DMPM Assignment 5

Name: Rushikesh Jyoti

Division: A

Roll no: 27

SRN: 201901139

Question: build a Linear Regression Model

Code

```
library(tidyverse)
library(ggplot2)
library(dplyr)
library(Metrics)
library(caret)
library(scales)
library(caTools)
library(corrplot)

dataset <- read.csv("AB_NYC_2019.csv")

summary(dataset)

# Find NA Values
```

```
print(colSums(is.na(dataset)))


# Fill 0 into NA
dataset$reviews_per_month[is.na(dataset$reviews_per_month) == TRUE] <- 0


data_new <- tidyr::separate(dataset, last_review, c("Year", "Month", "Day"), sep = "-")


data_new$Year[is.na(data_new$Year) == TRUE] <- 0
data_new$Month[is.na(data_new$Month) == TRUE] <- 0
data_new$Day[is.na(data_new$Day) == TRUE] <- 0


data_new$neighbourhood_group <- as.factor(data_new$neighbourhood_group)
data_new$room_type <- as.factor(data_new$room_type)
data_new$Year <- as.integer(data_new$Year)
data_new$Month <- as.integer(data_new$Month)
data_new$Day <- as.integer(data_new$Day)


head(data_new)
print(colSums(is.na(data_new)))
data_new <- na.omit(data_new)
summary(data_new)
```

```
# Correlation
correlation <- cor(data_new[, sapply(data_new, is.numeric)])
corrplot(cor(data_new[, sapply(data_new, is.numeric)]))

# Plotting the graphs
ggplot(data = data_new, mapping = aes(neighbourhood_group, fill =
room_type)) +
  geom_bar(colour = "Black", position = position_dodge())

price_roomtype <- data_new %>%
  group_by(neighbourhood_group, room_type) %>%
  summarise(Mean_Price = mean(price))

ggplot(price_roomtype, aes(x = reorder(neighbourhood_group, -
Mean_Price), y = Mean_Price, fill = room_type)) +
  geom_bar(stat = "identity", colour = "black", position = position_dodge())

ggplot(data_new, aes(y = price, x = minimum_nights, color =
neighbourhood_group)) +
  geom_jitter()

ggplot(data = data_new, mapping = aes(number_of_reviews, price)) +
  geom_point() +
```

```
  facet_wrap(data_new$room_type)


ggplot(data = data_new, mapping = aes(availability_365, price)) +
  geom_point()


ggplot(data = data_new, mapping = aes(neighbourhood_group,
availability_365)) +
  geom_boxplot()



# Building the model
model <- lm(price ~ host_id + neighbourhood_group + latitude + longitude +
room_type + minimum_nights + number_of_reviews +
  Year + calculated_host_listings_count + availability_365, data = data_new)


print(model)
print(summary(model))


pred1 <- predict(model)
resd1 <- residuals(model)


x <- cbind(data_new$price, pred1)
x <- data.matrix(x)
x <- rescale(x)
```

```
x <- as.data.frame(x)


mae <- MAE(x$V1, x$pred1)

mse <- mse(x$V1, x$pred1)

rmse <- RMSE(x$V1, x$pred1)

r2 <- R2(x$V1, x$pred1)


cat("\nMAE:", mae, "\n\nMSE:", mse, "\n\nRMSE:", rmse, "\n\nR-squared:",
r2, "\n\n")


ggplot(data_new, aes(y = pred1, x = data_new$price)) +

  geom_point() +

  geom_abline(intercept = 0, slope = 1, colour = "Red") +

  labs(y = "Predicted Values", x = "Actual Values", title = "Predicted vs. Actual
Values")
```

Output

Taking a look at dataset

```
> dataset ← read.csv("AB_NYC_2019.csv")
> head(dataset)
    id                                              name host_id   host_name
1 2539                Clean & quiet apt home by the park    2787        John
2 2595                             Skylit Midtown Castle    2845    Jennifer
3 3647               THE VILLAGE OF HARLEM....NEW YORK !    4632   Elisabeth
4 3831                   Cozy Entire Floor of Brownstone    4869 LisaRoxanne
5 5022 Entire Apt: Spacious Studio/Loft by central park    7192       Laura
6 5099         Large Cozy 1 BR Apartment In Midtown East    7322       Chris
  neighbourhood_group neighbourhood latitude longitude       room_type price
1            Brooklyn    Kensington 40.64749 -73.97237    Private room   149
2           Manhattan       Midtown 40.75362 -73.98377 Entire home/apt   225
3           Manhattan        Harlem 40.80902 -73.94190    Private room   150
4            Brooklyn  Clinton Hill 40.68514 -73.95976 Entire home/apt    89
5           Manhattan   East Harlem 40.79851 -73.94399 Entire home/apt    80
6           Manhattan   Murray Hill 40.74767 -73.97500 Entire home/apt   200
  minimum_nights number_of_reviews last_review reviews_per_month
1              1                 9  2018-10-19              0.21
2              1                45  2019-05-21              0.38
3              3                 0                            NA
4              1               270  2019-07-05              4.64
5             10                 9  2018-11-19              0.10
6              3                74  2019-06-22              0.59
  calculated_host_listings_count availability_365
1                              6              365
2                              2              355
3                              1              365
4                              1              194
5                              1                0
6                              1              129
```

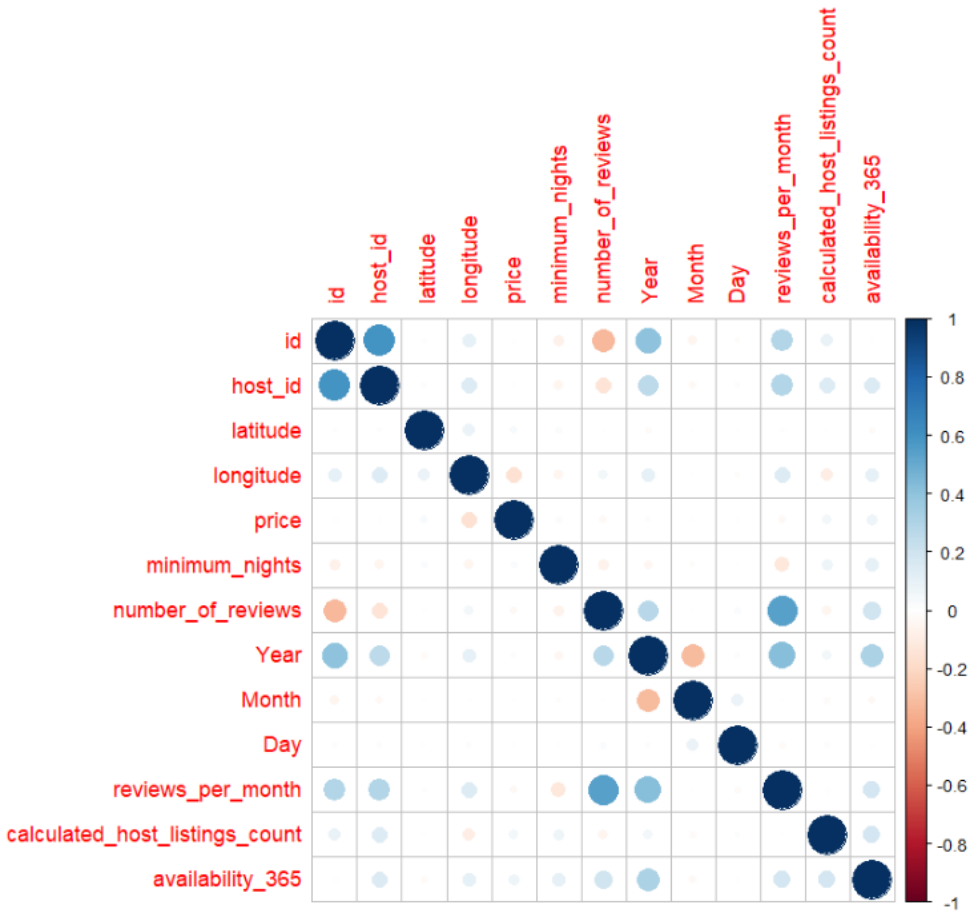
NA values in variables

```
> # Find NA Values
> print(colSums(is.na(dataset)))
                            id                          name
                             0                             0
                       host_id                     host_name
                             0                             0
          neighbourhood_group                 neighbourhood
                             0                             0
                      latitude                     longitude
                             0                             0
                     room_type                         price
                             0                             0
                minimum_nights             number_of_reviews
                             0                             0
                   last_review            reviews_per_month
                             0                         10052
calculated_host_listings_count              availability_365
                             0                             0
```

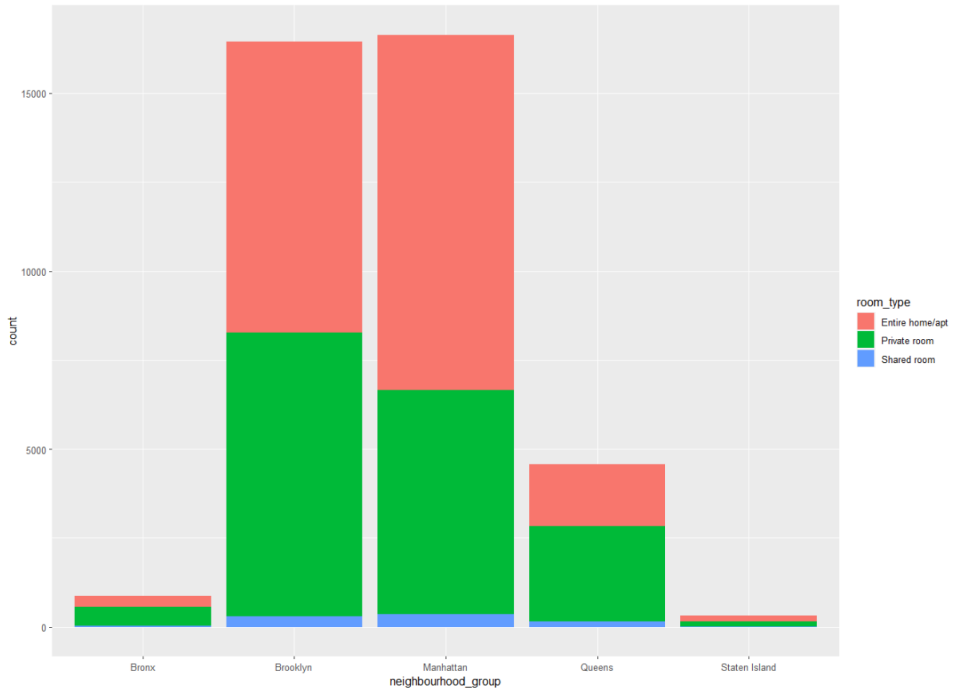
After omitting NA values

```
> print(colSums(is.na(data_new)))
                            id                           name
                             0                              0
                       host_id                      host_name
                             0                              0
          neighbourhood_group                  neighbourhood
                             0                              0
                      latitude                      longitude
                             0                              0
                     room_type                          price
                             0                              0
                minimum_nights                number_of_reviews
                             0                              0
                          Year                          Month
                         10052                              0
                           Day                reviews_per_month
                             0                              0
calculated_host_listings_count                availability_365
                             0                              0
```

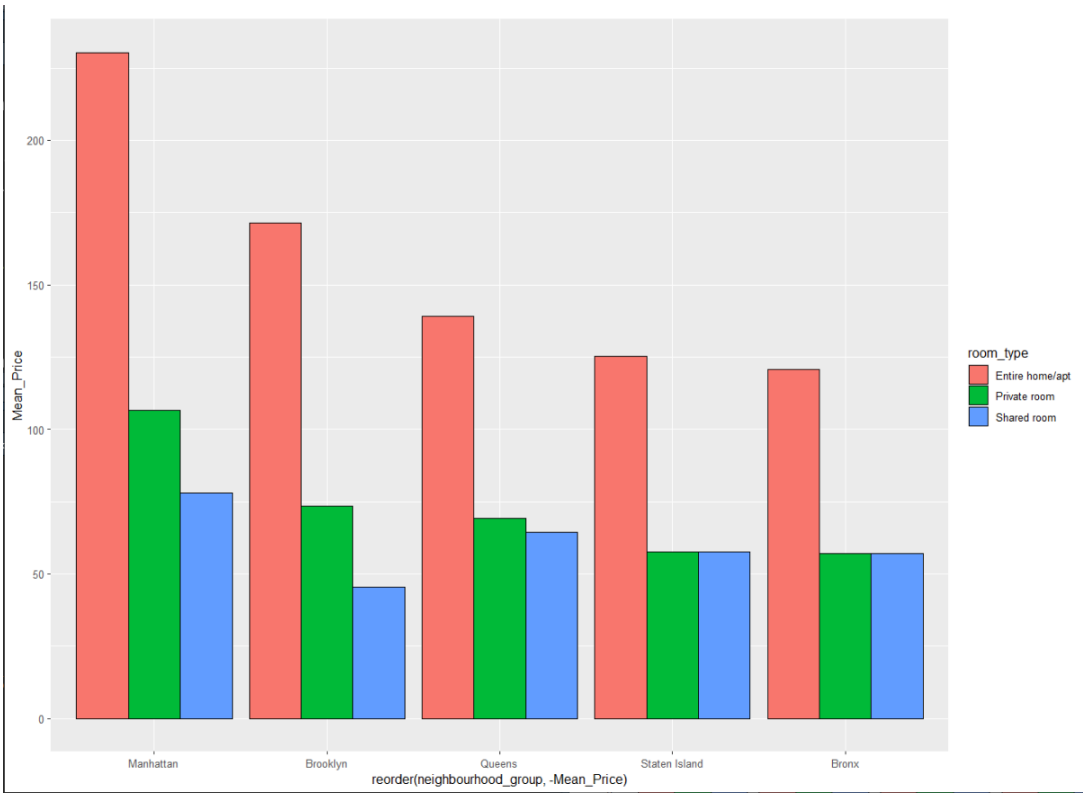
Correlation Matrix



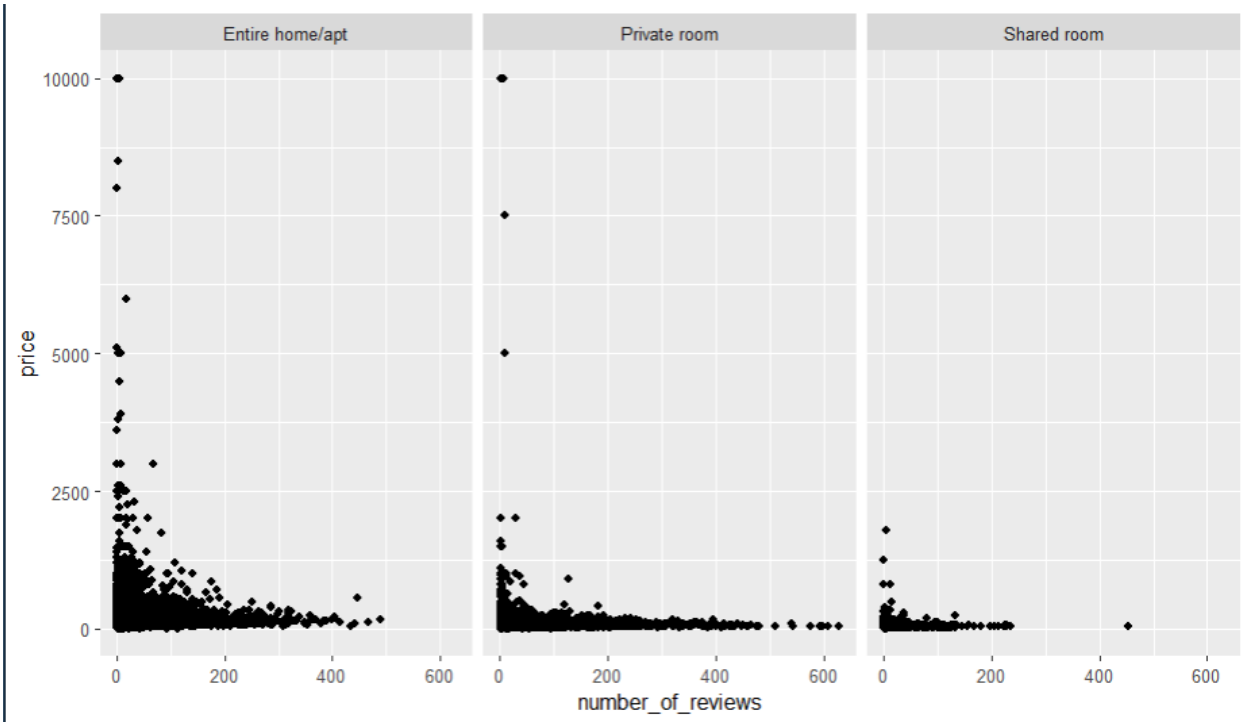
Count of each neighbourhood group



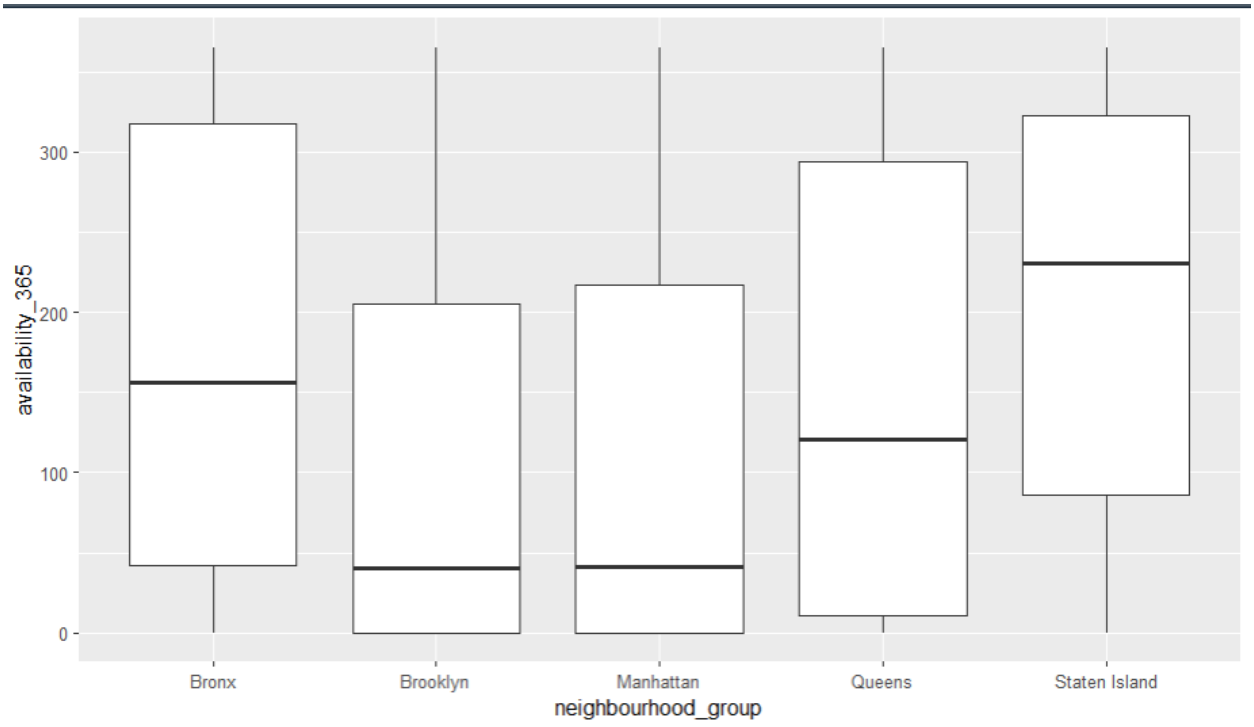
Mean price for each neighbourhood



Price of rooms based on reviews grouped by type of room



Box plot of room availability and neighbourhood



Building the model

```
> print(summary(model))

Call:
lm(formula = price ~ host_id + neighbourhood_group + latitude +
    longitude + room_type + minimum_nights + number_of_reviews +
    Year + calculated_host_listings_count + availability_365,
    data = data_new)

Residuals:
   Min     1Q Median     3Q    Max
-207.0  -53.3  -18.2   17.7 9959.8

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                    -1.630e+04  3.440e+03  -4.739 2.16e-06 ***
host_id                         7.821e-08  1.368e-08   5.718 1.09e-08 ***
neighbourhood_groupBrooklyn    -1.448e+01  7.972e+00  -1.816 0.069358 .
neighbourhood_groupManhattan    3.393e+01  7.205e+00   4.709 2.50e-06 ***
neighbourhood_groupQueens       6.149e+00  7.650e+00   0.804 0.421533
neighbourhood_groupStaten Island -1.422e+02  1.493e+01  -9.529  < 2e-16 ***
latitude                       -1.399e+02  2.849e+01  -4.911 9.08e-07 ***
longitude                      -4.552e+02  3.249e+01 -14.009  < 2e-16 ***
room_typePrivate room          -1.017e+02  1.958e+00 -51.937  < 2e-16 ***
room_typeShared room           -1.383e+02  6.530e+00 -21.180  < 2e-16 ***
minimum_nights                 -2.096e-01  5.501e-02  -3.810 0.000139 ***
number_of_reviews              -1.416e-01  2.116e-02  -6.690 2.26e-11 ***
Year                           -5.695e+00  8.979e-01  -6.343 2.28e-10 ***
calculated_host_listings_count -1.152e-01  3.739e-02  -3.082 0.002061 **
availability_365                1.807e-01  8.004e-03  22.574  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 184.9 on 38828 degrees of freedom
Multiple R-squared:  0.1194,    Adjusted R-squared:  0.119
F-statistic: 375.9 on 14 and 38828 DF,  p-value: < 2.2e-16
```

Metrics

```
> cat("\nMAE:", mae, "\nMSE:", mse, "\nRMSE:", rmse, "\nR-squared:", r2)

MAE: 0.006134365
MSE: 0.0003344305
RMSE: 0.01828744
R-squared: 0.1193614
```