**Bank Loan Default Case - R**

**By Avinash Sajeevan**

**Introduction**

Defaulting on a loan means that you have failed to make sufficient payments for an extended period. Bank will deem a loan in default when you haven't paid the minimum required payment for a certain number of months in a row, as detailed in your loan contract. Loan defaults can happen with any type of loan, whether a mortgage, credit card, or a corporate loan. Defaulting on a loan obligation is serious and can affect the creditworthiness of the individual or company in default. The risk is mainly for the bank and it can include complete or partial loss of principal, loss of interest, and disruption of cash flow.

Gone are the days where the borrowers went scot free after defaulting on bank loan. In the digital age, it is very easy for banks to locate bank defaulter. With the use of machine learning models it’s easy for the bank to predict the risk of default even at the early stage of loan application.

**Problem Statement**

The loan default dataset has 8 variables and 850 records, each record being loan default status for each customer. Each Applicant was rated as “Defaulted” or “Not-Defaulted”. New applicants for loan application can also be evaluated on these 8 predictor variables and classified as a default or non-default based on predictor variables.

**Data Summary**

The data is provided in the csv format, need to be imported using pandas library. Dataset has 8 variables and 850 records.

The data contains the details about customers:

|  |  |
| --- | --- |
| age | Age of each customer |
| ed | Education categories |
| employ | Employment status corresponds to job  status |
| address | Geographic area |
| income | Gross Income of each customer |
| debtinc | Individual’s debt payment to his or her  gross income |
| creddebt | Debt-to-credit ratio |
| othdebt | Any other debts |
| default | Customer defaulted in the past (1= defaulted, 0=Never defaulted) |

**Technological Requirements**

The following list summarizes the tools and packages used in this project

* R – 4.0.3
* Rstudio - 1.3.1093
* Shiny – 1.5.0
* Ggplot2 - 4.0.2
* Corrplot - 4.0.3
* Smotefamily - 1.3.1
* Caret - 4.0.3
* randomForest - 4.6-14
* ROCR - 1.0-11

**Methods Summary**

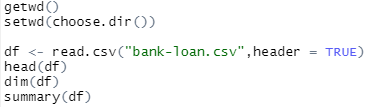
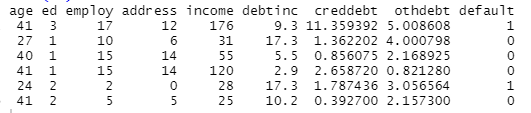
Table shows the list of data pre-processing, analysis, visualization and model building techniques applied to complete the project.

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Task Details | Analytical Techniques | Visualization Techniques |
| Data Manipulation &  Preparation | 1. Perform required data manipulation and cleaning.  2. Perform Univariate  and Bivariate analysis | 1. Descriptive statistics and outlier analysis.  2. T-test and VIF check to get important features. | 1. Histogram, Heat map, boxplots.  2. Boxplot segmentation |
| Model Building &  Performance Check | Create Model and assess the performance  of the models. | 1. Build Logistic Regression model and used cross validation to find best model.  2. Use hyper parameter to fine tune the model. | Plot ROC-AUC Curve and Precision-Recall curve to show the model performance. |

**Model to Predict Default Customers**

**- Data Manipulation & Preparation**

* Started by setting the working directory and load the dataset.

* There are 8 variables and 850 records in the data. Checked for missing values.



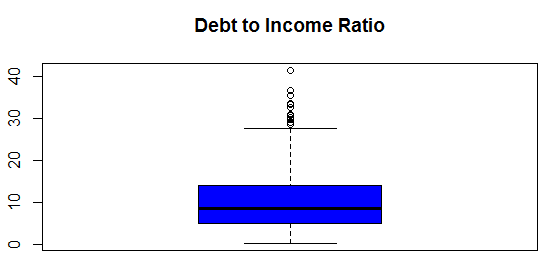
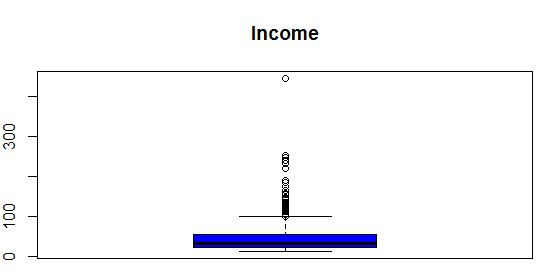


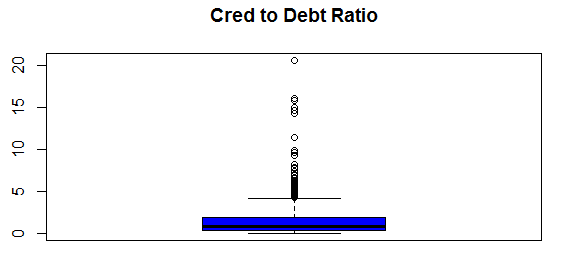
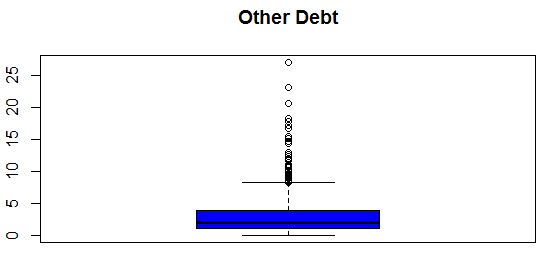
* There are 150 missing values in the dataset and it is found that missing values are in the default column. So dropped those missing values.



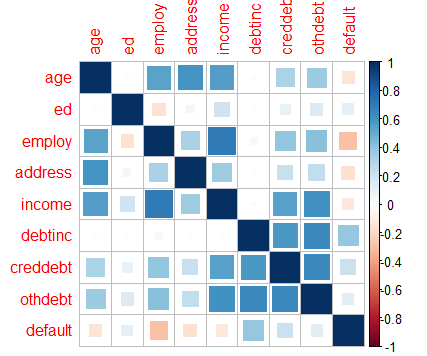


* Got 700 rows of data after dropping the missing values. That will be good for building the model.
* Checked for outliers by using box plots for each variables.

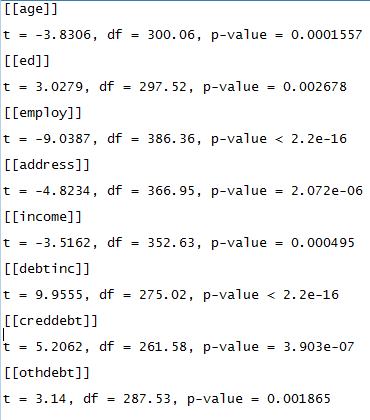


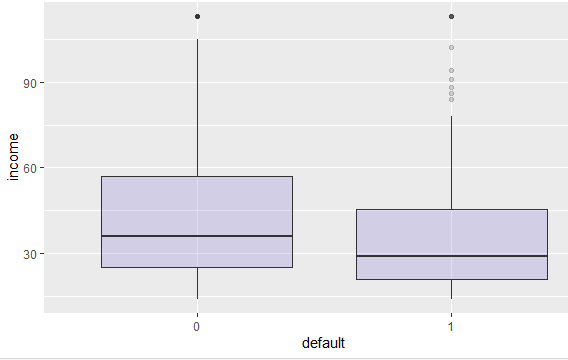
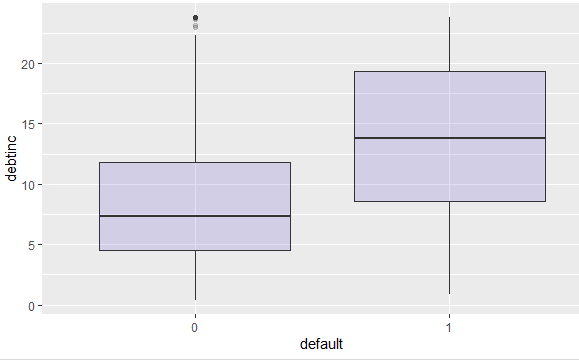
* Found few outliers in the data especially income, debtinc, creddebt, othdebt. So used winsorization method to handle the outliers.
* Plotted correlation plot and checked multicollinearity using T-test

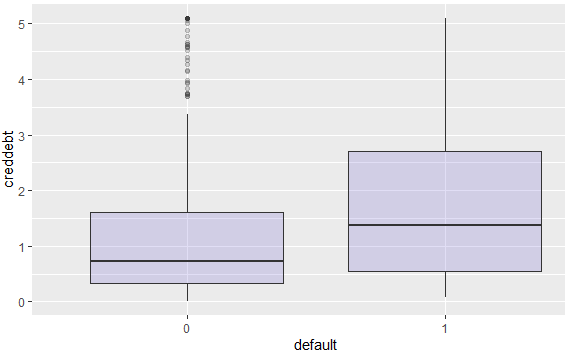
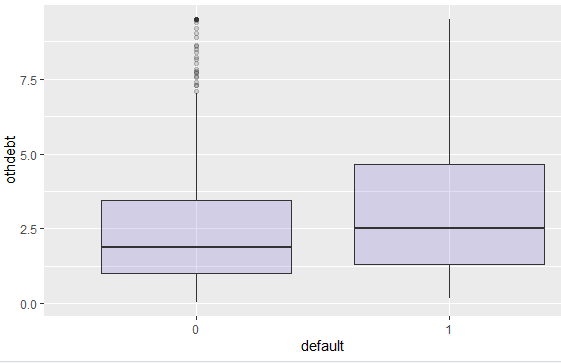


* Performed Independent T Test on each variable with 95% confidence level and found that all the variables are with in-significance level.

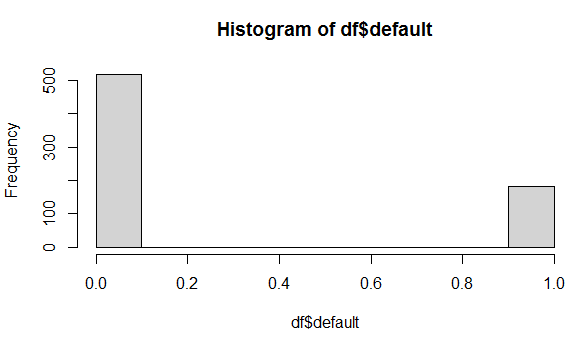


* Checked the effect of independent variables with the target variable using box plot and got few observations.
* Customers with lower values in age, geographic area, employment status, income are likely to default.
* Customers with higher values in individual’s debt payment, debt-to-credit ratio, any other debts are likely to default.

* Checked the distribution of default and non-defaulted customers in the dataset, to check whether the dataset is imbalanced or balanced data set.



* Found dataset is highly imbalanced with default value:



* Decided to use Generating Synthetic Minority Oversampling Technique (SMOTE) in order to avoid biasedness of the estimates and overfitting of the model.

**-** **Model Building & Performance Check**

* Splitted data into train and test using sample function and up-sample the default of training data using the SMOTE algorithm.

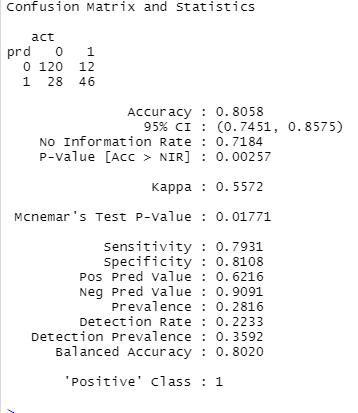


**Logistic Regression**

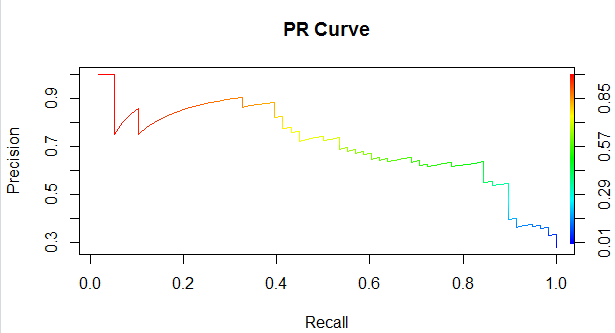
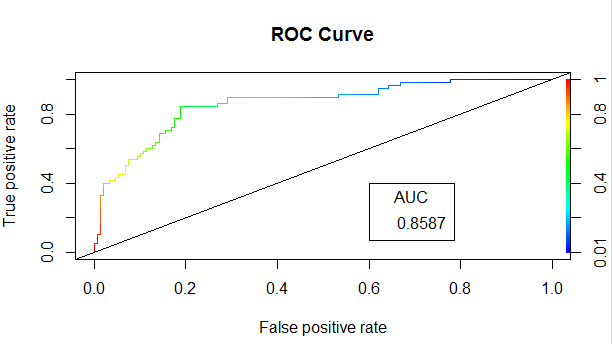
* Build the logistic regression model with all the variables and default probability for cut-off is taken as 0.5.
* Applied VIF test to find the multicollinearity between the variables, VIF Factor for these variables seems to be with in acceptance levels.



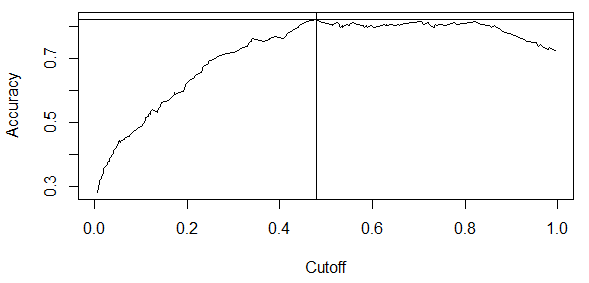
* Examined the model using confusion matrix, the overall accuracy of the default model is around 80.58% and recall/sensitivity score is 79.31%.

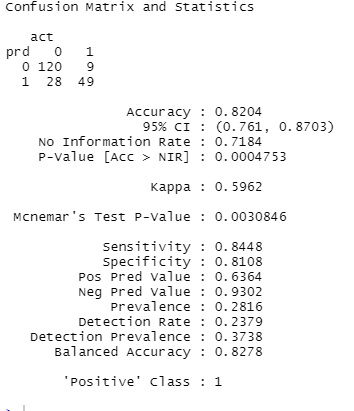

* Calculated Area Under the Curve score and plotted AUC - ROC Curve and Precision - Recall Curve

* Recall score of 79.31% is good. The objective of the model is to identify the customers who will default. In this case we need to find the optimum cut-off value.  
  
* Found the optimum cutoff value where the sensitivity and specificity is maximum: **0.4795718**

* Created confusion matrix report using this cut-off instead of default cutoff.



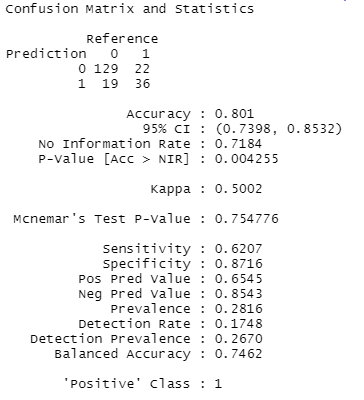


* The accuracy of the model with new cutoff is around 82.03% and recall/Sensitivity score is 84.48%.
* Overall accuracy of the model is increased from 80.58% to 82.03% by taking optimum cutoff as 0.4795718, Model performance i.e. recall score has increased from 79.31% to 84.48% and precision score/positive prediction value increased from 62.16% to 63.64%.

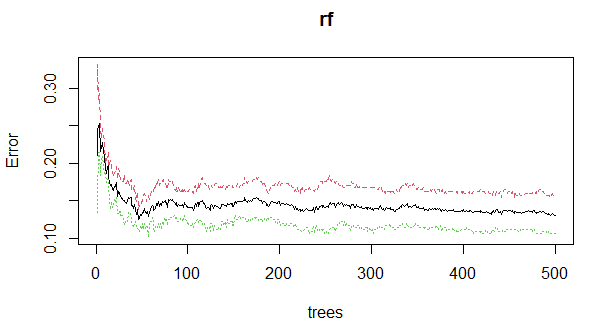
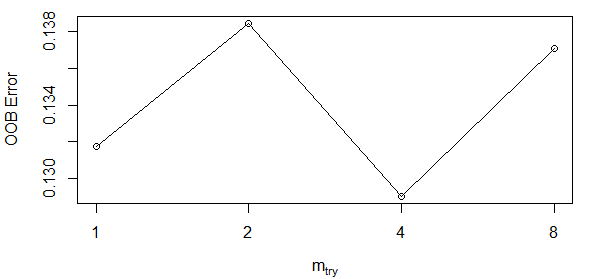
**Checking other models**

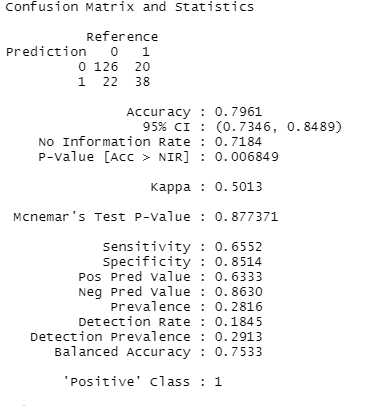
**Random Forest**

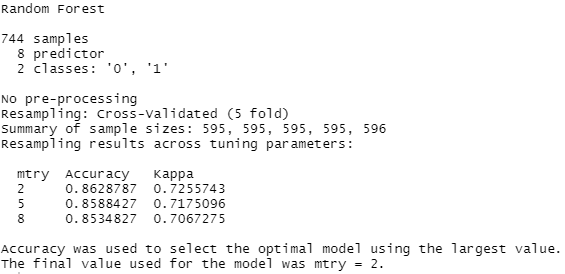
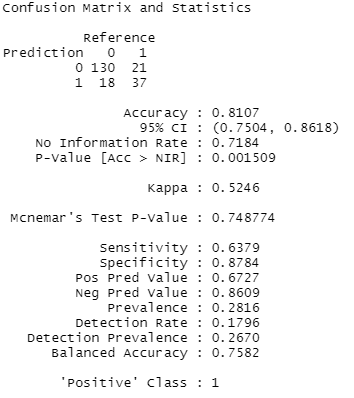
* Created Random Forest model and examined the model using confusion matrix, the overall accuracy of the model is around 80.09% and recall/sensitivity score is 62.07%.



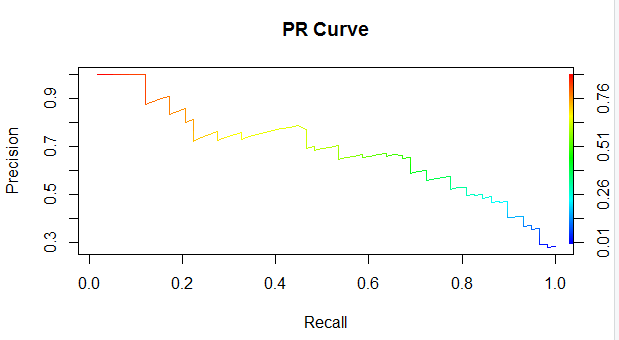


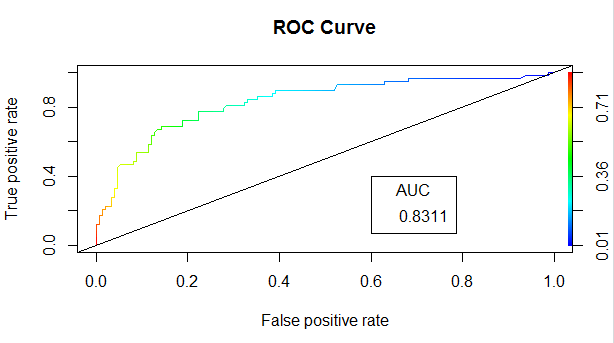
* Plotted error rate of Random Forest.  
   
* Tuned mtry and found mtry with the lower OOB error.  
  
* Trained the model with mtry=4. The overall accuracy of the model is now around 79.61% and recall/sensitivity score is 65.51%.

* Even after tuning random forest fall behind logistic regression.
* Checked other classification models like Decision Tree Classifier and SVM Classifier using cross validation.  
  
* Found Random Forest Classifier has more accuracy of 86.28% in cross validation.  
    
  
* Examined the Random Forest model using confusion matrix, the overall accuracy of the model is around 81.07% and recall/sensitivity score is 63.79%.  
     
  



* Calculated Area Under the Curve score and plotted AUC - ROC Curve and Precision - Recall Curve for Random Forest  
    
  



* Compared the precision, recall and F1 score of both the models.

Random Forest:



Logistic Regression (cutoff 0.4795718):



* Based on the F1-score (harmonic mean of precision and recall), logistic regression model (cutoff 0.4795718) with F1 score (for positive labels - default customers) of 0.726 is giving better results than random forest model with F1 score of 0.655. So we will use the **logistic regression model (cutoff 0.4795718)** to predict if the customer default or not.
* Save the model using saveRDS.



Model is now saved as ‘model.rds’

**Deployment**

* Created an app for using the model using Shiny library.
* Deployed the app on heroku.  
    
  App deployed: <https://bank-loan-default-rcode.herokuapp.com/>

Github repository: <https://github.com/avinashsajeevan/Bank-Loan-Default-Prediction--R>

**Summary**Banks play a big role in the market economies. In other to avoid another global financial crisis, it is important they give close attention to customer’s loan application, risk exposure and probability of default. Here, model building with Logistic regression seems very appropriate. The recall rate of 85.5% makes it best suitable model for bank loan default prediction. The high recall is better as the banks don’t want to lose money, and would be a good idea to alarm the bank even if there is a slight doubt about defaulter. It is important to mention that due to the relatively small sample size of the data which might not have been able to gain enough statistical and explanatory power. In the future data analysis, I’ll endeavor to apply large sample size.