Aoqin Yan
BUSQOM 1080
Final Project

Analysis on Bank Customer Churn

Introduction

For commercial banking, it is often challenging to deal with customer churn, for it is both hard to predict, and a major loss for the banks. This report will not only pinpoint the key factors contributing to customer attrition but will also provide actionable insights and strategies to enhance customer retention. Based on the logistic regression model and the supervised machine learning, the bank will have a better way to predict potential customer churn and improve retention strategy to reduce future customer loss.

The analysis is mainly can be divided into the following three parts: overview on customer portrait, building & testing the logistic regression model, supervised machine learning by applying k-nearest-neighbors algorithm.

Description of the Data

The BankChurners.csv dataset consists of 10127 observations in 21 variables, with all demographical and financial information, transaction histories, and account details (see below the columns of the data). Except for the transaction histories, all other variables are categorical variables including columns like "Income_Category".

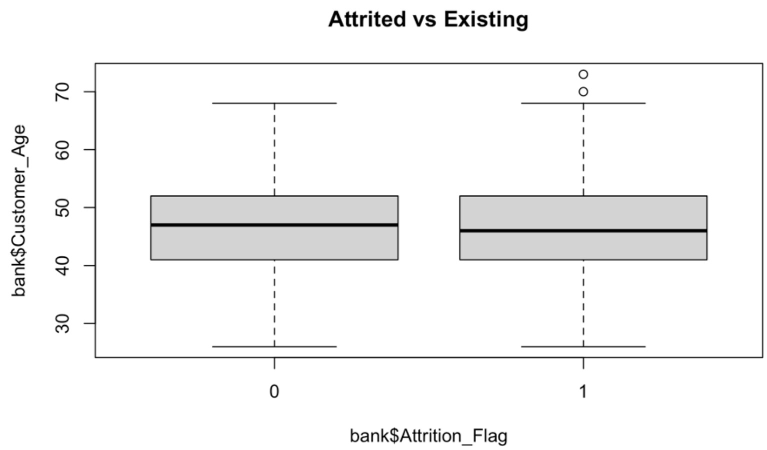
One of the very first step taken is to replace the customers category with "0" and "1", with "1" being existing customers and "0" being those who left.

```
bank                          10127 obs. of 21 variables
  $ CLIENTNUM               : int  768805383 818770008 713982108 769911858 709106358 713061558 810347208 818906208 710930508 719661558 ...
  $ Attrition_Flag          : chr  "Existing Customer" "Existing Customer" "Existing Customer" "Existing Customer" ...
  $ Customer_Age            : int  45 49 51 40 40 44 51 32 37 48 ...
  $ Gender                  : chr  "M" "F" "M" "F" ...
  $ Dependent_count         : int  3 5 3 4 3 2 4 0 3 2 ...
  $ Education_Level         : chr  "High School" "Graduate" "Graduate" "High School" ...
  $ Marital_Status          : chr  "Married" "Single" "Married" "Unknown" ...
  $ Income_Category         : chr  "$60K - $80K" "Less than $40K" "$80K - $120K" "Less than $40K" ...
  $ Card_Category           : chr  "Blue" "Blue" "Blue" "Blue" ...
  $ Months_on_book          : int  39 44 36 34 21 36 46 27 36 36 ...
  $ Total_Relationship_Count: int  5 6 4 3 5 3 6 2 5 6 ...
  $ Months_Inactive_12_mon  : int  1 1 1 4 1 1 1 2 2 3 ...
  $ Contacts_Count_12_mon   : int  3 2 0 1 0 2 3 2 0 3 ...
  $ Credit_Limit            : num  12691 8256 3418 3313 4716 ...
  $ Total_Revolving_Bal     : int  777 864 0 2517 0 1247 2264 1396 2517 1677 ...
  $ Avg_Open_To_Buy         : num  11914 7392 3418 796 4716 ...
  $ Total_Amt_Chng_Q4_Q1    : num  1.33 1.54 2.59 1.41 2.17 ...
  $ Total_Trans_Amt         : int  1144 1291 1887 1171 816 1088 1330 1538 1350 1441 ...
  $ Total_Trans_Ct          : int  42 33 20 20 28 24 31 36 24 32 ...
  $ Total_Ct_Chng_Q4_Q1     : num  1.62 3.71 2.33 2.33 2.5 ...
  $ Avg_Utilization_Ratio   : num  0.061 0.105 0 0.76 0 0.311 0.066 0.048 0.113 0.144 ...
```

First Analysis Method

For the first part, the main analysis methods used are statistical test. To be more specific, a t-test was used for testing if there is significant difference in average ages between existing customers and churn customers. And a Chi-squared test was used to check if the total amount of balance is related to the likelihood of customer churn.

In short, though there are some outliers in existing customer group (see the boxplot below), both tests can still be considered statistically significant (see Appendix Part A for detailed interpretation on the p-values and outliers). The tests indicate that there is no large difference in average ages for existing and lost customer, while the total amount of balance does relate to the likelihood of customer churn.



Second Analysis Method

For the second part, a logistic regression model was built and tested by dividing the scaled training and testing set (see Appendix Part B for more visualized results).

With the adjusted threshold of 0.75, this model is quite effective at predicting customer churn (see the result below). This model can correctly identify 87.06% of the customer churn, 71.06% of the customers who will stay, and has a high overall accuracy of 84.43%.

The area under the curve being 0.879 also suggests it has a strong ability to differentiate between churn customer who will churn and those who will not.

	Predicted	
Actual	Predicted Churn	Predicted Existing
Actual Churned	273	350
Actual Existing	145	3283

Third Analysis Method

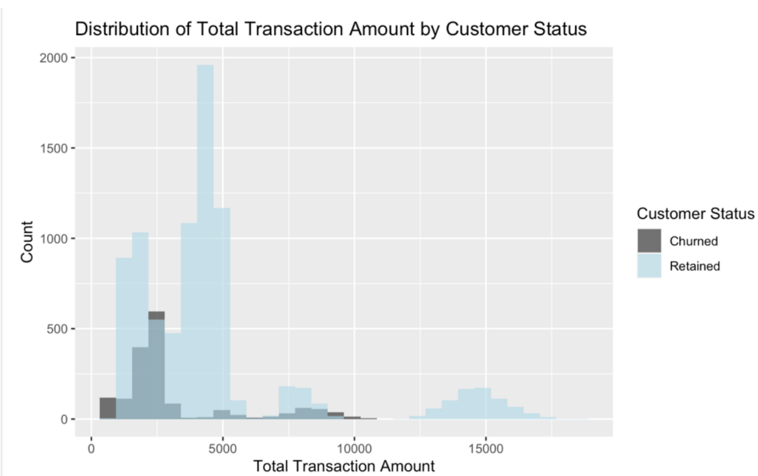
For the last part, knn is applied to train a better model than the logistic regression one. The best k (number of neighbors) found was 7, and the models does have a better performance in all aspects (see the result below). It has better sensitivity of 93.71%, specificity of 83.37%, and overall accuracy of 92.35%

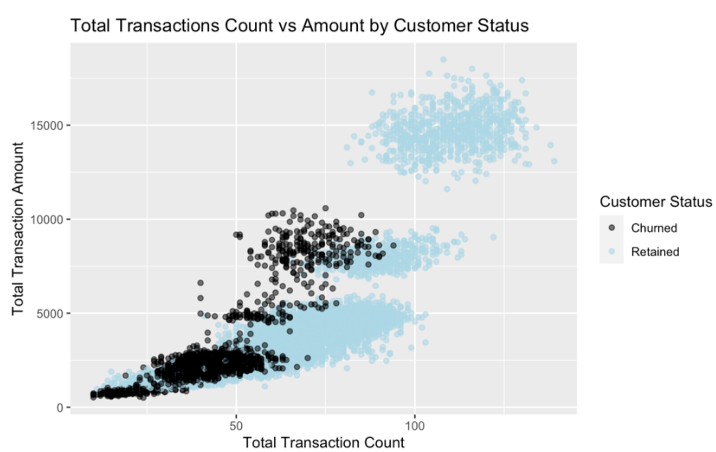
```
# Sensitivity = 0.9371445
TP / (TP+FN)
# Specificity = 0.8336449
TN / (TN+FP)
# Accuracy = 0.9234757
(TP + TN) / sum(conf_matrix_2)

```

```
                  Predicted
Actual            Predicted Churn Predicted Existing
  Actual Churned              404                103
  Actual Existing             219               3325
```

(For specific codes and data see Appendix Part C)

Additionally, the distribution of total transaction **amount** and the comparison between transactions **count** and **amount** were visualized using the "ggplot" package (see the histogram and scatter plots below).





Conclusions

Based on the testing result for the model, it is likely to be a valuable tool for the bank's retention strategies, helping to proactively identify and target customers who may be at risk of leaving. The bank can use this model to focus its customer engagement and retention efforts more efficiently.

Specifically, according to the two graphs on transaction amount and transaction counts, the obvious pattern here is that the more amount customers spend and the more often they process transactions, the more likely they are going to stay. This is intuitive that if customers have already formed spending habits with this bank account, they are more likely to be loyal and stick with this bank.

Therefore, the key here for the bank is to make sure customers can be more motivated to use their cards and should encourage them to use it more often and spend more. This can be achieved by launching special spending offers with message and email notifications to those who are likely to leave (predicted by the model, presumably those who used the card less often and spent less).

Appendix

Load in the data

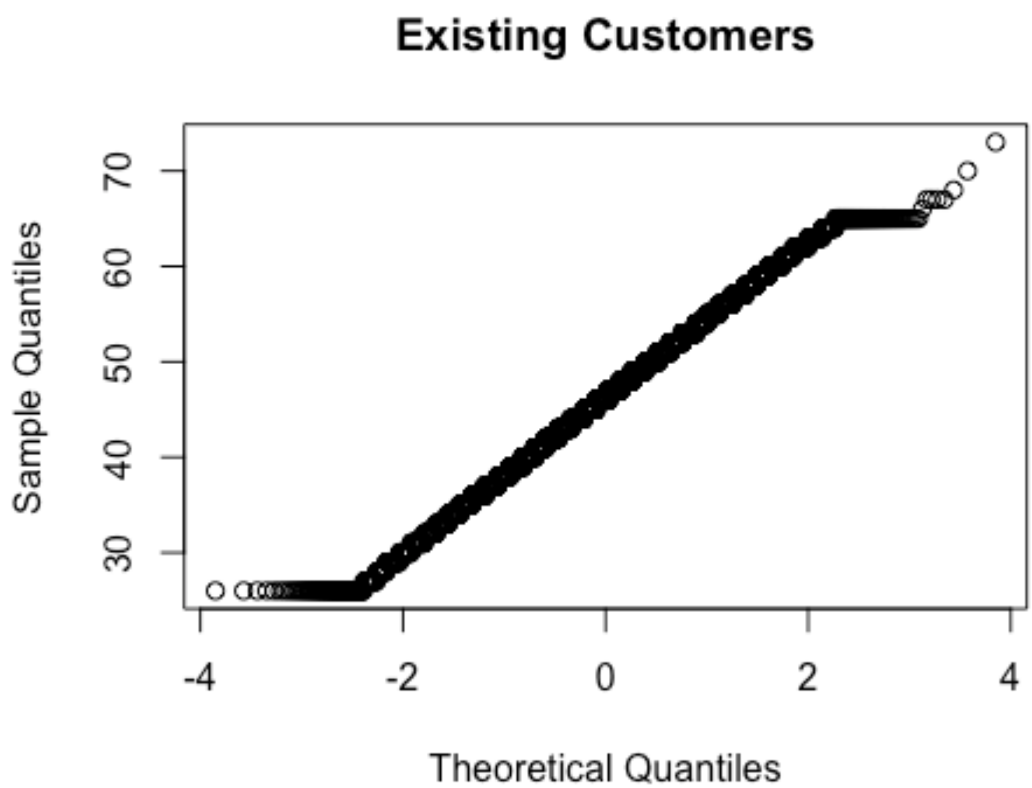
```
setwd("~/Desktop/1st semester/1080Data Analysis/project")
bank = read.csv("BankChurners_edited.csv", header = T)

library(pROC)

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

##
## Attaching package: 'pROC'

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

library(class)
library(ggplot2)
```

Part A. Portrait for both existing and attrited customers

(a) Is there a significant difference in average ages between existing customers and churn customers?

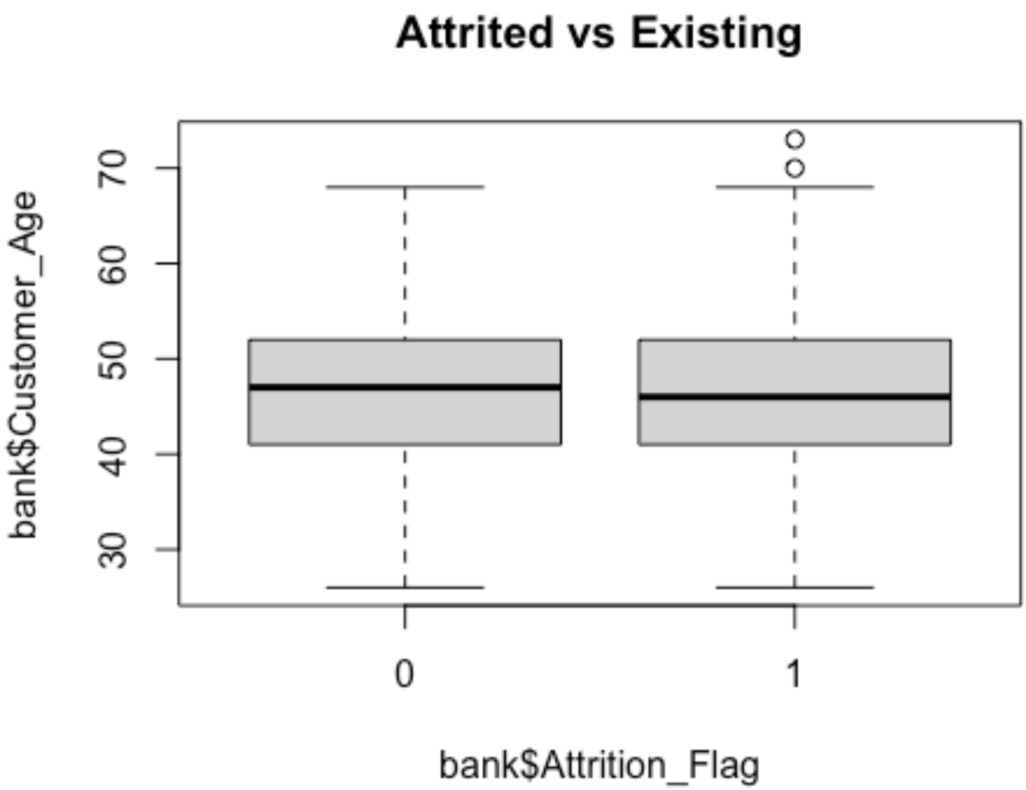
```
bank$Attrition_Flag = ifelse(bank$Attrition_Flag == "Existing Customer", 1,
0)

## Performing a t-test
age_test = t.test(data = bank, Customer_Age ~ Attrition_Flag)
print(age_test)

##
##  Welch Two Sample t-test
##
## data:  Customer_Age by Attrition_Flag
## t = 1.8988, df = 2370.8, p-value = 0.05772
## alternative hypothesis: true difference in means between group 0 and group
 1 is not equal to 0
## 95 percent confidence interval:
##  -0.01302059  0.80777731
## sample estimates:
## mean in group 0 mean in group 1
##        46.65950        46.26212

# Null Hypothesis: The average age among existing customers and attrited cust
omers are the same.
# Alt Hypothesis: The average age among existing customers and attrited custo
mers are NOT the same.

# T-test results
```

```
  # p-value = 0.05772
  # 95 percent confidence interval:
  # -0.01302059  0.80777731

  # sample estimates:
  # mean in group 0   mean in group 1
  #     46.65950        46.26212

# Testing assumptions of the t-test
  # Normality
  qqnorm(bank$Customer_Age[bank$Attrition_Flag == 1], main = "Existing Custom
ers")
```




```
  qqnorm(bank$Customer_Age[bank$Attrition_Flag == 0], main = "Attrited Custom
ers")
```


```
# Outliers
boxplot(bank$Customer_Age ~ bank$Attrition_Flag, main = "Attrited vs Existi
ng")
```




```
# Interpretations:
  # The p-value = 0.05772 is slightly above the conventional threshold of 0.0
5
  # The Q-Q plots of both groups seems to have noticeable diagonal, with some
 outliers in the existing customer group, which matches the boxplot result.
  # Given this is a large dataframe with more than 10,000 datapoints, the sli
ghtly higher p-value and some outliers should not affect the statistical sign
ificance of the t-test result.
  # Therefore, we can conclude that the difference in average ages between ex
isting and attrited customers is in between -0.01302059 and 0.80777731 (95% C
I).
```

(b) Does the Total_Revolving_Bal have a statistically significant effect on the likelihood of churn?

```
## Performing a Chi-squared test
Balance_table = table(bank$Total_Revolving_Bal, bank$Attrition_Flag)
Balance_test = chisq.test(Balance_table)

## Warning in chisq.test(Balance_table): Chi-squared approximation may be
## incorrect

print(Balance_test)
```

```
##
##  Pearson's Chi-squared test
##
## data:  Balance_table
## X-squared = 2898.8, df = 1973, p-value < 2.2e-16
```

```
# Null hypothesis: There is NO association between revolving balance amount a
nd attrition (churn).
# Alt hypothesis: There is an association between revolving balance amount an
d attrition (churn).

# Interpretation:
 # p-value < 2.2e-16. Therefore, we have evidence to reject the null hypothes
is and conclude that there is a statistically significant association between
 revolving balance amount and the likelihood of churn.
```

Part B. Building a logistic regression model

(a) Can we predict the likelihood of churn using a logistic regression model?

```
## Preparing the data
training_index = sample(1:nrow(bank), 0.6 * nrow(bank))
training_set = bank[training_index, ]
testing_set = bank[-training_index, ]

## Building the model
log_md = glm(data = training_set, Attrition_Flag ~ Months_Inactive_12_mon + T
otal_Revolving_Bal + Total_Trans_Amt + Total_Trans_Ct, family = "binomial")

summary(log_md)
```

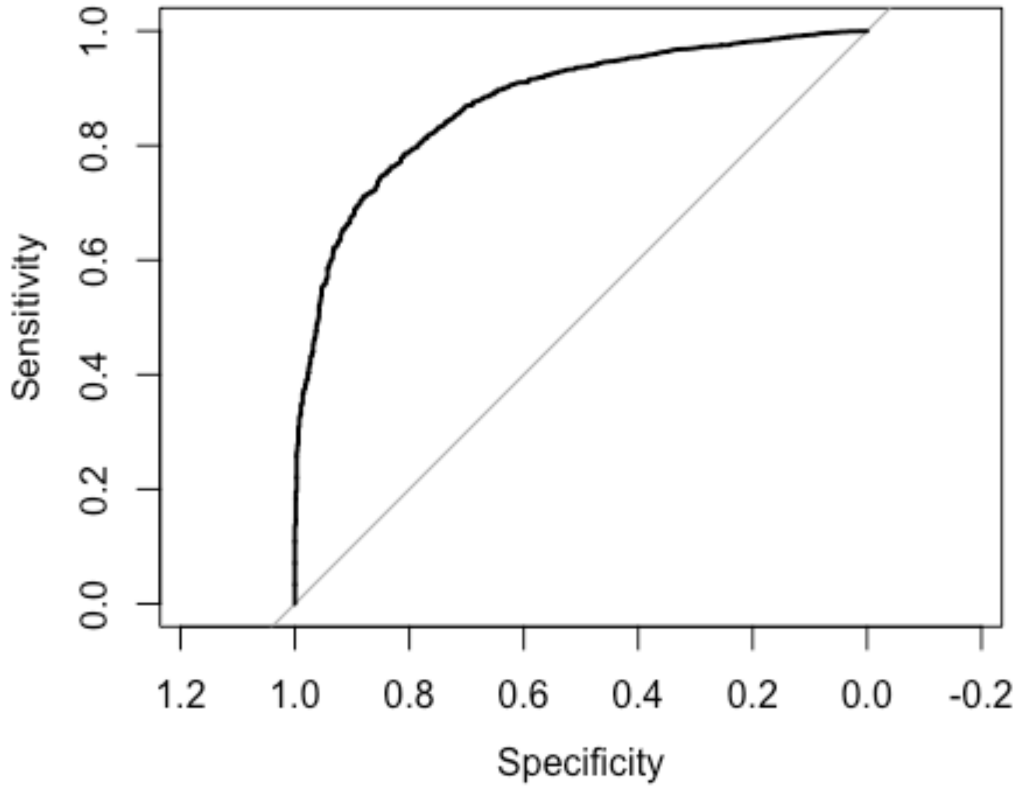
```
##
## Call:
## glm(formula = Attrition_Flag ~ Months_Inactive_12_mon + Total_Revolving_Ba
l +
##     Total_Trans_Amt + Total_Trans_Ct, family = "binomial", data = training
_set)
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -2.543e+00  1.794e-01  -14.18   <2e-16 ***
## Months_Inactive_12_mon -5.025e-01  4.309e-02  -11.66   <2e-16 ***
## Total_Revolving_Bal     1.105e-03  5.540e-05   19.95   <2e-16 ***
## Total_Trans_Amt        -4.762e-04  2.633e-05  -18.08   <2e-16 ***
## Total_Trans_Ct          1.113e-01  4.117e-03   27.04   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5312.4  on 6075  degrees of freedom
```

```
## Residual deviance: 3454.5  on 6071  degrees of freedom
## AIC: 3464.5
##
## Number of Fisher Scoring iterations: 6

# All variables and the intercept have p-value <2e-16, which indicates
```

(b) How good is this model on prediction?

```
predictions = predict(log_md, testing_set, type = "response")
threshold = 0.5
predictions_binary = ifelse(predictions > threshold, 1, 0)

conf_matrix = table(Actual = testing_set$Attrition_Flag, Predicted = predicti
ons_binary)
dimnames(conf_matrix) = list(Actual = c("Actual Churned", "Actual Existing"),
 Predicted = c("Predicted Churn", "Predicted Existing"))

print(conf_matrix)

##                    Predicted
## Actual              Predicted Churn Predicted Existing
##   Actual Churned                264                400
##   Actual Existing               154               3233

TN = conf_matrix[1,1]
FP = conf_matrix[1,2]
FN = conf_matrix[2,1]
TP = conf_matrix[2,2]

# Sensitivity = 0.9580378
TP / (TP+FN)

## [1] 0.954532

# Specificity = 0.4362819
TN / (TN+FP)

## [1] 0.3975904

# Accuracy = 0.8721303
(TP + TN) / sum(conf_matrix)

## [1] 0.8632436

## Adjust with a higher threshold so that sensitivity can be lower, and speci
ficity can be higher:


new_threshold = 0.75
predictions_binary = ifelse(predictions > new_threshold, 1, 0)

conf_matrix = table(Actual = testing_set$Attrition_Flag, Predicted = predicti
```

```
ons_binary)
dimnames(conf_matrix) = list(Actual = c("Actual Churned", "Actual Existing"),
 Predicted = c("Predicted Churn", "Predicted Existing"))

TN = conf_matrix[1,1]
FP = conf_matrix[1,2]
FN = conf_matrix[2,1]
TP = conf_matrix[2,2]

# Sensitivity = 0.8705674
TP / (TP+FN)

## [1] 0.8665486

# Specificity = 0.7106447
TN / (TN+FP)

## [1] 0.7033133

# Accuracy = 0.844236
(TP + TN) / sum(conf_matrix)

## [1] 0.8397926

## Graphing the Area Under the Curve
roc_curve = roc(testing_set$Attrition_Flag, predictions)

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

## Setting direction: controls < cases

print(roc_curve)

##
## Call:
## roc.default(response = testing_set$Attrition_Flag, predictor = prediction
s)
##
## Data: predictions in 664 controls (testing_set$Attrition_Flag 0) < 3387 ca
ses (testing_set$Attrition_Flag 1).
## Area under the curve: 0.877

# Area under the curve: 0.879
plot(roc_curve)
```




```
# Summary:
  # The regression model with default threshold (0.5) has a high sensitivity
of 95.8% and overall accuracy of 87.21%, yet its specificity remains as low a
s 43.63%. With the adjusted threshold of 0.75, there is a significant rise in
 specificity to 71.06%, without affecting the sensitivity and overall accurac
y too much. The high specificity can significantly lower the potential false
positive, help the bank better identify the true customers that are likely to
 leave, which can reduce the cost of avoiding customer churn.

  # Overall, this logistic regression model is quite effective at predicting
customer churn. This model can correctly identify 87.06% of the customer chur
n, 71.06% of the customers who will stay, and has a high overall accuracy of
84.43%. The area under the curve being 0.879 also suggests it has a strong ab
ility to differentiate between churn customer who will churn and those who wi
ll not.
```

Building a model by applying knn

(a) How good is this comparing to the logistic regression model?

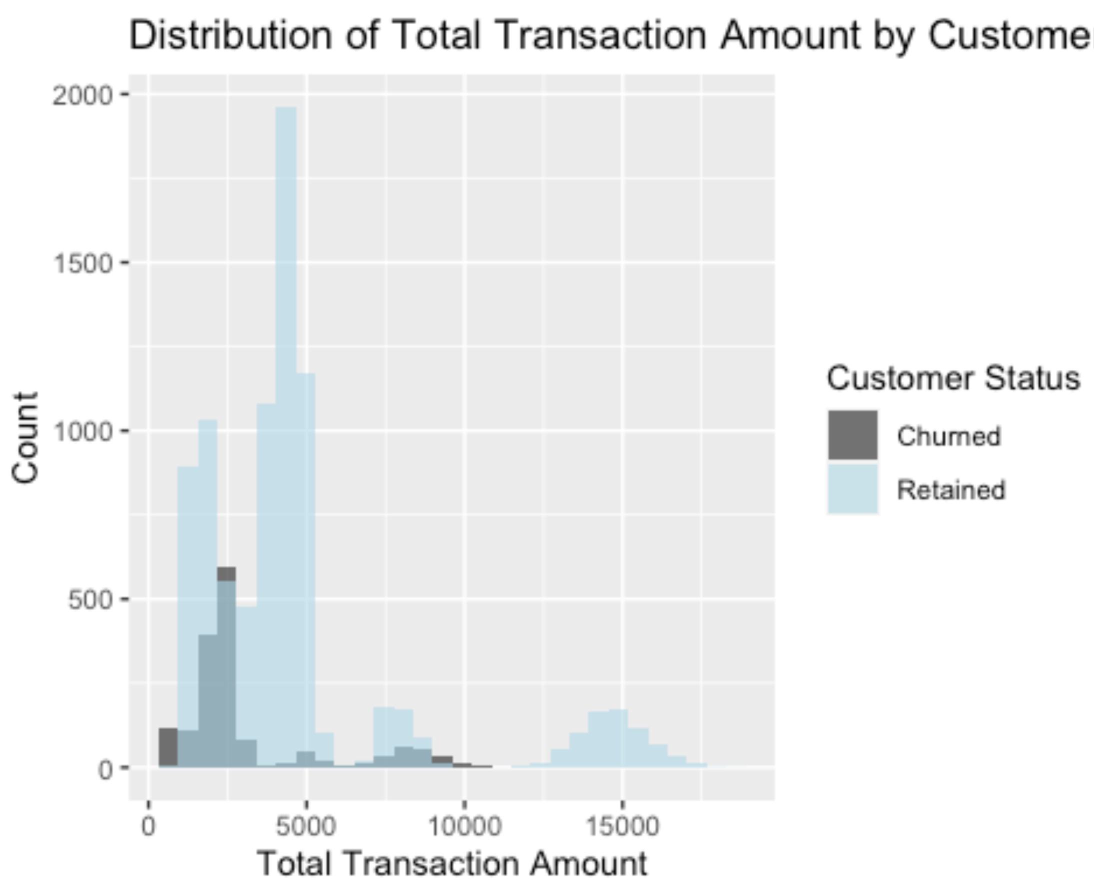
```
# Preparing the scaled data:
factors = c("Months_Inactive_12_mon", "Total_Revolving_Bal", "Total_Trans_Amt
", "Total_Trans_Ct")
```

```
train_scaled = training_set
train_scaled[, factors] = scale(training_set[, factors])

test_scaled = testing_set
test_scaled[, factors] = scale(testing_set[, factors])

# Running KNN with different values of k

accuracy_list = c()
for (k in 1:20) {
  knn_pred = knn(train = train_scaled[, factors],
                 test = test_scaled[, factors],
                 cl = training_set$Attrition_Flag, k = k)
  conf_matrix = table(Predicted = knn_pred, Actual = testing_set$Attrition_Fl
ag)
  accuracy = sum(diag(conf_matrix)) / sum(conf_matrix)
  accuracy_list = c(accuracy_list, accuracy)
}

# Find the k value with the highest accuracy
best_k = which.max(accuracy_list) # 7

# Final KNN model with the best k
knn_md = knn(train = train_scaled[, factors],
                      test = test_scaled[, factors],
                      cl = training_set$Attrition_Flag, k = best_k)

# Confusion matrix for the final model
conf_matrix_2 = table(Predicted = knn_md, Actual = testing_set$Attrition_Fla
g)
dimnames(conf_matrix_2) = list(Actual = c("Actual Churned", "Actual Existing
"), Predicted = c("Predicted Churn", "Predicted Existing"))
print(conf_matrix_2)

##                  Predicted
## Actual             Predicted Churn Predicted Existing
##   Actual Churned               433                102
##   Actual Existing              231               3285

TN = conf_matrix_2[1,1]
FP = conf_matrix_2[1,2]
FN = conf_matrix_2[2,1]
TP = conf_matrix_2[2,2]

# Sensitivity = 0.9371445
TP / (TP+FN)

## [1] 0.9343003

# Specificity = 0.8336449
TN / (TN+FP)
```

```
## [1] 0.8093458

# Accuracy = 0.9234757
(TP + TN) / sum(conf_matrix_2)

## [1] 0.9177981
```

(b) Based on the models and predictions, what can the bank do to reduce customer churn?

```
# Graphing the distribution of total transaction amount:
ggplot(bank, aes(x = Total_Trans_Amt, fill = as.factor(Attrition_Flag))) +
    geom_histogram(bins = 30, alpha = 0.6, position = "identity") +
    scale_fill_manual(values = c("black", "lightblue"),
                      labels = c("Churned", "Retained")) +
    labs(x = "Total Transaction Amount", y = "Count", fill = "Customer Status
") +
    ggtitle("Distribution of Total Transaction Amount by Customer Status")
```




```
# Graphing total transactions count vs amount
ggplot(bank, aes(x = Total_Trans_Ct, y = Total_Trans_Amt, color = as.factor(A
ttrition_Flag))) +
    geom_point(alpha = 0.6) +
    scale_color_manual(values = c("black", "lightblue"),
                       labels = c("Churned", "Retained")) +
    labs(x = "Total Transaction Count", y = "Total Transaction Amount", color
```

```
= "Customer Status") +
    ggtitle("Total Transactions Count vs Amount by Customer Status")
```