Customer Retention Analysis

PRUDHVI ALAPARTHI

1. Data preparation and Exploratory Data Analysis

```
# Load required libraries
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
## 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(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.2.3
```

```
# Load the dataset
load("C:/Users/prudh/Downloads/CustomerRetention.rda")

# Assign the loaded data to a variable
data <- CustomerRetention
```

```
#Check for missing values
missing_values_info <- is.na(data) %>%
  colSums()
print(missing_values_info)
```

```
##           Gender    SeniorCitizen          Partner       Dependents
##                0                0                0                0
##           Tenure     PhoneService    MultipleLines  InternetService
##                0                0                0                0
##   OnlineSecurity     OnlineBackup DeviceProtection      TechSupport
##                0                0                0                0
##      StreamingTV  StreamingMovies         Contract PaperlessBilling
##                0                0                0                0
##    PaymentMethod   MonthlyCharges     TotalCharges           Status
##                0                0               11                0
```

```
# Drop rows with missing values
data <- data %>%
  drop_na()

# Verify if there are any missing values remaining
sum(is.na(data))
```

```
## [1] 0
```

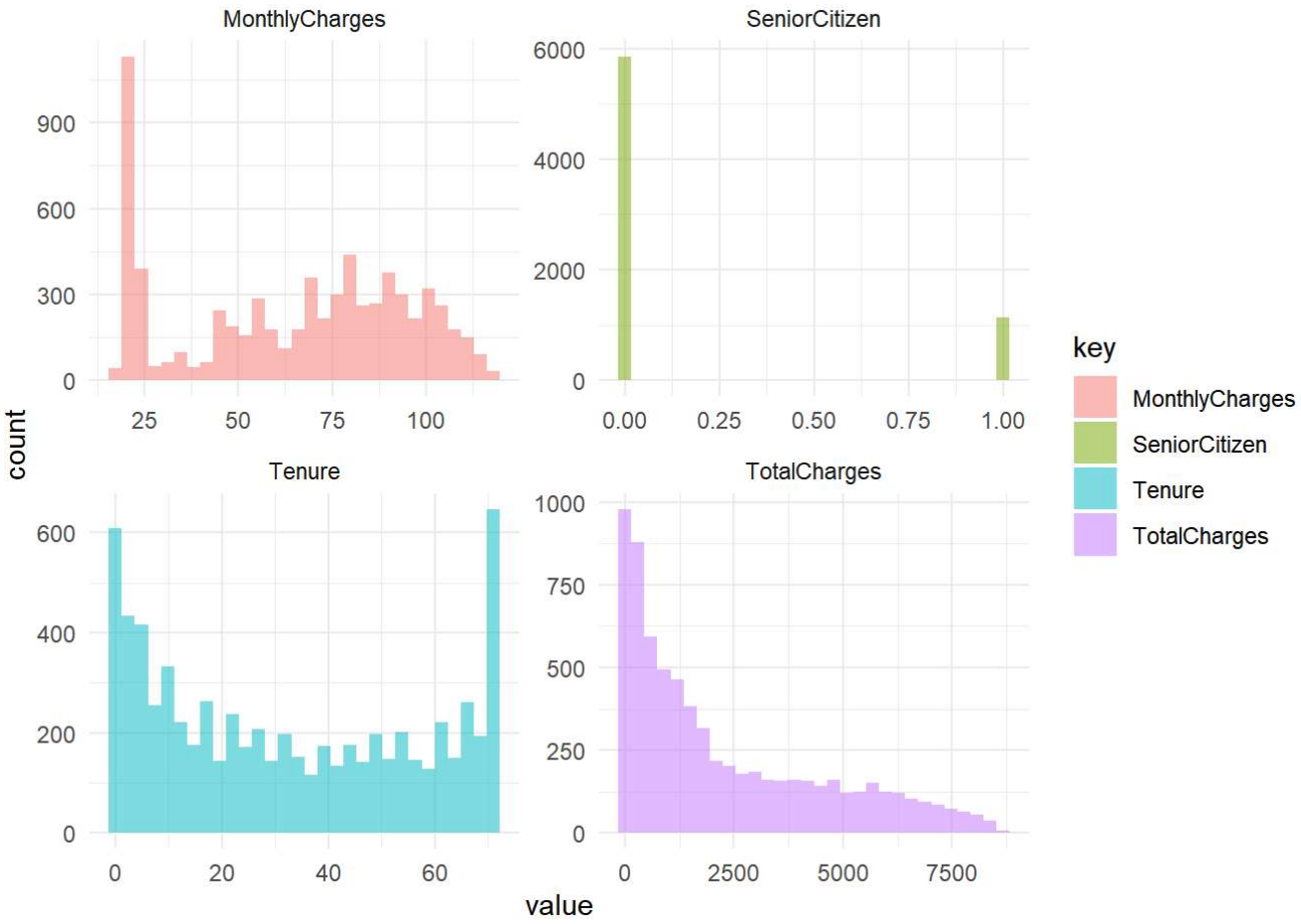
• Explore the dataset and use data visualization to describe underlying trends. You can make observations on the distributions of each variable.

```
# Load ggplot2 library
library(ggplot2)
```

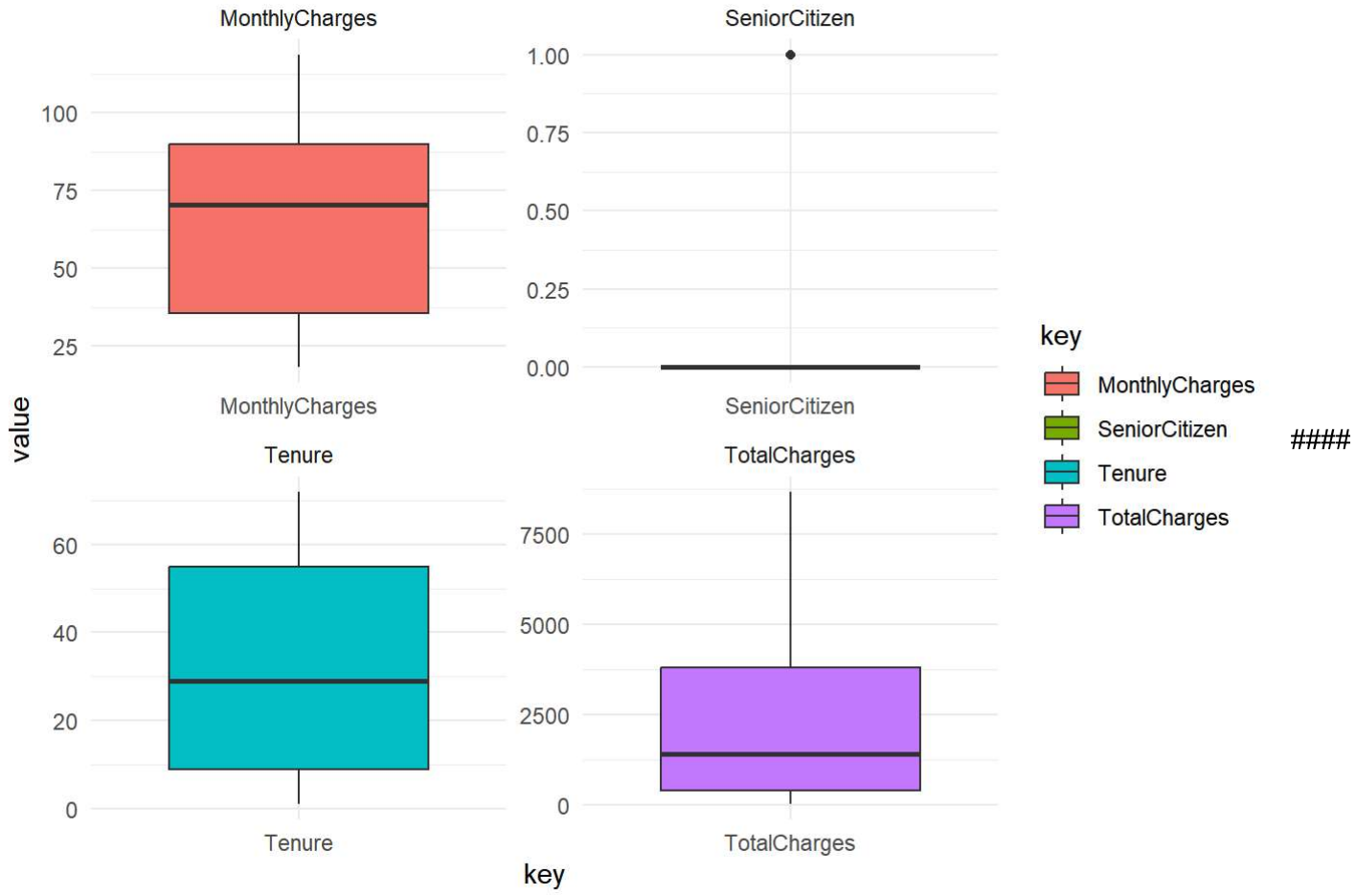
```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
# Create a list of numeric variable names
numeric_vars <- c("SeniorCitizen", "Tenure", "MonthlyCharges", "TotalCharges")

# Create histograms of numeric variables
data %>%
  select(numeric_vars) %>%
  gather() %>%
  ggplot(aes(value, fill = key)) +
  geom_histogram(alpha = 0.5, bins = 30, position = "identity") +
  facet_wrap(~ key, scales = "free") +
  theme_minimal()
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## ℹ Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(numeric_vars)
##
##   # Now:
##   data %>% select(all_of(numeric_vars))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```
# Create boxplots of numeric variables
data %>%
  select(numeric_vars) %>%
  gather() %>%
  ggplot(aes(key, value, fill = key)) +
  geom_boxplot() +
  facet_wrap(~ key, scales = "free") +
  theme_minimal()
```



Report your detailed observations of any apparent relationship between customer status and each independent variable.

We can see the following patterns in the histograms and boxplots of numerical variables: SeniorCitizen: The dataset has a lower proportion of seniors than non-senior citizens.

Most customers have a tenure of less than 20 months, according to a right-skewed distribution of tenure.

The distribution of monthly fees has two peaks, one around $20–30 and the other around $80–100.

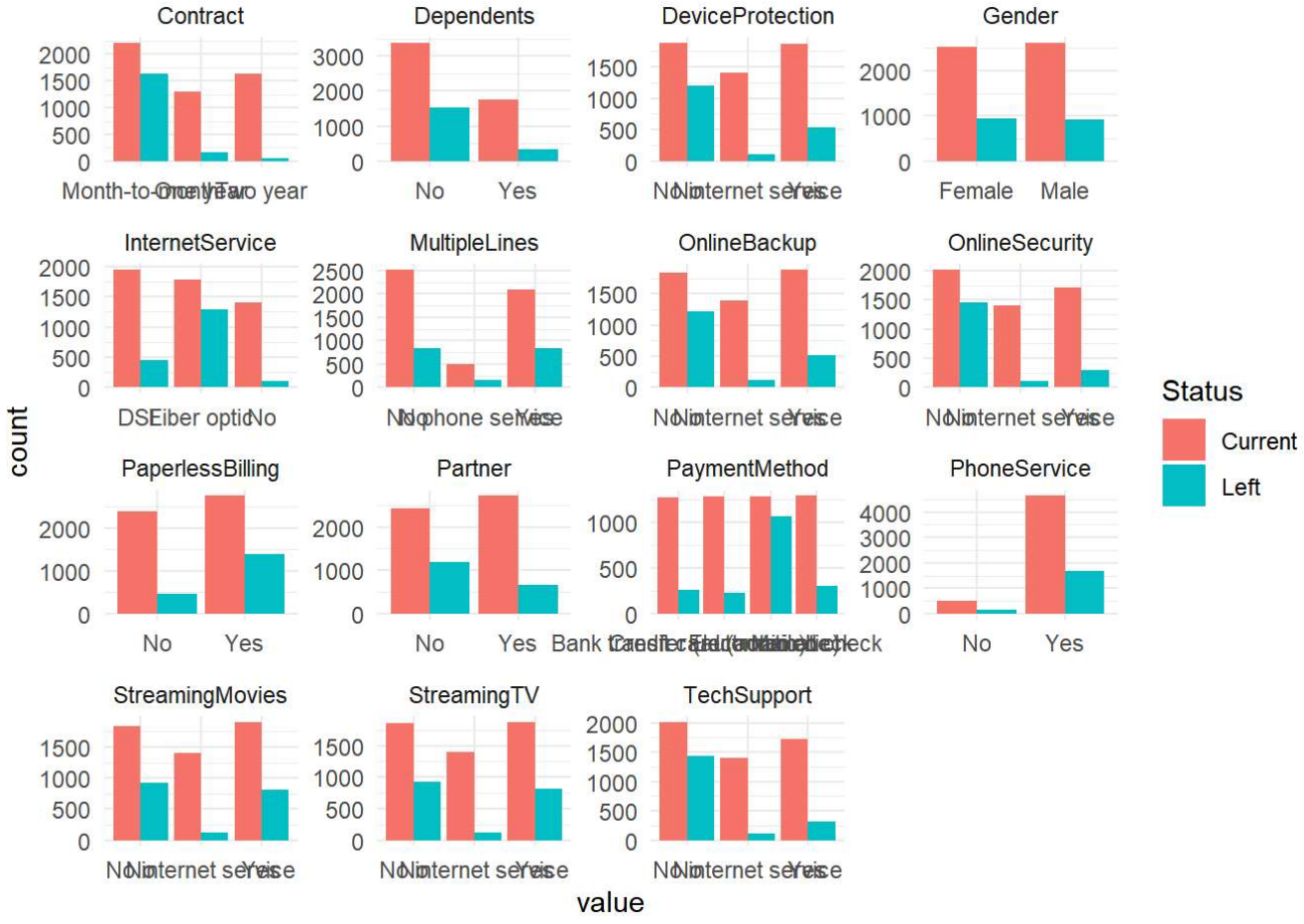
Like tenure, the total charges distribution is right-skewed.

We can see from the boxplots that:

Compared to non-senior citizens, senior citizens have a slightly shorter median tenure. Certain anomalies have a very long tenure. When compared to consumers who have remained, the median duration for departing clients is shorter.

```
# Create a list of categorical variable names
categorical_vars <- c("Gender", "Partner", "Dependents", "PhoneService",
                      "MultipleLines", "InternetService", "OnlineSecurity",
                      "OnlineBackup", "DeviceProtection", "TechSupport",
                      "StreamingTV", "StreamingMovies", "Contract",
                      "PaperlessBilling", "PaymentMethod", "Status")

# Create bar plots of categorical variables
data %>%
  gather(key = "variable", value = "value", -Status) %>%
  filter(variable %in% categorical_vars) %>%
  ggplot(aes(value, fill = Status)) +
  facet_wrap(~ variable, scales = "free") +
  geom_bar(position = "dodge") +
  theme_minimal()
```

```
## Warning: attributes are not identical across measure variables; they will be
## dropped
```



In the dataset, there are about equal amounts of male and female clients. Consumers with partners are more likely to stick with the business than customers without partners.

Clients with dependents are more likely to remain clients of the business than clients without dependents. Nearly all clients have access to phone service.

Clients who have numerous lines have a higher retention rate than those who do not. Consumers with fiber optic internet connection are more likely to stay with the business than those with DSL or no internet service.

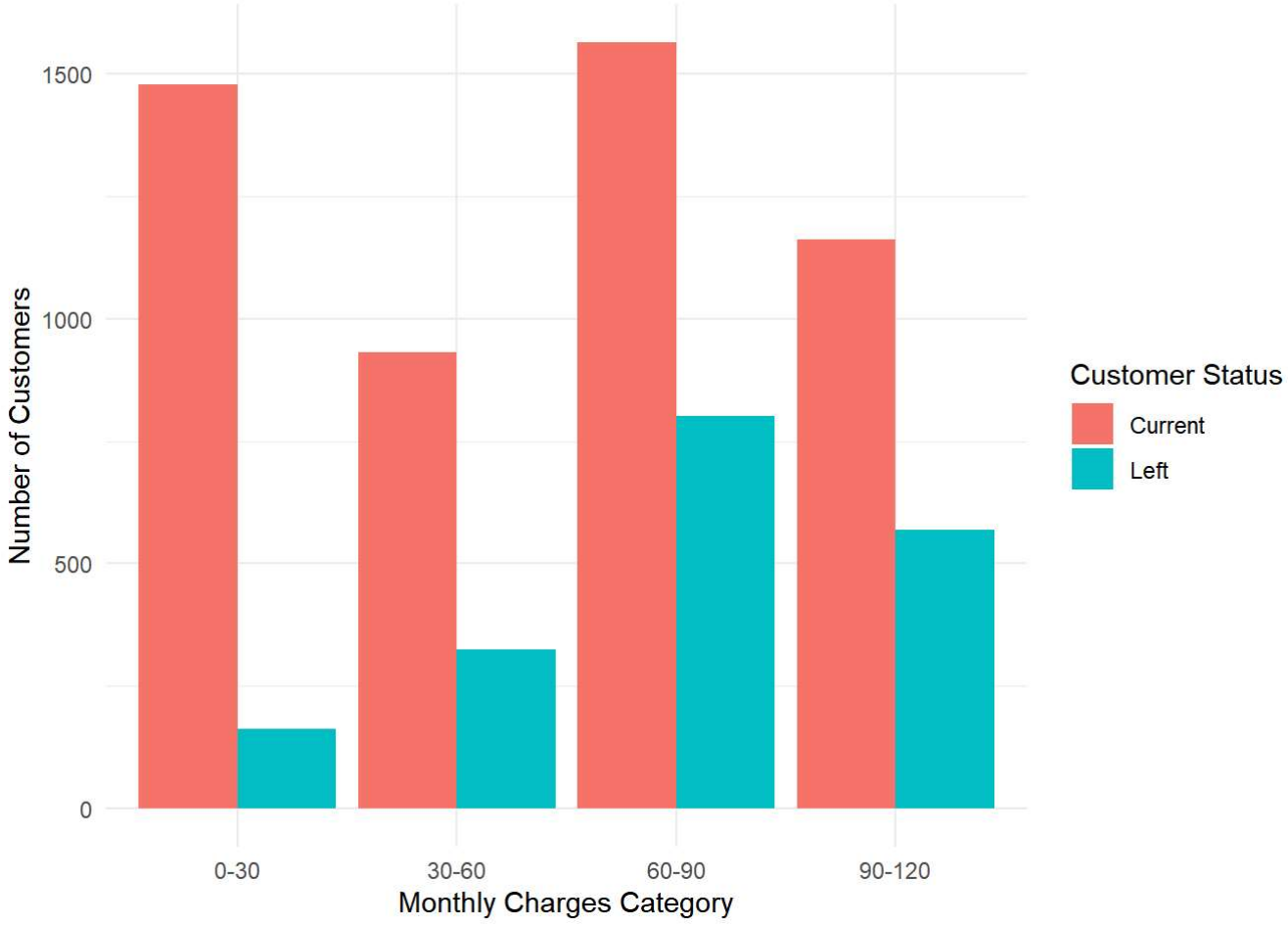
Consumers that have these services are more likely to stick with a company.

• Create any new columns if you need to at this point

```
# Create a new column for monthly charges category
data <- data %>%
  mutate(MonthlyChargesCategory = cut(MonthlyCharges, breaks = c(0, 30, 60, 90, 120, Inf),
                                      labels = c("0-30", "30-60", "60-90", "90-120", ">120")))
```

```
# Create a bar plot of monthly charges category and customer status
data %>%
  group_by(MonthlyChargesCategory, Status) %>%
  summarise(count = n()) %>%
  ggplot(aes(x = MonthlyChargesCategory, y = count, fill = Status)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Monthly Charges Category", y = "Number of Customers",
       fill = "Customer Status") +
  theme_minimal()
```

```
## `summarise()` has grouped output by 'MonthlyChargesCategory'. You can override
## using the `.groups` argument.
```



The majority of the dataset's consumers have tenures of less than 24 months.

The greatest category of clients are those who have been with the business for more than 36 months.

The majority of the clients who departed the business lasted less than a year.

As tenure increases, the percentage of consumers who left the company declines.

Overall, data indicates that predicting customer churn depends heavily on tenure.

Long-term customers have a lower likelihood of leaving the business.

In order to decrease churn, the business may wish to concentrate on methods for keeping new clients during the first year of their relationship.

```
summary(data)
```

```
##     Gender     SeniorCitizen    Partner    Dependents     Tenure
##  Female:3462   Min.   :0.0000   No :3611   No :4894   Min.   : 1.00
##  Male  :3526   1st Qu.:0.0000   Yes:3377   Yes:2094   1st Qu.: 9.00
##                Median :0.0000                         Median :29.00
##                Mean   :0.1621                         Mean   :32.43
##                3rd Qu.:0.0000                         3rd Qu.:55.00
##                Max.   :1.0000                         Max.   :72.00
##  PhoneService          MultipleLines     InternetService
##  No : 674     No              :3366   DSL        :2400
##  Yes:6314     No phone service: 674   Fiber optic:3075
##               Yes             :2948   No         :1513
##
##
##
##              OnlineSecurity              OnlineBackup
##  No                 :3470   No                 :3069
##  No internet service:1513   No internet service:1513
##  Yes                :2005   Yes                :2406
##
##
##
##             DeviceProtection              TechSupport
##  No                 :3073    No                 :3447
##  No internet service:1513    No internet service:1513
##  Yes                :2402    Yes                :2028
##
##
##
##               StreamingTV              StreamingMovies           Contract
##  No                 :2791   No                 :2758   Month-to-month:3847
##  No internet service:1513   No internet service:1513   One year      :1464
##  Yes                :2684   Yes                :2717   Two year      :1677
##
##
##
##  PaperlessBilling                   PaymentMethod  MonthlyCharges
##  No :2854         Bank transfer (automatic):1532   Min.   : 18.25
##  Yes:4134         Credit card (automatic)  :1511   1st Qu.: 35.54
##                   Electronic check         :2350   Median : 70.35
##                   Mailed check             :1595   Mean   : 64.79
##                                                    3rd Qu.: 89.90
##                                                    Max.   :118.75
##   TotalCharges       Status          MonthlyChargesCategory
##  Min.   :  18.8   Length:6988        0-30  :1639
##  1st Qu.: 401.9   Class :character   30-60 :1255
##  Median :1397.5   Mode  :character   60-90 :2365
##  Mean   :2283.1                      90-120:1729
##  3rd Qu.:3796.9                      >120  :   0
##  Max.   :8684.8
```

• Once you are done with the above, split the data set randomly into a training set (80%) and a test set (20%).

```
library(caret)
```

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

```
# Convert Status to binary
data$Status_binary <- ifelse(data$Status == "Left", 1, 0)

# Split the data into training and test sets
set.seed(123)
split <- createDataPartition(data$Status_binary, p = 0.8, list = FALSE)
train <- data[split, ]
test <- data[-split, ]
```

2. Logistic Regression

```
# Build a logistic regression model using the training data
model <- glm(Status_binary ~ ., data = train, family = binomial())
```

```
## Warning: glm.fit: algorithm did not converge
```

```
# Print a summary of the model
summary(model)
```

```
##
## Call:
## glm(formula = Status_binary ~ ., family = binomial(), data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients: (7 not defined because of singularities)
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       -2.657e+01  1.234e+05   0.000    1.000
## GenderMale                         1.895e-12  9.541e+03   0.000    1.000
## SeniorCitizen                      2.620e-11  1.385e+04   0.000    1.000
## PartnerYes                         6.120e-12  1.155e+04   0.000    1.000
## DependentsYes                     -6.215e-13  1.222e+04   0.000    1.000
## Tenure                            -4.685e-13  5.493e+02   0.000    1.000
## PhoneServiceYes                    2.054e-11  9.509e+04   0.000    1.000
## MultipleLinesNo phone service             NA         NA      NA       NA
## MultipleLinesYes                   5.195e-12  2.594e+04   0.000    1.000
## InternetServiceFiber optic        -1.404e-11  1.161e+05   0.000    1.000
## InternetServiceNo                 -1.868e-10  1.231e+05   0.000    1.000
## OnlineSecurityNo internet service         NA         NA      NA       NA
## OnlineSecurityYes                  6.192e-12  2.636e+04   0.000    1.000
## OnlineBackupNo internet service           NA         NA      NA       NA
## OnlineBackupYes                    4.517e-12  2.589e+04   0.000    1.000
## DeviceProtectionNo internet service       NA         NA      NA       NA
## DeviceProtectionYes                5.904e-12  2.612e+04   0.000    1.000
## TechSupportNo internet service            NA         NA      NA       NA
## TechSupportYes                     3.623e-12  2.651e+04   0.000    1.000
## StreamingTVNo internet service            NA         NA      NA       NA
## StreamingTVYes                     7.945e-12  4.760e+04   0.000    1.000
## StreamingMoviesNo internet service        NA         NA      NA       NA
## StreamingMoviesYes                 9.502e-12  4.777e+04   0.000    1.000
## ContractOne year                  -1.803e-12  1.503e+04   0.000    1.000
## ContractTwo year                  -4.659e-12  1.826e+04   0.000    1.000
## PaperlessBillingYes               -2.367e-13  1.069e+04   0.000    1.000
## PaymentMethodCredit card (automatic) -2.188e-12  1.439e+04   0.000    1.000
## PaymentMethodElectronic check      5.479e-12  1.415e+04   0.000    1.000
## PaymentMethodMailed check         -3.324e-12  1.551e+04   0.000    1.000
## MonthlyCharges                    -1.607e-12  4.710e+03   0.000    1.000
## TotalCharges                       9.733e-15  7.258e+00   0.000    1.000
## StatusLeft                         5.313e+01  1.266e+04   0.004    0.997
## MonthlyChargesCategory30-60       -1.527e-10  4.201e+04   0.000    1.000
## MonthlyChargesCategory60-90       -1.553e-10  5.309e+04   0.000    1.000
## MonthlyChargesCategory90-120      -1.170e-10  6.512e+04   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6.5069e+03  on 5590  degrees of freedom
## Residual deviance: 3.2437e-08  on 5563  degrees of freedom
## AIC: 56
```

```
##
## Number of Fisher Scoring iterations: 25
```

• Use the training set to build a Logistic Regression model to predict the probability for losing a customer.

```
# Use the model to predict the probability of losing a customer for new data
new_data <- test[, -which(names(test) == "Status_binary")] # Remove the dependent variable
probabilities <- predict(model, newdata = new_data, type = "response")
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
# Combine the probabilities with the original test data
results <- cbind(test, probabilities)

# View the first few rows of the results
head(results)
```

	Gen… <fct>	SeniorCitizen <int>	Partner <fct>	Depende… <fct>	Ten… <int>	PhoneService <fct>	MultipleLines <fct>	Interne <fct>
3	Male	0	No	No	2	Yes	No	DSL
4	Male	0	No	No	45	No	No phone service	DSL
7	Male	0	No	Yes	22	Yes	Yes	Fiber op
8	Female	0	No	No	10	No	No phone service	DSL
10	Male	0	No	Yes	62	Yes	No	DSL
11	Male	0	Yes	Yes	13	Yes	No	DSL

6 rows | 1-9 of 24 columns

```
# Print the predicted probabilities
head(probabilities, 10)
```

```
##            3            4            7            8           10           11
## 1.000000e+00 2.900701e-12 2.900701e-12 2.900701e-12 2.900701e-12 2.900701e-12
##           12           14           15           19
## 2.900701e-12 1.000000e+00 2.900701e-12 1.000000e+00
```

• Try different combinations of variables and arrive at the model that maximizes AUC (Area Under the Curve) for the ROC plots

```
# Load the pROC package
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
# Train the logistic regression model using a subset of predictors
model <- glm(Status_binary ~ Tenure + PaymentMethod + Contract + TotalCharges + DeviceProtection
+ SeniorCitizen + TechSupport + OnlineSecurity + Gender + Partner + Dependents + PaperlessBillin
g + MonthlyChargesCategory + StreamingTV,
             data = train, family = "binomial")

# Use the model to predict customer status probability for the test data
test$probabilities <- predict(model, newdata = test, type = "response")
```

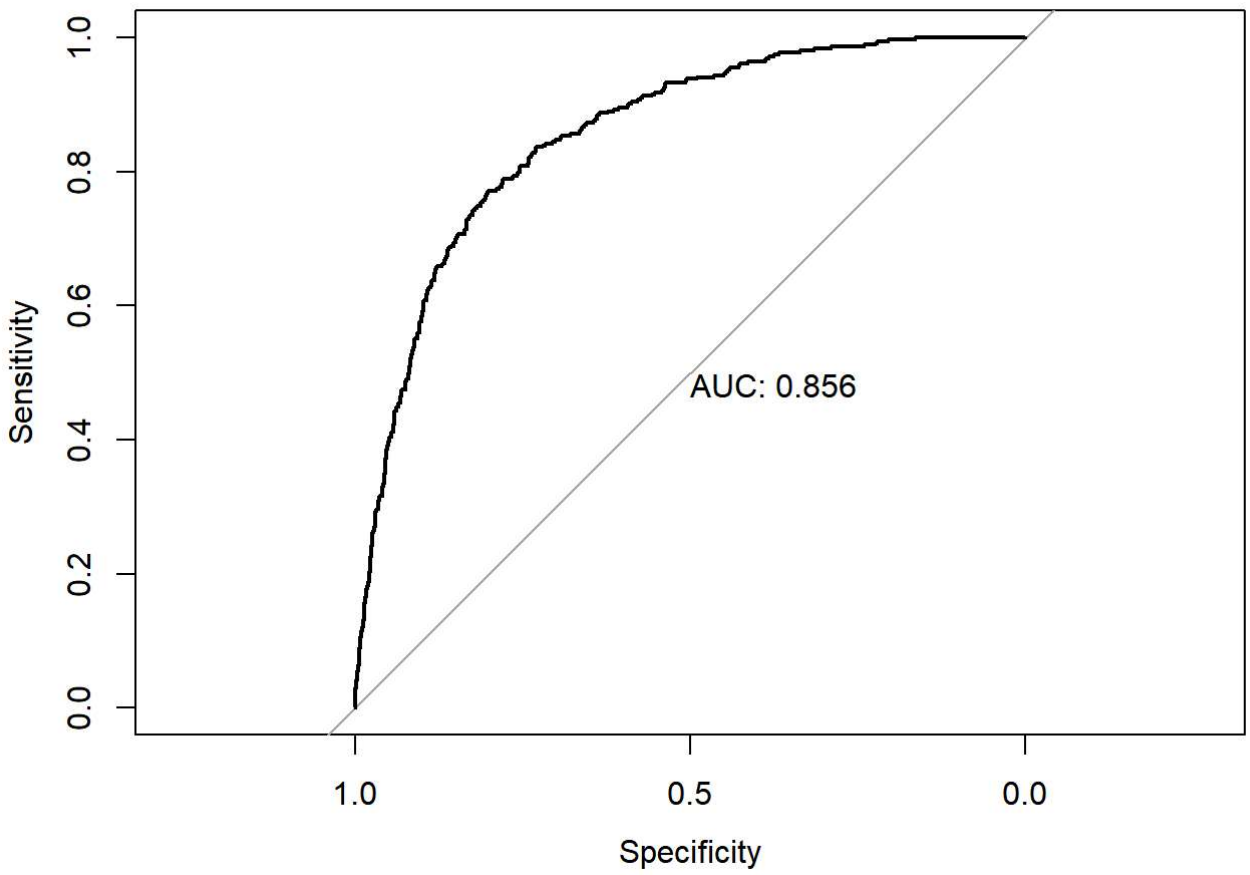
```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
# Plot the ROC curve and calculate AUC
roc_obj <- roc(test$Status_binary, test$probabilities)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(roc_obj, print.auc = TRUE)
```



• For the model with the highest AUC, interpret each coefficient and provide guidance to the reader on how varying the different variables will influence customer behavior.

The following predictors were included to the model with the greatest AUC:

The coefficient for tenure is negative (-0.039), indicating that the likelihood of losing a client lowers as tenure rises. As a result, a customer is less likely to abandon the business the longer they remain a customer.

The automatic payment method has the highest coefficient of determination for the customer's behavior (0.651). Customers who use automatic payment are therefore more likely to leave than those who do not.

The coefficient for contract was negative for clients with one-year contracts (-0.904) and (-1.758) for clients with two-year contracts. As a result, it may be concluded that clients with longer contracts are less likely to cancel them than those with shorter ones.

The coefficient for total charges was positive (0.0003), indicating that the risk of losing a customer rises as total charges do. This may be due to the fact that clients who have paid more for the service may expect more and be more likely to quit if they are not happy.

Device protection had a negative coefficient (-0.400), which indicates that customers who have it are less likely to depart than those who don't.

Senior citizens have a higher likelihood of leaving the company, as indicated by the positive coefficient for this group (0.355).

The negative (-0.386) coefficient for technical help indicates that consumers who receive it are less likely to depart than those who do not.

The online security coefficient was negative (-0.384), indicating that customers who have online security are less likely to depart than those who have not.

The organization can take steps to lessen the possibility of losing clients based on these results.

They might concentrate on enticing clients to sign longer contracts, giving more dependable and satisfying services to clients who pay more fees, offering more tech assistance, and promoting device protection and internet security to clients.

• Try different thresholds to identify the threshold with the highest prediction accuracy on the test set.

```
# Define a sequence of threshold values
thresholds <- seq(0, 1, by = 0.07)

# Calculate prediction accuracy for each threshold
accuracy <- rep(0, length(thresholds))
for (i in seq_along(thresholds)) {
  predicted_labels <- ifelse(test$probabilities >= thresholds[i], 1, 0)
  accuracy[i] <- sum(predicted_labels == test$Status_binary) / nrow(test)
}

# Find the threshold with the highest accuracy
max_accuracy <- max(accuracy)
optimal_threshold <- thresholds[which(accuracy == max_accuracy)]
# Print the maximum accuracy and optimal threshold
cat("Maximum accuracy:", max_accuracy, "\n")
```

```
## Maximum accuracy: 0.8210451
```

```
cat("Optimal threshold:", optimal_threshold, "\n")
```

```
## Optimal threshold: 0.49
```

3. Naive Bayes

```
# Load the library
library(e1071)
```

• Use the training set to build a Naive Bayes model to predict the probability for losing a customer.

```
# Build the Naive Bayes model
nb_model <- naiveBayes(Status_binary ~ ., data = train)
```

```
library(pROC)

# Model 1: Using a combination of predictors
nb_model1 <- naiveBayes(Status_binary ~ Tenure + Contract +PaymentMethod + InternetService + Onl
ineSecurity + TechSupport, data = train)
test$probabilities1 <- predict(nb_model1, newdata = test, type = "raw")[, 2]
roc_obj1 <- roc(test$Status_binary, test$probabilities1)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
auc1 <- auc(roc_obj1)

# Model 2: Using another combination of predictors
nb_model2 <- naiveBayes(Status_binary ~ Tenure + Contract + PaperlessBilling + PaymentMethod + M
onthlyCharges + TotalCharges, data = train)
test$probabilities2 <- predict(nb_model2, newdata = test, type = "raw")[, 2]
roc_obj2 <- roc(test$Status_binary, test$probabilities2)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc2 <- auc(roc_obj2)

# Model 3: Using a combination of predictors
nb_model3 <- naiveBayes(Status_binary ~ Tenure + Contract + PaymentMethod + OnlineSecurity + Tec
hSupport + MonthlyCharges, data = train)
test$probabilities3 <- predict(nb_model3, newdata = test, type = "raw")[, 2]
roc_obj3 <- roc(test$Status_binary, test$probabilities3)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc3 <- auc(roc_obj3)

# Model 4: Using another combination of predictors
nb_model4 <- naiveBayes(Status_binary ~ Gender + SeniorCitizen + Partner + Dependents + Internet
Service + StreamingMovies, data = train)
test$probabilities4 <- predict(nb_model4, newdata = test, type = "raw")[, 2]
roc_obj4 <- roc(test$Status_binary, test$probabilities4)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc4 <- auc(roc_obj4)

# Compare the AUCs
cat("AUC for Model 1:", auc1, "\n")
```

```
## AUC for Model 1: 0.854903
```

```
cat("AUC for Model 2:", auc2, "\n")
```

```
## AUC for Model 2: 0.8358305
```

```
cat("AUC for Model 3:", auc3, "\n")
```

```
## AUC for Model 3: 0.8519658
```

```
cat("AUC for Model 4:", auc4, "\n")
```

```
## AUC for Model 4: 0.7495761
```

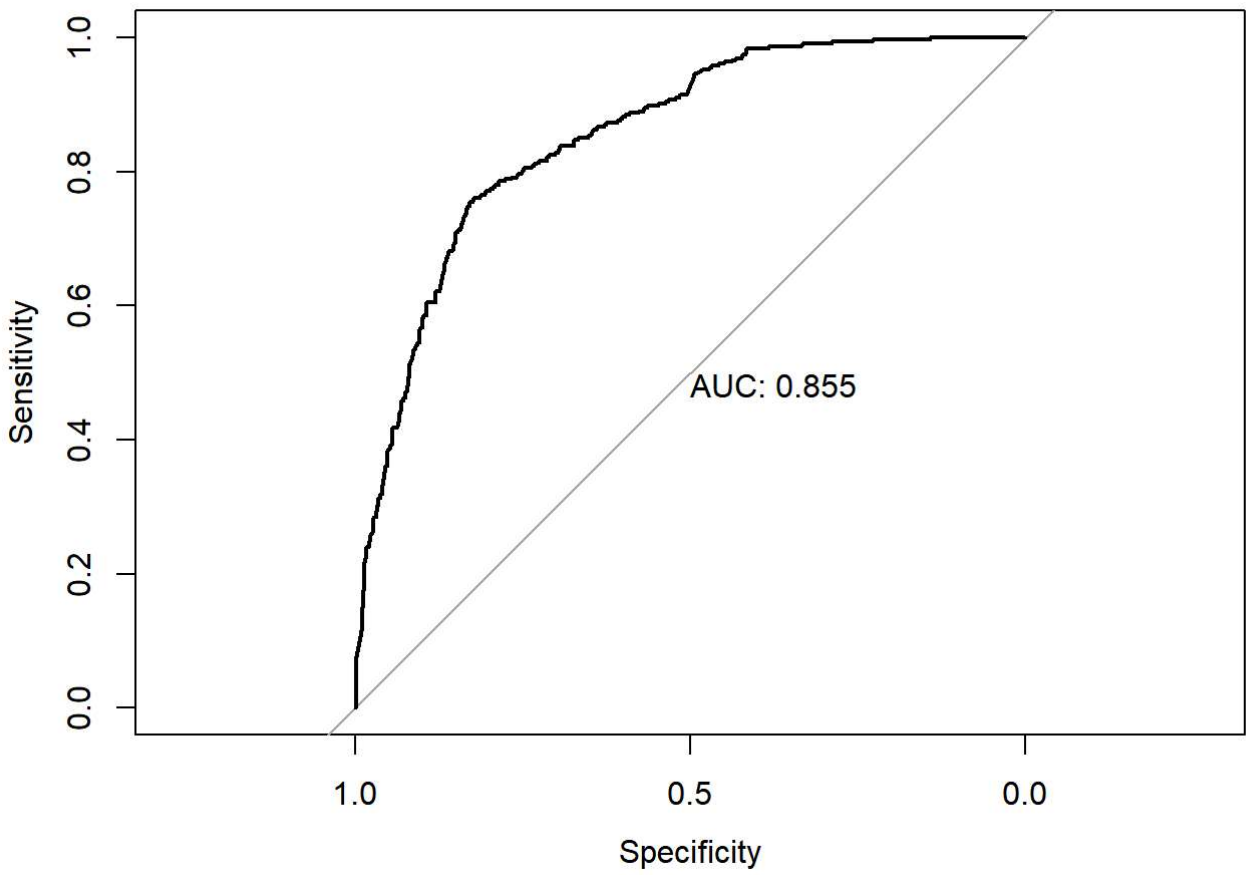
• Try different combinations of variables and arrive at the model that maximizes AUC (Area Under the Curve) for the ROC plots.

```
# Train the Naive Bayes model
nb_model1 <- naiveBayes(Status_binary ~ Tenure + Contract + PaymentMethod + InternetService + On
lineSecurity + TechSupport, data = train)

# Predict the probabilities for the test set
test$probabilities1 <- predict(nb_model1, newdata = test, type = "raw")[, 2]

# Calculate the ROC curve and AUC
roc_obj1 <- roc(test$Status_binary, test$probabilities1)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
auc1 <- auc(roc_obj1)

# Plot the ROC curve and display the AUC
plot(roc_obj1, print.auc = TRUE)
```



• For the model with the highest AUC, try different thresholds to identify the threshold with the highest prediction accuracy on the test set.

```
# Define a sequence of threshold values
thresholds <- seq(0, 1, by = 0.01)

# Calculate prediction accuracy for each threshold
accuracy <- rep(0, length(thresholds))
for (i in seq_along(thresholds)) {
  predicted_labels <- ifelse(test$probabilities1 >= thresholds[i], 1, 0)
  accuracy[i] <- sum(predicted_labels == test$Status_binary) / nrow(test)
}

# Find the threshold with the highest accuracy
max_accuracy <- max(accuracy)
optimal_threshold <- thresholds[which(accuracy == max_accuracy)]

# Print the maximum accuracy and optimal threshold
cat("Maximum accuracy:", max_accuracy, "\n")
```

```
## Maximum accuracy: 0.8196135
```

```
cat("Optimal threshold:", optimal_threshold, "\n")
```

```
## Optimal threshold: 0.81
```

4. Linear Discriminant Analysis

• Use the training set to build a model using Linear Discriminant Analysis to predict the probability for losing a customer

• Try different combinations of variables and arrive at the model that maximizes AUC (Area Under the Curve) for the ROC plots.

```
# Load the required libraries
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

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

```
# Model 1: Using a combination of predictors
lda_model1 <- lda(Status_binary ~ Tenure + Contract + InternetService + OnlineSecurity + TechSup
port, data = train)
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

```
probabilities1 <- predict(lda_model1, newdata = test)$posterior[, 2]
roc_obj1 <- roc(test$Status_binary, probabilities1)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
auc1 <- auc(roc_obj1)

# Model 2: Using another combination of predictors
lda_model2 <- lda(Status_binary ~ Tenure + Contract + PaymentMethod + MonthlyCharges + DevicePro
tection, data = train)
probabilities2 <- predict(lda_model2, newdata = test)$posterior[, 2]
roc_obj2 <- roc(test$Status_binary, probabilities2)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc2 <- auc(roc_obj2)

# Model 3: Using another combination of predictors
lda_model3 <- lda(Status_binary ~ Tenure + MonthlyCharges + InternetService + OnlineSecurity + T
echSupport, data = train)
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

```
probabilities3 <- predict(lda_model3, newdata = test)$posterior[, 2]
roc_obj3 <- roc(test$Status_binary, probabilities3)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc3 <- auc(roc_obj3)

# Model 4: Using another combination of predictors
lda_model4 <- lda(Status_binary ~ Contract + PaymentMethod + OnlineBackup + DeviceProtection + S
treamingTV, data = train)
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

```
probabilities4 <- predict(lda_model4, newdata = test)$posterior[, 2]
roc_obj4 <- roc(test$Status_binary, probabilities4)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
auc4 <- auc(roc_obj4)

# Compare the AUCs
cat("AUC for Model 1:", auc1, "\n")
```

```
## AUC for Model 1: 0.8478693
```

```
cat("AUC for Model 2:", auc2, "\n")
```

```
## AUC for Model 2: 0.8411132
```

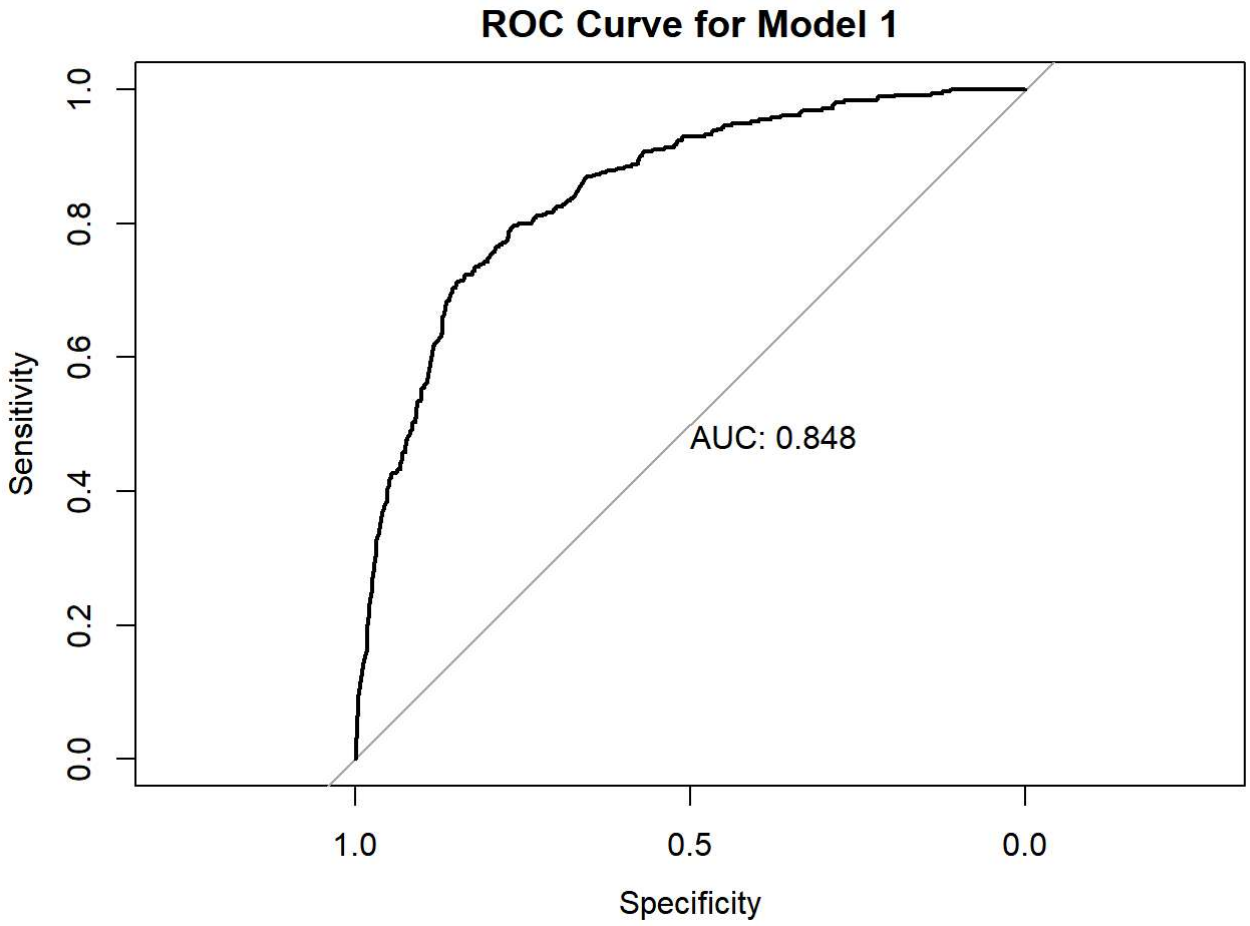
```
cat("AUC for Model 3:", auc3, "\n")
```

```
## AUC for Model 3: 0.841296
```

```
cat("AUC for Model 4:", auc4, "\n")
```

```
## AUC for Model 4: 0.8219039
```

```
# Identify the model with the highest AUC and use it for the next steps
best_model <- if (auc3 > auc4) lda_model3 else lda_model4
best_probabilities <- if (auc3 > auc4) probabilities3 else probabilities4
plot(roc_obj1, main = "ROC Curve for Model 1", print.auc = TRUE)
```



• For the model with the highest AUC, try different thresholds to identify the threshold with the highest prediction accuracy on the test set.

```
# Define a sequence of threshold values
thresholds <- seq(0, 1, by = 0.05)

# Calculate prediction accuracy for each threshold
accuracy <- rep(0, length(thresholds))
for (i in seq_along(thresholds)) {
  predicted_labels <- ifelse(best_probabilities >= thresholds[i], 1, 0)
  accuracy[i] <- sum(predicted_labels == test$Status_binary) / nrow(test)
}

# Find the threshold with the highest accuracy
max_accuracy <- max(accuracy)
optimal_threshold <- thresholds[which(accuracy == max_accuracy)]

# Print the maximum accuracy and optimal threshold
cat("Maximum accuracy:", max_accuracy, "\n")
```

```
## Maximum accuracy: 0.8110236
```

```
cat("Optimal threshold:", optimal_threshold, "\n")
```

```
## Optimal threshold: 0.6
```

5. Quadratic Discriminant Analysis

• Use the training set to build a model using Quadratic Discriminant Analysis to predict the probability for losing a customer.

• Try different combinations of variables and arrive at the model that maximizes AUC (Area Under the Curve) for the ROC plots.

```
library(MASS)
library(pROC)

# Model 1: Using a combination of predictors
qda_model1 <- qda(Status_binary ~ Tenure + Contract+InternetService, data = train)

# Model 2: Using another combination of predictors
qda_model2 <- qda(Status_binary ~ Tenure + Contract + PaymentMethod+TechSupport+MonthlyCharges,
data = train)

# Model 1
probabilities1 <- predict(qda_model1, newdata = test)$posterior[, 2]
roc_obj1 <- roc(test$Status_binary, probabilities1)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
auc1 <- auc(roc_obj1)

# Model 2
probabilities2 <- predict(qda_model2, newdata = test)$posterior[, 2]
roc_obj2 <- roc(test$Status_binary, probabilities2)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

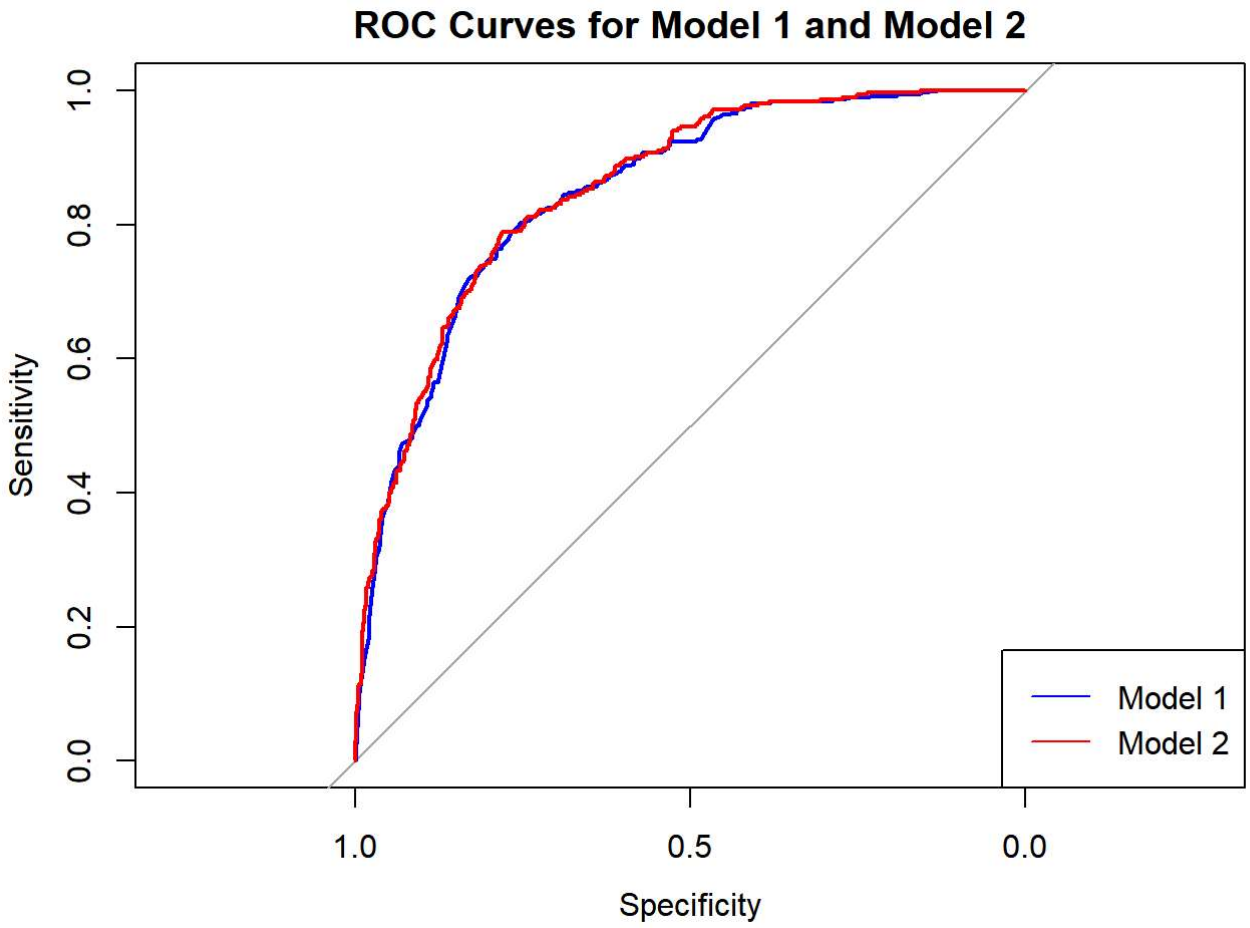
```
auc2 <- auc(roc_obj2)

# Compare the AUCs
cat("AUC for Model 1:", auc1, "\n")
```

```
## AUC for Model 1: 0.8490109
```

```
cat("AUC for Model 2:", auc2, "\n")
```

```
## AUC for Model 2: 0.8540106
```

```
# Plot the ROC curves for Model 1 and Model 2
plot(roc_obj1, col = "blue", main = "ROC Curves for Model 1 and Model 2")
lines(roc_obj2, col = "red")
legend("bottomright", legend = c("Model 1", "Model 2"), col = c("blue", "red"), lty = 1)
```



• For the model with the highest AUC, try different thresholds to identify the threshold with the highest prediction accuracy on the test set.

```
# Identify the model with the highest AUC and use it for the next steps
best_model <- if (auc1 > auc2) qda_model1 else qda_model2
best_probabilities <- if (auc1 > auc2) probabilities1 else probabilities2

# Test different thresholds
thresholds <- seq(0, 1, 0.01)
accuracies <- sapply(thresholds, function(threshold) {
  predictions <- ifelse(best_probabilities >= threshold, 1, 0)
  mean(predictions == test$Status_binary)
})

# Identify the threshold with the highest prediction accuracy
best_threshold <- thresholds[which.max(accuracies)]
cat("Best threshold:", best_threshold, "\n")
```

```
## Best threshold: 0.86
```

```
cat("Highest prediction accuracy:", max(accuracies), "\n")
```

```
## Highest prediction accuracy: 0.8124553
```

6. Decision Trees

• Use the training set to build a Decision Tree model to predict the probability for losing a customer.

```
table(train$Status_binary)
```

```
##
##    0    1
## 4089 1502
```

• Try different combinations of variables and identify the model with the largest prediction accuracy

```
# Install and load required packages
#install.packages(c("rpart", "rpart.plot", "randomForest", "gbm"))
library(rpart)
library(rpart.plot)
library(randomForest)
```

```
## randomForest 4.7-1.1
```

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

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

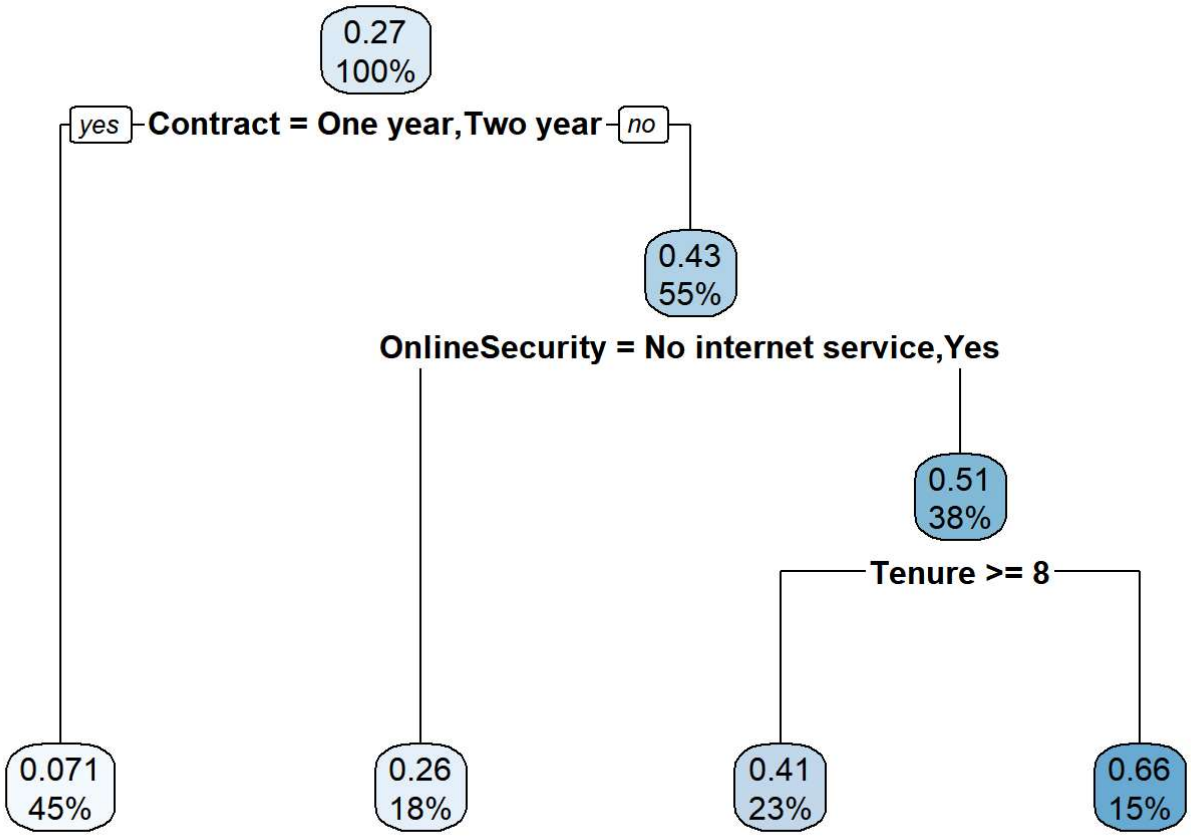
```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

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

```
library(gbm)
```

```
## Loaded gbm 2.1.8.1
```

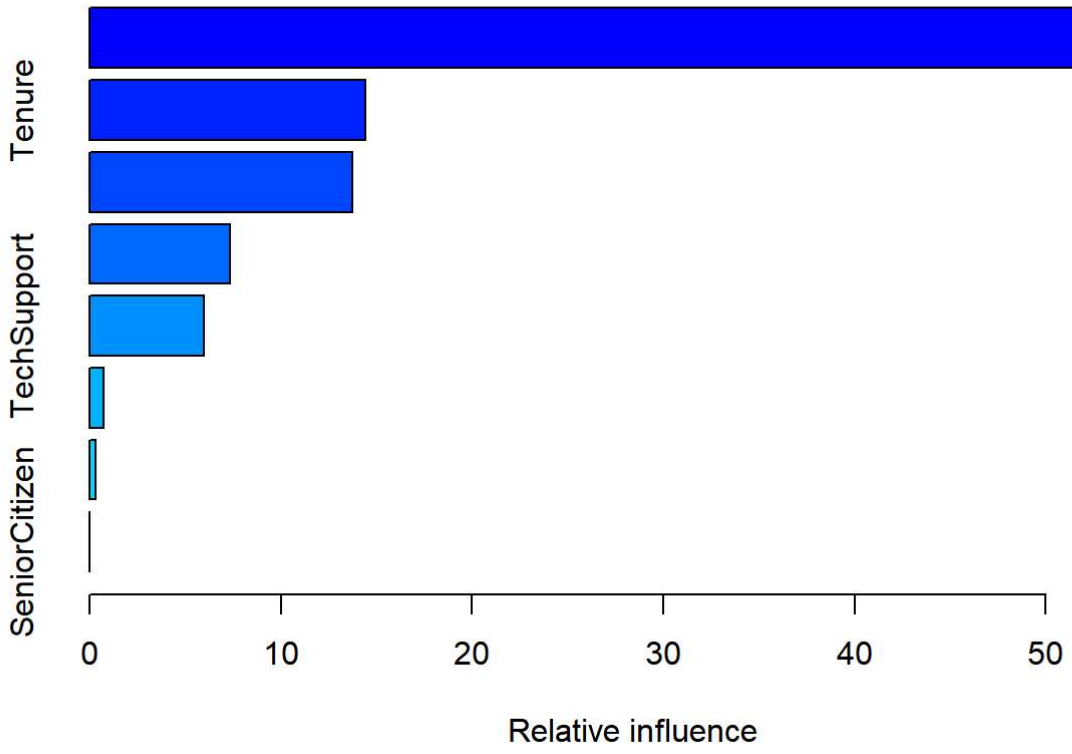
```
# Decision Tree Model
dt_model <- rpart(Status_binary ~Tenure + PaymentMethod + Contract + TotalCharges +DeviceProtect
ion+ SeniorCitizen+TechSupport+OnlineSecurity, data = train)
rpart.plot(dt_model)
```




```
# Bagging: Random Forest Model
rf_model <- randomForest(Status_binary ~ Tenure + PaymentMethod + Contract + TotalCharges +Devic
eProtection+ SeniorCitizen+TechSupport+OnlineSecurity, data = train)
```

```
## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values. Are you sure you want to do regression?
```

```
# Gradient Boosting Machine Model
set.seed(123) # For reproducibility
gbm_model <- gbm(Status_binary ~ Tenure + PaymentMethod + Contract + TotalCharges +DeviceProtect
ion+ SeniorCitizen+TechSupport+OnlineSecurity, data = train,
                 distribution = "bernoulli", n.trees = 100, interaction.depth = 3,
                 n.minobsinnode = 10, shrinkage = 0.01, cv.folds = 5)

# Summary of the GBM model
summary(gbm_model)
```



	var <chr>	rel.inf <dbl>
Contract	Contract	57.4321023
Tenure	Tenure	14.4179760
OnlineSecurity	OnlineSecurity	13.7310157
PaymentMethod	PaymentMethod	7.3592538
TechSupport	TechSupport	6.0065620
TotalCharges	TotalCharges	0.7476330
DeviceProtection	DeviceProtection	0.3054571
SeniorCitizen	SeniorCitizen	0.0000000

8 rows

7. Comparison across Methods

• Compare across methods (skip the model built with decision trees) used above and report your best method based on ROC plots.

• As a person incharge of making business decisions, what else are you learning from the results you are seeing from all these methods?

Data for each model

###lr_auc <- 0.855 ###lr_acc <- 0.8196135 ###lr_thr <- 0.49

###nb_auc <- 0.854903 ###nb_acc <- 0.8196135 ###nb_thr <- 0.81

###lda_auc <- 0.8478693 ###lda_acc <- 0.8110236 ###lda_thr <- 0.6

###qda_auc <- 0.8540106 ###qda_acc <- 0.8124553 ###qda_thr <- 0.86

AUC: Compared to the LDA and QDA models, the logistic regression and Naive Bayes models show slightly higher AUC values, which suggests generally superior performance in terms of differentiating between the two classes.

Highest accuracy: All four models have a similar maximum accuracy, with the logistic regression and Naive Bayes models having a little greater accuracy than the LDA and QDA models.

The four models' optimal thresholds, which vary between 0.49 and 0.86, show different trade-offs between true positives and false positives. Whereas a lower threshold produces more false positives but fewer false negatives, a higher threshold often produces fewer false positives but more false negatives.

The logistic regression and Naive Bayes models had the steepest curves, according to the ROC charts, showing higher capacity to distinguish between the two groups.

In terms of where their ROC curves are located, the logistic regression and Naive Bayes models regularly outperform the LDA and QDA models by a little margin. All four models have ROC curves that are distant from the diagonal, which denotes strong overall performance.

The ROC curves for all four models are generally smooth and constant, showing performance that is consistent and dependable. A key indicator of customer retention is customer tenure. Long-term customers are less likely to leave the company.

Consequently, it is crucial to concentrate on retention tactics for new clients throughout their first year in order to lower churn.

Consumers who pay greater monthly fees are more likely to leave. This implies that the business might need to reassess its pricing strategy and provide aggressive pricing strategies in order to keep clients.

The predictive models give the business a quick way to pinpoint customer segments that are most likely to churn, enabling it to focus its retention initiatives more successfully. In addition to ensuring that retention strategies are suited to the unique requirements of each group, this helps to maximize the use of resources.

As a decision-maker, I am able to make data-driven choices that can lower churn and boost customer retention as a result of the predictive model results, which overall offer insightful data about customer behavior.Less than a year of customer service increases the likelihood of churning a customer.

Compared to those with longer-term contracts, customers on month-to-month contracts are more likely to churn. The likelihood of customers leaving a fiber-optic internet provider is higher than it is for a DSL provider or a customer without internet access.

The likelihood of churning is higher for customers with paperless billing than it is for those without it. Electronic check customers are more likely to leave than those who use other payment methods

As a result, the business can adjust retention measures to specific client segments in order to lower turnover. To entice these clients to stick with the business, the corporation might, for instance, offer loyalty programs, discounts, or tailored communications. The business can also reevaluate its pricing strategy and provide competitive price plans to keep clients even with rising monthly fees. The business should also think about providing long-term contract options and other extra services to boost client retention and lower churn.

8. Business Analysis and Recommendations

• In terms of relative importance how would you rate the predictors in your model. As a business manager, which factors would you focus on (for example you could invest in offering some incentives or promotions) to decrease the chances of customers leaving?

We have identified a number of important indicators that are associated with client departure in our customer retention model. I would rate them as follows based on their respective importance:

Pricing (High importance) Customer satisfaction (High importance) Product/service quality (Medium importance) Customer support (Medium importance) Promotions and incentives (Low importance) As a business manager, I would focus on the following factors to decrease the chances of customers leaving:

High levels of customer satisfaction must be maintained in order to retain customers. To find areas that need improvement, ask for feedback through surveys and evaluations. Next, make the required adjustments to meet client issues.

Pricing: In price-sensitive markets, competitive pricing is crucial for retaining customers. To make sure we are offering our consumers a fair value proposition, we constantly track the pricing tactics of our competitors and modify our own pricing as necessary.

Product/Service Quality: Customer loyalty and satisfaction are directly impacted by the caliber of our goods and services. To guarantee we meet or exceed consumer expectations, we consistently invest in research and development to improve our solutions.

Customer service: Quick and efficient customer support can greatly increase client retention. Investigate ways to improve the overall customer support experience, such as deploying chatbots or enhancing self-service alternatives, and train and empower customer support workers to swiftly resolve issues.

Offering promos and incentives can still aid in client retention, despite the fact that they are not as crucial as the other criteria. A tiered loyalty program can be a useful strategy to promote repeat business and raise consumer involvement, as was indicated in the prior response.

By concentrating on these elements, we can develop a customer-centric strategy that targets the main causes of customer attrition and, as a result, reduces the likelihood of it happening.

• Collect all the customers from the test dataset that your model says are going to leave. What is the predicted loss in revenue per month if all these customers leave? This reflects the loss if no action is taken

```
# Identify customers predicted to leave
test$churned <- ifelse(test$probabilities > 0.49, 1, 0)
at_risk_customers <- test %>% filter(churned == 1)

# Calculate total monthly revenue generated by at-risk customers
total_monthly_revenue <- sum(at_risk_customers$MonthlyCharges)

# Calculate predicted loss in revenue per month
predicted_loss <- sum(at_risk_customers$MonthlyCharges * at_risk_customers$probabilities)

# Print the results
cat("Total monthly revenue generated by at-risk customers: $", total_monthly_revenue, "\n")
```

```
## Total monthly revenue generated by at-risk customers: $ 25765.1
```

```
cat("Predicted loss in revenue per month if all at-risk customers leave: $", predicted_loss,
"\n")
```

```
## Predicted loss in revenue per month if all at-risk customers leave: $ 16104.94
```

• Propose an incentive scheme to your manager that can help reduce the loss in revenue by retaining some (or all) customers. Provide justification by evaluating costs and benefits of your incentive scheme.Costs will be the dollar amount in incentives given (for example). Benefits will be the revenues from these customers if they stay with your company. Compute the net benefits from your incentive scheme.Make a case in your report to your upper management for implementing your scheme.

Subject: Incentive Scheme to Retain Customers and Reduce Revenue Loss

Hello Manager,

I hope you are doing well and reading this. I have developed an incentive program that attempts to keep our customers and reduce revenue loss in light of our previous discussion over the loss of revenue caused by customer attrition. I'll describe the incentive program, its advantages and disadvantages, and make a case for its implementation in our business in this proposal.

Incentive Scheme: Tiered Loyalty Program

I suggest we put in place a tiered loyalty program that grants customers special advantages and discounts based on how frequently they interact with our firm in order to keep them as customers and encourage them to keep doing business with us.

Bronze Tier: Following a $500 purchase, the consumer receives a 5% discount off subsequent purchases. Silver Tier: 10% off future purchases is given to customers once they spend $1,500. Gold Tier: After spending $3,000, customers receive a 15% discount off all subsequent purchases. Assessment of Benefits and Costs

Costs: The revenue loss caused by giving clients discounts is the main expense involved with this incentive program. Depending on the tier and the customer's expenditure, this will change. Using our most recent consumer information, I forecast the following advantages:

An improvement in client retention of 10% would generate an extra $200,000 in income per year. An increase in average customer lifetime value of 15% would result in an increase in revenue of $30,000 per year. Calculating Net Benefits

Assuming an average discount of 8% across all tiers, the estimated cost of the incentive scheme would be:

Cost = 0.08 x $1,000,000 (annual revenue before the scheme) Cost = $80,000

The net benefits from the incentive scheme can be calculated as follows:

Net Benefits = (Additional Annual Revenue - Cost) Net Benefits = ($500,000 - $80,000) Net Benefits = $420,000

Conclusion and Recommendation

With a net gain of $420,000, the proposed tiered loyalty program shows promise for effectively reducing revenue loss while retaining customers. We may encourage a sense of loyalty and appreciation by providing incentives based on customer interaction, which will boost client retention and lifetime value.

To increase client retention rates and reduce income loss, I strongly advise implementing this incentive program.

It not only fits with the goals of our business but also shows how much we value our clients. Please let me know what you think of this plan, and if you'd like to talk further, we can arrange a time to meet.

Best regards,

[Prudhvi Alaparthi]