Capital One : Data Science Coding Challenge

Ankush

November 9, 2017

Prelude :

The following analysis is done for the green taxis which can be hailed in Manhattan north of East 96th Street and West 110th Street, and all outer boroughs (the Bronx, Brooklyn, Queens, and Staten Island) except at the airports. The vehicles can drop passengers off anywhere, but will not be able to pick up new passengers within the “yellow zone” (south of East 96th and West 110th Streets) or within airports. (Source : Wikipedia)
The green taxi trip records include fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts. The data used in the attached datasets were collected and provided to the NYC Taxi and Limousine Commission (TLC) by technology providers authorized (corresponds to Vendor ID in the data) under the Taxicab & Livery Passenger Enhancement Program. (Source : http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml (http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml))
The following R packages have been used to perform this analysis :
1. caret
2. ggplot2
3. ggpubr
4. lubridate

Analysis :

Question 1
1. Programmatically download and load into your favorite analytical tool the trip data for September 2015.
2. Report how many rows and columns of data you have loaded.

```
options(scipen=999)

# Loading data in R
raw_data <- read.csv("https://s3.amazonaws.com/nyc-tlc/trip+data/green_tripdata_2015-0
9.csv")

# Analyzing the structure of the data
str(raw_data)
```

```
## 'data.frame':    1494926 obs. of  21 variables:
##  $ VendorID             : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ lpep_pickup_datetime : Factor w/ 1079075 levels "2015-09-01 00:00:00",..: 58 97
43 60 5 15 18 52 60 50 ...
##  $ Lpep_dropoff_datetime: Factor w/ 1077210 levels "2015-09-01 00:00:00",..: 3 6 6
22 5 11 14 12 27 28 ...
##  $ Store_and_fwd_flag   : Factor w/ 2 levels "N","Y": 1 1 1 1 1 1 1 1 1 1 ...
##  $ RateCodeID           : int  5 5 1 1 1 1 1 1 1 1 ...
##  $ Pickup_longitude     : num  -74 -74 -73.9 -73.9 -74 ...
##  $ Pickup_latitude      : num  40.7 40.9 40.8 40.8 40.7 ...
##  $ Dropoff_longitude    : num  -74 -74 -73.9 -73.9 -73.9 ...
##  $ Dropoff_latitude     : num  40.7 40.9 40.8 40.8 40.7 ...
##  $ Passenger_count      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Trip_distance        : num  0 0 0.59 0.74 0.61 1.07 1.43 0.9 1.33 0.84 ...
##  $ Fare_amount          : num  7.8 45 4 5 5 5.5 6.5 5 6 5.5 ...
##  $ Extra                : num  0 0 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ MTA_tax              : num  0 0 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
##  $ Tip_amount           : num  1.95 0 0.5 0 0 1.36 0 0 1.46 0 ...
##  $ Tolls_amount         : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Ehail_fee            : logi  NA NA NA NA NA NA ...
##  $ improvement_surcharge: num  0 0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 ...
##  $ Total_amount         : num  9.75 45 5.8 6.3 6.3 8.16 7.8 6.3 8.76 6.8 ...
##  $ Payment_type         : int  1 1 1 2 2 1 1 2 1 2 ...
##  $ Trip_type            : int  2 2 1 1 1 1 1 1 1 1 ...
```

The data consists of 1,494,926 rows and 21 columns.

Before we proceed further with the next question and delve deeper into our analysis, it is worth performing an exploratory data analysis (EDA) of the data and understand what we are dealing with. This will also help us prepare our data for further analysis. We will be looking at any abnormalities in our data (like datatype of variables, missing values, different factor levels for categorical data, outliers etc) as they might have repurcussions later on in this exercise.

Summarizing the data :

```
# Summary statistics for the raw data
summary(raw_data)
```

```
##     VendorID              lpep_pickup_datetime
##  Min.   :1.000   2015-09-20 02:00:32:      9
##  1st Qu.:2.000   2015-09-05 14:57:48:      8
##  Median :2.000   2015-09-10 17:43:49:      8
##  Mean   :1.782   2015-09-13 00:27:28:      8
##  3rd Qu.:2.000   2015-09-13 01:06:29:      8
##  Max.   :2.000   2015-09-26 22:48:40:      8
##                  (Other)            :1494877
##          Lpep_dropoff_datetime Store_and_fwd_flag   RateCodeID
##  2015-09-28 00:00:00:    172   N:1486192          Min.   : 1.000
##  2015-09-13 00:00:00:    153   Y:   8734          1st Qu.: 1.000
##  2015-09-19 00:00:00:    141                      Median : 1.000
##  2015-09-14 00:00:00:    126                      Mean   : 1.098
##  2015-09-21 00:00:00:    125                      3rd Qu.: 1.000
##  2015-09-12 00:00:00:    119                      Max.   :99.000
##  (Other)            :1494090
##  Pickup_longitude Pickup_latitude Dropoff_longitude Dropoff_latitude
##  Min.   :-83.32   Min.   : 0.00   Min.   :-83.43    Min.   : 0.00
##  1st Qu.:-73.96   1st Qu.:40.70   1st Qu.:-73.97    1st Qu.:40.70
##  Median :-73.95   Median :40.75   Median :-73.95    Median :40.75
##  Mean   :-73.83   Mean   :40.69   Mean   :-73.84    Mean   :40.69
##  3rd Qu.:-73.92   3rd Qu.:40.80   3rd Qu.:-73.91    3rd Qu.:40.79
##  Max.   :  0.00   Max.   :43.18   Max.   :  0.00    Max.   :42.80
##
##  Passenger_count Trip_distance      Fare_amount          Extra
##  Min.   :0.000   Min.   :  0.000   Min.   :-475.00   Min.   :-1.0000
##  1st Qu.:1.000   1st Qu.:  1.100   1st Qu.:   6.50   1st Qu.: 0.0000
##  Median :1.000   Median :  1.980   Median :   9.50   Median : 0.5000
##  Mean   :1.371   Mean   :  2.968   Mean   :  12.54   Mean   : 0.3513
##  3rd Qu.:1.000   3rd Qu.:  3.740   3rd Qu.:  15.50   3rd Qu.: 0.5000
##  Max.   :9.000   Max.   :603.100   Max.   : 580.50   Max.   :12.0000
##
##     MTA_tax          Tip_amount       Tolls_amount      Ehail_fee
##  Min.   :-0.5000   Min.   :-50.000   Min.   :-15.2900   Mode:logical
##  1st Qu.: 0.5000   1st Qu.:  0.000   1st Qu.:  0.0000   NA's:1494926
##  Median : 0.5000   Median :  0.000   Median :  0.0000
##  Mean   : 0.4866   Mean   :  1.236   Mean   :  0.1231
##  3rd Qu.: 0.5000   3rd Qu.:  2.000   3rd Qu.:  0.0000
##  Max.   : 0.5000   Max.   :300.000   Max.   : 95.7500
##
##  improvement_surcharge  Total_amount      Payment_type     Trip_type
##  Min.   :-0.3000       Min.   :-475.00   Min.   :1.000   Min.   :1.000
##  1st Qu.: 0.3000       1st Qu.:   8.16   1st Qu.:1.000   1st Qu.:1.000
##  Median : 0.3000       Median :  11.76   Median :2.000   Median :1.000
##  Mean   : 0.2921       Mean   :  15.03   Mean   :1.541   Mean   :1.022
##  3rd Qu.: 0.3000       3rd Qu.:  18.30   3rd Qu.:2.000   3rd Qu.:1.000
##  Max.   : 0.3000       Max.   : 581.30   Max.   :5.000   Max.   :2.000
##                                                          NA's   :4
```

Following are the issues and its remedy from summary statistics of our data :
1. Ehail_fee variable is entirely null so it can be removed totally from our data
2. RateCodeID = 99 which has no meaning as data dictionary suggest 1-6 as the possible values for RateCodeID. Again this could be a data issue and hence can be removed
3. The structure of the data and summary statistics also tell us that RateCodeID, Payment_type and Trip_type have “int” datatype whereas these should be factors as they are categorical data and take fixed set of values as mentioned in the data dictionary
4. Earlier while looking at the structure of the data, we found that lpep_pickup_datetime and Lpep_dropoff_datetime have “Factor” datatype whereas these should be date time format
5. 4 missing values in an otherwise well populated Trip_type variable. Could be data issue and hence can be removed
6. Presence of negative values in the payment amount related variables. It will be worth investigating the nature of such payments and how to treat them

Let’s start with issues in the order they are listed above :
1. Removing Ehail_fee from our data as it is completely missing

```
#removing ehail_fee as it's entirely blank
raw_data_2 <- raw_data[ , !(names(raw_data) %in% "Ehail_fee")]
```

2. RateCodeID =99 and removing such cases as they have no meaning from data dictionary and are a meagre 6 cases

```
# Checking rate code 99 in the data
ratecodeID_check <- raw_data_2[which(raw_data_2$RateCodeID==99),]

# Removing rate code 99 as such rate code has no meaning from data dictionary and als
o has 0 trip distance (as well as null value in trip type)
raw_data_2 <- raw_data_2[raw_data_2$RateCodeID!=99,]
```

3 and 4. Converting variables lpep_pickup_datetime, Lpep_dropoff_datetime, RateCodeID, Payment_type and Trip_type to the correct data type.

```
# Converting lpep_pickup_datetime and Lpep_dropoff_datetime from factors to date time
raw_data_2[,2] <- as.POSIXct(as.character(raw_data_2[,2]), format = "%Y-%m-%d %H:%M:%
S")
raw_data_2[,3] <- as.POSIXct(as.character(raw_data_2[,3]), format = "%Y-%m-%d %H:%M:%
S")

# Converting RateCodeID, Payment_type and Trip_type from int to factor
cols <- c(5, 19, 20)
raw_data_2[,cols] <- lapply(raw_data_2[,cols], factor)
```

This treatment takes care of null values in trip_type as well (our 5th point)
6. Presence of negative values in the payment amount related variables.

```
# Number of observations with negative values in Fare_amount, Extra, MTA_tax, Tip_amou
nt, Tolls_amount, improvement_surcharge and Total_amount
sum(raw_data_2$Fare_amount<0)
```

```
## [1] 2417
```

```
sum(raw_data_2$Extra<0)
```

```
## [1] 1255
```

```
sum(raw_data_2$MTA_tax<0)
```

```
## [1] 2187
```

```
sum(raw_data_2$Tip_amount<0)
```

```
## [1] 38
```

```
sum(raw_data_2$Tolls_amount<0)
```

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

```
sum(raw_data_2$improvement_surcharge<0)
```

```
## [1] 2215
```

```
sum(raw_data_2$Total_amount<0)
```

```
## [1] 2417
```

From these numbers, it appears all the cases where extra, mta_tax, tip_amount, tolls_amount and improvement_surcharge are negative are subsumed in the case where total_amount is also negative. We can check this hypothesis as follows :

```
# Check above hypothesis
sum(raw_data_2[raw_data_2$Fare_amount<0,"Total_amount"]<0)
```

```
## [1] 2417
```

```
sum(raw_data_2[raw_data_2$Extra<0,"Total_amount"]<0)
```

```
## [1] 1254
```

```
sum(raw_data_2[raw_data_2$MTA_tax<0,"Total_amount"]<0)
```

```
## [1] 2187
```

```
sum(raw_data_2[raw_data_2$Tip_amount<0,"Total_amount"]<0)
```

```
## [1] 38
```

```
sum(raw_data_2[raw_data_2$Tolls_amount<0,"Total_amount"]<0)
```

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

```
sum(raw_data_2[raw_data_2$improvement_surcharge<0,"Total_amount"]<0)
```

```
## [1] 2215
```

Our hypothesis was correct. Next, I want to know the distinctive feature of these cases where total_payment is negative like their payment type, trip type and rate code IDs

```
# Payment type for negative fares - most of them are no charge/dispute payment type
(3,4)
table(raw_data_2[raw_data_2$improvement_surcharge<0,"Payment_type"])
```

```
##
##    1    2    3    4    5
##    3  194 1209  809    0
```

```
# Trip type for negative fares - Shows overwhelming number of rides are street hail
(1)
table(raw_data_2[raw_data_2$improvement_surcharge<0,"Trip_type"])
```

```
##
##    1    2
## 2215    0
```

```
# Rate Code ID for negative fares - Shows overwhelming number of rides are std rate
(1)
table(raw_data_2[raw_data_2$improvement_surcharge<0,"RateCodeID"])
```

```
##
##    1    2    3    4    5    6
## 2063  116   24    1    7    4
```

We observe that most of the payment types were either no charge or disputed payments, overwhelmingly are street hail and metered on standard rate. The disputed payments characterisitic suggests that there exists an equally positive amount for these trips which was settled by cash. We can check this hypothesis :

```
# Merging cases with negative total_amount back to our data to see if there exists rec
ords for the same ride (i.e. for same pickup time, drop time, pickup lat, long, dropof
f lat, long)
merged_data <- merge ( raw_data_2[raw_data_2$Total_amount<0, ],
                       raw_data_2[raw_data_2$Total_amount >= 0,],
                       by=c("lpep_pickup_datetime", "Lpep_dropoff_datetime",
                            "Pickup_longitude", "Pickup_latitude",
                            "Dropoff_longitude", "Dropoff_latitude"),
                       all.x = TRUE)

# Checking if the absolute value of negative total amount is equal to positive total a
mount
merged_data$check <- ifelse(abs(merged_data$Total_amount.x)==merged_data$Total_amount.
y, 1, 0)
table(merged_data$check)
```

```
##
##    1
## 2403
```

```
table(merged_data$Payment_type.y)
```

```
##
##    1    2    3    4    5
##   81 2322    0    0    0
```

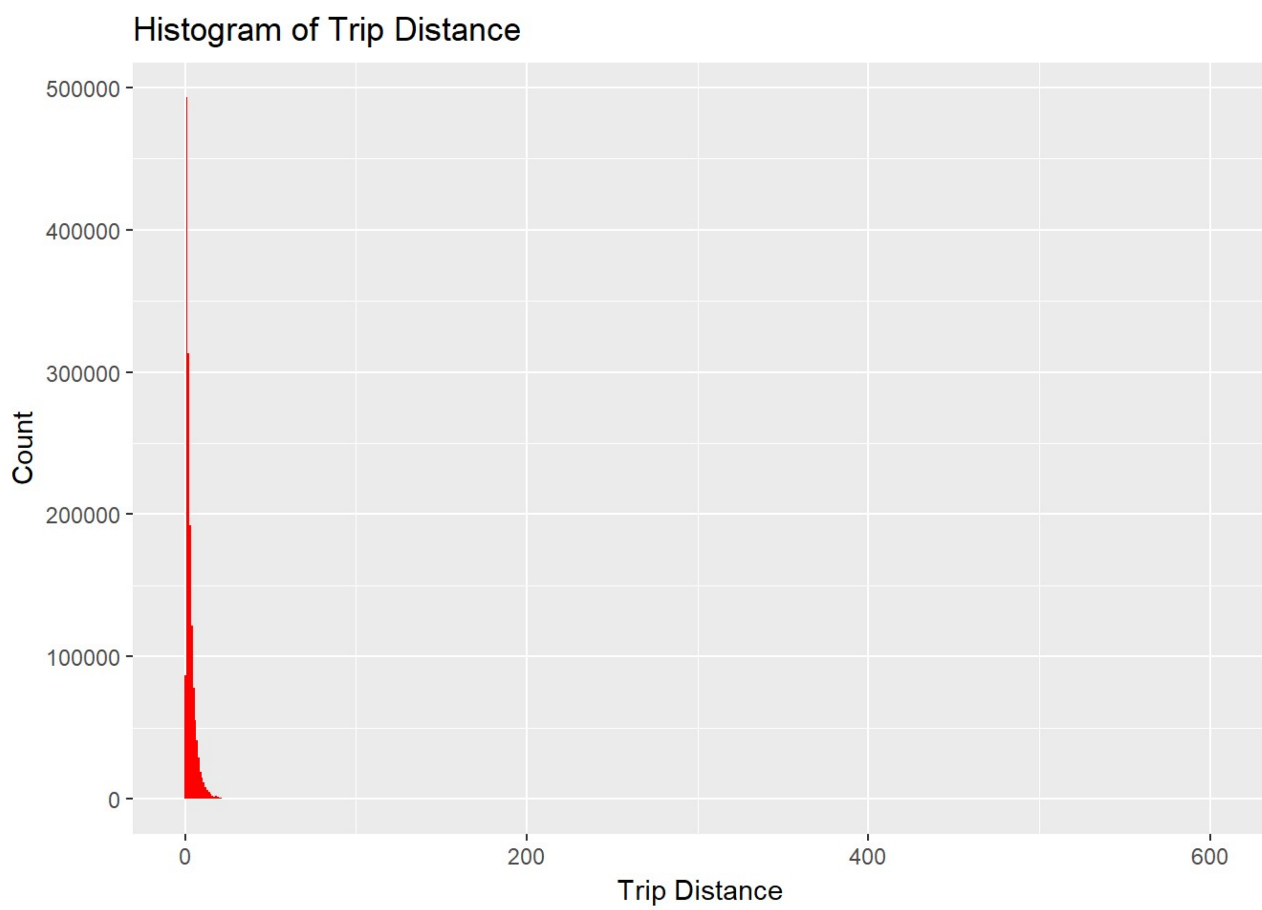
As we can see our hypothesis proved correct and we observe that all but 14 rides which have negative fare amounts also have a positive fare amount for the same ride which is predominantly paid by in cash and it suggests that this behavior could be because of payment failure via card. Hence the negative fares can be safely removed from our analyses.

```
# Removing observations with negative total_amount (0.16% of overall data)
raw_data_2 <- raw_data_2[raw_data_2$Total_amount>=0,]
```

Having prepared our data for various anomalies, we can now tackle our questions.

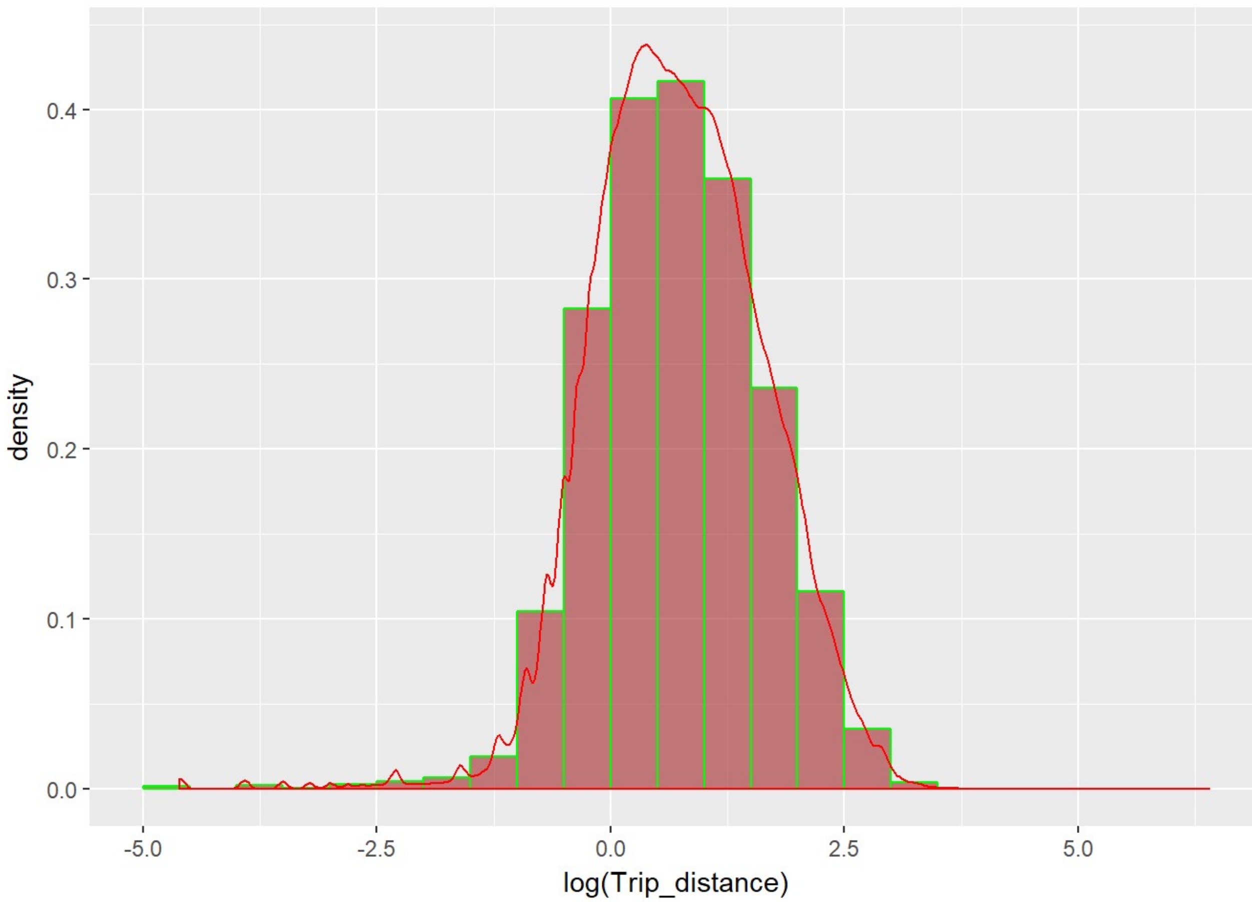
Question 2 :
1. Plot a histogram of the number of the trip distance (“Trip Distance”)
2. Report any structure you find and any hypotheses you have about that structure.

```
# Histogram of trip distance
library(ggplot2)
ggplot(data=raw_data_2, aes(Trip_distance)) +
  geom_histogram(binwidth = 1,
                 fill="red") +
  labs(title="Histogram of Trip Distance") +
  labs(x="Trip Distance", y="Count")
```



The histogram of the number of the trip distance shows the presence of extreme values (outliers) which is leading to a skew in the distribution. To improve the interpretability/appearance of the graphs, let’s look into logarithmic transformation of trip distance.

```
# Histogram of Log transformation of trip distance (removing 0 distance observations a
s log is undefined)
ggplot(data=raw_data_2[raw_data_2$Trip_distance>0,], aes(log(Trip_distance))) +
  geom_histogram(aes(y =..density..),
                 breaks=seq(-5, 5, by = 0.5),
                 col="green",
                 fill="brown",
                 alpha = 0.6) +
  geom_density(col=2)
```



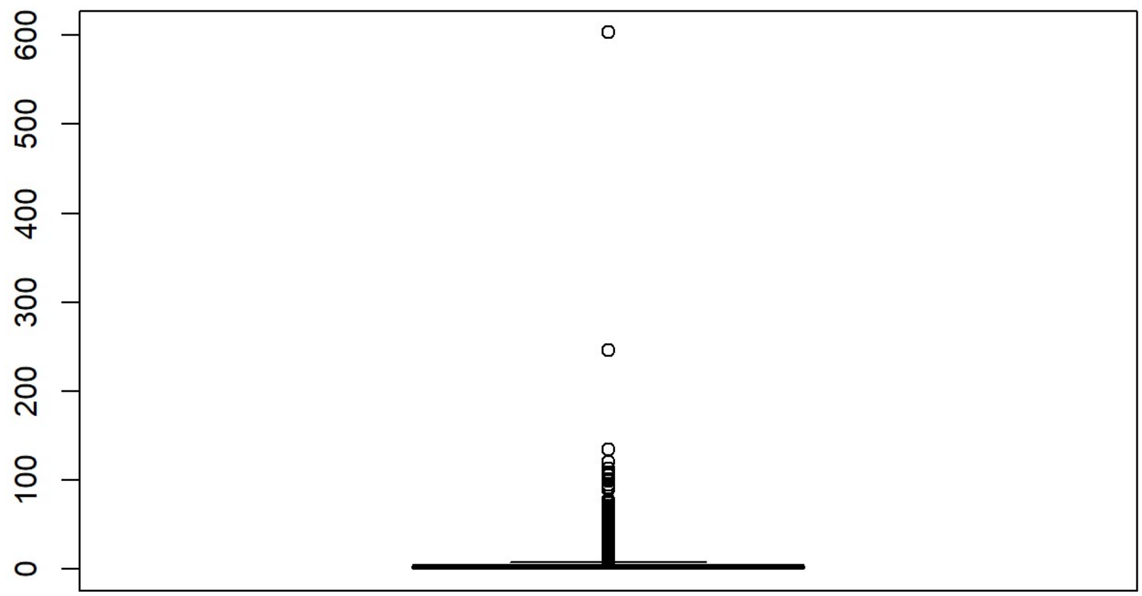

```
  labs(title="Histogram of Log transformation of Trip Distance")
```

```
## $title
## [1] "Histogram of Log transformation of Trip Distance"
##
## attr(,"class")
## [1] "labels"
```

Looking at the log transformation of trip distance, it appears that it’s compatible with normal distribution suggesting that trip distance is compatible with log normal distribution which is evident from the trendline on the graph.

Let’s investigate the outlier in trip_distance and generate a neat histogram of trip_distance

```
# Creating a boxplot of trip_distance
myboxplot <- boxplot(raw_data_2$Trip_distance, col = "blue")
```




```
# Out measure from boxplot obtains outlier observation and we can leverage that to rem
ove outliers from our data to bring a neat histogram plot
length(myboxplot$out)
```

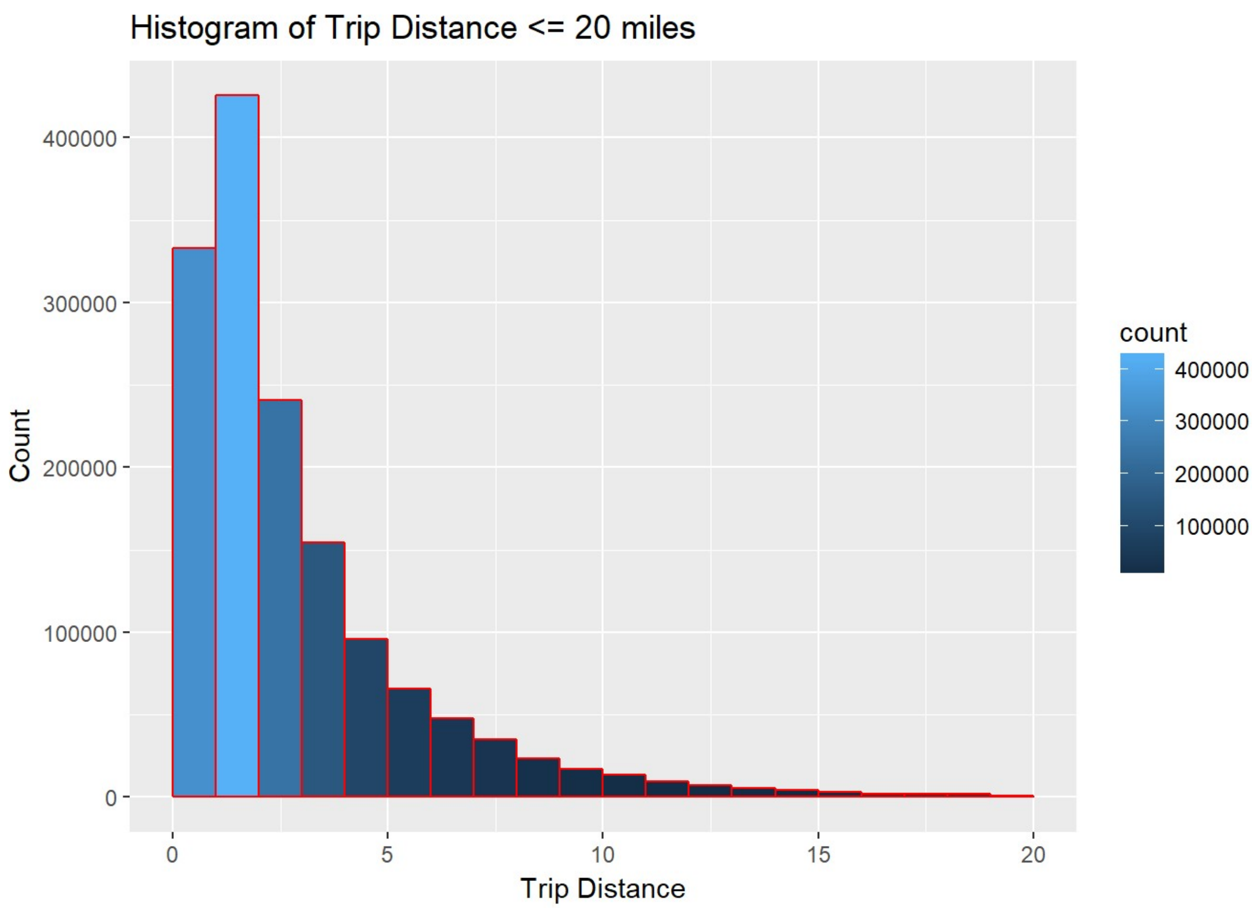
```
## [1] 102738
```

The boxplot result shows that there are roughly 100k outliers in trip distance which is very high (roughly 7% of our overall data). As the boxplot outliers are any value above the upper whisker (3rd quartile + 1.5 time Interquartile range), it is leading to a huge number of outlier cases. Restricting trip distance to 20 miles (as only 0.1% values are above this), we see a substantial improvement in the histogram of trip distance and its interpretability.

```
 # 1227 trips with more than 20 miles trip distance (comprising only 0.08% of overall
data)
sum(raw_data_2$Trip_distance>=20)
```

```
## [1] 3394
```

```
# Histogram with trip distance not more than 20 miles
ggplot(data=raw_data_2[raw_data_2$Trip_distance <=20,], aes(Trip_distance)) +
  geom_histogram(breaks=seq(0, 20, by = 1),
                 col="red",
                 aes(fill=..count..)) +
  labs(title="Histogram of Trip Distance <= 20 miles") +
  labs(x="Trip Distance", y="Count")
```



Question 3

1. Report mean and median trip distance grouped by hour of day
2. We'd like to get a rough sense of identifying trips that originate or terminate at one of the NYC area airports. Can you provide a count of how many transactions fit this criteria, the average fair, and any other interesting characteristics of these trips.

The mean and median trip distance grouped by hour of day is as follows :

```
ans_3a <- data.frame(
  "Time Of The Day" = c(0:23),
  Mean = round(aggregate(raw_data_2$Trip_distance,
                   list(format(raw_data_2$lpep_pickup_datetime,"%H")),
                   mean)[, 2], 2),
  Median = aggregate(raw_data_2$Trip_distance,
                     list(format(raw_data_2$lpep_pickup_datetime, "%H")),
                     median)[, 2]
  )

knitr::kable(ans_3a, align = 'c')
```

Time.Of.The.Day	Mean	Median
0	3.12	2.20
1	3.02	2.13
2	3.05	2.15
3	3.22	2.21
4	3.53	2.36
5	4.14	2.90
6	4.06	2.85
7	3.29	2.18
8	3.05	1.98
9	3.00	1.97
10	2.95	1.92
11	2.92	1.88
12	2.91	1.89
13	2.88	1.85
14	2.87	1.83
15	2.86	1.81
16	2.78	1.80
17	2.68	1.78
18	2.66	1.80
19	2.72	1.85

Time.Of.The.Day	Mean	Median
20	2.78	1.90
21	3.00	2.04
22	3.19	2.20
23	3.20	2.22

For the analysis to second part of the question, I will be using JFK Airport (represented as 2 in RateCodeID)

```
length(which(raw_data_2$RateCodeID==2))
```

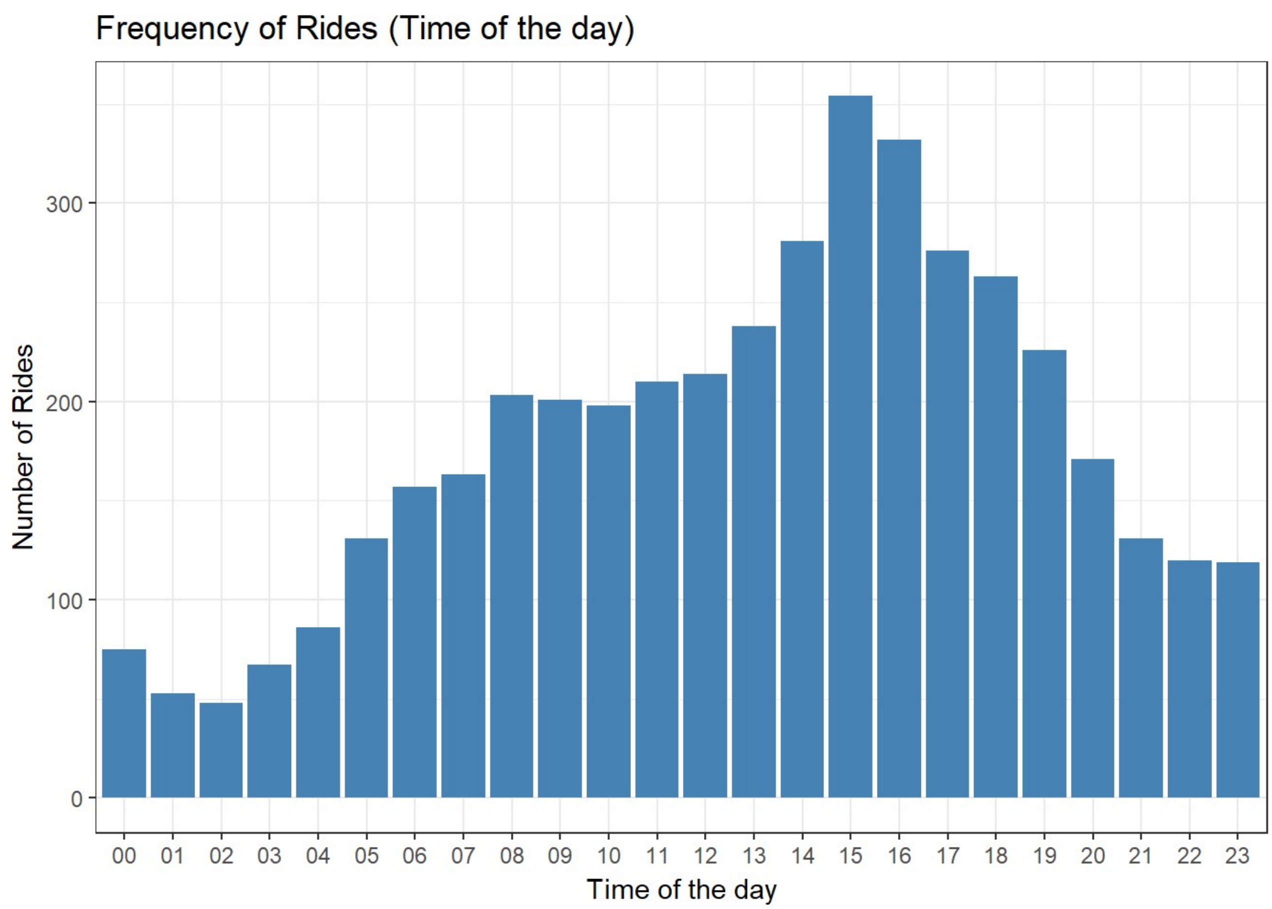
```
## [1] 4317
```

```
mean(raw_data_2[which(raw_data_2$RateCodeID==2), "Fare_amount"])
```

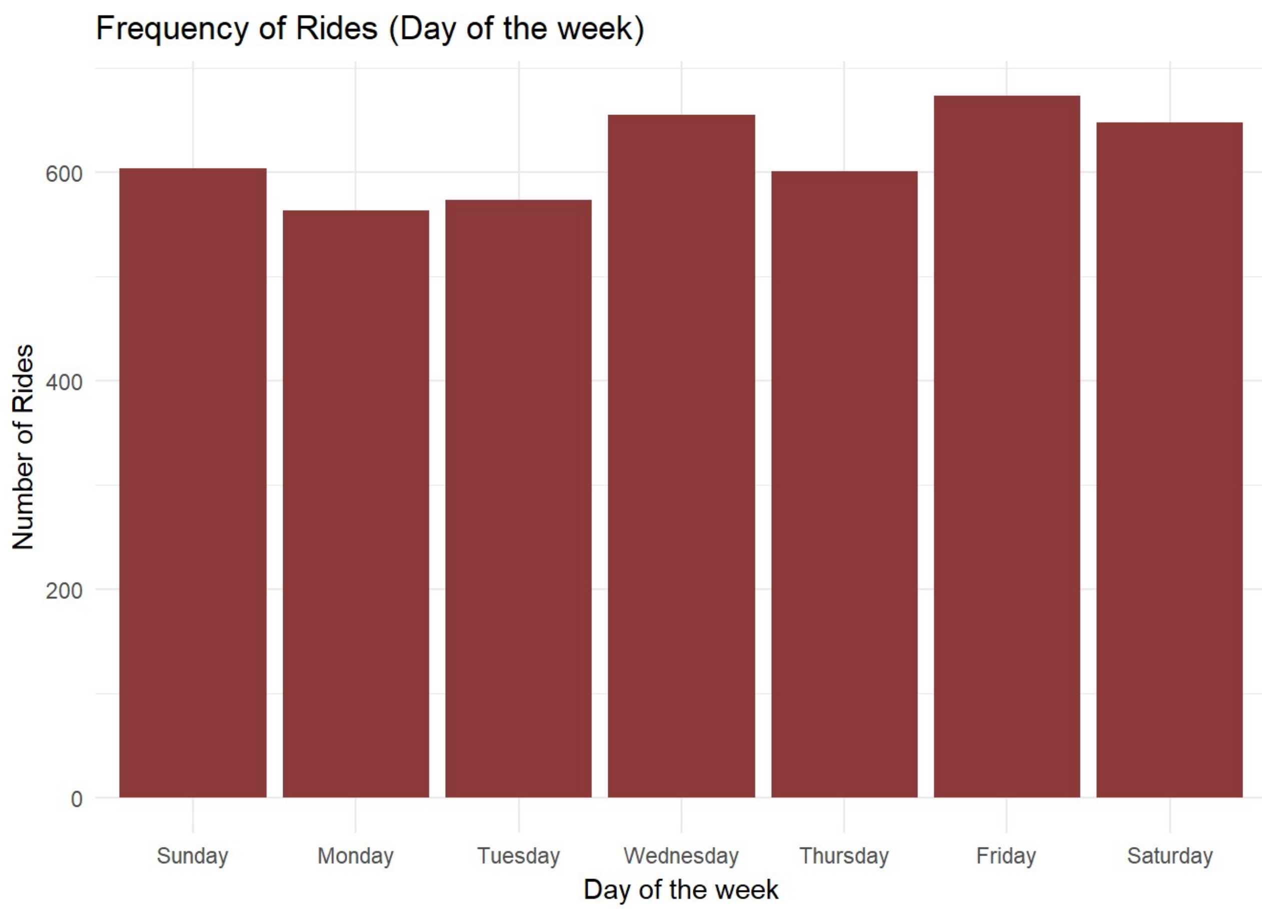
```
## [1] 51.78318
```

There are 4,317 total number of rides in the month of September 2015 with an average fare of 52 USD. Besides this we can also look at the distribution of rides across hours of the day.

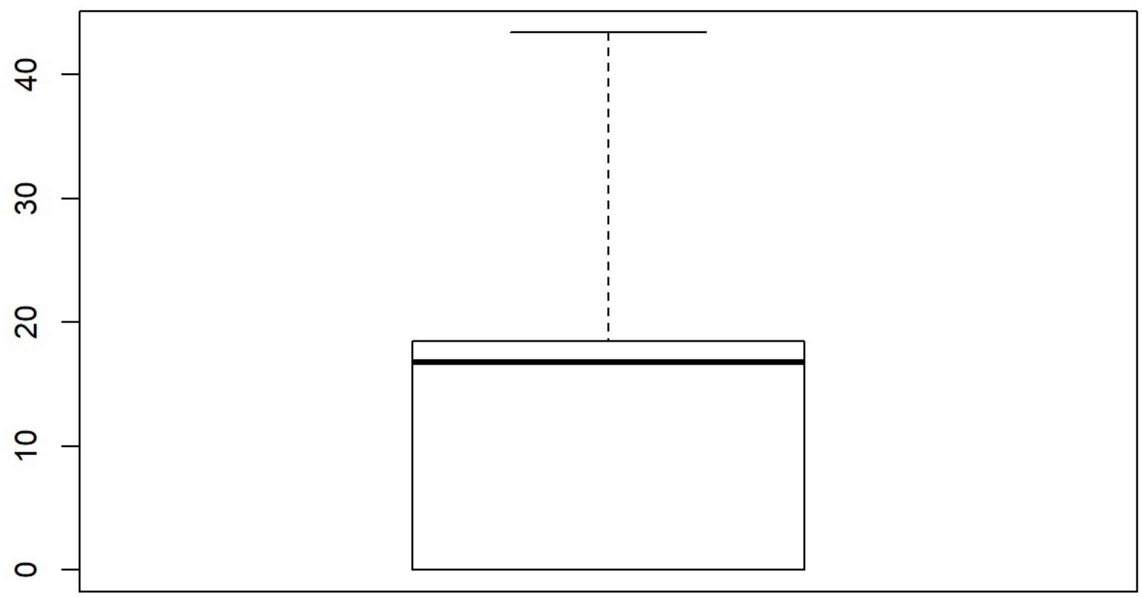
```
# Number of trips across time of the day
dist_hour <-
  aggregate(raw_data_2[which(raw_data_2$RateCodeID == 2), "RateCodeID"], list(format(r
aw_data_2[which(raw_data_2$RateCodeID ==
  2), "lpep_pickup_datetime"], "%H")), length)

# Number of trips across day of the week
  dist_week <-
  aggregate(raw_data_2[which(raw_data_2$RateCodeID == 2), "RateCodeID"], list(weekdays
(as.Date(raw_data_2[which(raw_data_2$RateCodeID ==
  2), "lpep_pickup_datetime"]))), length)

# Sorting weekdays (considering sunday as the first day) and assigning a shorter name
to wednesday
  dist_week$Group.1 <-factor(dist_week$Group.1,
                      levels = c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursda
y", "Friday",
                                 "Saturday"))

#Plotting the distribution of rides across the time of the day and day of the week
  ggplot(dist_hour, aes(Group.1, x)) +
  geom_bar(stat = "identity", fill="steelblue") +
  xlab("Time of the day") +
  ylab("Number of Rides") +
  ggtitle("Frequency of Rides (Time of the day)") +
  theme_bw()
```




```
ggplot(dist_week, aes(Group.1, x)) +
geom_bar(stat = "identity", fill="indianred4") +
xlab("Day of the week") +
ylab("Number of Rides") +
ggtitle("Frequency of Rides (Day of the week)") +
theme_minimal()
```




```
# Trip distance for green taxis on the JFK airport route
 boxplot(raw_data_2[which(raw_data_2$RateCodeID == 2), "Trip_distance"])
```




```
mean(raw_data_2[which(raw_data_2$RateCodeID == 2), "Trip_distance"])
```

```
## [1] 10.52321
```

```
median(raw_data_2[which(raw_data_2$RateCodeID == 2), "Trip_distance"])
```

```
## [1] 16.8
```

```
max(raw_data_2[which(raw_data_2$RateCodeID == 2), "Trip_distance"])
```

```
## [1] 43.37
```

We see interesting characteristics of trips to JFK. Maximum trips to JFK happen in the noon time from 2 PM to 5 PM and it peaks at 3 PM. There is also maximum number of trips on Friday/Saturday which makes intuitive sense as people like to travel during weekends. Only wednesday shows a considerable higher number of trips in weekdays which is surprising. The average distance travelled by Green Taxis to JFK airport is 10.5 miles whereas the maximum a ride has gone to JFK airport route is 43.37 miles.

Question 4

1. Build a derived variable for tip as a percentage of the total fare.
2. Build a predictive model for tip as a percentage of the total fare. Use as much of the data as you like (or all of it). We will validate a sample.

```
# Derived variable for tip as percentage of total fare
raw_data_2$percent_tip <- ifelse(raw_data_2$Total_amount == 0,
                                 0,
                                 raw_data_2$Tip_amount / raw_data_2$Total_amount)
```

We will be predicting the tip paid via credit card as cash tips are not captured (based on the data dictionary available on NYC TLC website)

To develop a model to predict percent_tip we will be performing the following steps:

1. Create few derived variable such as trip duration and pickup hour of the day
2. To develop the model, I will be using 5-fold cross validation technique (As Cross-validation produces much more stable test error estimates and will give me an idea as to how the model will perform on a sample) and utilize linear regression
3. We narrow down our list of variables to be leveraged to develop the model to the following 6 : RateCodeID, Passenger Count, trip distance, payment type, trip type, trip duration and hour of the day. My hypothesis behind including these variables are as follows :

a. RateCodeID : Tip paid might have a relation to the destination one's travelling to, like airport (JFK, Newark) which are baked into the ratecode. Also, the negotiated ratecode has a high chance of including a tip into the trip

b. Passenger_count : Higher passenger might lead to higher tendency of tipping due to reduced per head cost

c. Trip_distance : With low distance, passengers tend to tip less as they do not find it commensurate to the efforts

d. Payment_type : Credit card is convenient way of paying tip. Also when there's a dispute or no charge tips wouldn't be given

e. Trip_duration : A longer trip can lead to interaction between the driver and passenger and hence might lead to a generous tip

f. Hour_day : The tendency to tip might be less during rush hour as passengers are scrambling to reach their destination

4. The reason other variables were excluded are as follows :

a. VendorID/Store_and_fwd_flag : The technology provider of the cab or the server dynamics have little or no bearing on the tip one gives to a cab

b. lpep_pickup_datetime/lpep_dropoff_datetime : Date time have been incorporate by way of trip duration and hour of the day

c. Pickup_longitude/Pickup_latitude/Dropoff_longitude/Dropoff_ latitude : Latitude longitude of pickup and drop off location does not form a linear relationship with response variable

d. Payment related variables such as Fare_amount, Total_amount are leveraged to derive percent_tip and will be direclty collinear to response variable and hence I haven't included them

e. The pairwise scatterplot of percent_tip with extra, MTA_tax, Tolls_amount and improvement_surcharge shows no implicit trend and variation

```r
library("caret")
```

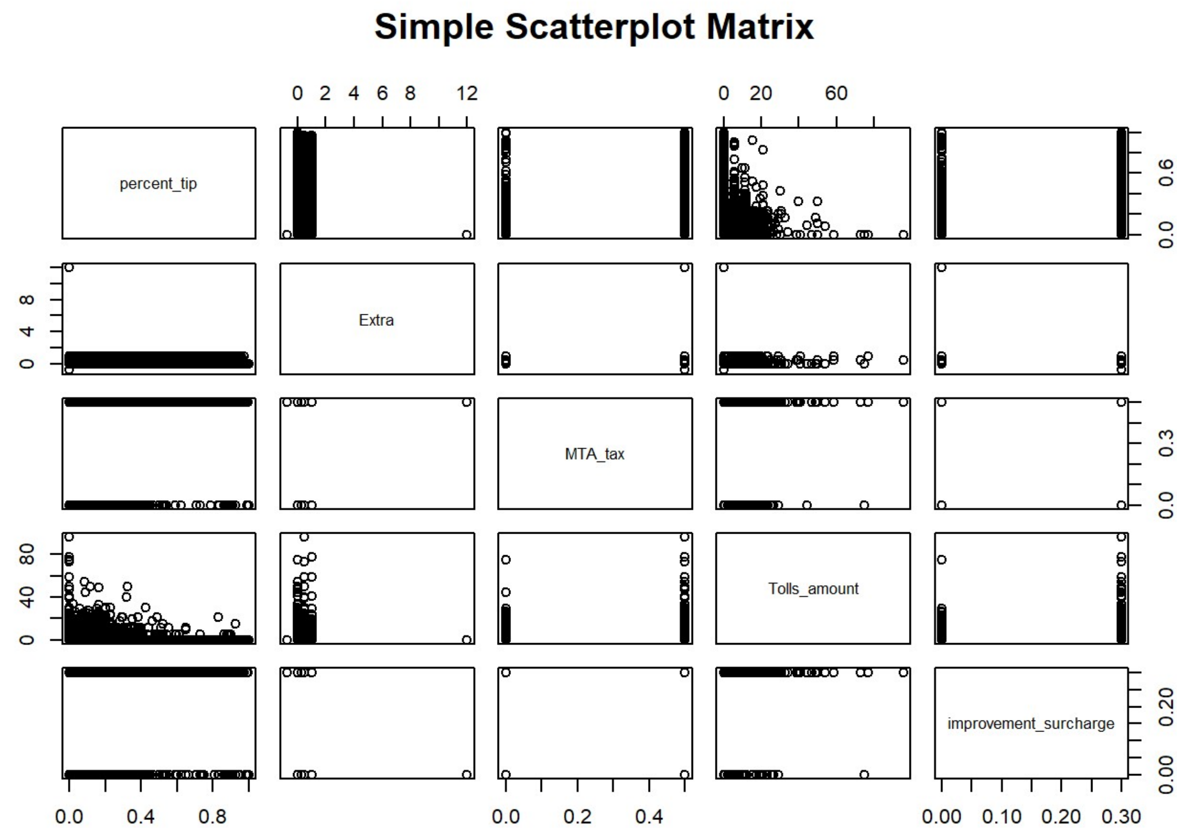
```
## Loading required package: lattice
```

```r
library("lubridate")
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```r
# Deriving a variable for trip duration and pickup hour of the day
raw_data_2$trip_duration <- as.numeric(difftime(raw_data_2$Lpep_dropoff_datetime, raw_
data_2$lpep_pickup_datetime,
                                                units = "mins"))
raw_data_2$hour_day <- hour(raw_data_2$lpep_pickup_datetime)

# Pairwise scatter plot
pairs(~percent_tip + Extra + MTA_tax + Tolls_amount + improvement_surcharge ,data=raw_
data_2,
   main="Simple Scatterplot Matrix")
```




```
# Correlation matrix between percent tip and numeric variables
cor_cols <- c(10:18, 22)
cor(raw_data_2[,21], raw_data_2[,cor_cols])
```

```
##      Passenger_count Trip_distance Fare_amount      Extra    MTA_tax
## [1,]     0.001663691    0.09492533  0.08580418 0.01386758 0.06835458
##      Tip_amount Tolls_amount improvement_surcharge Total_amount
## [1,]  0.7204813   0.04284545            0.06764871    0.2322158
##      trip_duration
## [1,]   -0.01130428
```

```
lm = train(
   percent_tip ~ RateCodeID + Passenger_count + Trip_distance + Payment_type + Trip_ty
pe + trip_duration + hour_day,
   data = raw_data_2,
   method = "lm",
   trControl = trainControl(method = "cv", number = 5)
 )
lm$results
```

```
##   intercept       RMSE  Rsquared        MAE       RMSESD  RsquaredSD
## 1      TRUE 0.05359996 0.6349772 0.02735852 0.0002258796 0.001897519
##           MAESD
## 1 0.00009311989
```

The test RMSE is observed to be 0.05359908 and we see that barring a few factor levels most of the variables turn out to be significant.

Question 5

Option A: Distributions
1. Build a derived variable representing the average speed over the course of a trip.
2. Can you perform a test to determine if the average trip speeds are materially the same in all weeks of September? If you decide they are not the same, can you form a hypothesis regarding why they differ?
3. Can you build up a hypothesis of average trip speed as a function of time of day?

1.

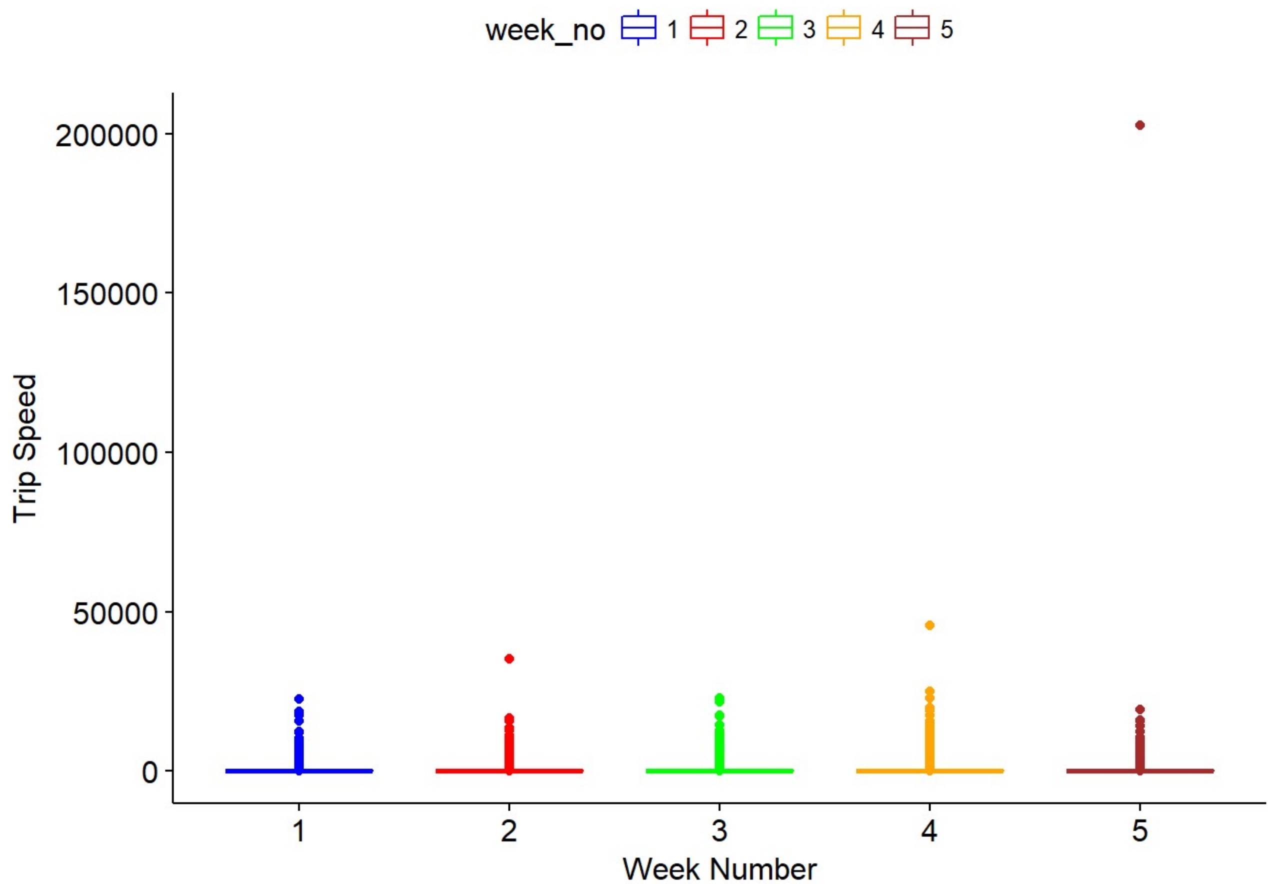
```
#Part a ()
# Calculating the average trip speed (in miles per hour)
raw_data_2$trip_speed <- ifelse(raw_data_2$trip_duration == 0,
                                0, 60*raw_data_2$Trip_distance/raw_data_2$trip_duratio
n)
```

2. I am going to use one-way anova test to test whether the average trip speeds are materially the same in all weeks of September. Anova test is suitable is used to determine whether there are any statistically significant differences between the means of three or more independent (unrelated) groups. I am assuming the means of the 5 weeks of september 2015 will be independent.
   Before that I am going to create a variable which signify different weeks of september and also use boxplot to see the distribution of trip speeds across 5 weeks of september.

```
# Given the date of the trip, calculating the week number of September it belongs to
raw_data_2$week_no <-  as.numeric(strftime(raw_data_2$lpep_pickup_datetime,format="%
W")) - 34

library("ggpubr")
```

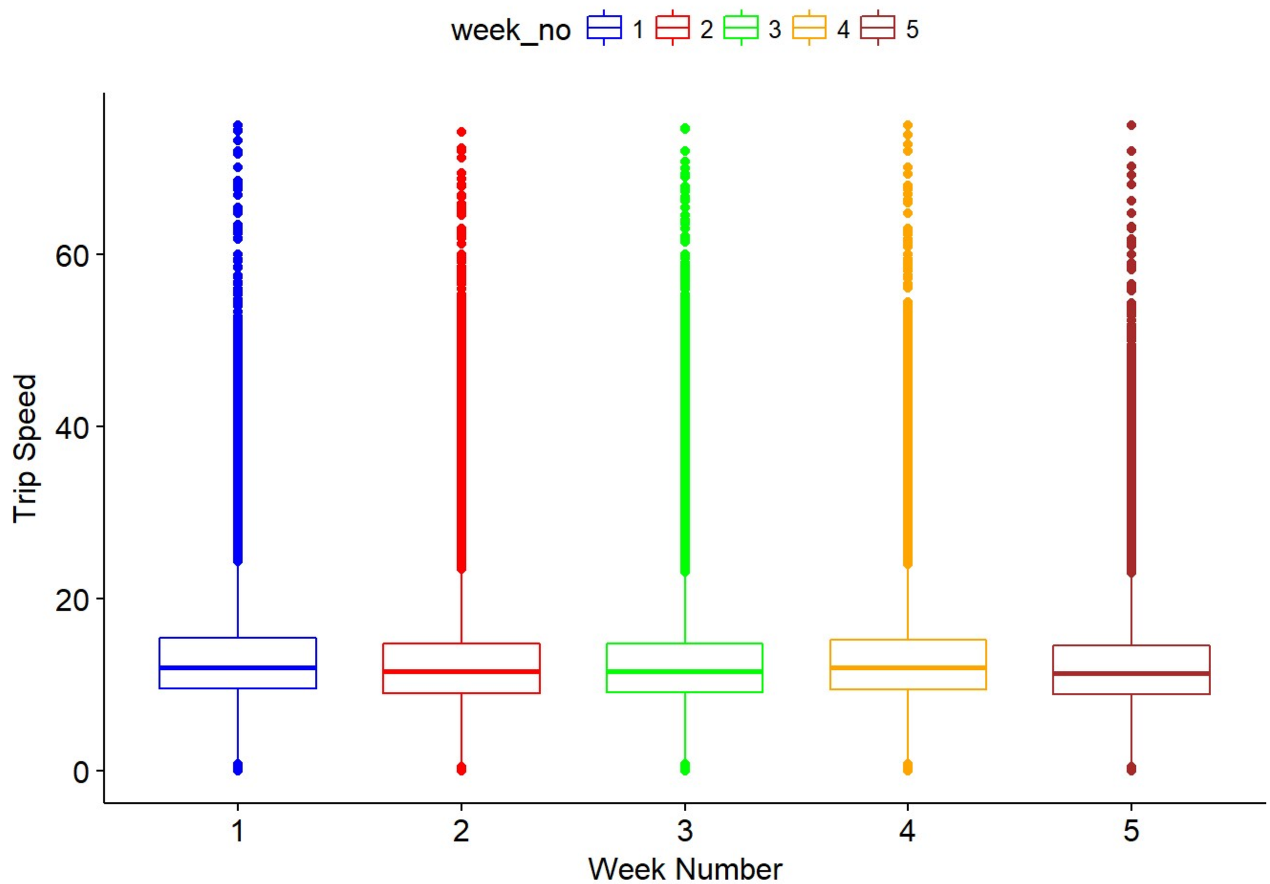
```
## Loading required package: magrittr
```

```
# Boxplot for different weeks
ggboxplot(raw_data_2, x = "week_no", y = "trip_speed",
          color = "week_no", palette = c("blue", "red", "green", "orange", "brown"),
          order = c(1:5),
          ylab = "Trip Speed", xlab = "Week Number")
```



The boxplot shows that we have unbelievably high trip speeds in the data (20,000mph which is not possible). Based on research online, I got to know the maximum permissible speed limit in New York City is 65 mph (link: http://www.safeny.ny.gov/spee-ndx.htm (http://www.safeny.ny.gov/spee-ndx.htm)) . I will use a margin of 10 and restrict my trip speeds to 75 mph which is a realistic expectation in most cases.

```
# Removing the speeds which are in excess of 75 miles per hour as they are comprise on
ly 0.2% of the overall data and is the maximum speed limit in new york city
speed_analysis <- raw_data_2[raw_data_2$trip_speed<=75, ]
ggboxplot(speed_analysis, x = "week_no", y = "trip_speed",
          color = "week_no", palette = c("blue", "red", "green", "orange", "brown"),
          order = c(1:5),
          ylab = "Trip Speed", xlab = "Week Number")
```



Now that we have treated trip speed of outliers, we can run the anova test with trip speed as response and weeks as my treatment groups.

```
res.aov <- aov(trip_speed ~ as.factor(week_no), data = speed_analysis)
summary(res.aov)
```

```
##                          Df   Sum Sq Mean Sq F value              Pr(>F)
## as.factor(week_no)        4   137875   34469     977 <0.0000000000000002
## Residuals           1489316 52543486      35
##
## as.factor(week_no) ***
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since p-value < 0.05, we reject the null hypothesis H0: the five means of each week are statistically equal and accept the alternate hypothesis that the average trip speeds are not materially the same in all weeks of September. One hypothesis regarding why average trip speeds differ across weeks is Seasonality : Start of september marks events like labor day and trigger events like these have a tendency to attrach large swathe of crowds thereby leading to traffic snarls and reduction in average speed.

3. Average trip speed could be low during the rush hour time (from 7 AM to 11 AM, from 3PM TO 9 PM) as humongous amount of people go about their day and travel leading to traffic snarls and thereby a reduction in average speed.

The analysis is not complete and there is always room for improvement. Following are a few ideas that I didn't get time to work on but could be explored in future:

1. Performing a more rigorous data exploration : Over the course of this analysis, I stumbled upon various issues which could have been investigated further such as trip distance being 0 (which means the trip did not even occur) and latitude/longitude having 0 values (which does not mean anything as it's a point in Atlantic Ocean off the west coast of Africa)
2. Checking assumptions of linear regression such as assessing normaliy (using QQ plot), non-constant variance (using residual plots) and correlated errors (using Durbin-Waston test)
3. Due to computational and time constraints, I did not get a chance to explore machine learning techniques such as Random Forests and Lasso regression which would have yielded a better prediction rate
4. Tukey HSD : In one-way ANOVA test, a significant p-value indicates that some of the group means are different, but we don't know which pairs of groups are different. we can compute Tukey HSD (Tukey Honest Significant Differences) for performing multiple pairwise-comparison between the means of groups.
5. Visualization : The internet is replete with cool and amazing visualization packages which could aid us in visualizing the different features of the data like plotly
6. Leveraging Latitude and Longitude values : In the entire analysis I did not make use of pickup latitude/longtiude and dropoff latitude/longitude information available in the data. I truly believe it has vast amount of information contained in it and it can be exploited. One such package to do so is ggmap.