**Summer 2023 R Project**

Patrick Doyne

*# Suppress dplyr summarise grouping warning messages*  
**options**(dplyr.summarise.inform = FALSE)  
  
***## Add R libraries here***  
**library**(tidyverse)  
**library**(tidymodels)  
**library**(vip)  
**library**(dplyr)  
**library**(discrim)  
**library**(klaR)  
**library**(tidyverse)  
  
*# Load data*  
loans\_df <- **read\_rds**("C:/Users/doyne/Downloads/loan\_data(1).rds")  
  
*#Split data*  
**set.seed**(345)  
loan\_split <- **initial\_split**(loans\_df, prop = 0.75,  
strata = loan\_default)  
*# Training set*  
loan\_training <- loan\_split **%>%** **training**()  
*# Testing Set*  
loan\_test <- loan\_split **%>%** **testing**()  
  
*#Creating a recipe*  
loan\_recipe <- **recipe**(loan\_default **~** ., data = loan\_training) **%>%**  
**step\_YeoJohnson**(**all\_numeric**(), **-all\_outcomes**()) **%>%**  
**step\_normalize**(**all\_numeric**(), **-all\_outcomes**()) **%>%**  
**step\_dummy**(**all\_nominal**(), **-all\_outcomes**())  
*#Train data quick view*  
loan\_recipe **%>%**  
**prep**() **%>%**  
**bake**(new\_data = loan\_training)

## # A tibble: 3,082 × 20  
## loan\_amount installment interest\_rate annual\_income current\_job\_years  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1.16 1.42 -0.551 1.89 1.10   
## 2 -1.62 -1.70 -0.345 0.189 -0.377  
## 3 -1.23 -1.22 -0.838 0.133 -0.377  
## 4 0.0145 0.0572 0.0439 -0.828 1.10   
## 5 1.64 0.990 -1.06 2.22 1.10   
## 6 -1.14 -1.10 0.0439 -1.36 1.10   
## 7 -0.369 -0.841 -0.912 -1.18 -0.104  
## 8 -0.554 -0.547 -1.06 -0.166 1.10   
## 9 -1.06 -1.05 -0.147 -0.437 -0.668  
## 10 -0.554 -0.518 -0.692 0.522 1.10   
## # ℹ 3,072 more rows  
## # ℹ 15 more variables: debt\_to\_income <dbl>, total\_credit\_lines <dbl>,  
## # years\_credit\_history <dbl>, loan\_default <fct>,  
## # loan\_purpose\_credit\_card <dbl>, loan\_purpose\_medical <dbl>,  
## # loan\_purpose\_small\_business <dbl>, loan\_purpose\_home\_improvement <dbl>,  
## # application\_type\_joint <dbl>, term\_five\_year <dbl>,  
## # homeownership\_rent <dbl>, homeownership\_own <dbl>, …

**Data Analysis**

**Question 1**

**Question**: Is there a difference in rate of defaulting on payments based on the amount of the loan and the term it is to be paid in?

**Answer**: The amount the loan is for seems to not be a major factor. In fact, the average loan amounts were larger for those who did not default in both the 3 and 5 year loans. However, the term of the loan is important. Nearly 55% of the defaulted loans were 5 year terms while the other 45% was the 3 year term. This shows you are having a problem with your 5 year term loans. Furthermore, only around 27% of people who took out a 3 year loan defaulted, while on the other hand, close to 55% of those whom took out a 5 year loan defaulted. This shows there is an issue with the 5 year loans being given.

loans\_df**%>%**  
 **group\_by**(loan\_default, term)**%>%**  
 **summarize**(avf\_loan\_amount= **mean**(loan\_amount), count = **n**())

## # A tibble: 4 × 4  
## # Groups: loan\_default [2]  
## loan\_default term avf\_loan\_amount count  
## <fct> <fct> <dbl> <int>  
## 1 yes three\_year 13062. 693  
## 2 yes five\_year 21079. 837  
## 3 no three\_year 13725. 1895  
## 4 no five\_year 23217. 685

**ggplot**(loans\_df, **aes**( x = loan\_amount, color = loan\_default))**+**  
 **geom\_histogram**()**+**  
 **facet\_wrap**(**~**term)**+**  
 **labs**(title="Loan Term and Amount vs Defaulting", y ="Defaulted",x="Amount of Loan")

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



**Question 2**

**Question**: How does previous bankruptcy affect the rates of loan defaulting?

**Answer**: Of the 1530 people that did default, around 13% have had a past bankruptcy in comparison to 11% of the other 2580 people who did not default, but do have a past bankruptcy. Roughly 42% of those with a history of bankruptcy defaulted. Further more, only about 37% of people who never had bankruptcy defaulted. There is a clear increase in rates of defaulting for those with a history of bankruptcy, however, with that said, it is still a mild difference.

loans\_df**%>%**  
 **group\_by**(loan\_default, history\_bankruptcy)**%>%**  
 **summarize**(count = **n**())

## # A tibble: 4 × 3  
## # Groups: loan\_default [2]  
## loan\_default history\_bankruptcy count  
## <fct> <fct> <int>  
## 1 yes yes 203  
## 2 yes no 1327  
## 3 no yes 283  
## 4 no no 2297

**Question 3**

**Question**: Are the people with higher installment amounts defaulting on their payments more frequently than those with lower rates? Does this vary among loan purpose?

**Answer**: It does appear that both the installment amount and the reason for the loan greatly affect weather a person defaults on their loan. For debt consolidation 25% defaulted and the average payment for those who defaulted was 16% higher. For credit card debt 54% of people defaulted and the average payment for those who defaulted was 15% higher. For medical assistance loans 52% defaulted and their average payment was around 11%. Small business loans saw a total of 26% defaulting and those who did had an average payment that was 23% higher. Lastly, those who took out a loan for home improvement had a total of 28% that defaulted, and had an average payment that was 18% higher. In every category of loan those who defaulted had higher average payments and they were all above 500$ on average. Additionally, credit card debt and medical loans had 2x the amount people defaulting in comparison to the other 3 categories.

loans\_df**%>%**  
 **group\_by**(loan\_default, loan\_purpose)**%>%**  
 **summarize**(defaulted = **n**(), mean\_installment = **mean**(installment))

## # A tibble: 10 × 4  
## # Groups: loan\_default [2]  
## loan\_default loan\_purpose defaulted mean\_installment  
## <fct> <fct> <int> <dbl>  
## 1 yes debt\_consolidation 308 540.  
## 2 yes credit\_card 470 526.  
## 3 yes medical 384 522.  
## 4 yes small\_business 221 565.  
## 5 yes home\_improvement 147 546.  
## 6 no debt\_consolidation 910 464.  
## 7 no credit\_card 409 457.  
## 8 no medical 251 472.  
## 9 no small\_business 632 459.  
## 10 no home\_improvement 378 464.

**ggplot**(loans\_df, **aes**( x = loan\_default, y = installment, color = loan\_default))**+**  
 **geom\_boxplot**()**+**  
 **facet\_wrap**(**~**loan\_purpose)**+**  
 **labs**(title="Loan Type and Installment vs Defaulting", y ="Installment",x="Defaulted")



**Predictive Modeling**

**Model 1**

*#Model 1*  
logistic\_model <- **logistic\_reg**() **%>%**  
**set\_engine**("glm") **%>%**  
**set\_mode**("classification")  
  
*#Workflow 1*  
loan\_log\_wf <- **workflow**() **%>%**  
**add\_model**(logistic\_model) **%>%**  
**add\_recipe**(loan\_recipe)  
  
*#Fit 1*  
loan\_log\_fit <- loan\_log\_wf**%>%**  
 **fit**(loan\_training)  
  
*# Extract trained model from workflow fit and view it 1*  
loan\_trained\_model\_log <- loan\_log\_fit **%>%**  
**pull\_workflow\_fit**()

## Warning: `pull\_workflow\_fit()` was deprecated in workflows 0.2.3.  
## ℹ Please use `extract\_fit\_parsnip()` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

loan\_trained\_model\_log

## parsnip model object  
##   
##   
## Call: stats::glm(formula = ..y ~ ., family = stats::binomial, data = data)  
##   
## Coefficients:  
## (Intercept) loan\_amount   
## 8.79197 28.02629   
## installment interest\_rate   
## -25.22909 -2.84596   
## annual\_income current\_job\_years   
## -0.16749 0.13550   
## debt\_to\_income total\_credit\_lines   
## -0.28770 0.24616   
## years\_credit\_history loan\_purpose\_credit\_card   
## -0.04054 -1.18667   
## loan\_purpose\_medical loan\_purpose\_small\_business   
## -1.75043 0.11032   
## loan\_purpose\_home\_improvement application\_type\_joint   
## -0.02612 -0.56007   
## term\_five\_year homeownership\_rent   
## -16.36598 -0.81587   
## homeownership\_own missed\_payment\_2\_yr\_no   
## -0.58275 0.64769   
## history\_bankruptcy\_no history\_tax\_liens\_no   
## -0.18180 -0.34375   
##   
## Degrees of Freedom: 3081 Total (i.e. Null); 3062 Residual  
## Null Deviance: 4069   
## Residual Deviance: 756.1 AIC: 796.1

*# variable importance 1*  
**vip**(loan\_trained\_model\_log)



*#MODEL ONE IS MADE THIS IS PREIDCTIONS FOR MODEL 1*  
  
*# Automate the metrics process 1*  
last\_fit\_model\_log <- loan\_log\_wf **%>%**  
**last\_fit**(split = loan\_split)  
  
*# Obtain data frame with predictions 1*  
last\_fit\_results\_log <- last\_fit\_model\_log **%>%**  
**collect\_predictions**()  
last\_fit\_results\_log

## # A tibble: 1,028 × 7  
## id .pred\_yes .pred\_no .row .pred\_class loan\_default .config  
## <chr> <dbl> <dbl> <int> <fct> <fct> <chr>   
## 1 train/test split 0.967 3.34e-2 1 yes yes Prepro…  
## 2 train/test split 0.637 3.63e-1 4 yes yes Prepro…  
## 3 train/test split 1.00 4.68e-8 6 yes yes Prepro…  
## 4 train/test split 0.0180 9.82e-1 8 no no Prepro…  
## 5 train/test split 0.000107 1.00e+0 10 no no Prepro…  
## 6 train/test split 0.848 1.52e-1 17 yes no Prepro…  
## 7 train/test split 0.652 3.48e-1 21 yes yes Prepro…  
## 8 train/test split 0.999 7.23e-4 24 yes yes Prepro…  
## 9 train/test split 0.642 3.58e-1 26 yes yes Prepro…  
## 10 train/test split 0.0903 9.10e-1 28 no no Prepro…  
## # ℹ 1,018 more rows

*# Accuracy and Area under ROC(table) 1*  
last\_fit\_model\_log **%>%** **collect\_metrics**()

## # A tibble: 2 × 4  
## .metric .estimator .estimate .config   
## <chr> <chr> <dbl> <chr>   
## 1 accuracy binary 0.952 Preprocessor1\_Model1  
## 2 roc\_auc binary 0.989 Preprocessor1\_Model1

*# Use dataframe to get the ROC Curve 1*  
last\_fit\_results\_log **%>%**  
**roc\_curve**(truth = loan\_default, .pred\_yes) **%>%**  
**autoplot**()



**Model 2**

*#Model 2*   
lda\_model <- **discrim\_regularized**(frac\_common\_cov = 1) **%>%**  
**set\_engine**('klaR') **%>%**  
**set\_mode**('classification')  
  
*#Workflow 2*  
lda\_wf <- **workflow**() **%>%**  
**add\_model**(lda\_model) **%>%**  
**add\_recipe**(loan\_recipe)  
  
*#Fit 2*  
loan\_lda\_fit <- lda\_wf**%>%**  
 **fit**(loan\_training)  
  
*# Extract trained model from workflow fit and view it 2*  
loan\_trained\_model\_lda <- loan\_lda\_fit **%>%**  
**pull\_workflow\_fit**()  
loan\_trained\_model\_lda

## parsnip model object  
##   
## Call:   
## rda(formula = ..y ~ ., data = data, lambda = ~1)  
##   
## Regularization parameters:   
## gamma lambda   
## 0.001061772 1.000000000   
##   
## Prior probabilities of groups:   
## yes no   
## 0.3721609 0.6278391   
##   
## Misclassification rate:   
## apparent: 5.516 %  
## cross-validated: 5.679 %

*#Train and evaluate with last\_fit() 2*  
last\_fit\_lda <- lda\_wf **%>%**  
**last\_fit**(split = loan\_split)  
  
*#Estimated Probablities 2*  
lda\_predictions <- last\_fit\_lda **%>%**  
**collect\_predictions**()  
  
*# Accuracy and Area under the ROC curve 2*  
last\_fit\_lda **%>%** **collect\_metrics**()

## # A tibble: 2 × 4  
## .metric .estimator .estimate .config   
## <chr> <chr> <dbl> <chr>   
## 1 accuracy binary 0.949 Preprocessor1\_Model1  
## 2 roc\_auc binary 0.987 Preprocessor1\_Model1

*# ROC Curve 2*  
lda\_predictions **%>%**  
**roc\_curve**(truth = loan\_default, .pred\_yes) **%>%**  
**autoplot**()



**Summary of Results**

**Introduction:**

When it comes to losing money, it’s simply something a successful business cannot afford to have happen. Much less with the banks record levels of customers defaulting on their loans in the past couple of years. I will be exploring the factors that lead to loan default and develop a machine learning algorithm that can predict the likelihood of an applicant defaulting on their loan in the future. This will allow your bank to flourish financially, and help pull it up by its bootstraps after the last several years of less than satisfactory performance. Three main questions being asked in this analysis include: Is there a difference in rate of defaulting on payments based on the amount of the loan and the term it is to be paid in? How does previous bankruptcy affect the rates of loan defaulting? And are the people with higher installment amounts defaulting on their payments more frequently than those with lower rates? Does this vary among loan purposes? This will allow us to explore the important factors that are adversely affecting the bank’s success in different areas, and give us a starting point with things to change, while the machine learning algorithm helps prevent any further defaults with your current setup. The data set being used contains information on over 3,500 individuals who secured a personal loan in 2017 from a national bank.

**Highlights and Key Findings:**

Q1) There is a clear increase in rates of defaulting for those with a history of bankruptcy, however, with that said, it is still a mild difference.

Q2) This shows there is an issue with the 5 year loans being given.

Q3) In every category of loan those who defaulted had higher average payments and they were all above 500$ on average. Additionally, credit card debt and medical loans had 2x the amount people defaulting in comparison to the other 3 categories.

**Recommendations Based Off Key Findings:**

Q1) I strongly recommend reducing or doing away with your 5-year loans.

Reason: Over half of those who take one out default on their payment at some point. This means you have a coin flip chance of losing the money attached to any 5-year loan you give out. Additionally, you are losing twice as many loans when compared to the 3-year loans, and the 5-year loans are typically worth around 50% more as well. Losing twice as many loans that are each worth 50% more is not good for business. This needs to change as soon as possible.

Q2) I would recommend the company to add slightly harsher penalties to those who have a proven past of financial irresponsibility, and to be more cautious of the situation surrounding these loan applicants. Perhaps flagging the application and putting it through a more rigorous financial screening process to ensure they have no other obligations.

Reason: Reduces the risk of loss for the company. It allows the company to have more confidence in giving out loans to those with a history of defaulting, which totals around 10% of your overall business. This is a large chunk and is something that should not be taken lightly.

Q3) I would recommend a few things:

One, I would recommend reducing the amount of loans that will have installments higher than 500.

Reason: The data shows every single category of loan purpose those who defaulted had an average payment of well over 500, while on the other hand, for those who did not default they had an average of closer to 450. This will reduce the number of people who struggle to make monthly payments and in turn will enable them to pay you back faster and more consistently.

Secondly, reduce the amount of credit card and medical loans you are giving out, maybe even all together.

Reason: They are your biggest risks when looking at reasons for loans. Not to mention credit card debt and medical expenses do not always have a foreseeable end, meaning the situation can get much worse for the person you have loaned the money to. This ultimately hurts the company as they cannot pay you back as fast, if at all.

Lastly, never under any circumstance give out a loan with an installment of over 1200.

Reason: The data shows about 1 percent of people who receive them can finish paying it back without defaulting. This is an unnecessary risk for the bank.

**Best Model?**

Both of the models we created were extremely accurate, however, the logistic model is deemed superior by a small margin. Looking at the accuracy of each model we see that in the logistic regression model we have an estimated score of 95.2%, while the LGA model scored in at 94.9%. Furthermore, this can be determined through using an ROC curve and measuring the area underneath it. In terms of model performance, an area under the ROC value between 0.9 - 1 indicates an “A”, 0.8 - 0.9 a “B”, and so forth. Anything below a 0.6 is an “F” and indicates poor model performance. You can think of it like a letter grade for the model. The logistic regression model scored an amazing 0.989 while the LGA model scored a 0.987. While both models perform extremely well, the logistic regression shows to be a better model and have a higher rate of accuracy. Additionally, it is important to note, of all the variables/factors in this data set, a few ranked with a much higher importance than the others: 5 year loans, installments, loan amount, interest rate, and purpose for loan (medical and credit card). This is useful information because you can further target these areas, improving performance in these areas will likely also improve the model, as seen in the recommendation section.

**Conclusion:**

In conclusion, there is a clear issue with the 5 year loans being given, the average installments needs to be less to keep people from defaulting, you need to look into changing or eliminating loans for credit cards and medical expenses, and you need to be more vigilant when it comes to applications of those with a history of bankruptcy. Additionally, the best model to use going forward, for predicting an applicant’s likelihood to default on a potential loan, is the logistic regression model.