R Notebook

By: Perry Son

Overview

This dataset consists of flight details and passenger surveys. Regarding surveys, passengers rate their experience on a scale of 1 to 5, with 0 being non-applicable. These surveys focused on various aspects of their flight. The dataset records passenger details(e.g., class, type of travel) and flight information(e.g., flight distance, delay). The dataset can be found on Kaggle (https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction).

It should be noted that the dataset did not specify the meaning of the scoring system. The notebook assumed that 1 and 5 respectively mapped to worst and best. The dataset also did not indicate the unit of measure for flight distance. The notebook also assumed that distance was measured in miles.

Originally, there were two files: train.csv and test.csv. After I loaded the files, I combined them to perform the same preprocessing techniques. Here, I printed the first 6 observations of the dataset.

```r
set.seed(1111)
current_path = rstudioapi::getActiveDocumentContext()$path
setwd(dirname(current_path ))
data1 <- read.csv("train.csv")
data2 <- read.csv("test.csv")
data <- rbind(data1, data2)
head(data)
```

```
##   X     id Gender     Customer.Type Age  Type.of.Travel    Class
## 1 0  70172   Male    Loyal Customer  13 Personal Travel Eco Plus
## 2 1   5047   Male disloyal Customer  25 Business travel Business
## 3 2 110028 Female    Loyal Customer  26 Business travel Business
## 4 3  24026 Female    Loyal Customer  25 Business travel Business
## 5 4 119299   Male    Loyal Customer  61 Business travel Business
## 6 5 111157 Female    Loyal Customer  26 Personal Travel      Eco
##   Flight.Distance Inflight.wifi.service Departure.Arrival.time.convenient
## 1             460                     3                                 4
## 2             235                     3                                 2
## 3            1142                     2                                 2
## 4             562                     2                                 5
## 5             214                     3                                 3
## 6            1180                     3                                 4
##   Ease.of.Online.booking Gate.location Food.and.drink Online.boarding
## 1                      3             1              5               3
## 2                      3             3              1               3
## 3                      2             2              5               5
## 4                      5             5              2               2
## 5                      3             3              4               5
## 6                      2             1              1               2
##   Seat.comfort Inflight.entertainment On.board.service Leg.room.service
## 1            5                      5                4                3
## 2            1                      1                1                5
## 3            5                      5                4                3
## 4            2                      2                2                5
## 5            5                      3                3                4
## 6            1                      1                3                4
##   Baggage.handling Checkin.service Inflight.service Cleanliness
## 1                4               4                5           5
## 2                3               1                4           1
## 3                4               4                4           5
## 4                3               1                4           2
## 5                4               3                3           3
## 6                4               4                4           1
##   Departure.Delay.in.Minutes Arrival.Delay.in.Minutes            satisfaction
## 1                         25                       18 neutral or dissatisfied
## 2                          1                        6 neutral or dissatisfied
## 3                          0                        0               satisfied
## 4                         11                        9 neutral or dissatisfied
## 5                          0                        0               satisfied
## 6                          0                        0 neutral or dissatisfied
```

Preprocessing and Data Cleaning

I dropped columns that contained non-essential predictors.

```
data <- subset(data, select = -c(X, id))
```

I mapped the categorical non-numerical predictors to different ranges.

```
data$Customer.Type <- ifelse(data$Customer.Type=="Local Customer", 1, 0)
data$Gender <- ifelse(data$Gender=="Female", 1, 0)
data$Type.of.Travel <- ifelse(data$Type.of.Travel=="Business travel", 1, 0)
data$Class[data$Class == "Eco"] <- 0
data$Class[data$Class == "Eco Plus"] <- 1
data$Class[data$Class == "Business"] <- 2
```

I listed the column names.

```
print(colnames(data))
```

```
##  [1] "Gender"                            "Customer.Type"
##  [3] "Age"                               "Type.of.Travel"
##  [5] "Class"                             "Flight.Distance"
##  [7] "Inflight.wifi.service"             "Departure.Arrival.time.convenient"
##  [9] "Ease.of.Online.booking"            "Gate.location"
## [11] "Food.and.drink"                    "Online.boarding"
## [13] "Seat.comfort"                      "Inflight.entertainment"
## [15] "On.board.service"                  "Leg.room.service"
## [17] "Baggage.handling"                  "Checkin.service"
## [19] "Inflight.service"                  "Cleanliness"
## [21] "Departure.Delay.in.Minutes"        "Arrival.Delay.in.Minutes"
## [23] "satisfaction"
```

. I checked which column contained NA's

```
print(sapply(data, function(y) sum(length(which(is.na(y))))))
```

```
##                            Gender                     Customer.Type
##                                 0                                 0
##                               Age                    Type.of.Travel
##                                 0                                 0
##                             Class                   Flight.Distance
##                                 0                                 0
##             Inflight.wifi.service Departure.Arrival.time.convenient
##                                 0                                 0
##            Ease.of.Online.booking                     Gate.location
##                                 0                                 0
##                    Food.and.drink                   Online.boarding
##                                 0                                 0
##                      Seat.comfort            Inflight.entertainment
##                                 0                                 0
##                  On.board.service                  Leg.room.service
##                                 0                                 0
##                  Baggage.handling                   Checkin.service
##                                 0                                 0
##                  Inflight.service                       Cleanliness
##                                 0                                 0
##        Departure.Delay.in.Minutes          Arrival.Delay.in.Minutes
##                                 0                               393
##                      satisfaction
##                                 0
```

The score of 0 indicated non-applicable reviews. I also checked which survey column contained scores of 0's.

```
print(sapply(data, function(y) sum(length(which(y==0)))))
```

```
##                            Gender                     Customer.Type
##                             63981                            129880
##                               Age                    Type.of.Travel
##                                 0                             40187
##                             Class                   Flight.Distance
##                             58309                                 0
##             Inflight.wifi.service Departure.Arrival.time.convenient
##                              3916                              6681
##            Ease.of.Online.booking                     Gate.location
##                              5682                                 1
##                    Food.and.drink                   Online.boarding
##                               132                              3080
##                      Seat.comfort            Inflight.entertainment
##                                 1                                18
##                  On.board.service                  Leg.room.service
##                                 5                               598
##                  Baggage.handling                   Checkin.service
##                                 0                                 1
##                  Inflight.service                       Cleanliness
##                                 5                                14
##        Departure.Delay.in.Minutes          Arrival.Delay.in.Minutes
##                             73356                             72753
##                      satisfaction
##                                 0
```

I dropped observations that contained NA's or scores of 0.

```
data <- data[!(is.na(data$Arrival.Delay.in.Minutes)),]
data <- data[!(data$Gate.location==0),]
data <- data[!(data$Food.and.drink==0),]
data <- data[!(data$Inflight.wifi.service==0),]
data <- data[!(data$Departure.Arrival.time.convenient==0),]
data <- data[!(data$Ease.of.Online.booking==0),]
data <- data[!(data$Online.boarding==0),]
data <- data[!(data$Seat.comfort==0),]
data <- data[!(data$Inflight.entertainment==0),]
data <- data[!(data$On.board.service==0),]
data <- data[!(data$Leg.room.service==0),]
data <- data[!(data$Checkin.service==0),]
data <- data[!(data$Inflight.service==0),]
data <- data[!(data$Cleanliness==0),]
```

I printed the number of observations in the dataset. There were 119204 observations.

```
print(nrow(data))
```

```
## [1] 119204
```

I converted the satisfaction column to a factor.

```
data$satisfaction<-as.factor(data$satisfaction)
```

I performed a train/test split with a ratio of 80:20.

```
i <- sample(1:nrow(data), nrow(data)*0.80, replace=FALSE)
train <- data[i,]
test <- data[-i,]
```

There are 95363 training observations.

```
print(nrow(train))
```

```
## [1] 95363
```

Data Exploration

Minimum age was 7 years old. Maximum age was 85 years old.

```
range(train$Age)
```

```
## [1]  7 85
```

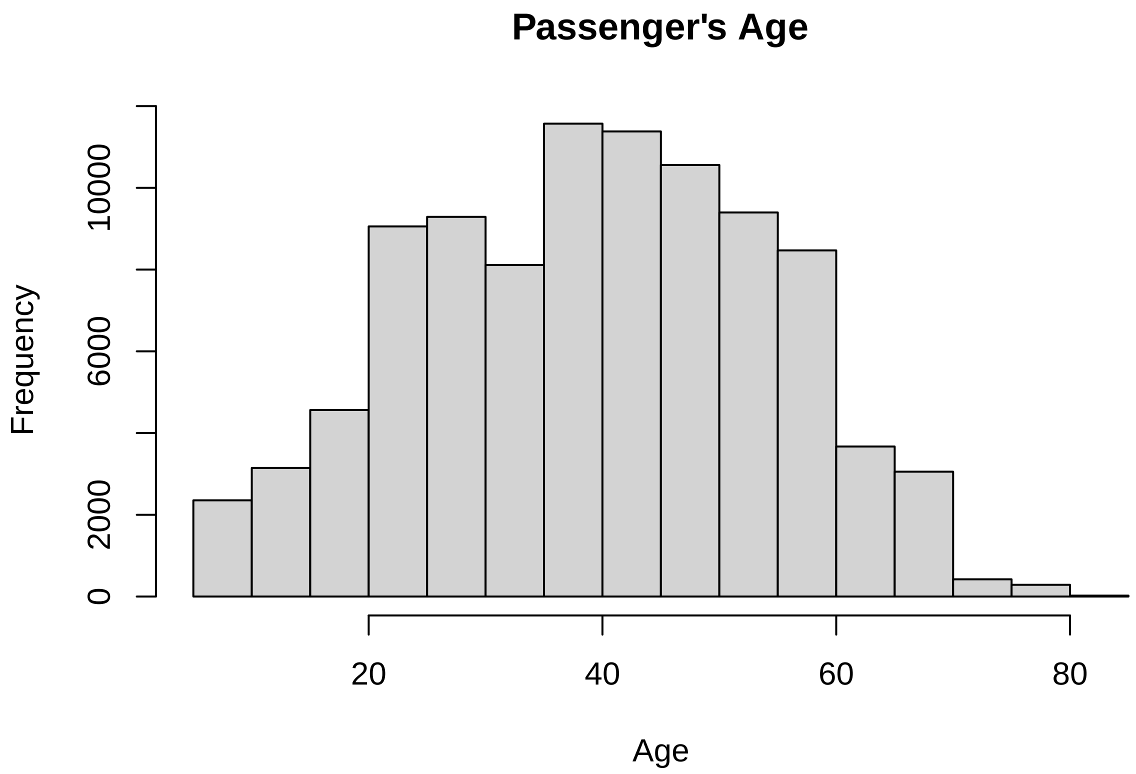
The average flight distance was 1225.62 miles.

```
mean(train$Flight.Distance)
```

```
## [1] 1225.62
```

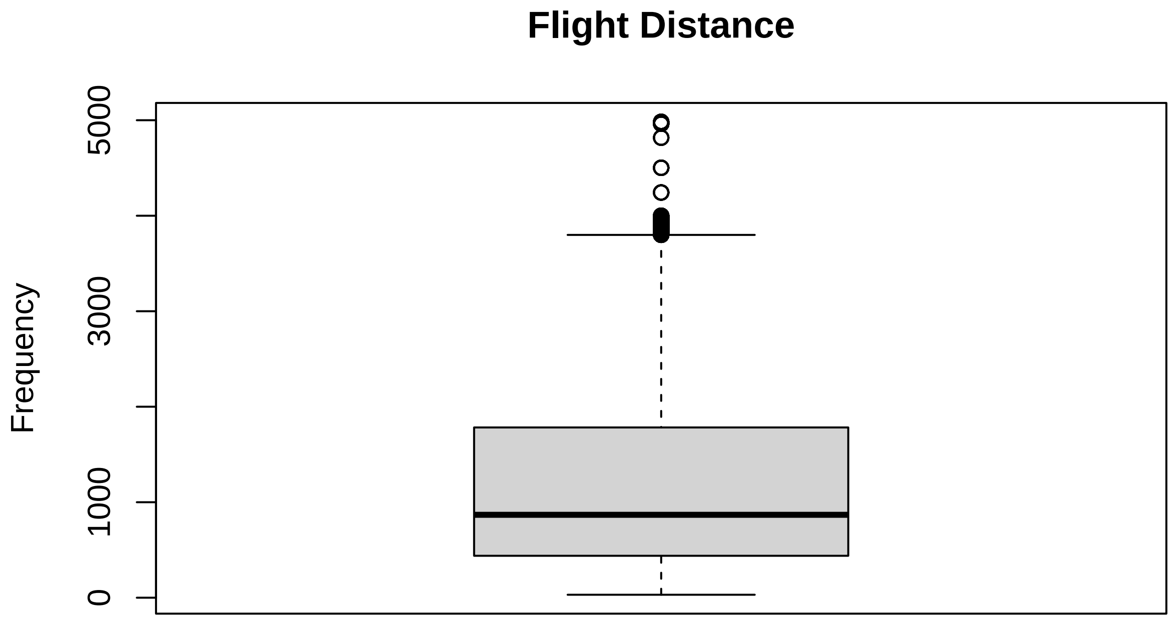
I plotted a histogram of the passengers' age. The ages were centered around the 20's to 60's.

```
hist(train$Age, main = "Passenger's Age", xlab="Age")
```



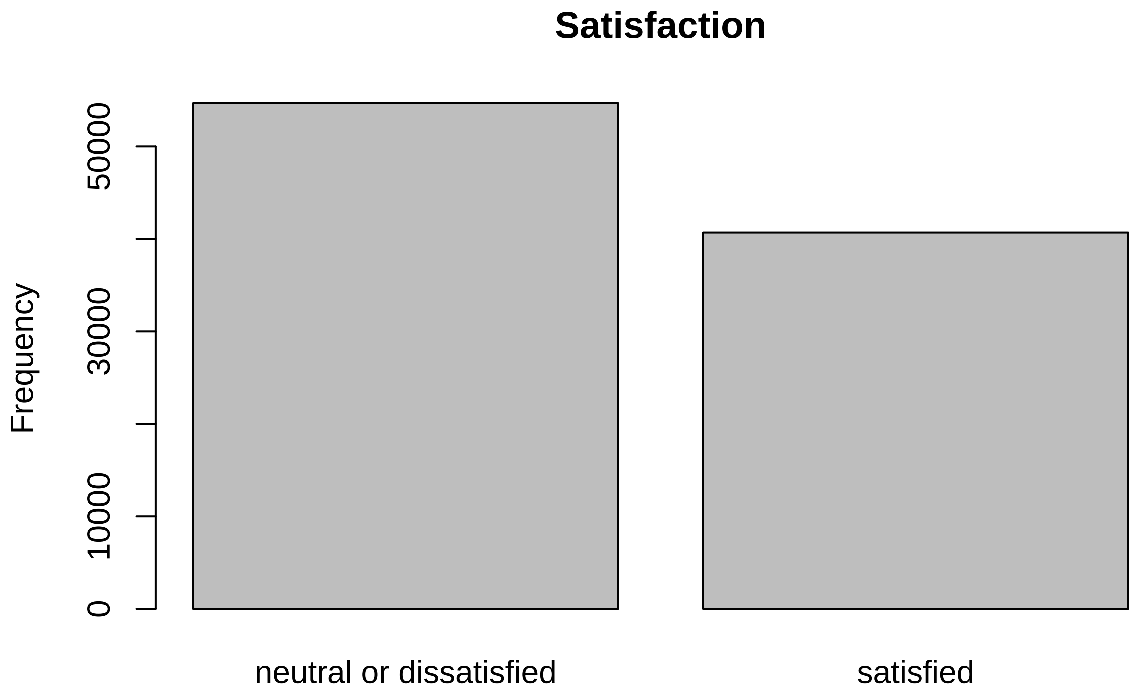
I plotted a box plot of the flight distance. A majority of flights had a flight distance of ~600 to ~1900 miles.

```
boxplot(train$Flight.Distance,  ylab="Frequency", main="Flight Distance")
```



I plotted a bar plot of the overall satisfaction level. More passengers were neutral or dissatisfied with their flight experience.

```
barplot(table(train$satisfaction),  ylab="Frequency", main="Satisfaction")
```



Machine Learning Models

I used Logistic Regression, K-nearest neighbors (KNN), and Decision Trees to predict passenger's satisfaction level. The formula used all of the predictors to determine overall flight satisfaction.

I fitted a logistic regression model on the training data. I also printed a summary of the model.

```
log_reg_model <- glm(satisfaction~., data=train, family=binomial)
summary(log_reg_model)
```

```
##
## Call:
## glm(formula = satisfaction ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3627  -0.3392  -0.0631   0.3476   3.9271
##
## Coefficients: (1 not defined because of singularities)
##                                    Estimate Std. Error  z value Pr(>|z|)
## (Intercept)                      -1.315e+01  1.048e-01 -125.529  < 2e-16 ***
## Gender                           -9.319e-02  2.237e-02   -4.166 3.11e-05 ***
## Customer.Type                            NA         NA       NA       NA
## Age                               1.425e-02  8.042e-04   17.721  < 2e-16 ***
## Type.of.Travel                    2.258e+00  3.497e-02   64.558  < 2e-16 ***
## Class1                             1.687e-01  4.624e-02    3.648 0.000264 ***
## Class2                             1.057e+00  3.003e-02   35.214  < 2e-16 ***
## Flight.Distance                    2.485e-04  1.236e-05   20.102  < 2e-16 ***
## Inflight.wifi.service              7.510e-01  1.392e-02   53.940  < 2e-16 ***
## Departure.Arrival.time.convenient -3.448e-01  1.267e-02  -27.218  < 2e-16 ***
## Ease.of.Online.booking             2.530e-01  1.448e-02   17.466  < 2e-16 ***
## Gate.location                     -1.892e-01  1.229e-02  -15.390  < 2e-16 ***
## Food.and.drink                    -7.915e-02  1.198e-02   -6.608 3.90e-11 ***
## Online.boarding                    9.505e-01  1.272e-02   74.729  < 2e-16 ***
## Seat.comfort                       5.041e-02  1.296e-02    3.891 9.99e-05 ***
## Inflight.entertainment             3.006e-01  1.600e-02   18.788  < 2e-16 ***
## On.board.service                   3.168e-01  1.169e-02   27.087  < 2e-16 ***
## Leg.room.service                   3.227e-01  1.011e-02   31.926  < 2e-16 ***
## Baggage.handling                   5.700e-02  1.298e-02    4.392 1.12e-05 ***
## Checkin.service                    2.902e-01  9.643e-03   30.095  < 2e-16 ***
## Inflight.service                   4.782e-02  1.362e-02    3.512 0.000445 ***
## Cleanliness                        1.693e-01  1.327e-02   12.753  < 2e-16 ***
## Departure.Delay.in.Minutes         3.702e-03  1.119e-03    3.308 0.000939 ***
## Arrival.Delay.in.Minutes          -7.410e-03  1.104e-03   -6.713 1.91e-11 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 130142  on 95362  degrees of freedom
## Residual deviance:  52767  on 95340  degrees of freedom
## AIC: 52813
##
## Number of Fisher Scoring iterations: 6
```

The warning indicates that some predictors are perfectly correlated. Here, I printed the confusion matrix and accuracy. In this case, logistic regression had an accuracy of ~0.89.

```
probs <- predict(log_reg_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
```

```
pred <- ifelse(probs>0.5, 2, 1)
acc1 <- mean(pred==as.integer(test$satisfaction))
pred <- ifelse(pred==2, "satisfied", "neutral or dissatisfied")
table(pred, test$satisfaction)
```

```
##
## pred                      neutral or dissatisfied satisfied
##   neutral or dissatisfied                   12362      1392
##   satisfied                                  1293      8794
```

```
print(paste("logistic reg accuracy = ", acc1))
```

```
## [1] "logistic reg accuracy =  0.887378885113879"
```

I fitted the KNN algorithm on the training data. I specified the number of neighbors to be the $\sqrt{N}/2$ with N being the number of training observations. Because there were two classes, I rounded K upward to get an odd number.

```
library(class)
labels <- ifelse(train$satisfaction=="satisfied", 1, 0)
k_amt <- ceiling(sqrt(length(labels))/2)
knn_pred <- knn(train=train[, -23], test=test[,-23],
cl=labels, k=k_amt)
```

I printed the confusion matrix and accuracy for KNN. KNN had an accuracy of ~0.71.

```
knn_pred <- ifelse(knn_pred==1, "satisfied", "neutral or dissatisfied")
acc <- length(which(knn_pred == test$satisfaction)) /length(knn_pred)
table(knn_pred, test$satisfaction)
```

```
##
## knn_pred                  neutral or dissatisfied satisfied
##   neutral or dissatisfied                   11157      4309
##   satisfied                                  2498      5877
```

```
print(paste("K Nearest Neighbors accuracy = ", acc))
```

```
## [1] "K Nearest Neighbors accuracy =  0.714483452875299"
```

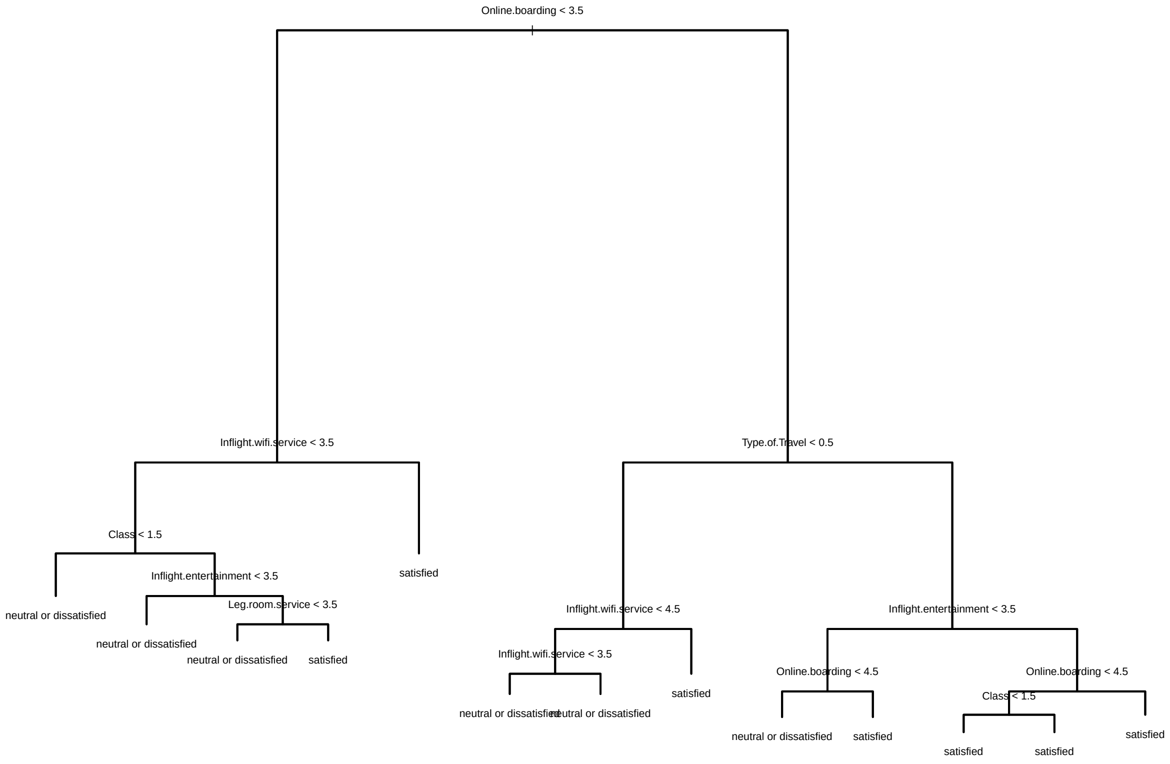
I fitted the decision tree algorithm on the training data. I also printed the tree structure.

Regarding the diagram, decision trees select features that best splits the data. Each node contains a feature and threshold value, which determines the path traversal.

```
library(tree)
```

```
## Warning: package 'tree' was built under R version 4.1.3
```

```
decision_tree <- tree(satisfaction~., data=train)
par(cex = .3)
plot(decision_tree)
text(decision_tree)
```



I printed the confusion matrix and accuracy for the decision tree. The decision tree had an accuracy of ~0.90.

```
tree_pred <- predict(decision_tree, newdata=test, type="class")
table(tree_pred, test$satisfaction)
```

```
##
## tree_pred                 neutral or dissatisfied satisfied
##   neutral or dissatisfied                   12535      1260
##   satisfied                                  1120      8926
```

```
acc <- mean(tree_pred == test$satisfaction)
print(paste("Decision Tree accuracy = ", acc))
```

```
## [1] "Decision Tree accuracy =  0.900171972652154"
```

Thoughts

Logistic Regression creates decision boundaries to partition observations into similar classes. For this dataset, the survey columns measured satisfaction on a level of 1 to 5. Due to the small set of choices, these columns didn't simulate any complex non-linear patterns. Logistic Regression assumes the predictor and target variables are linearly related with the log odds. Given that the model achieved an accuracy of ~0.89, the variables already possessed a linear relationship.

KNN didn't perform that well with an accuracy of ~0.71. For a given observation, KNN chooses a target class based on proximity to N nearest neighbors. I believe there were some underlying factors that affected overall flight satisfaction. Despite sharing similar values, the observations didn't incorporate these factors. Hence, KNN had a subpar performance.

The Decision Tree had the best performance with an accuracy of ~0.9. Since the data rarely varied, the decision tree didn't suffer from overfitting. As a result, it performed well on the test set.