**Problem Statement**

A Company wants a system of the loan eligibility criteria that based on customer detail when they provide information to the bank while filling the online application form. These information details consist of Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. They want a system that identifies the customers, those are eligible to get the loan from the bank.

## **Data Set**

### 

### Import the data

### First step is to import dataset from the system.

### To import data, we use this command.

### tr <- read.csv('train.csv', header = TRUE)

### Here tr is the variable where we are going to save our dataset.

### Summary of the dataset

### Next step is to look at the data. most of the time data is ambiguous, consist on errors and missing values.

### A picture containing text, receipt Description automatically generated

### We can see that there are NAs value in these variables Loan Amount, Loan\_Amount\_term and Credit History.

**Visualizing of the Data**

A screenshot of a map

Description automatically generated

Figure#1

A screenshot of a cell phone

Description automatically generated

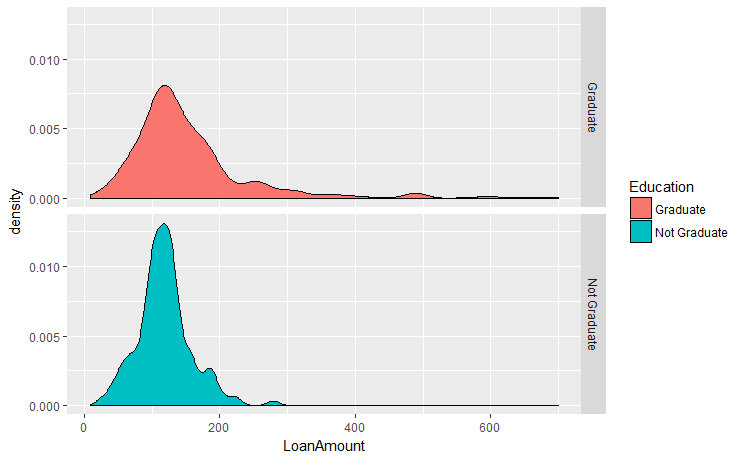
Figure#2

Now we take a look of numerical variables loan amount and the applicant income and draw the histograms and the boxplots of the variables:

A screenshot of a cell phone

Description automatically generated

Above boxplot and histogram shows that there are some extreme values in both variables. Let’s now we examine that the applicant’ loan amounts is affected by their educational level.



We noticed that graduates have more outliers and their loan amount distribution is wider.

Now let’s have a look at the categorical variables in the data.

**Categorical Variables**

A screenshot of a cell phone

Description automatically generated

If we look in the above Gender graph, we can notice that male’s applicant has more records, and more than half of the applications have been approved. And on the other side There are few females, less than males’ but more than half of their applications of the female applicant have been approved. Same as other graphs we can evaluate how each category performed in regard to the approval of the loan applications.

### Missing Values

### Now, we check the missing data in our dataset:

### By importing VIM library, we can obtain our missing values using these commands.

### library(VIM) mice\_plot <- aggr(tr, col=c('navyblue','red'),numbers=TRUE, sortVars=TRUE,

### labels=names(tr), cex.axis=.7,gap=3, ylab=c("Missing data","Pattern"))

### C:\Users\USER\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Missing Values.png

### We can see that in the chat there are seven variables that have some missing values.

### Tidying the data

Before start analysis, we need to fix errors and missing values in the data There are some missing values in a few variables. In distributions data, we noticed that Applicant Income and Loan Amount have outliers. Fixing outliers can be a tricky step in data preprocessing. In the data set, we found some missing values are on both numerical and categorical data, to handle these values, we are going to use the mice package in R. mice () function in mice package is used to imputing missing values from the data.

imputed\_Data <- mice(tr, m=2, maxit = 2, method = 'cart', seed = 500)

Here mice is a function, ‘m’ in the function shows how many rounds of imputation we want to do, ‘method’ argument indicates which of the methods for imputations we want to use. I used cart method that work with all type of variables. After imputing we merge the imputed data into our original dataset by using complete () function:

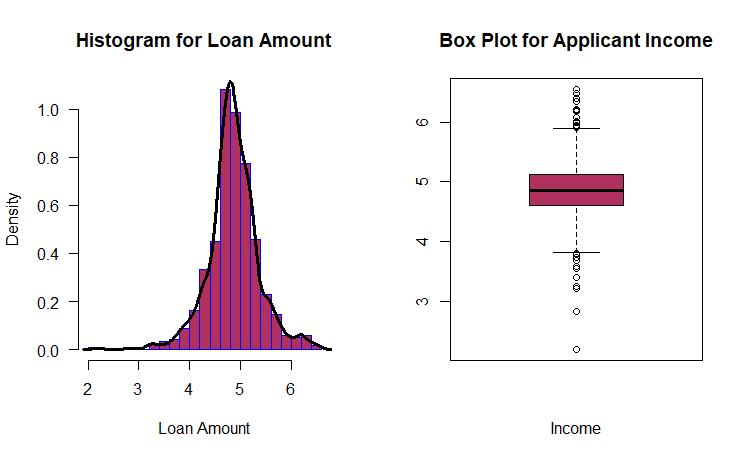
tr <- complete(imputed\_Data,2)

in complete function first argument is imputed data and second argument is 2 that tells the imputation round that we want to use in this function.

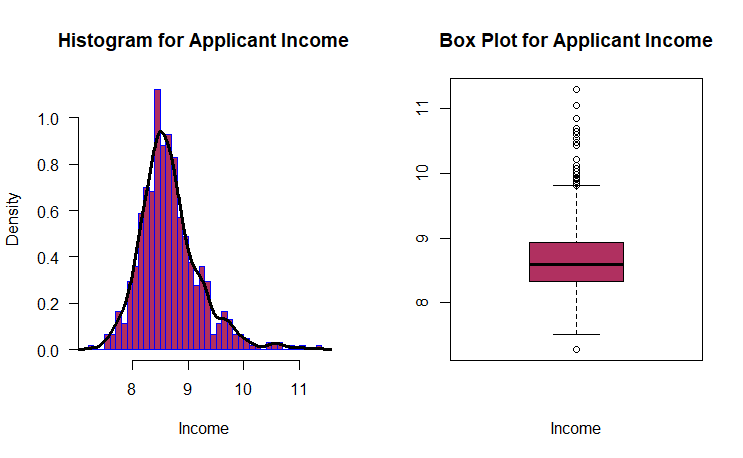
After merging the data, we can see that in our data set there is no missing data after the imputation:

sapply(tr, function(x) sum(is.na(x)))

Now It’s time to handle the extreme values. Looking at the Loan Amount variable, we can notice that high values are possible as just few customers, for some reason, or may they want to apply for higher loan amounts.



To see the extreme values, we will perform the log transformation method to normalize the data:



We can see that the normal distribution is better and closer in a normal distribution.

### Building Predictive Models

Now our next step is to apply model on our dataset. We are going to apply Logistic Regression and decision tree model.

### Logistic Regression

### Logistic Regression predicts the probability of occurrence of an event by fitting data to a logit function. Regression coefficients shows the change in mean of the variable for one unit of change in the predictor variable while holding other predictors in the model constant. In our logistic regression first, we do kitchen sink regression and put all variables in our model after that we do our model fit by using step wise regression model and will choose best variables for the model. That will be best fit variables for our model.

### Mod1 <- glm (Loan\_Status ~ Credit\_History+Education+Self\_Employed+Property\_Area+LogLoanAmount+ LogIncome,data = testnew, family = binomial)

### A screenshot of text Description automatically generated

### In the output we can notice that in our model p values of some variables are high than 0.05 .And ACI value of the model is 571 that is not better so we try to more reduce this values and find best fit variables in our model. When we reduce our variable by using step wise regression. We choose variables that will be best for our model.

### Mod2 <- glm (Loan\_Status ~ Credit\_History,data = testnew, family = binomial)

### A screenshot of text Description automatically generated

### In this output we can Cleary notice that by reducing our variables in the model we get can get best fit variables and good value of ACI that is 171 better that kitchen sink regression in which we use all variables. And accuracy of our model is Test data: 83.24% That shows our model is fit.

### Decision Tree

### Now we use another model on our dataset that is decision tree. Decision trees build a set of binary division on the independent variables to build a tree that will used to classify new observations into one of two groups.

### C:\Users\USER\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Decision tree.png

### In the above output tree, we can predict those customers that don’t have history but have income, their chances to get loan are more than those that have credit history. So, we can easily predict those customers who are willing to get the loan from the organization. Accuracy of the decision tree model is 85.4% that is good than regression model. so, decision tree model is best model than our regression model for our dataset.