GBUS Final Project - Loan Data Analysis Report

Shwetha Srinivasan Natesan

2023-10-29

Introduction

This report does the analysis of loan data collected from a national bank in 2017. This dataset consists of about 3,500 customers who secured a personal loan living in the Middle Atlantic and Northeast regions of the United States. This analysis is done to predict whether the customer will default on their loan or not. This bank also has a history of customers who tend to default on their loans in the past. So the goal of this analysis is to predict what are the factors that are causing to make the loan default and also how it can be improved.

Data Analysis

Libraries used

The following are the libraries that are used in this R code to perform summary, plots and regression models.

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── 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(dplyr)  
library(skimr)  
library(caret)

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

library(glmnet)

## Loading required package: Matrix  
##   
## Attaching package: 'Matrix'  
##   
## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack  
##   
## Loaded glmnet 4.1-8

library(tidymodels)

## ── Attaching packages ────────────────────────────────────── tidymodels 1.1.1 ──  
## ✔ broom 1.0.5 ✔ rsample 1.2.0  
## ✔ dials 1.2.0 ✔ tune 1.1.2  
## ✔ infer 1.0.5 ✔ workflows 1.1.3  
## ✔ modeldata 1.2.0 ✔ workflowsets 1.0.1  
## ✔ parsnip 1.1.1 ✔ yardstick 1.2.0  
## ✔ recipes 1.0.8   
## ── Conflicts ───────────────────────────────────────── tidymodels\_conflicts() ──  
## ✖ scales::discard() masks purrr::discard()  
## ✖ Matrix::expand() masks tidyr::expand()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ recipes::fixed() masks stringr::fixed()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ caret::lift() masks purrr::lift()  
## ✖ Matrix::pack() masks tidyr::pack()  
## ✖ yardstick::precision() masks caret::precision()  
## ✖ yardstick::recall() masks caret::recall()  
## ✖ yardstick::sensitivity() masks caret::sensitivity()  
## ✖ yardstick::spec() masks readr::spec()  
## ✖ yardstick::specificity() masks caret::specificity()  
## ✖ recipes::step() masks stats::step()  
## ✖ Matrix::unpack() masks tidyr::unpack()  
## ✖ recipes::update() masks Matrix::update(), stats::update()  
## • Learn how to get started at https://www.tidymodels.org/start/

library(ranger)  
library(pROC)

## Type 'citation("pROC")' for a citation.  
##   
## Attaching package: 'pROC'  
##   
## The following objects are masked from 'package:stats':  
##   
## cov, smooth, var

library(parsnip)  
library(tune)  
library(rsample)  
library(MASS)

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

##Loading the dataset The dataset loan\_data is loaded.

loan\_data <- readRDS("~/R/loan\_data.rds")  
loan\_data

## # A tibble: 4,110 × 16  
## loan\_default loan\_amount installment interest\_rate loan\_purpose   
## <fct> <int> <dbl> <dbl> <fct>   
## 1 yes 35000 927. 17.2 small\_business   
## 2 yes 10000 260. 11.5 small\_business   
## 3 no 28800 942. 8.97 debt\_consolidation  
## 4 yes 4475 165. 10 medical   
## 5 no 3600 111. 9.72 medical   
## 6 yes 12800 389. 20 medical   
## 7 yes 35000 927. 18.2 debt\_consolidation  
## 8 no 26000 619. 12.0 debt\_consolidation  
## 9 no 5500 176. 7.97 debt\_consolidation  
## 10 no 40000 952. 11.0 home\_improvement   
## # ℹ 4,100 more rows  
## # ℹ 11 more variables: application\_type <fct>, term <fct>, homeownership <fct>,  
## # annual\_income <dbl>, current\_job\_years <dbl>, debt\_to\_income <dbl>,  
## # total\_credit\_lines <int>, years\_credit\_history <dbl>,  
## # missed\_payment\_2\_yr <fct>, history\_bankruptcy <fct>,  
## # history\_tax\_liens <fct>

Exploratory data analysis

The following commands such as str(), dim(), head(), glimpse(),skim(), and summary() is used to view the variables and data types in loan\_data and to get an understanding about the statistical distribution in it.

str(loan\_data)

## tibble [4,110 × 16] (S3: tbl\_df/tbl/data.frame)  
## $ loan\_default : Factor w/ 2 levels "yes","no": 1 1 2 1 2 1 1 2 2 2 ...  
## $ loan\_amount : int [1:4110] 35000 10000 28800 4475 3600 12800 35000 26000 5500 40000 ...  
## $ installment : num [1:4110] 927 260 942 165 111 ...  
## $ interest\_rate : num [1:4110] 17.25 11.5 8.97 10 9.72 ...  
## $ loan\_purpose : Factor w/ 5 levels "debt\_consolidation",..: 4 4 1 3 3 3 1 1 1 5 ...  
## $ application\_type : Factor w/ 2 levels "individual","joint": 1 1 1 1 1 1 1 1 1 1 ...  
## $ term : Factor w/ 2 levels "three\_year","five\_year": 2 2 1 1 1 2 2 2 1 2 ...  
## $ homeownership : Factor w/ 3 levels "mortgage","rent",..: 2 1 2 2 1 2 1 1 2 1 ...  
## $ annual\_income : num [1:4110] 104660 57000 160000 37000 72000 ...  
## $ current\_job\_years : num [1:4110] 2 10 10 1 4 10 0 5 4 3 ...  
## $ debt\_to\_income : num [1:4110] 29.41 23.79 5.96 13.82 22.68 ...  
## $ total\_credit\_lines : int [1:4110] 27 14 35 7 35 57 34 24 12 12 ...  
## $ years\_credit\_history: num [1:4110] 15 4 17 5 11 14 22 16 9 12 ...  
## $ missed\_payment\_2\_yr : Factor w/ 2 levels "yes","no": 2 2 2 2 2 2 2 2 2 2 ...  
## $ history\_bankruptcy : Factor w/ 2 levels "yes","no": 2 2 1 2 2 2 2 2 2 2 ...  
## $ history\_tax\_liens : Factor w/ 2 levels "yes","no": 2 2 2 2 2 2 2 2 2 2 ...

dim(loan\_data)

## [1] 4110 16

head(loan\_data)

## # A tibble: 6 × 16  
## loan\_default loan\_amount installment interest\_rate loan\_purpose   
## <fct> <int> <dbl> <dbl> <fct>   
## 1 yes 35000 927. 17.2 small\_business   
## 2 yes 10000 260. 11.5 small\_business   
## 3 no 28800 942. 8.97 debt\_consolidation  
## 4 yes 4475 165. 10 medical   
## 5 no 3600 111. 9.72 medical   
## 6 yes 12800 389. 20 medical   
## # ℹ 11 more variables: application\_type <fct>, term <fct>, homeownership <fct>,  
## # annual\_income <dbl>, current\_job\_years <dbl>, debt\_to\_income <dbl>,  
## # total\_credit\_lines <int>, years\_credit\_history <dbl>,  
## # missed\_payment\_2\_yr <fct>, history\_bankruptcy <fct>,  
## # history\_tax\_liens <fct>

glimpse(loan\_data)

## Rows: 4,110  
## Columns: 16  
## $ loan\_default <fct> yes, yes, no, yes, no, yes, yes, no, no, no, no, …  
## $ loan\_amount <int> 35000, 10000, 28800, 4475, 3600, 12800, 35000, 26…  
## $ installment <dbl> 927.29, 259.58, 941.65, 164.99, 110.70, 389.10, 9…  
## $ interest\_rate <dbl> 17.25, 11.50, 8.97, 10.00, 9.72, 20.00, 18.25, 11…  
## $ loan\_purpose <fct> small\_business, small\_business, debt\_consolidatio…  
## $ application\_type <fct> individual, individual, individual, individual, i…  
## $ term <fct> five\_year, five\_year, three\_year, three\_year, thr…  
## $ homeownership <fct> rent, mortgage, rent, rent, mortgage, rent, mortg…  
## $ annual\_income <dbl> 104660, 57000, 160000, 37000, 72000, 73000, 16700…  
## $ current\_job\_years <dbl> 2, 10, 10, 1, 4, 10, 0, 5, 4, 3, 10, 10, 5, 10, 1…  
## $ debt\_to\_income <dbl> 29.41, 23.79, 5.96, 13.82, 22.68, 30.94, 25.91, 7…  
## $ total\_credit\_lines <int> 27, 14, 35, 7, 35, 57, 34, 24, 12, 12, 16, 9, 17,…  
## $ years\_credit\_history <dbl> 15, 4, 17, 5, 11, 14, 22, 16, 9, 12, 22, 9, 8, 17…  
## $ missed\_payment\_2\_yr <fct> no, no, no, no, no, no, no, no, no, no, no, no, n…  
## $ history\_bankruptcy <fct> no, no, yes, no, no, no, no, no, no, no, no, no, …  
## $ history\_tax\_liens <fct> no, no, no, no, no, no, no, no, no, no, no, no, n…

skim(loan\_data)

Data summary

|  |  |
| --- | --- |
| Name | loan\_data |
| Number of rows | 4110 |
| Number of columns | 16 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 8 |
| numeric | 8 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| loan\_default | 0 | 1 | FALSE | 2 | no: 2580, yes: 1530 |
| loan\_purpose | 0 | 1 | FALSE | 5 | deb: 1218, cre: 879, sma: 853, med: 635 |
| application\_type | 0 | 1 | FALSE | 2 | ind: 3494, joi: 616 |
| term | 0 | 1 | FALSE | 2 | thr: 2588, fiv: 1522 |
| homeownership | 0 | 1 | FALSE | 3 | mor: 1937, ren: 1666, own: 507 |
| missed\_payment\_2\_yr | 0 | 1 | FALSE | 2 | no: 3640, yes: 470 |
| history\_bankruptcy | 0 | 1 | FALSE | 2 | no: 3624, yes: 486 |
| history\_tax\_liens | 0 | 1 | FALSE | 2 | no: 4050, yes: 60 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| loan\_amount | 0 | 1 | 16692.79 | 10038.89 | 1000.00 | 9600.00 | 15000.00 | 24000.00 | 40000.00 | ▆▇▅▃▂ |
| installment | 0 | 1 | 489.42 | 289.50 | 31.04 | 274.82 | 421.97 | 663.98 | 1566.59 | ▇▇▅▂▁ |
| interest\_rate | 0 | 1 | 11.38 | 3.92 | 4.72 | 8.22 | 11.25 | 13.75 | 20.00 | ▆▆▇▃▃ |
| annual\_income | 0 | 1 | 73015.01 | 37203.11 | 3000.00 | 45000.00 | 65000.00 | 92000.00 | 200000.00 | ▃▇▃▁▁ |
| current\_job\_years | 0 | 1 | 5.80 | 3.69 | 0.00 | 2.00 | 5.00 | 10.00 | 10.00 | ▆▃▂▂▇ |
| debt\_to\_income | 0 | 1 | 20.04 | 14.23 | 0.00 | 11.85 | 18.59 | 26.13 | 437.61 | ▇▁▁▁▁ |
| total\_credit\_lines | 0 | 1 | 22.47 | 12.03 | 2.00 | 14.00 | 20.00 | 29.00 | 87.00 | ▇▇▂▁▁ |
| years\_credit\_history | 0 | 1 | 15.76 | 7.22 | 3.00 | 11.00 | 14.00 | 19.00 | 51.00 | ▆▇▂▁▁ |

summary(loan\_data)

## loan\_default loan\_amount installment interest\_rate   
## yes:1530 Min. : 1000 Min. : 31.04 Min. : 4.72   
## no :2580 1st Qu.: 9600 1st Qu.: 274.82 1st Qu.: 8.22   
## Median :15000 Median : 421.97 Median :11.25   
## Mean :16693 Mean : 489.42 Mean :11.38   
## 3rd Qu.:24000 3rd Qu.: 663.99 3rd Qu.:13.75   
## Max. :40000 Max. :1566.59 Max. :20.00   
## loan\_purpose application\_type term homeownership   
## debt\_consolidation:1218 individual:3494 three\_year:2588 mortgage:1937   
## credit\_card : 879 joint : 616 five\_year :1522 rent :1666   
## medical : 635 own : 507   
## small\_business : 853   
## home\_improvement : 525   
##   
## annual\_income current\_job\_years debt\_to\_income total\_credit\_lines  
## Min. : 3000 Min. : 0.000 Min. : 0.00 Min. : 2.00   
## 1st Qu.: 45000 1st Qu.: 2.000 1st Qu.: 11.85 1st Qu.:14.00   
## Median : 65000 Median : 5.000 Median : 18.59 Median :20.00   
## Mean : 73015 Mean : 5.802 Mean : 20.04 Mean :22.47   
## 3rd Qu.: 92000 3rd Qu.:10.000 3rd Qu.: 26.13 3rd Qu.:29.00   
## Max. :200000 Max. :10.000 Max. :437.61 Max. :87.00   
## years\_credit\_history missed\_payment\_2\_yr history\_bankruptcy history\_tax\_liens  
## Min. : 3.00 yes: 470 yes: 486 yes: 60   
## 1st Qu.:11.00 no :3640 no :3624 no :4050   
## Median :14.00   
## Mean :15.76   
## 3rd Qu.:19.00   
## Max. :51.00

Research Questions

1. Does the interest rate affect the loan default?

The summary of the interest rate is used here to calculate to see if that is the reason customers might take loan default. So from the analysis we can see that the interest rate gradually increases which is how normally a loan amount would have. So the average interest rate increases around 0.22% which is normal.

loan\_data %>% group\_by(interest\_rate) %>%  
 summarise(min\_value = min(interest\_rate),  
 avg\_value = mean(interest\_rate),  
 max\_value = max(interest\_rate),  
 sd\_value = sd(interest\_rate),  
 value\_greater\_ten = mean(interest\_rate >= 10))

## # A tibble: 80 × 6  
## interest\_rate min\_value avg\_value max\_value sd\_value value\_greater\_ten  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 4.72 4.72 4.72 4.72 0 0  
## 2 4.97 4.97 4.97 4.97 0 0  
## 3 5.22 5.22 5.22 5.22 0 0  
## 4 5.47 5.47 5.47 5.47 0 0  
## 5 5.72 5.72 5.72 5.72 0 0  
## 6 5.97 5.97 5.97 5.97 0 0  
## 7 6.22 6.22 6.22 6.22 0 0  
## 8 6.47 6.47 6.47 6.47 0 0  
## 9 6.72 6.72 6.72 6.72 0 0  
## 10 6.97 6.97 6.97 6.97 0 0  
## # ℹ 70 more rows

2. Are there any relation with credit history and loan defaut?

This summary is done just to the credit history of the customers over the years. The credit history over 5 years shows that there are maximum customers who managed to repay their loan debts. So, from this we can see that there are large amount of customers who pay their loans.

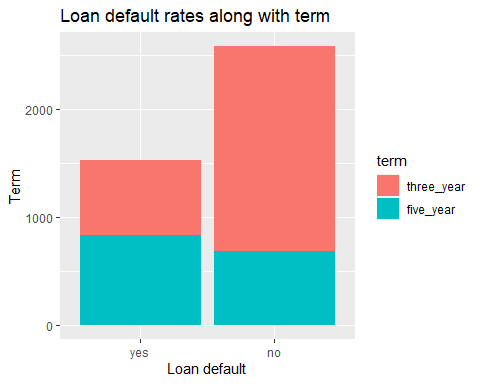
loan\_data %>% group\_by(years\_credit\_history) %>%  
 summarise(min\_years = min(years\_credit\_history),  
 avg\_years = mean(years\_credit\_history),  
 max\_years = max(years\_credit\_history),  
 sd\_years = sd(years\_credit\_history),  
 credityears\_greater\_five = mean(years\_credit\_history >= 5))

## # A tibble: 47 × 6  
## years\_credit\_history min\_years avg\_years max\_years sd\_years  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 3 3 3 3 0  
## 2 4 4 4 4 0  
## 3 5 5 5 5 0  
## 4 6 6 6 6 0  
## 5 7 7 7 7 0  
## 6 8 8 8 8 0  
## 7 9 9 9 9 0  
## 8 10 10 10 10 0  
## 9 11 11 11 11 0  
## 10 12 12 12 12 0  
## # ℹ 37 more rows  
## # ℹ 1 more variable: credityears\_greater\_five <dbl>

3. What is the term year when it comes to loan default?

The term year is the period where the customers choose to keep their loan. It consists of three year term and five year term. So from the bar chart we can see that the customers tend to prefer three year term and the loan default factor has maximum no for it.

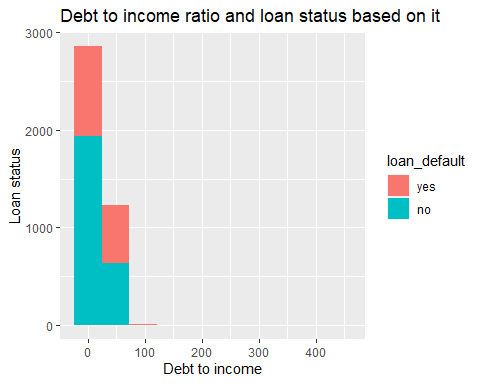
ggplot(data = loan\_data, aes(x = loan\_default, fill = term))+  
 geom\_bar()+  
 labs(title = "Loan default rates along with term",  
 x = "Loan default",y = "Term")



4. Does debt to income ratio affect the loan default status?

The debt to income ratio calculated based on the customers annual income and the debt that they have choosen to pay for their loan. So the histogram, shows that loan default has maximum no so the debt to income ratio doesn’t affect the loan payment.

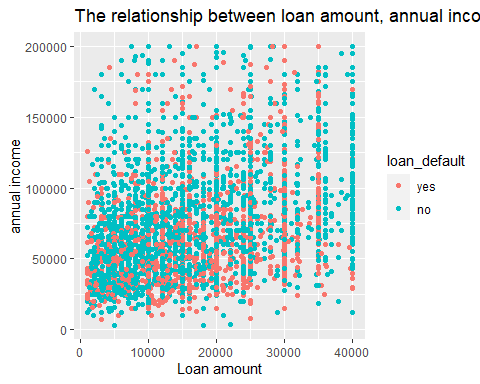
ggplot(data = loan\_data, aes(x = debt\_to\_income, fill = loan\_default))+  
 geom\_histogram(bins = 10)+  
 labs(title = "Debt to income ratio and loan status based on it", x = "Debt to income", y = "Loan status")



5. To see if the loan amount and annual income is related to loan default status?

The annual income of the customers is noted to see if the customer can afford to pay the loan. The loan amount is also plotted to see the average loan amount the customers tend to take. The scatterplot shows the maximum annual amount is around 200000. The maximum loan amount taken is usually from 40000 to the annual income that ranges 100000 to 150000. So in this range, the loan default value is no.

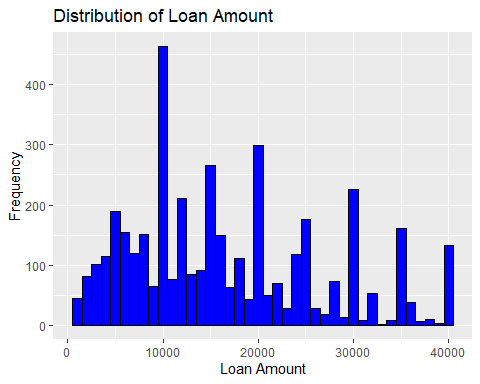
ggplot(data = loan\_data, aes(x = loan\_amount, y = annual\_income, col = loan\_default))+  
 geom\_point()+  
 labs(title = "The relationship between loan amount, annual income and loan status", x = "Loan amount", y = "annual income")



6. To see the maximum loan amount range that customers prefer

A histogram is plotted to see what is the loan amount that maximum customers prefer and we can see that 10000 is the amount that they prefer at a large rate.

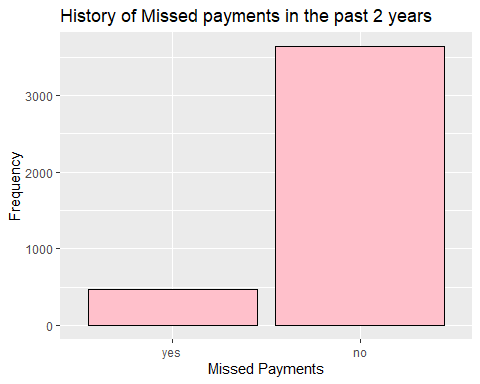
ggplot(data = loan\_data, aes(x = loan\_amount)) +  
 geom\_histogram(binwidth = 1000, fill = "blue", color = "black") +  
 labs(title = "Distribution of Loan Amount",  
 x = "Loan Amount",  
 y = "Frequency")



7. To check the factor for missed payment for the past two years

The missed payment factor for the past two years is to check the whether if it is also a reason for the loan default. The bar chart shows that the customers tend to pay their loans properly as the no factor is more. So, this is not the reason then.

ggplot(data = loan\_data, aes(x = missed\_payment\_2\_yr)) +  
 geom\_bar(fill = "pink", color = "black") +  
 labs(title = "History of Missed payments in the past 2 years",  
 x = "Missed Payments",  
 y = "Frequency")



8. What is the homeownership might be a reason for loan default?

The summary is based on the home ownership status. Basically there are three statuses namely mortgage, rent and own. It seems that the customers who have mortgage in their home ownership have more loan default value.

loan\_data %>%  
group\_by(homeownership) %>%  
summarise(n\_customers = n(),  
customers\_default = sum(loan\_default == 'yes'),  
default\_percent = 100 \* mean(loan\_default == 'yes'))

## # A tibble: 3 × 4  
## homeownership n\_customers customers\_default default\_percent  
## <fct> <int> <int> <dbl>  
## 1 mortgage 1937 628 32.4  
## 2 rent 1666 713 42.8  
## 3 own 507 189 37.3

Predictive Modeling

Logistic Regression

The first predictive model performed is Logistice Regression. The data is split into a training and testing set. First step is a seed is set for reproducibility.

set.seed(123)   
splitIndex <- createDataPartition(loan\_data$loan\_default, p = 0.7, list = FALSE)  
train\_data <- loan\_data[splitIndex, ]  
test\_data <- loan\_data[-splitIndex, ]

Next a logistic regression model is fit using the glm() function.

model <- glm(loan\_default ~ loan\_amount + installment + interest\_rate + loan\_purpose + application\_type + term + homeownership + annual\_income + current\_job\_years + debt\_to\_income + total\_credit\_lines + years\_credit\_history + missed\_payment\_2\_yr + history\_bankruptcy + history\_tax\_liens, data = train\_data, family = binomial)

The summary and coefficient of the fitted model is calculated.

summary(model)

##   
## Call:  
## glm(formula = loan\_default ~ loan\_amount + installment + interest\_rate +   
## loan\_purpose + application\_type + term + homeownership +   
## annual\_income + current\_job\_years + debt\_to\_income + total\_credit\_lines +   
## years\_credit\_history + missed\_payment\_2\_yr + history\_bankruptcy +   
## history\_tax\_liens, family = binomial, data = train\_data)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.059e+01 9.086e-01 11.655 < 2e-16 \*\*\*  
## loan\_amount 1.024e-03 6.105e-05 16.780 < 2e-16 \*\*\*  
## installment -3.247e-02 1.909e-03 -17.007 < 2e-16 \*\*\*  
## interest\_rate -7.187e-01 3.787e-02 -18.979 < 2e-16 \*\*\*  
## loan\_purposecredit\_card -1.201e+00 2.117e-01 -5.671 1.42e-08 \*\*\*  
## loan\_purposemedical -1.861e+00 2.437e-01 -7.637 2.22e-14 \*\*\*  
## loan\_purposesmall\_business 2.887e-02 2.216e-01 0.130 0.8963   
## loan\_purposehome\_improvement -9.482e-02 2.539e-01 -0.373 0.7088   
## application\_typejoint -5.170e-01 2.308e-01 -2.240 0.0251 \*   
## termfive\_year -7.053e+00 3.805e-01 -18.536 < 2e-16 \*\*\*  
## homeownershiprent -8.033e-01 1.738e-01 -4.622 3.81e-06 \*\*\*  
## homeownershipown -4.307e-01 2.443e-01 -1.763 0.0779 .   
## annual\_income 3.547e-06 2.478e-06 1.432 0.1522   
## current\_job\_years -1.045e-02 2.114e-02 -0.494 0.6212   
## debt\_to\_income -5.520e-03 4.299e-03 -1.284 0.1991   
## total\_credit\_lines 9.286e-03 6.710e-03 1.384 0.1664   
## years\_credit\_history 4.615e-03 1.170e-02 0.395 0.6932   
## missed\_payment\_2\_yrno 4.324e-01 2.313e-01 1.870 0.0615 .   
## history\_bankruptcyno -6.111e-02 2.252e-01 -0.271 0.7861   
## history\_tax\_liensno 5.511e-01 6.427e-01 0.857 0.3912   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 3798.5 on 2876 degrees of freedom  
## Residual deviance: 1192.0 on 2857 degrees of freedom  
## AIC: 1232  
##   
## Number of Fisher Scoring iterations: 7

coef(model)

## (Intercept) loan\_amount   
## 1.058905e+01 1.024375e-03   
## installment interest\_rate   
## -3.247401e-02 -7.187219e-01   
## loan\_purposecredit\_card loan\_purposemedical   
## -1.200690e+00 -1.861342e+00   
## loan\_purposesmall\_business loan\_purposehome\_improvement   
## 2.887050e-02 -9.481575e-02   
## application\_typejoint termfive\_year   
## -5.170365e-01 -7.053131e+00   
## homeownershiprent homeownershipown   
## -8.032520e-01 -4.306850e-01   
## annual\_income current\_job\_years   
## 3.547358e-06 -1.044797e-02   
## debt\_to\_income total\_credit\_lines   
## -5.520196e-03 9.285867e-03   
## years\_credit\_history missed\_payment\_2\_yrno   
## 4.615293e-03 4.324447e-01   
## history\_bankruptcyno history\_tax\_liensno   
## -6.110509e-02 5.511007e-01

Then predictions are made on the test data.

predictions <- predict(model, newdata = test\_data, type = "response")

Next the predictions are made on the test data and then converted into binary outcomes whether 0 or 1 based on a threshold like 0.5, 1, etc,.

predicted\_classes <- ifelse(predictions > 0.5, 1, 0)

The model’s performance is then evaluated by creating a confusion matrix and the accuracy is calculated.

confusion\_matrix <- table(predicted\_classes, test\_data$loan\_default)  
accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
print(confusion\_matrix)

##   
## predicted\_classes yes no  
## 0 406 47  
## 1 53 727

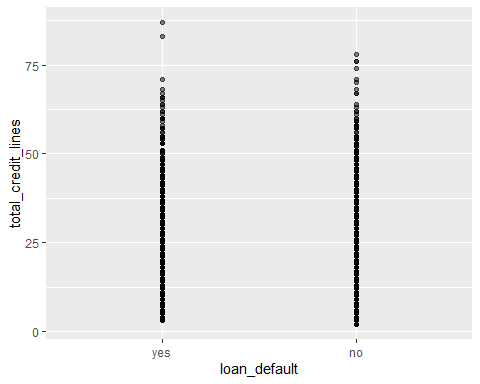
cat("Accuracy:", accuracy, "\n")

## Accuracy: 0.918897

A scatter plot is plotted to show the predicted model with x axis as loan default and y axis as total credit lines. So from the plot we can observe that the yes factor is more based on the credit lines. So this is a factor which is affecting the loan default status.

ggplot(loan\_data, aes(x = loan\_default, y = total\_credit\_lines)) +   
 geom\_point(alpha=.5) +  
 stat\_smooth(method="glm", se=FALSE, method.args = list(family=binomial))

## `geom\_smooth()` using formula = 'y ~ x'



Another logistic regression model is plotted by keeping the variables such as loan amount, interest rate, annual income, current job years, installment and loan purpose by keeping the loan default as the target variable.

model\_1 <- glm(loan\_default ~ loan\_amount + interest\_rate + annual\_income + current\_job\_years + installment + loan\_purpose,  
 family = binomial, data = train\_data)

Next the summary and coefficient of the above model is calculated.

summary(model\_1)

##   
## Call:  
## glm(formula = loan\_default ~ loan\_amount + interest\_rate + annual\_income +   
## current\_job\_years + installment + loan\_purpose, family = binomial,   
## data = train\_data)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 9.329e+00 4.013e-01 23.247 < 2e-16 \*\*\*  
## loan\_amount 9.436e-05 1.839e-05 5.132 2.86e-07 \*\*\*  
## interest\_rate -6.886e-01 2.822e-02 -24.404 < 2e-16 \*\*\*  
## annual\_income 8.596e-06 1.767e-06 4.865 1.14e-06 \*\*\*  
## current\_job\_years -1.676e-02 1.604e-02 -1.045 0.296   
## installment -4.232e-03 6.305e-04 -6.712 1.92e-11 \*\*\*  
## loan\_purposecredit\_card -1.244e+00 1.645e-01 -7.564 3.92e-14 \*\*\*  
## loan\_purposemedical -1.508e+00 1.844e-01 -8.179 2.86e-16 \*\*\*  
## loan\_purposesmall\_business -5.971e-02 1.703e-01 -0.351 0.726   
## loan\_purposehome\_improvement -1.519e-01 2.014e-01 -0.754 0.451   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 3798.5 on 2876 degrees of freedom  
## Residual deviance: 1869.0 on 2867 degrees of freedom  
## AIC: 1889  
##   
## Number of Fisher Scoring iterations: 6

coef(model\_1)

## (Intercept) loan\_amount   
## 9.329328e+00 9.435995e-05   
## interest\_rate annual\_income   
## -6.885525e-01 8.596277e-06   
## current\_job\_years installment   
## -1.675816e-02 -4.231842e-03   
## loan\_purposecredit\_card loan\_purposemedical   
## -1.244098e+00 -1.508496e+00   
## loan\_purposesmall\_business loan\_purposehome\_improvement   
## -5.971401e-02 -1.518603e-01

Random Forest

The first step is to create a recipe with the target variable as loan default.

recipe <- recipes::recipe(loan\_default ~ loan\_amount + installment + interest\_rate + loan\_purpose +   
 application\_type + term + homeownership + annual\_income +   
 current\_job\_years + debt\_to\_income + total\_credit\_lines +   
 years\_credit\_history + missed\_payment\_2\_yr + history\_bankruptcy +   
 history\_tax\_liens, data = train\_data)

Then create a random forest model specification.

rf\_spec <- parsnip::rand\_forest(mode = "classification", engine = "ranger")

Next step is to split the training data into 5 folds for 5-fold cross validation using vfold\_cv

resamples <- rsample::vfold\_cv(train\_data, v = 5)

Then create a grid of hyperparameters to tune.

param\_grid <- expand.grid(  
 mtry = c(3, 5, 7),   
 min\_n = c(5, 10, 15)   
)

Setup a workflow for the above hyperparameter tuning

loan\_workflow <- workflow()  
loan\_workflow <- add\_recipe(loan\_workflow, recipe)  
loan\_workflow <- add\_model(loan\_workflow, rf\_spec)

Then perform hyperparameter tuning

tuned\_workflow <- tune\_grid(  
 loan\_workflow,  
 resamples = resamples,  
 grid = param\_grid  
)

## Warning: No tuning parameters have been detected, performance will be evaluated  
## using the resamples with no tuning. Did you want to [tune()] parameters?

Fit the model with the best hyperparameters

final\_model <- fit(loan\_workflow, data = train\_data)

Perform predictions on the test data

test\_predictions <- predict(final\_model, new\_data = test\_data)

Final step is to evaluate model performance on the test data by plotting an ROC curve and calculating the area under the ROC curve on the test data

roc <- pROC::roc(test\_data$loan\_default, as.numeric(test\_predictions$.pred\_class))

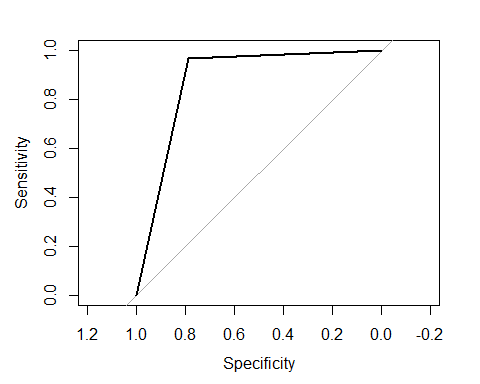
## Setting levels: control = yes, case = no

## Setting direction: controls < cases

auc <- pROC::auc(roc)

The roc curve is plotted

pROC::plot.roc(roc)



The accuracy is obtained and the value is AUC: 0.9046841.

cat("AUC:", auc, "\n")

## AUC: 0.8764503

Linear Discriminant Analysis

First step is to fit the linear discriminant analysis model using the lda() function

lda\_model <- lda(loan\_default ~ loan\_amount + installment + interest\_rate +   
 loan\_purpose + application\_type + term + homeownership +   
 annual\_income + current\_job\_years + debt\_to\_income +   
 total\_credit\_lines + years\_credit\_history +   
 missed\_payment\_2\_yr + history\_bankruptcy + history\_tax\_liens, data = train\_data)

Next is to make predictions on the test data

test\_predictions <- predict(lda\_model, newdata = test\_data)

Final step is to calculate the ROC curve and the accuracy.

roc <- pROC::roc(test\_data$loan\_default, as.numeric(test\_predictions$class))

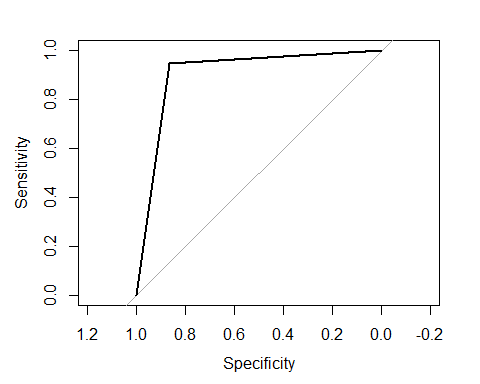
## Setting levels: control = yes, case = no

## Setting direction: controls < cases

auc <- pROC::auc(roc)

The ROC curve is ploted

pROC::plot.roc(roc)



The accuracy is obtained and the value is AUC: 0.9046841

cat("AUC:", auc, "\n")

## AUC: 0.9046841

##Error calculation

accuracy <- 0.918897  
error <- 1 - accuracy  
cat("Error:", error, "\n")

## Error: 0.081103

accuracy <- 0.9046841  
error <- 1 - accuracy  
cat("Error:", error, "\n")

## Error: 0.0953159

Conclusion

A national bank is facing loan defaults in large amount in recent years. This has led to a substantial financial loss to the bank. So to improve this the data analysis is made to identify the factors that contribute to this effect. Few machine learning models concepts to predict the cause and how to overcome it. The main research objectives were to determine the following: What variables are linked to loan default? and Is it possible to develop a predictive algorithm that would identify high-risk applicants and reduce financial losses?

The significant findings during the Exploratory Data Analysis (EDA) stage that are critical to the company are, \* Applicants with lower yearly salaries and larger debt-to-income ratios are more likely to experience loan defaults. \* The loan’s purpose and the kind of application (joint or individual) seem to have an impact on the loan default rate. \* Loan defaults are greatly impacted by credit history, as demonstrated by years of credit history and late payments. \* The home ownership also plays a role in the loan default status as the customers with mortgage tend to have more yes factor.

By gaining a better understanding of the risk variables linked to loan defaults, the firm may make more educated lending decisions and minimize losses.

Based on the predictive modeling, a Logistic Regression seems the most effective categorization model. The model is predicted using a test data to tell how well it would perform in the future with the accuracy of 0.918897 or 91% which is good. The Random Forest model is performed With the test data, the model obtained an Area Under the ROC Curve (AUC) of 0.9046841 or 90% which is also not bad. This suggests a high degree of capacity to distinguish between non-defaulting and defaulting loans. The estimated expected error of the model on future data is 0.081103, indicating that it may prove to be a useful instrument in identifying borrowers who pose a high risk.

In summary, by identifying risk variables and developing a predictive model, the analysis helps the organization manage its major challenge of lowering loan defaults. With an AUC of 0.918897, the Logistic Regression model is a good fit for forecasting loan defaults and can make a substantial contribution.

References

[1] r-bloggers.com, ‘ Linear Discriminant Analysis in R’, [Online] Available: <https://www.r-bloggers.com/2021/05/linear-discriminant-analysis-in-r/> [ Accessed on 10/27/2023].

[2] projectpro.io, ‘How to get Classification Accuracy in R’, [Online] Available: <https://www.projectpro.io/recipes/get-classification-accuracy> [ Accessed on 10/27/2023].

[3] Referred class notes.