BUAN4310 Group Project 1

Shayla Nguyen

2024-10-25

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(ROSE)

## Loaded ROSE 0.0-4

library(pROC)

## Type 'citation("pROC")' for a citation.

##   
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':  
##   
## cov, smooth, var

library(readr)  
  
# Load the credit31 data  
credit31 <- read\_csv("credit\_31.csv")

## New names:  
## • `` -> `...1`

## Rows: 30000 Columns: 68  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (12): NAME\_CONTRACT\_TYPE, CODE\_GENDER, FLAG\_OWN\_CAR, FLAG\_OWN\_REALTY, NA...  
## dbl (56): ...1, SK\_ID\_CURR, TARGET, CNT\_CHILDREN, AMT\_INCOME\_TOTAL, AMT\_CRED...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# Add new fields into data frame to improve model accuracy  
credit31$Income\_Credit\_Ratio <- credit31$AMT\_INCOME\_TOTAL / credit31$AMT\_CREDIT  
credit31$Annuity\_Income\_Ratio <- credit31$AMT\_ANNUITY / credit31$AMT\_INCOME\_TOTAL  
credit31$Credit\_As\_Percentage <- credit31$AMT\_CREDIT / credit31$AMT\_INCOME\_TOTAL  
credit31$Percent\_Days\_Employed <- credit31$DAYS\_EMPLOYED / credit31$DAYS\_BIRTH  
credit31$Income\_Per\_Person <- credit31$AMT\_INCOME\_TOTAL / credit31$CNT\_FAM\_MEMBERS  
  
# Remove XNA from CODE\_GENDER variable and convert to factor  
credit31 <- credit31[credit31$CODE\_GENDER != "XNA", ]  
credit31$CODE\_GENDER <- factor(credit31$CODE\_GENDER)  
  
# Explore data  
names(credit31)

## [1] "...1" "SK\_ID\_CURR"   
## [3] "TARGET" "NAME\_CONTRACT\_TYPE"   
## [5] "CODE\_GENDER" "FLAG\_OWN\_CAR"   
## [7] "FLAG\_OWN\_REALTY" "CNT\_CHILDREN"   
## [9] "AMT\_INCOME\_TOTAL" "AMT\_CREDIT"   
## [11] "AMT\_ANNUITY" "AMT\_GOODS\_PRICE"   
## [13] "NAME\_TYPE\_SUITE" "NAME\_INCOME\_TYPE"   
## [15] "NAME\_EDUCATION\_TYPE" "NAME\_FAMILY\_STATUS"   
## [17] "NAME\_HOUSING\_TYPE" "DAYS\_BIRTH"   
## [19] "DAYS\_EMPLOYED" "DAYS\_REGISTRATION"   
## [21] "DAYS\_ID\_PUBLISH" "OWN\_CAR\_AGE"   
## [23] "FLAG\_MOBIL" "FLAG\_EMP\_PHONE"   
## [25] "FLAG\_WORK\_PHONE" "FLAG\_CONT\_MOBILE"   
## [27] "FLAG\_PHONE" "FLAG\_EMAIL"   
## [29] "OCCUPATION\_TYPE" "CNT\_FAM\_MEMBERS"   
## [31] "REGION\_RATING\_CLIENT" "REGION\_RATING\_CLIENT\_W\_CITY"  
## [33] "WEEKDAY\_APPR\_PROCESS\_START" "HOUR\_APPR\_PROCESS\_START"   
## [35] "REG\_REGION\_NOT\_LIVE\_REGION" "REG\_REGION\_NOT\_WORK\_REGION"   
## [37] "LIVE\_REGION\_NOT\_WORK\_REGION" "REG\_CITY\_NOT\_LIVE\_CITY"   
## [39] "REG\_CITY\_NOT\_WORK\_CITY" "LIVE\_CITY\_NOT\_WORK\_CITY"   
## [41] "ORGANIZATION\_TYPE" "DAYS\_LAST\_PHONE\_CHANGE"   
## [43] "FLAG\_DOCUMENT\_2" "FLAG\_DOCUMENT\_3"   
## [45] "FLAG\_DOCUMENT\_4" "FLAG\_DOCUMENT\_5"   
## [47] "FLAG\_DOCUMENT\_6" "FLAG\_DOCUMENT\_7"   
## [49] "FLAG\_DOCUMENT\_8" "FLAG\_DOCUMENT\_9"   
## [51] "FLAG\_DOCUMENT\_10" "FLAG\_DOCUMENT\_11"   
## [53] "FLAG\_DOCUMENT\_12" "FLAG\_DOCUMENT\_13"   
## [55] "FLAG\_DOCUMENT\_14" "FLAG\_DOCUMENT\_15"   
## [57] "FLAG\_DOCUMENT\_16" "FLAG\_DOCUMENT\_17"   
## [59] "FLAG\_DOCUMENT\_18" "FLAG\_DOCUMENT\_19"   
## [61] "FLAG\_DOCUMENT\_20" "FLAG\_DOCUMENT\_21"   
## [63] "AMT\_REQ\_CREDIT\_BUREAU\_HOUR" "AMT\_REQ\_CREDIT\_BUREAU\_DAY"   
## [65] "AMT\_REQ\_CREDIT\_BUREAU\_WEEK" "AMT\_REQ\_CREDIT\_BUREAU\_MON"   
## [67] "AMT\_REQ\_CREDIT\_BUREAU\_QRT" "AMT\_REQ\_CREDIT\_BUREAU\_YEAR"   
## [69] "Income\_Credit\_Ratio" "Annuity\_Income\_Ratio"   
## [71] "Credit\_As\_Percentage" "Percent\_Days\_Employed"   
## [73] "Income\_Per\_Person"

str(credit31)

## tibble [30,000 × 73] (S3: tbl\_df/tbl/data.frame)  
## $ ...1 : num [1:30000] 284834 161354 132607 199508 99768 ...  
## $ SK\_ID\_CURR : num [1:30000] 429876 287055 253803 331291 215821 ...  
## $ TARGET : num [1:30000] 0 0 1 0 0 0 0 0 0 0 ...  
## $ NAME\_CONTRACT\_TYPE : chr [1:30000] "Cash loans" "Cash loans" "Cash loans" "Cash loans" ...  
## $ CODE\_GENDER : Factor w/ 2 levels "F","M": 1 2 2 1 1 2 1 1 1 1 ...  
## $ FLAG\_OWN\_CAR : chr [1:30000] "N" "Y" "N" "Y" ...  
## $ FLAG\_OWN\_REALTY : chr [1:30000] "N" "Y" "Y" "N" ...  
## $ CNT\_CHILDREN : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ AMT\_INCOME\_TOTAL : num [1:30000] 103500 81000 112500 225000 67500 ...  
## $ AMT\_CREDIT : num [1:30000] 675000 808650 423000 646920 135000 ...  
## $ AMT\_ANNUITY : num [1:30000] 21776 26217 28269 25065 16150 ...  
## $ AMT\_GOODS\_PRICE : num [1:30000] 675000 675000 423000 540000 135000 ...  
## $ NAME\_TYPE\_SUITE : chr [1:30000] "Family" "Family" "Spouse, partner" "Unaccompanied" ...  
## $ NAME\_INCOME\_TYPE : chr [1:30000] "State servant" "Working" "Working" "Working" ...  
## $ NAME\_EDUCATION\_TYPE : chr [1:30000] "Higher education" "Secondary / secondary special" "Secondary / secondary special" "Higher education" ...  
## $ NAME\_FAMILY\_STATUS : chr [1:30000] "Separated" "Married" "Married" "Married" ...  
## $ NAME\_HOUSING\_TYPE : chr [1:30000] "House / apartment" "House / apartment" "House / apartment" "House / apartment" ...  
## $ DAYS\_BIRTH : num [1:30000] -14211 -17884 -14629 -12894 -16825 ...  
## $ DAYS\_EMPLOYED : num [1:30000] -2875 -2192 -984 -1994 -1087 ...  
## $ DAYS\_REGISTRATION : num [1:30000] -4018 -7442 -741 -1278 -8220 ...  
## $ DAYS\_ID\_PUBLISH : num [1:30000] -4693 -1428 -1747 -3897 -367 ...  
## $ OWN\_CAR\_AGE : num [1:30000] NA 1 NA 1 18 NA NA NA NA NA ...  
## $ FLAG\_MOBIL : num [1:30000] 1 1 1 1 1 1 1 1 1 1 ...  
## $ FLAG\_EMP\_PHONE : num [1:30000] 1 1 1 1 1 1 1 0 1 0 ...  
## $ FLAG\_WORK\_PHONE : num [1:30000] 1 1 0 1 0 0 0 0 0 0 ...  
## $ FLAG\_CONT\_MOBILE : num [1:30000] 1 1 1 1 1 1 1 1 1 1 ...  
## $ FLAG\_PHONE : num [1:30000] 0 1 0 1 0 0 0 0 0 0 ...  
## $ FLAG\_EMAIL : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ OCCUPATION\_TYPE : chr [1:30000] "Laborers" "Security staff" "Laborers" "Laborers" ...  
## $ CNT\_FAM\_MEMBERS : num [1:30000] 1 2 2 2 2 1 2 2 1 2 ...  
## $ REGION\_RATING\_CLIENT : num [1:30000] 2 2 2 2 2 2 2 2 2 3 ...  
## $ REGION\_RATING\_CLIENT\_W\_CITY: num [1:30000] 2 2 2 2 2 2 2 2 2 3 ...  
## $ WEEKDAY\_APPR\_PROCESS\_START : chr [1:30000] "SUNDAY" "TUESDAY" "MONDAY" "SATURDAY" ...  
## $ HOUR\_APPR\_PROCESS\_START : num [1:30000] 11 11 8 10 9 12 14 11 11 11 ...  
## $ REG\_REGION\_NOT\_LIVE\_REGION : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ REG\_REGION\_NOT\_WORK\_REGION : num [1:30000] 0 0 1 0 0 0 0 0 0 0 ...  
## $ LIVE\_REGION\_NOT\_WORK\_REGION: num [1:30000] 0 0 1 0 0 0 0 0 0 0 ...  
## $ REG\_CITY\_NOT\_LIVE\_CITY : num [1:30000] 0 0 0 0 0 1 0 0 0 0 ...  
## $ REG\_CITY\_NOT\_WORK\_CITY : num [1:30000] 0 1 1 0 1 1 1 0 0 0 ...  
## $ LIVE\_CITY\_NOT\_WORK\_CITY : num [1:30000] 0 1 1 0 1 0 1 0 0 0 ...  
## $ ORGANIZATION\_TYPE : chr [1:30000] "Postal" "Business Entity Type 3" "Industry: type 9" "Business Entity Type 1" ...  
## $ DAYS\_LAST\_PHONE\_CHANGE : num [1:30000] -1735 0 -570 -1748 -1204 ...  
## $ FLAG\_DOCUMENT\_2 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_3 : num [1:30000] 1 1 1 1 1 1 1 0 0 0 ...  
## $ FLAG\_DOCUMENT\_4 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_5 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_6 : num [1:30000] 0 0 0 0 0 0 0 0 0 1 ...  
## $ FLAG\_DOCUMENT\_7 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_8 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_9 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_10 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_11 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_12 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_13 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_14 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_15 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_16 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_17 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_18 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_19 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_20 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ FLAG\_DOCUMENT\_21 : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_HOUR : num [1:30000] 0 0 0 1 0 0 0 0 0 0 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_DAY : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_WEEK : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_MON : num [1:30000] 0 0 0 1 0 0 0 0 1 1 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_QRT : num [1:30000] 0 0 0 0 0 0 0 0 0 0 ...  
## $ AMT\_REQ\_CREDIT\_BUREAU\_YEAR : num [1:30000] 2 2 0 2 1 1 2 5 0 2 ...  
## $ Income\_Credit\_Ratio : num [1:30000] 0.153 0.1 0.266 0.348 0.5 ...  
## $ Annuity\_Income\_Ratio : num [1:30000] 0.21 0.324 0.251 0.111 0.239 ...  
## $ Credit\_As\_Percentage : num [1:30000] 6.52 9.98 3.76 2.88 2 ...  
## $ Percent\_Days\_Employed : num [1:30000] 0.2023 0.1226 0.0673 0.1546 0.0646 ...  
## $ Income\_Per\_Person : num [1:30000] 103500 40500 56250 112500 33750 ...

summary(credit31$TARGET)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 0.0000 0.0000 0.1949 0.0000 1.0000

# Convert education type to factor with levels across education   
credit31$NAME\_EDUCATION\_TYPE <- factor(credit31$NAME\_EDUCATION\_TYPE, levels = c(  
 "Secondary / secondary special",  
 "Higher education",  
 "Lower secondary",  
 "Incomplete higher",  
 "Academic degree"))  
  
# Set Target variable as factor   
credit31$TARGET <- as.factor(credit31$TARGET)  
  
# Variable list   
# Percent\_Days\_Employed, NAME\_EDUCATION\_TYPE, REGION\_RATING\_CLIENT\_W\_CITY, AMT\_GOODS\_PRICE, CODE\_GENDER, DAYS\_BIRTH, AMT\_CREDIT, AMT\_ANNUITY, DAYS\_EMPLOYED, DAYS\_REGISTRATION, DAYS\_ID\_PUBLISH, Annuity\_Income\_Ratio  
  
# Remove unused variables  
credit31 <- credit31[ , -c(1:2, 4, 6:9, 13:14, 16:17, 22:31, 33:69, 71, 73)]  
names(credit31)

## [1] "TARGET" "CODE\_GENDER"   
## [3] "AMT\_CREDIT" "AMT\_ANNUITY"   
## [5] "AMT\_GOODS\_PRICE" "NAME\_EDUCATION\_TYPE"   
## [7] "DAYS\_BIRTH" "DAYS\_EMPLOYED"   
## [9] "DAYS\_REGISTRATION" "DAYS\_ID\_PUBLISH"   
## [11] "REGION\_RATING\_CLIENT\_W\_CITY" "Annuity\_Income\_Ratio"   
## [13] "Percent\_Days\_Employed"

# Training - Validation split   
set.seed(666)  
train\_index <- sample(1:nrow(credit31), 0.7 \* nrow(credit31))  
valid\_index <- setdiff(1:nrow(credit31), train\_index)  
train\_df <- credit31[train\_index, ]  
valid\_df <- credit31[valid\_index, ]  
  
# Double check   
nrow(train\_df)

## [1] 21000

nrow(valid\_df)

## [1] 9000

head(train\_df)

## # A tibble: 6 × 13  
## TARGET CODE\_GENDER AMT\_CREDIT AMT\_ANNUITY AMT\_GOODS\_PRICE NAME\_EDUCATION\_TYPE   
## <fct> <fct> <dbl> <dbl> <dbl> <fct>   
## 1 0 M 956574 38066. 855000 Secondary / seconda…  
## 2 0 F 1633473 45050. 1363500 Secondary / seconda…  
## 3 1 F 279000 15134. 279000 Secondary / seconda…  
## 4 0 F 405000 20250 405000 Higher education   
## 5 1 M 279000 22041 279000 Secondary / seconda…  
## 6 0 F 808650 26217 675000 Secondary / seconda…  
## # ℹ 7 more variables: DAYS\_BIRTH <dbl>, DAYS\_EMPLOYED <dbl>,  
## # DAYS\_REGISTRATION <dbl>, DAYS\_ID\_PUBLISH <dbl>,  
## # REGION\_RATING\_CLIENT\_W\_CITY <dbl>, Annuity\_Income\_Ratio <dbl>,  
## # Percent\_Days\_Employed <dbl>

head(valid\_df)

## # A tibble: 6 × 13  
## TARGET CODE\_GENDER AMT\_CREDIT AMT\_ANNUITY AMT\_GOODS\_PRICE NAME\_EDUCATION\_TYPE   
## <fct> <fct> <dbl> <dbl> <dbl> <fct>   
## 1 0 M 808650 26217 675000 Secondary / seconda…  
## 2 1 M 423000 28269 423000 Secondary / seconda…  
## 3 0 M 450000 30074. 450000 Secondary / seconda…  
## 4 0 F 202500 10125 202500 Secondary / seconda…  
## 5 0 F 269550 12002. 225000 Secondary / seconda…  
## 6 0 F 1125000 36292. 1125000 Secondary / seconda…  
## # ℹ 7 more variables: DAYS\_BIRTH <dbl>, DAYS\_EMPLOYED <dbl>,  
## # DAYS\_REGISTRATION <dbl>, DAYS\_ID\_PUBLISH <dbl>,  
## # REGION\_RATING\_CLIENT\_W\_CITY <dbl>, Annuity\_Income\_Ratio <dbl>,  
## # Percent\_Days\_Employed <dbl>

str(train\_df)

## tibble [21,000 × 13] (S3: tbl\_df/tbl/data.frame)  
## $ TARGET : Factor w/ 2 levels "0","1": 1 1 2 1 2 1 1 1 1 1 ...  
## $ CODE\_GENDER : Factor w/ 2 levels "F","M": 2 1 1 1 2 1 2 2 1 2 ...  
## $ AMT\_CREDIT : num [1:21000] 956574 1633473 279000 405000 279000 ...  
## $ AMT\_ANNUITY : num [1:21000] 38066 45050 15134 20250 22041 ...  
## $ AMT\_GOODS\_PRICE : num [1:21000] 855000 1363500 279000 405000 279000 ...  
## $ NAME\_EDUCATION\_TYPE : Factor w/ 5 levels "Secondary / secondary special",..: 1 1 1 2 1 1 1 2 1 1 ...  
## $ DAYS\_BIRTH : num [1:21000] -14523 -13597 -21630 -14352 -23121 ...  
## $ DAYS\_EMPLOYED : num [1:21000] -926 -247 -14068 -4132 365243 ...  
## $ DAYS\_REGISTRATION : num [1:21000] -8452 -7709 -5517 -10 -935 ...  
## $ DAYS\_ID\_PUBLISH : num [1:21000] -4476 -4795 -5024 -2199 -4450 ...  
## $ REGION\_RATING\_CLIENT\_W\_CITY: num [1:21000] 2 3 2 2 2 2 2 2 2 2 ...  
## $ Annuity\_Income\_Ratio : num [1:21000] 0.169 0.222 0.108 0.15 0.109 ...  
## $ Percent\_Days\_Employed : num [1:21000] 0.0638 0.0182 0.6504 0.2879 -15.797 ...

str(valid\_df)

## tibble [9,000 × 13] (S3: tbl\_df/tbl/data.frame)  
## $ TARGET : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 2 1 1 1 ...  
## $ CODE\_GENDER : Factor w/ 2 levels "F","M": 2 2 2 1 1 1 1 1 1 2 ...  
## $ AMT\_CREDIT : num [1:9000] 808650 423000 450000 202500 269550 ...  
## $ AMT\_ANNUITY : num [1:9000] 26217 28269 30074 10125 12002 ...  
## $ AMT\_GOODS\_PRICE : num [1:9000] 675000 423000 450000 202500 225000 ...  
## $ NAME\_EDUCATION\_TYPE : Factor w/ 5 levels "Secondary / secondary special",..: 1 1 1 1 1 1 4 1 1 1 ...  
## $ DAYS\_BIRTH : num [1:9000] -17884 -14629 -9655 -21512 -22485 ...  
## $ DAYS\_EMPLOYED : num [1:9000] -2192 -984 -2940 -1874 365243 ...  
## $ DAYS\_REGISTRATION : num [1:9000] -7442 -741 -8153 -10778 -14544 ...  
## $ DAYS\_ID\_PUBLISH : num [1:9000] -1428 -1747 -2298 -4811 -4620 ...  
## $ REGION\_RATING\_CLIENT\_W\_CITY: num [1:9000] 2 2 2 2 3 2 2 2 1 2 ...  
## $ Annuity\_Income\_Ratio : num [1:9000] 0.324 0.251 0.122 0.113 0.133 ...  
## $ Percent\_Days\_Employed : num [1:9000] 0.1226 0.0673 0.3045 0.0871 -16.2439 ...

# Use ROSE to balance model  
train\_df\_rose <- ROSE(TARGET ~ Percent\_Days\_Employed + NAME\_EDUCATION\_TYPE + REGION\_RATING\_CLIENT\_W\_CITY + AMT\_GOODS\_PRICE + CODE\_GENDER + DAYS\_BIRTH + AMT\_CREDIT + AMT\_ANNUITY + DAYS\_EMPLOYED + DAYS\_REGISTRATION + DAYS\_ID\_PUBLISH + Annuity\_Income\_Ratio,  
 data = train\_df, seed = 666)$data  
  
table(train\_df\_rose$TARGET)

##   
## 0 1   
## 10339 10642

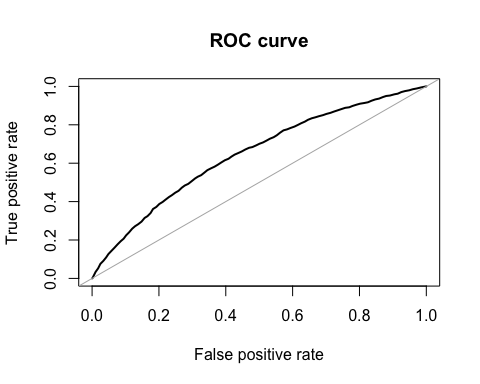
# Normalization algorithm  
train\_norm <- train\_df\_rose  
valid\_norm <- valid\_df  
  
norm\_values <- preProcess(train\_df\_rose[, -c(1)],  
 method = c("center", "scale"))  
train\_norm[, -c(1)] <- predict(norm\_values,  
 train\_df\_rose[, -c(1)])  
  
# Apply to validation set   
valid\_norm[, -c(1)] <- predict(norm\_values,  
 valid\_df[, -c(1)])  
  
# Drop missing values  
library(tidyr)  
valid\_norm <- drop\_na(valid\_norm)  
  
# Train logistic regression model  
logistic\_model <- glm(TARGET ~ Percent\_Days\_Employed + NAME\_EDUCATION\_TYPE + REGION\_RATING\_CLIENT\_W\_CITY + AMT\_GOODS\_PRICE + CODE\_GENDER + DAYS\_BIRTH + AMT\_CREDIT + AMT\_ANNUITY + DAYS\_EMPLOYED + DAYS\_REGISTRATION + DAYS\_ID\_PUBLISH + Annuity\_Income\_Ratio,  
 data = train\_norm, family = binomial)  
  
# Prediction on training set  
logistic\_pred\_train <- predict(logistic\_model, newdata = train\_norm, type = "response")  
logistic\_pred\_train\_class <- ifelse(logistic\_pred\_train > 0.5, 1, 0)  
  
# Prediction on validation set   
logistic\_pred\_valid <- predict(logistic\_model, newdata = valid\_norm, type = "response")  
logistic\_pred\_valid\_class <- ifelse(logistic\_pred\_valid > 0.5, 1, 0)  
  
# Confusion matrix on training set   
confusionMatrix(as.factor(logistic\_pred\_train\_class), as.factor(train\_norm$TARGET), positive = "1")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 5773 3900  
## 1 4566 6742  
##   
## Accuracy : 0.5965   
## 95% CI : (0.5898, 0.6031)  
## No Information Rate : 0.5072   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.1921   
##   
## Mcnemar's Test P-Value : 4.923e-13   
##   
## Sensitivity : 0.6335   
## Specificity : 0.5584   
## Pos Pred Value : 0.5962   
## Neg Pred Value : 0.5968   
## Prevalence : 0.5072   
## Detection Rate : 0.3213   
## Detection Prevalence : 0.5390   
## Balanced Accuracy : 0.5959   
##   
## 'Positive' Class : 1   
##

# Confusion matrix on validation set   
confusionMatrix(as.factor(logistic\_pred\_valid\_class), as.factor(valid\_norm$TARGET), positive = "1")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 4027 600  
## 1 3198 1165  
##   
## Accuracy : 0.5775   
## 95% CI : (0.5672, 0.5878)  
## No Information Rate : 0.8037   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.1397   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.6601   
## Specificity : 0.5574   
## Pos Pred Value : 0.2670   
## Neg Pred Value : 0.8703   
## Prevalence : 0.1963   
## Detection Rate : 0.1296   
## Detection Prevalence : 0.4853   
## Balanced Accuracy : 0.6087   
##   
## 'Positive' Class : 1   
##

# Model Evaluation  
library(ROSE)  
ROSE::roc.curve(valid\_norm$TARGET, logistic\_pred\_valid)



## Area under the curve (AUC): 0.646

# Load new customer data  
new\_customers <- read\_csv("credit\_test\_31.csv")

## New names:  
## Rows: 5 Columns: 67  
## ── Column specification  
## ──────────────────────────────────────────────────────── Delimiter: "," chr  
## (12): NAME\_CONTRACT\_TYPE, CODE\_GENDER, FLAG\_OWN\_CAR, FLAG\_OWN\_REALTY, NA... dbl  
## (55): ...1, SK\_ID\_CURR, CNT\_CHILDREN, AMT\_INCOME\_TOTAL, AMT\_CREDIT, AMT\_...  
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ  
## Specify the column types or set `show\_col\_types = FALSE` to quiet this message.  
## • `` -> `...1`

# Preprocess new customer data  
new\_customers$Income\_Credit\_Ratio <- new\_customers$AMT\_INCOME\_TOTAL / new\_customers$AMT\_CREDIT  
new\_customers$Annuity\_Income\_Ratio <- new\_customers$AMT\_ANNUITY / new\_customers$AMT\_INCOME\_TOTAL  
new\_customers$Credit\_As\_Percentage <- new\_customers$AMT\_CREDIT / new\_customers$AMT\_INCOME\_TOTAL  
new\_customers$Percent\_Days\_Employed <- new\_customers$DAYS\_EMPLOYED / new\_customers$DAYS\_BIRTH  
new\_customers$Income\_Per\_Person <- new\_customers$AMT\_INCOME\_TOTAL / new\_customers$CNT\_FAM\_MEMBERS  
  
# Remove XNA from CODE\_GENDER variable and convert to factor  
new\_customers <- new\_customers[new\_customers$CODE\_GENDER != "XNA", ]  
new\_customers$CODE\_GENDER <- factor(new\_customers$CODE\_GENDER)  
  
# Convert education type to factor with levels across education   
new\_customers$NAME\_EDUCATION\_TYPE <- factor(new\_customers$NAME\_EDUCATION\_TYPE, levels = c(  
 "Secondary / secondary special",  
 "Higher education",  
 "Lower secondary",  
 "Incomplete higher",  
 "Academic degree"))  
  
# Normalize new customer data using the same scaling as the training data  
new\_customers\_norm <- predict(norm\_values, new\_customers[, -c(1)])  
  
# Predict risk of new customers  
new\_customer\_predictions <- predict(logistic\_model, newdata = new\_customers\_norm, type = "response")  
new\_customer\_predictions\_class <- ifelse(new\_customer\_predictions > 0.5, 1, 0)  
  
# Display predictions for new customers  
new\_customer\_results <- data.frame(new\_customers, Predicted\_Risk = new\_customer\_predictions\_class)  
head(new\_customer\_results)

## ...1 SK\_ID\_CURR NAME\_CONTRACT\_TYPE CODE\_GENDER FLAG\_OWN\_CAR FLAG\_OWN\_REALTY  
## 1 1 402254 Cash loans M N Y  
## 2 2 440463 Cash loans F N Y  
## 3 3 242185 Cash loans F N N  
## 4 4 235118 Cash loans M Y Y  
## 5 5 407346 Cash loans F N Y  
## CNT\_CHILDREN AMT\_INCOME\_TOTAL AMT\_CREDIT AMT\_ANNUITY AMT\_GOODS\_PRICE  
## 1 0 247500 746280 59094.0 675000  
## 2 0 112500 589500 32107.5 589500  
## 3 1 112500 272520 16803.0 225000  
## 4 0 157500 533313 37246.5 472500  
## 5 2 112500 283500 22527.0 283500  
## NAME\_TYPE\_SUITE NAME\_INCOME\_TYPE NAME\_EDUCATION\_TYPE  
## 1 Unaccompanied Working Secondary / secondary special  
## 2 Unaccompanied Commercial associate Incomplete higher  
## 3 Unaccompanied Working Secondary / secondary special  
## 4 Unaccompanied Working Higher education  
## 5 Unaccompanied Working Secondary / secondary special  
## NAME\_FAMILY\_STATUS NAME\_HOUSING\_TYPE DAYS\_BIRTH DAYS\_EMPLOYED  
## 1 Separated House / apartment -9889 -2077  
## 2 Married House / apartment -9843 -2772  
## 3 Married House / apartment -10208 -853  
## 4 Married House / apartment -21121 -3561  
## 5 Civil marriage House / apartment -19354 -5103  
## DAYS\_REGISTRATION DAYS\_ID\_PUBLISH OWN\_CAR\_AGE FLAG\_MOBIL FLAG\_EMP\_PHONE  
## 1 -417 -1342 NA 1 1  
## 2 -524 -2523 NA 1 1  
## 3 -1893 -1946 NA 1 1  
## 4 -7328 -3506 3 1 1  
## 5 -9478 -2873 NA 1 1  
## FLAG\_WORK\_PHONE FLAG\_CONT\_MOBILE FLAG\_PHONE FLAG\_EMAIL OCCUPATION\_TYPE  
## 1 0 1 0 0 Laborers  
## 2 0 1 0 0 Cooking staff  
## 3 1 1 0 0 Sales staff  
## 4 0 1 0 1 Drivers  
## 5 0 1 0 0 Laborers  
## CNT\_FAM\_MEMBERS REGION\_RATING\_CLIENT REGION\_RATING\_CLIENT\_W\_CITY  
## 1 1 3 3  
## 2 2 2 2  
## 3 3 2 2  
## 4 2 2 2  
## 5 4 2 2  
## WEEKDAY\_APPR\_PROCESS\_START HOUR\_APPR\_PROCESS\_START REG\_REGION\_NOT\_LIVE\_REGION  
## 1 TUESDAY 9 0  
## 2 THURSDAY 12 0  
## 3 THURSDAY 13 0  
## 4 TUESDAY 17 0  
## 5 FRIDAY 13 0  
## REG\_REGION\_NOT\_WORK\_REGION LIVE\_REGION\_NOT\_WORK\_REGION REG\_CITY\_NOT\_LIVE\_CITY  
## 1 0 0 0  
## 2 0 0 1  
## 3 0 0 0  
## 4 0 0 0  
## 5 0 0 1  
## REG\_CITY\_NOT\_WORK\_CITY LIVE\_CITY\_NOT\_WORK\_CITY ORGANIZATION\_TYPE  
## 1 0 0 Other  
## 2 1 0 Business Entity Type 2  
## 3 1 1 Trade: type 3  
## 4 0 0 Business Entity Type 3  
## 5 1 0 Business Entity Type 3  
## DAYS\_LAST\_PHONE\_CHANGE FLAG\_DOCUMENT\_2 FLAG\_DOCUMENT\_3 FLAG\_DOCUMENT\_4  
## 1 -785 0 1 0  
## 2 -202 0 0 0  
## 3 -1474 0 1 0  
## 4 -618 0 1 0  
## 5 -510 0 1 0  
## FLAG\_DOCUMENT\_5 FLAG\_DOCUMENT\_6 FLAG\_DOCUMENT\_7 FLAG\_DOCUMENT\_8  
## 1 0 0 0 0  
## 2 0 0 0 1  
## 3 0 0 0 0  
## 4 0 0 0 0  
## 5 0 0 0 0  
## FLAG\_DOCUMENT\_9 FLAG\_DOCUMENT\_10 FLAG\_DOCUMENT\_11 FLAG\_DOCUMENT\_12  
## 1 0 0 0 0  
## 2 0 0 0 0  
## 3 0 0 0 0  
## 4 0 0 0 0  
## 5 0 0 0 0  
## FLAG\_DOCUMENT\_13 FLAG\_DOCUMENT\_14 FLAG\_DOCUMENT\_15 FLAG\_DOCUMENT\_16  
## 1 0 0 0 0  
## 2 0 0 0 0  
## 3 0 0 0 0  
## 4 0 0 0 0  
## 5 0 0 0 0  
## FLAG\_DOCUMENT\_17 FLAG\_DOCUMENT\_18 FLAG\_DOCUMENT\_19 FLAG\_DOCUMENT\_20  
## 1 0 0 0 0  
## 2 0 0 0 0  
## 3 0 0 0 0  
## 4 0 0 0 0  
## 5 0 0 0 0  
## FLAG\_DOCUMENT\_21 AMT\_REQ\_CREDIT\_BUREAU\_HOUR AMT\_REQ\_CREDIT\_BUREAU\_DAY  
## 1 0 0 0  
## 2 0 0 0  
## 3 0 NA NA  
## 4 0 0 0  
## 5 0 0 0  
## AMT\_REQ\_CREDIT\_BUREAU\_WEEK AMT\_REQ\_CREDIT\_BUREAU\_MON  
## 1 0 0  
## 2 0 0  
## 3 NA NA  
## 4 0 0  
## 5 0 0  
## AMT\_REQ\_CREDIT\_BUREAU\_QRT AMT\_REQ\_CREDIT\_BUREAU\_YEAR Income\_Credit\_Ratio  
## 1 1 3 0.3316450  
## 2 0 2 0.1908397  
## 3 NA NA 0.4128137  
## 4 0 0 0.2953238  
## 5 0 3 0.3968254  
## Annuity\_Income\_Ratio Credit\_As\_Percentage Percent\_Days\_Employed  
## 1 0.2387636 3.015273 0.21003135  
## 2 0.2854000 5.240000 0.28162146  
## 3 0.1493600 2.422400 0.08356191  
## 4 0.2364857 3.386114 0.16859997  
## 5 0.2002400 2.520000 0.26366643  
## Income\_Per\_Person Predicted\_Risk  
## 1 247500 1  
## 2 56250 1  
## 3 37500 1  
## 4 78750 0  
## 5 28125 0

new\_customer\_results <- data.frame(  
 Customer\_ID = new\_customers$SK\_ID\_CURR, # Replace with the actual identifier column if different  
 Prediction = new\_customer\_predictions\_class,  
 Probability = new\_customer\_predictions  
)  
  
# Format and display top results for clarity  
head(new\_customer\_results[order(-new\_customer\_results$Probability), ]) # Top predictions with high probability

## Customer\_ID Prediction Probability  
## 1 402254 1 0.7369819  
## 3 242185 1 0.5901293  
## 2 440463 1 0.5713890  
## 5 407346 0 0.4692277  
## 4 235118 0 0.4174471

# Calculate accuracy for training set  
train\_accuracy <- mean(logistic\_pred\_train\_class == train\_norm$TARGET) \* 100  
  
# Calculate accuracy for validation set  
valid\_accuracy <- mean(logistic\_pred\_valid\_class == valid\_norm$TARGET) \* 100  
  
# Print accuracy results  
cat("Training Accuracy:", round(train\_accuracy, 2), "%\n")

## Training Accuracy: 59.65 %

cat("Validation Accuracy:", round(valid\_accuracy, 2), "%\n")

## Validation Accuracy: 57.75 %

# Load required libraries  
library(caret)  
  
# Confusion matrix for training set  
train\_conf\_matrix <- confusionMatrix(as.factor(logistic\_pred\_train\_class), as.factor(train\_norm$TARGET), positive = "1")  
cat("Training Confusion Matrix:\n")

## Training Confusion Matrix:

print(train\_conf\_matrix)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 5773 3900  
## 1 4566 6742  
##   
## Accuracy : 0.5965   
## 95% CI : (0.5898, 0.6031)  
## No Information Rate : 0.5072   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.1921   
##   
## Mcnemar's Test P-Value : 4.923e-13   
##   
## Sensitivity : 0.6335   
## Specificity : 0.5584   
## Pos Pred Value : 0.5962   
## Neg Pred Value : 0.5968   
## Prevalence : 0.5072   
## Detection Rate : 0.3213   
## Detection Prevalence : 0.5390   
## Balanced Accuracy : 0.5959   
##   
## 'Positive' Class : 1   
##

# Calculate F1 score for training set  
train\_precision <- train\_conf\_matrix$byClass["Pos Pred Value"]  
train\_recall <- train\_conf\_matrix$byClass["Sensitivity"]  
train\_f1 <- 2 \* ((train\_precision \* train\_recall) / (train\_precision + train\_recall))  
cat("Training F1 Score:", round(train\_f1, 2), "\n")

## Training F1 Score: 0.61

# Confusion matrix for validation set  
valid\_conf\_matrix <- confusionMatrix(as.factor(logistic\_pred\_valid\_class), as.factor(valid\_norm$TARGET), positive = "1")  
cat("\nValidation Confusion Matrix:\n")

##   
## Validation Confusion Matrix:

print(valid\_conf\_matrix)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 4027 600  
## 1 3198 1165  
##   
## Accuracy : 0.5775   
## 95% CI : (0.5672, 0.5878)  
## No Information Rate : 0.8037   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.1397   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.6601   
## Specificity : 0.5574   
## Pos Pred Value : 0.2670   
## Neg Pred Value : 0.8703   
## Prevalence : 0.1963   
## Detection Rate : 0.1296   
## Detection Prevalence : 0.4853   
## Balanced Accuracy : 0.6087   
##   
## 'Positive' Class : 1   
##

# Calculate F1 score for validation set  
valid\_precision <- valid\_conf\_matrix$byClass["Pos Pred Value"]  
valid\_recall <- valid\_conf\_matrix$byClass["Sensitivity"]  
valid\_f1 <- 2 \* ((valid\_precision \* valid\_recall) / (valid\_precision + valid\_recall))  
cat("Validation F1 Score:", round(valid\_f1, 2), "\n")

## Validation F1 Score: 0.38