Machine Learning Project

Your Name:Bhargav Teja Jakku
Your G Number: G01334185

```
# Suppress dplyr summarise grouping warning messages
options(dplyr.summarise.inform = FALSE)

library(tidyverse)
library(tidymodels)
library(discrim)
credit_card_df <- readRDS(url('https://gmubusinessanalytics.netlify.app/data/credit_card_df.rds'))
```

Data Analysis

In this section, you must think of at least 5 relevant questions that explore the relationship between `customer_status` and the other variables in the `credit_card_df` data set. The goal of your analysis should be discovering which variables drive the differences between customers who do and do not close their account.

You must answer each question and provide supporting data summaries with either a summary data frame (using `dplyr`/`tidyr`) or a plot (using `ggplot`) or both.

In total, you must have a minimum of 3 plots (created with `ggplot`) and 3 summary data frames (created with `dplyr`) for the exploratory data analysis section. Among the plots you produce, you must have at least 3 different types (ex. box plot, bar chart, histogram, scatter plot, etc. . .)

See the Data Analysis Project for an example of a question answered with a summary table and plot.

Note: To add an R code chunk to any section of your project, you can use the keyboard shortcut `Ctrl` + `Alt` + `i` or the `insert` button at the top of your R project template notebook file.

Question 1

Question: Is there a relationship between customer closing their account with their income?

Answer:

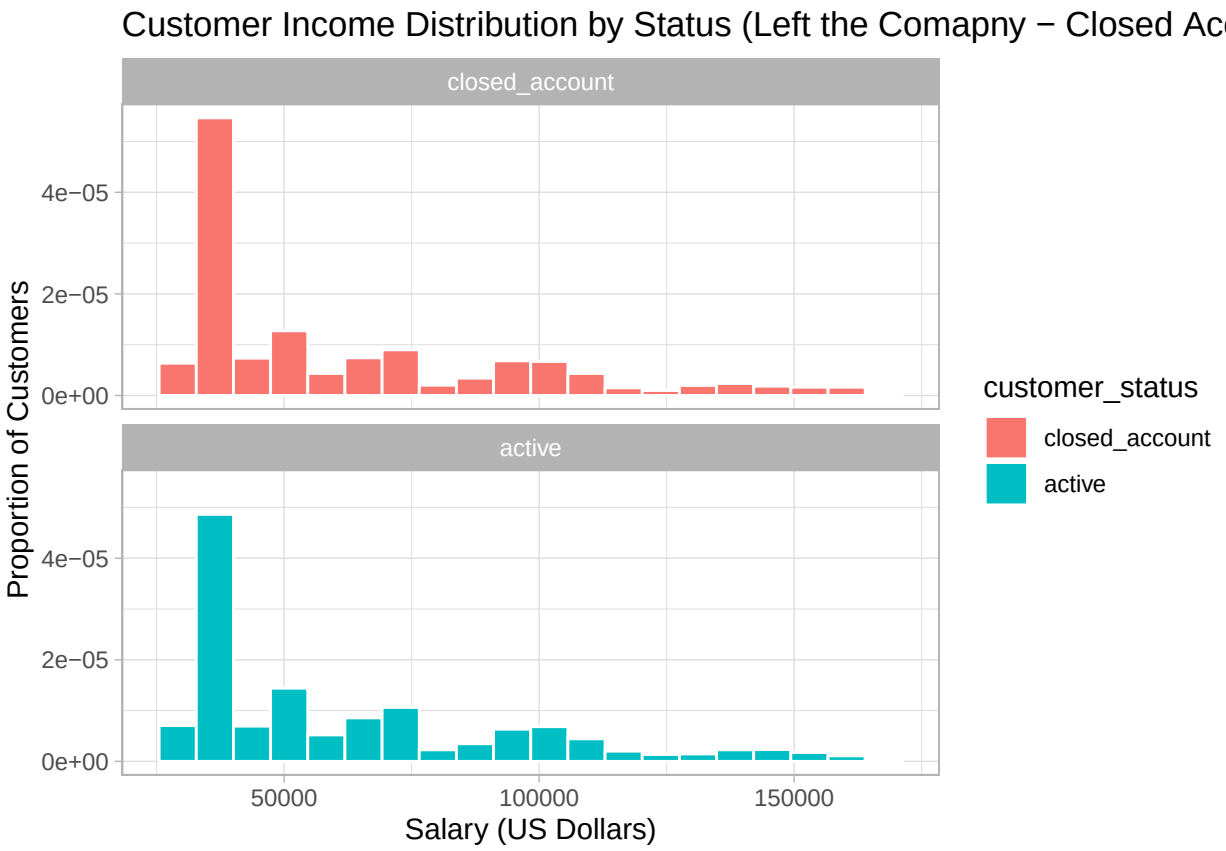
Summary Table

```
credit_card_df %>% group_by(customer_status) %>%
                   summarise(n_status = n(),
                             min_salary = min(income),
                             avg_salary = mean(income),
                             max_salary = max(income),
                             sd_salary = sd(income))
```

```
## # A tibble: 2 x 6
##   customer_status n_status min_salary avg_salary max_salary sd_salary
##   <fct>              <int>      <dbl>      <dbl>      <dbl>     <dbl>
## 1 closed_account      2092      30198     61602.     168522    33659.
## 2 active              2535      30094     62843.     166225    33306.
```

Data Visualization

```
ggplot(data =credit_card_df, aes(x = income, fill = customer_status)) +
   geom_histogram(aes(y = ..density..), color = "white", bins = 20) +
   facet_wrap(~ customer_status, nrow = 2) +
   labs(title = "Customer Income Distribution by Status (Left the Comapny - Closed Account/Active)",
          x = "Salary (US Dollars)", y = "Proportion of Customers")+theme_classic()+theme_light()
```



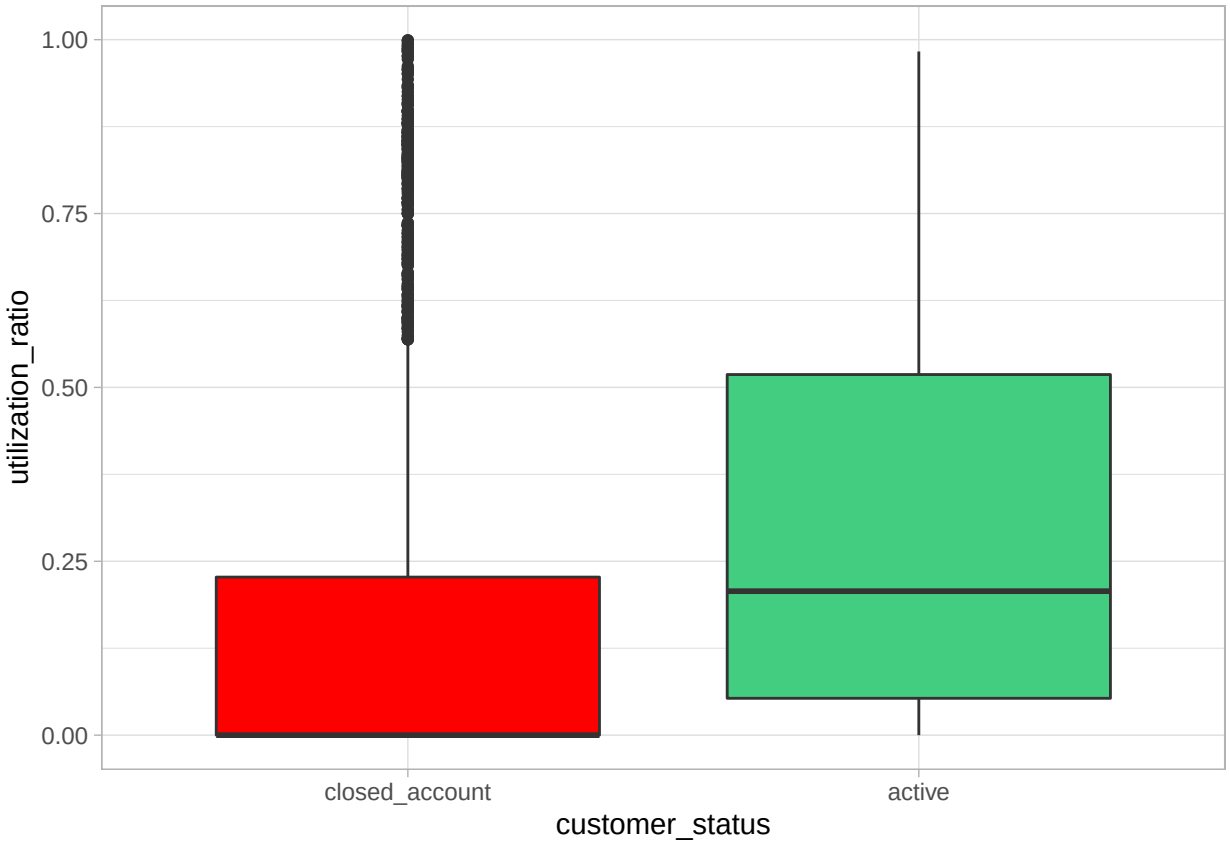
Question 2

Question: Is there any relationship between the average monthly balance to credit limit of the customer with their account status?

Answer:

Data Visualization

```
ggplot(credit_card_df, aes(x = customer_status, y = utilization_ratio, fill = customer_status))+theme_cl
```



Question 3

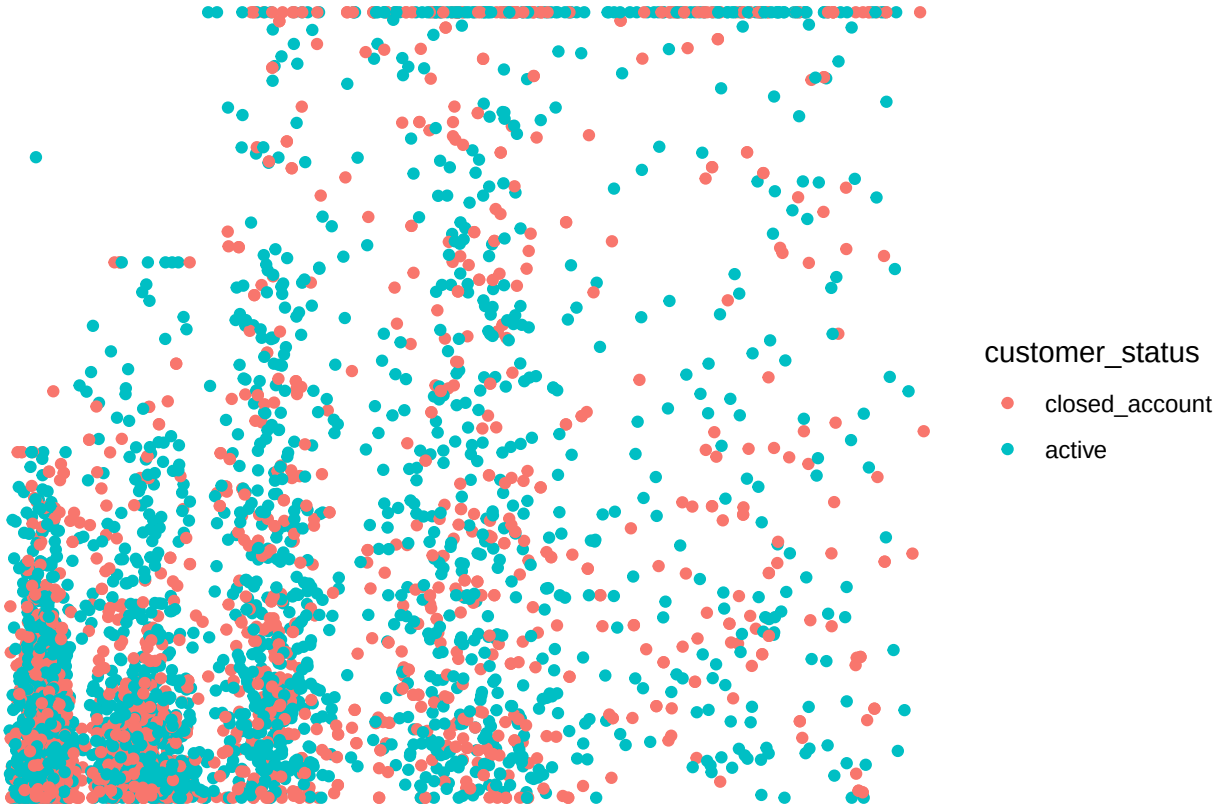
Is there any relationship between the customer status with income and credit limit?

Question:

Answer:

Data Visualization

```
credit_card_df%>%ggplot(aes(x=income,y=credit_limit,
                            color=customer_status))+
  geom_point()+theme_classic()+theme_void()
```



Question 4

Question: The effect of the education on the account status of the customers.

Answer:

Summary Table

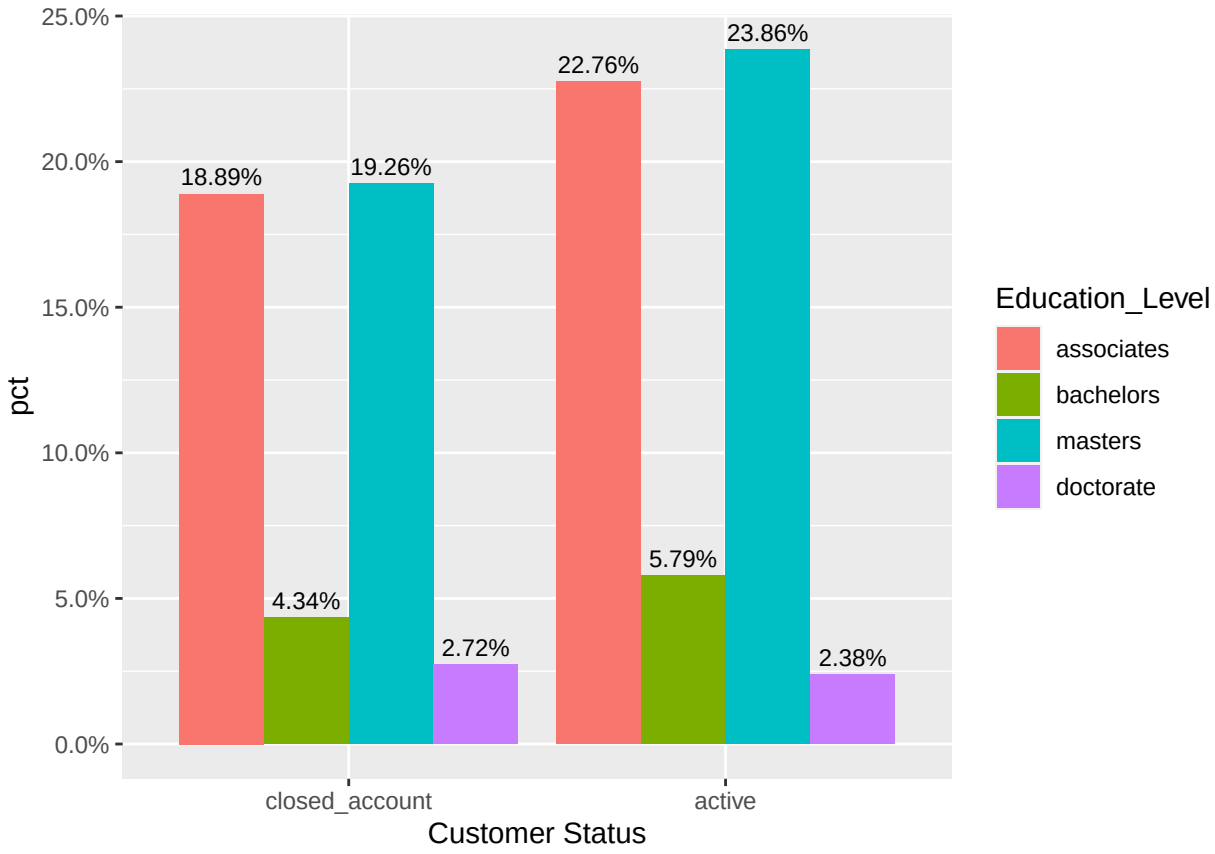
```
credit_card_df %>% count(customer_status, education) %>%
    group_by(customer_status) %>%
    mutate(pct=n/sum(n))
```

```
## # A tibble: 8 x 4
## # Groups:   customer_status [2]
##   customer_status education      n    pct
##   <fct>           <fct>      <int>  <dbl>
## 1 closed_account  associates   874 0.418
## 2 closed_account  bachelors    201 0.0961
## 3 closed_account  masters      891 0.426
## 4 closed_account  doctorate    126 0.0602
## 5 active          associates  1053 0.415
## 6 active          bachelors    268 0.106
```

```
## 7 active          masters     1104 0.436
## 8 active          doctorate    110 0.0434
```

Data Visualization

```
credit_card_df %>%
  ggplot(aes(x=factor(customer_status),
             y=prop.table(stat(count)),
             fill = factor(education),
             label = scales::percent(prop.table(stat(count))))) +
  geom_bar(position = 'dodge') +
  geom_text(stat = 'count',
            position = position_dodge(.9),
            vjust = -0.5,
            size = 3) +
  scale_y_continuous(labels = scales::percent) +
  labs(x= 'Customer Status', y='pct', fill = 'Education_Level')
```



Question 5

Question: Is there any relationship between the expenditure performed by the clients with their account status?

Answer:

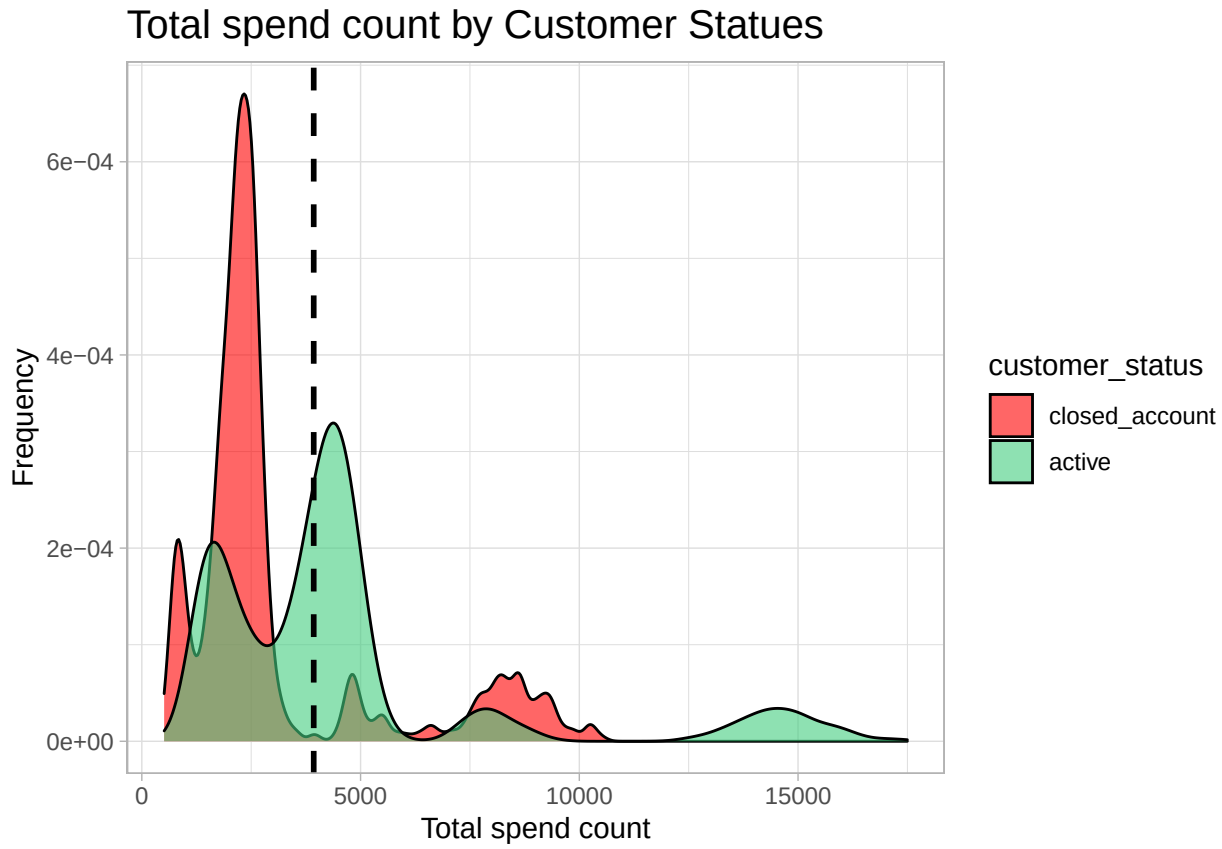
Summary Table

```
credit_card_df %>% group_by(customer_status) %>%
  summarise(avg_trans=mean(total_spend_last_year),
            max_trans= max(total_spend_last_year),
            min_trans= min(total_spend_last_year))
```

```
## # A tibble: 2 x 4
##   customer_status avg_trans max_trans min_trans
##   <fct>               <dbl>     <dbl>     <dbl>
## 1 closed_account      3121.     10583       510
## 2 active              4597.     17498       893
```

Data Visualization

```
credit_card_df %>%ggplot( aes(x=total_spend_last_year, fill=customer_status)) +geom_density(alpha=0.6) +
```



Machine Learning

In this section of the project, you will fit **three classification algorithms** to predict the outcome variable,`customer_status`.

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 `credit_card_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, correlation filters, 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
    - Refer to the following tutorial for an example - Random Grid Search
    - 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 plotting an ROC curve using `autoplot()` and calculating the area under the ROC curve on your test data

Dividing data into train and test split

```
set.seed(123)
customer_split <- initial_split(credit_card_df, prop = 0.80,
                                strata = customer_status)
customer_training <- customer_split %>% training()
customer_test <- customer_split %>% testing()
```

Feature Engineering

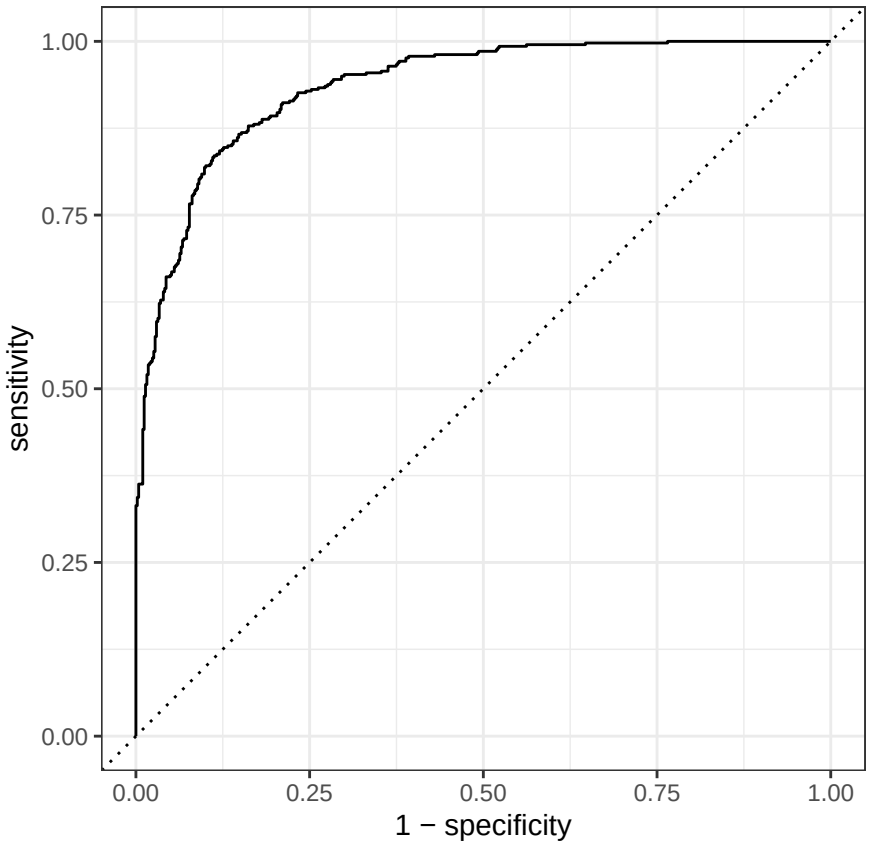
```
customer_recipe <- recipe(customer_status ~ ., data = customer_training) %>%
                step_YeoJohnson(all_numeric(), -all_outcomes()) %>%
                step_normalize(all_numeric(), -all_outcomes()) %>%
                step_dummy(all_nominal(), -all_outcomes()) %>% prep()
```

Model 1 (Logistic Regression)

```
logistic_model <- logistic_reg() %>%
                  set_engine('glm') %>%
                  set_mode('classification')
customer_wf <- workflow() %>%
            add_model(logistic_model) %>%
            add_recipe(customer_recipe)
customer_logistic_fit <- customer_wf %>%
                     fit(data = customer_training)

predictions_categories <- predict(customer_logistic_fit,
                                  new_data = customer_test)

predictions_probabilities <- predict(customer_logistic_fit,
                                     new_data = customer_test,
                                     type = 'prob')


test_results <- customer_test %>% select(customer_status) %>%
                bind_cols(predictions_categories) %>%
                bind_cols(predictions_probabilities)
my_metrics <- metric_set(accuracy, sens, spec, f_meas, roc_auc)

last_fit_model <- customer_wf %>%
                  last_fit(split = customer_split,
                           metrics = my_metrics)
last_fit_results <- last_fit_model %>%
                    collect_predictions()
last_fit_model %>% collect_metrics()
```

```
## # A tibble: 5 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.859 Preprocessor1_Model1
## 2 sens     binary         0.857 Preprocessor1_Model1
## 3 spec     binary         0.860 Preprocessor1_Model1
## 4 f_meas   binary         0.846 Preprocessor1_Model1
## 5 roc_auc  binary         0.935 Preprocessor1_Model1
```

```
last_fit_results %>%
  roc_curve(truth = customer_status, estimate = .pred_closed_account) %>%
  autoplot()
```

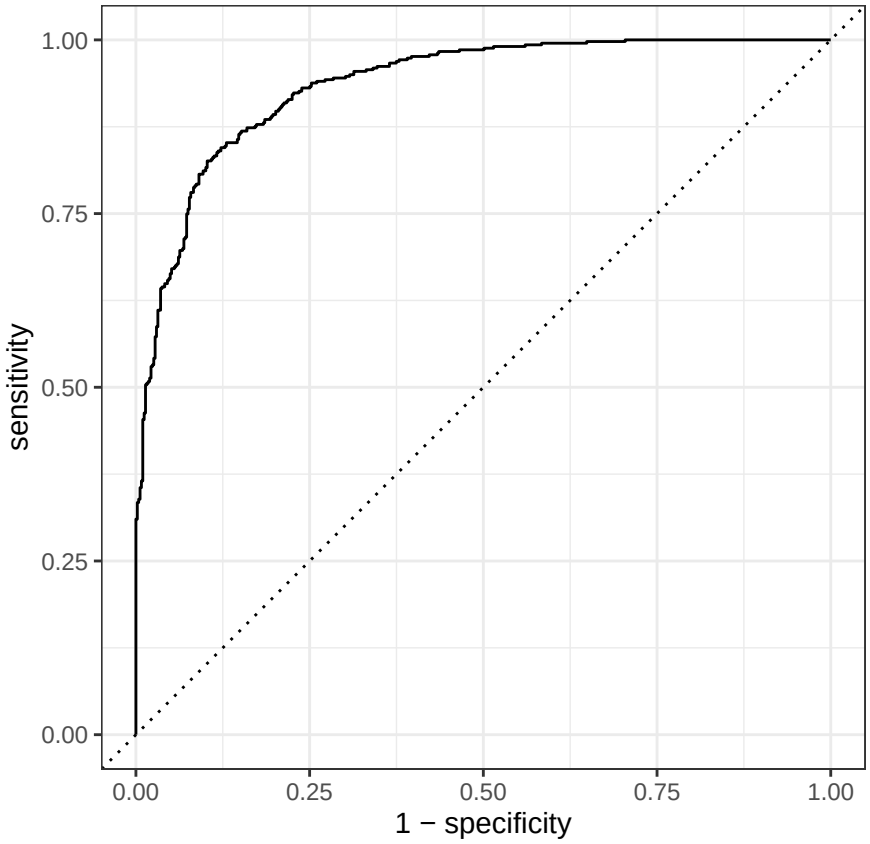


Model 2 (LDA)

```
lda_model <- discrim_regularized(frac_common_cov = 1) %>%
             set_engine('klaR') %>%
             set_mode('classification')

lda_wf <- workflow() %>%
          add_model(lda_model) %>%
          add_recipe(customer_recipe)

last_fit_lda <- lda_wf %>%
                last_fit(split = customer_split)
lda_predictions <- last_fit_lda %>%
                     collect_predictions()
last_fit_lda %>% collect_metrics()
```

```
## # A tibble: 2 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.861 Preprocessor1_Model1
## 2 roc_auc  binary         0.936 Preprocessor1_Model1
```

```
lda_predictions %>%
  roc_curve(truth = customer_status, .pred_closed_account) %>%
  autoplot()
```

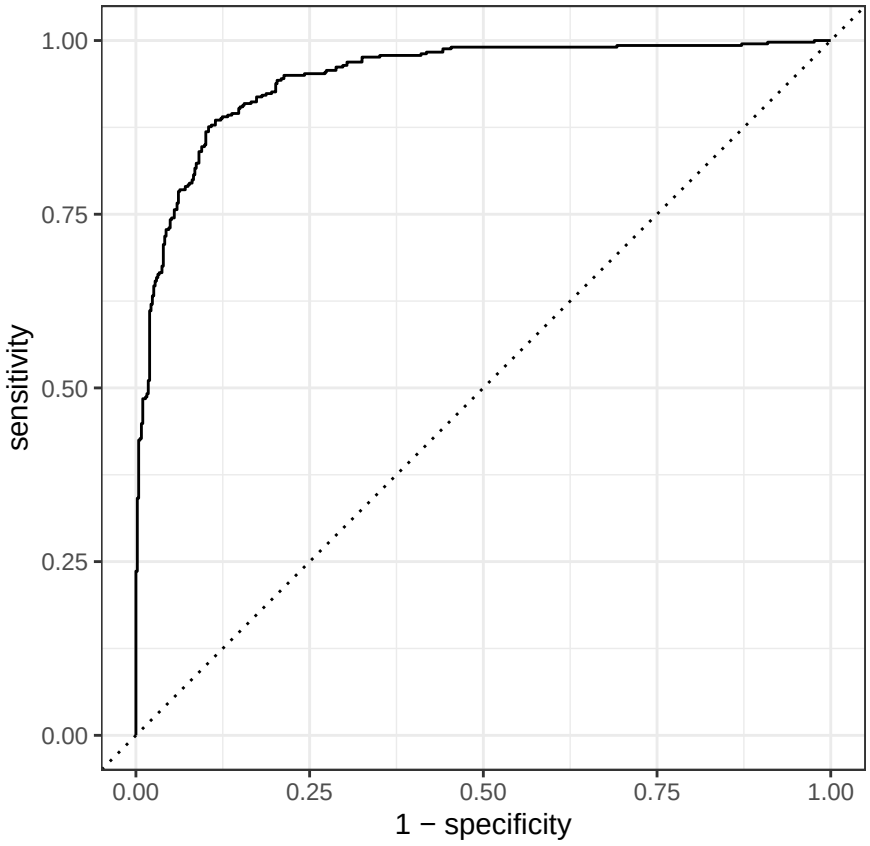


Model 3 (QDA)

```
qda_model <- discrim_regularized(frac_common_cov = 0) %>%
             set_engine('klaR') %>%
             set_mode('classification')
qda_wf <- workflow() %>%
          add_model(qda_model) %>%
          add_recipe(customer_recipe)
last_fit_qda <- qda_wf %>%
                last_fit(split = customer_split)
last_fit_qda %>% collect_metrics()
```

```
## # A tibble: 2 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.882 Preprocessor1_Model1
## 2 roc_auc  binary         0.947 Preprocessor1_Model1
```

```
qda_predictions <- last_fit_qda %>%
                     collect_predictions()

qda_predictions %>%
  roc_curve(truth = customer_status, .pred_closed_account) %>%
  autoplot()
```



Summary of Results

Write a summary of your overall findings and recommendations to the executives at the bank. Think of this section as your closing remarks of a presentation, where you summarize your key findings, model performance, and make recommendations to improve customer retention and service at the bank.

Your executive summary must be written in a professional tone, with minimal grammatical errors, and should include the following sections:

1. An introduction where you explain the business problem and goals of your data analysis

The banks want to investigate the reason and traits of their customers that have an active account and for those whose account got closed, the bank wants to use machine learning to create a system that would help them in advance to see if an account is going to be closed soon so that they can take necessary actions to avoid losing their valuable customers to other banks. The goal of the analysis was to find out the Similarities and relationships of the customer account status with their attributes like Average monthly balance to credit

limit, income, expenditure, credit limit, and education level. The questions aimed to give answers and insights to the customer trait that is most closely related to the account status so an understanding of the traits with the customer behavior is of paramount importance.

2. Highlights and key findings from your Exploratory Data Analysis section

    - There is a very thin line between the salary a person is getting with their account status, the attributes are almost identical in this scenario, so a feature that sought to be important with the account closure turned out to be not that helpful, this also reflects great information that a person salary has a very little to do with their account status.
    - There is a strong relationship between the Average monthly balance to credit limit with the customer status, the closed account seems to have lesser value as compared to the active amount, in fact the average value for active customer is equal to the 75 percent of the closed customer, so the bank must pay close attention to such user with extreme value of utlization ratio.
    - The income and credit limit are closely aligned with each other, meaning that they have a direct effect on each other, the customers with active account tend to have lower income and credit limit as compared to the inactive account, this is maybe due to the fact that they are regular customers of the bank and spend their income on their needs, the bank should try to facilitate their active customers by offering them discounts on their expenditures.
    - The total transaction of existing customs has a huge difference between the inactive customers. Inactive customers distribution is more positively skewed and have a mean value lower than 50, other than the existing customers who have a mean value around 75, this means that the inactive customer empty their accounts before being inactive, this raises questions on the bank polices and their behaviour, are they moving to a new bank or is there some other reason behind this, this figure is alarming.
    - The level of education shows that the customers with a masters degree tend to be the best customers of the bank followed by the customers with associate degree, so level of education affects the customer decision of sticking with the bank.

3. Your "best" classification model and an analysis of its performance

The QDA model performed exceptionally well on the test set (new data) with an error rate of around 5 %, so the predictions are made with around 95 percent of confidence with a very low error percentage of 5 percent, the performance is judged on the AUC metric which is also knows as the Area Under the Curve that defines the model's ability to distinguish between outcomes, as mentioned earlier that their is very thin line when seen through human eye between the income of the customer with their account status but this is no problem for our best trained classification model.

4. Your recommendations to the bank on how to reduce the number of customers closing their credit card accounts

- The bank should focus on the customers that have alarming value for utilization ratio and should try to offer them better discounts on their shopping habits.
- The bank should also collect data regarding the customer shopping habits to make targeted marketing campaigns to the customers who spend a lot because they are the ones that are at a risk of closing their accounts.
- The bank should pay attention to their customers with highest transaction ratio for the past year.
- The bank should try to establish good relationships with their most spending and educated people .

Summary

The machine learning models can certainly help the banking sector in deciding their next move and getting warning about a potential customer who can close their account based on the predictions generated by the

model, so it is safe to say that the Machine learning models combined with the knowledge obtained from the data analysis can really help formulate the strategy for maximization profits for the banks. The insights from the data suggested that when seen through the human eye there is a very small line between the income of the client and their status of the account but rigorous machine learning models can learn on those features too to give a very good quality predictions. Another fact that is witnessed was that utilization ratio really affects the status of the account. All of the models gave good predictions meaning that there is still room for improvement in the modelling to achieve even higher possible accuracy on the test set. The AUC of 95 indicates that if new data is given to the model it would give the predictions with only 5 percent chance of an error in other words with 95 percent of accuracy.