Machine Learning Project

Name: Sai Jaswanth Kumar Kunku
G Number : G01331645

Library Import

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

library(tidyverse)
library(tidymodels)
library(vip)
library(discrim)
library(ggcorrplot)
library(rpart.plot)
library(psych)
library(gridExtra)
```

Raw Data

```
credit_card_df <- readRDS(url('https://gmubusinessanalytics.netlify.app/data/credit_card_df.rds'))
summary(credit_card_df)
```

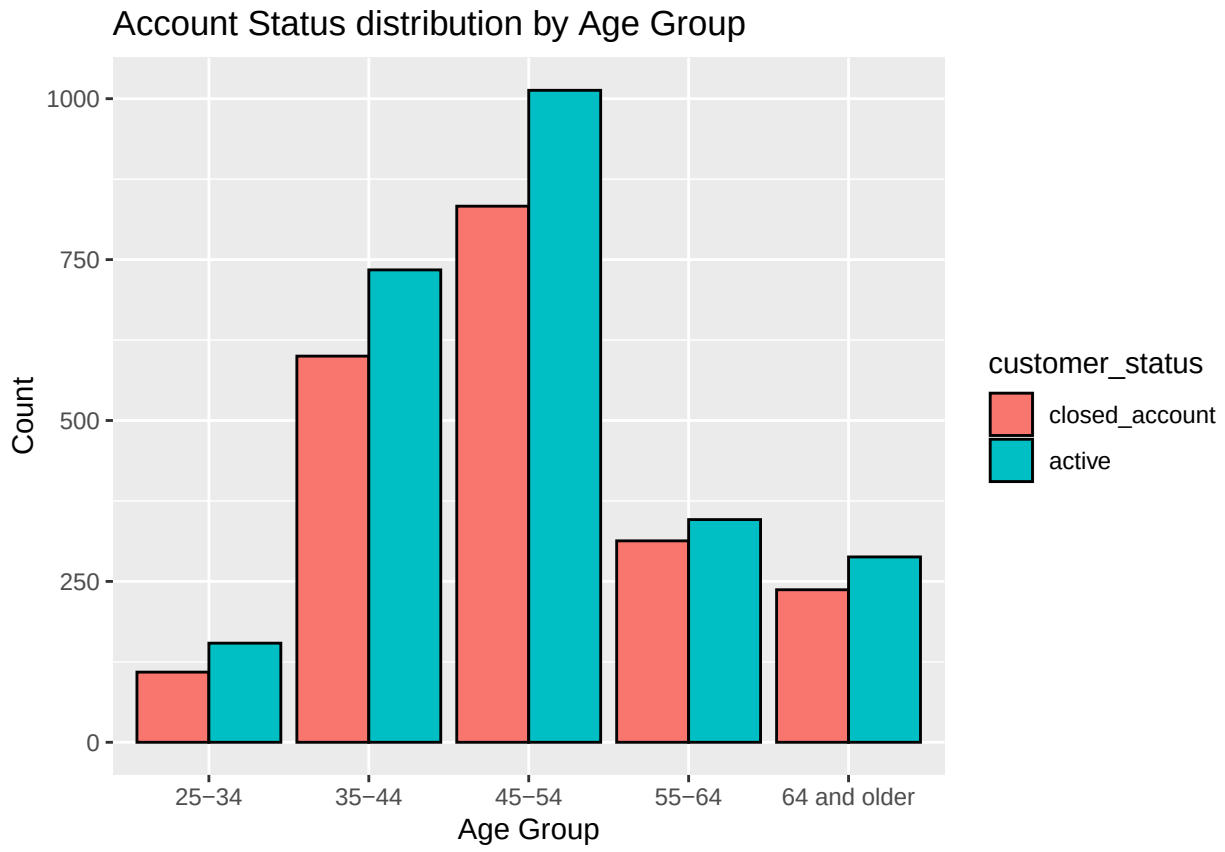
```
##        customer_status      age          dependents         education
##  closed_account:2092   Min.   :26.00   Min.   :0.000   associates:1927
##  active        :2535   1st Qu.:41.00   1st Qu.:1.000   bachelors : 469
##                        Median :46.00   Median :2.000   masters   :1995
##                        Mean   :46.37   Mean   :2.379   doctorate : 236
##                        3rd Qu.:52.00   3rd Qu.:3.000
##                        Max.   :70.00   Max.   :5.000
##   marital_status     employment_status     income        card_type
##  single  :2007   full_time    :2390    Min.   : 30094   blue  :2551
##  married :2266   part_time    :1527    1st Qu.: 35423   silver:1168
##  divorced: 354   self_employed: 710    Median : 49184   gold  : 908
##                                        Mean   : 62282
##                                        3rd Qu.: 79796
##                                        Max.   :168522
##  months_since_first_account total_accounts  months_inactive_last_year
##  Min.   :13.00              Min.   :1.000   Min.   :0.000
##  1st Qu.:32.00              1st Qu.:2.000   1st Qu.:2.000
##  Median :36.00              Median :4.000   Median :3.000
##  Mean   :36.06              Mean   :3.645   Mean   :2.456
##  3rd Qu.:40.00              3rd Qu.:5.000   3rd Qu.:3.000
##  Max.   :56.00              Max.   :6.000   Max.   :6.000
##  contacted_last_year  credit_limit   utilization_ratio spend_ratio_q4_q1
```

```
##  Min.   :0.000       Min.   : 1430   Min.   :0.0000    Min.   :0.0000
##  1st Qu.:2.000       1st Qu.: 2440   1st Qu.:0.0000    1st Qu.:0.6020
##  Median :3.000       Median : 4500   Median :0.1050    Median :0.7230
##  Mean   :2.656       Mean   : 8501   Mean   :0.2314    Mean   :0.7368
##  3rd Qu.:3.000       3rd Qu.:10626   3rd Qu.:0.4180    3rd Qu.:0.8590
##  Max.   :6.000       Max.   :34516   Max.   :0.9990    Max.   :2.2820
##  total_spend_last_year transactions_last_year transaction_ratio_q4_q1
##  Min.   :  510         Min.   : 10.00         Min.   :0.0000
##  1st Qu.: 2021         1st Qu.: 40.00         1st Qu.:0.5000
##  Median : 2753         Median : 55.00         Median :0.6500
##  Mean   : 3929         Mean   : 57.87         Mean   :0.6548
##  3rd Qu.: 4613         3rd Qu.: 74.00         3rd Qu.:0.7840
##  Max.   :17498         Max.   :131.00         Max.   :3.0000
```

Question 1: What is the distribution of open and closed accounts for different age groups?

Answer:The frequency of customers who are closing their accounts are high in the age groups between 35-44 and 45-54 respectively.This is the age group that has more number of active accounts as well.

```
credit_card_df <- credit_card_df %>%
  mutate(age_group = case_when(
    age >= 25 & age <34 ~ '25-34',
    age >= 35  & age <44 ~ '35-44',
    age >= 45 & age <54 ~ '45-54',
    age >= 55 & age <64 ~ '55-64',
    TRUE ~ '64 and older'))

df1<-credit_card_df %>% group_by(age_group,customer_status) %>%
summarize(total_count=n()) %>%
mutate(percentage =  round(100*(total_count/sum(total_count)),2))

ggplot(df1,aes(x=age_group,y=total_count,fill=customer_status))+
  geom_bar(stat="identity",color="black",position = "dodge") +
  labs(title = "Account Status distribution by Age Group",
              x = "Age Group",
              y = "Count")
```



Question 2:How the income levels of different customers vary by their employment status and marital status?

Answer:The median income of full-time,part-time and self-employees customers lies between 40000 to 50000 and there is no noticeable differences in active vs closed accounts for married and unmarried status.

```
fig1<-credit_card_df %>% ggplot(aes(x=employment_status,
                          y=income,
                          fill=customer_status)) +
geom_boxplot()  +  labs(title = "Summary Statistics of Income by Employement Type", y = "Income",
         x = "Employement Status")

fig2<-credit_card_df %>% ggplot(aes(x=marital_status,
                          y=income,
                          fill=customer_status)) +
geom_boxplot()  +  labs(title = "Summary Statistics of Income by Marital Status", y = "Income",
         x = "Marital Status")

grid.arrange(fig1, fig2)
```

Question 3: What percentage of active and churned customers with respect to their employment status

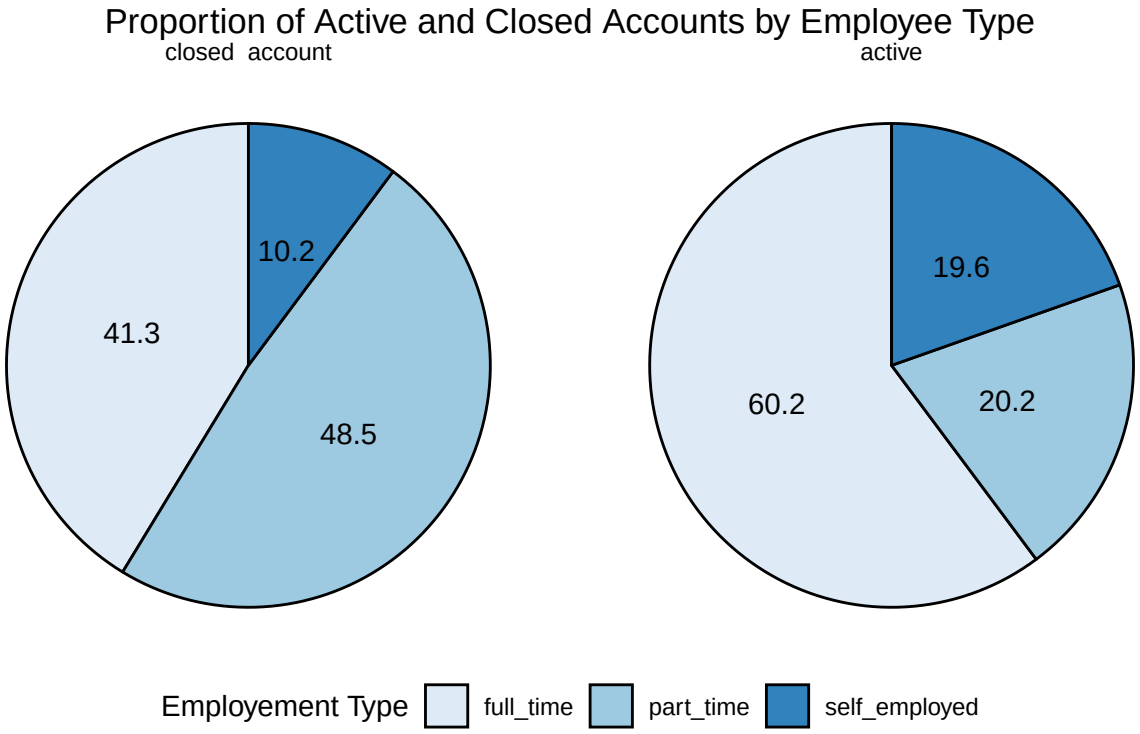
Answer:The results from pie chart shows that 90% of closed accounts are from part-time and full-time employees.It should be noticed that the proportion of closed accounts is 28% higher than that of corresponding active accounts.

```
emp_data<-credit_card_df %>% group_by(customer_status,employment_status) %>% summarize(count=n()) %>%  m

emp_data
```

```
## # A tibble: 6 x 4
## # Groups:   customer_status [2]
##   customer_status employment_status count percentage
##   <fct>           <fct>             <int>      <dbl>
## 1 closed_account  full_time           865       41.3
## 2 closed_account  part_time          1014       48.5
## 3 closed_account  self_employed       213       10.2
## 4 active          full_time          1525       60.2
## 5 active          part_time           513       20.2
## 6 active          self_employed       497       19.6
```

```
ggplot(emp_data, aes(x = "", y = percentage, fill = employment_status)) +
  geom_col(color = "black") +guides(fill=guide_legend(title="Employement Type"))+
  geom_text(aes(label = percentage),
            position = position_stack(vjust = 0.5)) +
```

```
  coord_polar(theta = "y")+theme_void() +  scale_fill_brewer() +facet_wrap(~customer_status)+
  ggtitle("Proportion of Active and Closed Accounts by Employee Type")+
  theme(plot.title = element_text(hjust = 0.5),
        legend.position = "bottom",panel.spacing = unit(1, "lines"))
```



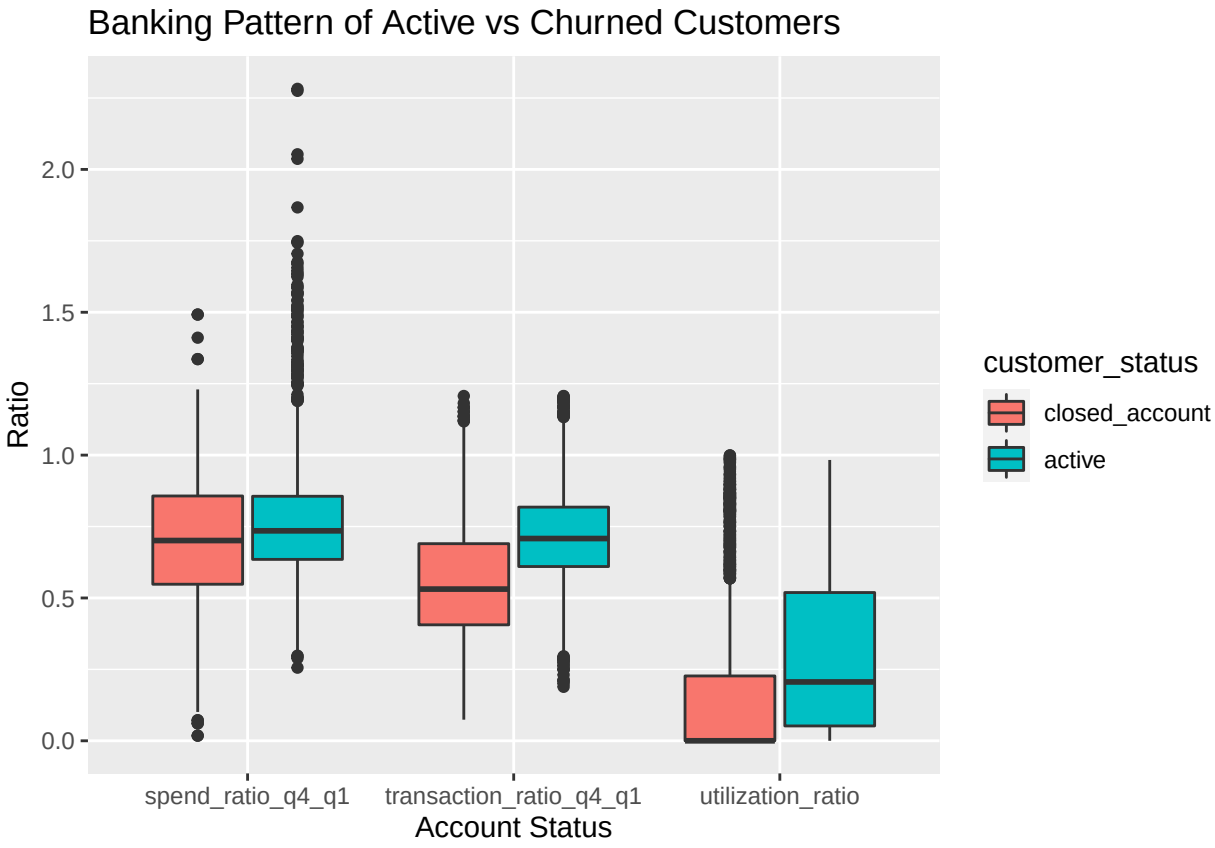
Question 4: How does the Transaction ratio,number of transactions and spend ratio vary among different customers?

Answer:The average transaction ratio of active accounts is 0.74 whereas for the closed accounts is 0.56.The spending ratio of both active and closed accounts is almost identical with values at 0.77 and 0.7 respectively.

```
ratio_data<-credit_card_df %>%
  group_by(customer_status) %>%
  summarise(avg_transaction_ratio = round(mean(transaction_ratio_q4_q1),2),
            total_transactions = round(median(transactions_last_year),2),
            avg_spend_ratio = round(mean(spend_ratio_q4_q1),2))

ratio_data
```

```
## # A tibble: 2 x 4
##   customer_status avg_transaction_ratio total_transactions avg_spend_ratio
##   <fct>                           <dbl>              <dbl>           <dbl>
## 1 closed_account                   0.56                 43             0.7
## 2 active                           0.74                 71             0.77
```

```
outlier1 <- boxplot(credit_card_df$transaction_ratio_q4_q1, plot=FALSE)$out
x<- as.data.frame(credit_card_df[-which(credit_card_df$transaction_ratio_q4_q1 %in% outlier1),])

x %>%
  pivot_longer(cols = c(transaction_ratio_q4_q1, utilization_ratio,spend_ratio_q4_q1),
               names_to = 'value_type',
               values_to = 'Ratio') %>%  ggplot(aes(x=value_type,
               y=Ratio,
               fill=customer_status)) +
  geom_boxplot()+
  labs(title = "Banking Pattern of Active vs Churned Customers",
       x = "Account Status")
```



Question 5: Does no:of dependents has any impact on the status of the customer?

Answer:The increase in number of dependents doesn't show any impact on the customer status as the percentage of closed accounts is always 10-15% lesser than active accounts within each dependent group.

```
credit_card_df %>% group_by(dependents,customer_status) %>% summarize(count=n()) %>%  mutate(percentage
```

```
## # A tibble: 12 x 4
## # Groups:   dependents [6]
##    dependents customer_status count percentage
##         <dbl> <fct>           <int>      <dbl>
```

```
##  1          0 closed_account    178       43.3
##  2          0 active            233       56.7
##  3          1 closed_account    345       42.1
##  4          1 active            475       57.9
##  5          2 closed_account    521       46.0
##  6          2 active            612       54.0
##  7          3 closed_account    630       47.6
##  8          3 active            693       52.4
##  9          4 closed_account    336       44.9
## 10          4 active            413       55.1
## 11          5 closed_account     82       42.9
## 12          5 active            109       57.1
```

Question 6:Which credit card holders are more likely to close their accounts and what is their average credit utilization ratio?

Answer:Blue card holders are more likely to close their accounts as they constitute 71.5% of overall closed accounts.Out of total blue card holders 58.6% has closed their accounts whereas followed by gold and silver card users with 32.9% and 25.3% respectively.

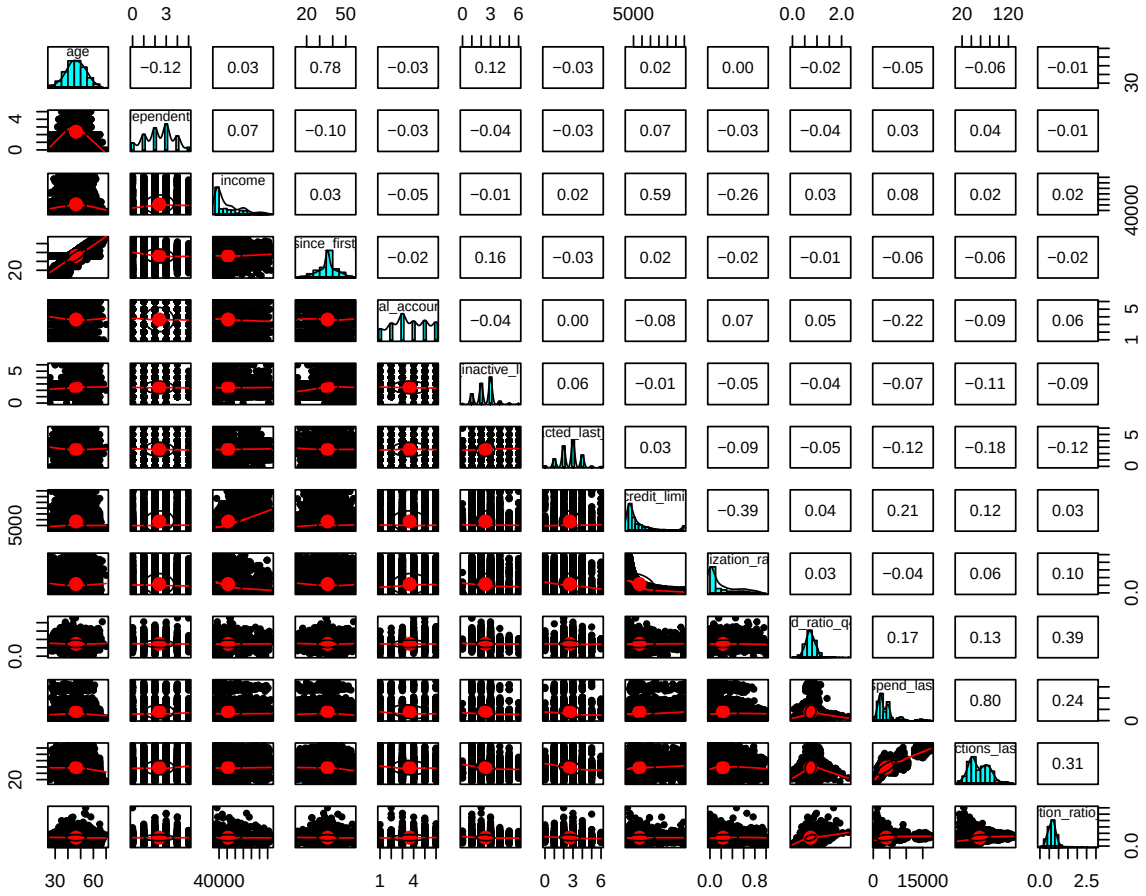
```
credit_card_df %>%
  group_by(card_type,customer_status) %>% summarize(count=n(),avg_utilization=mean(utilization_ratio)) %
  mutate(percentage=round((count/sum(count))*100,2)) %>% arrange(customer_status
) %>% ungroup() %>% group_by(customer_status) %>%
  mutate(prop=round((count/sum(count))*100,2))
```

```
## # A tibble: 6 x 6
## # Groups:   customer_status [2]
##   card_type customer_status count avg_utilization percentage  prop
##   <fct>     <fct>           <int>           <dbl>      <dbl> <dbl>
## 1 blue      closed_account   1497           0.162       58.7  71.6
## 2 silver    closed_account    296           0.159       25.3  14.2
## 3 gold      closed_account    299           0.159       32.9  14.3
## 4 blue      active           1054           0.278       41.3  41.6
## 5 silver    active            872           0.292       74.7  34.4
## 6 gold      active            609           0.306       67.1  24.0
```

Question 7::How are various characterstics related to each other?

Answer:There is positive correlation of 0.59 between the customer income and the credit limit.In addition to that the number of transactions made last year is positively related to the amount spent last year with a correlation of 0.80.On the flipside,credit limit and utilization ratio has negative correaltion of 0.39.

```
numeric_df<-Filter(is.numeric, credit_card_df)
pairs.panels(numeric_df)
```



Machine Learning

Factoring & Data Splitting

```
credit_card_df <- readRDS(url('https://gmubusinessanalytics.netlify.app/data/credit_card_df.rds'))
credit_card_df$total_accounts<-factor(credit_card_df$total_accounts)
credit_card_df$contacted_last_year  <-factor(credit_card_df$total_accounts)
credit_card_df$months_inactive_last_year  <-factor(credit_card_df$months_inactive_last_year)
credit_card_df$years_active<-as.factor(round(credit_card_df$months_since_first_account/12))
credit_card_df$dependents<-factor(credit_card_df$dependents)
credit_card_df <- credit_card_df %>%
  mutate(total_transactions = case_when(
    transactions_last_year <=55 ~ '<55',
    TRUE ~ '>55'))
credit_card_df<-credit_card_df %>% dplyr::select(-c("months_since_first_account","transactions_last_yea

set.seed(1)
#Test/Train Split
credit_split <- initial_split(credit_card_df, prop = 0.75,strata = customer_status)
credit_training <- credit_split %>% training()
```

```
credit_test <- credit_split %>% testing()

set.seed(10)
#Cross Validation
credit_folds <- vfold_cv(credit_training, v = 5)
#Metrics
my_metrics <- metric_set(accuracy, sens, spec, f_meas, roc_auc)
```

Feature Engineering

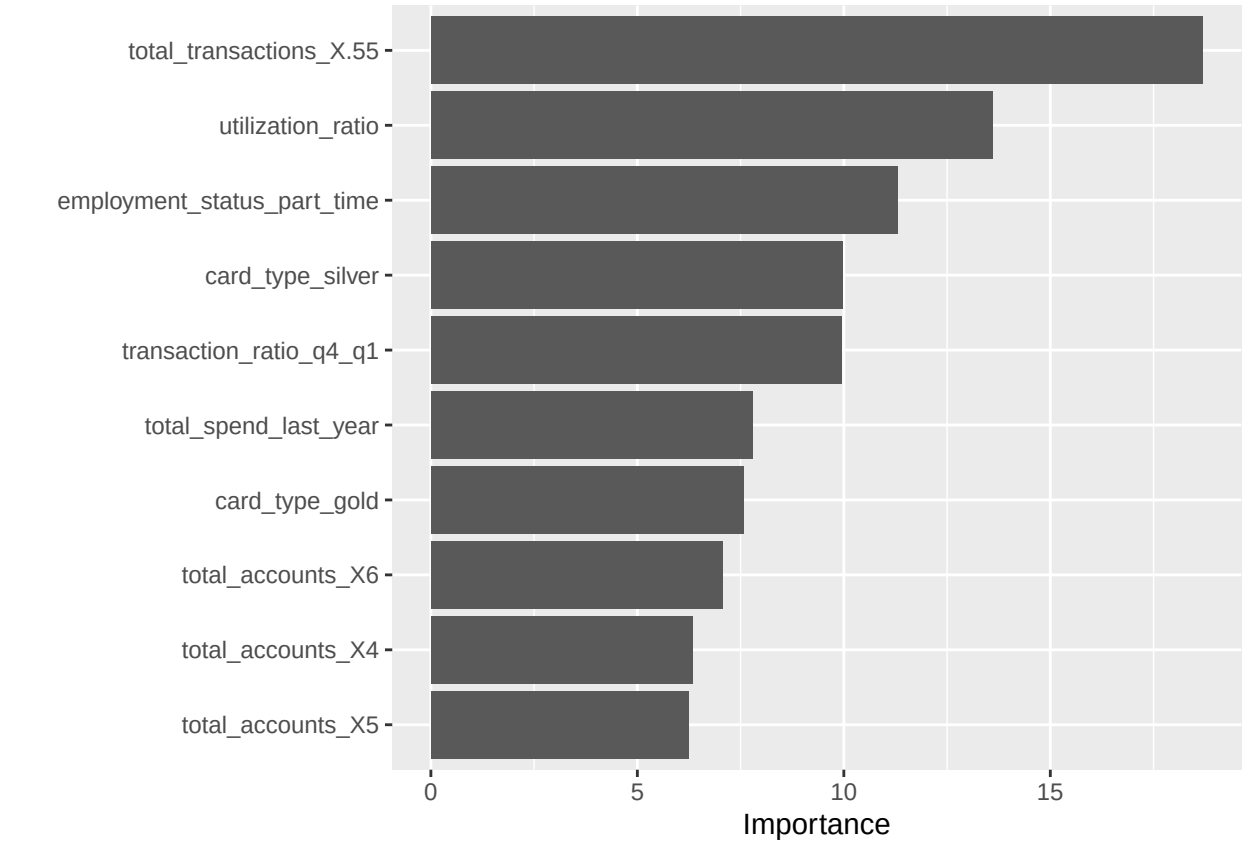
```
credit_recipe <- recipe(customer_status ~ ., data = credit_training) %>%
                  step_YeoJohnson(all_numeric(), -all_outcomes()) %>%
                  step_normalize(all_numeric(), -all_outcomes()) %>%
                  step_dummy(all_nominal(), -all_outcomes())

credit_recipe %>%
  prep(training = credit_training) %>%
  bake(new_data = NULL)
```

```
## # A tibble: 3,470 x 42
##    income credit_limit utilization_ratio spend_ratio_q4_q1 total_spend_last_year
##     <dbl>        <dbl>             <dbl>             <dbl>                 <dbl>
##  1 -0.998      -0.175             0.0712            0.237                 0.722
##  2  0.723       0.135             1.06             -0.250                 0.645
##  3 -0.822      -0.0564            1.11             -0.0840               -0.0448
##  4  1.25        1.27             -0.653            -0.0280               -0.796
##  5  0.662       0.905             0.297             0.532                 1.40
##  6 -0.906      -1.59              1.26             -0.337                 0.737
##  7 -0.810      -0.651             1.67              0.331                 0.596
##  8  1.28        0.555            -1.03             -0.613                 0.676
##  9 -1.15       -0.673            -1.03             -0.356                 0.582
## 10 -0.946      -0.0977            0.835             0.300                 0.366
## # ... with 3,460 more rows, and 37 more variables:
## #   transaction_ratio_q4_q1 <dbl>, customer_status <fct>, dependents_X1 <dbl>,
## #   dependents_X2 <dbl>, dependents_X3 <dbl>, dependents_X4 <dbl>,
## #   dependents_X5 <dbl>, education_bachelors <dbl>, education_masters <dbl>,
## #   education_doctorate <dbl>, marital_status_married <dbl>,
## #   marital_status_divorced <dbl>, employment_status_part_time <dbl>,
## #   employment_status_self_employed <dbl>, card_type_silver <dbl>, ...
```

Model 1 :Logistic Regression

```
#Model Spec
logistic_model <- logistic_reg() %>%
                  set_engine('glm') %>%
                  set_mode('classification')

credit_wf <- workflow() %>%
             add_model(logistic_model) %>%
             add_recipe(credit_recipe)
```

```
credit_logistic_fit <- credit_wf %>%
                       fit(data = credit_training)

credit_trained_model <- credit_logistic_fit %>%
                        extract_fit_parsnip()

vip(credit_trained_model)
```




```
last_fit_model <- credit_wf %>%
                  last_fit(split = credit_split,
                           metrics = my_metrics)
```

```
## ! train/test split: preprocessor 1/1, model 1/1 (predictions): prediction from a rank-defici...
```

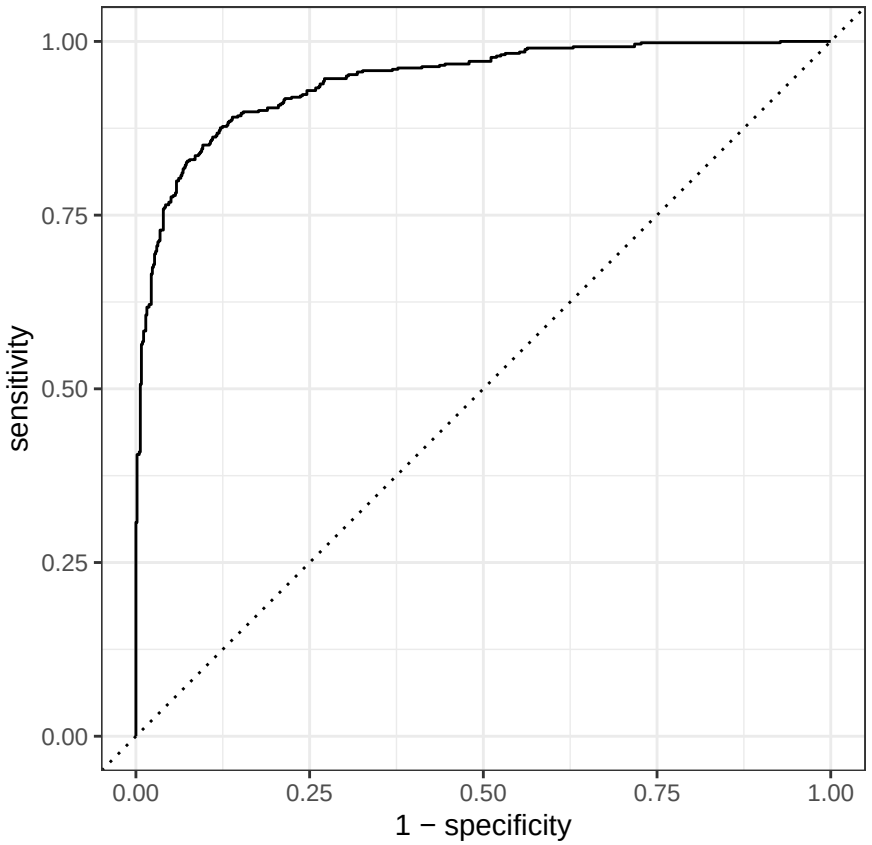
```
metrics<-last_fit_model %>% collect_metrics()

metrics
```

```
## # A tibble: 5 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.876 Preprocessor1_Model1
## 2 sens     binary         0.872 Preprocessor1_Model1
## 3 spec     binary         0.880 Preprocessor1_Model1
## 4 f_meas   binary         0.864 Preprocessor1_Model1
## 5 roc_auc  binary         0.944 Preprocessor1_Model1
```

```
last_fit_results <- last_fit_model %>%  collect_predictions()

last_fit_results
```

```
## # A tibble: 1,157 x 7
##    id    .pred_class  .row .pred_closed_ac~ .pred_active customer_status .config
##    <chr> <fct>       <int>            <dbl>        <dbl> <fct>           <chr>
##  1 trai~ closed_acc~     2           0.978       0.0223  closed_account  Prepro~
##  2 trai~ closed_acc~     8           0.990       0.00983 closed_account  Prepro~
##  3 trai~ active          9           0.0423      0.958   active          Prepro~
##  4 trai~ closed_acc~    12           0.601       0.399   closed_account  Prepro~
##  5 trai~ closed_acc~    16           0.644       0.356   closed_account  Prepro~
##  6 trai~ active         18           0.101       0.899   active          Prepro~
##  7 trai~ active         21           0.302       0.698   closed_account  Prepro~
##  8 trai~ active         24           0.248       0.752   active          Prepro~
##  9 trai~ closed_acc~    27           0.908       0.0921  closed_account  Prepro~
## 10 trai~ closed_acc~    37           0.512       0.488   closed_account  Prepro~
## # ... with 1,147 more rows
```

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




```
conf_mat(last_fit_results, truth = customer_status, estimate = .pred_class)
```

```
##                 Truth
## Prediction       closed_account active
```

```
##   closed_account            456     76
##   active                     67    558
```

Model 2 :KNN

```
knn_model <- nearest_neighbor(neighbors = tune()) %>%
             set_engine('kknn') %>%
             set_mode('classification')

knn_wf <- workflow() %>%
          add_model(knn_model) %>%
          add_recipe(credit_recipe)

k_grid <- tibble(neighbors = c(10, 20, 30, 50, 70, 100))

## Tune  workflow
set.seed(10)

knn_tuning <- knn_wf %>%
              tune_grid(resamples = credit_folds,
                        grid = k_grid)
```

```
## Warning: package 'kknn' was built under R version 4.1.2
```

```
knn_tuning %>% collect_metrics()
```

```
## # A tibble: 12 x 7
##    neighbors .metric  .estimator  mean     n std_err .config
##        <dbl> <chr>    <chr>      <dbl> <int>   <dbl> <chr>
##  1        10 accuracy binary     0.786     5 0.00731 Preprocessor1_Model1
##  2        10 roc_auc  binary     0.880     5 0.00624 Preprocessor1_Model1
##  3        20 accuracy binary     0.8       5 0.00999 Preprocessor1_Model2
##  4        20 roc_auc  binary     0.895     5 0.00744 Preprocessor1_Model2
##  5        30 accuracy binary     0.809     5 0.00811 Preprocessor1_Model3
##  6        30 roc_auc  binary     0.899     5 0.00836 Preprocessor1_Model3
##  7        50 accuracy binary     0.820     5 0.0106  Preprocessor1_Model4
##  8        50 roc_auc  binary     0.903     5 0.00918 Preprocessor1_Model4
##  9        70 accuracy binary     0.821     5 0.00977 Preprocessor1_Model5
## 10        70 roc_auc  binary     0.904     5 0.00934 Preprocessor1_Model5
## 11       100 accuracy binary     0.823     5 0.00949 Preprocessor1_Model6
## 12       100 roc_auc  binary     0.906     5 0.00875 Preprocessor1_Model6
```

```
best_k <- knn_tuning %>% select_best(metric = 'roc_auc')

final_knn_wf <- knn_wf %>% finalize_workflow(best_k)

last_fit_knn <- final_knn_wf %>% last_fit(split = credit_split,metrics=my_metrics)

last_fit_knn %>% collect_metrics()
```

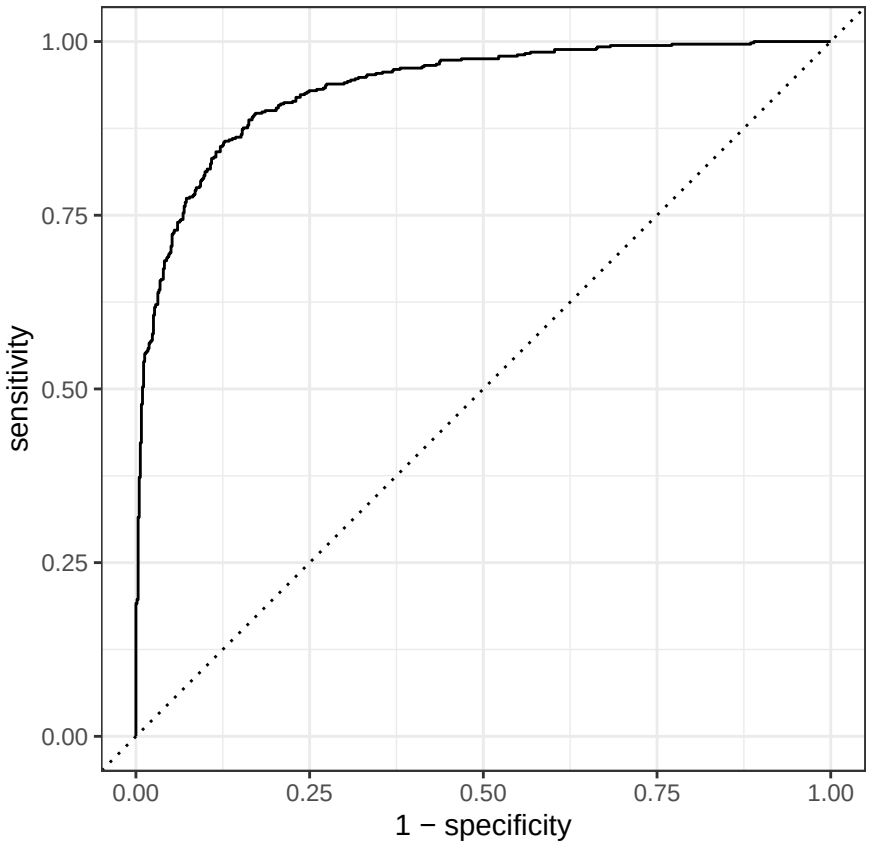
```
## # A tibble: 5 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.862 Preprocessor1_Model1
## 2 sens     binary         0.834 Preprocessor1_Model1
```

```
## 3 spec     binary         0.885 Preprocessor1_Model1
## 4 f_meas   binary         0.845 Preprocessor1_Model1
## 5 roc_auc  binary         0.934 Preprocessor1_Model1
```

```
knn_predictions <- last_fit_knn %>% collect_predictions()

knn_predictions
```

```
## # A tibble: 1,157 x 7
##    id    .pred_class  .row .pred_closed_ac~ .pred_active customer_status .config
##    <chr> <fct>       <int>            <dbl>        <dbl> <fct>           <chr>
##  1 trai~ closed_acc~     2            0.928       0.0720 closed_account  Prepro~
##  2 trai~ closed_acc~     8            0.728       0.272  closed_account  Prepro~
##  3 trai~ active          9            0.255       0.745  active          Prepro~
##  4 trai~ closed_acc~    12            0.575       0.425  closed_account  Prepro~
##  5 trai~ closed_acc~    16            0.600       0.400  closed_account  Prepro~
##  6 trai~ active         18            0.175       0.825  active          Prepro~
##  7 trai~ closed_acc~    21            0.698       0.302  closed_account  Prepro~
##  8 trai~ active         24            0.379       0.621  active          Prepro~
##  9 trai~ closed_acc~    27            0.678       0.322  closed_account  Prepro~
## 10 trai~ closed_acc~    37            0.674       0.326  closed_account  Prepro~
## # ... with 1,147 more rows
```

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



```
conf_mat(knn_predictions, truth = customer_status, estimate = .pred_class)
```

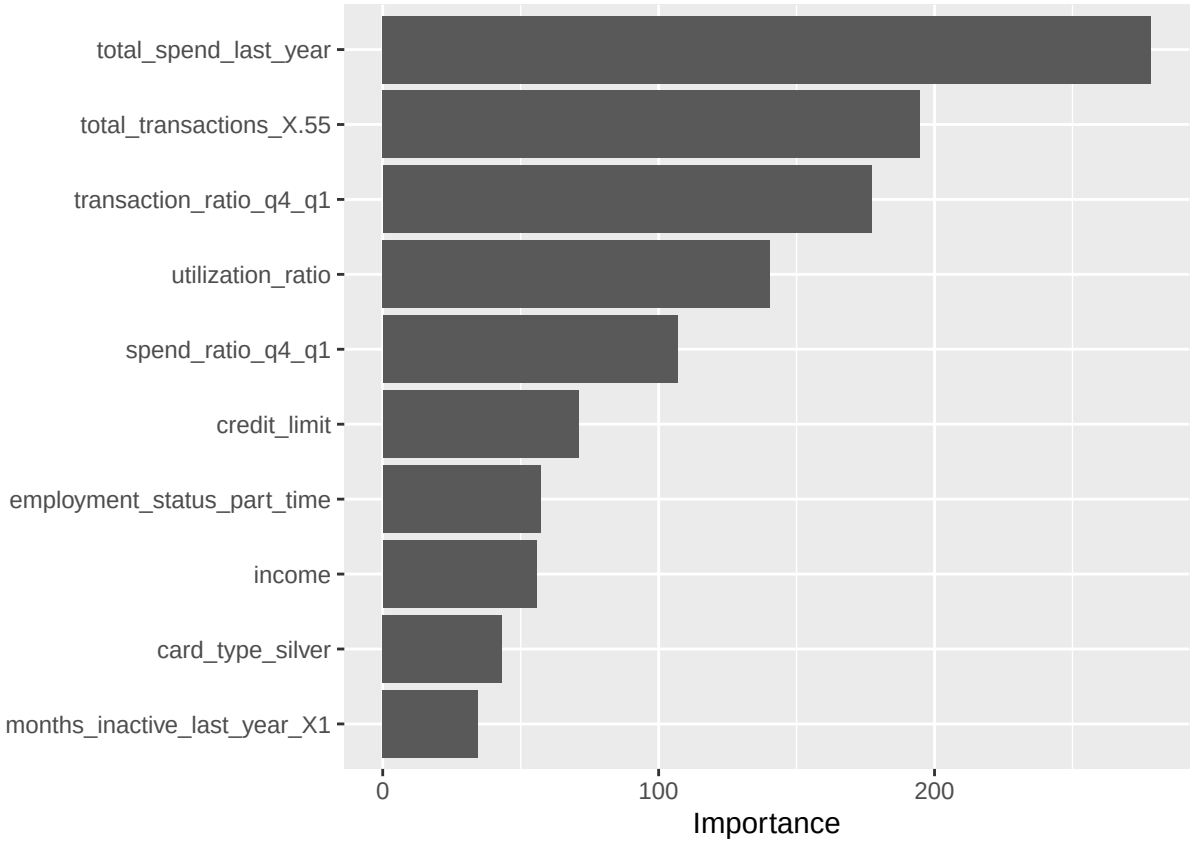
```
##                 Truth
## Prediction       closed_account active
##   closed_account            436     73
##   active                     87    561
```

Model 3 :Random Forest

```
rf_model <- rand_forest(mtry = tune(),
                        trees = tune(),
                        min_n = tune()) %>%
            set_engine('ranger', importance = "impurity") %>%
            set_mode('classification')

rf_workflow <- workflow() %>%
               add_model(rf_model) %>%
               add_recipe(credit_recipe)

set.seed(10)

rf_grid <- grid_random(mtry() %>% range_set(c(2, round(sqrt(ncol(credit_training))))),
                       trees(),
                       min_n(),
                       size = 9)

set.seed(10)

rf_tuning <- rf_workflow %>%
             tune_grid(resamples = credit_folds,
                       grid = rf_grid)
```

```
## Warning: package 'ranger' was built under R version 4.1.2
```

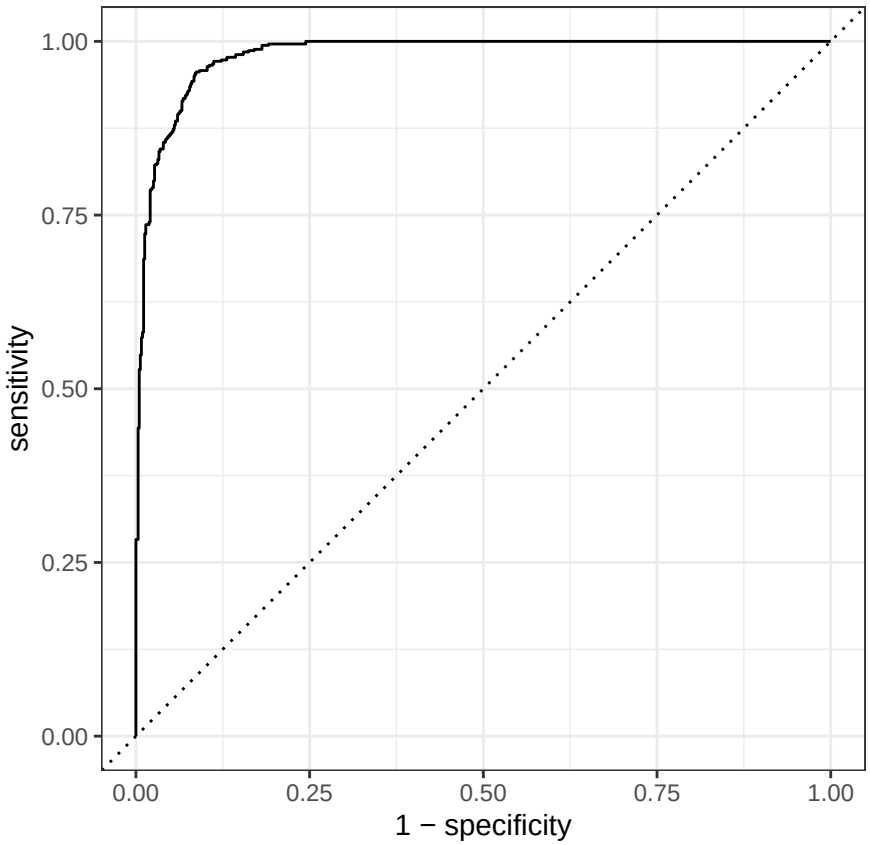
```
best_rf <- rf_tuning %>%
           select_best(metric = 'roc_auc')

final_rf_workflow <- rf_workflow %>%
                     finalize_workflow(best_rf)

rf_last_fit <- final_rf_workflow %>%
               last_fit(split = credit_split,metrics=my_metrics)

credit_rf_fit <- final_rf_workflow %>%
                      fit(data = credit_training)

rf_trained_model <- credit_rf_fit %>%
                        extract_fit_parsnip()

vip(rf_trained_model)
```




```
metrics<-rf_last_fit %>% collect_metrics()

metrics
```

```
## # A tibble: 5 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy binary         0.927 Preprocessor1_Model1
## 2 sens     binary         0.931 Preprocessor1_Model1
## 3 spec     binary         0.923 Preprocessor1_Model1
## 4 f_meas   binary         0.920 Preprocessor1_Model1
## 5 roc_auc  binary         0.981 Preprocessor1_Model1
```

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




```
rf_last_fit %>% collect_predictions() %>% conf_mat(truth = customer_status, estimate = .pred_class)
```

```
##                 Truth
## Prediction       closed_account active
##   closed_account            487     49
##   active                     36    585
```

Summary of Results

Introduction

Banking sector is one of the important sectors in financial paradigm as it plays an important role in our everyday life and in achieving economic growth.The number of customers closing their credit account is a concerning issue to banks as it affects their business and impacts overall profitability.Hence it is important for them to find effective ways to retain existing customers and prevent them from closing their accounts.The goal of this analysis to identify the key factors that are driving the customers to close their accounts,predict the likelihood of them and provide recommendations to prevent it from happening in the future.

Key Findings

Some of the key findings from my analysis reveal that majority of account closures are happening among the customers with an age group between 35 to 55.It is also worth noticing that these are the age groups with more active accounts.When it comes to the employment status,48.5% of customers who close their accounts are working part-time followed by full-time employees with 41% despite having 60% and 20% of active accounts respectively.The median income of these people is around 40,000 to 50,000 USD and is not

explaining anything related to the customer status.The same applies to customers with different marital status customers.While the spending ratio of two different customers is around 0.7 ;the transaction ratio of customers who are closing their account is 0.18 lower than that of active customers(0.75).The total transactions of closed account customers are at 43 while this number is 71 for active users.It is surprising to observe that the number of dependents doesn't seem to have any effect on the status of the customer.Finally the type of credit which a customer uses clearly distinguishes the active accounts from the closed ones. 58% of account closures are from blue cards users followed by gold card and silver card users with 32.9% and 25.3% respectively.Finally,the total number of blue cards customers constitute about 70% of overall closed accounts.

Best Classification Model

In order to predict the likelihood of account closures, models such as Logistic Regression,KNN and Random Forest models has been implemented.Among these three models,Random Forest model performed so well on the unforeseen data.This model was able to predict the a customer will close his/her account 93.8% of the time leaving an error rate of 7.2%.It is also capable of distinguishing active and closed account customers 98.5% of time while the other models were struggling at 94% and 93% respectively.The model also tells that total amount spend last year,total transactions,transaction ratio,utilization ratio and spend ratio are the important variables in determining the target customer status.

Recommendations

Based on the results from my analysis I would recommend bank executives to consider following steps in order to reduce the number of customers from closing their credit card accounts.Since,customers between ages 35-55 are more prone to close their accounts the bank officials should target these groups by understanding their problems through continuous feedback.Since,there is 90% of account closures are from full-time and part-time customers ,there is a high possibility that they are dissatisfied with the banking functions such as delay in payroll processing/check deposits etc.To overcome this, the bank account should frequency reach out to them through telephonic surveys with proper questionnaires.Since majority of account closures are from blue cards users when compared to other card holders they should work on them credit card offers in the form of discounts and cash backs.The banks should periodically track the customer transaction and spending ratio and observe abnormalities in the banking activity.Based on the predictions obtained from the model,the banks should focus on the customers who are likely to close their accounts and work on them by re-engineering their sales, service, and marketing strategies.If the banks were able to successfully work on all these recommendations they can improve their customer retention and overall brand reputation.