Uber Data: Ride Cancellation Probability.

Arthur Richardson

7/5/2023

Contents

Dedication:

This project is dedicated to my two daughters.

Love self first to understand how to love another.

Love, Dad.

Introduction:

In this project, we will analyze Uber Request Data from ANUPAM MAJHI's Kaggle(https://www.kaggle.com/datasets/anupammajhi/uber-request-data). Uber is a ridesharing company. Rideshare is a travel in a private vehicle driven by its owner, free or for a fee, especially as part of an arrangement made using a website or app. In this project, we will evaluate the trip data and create a machine-learning algorithm to predict future trip completions or cancellations.

DISCLAIMER: This data and its analysis are provided for informational purposes only. The information presented here is not endorsed, affiliated with, or sponsored by Uber or any related entities. The data used in this analysis is publicly available and has been collected from various sources. We make no representations or warranties of any kind, express or implied, about the data's completeness, accuracy, reliability, or suitability. Any reliance you place on the information provided is strictly at your own risk. We will not be liable for any loss or damage arising from using this data. The use of this data does not create a professional-client relationship. We recommend verifying the data with official sources before making decisions or conclusions.

Objective:

Uber is one of the top ride-sharing companies in the world. Uber has a 68% share of the US rideshare market. Uber is a global company. Its service is available in over 250 cities in more than 45 countries. Uber drivers completed 7.6 billion trips in 2022, surpassing its peak of 6.9 billion in 2019. In this project, we will explore Uber's request data.

Data Installation:

Upload following packages and libraries for data exploration.

```
library(tidyverse)
library(caret)
library(data.table)
library(RColorBrewer)
library(rmarkdown)
library(dslabs)
library(gtable)
library(ggplot2)
```

```r
library(hexbin)
library(gt)
library(dplyr)
library(ggpmisc)
library(gridExtra)
library(janitor)
library(lubridate)
library(highcharter)
library(viridisLite)
library(broom)
library(scales)
library(xfun)
library(htmltools)
library(mime)
library(ggfortify)
library(gtsummary)
library(tinytex)
library(vroom)
library(curl)
library(gtools)
library(hrbrthemes)
library(plotrix)
library(timeDate)
library(parsnip)
library(viridis)
library(latexpdf)
library(kableExtra)
library(showtext)
library(remotes)
library(extrafont)
```

Data Analysis:

Upload the data set. The file can be downloaded from https://www.kaggle.com/datasets/anupammajhi/uber-request-data

```r
xfun::pkg_load2(c("htmltools", "mime"))
xfun::embed_files('UBER_Request.csv')
```

```r
UBER_Data <- read.csv(
  'UBER_Request.csv')
```

Dimensions and Summary

```
## [1] 6745    6

##    Request.id   Pickup.point         Driver.id        Status
##  Min.   :   1   Length:6745        Min.   :  1.0   Length:6745
##  1st Qu.:1691   Class :character   1st Qu.: 75.0   Class :character
##  Median :3387   Mode  :character   Median :149.0   Mode  :character
##  Mean   :3385                      Mean   :149.5
##  3rd Qu.:5080                      3rd Qu.:224.0
##  Max.   :6766                      Max.   :300.0
##                                    NA's   :2650
##  Request.timestamp  Drop.timestamp
##  Length:6745        Length:6745
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
##
```

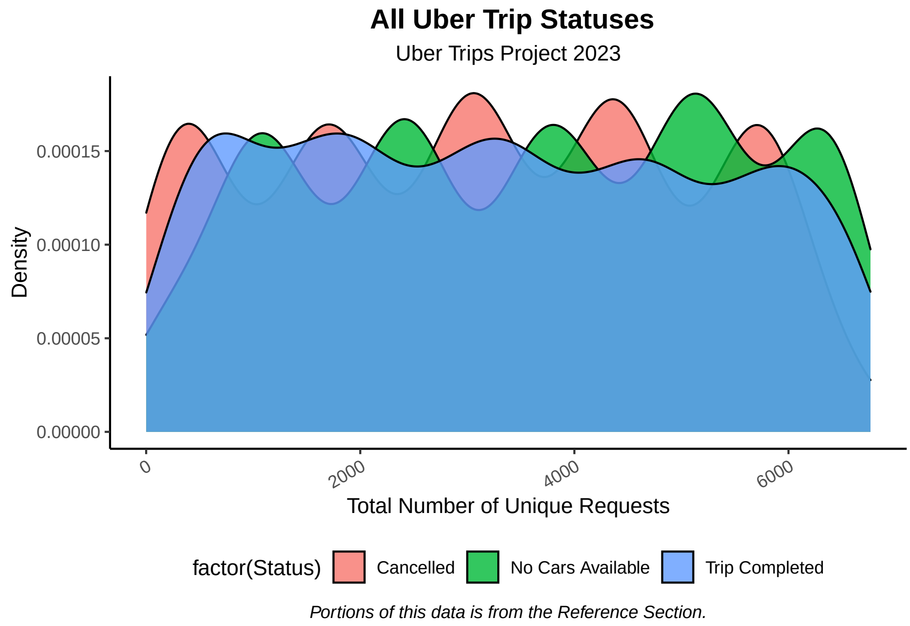
Based on this data we know that we have 300 unique driver ids, two main pick up points (Airport / City), and 6745 unique requests. We also have three Trip statuses: Cancelled— 1264, No Cars Available— 2650, Trip Completed— 2831.

We will explore the data. First, lets determine how many different trips were completed/not completed, what is the location of the completed trips, what dates does this data set cover.

Uber Data Set Glossary and Terminology

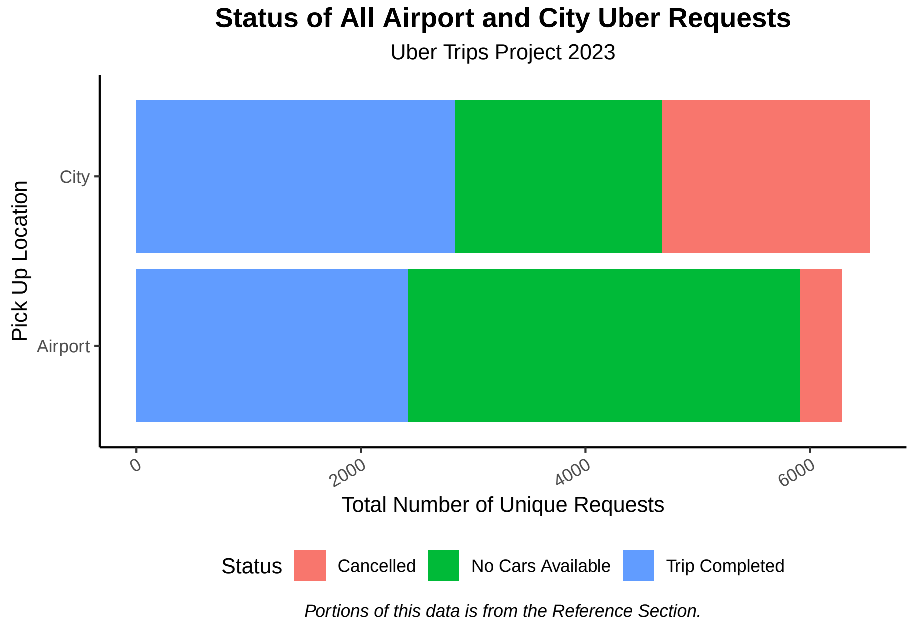
1. Request id: A unique identifier of the request.
2. Pickup point: The point from which the request was made.
3. Driver id: The unique identification number of the driver.
4. Status: The final status of the trip, that can be either completed, cancelled by the driver or no cars available.
5. Request timestamp: The date and time at which the customer made the trip request.
6. Drop timestamp: The drop-off date and time, in case the trip was completed.

Deep dive into the data


All Uber Trip Statuses

Uber Trips Project 2023

Portions of this data is from the Reference Section.


Status of All Airport and City Uber Requests

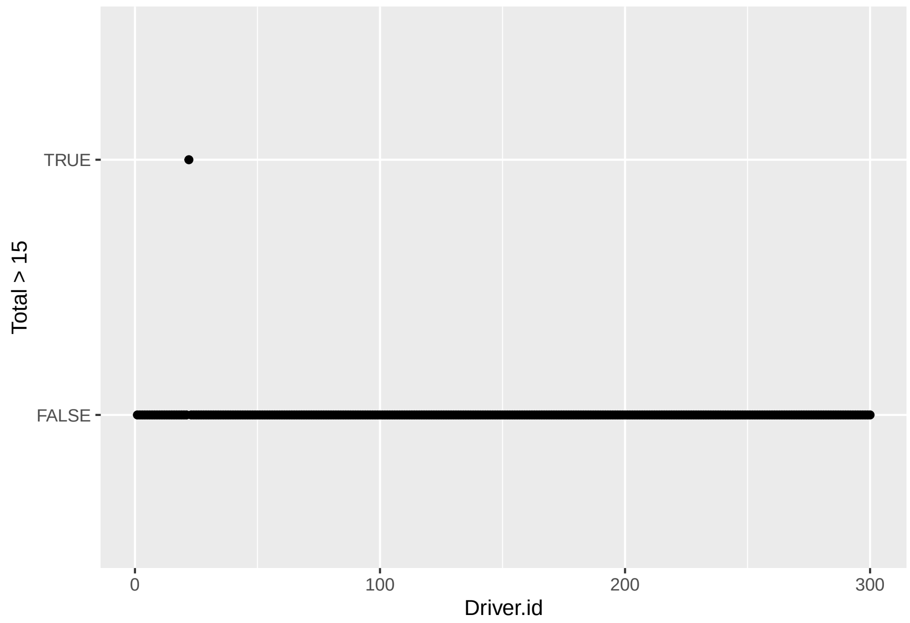
Uber Trips Project 2023

Portions of this data is from the Reference Section.

Which driver completed the most trips?

```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Trip_Completed Total
##       <int> <lgl>          <int>
## 1        22 TRUE              16
```

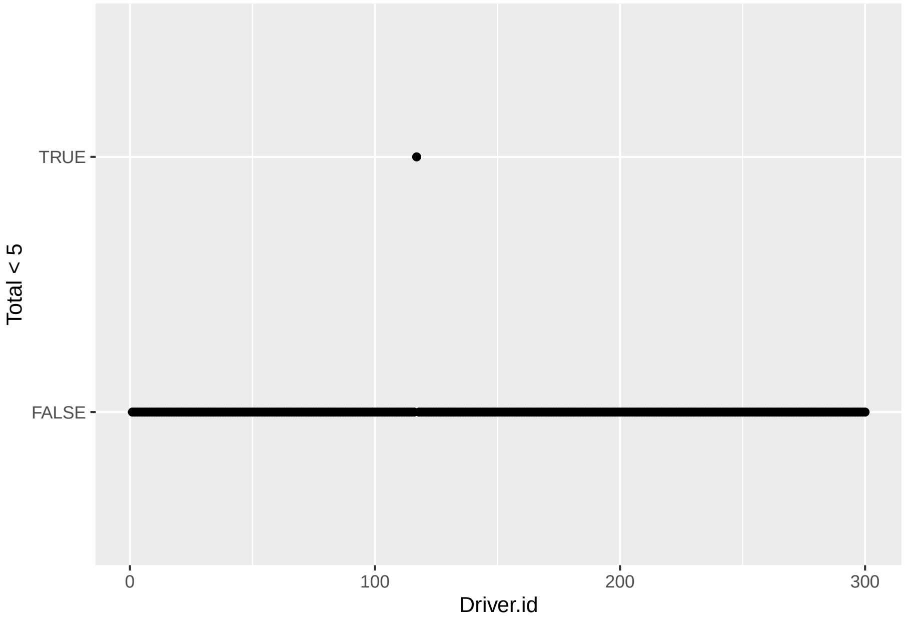
```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Trip_Completed Total
##       <int> <lgl>          <int>
## 1        22 TRUE              16
```



Who completed the least amount of trips?

```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Trip_Completed Total
##       <int> <lgl>          <int>
## 1       117 TRUE               4
```

```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Trip_Completed Total
##       <int> <lgl>          <int>
## 1       117 TRUE               4
```



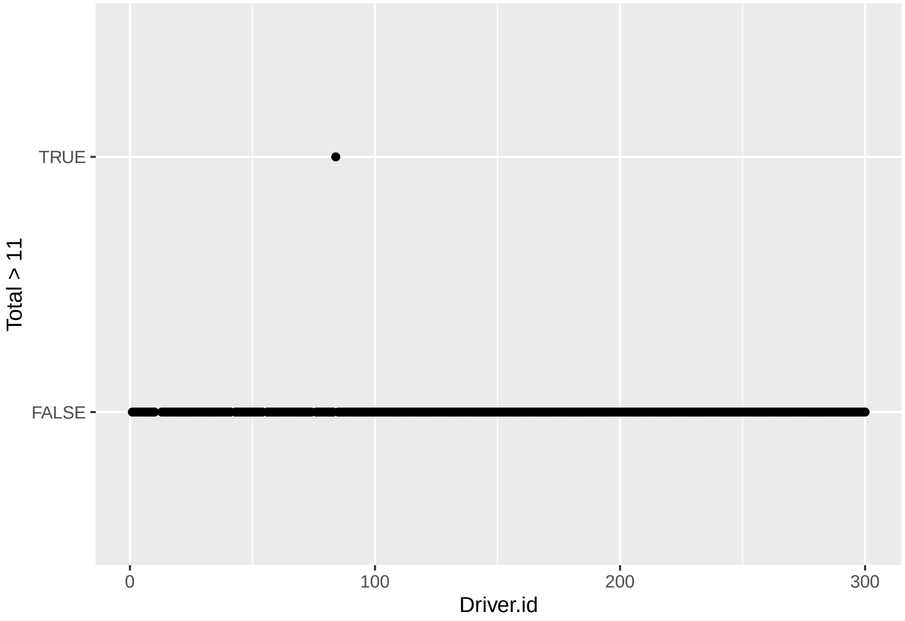
Average total of cancellations

```
## [1] 9.436667
```

Who had the most cancellations?

```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Cancelled Total
##       <int> <lgl>     <int>
## 1        84 TRUE         12
```

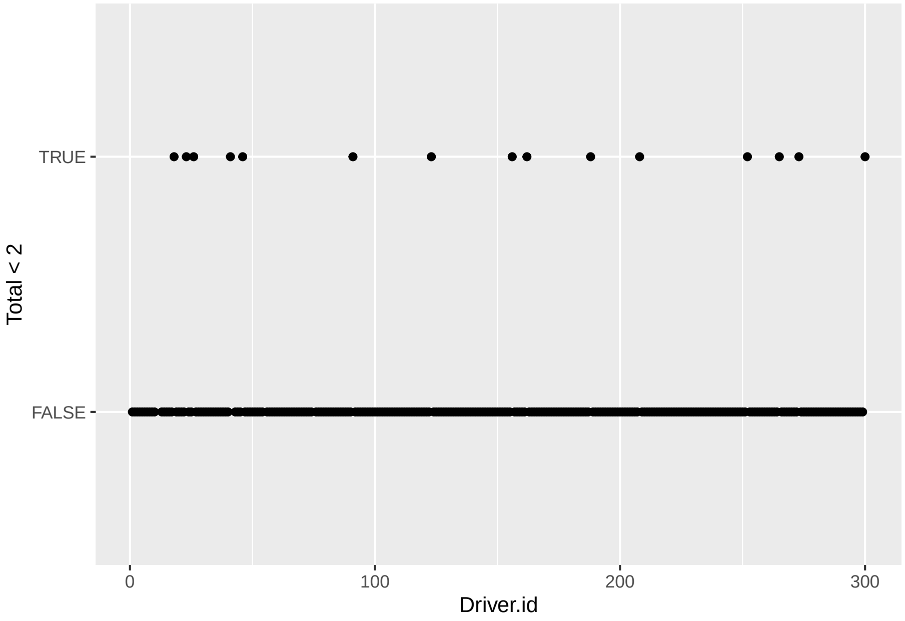
```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Cancelled Total
##       <int> <lgl>     <int>
## 1        84 TRUE         12
```



Who had the least amount of cancellations?

```
## # A tibble: 1 x 3
## # Groups:   Driver.id [1]
##   Driver.id Cancelled Total
##       <int> <lgl>     <int>
## 1        18 TRUE          1
```

```
## # A tibble: 15 x 3
## # Groups:   Driver.id [15]
##    Driver.id Cancelled Total
##        <int> <lgl>     <int>
##  1        18 TRUE          1
##  2        23 TRUE          1
##  3        26 TRUE          1
##  4        41 TRUE          1
##  5        46 TRUE          1
##  6        91 TRUE          1
##  7       123 TRUE          1
##  8       156 TRUE          1
##  9       162 TRUE          1
## 10       188 TRUE          1
## 11       208 TRUE          1
## 12       252 TRUE          1
## 13       265 TRUE          1
## 14       273 TRUE          1
## 15       300 TRUE          1
```



Average total of cancellations?

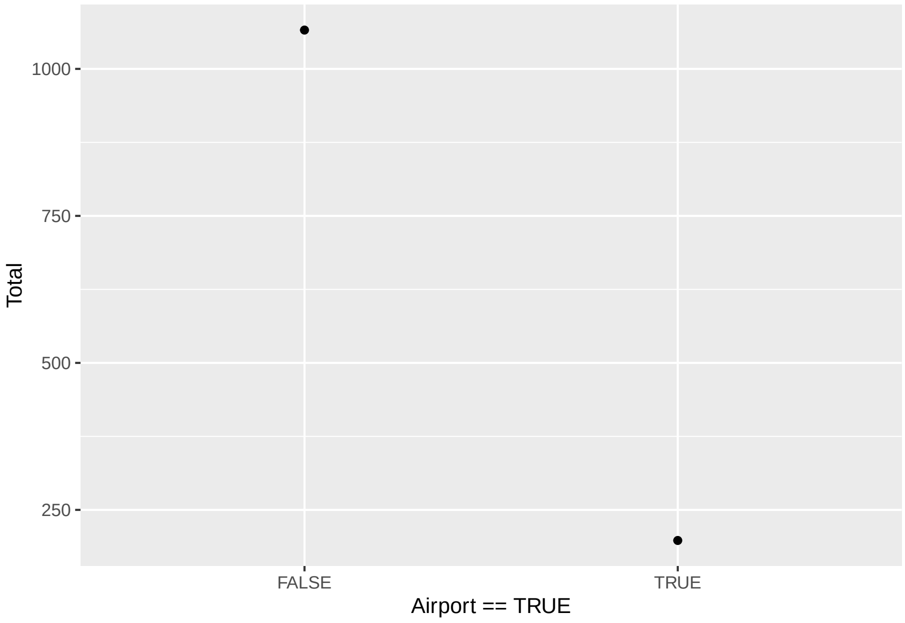
```
## [1] 4.284746
```

What is the ratio of completed trips vice cancelled trips?

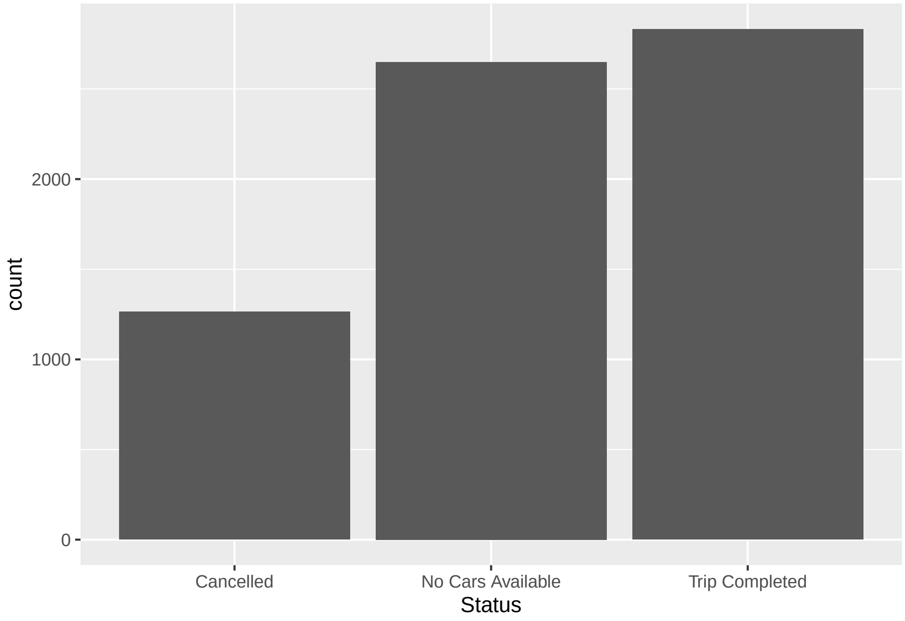
```
## [1] 2.202387
```

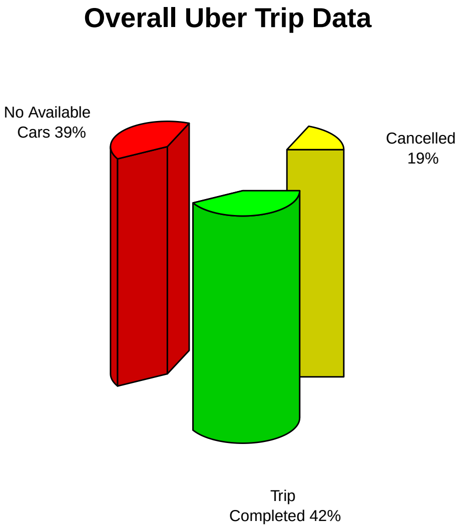
Which had the most cancellations, Airport or City?

```
## # A tibble: 1 x 4
## # Groups:   Airport, City [1]
##   Airport City  Status Total
##   <lgl>   <lgl> <lgl>  <int>
## 1 FALSE   TRUE  TRUE    1066
## # A tibble: 1 x 4
## # Groups:   Airport, City [1]
##   Airport City  Status Total
##   <lgl>   <lgl> <lgl>  <int>
## 1 TRUE    FALSE TRUE     198
```



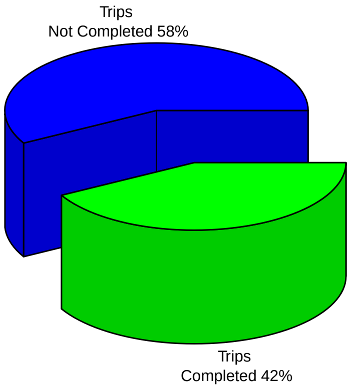
To get more from the data, lets break the specific trips down so we can determine more specific data





Trips Completed vs Trips Not Completed

Completed vs Not Competed Uber Trip Data



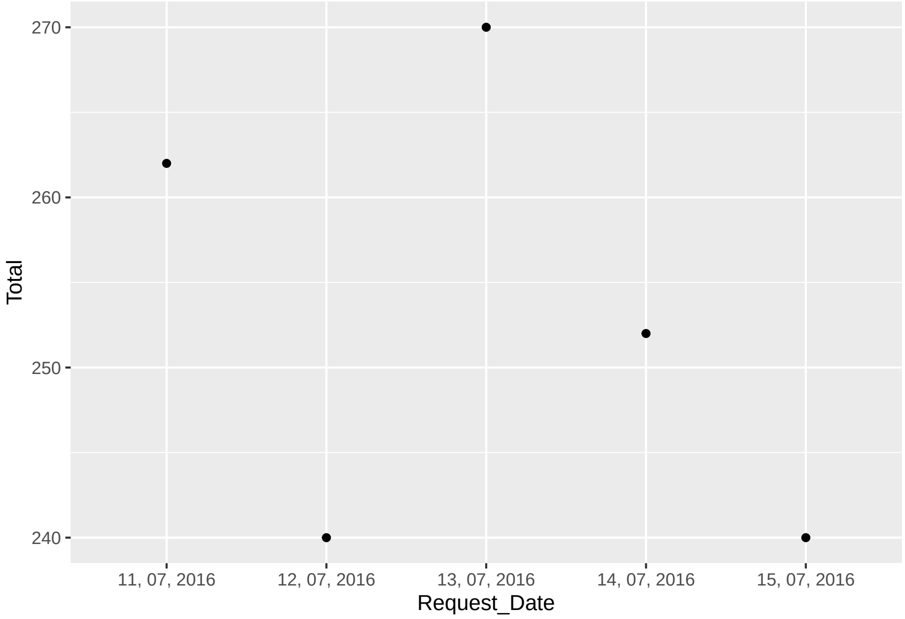
What time was the most cancellations?

```
## # A tibble: 1 x 3
## # Groups:   Request_Time [1]
##   Request_Time Status Total
##   <chr>        <lgl>  <int>
## 1 10:04        TRUE       9
```

```
## # A tibble: 1 x 3
## # Groups:   Request_Time [1]
##   Request_Time Status Total
##   <chr>        <lgl>  <int>
## 1 00:00        TRUE       1
```

What day were the most cancellations?

```
## # A tibble: 1 x 3
## # Groups:   Request_Date [1]
##   Request_Date Status Total
##   <chr>        <lgl>  <int>
## 1 13, 07, 2016 TRUE     270
## # A tibble: 1 x 3
## # Groups:   Request_Date [1]
##   Request_Date Status Total
##   <chr>        <lgl>  <int>
## 1 12, 07, 2016 TRUE     240
```



What time has the most trips?

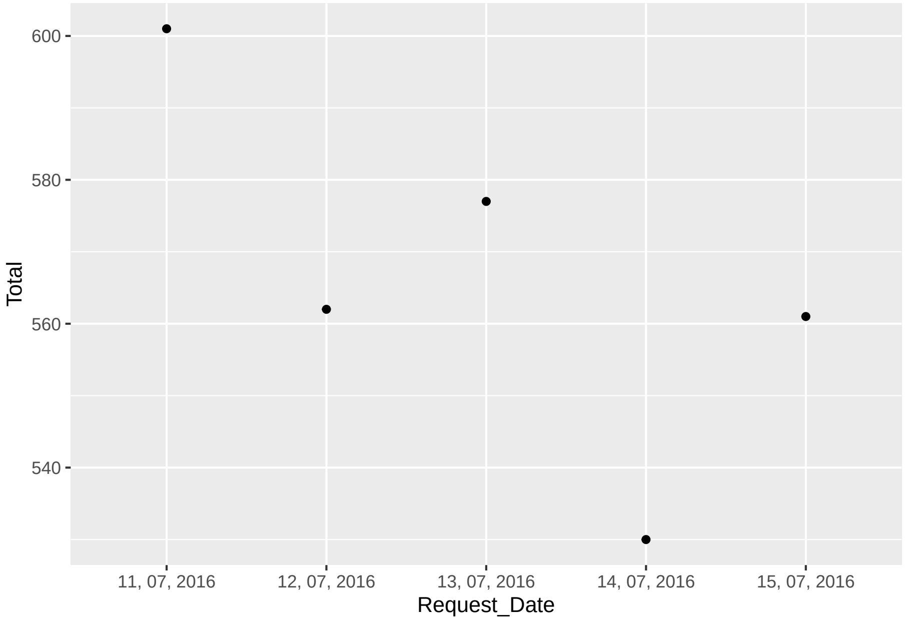
```
## # A tibble: 1 x 3
## # Groups:   Request_Time [1]
##   Request_Time Status Total
##   <chr>        <lgl>  <int>
## 1 04:54        TRUE       8
```

```
## # A tibble: 1 x 3
## # Groups:   Request_Time [1]
##   Request_Time Status Total
##   <chr>        <lgl>  <int>
## 1 00:00        TRUE       1
```

What day had the most trips?

```
## # A tibble: 1 x 3
## # Groups:   Request_Date [1]
##   Request_Date Status Total
##   <chr>        <lgl>  <int>
## 1 11, 07, 2016 TRUE     601
```

```
## # A tibble: 1 x 3
## # Groups:   Request_Date [1]
##   Request_Date Status Total
##   <chr>        <lgl>  <int>
## 1 14, 07, 2016 TRUE     530
```



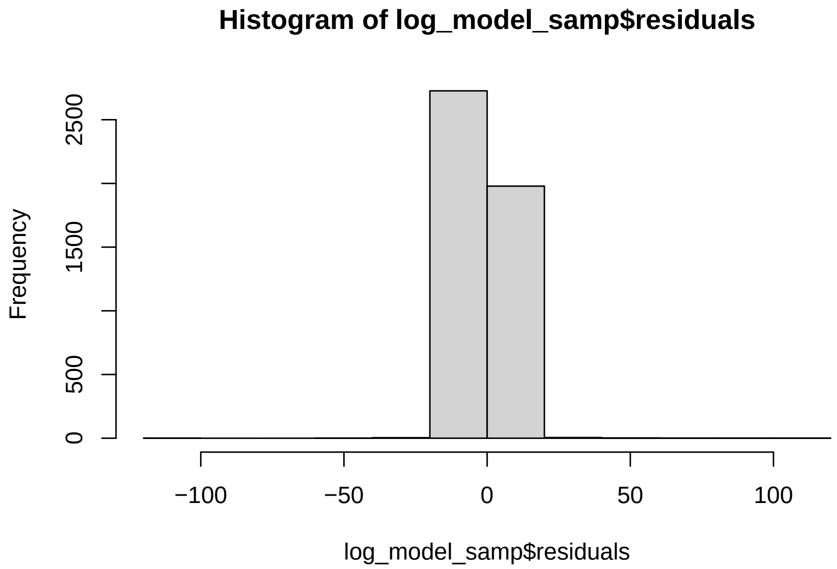
Machine Learning

Now lets create a Logistic regression machine learning algorithm that will predict if a trip will be completed or not.

```
y$Trips <- ifelse(y$Status == "Trip Completed", "Trip Completed", "Trip Not Completed")

y <- y %>%
  mutate(Trip_Not_Completed = ifelse(Status %in% c("No Cars Available", "Cancelled"), 1, 0
         Trips = ifelse(Status == "Trip Completed", 1, 0)) %>%
  select(-Status)
```

```
y = subset(y, select = -c(Trip_Not_Completed) )

any(is.na(y))
sum(is.na(y$Driver.id))
table(which(is.na(y), arr.ind=TRUE))

as.numeric(y$Request.id, y$Driver.id, y$Dropoff.Time, y$Request.time)

y$Request.time <- factor(y$Request.time, ordered = FALSE)
y$Dropoff.Time  <- factor(y$Dropoff.Time, ordered = FALSE)

set.seed(123)
train_index <- createDataPartition(y$Trips, p = 0.7, list = FALSE, times = 1)
train <- y[train_index, ]
test <- y[-train_index, ]

log_model_samp <- glm(Trips ~ Request.id + Pickup.point + Request.time + Driver.id, data =
                      family = binomial(link = "logit"))
```

View the model output and use the model to make predictions on the testing set.



```
accuracy <- mean((predictions <= 0.5) == (test$Trips == "0"))
```

Calculate the accuracy of the model

```
cat("Accuracy of the model:", accuracy)
```

```
## Accuracy of the model: 0.5743945
```

Average Trip notcompleted in the model

```
sum(log_model_samp$model$Trips == 0)/sum(log_model_samp$model$Trips == 0,
                                         log_model_samp$model$Trips == 1)
```

```
## [1] 0.5787802
```

Conclusion

The accuracy of the model is less than 60%. This is basically a coin flip. Without the distance or other critical data it is hard to train the algorithm.

Reference Section

1. Irizarry, R. A. (2022, July 7). Introduction to Data Science. HARVARD Data Science. Retrieved August 8, 2022, from Https://rafalab.github.io/dsbook/ This project utilized "Introduction to Data Science Data Analysis and Prediction Algorithms with R" by our course instructor Rafael A. Irizarry published 2022-07-07.

2. (2018, January 1). Uber Request Data. Kaggle. Retrieved October 1, 2022, from https://www.kaggle.com/datasets/anupammajhi/uber-request-data