

# RandomForest\_Model.R

rahul

2024-11-11

```
#Required libraries
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.3.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
## Loading required package: lattice
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.3.3
```

```
## randomForest 4.7-1.2
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggplot2)
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 4.3.3
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.3.3
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:randomForest':
##
##     combine

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(writexl)
```

```
## Warning: package 'writexl' was built under R version 4.3.3
```

```
#Loading Data
data <- read_excel("ACCT_Monitoring_FinalData.xlsx")

#Data Preprocessing - Conversion of categorical variables to factors
data$STATE <- as.factor(data$STATE)
data$LOCK_CODE <- as.factor(data$LOCK_CODE)
data$LOCK_REASON <- as.factor(data$LOCK_REASON)
data$PORTFOLIO <- as.factor(data$PORTFOLIO)

#Defining the Target Variable
data$PerformanceLabel <- as.factor(ifelse(data$TOT_NET_REV > median(data$TOT_NET_REV), "High", "Low"))

#Splitting data into test and train.
set.seed(123)
trainIndex <- createDataPartition(data$PerformanceLabel, p = 0.7, list = FALSE)
```

```

trainData <- data[trainIndex, ]
testData <- data[-trainIndex, ]

#Random Forest Model
RandomForest_model <- randomForest(PerformanceLabel ~ CLI_AMOUNT + TERM_DAYS + DUE_DAYS + PAYDEX + VANT
NO_OF_PAYMENT + PAYMENT_AMOUNT + FUEL_SPEND + NONFUEL_SPEND + TOT_
FUEL_NUM_TRX + NONFUEL_NUM_TRX + FUEL_TRX_AMT + NONFUEL_TRX_AMT + I
NSF_PMTS + LOCK_DAYS + TOT_NET_REV + TOT_TRX_AMT + TOT_NUM_TRX + S
data = trainData, importance = TRUE, ntree = 100)

#Model Evaluation
predictions <- predict(RandomForest_model, testData)
confusionMatrix(predictions, testData$PerformanceLabel)

```

```

## Confusion Matrix and Statistics
##
##           Reference
## Prediction High Low
##      High    73    0
##      Low     2    75
##
##              Accuracy : 0.9867
##              95% CI : (0.9527, 0.9984)
##      No Information Rate : 0.5
##      P-Value [Acc > NIR] : <2e-16
##
##              Kappa : 0.9733
##
##  Mcnemar's Test P-Value : 0.4795
##
##              Sensitivity : 0.9733
##              Specificity : 1.0000
##      Pos Pred Value : 1.0000
##      Neg Pred Value : 0.9740
##      Prevalence : 0.5000
##      Detection Rate : 0.4867
##      Detection Prevalence : 0.4867
##      Balanced Accuracy : 0.9867
##
##      'Positive' Class : High
##

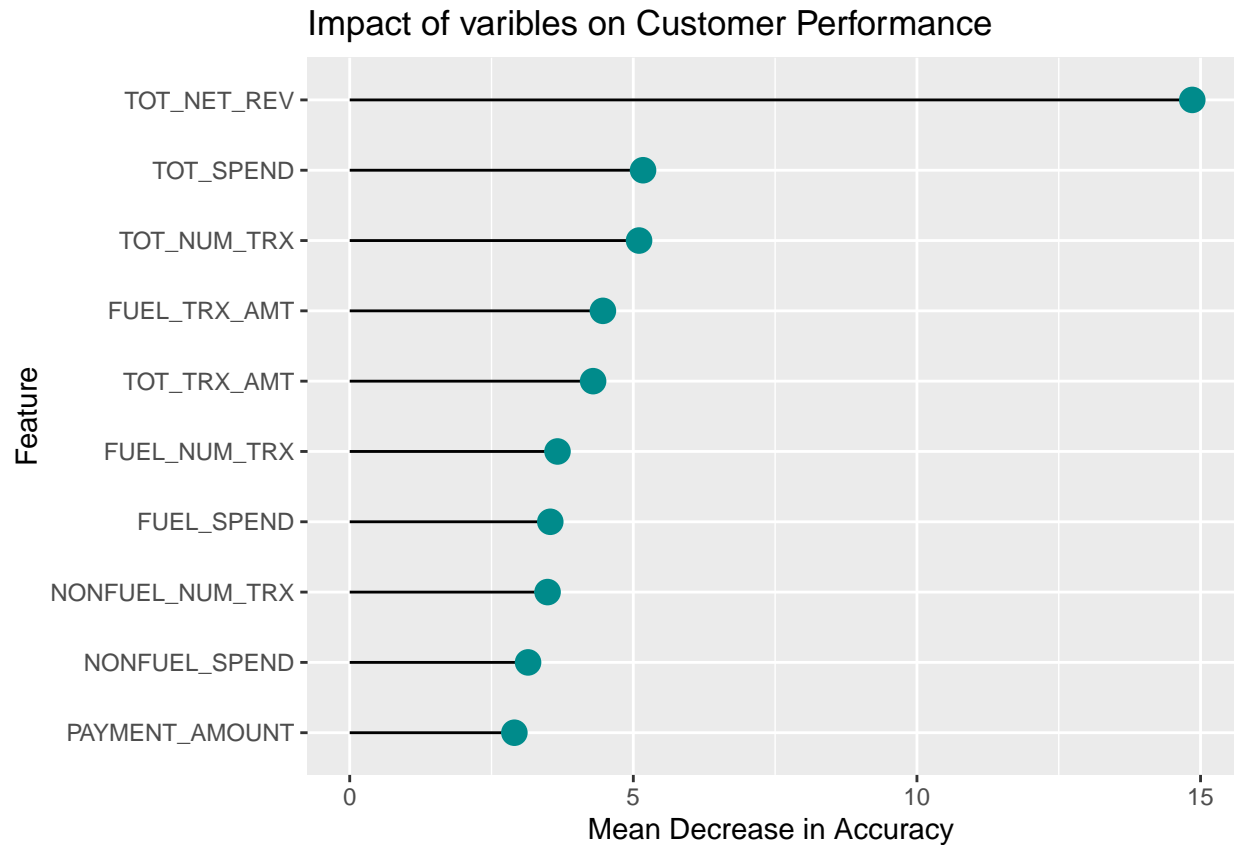
```

```

#Customer performance Evaluation:
importance_df <- as.data.frame(importance(RandomForest_model))
importance_df$Variable <- rownames(importance_df)
top_importance <- importance_df %>% arrange(desc(MeanDecreaseAccuracy)) %>% head(10)

ggplot(top_importance, aes(x = reorder(Variable, MeanDecreaseAccuracy), y = MeanDecreaseAccuracy)) +
  geom_segment(aes(x = Variable, xend = Variable, y = 0, yend = MeanDecreaseAccuracy), color="black") +
  geom_point(color="cyan4", size=4) +
  coord_flip() +
  labs(title = "Impact of variables on Customer Performance ", y = "Mean Decrease in Accuracy", x = "Fea

```



*#Transaction Evaluation:*

```
p1 <- ggplot(testData, aes(x = PerformanceLabel, y = FUEL_SPEND, fill = PerformanceLabel)) +
  geom_boxplot() +
  labs(title = "Fuel Spend by Performance Label", x = "Performance Label", y = "Fuel Spend") +
  theme_minimal() +
  scale_fill_manual(values = c("High" = "chartreuse4", "Low" = "orange"))

p2 <- ggplot(testData, aes(x = PerformanceLabel, y = NONFUEL_SPEND, fill = PerformanceLabel)) +
  geom_boxplot() +
  labs(title = "Non Fuel Spend by Performance Label", x = "Performance Label", y = "Non Fuel Spend") +
  theme_minimal() +
  scale_fill_manual(values = c("High" = "chartreuse4", "Low" = "orange"))

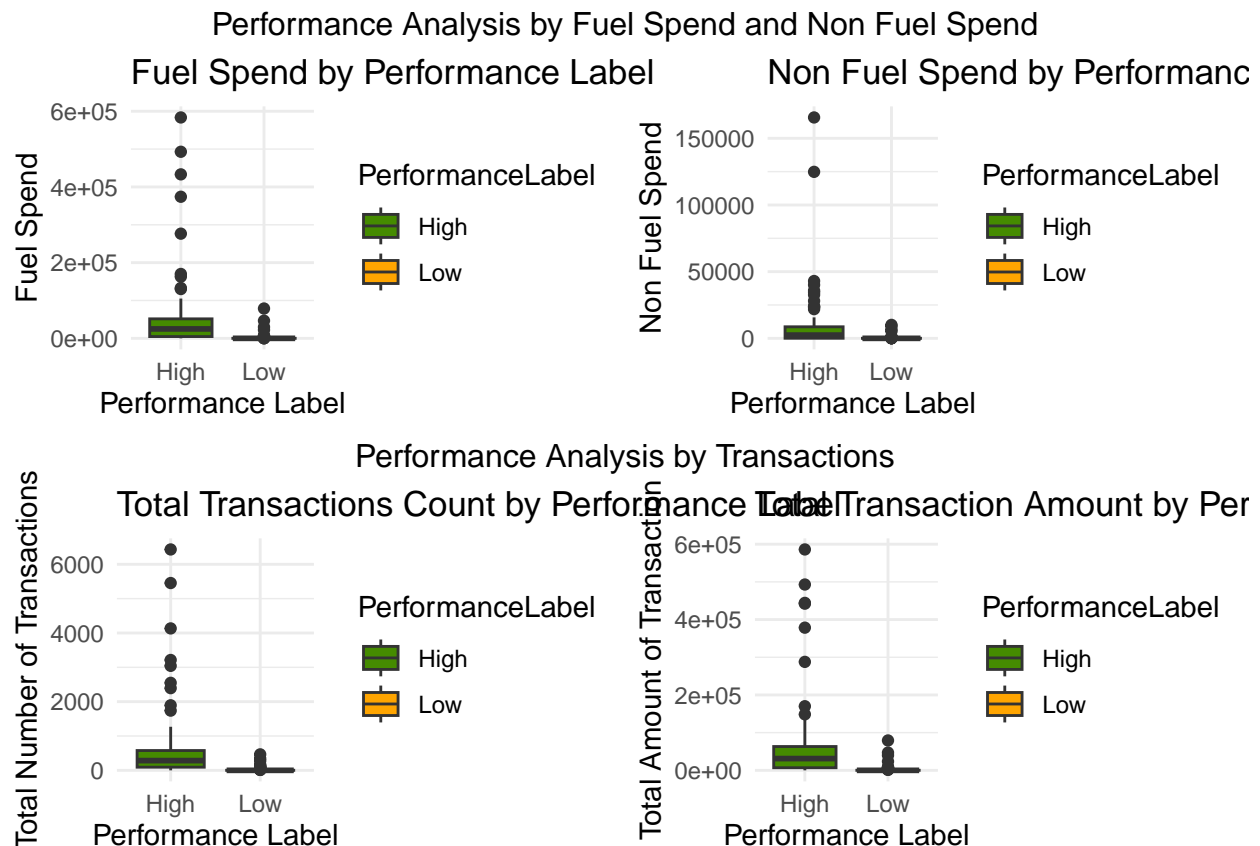
p3<- ggplot(testData, aes(x = PerformanceLabel, y = TOT_NUM_TRX, fill = PerformanceLabel)) +
  geom_boxplot() +
  labs(title = "Total Transactions Count by Performance Label", x = "Performance Label", y = "Total Num") +
  theme_minimal() +
  scale_fill_manual(values = c("High" = "chartreuse4", "Low" = "orange"))

p4<- ggplot(testData, aes(x = PerformanceLabel, y = TOT_TRX_AMT, fill = PerformanceLabel)) +
  geom_boxplot() +
  labs(title = "Total Transaction Amount by Performance Label", x = "Performance Label", y = "Total Amou") +
  theme_minimal() +
  scale_fill_manual(values = c("High" = "chartreuse4", "Low" = "orange"))
```

```

grid.arrange(
  arrangeGrob(p1, p2, ncol = 2, top = "Performance Analysis by Fuel Spend and Non Fuel Spend"),
  arrangeGrob(p3, p4, ncol = 2, top = "Performance Analysis by Transactions"),
  heights = c(1, 1),
  padding = unit(1, "line")
)

```



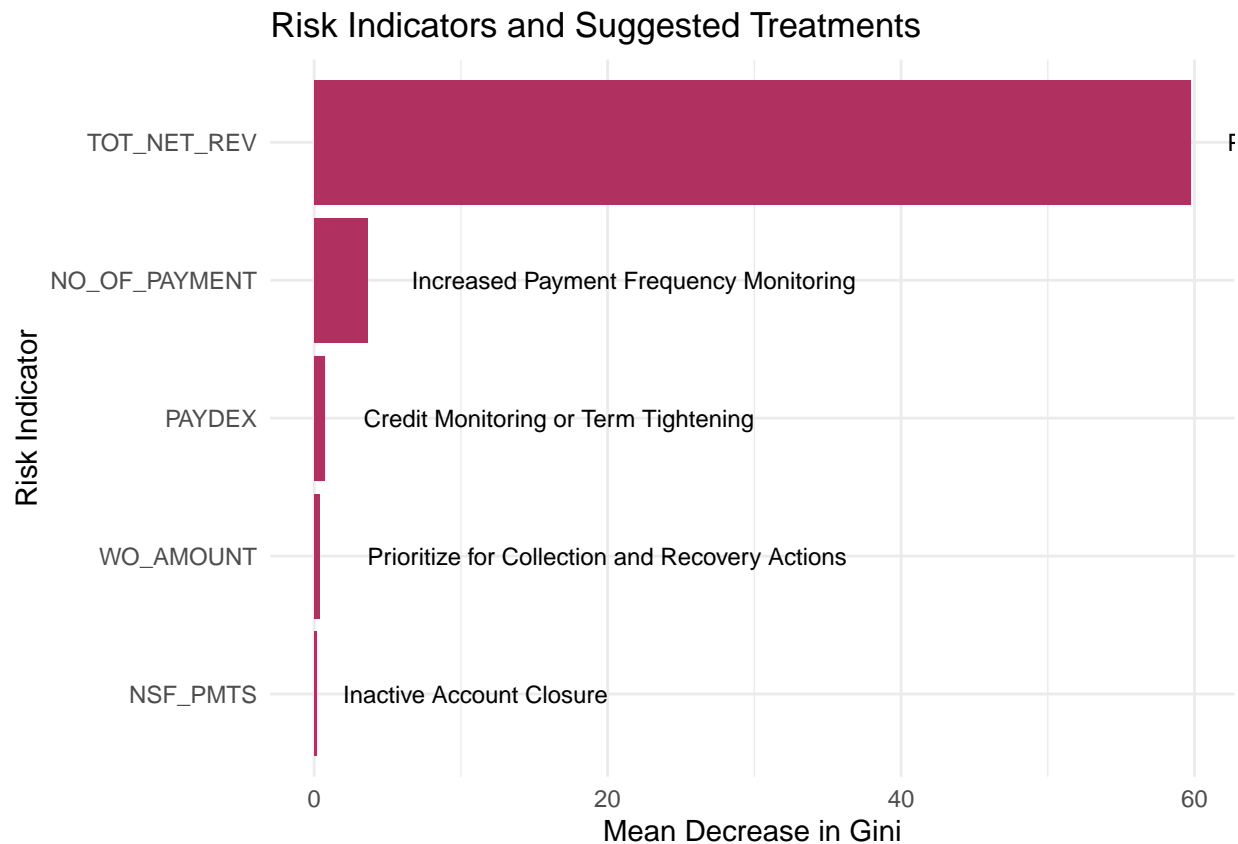
```

#Risk Assessment
#Risk Indicators and Suggested Treatments
risk_related_features <- c("PAYDEX", "TOT_NET_REV", "NSF_PMTS", "NO_OF_PAYMENT", "WO_AMOUNT")
risk_importance_df <- importance_df %>% filter(Variable %in% risk_related_features)
risk_importance_df <- risk_importance_df %>%
  mutate(Treatment = case_when(
    Variable == "PAYDEX" ~ "Credit Monitoring or Term Tightening",
    Variable == "TOT_NET_REV" ~ "Proactive Communication on Dues",
    Variable == "NSF_PMTS" ~ "Inactive Account Closure",
    Variable == "NO_OF_PAYMENT" ~ "Increased Payment Frequency Monitoring",
    Variable == "WO_AMOUNT" ~ "Prioritize for Collection and Recovery Actions"
  ))

ggplot(risk_importance_df, aes(x = reorder(Variable, MeanDecreaseGini), y = MeanDecreaseGini)) +
  geom_bar(stat = "identity", fill = "maroon") +
  coord_flip() +
  labs(title = "Risk Indicators and Suggested Treatments",
       x = "Risk Indicator",

```

```
y = "Mean Decrease in Gini" +
geom_text(aes(label = Treatment), hjust = -0.1, size = 3, color = "black") +
theme_minimal()
```



```
#Risk Associated Accounts
data$PredictedRisk <- predict(RandomForest_model, data)
high_risk_customers <- data %>% filter(PredictedRisk == "High")
View(high_risk_customers)

#Download Excel of High Risk Customers
write.csv(high_risk_customers, "High_Risk_Customers.csv", row.names = FALSE)
write.xlsx(high_risk_customers, path = "High_Risk_Customers.xlsx")

# View or print high-risk customers for verification
print(high_risk_customers)
```

```
## # A tibble: 248 x 38
##   FAKE_ACCTCODE OPT_IN_DATE          FUEL_ONLY_PARENT_ACCT CLI_AMOUNT
##   <dbl> <dtm>          <chr>          <dbl>
## 1 111111 2021-09-11 00:00:00 0          11000
## 2 111114 2022-07-08 00:00:00 0           6000
## 3 111116 2022-01-12 00:00:00 0           3000
## 4 111117 2022-08-10 00:00:00 0          62000
## 5 111120 2023-04-07 00:00:00 0          58000
## 6 111126 2022-03-24 00:00:00 0          60000
```

```

## 7      111130 2022-10-19 00:00:00 0      11500
## 8      111139 2021-05-21 00:00:00 0      4000
## 9      111143 2022-08-16 00:00:00 0      4500
## 10     111157 2022-09-27 00:00:00 0      7500
## # i 238 more rows
## # i 34 more variables: FLEETCOR_OPEN_DATE <dtm>, CITY <chr>, STATE <fct>,
## #   ZIP <chr>, TERM_DAYS <dbl>, DUE_DAYS <dbl>, LOCK_DAYS <dbl>,
## #   CREDIT_LIMIT <dbl>, LOCK_CODE <fct>, LOCK_REASON <fct>, LOCK_TYPE <chr>,
## #   DEPOSIT_FLAG <dbl>, PORTFOLIO <fct>, LOB_REPORTING <chr>, FUEL_SPEND <dbl>,
## #   NONFUEL_SPEND <dbl>, TOT_SPEND <dbl>, PAYMENT_AMOUNT <dbl>,
## #   NO_OF_PAYMENT <dbl>, NSF_AMT <dbl>, NSF_PMTS <dbl>, PAYDEX <dbl>, ...

```