Final Group Project Report

Data and Prediction Analysis of Airline Passenger Satisfaction

Project Members

Vikas Pathak (vpathak1) - A20460927

Ruturaj Joshi (rjoshi17) - A20497857

Illinois Institute of Technology
CSP571 - Data Preparation and Analysis
Professor: Jawahar Panchal

ABSTRACT

The airline industry is currently the biggest industry of transportation in the world. It focuses on customer acquisition strategies because providing excellent service to customers is a good method to win over repeat business. With this project, we aim to extract valuable insights from data that can be utilized to make business decisions toward improving customer satisfaction. In addition to this, we aim to build a classifier that can efficiently identify customer satisfaction levels.

OVERVIEW

Transportation services have become core community necessities for both daily activities and travel. Most individuals choose air transportation for long-distance travel since it is more efficient and effective in terms of time. Air transport can reach regions where other forms of transportation, such as land and water, cannot.

Global air transport increased by 5% per year, or doubling every 15 years.As a result aviation professionals must ensure that the quality of service they deliver has an impact on customer satisfaction. Satisfaction is viewed not only as a customer goal to be achieved as a result of poor service, but also as a company objective, as a means of increasing customer retention rates and creating profits. If the service or product meets the customer's expectations, he will be satisfied, raising the level of consumer loyalty. In contrast, if service delivery fails miserably of customer expectations, service quality will be deemed poor, resulting in a decrease in customer loyalty; improving service quality is one strategy to boost customer loyalty. Customer loyalty is defined as a commitment or customer principle to always choose the same service or product in the future.

Customer behavior analysis is crucial in today's market. If a correlation between client loyalty and certain of their features can be found, improvements can be made by moving beyond these aspects.

In order to become more competitive, specially after Covid-19 pandemic, it has become even more crucial for airline companies to understand the satisfaction of their customers.

Understanding the variables that affect airline passengers' pleasure is essential to marketing techniques like promotional vouchers, target campaigns, etc., and goes far beyond simply allowing for service improvement. And our objective is to build a model that can detect what factors result in satisfaction and where these companies could improve their performance.

SPECIFIC QUESTIONS

- Which predictors influence customer satisfaction significantly?
- Determine how predictors are correlated with each other
- Identify areas in which airline operator needs improvement
- Build a model that can predict the customer satisfaction with acceptable performance matrix

PROPOSED METHODOLOGY

- Gather relevant datasets
- Preparing the data includes that data in the provided data set are having a proper recognizable data type, filling in null or empty data points with some value, where it can be mean, median or zero, adding missing values in the data set, detection and removal of any outliers
- Using R visualization tools to visualize the data. Finding relationships between numerous factors by using correlation plots
- Selecting significant and useful model features
- Dividing the dataset into a training and a testing set
- Selection of a model and training it with the train dataset
- Putting the model to the test with a test dataset
- Checking the model's correctness by comparing the anticipated and actual output values for the test data

DATASET

The data we utilized for this research came from Kaggle. There are 24 attributes in the data set, of which the input variables include 4 numerical continuous variables, 4 class discrete variables and 14 qualitative sequential variables, indicating the customer's satisfaction with relevant services (0 – 5 points). The data used in this study mainly contain three dimensions of information, including basic information, flight information and satisfaction information. The constituent elements and the specific variable names and variable attributes are shown in Table below. The output variables are category variables, that is, passengers' final satisfaction and dissatisfaction or neutral attitude.

The data set includes information about:

Field	DataType	Description
ID	int	Unique passenger identifier
Gender	chr	Gender of the passenger (Female/Male)
Age	int	Age of the passenger
Customer Type	chr	Type of airline customer (First-time/Returning)
Type of Travel	chr	Purpose of the flight (Business/Personal)
Class	chr	Travel class in the airplane for the passenger seat
Flight Distance	int	Flight distance in miles
Departure Delay	int	Flight departure delay in minutes
Arrival Delay	int	Flight arrival delay in minutes
Departure and Arrival Time Convenience	int	Satisfaction level with the convenience of the flight departure and arrival times from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Ease of Online Booking	int	Satisfaction level with the online booking experience from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Check-in Service	int	Satisfaction level with the check-in service from 1 (lowest) to 5 (highest) - 0 means "not applicable"

Online Boarding	int	Satisfaction level with the online boarding experience from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Gate Location	int	Satisfaction level with the gate location in the airport from 1 (lowest) to 5 (highest) - 0 means "not applicable"
On-board Service	int	Satisfaction level with the on-boarding service in the airport from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Seat Comfort	int	Satisfaction level with the comfort of the airplane seat from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Leg Room Service	int	Satisfaction level with the leg room of the airplane seat from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Cleanliness	int	Satisfaction level with the cleanliness of the airplane from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Food and Drink	int	Satisfaction level with the food and drinks on the airplane from 1 (lowest) to 5 (highest) - 0 means "not applicable"
In-flight Service	int	Satisfaction level with the in-flight service from 1 (lowest) to 5 (highest) - 0 means "not applicable"
In-flight Wifi Service	int	Satisfaction level with the in-flight Wifi service from 1 (lowest) to 5 (highest) - 0 means "not applicable"
In-flight Entertainment	int	Satisfaction level with the in-flight entertainment from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Baggage Handling	int	Satisfaction level with the baggage handling from the airline from 1 (lowest) to 5 (highest) - 0 means "not applicable"
Satisfaction	chr	Overall satisfaction level with the airline (Satisfied/Neutral or unsatisfied)

DATA PREPARATION

As part of data preparation, we have performed below steps:

1. Typecast string data values into a factor data type to interpret those values as categorical values.
2. Determine the number of rows that NA values and form a strategy to handle such rows.
3. Review the dataset closely and act on assumptions if any.

```
# Read data set
airlineData <- read.csv("airline_passenger_satisfaction.csv")

# Convert string data types into factors i.e. categorical data
airlineData$Gender <- as.factor(airlineData$Gender)
airlineData$Customer.Type <- as.factor(airlineData$Customer.Type)
airlineData$Type.of.Travel <- as.factor(airlineData$Type.of.Travel)
airlineData$Class <- as.factor(airlineData$Class)
airlineData$Satisfaction <- as.factor(airlineData$Satisfaction)

# Describe data set
str(airlineData)
```

```
'data.frame':129880 obs. of  24 variables:
 $ ID                                : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Gender                            : Factor w/ 2 levels "Female","Male": 2 1
2 2 1 2 2 1 2 1 ...
 $ Age                               : int  48 35 41 50 49 43 43 60 50 38 ...
 $ Customer.Type                     : Factor w/ 2 levels
"First-time","Returning": 1 2 2 2 2 2 2 2 2 2 ...
 $ Type.of.Travel                    : Factor w/ 2 levels
"Business","Personal": 1 1 1 1 1 1 1 1 1 1 ...
 $ Class                             : Factor w/ 3 levels
"Business","Economy",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Flight.Distance                   : int  821 821 853 1905 3470 3788 1963 853
2607 2822 ...
 $ Departure.Delay                   : int  2 26 0 0 0 0 0 0 0 13 ...
 $ Arrival.Delay                     : int  5 39 0 0 1 0 0 3 0 0 ...
 $ Departure.and.Arrival.Time.Convenience: int  3 2 4 2 3 4 3 3 1 2 ...
 $ Ease.of.Online.Booking            : int  3 2 4 2 3 4 3 4 1 5 ...
 $ Check.in.Service                  : int  4 3 4 3 3 3 4 3 3 3 ...
 $ Online.Boarding                   : int  3 5 5 4 5 5 4 4 2 5 ...
 $ Gate.Location                     : int  3 2 4 2 3 4 3 4 1 2 ...
 $ On.board.Service                  : int  3 5 3 5 3 4 5 3 4 5 ...
```

```
 $ Seat.Comfort                     : int  5 4 5 5 4 4 5 4 3 4 ...
 $ Leg.Room.Service                 : int  2 5 3 5 4 4 5 4 4 5 ...
 $ Cleanliness                      : int  5 5 5 4 5 3 4 4 3 4 ...
 $ Food.and.Drink                   : int  5 3 5 4 4 3 5 4 3 2 ...
 $ In.flight.Service                : int  5 5 3 5 3 4 5 3 4 5 ...
 $ In.flight.Wifi.Service           : int  3 2 4 2 3 4 3 4 4 2 ...
 $ In.flight.Entertainment          : int  5 5 3 5 3 4 5 3 4 5 ...
 $ Baggage.Handling                 : int  5 5 3 5 3 4 5 3 4 5 ...
 $ Satisfaction                     : Factor w/ 2 levels "Neutral or
Dissatisfied",..: 1 2 2 2 2 2 2 2 1 2 ...
```

```
df <- airlineData[rowSums(is.na(airlineData)) > 0, ]
df
```

	ID	Gender	Age	Customer.Type	Type.of.Travel	Class	Flight.Distance	Departure.Delay	Arrival.Delay	Departure.and.Arrival.Time.Convenience	Ease.of.C
247	247	Male	11	Returning	Business	Business	719	38	NA	1	
884	884	Male	39	Returning	Business	Business	396	0	NA	3	
1966	1966	Male	36	Returning	Business	Economy	383	2	NA	4	
2408	2408	Female	55	Returning	Business	Business	2904	58	NA	5	
2449	2449	Male	21	Returning	Personal	Economy Plus	767	5	NA	3	
2615	2615	Male	46	Returning	Personal	Economy	181	0	NA	4	
2679	2679	Female	64	Returning	Personal	Economy	622	109	NA	3	
2755	2755	Female	18	Returning	Personal	Economy	772	0	NA	4	
2862	2862	Male	72	Returning	Business	Business	475	15	NA	4	
3449	3449	Male	34	Returning	Personal	Economy	213	0	NA	4	
3537	3537	Female	25	Returning	Personal	Economy Plus	557	32	NA	4	
3638	3638	Male	35	Returning	Business	Business	1534	0	NA	1	
3639	3639	Male	44	Returning	Business	Business	431	6	NA	5	
3699	3699	Female	36	Returning	Business	Business	1840	166	NA	2	
3711	3711	Male	9	Returning	Personal	Economy	427	0	NA	0	
3852	3852	Male	32	Returning	Personal	Economy	650	22	NA	1	
4900	4900	Male	52	Returning	Business	Business	1020	118	NA	5	
5136	5136	Female	58	Returning	Business	Business	184	14	NA	1	
5298	5298	Female	34	Returning	Personal	Economy	285	0	NA	1	
5411	5411	Male	9	Returning	Personal	Economy	785	0	NA	5	
5978	5978	Male	51	Returning	Business	Business	691	0	NA	5	
6311	6311	Male	32	Returning	Business	Business	240	70	NA	1	
6557	6557	Male	47	Returning	Business	Business	2022	0	NA	5	

```
cat("Dataset has ", nrow(df), " rows that have NA values for at least one column.")

Dataset has  393  rows that have NA values for at least one column.
```

In our dataset, 393 rows have an *NA* value for at least one column. We have decided to remove these rows since these are relatively low row counts when compared with our entire dataset and hence do not have any significant impact on the resulting dataset.

```
airlineData <- airlineData[rowSums(is.na(airlineData)) == 0, ]
cat("Row count after removing rows: ", nrow(airlineData))
```

```
Row count after removing rows:  129487
```

Now, we will be removing rows that have a rating of 0 for at least one of the survey criteria. Here, we are assuming that having a 0 rating means it has been skipped by the customer and not the other way around. In this case, the number of such rows is relatively low when compared to the size of our entire dataset. Therefore, there is no significant impact of removing these rows.

```
airlineData <- airlineData[!(airlineData$Departure.and.Arrival.Time.Convenience ==
0 | airlineData$Ease.of.Online.Booking == 0 | airlineData$Check.in.Service == 0 |
airlineData$Online.Boarding == 0 | airlineData$Gate.Location == 0 |
airlineData$On.board.Service == 0 | airlineData$Seat.Comfort == 0 |
airlineData$Leg.Room.Service == 0 | airlineData$Cleanliness == 0 |
airlineData$Food.and.Drink == 0 | airlineData$In.flight.Service == 0 |
airlineData$In.flight.Wifi.Service == 0 | airlineData$In.flight.Entertainment ==
0),]

cat("Row count after removing rows: ", nrow(airlineData))
```

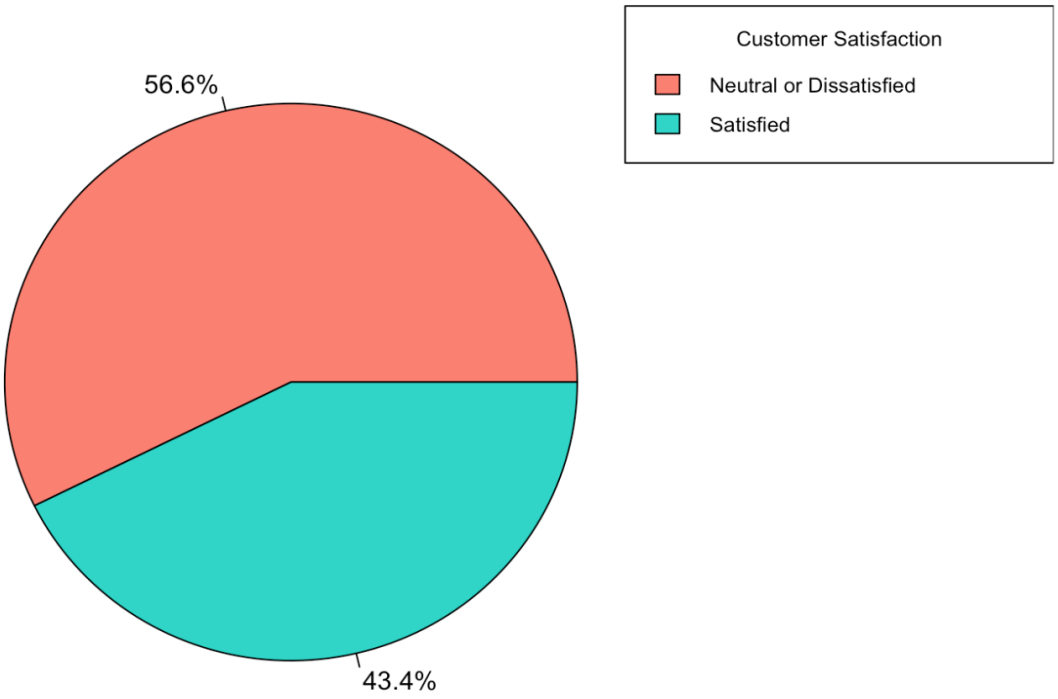
```
Row count after removing rows:  119204
```

EXPLORATORY ANALYSIS

1. Response variable distribution

Here we are interested in knowing the satisfaction of the customer and hence our response variable, in this case, is "Satisfaction".

```
pie(table(airlineData$Satisfaction), labels = c("56.6%", "43.4%"), col =
c("#FA8072", "#30D5C8"))
legend("topright", c("Neutral or Dissatisfied", "Satisfied"), cex = 0.8, fill =
c("#FA8072", "#30D5C8"), title = "Customer Satisfaction")
```

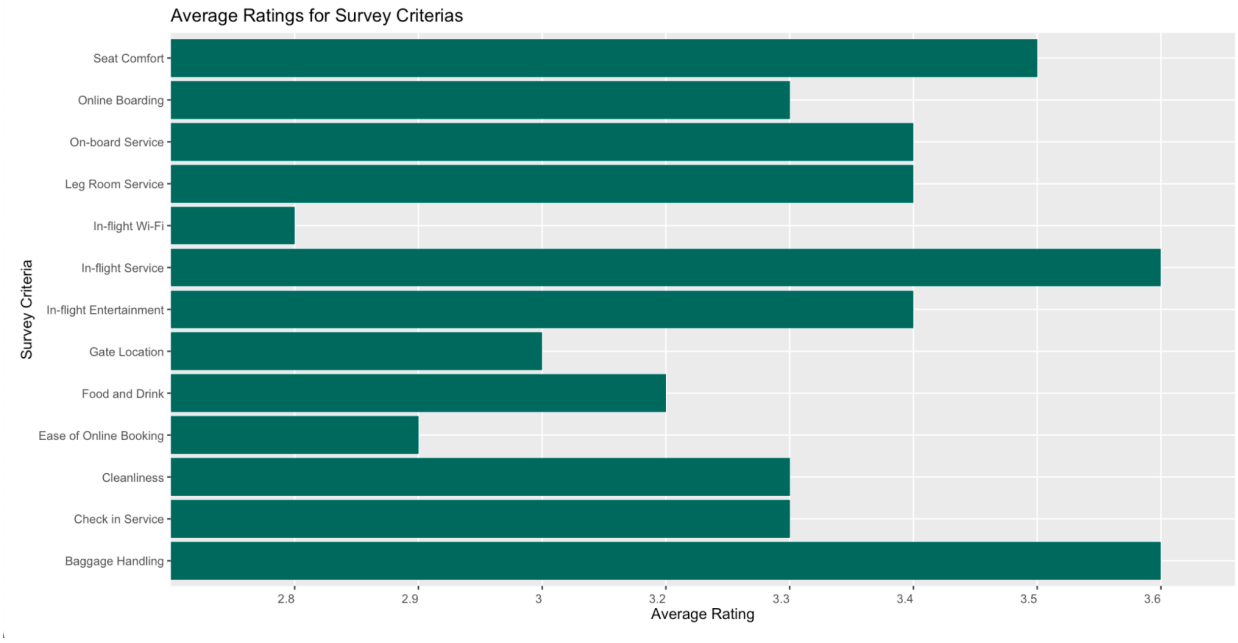


In our dataset, 56.6% of customers are "Neutral or Dissatisfied" and the remaining 43.4% are "Satisfied". With this observation, we can say that our dataset is balanced as there are enough data points to represent each class.

2. Average rating for provided services in survey

```
surveyCriteria <- c("Ease of Online Booking", "Check in Service", "Online
Boarding", "Gate Location", "On-board Service", "Seat Comfort", "Leg Room
Service", "Cleanliness", "Food and Drink", "In-flight Service", "In-flight
Wi-Fi", "In-flight Entertainment", "Baggage Handling")
avgRatings <- c(mean(airlineData$Ease.of.Online.Booking),
mean(airlineData$Check.in.Service), mean(airlineData$Online.Boarding),
mean(airlineData$Gate.Location), mean(airlineData$On.board.Service),
mean(airlineData$Seat.Comfort), mean(airlineData$Leg.Room.Service),
mean(airlineData$Cleanliness), mean(airlineData$Food.and.Drink),
mean(airlineData$In.flight.Service),
mean(airlineData$In.flight.Wifi.Service),
mean(airlineData$In.flight.Entertainment),
mean(airlineData$Baggage.Handling))
avgRatings <- round(avgRatings, digits = 1)

ggplot(as.data.frame(cbind(surveyCriteria, avgRatings)), aes(x =
surveyCriteria, y = avgRatings)) +
  geom_bar(stat = 'identity', fill = "#01675E") +
  theme(axis.text.x = element_text(angle = 0, vjust = 0.5, hjust = 1)) +
  ggtitle("Average Ratings for Survey Criterias") +
  xlab("Survey Criteria") +
  ylab("Average Rating") +
  coord_flip()
```



Top 5 highly rated services:

1. Baggage handling
2. In-flight service
3. Seat comfort
4. In-flight entertainment
5. Leg room service

Top 5 worst rated services:

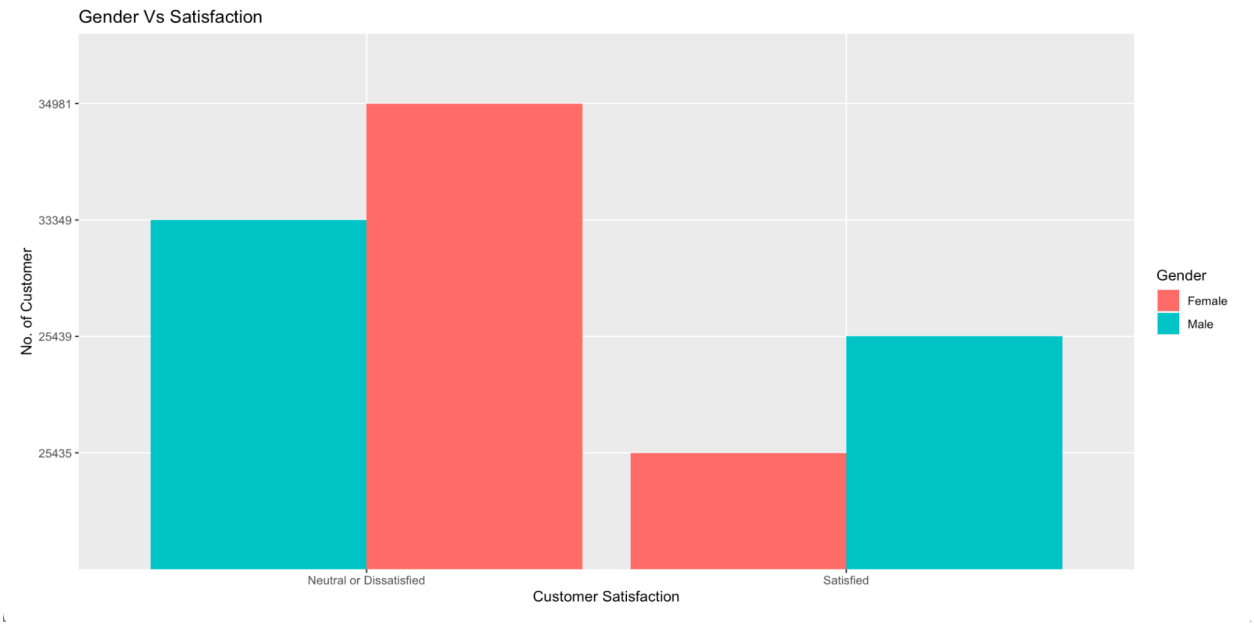
1. In-flight Wi-Fi
2. Ease of online booking
3. Gate location
4. Food and drink
5. Cleanliness

3. Satisfaction by gender

```
dissatisfiedMales <- nrow(airlineData %>% filter(Gender == "Male" & Satisfaction ==
"Neutral or Dissatisfied"))
satisfiedMales <- nrow(airlineData %>% filter(Gender == "Male" & Satisfaction ==
"Satisfied"))
dissatisfiedFemales <- nrow(airlineData %>% filter(Gender == "Female" &
Satisfaction == "Neutral or Dissatisfied"))
satisfiedFemales <- nrow(airlineData %>% filter(Gender == "Female" & Satisfaction
== "Satisfied"))

gender <- c("Male", "Male", "Female", "Female")
response <- c("Satisfied", "Neutral or Dissatisfied", "Satisfied", "Neutral or
Dissatisfied")
cnt <- c(satisfiedMales, dissatisfiedMales, satisfiedFemales, dissatisfiedFemales)

ggplot(as.data.frame(cbind(gender, response, cnt)), aes(x = response, y = cnt, fill
= gender)) +
  geom_bar(stat = 'identity', position = 'dodge') +
  ggtitle("Gender Vs Satisfaction") +
  labs(fill = "Gender") +
  xlab("Customer Satisfaction") +
  ylab("No. of Customer")
```



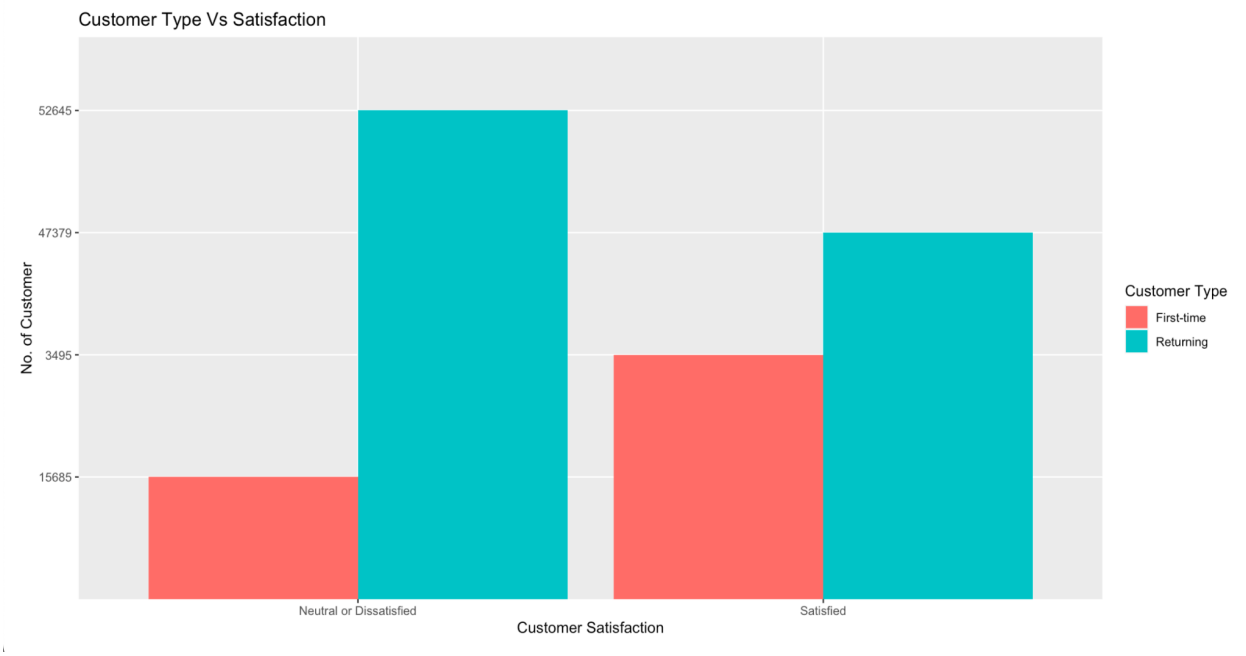
Number of male passengers satisfied with the airline service is higer than female passengers. On the other hand, number of female passengers dissatisfied with airline service is higher than male passengers.

4. Satisfaction by customer type

```
firstTimeSatisfiedCust <- nrow(airlineData %>% filter(Customer.Type == "First-time"
& Satisfaction == "Satisfied"))
firstTimeDissatisfiedCust <- nrow(airlineData %>% filter(Customer.Type ==
"First-time" & Satisfaction == "Neutral or Dissatisfied"))
returningSatisfiedCust <- nrow(airlineData %>% filter(Customer.Type == "Returning"
& Satisfaction == "Satisfied"))
returningDissatisfiedCust <- nrow(airlineData %>% filter(Customer.Type ==
"Returning" & Satisfaction == "Neutral or Dissatisfied"))

custType <- c('First-time', 'First-time', 'Returning', 'Returning')
cnt <- c(firstTimeSatisfiedCust, firstTimeDissatisfiedCust, returningSatisfiedCust,
returningDissatisfiedCust)

ggplot(as.data.frame(cbind(custType, response, cnt)), aes(x = response, y = cnt,
fill = custType)) +
  geom_bar(stat = 'identity', position = 'dodge') +
  ggtitle("Customer Type Vs Satisfaction") +
  labs(fill = "Customer Type") +
  xlab("Customer Satisfaction") +
  ylab("No. of Customer")
```



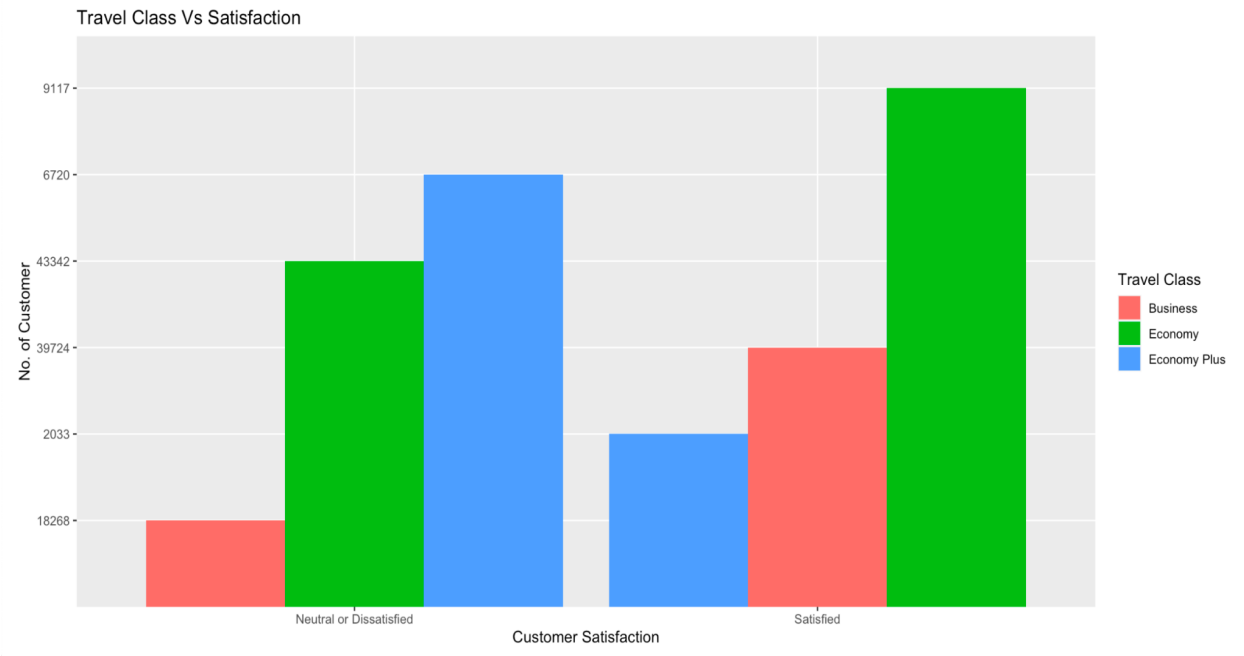
Majority of the first-time customers finds airline service satisfactory. On the contrary, majority of returning customers finds airline service not satisfactory or have neutral opinions.

5. Satisfaction by travel class

```
businessClassSatisfiedCust <- nrow(airlineData %>% filter(Class == "Business" &
Satisfaction == "Satisfied"))
businessClassDissatisfiedCust <- nrow(airlineData %>% filter(Class == "Business" &
Satisfaction == "Neutral or Dissatisfied"))
economyPlusClassSatisfiedCust <- nrow(airlineData %>% filter(Class == "Economy
Plus" & Satisfaction == "Satisfied"))
economyPlusClassDissatisfiedCust <- nrow(airlineData %>% filter(Class == "Economy
Plus" & Satisfaction == "Neutral or Dissatisfied"))
economyClassSatisfiedCust <- nrow(airlineData %>% filter(Class == "Economy" &
Satisfaction == "Satisfied"))
economyClassDissatisfiedCust <- nrow(airlineData %>% filter(Class == "Economy" &
Satisfaction == "Neutral or Dissatisfied"))

businessClass <- c("Business", "Business", "Economy Plus", "Economy Plus",
"Economy", "Economy")
response <- c("Satisfied", "Neutral or Dissatisfied", "Satisfied", "Neutral or
Dissatisfied", "Satisfied", "Neutral or Dissatisfied")
cnt <- c(businessClassSatisfiedCust, businessClassDissatisfiedCust,
economyPlusClassSatisfiedCust, economyPlusClassDissatisfiedCust,
economyClassSatisfiedCust, economyClassDissatisfiedCust)
```

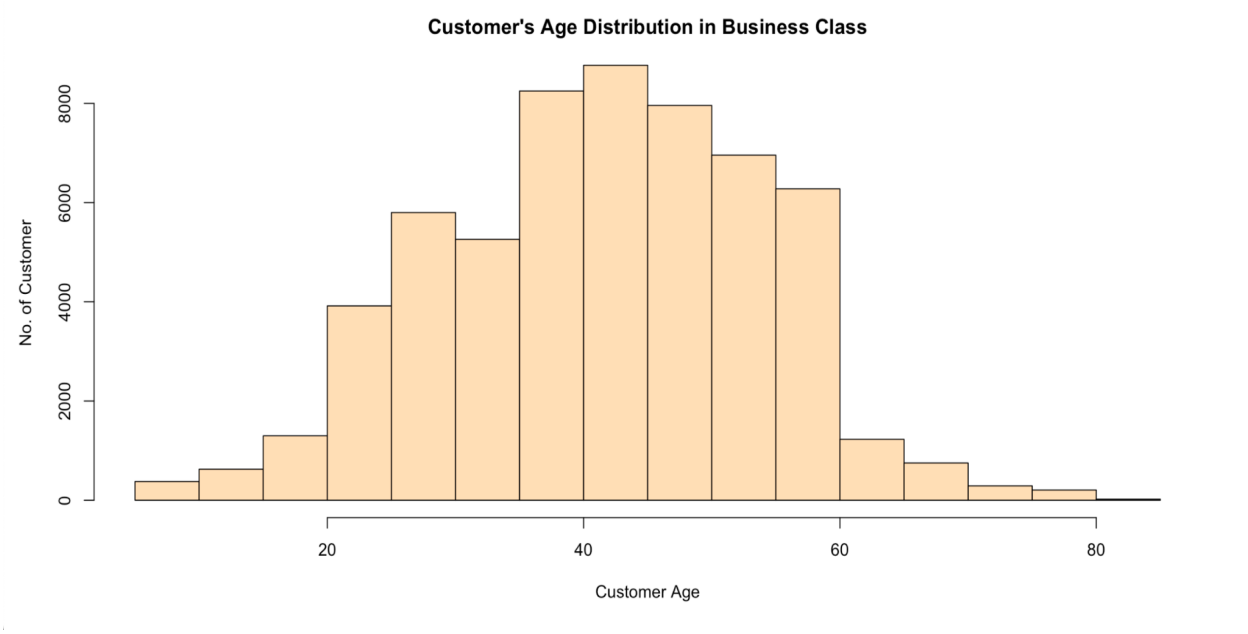
```
ggplot(as.data.frame(cbind(businessClass, response, cnt)), aes(x = response, y =
cnt, fill = businessClass)) +
  geom_bar(stat = 'identity', position = 'dodge') +
  ggtitle("Travel Class Vs Satisfaction") +
  labs(fill = "Travel Class") +
  xlab("Customer Satisfaction") +
  ylab("No. of Customer")
```



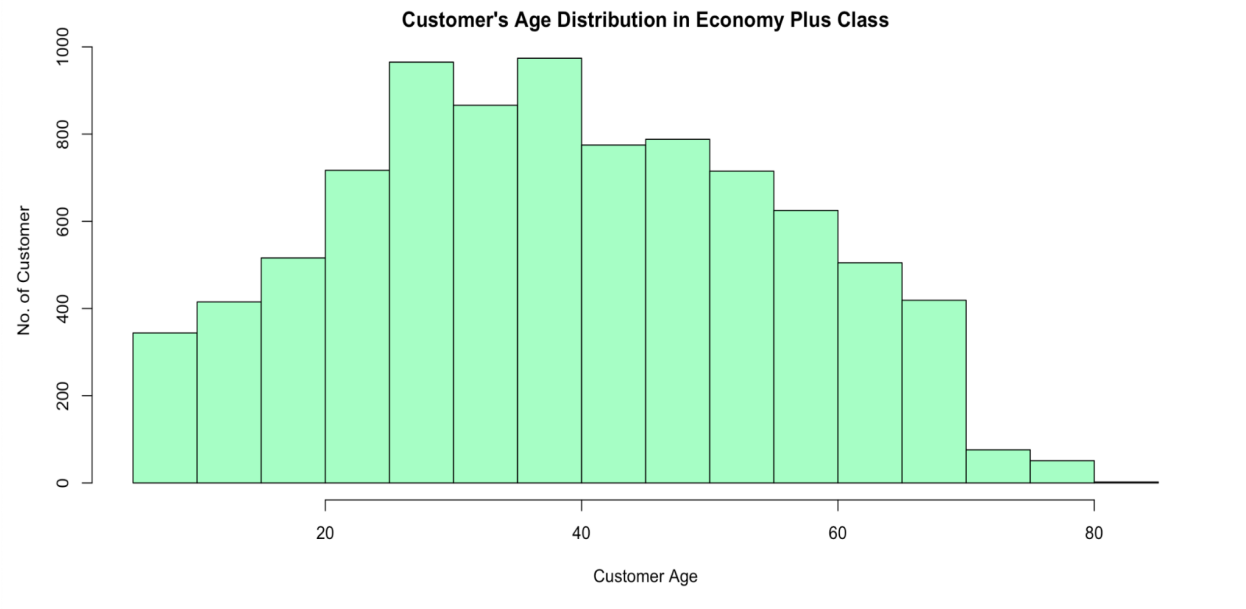
Based on the above graph, passengers traveling with economy class tend to provide satisfactory feedback, and passengers traveling with economy plus class tend to provide dissatisfactory feedback or have neutral opinions about airline service.

6. Age distribution by travel class

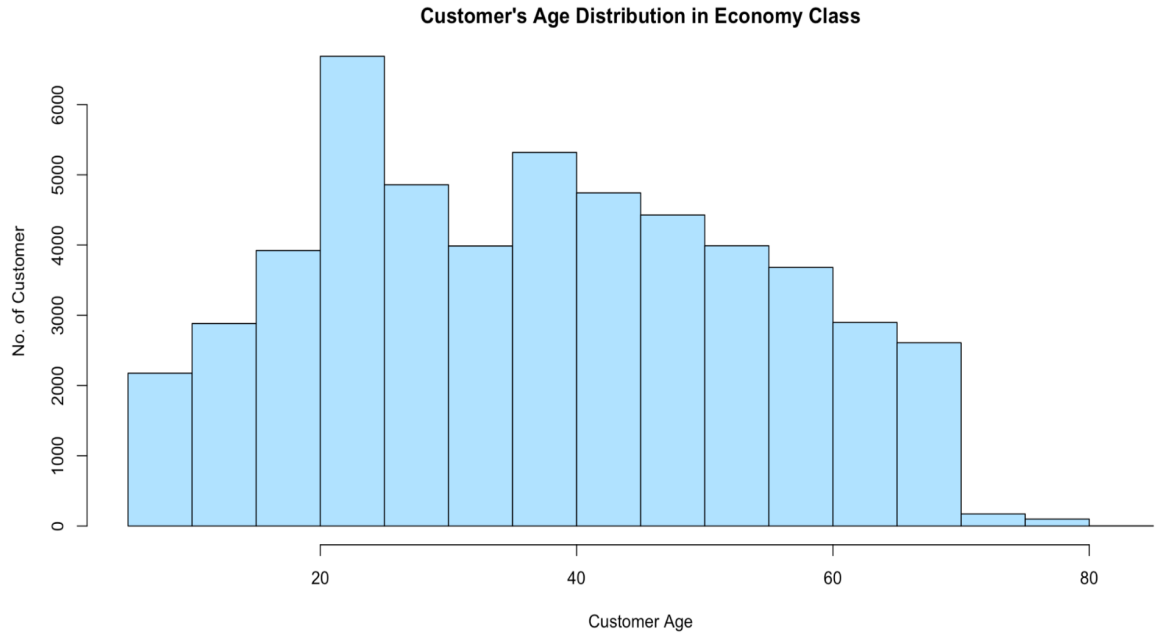
```
businessClassCust <- airlineData %>% filter(Class == "Business")
hist(businessClassCust$Age, col = '#ffdfba', main = "Customer's Age Distribution in
Business Class", xlab = "Customer Age", ylab = "No. of Customer")
```



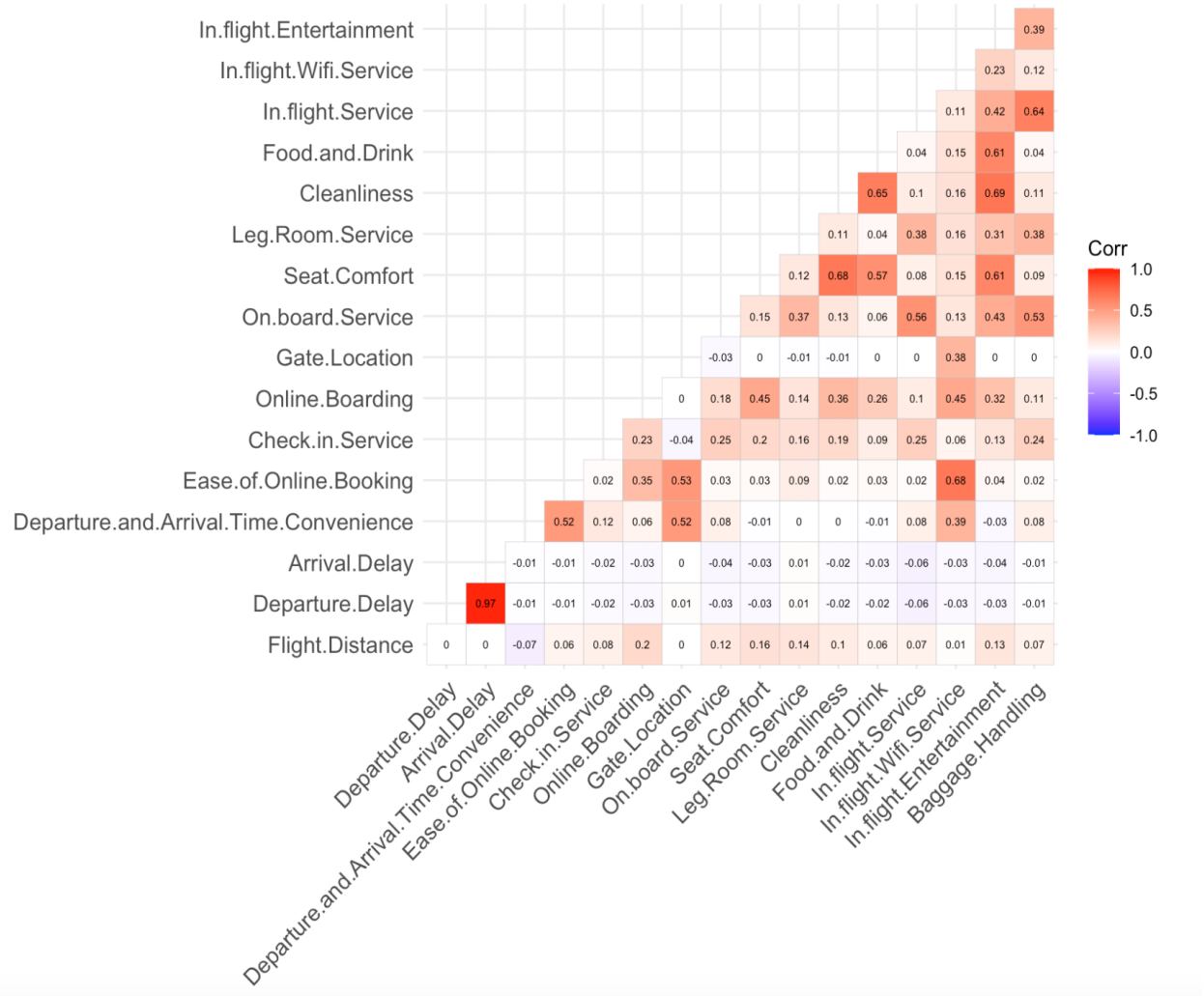

```
economyPlusClassCust <- airlineData %>% filter(Class == "Economy Plus")
hist(economyPlusClassCust$Age, col = '#baffc9', main = "Customer's Age Distribution
in Economy Plus Class", xlab = "Customer Age", ylab = "No. of Customer")
```




```
economyClassCust <- airlineData %>% filter(Class == "Economy")
hist(economyClassCust$Age, col = '#bae1ff', main = "Customer's Age Distribution in
Economy Class", xlab = "Customer Age", ylab = "No. of Customer")
```


Customer's Age Distribution in Economy Class
No. of Customer
0
1000
2000
3000
4000
5000
6000
20
40
60
80
Customer Age

7. Heatmap for co-relation matrix



From the co-relation matrix, almost all the numeric predictors are positively co-related to each other.

DATA PARTITION

We are using 80-20% split for training and testing dataset.

```
set.seed(1234)
index <- sample(1:nrow(airlineData), 0.20*nrow(airlineData))
test.df <- airlineData[index, ]
train.df <- airlineData[-index, ]
```

```
cat("Number of rows:\n1. Training dataset: ", nrow(train.df), "\n2. Testing
dataset: ", nrow(test.df))
```

```
Number of rows:
1. Training dataset:  95364
2. Testing dataset:  23840
```

MODEL TRAINING

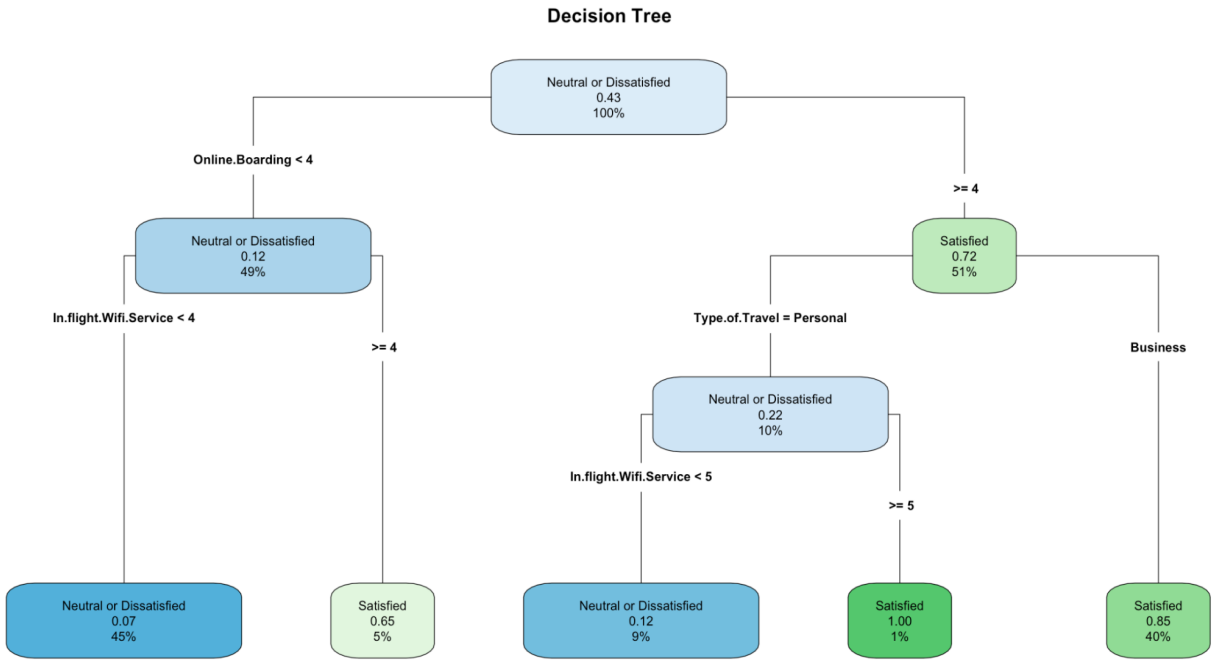
For training our models, we are using all available predictors except “ID” since it is unique for each row and does not provide any predictive power to models.

DECISION TREE

Data is represented by decision trees as a tree with hierarchical branches. It is a flowchart-like structure in which each internal node represents a test on an attribute (for example, whether a coin flip comes up heads or tails) [9], each branch represents the test result, and each leaf node represents a class label (decision taken after computing all attributes). Classification rules are represented by the pathways from the root to the leaf. Decision Trees are capable of performing both regression and classification problems.

```
# Generate decision tree
decisionTree <- rpart(Satisfaction ~ Online.Boarding + In.flight.Wifi.Service +
Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel + Class +
In.flight.Entertainment + Age + Leg.Room.Service + Cleanliness, data = train.df,
method = "class")
```

```
# Plot decision tree
rpart.plot(decisionTree, type = 4, main = "Decision Tree")
```




```
# Print model summary
summary(decisionTree)
```

```
Call:
rpart(formula = Satisfaction ~ Online.Boarding + In.flight.Wifi.Service +
    Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel +
    Class + In.flight.Entertainment + Age + Leg.Room.Service +
    Cleanliness, data = train.df, method = "class")
  n= 95364

          CP nsplit rel error    xerror        xstd
1 0.52110561      0 1.0000000 1.0000000 0.003758567
2 0.13901400      1 0.4788944 0.4788944 0.003063025
3 0.03357208      2 0.3398804 0.3398804 0.002674718
4 0.02685274      3 0.3063083 0.3063083 0.002560331
5 0.01000000      4 0.2794556 0.2794556 0.002461567

Variable importance
        Online.Boarding  In.flight.Wifi.Service           Seat.Comfort
Class          Type.of.Travel  Ease.of.Online.Booking
                    28                      18                      13
11                     10                      10
In.flight.Entertainment                     Age
```

```
                    9                     1

Node number 1: 95364 observations,    complexity param=0.5211056
  predicted class=Neutral or Dissatisfied  expected loss=0.4260413  P(node) =1
    class counts: 54735 40629
   probabilities: 0.574 0.426
  left son=2 (47024 obs) right son=3 (48340 obs)
  Primary splits:
      Online.Boarding         < 3.5  to the left,  improve=16826.210, (0 missing)
      Class                   splits as  RLL,      improve=12136.160, (0 missing)
      Type.of.Travel          splits as  RL,       improve=10060.030, (0 missing)
      In.flight.Entertainment < 3.5  to the left,  improve= 9613.947, (0 missing)
      In.flight.Wifi.Service  < 3.5  to the left,  improve= 8687.950, (0 missing)
  Surrogate splits:
      Seat.Comfort            < 3.5  to the left,  agree=0.737, adj=0.466, (0
split)
      In.flight.Wifi.Service  < 3.5  to the left,  agree=0.709, adj=0.410, (0
split)
      In.flight.Entertainment < 3.5  to the left,  agree=0.668, adj=0.327, (0
split)
      Class                   splits as  RLL,      agree=0.667, adj=0.325, (0
split)
      Ease.of.Online.Booking  < 3.5  to the left,  agree=0.667, adj=0.324, (0
split)

Node number 2: 47024 observations,    complexity param=0.03357208
  predicted class=Neutral or Dissatisfied  expected loss=0.1248937  P(node)
=0.4931001
    class counts: 41151  5873
   probabilities: 0.875 0.125
  left son=4 (42552 obs) right son=5 (4472 obs)
  Primary splits:
      In.flight.Wifi.Service < 3.5  to the left,  improve=2751.4320, (0 missing)
      Class                 splits as  RLL,      improve=1061.7760, (0 missing)
      Type.of.Travel        splits as  RL,       improve= 784.5242, (0 missing)
      Ease.of.Online.Booking < 3.5  to the left,  improve= 729.1169, (0 missing)
      Leg.Room.Service      < 3.5  to the left,  improve= 713.4416, (0 missing)

Node number 3: 48340 observations,    complexity param=0.139014
  predicted class=Satisfied               expected loss=0.2810095  P(node)
=0.5068999
    class counts: 13584 34756
   probabilities: 0.281 0.719
  left son=6 (9910 obs) right son=7 (38430 obs)
  Primary splits:
      Type.of.Travel          splits as  RL,       improve=6331.749, (0 missing)
      Class                   splits as  RLL,      improve=4466.577, (0 missing)
```

```
      In.flight.Entertainment < 3.5  to the left,  improve=3378.791, (0 missing)
      Leg.Room.Service        < 3.5  to the left,  improve=3126.732, (0 missing)
      In.flight.Wifi.Service  < 4.5  to the left,  improve=1724.592, (0 missing)
  Surrogate splits:
      Class                   splits as  RLR,      agree=0.827, adj=0.156, (0
split)
      Age                     < 60.5 to the right, agree=0.812, adj=0.084, (0
split)
      In.flight.Entertainment < 1.5  to the left,  agree=0.801, adj=0.029, (0
split)

Node number 4: 42552 observations
  predicted class=Neutral or Dissatisfied  expected loss=0.06944444  P(node)
=0.4462061
    class counts: 39597  2955
   probabilities: 0.931 0.069

Node number 5: 4472 observations
  predicted class=Satisfied                expected loss=0.3474955  P(node)
=0.04689401
    class counts:  1554  2918
   probabilities: 0.347 0.653

Node number 6: 9910 observations,    complexity param=0.02685274
  predicted class=Neutral or Dissatisfied  expected loss=0.2150353  P(node)
=0.1039176
    class counts:  7779  2131
   probabilities: 0.785 0.215
  left son=12 (8819 obs) right son=13 (1091 obs)
  Primary splits:
      In.flight.Wifi.Service < 4.5  to the left,  improve=1510.80800, (0 missing)
      Ease.of.Online.Booking < 4.5  to the left,  improve= 947.57530, (0 missing)
      Age                    < 41.5 to the right, improve= 193.14710, (0 missing)
      Leg.Room.Service       < 3.5  to the left,  improve= 126.42480, (0 missing)
      Online.Boarding        < 4.5  to the left,  improve=  96.44088, (0 missing)
  Surrogate splits:
      Ease.of.Online.Booking < 4.5  to the left,  agree=0.947, adj=0.52, (0 split)

Node number 7: 38430 observations
  predicted class=Satisfied                expected loss=0.1510539  P(node)
=0.4029823
    class counts:  5805 32625
   probabilities: 0.151 0.849

Node number 12: 8819 observations
  predicted class=Neutral or Dissatisfied  expected loss=0.1179272  P(node)
=0.09247725
```

```
    class counts:  7779  1040
   probabilities: 0.882 0.118

Node number 13: 1091 observations
  predicted class=Satisfied               expected loss=0  P(node) =0.01144038
    class counts:     0  1091
   probabilities: 0.000 1.000
```

```
# Print variables of importance
print(as.data.frame(decisionTree$variable.importance))
```

Description: df [8 × 1]

	decisionTree$variable.importance <dbl>
Online.Boarding	16826.206
In.flight.Wifi.Service	11159.968
Seat.Comfort	7840.588
Class	6462.169
Type.of.Travel	6331.749
Ease.of.Online.Booking	6238.379
In.flight.Entertainment	5688.867
Age	530.308

8 rows

Out of all predictors, online boarding is the most significant predictor and hence the first split in our decision tree is based on online boarding. On the other hand, age is the least significant predictor of all.

```
# Test decision tree on test data set
predictions <- predict(decisionTree, test.df, type = "class")

# Generate confusion matrix
decisionTreeConfMatrix <- confusionMatrix(predictions, test.df$Satisfaction)
decisionTreeConfMatrix
```

```
Confusion Matrix and Statistics

                          Reference
Prediction                 Neutral or Dissatisfied Satisfied
  Neutral or Dissatisfied                   11769      1042
  Satisfied                                  1826      9203

               Accuracy : 0.8797
                 95% CI : (0.8755, 0.8838)
    No Information Rate : 0.5703
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7568

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.8657
            Specificity : 0.8983
         Pos Pred Value : 0.9187
         Neg Pred Value : 0.8344
             Prevalence : 0.5703
         Detection Rate : 0.4937
   Detection Prevalence : 0.5374
      Balanced Accuracy : 0.8820

       'Positive' Class : Neutral or Dissatisfied
```

RANDOM FOREST

Random Forest, as the name implies, is made up of a vast number of decision trees that work together to classify data. An ensemble is a group of several predictive models that decide on the anticipated output collectively. Each individual tree in a random forest produces a class as an output. The class with the most votes is chosen as the model's final output.

The premise behind random forests is that a large number of uncorrelated decision trees functioning separately will outperform any one tree.

```
randomForestModel <- randomForest(Satisfaction ~ Online.Boarding +
In.flight.Wifi.Service + Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel +
Class + In.flight.Entertainment + Age + Leg.Room.Service + Cleanliness, data =
train.df)

# Print model
print(randomForestModel)
```

```
Call:
 randomForest(formula = Satisfaction ~ Online.Boarding + In.flight.Wifi.Service +
Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel +      Class +
In.flight.Entertainment + Age + Leg.Room.Service +      Cleanliness, data =
train.df)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 3

        OOB estimate of  error rate: 5.3%
Confusion matrix:
                       Neutral or Dissatisfied Satisfied class.error
Neutral or Dissatisfied                   52828      1907  0.03484060
Satisfied                                  3145     37484  0.07740776
```

```
# Print variables of importance
print(as.data.frame(randomForestModel$importance))
```

Description: df [10 × 1]

	MeanDecreaseGini <dbl>
Online.Boarding	10265.293
In.flight.Wifi.Service	6376.310
Seat.Comfort	2442.769
Ease.of.Online.Booking	1490.928
Type.of.Travel	4814.601
Class	5901.164
In.flight.Entertainment	4013.194
Age	2299.436
Leg.Room.Service	2457.694
Cleanliness	1523.537

1–10 of 10 rows

```
# Test random forest model on test data set
randomForestPredictions <- predict(randomForestModel, test.df, type = "class")

# Generate confusion matrix
randomForestConfMatrix <- confusionMatrix(randomForestPredictions,
test.df$Satisfaction)
randomForestConfMatrix
```

```
Confusion Matrix and Statistics

                          Reference
Prediction                 Neutral or Dissatisfied Satisfied
  Neutral or Dissatisfied                   13107       804
  Satisfied                                   488      9441

               Accuracy : 0.9458
                 95% CI : (0.9429, 0.9486)
    No Information Rate : 0.5703
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.889

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9641
            Specificity : 0.9215
         Pos Pred Value : 0.9422
         Neg Pred Value : 0.9509
             Prevalence : 0.5703
         Detection Rate : 0.5498
   Detection Prevalence : 0.5835
      Balanced Accuracy : 0.9428

       'Positive' Class : Neutral or Dissatisfied
```

NAIVE BAYES

Naive Bayes classifiers are a type of simple probabilistic classifier that uses Bayes' theorem with strong (naive) independence assumptions between features.

```
naiveBayesModel <- naiveBayes(Satisfaction ~ Online.Boarding +
In.flight.Wifi.Service + Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel +
Class + In.flight.Entertainment + Age + Leg.Room.Service + Cleanliness, data =
train.df)
```

```
# Print model summary
print(naiveBayesModel)
```

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
Neutral or Dissatisfied              Satisfied
              0.5739587              0.4260413

Conditional probabilities:
                        Online.Boarding
Y                            [,1]      [,2]
  Neutral or Dissatisfied 2.711921 1.0936989
  Satisfied               4.162667 0.9562644

                        In.flight.Wifi.Service
Y                            [,1]      [,2]
  Neutral or Dissatisfied 2.415475 0.9578836
  Satisfied               3.356839 1.3911611

                        Seat.Comfort
Y                            [,1]     [,2]
  Neutral or Dissatisfied 3.034293 1.297258
  Satisfied               4.018927 1.100006

                        Ease.of.Online.Booking
Y                            [,1]     [,2]
  Neutral or Dissatisfied 2.627039 1.154382
  Satisfied               3.219843 1.398324

                        Type.of.Travel
Y                          Business   Personal
  Neutral or Dissatisfied 0.50485064 0.49514936
  Satisfied               0.93925521 0.06074479

                        Class
Y                          Business    Economy Economy Plus
  Neutral or Dissatisfied 0.26538778 0.63717914   0.09743309
  Satisfied               0.78094465 0.17935465   0.03970071

                        In.flight.Entertainment
```

```
Y                            [,1]     [,2]
  Neutral or Dissatisfied 2.882415 1.318468
  Satisfied               4.054247 1.002415

                         Age
Y                            [,1]     [,2]
  Neutral or Dissatisfied 37.84805 16.49876
  Satisfied               42.48286 12.33884

                         Leg.Room.Service
Y                            [,1]     [,2]
  Neutral or Dissatisfied 2.997314 1.285316
  Satisfied               3.899727 1.116100

                         Cleanliness
Y                            [,1]     [,2]
  Neutral or Dissatisfied 2.923395 1.321835
  Satisfied               3.791848 1.114458
```

```
# Test classifier on test data set
naiveBayesPredictions <- predict(naiveBayesModel, test.df, type = "class")

# Generate confusion matrix
naiveBayesConfMatrix <- confusionMatrix(naiveBayesPredictions,
test.df$Satisfaction)
naiveBayesConfMatrix
```

```
Confusion Matrix and Statistics

                         Reference
Prediction                Neutral or Dissatisfied Satisfied
  Neutral or Dissatisfied                   11785      1375
  Satisfied                                  1810      8870

               Accuracy : 0.8664
                 95% CI : (0.862, 0.8707)
    No Information Rate : 0.5703
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7288

 Mcnemar's Test P-Value : 1.47e-14

            Sensitivity : 0.8669
            Specificity : 0.8658
```

```
         Pos Pred Value : 0.8955
         Neg Pred Value : 0.8305
             Prevalence : 0.5703
         Detection Rate : 0.4943
   Detection Prevalence : 0.5520
      Balanced Accuracy : 0.8663

       'Positive' Class : Neutral or Dissatisfied
```

LOGISTIC REGRESSION

Logistic Regression is a classification method that is built on the same concept as linear regression. In linear regression, we take a linear combination of different variables plus an intercept term to predict the output. But in classification problems, the predicted variable is categorical. The simplest case of classification is when the predicted variable is binary, i.e., it has only two classes, e.g., yes/no, pass/fail, win/lose, male/female, etc. Logistic regression also takes the linear combination of different variables plus the intercept term, but afterward, it takes the result and passes it through a logistic function.

```
# "Satisfied" = 1 & "Neutral or Dissatisfied" = 0
train.df$satisfactionCode <- ifelse(train.df$Satisfaction == "Satisfied", 1, 0)
test.df$satisfactionCode <- ifelse(test.df$Satisfaction == "Satisfied", 1, 0)

logisticModel <- glm(satisfactionCode ~ Online.Boarding + In.flight.Wifi.Service +
Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel + Class +
In.flight.Entertainment + Age + Leg.Room.Service + Cleanliness, data = train.df,
family = "binomial")

# Print summary of logistic model
summary(logisticModel)
```

```
Call:
glm(formula = satisfactionCode ~ Online.Boarding + In.flight.Wifi.Service +
    Seat.Comfort + Ease.of.Online.Booking + Type.of.Travel +
    Class + In.flight.Entertainment + Age + Leg.Room.Service +
    Cleanliness, family = "binomial", data = train.df)

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-2.9494  -0.3902  -0.0870   0.3876   3.5765

Coefficients:
                          Estimate Std. Error  z value Pr(>|z|)
(Intercept)             -8.3393059  0.0750752 -111.079  < 2e-16 ***
Online.Boarding          1.0839300  0.0120441   89.997  < 2e-16 ***
In.flight.Wifi.Service   0.6217350  0.0125443   49.563  < 2e-16 ***
Seat.Comfort            -0.0302550  0.0121182   -2.497   0.0125 *
Ease.of.Online.Booking  -0.0836322  0.0108327   -7.720 1.16e-14 ***
Type.of.TravelPersonal  -2.0347197  0.0312029  -65.209  < 2e-16 ***
ClassEconomy            -1.5068005  0.0267072  -56.419  < 2e-16 ***
ClassEconomy Plus       -1.3769760  0.0456471  -30.166  < 2e-16 ***
In.flight.Entertainment  0.4814265  0.0121632   39.581  < 2e-16 ***
Age                      0.0113342  0.0007621   14.873  < 2e-16 ***
Leg.Room.Service         0.4501659  0.0093405   48.195  < 2e-16 ***
Cleanliness              0.0803937  0.0119664    6.718 1.84e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 130108  on 95363  degrees of freedom
Residual deviance:  57149  on 95352  degrees of freedom
AIC: 57173

Number of Fisher Scoring iterations: 6
```

From the p-values, we can see that all the predictors are statistically significant in determining response variable.

```
# Print coefficients of predictors
as.data.frame(logisticModel$coefficients)
```

Description: df [12 × 1]

	logisticModel$coefficients <dbl>
(Intercept)	-8.33930589
Online.Boarding	1.08393000
In.flight.Wifi.Service	0.62173501
Seat.Comfort	-0.03025500
Ease.of.Online.Booking	-0.08363216
Type.of.TravelPersonal	-2.03471971
ClassEconomy	-1.50680045
ClassEconomy Plus	-1.37697604
In.flight.Entertainment	0.48142648
Age	0.01133424
Leg.Room.Service	0.45016594
Cleanliness	0.08039372

12 rows

```
# Test classifier on test data set
logisticModelPredictions <- predict(logisticModel, test.df)
logisticModelPredictions <- ifelse(logisticModelPredictions > 0.5, 1, 0)

# Generate confusion matrix
logisticModelConfMatrix <- confusionMatrix(as.factor(logisticModelPredictions),
as.factor(test.df$satisfactionCode))
logisticModelConfMatrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 12609  2069
         1   986  8176

               Accuracy : 0.8719
                 95% CI : (0.8675, 0.8761)
    No Information Rate : 0.5703
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7351

 Mcnemar's Test P-Value : < 2.2e-16
```

```
          Sensitivity : 0.9275
          Specificity : 0.7980
       Pos Pred Value : 0.8590
       Neg Pred Value : 0.8924
           Prevalence : 0.5703
       Detection Rate : 0.5289
 Detection Prevalence : 0.6157
    Balanced Accuracy : 0.8628

     'Positive' Class : 0
```

MODEL EVALUATION

We have evaluated models based on below criterias:

1. **Accuracy:** It is the ratio of the total number of correct predictions to the total number of predictions. It indicates how well the model is able to capture patterns in the dataset.

2. **Balanced accuracy:** It is a metric developed over the standard accuracy metric to determine how well the model can predict individual class labels from the response variable. This metric is typically used for models built on the unbalanced dataset.

3. **Sensitivity:** It is a metric that indicates the model's ability to correctly identify positive labels. It is also called the true positive rate or TPR.

4. **Specificity:** Similar to sensitivity, it is a metric used to measure the model's ability to correctly identify negative labels. It is also called the true negative rate or TNR.

5. **Precision:** It is a metric that indicates the reproducibility of the results. A good model should have high precision and high accuracy.

6. **Error:** It is a measure of error and is calculated as (1 - accuracy).

Calculating performance metric:

```
models <- c("Logistic Model", "Decision Tree", "Random Forest", "Naive Bayes")

accuracy <- c(round(logisticModelConfMatrix$overall['Accuracy'], 3),
round(decisionTreeConfMatrix$overall['Accuracy'], 3),
round(randomForestConfMatrix$overall['Accuracy'], 3),
round(naiveBayesConfMatrix$overall['Accuracy'], 3))

balancedAccuracy <- c(round(logisticModelConfMatrix$byClass['Balanced Accuracy'],
3), round(decisionTreeConfMatrix$byClass['Balanced Accuracy'], 3),
round(randomForestConfMatrix$byClass['Balanced Accuracy'], 3),
round(naiveBayesConfMatrix$byClass['Balanced Accuracy'], 3))

specificity <- c(round(logisticModelConfMatrix$byClass['Specificity'], 3),
round(decisionTreeConfMatrix$byClass['Specificity'], 3),
round(randomForestConfMatrix$byClass['Specificity'], 3),
round(naiveBayesConfMatrix$byClass['Specificity'], 3))

sensitivity <- c(round(logisticModelConfMatrix$byClass['Sensitivity'], 3),
round(decisionTreeConfMatrix$byClass['Sensitivity'], 3),
round(randomForestConfMatrix$byClass['Sensitivity'], 3),
round(naiveBayesConfMatrix$byClass['Sensitivity'], 3))

precision <- c(round(logisticModelConfMatrix$byClass['Precision'], 3),
round(decisionTreeConfMatrix$byClass['Precision'], 3),
round(randomForestConfMatrix$byClass['Precision'], 3),
round(naiveBayesConfMatrix$byClass['Precision'], 3))

error <- c(1 - round(logisticModelConfMatrix$overall['Accuracy'], 3), 1 -
round(decisionTreeConfMatrix$overall['Accuracy'], 3), 1 -
round(randomForestConfMatrix$overall['Accuracy'], 3), 1 -
round(naiveBayesConfMatrix$overall['Accuracy'], 3))

performanceMatrix <- as.data.frame(cbind(models, accuracy, balancedAccuracy,
specificity, sensitivity, precision, error), row.names = FALSE)
colnames(performanceMatrix) <- c("Models", "Accuracy", "Balanced Accuracy",
"Specificity", "Sensitivity", "Precision", "Error")
performanceMatrix
```

Performance metric:

Description: df [4 × 7]

Models <chr>	Accuracy <chr>	Balanced Accuracy <chr>	Specificity <chr>	Sensitivity <chr>	Precision <chr>	Error <chr>
Logistic Model	0.872	0.863	0.798	0.927	0.859	0.128
Decision Tree	0.88	0.882	0.898	0.866	0.919	0.12
Random Forest	0.946	0.943	0.922	0.964	0.942	0.054
Naive Bayes	0.866	0.866	0.866	0.867	0.896	0.134

4 rows

From the performance metric, we could see random forest dominates every other model in all aspects. Hence, random forest is the best suitable model for our application with the given dataset.

CONCLUSION

In conclusion, we were able to identify anomalies in the underlying data set and counter them with strategically sound decisions. We were also able to produce some insightful visualization that depicts trends in the dataset. With visualization, we could also point out areas the airline operator is doing well and also the areas that need improvements to achieve a better customer experience. We also built several classification models to determine customer satisfaction. From the experimentation results, we conclude that random forest is the best suitable for our application.

In future work, we can create a data pipeline to utilize data from different sources. We can also consider including other classifiers such as KNN, Support vector machine, Artificial neural networks, etc., and perform comparative analysis like we did to identify the best classifier for this application.

DATA SOURCES

The data we utilized for this research came from Kaggle; the airline passenger satisfaction dataset is available at https://www.kaggle.com/code/mennatallahnasr/airline-passenger-satisfaction/data

SOURCE CODE

GitHub Link: https://github.com/pathak-vikas/CSP571_Data_Preparation_Analysis

REFERENCES

[1] Siahaan, Vivian, and Sianipar, Rismon Hasiholan. AIRLINE PASSENGER SATISFACTION Analysis and Prediction Using Machine Learning and Deep Learning with Python. N.p.: BALIGE PUBLISHING, 2022.

[2] Jiang, Xuchu, Ying Zhang, Ying Li, and Biao Zhang. “Forecast and Analysis of Aircraft Passenger Satisfaction Based on RF-RFE-LR Model.” Nature News. Nature Publishing Group, July 1, 2022. https://www.nature.com/articles/s41598-022-14566-3.

[3] B.S., Kevin Tegar, Lestari, Anggun, and Pratiwi, Sekar Widyastuti. “AN ANALYSIS OF AIRLINES CUSTOMER SATISFACTION BY IMPROVING CUSTOMER SERVICE PERFORMANCE” Atlantis Press. Conference on Global Research on Sustainable Transport (GROST 2017). https://www.atlantis-press.com/article/25889417.pdf.

[4] Mohd Suki, Norazah. “Passenger satisfaction with airline service quality in Malaysia: A structural equation modeling approach.” Research in Transportation Business & Management. 10. 10.1016/j.rtbm.2014.04.001. Accessed November 30, 2022. https://www.researchgate.net/publication/262051305_Passenger_satisfaction_with_airline_service_quality_in_Malaysia_A_structural_equation_modeling_approach.

[5] Khatib, Fahed Salim. “An Investigation of Airline Service Quality, Passenger Satisfaction and Loyalty: The Jordanian Airline.” White Rose eTheses Online. University of Sheffield, January 1, 1998. https://etheses.whiterose.ac.uk/3647/.

[6] Yadav, Mith Follow Banking at ICICI Bank, Mithilesh. “US Falcon Airline Passenger Satisfaction.” Us falcon airline passenger satisfaction. slideshare. Accessed November 29, 2022. https://www.slideshare.net/Mithileshyadav11/us-falcon-airline-passenger-satisfaction.

[7] Melo, Pedro Augusto Oliveira. “Is It Possible to Predict How Satisfied a Passenger Will Be?” Medium. Medium, September 13, 2021. https://medium.com/@pedromelonet22/is-it-possible-to-predict-how-satisfied-a-passenger-will-be-e204f2bfbb0f.

[8] “US Airline.” Data Science + Melanin. Accessed November 29, 2022. https://www.mmoorer.com/us-airline.

[9] Leo Breiman. Random Forests. 2001. https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf.