Assignment: Project step 2

Name: Rapuru, Supraja

Date: 2021-08-07

Analysis of how AirBnB rentals prices affects the nearby housing rental prices in Chicago

```
## Load the readxl package
library(readxl)
## Load the plyr package
library(dplyr)
## Load the plyr package
library(plyr)
## Load the tidyverse package
library(tidyverse)
library(ggplot2)
library(readr)
## Set the working directory to the root of your DSC 520 directory
setwd('C:/Users/Supraja/Desktop/3. DSC520/Project')
#The data set contains information across Chicago City
airbnb_chicago_df <- readr::read_csv('airbnb-listings.csv')
summary(airbnb_chicago_df)
## Load the Affordable rental housing dataset
housing_df=read.csv("Affordable_Rental_Housing_Developments.csv")
glimpse(housing_df)
```

```
Rows: 488
Columns: 14
$ neighbourhood_cleansed <chr> "Edgewater", "Roseland", "Humboldt Park", "~
$ Community.Area.Number  <int> 77, 49, 23, 38, 42, 36, 36, 8, 24, 18, 14, ~
$ Property.Type          <chr> "Multifamily", "Senior", "Multifamily", "Mu~
$ Property.Name          <chr> "Winthrop Apts.", "Victory Center of Rosela~
$ Address                <chr> "6214 N. Winthrop Ave.", "10450 S. Michigan~
$ Zip.Code               <int> 60660, 60628, 60624, 60615, 60637, 60653, 6~
$ Phone.Number           <chr> "773-477-7070", "773-468-6400", "773-227-63~
$ Management.Company     <chr> "Hunter Properties", "Pathway Senior Living~
$ Units                  <int> 108, 81, 6, 8, 33, 148, 76, 7, 3, 3, 97, 22~
$ X.Coordinate           <dbl> 1167689, 1178829, 1155445, 1181237, 1182661~
$ Y.Coordinate           <dbl> 1941496, 1835494, 1903207, 1871959, 1864419~
$ Latitude               <dbl> 41.99502, 41.70389, 41.89020, 41.80390, 41.~
$ Longitude              <dbl> -87.65852, -87.62077, -87.70459, -87.61083,~
$ Location               <chr> "(41.9950154575665, -87.6585160357341)", "(~
```

Load the Average rent Chicago neighborhood dataset

avg_rent_df <- read_excel("Average_rent_Chicago_neighbourhood.xlsx")

glimpse(avg_rent_df)

```
Rows: 70
Columns: 2
$ neighbourhood_cleansed <chr> "Near North Side", "Lakeview", "West Town",~
$ `Average Rent`         <dbl> 2200, 1395, 1600, 2350, 1299, 1500, 1180, 1~
```

#Merge the airbnb df with rental housing df based on neighbourhood

final_1_df <- left_join(airbnb_chicago_df,housing_df,by="neighbourhood_cleansed")

glimpse(final_1_df)

head(final_1_df)

#Merge the above df with Average rent df based on neighbourhood

final_2_df <- inner_join(x=final_1_df,y=avg_rent_df,by=c("neighbourhood_cleansed"))

glimpse(final_2_df)

```
Rows: 78,313
Columns: 88
$ id                                  <dbl> 2384, 2384, 2384, 238~
$ listing_url                         <chr> "https://www.airbnb.c~
$ scrape_id                           <dbl> 2.02e+13, 2.02e+13, 2~
$ last_scraped                        <chr> "7/11/2021", "7/11/20~
$ name                                <chr> "Hyde Park - Walk to ~
$ description                         <chr> "If you have been ful~
$ neighborhood_overview               <chr> "The apartment is les~
$ picture_url                         <chr> "https://a0.muscache.~
$ host_id                             <dbl> 2613, 2613, 2613, 261~
$ host_url                            <chr> "https://www.airbnb.c~
$ host_name                           <chr> "Rebecca", "Rebecca",~
$ host_since                          <chr> "8/29/2008", "8/29/20~
$ host_location                       <chr> "Chicago, Illinois, U~
$ host_about                          <chr> "My 2 bdrm apartment ~
$ host_response_time                  <chr> "within an hour", "wi~
$ host_response_rate                  <chr> "100%", "100%", "100%~
$ host_acceptance_rate                <chr> "93%", "93%", "93%", ~
$ host_is_superhost                   <lgl> TRUE, TRUE, TRUE, TRU~
$ host_thumbnail_url                  <chr> "https://a0.muscache.~
$ host_picture_url                    <chr> "https://a0.muscache.~
$ host_neighbourhood                  <chr> "Hyde Park", "Hyde Pa~
$ host_listings_count                 <dbl> 1, 1, 1, 1, 1, 2, 2, ~
$ host_total_listings_count           <dbl> 1, 1, 1, 1, 1, 2, 2, ~
$ host_verifications                  <chr> "['email', 'phone', '~
$ host_has_profile_pic                <lgl> TRUE, TRUE, TRUE, TRU~
$ host_identity_verified              <lgl> TRUE, TRUE, TRUE, TRU~
```

```
$ neighbourhood                  <chr> "Chicago, Illinois, U~
$ neighbourhood_cleansed         <chr> "Hyde Park", "Hyde Pa~
$ neighbourhood_group_cleansed   <lgl> NA, NA, NA, NA, NA, N~
$ latitude                       <dbl> 41.78790, 41.78790, 4~
$ longitude                      <dbl> -87.58780, -87.58780,~
$ property_type                  <chr> "Private room in cond~
$ room_type                      <chr> "Private room", "Priv~
$ accommodates                   <dbl> 1, 1, 1, 1, 1, 2, 2, ~
$ bathrooms                      <lgl> NA, NA, NA, NA, NA, N~
$ bathrooms_text                 <chr> "1 shared bath", "1 s~
$ bedrooms                       <dbl> 1, 1, 1, 1, 1, 1, 1, ~
$ beds                           <dbl> 1, 1, 1, 1, 1, 1, 1, ~
$ amenities                      <chr> "[\"Hot water kettle\~
$ price                          <dbl> 85, 85, 85, 85, 85, 6~
$ minimum_nights                 <dbl> 1, 1, 1, 1, 1, 2, 2, ~
$ maximum_nights                 <dbl> 90, 90, 90, 90, 90, 6~
$ minimum_minimum_nights         <dbl> 2, 2, 2, 2, 2, 2, 2, ~
$ maximum_minimum_nights         <dbl> 4, 4, 4, 4, 4, 2, 2, ~
$ minimum_maximum_nights         <dbl> 90, 90, 90, 90, 90, 1~
$ maximum_maximum_nights         <dbl> 90, 90, 90, 90, 90, 1~
$ minimum_nights_avg_ntm         <dbl> 2, 2, 2, 2, 2, 2, 2, ~
$ maximum_nights_avg_ntm         <dbl> 90, 90, 90, 90, 90, 1~
$ calendar_updated               <lgl> NA, NA, NA, NA, NA, N~
$ has_availability               <lgl> TRUE, TRUE, TRUE, TRU~
$ availability_30                <dbl> 10, 10, 10, 10, 10, 1~
$ availability_60                <dbl> 33, 33, 33, 33, 33, 1~
$ availability_90                <dbl> 63, 63, 63, 63, 63, 3~
$ availability_365               <dbl> 338, 338, 338, 338, 3~
$ calendar_last_scraped          <chr> "7/11/2021", "7/11/20~
$ number_of_reviews              <dbl> 185, 185, 185, 185, 1~
$ number_of_reviews_ltm          <dbl> 7, 7, 7, 7, 7, 17, 17~
$ number_of_reviews_l30d         <dbl> 2, 2, 2, 2, 2, 0, 0, ~
$ first_review                   <chr> "4/30/2015", "4/30/20~
$ last_review                    <chr> "6/21/2021", "6/21/20~
```

```
$ review_scores_rating                          <dbl> 4.99, 4.99, 4.99, 4.9~
$ review_scores_accuracy                        <dbl> 4.98, 4.98, 4.98, 4.9~
$ review_scores_cleanliness                     <dbl> 4.99, 4.99, 4.99, 4.9~
$ review_scores_checkin                         <dbl> 4.98, 4.98, 4.98, 4.9~
$ review_scores_communication                   <dbl> 4.98, 4.98, 4.98, 4.9~
$ review_scores_location                        <dbl> 4.95, 4.95, 4.95, 4.9~
$ review_scores_value                           <dbl> 4.94, 4.94, 4.94, 4.9~
$ license                                       <chr> "R17000015609", "R170~
$ instant_bookable                              <lgl> FALSE, FALSE, FALSE, ~
$ calculated_host_listings_count                <dbl> 1, 1, 1, 1, 1, 1, 1, ~
$ calculated_host_listings_count_entire_homes   <dbl> 0, 0, 0, 0, 0, 1, 1, ~
$ calculated_host_listings_count_private_rooms  <dbl> 1, 1, 1, 1, 1, 0, 0, ~
$ calculated_host_listings_count_shared_rooms   <dbl> 0, 0, 0, 0, 0, 0, 0, ~
$ reviews_per_month                             <dbl> 2.45, 2.45, 2.45, 2.4~
$ Community.Area.Number                         <int> 41, 41, 41, 41, 41, 2~
$ Property.Type                                 <chr> "ARO", "ARO", "ARO", ~
$ Property.Name                                 <chr> "City Hyde Park", "Vu~
$ Address                                       <chr> "5105 S. Harper Ave."~
$ Zip.Code                                      <int> 60615, 60615, 60615, ~
$ Phone.Number                                  <chr> "773-548-5077", "773-~
$ Management.Company                            <chr> "Mac Properties", "Pe~
$ Units                                         <int> 36, 27, 2, 10, 36, 3,~
$ X.Coordinate                                  <dbl> 1187194, 1185905, 118~
$ Y.Coordinate                                  <dbl> 1871413, 1870431, 187~
$ Latitude                                      <dbl> 41.80226, 41.79960, 4~
$ Longitude                                     <dbl> -87.58900, -87.59376,~
$ Location                                      <chr> "(41.8022605698632, -~
```

#By looking at the data we can say that Airbnb data

 # 1. Variable id is just an identifier, and we can ignore it.

 # 2. We can factor the field room.type - Private room, Entire home/apt,Hotel room, Shared room

 # 3. We can drop the host.id and host.name, neighbourhood.group, name fields from the dataset

 # 4. We can drop fields like last.review, number.of.reviews, reviews.per.month, calculated.host.listings.count

#Average rent Chicago neighborhood data

 # 5. We can drop Property Name, Phone Number, Management Company, Units, Zip Codes from the dataset

#Average rent Chicago neighborhood data

 # 6. rename the Average Rent to Average_Rent

Apply above transformation to the dataframe – Select only required attributes

```
final_df <- subset(final_2_df, select = c("neighbourhood_cleansed",
                                          "latitude",
                                          "longitude",
```

"room_type",

"price","minimum_nights",

"availability_365",

"property_type",

"Zip.Code","X.Coordinate",

"Y.Coordinate",

"Latitude","Longitude",

"Average Rent"))

glimpse(final_df)

```
Rows: 78,313
Columns: 14
$ neighbourhood_cleansed <chr> "Hyde Park", "Hyde Park", "Hyde Park", "Hyd~
$ latitude               <dbl> 41.78790, 41.78790, 41.78790, 41.78790, 41.~
$ longitude              <dbl> -87.58780, -87.58780, -87.58780, -87.58780,~
$ room_type              <chr> "Private room", "Private room", "Private ro~
$ price                  <dbl> 85, 85, 85, 85, 85, 65, 65, 65, 65, 65, 65,~
$ minimum_nights         <dbl> 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2~
$ availability_365       <dbl> 338, 338, 338, 338, 338, 59, 59, 59, 59, 59~
$ property_type          <chr> "Private room in condominium", "Private roo~
$ Zip.Code               <int> 60615, 60615, 60615, 60615, 60615, 60642, 6~
$ X.Coordinate           <dbl> 1187194, 1185905, 1186745, 1185103, 1187148~
$ Y.Coordinate           <dbl> 1871413, 1870431, 1870452, 1869464, 1870068~
$ Latitude               <dbl> 41.80226, 41.79960, 41.79963, 41.79696, 41.~
$ Longitude              <dbl> -87.58900, -87.59376, -87.59068, -87.59674,~
$ `Average Rent`         <dbl> 1450, 1450, 1450, 1450, 1450, 1600, 1600, 1~
```

#Rename Average Rent to Average_Rent

colnames(final_df)[14] <- "Average_Rent"

Checking the summary of data set to gauge the value range of each numerical variable

summary(final_df)

```
 neighbourhood_cleansed    latitude       longitude       room_type
 Length:78313           Min.   :41.66   Min.   :-87.84   Length:78313
 Class :character       1st Qu.:41.88   1st Qu.:-87.69   Class :character
 Mode  :character       Median :41.90   Median :-87.67   Mode  :character
                        Mean   :41.90   Mean   :-87.67
                        3rd Qu.:41.92   3rd Qu.:-87.64
                        Max.   :42.02   Max.   :-87.54

     price         minimum_nights    availability_365 property_type
 Min.   :   0.0   Min.   :  1.000   Min.   :  0.0    Length:78313
 1st Qu.:  79.0   1st Qu.:  1.000   1st Qu.: 19.0    Class :character
 Median : 122.0   Median :  2.000   Median :134.0    Mode  :character
 Mean   : 182.1   Mean   :  8.943   Mean   :158.2
 3rd Qu.: 196.0   3rd Qu.:  4.000   3rd Qu.:302.0
 Max.   :9999.0   Max.   :365.000   Max.   :365.0

    Zip.Code      X.Coordinate      Y.Coordinate        Latitude
 Min.   :60601   Min.   :1127329   Min.   :1824810   Min.   :41.67
 1st Qu.:60612   1st Qu.:1158284   1st Qu.:1901307   1st Qu.:41.88
 Median :60622   Median :1165062   Median :1908210   Median :41.90
 Mean   :60655   Mean   :1164788   Mean   :1906400   Mean   :41.90
 3rd Qu.:60647   3rd Qu.:1170456   3rd Qu.:1912027   3rd Qu.:41.91
 Max.   :66007   Max.   :1199523   Max.   :1949531   Max.   :42.02
 NA's   :120     NA's   :135       NA's   :135       NA's   :135
   Longitude       Average_Rent
 Min.   :-87.81   Min.   : 675
 1st Qu.:-87.69   1st Qu.:1299
 Median :-87.67   Median :1600
 Mean   :-87.67   Mean   :1605
 3rd Qu.:-87.65   3rd Qu.:2200
 Max.   :-87.54   Max.   :2350
 NA's   :135
```

 # 7. Range of values prices are varies from 0 to 10000. It looks like there are outliers in the field.

 # 8. Range of values minimum_nights varies from 1 to 365. It looks like there are outliers in the field.

 # 9. Range of values for availability_365 varies from 0 to 365.

 # 10. Range of values for Average_Rent varies from 675 to 2350.

#Calculate the 30 days price for airbnb property.

final_df$airbnb_30_days_price=final_df$price * 30

summary(final_df)

```
neighbourhood_cleansed    latitude       longitude       room_type
Length:78313           Min.   :41.66   Min.   :-87.84   Length:78313
Class :character       1st Qu.:41.88   1st Qu.:-87.69   Class :character
Mode  :character       Median :41.90   Median :-87.67   Mode  :character
                       Mean   :41.90   Mean   :-87.67
                       3rd Qu.:41.92   3rd Qu.:-87.64
                       Max.   :42.02   Max.   :-87.54

    price         minimum_nights    availability_365 property_type
Min.   :   0.0   Min.   :  1.000   Min.   :  0.0    Length:78313
1st Qu.:  79.0   1st Qu.:  1.000   1st Qu.: 19.0    Class :character
Median : 122.0   Median :  2.000   Median :134.0    Mode  :character
Mean   : 182.1   Mean   :  8.943   Mean   :158.2
3rd Qu.: 196.0   3rd Qu.:  4.000   3rd Qu.:302.0
Max.   :9999.0   Max.   :365.000   Max.   :365.0

   Zip.Code      X.Coordinate      Y.Coordinate        Latitude
Min.   :60601   Min.   :1127329   Min.   :1824810   Min.   :41.67
1st Qu.:60612   1st Qu.:1158284   1st Qu.:1901307   1st Qu.:41.88
Median :60622   Median :1165062   Median :1908210   Median :41.90
Mean   :60655   Mean   :1164788   Mean   :1906400   Mean   :41.90
3rd Qu.:60647   3rd Qu.:1170456   3rd Qu.:1912027   3rd Qu.:41.91
Max.   :66007   Max.   :1199523   Max.   :1949531   Max.   :42.02
NA's   :120     NA's   :135       NA's   :135       NA's   :135
  Longitude       Average_Rent  airbnb_30_days_price
Min.   :-87.81   Min.   : 675   Min.   :     0
1st Qu.:-87.69   1st Qu.:1299   1st Qu.:  2370
Median :-87.67   Median :1600   Median :  3660
Mean   :-87.67   Mean   :1605   Mean   :  5463
3rd Qu.:-87.65   3rd Qu.:2200   3rd Qu.:  5880
Max.   :-87.54   Max.   :2350   Max.   :299970
NA's   :135
```

#Check missing values

apply(final_df, 2, function(x) any(is.na(x)))

```
neighbourhood_cleansed                latitude                 longitude
                 FALSE                    FALSE                     FALSE
             room_type                    price           minimum_nights
                 FALSE                    FALSE                     FALSE
      availability_365            property_type                  Zip.Code
                 FALSE                    FALSE                      TRUE
          X.Coordinate          Y.Coordinate                  Latitude
                  TRUE                     TRUE                      TRUE
             Longitude           Average_Rent     airbnb_30_days_price
                  TRUE                    FALSE                     FALSE
```

#It looks like there are some missing values for X.Coordinate ,Y.Coordinate, Latitude, Longitude, Zip.Code

2.What does the final data set look like?

glimpse(final_df)

```
Rows: 78,313
Columns: 15
$ neighbourhood_cleansed <chr> "Hyde Park", "Hyde Park", "Hyde Park", "Hyd~
$ latitude               <dbl> 41.78790, 41.78790, 41.78790, 41.78790, 41.~
$ longitude              <dbl> -87.58780, -87.58780, -87.58780, -87.58780,~
$ room_type              <chr> "Private room", "Private room", "Private ro~
$ price                  <dbl> 85, 85, 85, 85, 85, 65, 65, 65, 65, 65, 65,~
$ minimum_nights         <dbl> 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2~
$ availability_365       <dbl> 338, 338, 338, 338, 338, 59, 59, 59, 59, 59~
$ property_type          <chr> "Private room in condominium", "Private roo~
$ Zip.Code               <int> 60615, 60615, 60615, 60615, 60615, 60642, 6~
$ X.Coordinate           <dbl> 1187194, 1185905, 1186745, 1185103, 1187148~
$ Y.Coordinate           <dbl> 1871413, 1870431, 1870452, 1869464, 1870068~
$ Latitude               <dbl> 41.80226, 41.79960, 41.79963, 41.79696, 41.~
$ Longitude              <dbl> -87.58900, -87.59376, -87.59068, -87.59674,~
$ Average_Rent           <dbl> 1450, 1450, 1450, 1450, 1450, 1600, 1600, 1~
$ airbnb_30_days_price   <dbl> 2550, 2550, 2550, 2550, 2550, 1950, 1950, 1~
```

3. Questions for future steps.

 # a) Need to learn how to visualize more than two variables.

 # b) Need to learn application of variable scaling and techniques.

 # c) Need to learn how lm() function takes care of variable scaling.

 # d) Need to learn correlation between different variables.

4. What information is not self-evident?

 # To uncover new information in the data that is not self-evident -

 # 1. visualize data to uncover patterns and trends

 # 2. correlation among variables

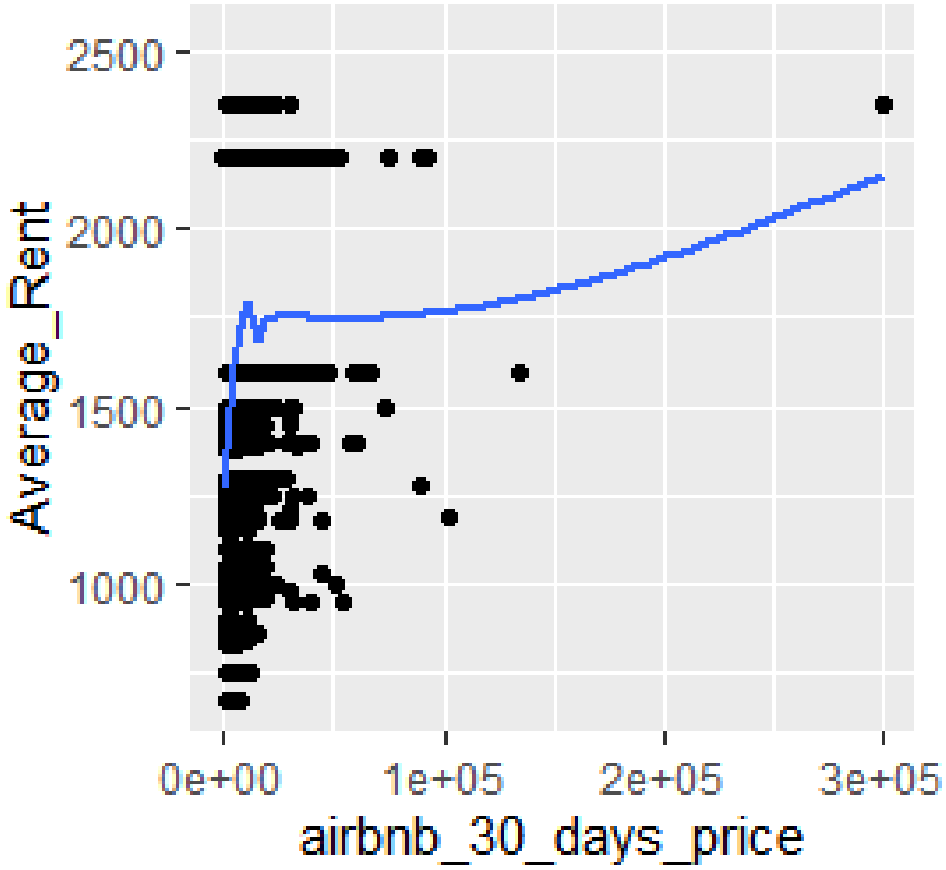
 # 3. Check data distribution of variables

 # 4. detect outliers and influencial cases

 # 5.What are different ways you could look at this data?

Checking relation between airbnb_30_days_price and Average_Rent using ggplot()

```
library(ggplot2)

ggplot(data = final_df, aes(x = airbnb_30_days_price, y = Average_Rent)) +

  geom_point() + geom_smooth(fill=NA)
```

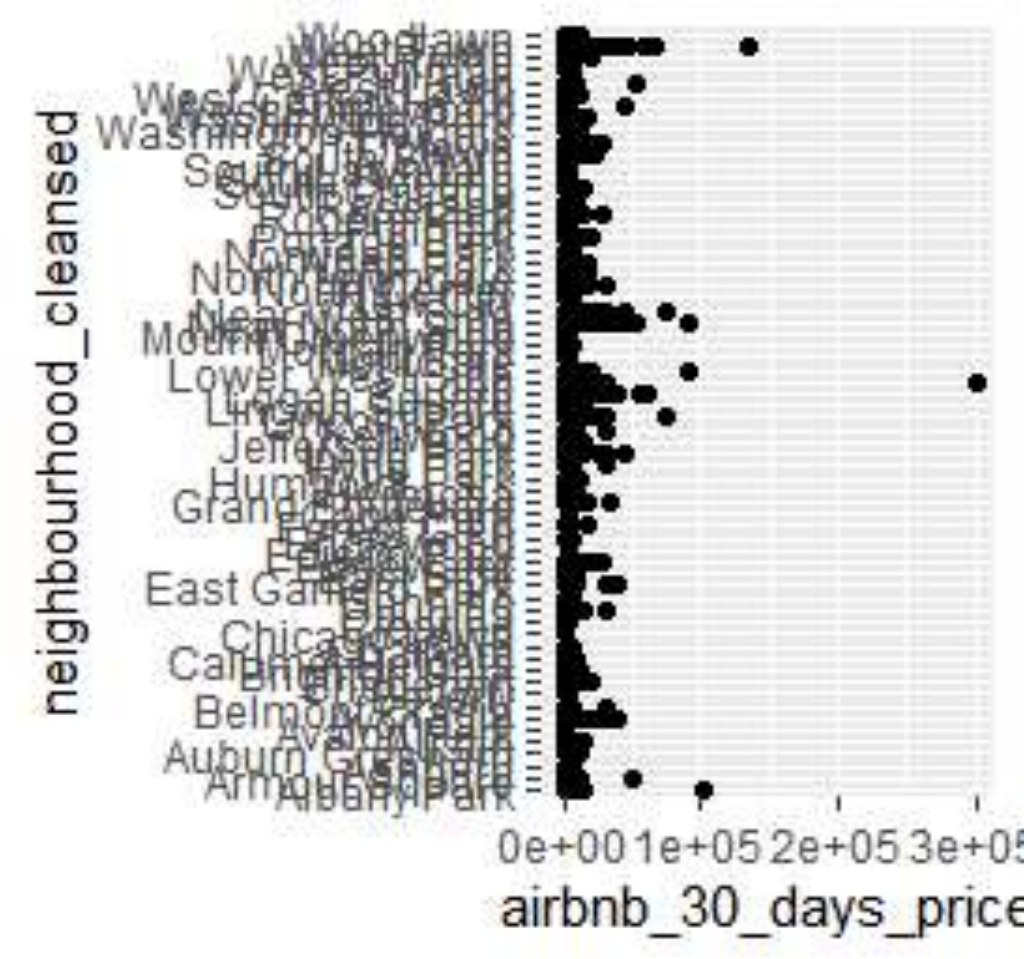


Checking relation between neighbourhood_cleansed and Average_Rent using ggplot()

```
library(ggplot2)

ggplot(data = final_df, aes(y = neighbourhood_cleansed, x = Average_Rent)) +

  geom_point() + geom_smooth(fill=NA)
```

Checking relation between neighbourhood_cleansed and airbnb_30_days_price using ggplot()

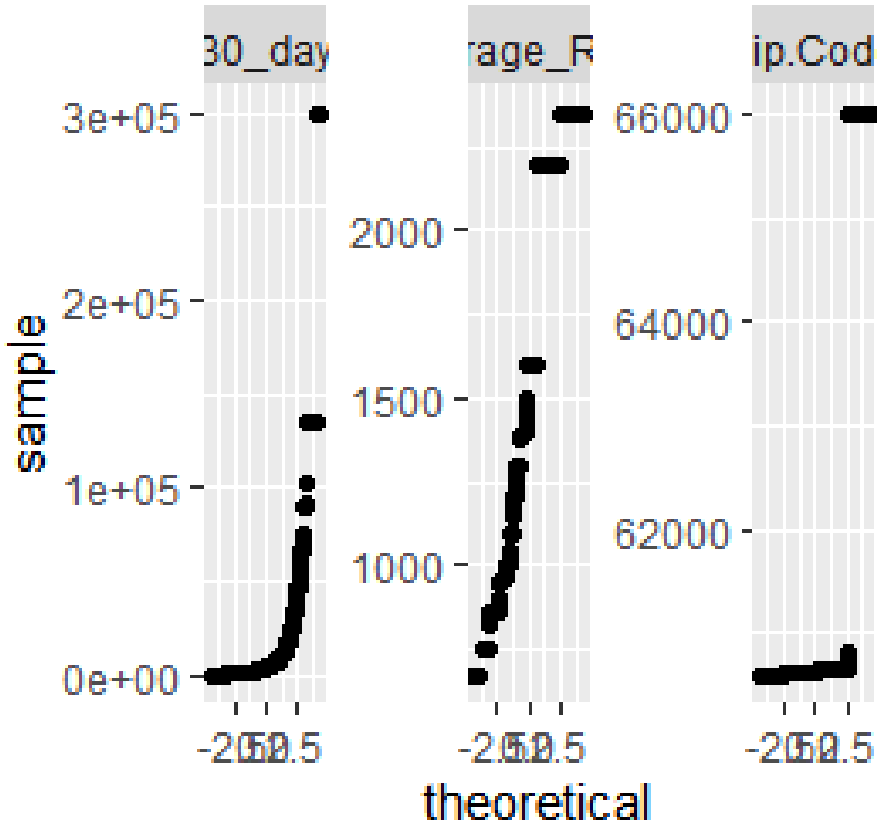
```
library(ggplot2)
ggplot(data = final_df, aes(y = neighbourhood_cleansed, x = airbnb_30_days_price)) +
  geom_point() + geom_smooth(fill=NA)
```



#We can see that there is relationship between neighbourhood and prices

Checking if data distribution of numeric variables is normal combining pipe operator between dplyr transformation and ggplot

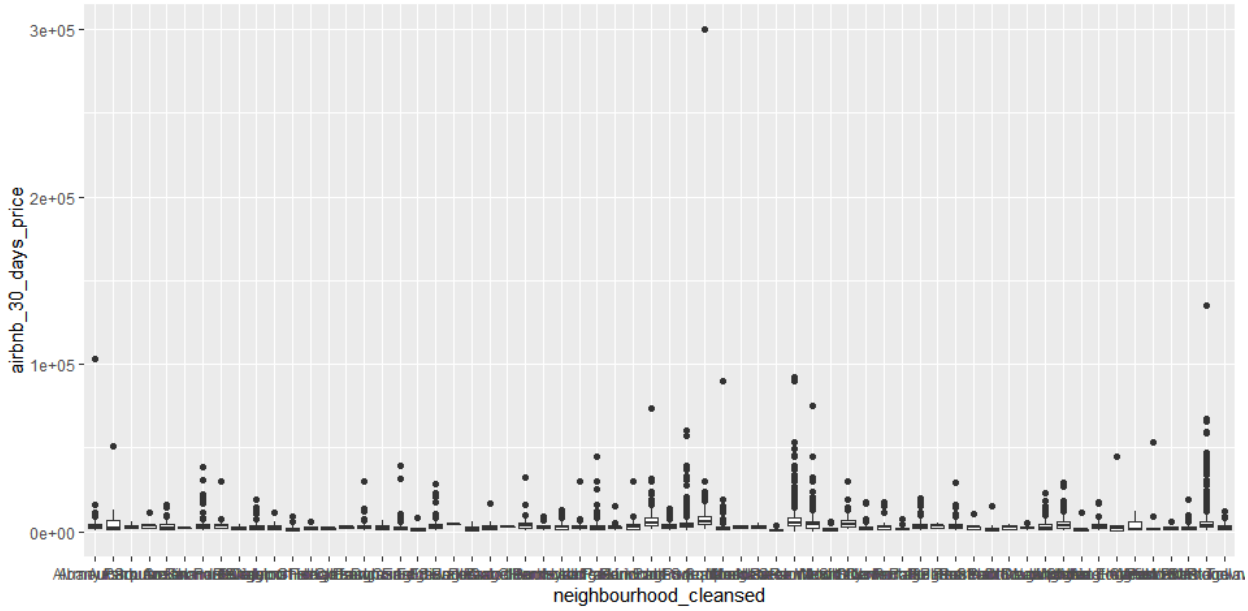
```
final_df %>% select(airbnb_30_days_price, Zip.Code, Average_Rent) %>%
  gather() %>%
  ggplot(., aes(sample = value)) +
  stat_qq() +
  facet_wrap(vars(key), scales ='free_y')
```



#None of the variables looks normally distributed

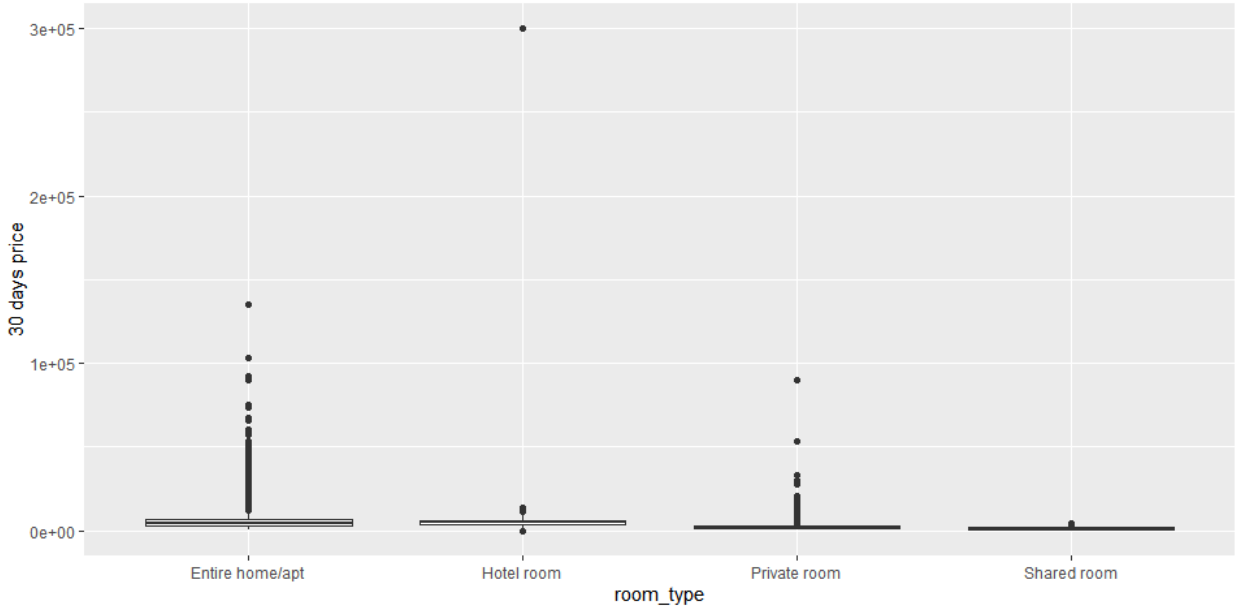
ggplot(data = final_df, aes(x = neighbourhood_cleansed , y = airbnb_30_days_price)) +

geom_boxplot() + ylab("airbnb_30_days_price")



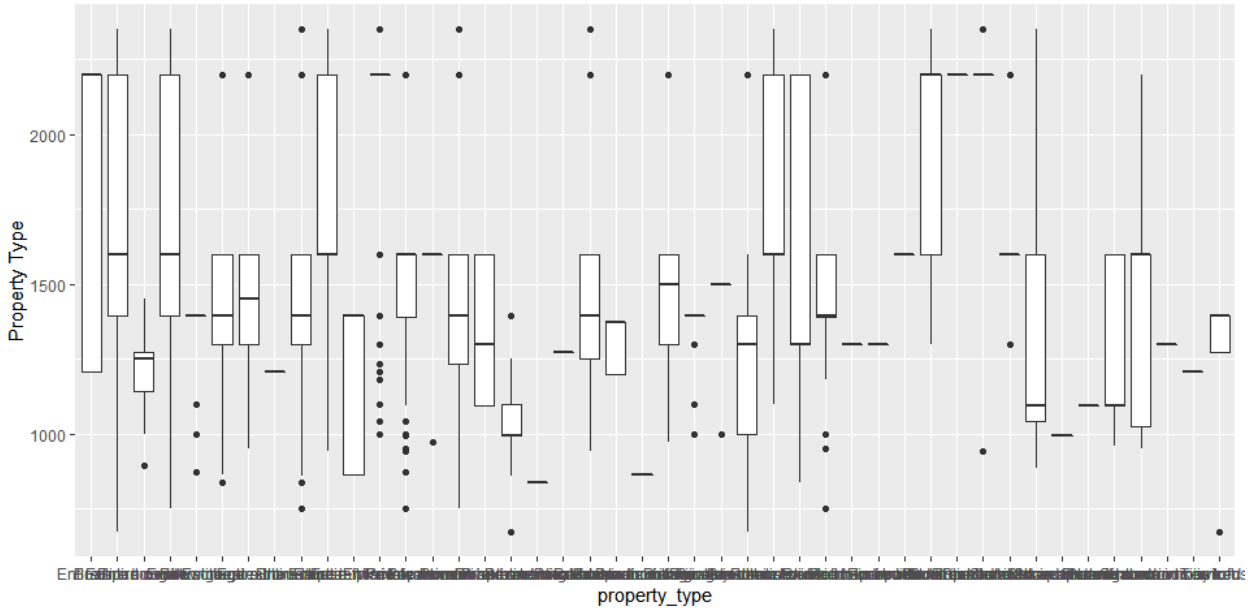
 # We can see that there are so many outliers for many neighbourhoods thus data is not normally distributed

```
ggplot(data = final_df, aes(x = room_type , y = airbnb_30_days_price)) +
  geom_boxplot() + ylab("30 days price")
```



We can see that there are so many outliers for room_type thus data is not normally distributed

```
ggplot(data = final_df, aes(x = property_type , y = Average_Rent)) +
  geom_boxplot() + ylab("Average Rent")
```



 # We can see that there are so many outliers for Property_Type thus data is not normally distributed

6.How do you plan to slice and dice the data?

unique(final_df[c("Zip.Code")])

```
# A tibble: 49 x 1
   Zip.Code
      <int>
 1    60615
 2    60642
 3    60647
 4    60622
 5    60654
 6    60614
 7    60610
 8    60612
 9    60640
10    60613
# ... with 39 more rows
```

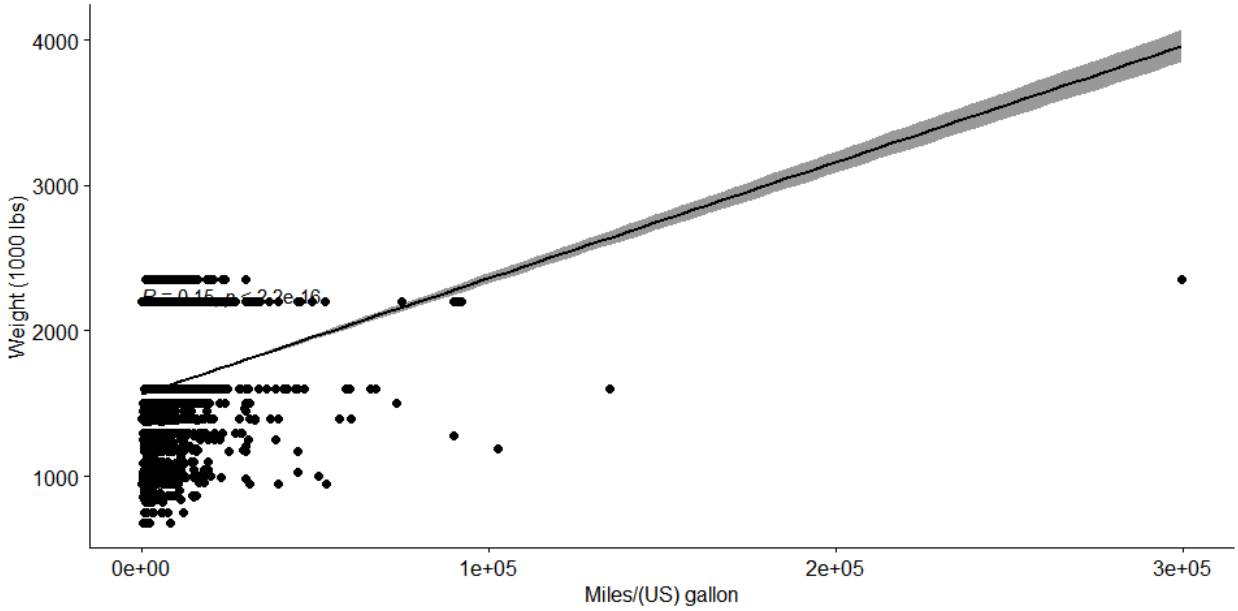
unique(final_df[c("neighbourhood_cleansed")])

```
# A tibble: 64 x 1
   neighbourhood_cleansed
   <chr>
 1 Hyde Park
 2 West Town
 3 Lincoln Park
 4 Near North Side
 5 Logan Square
 6 Uptown
 7 North Center
 8 Albany Park
 9 Pullman
10 West Ridge
# ... with 54 more rows
```

#I think need to slice the datasets by zip codes or neighbourhood to analyze the data in more granular level

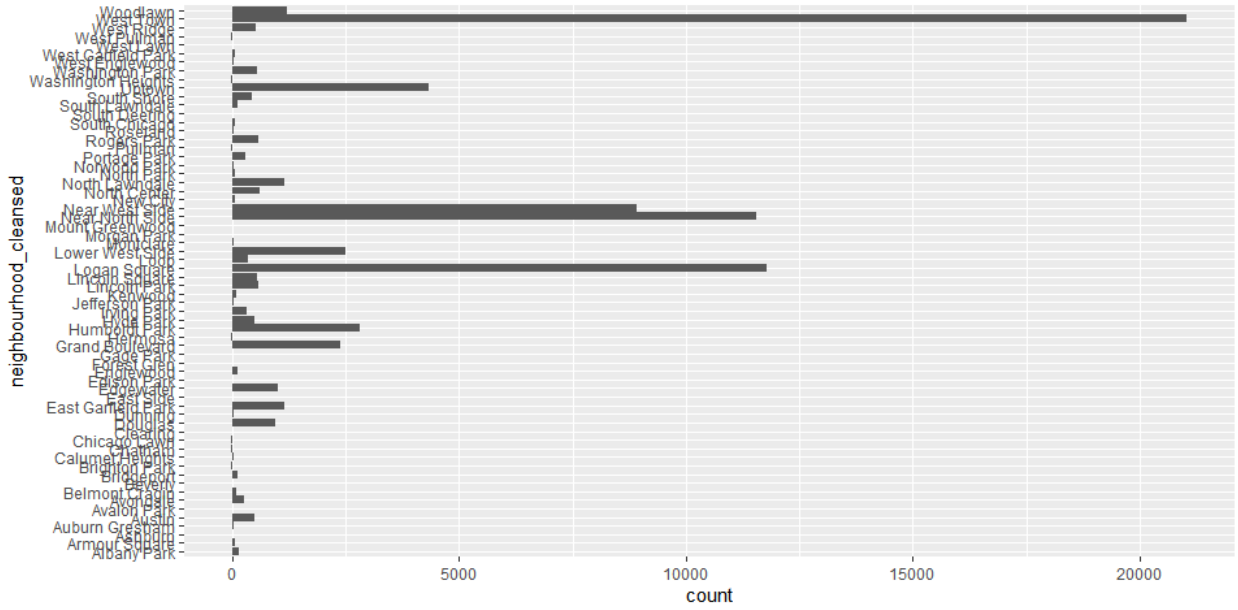
7.How could you summarize your data to answer key questions?

```
library("ggpubr")

ggscatter(final_df, x = "airbnb_30_days_price", y = "Average_Rent",

      add = "reg.line", conf.int = TRUE,

      cor.coef = TRUE, cor.method = "pearson",

      xlab = "Miles/(US) gallon", ylab = "Weight (1000 lbs)")
```



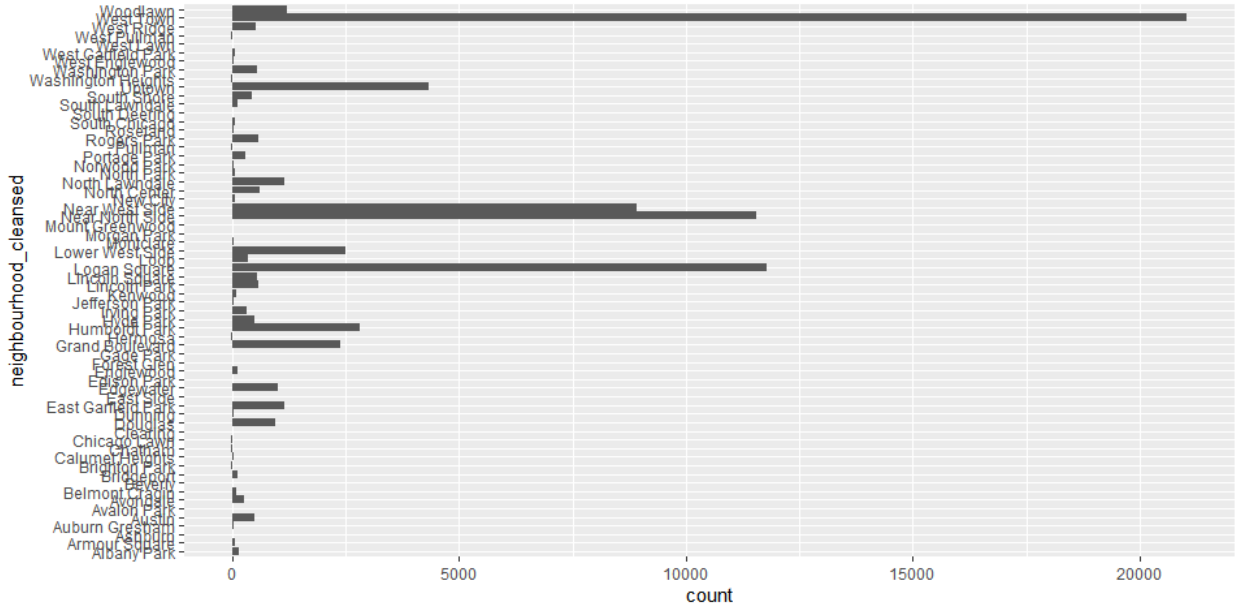
#a) What are the Airbnb rental prices for different areas in Chicago?

ggplot(data=final_df,aes(y=neighbourhood_cleansed)) + geom_histogram(stat = "count")



ggplot(aes(y=neighbourhood_cleansed,x=airbnb_30_days_price),data=final_df)+

geom_point()



 # From graph it looks like "West town" have major number of airbnb properties. Also the prices of "West town" properties are high for airbnb rental.

b) What is the correlation between the Airbnb rental prices and Chicago neighborhood rent prices?

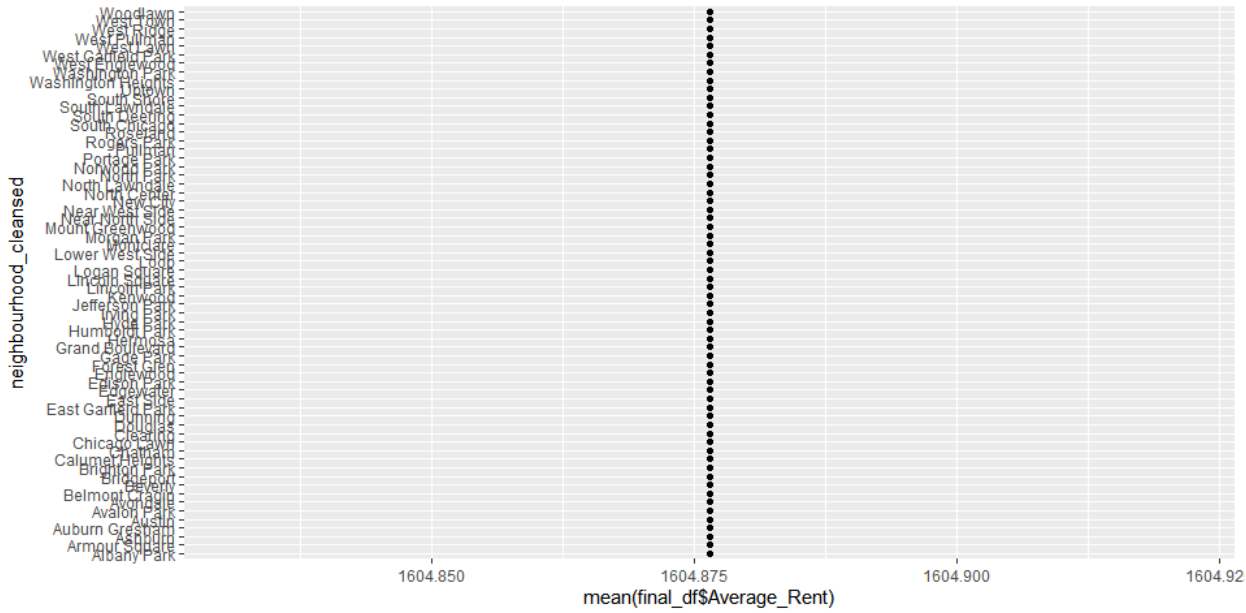
```
cor(final_df$airbnb_30_days_price,final_df$Average_Rent)
```

```
[1] 0.1470344
```

 # It is evident from the plots that there is positive correlation between airbnb prices and average rent

c)What are the average rent prices by the neighborhood?

```
ggplot(aes(y=neighbourhood_cleansed,x=mean(final_df$Average_Rent)),data=final_df)+
  geom_point()
```

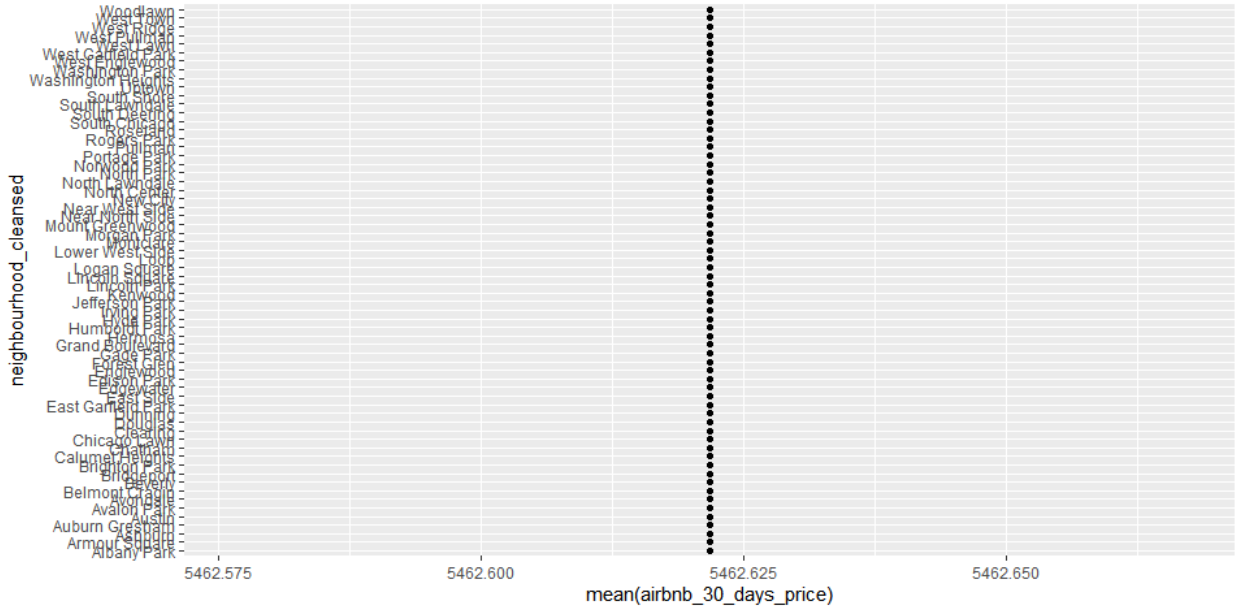


#The average rent price is ~1600 per month

d)What are the average rent prices for Airbnb by the neighborhood?

ggplot(aes(y=neighbourhood_cleansed,x=mean(airbnb_30_days_price)),data=final_df)+

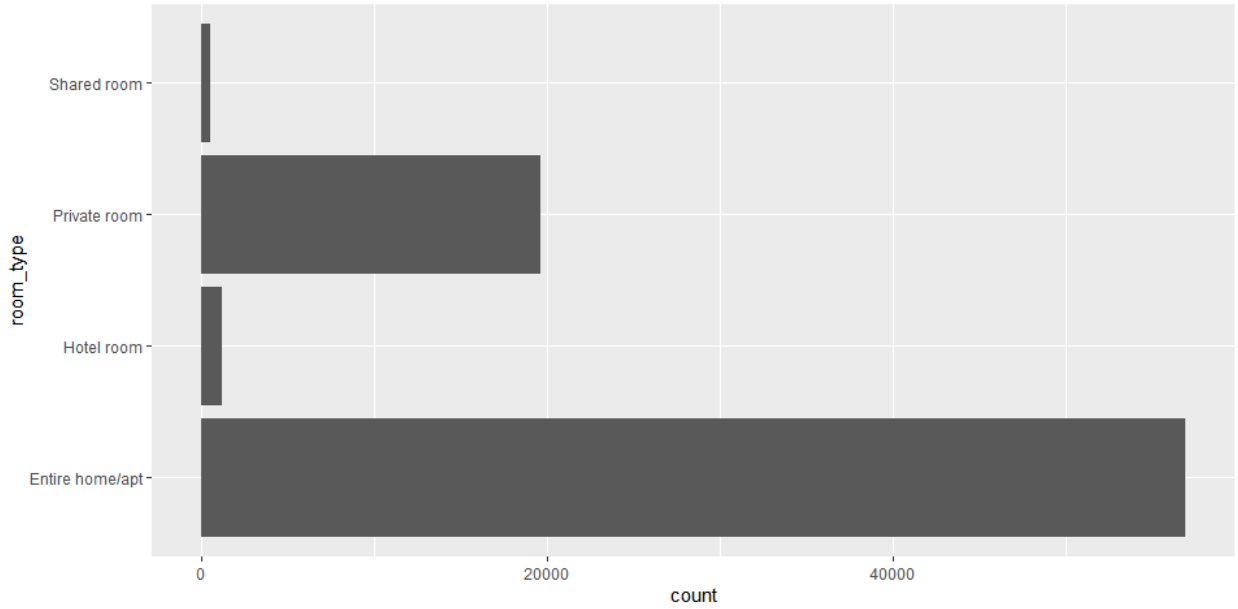
geom_point()



#The average airbnb price is ~ 5400 per month

e) What type of houses are most rented on Airbnb?

ggplot(data=final_df,aes(y=room_type)) + geom_histogram(stat ="count")



#It looks like Entire home/apt are most rented on Airbnb

f)What is the monthly rent from the Airbnb properties?

df1 <-final_df%>%select(neighbourhood_cleansed, airbnb_30_days_price, Average_Rent)

df1 %>% group_by(neighbourhood_cleansed) %>% summarize(mean_airbnb_30_days_price = mean(airbnb_30_days_price))

```
  mean_airbnb_30_days_price
1                  5462.622
```

#Airbnb monthly average rent is 5462.622

 # 9)Do you plan on incorporating any machine learning techniques to answer your research questions? Explain.

 # performing multiple linear regression

 # splitting the data into training and test set

library(caTools)

mymodel_1 <-lm(airbnb_30_days_price ~ neighbourhood_cleansed,data = final_df)

summary(mymodel_1)

```
Call:
lm(formula = airbnb_30_days_price ~ neighbourhood_cleansed, data = final_df)

Residuals:
   Min     1Q Median     3Q    Max
 -8592  -2980  -1513    342 290028

Coefficients:
                                    Estimate Std. Error t value
(Intercept)                          5483.21     573.47   9.561
neighbourhood_cleansedArmour Square  -408.37    1082.71  -0.377
neighbourhood_cleansedAshburn       -2558.21    2357.51  -1.085
neighbourhood_cleansedAuburn Gresham -1158.21    1334.68  -0.868
neighbourhood_cleansedAustin        -2015.48     660.99  -3.049
neighbourhood_cleansedAvalon Park   -3533.21    7253.88  -0.487
neighbourhood_cleansedAvondale      -1028.10     722.87  -1.422
neighbourhood_cleansedBelmont Cragin  -776.21     922.91  -0.841
neighbourhood_cleansedBeverly       -3190.35    2792.64  -1.142
neighbourhood_cleansedBridgeport    -2362.47     870.33  -2.714
neighbourhood_cleansedBrighton Park -2268.47    1755.27  -1.292
neighbourhood_cleansedCalumet Heights -3804.36  1529.71  -2.487
neighbourhood_cleansedChatham       -2683.21    1798.29  -1.492
neighbourhood_cleansedChicago Lawn  -3705.71    1896.57  -1.954
neighbourhood_cleansedClearing      -2783.21    4214.12  -0.660
neighbourhood_cleansedDouglas       -1630.08     619.14  -2.633
neighbourhood_cleansedDunning       -2677.65    1505.17  -1.779
neighbourhood_cleansedEast Garfield Park -2425.49  611.67  -3.965
neighbourhood_cleansedEast Side     -3213.21    3007.30  -1.068
neighbourhood_cleansedEdgewater     -1515.71     617.04  -2.456
neighbourhood_cleansedEdison Park   -1013.21    7253.88  -0.140
neighbourhood_cleansedEnglewood     -3309.21     846.29  -3.910
neighbourhood_cleansedForest Glen   -1353.21    2477.67  -0.546
neighbourhood_cleansedGage Park     -2513.21    7253.88  -0.346
neighbourhood_cleansedGrand Boulevard -1226.38   592.32  -2.070
```

neighbourhood_cleansedHermosa	-2099.21	1715.62	-1.224
neighbourhood_cleansedHumboldt Park	-2192.68	589.46	-3.720
neighbourhood_cleansedHyde Park	-2330.78	659.17	-3.536
neighbourhood_cleansedIrving Park	-1816.23	702.35	-2.586
neighbourhood_cleansedJefferson Park	-2425.93	1383.26	-1.754
neighbourhood_cleansedKenwood	-1838.21	934.64	-1.967
neighbourhood_cleansedLincoln Park	1511.69	647.80	2.334
neighbourhood_cleansedLincoln Square	-1749.84	650.84	-2.689
neighbourhood_cleansedLogan Square	-455.04	577.32	-0.788
neighbourhood_cleansedLoop	4458.63	694.09	6.424
neighbourhood_cleansedLower West Side	-1994.76	591.39	-3.373
neighbourhood_cleansedMontclare	-2765.21	1555.78	-1.777
neighbourhood_cleansedMorgan Park	-2426.96	2620.14	-0.926
neighbourhood_cleansedMount Greenwood	-4065.71	3660.79	-1.111
neighbourhood_cleansedNear North Side	1997.10	577.40	3.459
neighbourhood_cleansedNear West Side	62.76	578.55	0.108
neighbourhood_cleansedNew City	-3466.28	1155.18	-3.001
neighbourhood_cleansedNorth Center	109.69	644.44	0.170
neighbourhood_cleansedNorth Lawndale	-2449.68	611.64	-4.005
neighbourhood_cleansedNorth Park	-1515.38	1042.45	-1.454
neighbourhood_cleansedNorwood Park	-3511.78	1482.01	-2.370
neighbourhood_cleansedPortage Park	-1580.67	716.23	-2.207
neighbourhood_cleansedPullman	-2393.21	1715.62	-1.395
neighbourhood_cleansedRogers Park	-1769.62	646.72	-2.736
neighbourhood_cleansedRoseland	-2344.46	1190.90	-1.969
neighbourhood_cleansedSouth Chicago	-3429.09	1047.78	-3.273
neighbourhood_cleansedSouth Deering	-2858.21	5145.27	-0.556
neighbourhood_cleansedSouth Lawndale	-3129.42	894.40	-3.499
neighbourhood_cleansedSouth Shore	-1913.21	667.13	-2.868
neighbourhood_cleansedUptown	-542.04	583.93	-0.928
neighbourhood_cleansedWashington Heights	-2447.21	1953.17	-1.253
neighbourhood_cleansedWashington Park	-1445.37	650.45	-2.222
neighbourhood_cleansedWest Englewood	2573.94	1678.95	1.533
neighbourhood_cleansedWest Garfield Park	-2157.49	1076.51	-2.004

```
neighbourhood_cleansedWest Lawn           2440.13    2477.67   0.985
neighbourhood_cleansedWest Pullman       -2783.21    1953.17  -1.425
neighbourhood_cleansedWest Ridge         -2621.21     656.81  -3.991
neighbourhood_cleansedWest Town            829.30     575.63   1.441
neighbourhood_cleansedWoodlawn           -2688.14     610.07  -4.406
                                                    Pr(>|t|)
(Intercept)                                          < 2e-16 ***
neighbourhood_cleansedArmour Square                 0.706045
neighbourhood_cleansedAshburn                       0.277867
neighbourhood_cleansedAuburn Gresham                0.385517
neighbourhood_cleansedAustin                        0.002295 **
neighbourhood_cleansedAvalon Park                   0.626204
neighbourhood_cleansedAvondale                      0.154957
neighbourhood_cleansedBelmont Cragin                0.400327
neighbourhood_cleansedBeverly                       0.253286
neighbourhood_cleansedBridgeport                    0.006640 **
neighbourhood_cleansedBrighton Park                 0.196230
neighbourhood_cleansedCalumet Heights               0.012885 *
neighbourhood_cleansedChatham                       0.135681
neighbourhood_cleansedChicago Lawn                  0.050717 .
neighbourhood_cleansedClearing                      0.508969
neighbourhood_cleansedDouglas                       0.008470 **
neighbourhood_cleansedDunning                       0.075248 .
neighbourhood_cleansedEast Garfield Park            7.33e-05 ***
neighbourhood_cleansedEast Side                     0.285312
neighbourhood_cleansedEdgewater                     0.014036 *
neighbourhood_cleansedEdison Park                   0.888915
neighbourhood_cleansedEnglewood                     9.23e-05 ***
neighbourhood_cleansedForest Glen                   0.584957
neighbourhood_cleansedGage Park                     0.728995
neighbourhood_cleansedGrand Boulevard               0.038410 *
neighbourhood_cleansedHermosa                       0.221113
neighbourhood_cleansedHumboldt Park                 0.000200 ***
neighbourhood_cleansedHyde Park                     0.000407 ***
```

```
neighbourhood_cleansedIrving Park         0.009714 **
neighbourhood_cleansedJefferson Park      0.079472 .
neighbourhood_cleansedKenwood             0.049215 *
neighbourhood_cleansedLincoln Park        0.019621 *
neighbourhood_cleansedLincoln Square      0.007177 **
neighbourhood_cleansedLogan Square        0.430589
neighbourhood_cleansedLoop                1.34e-10 ***
neighbourhood_cleansedLower West Side     0.000744 ***
neighbourhood_cleansedMontclare           0.075511 .
neighbourhood_cleansedMorgan Park         0.354308
neighbourhood_cleansedMount Greenwood     0.266739
neighbourhood_cleansedNear North Side     0.000543 ***
neighbourhood_cleansedNear West Side      0.913617
neighbourhood_cleansedNew City            0.002695 **
neighbourhood_cleansedNorth Center        0.864851
neighbourhood_cleansedNorth Lawndale      6.20e-05 ***
neighbourhood_cleansedNorth Park          0.146039
neighbourhood_cleansedNorwood Park        0.017810 *
neighbourhood_cleansedPortage Park        0.027322 *
neighbourhood_cleansedPullman             0.163035
neighbourhood_cleansedRogers Park         0.006215 **
neighbourhood_cleansedRoseland            0.048997 *
neighbourhood_cleansedSouth Chicago       0.001066 **
neighbourhood_cleansedSouth Deering       0.578553
neighbourhood_cleansedSouth Lawndale      0.000467 ***
neighbourhood_cleansedSouth Shore         0.004134 **
neighbourhood_cleansedUptown              0.353272
neighbourhood_cleansedWashington Heights  0.210230
neighbourhood_cleansedWashington Park     0.026279 *
neighbourhood_cleansedWest Englewood      0.125264
neighbourhood_cleansedWest Garfield Park  0.045056 *
neighbourhood_cleansedWest Lawn           0.324703
neighbourhood_cleansedWest Pullman        0.154170
neighbourhood_cleansedWest Ridge          6.59e-05 ***
neighbourhood_cleansedWest Town           0.149681
neighbourhood_cleansedWoodlawn            1.05e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7231 on 78249 degrees of freedom
Multiple R-squared:  0.03579,   Adjusted R-squared:  0.03501
F-statistic:  46.1 on 63 and 78249 DF,  p-value: < 2.2e-16
```

mymodel_2 <-lm(airbnb_30_days_price ~ Zip.Code,data = final_df)

summary(mymodel_2)

```
Call:
lm(formula = airbnb_30_days_price ~ Zip.Code, data = final_df)

Residuals:
   Min     1Q Median     3Q    Max
 -5466  -3095  -1805    414 294504

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6839.40684 4311.83334   1.586    0.113
Zip.Code      -0.02266    0.07109  -0.319    0.750

Residual standard error: 7364 on 78191 degrees of freedom
  (120 observations deleted due to missingness)
Multiple R-squared:  1.3e-06,   Adjusted R-squared:  -1.149e-05
F-statistic: 0.1016 on 1 and 78191 DF,  p-value: 0.7499
```

 # Questions for future steps?

 # 1. I would like to plot the airbnb properties on map

 # 2. I think I need to look for more data to determine the correlation and to predict prices accurately