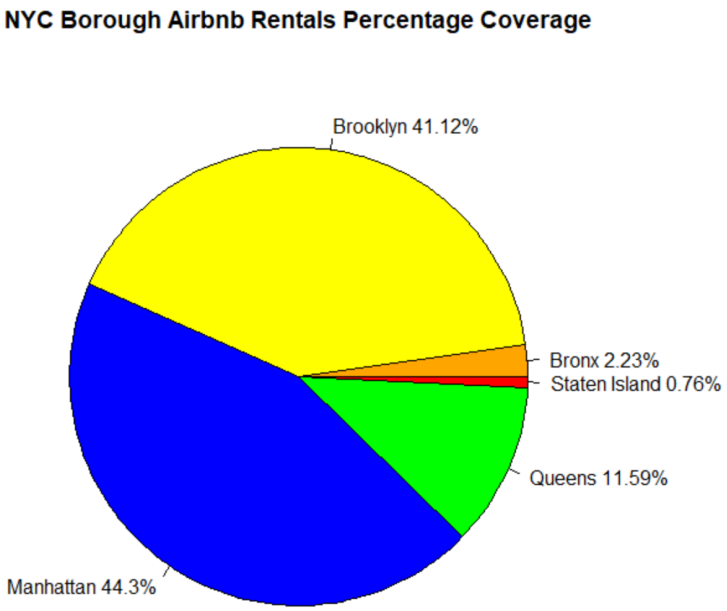
R_Console

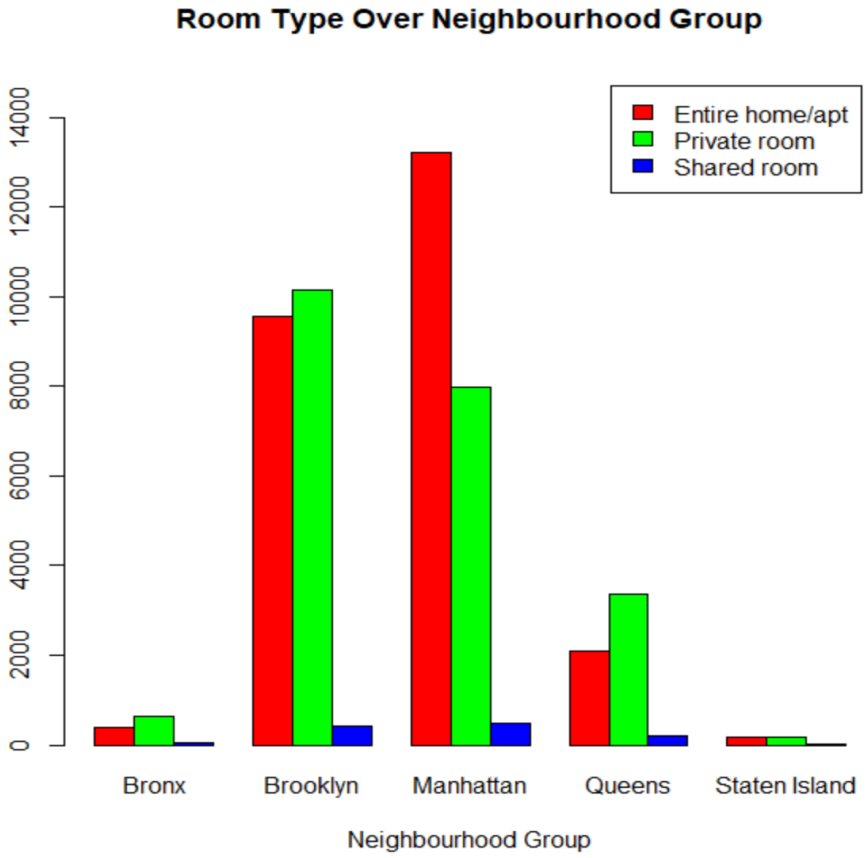
- This is Console file or Output for the R code
- Some calculation might differ than other calculation from final document
- For plotly graphs look up their individual file, and Graphs are below after respective section code is run.

```
> getwd()
[1] "C:/Users/Personal/Documents"
> setwd("Documents/Boston/")
> getwd()
[1] "C:/Users/Personal/Documents/Documents/Boston"
> df = read.csv("AB_NYC_2019.csv")
> head(df)
    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
> colnames(df)
 [1] "id"                                "name"
 [3] "host_id"                           "host_name"
 [5] "neighbourhood_group"               "neighbourhood"
 [7] "latitude"                          "longitude"
 [9] "room_type"                         "price"
[11] "minimum_nights"                    "number_of_reviews"
[13] "last_review"                       "reviews_per_month"
[15] "calculated_host_listings_count" "availability_365"
> na.cols = c()
> # This is to check weather we have any columns with NA
> for (i in colnames(df)) {
+   if(any(is.na(df[i]))){
+     na.cols=c(i)
+     }
+ }
```

```
> na.cols
[1] "reviews_per_month"
> # It gives one col that has NA's in it and it is reviews_per_month, which we can
compute it
> #   from reviews by dividing with 12
> df$reviews_per_month = df$number_of_reviews/12
> any(is.na(df$reviews_per_month))
[1] FALSE
> # Hence we filtered our data with NA's
> # Now by looking at data we can filter columns so lets make data set just for col
umns we need
> col2keep = c("name","host_id","host_name","neighbourhood_group","neighbourhood",
+                "latitude","longitude","room_type","price","minimum_nights",
+                 "number_of_reviews","reviews_per_month")
> df = df[,col2keep]
> head(df)
                                              name host_id   host_name
1               Clean & quiet apt home by the park    2787        John
2                            Skylit Midtown Castle    2845    Jennifer
3              THE VILLAGE OF HARLEM....NEW YORK !    4632   Elisabeth
4                  Cozy Entire Floor of Brownstone    4869 LisaRoxanne
5 Entire Apt: Spacious Studio/Loft by central park    7192       Laura
6        Large Cozy 1 BR Apartment In Midtown East    7322       Chris
  neighbourhood_group neighbourhood latitude longitude       room_type price
1            Brooklyn    Kensington 40.64749 -73.97237    Private room   149
2           Manhattan       Midtown 40.75362 -73.98377 Entire home/apt   225
3           Manhattan        Harlem 40.80902 -73.94190    Private room   150
4            Brooklyn  Clinton Hill 40.68514 -73.95976 Entire home/apt    89
5           Manhattan   East Harlem 40.79851 -73.94399 Entire home/apt    80
6           Manhattan   Murray Hill 40.74767 -73.97500 Entire home/apt   200
  minimum_nights number_of_reviews reviews_per_month
1              1                9          0.750000
2              1               45          3.750000
3              3                0          0.000000
4              1              270         22.500000
5             10                9          0.750000
6              3               74          6.166667
> ### Descreptive analysis
> # We will analyze the nighbourhood group as our categorieal varible
> x = table(df$neighbourhood_group)
> x

       Bronx      Brooklyn     Manhattan        Queens Staten Island
        1091         20104         21661          5666           373
>
> # Lets see their respective percentage and as seen we expect people use Airbnb mo
re in Manhattan
> prop.table(x)

       Bronx      Brooklyn     Manhattan        Queens Staten Island
 0.022313120   0.411166786   0.443010533   0.115880969   0.007628592
>
> # It looks Brooklyn is not laggig behind with much wheare as Staten Island is out
of game
> # Let describe with pie chart
> y = paste(names(x),paste(round(prop.table(x)*100,2),"%",sep = ""))
> pie(x, main = "NYC Borough Airbnb Rentals Percentage Coverage",labels = y, col=c(
"Orange","Yellow","Blue","Green","Red"))
>
```

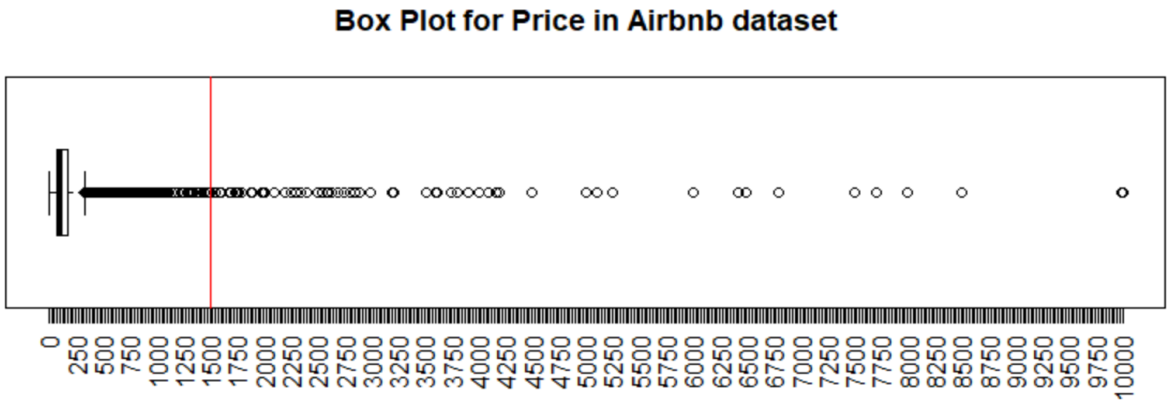


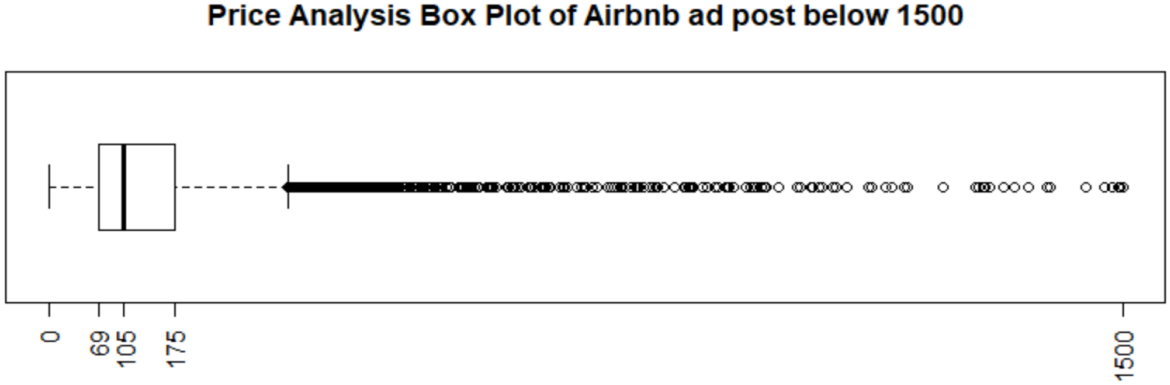

```
>
> # SO let also see what are most types of properties are in Airbnb in respective n
eighbour hood group
> y = table(df$room_type,df$neighbourhood_group)
> y.prop = prop.table(y)
> addmargins(y)

                 Bronx Brooklyn Manhattan Queens Staten Island   Sum
  Entire home/apt   379     9559     13199   2096           176 25409
  Private room      652    10132      7982   3372           188 22326
  Shared room        60      413       480    198             9  1160
  Sum              1091    20104     21661   5666           373 48895
> addmargins(y.prop)*100

                       Bronx     Brooklyn    Manhattan       Queens Staten Island
  Entire home/apt   0.77513038  19.55005624  26.99458022   4.28673689    0.35995501
  Private room      1.33346968  20.72195521  16.32477758   6.89641068    0.38449739
  Shared room       0.12271193   0.84466714   0.98169547   0.40494938    0.01840679
  Sum               2.23131200  41.11667860  44.30105328  11.58809694    0.76285919

                         Sum
  Entire home/apt  51.96645874
  Private room     45.66111054
  Shared room       2.37243072
  Sum             100.00000000
> barplot(y, xlab = "Neighbourhood Group", beside = TRUE, legend.text = TRUE,ylim =
c(0,15000),
+         main = "Room Type Over Neighbourhood Group",col=c("red", "green", "blue")
)
>
```




```
> # As we knew Manhatten would have more Airbnb rentals, where as bronx and Staten
Island doesn't
> # have good enough amount of rentals.
> # As expected from the idea of Airbnb people will rent rooms more than a whole ap
t/house which is
> # completely seen in our data, but I am surprised to see different to that in Man
hatten.
> # Now we would have thought since Manhattan is so expensive people would rent pri
vate room or
> # shared room but from our data points which is completely different than our exp
ectations.
> # Although now we know that if someone is going to rent on Airbnb in queens they
are likely to go for
> # privet rooom and entire apt/house in Manhattan, we can't say anything sure abou
t Brooklyn.
> # However second choice for NYC Airbnb booking will be Brooklyn over Queens for M
anhatten
>
>
>
>
> par(mfrow=c(2,1))
> # Lets see how would box plot look for the prices per night in nyc
>
>
> f = fivenum(df$price)
> IQR = f[4]-f[2]
> IQR
```

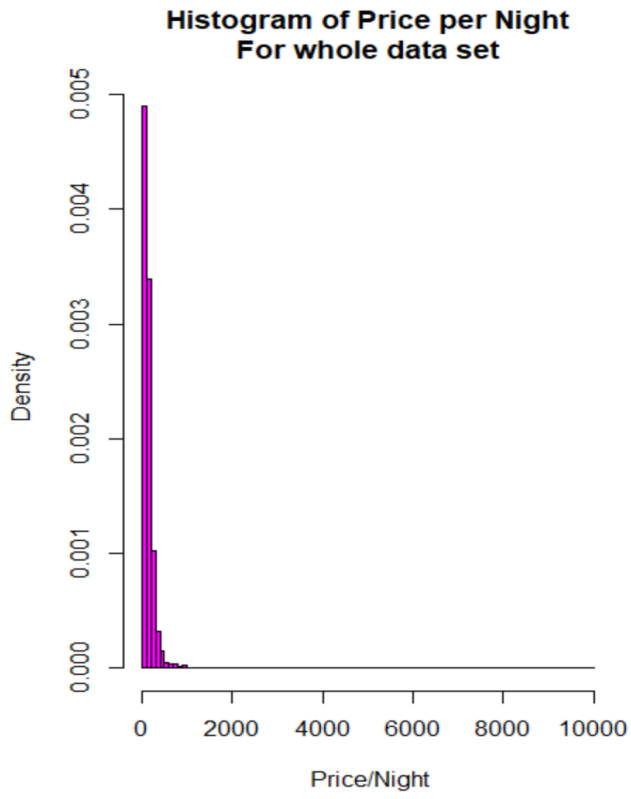
```
[1] 106
> summary(df$price)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0    69.0   106.0   152.7   175.0 10000.0
> # So Mean is 152.7 and Median in 106 and Inter Quaraltile Range is 106
>
> # No of 0's
> sum(df$price==0)
[1] 11
>
> boxplot(df$price, horizontal = TRUE, main = "Box Plot for Price in Airbnb dataset
",xaxt = "n")
> axis(side = 1, at = seq(0,max(df$price)+10,25), labels = TRUE,  las=2)
> abline(v=1500,col="RED")
>
> # As it seems there way more outlier in our data and there are many but since the
re many points
> #  together below 1500 and it's darker line till 1500 so we will still considered
them valid
>
> # Also let's see how would box plot look without over 1500
> price_below_1500 = subset(df, df$price <= 1500)
> boxplot(price_below_1500$price, horizontal = TRUE, xaxt = "n", main = "Price Anal
ysis Box Plot of Airbnb ad post below 1500")
> axis(side = 1, at = fivenum(price_below_1500$price), labels = TRUE,  las=2)
>
```

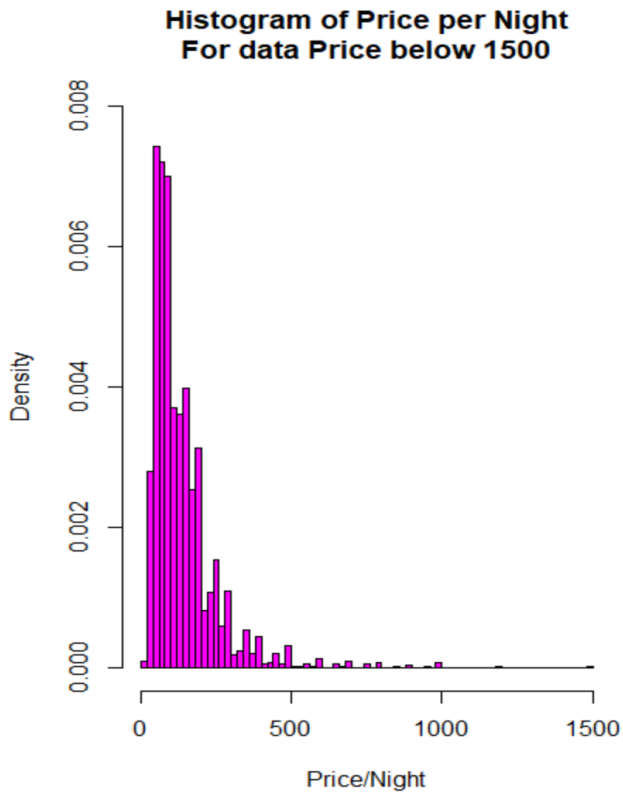




```
> #   Over 1500 we need to study more, why are they so high, is the data set wrong?
> df_price_1500 = subset(df,df$price>1500)
> nrow(df_price_1500)
[1] 139
> table(df_price_1500$neighbourhood_group)

        Bronx      Brooklyn     Manhattan        Queens Staten Island
            1            34            97             6             1
> # As it's quite often that manhatten and brooklyn can cost more beacuase of manha
tten parties
> #   However, staten_island, bronx and queens will be intersting to look
> # let's see why they cost more
> df_price_1500[df_price_1500$neighbourhood_group %in% c("Bronx","Queens","Staten I
sland"),]
                                        name   host_id host_name neighbourhood_grou
p
9152     Furnished room in Astoria apartment  20582832  Kathrine              Queen
s
14381 Mins away to Manhattan Suite Residence  24146326    Julien              Queen
s
17812           Gorgeous 2 Bedroom apartment  41870118     Iveta              Queen
s
22354                Victorian Film location   2675644    Alissa       Staten Islan
d
24478                "The luxury of Comfort" 131826530     Kathy               Bron
x
42681          Majestic Mansion LifeStyle :)  74373729      Shah              Queen
s
44430     Room with sofa bed or air mattress   9295237    Noelle              Queen
s
47351                        wait until later  35741633      Chen              Queen
s
      neighbourhood latitude longitude       room_type price minimum_nights
9152          Astoria 40.76810 -73.91651    Private room 10000             100
14381         Astoria 40.76626 -73.93054     Shared room  1800               3
17812     Forest Hills 40.72064 -73.83746 Entire home/apt  2350             365
22354    Randall Manor 40.63952 -74.09730 Entire home/apt  5000               1
24478        Riverdale 40.88671 -73.91510    Private room  2500               2
42681          Bayside 40.77811 -73.77069 Entire home/apt  2600               6
44430          Astoria 40.75593 -73.91276    Private room  2000             365
47351 Long Island City 40.74869 -73.94294 Entire home/apt  2000               1
      number_of_reviews reviews_per_month
9152                  2         0.1666667
14381                 5         0.4166667
17812                 0         0.0000000
22354                 0         0.0000000
24478                 0         0.0000000
42681                 3         0.2500000
44430                 0         0.0000000
47351                 0         0.0000000
>
> # As it seems they are entire home/apt we can expect someone will pay more than 1
500 if it's
> #  some important occation but it's still highly doubtable about shared room or p
rivate room.
> # Let also look on top 10 expensive stays
> df_price_1500[order(df_price_1500$price,decreasing = TRUE)[1:10],]
                                                 name   host_id host_name
9152                Furnished room in Astoria apartment  20582832  Kathrine
17693    Luxury 1 bedroom apt. -stunning Manhattan views   5143901      Erin
29239                                1-BR Lincoln Center  72390391    Jelena
6531                                  Spanish Harlem Apt   1235070     Olson
12343               Quiet, Clean, Lit @ LES & Chinatown   3906464       Amy
40434 2br - The Heart of NYC: Manhattans Lower East Side   4382127      Matt
```

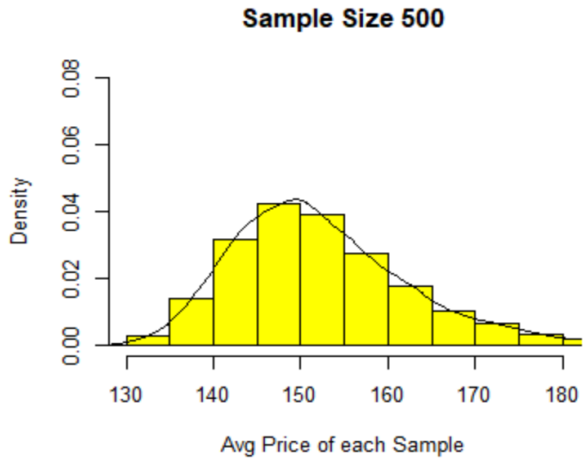
```
30269  Beautiful/Spacious 1 bed luxury flat-TriBeCa/Soho  18128455       Rum
4378                                      Film Location   1177497   Jessica
29663           East 72nd Townhouse by (Hidden by Airbnb) 156158778     Sally
42524                70' Luxury MotorYacht on the Hudson   7407743      Jack
      neighbourhood_group     neighbourhood latitude longitude       room_type pric
e
9152                Queens           Astoria 40.76810 -73.91651    Private room 1000
0
17693            Brooklyn        Greenpoint 40.73260 -73.95739 Entire home/apt 1000
0
29239           Manhattan   Upper West Side 40.77213 -73.98665 Entire home/apt 1000
0
6531            Manhattan       East Harlem 40.79264 -73.93898 Entire home/apt  999
9
12343           Manhattan   Lower East Side 40.71355 -73.98507    Private room  999
9
40434           Manhattan   Lower East Side 40.71980 -73.98566 Entire home/apt  999
9
30269           Manhattan           Tribeca 40.72197 -74.00633 Entire home/apt  850
0
4378             Brooklyn      Clinton Hill 40.69137 -73.96723 Entire home/apt  800
0
29663           Manhattan   Upper East Side 40.76824 -73.95989 Entire home/apt  770
3
42524           Manhattan Battery Park City 40.71162 -74.01693 Entire home/apt  750
0
      minimum_nights number_of_reviews reviews_per_month
9152              100                2        0.16666667
17693               5                5        0.41666667
29239              30                0        0.00000000
6531                5                1        0.08333333
12343              99                6        0.50000000
40434              30                0        0.00000000
30269              30                2        0.16666667
4378                1                1        0.08333333
29663               1                0        0.00000000
42524               1                0        0.00000000
> # As it seems they are more premium places but it's stll highly doubtable to pay
that much. Though
> #   it is intersting to learn that people are posting ads with high prices.
>
>
> # Conclusion
> # Looking at all the calculations, I think there is quite missing or incorrect ob
servation
> # in the data. Although R calculation might say they are outliears, we do not hav
e any evidence
> # to tell. Since living in NY I know it is not impossible for someone to rent at
3000 or 5000
> # in Manhatten. Hence let's consider this data not corrupt.
>
>
>
> par(mfrow = c(1,2))
> hist(df$price,probability = TRUE,main= "Histogram of Price per Night\nFor whole d
ata set",
+           xlab = " Price/Night",breaks=100,col = "Magenta")
> hist(price_below_1500$price,main= "Histogram of Price per Night\nFor data Price b
elow 1500",
+           xlab = "Price/Night", ylim= c(0,0.008),probability = TRUE, breaks = 100
,col = "Magenta")
>
```

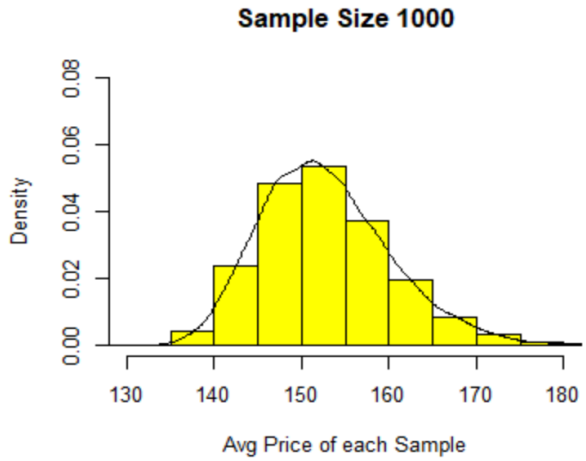


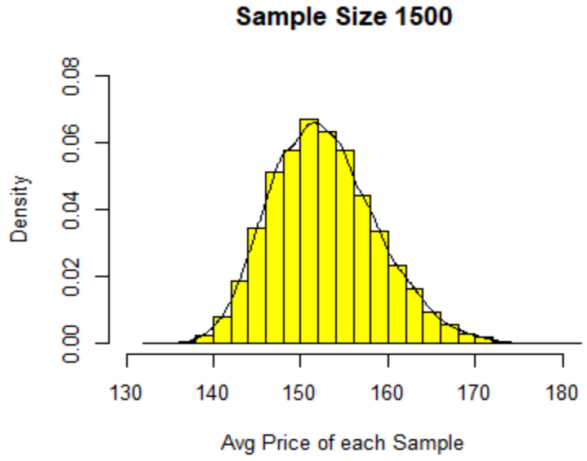


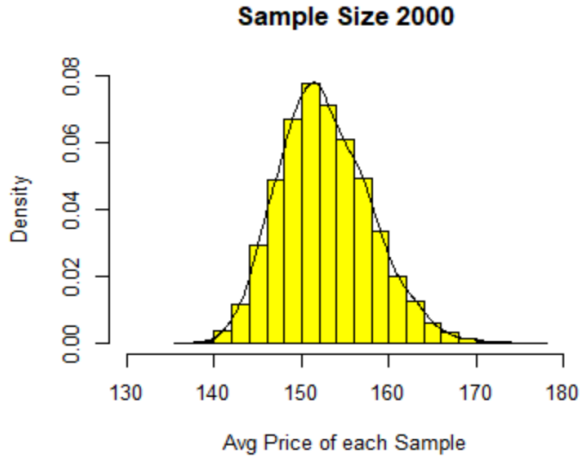

```
> mean.price = mean(df$price)
> sd.price = sd(df$price)
>
> summary(df$price)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0    69.0   106.0   152.7   175.0 10000.0
> sd.price
[1] 240.1542
> summary(price_below_1500$price)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0    69.0   105.0   143.6   175.0  1500.0
> sd(price_below_1500$price)
[1] 127.8256
> # As it is seen the data is more skewed towards right. Hence it is Right skewed d
istribution
> # That's why mean is greather than median. As for calculation purpose it is bette
r to work with
> # median here.
>
>
> #### Central Limit Theorem
> par(mfrow =c(2,2))
> samples = 10000
> for (size in c(500,1000,1500,2000)){
+   xbar <- numeric(samples)
+     for (i in 1: samples) {
+       xbar[i] <- mean(sample(df$price,size))
+     }
+   hist(xbar,prob = TRUE, col = "Yellow", ylim = c(0,0.08),xlim = c(130,180),
+        main = paste("Sample Size",size), xlab = "Avg Price of each Sample", break
s = 20)
+   lines(density(xbar))
```

```
+
+    cat("Sample Size = ", size, " Mean = ", mean(xbar),
+        " SD = ", sd(xbar), "\n")
+ }
Sample Size =  500  Mean =  152.5934  SD =  10.70393
Sample Size =  1000  Mean =  152.8188  SD =  7.575459
Sample Size =  1500  Mean =  152.7379  SD =  6.095657
Sample Size =  2000  Mean =  152.6865  SD =  5.279374
>
```










```
> head(df)
                                              name host_id   host_name
1               Clean & quiet apt home by the park    2787        John
2                            Skylit Midtown Castle    2845    Jennifer
3              THE VILLAGE OF HARLEM....NEW YORK !    4632   Elisabeth
4                  Cozy Entire Floor of Brownstone    4869 LisaRoxanne
5 Entire Apt: Spacious Studio/Loft by central park    7192       Laura
6        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
```

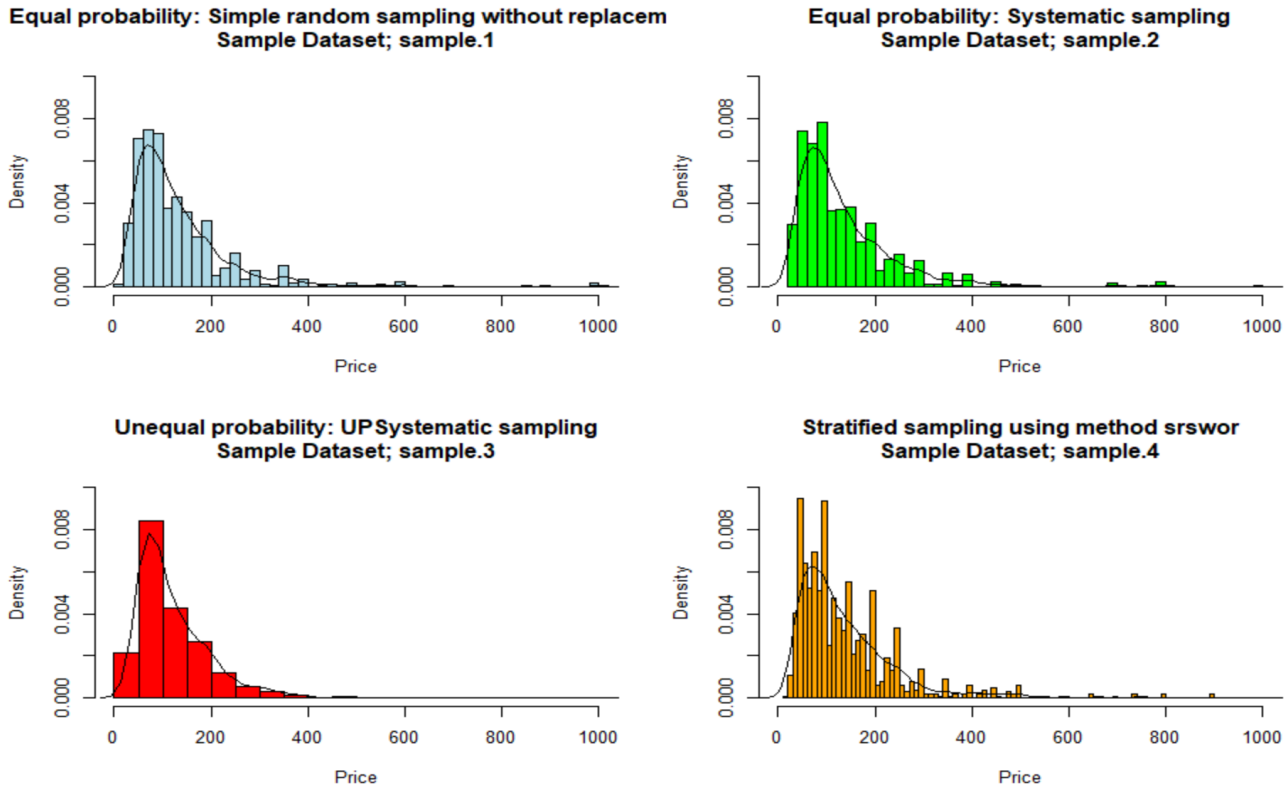
```
  minimum_nights number_of_reviews reviews_per_month
1              1                 9          0.750000
2              1                45          3.750000
3              3                 0          0.000000
4              1               270         22.500000
5             10                 9          0.750000
6              3                74          6.166667
>
>
> #### Sampling Methods
> library(sampling)
> library(prob)
Loading required package: combinat

Attaching package: 'combinat'

The following object is masked from 'package:utils':

    combn

Loading required package: fAsianOptions
Loading required package: timeDate
Loading required package: timeSeries
Loading required package: fBasics
Loading required package: fOptions

Attaching package: 'prob'

The following objects are masked from 'package:base':

    intersect, setdiff, union

> name_sample = c()
> set.seed(123)
>
> # SRSWOR
> # Equal Probability
> s = srswor(1000,nrow(df))
> sample.1 = df[as.logical(s),]
> head(sample.1)
                                                 name host_id        host_name
31                             front room/double bed   32294  Ssameer Or Trip
41                               ENJOY Downtown NYC!   46978            Edward
76            Charming East Village One Bedroom Flat   69829              Josh
79                     Fort Greene Retreat on the Park   71512           Blaise
232 Artistic, Cozy, and Spacious w/ Patio! Sleeps 5  186084 Ricardo & Ashlie
242                  Colorful Private One Bedroom Apt  295760             Greta
    neighbourhood_group neighbourhood latitude longitude       room_type price
31            Manhattan        Harlem 40.82245 -73.95104    Private room    50
41            Manhattan  East Village 40.72290 -73.98199    Private room    68
76            Manhattan  East Village 40.72828 -73.98801 Entire home/apt   190
79             Brooklyn   Fort Greene 40.69320 -73.97267    Private room    95
232           Manhattan     Chinatown 40.71756 -73.99503 Entire home/apt   250
242           Manhattan  Little Italy 40.71961 -73.99540 Entire home/apt   135
    minimum_nights number_of_reviews reviews_per_month
31               3               242          20.16667
41               2               245          20.41667
76               5                21           1.75000
79               3               143          11.91667
232              4                18           1.50000
242              2                21           1.75000
> name_sample = c("Equal probability: Simple random sampling without replacement")
>
> # Systematic Sampling
```

```
> set.seed(151)
> N = nrow(df)
> n = 1000
> k = ceiling(N / n)
> r <- sample(k, 1)
> r
[1] 13
> s = seq(r, by = k, length = n)
> sample.2 = df[s, ]
> tail(sample.2)
                                                    name   host_id host_name
48719             Nice bedroom - 3 Stops to Times Square 268796947   Derreck
48768 Romantic studio in New York artistic best Bushwick 273849259 Alexander
48817         Entire first floor apartment in Park Slope  71142174   Toniann
48866                       1 bedroom in sunlit apartment  99144947    Brenda
NA                                                 <NA>        NA      <NA>
NA.1                                               <NA>        NA      <NA>
      neighbourhood_group    neighbourhood latitude longitude       room_type price
48719              Queens Long Island City 40.75255 -73.93128    Private room    89
48768            Brooklyn         Bushwick 40.69958 -73.92772 Entire home/apt   120
48817            Brooklyn      Sunset Park 40.66266 -73.98908 Entire home/apt   100
48866           Manhattan           Inwood 40.86845 -73.92449    Private room    80
NA                   <NA>             <NA>       NA        NA            <NA>    NA
NA.1                 <NA>             <NA>       NA        NA            <NA>    NA
      minimum_nights number_of_reviews reviews_per_month
48719              1                 0                 0
48768              1                 0                 0
48817              4                 0                 0
48866              1                 0                 0
NA                NA                NA                NA
NA.1              NA                NA                NA
> # Hence we got some NA we need to change that
>
> extra = sample((1:nrow(df))[-s], 2)
> # Now we can omit NA nd add last two rows
> new_s =  sort(c(s[s<= nrow(df)],extra))
> sample.2 = df[new_s,]
> tail(sample.2)
                                                    name   host_id host_name
48621        SUPER COZY 2 BEDS IN BEST PART OF CHELSEA!!  91268177   Beatriz
48670                                  Harman st Loft 273546395     Allen
48719             Nice bedroom - 3 Stops to Times Square 268796947   Derreck
48768 Romantic studio in New York artistic best Bushwick 273849259 Alexander
48817         Entire first floor apartment in Park Slope  71142174   Toniann
48866                       1 bedroom in sunlit apartment  99144947    Brenda
      neighbourhood_group    neighbourhood latitude longitude       room_type price
48621           Manhattan          Chelsea 40.74480 -74.00761 Entire home/apt   299
48670            Brooklyn         Bushwick 40.69980 -73.91913 Entire home/apt   165
48719              Queens Long Island City 40.75255 -73.93128    Private room    89
48768            Brooklyn         Bushwick 40.69958 -73.92772 Entire home/apt   120
48817            Brooklyn      Sunset Park 40.66266 -73.98908 Entire home/apt   100
48866           Manhattan           Inwood 40.86845 -73.92449    Private room    80
      minimum_nights number_of_reviews reviews_per_month
48621              3                 0                 0
48670              3                 0                 0
48719              1                 0                 0
48768              1                 0                 0
48817              4                 0                 0
48866              1                 0                 0
> name_sample = c(name_sample ,"Equal probability: Systematic sampling")
>
>
> # UPsystematic
> # Using the probabilities of reviews to determine our sample
```

```
> pik = inclusionprobabilities(df$number_of_reviews, 1000)
Warning message:
In inclusionprobabilities(df$number_of_reviews, 1000) :
  there are zero values in the initial vector a

> length(pik)
[1] 48895
> sum(pik)
[1] 1000
> s = UPsystematic(pik)
> sample.3 = df[as.logical(s),]
> tail(sample.3)
                                                    name   host_id  host_name
44556     Cosy shiny bedroom close to Manhattan 25min 254104585       Mila
44894 Quiet Room Next to Times Square and Bryant Park 260191397 Hotel Mela
45408                      NEW! Chic Designer Vanilla 259468466       Jack
45893 Amazing 3bd/2ba Town-Home In Prime Williamsburg  16625273    Richard
46431                   Spacious 2Bedroom Village Home 265248832        Noe
47323                     The Outback Private's Escape 175730239  Baboucarr
      neighbourhood_group      neighbourhood latitude longitude       room_type
44556            Brooklyn Bedford-Stuyvesant 40.68316 -73.92870    Private room
44894           Manhattan   Theater District 40.75745 -73.98596    Private room
45408           Manhattan    Lower East Side 40.71461 -73.98755    Private room
45893            Brooklyn       Williamsburg 40.71490 -73.94585 Entire home/apt
46431           Manhattan  Greenwich Village 40.72757 -74.00059 Entire home/apt
47323              Queens          Sunnyside 40.73902 -73.92686    Private room
      price minimum_nights number_of_reviews reviews_per_month
44556    70              2                12         1.0000000
44894   100              1                 7         0.5833333
45408    89              4                 4         0.3333333
45893   295              2                 2         0.1666667
46431   349              2                 2         0.1666667
47323    59              3                 2         0.1666667
> name_sample = c(name_sample ,"Unequal probability: UPSystematic sampling")
>
> ## Stratified sampling
> set.seed(146)
> df <- df[order(df$neighbourhood_group), ]
> st.size = 1000*prop.table(table(df$neighbourhood_group))
>
> st = strata(df, stratanames = c("neighbourhood_group"),
+               size = st.size, method = "srswor",
+               description = TRUE)
Stratum 1

Population total and number of selected units: 1091 22.31312
Stratum 2

Population total and number of selected units: 20104 411.1668
Stratum 3

Population total and number of selected units: 21661 443.0105
Stratum 4

Population total and number of selected units: 5666 115.881
Stratum 5

Population total and number of selected units: 373 7.628592
Number of strata  5
Total number of selected units 1000
> head(st)
      neighbourhood_group ID_unit       Prob Stratum
6                   Bronx       6 0.02045199       1
95                  Bronx      95 0.02045199       1
```

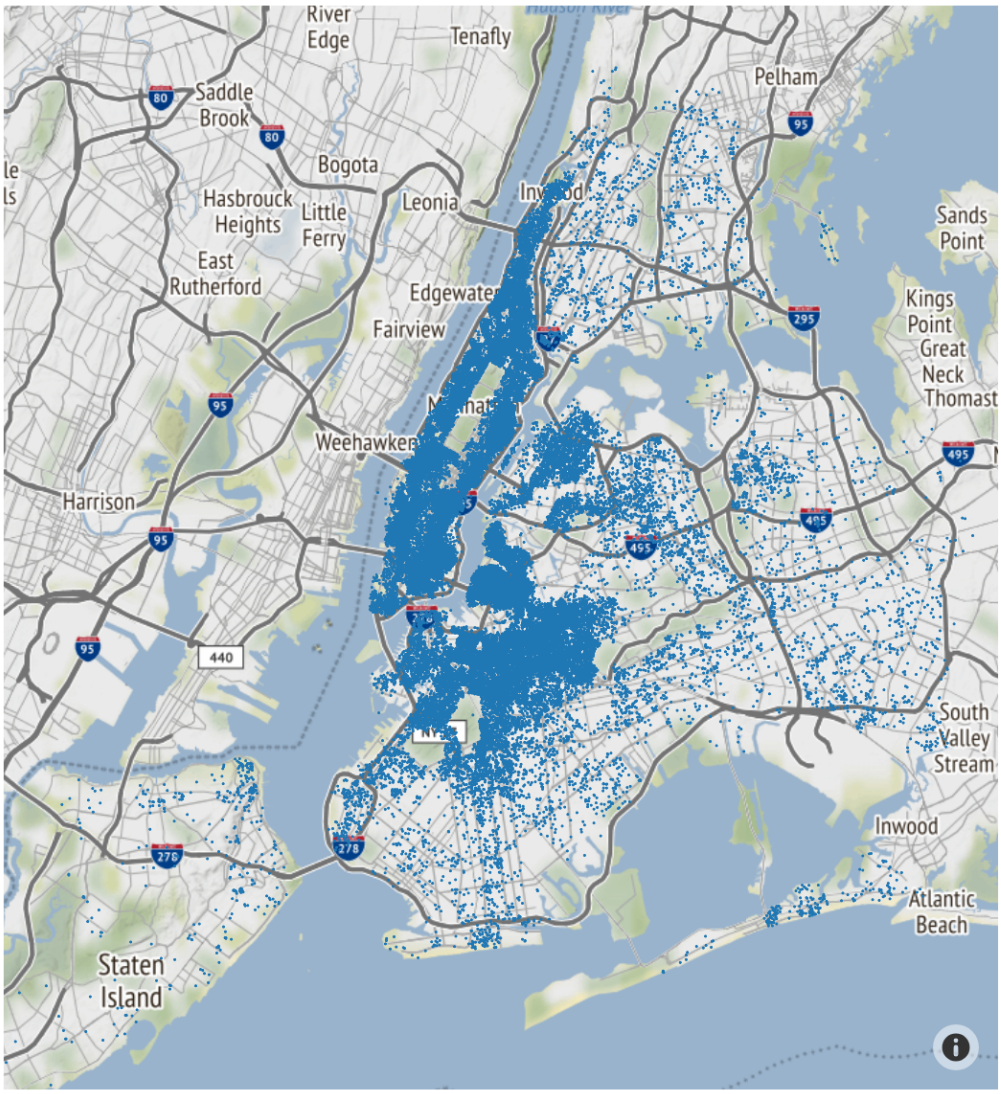
```
207                 Bronx     207 0.02045199       1
216                 Bronx     216 0.02045199       1
222                 Bronx     222 0.02045199       1
304                 Bronx     304 0.02045199       1
448                 Bronx     448 0.02045199       1
516                 Bronx     516 0.02045199       1
536                 Bronx     536 0.02045199       1
643                 Bronx     643 0.02045199       1
664                 Bronx     664 0.02045199       1
690                 Bronx     690 0.02045199       1
> sample.4 = getdata(df,st)
> head(sample.4)
                                                    name  host_id
434   Artsy 1 bedroom Apt. 20 min to 42nd Grand Central!   716306
6933                          Bronx, 1Bdrm in 3Bdrm Apt. 25605654
16327                             Uptown Bronx Apartment 73126926
16705   Budget Room in large apartment (Solo Travelers+) 11070120
16939                         cosy room with all confort 51329030
21298   Large *COZY* Private Bedroom near Yankee Stadium 67130668
                  host_name     neighbourhood latitude longitude       room_type
434   Dee, Dre & Mama Shelley          Woodlawn 40.89747 -73.86390 Entire home/apt
6933                  Cindy       Morris Park 40.85465 -73.85669    Private room
16327       Real Estate To Go    Williamsbridge 40.88493 -73.86180 Entire home/apt
16705                Vernon           Fordham 40.85927 -73.90142    Private room
16939                Zakaria          Longwood 40.82716 -73.89911    Private room
21298                Melissa Concourse Village 40.82692 -73.92183    Private room
      price minimum_nights number_of_reviews reviews_per_month neighbourhood_group
434      77              1               197        16.4166667               Bronx
6933     50              1                 2         0.1666667               Bronx
16327    67              3               205        17.0833333               Bronx
16705    40             21                23         1.9166667               Bronx
16939    30              1                11         0.9166667               Bronx
21298    45              5                46         3.8333333               Bronx
      ID_unit       Prob Stratum
434         6 0.02045199       1
6933       95 0.02045199       1
16327     207 0.02045199       1
16705     216 0.02045199       1
16939     222 0.02045199       1
21298     304 0.02045199       1
> name_sample = c(name_sample ,"Stratified sampling using method srswor")
> prop.table(table(sample.4$neighbourhood_group))

       Bronx      Brooklyn     Manhattan        Queens Staten Island
  0.022044088   0.411823647   0.443887776   0.115230461   0.007014028
> prop.table(table(df$neighbourhood_group))

       Bronx      Brooklyn     Manhattan        Queens Staten Island
  0.022313120   0.411166786   0.443010533   0.115880969   0.007628592
> # As we can see that propotion are same
>
> library(stringr)
> ans = str_c("Sample Dataset; Original","   Mean = ",round(mean(df$price),3),"   S
D = ", sd(df$price),
+               "   Min = ",min(df$price),"   Max = ",max(df$price))
>
> par(mfrow =c(2,2))
> #1
> hist(sample.1$price,prob = TRUE, col = "Light Blue", main = paste(name_sample[1],
"Sample Dataset; sample.1",sep = "\n"),
+      ylim = c(0,0.01),xlim= c(0,1000),xlab = "Price", breaks = 200)
> lines(density(sample.1$price))
> ans = c(ans,str_c("Sample Dataset; sample.1", "   Mean = ", round(mean(sample.1$p
rice),3),  "   SD = ", sd(sample.1$price),
```

```
+                   "   Min = ",min(sample.1$price),"   Max = ",max(sample.1$price)
))
>
> #2
> hist(sample.2$price,prob = TRUE, col = "green", main = paste(name_sample[2],"Samp
le Dataset; sample.2",sep = "\n"),
+      ylim = c(0,0.01),xlim= c(0,1000),xlab = "Price", breaks = 200)
> lines(density(sample.2$price))
> ans = c(ans , str_c("Sample Dataset; sample.2", "   Mean = ", round(mean(sample.2
$price),3),   "   SD = ", sd(sample.2$price),
+                     "   Min = ",min(sample.2$price),"   Max = ",max(sample.2$pric
e)))
>
> #3
> hist(sample.3$price,prob = TRUE, col = "Red", main = paste(name_sample[3],"Sample
Dataset; sample.3",sep = "\n"),
+      ylim = c(0,0.01),xlim= c(0,1000),xlab = "Price", breaks = 200)
> lines(density(sample.3$price))
> ans = c(ans , str_c("Sample Dataset; sample.3", "   Mean = ", round(mean(sample.3
$price),3),   "   SD = ", sd(sample.3$price),
+                     "   Min = ",min(sample.3$price),"   Max = ",max(sample.3$pric
e)))
>
> #4
> hist(sample.4$price,prob = TRUE, col = "orange", main = paste(name_sample[4],"Sam
ple Dataset; sample.4",sep = "\n"),
+      ylim = c(0,0.01),xlim= c(0,1000),xlab = "Price", breaks = 200)
> lines(density(sample.4$price))
> ans = c(ans , str_c("Sample Dataset; sample.4", "   Mean = ", round(mean(sample.4
$price),3),   "   SD = ", sd(sample.4$price),
+                     "   Min = ",min(sample.4$price),"   Max = ",max(sample.4$pric
e)))
>
```



```
> ans[1:5]
[1] "Sample Dataset; Original   Mean = 152.721   SD = 240.154169747188   Min = 0
Max = 10000"
[2] "Sample Dataset; sample.1   Mean = 154.152   SD = 229.584267387449   Min = 0
Max = 4000"
[3] "Sample Dataset; sample.2   Mean = 151.591   SD = 206.126888136927   Min = 20
Max = 3750"
[4] "Sample Dataset; sample.3   Mean = 142.907   SD = 346.6852997288   Min = 19   M
ax = 10000"
[5] "Sample Dataset; sample.4   Mean = 151.741   SD = 165.702387267183   Min = 10
Max = 2545"
>
>
>
>
> ########### Extra credit
> # Actually it really took me 3 days to figure out this
> library(plotly)
Loading required package: ggplot2
Want to understand how all the pieces fit together? See the R for Data
Science book: http://r4ds.had.co.nz/

Attaching package: ‘plotly’

The following object is masked from ‘package:ggplot2’:

    last_plot

The following object is masked from ‘package:timeSeries’:

    filter

The following object is masked from ‘package:stats’:

    filter

The following object is masked from ‘package:graphics’:

    layout

> library(mvtnorm)
> fig <- plot_ly(
+   fill = "toself",
+   lon = df$longitude,
+   lat = df$latitude,
+   type = 'scattermapbox',
+   text = paste("Price : ",df$price),
+   marker = list(size = 2,color = "Light Blue"),
+   fillcolor = 'color'
+   )
> fig <- fig %>%
+   layout(
+     mapbox = list(
+       style = "stamen-terrain",
+       center = list(lon = mean(df$longitude), lat = mean(df$latitude)),
+       zoom = 9.5),
+     showlegend = TRUE)
>
> fig
>>>>>>> Lookup it’s individual file attached as well,

No scattermapbox mode specifed:
  Setting the mode to markers
  Read more about this attribute -> https://plot.ly/r/reference/#scatter-mode
```

2019 Airbnb Listing's rental precise location with Price




```
>
> # heat map and 2d histogram
> s <- matrix(c(1, -.75, -.75, 1), ncol = 2)
> obs <- mvtnorm::rmvnorm(500, sigma = s)
> fig <- plot_ly(y = df$latitude, x = df$longitude) %>%
+   layout(title= "Properties Location of Airbnb ad post alongside heatmap in inter
active plotly plots")
> fig2 <- subplot(
+   fig %>% add_markers(alpha = 0.5),
+   fig %>% add_histogram2d(colorscale = 'YlOrRd')
+ )
>
> fig2
```

>>>>>> Lookup it's individual file attached