CSP-571 Project Spring 24

Kyung Jin Kwak (kkwak4@hawk.iit.edu (mailto:kkwak4@hawk.iit.edu)), Srujan Ramesh (sramesh19@hawk.iit.edu (mailto:sramesh19@hawk.iit.edu))

Install necessary libraries

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(knitr)
```

Loading & Checking Data

```
taxi_df <- read.csv("taxi_Trips__2024.csv")
```

```
head(taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
2	000072ee076c9038868e239ca54185eb43959db0
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
4	00007572c5f92e2ff067e6f838a5ad74e83665d3
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25

6 rows | 1-2 of 24 columns

{r}tlsg print(names(taxi_df)) print(nrow(taxi_df))

```
colSums(is.na(taxi_df))
```

```
##                     Trip.ID                     Taxi.ID
##                           0                           1
##       Trip.Start.Timestamp         Trip.End.Timestamp
##                           0                           0
##               Trip.Seconds                 Trip.Miles
##                          87                           5
##        Pickup.Census.Tract       Dropoff.Census.Tract
##                      266886                      273775
##      Pickup.Community.Area     Dropoff.Community.Area
##                      11105                       42428
##                       Fare                        Tips
##                       1014                        1014
##                      Tolls                      Extras
##                       1014                        1014
##                 Trip.Total                Payment.Type
##                       1014                           0
##                    Company   Pickup.Centroid.Latitude
##                           0                       10989
##  Pickup.Centroid.Longitude   Pickup.Centroid.Location
##                      10989                           0
##  Dropoff.Centroid.Latitude Dropoff.Centroid.Longitude
##                      39982                       39982
## Dropoff.Centroid..Location
##                           0
```

```
(unique(taxi_df$Company))
```

```
##  [1] "Flash Cab"
##  [2] "Taxicab Insurance Agency Llc"
##  [3] "Globe Taxi"
##  [4] "5 Star Taxi"
##  [5] "City Service"
##  [6] "Chicago Independents"
##  [7] "Blue Ribbon Taxi Association"
##  [8] "Taxi Affiliation Services"
##  [9] "Chicago City Taxi Association"
## [10] "Choice Taxi Association"
## [11] "Medallion Leasin"
## [12] "Sun Taxi"
## [13] "U Taxicab"
## [14] "Taxicab Insurance Agency, LLC"
## [15] "Choice Taxi Association Inc"
## [16] "Chicago Taxicab"
## [17] "Patriot Taxi Dba Peace Taxi Associat"
## [18] "Setare Inc"
## [19] "Taxi Affiliation Services Llc - Yell"
## [20] "3556 - 36214 RC Andrews Cab"
## [21] "Top Cab"
## [22] "Koam Taxi Association"
## [23] "312 Medallion Management Corp"
## [24] "Star North Taxi Management Llc"
## [25] "6574 - Babylon Express Inc."
## [26] "5167 - 71969 5167 Taxi Inc"
## [27] "2733 - 74600 Benny Jona"
## [28] "3591 - 63480 Chuks Cab"
## [29] "Tac - Yellow Cab Association"
## [30] "Metro Jet Taxi A."
## [31] "4787 - 56058 Reny Cab Co"
## [32] "4623 - 27290 Jay Kim"
## [33] "4053 - 40193 Adwar H. Nikola"
## [34] "Petani Cab Corp"
## [35] "Tac - Checker Cab Dispatch"
```

Understanding the features

```
feature_desc <- read.csv("taxi_Trips_2024_Feature_descriptions.csv")
feature_desc
```

Column.Name <chr>
Trip ID
Taxi ID
Trip Start Timestamp
Trip End Timestamp

Column.Name <chr>
Trip Seconds
Trip Miles
Pickup Census Tract
Dropoff Census Tract
Pickup Community Area
Dropoff Community Area

1-10 of 23 rows | 1-1 of 3 columns Previous **1** 2 3 Next

```
notes <- list()
for (feature in names(taxi_df)) {

  curr_note <- paste("Valid rows:",(nrow(taxi_df) - sum(is.na(taxi_df[,feature]))),
                     "; N/A rows:", sum(is.na(taxi_df[,feature])),
                     "; Unique values:", length(unique(taxi_df[,feature])))
  notes <- append(notes, curr_note)
}
notes
```

```
## [[1]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 425229"
##
## [[2]]
## [1] "Valid rows: 425228 ; N/A rows: 1 ; Unique values: 2520"
##
## [[3]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 2977"
##
## [[4]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 2988"
##
## [[5]]
## [1] "Valid rows: 425142 ; N/A rows: 87 ; Unique values: 6077"
##
## [[6]]
## [1] "Valid rows: 425224 ; N/A rows: 5 ; Unique values: 4278"
##
## [[7]]
## [1] "Valid rows: 158343 ; N/A rows: 266886 ; Unique values: 221"
##
## [[8]]
## [1] "Valid rows: 151454 ; N/A rows: 273775 ; Unique values: 347"
##
## [[9]]
## [1] "Valid rows: 414124 ; N/A rows: 11105 ; Unique values: 78"
##
## [[10]]
## [1] "Valid rows: 382801 ; N/A rows: 42428 ; Unique values: 78"
##
## [[11]]
## [1] "Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 4642"
##
## [[12]]
## [1] "Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 2161"
##
## [[13]]
## [1] "Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 139"
##
## [[14]]
## [1] "Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 865"
##
## [[15]]
## [1] "Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 7816"
##
## [[16]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 7"
##
## [[17]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 35"
##
## [[18]]
```

```
## [1] "Valid rows: 414240 ; N/A rows: 10989 ; Unique values: 276"
##
## [[19]]
## [1] "Valid rows: 414240 ; N/A rows: 10989 ; Unique values: 276"
##
## [[20]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 277"
##
## [[21]]
## [1] "Valid rows: 385247 ; N/A rows: 39982 ; Unique values: 366"
##
## [[22]]
## [1] "Valid rows: 385247 ; N/A rows: 39982 ; Unique values: 365"
##
## [[23]]
## [1] "Valid rows: 425229 ; N/A rows: 0 ; Unique values: 366"
```

```
feature_desc$Notes <- unlist(notes)
feature_desc
```

Column.Name <chr>
Trip ID
Taxi ID
Trip Start Timestamp
Trip End Timestamp
Trip Seconds
Trip Miles
Pickup Census Tract
Dropoff Census Tract
Pickup Community Area
Dropoff Community Area

1-10 of 23 rows | 1-1 of 4 columns Previous **1** 2 3 Next

```
# Install the formattable package if not already installed
if (!require(formattable)) {
  install.packages("formattable")
}
```

```
## Loading required package: formattable
```

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

# Pretty print the table with color
formattable(feature_desc, align = c("l", "l", "l", "l"), list(Notes = formatter("span",
style = "color:blue"), Type = formatter("span", style = "color:green"))
)
```

Column.Name	Description	Type	Notes
Trip ID	A unique identifier for the trip.	Plain Text	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 425229
Taxi ID	A unique identifier for the taxi.	Plain Text	Valid rows: 425228 ; N/A rows: 1 ; Unique values: 2520
Trip Start Timestamp	When the trip started, rounded to the nearest 15 minutes.	Date & Time	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 2977
Trip End Timestamp	When the trip ended, rounded to the nearest 15 minutes.	Date & Time	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 2988
Trip Seconds	Time of the trip in seconds.	Number	Valid rows: 425142 ; N/A rows: 87 ; Unique values: 6077
Trip Miles	Distance of the trip in miles.	Number	Valid rows: 425224 ; N/A rows: 5 ; Unique values: 4278
Pickup Census Tract	The Census Tract where the trip began. For privacy, this Census Tract is not shown for some trips. This column often will be blank for locations outside Chicago.	Plain Text	Valid rows: 158343 ; N/A rows: 266886 ; Unique values: 221
Dropoff Census Tract	The Census Tract where the trip ended. For privacy, this Census Tract is not shown for some trips. This column often will be blank for locations outside Chicago.	Plain Text	Valid rows: 151454 ; N/A rows: 273775 ; Unique values: 347
Pickup Community Area	The Community Area where the trip began. This column will be blank for locations outside Chicago.	Number	Valid rows: 414124 ; N/A rows: 11105 ; Unique values: 78
Dropoff Community Area	The Community Area where the trip ended. This column will be blank for locations outside Chicago.	Number	Valid rows: 382801 ; N/A rows: 42428 ; Unique values: 78
Fare	The fare for the trip.	Number	Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 4642

Column.Name	Description	Type	Notes
Tips	The tip for the trip. Cash tips generally will not be recorded.	Number	Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 2161
Tolls	The tolls for the trip.	Number	Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 139
Extras	Extra charges for the trip.	Number	Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 865
Trip Total	Total cost of the trip, the total of the previous columns.	Number	Valid rows: 424215 ; N/A rows: 1014 ; Unique values: 7816
Payment Type	Type of payment for the trip.	Plain Text	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 7
Company	The taxi company.	Plain Text	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 35
Pickup Centroid Latitude	The latitude of the center of the pickup census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Number	Valid rows: 414240 ; N/A rows: 10989 ; Unique values: 276
Pickup Centroid Longitude	The longitude of the center of the pickup census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Number	Valid rows: 414240 ; N/A rows: 10989 ; Unique values: 276
Pickup Centroid Location	The location of the center of the pickup census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Point	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 277
Dropoff Centroid Latitude	The latitude of the center of the dropoff census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Number	Valid rows: 385247 ; N/A rows: 39982 ; Unique values: 366
Dropoff Centroid Longitude	The longitude of the center of the dropoff census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Number	Valid rows: 385247 ; N/A rows: 39982 ; Unique values: 365

Column.Name	Description	Type	Notes
Dropoff Centroid Location	The location of the center of the dropoff census tract or the community area if the census tract has been hidden for privacy. This column often will be blank for locations outside Chicago.	Point	Valid rows: 425229 ; N/A rows: 0 ; Unique values: 366

```
length(which(taxi_df$Fare + taxi_df$Tips + taxi_df$Tolls + taxi_df$Extras != taxi_df$Trip.Total))
```

```
## [1] 136222
```

Data Cleaning

Dropping unwanted features

```
features_to_drop <- c("Trip.End.Timestamp", "Pickup.Census.Tract", "Dropoff.Census.Tract", "Pickup.Centroid.Latitude", "Pickup.Centroid.Longitude", "Pickup.Centroid.Location", "Dropoff.Centroid.Latitude", "Dropoff.Centroid.Longitude", "Dropoff.Centroid..Location")

simplified_taxi_df <- subset(taxi_df, select = -c(Trip.End.Timestamp, Pickup.Census.Tract, Dropoff.Census.Tract, Pickup.Centroid.Latitude, Pickup.Centroid.Longitude, Pickup.Centroid.Location, Dropoff.Centroid.Latitude, Dropoff.Centroid.Longitude, Dropoff.Centroid..Location))
head(simplified_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
2	000072ee076c9038868e239ca54185eb43959db0
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
4	00007572c5f92e2ff067e6f838a5ad74e83665d3
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25

6 rows | 1-2 of 15 columns

Removing Null rows

```
dim(simplified_taxi_df)
```

```
## [1] 425229     14
```

```
colSums(is.na(simplified_taxi_df))
```

```
##                Trip.ID                Taxi.ID   Trip.Start.Timestamp
##                      0                       1                      0
##           Trip.Seconds             Trip.Miles  Pickup.Community.Area
##                     87                       5                  11105
## Dropoff.Community.Area                   Fare                   Tips
##                  42428                   1014                   1014
##                  Tolls                 Extras             Trip.Total
##                   1014                   1014                   1014
##           Payment.Type                Company
##                      0                       0
```

```
cleaned_taxi_df <- simplified_taxi_df[!apply(is.na(simplified_taxi_df), 1, any), ]
colSums(is.na(cleaned_taxi_df))
```

```
##                Trip.ID                Taxi.ID   Trip.Start.Timestamp
##                      0                       0                      0
##           Trip.Seconds             Trip.Miles  Pickup.Community.Area
##                      0                       0                      0
## Dropoff.Community.Area                   Fare                   Tips
##                      0                       0                      0
##                  Tolls                 Extras             Trip.Total
##                      0                       0                      0
##           Payment.Type                Company
##                      0                       0
```

```
dim(cleaned_taxi_df)
```

```
## [1] 379024     14
```

```
head(cleaned_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25
7	0000cf293ada965f89a98c8ccfae7b0ce3a03e41

	Trip.ID <chr>
8	0001235258d46a21317b6691ade9386c4d7e02c4

6 rows | 1-2 of 15 columns

Converting seconds to mins

```
cleaned_taxi_df$Trip.Minutes <- round(cleaned_taxi_df$Trip.Seconds / 60, digits = 2)
cleaned_taxi_df$Trip.Seconds <- NULL
head(cleaned_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25
7	0000cf293ada965f89a98c8ccfae7b0ce3a03e41
8	0001235258d46a21317b6691ade9386c4d7e02c4

6 rows | 1-2 of 15 columns

Converting timestamp to hour of the day & day of the week

```
cleaned_taxi_df$Trip.Start.Timestamp <- as.POSIXct(cleaned_taxi_df$Trip.Start.Timestamp,
                                                   format = "%m/%d/%y %H:%M")

cleaned_taxi_df$Trip.Start.Date <- as.Date(cleaned_taxi_df$Trip.Start.Timestamp)

cleaned_taxi_df$Trip.Hour.Of.The.Day <- as.integer(format(cleaned_taxi_df$Trip.Start.Tim
estamp, format = "%H"))

days_of_week <- c("Sunday" = 1, "Monday" = 2, "Tuesday" = 3, "Wednesday" = 4, "Thursday"
= 5, "Friday" = 6, "Saturday" = 7)
cleaned_taxi_df$Trip.Day.Of.The.Week <- as.integer(days_of_week[weekdays(cleaned_taxi_df
$Trip.Start.Timestamp)])
cleaned_taxi_df$Trip.Day.Of.The.Week <- as.factor(cleaned_taxi_df$Trip.Day.Of.The.Week)
head(cleaned_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25
7	0000cf293ada965f89a98c8ccfae7b0ce3a03e41
8	0001235258d46a21317b6691ade9386c4d7e02c4

6 rows | 1-2 of 18 columns

Convert datatype of payment type

```
cleaned_taxi_df$Payment.Type <- as.factor(cleaned_taxi_df$Payment.Type)
cleaned_taxi_df$Company <- as.factor(cleaned_taxi_df$Company)
cleaned_taxi_df$Taxi.ID <- as.factor(cleaned_taxi_df$Taxi.ID)
head(cleaned_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25
7	0000cf293ada965f89a98c8ccfae7b0ce3a03e41
8	0001235258d46a21317b6691ade9386c4d7e02c4

6 rows | 1-2 of 18 columns

Convert datatype of community area to string

```
cleaned_taxi_df$Pickup.Community.Area <- as.factor(cleaned_taxi_df$Pickup.Community.Are
a)
cleaned_taxi_df$Dropoff.Community.Area <- as.factor(cleaned_taxi_df$Dropoff.Community.Ar
ea)
```

Check dimension and summary of the cleaned dataset

```
dim(cleaned_taxi_df)
```

```
## [1] 379024     17
```

```
names(cleaned_taxi_df)
```

```
##  [1] "Trip.ID"                "Taxi.ID"                "Trip.Start.Timestamp"
##  [4] "Trip.Miles"             "Pickup.Community.Area"  "Dropoff.Community.Area"
##  [7] "Fare"                   "Tips"                   "Tolls"
## [10] "Extras"                 "Trip.Total"             "Payment.Type"
## [13] "Company"                "Trip.Minutes"           "Trip.Start.Date"
## [16] "Trip.Hour.Of.The.Day"   "Trip.Day.Of.The.Week"
```

```
summary(cleaned_taxi_df)
```

```
##    Trip.ID
##  Length:379024
##  Class :character
##  Mode  :character
##
##
##
##
##
Taxi.ID
##  d40dae7ea46d61abca67eb53b157fe9cf0b485cca6dce122604588a69aa6c4b6b78e0e5c5fd11f9702ba
bd94016122df1d328a459c8b7de2cb37a1bad947b1fe:   828
##  2780ead18beaa862cc67315ddabd9d1acaadcd6da82eba38b064d7d6f4acc260b68ef1ae3ce06dad8451
78107940b3493fa99640f0f70c25d15cf57336ab7b8f:   739
##  abd1ffa32433ceabeb49f4461015b38ddc252847ed3a29320aee6af650ba1e927195d191bf191f4f6f32
9ad7512a3f0f8e43ea844f3ead6f7c50fc4f0ccff08a:   637
##  37073e8c9e454886fe4a916f80a9a3478570e7dd3e663f40c5b81eae90f8f611027c67455f43b426f4c3
4dcb7fdb6697c82a3c6d00237f11a4a6cf5b1d1ce0c7:   635
##  13016372e777da1289d557edbe4ce2be8a68e77bc64768acaf5e0539b10be2ca089238dc27408b49b178
99014e6e178e17c3ba455812fd84024f93e266324439:   633
##  8da9e1d18757022c6a6a614fc2d38483e38aae441feff500095a83ebc68006cf88329f2c28e35ba92ead
14037739f9971a8a2852f946ebc59d0160c4f1104ec8:   630
##  (Other)
:374922
##  Trip.Start.Timestamp               Trip.Miles      Pickup.Community.Area
##  Min.   :2024-01-01 00:00:00.00   Min.   :  0.000   8      :82654
##  1st Qu.:2024-01-09 20:45:00.00   1st Qu.:  0.880   76     :72260
##  Median :2024-01-18 10:00:00.00   Median :  2.570   32     :63444
##  Mean   :2024-01-17 10:42:18.18   Mean   :  6.085   28     :42272
##  3rd Qu.:2024-01-24 16:30:00.00   3rd Qu.: 11.270   6      :12880
##  Max.   :2024-02-01 00:00:00.00   Max.   :664.900   56     :12047
##                                                     (Other):93467
##  Dropoff.Community.Area      Fare              Tips             Tolls
##  8      : 92741         Min.   :   0.00   Min.   :  0.000   Min.   :   0.000
##  32     : 64694         1st Qu.:   7.75   1st Qu.:  0.000   1st Qu.:   0.000
##  28     : 40248         Median :  14.00   Median :  0.040   Median :   0.000
##  6      : 19276         Mean   :  20.72   Mean   :  2.592   Mean   :   0.054
##  76     : 18269         3rd Qu.:  32.25   3rd Qu.:  3.700   3rd Qu.:   0.000
##  7      : 16568         Max.   :1525.00   Max.   :200.000   Max.   :4444.440
##  (Other):127228
##      Extras           Trip.Total           Payment.Type
##  Min.   :   0.000   Min.   :   0.00   Cash       :108481
##  1st Qu.:   0.000   1st Qu.:   9.75   Credit Card:136043
##  Median :   0.000   Median :  16.50   Dispute    :   102
##  Mean   :   1.327   Mean   :  24.86   Mobile     : 62672
##  3rd Qu.:   1.000   3rd Qu.:  36.30   No Charge  :   244
##  Max.   :5051.100   Max.   :8912.13   Prcard     : 50556
##                                       Unknown    : 20926
##                          Company       Trip.Minutes     Trip.Start.Date
##  Flash Cab                   :86779   Min.   :   0.00   Min.   :2024-01-01
##  Taxi Affiliation Services   :71541   1st Qu.:   7.28   1st Qu.:2024-01-10
##  Sun Taxi                    :40352   Median :  14.23   Median :2024-01-18
```

```
##  Taxicab Insurance Agency Llc:39522   Mean   :  18.48   Mean   :2024-01-17
##  City Service                :35399   3rd Qu.:  25.72   3rd Qu.:2024-01-24
##  Chicago Independents        :21779   Max.   :1435.58   Max.   :2024-02-01
##  (Other)                     :83652
##  Trip.Hour.Of.The.Day Trip.Day.Of.The.Week
##  Min.   : 0.00        1:34446
##  1st Qu.:10.00        2:59096
##  Median :14.00        3:67076
##  Mean   :13.69        4:72588
##  3rd Qu.:18.00        5:59096
##  Max.   :23.00        6:50350
##                       7:36372
```

```
head(cleaned_taxi_df)
```

	Trip.ID <chr>
1	0000184e7cd53cee95af32eba49c44e4d20adcd8
3	000074019d598c2b1d6e77fbae79e40b0461a2fc
5	00007c3e7546e2c7d15168586943a9c22c3856cf
6	0000bab44d0d673a222e7f1a0a6026563519aa25
7	0000cf293ada965f89a98c8ccfae7b0ce3a03e41
8	0001235258d46a21317b6691ade9386c4d7e02c4

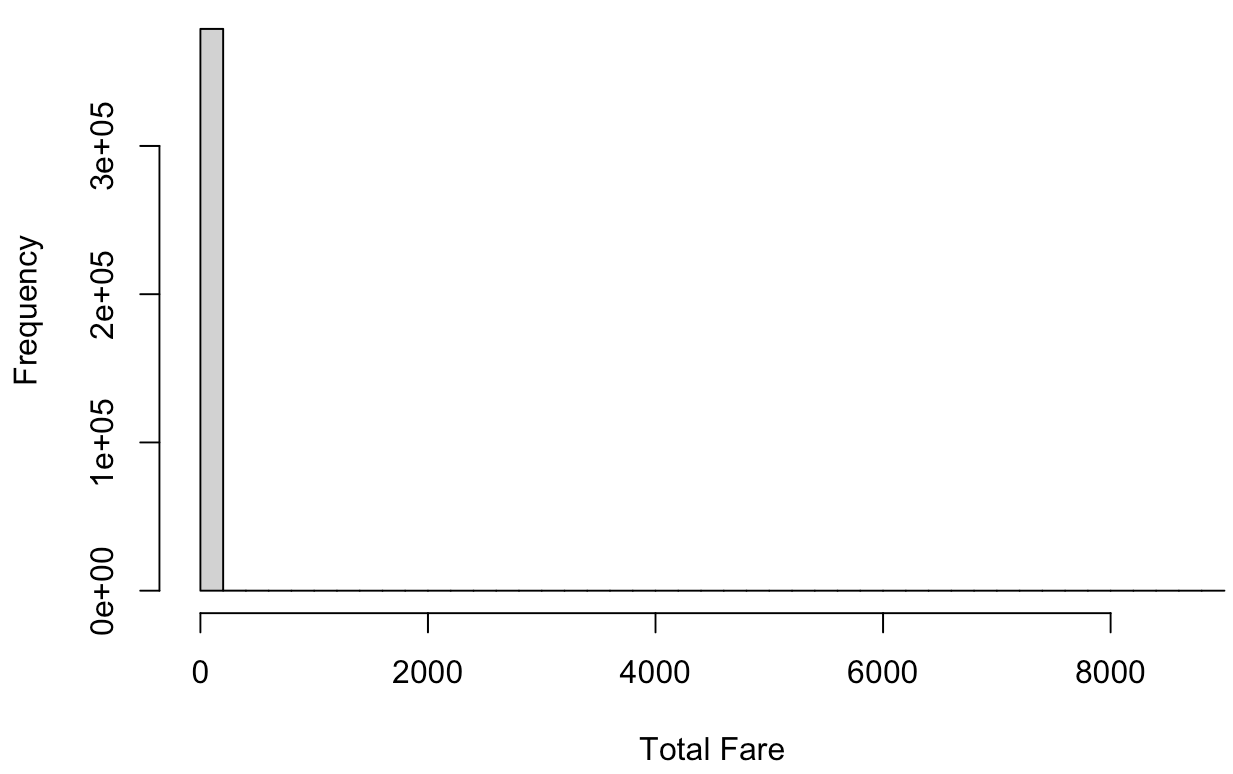
6 rows | 1-2 of 18 columns

2. Exploratory Data Analysis

2.1 Check and remove rows that has outliers

```
attach(cleaned_taxi_df)
```

```
hist(Trip.Total, breaks = 50, main = "Histogram of Total fare", xlab = "Total Fare")
```

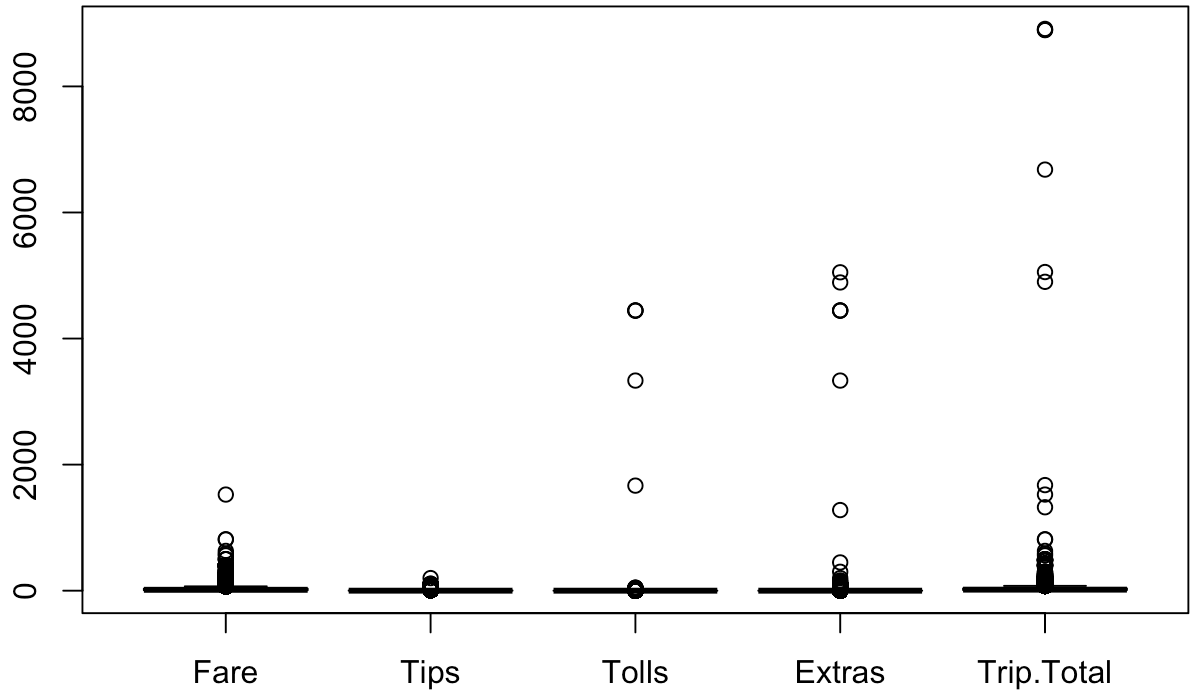



```
summary(Trip.Total)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.75   16.50   24.86   36.30 8912.13
```

Looking at the histogram and the boxplot, there are some extreme values that hinders accuracy of our future model, hence needs to be removed.

```
fare_related_features <- cleaned_taxi_df[, c('Fare','Tips','Tolls','Extras','Trip.Tota
l')]
boxplot(fare_related_features)
```




```
# Define the function to detect outliers for a single column
is.outlier <- function(x) {
  iqr <- IQR(x, na.rm = TRUE)
  lower <- quantile(x, 0.25, na.rm = TRUE) - 1.5 * iqr
  upper <- quantile(x, 0.75, na.rm = TRUE) + 1.5 * iqr
  return(x < lower | x > upper)
}

outliers <- is.outlier(Trip.Total)
```

number of outliers

```
sum(outliers)
```

```
## [1] 3889
```

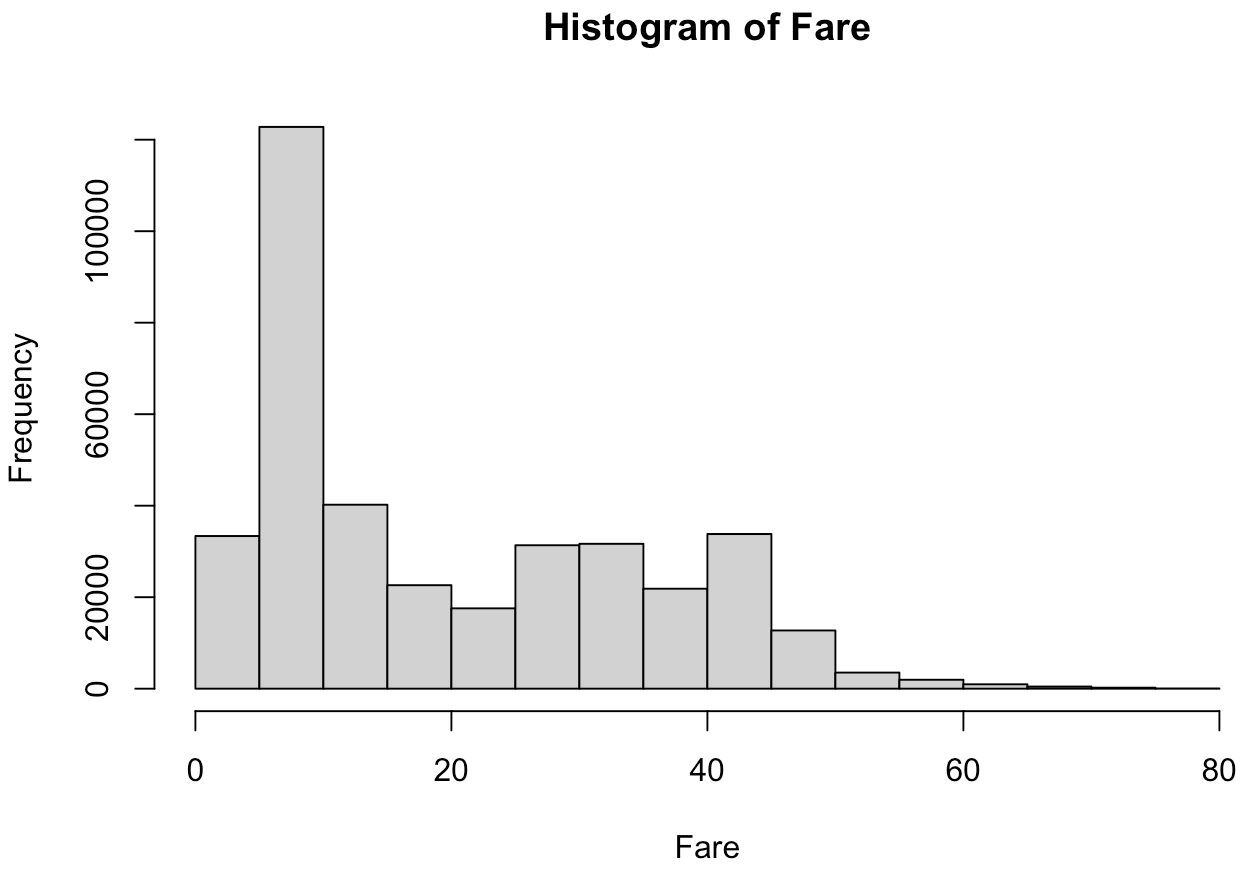
Remove all the rows that are outliers of Trip.Total

```
cleaned_taxi_df <- cleaned_taxi_df[!outliers, ]
dim(cleaned_taxi_df)
```

```
## [1] 375135     17
```

2.2 Histrogram of each features

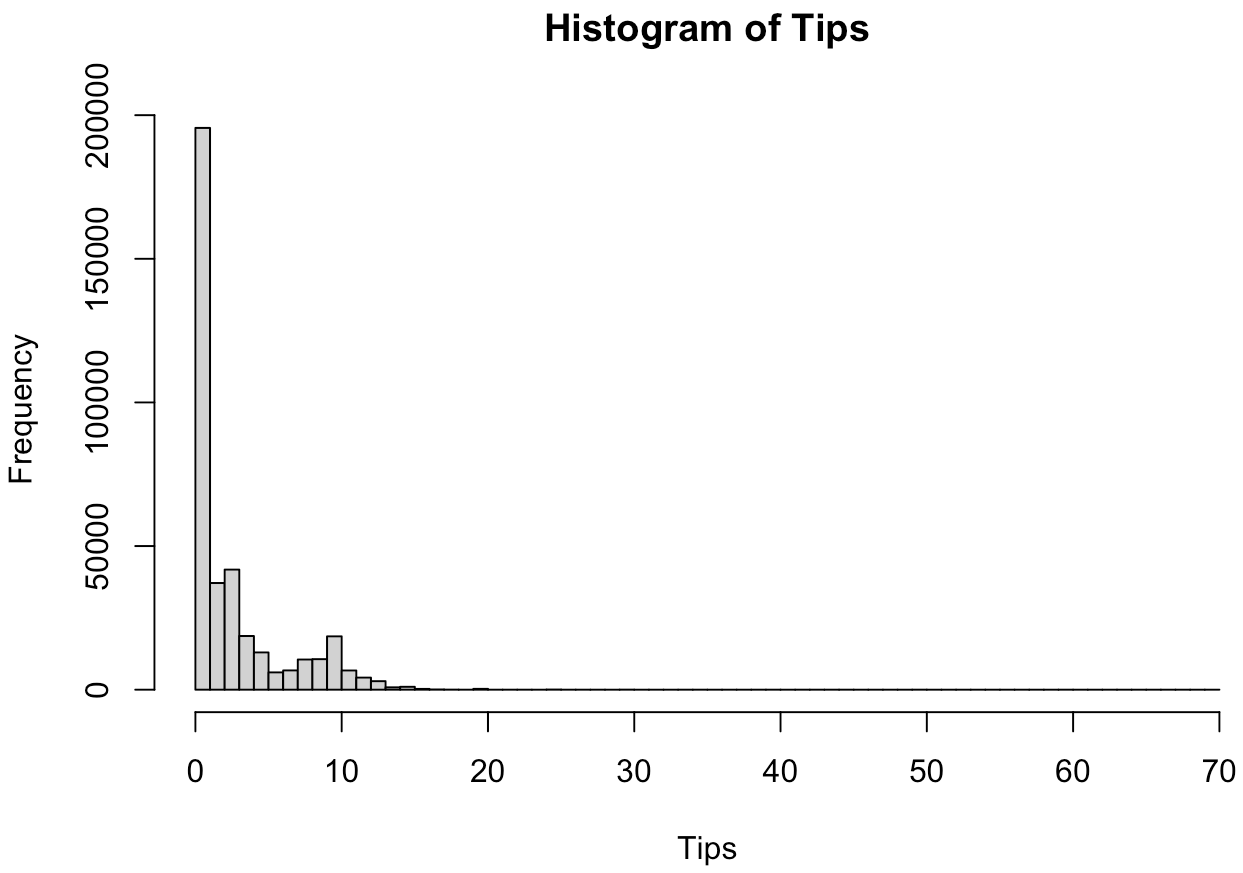
```
hist(cleaned_taxi_df$Fare, breaks = 25, main = "Histogram of Fare", xlab = "Fare")
```




```
summary(cleaned_taxi_df$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    7.75   13.75   20.13   31.75   76.00
```

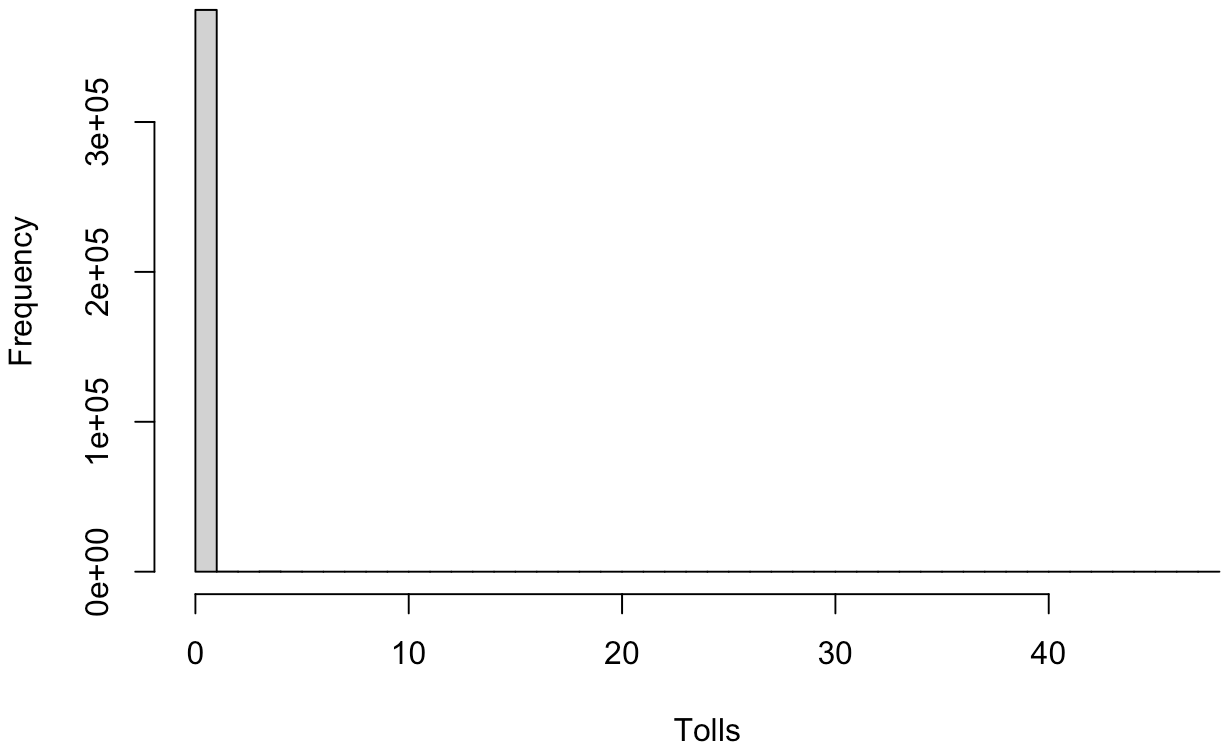
```
hist(cleaned_taxi_df$Tips, breaks = 50, main = "Histogram of Tips", xlab = "Tips")
```




```
summary(cleaned_taxi_df$Tips)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   2.507   3.550  70.000
```

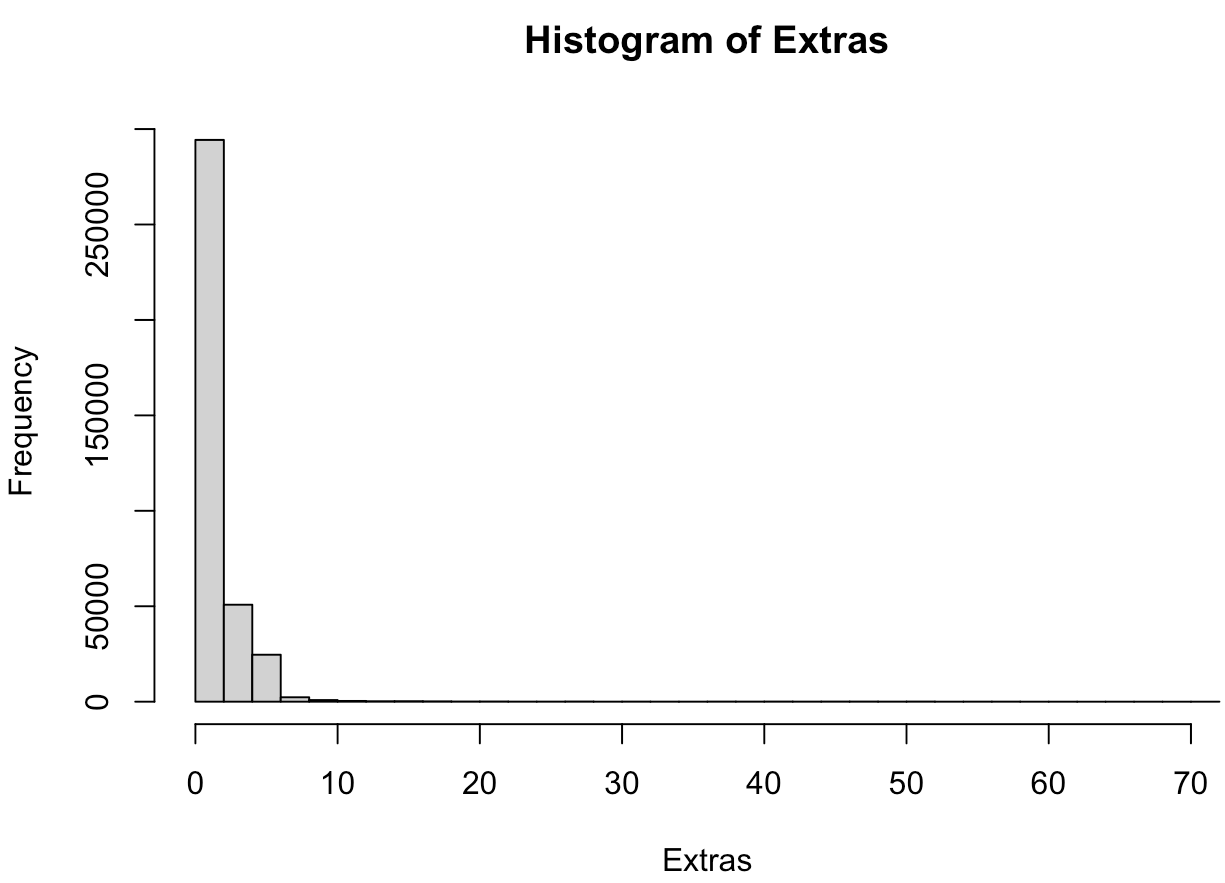
```
hist(cleaned_taxi_df$Tolls, breaks = 50, main = "Histogram of Tolls", xlab = "Tolls")
```




```
summary(cleaned_taxi_df$Tolls)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00000  0.00000  0.00000  0.00453  0.00000 48.00000
```

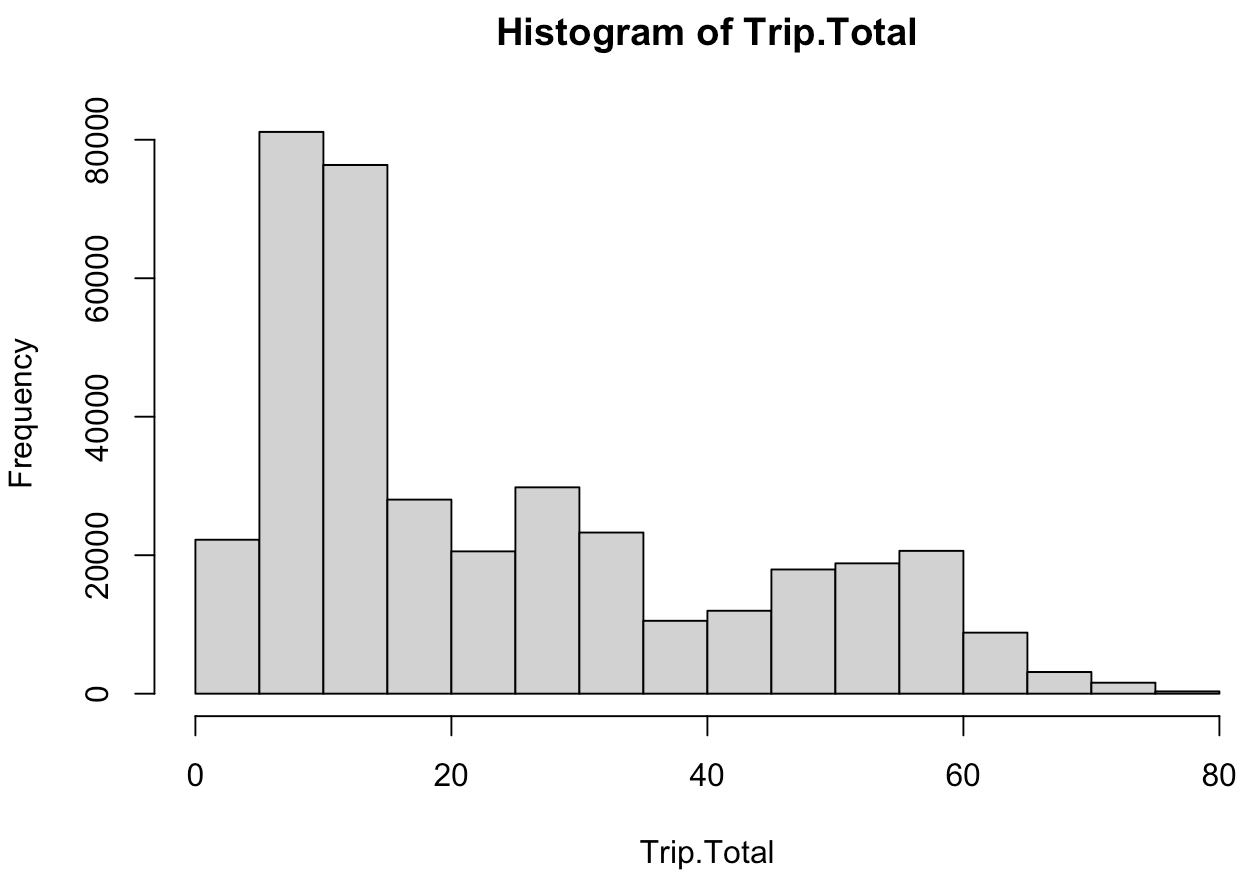
```
hist(cleaned_taxi_df$Extras, breaks = 50, main = "Histogram of Extras", xlab = "Extras")
```




```
summary(cleaned_taxi_df$Extras)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   1.199   1.000  72.000
```

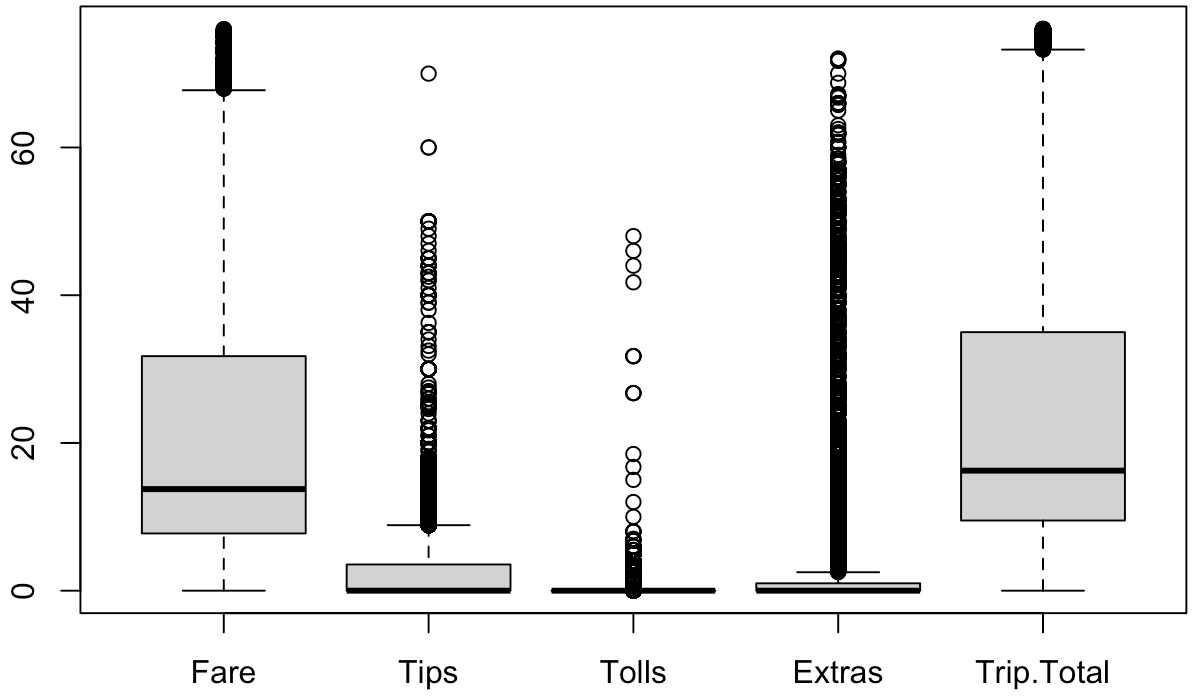
```
hist(cleaned_taxi_df$Trip.Total, breaks = 25, main = "Histogram of Trip.Total", xlab =
"Trip.Total")
```




```
summary(cleaned_taxi_df$Trip.Total)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.50   16.25   24.01   35.00   76.05
```

```
boxplot(cleaned_taxi_df[, c('Fare','Tips','Tolls','Extras','Trip.Total')])
```

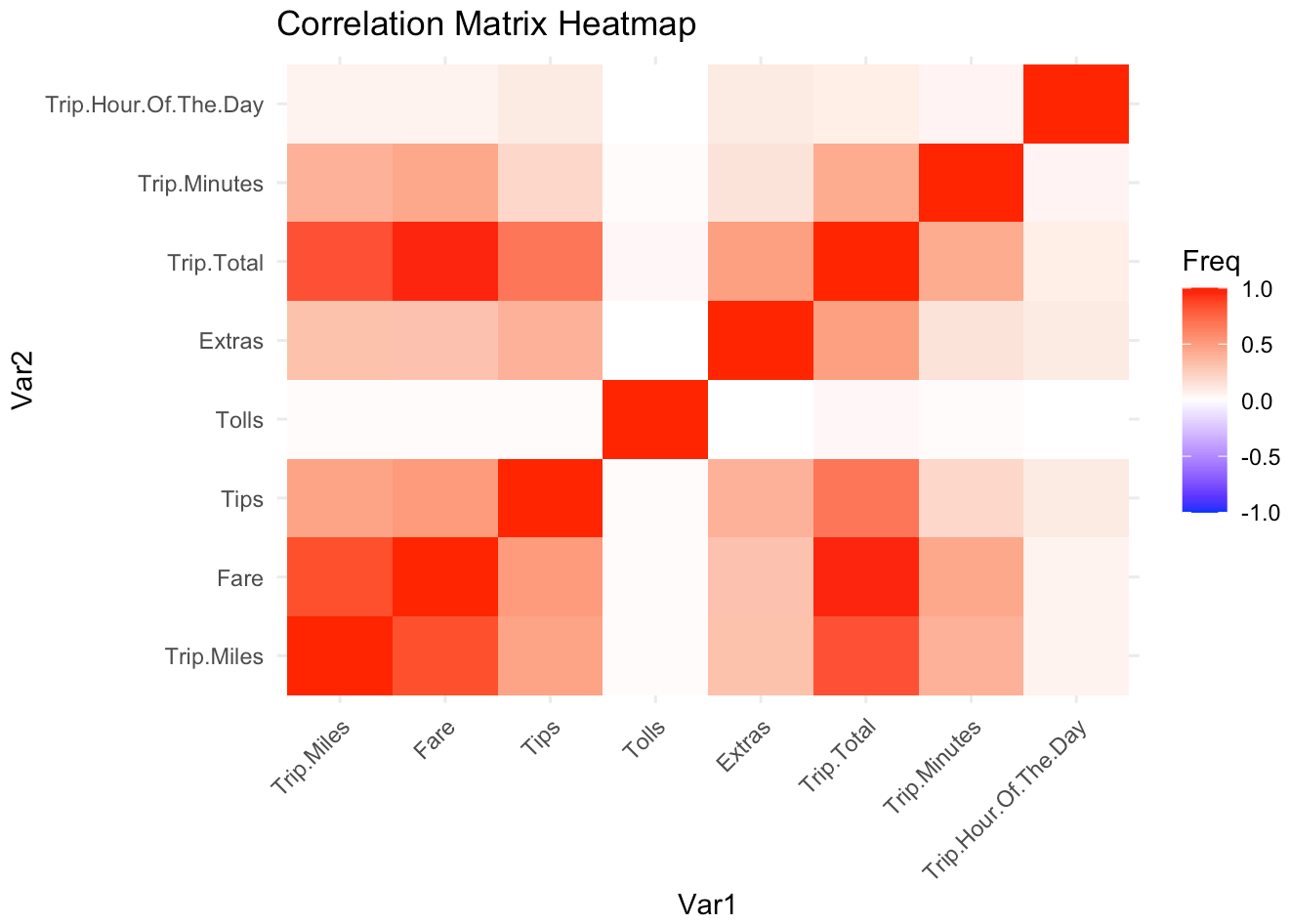


2.3 Heatmap of correlation of each features

```
# Select only numeric columns for correlation
numeric_columns <- sapply(cleaned_taxi_df, is.numeric)
cor_matrix <- cor(cleaned_taxi_df[, numeric_columns], use = "complete.obs")

print(cor_matrix)
```

```
##                      Trip.Miles       Fare       Tips         Tolls
## Trip.Miles           1.00000000 0.84004896 0.46749790  1.323617e-02
## Fare                 0.84004896 1.00000000 0.50713396  1.017268e-02
## Tips                 0.46749790 0.50713396 1.00000000  1.559658e-02
## Tolls                0.01323617 0.01017268 0.01559658  1.000000e+00
## Extras               0.31762504 0.31177482 0.38864411 -6.985914e-05
## Trip.Total           0.82845693 0.96424450 0.67724999  2.406436e-02
## Trip.Minutes         0.40196754 0.45073709 0.19767514  3.401189e-03
## Trip.Hour.Of.The.Day 0.05298319 0.05839975 0.09499803  1.705053e-03
##                             Extras Trip.Total Trip.Minutes Trip.Hour.Of.The.Day
## Trip.Miles            3.176250e-01 0.82845693  0.401967535          0.052983193
## Fare                  3.117748e-01 0.96424450  0.450737087          0.058399754
## Tips                  3.886441e-01 0.67724999  0.197675144          0.094998035
## Tolls                -6.985914e-05 0.02406436  0.003401189          0.001705053
## Extras                1.000000e+00 0.49444261  0.133194004          0.099926738
## Trip.Total            4.944426e-01 1.00000000  0.426929339          0.082680457
## Trip.Minutes          1.331940e-01 0.42692934  1.000000000          0.046674751
## Trip.Hour.Of.The.Day  9.992674e-02 0.08268046  0.046674751          1.000000000
```

```
# Convert the correlation matrix to a long format for ggplot2
cor_data <- as.data.frame(as.table(cor_matrix))

# Create a heatmap with ggplot2
ggplot(cor_data, aes(Var1, Var2, fill = Freq)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit =
c(-1,1)) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  ggtitle("Correlation Matrix Heatmap")
```

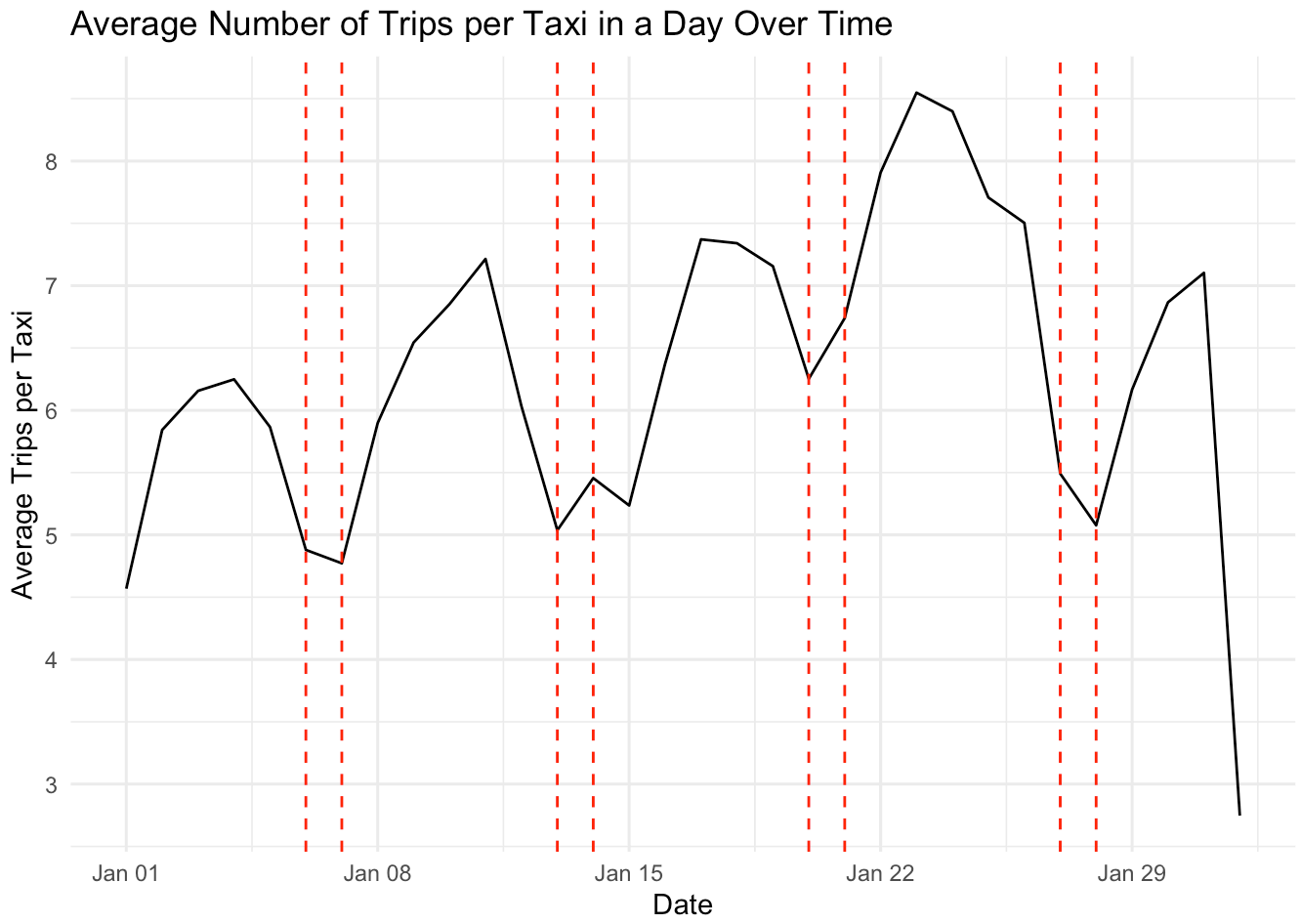


Trip.Total is in high correlation relationship with 'Fare, Trip.Miles, Tips', but almost no relationship with 'Tolls, Hour of the trip'

2.4 Other data exploration

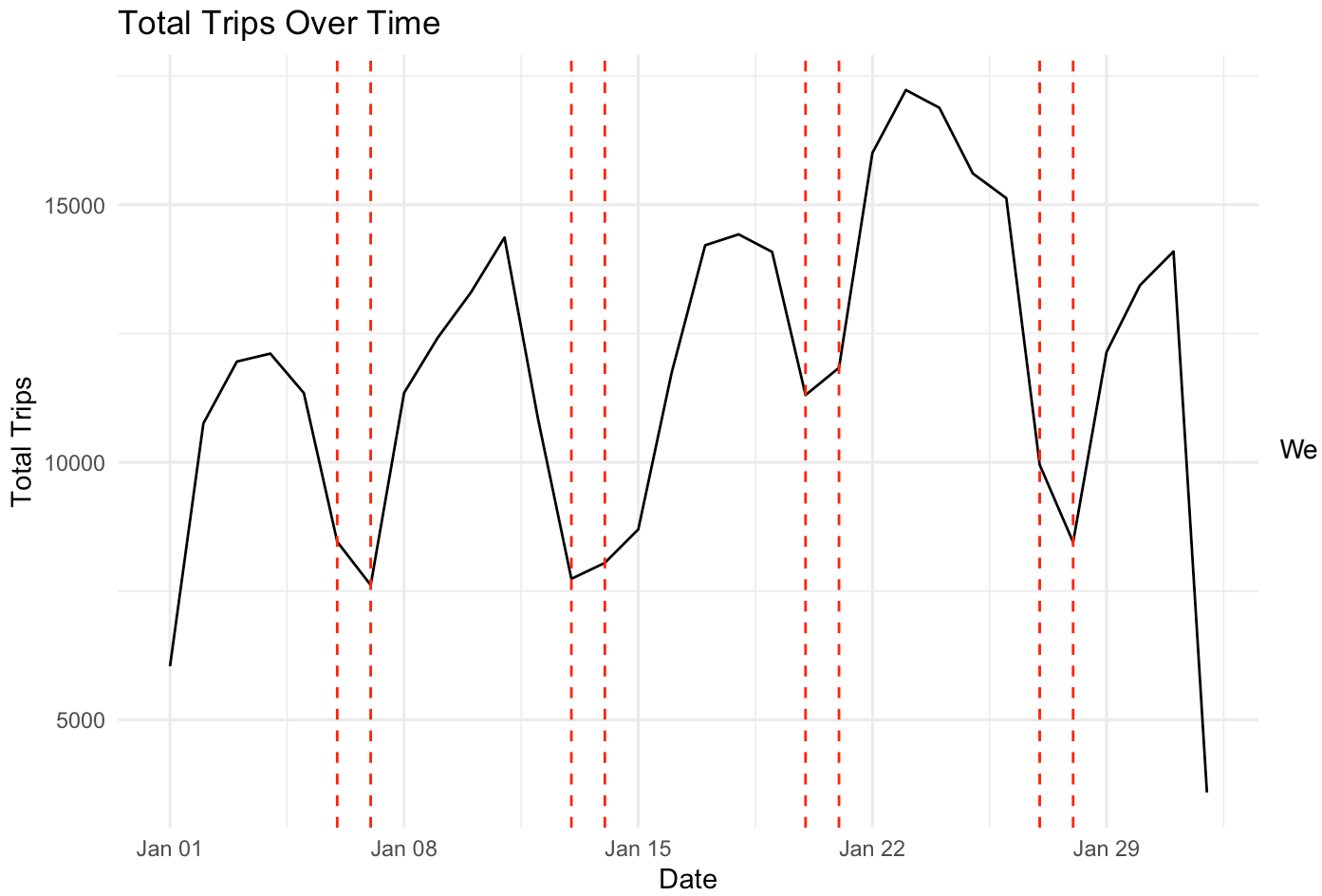
1) Average Number of Trips per Taxi in a Day Over Time

```
# Generate a sequence of dates within the range of your data
date_range <- seq(min(cleaned_taxi_df$Trip.Start.Date), max(cleaned_taxi_df$Trip.Start.D
ate), by = "day")

# Create a data frame of weekend dates
weekend_dates <- data.frame(Date = date_range[weekdays(date_range) %in% c("Saturday", "S
unday")])

# Group by Taxi ID and Date, then summarize the average trips
average_trips_per_taxi <- cleaned_taxi_df %>%
  group_by(Taxi.ID, Trip.Start.Date) %>%
  summarise(Trips = n(), .groups = 'drop') %>%
  group_by(Trip.Start.Date) %>%
  summarise(AvgTrips = mean(Trips), .groups = 'drop')

# Plot the average trips per taxi over time
plot <- ggplot(average_trips_per_taxi, aes(x = Trip.Start.Date, y = AvgTrips)) +
  geom_line() +
  labs(title = "Average Number of Trips per Taxi in a Day Over Time",
       x = "Date",
       y = "Average Trips per Taxi") +
  theme_minimal()

# Add vertical lines for weekends
plot + geom_vline(data = weekend_dates, aes(xintercept = as.numeric(Date)), color = "re
d", linetype = "dashed")
```



2) Total Trips Over Time

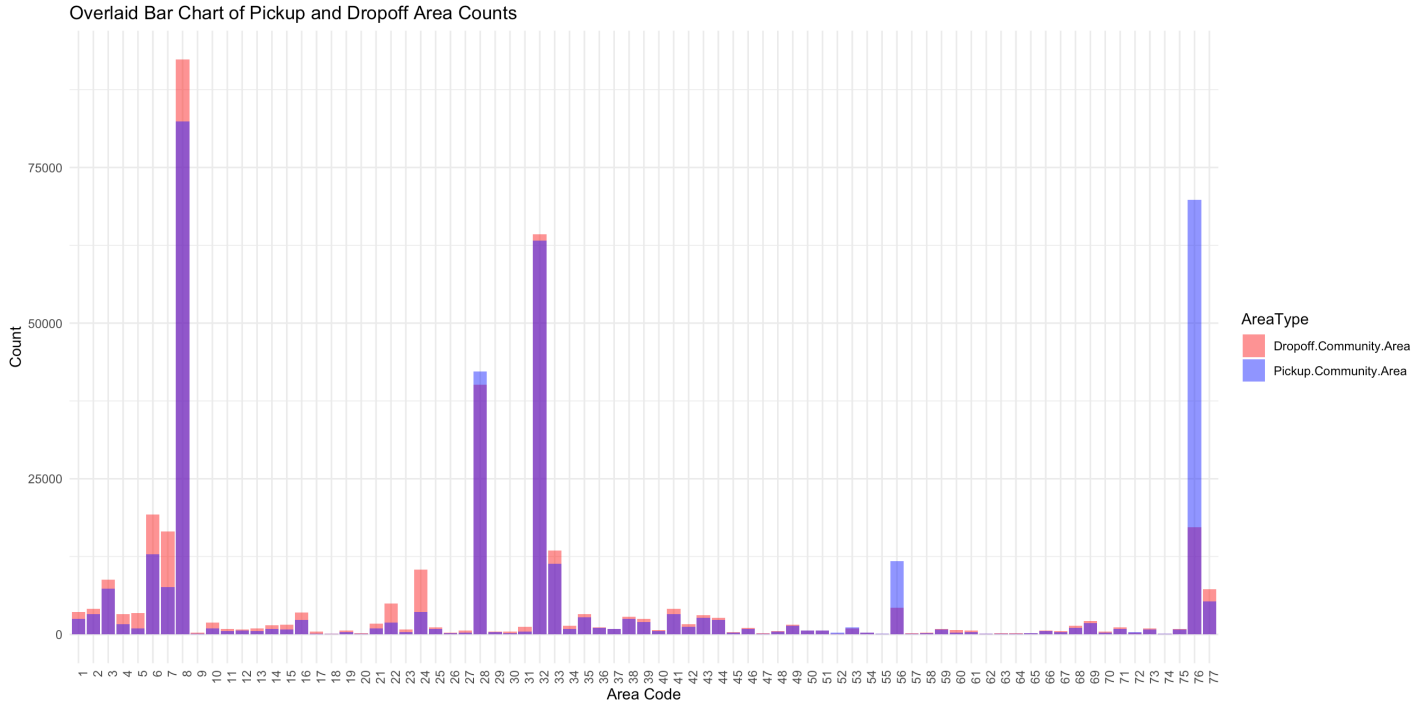
```
# First, calculate the total trips per day
total_trips_over_time <- cleaned_taxi_df %>%
  group_by(Trip.Start.Date) %>%
  summarise(TotalTrips = n(), .groups = 'drop')

# Plot the total trips over time
total_trips_plot <- ggplot(total_trips_over_time, aes(x = Trip.Start.Date, y = TotalTrip
s)) +
  geom_line() +
  labs(title = "Total Trips Over Time",
       x = "Date",
       y = "Total Trips") +
  theme_minimal()

# Add vertical lines for weekends
total_trips_plot + geom_vline(data = weekend_dates, aes(xintercept = as.numeric(Date)),
color = "red", linetype = "dashed")
```



can see that people tends not to use taxi on weekends than weekdays.

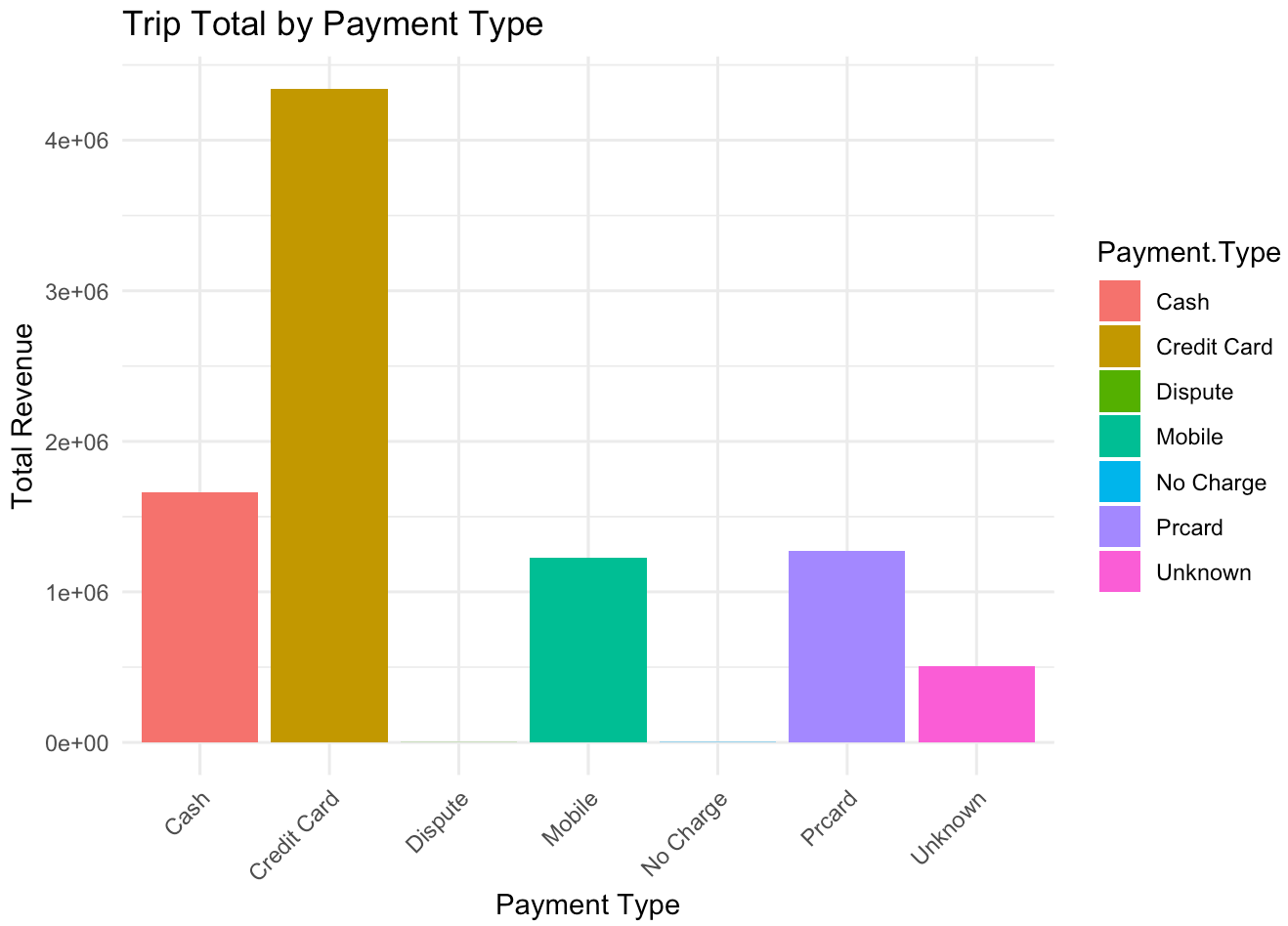
3) Pickup and Dropoff Area Count - Overlaid Histogram

```
# Create a long format data frame for pickup and dropoff areas
area_data <- tidyr::pivot_longer(
  cleaned_taxi_df,
  cols = c("Pickup.Community.Area", "Dropoff.Community.Area"),
  names_to = "AreaType",
  values_to = "Area"
)

# Plot overlaid bar charts for pickup and dropoff area counts
ggplot(area_data, aes(x = as.factor(Area), fill = AreaType)) +
  geom_bar(position = "identity", alpha = 0.5) +  # Set alpha for transparency
  scale_fill_manual(values = c("Pickup.Community.Area" = "blue", "Dropoff.Community.Are
a" = "red")) +
  labs(title = "Overlaid Bar Chart of Pickup and Dropoff Area Counts",
       x = "Area Code",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



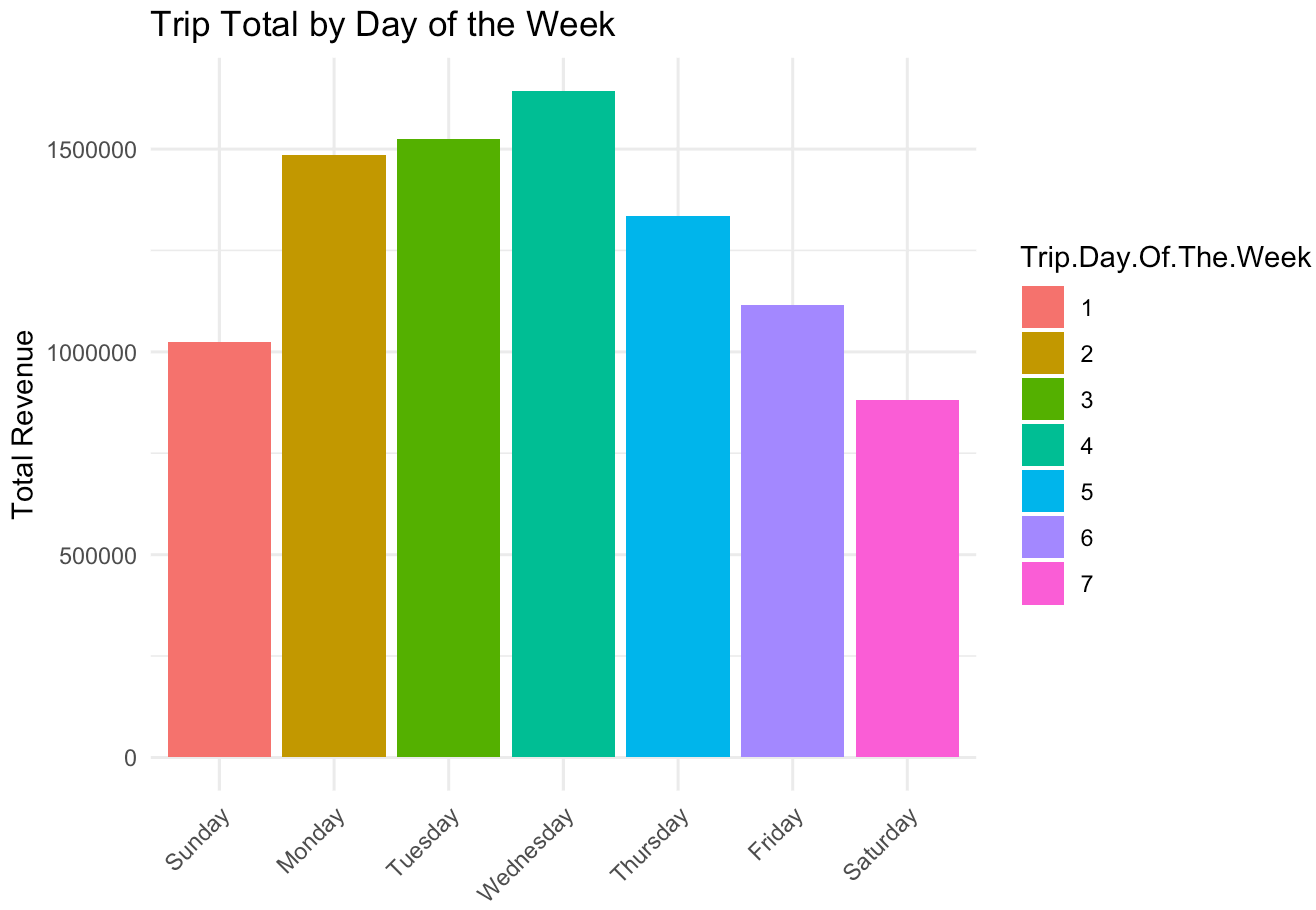
4) Trip.Total by Payment Type

```
# Summarise Trip.Total by Payment Type
trip_total_by_PaymentType <- cleaned_taxi_df %>%
  group_by(Payment.Type) %>%
  summarise(TotalRevenue = sum(Trip.Total), .groups = 'drop')

# Create a bar plot of Trip Total by Payment Type
ggplot(trip_total_by_PaymentType, aes(x = Payment.Type, y = TotalRevenue, fill = Paymen
t.Type)) +
  geom_col() +  # This creates a bar chart with pre-summarized data
  labs(title = "Trip Total by Payment Type",
       x = "Payment Type",
       y = "Total Revenue") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



5) Trip.Total by Day of the week

```
trip_total_by_DayoftheWeek <- cleaned_taxi_df %>%
  group_by(Trip.Day.Of.The.Week) %>%
  summarise(TotalRevenue = sum(Trip.Total), .groups = 'drop')

# Define a named vector to map day numbers to day names
day_names <- c("1" = "Sunday", "2" = "Monday", "3" = "Tuesday", "4" = "Wednesday",
               "5" = "Thursday", "6" = "Friday", "7" = "Saturday")

# Create the bar plot, using the named vector for axis labels
ggplot(trip_total_by_DayoftheWeek, aes(x = Trip.Day.Of.The.Week, y = TotalRevenue, fill
= Trip.Day.Of.The.Week)) +
  geom_col() +
  scale_x_discrete(labels = day_names) +  # Use the day_names vector for axis labels
  labs(title = "Trip Total by Day of the Week",
       x = " ",
       y = "Total Revenue") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



6) Trip.Total by Company

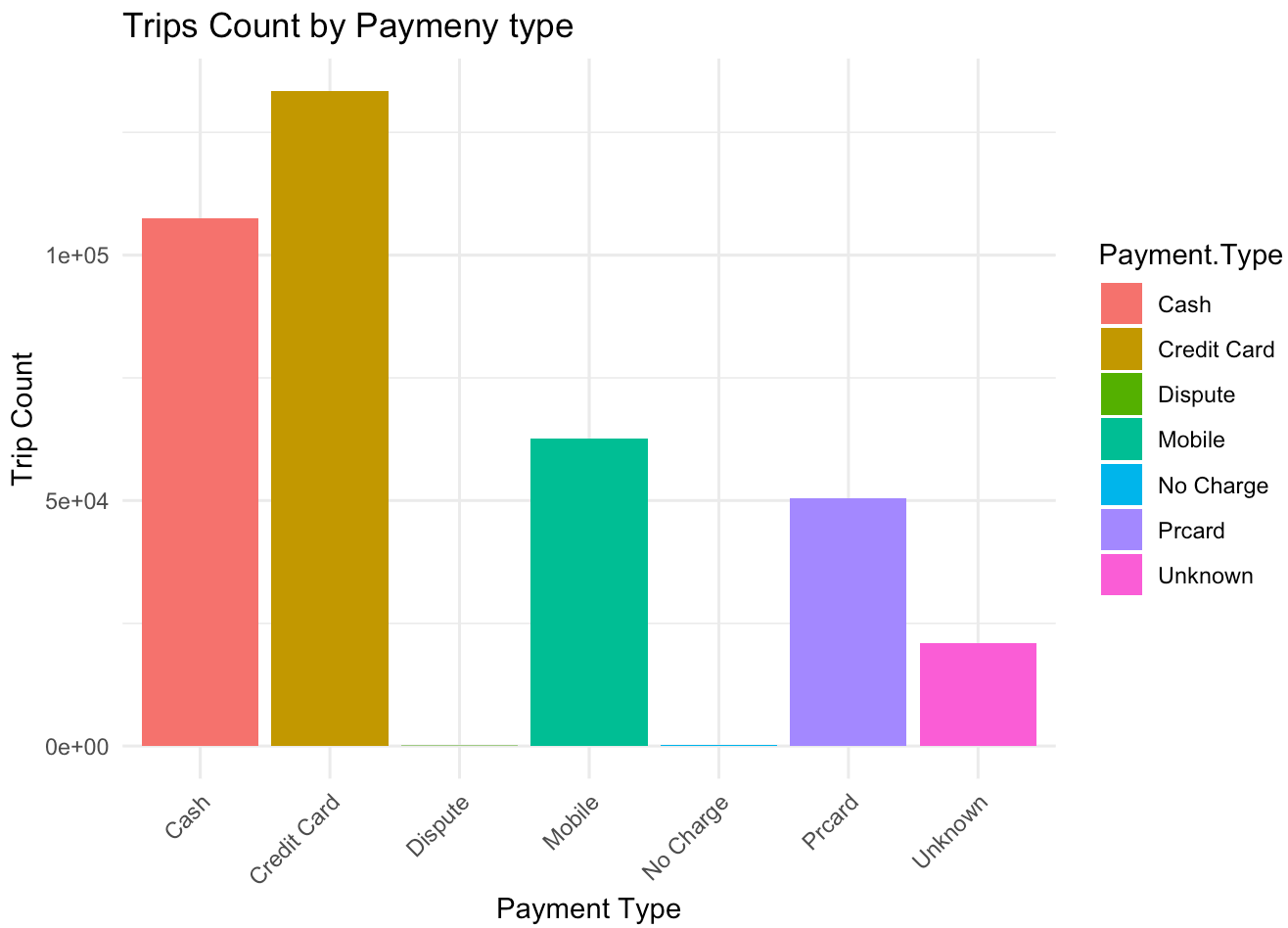
```
# Summarise Trip.Total by Company and arrange by TotalRevenue in descending order
trip_total_by_Company <- cleaned_taxi_df %>%
  group_by(Company) %>%
  summarise(TotalRevenue = sum(Trip.Total), .groups = 'drop') %>%
  arrange(desc(TotalRevenue))

# Display the full table sorted by Total Revenue
kable(trip_total_by_Company)
```

Company	TotalRevenue
Flash Cab	1990169.36
Taxi Affiliation Services	1675736.31
Taxicab Insurance Agency Llc	1006494.48
Sun Taxi	1006423.09
City Service	817276.91
Chicago Independents	556686.13
5 Star Taxi	482365.54
Globe Taxi	331868.53

Company	TotalRevenue
Blue Ribbon Taxi Association	271361.15
Medallion Leasin	253900.09
Taxicab Insurance Agency, LLC	151921.61
Choice Taxi Association	106725.28
Choice Taxi Association Inc	79464.05
Chicago City Taxi Association	64120.54
U Taxicab	48885.16
Top Cab	32126.52
Koam Taxi Association	18369.99
Chicago Taxicab	17667.09
Taxi Affiliation Services Llc - Yell	16331.27
Star North Taxi Management Llc	15055.09
312 Medallion Management Corp	13741.55
Patriot Taxi Dba Peace Taxi Associat	10606.40
Setare Inc	10504.64
Metro Jet Taxi A.	6202.93
5167 - 71969 5167 Taxi Inc	4909.89
3591 - 63480 Chuks Cab	3561.23
Petani Cab Corp	3191.65
6574 - Babylon Express Inc.	2965.80
Tac - Yellow Cab Association	2710.99
4053 - 40193 Adwar H. Nikola	2392.50
2733 - 74600 Benny Jona	2122.88
3556 - 36214 RC Andrews Cab	1712.25
Tac - Checker Cab Dispatch	991.72

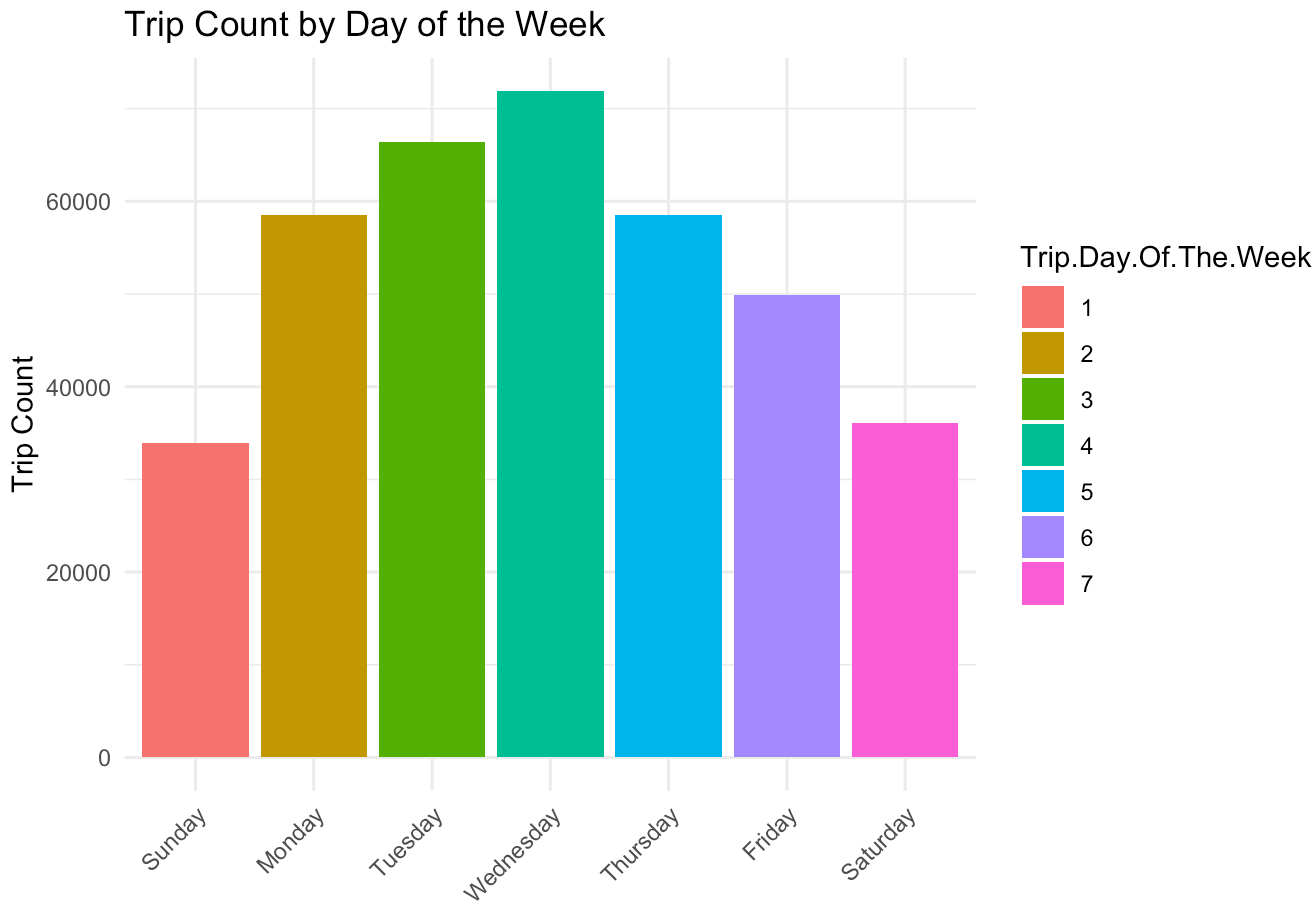
7) Trips Count by Paymeny type

```
# Summarise Trip id by Payment Type
trip_count_by_PaymentType <- cleaned_taxi_df %>%
  group_by(Payment.Type) %>%
  summarise(TripCount = n_distinct(Trip.ID))

# Create a bar plot of Trip Total by Payment Type
ggplot(trip_count_by_PaymentType, aes(x = Payment.Type, y = TripCount, fill = Payment.Ty
pe)) +
  geom_col() +  # This creates a bar chart with pre-summarized data
  labs(title = "Trips Count by Paymeny type",
       x = "Payment Type",
       y = "Trip Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



8) Trips Count by day of the week

```
trip_count_by_DayoftheWeek <- cleaned_taxi_df %>%
  group_by(Trip.Day.Of.The.Week) %>%
  summarise(TripCount = n_distinct(Trip.ID))

# Define a named vector to map day numbers to day names
day_names <- c("1" = "Sunday", "2" = "Monday", "3" = "Tuesday", "4" = "Wednesday",
               "5" = "Thursday", "6" = "Friday", "7" = "Saturday")

# Create the bar plot, using the named vector for axis labels
ggplot(trip_count_by_DayoftheWeek, aes(x = Trip.Day.Of.The.Week, y = TripCount, fill = T
rip.Day.Of.The.Week)) +
  geom_col() +
  scale_x_discrete(labels = day_names) +  # Use the day_names vector for axis labels
  labs(title = "Trip Count by Day of the Week",
       x = " ",
       y = "Trip Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



9) Trips Count by Company

```
# Summarise Trip.Total by Company and arrange by TotalRevenue in descending order
trip_count_by_Company <- cleaned_taxi_df %>%
  group_by(Company) %>%
  summarise(TripCount = n_distinct(Trip.ID), .groups = 'drop') %>%
  arrange(desc(TripCount))

# Display the full table sorted by Total Revenue
kable(trip_count_by_Company)
```

Company	TripCount
Flash Cab	86155
Taxi Affiliation Services	70815
Sun Taxi	39892
Taxicab Insurance Agency Llc	38976
City Service	35099
Chicago Independents	21511
5 Star Taxi	16979
Globe Taxi	14263
Blue Ribbon Taxi Association	12996
Medallion Leasin	11036
Taxicab Insurance Agency, LLC	6118
Choice Taxi Association	4194
Chicago City Taxi Association	3765
Choice Taxi Association Inc	3232
U Taxicab	2316
Top Cab	1629
Koam Taxi Association	1138
Taxi Affiliation Services Llc - Yell	829
Patriot Taxi Dba Peace Taxi Associat	762
Chicago Taxicab	666
Star North Taxi Management Llc	628
Setare Inc	412
312 Medallion Management Corp	347
3591 - 63480 Chuks Cab	328

Company	TripCount
Metro Jet Taxi A.	299
Tac - Yellow Cab Association	182
5167 - 71969 5167 Taxi Inc	176
3556 - 36214 RC Andrews Cab	91
6574 - Babylon Express Inc.	82
Petani Cab Corp	67
Tac - Checker Cab Dispatch	58
4053 - 40193 Adwar H. Nikola	51
2733 - 74600 Benny Jona	43

Some Key Findings from EDA

1. Total taxi fare correlates with original fare, trip miles, tips, where does not correlates with tolls and trip hours of the day.
2. People tends to use taxi on weekdays (especially on Wed > Tue > Mon), but not on weenkends.
3. Majority of people use Credit Card then Cash to pay the taxi fare.
4. The most popular Pickup area are ‘8, 76, 32, 28’.
5. The most popular Dropoff area are ‘8, 32, 28’.
6. Top 5 most frequently used taxi companies in Chicago are Flash Cab, Taxi Affiliation Services, Sun Taxi, Taxicab Insurance Agency Llc, and City Service.