ThottiyamVenkatakrishnan\_DSC520\_Research\_Paper

2023-03-01

# Load the required libraries  
library(class)  
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(bootStepAIC)

## Loading required package: MASS

library(ggplot2)  
library(gridExtra)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:gridExtra':  
##   
## combine

## The following object is masked from 'package:MASS':  
##   
## select

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyr)  
library(caTools)  
library(caret)  
library(e1071)  
  
## Set the working directory to the root of your DSC 520 directory  
setwd("/Users/tvvr/Downloads")  
  
## Load the dataset into the dataframe  
defaulters <- read.csv("UCI\_Credit\_Card.csv")  
  
## Create an alias for the dependent variable  
colnames(defaulters)[colnames(defaulters)=="default.payment.next.month"]<-"dfpy"  
  
# Check for missing values  
sum(is.na(defaulters))

## [1] 0

head(defaulters)

## ID LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5 PAY\_6  
## 1 1 20000 2 2 1 24 2 2 -1 -1 -2 -2  
## 2 2 120000 2 2 2 26 -1 2 0 0 0 2  
## 3 3 90000 2 2 2 34 0 0 0 0 0 0  
## 4 4 50000 2 2 1 37 0 0 0 0 0 0  
## 5 5 50000 1 2 1 57 -1 0 -1 0 0 0  
## 6 6 50000 1 1 2 37 0 0 0 0 0 0  
## BILL\_AMT1 BILL\_AMT2 BILL\_AMT3 BILL\_AMT4 BILL\_AMT5 BILL\_AMT6 PAY\_AMT1 PAY\_AMT2  
## 1 3913 3102 689 0 0 0 0 689  
## 2 2682 1725 2682 3272 3455 3261 0 1000  
## 3 29239 14027 13559 14331 14948 15549 1518 1500  
## 4 46990 48233 49291 28314 28959 29547 2000 2019  
## 5 8617 5670 35835 20940 19146 19131 2000 36681  
## 6 64400 57069 57608 19394 19619 20024 2500 1815  
## PAY\_AMT3 PAY\_AMT4 PAY\_AMT5 PAY\_AMT6 dfpy  
## 1 0 0 0 0 1  
## 2 1000 1000 0 2000 1  
## 3 1000 1000 1000 5000 0  
## 4 1200 1100 1069 1000 0  
## 5 10000 9000 689 679 0  
## 6 657 1000 1000 800 0

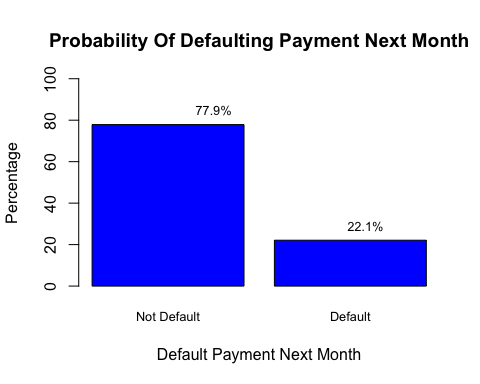
# This step is converting the categorical variables "EDUCATION" and "MARRIAGE"   
# from numeric values to factors with meaningful labels.  
# By converting these variables to factors with meaningful labels, it becomes   
# easier to analyze the data and perform statistical analyses on   
# the categorical variables.  
  
defaulters$SEX <- factor(defaulters$SEX, levels = c(1, 2), labels = c("Male",  
 "Female"))  
defaulters$EDUCATION <- factor(defaulters$EDUCATION, levels = c(0, 1, 2, 3, 4,  
 5, 6),   
 labels = c("Unknown", "Graduate School","University",  
 "High School", "Others","Others","Others"))   
  
defaulters$MARRIAGE <- factor(defaulters$MARRIAGE, levels = c(0, 1, 2, 3),   
 labels = c("Unknown", "Married", "Single","Others"))  
   
defaulters[, c("PAY\_0", "PAY\_2", "PAY\_3", "PAY\_4", "PAY\_5", "PAY\_6")] <-   
 lapply(defaulters[, c("PAY\_0", "PAY\_2", "PAY\_3", "PAY\_4", "PAY\_5", "PAY\_6")],   
 function(x) factor(x, levels = c(-2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8),  
 labels = c("No consumption", "Paid in full", "Revolving credit",  
 "Delay 1 month", "Delay 2 months", "Delay 3 months",  
 "Delay 4 months", "Delay 5 months", "Delay 6 months",  
 "Delay 7 months", "Delay 8 months or more")))  
  
# Convert categorical variables to dummy variables and create new columns  
# This is necessary because many machine learning algorithms require numerical   
# input, and binary dummy variables provide a way to represent categorical   
# data as numerical data  
  
defaulters <- defaulters %>%   
 mutate(SEX\_Male = ifelse(SEX == "Male", 1, 0),  
 EDUCATION\_Graduate\_School = ifelse(EDUCATION == "Graduate School",1,0),  
 EDUCATION\_University = ifelse(EDUCATION == "University", 1, 0),  
 EDUCATION\_High\_School = ifelse(EDUCATION == "High School", 1, 0),  
 EDUCATION\_Others = ifelse(EDUCATION == "Others", 1, 0),  
 MARRIAGE\_Married = ifelse(MARRIAGE == "Married", 1, 0),  
 MARRIAGE\_Single = ifelse(MARRIAGE == "Single", 1, 0))

# Sample output of the dataset after manipulation and cleansing

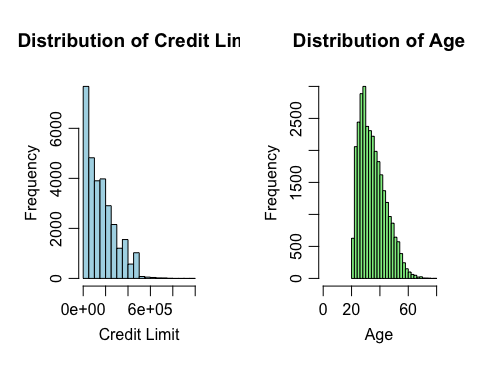
head(defaulters)

## ID LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0  
## 1 1 20000 Female University Married 24 Delay 2 months  
## 2 2 120000 Female University Single 26 Paid in full  
## 3 3 90000 Female University Single 34 Revolving credit  
## 4 4 50000 Female University Married 37 Revolving credit  
## 5 5 50000 Male University Married 57 Paid in full  
## 6 6 50000 Male Graduate School Single 37 Revolving credit  
## PAY\_2 PAY\_3 PAY\_4 PAY\_5  
## 1 Delay 2 months Paid in full Paid in full No consumption  
## 2 Delay 2 months Revolving credit Revolving credit Revolving credit  
## 3 Revolving credit Revolving credit Revolving credit Revolving credit  
## 4 Revolving credit Revolving credit Revolving credit Revolving credit  
## 5 Revolving credit Paid in full Revolving credit Revolving credit  
## 6 Revolving credit Revolving credit Revolving credit Revolving credit  
## PAY\_6 BILL\_AMT1 BILL\_AMT2 BILL\_AMT3 BILL\_AMT4 BILL\_AMT5 BILL\_AMT6  
## 1 No consumption 3913 3102 689 0 0 0  
## 2 Delay 2 months 2682 1725 2682 3272 3455 3261  
## 3 Revolving credit 29239 14027 13559 14331 14948 15549  
## 4 Revolving credit 46990 48233 49291 28314 28959 29547  
## 5 Revolving credit 8617 5670 35835 20940 19146 19131  
## 6 Revolving credit 64400 57069 57608 19394 19619 20024  
## PAY\_AMT1 PAY\_AMT2 PAY\_AMT3 PAY\_AMT4 PAY\_AMT5 PAY\_AMT6 dfpy SEX\_Male  
## 1 0 689 0 0 0 0 1 0  
## 2 0 1000 1000 1000 0 2000 1 0  
## 3 1518 1500 1000 1000 1000 5000 0 0  
## 4 2000 2019 1200 1100 1069 1000 0 0  
## 5 2000 36681 10000 9000 689 679 0 1  
## 6 2500 1815 657 1000 1000 800 0 1  
## EDUCATION\_Graduate\_School EDUCATION\_University EDUCATION\_High\_School  
## 1 0 1 0  
## 2 0 1 0  
## 3 0 1 0  
## 4 0 1 0  
## 5 0 1 0  
## 6 1 0 0  
## EDUCATION\_Others MARRIAGE\_Married MARRIAGE\_Single  
## 1 0 1 0  
## 2 0 0 1  
## 3 0 0 1  
## 4 0 1 0  
## 5 0 1 0  
## 6 0 0 1

# Data Analysis  
# This step creates a bar plot that shows the percentage of defaulting payments   
# next month versus not defaulting payments next month.   
# The purpose of this step is to visualize the distribution of the target   
# variable and to see how imbalanced the data is.   
# It helps to understand the probability of defaulting payments next month   
# and to identify any class imbalance issues affecting the model's performance  
  
  
def\_cnt <- prop.table(table(defaulters$dfpy)) \* 100  
barplot(def\_cnt, main = "Probability Of Defaulting Payment Next Month",  
 xlab = "Default Payment Next Month", ylab = "Percentage", col = "blue",  
 ylim = c(0, 100), names.arg = c("Not Default", "Default"),  
 cex.names = 0.8)  
text(x = 1:2, y = def\_cnt, labels = paste0(round(def\_cnt, 1), "%"),   
 pos = 3, cex = 0.8)

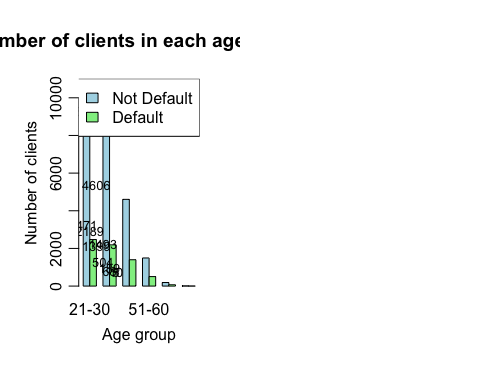


# create a two-panel plot using the par() function.  
# The two panels display the distribution of credit limit (LIMIT\_BAL) and  
# age (AGE) using the hist() function.  
  
par(mfrow = c(1, 2), mar = c(5, 4, 4, 2) + 0.1, mgp = c(2, 0.7, 0))  
hist(defaulters$LIMIT\_BAL, main = "Distribution of Credit Limit",   
 xlab = "Credit Limit", col = "lightblue",   
 xlim = c(0, max(defaulters$LIMIT\_BAL)), breaks = 30)  
  
hist(defaulters$AGE, main = "Distribution of Age", xlab = "Age",   
 col = "lightgreen", xlim = c(0, max(defaulters$AGE)), breaks = 30)



# creating age bins to visualize the number of clients in each age group  
# by default status  
  
  
bins <- c(20, 30, 40, 50, 60, 70, 80)  
names <- c("21-30", "31-40", "41-50", "51-60", "61-70", "71-80")  
defaulters$AGE\_BIN <- cut.default(defaulters$AGE, breaks = bins, labels = names,  
 right = TRUE)  
  
age\_cnt <- table(defaulters$AGE\_BIN)  
age\_0 <- table(defaulters$AGE\_BIN[defaulters$dfpy == 0])  
age\_1 <- table(defaulters$AGE\_BIN[defaulters$dfpy == 1])  
  
barplot(rbind(age\_0, age\_1), main = "Number of clients in each age group",   
 xlab = "Age group", ylab = "Number of clients",   
 col = c("lightblue", "lightgreen"),   
 beside = TRUE, ylim = c(0, max(age\_cnt)), names.arg = names)  
  
text(x = rep(1:6, 2), y = c(age\_0, age\_1), labels = c(age\_0, age\_1),   
 pos = 3, cex = 0.8)  
legend("topright", legend = c("Not Default", "Default"),   
 fill = c("lightblue", "lightgreen"))  
  
par(mfrow=c(2,3),figsize=c(20,10))

## Warning in par(mfrow = c(2, 3), figsize = c(20, 10)): "figsize" is not a  
## graphical parameter



# The following steps are creating barplots of the repayment status for the   
# first six months (PAY\_0 to PAY\_5) separately for defaulters and non-defaulters.  
  
ind <- sort(unique(defaulters$PAY\_0))  
pay\_0 <- prop.table(table(defaulters$PAY\_0[defaulters$dfpy == 0]))  
pay\_1 <- prop.table(table(defaulters$PAY\_0[defaulters$dfpy == 1]))  
total <- pay\_0+pay\_1  
pay\_0\_prop <- (pay\_0/total)\*100  
pay\_1\_prop <- (pay\_1/total)\*100  
barplot(pay\_0\_prop, ylim=c(0, 70), col='green', main='Repayment Status M-0',   
 xlab='Repayment Status', ylab='Percentage', legend.text='0')  
barplot(pay\_1\_prop, ylim=c(0, 70), col='red', add=T, legend.text='1')  
  
  
ind <- sort(unique(defaulters$PAY\_2))  
pay\_0 <- prop.table(table(defaulters$PAY\_2[defaulters$dfpy == 0]))  
pay\_1 <- prop.table(table(defaulters$PAY\_2[defaulters$dfpy == 1]))  
total <- pay\_0+pay\_1  
pay\_0\_prop <- (pay\_0/total)\*100  
pay\_1\_prop <- (pay\_1/total)\*100  
barplot(pay\_0\_prop, ylim=c(0, 70), col='green', main='Repayment Status M-1',   
 xlab='Repayment Status', ylab='Percentage', legend.text='0')  
barplot(pay\_1\_prop, ylim=c(0, 70), col='red', add=T, legend.text='1')  
  
ind <- sort(unique(defaulters$PAY\_3))  
pay\_0 <- prop.table(table(defaulters$PAY\_3[defaulters$dfpy == 0]))  
pay\_1 <- prop.table(table(defaulters$PAY\_3[defaulters$dfpy == 1]))  
total <- pay\_0+pay\_1  
pay\_0\_prop <- (pay\_0/total)\*100  
pay\_1\_prop <- (pay\_1/total)\*100  
barplot(pay\_0\_prop, ylim=c(0, 70), col='green', main='Repayment Status M-2',   
 xlab='Repayment Status', ylab='Percentage', legend.text='0')  
barplot(pay\_1\_prop, ylim=c(0, 70), col='red', add=T, legend.text='1')  
  
ind <- sort(unique(defaulters$PAY\_4))  
pay\_0 <- prop.table(table(defaulters$PAY\_4[defaulters$dfpy == 0]))  
pay\_1 <- prop.table(table(defaulters$PAY\_4[defaulters$dfpy == 1]))  
total <- pay\_0+pay\_1  
pay\_0\_prop <- (pay\_0/total)\*100  
pay\_1\_prop <- (pay\_1/total)\*100  
barplot(pay\_0\_prop, ylim=c(0, 70), col='green', main='Repayment Status M-3',   
 xlab='Repayment Status', ylab='Percentage', legend.text='0')  
barplot(pay\_1\_prop, ylim=c(0, 70), col='red', add=T, legend.text='1')  
  
ind <- sort(unique(defaulters$PAY\_5))  
pay\_0 <- prop.table(table(defaulters$PAY\_5[defaulters$dfpy == 0]))  
pay\_1 <- prop.table(table(defaulters$PAY\_5[defaulters$dfpy == 1]))  
pay\_1

##   
## No consumption Paid in full Revolving credit   
## 0.1348704039 0.1351717902 0.4814647378   
## Delay 1 month Delay 2 months Delay 3 months   
## 0.0000000000 0.2144364075 0.0170283303   
## Delay 4 months Delay 5 months Delay 6 months   
## 0.0076853526 0.0015069319 0.0004520796   
## Delay 7 months Delay 8 months or more   
## 0.0072332731 0.0001506932

total <- pay\_0+pay\_1  
  
g <- ggplot(defaulters, aes(x=AGE)) +  
 geom\_histogram() +  
 facet\_grid(dfpy ~ MARRIAGE)  
  
grid.arrange(g)

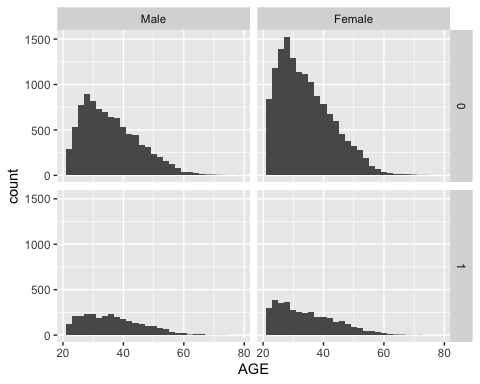
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

ggplot(defaulters, aes(x = AGE)) +  
 geom\_histogram() +  
 facet\_grid(rows = vars(dfpy), cols = vars(SEX))

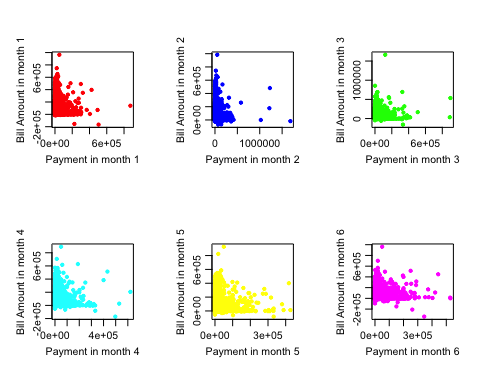
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

# This code creates a grid of six scatter plots using the plot() function  
# this code is useful for comparing the relationship between payment amounts   
# and bill amounts across different months in the defaulters data set.   
# By plotting the data for each month in a separate plot, it allows for easy   
# visual comparison of the patterns over time. The grid layout also makes it   
# easy to compare the plots side-by-side.  
  
  
par(mfrow=c(2,3),figsize=c(20,10))

## Warning in par(mfrow = c(2, 3), figsize = c(20, 10)): "figsize" is not a  
## graphical parameter



plot(defaulters$PAY\_AMT1, defaulters$BILL\_AMT1, col='red', pch=20,   
 xlab="Payment in month 1", ylab="Bill Amount in month 1", cex=1)  
plot(defaulters$PAY\_AMT2, defaulters$BILL\_AMT2, col='blue', pch=20,   
 xlab="Payment in month 2", ylab="Bill Amount in month 2", cex=1)  
plot(defaulters$PAY\_AMT3, defaulters$BILL\_AMT3, col='green', pch=20,   
 xlab="Payment in month 3", ylab="Bill Amount in month 3", cex=1)  
plot(defaulters$PAY\_AMT4, defaulters$BILL\_AMT4, col='cyan', pch=20,   
 xlab="Payment in month 4", ylab="Bill Amount in month 4", cex=1)  
plot(defaulters$PAY\_AMT5, defaulters$BILL\_AMT5, col='yellow', pch=20,   
 xlab="Payment in month 5", ylab="Bill Amount in month 5", cex=1)  
plot(defaulters$PAY\_AMT6, defaulters$BILL\_AMT6, col='magenta', pch=20,   
 xlab="Payment in month 6", ylab="Bill Amount in month 6", cex=1)



# Create a scatter plot of the relationship between age and credit limit balance  
# in the defaulters dataset in R, with points colored by the binary target   
# variable dfpy.  
  
y1 <- defaulters$AGE[defaulters$dfpy == 0]  
y2 <- defaulters$AGE[defaulters$dfpy == 1]  
x1 <- defaulters$LIMIT\_BAL[defaulters$dfpy == 0]  
x2 <- defaulters$LIMIT\_BAL[defaulters$dfpy == 1]  
  
plot(x1, y1, col="red", pch="", xlab="LIMITING BALANCE", ylab="AGE",  
 xlim=c(min(defaulters$LIMIT\_BAL), max(defaulters$LIMIT\_BAL)),  
 ylim=c(min(defaulters$AGE), max(defaulters$AGE)),  
 main="Age vs. Limiting Balance for Defaulters")  
points(x2, y2, col="blue", pch=".", xlab="LIMITING BALANCE", ylab="AGE")  
legend("topright",c("0","1"),col=c("red", "blue"),pch=c("", "."), title="dfpy")  
  
defaulters$AGE\_BIN <- cut.default(defaulters$AGE, breaks = bins, labels = names,  
 right = TRUE)

# Calculate the correlation coefficient between LIMIT\_BAL and BILL\_AMT1  
matrix<-data.frame(rnorm(defaulters$dfpy),  
 rnorm(defaulters$LIMIT\_BAL),rnorm(defaulters$AGE),  
 rnorm(defaulters$PAY\_0),rnorm(defaulters$BILL\_AMT1))  
cor\_matrix<-cor(matrix)  
  
print(cor\_matrix)

## rnorm.defaulters.dfpy. rnorm.defaulters.LIMIT\_BAL.  
## rnorm.defaulters.dfpy. 1.000000000 -0.009330412  
## rnorm.defaulters.LIMIT\_BAL. -0.009330412 1.000000000  
## rnorm.defaulters.AGE. 0.008286181 0.009205837  
## rnorm.defaulters.PAY\_0. -0.006379845 0.014523830  
## rnorm.defaulters.BILL\_AMT1. -0.007496073 0.006674637  
## rnorm.defaulters.AGE. rnorm.defaulters.PAY\_0.  
## rnorm.defaulters.dfpy. 0.008286181 -0.0063798449  
## rnorm.defaulters.LIMIT\_BAL. 0.009205837 0.0145238300  
## rnorm.defaulters.AGE. 1.000000000 0.0065613872  
## rnorm.defaulters.PAY\_0. 0.006561387 1.0000000000  
## rnorm.defaulters.BILL\_AMT1. -0.002628565 -0.0008093272  
## rnorm.defaulters.BILL\_AMT1.  
## rnorm.defaulters.dfpy. -0.0074960727  
## rnorm.defaulters.LIMIT\_BAL. 0.0066746370  
## rnorm.defaulters.AGE. -0.0026285652  
## rnorm.defaulters.PAY\_0. -0.0008093272  
## rnorm.defaulters.BILL\_AMT1. 1.0000000000

# Fit a stepwise regression model using all predictors  
step.model <- stepAIC(lm(dfpy ~ ., data = defaulters),   
 direction="both")

## Start: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + EDUCATION\_University + EDUCATION\_High\_School +   
## EDUCATION\_Others + MARRIAGE\_Married + MARRIAGE\_Single + AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + EDUCATION\_University + EDUCATION\_High\_School +   
## EDUCATION\_Others + MARRIAGE\_Married + AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + EDUCATION\_University + EDUCATION\_High\_School +   
## EDUCATION\_Others + AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + EDUCATION\_University + EDUCATION\_High\_School +   
## AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + EDUCATION\_University + AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## EDUCATION\_Graduate\_School + AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + SEX\_Male +   
## AGE\_BIN  
##   
##   
## Step: AIC=-59761.08  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6 + AGE\_BIN  
##   
## Df Sum of Sq RSS AIC  
## - AGE\_BIN 5 0.593 4069.4 -59767  
## - PAY\_AMT3 1 0.002 4068.9 -59763  
## - BILL\_AMT6 1 0.004 4068.9 -59763  
## - BILL\_AMT4 1 0.007 4068.9 -59763  
## - AGE 1 0.008 4068.9 -59763  
## - BILL\_AMT1 1 0.008 4068.9 -59763  
## - PAY\_AMT4 1 0.009 4068.9 -59763  
## - ID 1 0.020 4068.9 -59763  
## - BILL\_AMT5 1 0.039 4068.9 -59763  
## - BILL\_AMT3 1 0.120 4069.0 -59762  
## - BILL\_AMT2 1 0.224 4069.1 -59761  
## <none> 4068.9 -59761  
## - PAY\_AMT5 1 0.287 4069.1 -59761  
## - PAY\_AMT2 1 0.350 4069.2 -59760  
## - PAY\_AMT6 1 0.375 4069.2 -59760  
## - PAY\_AMT1 1 2.038 4070.9 -59748  
## - SEX 1 2.716 4071.6 -59743  
## - PAY\_5 9 4.948 4073.8 -59743  
## - MARRIAGE 3 3.521 4072.4 -59741  
## - PAY\_2 10 5.845 4074.7 -59738  
## - EDUCATION 4 4.723 4073.6 -59734  
## - PAY\_4 10 7.003 4075.9 -59729  
## - PAY\_3 10 8.154 4077.0 -59721  
## - PAY\_6 9 9.401 4078.3 -59710  
## - LIMIT\_BAL 1 15.663 4084.5 -59648  
## - PAY\_0 10 290.389 4359.2 -57713  
##   
## Step: AIC=-59766.71  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - PAY\_AMT3 1 0.003 4069.4 -59769  
## - BILL\_AMT6 1 0.004 4069.5 -59769  
## - BILL\_AMT4 1 0.008 4069.5 -59769  
## - PAY\_AMT4 1 0.008 4069.5 -59769  
## - BILL\_AMT1 1 0.008 4069.5 -59769  
## - ID 1 0.021 4069.5 -59769  
## - BILL\_AMT5 1 0.041 4069.5 -59768  
## - BILL\_AMT3 1 0.123 4069.6 -59768  
## - BILL\_AMT2 1 0.224 4069.7 -59767  
## <none> 4069.4 -59767  
## - PAY\_AMT5 1 0.287 4069.7 -59767  
## - PAY\_AMT2 1 0.352 4069.8 -59766  
## - PAY\_AMT6 1 0.373 4069.8 -59766  
## - AGE 1 0.524 4070.0 -59765  
## + AGE\_BIN 5 0.593 4068.9 -59761  
## - PAY\_AMT1 1 2.035 4071.5 -59754  
## - SEX 1 2.637 4072.1 -59749  
## - PAY\_5 9 4.940 4074.4 -59748  
## - MARRIAGE 3 3.402 4072.8 -59748  
## - PAY\_2 10 5.846 4075.3 -59744  
## - EDUCATION 4 4.722 4074.2 -59740  
## - PAY\_4 10 6.990 4076.4 -59735  
## - PAY\_3 10 8.151 4077.6 -59727  
## - PAY\_6 9 9.417 4078.9 -59715  
## - LIMIT\_BAL 1 16.592 4086.0 -59647  
## - PAY\_0 10 290.538 4360.0 -57718  
##   
## Step: AIC=-59768.69  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + BILL\_AMT6 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - BILL\_AMT6 1 0.004 4069.5 -59771  
## - BILL\_AMT4 1 0.005 4069.5 -59771  
## - PAY\_AMT4 1 0.007 4069.5 -59771  
## - BILL\_AMT1 1 0.008 4069.5 -59771  
## - ID 1 0.021 4069.5 -59771  
## - BILL\_AMT5 1 0.043 4069.5 -59770  
## - BILL\_AMT3 1 0.134 4069.6 -59770  
## - BILL\_AMT2 1 0.226 4069.7 -59769  
## <none> 4069.4 -59769  
## - PAY\_AMT5 1 0.285 4069.7 -59769  
## - PAY\_AMT2 1 0.369 4069.8 -59768  
## - PAY\_AMT6 1 0.371 4069.8 -59768  
## - AGE 1 0.524 4070.0 -59767  
## + PAY\_AMT3 1 0.003 4069.4 -59767  
## + AGE\_BIN 5 0.593 4068.9 -59763  
## - PAY\_AMT1 1 2.043 4071.5 -59756  
## - SEX 1 2.639 4072.1 -59751  
## - PAY\_5 9 4.954 4074.4 -59750  
## - MARRIAGE 3 3.402 4072.9 -59750  
## - PAY\_2 10 5.848 4075.3 -59746  
## - EDUCATION 4 4.719 4074.2 -59742  
## - PAY\_4 10 7.005 4076.5 -59737  
## - PAY\_3 10 8.161 4077.6 -59729  
## - PAY\_6 9 9.415 4078.9 -59717  
## - LIMIT\_BAL 1 16.591 4086.0 -59649  
## - PAY\_0 10 290.536 4360.0 -57720  
##   
## Step: AIC=-59770.66  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT4 + BILL\_AMT5 + PAY\_AMT1 + PAY\_AMT2 +   
## PAY\_AMT4 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - BILL\_AMT4 1 0.005 4069.5 -59773  
## - PAY\_AMT4 1 0.008 4069.5 -59773  
## - BILL\_AMT1 1 0.008 4069.5 -59773  
## - ID 1 0.021 4069.5 -59773  
## - BILL\_AMT5 1 0.052 4069.5 -59772  
## - BILL\_AMT3 1 0.134 4069.6 -59772  
## - BILL\_AMT2 1 0.230 4069.7 -59771  
## <none> 4069.5 -59771  
## - PAY\_AMT2 1 0.371 4069.8 -59770  
## - PAY\_AMT5 1 0.374 4069.8 -59770  
## - PAY\_AMT6 1 0.404 4069.9 -59770  
## - AGE 1 0.524 4070.0 -59769  
## + BILL\_AMT6 1 0.004 4069.4 -59769  
## + PAY\_AMT3 1 0.002 4069.5 -59769  
## + AGE\_BIN 5 0.593 4068.9 -59765  
## - PAY\_AMT1 1 2.061 4071.5 -59757  
## - SEX 1 2.639 4072.1 -59753  
## - PAY\_5 9 4.952 4074.4 -59752  
## - MARRIAGE 3 3.401 4072.9 -59752  
## - PAY\_2 10 5.848 4075.3 -59748  
## - EDUCATION 4 4.732 4074.2 -59744  
## - PAY\_4 10 7.010 4076.5 -59739  
## - PAY\_3 10 8.159 4077.6 -59731  
## - PAY\_6 9 9.413 4078.9 -59719  
## - LIMIT\_BAL 1 16.587 4086.0 -59651  
## - PAY\_0 10 290.532 4360.0 -57722  
##   
## Step: AIC=-59772.62  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT5 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT4 +   
## PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - PAY\_AMT4 1 0.005 4069.5 -59775  
## - BILL\_AMT1 1 0.009 4069.5 -59775  
## - ID 1 0.023 4069.5 -59774  
## - BILL\_AMT3 1 0.130 4069.6 -59774  
## - BILL\_AMT5 1 0.155 4069.6 -59773  
## - BILL\_AMT2 1 0.229 4069.7 -59773  
## <none> 4069.5 -59773  
## - PAY\_AMT2 1 0.370 4069.8 -59772  
## - PAY\_AMT5 1 0.380 4069.8 -59772  
## - PAY\_AMT6 1 0.410 4069.9 -59772  
## - AGE 1 0.524 4070.0 -59771  
## + BILL\_AMT4 1 0.005 4069.5 -59771  
## + BILL\_AMT6 1 0.004 4069.5 -59771  
## + PAY\_AMT3 1 0.000 4069.5 -59771  
## + AGE\_BIN 5 0.593 4068.9 -59767  
## - PAY\_AMT1 1 2.090 4071.5 -59759  
## - SEX 1 2.637 4072.1 -59755  
## - PAY\_5 9 4.948 4074.4 -59754  
## - MARRIAGE 3 3.403 4072.9 -59754  
## - PAY\_2 10 5.850 4075.3 -59750  
## - EDUCATION 4 4.733 4074.2 -59746  
## - PAY\_4 10 7.013 4076.5 -59741  
## - PAY\_3 10 8.158 4077.6 -59733  
## - PAY\_6 9 9.408 4078.9 -59721  
## - LIMIT\_BAL 1 16.634 4086.1 -59652  
## - PAY\_0 10 290.557 4360.0 -57724  
##   
## Step: AIC=-59774.59  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT1 + BILL\_AMT2 +   
## BILL\_AMT3 + BILL\_AMT5 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 +   
## PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - BILL\_AMT1 1 0.011 4069.5 -59777  
## - ID 1 0.023 4069.5 -59776  
## - BILL\_AMT3 1 0.150 4069.6 -59775  
## - BILL\_AMT5 1 0.210 4069.7 -59775  
## - BILL\_AMT2 1 0.232 4069.7 -59775  
## <none> 4069.5 -59775  
## - PAY\_AMT5 1 0.391 4069.9 -59774  
## - PAY\_AMT2 1 0.397 4069.9 -59774  
## - PAY\_AMT6 1 0.419 4069.9 -59774  
## - AGE 1 0.525 4070.0 -59773  
## + BILL\_AMT6 1 0.005 4069.5 -59773  
## + PAY\_AMT4 1 0.005 4069.5 -59773  
## + BILL\_AMT4 1 0.002 4069.5 -59773  
## + PAY\_AMT3 1 0.000 4069.5 -59773  
## + AGE\_BIN 5 0.593 4068.9 -59769  
## - PAY\_AMT1 1 2.146 4071.6 -59761  
## - SEX 1 2.635 4072.1 -59757  
## - MARRIAGE 3 3.401 4072.9 -59756  
## - PAY\_5 9 5.049 4074.5 -59755  
## - PAY\_2 10 5.846 4075.3 -59752  
## - EDUCATION 4 4.734 4074.2 -59748  
## - PAY\_4 10 7.014 4076.5 -59743  
## - PAY\_3 10 8.156 4077.6 -59735  
## - PAY\_6 9 9.423 4078.9 -59723  
## - LIMIT\_BAL 1 16.669 4086.1 -59654  
## - PAY\_0 10 290.688 4360.2 -57725  
##   
## Step: AIC=-59776.51  
## dfpy ~ ID + LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 + BILL\_AMT3 +   
## BILL\_AMT5 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - ID 1 0.023 4069.5 -59778  
## - BILL\_AMT3 1 0.154 4069.6 -59777  
## - BILL\_AMT5 1 0.215 4069.7 -59777  
## <none> 4069.5 -59777  
## - BILL\_AMT2 1 0.345 4069.8 -59776  
## - PAY\_AMT5 1 0.402 4069.9 -59776  
## - PAY\_AMT2 1 0.415 4069.9 -59775  
## - PAY\_AMT6 1 0.433 4069.9 -59775  
## - AGE 1 0.525 4070.0 -59775  
## + BILL\_AMT1 1 0.011 4069.5 -59775  
## + PAY\_AMT4 1 0.006 4069.5 -59775  
## + BILL\_AMT6 1 0.006 4069.5 -59775  
## + BILL\_AMT4 1 0.002 4069.5 -59775  
## + PAY\_AMT3 1 0.000 4069.5 -59775  
## + AGE\_BIN 5 0.593 4068.9 -59771  
## - PAY\_AMT1 1 2.499 4072.0 -59760  
## - SEX 1 2.631 4072.1 -59759  
## - MARRIAGE 3 3.398 4072.9 -59757  
## - PAY\_5 9 5.052 4074.5 -59757  
## - PAY\_2 10 5.837 4075.3 -59754  
## - EDUCATION 4 4.743 4074.2 -59750  
## - PAY\_4 10 7.021 4076.5 -59745  
## - PAY\_3 10 8.193 4077.7 -59736  
## - PAY\_6 9 9.422 4078.9 -59725  
## - LIMIT\_BAL 1 16.917 4086.4 -59654  
## - PAY\_0 10 290.969 4360.4 -57725  
##   
## Step: AIC=-59778.34  
## dfpy ~ LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 + BILL\_AMT3 +   
## BILL\_AMT5 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - BILL\_AMT3 1 0.150 4069.6 -59779  
## - BILL\_AMT5 1 0.213 4069.7 -59779  
## <none> 4069.5 -59778  
## - BILL\_AMT2 1 0.349 4069.8 -59778  
## - PAY\_AMT5 1 0.402 4069.9 -59777  
## - PAY\_AMT2 1 0.413 4069.9 -59777  
## - PAY\_AMT6 1 0.433 4069.9 -59777  
## + ID 1 0.023 4069.5 -59777  
## - AGE 1 0.524 4070.0 -59776  
## + BILL\_AMT1 1 0.011 4069.5 -59776  
## + PAY\_AMT4 1 0.006 4069.5 -59776  
## + BILL\_AMT6 1 0.006 4069.5 -59776  
## + BILL\_AMT4 1 0.003 4069.5 -59776  
## + PAY\_AMT3 1 0.000 4069.5 -59776  
## + AGE\_BIN 5 0.595 4068.9 -59773  
## - PAY\_AMT1 1 2.504 4072.0 -59762  
## - SEX 1 2.640 4072.1 -59761  
## - MARRIAGE 3 3.394 4072.9 -59759  
## - PAY\_5 9 5.065 4074.6 -59759  
## - PAY\_2 10 5.833 4075.3 -59755  
## - EDUCATION 4 4.770 4074.3 -59751  
## - PAY\_4 10 6.999 4076.5 -59747  
## - PAY\_3 10 8.207 4077.7 -59738  
## - PAY\_6 9 9.472 4079.0 -59727  
## - LIMIT\_BAL 1 16.952 4086.4 -59656  
## - PAY\_0 10 291.178 4360.7 -57725  
##   
## Step: AIC=-59779.23  
## dfpy ~ LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 + BILL\_AMT5 +   
## PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## - BILL\_AMT5 1 0.110 4069.8 -59780  
## - PAY\_AMT2 1 0.263 4069.9 -59779  
## <none> 4069.6 -59779  
## - PAY\_AMT5 1 0.391 4070.0 -59778  
## + BILL\_AMT3 1 0.150 4069.5 -59778  
## - PAY\_AMT6 1 0.438 4070.1 -59778  
## + PAY\_AMT4 1 0.029 4069.6 -59777  
## + ID 1 0.020 4069.6 -59777  
## + PAY\_AMT3 1 0.015 4069.6 -59777  
## - AGE 1 0.528 4070.2 -59777  
## + BILL\_AMT1 1 0.014 4069.6 -59777  
## + BILL\_AMT6 1 0.008 4069.6 -59777  
## + BILL\_AMT4 1 0.007 4069.6 -59777  
## + AGE\_BIN 5 0.596 4069.1 -59774  
## - BILL\_AMT2 1 2.132 4071.8 -59766  
## - SEX 1 2.624 4072.3 -59762  
## - MARRIAGE 3 3.405 4073.1 -59760  
## - PAY\_5 9 5.069 4074.7 -59760  
## - PAY\_AMT1 1 3.015 4072.7 -59759  
## - PAY\_2 10 5.754 4075.4 -59757  
## - EDUCATION 4 4.752 4074.4 -59752  
## - PAY\_4 10 7.037 4076.7 -59747  
## - PAY\_3 10 8.262 4077.9 -59738  
## - PAY\_6 9 9.501 4079.1 -59727  
## - LIMIT\_BAL 1 17.031 4086.7 -59656  
## - PAY\_0 10 291.117 4360.8 -57727  
##   
## Step: AIC=-59780.42  
## dfpy ~ LIMIT\_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY\_0 +   
## PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 + PAY\_AMT1 +   
## PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6  
##   
## Df Sum of Sq RSS AIC  
## <none> 4069.8 -59780  
## - PAY\_AMT2 1 0.340 4070.1 -59780  
## - PAY\_AMT5 1 0.389 4070.1 -59780  
## + BILL\_AMT5 1 0.110 4069.6 -59779  
## - PAY\_AMT6 1 0.441 4070.2 -59779  
## + PAY\_AMT4 1 0.073 4069.7 -59779  
## + BILL\_AMT6 1 0.054 4069.7 -59779  
## + BILL\_AMT3 1 0.048 4069.7 -59779  
## + PAY\_AMT3 1 0.037 4069.7 -59779  
## + BILL\_AMT4 1 0.033 4069.7 -59779  
## + ID 1 0.020 4069.7 -59779  
## + BILL\_AMT1 1 0.017 4069.7 -59779  
## - AGE 1 0.528 4070.3 -59779  
## + AGE\_BIN 5 0.600 4069.2 -59775  
## - SEX 1 2.645 4072.4 -59763  
## - MARRIAGE 3 3.403 4073.2 -59761  
## - PAY\_5 9 5.034 4074.8 -59761  
## - PAY\_AMT1 1 2.924 4072.7 -59761  
## - PAY\_2 10 5.804 4075.6 -59758  
## - BILL\_AMT2 1 3.809 4073.6 -59754  
## - EDUCATION 4 4.724 4074.5 -59754  
## - PAY\_4 10 7.018 4076.8 -59749  
## - PAY\_3 10 8.256 4078.0 -59740  
## - PAY\_6 9 9.594 4079.4 -59728  
## - LIMIT\_BAL 1 17.781 4087.5 -59652  
## - PAY\_0 10 291.074 4360.8 -57728

# Print the selected variables  
summary(step.model)

##   
## Call:  
## lm(formula = dfpy ~ LIMIT\_BAL + SEX + EDUCATION + MARRIAGE +   
## AGE + PAY\_0 + PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 +   
## PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6, data = defaulters)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.98471 -0.16860 -0.12215 -0.03512 1.14383   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.031e-01 1.116e-01 -0.923 0.355759   
## LIMIT\_BAL -2.386e-07 2.087e-08 -11.434 < 2e-16 \*\*\*  
## SEXFemale -1.946e-02 4.413e-03 -4.410 1.04e-05 \*\*\*  
## EDUCATIONGraduate School 1.630e-01 9.869e-02 1.651 0.098678 .   
## EDUCATIONUniversity 1.682e-01 9.870e-02 1.704 0.088358 .   
## EDUCATIONHigh School 1.618e-01 9.879e-02 1.638 0.101463   
## EDUCATIONOthers 6.890e-02 1.001e-01 0.688 0.491474   
## MARRIAGEMarried 1.532e-01 5.047e-02 3.035 0.002410 \*\*   
## MARRIAGESingle 1.335e-01 5.049e-02 2.645 0.008182 \*\*   
## MARRIAGEOthers 1.662e-01 5.441e-02 3.055 0.002252 \*\*   
## AGE 5.374e-04 2.727e-04 1.970 0.048792 \*   
## PAY\_0Paid in full 6.705e-02 1.509e-02 4.443 8.89e-06 \*\*\*  
## PAY\_0Revolving credit -2.762e-02 1.640e-02 -1.684 0.092206 .   
## PAY\_0Delay 1 month 1.371e-01 1.196e-02 11.468 < 2e-16 \*\*\*  
## PAY\_0Delay 2 months 4.218e-01 1.556e-02 27.115 < 2e-16 \*\*\*  
## PAY\_0Delay 3 months 4.242e-01 2.599e-02 16.319 < 2e-16 \*\*\*  
## PAY\_0Delay 4 months 3.563e-01 4.875e-02 7.309 2.77e-13 \*\*\*  
## PAY\_0Delay 5 months 2.644e-01 8.566e-02 3.087 0.002027 \*\*   
## PAY\_0Delay 6 months 5.022e-02 1.526e-01 0.329 0.742116   
## PAY\_0Delay 7 months 3.016e-01 2.488e-01 1.212 0.225471   
## PAY\_0Delay 8 months or more -2.574e-01 3.786e-01 -0.680 0.496643   
## PAY\_2Paid in full -2.182e-02 1.636e-02 -1.333 0.182402   
## PAY\_2Revolving credit 7.845e-03 1.960e-02 0.400 0.689021   
## PAY\_2Delay 1 month -1.248e-01 7.612e-02 -1.639 0.101251   
## PAY\_2Delay 2 months 3.539e-02 1.716e-02 2.063 0.039155 \*   
## PAY\_2Delay 3 months 5.296e-02 2.825e-02 1.874 0.060886 .   
## PAY\_2Delay 4 months -9.927e-02 5.397e-02 -1.839 0.065884 .   
## PAY\_2Delay 5 months 2.859e-01 1.206e-01 2.370 0.017782 \*   
## PAY\_2Delay 6 months 3.180e-01 2.539e-01 1.252 0.210424   
## PAY\_2Delay 7 months 4.016e-01 4.737e-01 0.848 0.396478   
## PAY\_2Delay 8 months or more 2.405e-01 4.841e-01 0.497 0.619370   
## PAY\_3Paid in full -6.539e-03 1.493e-02 -0.438 0.661374   
## PAY\_3Revolving credit 7.509e-03 1.708e-02 0.440 0.660128   
## PAY\_3Delay 1 month -1.520e-01 2.718e-01 -0.559 0.575947   
## PAY\_3Delay 2 months 6.915e-02 1.796e-02 3.850 0.000118 \*\*\*  
## PAY\_3Delay 3 months 8.300e-02 3.452e-02 2.405 0.016190 \*   
## PAY\_3Delay 4 months -3.288e-02 6.871e-02 -0.479 0.632242   
## PAY\_3Delay 5 months -1.745e-01 1.398e-01 -1.248 0.212034   
## PAY\_3Delay 6 months 3.949e-01 3.046e-01 1.296 0.194852   
## PAY\_3Delay 7 months 5.907e-02 1.041e-01 0.568 0.570351   
## PAY\_3Delay 8 months or more -1.908e-01 3.006e-01 -0.635 0.525551   
## PAY\_4Paid in full -1.806e-02 1.508e-02 -1.198 0.231038   
## PAY\_4Revolving credit -1.058e-02 1.674e-02 -0.632 0.527578   
## PAY\_4Delay 1 month 4.920e-01 3.697e-01 1.331 0.183352   
## PAY\_4Delay 2 months 4.241e-02 1.856e-02 2.285 0.022341 \*   
## PAY\_4Delay 3 months 7.699e-03 3.815e-02 0.202 0.840076   
## PAY\_4Delay 4 months 8.032e-02 7.099e-02 1.131 0.257937   
## PAY\_4Delay 5 months -2.640e-01 1.254e-01 -2.105 0.035327 \*   
## PAY\_4Delay 6 months -8.510e-01 3.208e-01 -2.653 0.007994 \*\*   
## PAY\_4Delay 7 months -5.618e-01 3.317e-01 -1.694 0.090338 .   
## PAY\_4Delay 8 months or more -1.382e+00 4.967e-01 -2.783 0.005393 \*\*   
## PAY\_5Paid in full -1.511e-02 1.486e-02 -1.017 0.309217   
## PAY\_5Revolving credit -5.250e-04 1.635e-02 -0.032 0.974387   
## PAY\_5Delay 2 months 5.517e-02 1.907e-02 2.893 0.003824 \*\*   
## PAY\_5Delay 3 months 1.745e-02 3.770e-02 0.463 0.643567   
## PAY\_5Delay 4 months -3.624e-02 7.005e-02 -0.517 0.604885   
## PAY\_5Delay 5 months 1.879e-01 1.290e-01 1.457 0.145133   
## PAY\_5Delay 6 months 7.255e-01 3.500e-01 2.073 0.038224 \*   
## PAY\_5Delay 7 months 7.041e-01 3.621e-01 1.944 0.051847 .   
## PAY\_5Delay 8 months or more 1.183e+00 9.766e-01 1.211 0.225724   
## PAY\_6Paid in full -1.980e-02 1.156e-02 -1.713 0.086713 .   
## PAY\_6Revolving credit -4.629e-02 1.220e-02 -3.793 0.000149 \*\*\*  
## PAY\_6Delay 2 months 1.821e-02 1.483e-02 1.228 0.219588   
## PAY\_6Delay 3 months 1.253e-01 3.617e-02 3.464 0.000532 \*\*\*  
## PAY\_6Delay 4 months 2.514e-02 7.214e-02 0.348 0.727493   
## PAY\_6Delay 5 months -4.525e-02 1.242e-01 -0.364 0.715595   
## PAY\_6Delay 6 months 7.880e-02 1.426e-01 0.553 0.580603   
## PAY\_6Delay 7 months -4.013e-02 1.958e-01 -0.205 0.837585   
## PAY\_6Delay 8 months or more 7.010e-01 5.876e-01 1.193 0.232935   
## BILL\_AMT2 2.123e-07 4.011e-08 5.292 1.22e-07 \*\*\*  
## PAY\_AMT1 -6.707e-07 1.446e-07 -4.637 3.56e-06 \*\*\*  
## PAY\_AMT2 -1.586e-07 1.004e-07 -1.580 0.114097   
## PAY\_AMT5 -2.512e-07 1.484e-07 -1.692 0.090618 .   
## PAY\_AMT6 -2.273e-07 1.262e-07 -1.801 0.071792 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3688 on 29926 degrees of freedom  
## Multiple R-squared: 0.2125, Adjusted R-squared: 0.2106   
## F-statistic: 110.6 on 73 and 29926 DF, p-value: < 2.2e-16

# Fit a ridge regression model using the selected variables  
library(glmnet)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

## Loaded glmnet 4.1-6

x <- model.matrix(dfpy ~ ., data = defaulters)[,-1]  
y <- defaulters$dfpy  
ridge.model <- cv.glmnet(x, y, alpha = 0.5, family = "binomial")  
  
# Plot the cross-validation error  
plot(ridge.model)  
  
reg\_model<-lm(formula = dfpy ~ LIMIT\_BAL + SEX + EDUCATION +   
 MARRIAGE + AGE + PAY\_0 + PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 +   
 PAY\_6 + BILL\_AMT2 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6,   
 data = defaulters)  
  
# View the summary of the model  
summary(reg\_model)

##   
## Call:  
## lm(formula = dfpy ~ LIMIT\_BAL + SEX + EDUCATION + MARRIAGE +   
## AGE + PAY\_0 + PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 + PAY\_6 + BILL\_AMT2 +   
## PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT5 + PAY\_AMT6, data = defaulters)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.98471 -0.16860 -0.12215 -0.03512 1.14383   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.031e-01 1.116e-01 -0.923 0.355759   
## LIMIT\_BAL -2.386e-07 2.087e-08 -11.434 < 2e-16 \*\*\*  
## SEXFemale -1.946e-02 4.413e-03 -4.410 1.04e-05 \*\*\*  
## EDUCATIONGraduate School 1.630e-01 9.869e-02 1.651 0.098678 .   
## EDUCATIONUniversity 1.682e-01 9.870e-02 1.704 0.088358 .   
## EDUCATIONHigh School 1.618e-01 9.879e-02 1.638 0.101463   
## EDUCATIONOthers 6.890e-02 1.001e-01 0.688 0.491474   
## MARRIAGEMarried 1.532e-01 5.047e-02 3.035 0.002410 \*\*   
## MARRIAGESingle 1.335e-01 5.049e-02 2.645 0.008182 \*\*   
## MARRIAGEOthers 1.662e-01 5.441e-02 3.055 0.002252 \*\*   
## AGE 5.374e-04 2.727e-04 1.970 0.048792 \*   
## PAY\_0Paid in full 6.705e-02 1.509e-02 4.443 8.89e-06 \*\*\*  
## PAY\_0Revolving credit -2.762e-02 1.640e-02 -1.684 0.092206 .   
## PAY\_0Delay 1 month 1.371e-01 1.196e-02 11.468 < 2e-16 \*\*\*  
## PAY\_0Delay 2 months 4.218e-01 1.556e-02 27.115 < 2e-16 \*\*\*  
## PAY\_0Delay 3 months 4.242e-01 2.599e-02 16.319 < 2e-16 \*\*\*  
## PAY\_0Delay 4 months 3.563e-01 4.875e-02 7.309 2.77e-13 \*\*\*  
## PAY\_0Delay 5 months 2.644e-01 8.566e-02 3.087 0.002027 \*\*   
## PAY\_0Delay 6 months 5.022e-02 1.526e-01 0.329 0.742116   
## PAY\_0Delay 7 months 3.016e-01 2.488e-01 1.212 0.225471   
## PAY\_0Delay 8 months or more -2.574e-01 3.786e-01 -0.680 0.496643   
## PAY\_2Paid in full -2.182e-02 1.636e-02 -1.333 0.182402   
## PAY\_2Revolving credit 7.845e-03 1.960e-02 0.400 0.689021   
## PAY\_2Delay 1 month -1.248e-01 7.612e-02 -1.639 0.101251   
## PAY\_2Delay 2 months 3.539e-02 1.716e-02 2.063 0.039155 \*   
## PAY\_2Delay 3 months 5.296e-02 2.825e-02 1.874 0.060886 .   
## PAY\_2Delay 4 months -9.927e-02 5.397e-02 -1.839 0.065884 .   
## PAY\_2Delay 5 months 2.859e-01 1.206e-01 2.370 0.017782 \*   
## PAY\_2Delay 6 months 3.180e-01 2.539e-01 1.252 0.210424   
## PAY\_2Delay 7 months 4.016e-01 4.737e-01 0.848 0.396478   
## PAY\_2Delay 8 months or more 2.405e-01 4.841e-01 0.497 0.619370   
## PAY\_3Paid in full -6.539e-03 1.493e-02 -0.438 0.661374   
## PAY\_3Revolving credit 7.509e-03 1.708e-02 0.440 0.660128   
## PAY\_3Delay 1 month -1.520e-01 2.718e-01 -0.559 0.575947   
## PAY\_3Delay 2 months 6.915e-02 1.796e-02 3.850 0.000118 \*\*\*  
## PAY\_3Delay 3 months 8.300e-02 3.452e-02 2.405 0.016190 \*   
## PAY\_3Delay 4 months -3.288e-02 6.871e-02 -0.479 0.632242   
## PAY\_3Delay 5 months -1.745e-01 1.398e-01 -1.248 0.212034   
## PAY\_3Delay 6 months 3.949e-01 3.046e-01 1.296 0.194852   
## PAY\_3Delay 7 months 5.907e-02 1.041e-01 0.568 0.570351   
## PAY\_3Delay 8 months or more -1.908e-01 3.006e-01 -0.635 0.525551   
## PAY\_4Paid in full -1.806e-02 1.508e-02 -1.198 0.231038   
## PAY\_4Revolving credit -1.058e-02 1.674e-02 -0.632 0.527578   
## PAY\_4Delay 1 month 4.920e-01 3.697e-01 1.331 0.183352   
## PAY\_4Delay 2 months 4.241e-02 1.856e-02 2.285 0.022341 \*   
## PAY\_4Delay 3 months 7.699e-03 3.815e-02 0.202 0.840076   
## PAY\_4Delay 4 months 8.032e-02 7.099e-02 1.131 0.257937   
## PAY\_4Delay 5 months -2.640e-01 1.254e-01 -2.105 0.035327 \*   
## PAY\_4Delay 6 months -8.510e-01 3.208e-01 -2.653 0.007994 \*\*   
## PAY\_4Delay 7 months -5.618e-01 3.317e-01 -1.694 0.090338 .   
## PAY\_4Delay 8 months or more -1.382e+00 4.967e-01 -2.783 0.005393 \*\*   
## PAY\_5Paid in full -1.511e-02 1.486e-02 -1.017 0.309217   
## PAY\_5Revolving credit -5.250e-04 1.635e-02 -0.032 0.974387   
## PAY\_5Delay 2 months 5.517e-02 1.907e-02 2.893 0.003824 \*\*   
## PAY\_5Delay 3 months 1.745e-02 3.770e-02 0.463 0.643567   
## PAY\_5Delay 4 months -3.624e-02 7.005e-02 -0.517 0.604885   
## PAY\_5Delay 5 months 1.879e-01 1.290e-01 1.457 0.145133   
## PAY\_5Delay 6 months 7.255e-01 3.500e-01 2.073 0.038224 \*   
## PAY\_5Delay 7 months 7.041e-01 3.621e-01 1.944 0.051847 .   
## PAY\_5Delay 8 months or more 1.183e+00 9.766e-01 1.211 0.225724   
## PAY\_6Paid in full -1.980e-02 1.156e-02 -1.713 0.086713 .   
## PAY\_6Revolving credit -4.629e-02 1.220e-02 -3.793 0.000149 \*\*\*  
## PAY\_6Delay 2 months 1.821e-02 1.483e-02 1.228 0.219588   
## PAY\_6Delay 3 months 1.253e-01 3.617e-02 3.464 0.000532 \*\*\*  
## PAY\_6Delay 4 months 2.514e-02 7.214e-02 0.348 0.727493   
## PAY\_6Delay 5 months -4.525e-02 1.242e-01 -0.364 0.715595   
## PAY\_6Delay 6 months 7.880e-02 1.426e-01 0.553 0.580603   
## PAY\_6Delay 7 months -4.013e-02 1.958e-01 -0.205 0.837585   
## PAY\_6Delay 8 months or more 7.010e-01 5.876e-01 1.193 0.232935   
## BILL\_AMT2 2.123e-07 4.011e-08 5.292 1.22e-07 \*\*\*  
## PAY\_AMT1 -6.707e-07 1.446e-07 -4.637 3.56e-06 \*\*\*  
## PAY\_AMT2 -1.586e-07 1.004e-07 -1.580 0.114097   
## PAY\_AMT5 -2.512e-07 1.484e-07 -1.692 0.090618 .   
## PAY\_AMT6 -2.273e-07 1.262e-07 -1.801 0.071792 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3688 on 29926 degrees of freedom  
## Multiple R-squared: 0.2125, Adjusted R-squared: 0.2106   
## F-statistic: 110.6 on 73 and 29926 DF, p-value: < 2.2e-16

# Define the training control  
train\_control <- trainControl(method = "cv", number = 10)  
  
# Train the linear regression model using 10-fold cross-validation  
lm\_model <- train(dfpy ~ SEX + MARRIAGE + AGE + BILL\_AMT1 +   
 EDUCATION + PAY\_0, data = defaulters,  
 method = "lm",  
 trControl = train\_control)

## Warning in train.default(x, y, weights = w, ...): You are trying to do  
## regression and your outcome only has two possible values Are you trying to do  
## classification? If so, use a 2 level factor as your outcome column.

# Print the cross-validation results  
lm\_model

## Linear Regression   
##   
## 30000 samples  
## 6 predictor  
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 27000, 27000, 27000, 27000, 27000, 27000, ...   
## Resampling results:  
##   
## RMSE Rsquared MAE   
## 0.3755361 0.1815043 0.2819712  
##   
## Tuning parameter 'intercept' was held constant at a value of TRUE

# set seed for reproducibility  
set.seed(123)  
  
# split data into training and testing sets (70/30 split)  
trainIndex <- createDataPartition(defaulters$dfpy, p = 0.7, list = FALSE)  
training <- defaulters[trainIndex,]  
testing <- defaulters[-trainIndex,]  
  
# check number of observations in training and testing sets  
nrow(training) # should be approximately 21000

## [1] 21000

nrow(testing) # should be approximately 9000

## [1] 9000

# fit logistic regression model on training datxa  
logit\_model <- glm(dfpy ~ SEX + MARRIAGE + AGE + BILL\_AMT1 +   
 EDUCATION + PAY\_0, data = training,   
 family = "binomial")  
summary(logit\_model)

##   
## Call:  
## glm(formula = dfpy ~ SEX + MARRIAGE + AGE + BILL\_AMT1 + EDUCATION +   
## PAY\_0, family = "binomial", data = training)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.9613 -0.5897 -0.5327 -0.4676 2.6571   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.451e+01 8.740e+01 -0.166 0.86812   
## SEXFemale -1.974e-01 3.763e-02 -5.245 1.56e-07 \*\*\*  
## MARRIAGEMarried 1.686e+00 7.662e-01 2.200 0.02778 \*   
## MARRIAGESingle 1.584e+00 7.662e-01 2.068 0.03869 \*   
## MARRIAGEOthers 1.957e+00 7.829e-01 2.500 0.01242 \*   
## AGE 4.120e-04 2.318e-03 0.178 0.85894   
## BILL\_AMT1 -4.770e-07 2.897e-07 -1.647 0.09960 .   
## EDUCATIONGraduate School 1.103e+01 8.740e+01 0.126 0.89953   
## EDUCATIONUniversity 1.122e+01 8.740e+01 0.128 0.89785   
## EDUCATIONHigh School 1.118e+01 8.740e+01 0.128 0.89825   
## EDUCATIONOthers 9.834e+00 8.740e+01 0.113 0.91041   
## PAY\_0Paid in full 2.450e-01 8.053e-02 3.043 0.00234 \*\*   
## PAY\_0Revolving credit -2.321e-02 7.758e-02 -0.299 0.76484   
## PAY\_0Delay 1 month 1.194e+00 8.033e-02 14.859 < 2e-16 \*\*\*  
## PAY\_0Delay 2 months 2.678e+00 8.698e-02 30.789 < 2e-16 \*\*\*  
## PAY\_0Delay 3 months 3.088e+00 1.754e-01 17.608 < 2e-16 \*\*\*  
## PAY\_0Delay 4 months 2.668e+00 3.089e-01 8.640 < 2e-16 \*\*\*  
## PAY\_0Delay 5 months 1.990e+00 4.727e-01 4.211 2.55e-05 \*\*\*  
## PAY\_0Delay 6 months 2.735e+00 8.410e-01 3.252 0.00115 \*\*   
## PAY\_0Delay 7 months 3.436e+00 1.100e+00 3.122 0.00180 \*\*   
## PAY\_0Delay 8 months or more 1.828e+00 5.839e-01 3.131 0.00174 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 22279 on 20999 degrees of freedom  
## Residual deviance: 18917 on 20979 degrees of freedom  
## AIC: 18959  
##   
## Number of Fisher Scoring iterations: 11

reg\_model<-lm(formula = dfpy ~ SEX + MARRIAGE + AGE + BILL\_AMT1 +   
 EDUCATION + PAY\_0, data = defaulters)  
summary(reg\_model)

##   
## Call:  
## lm(formula = dfpy ~ SEX + MARRIAGE + AGE + BILL\_AMT1 + EDUCATION +   
## PAY\_0, data = defaulters)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.80660 -0.16323 -0.13373 -0.09352 1.00654   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.657e-01 1.135e-01 -1.461 0.144101   
## SEXFemale -2.550e-02 4.480e-03 -5.692 1.27e-08 \*\*\*  
## MARRIAGEMarried 1.400e-01 5.136e-02 2.726 0.006417 \*\*   
## MARRIAGESingle 1.238e-01 5.138e-02 2.409 0.016015 \*   
## MARRIAGEOthers 1.593e-01 5.534e-02 2.878 0.003999 \*\*   
## AGE -2.704e-05 2.753e-04 -0.098 0.921748   
## BILL\_AMT1 -6.347e-08 3.267e-08 -1.943 0.052034 .   
## EDUCATIONGraduate School 1.759e-01 1.004e-01 1.752 0.079811 .   
## EDUCATIONUniversity 2.007e-01 1.004e-01 1.999 0.045636 \*   
## EDUCATIONHigh School 2.019e-01 1.005e-01 2.009 0.044597 \*   
## EDUCATIONOthers 8.044e-02 1.019e-01 0.789 0.429936   
## PAY\_0Paid in full 3.212e-02 8.723e-03 3.682 0.000232 \*\*\*  
## PAY\_0Revolving credit -7.560e-03 8.247e-03 -0.917 0.359316   
## PAY\_0Delay 1 month 2.009e-01 9.523e-03 21.100 < 2e-16 \*\*\*  
## PAY\_0Delay 2 months 5.518e-01 1.048e-02 52.639 < 2e-16 \*\*\*  
## PAY\_0Delay 3 months 6.134e-01 2.221e-02 27.623 < 2e-16 \*\*\*  
## PAY\_0Delay 4 months 5.383e-01 4.375e-02 12.304 < 2e-16 \*\*\*  
## PAY\_0Delay 5 months 3.613e-01 7.409e-02 4.876 1.09e-06 \*\*\*  
## PAY\_0Delay 6 months 3.987e-01 1.135e-01 3.513 0.000444 \*\*\*  
## PAY\_0Delay 7 months 6.330e-01 1.254e-01 5.046 4.55e-07 \*\*\*  
## PAY\_0Delay 8 months or more 4.378e-01 8.651e-02 5.061 4.19e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3754 on 29979 degrees of freedom  
## Multiple R-squared: 0.1825, Adjusted R-squared: 0.182   
## F-statistic: 334.7 on 20 and 29979 DF, p-value: < 2.2e-16

# Split data into training and testing sets  
set.seed(10)  
split <- sample.split(defaulters$dfpy, SplitRatio = 0.8)  
train <- subset(defaulters, split == TRUE)  
test <- subset(defaulters, split == FALSE)  
  
# Remove 'dfpy' and 'AGE\_BIN' from training and testing sets  
X\_train <- train[, !names(train) %in% c("dfpy", "defaulters$AGE\_BIN")]  
X\_test <- test[, !names(test) %in% c("dfpy", "defaulters$AGE\_BIN")]  
y\_train <- train$dfpy  
y\_test <- test$dfpy  
  
# Fit logistic regression model  
model1 <- glm(dfpy ~ ., data = train, family = binomial(link = "logit"))  
  
# Make predictions on test set  
y\_pred <- predict(model1, newdata = X\_test, type = "response")

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
## prediction from a rank-deficient fit may be misleading

y\_pred <- ifelse(y\_pred > 0.5, 1, 0)  
  
# Evaluate model performance  
print(confusionMatrix(table(y\_pred, y\_test)))

## Confusion Matrix and Statistics  
##   
## y\_test  
## y\_pred 0 1  
## 0 4449 875  
## 1 224 452  
##   
## Accuracy : 0.8168   
## 95% CI : (0.8068, 0.8265)  
## No Information Rate : 0.7788   
## P-Value [Acc > NIR] : 2.381e-13   
##   
## Kappa : 0.355   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9521   
## Specificity : 0.3406   
## Pos Pred Value : 0.8356   
## Neg Pred Value : 0.6686   
## Prevalence : 0.7788   
## Detection Rate : 0.7415   
## Detection Prevalence : 0.8873   
## Balanced Accuracy : 0.6463   
##   
## 'Positive' Class : 0   
##

print(paste0("Accuracy Score for model1: ", mean(y\_pred == y\_test)))

## [1] "Accuracy Score for model1: 0.816833333333333"

# Change the datatype of categorical features from integer to factor  
defaulters$SEX <- as.factor(defaulters$SEX)  
defaulters$EDUCATION <- as.factor(defaulters$EDUCATION)  
defaulters$MARRIAGE <- as.factor(defaulters$MARRIAGE)  
defaulters$PAY\_0 <- as.factor(defaulters$PAY\_0)  
defaulters$PAY\_2 <- as.factor(defaulters$PAY\_2)  
defaulters$PAY\_3 <- as.factor(defaulters$PAY\_3)  
defaulters$PAY\_4 <- as.factor(defaulters$PAY\_4)  
defaulters$PAY\_5 <- as.factor(defaulters$PAY\_5)  
defaulters$PAY\_6 <- as.factor(defaulters$PAY\_6)  
defaulters$def\_type <- as.factor(defaulters$dfpy)  
  
  
# Split data into training and testing sets  
set.seed(10)  
split <- sample.split(defaulters$dfpy, SplitRatio = 0.8)  
train <- subset(defaulters, split == TRUE)  
test <- subset(defaulters, split == FALSE)  
  
# Remove 'def\_pay' and 'AGE\_BIN' from training and testing sets  
X\_train <- train[, !names(train) %in% c("dfpy", "defaulters$AGE\_BIN")]  
X\_test <- test[, !names(test) %in% c("dfpy", "defaulters$AGE\_BIN")]  
y\_train <- train$dfpy  
y\_test <- test$dfpy  
  
# Fit logistic regression model  
model2 <- glm(dfpy ~ ., data = train, family = binomial(link = "logit"))

## Warning: glm.fit: algorithm did not converge

# Make predictions on test set  
y\_pred <- predict(model2, newdata = X\_test, type = "response")

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
## prediction from a rank-deficient fit may be misleading

y\_pred <- ifelse(y\_pred > 0.5, 1, 0)  
  
# Evaluate model performance  
print(confusionMatrix(table(y\_pred, y\_test)))

## Confusion Matrix and Statistics  
##   
## y\_test  
## y\_pred 0 1  
## 0 4673 0  
## 1 0 1327  
##   
## Accuracy : 1   
## 95% CI : (0.9994, 1)  
## No Information Rate : 0.7788   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 1   
##   
## Mcnemar's Test P-Value : NA   
##   
## Sensitivity : 1.0000   
## Specificity : 1.0000   
## Pos Pred Value : 1.0000   
## Neg Pred Value : 1.0000   
## Prevalence : 0.7788   
## Detection Rate : 0.7788   
## Detection Prevalence : 0.7788   
## Balanced Accuracy : 1.0000   
##   
## 'Positive' Class : 0   
##

print(paste0("Accuracy Score for model2: ", mean(y\_pred == y\_test)))

## [1] "Accuracy Score for model2: 1"

df\_X <- defaulters[c('SEX', 'MARRIAGE','AGE','BILL\_AMT1','EDUCATION','PAY\_0')]  
df\_y <- defaulters$dfpy  
  
set.seed(20) # setting the random seed for reproducibility  
split <- sample.split(df\_y, SplitRatio = 0.1)  
train\_X <- df\_X[split, ]  
train\_y <- df\_y[split]  
test\_X <- df\_X[!split, ]  
test\_y <- df\_y[!split]  
  
model4 <- glm(dfpy ~ SEX + AGE, data = defaulters[c("dfpy", "SEX", "AGE")],   
 family = "binomial")  
  
y\_pred <- ifelse(predict(model4, test\_X, type = "response") > 0.5, 1, 0)  
y\_train\_pred <- ifelse(predict(model4, train\_X, type = "response") > 0.5, 1, 0)  
  
cat("Test Accuracy Score for model4: ", mean(y\_pred == test\_y), "\n")

## Test Accuracy Score for model4: 0.7788148

cat("Train Accuracy Score for model4: ", mean(y\_train\_pred == train\_y), "\n")

## Train Accuracy Score for model4: 0.7786667

defaulters <- read.csv("UCI\_Credit\_Card.csv")  
  
df\_X <- defaulters[c('SEX', 'MARRIAGE', 'AGE','BILL\_AMT1','EDUCATION','PAY\_0')]  
#df\_y <- defaulters$dfpy  
df\_y <- as.factor(defaulters$default.payment.next.month)  
  
set.seed(10)  
trainIndex <- createDataPartition(df\_y, p = .7, list = FALSE, times = 1)  
  
train\_X <- df\_X[trainIndex, ]  
test\_X <- df\_X[-trainIndex, ]  
train\_y <- df\_y[trainIndex]  
test\_y <- df\_y[-trainIndex]  
  
#warnings()  
model5 <- train(train\_X, train\_y, method = "glmnet",   
 trControl = trainControl(method = "cv", number = 10))  
y\_pred <- predict(model5, newdata = test\_X)  
y\_train\_pred <- predict(model5, newdata = train\_X)  
  
print(confusionMatrix(y\_pred, test\_y))

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 6809 1505  
## 1 200 485  
##   
## Accuracy : 0.8105   
## 95% CI : (0.8023, 0.8186)  
## No Information Rate : 0.7789   
## P-Value [Acc > NIR] : 9.421e-14   
##   
## Kappa : 0.2812   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9715   
## Specificity : 0.2437   
## Pos Pred Value : 0.8190   
## Neg Pred Value : 0.7080   
## Prevalence : 0.7789   
## Detection Rate : 0.7566   
## Detection Prevalence : 0.9239   
## Balanced Accuracy : 0.6076   
##   
## 'Positive' Class : 0   
##

cat("Test Accuracy Score for model5: ", mean(y\_pred == test\_y), "\n")

## Test Accuracy Score for model5: 0.8105345

cat("Train Accuracy Score for model5: ", mean(y\_train\_pred == train\_y), "\n")

## Train Accuracy Score for model5: 0.8102947

