credit\_default\_logistic

authour

2023-11-30

Introduction

## Data description

Link to the dataset <https://archive.ics.uci.edu/dataset/350/default+of+credit+card+clients> We will be utilizing the Credit Card Clients Data Set Default. There are 24 columns and 30,000 rows in this data collection. With the information supplied, the data set might be used to calculate the likelihood that a credit card client would default on their payment. These characteristics are connected to a customer’s prior payment history, bill statements, and other facts. Significance This study is significant because it attempts to estimate the probability that credit card customers would miss payments based on past payment histories, bill statements, and other relevant information. The research holds relevance as it has the ability to aid in risk assessment and management within the credit card industry. Credit card firms can reduce financial risk and save possible losses by making well-informed judgments about lending and credit limits based on accurate default prediction. Furthermore, by giving customers a greater understanding of their own creditworthiness and assisting them in making wiser financial decisions, this research can also benefit consumers. In the end, this study has applications for credit card issuers and consumers alike, contributing to the advancement of credit card services and finance Objective: • Understand the Dataset  
• Build classification models to predict default rate

## The research question

Build a classification model, logistic regression to predict whether a particular customer will default payment next month or not. Target variable is default payment (Yes = 1, No = 0), as the response variable. The factors or parameters from the dataset to be utilized are  
Amount of the given credit, it includes both the individual consumer credit and his/her family (supplementary) Gender (1 = male; 2 = female). Education (1 = graduate school; 2 = university; 3 = high school; 4 = others). Marital status (1 = married; 2 = single; 3 = others). Age (year). History of past payment the repayment status

## Data-driven, computational approach may be useful

Because a data-driven, computational method makes it possible to analyze a lot of data and find patterns and interactions between variables, it might be helpful in addressing the research topic. In this situation, a data-driven, computing approach is helpful because it enables us to examine a lot of data and find insights, linkages, and patterns that might not be immediately obvious via traditional research. We can forecast whether or not a consumer in this particular circumstance would default on their payment the next month by utilizing a categorization model.

Using the provided parameters—credit amount, gender, age, marital status, education, and previous payment history—along with the payment history, we can create a model that learns from prior data and recognizes patterns that point to defaulting behavior. Based on the trends found, this enables us to forecast situations in the future. Employing a computational, data-driven approach has several benefits. First of all, it enables a bias-free, impartial analysis of the data. Second, it has the speed and efficiency to process vast amounts of data, allowing us to swiftly make predictions. Thirdly, it has the ability to automatically recognize intricate connections between several factors that human analysts might not be able to see. Finally, it enables the model to be improved and refined iteratively using fresh data, which promotes ongoing learning and increasingly accurate predictions over time.

All things considered, a computational, data-driven strategy is an effective means of forecasting credit card payment default as it makes use of the available data to build a precise and trustworthy model that may help reduce risks and make well-informed judgments.

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(readxl)  
library(dplyr)  
  
df <- read\_excel("C:/Users/Desktop/default of credit card clients.xls")  
  
head(df)

## # A tibble: 6 × 25  
## ID LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 20000 2 2 1 24 2 2 -1 -1 -2  
## 2 2 120000 2 2 2 26 -1 2 0 0 0  
## 3 3 90000 2 2 2 34 0 0 0 0 0  
## 4 4 50000 2 2 1 37 0 0 0 0 0  
## 5 5 50000 1 2 1 57 -1 0 -1 0 0  
## 6 6 50000 1 1 2 37 0 0 0 0 0  
## # ℹ 14 more variables: PAY\_6 <dbl>, BILL\_AMT1 <dbl>, BILL\_AMT2 <dbl>,  
## # BILL\_AMT3 <dbl>, BILL\_AMT4 <dbl>, BILL\_AMT5 <dbl>, BILL\_AMT6 <dbl>,  
## # PAY\_AMT1 <dbl>, PAY\_AMT2 <dbl>, PAY\_AMT3 <dbl>, PAY\_AMT4 <dbl>,  
## # PAY\_AMT5 <dbl>, PAY\_AMT6 <dbl>, `default payment next month` <dbl>

sum(is.na(df))

## [1] 0

#renaming column  
# Assuming you have a data frame named "df"  
#colnames(df)[colnames(df) == "old\_column\_name"] <- "new\_column\_name"  
colnames(df)[colnames(df)=="default payment next month"] <- "default\_payment"  
   
df\_clean <- na.omit(df)  
  
head(df\_clean)

## # A tibble: 6 × 25  
## ID LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 20000 2 2 1 24 2 2 -1 -1 -2  
## 2 2 120000 2 2 2 26 -1 2 0 0 0  
## 3 3 90000 2 2 2 34 0 0 0 0 0  
## 4 4 50000 2 2 1 37 0 0 0 0 0  
## 5 5 50000 1 2 1 57 -1 0 -1 0 0  
## 6 6 50000 1 1 2 37 0 0 0 0 0  
## # ℹ 14 more variables: PAY\_6 <dbl>, BILL\_AMT1 <dbl>, BILL\_AMT2 <dbl>,  
## # BILL\_AMT3 <dbl>, BILL\_AMT4 <dbl>, BILL\_AMT5 <dbl>, BILL\_AMT6 <dbl>,  
## # PAY\_AMT1 <dbl>, PAY\_AMT2 <dbl>, PAY\_AMT3 <dbl>, PAY\_AMT4 <dbl>,  
## # PAY\_AMT5 <dbl>, PAY\_AMT6 <dbl>, default\_payment <dbl>

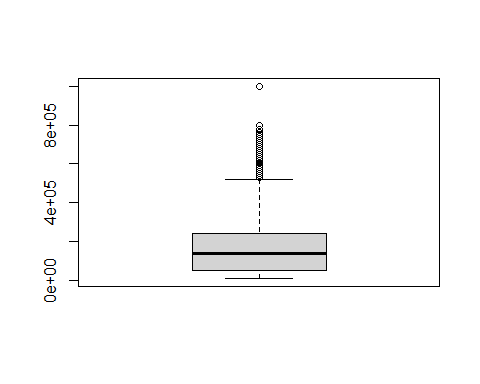
## eXPLORATORY DATA ANALYSIS

dim(df\_clean)

## [1] 30000 25

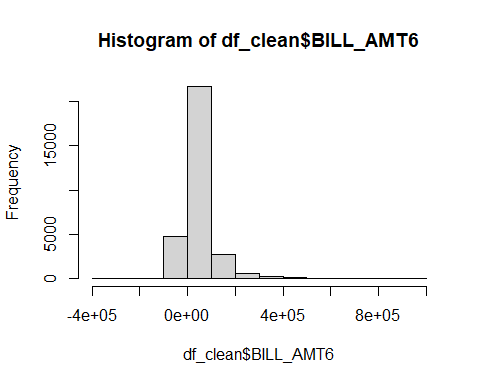
#For the choosen dataset, what are the necessary data wrangling steps to make the data ready this step involve checking for missing values, making sure that the data types of the features are in correct fomart, checking for outliers using boxplot

boxplot(df\_clean$LIMIT\_BAL)

 There are some outliers in the balance variable

str(df\_clean)

## tibble [30,000 × 25] (S3: tbl\_df/tbl/data.frame)  
## $ ID : num [1:30000] 1 2 3 4 5 6 7 8 9 10 ...  
## $ LIMIT\_BAL : num [1:30000] 20000 120000 90000 50000 50000 50000 500000 100000 140000 20000 ...  
## $ SEX : num [1:30000] 2 2 2 2 1 1 1 2 2 1 ...  
## $ EDUCATION : num [1:30000] 2 2 2 2 2 1 1 2 3 3 ...  
## $ MARRIAGE : num [1:30000] 1 2 2 1 1 2 2 2 1 2 ...  
## $ AGE : num [1:30000] 24 26 34 37 57 37 29 23 28 35 ...  
## $ PAY\_0 : num [1:30000] 2 -1 0 0 -1 0 0 0 0 -2 ...  
## $ PAY\_2 : num [1:30000] 2 2 0 0 0 0 0 -1 0 -2 ...  
## $ PAY\_3 : num [1:30000] -1 0 0 0 -1 0 0 -1 2 -2 ...  
## $ PAY\_4 : num [1:30000] -1 0 0 0 0 0 0 0 0 -2 ...  
## $ PAY\_5 : num [1:30000] -2 0 0 0 0 0 0 0 0 -1 ...  
## $ PAY\_6 : num [1:30000] -2 2 0 0 0 0 0 -1 0 -1 ...  
## $ BILL\_AMT1 : num [1:30000] 3913 2682 29239 46990 8617 ...  
## $ BILL\_AMT2 : num [1:30000] 3102 1725 14027 48233 5670 ...  
## $ BILL\_AMT3 : num [1:30000] 689 2682 13559 49291 35835 ...  
## $ BILL\_AMT4 : num [1:30000] 0 3272 14331 28314 20940 ...  
## $ BILL\_AMT5 : num [1:30000] 0 3455 14948 28959 19146 ...  
## $ BILL\_AMT6 : num [1:30000] 0 3261 15549 29547 19131 ...  
## $ PAY\_AMT1 : num [1:30000] 0 0 1518 2000 2000 ...  
## $ PAY\_AMT2 : num [1:30000] 689 1000 1500 2019 36681 ...  
## $ PAY\_AMT3 : num [1:30000] 0 1000 1000 1200 10000 657 38000 0 432 0 ...  
## $ PAY\_AMT4 : num [1:30000] 0 1000 1000 1100 9000 ...  
## $ PAY\_AMT5 : num [1:30000] 0 0 1000 1069 689 ...  
## $ PAY\_AMT6 : num [1:30000] 0 2000 5000 1000 679 ...  
## $ default\_payment: num [1:30000] 1 1 0 0 0 0 0 0 0 0 ...



colnames(df\_clean)

## [1] "ID" "LIMIT\_BAL" "SEX" "EDUCATION"   
## [5] "MARRIAGE" "AGE" "PAY\_0" "PAY\_2"   
## [9] "PAY\_3" "PAY\_4" "PAY\_5" "PAY\_6"   
## [13] "BILL\_AMT1" "BILL\_AMT2" "BILL\_AMT3" "BILL\_AMT4"   
## [17] "BILL\_AMT5" "BILL\_AMT6" "PAY\_AMT1" "PAY\_AMT2"   
## [21] "PAY\_AMT3" "PAY\_AMT4" "PAY\_AMT5" "PAY\_AMT6"   
## [25] "default\_payment"

count(df\_clean, vars = EDUCATION)

## # A tibble: 7 × 2  
## vars n  
## <dbl> <int>  
## 1 0 14  
## 2 1 10585  
## 3 2 14030  
## 4 3 4917  
## 5 4 123  
## 6 5 280  
## 7 6 51

#head(df\_clean)

DUCATION: “Education (1 = graduate school; 2 = university; 3 = high school; 4 = others)

summary(df\_clean)

## ID LIMIT\_BAL SEX EDUCATION   
## Min. : 1 Min. : 10000 Min. :1.000 Min. :0.000   
## 1st Qu.: 7501 1st Qu.: 50000 1st Qu.:1.000 1st Qu.:1.000   
## Median :15000 Median : 140000 Median :2.000 Median :2.000   
## Mean :15000 Mean : 167484 Mean :1.604 Mean :1.853   
## 3rd Qu.:22500 3rd Qu.: 240000 3rd Qu.:2.000 3rd Qu.:2.000   
## Max. :30000 Max. :1000000 Max. :2.000 Max. :6.000   
## MARRIAGE AGE PAY\_0 PAY\_2   
## Min. :0.000 Min. :21.00 Min. :-2.0000 Min. :-2.0000   
## 1st Qu.:1.000 1st Qu.:28.00 1st Qu.:-1.0000 1st Qu.:-1.0000   
## Median :2.000 Median :34.00 Median : 0.0000 Median : 0.0000   
## Mean :1.552 Mean :35.49 Mean :-0.0167 Mean :-0.1338   
## 3rd Qu.:2.000 3rd Qu.:41.00 3rd Qu.: 0.0000 3rd Qu.: 0.0000   
## Max. :3.000 Max. :79.00 Max. : 8.0000 Max. : 8.0000   
## PAY\_3 PAY\_4 PAY\_5 PAY\_6   
## Min. :-2.0000 Min. :-2.0000 Min. :-2.0000 Min. :-2.0000   
## 1st Qu.:-1.0000 1st Qu.:-1.0000 1st Qu.:-1.0000 1st Qu.:-1.0000   
## Median : 0.0000 Median : 0.0000 Median : 0.0000 Median : 0.0000   
## Mean :-0.1662 Mean :-0.2207 Mean :-0.2662 Mean :-0.2911   
## 3rd Qu.: 0.0000 3rd Qu.: 0.0000 3rd Qu.: 0.0000 3rd Qu.: 0.0000   
## Max. : 8.0000 Max. : 8.0000 Max. : 8.0000 Max. : 8.0000   
## BILL\_AMT1 BILL\_AMT2 BILL\_AMT3 BILL\_AMT4   
## Min. :-165580 Min. :-69777 Min. :-157264 Min. :-170000   
## 1st Qu.: 3559 1st Qu.: 2985 1st Qu.: 2666 1st Qu.: 2327   
## Median : 22382 Median : 21200 Median : 20089 Median : 19052   
## Mean : 51223 Mean : 49179 Mean : 47013 Mean : 43263   
## 3rd Qu.: 67091 3rd Qu.: 64006 3rd Qu.: 60165 3rd Qu.: 54506   
## Max. : 964511 Max. :983931 Max. :1664089 Max. : 891586   
## BILL\_AMT5 BILL\_AMT6 PAY\_AMT1 PAY\_AMT2   
## Min. :-81334 Min. :-339603 Min. : 0 Min. : 0   
## 1st Qu.: 1763 1st Qu.: 1256 1st Qu.: 1000 1st Qu.: 833   
## Median : 18105 Median : 17071 Median : 2100 Median : 2009   
## Mean : 40311 Mean : 38872 Mean : 5664 Mean : 5921   
## 3rd Qu.: 50191 3rd Qu.: 49198 3rd Qu.: 5006 3rd Qu.: 5000   
## Max. :927171 Max. : 961664 Max. :873552 Max. :1684259   
## PAY\_AMT3 PAY\_AMT4 PAY\_AMT5 PAY\_AMT6   
## Min. : 0 Min. : 0 Min. : 0.0 Min. : 0.0   
## 1st Qu.: 390 1st Qu.: 296 1st Qu.: 252.5 1st Qu.: 117.8   
## Median : 1800 Median : 1500 Median : 1500.0 Median : 1500.0   
## Mean : 5226 Mean : 4826 Mean : 4799.4 Mean : 5215.5   
## 3rd Qu.: 4505 3rd Qu.: 4013 3rd Qu.: 4031.5 3rd Qu.: 4000.0   
## Max. :896040 Max. :621000 Max. :426529.0 Max. :528666.0   
## default\_payment   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0000   
## Mean :0.2212   
## 3rd Qu.:0.0000   
## Max. :1.0000

## Warning in pal\_name(palette, type): Unknown palette steelblue

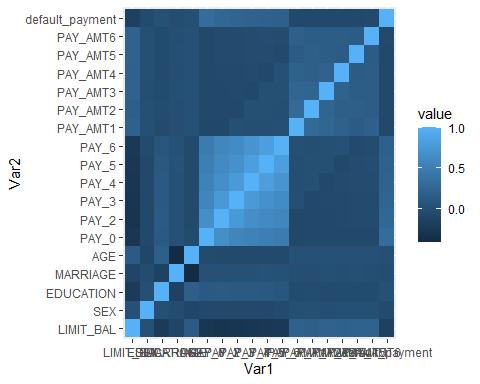
 Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.

## # A tibble: 6 × 18  
## LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5 PAY\_6  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 20000 2 2 1 24 2 2 -1 -1 -2 -2  
## 2 120000 2 2 2 26 -1 2 0 0 0 2  
## 3 90000 2 2 2 34 0 0 0 0 0 0  
## 4 50000 2 2 1 37 0 0 0 0 0 0  
## 5 50000 1 2 1 57 -1 0 -1 0 0 0  
## 6 50000 1 1 2 37 0 0 0 0 0 0  
## # ℹ 7 more variables: PAY\_AMT1 <dbl>, PAY\_AMT2 <dbl>, PAY\_AMT3 <dbl>,  
## # PAY\_AMT4 <dbl>, PAY\_AMT5 <dbl>, PAY\_AMT6 <dbl>, default\_payment <dbl>

## LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4  
## LIMIT\_BAL 1.00 0.02 -0.22 -0.11 0.14 -0.27 -0.30 -0.29 -0.27  
## SEX 0.02 1.00 0.01 -0.03 -0.09 -0.06 -0.07 -0.07 -0.06  
## EDUCATION -0.22 0.01 1.00 -0.14 0.18 0.11 0.12 0.11 0.11  
## MARRIAGE -0.11 -0.03 -0.14 1.00 -0.41 0.02 0.02 0.03 0.03  
## AGE 0.14 -0.09 0.18 -0.41 1.00 -0.04 -0.05 -0.05 -0.05  
## PAY\_0 -0.27 -0.06 0.11 0.02 -0.04 1.00 0.67 0.57 0.54  
## PAY\_5 PAY\_6 PAY\_AMT1 PAY\_AMT2 PAY\_AMT3 PAY\_AMT4 PAY\_AMT5 PAY\_AMT6  
## LIMIT\_BAL -0.25 -0.24 0.20 0.18 0.21 0.20 0.22 0.22  
## SEX -0.06 -0.04 0.00 0.00 -0.01 0.00 0.00 0.00  
## EDUCATION 0.10 0.08 -0.04 -0.03 -0.04 -0.04 -0.04 -0.04  
## MARRIAGE 0.04 0.03 -0.01 -0.01 0.00 -0.01 0.00 -0.01  
## AGE -0.05 -0.05 0.03 0.02 0.03 0.02 0.02 0.02  
## PAY\_0 0.51 0.47 -0.08 -0.07 -0.07 -0.06 -0.06 -0.06  
## default\_payment  
## LIMIT\_BAL -0.15  
## SEX -0.04  
## EDUCATION 0.03  
## MARRIAGE -0.02  
## AGE 0.01  
## PAY\_0 0.32

##   
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':  
##   
## smiths



## modeling techniques can be used to answer the research

selecting columns for logistic regression

selected\_columns\_forML <- df[c( "LIMIT\_BAL" , "SEX" , "EDUCATION", "MARRIAGE" ,"AGE" , "PAY\_0" ,"PAY\_2" , "PAY\_3","PAY\_4" , "PAY\_5" , "PAY\_6" , "PAY\_AMT1" , "PAY\_AMT2" ,"PAY\_AMT3" , "PAY\_AMT4" , "PAY\_AMT5","PAY\_AMT6" , "default\_payment")]  
head(selected\_columns\_forML)

## # A tibble: 6 × 18  
## LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5 PAY\_6  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 20000 2 2 1 24 2 2 -1 -1 -2 -2  
## 2 120000 2 2 2 26 -1 2 0 0 0 2  
## 3 90000 2 2 2 34 0 0 0 0 0 0  
## 4 50000 2 2 1 37 0 0 0 0 0 0  
## 5 50000 1 2 1 57 -1 0 -1 0 0 0  
## 6 50000 1 1 2 37 0 0 0 0 0 0  
## # ℹ 7 more variables: PAY\_AMT1 <dbl>, PAY\_AMT2 <dbl>, PAY\_AMT3 <dbl>,  
## # PAY\_AMT4 <dbl>, PAY\_AMT5 <dbl>, PAY\_AMT6 <dbl>, default\_payment <dbl>

dim(selected\_columns\_forML)

## [1] 30000 18

#standardisation  
selected\_columns\_forML[, 1:17] <- scale(selected\_columns\_forML[, 1:17])  
  
head(selected\_columns\_forML)

## # A tibble: 6 × 18  
## LIMIT\_BAL SEX EDUCATION MARRIAGE AGE PAY\_0 PAY\_2 PAY\_3 PAY\_4 PAY\_5  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 -1.14 0.810 0.186 -1.06 -1.25 1.79 1.78 -0.697 -0.667 -1.53   
## 2 -0.366 0.810 0.186 0.859 -1.03 -0.875 1.78 0.139 0.189 0.235  
## 3 -0.597 0.810 0.186 0.859 -0.161 0.0149 0.112 0.139 0.189 0.235  
## 4 -0.905 0.810 0.186 -1.06 0.164 0.0149 0.112 0.139 0.189 0.235  
## 5 -0.905 -1.23 0.186 -1.06 2.33 -0.875 0.112 -0.697 0.189 0.235  
## 6 -0.905 -1.23 -1.08 0.859 0.164 0.0149 0.112 0.139 0.189 0.235  
## # ℹ 8 more variables: PAY\_6 <dbl>, PAY\_AMT1 <dbl>, PAY\_AMT2 <dbl>,  
## # PAY\_AMT3 <dbl>, PAY\_AMT4 <dbl>, PAY\_AMT5 <dbl>, PAY\_AMT6 <dbl>,  
## # default\_payment <dbl>

#create a list of random number ranging from 1 to number of rows from actual data   
#and 70% of the data into training data   
  
data2 = sort(sample(nrow(selected\_columns\_forML), nrow(selected\_columns\_forML)\*.7))  
  
#creating training data set by selecting the output row values  
train <- selected\_columns\_forML[data2,]  
  
#creating test data set by not selecting the output row values  
test <- selected\_columns\_forML[-data2,]

# Create a logistic regression model  
model <- glm(default\_payment ~ ., data = train, family = binomial(link = "logit"))  
  
# Predict on the testing set  
predicted <- predict(model, newdata = test, type = "response")  
predicted <- ifelse(predicted > 0.5, 1, 0)  
  
# Create the confusion matrix  
confusionMatrixData <- table(predicted, test$default\_payment)  
print(confusionMatrixData)

##   
## predicted 0 1  
## 0 6842 1546  
## 1 175 437

# Plot the confusion matrix  
library(caret) # For confusion matrix plotting

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

confusionMatrix(confusionMatrixData)

## Confusion Matrix and Statistics  
##   
##   
## predicted 0 1  
## 0 6842 1546  
## 1 175 437  
##   
## Accuracy : 0.8088   
## 95% CI : (0.8005, 0.8169)  
## No Information Rate : 0.7797   
## P-Value [Acc > NIR] : 6.886e-12   
##   
## Kappa : 0.2599   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9751   
## Specificity : 0.2204   
## Pos Pred Value : 0.8157   
## Neg Pred Value : 0.7141   
## Prevalence : 0.7797   
## Detection Rate : 0.7602   
## Detection Prevalence : 0.9320   
## Balanced Accuracy : 0.5977   
##   
## 'Positive' Class : 0   
##

## What metrics will be used to evaluate the quality of the data analysis?

One indicator to assess the caliber of the data analysis is the classification model’s accuracy. The accuracy in this instance is 0.8121, meaning that 81.21% of the time the model accurately predicts whether or not a client would fail on a payment.

The accuracy measure, which is the percentage of true positives (cases properly predicted as default payment) out of all cases forecasted as default payment, is another useful metric. The precision for forecasting default payment in this instance is 0.2355, meaning that only 23.55% of the instances that were projected to be default payments were in fact true positives.

summary(model)

##   
## Call:  
## glm(formula = default\_payment ~ ., family = binomial(link = "logit"),   
## data = train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.466692 0.019744 -74.285 < 2e-16 \*\*\*  
## LIMIT\_BAL -0.109961 0.023072 -4.766 1.88e-06 \*\*\*  
## SEX -0.050915 0.017865 -2.850 0.004372 \*\*   
## EDUCATION -0.088586 0.019654 -4.507 6.57e-06 \*\*\*  
## MARRIAGE -0.084815 0.019791 -4.286 1.82e-05 \*\*\*  
## AGE 0.049238 0.019652 2.505 0.012229 \*   
## PAY\_0 0.630052 0.023851 26.416 < 2e-16 \*\*\*  
## PAY\_2 0.083925 0.028599 2.935 0.003340 \*\*   
## PAY\_3 0.127341 0.031833 4.000 6.33e-05 \*\*\*  
## PAY\_4 0.019632 0.034849 0.563 0.573199   
## PAY\_5 0.007184 0.036643 0.196 0.844573   
## PAY\_6 0.029340 0.029949 0.980 0.327248   
## PAY\_AMT1 -0.216190 0.043358 -4.986 6.16e-07 \*\*\*  
## PAY\_AMT2 -0.157934 0.048180 -3.278 0.001046 \*\*   
## PAY\_AMT3 -0.033375 0.028137 -1.186 0.235563   
## PAY\_AMT4 -0.044218 0.027076 -1.633 0.102450   
## PAY\_AMT5 -0.113631 0.033627 -3.379 0.000727 \*\*\*  
## PAY\_AMT6 -0.077128 0.029591 -2.606 0.009148 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 22213 on 20999 degrees of freedom  
## Residual deviance: 19614 on 20982 degrees of freedom  
## AIC: 19650  
##   
## Number of Fisher Scoring iterations: 6

## Results

Based on the logistic regression results, several variables were found to be significant predictors of default payment for next month. These variables include LIMIT\_BAL (credit limit), SEX, EDUCATION, MARRIAGE, AGE, PAY\_0, PAY\_2, and PAY\_3. The coefficients of these variables indicate the direction and magnitude of their effect on the likelihood of default payment. For example, a higher credit limit (LIMIT\_BAL) is associated with a lower probability of default payment, while being female (SEX) is associated with a higher probability of default payment. Overall, this classification model can be useful in predicting whether a customer is likely to default on their payment next month.

## conclusion

The research issue is partially addressed by the analysis; a full response will require more investigation. There may be some predictive potential in the model since it correctly classifies 81.21% of instances. Nevertheless, the low accuracy (23.55%) suggests that a large number of the anticipated defaults are false positives. This makes one wonder if the model will work well in practical settings.

Furthermore, the narrow scope of the analysis, only predicting next month’s defaults, restricts the depth of insights that can be gained. Exploring longer timeframes or incorporating other financial metrics could provide a more comprehensive understanding of customer default behavior

Various classification methods might be investigated and hyperparameters adjusted to increase accuracy in order to improve the analysis. Optimizing model performance can also involve addressing data imbalances using methods like oversampling or undersampling. To improve model accuracy, extra information may be captured using feature engineering, which is the process of developing new features based on preexisting data. Predictions may also be improved by adding data from other sources, such as social media or economic indicators.