```
# Assignment: Project step 2
# Name: Anjale, Jiteshwar
# Date: 2021-05-22
#Analysis of how AirBnB rentals prices affects the nearby housing rental
prices in Chicago

## Load the readxl package
library(readxl)

## Warning: package 'readxl' was built under R version 4.0.5

## Load the plyr package
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.5

##
## 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

## Load the plyr package
library(plyr)

## Warning: package 'plyr' was built under R version 4.0.5

## ------------------------------------------------------------------------
----

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first,
then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------
----

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

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

```
## Warning: package 'purrr' was built under R version 4.0.5

##
## Attaching package: 'purrr'

## The following object is masked from 'package:plyr':
##
##     compact

## Load the tidyverse package
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.0.5

## -- Attaching packages --------------------------------------- tidyverse
1.3.1 --

## v ggplot2 3.3.3     v readr   1.4.0
## v tibble  3.1.0     v stringr 1.4.0
## v tidyr   1.1.3     v forcats 0.5.1

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'readr' was built under R version 4.0.5

## Warning: package 'stringr' was built under R version 4.0.5

## Warning: package 'forcats' was built under R version 4.0.5

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x plyr::arrange()   masks dplyr::arrange()
## x purrr::compact()  masks plyr::compact()
## x plyr::count()     masks dplyr::count()
## x plyr::failwith()  masks dplyr::failwith()
## x dplyr::filter()   masks stats::filter()
## x plyr::id()        masks dplyr::id()
## x dplyr::lag()      masks stats::lag()
## x plyr::mutate()    masks dplyr::mutate()
## x plyr::rename()    masks dplyr::rename()
## x plyr::summarise() masks dplyr::summarise()
## x plyr::summarize() masks dplyr::summarize()

library(ggplot2)


## Set the working directory to the root of your DSC 520 directory
setwd('C:/Users/anjal/OneDrive/Desktop/MS/DSC520/project')

## Load the Airbnb dataset
airbnb_df <-
read.csv("C:/Users/anjal/OneDrive/Desktop/MS/DSC520/project/AirBnb-listing-
```

```
data.csv")
glimpse(airbnb_df)

## Rows: 226,030
## Columns: 17
## $ ï..id                          <int> 38585, 80905, 108061, 155305,
160594, 2~
## $ name                           <chr> "Charming Victorian home - twin
beds + ~
## $ host_id                        <int> 165529, 427027, 320564, 746673,
769252,~
## $ host_name                      <chr> "Evelyne", "Celeste", "Lisa",
"BonPaul"~
## $ neighbourhood_group            <chr> "", "", "", "", "", "", "", "", "",
"",~
## $ neighbourhood                  <chr> "28804", "28801", "28801", "28806",
"28~
## $ latitude                       <dbl> 35.65146, 35.59779, 35.60670,
35.57864,~
## $ longitude                      <dbl> -82.62792, -82.55540, -82.55563, -
82.59~
## $ room_type                      <chr> "Private room", "Entire home/apt",
"Ent~
## $ price                          <int> 60, 470, 75, 90, 125, 134, 48, 65,
71, ~
## $ minimum_nights                 <int> 1, 1, 30, 1, 30, 7, 1, 3, 28, 90,
30, 4~
## $ number_of_reviews              <int> 138, 114, 89, 267, 58, 54, 137, 57,
537~
## $ last_review                    <chr> "16/02/20", "7/9/2020", "30/11/19",
"22~
## $ reviews_per_month              <dbl> 1.14, 1.03, 0.81, 2.39, 0.52, 0.49,
1.3~
## $ calculated_host_listings_count <int> 1, 11, 2, 5, 1, 1, 1, 2, 1, 1, 2,
1, 1,~
## $ availability_365               <int> 0, 288, 298, 0, 0, 294, 0, 106,
207, 33~
## $ city                           <chr> "Asheville", "Asheville",
"Asheville", ~

#Above data set contains information across US cities
#Filtering the data based on city==Chicago as we are focusing on Chicago
airbnb_chicago_df <- filter(airbnb_df,city=="Chicago")
glimpse(airbnb_chicago_df)

## Rows: 6,397
## Columns: 17
## $ ï..id                          <int> 2384, 4505, 7126, 9811, 10610,
10945, 1~
## $ name                           <chr> "Hyde Park - Walk to UChicago, 10
```

```
min t~
## $ host_id                          <int> 2613, 5775, 17928, 33004, 2140,
33004, ~
## $ host_name                        <chr> "Rebecca", "Craig & Kathleen",
"Sarah",~
## $ neighbourhood_group              <chr> "", "", "", "", "", "", "", "", "",
"",~
## $ neighbourhood                    <chr> "Hyde Park", "South Lawndale",
"West To~
## $ latitude                         <dbl> 41.78790, 41.85495, 41.90289,
41.91769,~
## $ longitude                        <dbl> -87.58780, -87.69696, -87.68182, -
87.63~
## $ room_type                        <chr> "Private room", "Entire home/apt",
"Ent~
## $ price                            <int> 60, 105, 60, 65, 21, 115, 99, 289,
99, ~
## $ minimum_nights                   <int> 2, 2, 2, 4, 1, 4, 5, 2, 91, 32, 32,
2, ~
## $ number_of_reviews                <int> 178, 395, 384, 49, 44, 19, 9, 4, 9,
37,~
## $ last_review                      <chr> "15/12/19", "14/07/20", "8/3/2020",
"23~
## $ reviews_per_month                <dbl> 2.56, 2.81, 2.81, 0.63, 0.61, 0.24,
0.1~
## $ calculated_host_listings_count <int> 1, 1, 1, 9, 5, 9, 1, 1, 2, 4, 4, 1,
2, ~
## $ availability_365                 <int> 353, 155, 321, 300, 168, 325, 316,
179,~
## $ city                             <chr> "Chicago", "Chicago", "Chicago",
"Chica~
```

```
## Load the Affordable rental housing dataset
housing_df=read.csv("C:/Users/anjal/OneDrive/Desktop/MS/DSC520/project/afford
able-rental-housing-developments.csv")
glimpse(housing_df)
```

```
## Rows: 428
## Columns: 14
## $ neighbourhood         <chr> "Rogers Park", "Rogers Park", "Rogers Park",
"Ro~
## $ Community_Area_Number <int> 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3,
3, ~
## $ Property_Type         <chr> "Senior", "Supportive Housing", "Senior",
"Senio~
## $ Property_Name         <chr> "Morse Senior Apts.", "Wayne Street Apts.",
"Jam~
## $ Address               <chr> "6928 N. Wayne Ave.", "6808 N. Wayne Ave.",
"745~
## $ Zip_Code              <int> 60626, 60626, 60626, 60626, 60659, 60659,
```

```
60645,~
## $ Phone_Number          <chr> "312-602-6207", "773-572-5272", "773-743-
3699", ~
## $ Management_Company    <chr> "Morse Urban Dev.", "The Thresholds",
"Hispanic ~
## $ Units                 <int> 44, 297, 57, 119, 99, 117, 3, 2, 89, 60, 4,
13, ~
## $ X_Coordinate          <dbl> 1165844, 1165865, 1163641, 1165844, 1153826,
115~
## $ Y_Coordinate          <dbl> 1946059, 1945402, 1949531, 1946059, 1940243,
194~
## $ Latitude              <dbl> 42.00757, 42.00577, 42.01715, 42.00757,
41.99187~
## $ Longitude             <dbl> -87.66517, -87.66511, -87.67318, -87.66517,
-87.~
## $ Zip.Codes             <int> 21853, 21853, 21853, 21853, 4450, 4450,
4450, 44~
```

```
## Load the Average rent Chicago neighborhood dataset
avg_rent_df <-
read_excel("C:/Users/anjal/OneDrive/Desktop/MS/DSC520/project/Average_rent_Ch
icago_neighbourhood.xls")
glimpse(avg_rent_df)
```

```
## Rows: 180
## Columns: 2
## $ neighbourhood  <chr> "Dearborn Park", "Printer's Row", "Streeterville",
"Nea~
## $ `Average Rent` <dbl> 2419, 2419, 2410, 2316, 2308, 2307, 2307, 2294,
2291, 2~
```

```
#Merge the airbnb df with rental housing df based on neighbourhood
final_1_df <- left_join(airbnb_chicago_df,housing_df,by="neighbourhood" )
glimpse(final_1_df)
```

```
## Rows: 63,486
## Columns: 30
## $ ï..id                         <int> 2384, 2384, 2384, 2384, 4505, 4505,
450~
## $ name                          <chr> "Hyde Park - Walk to UChicago, 10
min t~
## $ host_id                       <int> 2613, 2613, 2613, 2613, 5775, 5775,
577~
## $ host_name                     <chr> "Rebecca", "Rebecca", "Rebecca",
"Rebec~
## $ neighbourhood_group           <chr> "", "", "", "", "", "", "", "", "",
"",~
## $ neighbourhood                 <chr> "Hyde Park", "Hyde Park", "Hyde
Park", ~
## $ latitude                      <dbl> 41.78790, 41.78790, 41.78790,
41.78790,~
```

```
## $ longitude                          <dbl> -87.58780, -87.58780, -87.58780, -
87.58~
## $ room_type                          <chr> "Private room", "Private room",
"Privat~
## $ price                              <int> 60, 60, 60, 60, 105, 105, 105, 60,
60, ~
## $ minimum_nights                     <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, ~
## $ number_of_reviews                  <int> 178, 178, 178, 178, 395, 395, 395,
384,~
## $ last_review                        <chr> "15/12/19", "15/12/19", "15/12/19",
"15~
## $ reviews_per_month                  <dbl> 2.56, 2.56, 2.56, 2.56, 2.81, 2.81,
2.8~
## $ calculated_host_listings_count <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, ~
## $ availability_365                   <int> 353, 353, 353, 353, 155, 155, 155,
321,~
## $ city                               <chr> "Chicago", "Chicago", "Chicago",
"Chica~
## $ Community_Area_Number              <int> 41, 41, 41, 41, 30, 30, 30, 24, 24,
24,~
## $ Property_Type                      <chr> "ARO", "ARO", "ARO", "Multifamily",
"Mu~
## $ Property_Name                      <chr> "5432-44 S. Woodlawn", "Vue53",
"City H~
## $ Address                            <chr> "5432 S. Woodlawn Ave.", "1330 E.
53rd ~
## $ Zip_Code                           <int> 60615, 60615, 60615, 60615, 60623,
6062~
## $ Phone_Number                       <chr> "312-480-0933", "773-355-4972",
"773-54~
## $ Management_Company                 <chr> "Chicago Apartment Finders", "Peak
Camp~
## $ Units                              <int> 10, 27, 36, 36, 8, 2, 29, 3, 1, 61,
10,~
## $ X_Coordinate                       <dbl> 1185103, 1185905, 1187194, 1187148,
115~
## $ Y_Coordinate                       <dbl> 1869464, 1870431, 1871413, 1870068,
188~
## $ Latitude                           <dbl> 41.79696, 41.79960, 41.80226,
41.79857,~
## $ Longitude                          <dbl> -87.59674, -87.59376, -87.58900, -
87.58~
## $ Zip.Codes                          <int> 21192, 21192, 21192, 21192, 21569,
2156~
```

```
#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") )
glimpse(final_2_df)
```

```
## Rows: 43,334
## Columns: 31
## $ ï..id                           <int> 2384, 2384, 2384, 2384, 7126, 7126,
712~
## $ name                            <chr> "Hyde Park - Walk to UChicago, 10
min t~
## $ host_id                         <int> 2613, 2613, 2613, 2613, 17928,
17928, 1~
## $ host_name                       <chr> "Rebecca", "Rebecca", "Rebecca",
"Rebec~
## $ neighbourhood_group             <chr> "", "", "", "", "", "", "", "", "",
"",~
## $ neighbourhood                   <chr> "Hyde Park", "Hyde Park", "Hyde
Park", ~
## $ latitude                        <dbl> 41.78790, 41.78790, 41.78790,
41.78790,~
## $ longitude                       <dbl> -87.58780, -87.58780, -87.58780, -
87.58~
## $ room_type                       <chr> "Private room", "Private room",
"Privat~
## $ price                           <int> 60, 60, 60, 60, 60, 60, 60, 60, 60,
60,~
## $ minimum_nights                  <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, ~
## $ number_of_reviews               <int> 178, 178, 178, 178, 384, 384, 384,
384,~
## $ last_review                     <chr> "15/12/19", "15/12/19", "15/12/19",
"15~
## $ reviews_per_month               <dbl> 2.56, 2.56, 2.56, 2.56, 2.81, 2.81,
2.8~
## $ calculated_host_listings_count <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, ~
## $ availability_365                <int> 353, 353, 353, 353, 321, 321, 321,
321,~
## $ city                            <chr> "Chicago", "Chicago", "Chicago",
"Chica~
## $ Community_Area_Number           <int> 41, 41, 41, 41, 24, 24, 24, 24, 24,
24,~
## $ Property_Type                   <chr> "ARO", "ARO", "ARO", "Multifamily",
"Mu~
## $ Property_Name                   <chr> "5432-44 S. Woodlawn", "Vue53",
"City H~
## $ Address                         <chr> "5432 S. Woodlawn Ave.", "1330 E.
53rd ~
## $ Zip_Code                        <int> 60615, 60615, 60615, 60615, 60622,
6062~
## $ Phone_Number                    <chr> "312-480-0933", "773-355-4972",
"773-54~
## $ Management_Company              <chr> "Chicago Apartment Finders", "Peak
Camp~
```

```
## $ Units                          <int> 10, 27, 36, 36, 3, 1, 61, 10, 24,
24, 2~
## $ X_Coordinate                   <dbl> 1185103, 1185905, 1187194, 1187148,
NA,~
## $ Y_Coordinate                   <dbl> 1869464, 1870431, 1871413, 1870068,
NA,~
## $ Latitude                       <dbl> 41.79696, 41.79960, 41.80226,
41.79857,~
## $ Longitude                      <dbl> -87.59674, -87.59376, -87.58900, -
87.58~
## $ Zip.Codes                      <int> 21192, 21192, 21192, 21192, NA,
21560, ~
## $ `Average Rent`                 <dbl> 1431, 1431, 1431, 1431, 2147, 2147,
214~
```

```
#By looking at the data we can say that
#Airbnb data
# 1. Variable id is just an identifier and we can ignore it.
# 2. The dataframe have data of many cities, we need to filter it for
Chicago.
# 3. We can factor the field room_type - Private room,Entire home/apt,Hotel
room,   Shared room
# 4. We can drop the host_id and host_name,neighbourhood_group,name fields
from the dataset
# 5. We can drop fields like
last_review,number_of_reviews,reviews_per_month,calculated_host_listings_coun
t

#Average rent Chicago neighborhood data
# 6. We can drop
Property_Name,Phone_Number,Management_Company,Units,Zip.Codes from the
dataset

#Average rent Chicago neighborhood data
# 7. rename the Average Rent to Average_Rent

# Apply above transformation to the dataframe
final_df <- subset(final_2_df, select = -
c(ï..id,name,host_id,host_name,last_review,neighbourhood_group,number_of_revi
ews,reviews_per_month,calculated_host_listings_count,Property_Name,Phone_Numb
er,Management_Company,Units,Community_Area_Number,Address,Zip.Codes) )
glimpse(final_df)
```

```
## Rows: 43,334
## Columns: 15
## $ neighbourhood    <chr> "Hyde Park", "Hyde Park", "Hyde Park", "Hyde
Park", "~
## $ latitude         <dbl> 41.78790, 41.78790, 41.78790, 41.78790, 41.90289,
41.~
## $ longitude        <dbl> -87.58780, -87.58780, -87.58780, -87.58780, -
```

```
87.68182~
## $ room_type        <chr> "Private room", "Private room", "Private room",
"Priv~
## $ price            <int> 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60,
60, 6~
## $ minimum_nights   <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2,~
## $ availability_365 <int> 353, 353, 353, 353, 321, 321, 321, 321, 321, 321,
321~
## $ city             <chr> "Chicago", "Chicago", "Chicago", "Chicago",
"Chicago"~
## $ Property_Type    <chr> "ARO", "ARO", "ARO", "Multifamily",
"Multifamily", "A~
## $ Zip_Code         <int> 60615, 60615, 60615, 60615, 60622, 60622, 60622,
6062~
## $ X_Coordinate     <dbl> 1185103, 1185905, 1187194, 1187148, NA, 1165486,
1159~
## $ Y_Coordinate     <dbl> 1869464, 1870431, 1871413, 1870068, NA, 1907421,
1907~
## $ Latitude         <dbl> 41.79696, 41.79960, 41.80226, 41.79857, NA,
41.90156,~
## $ Longitude        <dbl> -87.59674, -87.59376, -87.58900, -87.58921, NA, -
87.6~
## $ `Average Rent`   <dbl> 1431, 1431, 1431, 1431, 2147, 2147, 2147, 2147,
2147,~

#Rename Average Rent to Average_Rent
colnames(final_df)[15] <- "Average_Rent"

# Checking the summary of data set to gauge the value range of each numerical
variable
summary(final_df)

##  neighbourhood         latitude       longitude       room_type
##  Length:43334       Min.   :41.65   Min.   :-87.84   Length:43334
##  Class :character   1st Qu.:41.89   1st Qu.:-87.70   Class :character
##  Mode  :character   Median :41.90   Median :-87.68   Mode  :character
##                     Mean   :41.90   Mean   :-87.68
##                     3rd Qu.:41.92   3rd Qu.:-87.66
##                     Max.   :42.02   Max.   :-87.55
##
##      price          minimum_nights    availability_365     city
##  Min.   :   10.0   Min.   :  1.000   Min.   :  0      Length:43334
##  1st Qu.:   60.0   1st Qu.:  1.000   1st Qu.: 13      Class :character
##  Median :   94.0   Median :  2.000   Median :137      Mode  :character
##  Mean   :  143.9   Mean   :  6.146   Mean   :161
##  3rd Qu.:  145.0   3rd Qu.:  3.000   3rd Qu.:321
##  Max.   :10000.0   Max.   :500.000   Max.   :365
##
##  Property_Type         Zip_Code      X_Coordinate      Y_Coordinate
```

```
##  Length:43334       Min.   :60607   Min.   :1127615   Min.   :1818758
##  Class :character   1st Qu.:60622   1st Qu.:1156537   1st Qu.:1900741
##  Mode  :character   Median :60624   Median :1159393   Median :1907821
##                     Mean   :60632   Mean   :1161973   Mean   :1904333
##                     3rd Qu.:60647   3rd Qu.:1165486   3rd Qu.:1910798
##                     Max.   :60808   Max.   :1196632   Max.   :1949531
##                     NA's   :438     NA's   :9946      NA's   :9946
##     Latitude       Longitude       Average_Rent
##  Min.   :41.66   Min.   :-87.81   Min.   : 728
##  1st Qu.:41.88   1st Qu.:-87.70   1st Qu.:1785
##  Median :41.90   Median :-87.69   Median :2147
##  Mean   :41.89   Mean   :-87.68   Mean   :1880
##  3rd Qu.:41.91   3rd Qu.:-87.67   3rd Qu.:2147
##  Max.   :42.02   Max.   :-87.56   Max.   :2291
##  NA's   :9946    NA's   :9946

# 8. Range of values prices are varies from 0 to 10000. It looks like there
is outliers in the field.
# 9. Range of values minimum_nights varies from 1 to 500. It looks like there
is outliers in the field.
# 10. Range of values for availability_365 varies from 0 to 365.
# 11. Range of values for Average_Rent  varies from 728 to 2291.

#Calculate the 30 days price for airbnb property.
final_df$airbnb_30_days_price=final_df$price * 30
summary(final_df)

##  neighbourhood         latitude       longitude       room_type
##  Length:43334       Min.   :41.65   Min.   :-87.84   Length:43334
##  Class :character   1st Qu.:41.89   1st Qu.:-87.70   Class :character
##  Mode  :character   Median :41.90   Median :-87.68   Mode  :character
##                     Mean   :41.90   Mean   :-87.68
##                     3rd Qu.:41.92   3rd Qu.:-87.66
##                     Max.   :42.02   Max.   :-87.55
##
##      price          minimum_nights    availability_365     city
##  Min.   :   10.0   Min.   :  1.000   Min.   :  0      Length:43334
##  1st Qu.:   60.0   1st Qu.:  1.000   1st Qu.: 13      Class :character
##  Median :   94.0   Median :  2.000   Median :137      Mode  :character
##  Mean   :  143.9   Mean   :  6.146   Mean   :161
##  3rd Qu.:  145.0   3rd Qu.:  3.000   3rd Qu.:321
##  Max.   :10000.0   Max.   :500.000   Max.   :365
##
##  Property_Type         Zip_Code      X_Coordinate      Y_Coordinate
##  Length:43334       Min.   :60607   Min.   :1127615   Min.   :1818758
##  Class :character   1st Qu.:60622   1st Qu.:1156537   1st Qu.:1900741
##  Mode  :character   Median :60624   Median :1159393   Median :1907821
##                     Mean   :60632   Mean   :1161973   Mean   :1904333
##                     3rd Qu.:60647   3rd Qu.:1165486   3rd Qu.:1910798
##                     Max.   :60808   Max.   :1196632   Max.   :1949531
```

```
##                     NA's   :438     NA's   :9946      NA's   :9946
##     Latitude       Longitude       Average_Rent  airbnb_30_days_price
##  Min.   :41.66   Min.   :-87.81   Min.   : 728   Min.   :   300
##  1st Qu.:41.88   1st Qu.:-87.70   1st Qu.:1785   1st Qu.:  1800
##  Median :41.90   Median :-87.69   Median :2147   Median :  2820
##  Mean   :41.89   Mean   :-87.68   Mean   :1880   Mean   :  4318
##  3rd Qu.:41.91   3rd Qu.:-87.67   3rd Qu.:2147   3rd Qu.:  4350
##  Max.   :42.02   Max.   :-87.56   Max.   :2291   Max.   :300000
##  NA's   :9946    NA's   :9946
```

```
#Check missing values
apply(final_df, 2, function(x) any(is.na(x)))
```

```
##        neighbourhood             latitude            longitude
##                FALSE                FALSE                FALSE
##            room_type                price       minimum_nights
##                FALSE                FALSE                FALSE
##     availability_365                 city        Property_Type
##                FALSE                FALSE                 TRUE
##             Zip_Code         X_Coordinate         Y_Coordinate
##                 TRUE                 TRUE                 TRUE
##             Latitude            Longitude         Average_Rent
##                 TRUE                 TRUE                FALSE
## airbnb_30_days_price
##                FALSE
```

```
#It looks like there are some missing values for
#X_Coordinate ,Y_Coordinate, Latitude, Longitude
```

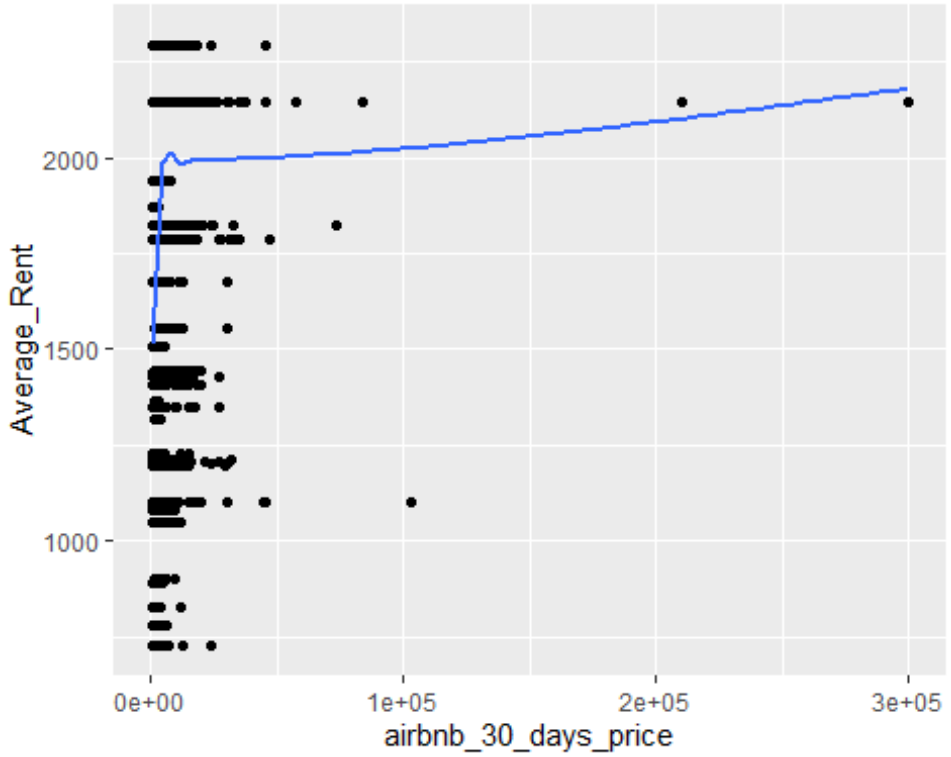
```
## 2.What does the final data set look like?
glimpse(final_df)
```

```
## Rows: 43,334
## Columns: 16
## $ neighbourhood        <chr> "Hyde Park", "Hyde Park", "Hyde Park", "Hyde
Park~
## $ latitude             <dbl> 41.78790, 41.78790, 41.78790, 41.78790,
41.90289,~
## $ longitude            <dbl> -87.58780, -87.58780, -87.58780, -87.58780, -
87.6~
## $ room_type            <chr> "Private room", "Private room", "Private
room", "~
## $ price                <int> 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60,
60, 6~
## $ minimum_nights       <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2~
## $ availability_365     <int> 353, 353, 353, 353, 321, 321, 321, 321, 321,
321,~
## $ city                 <chr> "Chicago", "Chicago", "Chicago", "Chicago",
"Chic~
```

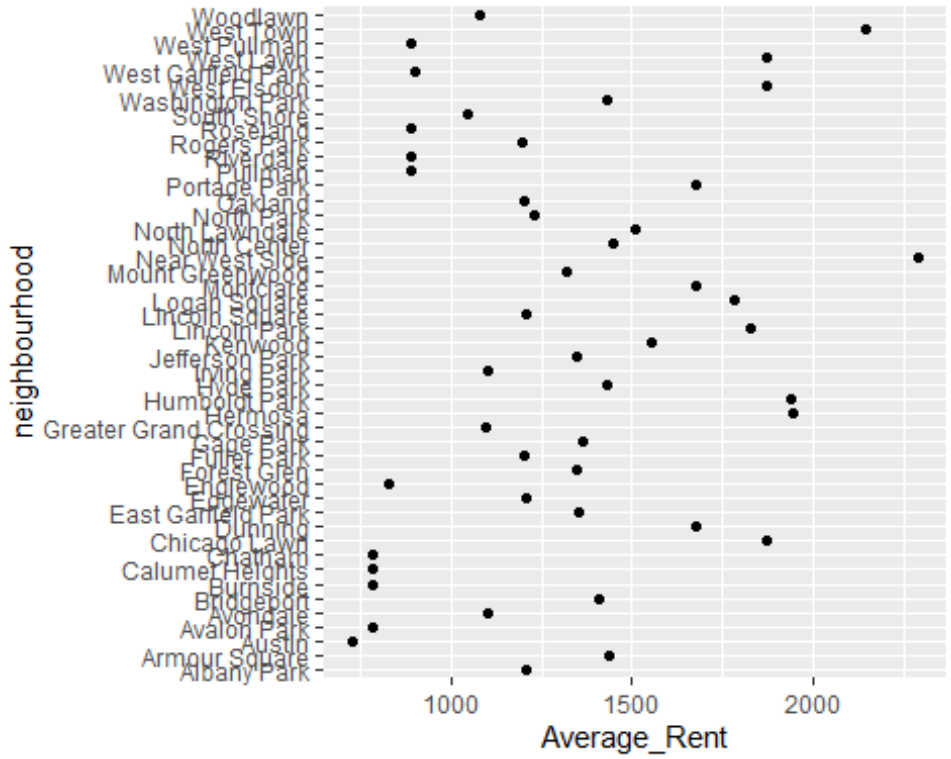
```
## $ Property_Type        <chr> "ARO", "ARO", "ARO", "Multifamily",
"Multifamily"~
## $ Zip_Code             <int> 60615, 60615, 60615, 60615, 60622, 60622,
60622, ~
## $ X_Coordinate         <dbl> 1185103, 1185905, 1187194, 1187148, NA,
1165486, ~
## $ Y_Coordinate         <dbl> 1869464, 1870431, 1871413, 1870068, NA,
1907421, ~
## $ Latitude             <dbl> 41.79696, 41.79960, 41.80226, 41.79857, NA,
41.90~
## $ Longitude            <dbl> -87.59674, -87.59376, -87.58900, -87.58921,
NA, -~
## $ Average_Rent         <dbl> 1431, 1431, 1431, 1431, 2147, 2147, 2147,
2147, 2~
## $ airbnb_30_days_price <dbl> 1800, 1800, 1800, 1800, 1800, 1800, 1800,
1800, 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)
```

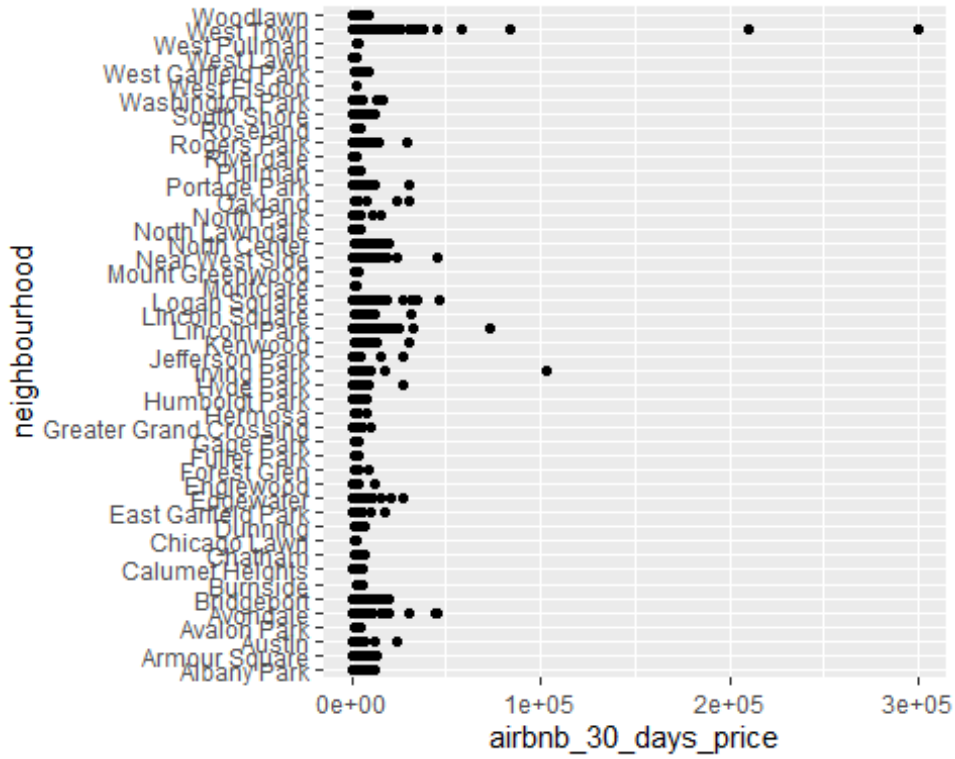
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



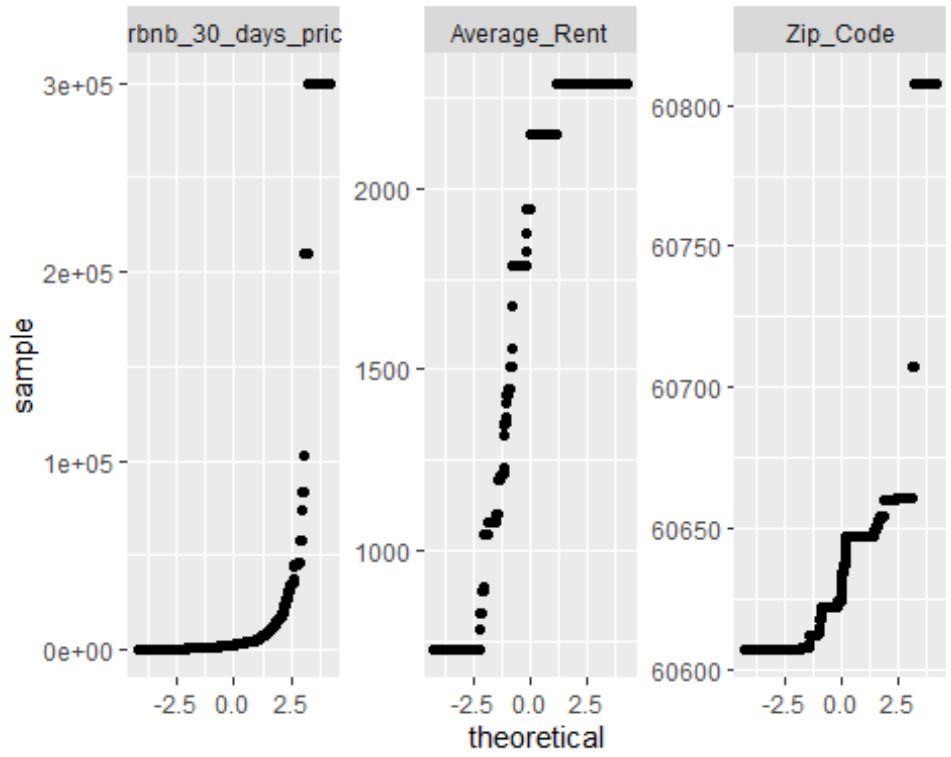

```
# Checking relation between airbnb_30_days_price and Average_Rent using
ggplot()
library(ggplot2)
ggplot(data = final_df, aes(y = neighbourhood, x = Average_Rent)) +
  geom_point() + geom_smooth(fill=NA)

## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



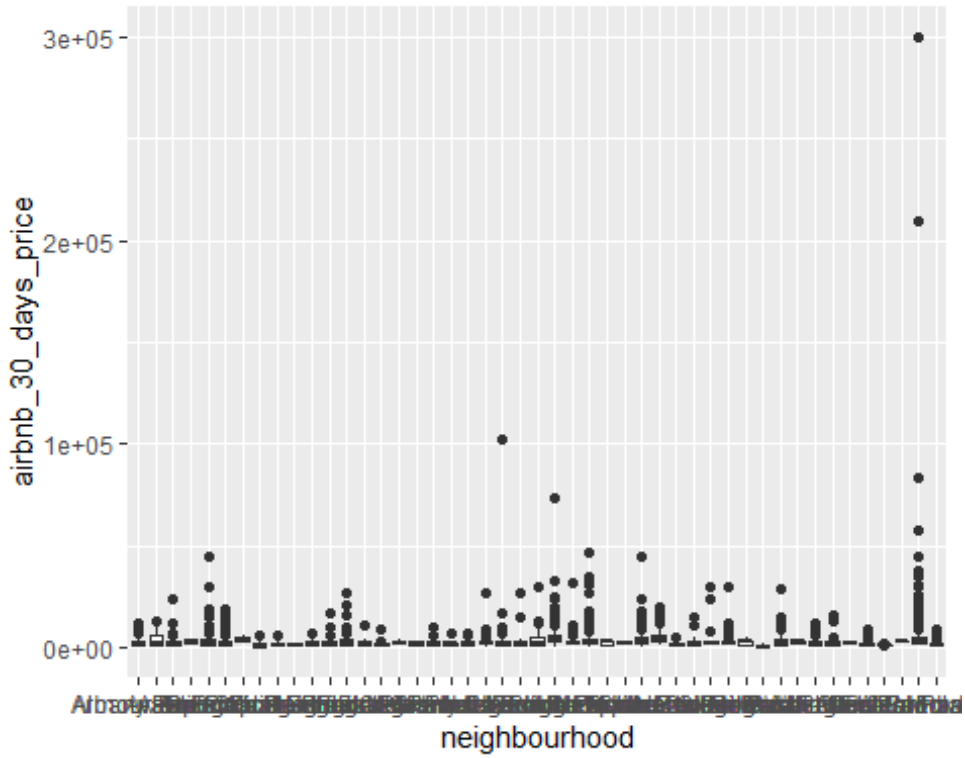

```
# Checking relation between airbnb_30_days_price and Average_Rent using
ggplot()
library(ggplot2)
ggplot(data = final_df, aes(y = neighbourhood, x = airbnb_30_days_price)) +
  geom_point() + geom_smooth(fill=NA)

## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

## Warning: Computation failed in `stat_smooth()`:
## NA/NaN/Inf in foreign function call (arg 3)
```



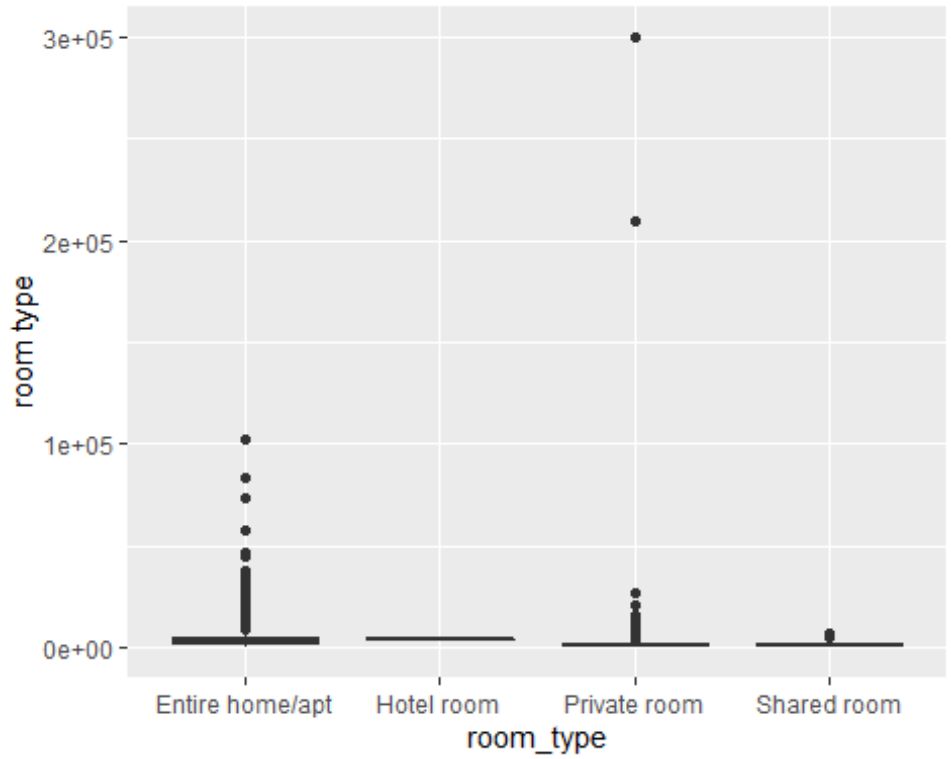

```
#We can see that there is relationship between neighbourhood and prices

# 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')

## Warning: Removed 438 rows containing non-finite values (stat_qq).
```



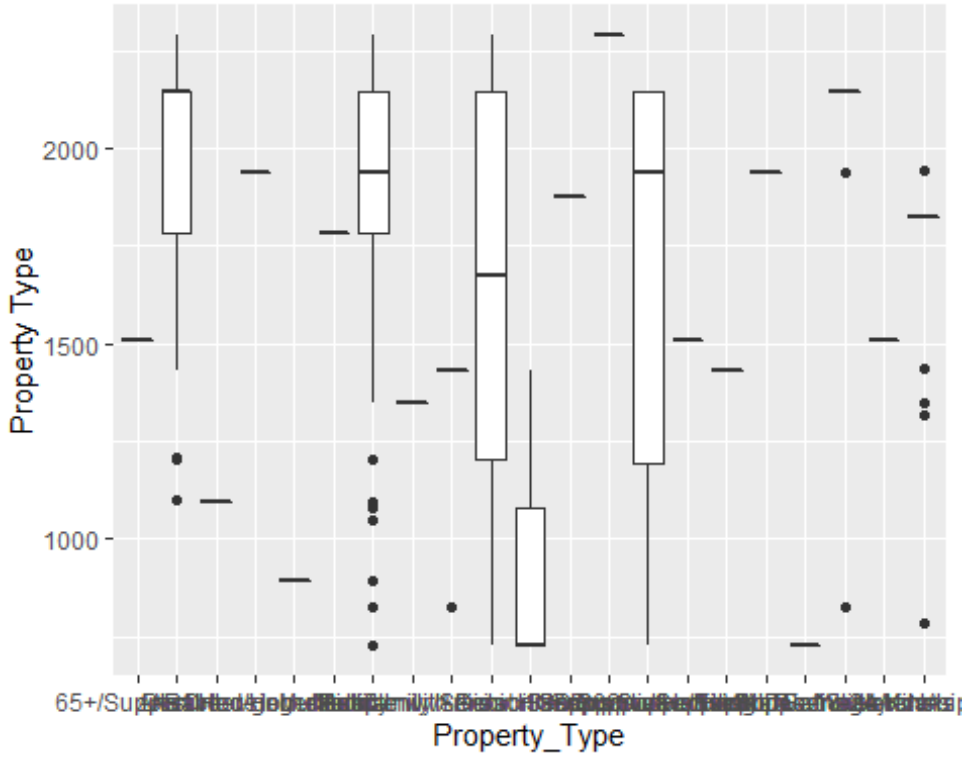

```
#None of the variables looks normally distributed



ggplot(data = final_df, aes(x = neighbourhood , 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("room type")
```




```
# 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("Property Type")
```




```
# 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")])
```

```
##       Zip_Code
## 1        60615
## 5        60622
## 10       60647
## 12       60654
## 28          NA
## 68       60612
## 130      60613
## 131      60618
## 138      60641
## 139      60634
## 167      60628
## 220      60661
## 221      60607
## 222      60608
## 275      60660
## 304      60637
```

```
## 322      60630
## 324      60625
## 325      60653
## 420      60639
## 421      60626
## 427      60640
## 789      60624
## 865      60623
## 892      60808
## 1188     60619
## 1665     60651
## 1858     60629
## 1883     60649
## 3917     60646
## 4404     60632
## 4925     60644
## 5115     60621
## 15733    60707
## 20164    60609
## 22189    60643
## 34924    60627
```

```
unique(final_df[c("neighbourhood")])
```

```
##                neighbourhood
## 1                  Hyde Park
## 5                  West Town
## 28               Lincoln Park
## 60               Logan Square
## 130              North Center
## 137               Irving Park
## 138              Portage Park
## 167                   Pullman
## 218           Near West Side
## 275                Edgewater
## 303               Bridgeport
## 304                 Woodlawn
## 322              Albany Park
## 325                  Kenwood
## 419                 Avondale
## 421              Rogers Park
## 425           Lincoln Square
## 597              Forest Glen
## 789       East Garfield Park
## 861           North Lawndale
## 1124                 Oakland
## 1188                Chatham
## 1606         Washington Park
## 1655           Humboldt Park
## 1772                 Dunning
```

```
## 1858               West Lawn
## 1883             South Shore
## 2577           Armour Square
## 3297      West Garfield Park
## 3373                 Hermosa
## 3917              North Park
## 4056          Jefferson Park
## 4404             West Elsdon
## 4924                  Austin
## 5114  Greater Grand Crossing
## 11918              Englewood
## 13348            Avalon Park
## 14423           Chicago Lawn
## 14746              Gage Park
## 15733              Montclare
## 17074               Roseland
## 17288        Calumet Heights
## 20164            Fuller Park
## 22187           West Pullman
## 33026               Burnside
## 33366        Mount Greenwood
## 34924              Riverdale
```

#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")
```

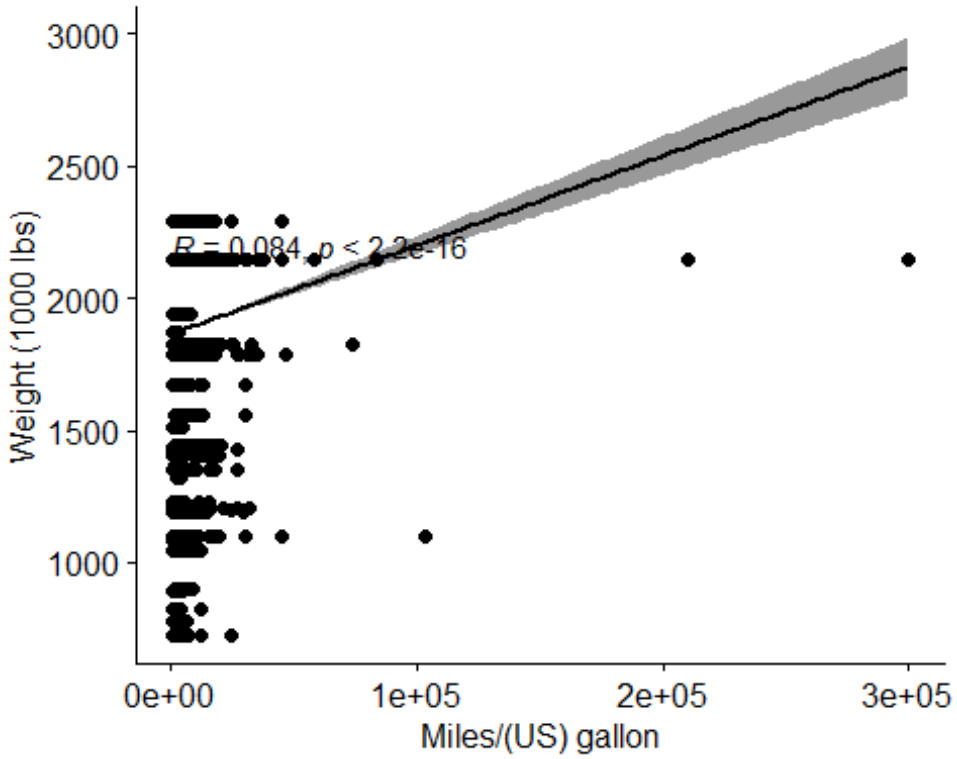
```
## Warning: package 'ggpubr' was built under R version 4.0.5
```

```
##
## Attaching package: 'ggpubr'
```

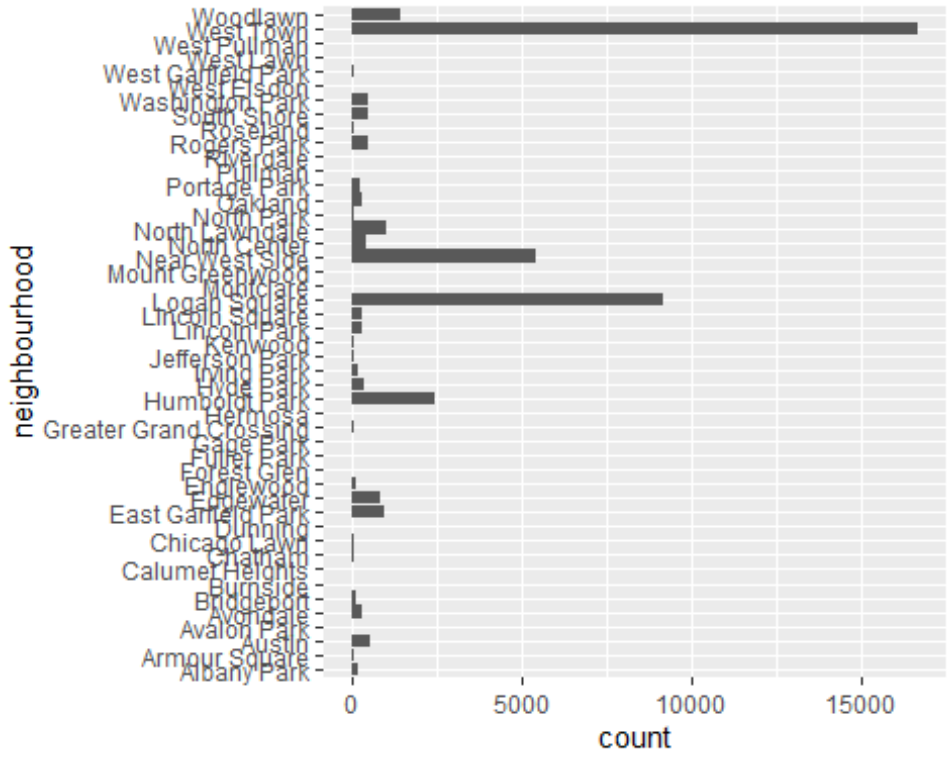
```
## The following object is masked from 'package:plyr':
##
##     mutate
```

```
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)")
```

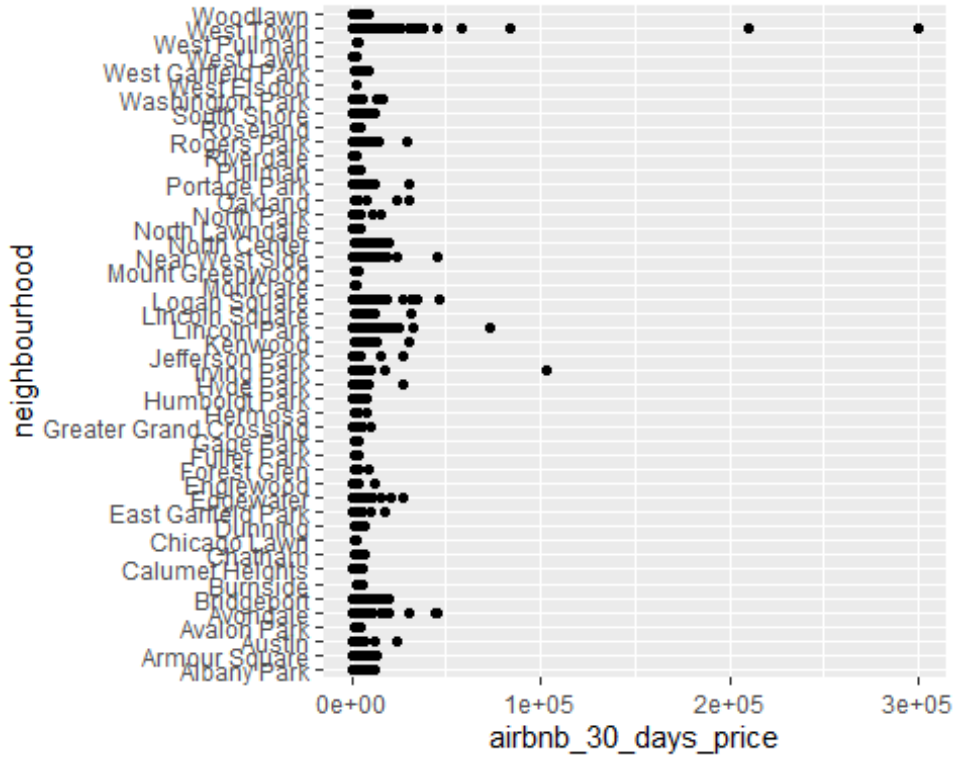
```
## `geom_smooth()` using formula 'y ~ x'
```




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

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

## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
ggplot(aes(y=neighbourhood,x=airbnb_30_days_price),data=final_df)+
  geom_point()
```

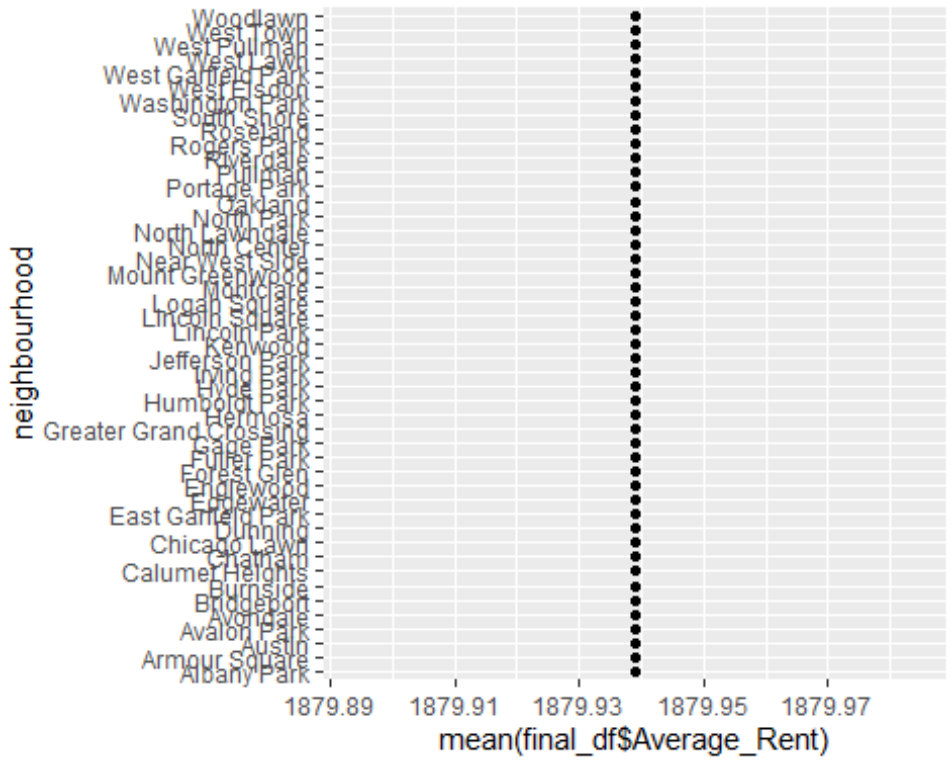


```
# From graph it looks like "West town" have major number of airbnb properties
# Also the prices of "West town" properties are high for airbnb rental.

# 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] 0.08402284
```

```
# 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,x=mean(final_df$Average_Rent)),data=final_df)+
  geom_point()
```

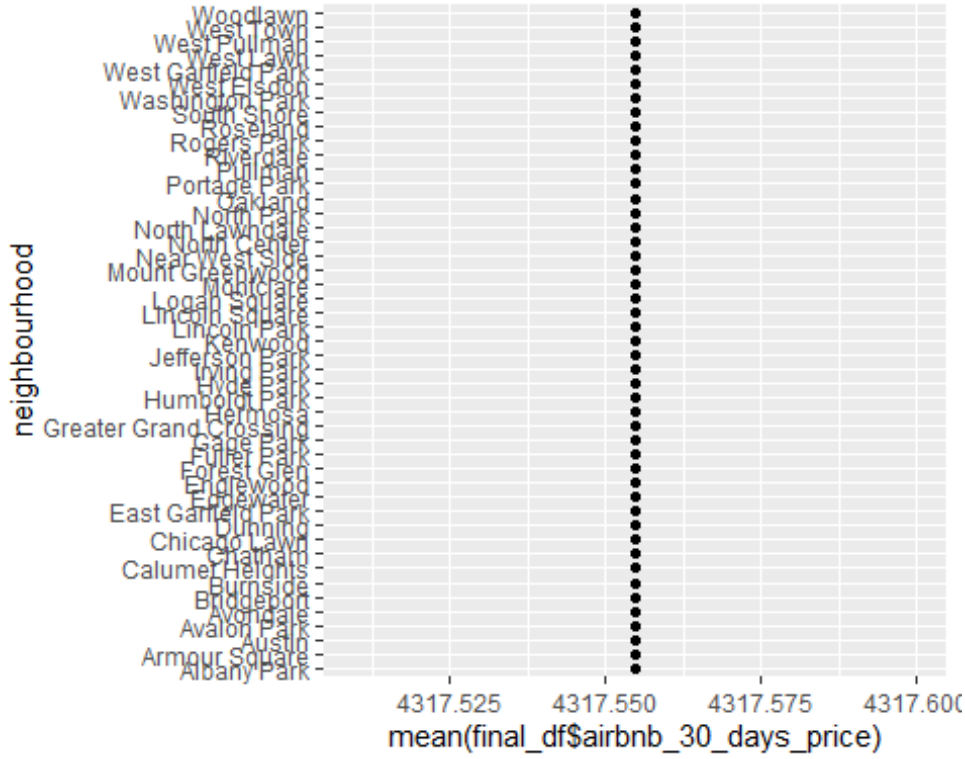
```
## Warning: Use of `final_df$Average_Rent` is discouraged. Use `Average_Rent`
## instead.
```




```
#The average rent price is ~1800 per month

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

ggplot(aes(y=neighbourhood,x=mean(final_df$airbnb_30_days_price)),data=final_
```

```
df)+
  geom_point()
```

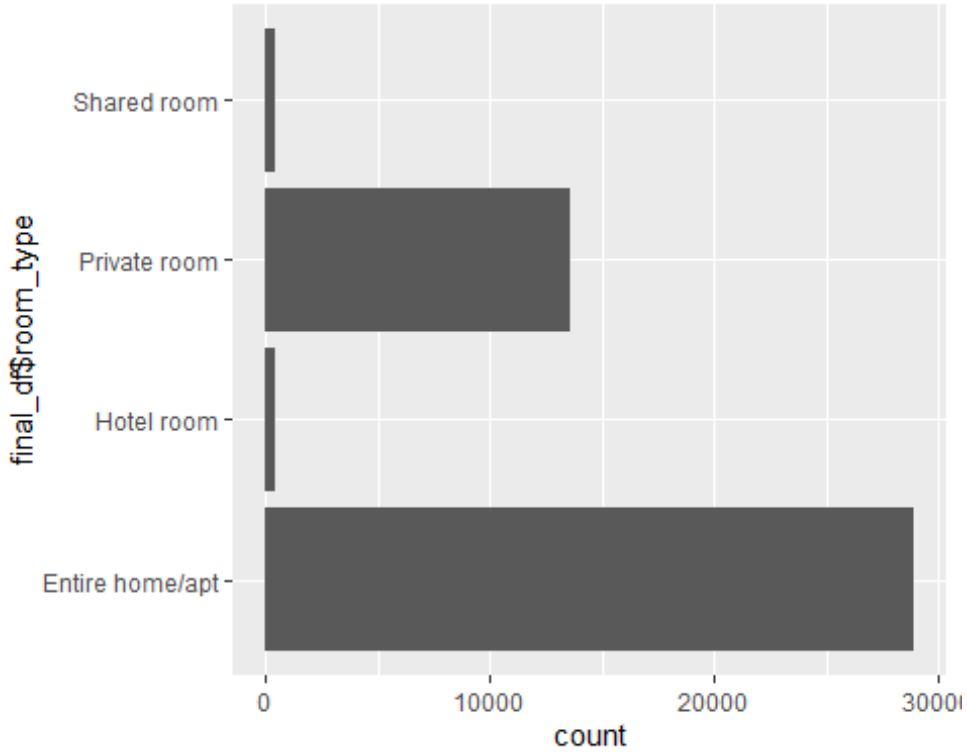
```
## Warning: Use of `final_df$airbnb_30_days_price` is discouraged. Use
## `airbnb_30_days_price` instead.
```




```
#The average airbnb price is ~ 4300 per month

# e) What type of houses are most rented on Airbnb?
ggplot(data=final_df,aes(y=final_df$room_type)) + geom_histogram(stat =
"count")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
## Warning: Use of `final_df$room_type` is discouraged. Use `room_type`
instead.
```




```
#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, airbnb_30_days_price, Average_Rent)

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

##   mean_airbnb_30_days_price
## 1                  4317.555

#Airbnb monrthly average rent is 4312.728

# 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)

## Warning: package 'caTools' was built under R version 4.0.5

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

summary(mymodel_1)

##
## Call:
```

```
## lm(formula = airbnb_30_days_price ~ neighbourhood, data = final_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -5137  -2417  -1217     71 294620
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       2640.00     673.33   3.921 8.84e-05
***
## neighbourhoodArmour Square         1348.21    1354.17   0.996 0.319450
## neighbourhoodAustin                 290.82     790.54   0.368 0.712972
## neighbourhoodAvalon Park            427.50    4855.42   0.088 0.929841
## neighbourhoodAvondale              1338.24     875.11   1.529 0.126217
## neighbourhoodBridgeport             716.79    1130.99   0.634 0.526236
## neighbourhoodBurnside              1335.00    6833.51   0.195 0.845111
## neighbourhoodCalumet Heights      -1054.50    2253.38  -0.468 0.639813
## neighbourhoodChatham               -460.71    1938.16  -0.238 0.812110
## neighbourhoodChicago Lawn         -1102.50    1828.55  -0.603 0.546553
## neighbourhoodDunning               -446.54    2002.64  -0.223 0.823556
## neighbourhoodEast Garfield Park    -340.53     742.80  -0.458 0.646638
## neighbourhoodEdgewater              664.59     749.78   0.886 0.375421
## neighbourhoodEnglewood               68.00    1106.39   0.061 0.950992
## neighbourhoodForest Glen           -275.00    2856.68  -0.096 0.923310
## neighbourhoodFuller Park            -20.00    3983.45  -0.005 0.995994
## neighbourhoodGage Park             -696.00    4353.25  -0.160 0.872976
## neighbourhoodGreater Grand Crossing -177.00   1412.38  -0.125 0.900270
## neighbourhoodHermosa               -558.95    2306.75  -0.242 0.808542
## neighbourhoodHumboldt Park          -81.18     700.82  -0.116 0.907787
## neighbourhoodHyde Park              109.58     834.72   0.131 0.895557
## neighbourhoodIrving Park            833.81    1024.72   0.814 0.415827
## neighbourhoodJefferson Park         321.82    1598.54   0.201 0.840450
## neighbourhoodKenwood               1641.43    1246.76   1.317 0.187994
## neighbourhoodLincoln Park          2797.50     863.74   3.239 0.001201
**
## neighbourhoodLincoln Square         652.42     890.03   0.733 0.463543
## neighbourhoodLogan Square          1576.96     680.79   2.316 0.020542 *
## neighbourhoodMontclare             -380.00    2572.77  -0.148 0.882580
## neighbourhoodMount Greenwood        -90.00    6833.51  -0.013 0.989492
## neighbourhoodNear West Side        1839.94     685.83   2.683 0.007304
**
## neighbourhoodNorth Center          2321.17     823.65   2.818 0.004832
**
## neighbourhoodNorth Lawndale       -1112.00     739.45  -1.504 0.132636
## neighbourhoodNorth Park             556.67    1263.00   0.441 0.659398
## neighbourhoodOakland               2955.00     890.73   3.318 0.000909
***
## neighbourhoodPortage Park           623.86     908.19   0.687 0.492134
## neighbourhoodPullman                -60.00    2253.38  -0.027 0.978758
## neighbourhoodRiverdale            -1510.00    5593.06  -0.270 0.787179
```

```
## neighbourhoodRogers Park               743.04     798.96   0.930 0.352373
## neighbourhoodRoseland                  188.57    1629.55   0.116 0.907875
## neighbourhoodSouth Shore              -276.26     810.33  -0.341 0.733160
## neighbourhoodWashington Park           413.64     800.13   0.517 0.605186
## neighbourhoodWest Elsdon               -75.00    6833.51  -0.011 0.991243
## neighbourhoodWest Garfield Park       -272.14    1629.55  -0.167 0.867367
## neighbourhoodWest Lawn               -1503.00    3114.81  -0.483 0.629430
## neighbourhoodWest Pullman              210.00    3983.45   0.053 0.957957
## neighbourhoodWest Town                2740.47     677.43   4.045 5.23e-05
***
## neighbourhoodWoodlawn                 -548.93     719.82  -0.763 0.445708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9617 on 43287 degrees of freedom
## Multiple R-squared:  0.01368,    Adjusted R-squared:  0.01263
## F-statistic: 13.05 on 46 and 43287 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
##  -4092  -2485  -1495     54 295756
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 195266.742 177135.974   1.102    0.270
## Zip_Code        -3.149      2.921  -1.078    0.281
##
## Residual standard error: 9713 on 42894 degrees of freedom
##   (438 observations deleted due to missingness)
## Multiple R-squared:  2.709e-05,  Adjusted R-squared:  3.779e-06
## F-statistic: 1.162 on 1 and 42894 DF,  p-value: 0.281
```

```
# 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
```