# The-Financial-analysis-using-classification-Models

# Suppress dplyr summarise grouping warning messages  
options(dplyr.summarise.inform = FALSE)  
  
## Add R libraries here  
library(tidyverse)  
library(tidymodels)  
library(vip)  
library(discrim)  
  
# Load data  
loans\_df <- read\_rds("~/Desktop/MIS431 FINAL/loan\_data.rds")

# Summary of Results

* What problem(s) is this company trying to solve? Why are they important to their future success?
* Banks have experienced a large scale of default on customer loans through the years. To them, it has resulted in substantial loss from defaulting on loans, and it will be enable them to prevent further costs as the determinants become clear. Thereby, reducing such factors might lead the banks to reduce the default rates in the future; and in the end, they will be able to achieve more gains.
  + What was the goal of your analysis? What questions were you trying to answer and why do they matter? The primary goal of the analysis is to find the factors leading to loan default and develop a machine learning algorithms to predict the possibility of an applicants. The analysis will be based on the data of over 4,000 customers who are using loans from the National bank, and variables are used to find what ar driving to the loan default: i.interest rate, ii.debt-to-income ratio,iii.history of bankruptcy, iv.annual income, v. home ownership, vi. loan term.

1. Highlights and key findings from your Exploratory Data Analysis section
   * What were the interesting findings from your analysis and **why are they important for the business**?

Interestingly, among 6 variables, history of bankruptcy, debt-to-income ratio, and home ownership have insignificant influences on defaulting rate of loans. Thus, these three factors should be not be the consideration when banks approve the loan customers. On the other hand, loan interest rate, annual income, and loan term are the determinants showing the huge difference. Specifically, loan interest rate is shown to be higher to the people who default the loans. On top of that, the longer the loan term is, the higher default rate is based on the finding from comparing the difference between the 5 years loan and 3 years loan term. Finding three dominant factors to defaulting loans is important for the company since they can reduce the loss.

- This section is meant to \*\*establish the need for your recommendations\*\* in the following section

Related to three major factors, loan interest rate, annual income, and loan term, recommendation should be related to these three factors, not the history of bankruptcy, debt-to-income ratio, or home ownership. Such Recommendations will be helpful for bank to get more money back from customers and lower the default rate.

1. My “best” classification model and an analysis of its performance
   * In this section you should talk about the expected error of your model on future data
     + To estimate future performance, you can use your model performance results on the **test data**
   * You should discuss at least one performance metric, such as an F1 or ROC AUC for your model. However, you must explain the results in an **intuitive, non-technical manner**. Your audience in this case are executives at a bank with limited knowledge of machine learning.

* In this analysis, three classification models are used: Logistic Regression Model, Random Forest Model, and LDA Model.Through Machine Learning modeling, three dominant factors are found, and ROC AUC is also used for the accuracy, 99%, 99%, and 97.6% are measured in logistic regression model, LDA model, and Random Forest model for each. It manifests that higher ROC AUC rate proves that it has more accurate result.Therefore, logistic regression model supports loan defaulting rate is greatly affected by installment, the amount of loan, interest rate, and loan term.

1. My recommendations to the company on how to reduce loan default rates

First, bank should change the loan term from 5 years to 3 years. Based on the results from logistic regression model, the number of people with 5 years loan account for the second biggest part in defaulting on the loans. In statistics, 3 years loan has a half of defaulting rate, and thereby the it is recommend to reduce the loan term from 5 years to 3 years. It will not only reduce loan default rate but also encourage more customers to continuously loan the money.

Second, bank should loan the people who have higher annual income rate. In the data, it was noticeable that people with higher annual rate show more less default rate. Therefore, based on this result, banks should try to keep customers who are with higher annual income and higher the standard for the people who can be eligible for loan program.

Third, bank should reduce interest rate to lower the loan default rate since it is turned out as the most determining factor for the loan default rate. In summary table, 90% of customers who have not paid were using loan program at the interest rate higher than 10% while other customers with less interest rate were more likely to pay their money back to the bank. Default rate is 0 within the customers with the interest rate lower than 9%. If the bank is willing to focus on reducing loan defaulting rates, lower interest rate will result in significant difference.

1. Conclusion

Wrap up the report with concluding remarks by summarizing the results and your recommendations in two or three paragraphs.

In this report, six variables have been reviewed to see if each of them can be the determinant for loan default rate. The statistics given from the analysis will contribute to the banks’ gains in the future by reducing loan default rate. In analysis through classification model(logitic regression, Random Forest, and LDA), three factors leading to higher loan default rate are loan interest rate, annual income, and loan term. Based on the analysis, three recommendations have been made. First, company should lower the interest rate since interest rate is the biggest determinant. Second, bank should loan the people who have higher annual income rate since they are more likely to pay the money back. Third, bank should change to the short loan term from 5 years to 3 years. Since these recommendations are highly related to the biggest three factors, it will show the significant difference.

**Are there differences in loan default rates by loan purpose?**

**Answer**: Yes, the data indicates that credit card and medical loans have significantly larger default rates than any other type of loan. In fact, both of these loan types have default rates at more than 50%. This is nearly two times the average default rate for all other loan types.

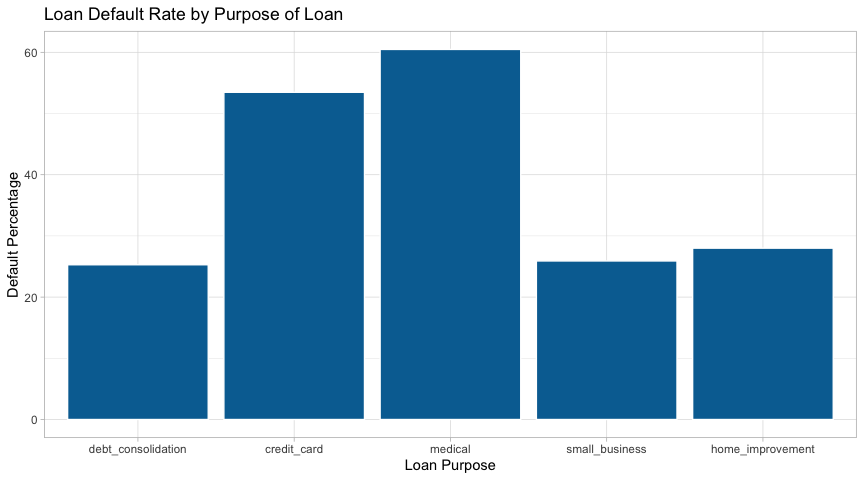
### Summary Table

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

## # A tibble: 5 × 4  
## loan\_purpose n\_customers customers\_default default\_percent  
## <fct> <int> <int> <dbl>  
## 1 debt\_consolidation 1218 308 25.3  
## 2 credit\_card 879 470 53.5  
## 3 medical 635 384 60.5  
## 4 small\_business 853 221 25.9  
## 5 home\_improvement 525 147 28

### Data Visulatization

default\_rates <- loans\_df %>%  
 group\_by(loan\_purpose) %>%   
 summarise(n\_customers = n(),  
 customers\_default = sum(loan\_default == 'yes'),  
 default\_percent = 100 \* mean(loan\_default == 'yes'))  
  
  
ggplot(data = default\_rates, mapping = aes(x = loan\_purpose, y = default\_percent)) +  
 geom\_bar(stat = 'identity', fill = '#006EA1', color = 'white') +  
 labs(title = 'Loan Default Rate by Purpose of Loan',  
 x = 'Loan Purpose',  
 y = 'Default Percentage') +  
 theme\_light()



## Question 1

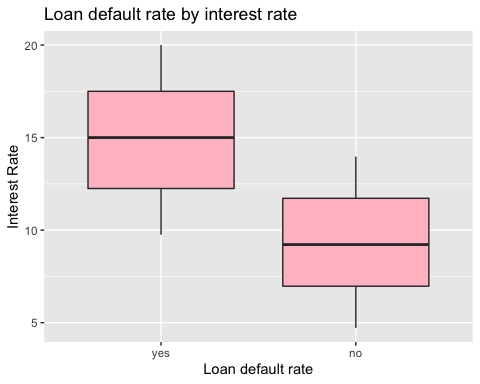
**Question**: Does higher interest rate result in higher loan default rate?

**Answer**: Yes. Higher interest rate is more likely to result in loan default rate based on the box plot below. The average interest rate among customers who default lone is 14.89 while people who do not default lone use the program at 9.34. It proves that higher interest rate is more likely to result in higher loan default rate. Among customers who have default loans, minimum interest rate is 9.75 and the highest interest rate is 20.00. Compared to them, customers who have not default loans show the far lower interest rate as 4.72 and 13.97 for minimum and maximum for each. Therefore, it shows that there is the clear relationship that higher interest rate leads to the higher defaulting loan rate to customers.

loans\_df %>%  
 group\_by(loan\_default) %>%  
 summarize(minimum = min(interest\_rate),  
 maximum = max(interest\_rate),  
 standard\_deviation = sd(interest\_rate),  
 average = mean(interest\_rate),  
 interestrate\_over\_10 = mean(interest\_rate >= 10),  
 interestrate\_less\_9 = mean(interest\_rate < 9))

## # A tibble: 2 × 7  
## loan\_default minimum maximum standard\_deviation average interestrate\_over\_10  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 yes 9.75 20 3.00 14.9 0.978  
## 2 no 4.72 14.0 2.75 9.30 0.408  
## # … with 1 more variable: interestrate\_less\_9 <dbl>

ggplot(data = loans\_df, mapping = aes(x = loan\_default, y = interest\_rate)) +  
 geom\_boxplot(fill = 'pink') +  
 labs(title = 'Loan default rate by interest rate', x= "Loan default rate", y = "Interest Rate")



## Question 2

**Question**: Are customers with bankrutpcy history more likely to default the loans in the future? **Answer**: No. There is no significant difference in loan default rates. The data collected based on 486 customers, 41.76% of them have the history of bankruptcy. Among 3264, 36.6% of them do not default the loans, which is significant. Therefore, compared to other factors, the history of bankruptcy does not make the big difference in lowering loan default rate.

loans\_df %>%  
 group\_by(history\_bankruptcy) %>%  
 summarise(n\_customers = n(),  
 customers\_default = sum(loan\_default == 'yes'),  
 default\_percent = 100 \* mean(loan\_default == 'yes'))

## # A tibble: 2 × 4  
## history\_bankruptcy n\_customers customers\_default default\_percent  
## <fct> <int> <int> <dbl>  
## 1 yes 486 203 41.8  
## 2 no 3624 1327 36.6

## Question 3

**Question**: Is short term loan period likely to default the loans in the future? **Answer**: Yes. It can be considered as one of the biggest determinants. Comparing three year to five years, default rate of five years is twice higher at 54.99%. Therefore, it makes the huge difference.

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

## # A tibble: 2 × 4  
## term n\_customers customers\_default default\_percent  
## <fct> <int> <int> <dbl>  
## 1 three\_year 2588 693 26.8  
## 2 five\_year 1522 837 55.0

## Question 4

**Question**: Is the current debt affecting to people when it comes to paying the money back? **Answer**: No. they do not have any relationship. debt-to-income ratio is appeared same in average rate as 22.47 and 18.59 for the number of people who have defaulted the loan who have not.

loans\_df %>%  
 group\_by(loan\_default) %>%  
 summarise(n\_customers = n(),  
 minimum = min(debt\_to\_income),  
 maximum = max(debt\_to\_income),  
 standard\_deviation = sd(debt\_to\_income),  
 average = mean(debt\_to\_income))

## # A tibble: 2 × 6  
## loan\_default n\_customers minimum maximum standard\_deviation average  
## <fct> <int> <dbl> <dbl> <dbl> <dbl>  
## 1 yes 1530 0 182. 12.7 22.5  
## 2 no 2580 0 438. 14.9 18.6

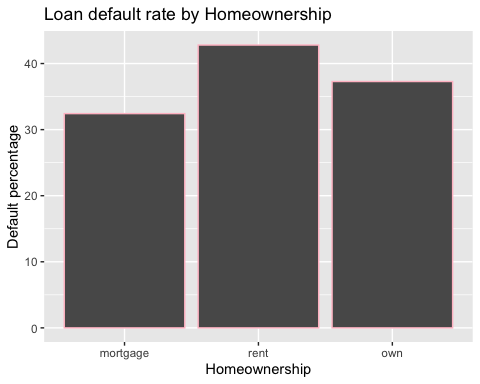
## Question 5

**Question**: Do people who have home are less likely to default loans? **Answer**: No. homeownership is not the factor of leading to defaulting loan. people who own their home show 37% of loan\_default rate while people who rent the house show 42.79. The rate is slightly higher at 5%, which is significant.

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

## # 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

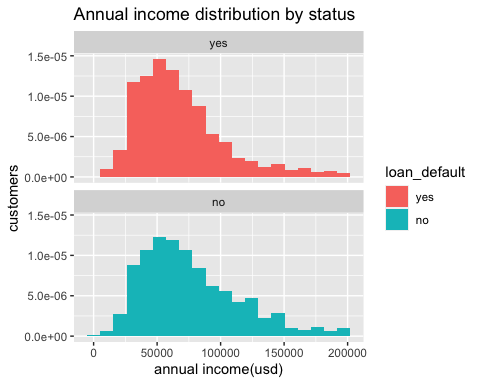
ggplot(data = default\_rates, mapping = aes(x = homeownership, y = default\_percent)) +  
 geom\_bar(color = "pink", stat = "identity") +  
 labs(title = 'Loan default rate by Homeownership', x= "Homeownership", y = "Default percentage")



## Question 6

**Question**: Do people who have higher income are less likely to default the loans? **Answer**: Yes. Customers who default on their loans have lower income than customers who have not. From 100k, the proportion of customers who have not default on their loans is far higher than those who have not.

ggplot(data = loans\_df, aes( x= annual\_income, fill = loan\_default)) +  
 geom\_histogram(aes(y = ..density..), bins = 20) +  
 facet\_wrap(~ loan\_default, nrow =2) +   
 labs(title = "Annual income distribution by status",  
 x = "annual income(usd)",  
 y = "customers")



# Predictive Modeling [70 Points]

In this section of the project, you will fit **three classification algorithms** to predict the response variable,loan\_default. You should use all of the other variables in the loans\_df data as predictor variables for each model.

You must follow the machine learning steps below.

The data splitting and feature engineering steps should only be done once so that your models are using the same data and feature engineering steps for training.

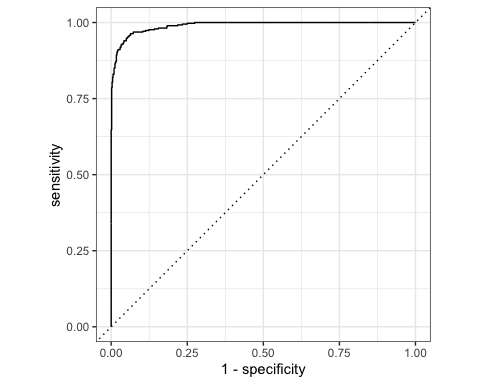
* Split the loans\_df data into a training and test set (remember to set your seed)
* Specify a feature engineering pipeline with the recipes package
  + You can include steps such as skewness transformation, dummy variable encoding or any other steps you find appropriate
* Specify a parsnip model object
  + You may choose from the following classification algorithms:
    - Logistic Regression
    - LDA
    - QDA
    - KNN
    - Decision Tree
    - Random Forest
* Package your recipe and model into a workflow
* Fit your workflow to the training data
  + If your model has hyperparameters:
    - Split the training data into 5 folds for 5-fold cross validation using vfold\_cv (remember to set your seed)
    - Perform hyperparamter tuning with a random grid search using the grid\_random() function
    - Hyperparameter tuning can take a significant amount of computing time. Be careful not to set the size argument of grid\_random() too large. I recommend size = 10 or smaller.
    - Select the best model with select\_best() and finalize your workflow
* Evaluate model performance on the test set by Select the best model with select\_best() and calculating the area under the ROC curve on your test data

## Model 1

#Split the `loans\_df` data into a training and test set   
set.seed(100)  
loan\_split <- initial\_split(loans\_df, prop = 0.75, strata = loan\_default)  
loan\_training <- loan\_split %>% training()  
loan\_test <- loan\_split %>% testing()  
  
#Specify a feature engineering pipeline with the `recipes` package  
recipes <- recipe(loan\_default ~., data = loan\_training) %>%  
 step\_YeoJohnson(all\_numeric(), -all\_outcomes()) %>%  
 step\_normalize(all\_numeric(), -all\_outcomes()) %>%  
 step\_dummy(all\_nominal(), -all\_outcomes())  
  
recipes %>%  
 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 1.17 1.42 -0.555 1.88 1.10   
## 2 -1.60 -1.68 -0.348 0.174 -0.399  
## 3 0.975 0.613 0.233 1.33 -0.124  
## 4 -1.21 -1.20 -0.844 0.119 -0.399  
## 5 1.87 1.45 -0.0186 0.119 -0.691  
## 6 0.0362 0.0786 0.0452 -0.844 1.10   
## 7 1.64 1.00 -1.07 2.22 1.10   
## 8 -1.12 -1.08 0.0452 -1.37 1.10   
## 9 -0.531 -0.524 -1.07 -0.181 1.10   
## 10 -1.04 -1.03 -0.148 -0.452 -0.691  
## # … with 3,072 more rows, and 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>,  
## # missed\_payment\_2\_yr\_no <dbl>, history\_bankruptcy\_no <dbl>, …

#Specify a `parsnip` model object(Logistic Regression/LDA/QDA/KNN/Decision Tree/Random Forest)  
logistic\_model <-logistic\_reg() %>%  
 set\_engine("glm") %>%  
 set\_mode("classification")  
#Package your recipe and model into a workflow  
logistic\_wf <- workflow() %>%  
 add\_model(logistic\_model)%>%  
 add\_recipe(recipes)  
  
#Fit your workflow to the training data(5-fold cross validation using `vfold\_cv`, grid search using the `grid\_random()` ,Hyperparameter tuning can take a significant amount of computing time,Select the best model with `select\_best()`, Select the best model with `select\_best()`, calculating the area under the ROC curve)  
loan\_folds <- vfold\_cv(loan\_training, v =5)  
logistic\_fit <- logistic\_wf %>%  
 last\_fit(split = loan\_split)  
  
logistic\_result <- logistic\_fit %>%  
 collect\_predictions()  
#ROC Curve  
roc\_curve(logistic\_result, truth = loan\_default, estimate = .pred\_yes) %>% autoplot()



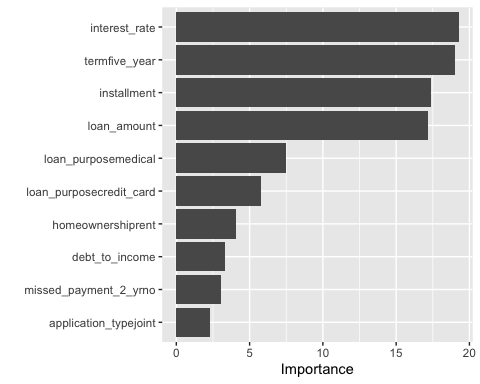
roc\_auc(logistic\_result, truth = loan\_default, .pred\_yes)

## # A tibble: 1 × 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 roc\_auc binary 0.990

model\_1<- glm(loan\_default ~., data = loan\_training, family = binomial)  
tidy\_model1 <- tidy(model\_1)  
summary(tidy\_model1)

## term estimate std.error statistic   
## Length:20 Min. :-7.340463 Min. :0.0000026 Min. :-19.2594   
## Class :character 1st Qu.:-0.579554 1st Qu.:0.0105427 1st Qu.: -4.4857   
## Mode :character Median :-0.121720 Median :0.2168513 Median : -1.1875   
## Mean :-0.055533 Mean :0.2071372 Mean : -2.3694   
## 3rd Qu.: 0.000267 3rd Qu.:0.2371219 3rd Qu.: 0.9784   
## Max. :11.606140 Max. :0.9606483 Max. : 17.1585   
## p.value   
## Min. :0.00000   
## 1st Qu.:0.00000   
## Median :0.01187   
## Mean :0.17616   
## 3rd Qu.:0.20944   
## Max. :0.95583

vip(model\_1)

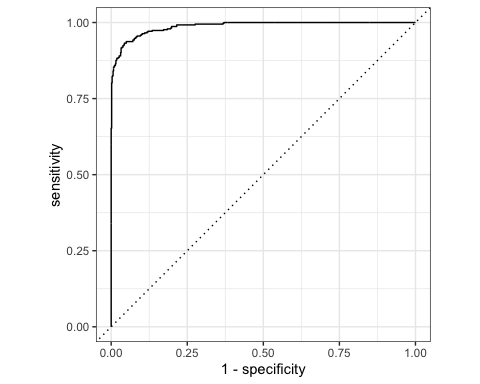


## Model 2

# Specify LDA Model  
LDA\_model <- discrim\_regularized(frac\_common\_cov = 1) %>%  
 set\_engine('klaR') %>%  
 set\_mode('classification')  
# Create a Workflow  
LDA\_wf <- workflow() %>%  
 add\_model(LDA\_model) %>%  
 add\_recipe(recipes)  
# Fit Model  
LDA\_fit <- LDA\_wf %>%  
 last\_fit(split = loan\_split)

## Warning: package 'klaR' was built under R version 4.1.2

# Collect Predictions  
LDA\_results <- LDA\_fit %>%  
 collect\_predictions()  
# ROC Curve  
roc\_curve(LDA\_results, truth = loan\_default, estimate = .pred\_yes) %>%  
 autoplot()



## Model 3

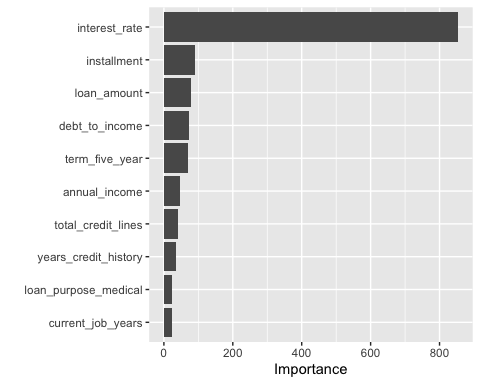
# Specify Random Forest Model  
rf\_model <- rand\_forest(mtry = tune(),  
 trees = tune(),  
 min\_n = tune()) %>%  
 set\_engine('ranger', importance = "impurity") %>%  
 set\_mode('classification')  
# Create a Workflow  
rf\_wf <- workflow() %>%  
 add\_model(rf\_model) %>%  
 add\_recipe(recipes)  
# Hyperparameter Tuning  
set.seed(01174206)  
  
rf\_grid <- grid\_random(mtry() %>% range\_set(c(2, 10)),  
 trees(),  
 min\_n(),  
 size = 10)  
  
rf\_tuning <- rf\_wf %>%  
 tune\_grid(resamples = loan\_folds,  
 grid = rf\_grid)  
# View the Best model  
best\_rf <- rf\_tuning %>%  
 select\_best(metric = 'roc\_auc')  
best\_rf

## # A tibble: 1 × 4  
## mtry trees min\_n .config   
## <int> <int> <int> <chr>   
## 1 10 1803 6 Preprocessor1\_Model10

final\_rf\_workflow <- rf\_wf %>%  
 finalize\_workflow(best\_rf)  
  
final\_rf\_workflow

## ══ Workflow ════════════════════════════════════════════════════════════════════  
## Preprocessor: Recipe  
## Model: rand\_forest()  
##   
## ── Preprocessor ────────────────────────────────────────────────────────────────  
## 3 Recipe Steps  
##   
## • step\_YeoJohnson()  
## • step\_normalize()  
## • step\_dummy()  
##   
## ── Model ───────────────────────────────────────────────────────────────────────  
## Random Forest Model Specification (classification)  
##   
## Main Arguments:  
## mtry = 10  
## trees = 1803  
## min\_n = 6  
##   
## Engine-Specific Arguments:  
## importance = impurity  
##   
## Computational engine: ranger

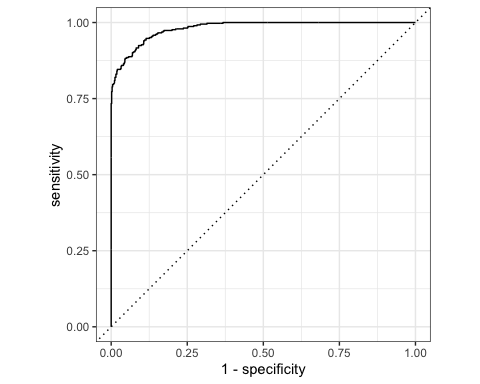
# Fit the Model  
rf\_wf\_fit <- final\_rf\_workflow %>%  
 fit(data = loan\_training)  
# Extract Trained Model  
rf\_fit <- rf\_wf\_fit %>%  
 extract\_fit\_parsnip()  
  
vip(rf\_fit)



# Train and Evaluate with Last fit  
rf\_last\_fit <- final\_rf\_workflow %>%  
 last\_fit(loan\_split)  
# Accuracy and ROC AUC  
rf\_last\_fit %>% collect\_metrics()

## # A tibble: 2 × 4  
## .metric .estimator .estimate .config   
## <chr> <chr> <dbl> <chr>   
## 1 accuracy binary 0.927 Preprocessor1\_Model1  
## 2 roc\_auc binary 0.981 Preprocessor1\_Model1

# ROC Curve  
rf\_last\_fit %>% collect\_predictions() %>%  
 roc\_curve(truth = loan\_default, estimate = .pred\_yes) %>%  
 autoplot()



— End of the Project —