(). For numeric features the boxplot technique is used for outlier detection and this is implemented using the daisy() function of the **cluster** package. But, before this the numeric data has to be normalized into a domain of [0, 1]. The agglomerative hierarchical clustering algorithm is used for outlier ranking. This is done using the outliers.ranking() function of the **DMwR** package. After ranking the outlier data, the ones that is out of range is disregarded and the remaining outliers are filled with null values.

The inconsistencies in the data like unbalanced dataset have to be balanced before building the classification model. Many real time datasets have this problem and hence need to be rectified for better results. But, before this step, it is required to split the sample dataset into training and test datasets which will be in the ratio 4:1 (i.e. Training dataset 80% of data and 20% of data will be test dataset). Now the balancing step will be executed on the training dataset using the SMOTE() function of the **DMwR** package.

Next using the training dataset the correlation between the various attributes need to be checked to see if there are any redundant information represented using two attributes. This is implemented using the plotcorr() function the **ellipse** package. The unique features will then be ranked and based on some threshold limit the

number of highly ranked features will be chosen for model building. For ranking the features the randomForest() function of the **randomForest** package is used. The threshold for selecting the number of important features is chosen by using the rfcv() function of the **randomForest** package.

Now the resultant dataset with the reduced number of features is ready for use by the classification algorithms. Classification is one of the data analysis methods that predict the class labels [19]. Classification can be done in several ways and one of the most appropriate for the chosen problem is using decision trees. Classification is done in two steps – (i) the class labels of the training dataset is used to build the decision tree model and (ii) This model will be applied on the test dataset to predict the class labels of the test dataset. For the first step the function rpart() of the rpart package will be used. The predict() function is used to execute the second step. The resultant prediction is then evaluated against the original class labels of the test dataset to find the accuracy of the model.

The steps involved in this model building methodology are represented as below and the same are presented as block diagrams in Fig. 1, Fig. 2, Fig. 3 and Fig. 4.

Step 1 – Data Selection
Step 2 – Data Pre-Processing
- Step 2.1 – Outlier Detection
- Step 2.2 – Outlier Ranking
- Step 2.3 – Outlier Removal
- Step 2.4 – Imputations Removal
- Step 2.5 – Splitting Training & Test Datasets
- Step 2.6 – Balancing Training Dataset

Step 3 – Features Selection
- Step 3.1 – Correlation Analysis of Features
- Step 3.2 – Ranking Features
- Step 3.3 – Feature Selection

Step 4 – Building Classification Model
Step 5 – Predicting Class Labels of Test Dataset
Step 6 – Evaluating Predictions

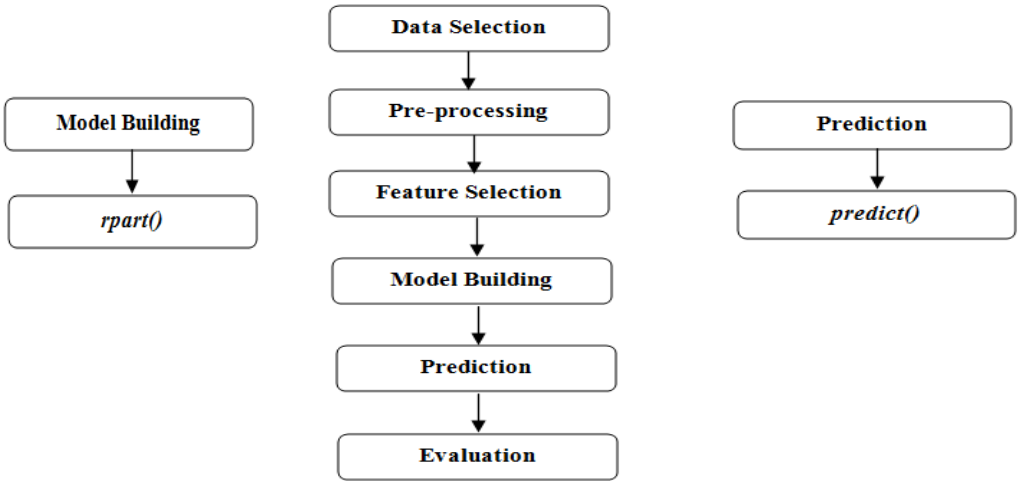

Fig. 1. Major Steps of the Credit Risk Analysis and Prediction Modelling Using R

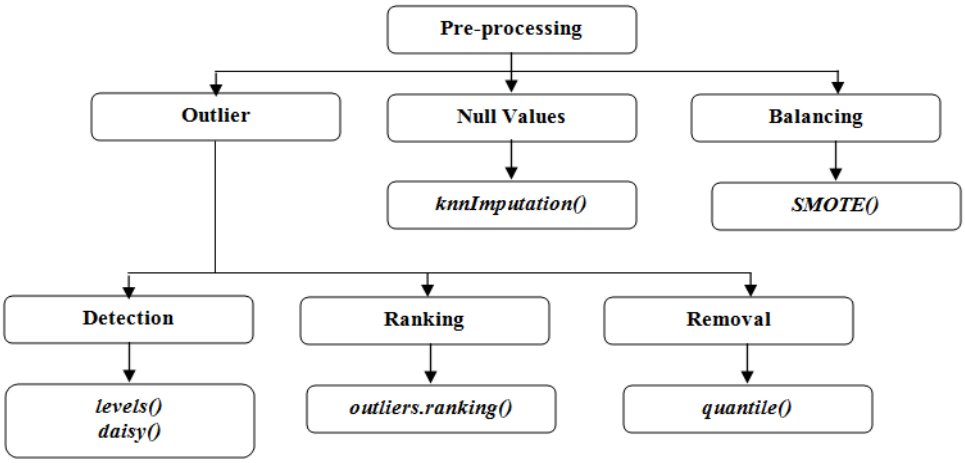

Fig. 2. Sub Steps under the Pre-Processing Step

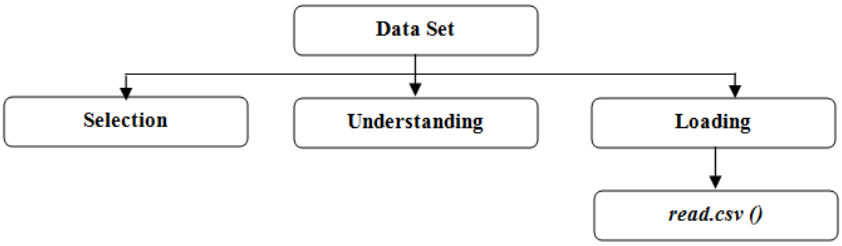

Fig. 3. Sub Steps under the Dataset Selection Process

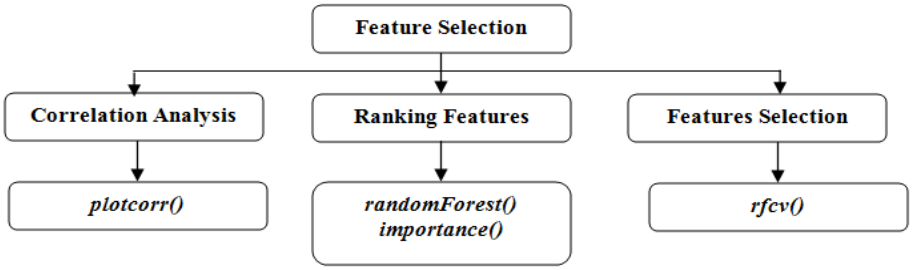

Fig. 4. Sub Steps under the Feature Selection Step

IV. Results and Discussions

A. Dataset Selection

The German Credit Scoring dataset in the numeric format which is used for the implementation of this model has the below attributes and the descriptions of the same are given in the below Table I.

TABLE I Dataset Attribute Types

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	A20	Def
Q	N	Q	Q	N	Q	Q	N	Q	Q	N	Q	N	Q	Q	N	Q	N	B	B	B

***Q**: Quantitative* ***N**: Numeric* ***B**: Binary*

A1: Status of Existing Account
(1: < 0 DM, 2: < 200 DM, 3: >= 200 DM, 4: No existing Account)
A2: Loan Duration in Month
A3: Credit History
(0: No credits taken so far, 1: All credit in this Bank paid back duly,
2: Existing credits paid back dully till now,
3: Delay in paying off in the past,
4: Credits existing in other banks)
A4: Loan Purpose
(0: new car purchase, 1: used car purchase,
2: furniture or equipment purchase, 3: radio or television purchase,
4: domestic appliances purchase,
5: repairs, 6: education, 7: vacation, 8: retraining,
9: Business, 10: others)
A5: Credit Amount (in DM)
A6: Bonds / Savings
(1: < 100 DM, 2: >= 100 and < 500 DM, 3: >= 500 DM and 1000 DM,
4: >= 1000 DM, 5: no savings / bonds)
A7: Present Employment Since
(1: unemployed, 2: < 1 year, 3: >= 1 and < 4 years,
4: >= 4 and < 7 years, 5: >= 7 years)
A8: Instalment rate in percentage of disposable income
A9: Personal Status and Sex
(1: Divorced Male, 2: Divorced/Married Female, 3: Male Single,
4: Married Male, 5: Female Single)
A10: *Other Debtors / Guarantors*
(1: None, 2: Co-applicant, 3; Guarantor)

A11: Present Residence Since (in Years)
A12: Property
(1: Real Estate, 2: Life Insurance, 3: Car or others, 4: No property)
A13: Age in years
A14: Other instalment plans
(1: Bank, 2: Stores, 3: None)
A15: Housing
(1: Rented, 2: Owned, 3: For Free)
A16: Number of existing credits at this bank
A17: Job Status
(1: Unemployed non-resident, 2: Unemployed resident,
3: Skilled Employee, 4: Self-Employed)
A18: Number of People being liable to provide maintenance for
A19: Telephone
(0: Not Available, 1: Available)
A20: Foreign Worker
(0: No, 1: Yes)
Def: Class Label
(0: Non Default, 1: Default)

After selecting and understanding the dataset it is loaded into the R software using the below code. The dataset is loaded into R with the name creditdata.

```
creditdata <- read.csv("UCI German Credit Data Numeric.csv", header = TRUE, sep = ",")
nrow(creditdata)
[1] 1000
```

B. Data Pre-Processing

1) Outlier Detection: To identify the outliers of the numeric attributes, the values of the values of the numeric attributes are normalized into the domain range of [0, 1] and they are plotted as boxplot to view the outlier values. The code and the result for this step are given as below.

```
normalization <- function(data,x)
{for(j in x)
{data[!(is.na(data[,j])),j]=
(data[!(is.na(data[,j])),j]-min(data[!(is.na(data[,j])),j]))/
(max(data[!(is.na(data[,j])),j])-min(data[!(is.na(data[,j])),j]))}
return(data)}
```

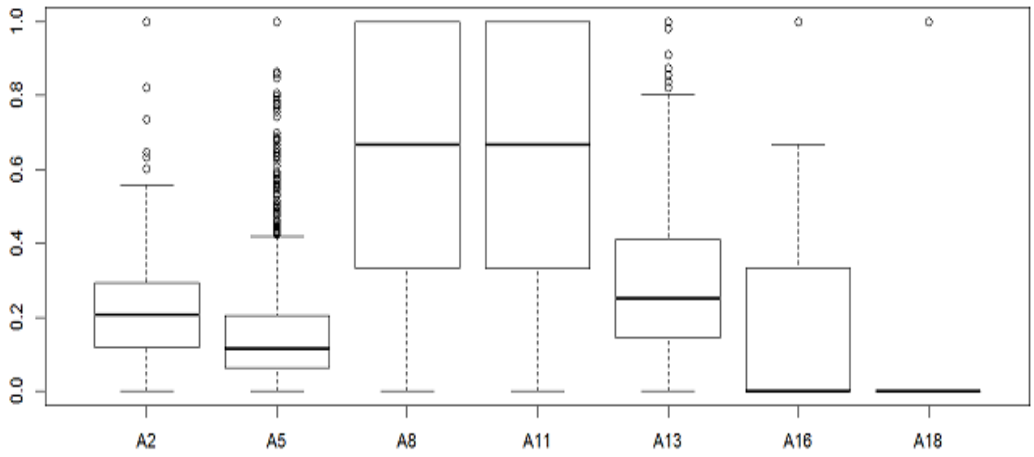

Fig. 5. Box Plot of Outliers in Numeric Attributes

```
c <- c(2,5,8,11,13,16,18)
normdata <- normalization(creditdata,c)
boxplot(normdata[,c])
```

To identify the outliers of the quantitative attributes, the below commands are used. From the results of the same, one can identify the values that do not fall under the allowed values. For example the attribute “A1” can only take the values “1”, “2”, “3”, “4”. If there is any observation that has data other than these allowed values, it is removed. Similarly, the allowed values for each quantitative attribute can be checked and outliers removed.

```
levels(as.factor(creditdata["A1"]))
[1] "1" "2" "3" "4"
```

2) Outliers Ranking: The agglomerative hierarchical clustering algorithm chosen for ranking the outliers is less complex and easy to understand. Each observation is assumed to be a cluster and in each step, the observations are grouped based on the distance between them. Each observation that is observed later has lower rank. It is seen that the observations with lower rank are outliers because there are dissimilarities between them and the other observations [18]. For outlier ranking the following code is used.

```
require(cluster)
distance=daisy(creditdata[,-19],stand=TRUE,metric=c("gower"), type =
list(interval=c(2,5,8,11,13,16,18), nominal=c(1,3,4,6,7,9,10,12,14,15,17),binary=c(19,20)))
require(DMwR)
outlierdata=outliers.ranking(distance,test.data=NULL,method="sizeDiff",clus = list(dist="euclidean",
alg = "hclust", meth="average"), power = 1, verb = F)
```

3) Outliers Removal: The observations which are out of range (based on the rankings) are removed using the below code. After outlier removal the dataset creditdata is renamed as creditdata_noout.

```
boxplot(outlierdata$prob.outliers[outlierdata$rank.outliers])
n=quantile(outlierdata$rank.outliers)
```

```
n1=n[1]
n4=n[4]
filler=(outlierdata$rank.outlier > n4*1.3)
creditdata_noout=creditdata[!filler,]
nrow(creditdata_noout)
[1] 975
```

4) Imputations Removal: The method used for null values removal is multiple imputation method in which the k nearest neighbours' algorithm is used for both numeric and quantitative attributes. The numeric features are normalized before calculating the distance between objects. The following code is used for imputations removal. After imputations removal the dataset creditdata_noout is renamed as creditdata_noout_noimp.

```
require(DMwR)
creditdata_noout_noimp=knnImputation(creditdata_noout, k = 5, scale = T, meth = "weighAvg",
distData = NULL)
nrow(creditdata_noout_noimp)
[1] 975
```

There were no null values for the attributes in the dataset we have chosen and hence the number of records remains unchanged after the above step.

5) Splitting Training and Test Datasets: Before proceeding to the further steps, the dataset has to be split into training and test datasets so that the model can be built using the training dataset. The code for splitting the database is listed below.

```
library(DMwR)
split<-sample(nrow(creditdata_noout_noimp), round(nrow(creditdata_noout_noimp)*0.8))
trainingdata=creditdata_noout_noimp[split,]
testdata=creditdata_noout_noimp[-split,]
```

6) Balancing Training Dataset: The SMOTE function handles unbalanced classification problems and it generates the new smoted dataset that addresses the unbalanced class problem. It artificially generates observations of minority classes using the nearest neighbours of this class of elements to balance the training dataset [18]. The following code is used for balancing the training dataset.

```
creditdata_noout_noimp_train=trainingdata
creditdata_noout_noimp_train$default <- factor(ifelse(creditdata_noout_noimp_train$Def == 1, "def",
"nondef"))
creditdata_noout_noimp_train_smot <- SMOTE(default ~ ., creditdata_noout_noimp_train,
k=5,perc.over = 500)
```

The data distribution before and after balancing the data are shown in the Fig. 6 and Fig. 7 respectively. This method is based on proximities between objects and produces a spatial representation of these objects. Proximities represent the similarity or dissimilarity between data objects. The code used to plot these objects is shown below.

```
library(cluster)
dist1=daisy(creditdata_noout_noimp_train[,-21],stand=TRUE,metric=c("gower"), type =
list(interval=c(2,5,8,11,13,16,18), nominal=c(1,3,4,6,7,9,10,12,14,15,17),binary=c(19,20)))
dist2=daisy(creditdata_noout_noimp_train_smot[,-21],stand=TRUE,metric=c("gower"), type =
list(interval=c(2,5,8,11,13,16,18), nominal=c(1,3,4,6,7,9,10,12,14,15,17),binary=c(19,20)))
loc1=cmdscale(dist1,k=2)
loc2=cmdscale(dist2,k=2)
x1=loc1[,1]
y1=loc1[,2]
x2=loc2[,1]
y2=loc2[,2]
```

```
plot(x1,y1,type="n")
text(x1,y1,labels=creditdata_noout_noimp_train[,22],
col=as.numeric(creditdata_noout_noimp_train[,22])+4)
plot(x2,y2,type="n")
text(x2,y2,labels=creditdata_noout_noimp_train_smot[,22],
col=as.numeric(creditdata_noout_noimp_train_smot[,22])+4)
```

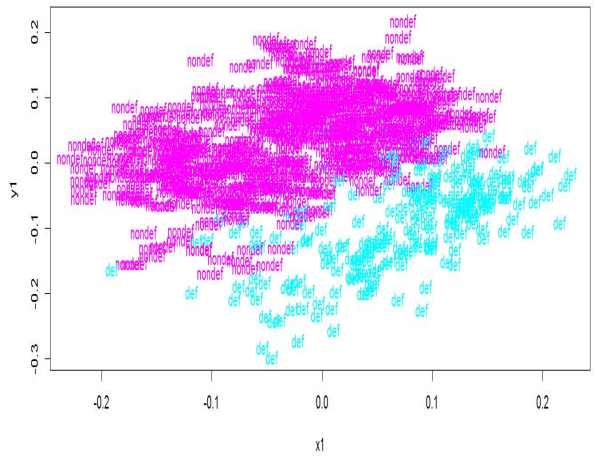

Fig. 6. Data Distribution before Balancing

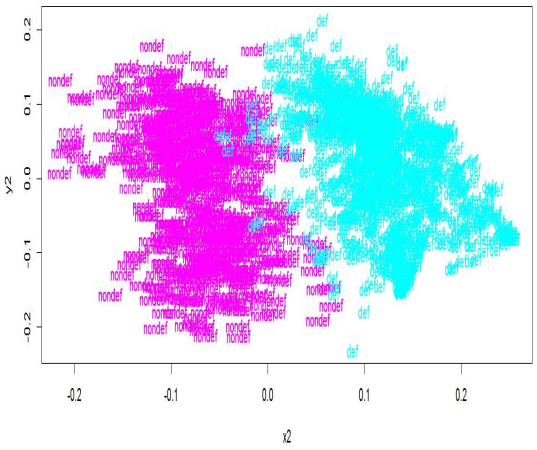

Fig. 7. Data Distribution after Balancing

C. Features Selection

1) Correlation Analysis: Datasets may contain irrelevant or redundant features which might make the model more complicated. Hence removing such redundant features will speed up the model. The function *plotcorr()* plots a correlation matrix using ellipse shaped glyphs for each entry. It shows the correlation between the features in an easy way. The plot is coloured for more clarity. The following code displays the correlation. Correlation is checked independently for each data type: numeric and nominal. From the results in Fig. 8 and Fig. 9, it is observed that there is no positive correlation between any of the features, both numeric and quantitative. Hence, in this step none of the features are removed.

```
library(package="ellipse")
c= c(2,5,8,11,13,16,18)
plotcorr(cor(creditdata_noout_noimp_train[,c]),col=cl<-c(7,6,3))
```

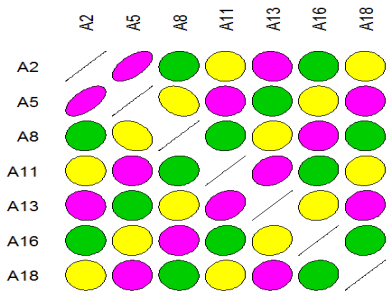

Fig. 8. Correlation between Numeric Features

```
c= c(1,3,4,6,7,9,10,12,14,15,17)
plotcorr(cor(creditdata_noout_noimp_train [,c]),col=cl<-c("green","red","blue"))
```

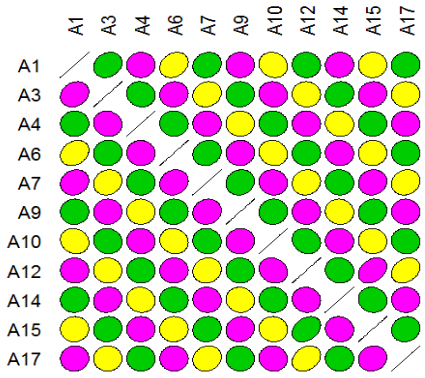

Fig. 9. Correlation between Quantitative Features

2) Ranking Features: The aim of this step is to find the subset of features that will be really relevant for the analysis as irrelevant features causes drawbacks like increased runtime, complex patterns etc. This resultant subset of features should give the same results as that of the original dataset. The proposed method picks a random object from the observations and generates several trees and on the basis of the accuracy of classifier or error ratio, features are weighted. To make the table of important features the following code is used.

library(randomForest)

set.seed(454)

data.frame(creditdata_noout_noimp_train)

randf<-randomForest(Def~ ., data=creditdata_noout_noimp_train, ntree=700, importance=TRUE, proximity=TRUE)

importance(randf, type=1, scale=TRUE)

The above function importance() displays the features importance using the “mean decrease accuracy” measure in Table II. The measures can be plotted using the function varImpPlot()as shown in Fig. 10.

varImpPlot(randf)

TABLE II Importance of Features

Features	Mean Decrease Accuracy
A1	8.085083
A2	7.070556
A3	4.691744
A4	-0.10716
A5	6.238347
A6	4.554283
A7	3.316346
A8	0.59622
A9	1.634721
A10	1.383725
A11	0.541585
A12	2.344433
A13	2.621854
A14	4.629331
A15	0.825801
A16	1.225997
A17	0.635881
A18	0.037408
A19	1.117891
A20	1.388876

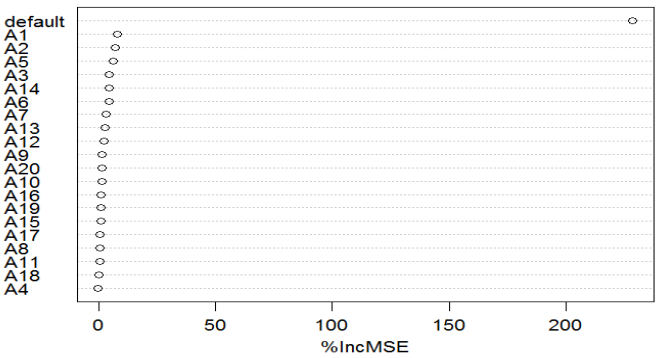

Fig. 10. Ranking Features

3) Features Selection: To fix the number of features to be selected based on the ranking, a threshold is required. This is accomplished using the below code.

```
findopt=rfcv(creditdata_noout_noimp_train[,-21],
creditdata_noout_noimp_train[,21], cv.fold=10, scale="log", step=0.9)
opt <- which.max(findopt$error.cv)
plot( findopt$n.var, findopt$error.cv, type= "h", main = "Importance", xlab="Number of Features",
ylab = "Classifier Error Rate")
axis(1, opt, paste("Threshold", opt, sep="\n"), col = "red", col.axis = "red")
```

The result of this code is shown in the Fig. 11 and it shows the best number of features is 15. Hence we select the features A1, A2, A3, A5, A6, A7, A9, A10, A12, A13, A14, A16, A19, A20, Def to build the model.

D. Building Model

Classification is one of the data analysis forms that predicts categorical labels [19]. We used the decision tree model to predict the probability of default. The following code uses the function rpart() and finds a model from the training dataset.

```
library(rpart)
c = c(4, 8, 11, 15, 17, 18, 22)
trdata=data.frame(creditdata_noout_noimp_train[,-c])
tree=rpart(trdata$Def~.,data=trdata,method="class")
printcp(tree)
```

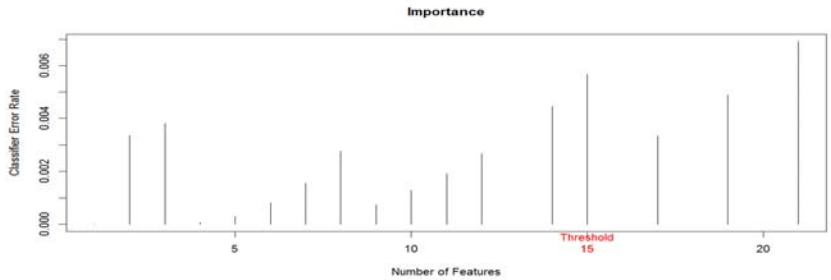

Fig. 11. Threshold for Features Selection

The result of this code is displayed below and in Table III:

Classification tree:

rpart(formula = trdata$Def ~ ., data = trdata, method = "class")

Variables actually used in tree construction:

[1] A1 A12 A13 A2 A3 A5 A6 A9

Root node error: 232/780 = 0.29744

n= 780

TABLE III Results of rpart() Function

CP	nsplit	rel	error	xerror	xstd
1	0.049569	0	1	1	0.05503
2	0.012931	4	0.78448	0.84483	0.052215
3	0.011494	5	0.77155	0.88793	0.053071
4	0.010057	9	0.72414	0.89655	0.053235
5	0.01	18	0.61207	0.89655	0.053235

The command to plot the classification tree is shown below.

plot(tree, uniform=TRUE,main="Classification Tree")

text(tree, use.n=TRUE, all=TRUE, cex=0.7)

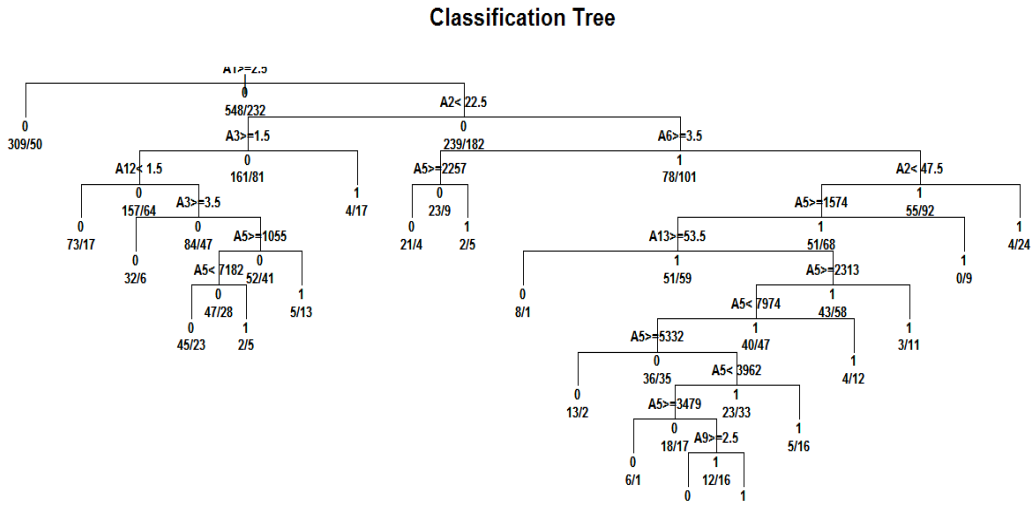

Fig. 12. Classification Tree Model

E. Prediction

The model is tested using the test dataset by using the predict() function. The code for the same and the results of the prediction are displayed below and in Table IV.

predicttest=data.frame(testdata)

pred=predict(tree,predicttest)

c=c(21)

table(predict(tree, testdata, type="class",na.action=na.pass), testdata[, c])

TABLE IV Results of Prediction

	def	nondef
def	30	5
nondef	6	154

F. Evaluation

Common metrics calculated from the confusion matrix are Precision, Accuracy, TP Rate and FP Rate. The calculations for the same are listed below.

$$Precision = \frac{True\ Defaults}{True\ Defaults + False\ Defaults}$$

$$Accuracy = \frac{True\ Defaults + True\ Nondefaults}{Total\ Testset}$$

$$TP\ Rate = \frac{True\ Defaults}{Total\ Defaults}$$

$$FP\ Rate = \frac{False\ Defaults}{Total\ Nondefaults}$$

From our resultant data we get the values of the above metrics by applying the values as derived below.

True Defaults = *30*
False Default = *6*
Total Default = *35*
True Nondefault = *154*
False Nondefault = *5*
Total Nondefault = *160*
Total Testset = *195*
Precision = *30 / (30 + 6)* = *0.833*
Accuracy = *(30 + 154) / 195* = *0.943*
TP Rate = *30 / 35* = *0.857*
FP Rate = *6 / 160* = *0.037*

TABLE V Measures from the Confusion Matrix

Precision	**Accuracy**	**TP Rate**	**FP Rate**
0.833	0.943	0.857	0.037

These results show that the proposed model is performing with high accuracy and precision and hence can be applied for credit scoring.

V. CONCLUSION

In this paper we presented a framework to effectively identify the Probability of Default of a Bank Loan applicant. Probability of Default estimation can help banks to avoid huge losses. This model is built using the data mining functions available in the R package and dataset is taken from the UCI repository. As the pre-processing step is the most important and time consuming one, classification and clustering techniques in R were used to make the data ready for further use. Pre-processed dataset is then used for building the decision tree classifier. The tree model is then used to predict the class labels of the new loan applicants, their Probability of Default. Several R functions and packages were used to prepare the data and to build the classification model. The work proves that the R package is an efficient visualizing tool that applies data mining techniques. The metrics derived from the predictions reveal the high accuracy and precision of the built model.

REFERENCES

[1] M. Sudhakar, and C.V.K. Reddy, "Two Step Credit Risk Assessment Model For Retail Bank Loan Applications Using Decision Tree Data Mining Technique", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 5(3), pp. 705-718, 2016.
[2] J. H. Aboobyda, and M.A. Tarig, "Developing Prediction Model Of Loan Risk In Banks Using Data Mining", *Machine Learning and Applications: An International Journal (MLAIJ)*, vol. 3(1), pp. 1–9, 2016.
[3] K. Kavitha, "Clustering Loan Applicants based on Risk Percentage using K-Means Clustering Techniques", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 6(2), pp. 162–166, 2016.
[4] Z. Somayyeh, and M. Abdolkarim, "Natural Customer Ranking of Banks in Terms of Credit Risk by Using Data Mining A Case Study: Branches of Mellat Bank of Iran", *Jurnal UMP Social Sciences and Technology Management*, vol. 3(2), pp. 307–316, 2015.
[5] A.B. Hussain, and F.K.E. Shorouq, "Credit risk assessment model for Jordanian commercial banks: Neuralscoring approach", *Review of Development Finance, Elsevier*, vol. 4, pp. 20–28, 2014.
[6] A. Blanco, R. Mejias, J. Lara, and S. Rayo, "Credit scoring models for the microfinance industry using neural networks: evidence from Peru", *Expert Systems with Applications*, vol. 40, pp. 356–364, 2013.
[7] T. Harris, "Quantitative credit risk assessment using support vector machines: Broad versus Narrow default definitions", *Expert Systems with Applications*, vol. 40, pp. 4404–4413, 2013.
[8] A. Abhijit, and P.M. Chawan, "Study of Data Mining Techniques used for Financial Data Analysis", *International Journal of Engineering Science and Innovative Technology*, vol. 2(3), pp. 503-509, 2013.
[9] D. Adnan, and D. Dzenana, "Data Mining Techniques for Credit Risk Assessment Task", in *Proceedings of the 4th International Conference on Applied Informatics and Computing Theory (AICT '13)*, 2013, p. 105-110.
[10] G. Francesca, "A Discrete-Time Hazard Model for Loans: Some Evidence from Italian Banking System", *American Journal of Applied Sciences*, 9(9), pp. 1337–1346, 2012.
[11] P. Seema, and K. Anjali, "Credit Evaluation Model of Loan Proposals for Indian Banks", *World Congress on Information and Communication Technologies, IEEE*, pp. 868–873, 2011.
[12] E.N. Hamid, and N. Ahmad, "A New Approach for Labeling the Class of Bank Credit Customers via Classification Method in Data Mining", *International Journal of Information and Education Technology*, vol. 1(2), pp. 150-155, 2011.
[13] K. Abbas, and Y. Niloofar, "A Proposed Classification of Data Mining Techniques in Credit Scoring", in *Proceedings of the 2011 International Conference on Industrial Engineering and Operations Management, Kuala Lumpur, Malaysia*, 2011, p. 416-424.
[14] B. Twala, "Multiple classifier application to credit risk assessment", *Expert Systems with Applications*, vol. 37(4), pp. 3326–3336, 2010.
[15] N.C. Hsieh, and L.P. Hung, "A data driven ensemble classifier for credit scoring analysis", *Expert Systems with Applications*, vol. 37, pp. 534–545, 2010.
[16] Z. Defu, Z. Xiyue, C.H.L. Stephen, and Z. Jiemin, "Vertical bagging decision trees model for credit scoring", *Expert Systems with Applications*, vol. 37, pp. 7838-7843, 2010.
[17] L. Torgo, *Functions and data for "data mining with r" R package version 0.2.3*, 2012.
[18] L. Torgo, *Data Mining with R: Learning with Case Studies*, Chapman Hall/CRC, Boca Raton, 2011.
[19] J. Han, and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, San Francisco, 2006.

AUTHOR PROFILE

Dr. G. Sudhamathy has obtained an undergraduate degree B.Sc. (Spl) Mathematics from Lady Doak College, Madurai, India in 1995. She holds a post graduate degree, Master of Computer Applications (MCA) at Thiagarajar School of Management, Madurai, India in 1998. She has acquired her doctorate degree in Computer Science from Bharathiar University, Coimbatore, India in 2013. She is currently working as Assistant professor in the Department of Computer Science in Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamilnadu, India. She has a rich industrial experience of around 11 years working in various multinational Information Technology companies like Cognizant Technologies Solutions, L&T Infotech, etc. She has worked with international clients and have worked in London for a year. She also has around 7 years of academic and research experience. Her research interests are in Web Usage Mining and Applications of Data Mining using R.