ASSIGNMENT ON

FINANCE AND RISK ANALYTICS

Done & Submitted by

Akshay Subramanian

Table of Contents

OVERVIEW AND OBJECTIVE OF THE DATASET

We are provided with two excel sheets (datasets) namely:

1. **raw-data** which consists of 3541 records and 52 columns **.**
2. **validation_data** which consists of 715 records and 52 columns.

```
> company <- read_excel("raw-data.xlsx", sheet = "raw data")
> dim(company)
[1] 3541   52
> validation <- read_excel("validation_data.xlsx", sheet = "valdata")
> dim(validation)
[1] 715  52
```

The objective of the dataset is to:

1. Create a new variable "**Default**" in the raw-data based on "**Net worth next year**" column.
2. Handle the missing data by imputing and then treat the outliers.
3. Create a Credit-risk model (Default) using the **logistic regression** technique.
4. Evaluate the model on the "**validation_data**" to find out how effective the model is.

IMPORTING THE DATASETS

In order to access the dataset from the directory where it has been stored, the working directory must be set.

```
> setwd("D:\\BABI\\BABI-18th Residency\\Assignment")
> getwd()
[1] "D:/BABI/BABI-18th Residency/Assignment"
```

Then both the datasets namely the **raw-data** and **validation_data** must be loaded.

```
> company <- read_excel("raw-data.xlsx", sheet = "raw data")
> dim(company)
[1] 3541   52
> validation <- read_excel("validation_data.xlsx", sheet = "valdata")
> dim(validation)
[1] 715  52
>
```

LOADING THE REQUIRED LIBRARIES

In order to work further on the datasets, the required libraries must be installed and loaded. The key library files that have to be loaded are:

1. **readxl**, 2. **writexl**, 3. **ggplot2**, 4. **DataExplorer**, 5. **lmtest**, 6. **corrplot**, 7. **explore**, 8. **caret**

```
# loading the library files #

library(readxl)
library(writexl)
library(dplyr)
library(readr)
library(kableExtra)
library(ggplot2)
library(naniar)
library(visdat)
library(corrplot)
library(StatMeasures)
library(crayon)
library(gridExtra)
library(DataExplorer)
library(lattice)
library(mlr)
library(explore)
library(lmtest)
library(DMwR)
library(HH)
library(pROC)
library(e1071)
library(caret)
```

CREATION OF A NEW VARIABLE IN raw-data DATASET

A new variable called "**Default**" is created based on the column "**Networth Next Year**". If the next year's networth of the company is less than 0, then the company will be prone to be a Defaulter else, the company is not prone to be a Defaulter. The same is done in the creation of the variable.

```
# creation of a new variable "Default" #

company$Default <- ifelse(company$`Networth Next Year`>0,0,1)
str(company$Default)
company$Default <- as.factor(company$Default)
```

Since the "**Default**" variable is already present in the **validation_data** dataset, it is converted into a factor variable.

```
# renaming the "Default-1" column in validation dataset and then converting the datatype of the same #

validation = validation%>% rename(Default = "Default - 1")
str(validation$Default)
validation$Default <- as.factor(validation$Default)
```

DATA HANDLING

On preliminary examination of both the **raw-data** and **validation_data** datasets, we can see that the column "**Deposits (accepted by commercial banks)**" doesn't have any records. Hence, we can remove it from both of the datasets.

```
# Removing the "Deposits column" in company and validation dataset #

company <- company[,-22]
validation <- validation[,-22]
```

Also, on looking into both the datasets, we can see that some columns are on a different datatype apart from being numeric. The columns that are not numeric are:

1. PE on BSE
2. Creditors turnover
3. Debtors turnover
4. Finished goods turnover
5. WIP turnover
6. Raw material turnover
7. Shares outstanding and
8. Equity face value

Hence, the above variables or columns have to be converted into numeric datatype in both **raw-data** and **validation_data** datasets.

```
######### Datatype conversion ##########

# company dataset #

company$`PE on BSE` <- as.numeric(company$`PE on BSE`)
company$`Creditors turnover` <- as.numeric(company$`Creditors turnover`)
company$`Debtors turnover` <- as.numeric(company$`Debtors turnover`)
company$`Finished goods turnover` <- as.numeric(company$`Finished goods turnover`)
company$`WIP turnover` <- as.numeric(company$`WIP turnover`)
company$`Raw material turnover` <- as.numeric(company$`Raw material turnover`)
company$`Shares outstanding` <- as.numeric(company$`Shares outstanding`)
company$`Equity face value` <- as.numeric(company$`Equity face value`)

# validation dataset #

validation$`Creditors turnover` <- as.numeric(validation$`Creditors turnover`)
validation$`Debtors turnover` <- as.numeric(validation$`Debtors turnover`)
validation$`Finished goods turnover` <- as.numeric(validation$`Finished goods turnover`)
validation$`WIP turnover` <- as.numeric(validation$`WIP turnover`)
validation$`Raw material turnover` <- as.numeric(validation$`Raw material turnover`)
validation$`Shares outstanding` <- as.numeric(validation$`Shares outstanding`)
validation$`Equity face value` <- as.numeric(validation$`Equity face value`)
validation$`PE on BSE` <- as.numeric(validation$`PE on BSE`)
```

PERCENTAGE OF MISSING RECORDS IN EVERY COLUMN

On looking at the summary of our raw-data dataset, we can see that there are blank values in various columns of the dataset. Hence, we run a function to create a .csv file that tells us how much percentage of the missing values are there in each column. (*Refer Annexure 1 - "missing_cols.csv"*)

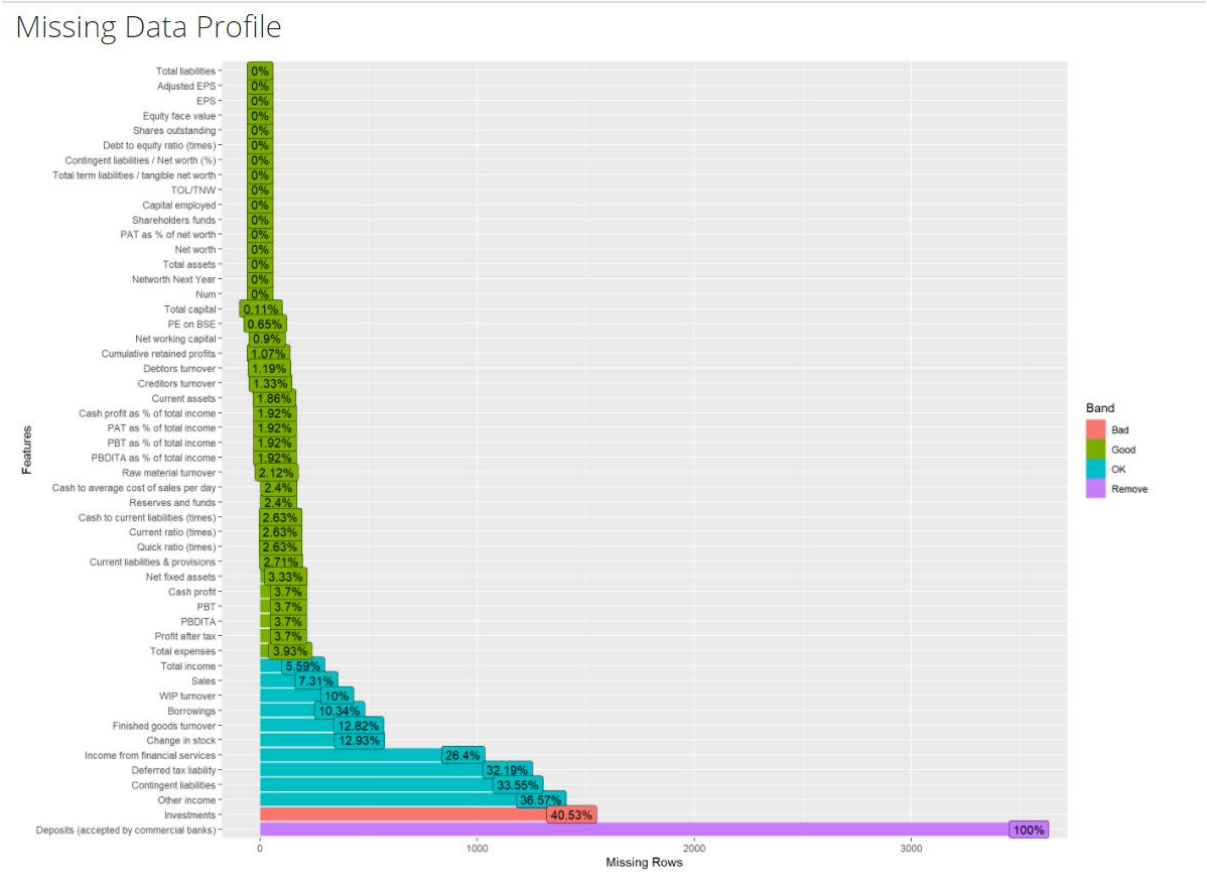
```
# percentage of missing values in each column #

contains_any_na <- sapply(company, function(x) {
  count<-as.integer(sum(is.na(x)),length=0)
  percentage<-round(count/NROW(x),4)*100
  datatype<-class(x)
  frame<-c(datatype,count,percentage)
}
)

missing_frame <- t(contains_any_na)
dim(missing_frame)
colnames(missing_frame) <- c("datatype","missing records count", "percentage")
print(missing_frame)
write.csv(missing_frame,"missing_cols.csv")
```

On looking into the .csv file, the top five columns (**more than 30 percent**) that have more missing records are:

Table 1: Top five columns of missing records

	datatype	missing records count	percentage
PE on BSE	numeric	2194	61.96
Investments	numeric	1435	40.53
Other income	numeric	1295	36.57
Contingent liabilities	numeric	1188	33.55
Deferred tax liability	numeric	1140	32.19

Hence, the above five columns have majority of the missing records; these columns can be removed along with the "**Networth Next Year**" and "**Num**" columns in both the datasets.



TREATING OF OUTLIERS AND MISSING VALUES

After removing the above columns, the remaining has to be treated for the missing values and the outliers have to be treated.

1. For the **missing values**, the columns have to be imputed with the **median values**.
2. For the **outlier treatment**, the outliers on the left side (less than one percentile) are capped at **1 percentile**. The outliers on the right side (more than 99 percentile) are capped at **99 percentile**.

```r
# outlier function #

outlierpercentage <- function(dt, var) {
  var_name <- eval(substitute(var),eval(dt))
  tot <- sum(!is.na(var_name))
  na1 <- sum(is.na(var_name))
  mean1 <- mean(var_name, na.rm = T)
  outlier <- boxplot.stats(var_name)$out
  mo <- mean(outlier)
  var_name <- ifelse(var_name %in% outlier, NA, var_name)
  na2 <- sum(is.na(var_name))
  cat(paste("Outliers identified: ",(na2 - na1), " from ", tot, " observations \n"))
  cat(paste("Proportion (%) of outliers: ", round((na2 - na1) / tot*100,3),"\n"))
  cat(paste("Number of NA's: ",na1,"\n"))
  cat(paste("NA percentage is:", round(na1/(tot+na1)*100,3),"\n"))
  cat(paste("Mean of the outliers: ", mo,"\n"))
  mean2 <- mean(var_name, na.rm = T)
  cat(paste("Mean without removing outliers: ", round(mean1,3),"\n"))
  cat(paste("Mean if we remove outliers: ", round(mean2,3),"\n"))

}

# replace median function #

replace_median<-function(df,colname){
  colname <- eval(substitute(colname),eval(df))
  colname <- ifelse(is.na(colname),median(colname, na.rm = TRUE),colname)
  return (colname)
}

# treat outliers #

treat_outliers<-function(df,colname){
  colname <- eval(substitute(colname),eval(df))
  colname<-ifelse(colname>=quantile(colname,0.99),quantile(colname,0.99),colname)
  colname<-ifelse(colname<=quantile(colname,0.01),quantile(colname,0.01),colname)
  return (colname)
}
```

The above three functions defined are:

1. Finding the percentage of outliers.
2. Imputing the median value in the place of blank values.
3. Treating the outliers.

Percentage of Outliers:

The percentage of outliers is computed by counting the number of outliers on the actual number of records (excluding the blank values). Also, additional details like “Mean including the blank values” and “Mean after imputing the blank values” are given in order to get an understanding of how the imputing of the median values in the blank records change the skewness of the data.

Imputing the median value in the place of blank values:

The blank records as such cannot be used in creation of model. Hence, the blank records in various columns are imputed with the median values.

Treating the Outliers:

Since the outliers affect the performance of logistic regression, we are capping the outliers. In our problem, we are capping the outliers which are greater than 99 percentile at 99th percentile, and the outliers that are lesser than 1 percentile at 1st percentile. By this, the maximum value and the minimum value of the columns can be capped and the outliers can also be reduced.

EXPLORATORY DATA ANALYSIS

(Refer Annexure 2 – "EXPLORATORY DATA ANALYSIS.docx")

Profit after Tax

```
> outlierpercentage(company1, `Profit after tax`)
Outliers identified:  577  from  3410  observations
Proportion (%) of outliers:  16.921
Number of NA's:  131
NA percentage is: 3.7
Mean of the outliers:  1560.80103986135
Mean without removing outliers:  277.36
Mean if we remove outliers:  15.961
```

On looking at the above summary, we can see that 16.921% of the recorded observations are considered as outliers. Also, 131 records (3.7% of the total records) are blank values. Also, if the outliers are treated, we can observe that the mean is reduced from 277.36 to 15.961.

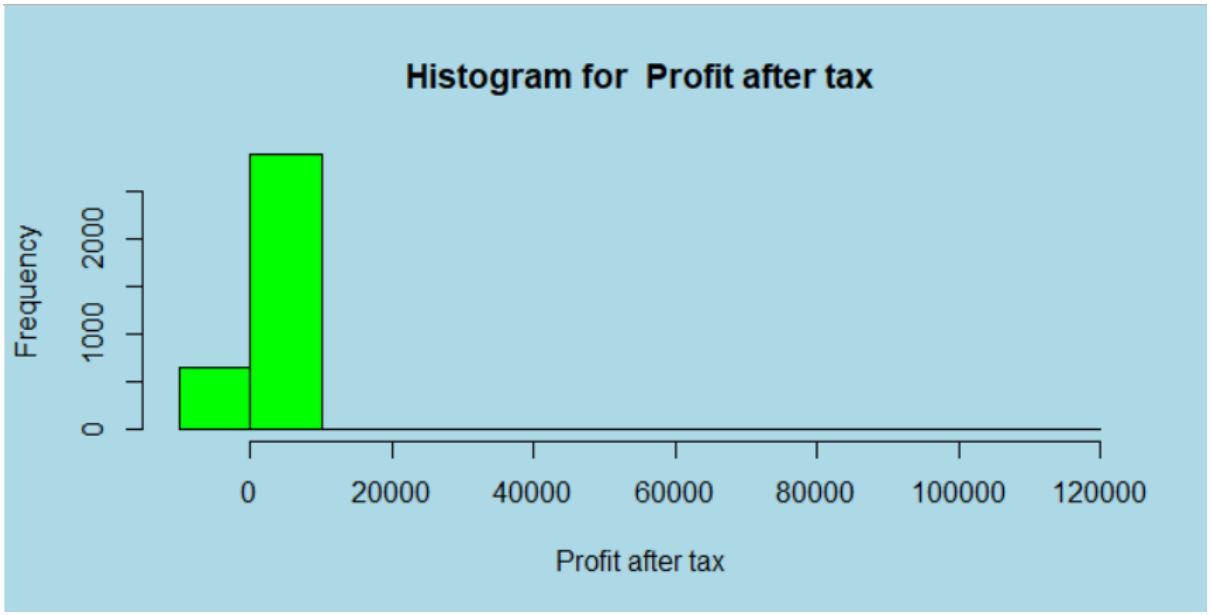
```
summary before treatment
> summary(company1$`Profit after tax`)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
 -3908.30      0.50      8.80    277.36     52.27 119439.10       131
```

On looking into the summary, we can see that the maximum value is at 119439.10 and the minimum value is at -3908.30. Also, the median and mean value is at 8.8 and 277.36 respectively.

```
> company1$`Profit after tax`<-replace_median(company1,`Profit after tax`)
> company1$Profit.after.tax<-treat_outliers(company1,`Profit after tax`)
> cat("summary after treatment")
summary after treatment
> summary(company1$`Profit after tax`)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
 -3908.3      0.6      8.8    267.4     48.1 119439.1
```

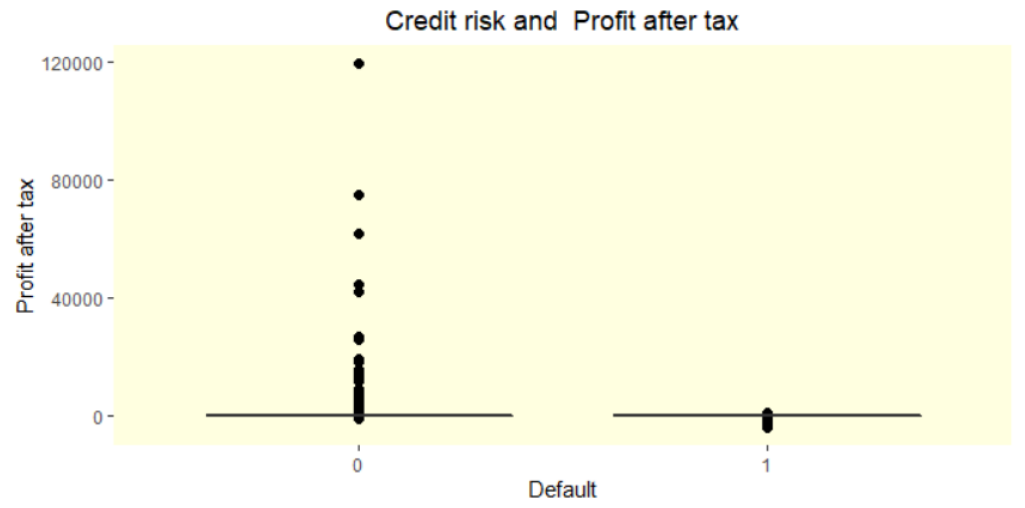
After imputing with blank values and treating the outliers, we can see that the 1st quartile and 3rd quartile is changed to 0.6 and 48.1 respectively. Also, the mean has been reduced to 267.4. Whereas, the minimum and the maximum value remains unchanged.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that the companies have the profit in between less than 0 and 12000.

BIVARIATE ANALYSIS



On looking at the graph, we can say that the companies have less profit after tax deductions are prone to default the loans.

PBT

```
> outlierpercentage(company1, PBT)
Outliers identified:  576  from  3410  observations
Proportion (%) of outliers:  16.891
Number of NA's:  131
NA percentage is: 3.7
Mean of the outliers:  2163.603125
Mean without removing outliers:  383.81
Mean if we remove outliers:  22.074
```

On looking at the executive summary, we can see that 16.891% of the recorded observations (576 observations) are observed as outliers. Also, 3.7% of the total records (131 records) are blank records. Also, if all the outliers are removed, the mean value changes from 383.81 to 22.074.

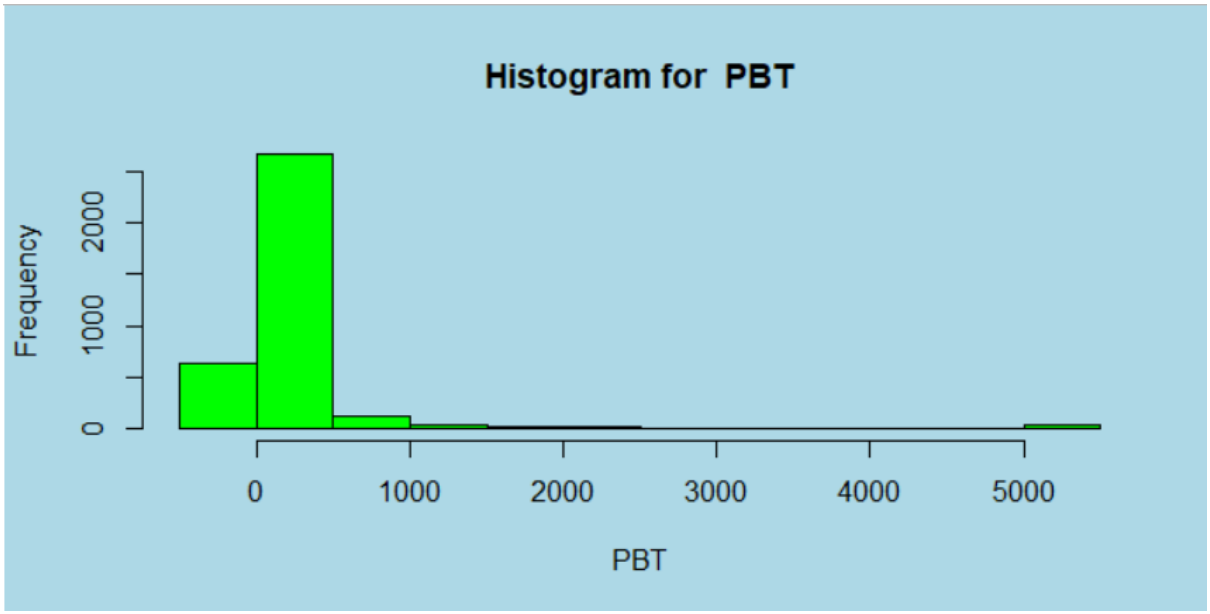
```
summary before treatment
> summary(company1$PBT)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
 -3894.80      0.70     12.40    383.81     71.97 145292.60       131
```

On looking at the summary, we can see that the maximum value is at 145292.60 and the minimum value is at -3894.80.

```
> company1$PBT<-replace_median(company1,PBT)
> company1$PBT<-treat_outliers(company1,PBT)
> cat("summary after treatment")
summary after treatment
> summary(company1$PBT)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -191.7     0.9    12.4   172.3    67.5  5421.4
```

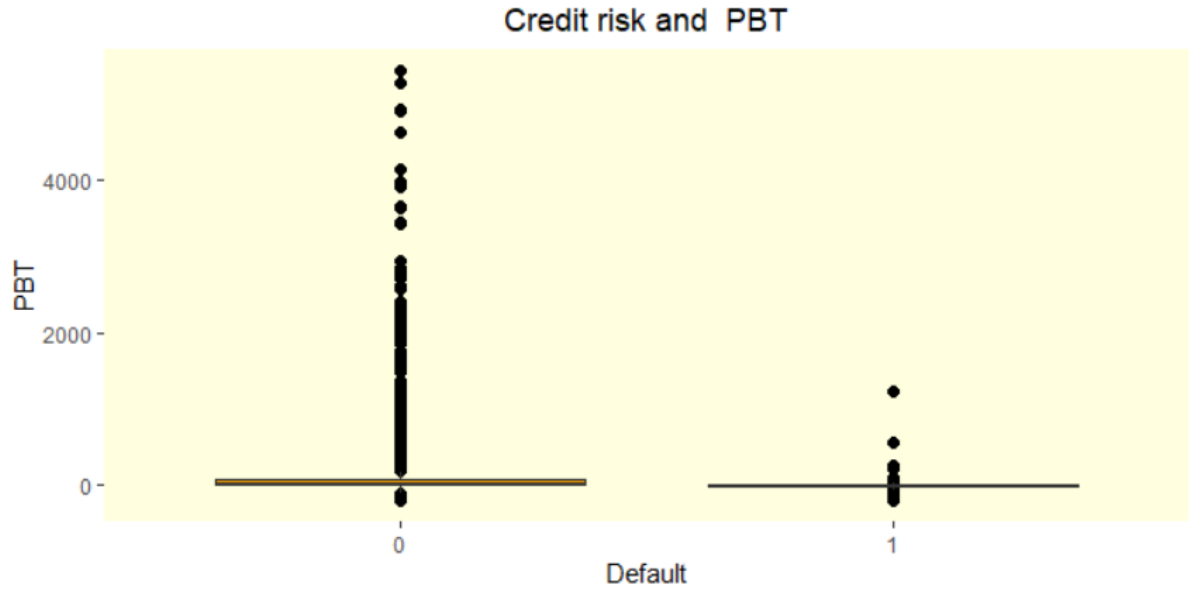
On looking at the summary, we can see that the maximum value is capped at 5421.4 and the minimum value at -191.7. Also, the mean is reduced from 383.81 to 172.3.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that majority of the companies have profit before tax in between 0 and 500. Also, some companies have negative profit even before taxes are deducted. Some companies have profit more than 5000. Those companies are seen as outliers.

BIVARIATE ANALYSIS



On looking at the graph, we can see that the companies having less profit even before taxes are being deducted are more prone to default.

Cash Profit

```
> outlierpercentage(company1, `Cash profit`)
Outliers identified:  515  from  3410  observations
Proportion (%) of outliers:  15.103
Number of NA's:  131
NA percentage is: 3.7
Mean of the outliers:  2410.89242718447
Mean without removing outliers:  392.065
Mean if we remove outliers:  32.93
```

On looking at the summary, we can see that the 515 of the recorded observations are identified as outliers. Also, 131 records are blank values. These blank values constitute of 3.7% of the total number of records.

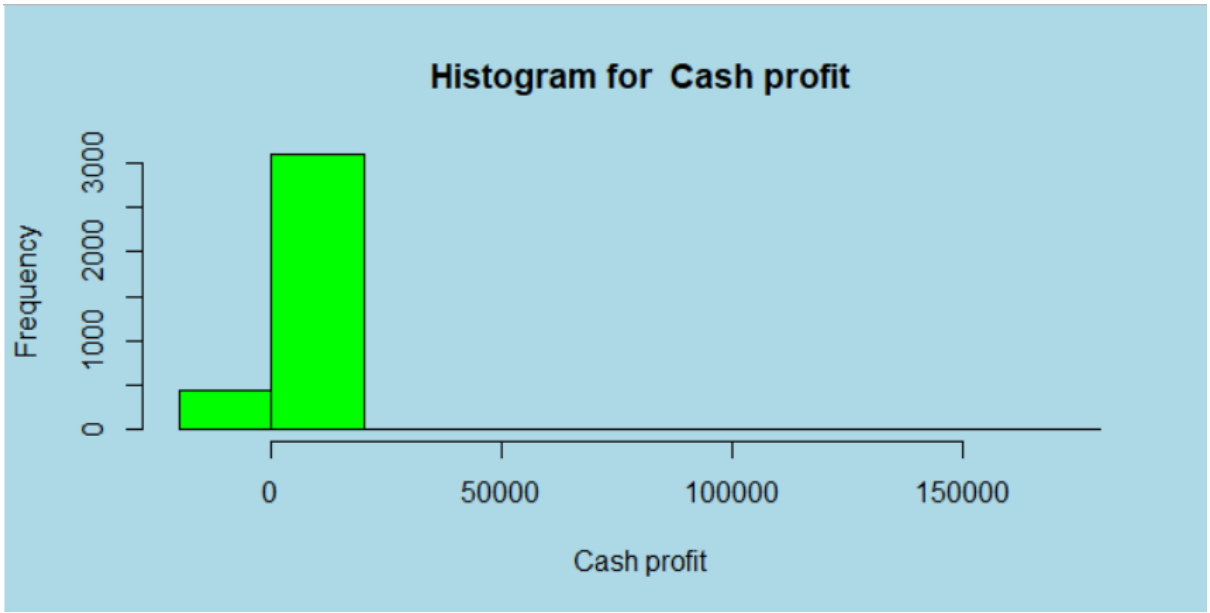
```
summary before treatment
> summary(company1$`Cash profit`)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
 -2245.70      2.90     18.85    392.07     93.20 176911.80       131
```

On looking at the summary, we can see that the maximum value is 176911.80 and the minimum value is -2245.70. The median and mean values are 18.85 and 392.07 respectively.

```
> company1$`Cash profit`<-replace_median(company1,`Cash profit`)
> company1$Cash.profit<-treat_outliers(company1,`Cash profit`)
> cat("summary after treatment")
summary after treatment
> summary(company1$`Cash profit`)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
 -2245.70      3.10     18.85    378.26     86.80 176911.80
```

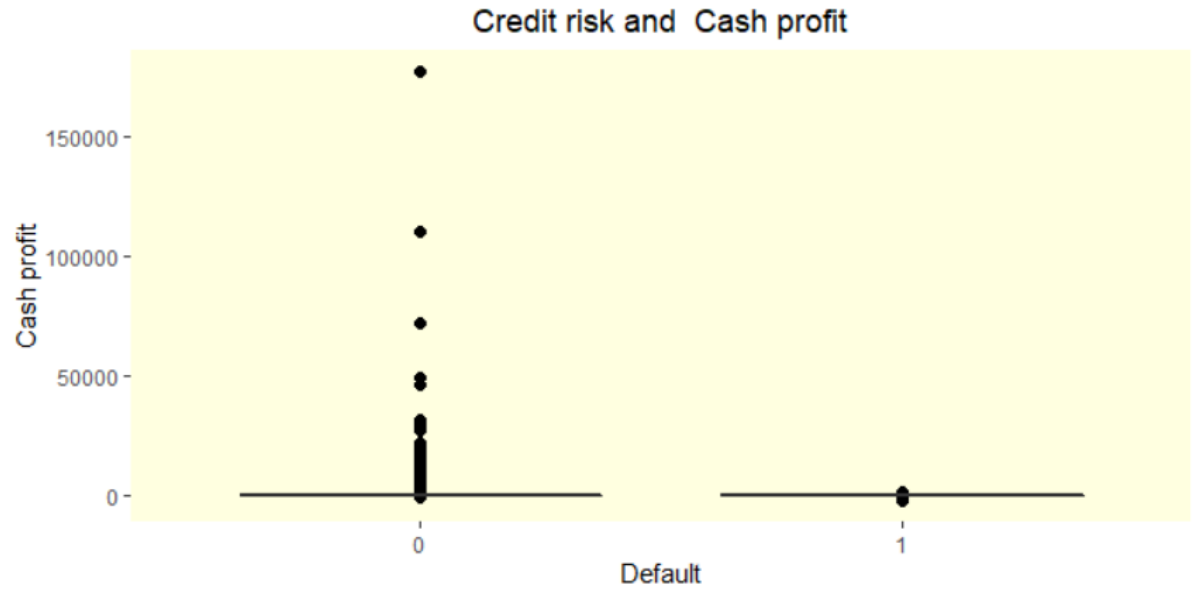
Once the blank values are imputed and the outliers are treated, we can see that the mean value is reduced from 392.07 to 378.26. The maximum and minimum values are unchanged.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that the majority of the companies have cash profit in between 0 and 2000. Also, some companies have negative profit which means those companies have incurred loss.

BIVARIATE ANALYSIS



On looking at the graph, we can see that the companies having less Cash profit or negative Cash profit are prone to loan default.

Borrowings

```
> outlierpercentage(company1, Borrowings)
Outliers identified:  432  from  3175  observations
Proportion (%) of outliers:  13.606
Number of NA's:  366
NA percentage is: 10.336
Mean of the outliers:  7310.75162037037
Mean without removing outliers:  1122.279
Mean if we remove outliers:  147.646
```

On looking at the executive summary, we can see that 13.606% of the recorded observations are identified as outliers. Also, 10.336% of the total observations are blank values.

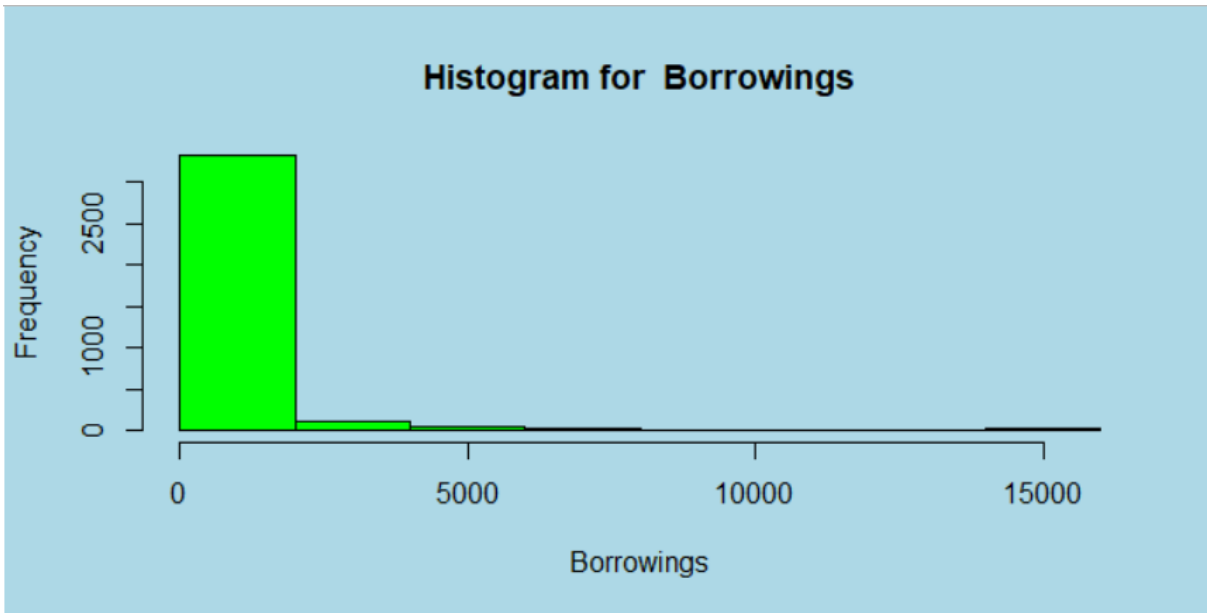
```
summary before treatment
> summary(company1$Borrowings)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
     0.10     23.95     99.20   1122.28    352.60 278257.30       366
```

From the summary, we can see that the minimum and maximum values are 0.10 and 278257.30 respectively. There are 366 blank values in the column.

```
> company1$Borrowings<-replace_median(company1,Borrowings)
> company1$Borrowings<-treat_outliers(company1,Borrowings)
> cat("summary after treatment")
summary after treatment
> summary(company1$Borrowings)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.2    29.7    99.2   600.7   296.0 14803.4
```

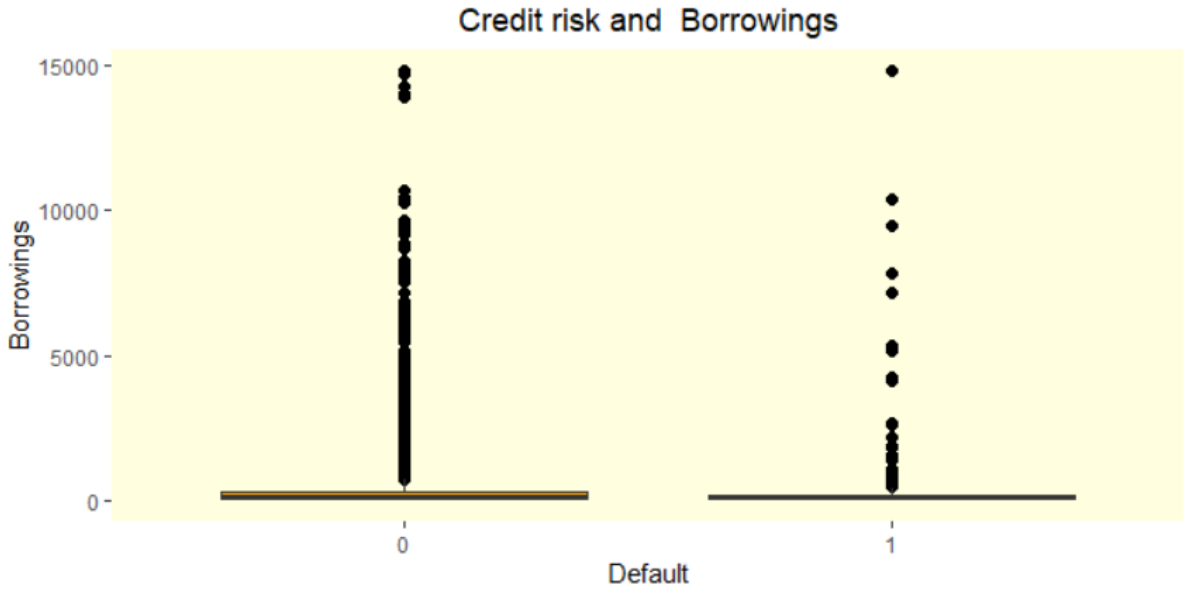
After the blank value imputation and outlier treatment, we can see that the mean is 600.7 and the maximum value is capped at 14803.4. The minimum value is capped at 0.2.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that majority of the companies have borrowings between 0 and 2000. Companies having borrowings more than 15000 are considered as outliers.

BIVARIATE ANALYSIS



On looking at the graph, we can see that even though some companies have high Borrowings, they are not prone to default.

Sales

```
> outlierpercentage(company1, Sales)
Outliers identified:  417  from  3282  observations
Proportion (%) of outliers:  12.706
Number of NA's:  259
NA percentage is: 7.314
Mean of the outliers:  31422.3503597122
Mean without removing outliers:  4549.52
Mean if we remove outliers:  638.187
```

On looking at the above screenshot, we can see that 12.706% of the observed records are identified as outliers. Also, 259 records out of 3541 records are blank values.

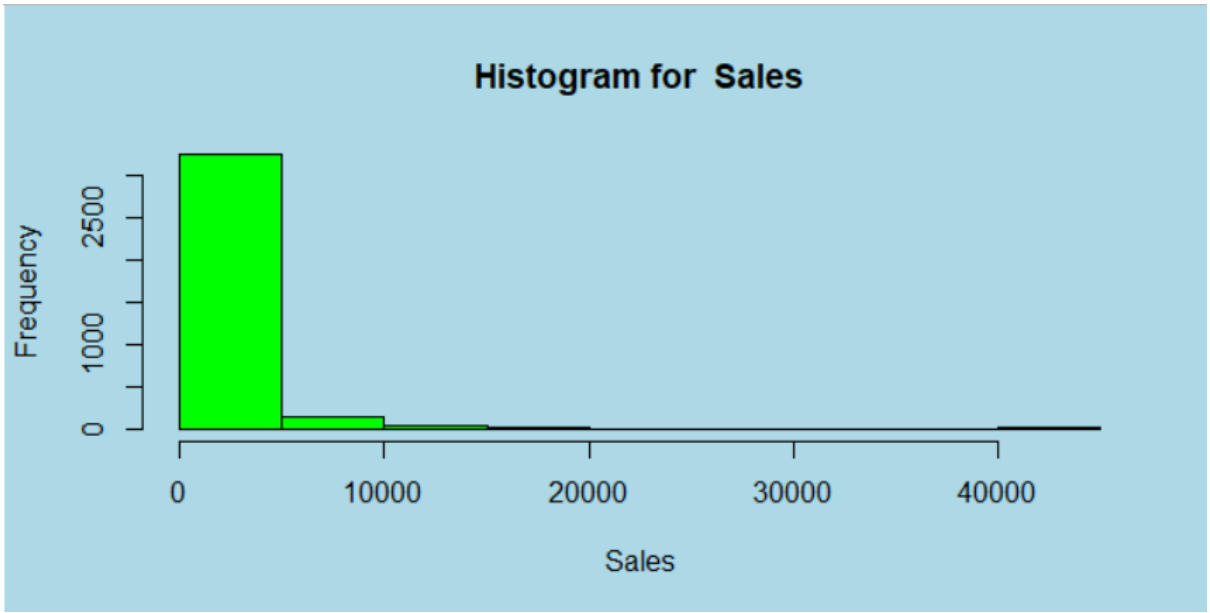
```
summary before treatment
> summary(company1$Sales)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
     0.1    112.7    453.1   4549.5   1433.5 2384984.4      259
```

From the above summary, we can see that the minimum value is 0.1 and the maximum value is 2384984.4. The mean and median are 4549.5 and 453.1 respectively.

```
> company1$Sales<-replace_median(company1,Sales)
> company1$Sales<-treat_outliers(company1,Sales)
> cat("summary after treatment")
summary after treatment
> summary(company1$Sales)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.5   133.3   453.1  1987.5  1314.7 40605.1
```

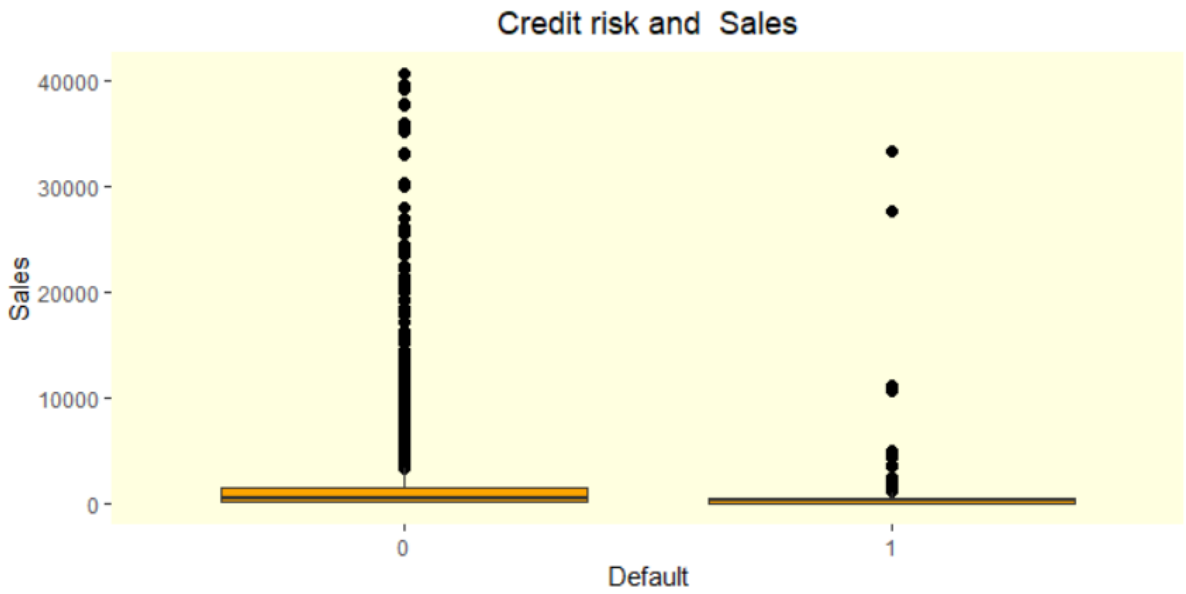
After the imputations and outlier treatments, we can see that the maximum value is capped at 40605.1 and the minimum value is capped at 0.5. The mean of the Sales is at 1987.5.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that majority of the companies have sales between 0 and 1000. The companies having sales more than 4000 are considered as outliers.

BIVARIATE ANALYSIS



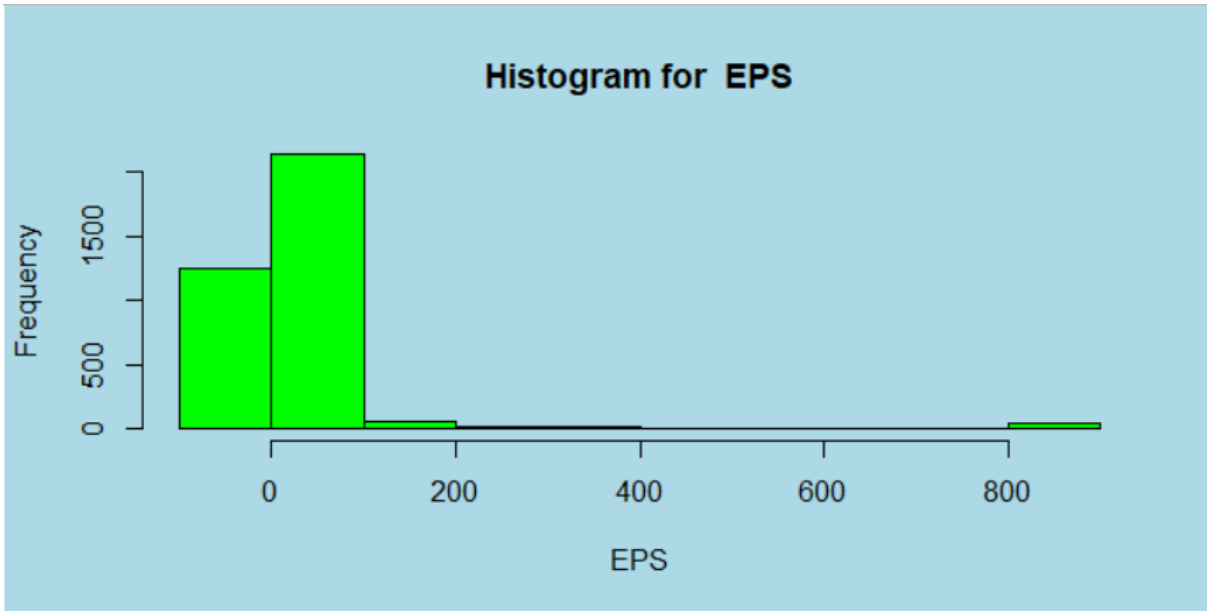
By looking at the graph, we can say that the companies having fewer Sales are prone to loan default.

EPS

```
> outlierpercentage(company1, EPS)
Outliers identified:  540  from  3541  observations
Proportion (%) of outliers:  15.25
Number of NA's:  0
NA percentage is: 0
Mean of the outliers:  -1461.85388888889
Mean without removing outliers:  -220.316
Mean if we remove outliers:  3.086
> cat("summary before treatment")
summary before treatment
> summary(company1$EPS)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-843181.8       0.0       1.4    -220.3       9.6   34522.5
> company1$EPS<-treat_outliers(company1,EPS)
> cat("summary after treatment")
summary after treatment
> summary(company1$EPS)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -60.32    0.00    1.43   25.91    9.62  896.14
```

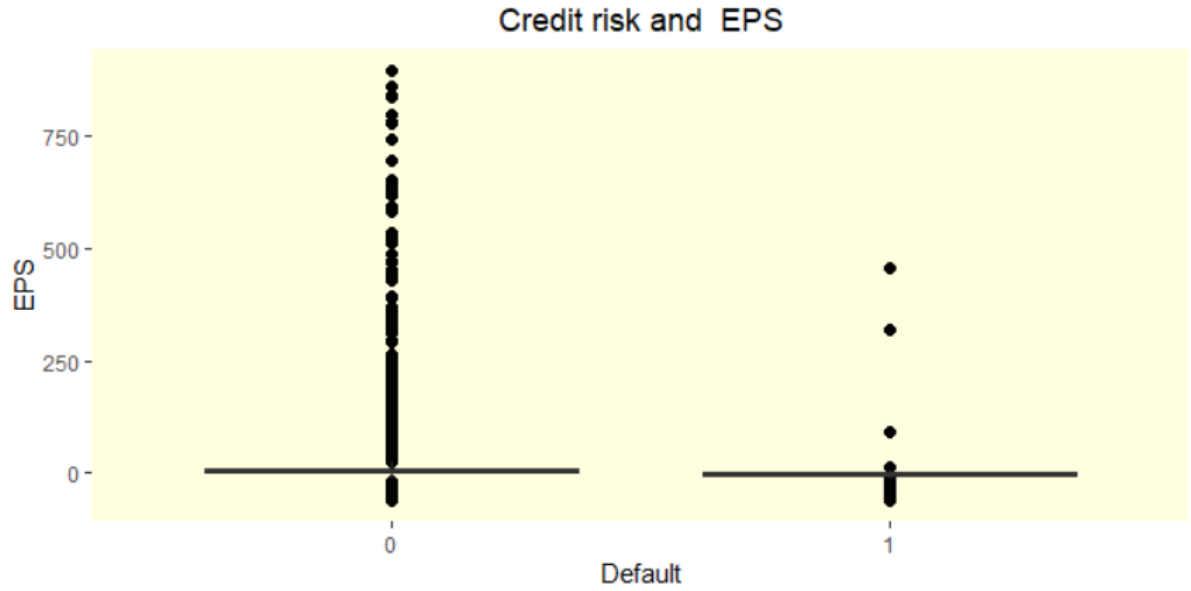
On looking at the summaries, we can see that 15.25% of the observations are identified as outliers. This column doesn't have any blank values. After the outliers are treated, we can see that the maximum value is capped at 896.14 and the minimum value at -60.32. The mean and the median values are at 25.91 and 1.43 respectively.

UNIVARIATE ANALYSIS



On looking at the histogram, we can see that the majority of the companies have EPS in between -100 and 200. The companies having EPS more than 800 are considered as outliers.

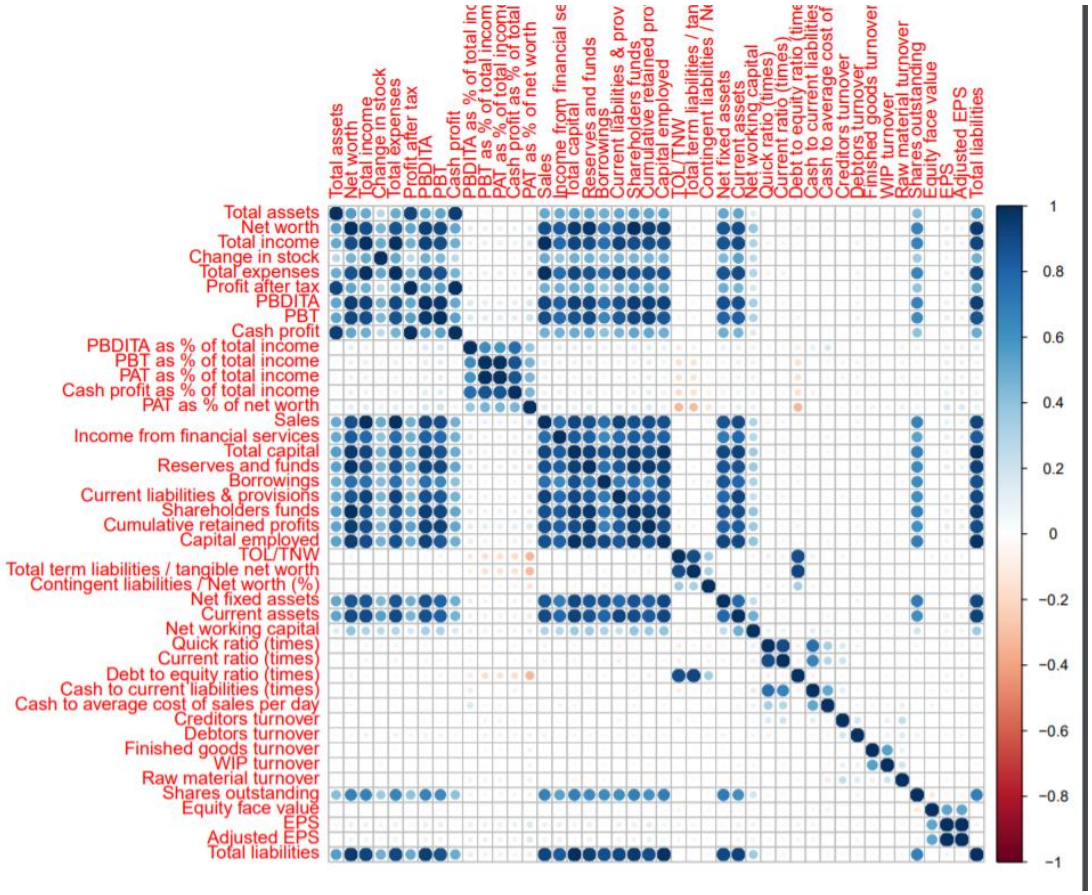
BIVARIATE ANALYSIS



From the above graph, we can say that the companies with low EPS value are tending to default. On the contrary, even though some companies have low EPS value, they are not prone to default.

CORRELATION MATRIX

Excluding the "Default" variable from the company dataset, we run a correlation command in order to find how one column is correlated to another. *(Refer Annexure 3 – "correlation_matrix.pdf")*



From the above correlation plot, we can see that:

1. **Debt to equity ratio (times)** column is highly correlated (positive correlation) to **TOL/TNW** and also to **Total term liabilities/tangible net worth**.
2. **Equity face value** is correlated (positive correlation) to **EPS** and **Adjusted EPS**.
3. **Net fixed assets** is highly correlated to **Total liabilities**.
4. **TOL/TNW** is negatively correlated to **PAT as % of net worth**.

GROUPING OF COLUMNS

As described in the problem statement, there are columns that fall under the categories:

1. Profitability
2. Leverage
3. Liquidity
4. Company's size

The below table consists of the count of columns that are under the above mentioned categories.

Table 2 – Count of columns under each category

Category	no. of columns	no. of columns removed due to high blank records
Profitability	2	no columns removed
Leverage	6	no columns removed
Liquidity	12	2 columns removed
Company's size	12	1 column removed

Also, grouping the number columns under the above mentioned categories:

Table 3 – Columns under the mentioned categories

Column names	Category
EPS	Profitability
Adjusted EPS	
Borrowings	Leverage
TOL/TNW	
Total term liabilities / tangible net worth	
Contingent liabilities / Net worth (%)	
Current ratio (times)	
Total liabilities	
Current liabilities & provisions	Liquidity
Quick ratio (times)	
Debt to equity ratio (times)	

Cash to current liabilities (times)	
Cash to average cost of sales per day	
Creditors turnover	
Debtors turnover	
Finished goods turnover	
WIP turnover	
Raw material turnover	
Total assets	Size
Net worth	
Total income	
Change in stock	
Sales	
Total capital	
Shareholders funds	
Capital employed	
Net fixed assets	
Current assets	
Shares outstanding	
Equity face value	

IMPUTATION OF VALUES IN VALIDATION DATASET

On looking at the validation dataset, we can see that there are blank values in the validation dataset too. Since the imputation will bias the data, we are considering the blank values as 0s.

```
#replacing the zeroes in the blank values of validation dataset

sapply(validation1, function(x){sum(is.na(x))})
validation1[is.na(validation1)]=0  #replace the zeroes in validation dataset
any(is.na(validation1)) #re-checking for null values
```

MODELLING PART

Since, the target variable "**Default**" is categorical in nature with two levels (**either 1 or 0**), the **logistic regression** technique can be used to device a credit risk model.

First model

At first, the model is created with all the columns or variables into consideration. And then, the multi-collinearity of the variables is measured by running the "**vif**" command.

```
#Model with all the columns

lg_model1 <- glm(Default~., data = company1, family = binomial())
summary(lg_model1)
vif(lg_model1)
```

For the "vif" values, Refer (*"Annexure 4 – vif values_LR.xlsx"*).

Second model

Since the vif value of "Total liabilities" is **Inf**, the column is removed and again the model is run.

```
#Removing the "Total liabilities" column

lg_model2 <- glm(Default~. -`Total liabilities`, data = company1, family = binomial())
summary(lg_model2)
vif(lg_model2)
```

Again, to check for multi-collinearity, the "**vif**" command is run to find out which variable exhibits highest vif value.

Likewise, the same procedure is done for every other column with highest vif values. After that, all the columns whose vif values are less than 20 are taken into consideration.

Twelfth model

The columns or variables that are removed in 12th iteration of our model are:

1. Total liabilities
2. Total income
3. Sales
4. Net worth
5. Capital employed
6. Profit after tax
7. EPS
8. PBT as % of total income
9. Total capital
10. PBDITA
11. Reserves and funds

```
#Removing "Reserves and funds" column

lg_model12 <- glm(Default~. -`Total liabilities` -`Total income` -Sales -`Net worth`
                  -`Capital employed` -`Profit after tax`
                  -EPS -`PBT as % of total income` -`Total capital` -PBDITA
                  -`Reserves and funds`,data = company1, family = binomial())
summary(lg_model12)
vif(lg_model12)
```

After removing and taken a summary of the model, we can see that the following columns are most significant (denoted by "*******"):

1. Cash profit as % of total income
2. Cumulative retained profits
3. TOL/TNW
4. Debt to equity ratio (times)

The columns that are somewhat significant are (denoted by "******"):

1. Cash to average cost of sales per day
2. Total term liabilities / tangible net worth

The columns that are least significant are (denoted by "*****"):

1. Total assets

The columns that are very less significant are (denoted by "**.**"):

1. Cash profit
2. Cash profit as % of total income
3. Net fixed assets
4. Creditors turnover

For the summary of this twelfth model, Refer (*"Annexure 4 – vif values_LR.xlsx"*).

Taking only the significant columns

At first, we are taking the following columns (denoted in "*******", "******", "*****" and one column from the least significant column which are denoted by "**.**") to create another model. And also, the same model is used for evaluation purpose.

```
lg_model13 <- glm(Default~`Cash profit as % of total income`+`Cumulative retained profits`+
                    `TOL/TNW`+`Debt to equity ratio (times)`+`Cash to average cost of sales per day`+
                    `Total term liabilities / tangible net worth`+`Total assets`+
                    `Net fixed assets`,data = company1,
                  family = binomial())
summary(lg_model13)
vif(lg_model13)
```

Also, on looking into the values of the significant values:

```
> summary(lg_model13)

Call:
glm(formula = Default ~ `Cash profit as % of total income` +
    `Cumulative retained profits` + `TOL/TNW` + `Debt to equity ratio (times)` +
    `Cash to average cost of sales per day` + `Total term liabilities / tangible net worth` +
    `Total assets` + `Net fixed assets`, family = binomial(),
    data = company1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7298  -0.3075  -0.2617  -0.1315   3.7720

Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                             -3.102e+00  1.036e-01 -29.949  < 2e-16 ***
`Cash profit as % of total income`      -2.093e-02  2.396e-03  -8.733  < 2e-16 ***
`Cumulative retained profits`           -4.447e-03  6.052e-04  -7.349 2.00e-13 ***
`TOL/TNW`                                7.152e-02  1.485e-02   4.815 1.47e-06 ***
`Debt to equity ratio (times)`           1.259e-01  2.832e-02   4.445 8.80e-06 ***
`Cash to average cost of sales per day`  1.131e-03  3.360e-04   3.366 0.000762 ***
`Total term liabilities / tangible net worth` -1.241e-01  3.742e-02  -3.315 0.000915 ***
`Total assets`                          -4.088e-06  1.730e-05  -0.236 0.813195
`Net fixed assets`                       6.952e-06  8.424e-05   0.083 0.934236
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1771.0  on 3540  degrees of freedom
Residual deviance: 1195.3  on 3532  degrees of freedom
AIC: 1213.3

Number of Fisher Scoring iterations: 10
```

On looking at the above screenshot, we can see that the variables "**Debt to equity ratio (times)**", "**TOL/TNW**", "**Cash to average cost of sales per day**" have a significant impact in the model.

```
> vif(lg_model13)
          `Cash profit as % of total income`                `Cumulative retained profits`
                                   1.049924                                     3.154003
                                  `TOL/TNW`                `Debt to equity ratio (times)`
                                   5.685444                                     7.343447
     `Cash to average cost of sales per day` `Total term liabilities / tangible net worth`
                                   1.005035                                     6.981683
                              `Total assets`                             `Net fixed assets`
                                   1.433968                                     3.085980
```

Also, the **variable inflation factor** (vif) values are also less than 10 for these variables. Also, below is the table that tabulates the above columns based on the categories:

Table 4 – Categories of the variables under lg_model13

Variable name	Category of the variable
Cash profit as % of total income	profit
Cumulative retained profits	profit
TOL/TNW	leverage
Debt to equity ratio (times)	liquidity
Cash to average cost of sales per day	liquidity
Total term liabilities / tangible net worth	leverage
Total assets	size
Net fixed assets	size

EVALUATION OF THE MODEL

We set the threshold value for the predicted as **0.07**, above which the values will get rounded to **1** and the rest of them rounded to **0**. Also, we evaluate the models **lg_model12** and **lg_model13** with the same level of threshold to see which model performs better.

```
#Prediction and Confusion matrix of validation data


tdata <- predict(lg_model13, validation1, type = "response")
t_confmat = table(predicted = ifelse(tdata>0.07,1,0), actual=validation1$Default)
confusionMatrix(t_confmat,positive = "1", mode = "everything")

tdata2 <- predict(lg_model12, validation1, type = "response")
t_confmat2 = table(predicted = ifelse(tdata6>0.07,1,0), actual=validation1$Default)
confusionMatrix(t_confmat2,positive = "1", mode = "everything")
```

On looking at the screenshot, we can see that the predicted values are the ones obtained from running the model on validation dataset and the actual values are the values under "**Default**" variable of the validation dataset.

model used:	**lg_model13**	
Confusion Matrix and Statistics		
	actual	
predicted	0	1
0	582	8
1	79	46
Accuracy :	0.8783	
95% CI :	(0.8521, 0.9014)	
No Information Rate :	0.9245	
P-Value [Acc > NIR] :	1	
Kappa :	0.4567	
Mcnemar's Test P-Value :	6.15E-14	
Sensitivity :	0.85185	
Specificity :	0.88048	
Pos Pred Value :	0.368	
Neg Pred Value :	0.98644	
Precision :	0.368	
Recall :	0.85185	
F1 :	0.51397	
Prevalence :	0.07552	
Detection Rate :	0.06434	
Detection Prevalence :	0.17483	
Balanced Accuracy :	0.86617	
'Positive' Class :	1	

model used:	**lg_model12**	
Confusion Matrix and Statistics		
	actual	
predicted	0	1
0	560	3
1	101	51
Accuracy :	0.8545	
95% CI :	(0.8265, 0.8796)	
No Information Rate :	0.9245	
P-Value [Acc > NIR] :	1	
Kappa :	0.4318	
Mcnemar's Test P-Value :	<2e-16	
Sensitivity :	0.94444	
Specificity :	0.8472	
Pos Pred Value :	0.33553	
Neg Pred Value :	0.99467	
Precision :	0.33553	
Recall :	0.94444	
F1 :	0.49515	
Prevalence :	0.07552	
Detection Rate :	0.07133	
Detection Prevalence :	0.21259	
Balanced Accuracy :	0.89582	
'Positive' Class :	1	

On looking at the Accuracy, we can say that **lg_model13** performs better than **lg_model12**. But, in terms of sensitivity (also known as true positive rate (TPR)), **lg_model12** is better than **lg_model13**.

ANNEXURES

R Code

Finance_and_Risk_Analytics_Assignment.R

Annexure 1 - missing_cols

missing_cols.csv

Annexure 2 - EXPLORATORY DATA ANALYSIS

EXPLORATORY DATA ANALYSIS.docx

Annexure 3 – Correlation_matrix

correlation_matrix.pdf

Annexure 4 – vif values_LR

vif values_LR.xlsx

Other Annexures

report.html

Explore.pdf