***Data Mining and Predictive Modelling Assignment 1***

# Name: Rushikesh Jyoti

# Division: A

# Roll no: 27

# SRN: 201901139

Code

library(moments)

library(corrplot)

dataset <- read.csv('C:\\VS\_Workshop\\Sem 6\\Data Mining and Predictive Modelling\\Assignments\\Ass1\\pva97nk.csv')

# 2. Identify the variables in the file "pva97nk.csv" and

# determine whether any variable has any missing values.

colnames(dataset)

sprintf("There are %d NA values in dataset", sum(is.na(dataset)))

# OR

# table(is.na(dataset))

# 3. Impute some of the variables that have missing values using their corresponding mean values.

# Verify whether your task has been correctly done.

for(i in 1:ncol(dataset)){

if (is.numeric(dataset[,i])){

dataset[is.na(dataset[,i]), i] <- mean(dataset[,i], na.rm = TRUE)

}

}

# Verification

sprintf("There are %d NA values in dataset", sum(is.na(dataset)))

# 4. Compute Skewness and Kurtosis

skurtosis <- data.frame("Category", "Skewness", "Kurtosis")

for(i in 1:ncol(dataset)) {

if(is.numeric(dataset[,i])){

skurtosis[nrow(skurtosis) + 1,] = c(

colnames(dataset)[i],

round(skewness(dataset[,i]), 5),

round(kurtosis(dataset[,i]), 5)

)

}

}

skurtosis

# Histogram of GiftCntAll

hist(dataset$GiftCntAll)

# 5. Determine the "summary" information for the numerical variables.

summary(dataset)

# 6. Identify the "distributions" of the numerical variables

# and plot the distributions.

for(i in 1:ncol(dataset)) {

if (is.numeric(dataset[,i])) {

hist(dataset[,i], main=colnames(dataset)[i])

}

}

# 7. Transform the numeric variables into their natural log values

# and scale [0 - 1] values.

numericset = Filter(is.numeric, dataset)

for (i in 1:ncol(numericset)) {

print(colnames(numericset)[i])

print(head(log(numericset[,i])))

}

# 8. Check whether the numeric variables follow normality conditions.

qqnorm(numericset$GiftCntAll)

qqline(numericset$GiftCntAll)

qqnorm(numericset$PromCntAll)

qqline(numericset$PromCntAll)

qqnorm(numericset$DemAge)

qqline(numericset$DemAge)

# 9. Find the correlation matrix for all the variables in the dataset

# and plot the graph of the correlation matrix.

corrplot(cor(numericset, method = c("spearman")), diag=FALSE)

# 10. From the given dataset partition the data into 70-15-15 divisions

# so to construct the training, validation and test datasets.

spec = c(train = .70, test = .15, validate = .15)

g = sample(cut(

seq(nrow(numericset)),

nrow(numericset) \* cumsum(c(0, spec)),

labels = names(spec)

))

result = split(numericset, g)

sapply(result, nrow) / nrow(numericset)

# To see the dataset

# head(result$train)

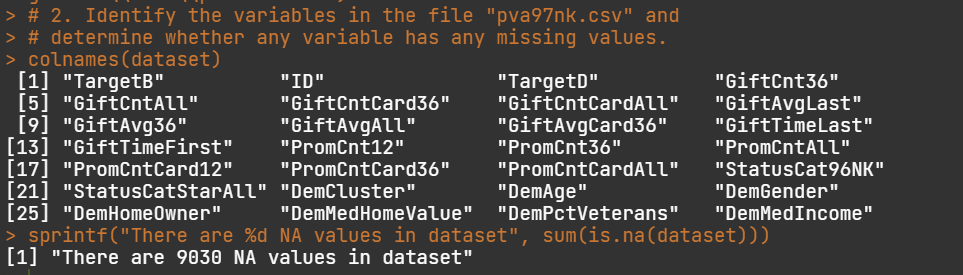
# head(result$test)

# head(result$validate)

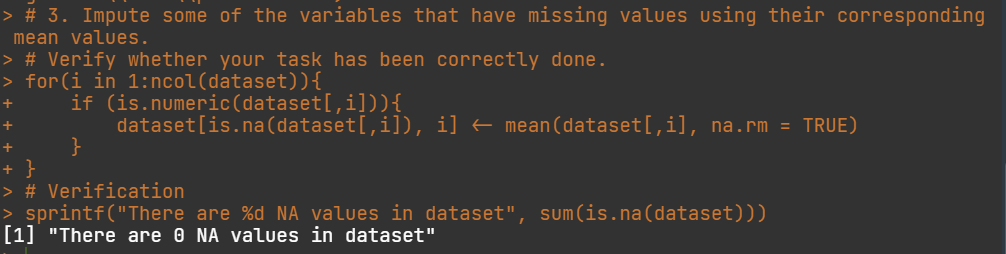
Output

1. Read the file

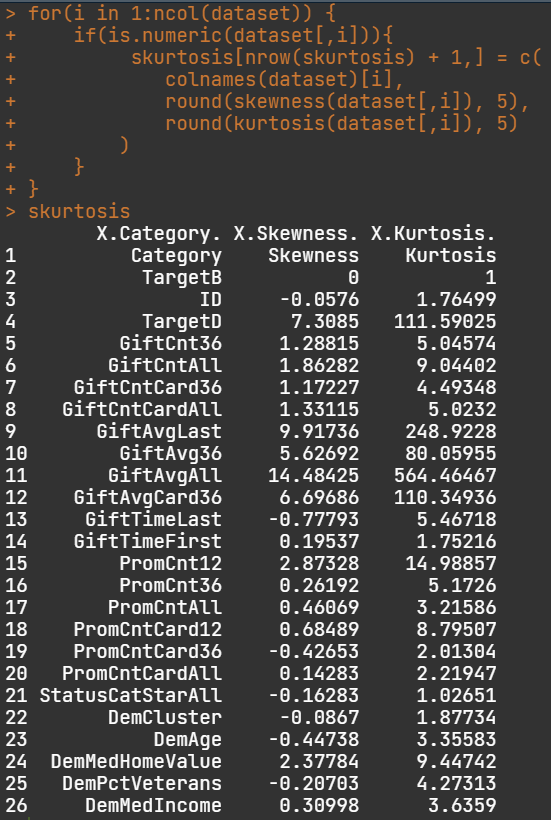
2. The variables and NA values



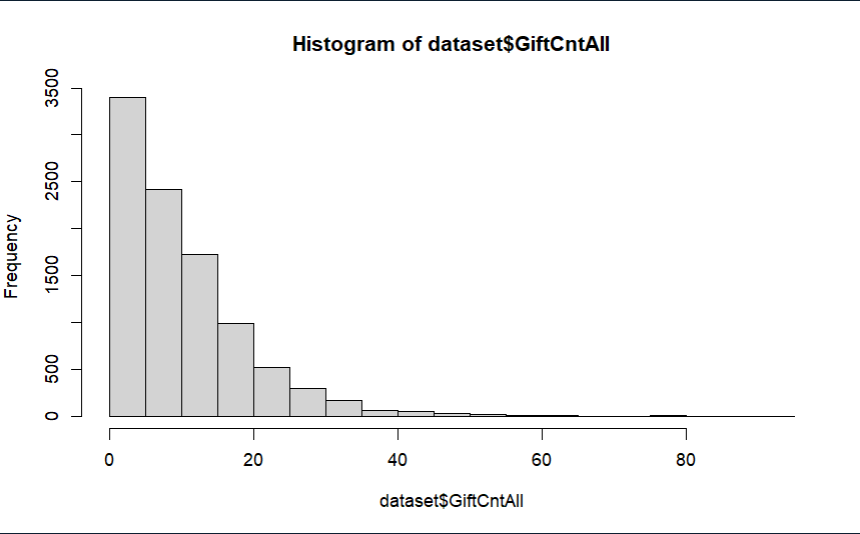
3. Filling NA values with mean and verification of values



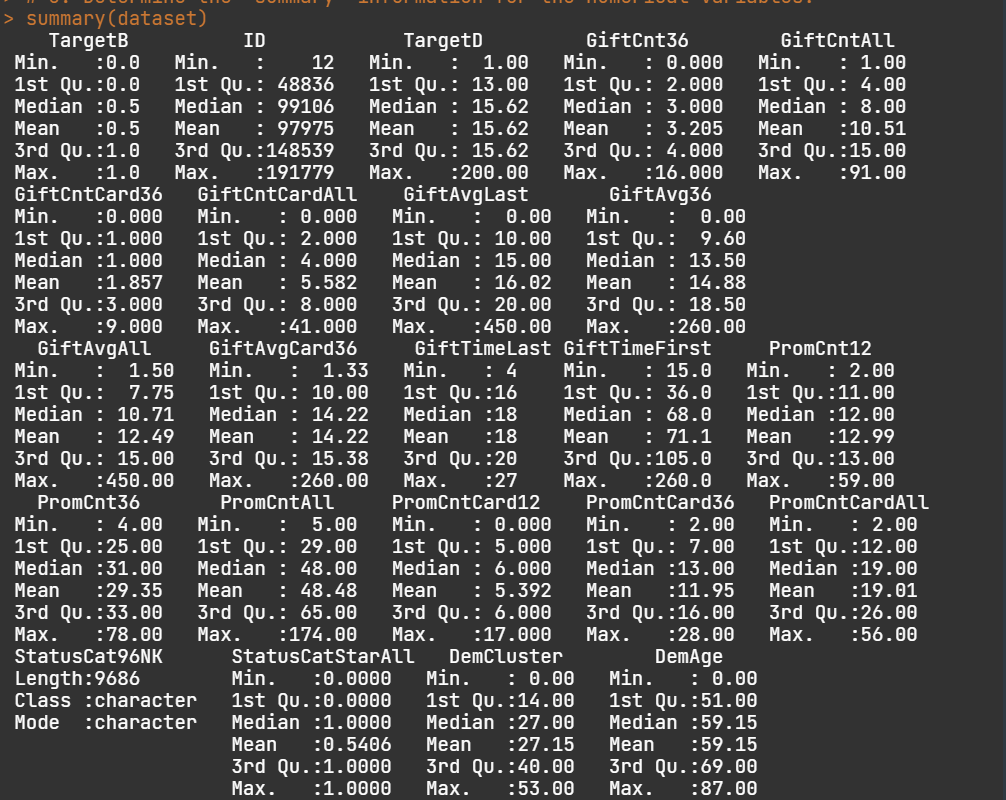
4. Computing the skewness and kurtosis

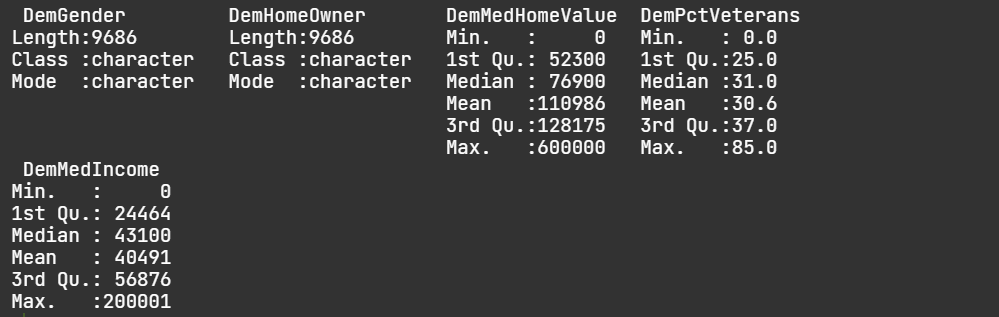


Skewness and Kurtosis of GiftCntAll

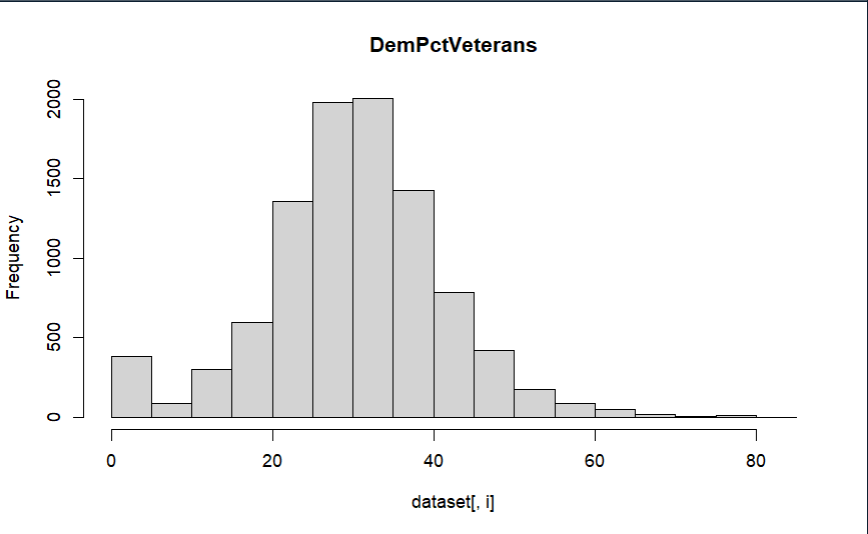


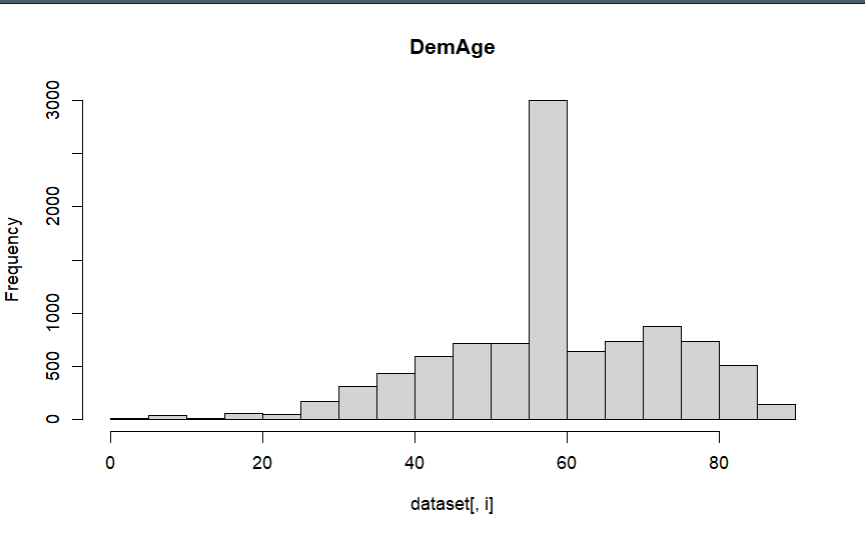
5. Summary of dataset

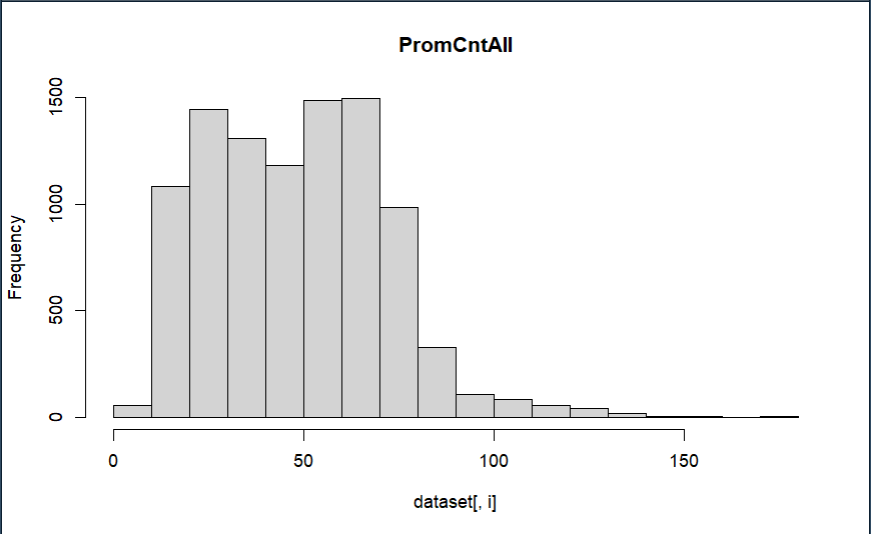




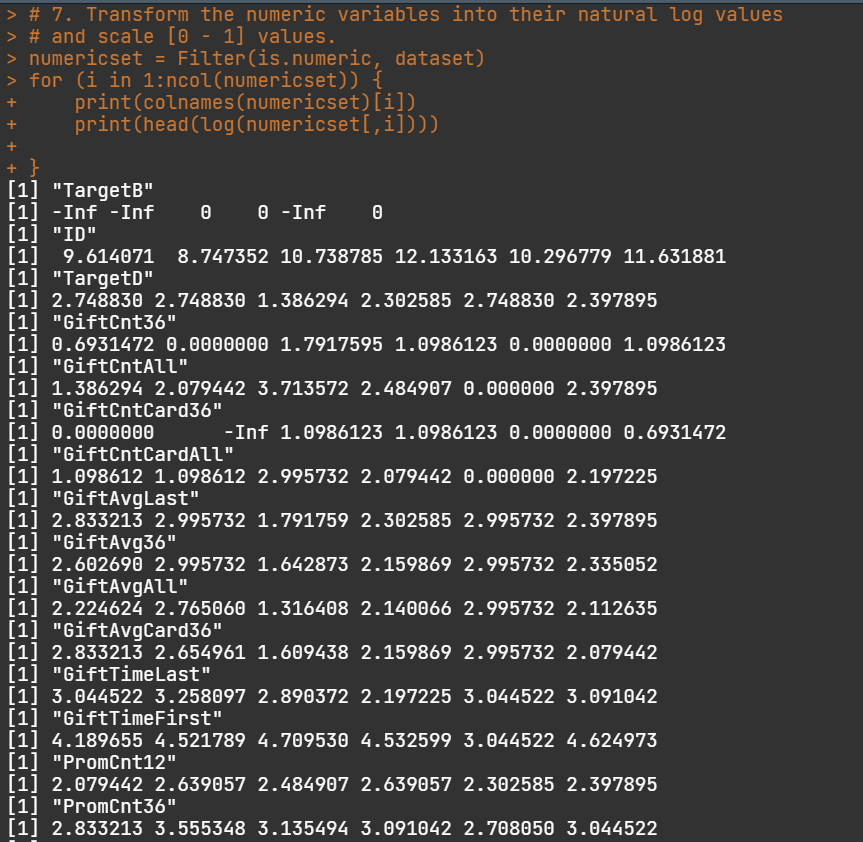
6. Distributions of numeric variables and plotting the distributions

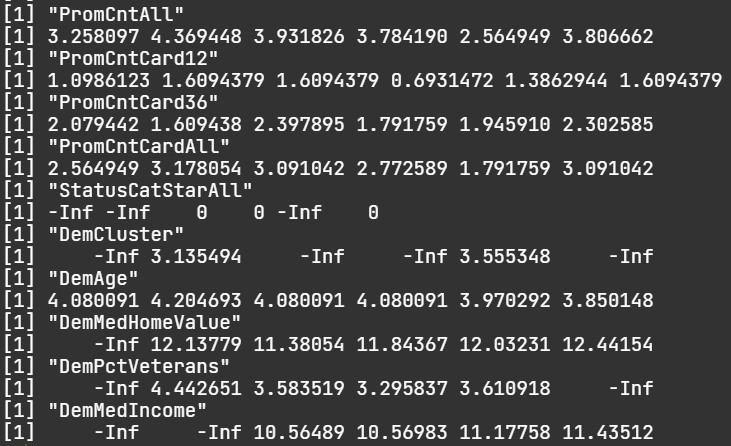






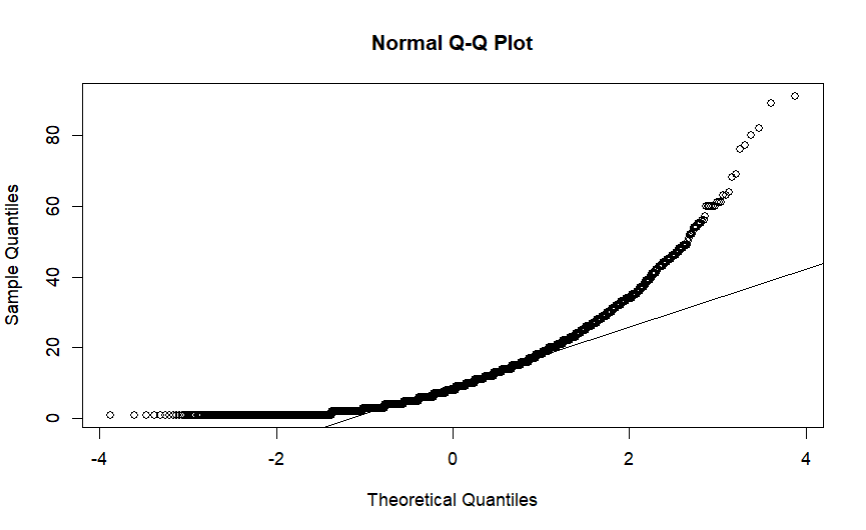
7. Transform numeric variables to their natural log



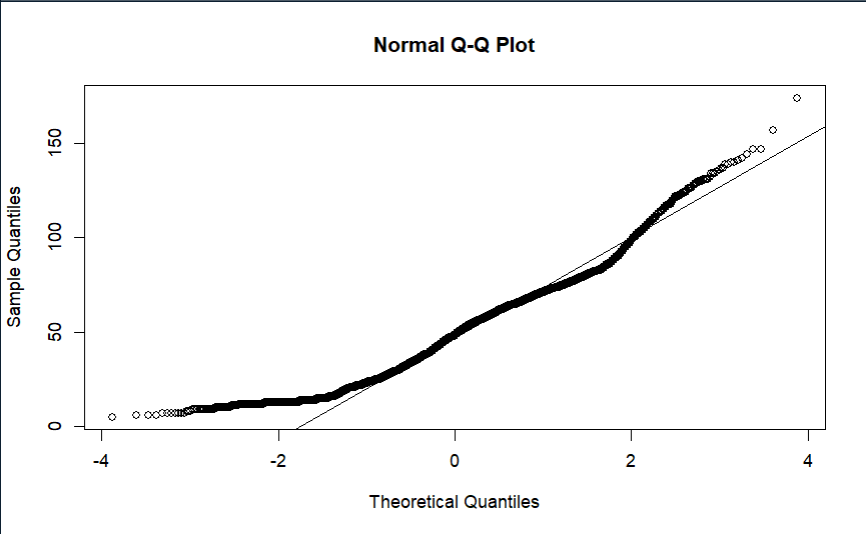


8. Check whether the numeric variables follow normality conditions.

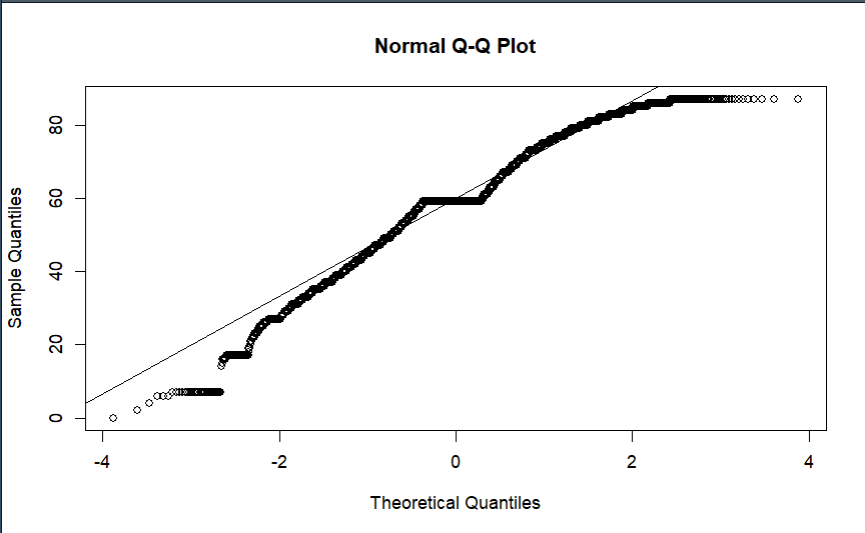
Normality of GiftCntAll



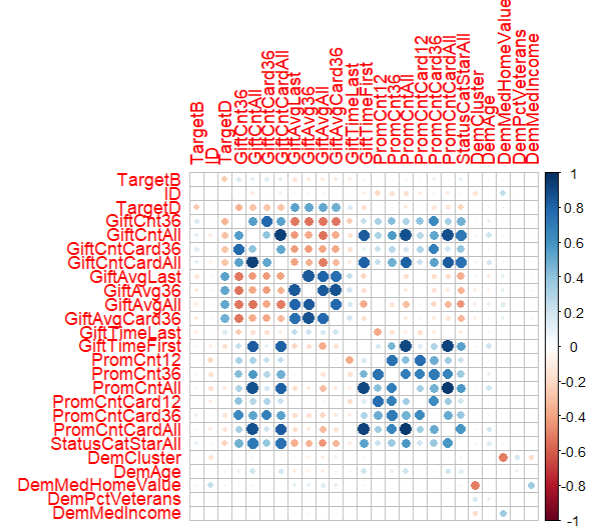
Normality of PromCntAll



Normality of DemAge



9. Find the correlation matrix



10. From the given dataset partition the data into 70-15-15 divisions so to construct the training, validation and test datasets.

