Singapore Airbnb

David Santos

15/01/2022

Abstract

Airbnb Inc. is an American company that operates an online marketplace for lodging, primarily homestays for vacation rentals, and tourism activities since 2008, giving people a different experience of the world while travelling. Airbnb has became a must in our society and is recognized all over the world. There are millions of listings, an those listings generate a lot of data that can be analysed and used for many differenty purposes.

Data Source

The data set has around 7,900 observations of 16 variables (columns) and was obtained from Keggle (https://www.kaggle.com/jojoker/singapore-airbnb)

Acquaring and Loading Data aswell as setting up our environment

```
library(grid)
library(pacman)
library(patchwork)
library(png)
library(tidyverse)
```

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

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

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

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)

# Loading the data set.

dataset <- read.csv('C:\\Users\\Dacs\\Documents\\R Working Directory\\Singapore Airbnb Analysis\\listings.csv')
```

Understanding and Cleaning Data

```
nrow(dataset)
```

```
## [1] 7907
```

```
colnames(dataset)
```

```
##  [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"
```

```
str(dataset)
```

```
## 'data.frame':    7907 obs. of  16 variables:
##  $ id                            : int  49091 50646 56334 71609 71896 71903 71907 241503 241508 241510 ...
##  $ name                          : chr  "COZICOMFORT LONG TERM STAY ROOM 2" "Pleasant Room along Bukit Timah" "COZICOMFORT" "Ensuite Room (Room 1 & 2) near EXPO" ...
##  $ host_id                       : int  266763 227796 266763 367042 367042 367042 367042 1017645 1017645 10176
45 ...
##  $ host_name                     : chr  "Francesca" "Sujatha" "Francesca" "Belinda" ...
##  $ neighbourhood_group           : chr  "North Region" "Central Region" "North Region" "East Region" ...
##  $ neighbourhood                 : chr  "Woodlands" "Bukit Timah" "Woodlands" "Tampines" ...
##  $ latitude                      : num  1.44 1.33 1.44 1.35 1.35 ...
##  $ longitude                     : num  104 104 104 104 104 ...
##  $ room_type                     : chr  "Private room" "Private room" "Private room" "Private room" ...
##  $ price                         : int  83 81 69 206 94 104 208 50 54 42 ...
##  $ minimum_nights                : int  180 90 6 1 1 1 1 90 90 90 ...
##  $ number_of_reviews             : int  1 18 20 14 22 39 25 174 198 236 ...
##  $ last_review                   : chr  "2013-10-21" "2014-12-26" "2015-10-01" "2019-08-11" ...
##  $ reviews_per_month             : num  0.01 0.28 0.2 0.15 0.22 0.38 0.25 1.88 2.08 2.53 ...
##  $ calculated_host_listings_count: int  2 1 2 9 9 9 9 4 4 4 ...
##  $ availability_365              : int  365 365 365 353 355 346 172 59 133 147 ...
```

```
head(dataset)
```

```
##      id                                name host_id host_name
## 1 49091   COZICOMFORT LONG TERM STAY ROOM 2  266763 Francesca
## 2 50646     Pleasant Room along Bukit Timah  227796   Sujatha
## 3 56334                         COZICOMFORT  266763 Francesca
## 4 71609 Ensuite Room (Room 1 & 2) near EXPO  367042   Belinda
## 5 71896     B&B  Room 1 near Airport & EXPO  367042   Belinda
## 6 71903         Room 2-near Airport & EXPO  367042   Belinda
##   neighbourhood_group neighbourhood latitude longitude    room_type price
## 1        North Region     Woodlands  1.44255  103.7958 Private room    83
## 2      Central Region   Bukit Timah  1.33235  103.7852 Private room    81
## 3        North Region     Woodlands  1.44246  103.7967 Private room    69
## 4         East Region      Tampines  1.34541  103.9571 Private room   206
## 5         East Region      Tampines  1.34567  103.9596 Private room    94
## 6         East Region      Tampines  1.34702  103.9610 Private room   104
##   minimum_nights number_of_reviews last_review reviews_per_month
## 1            180                 1  2013-10-21              0.01
## 2             90                18  2014-12-26              0.28
## 3              6                20  2015-10-01              0.20
## 4              1                14  2019-08-11              0.15
## 5              1                22  2019-07-28              0.22
## 6              1                39  2019-08-15              0.38
##   calculated_host_listings_count availability_365
## 1                              2              365
## 2                              1              365
## 3                              2              365
## 4                              9              353
## 5                              9              355
## 6                              9              346
```

After checking the data set's head and str, we realize that this is a dense data set, with 16 columns, it appears to provide enough data for further exploration. We can already see some missing values (for example in the column "last_review" or "reviews_per_month"), which will require cleaning and handling of these NaN values.

- Looking to find out first what columns have null values.
- Using the code below will show us how many NA's are found in each column of the data set.

```
sapply(dataset, function(x) sum(is.na(x)))
```

```
##                             id                           name
##                              0                              0
##                        host_id                      host_name
##                              0                              0
##            neighbourhood_group                  neighbourhood
##                              0                              0
##                       latitude                      longitude
##                              0                              0
##                      room_type                          price
##                              0                              0
##                 minimum_nights              number_of_reviews
##                              0                              0
##                    last_review              reviews_per_month
##                              0                           2758
## calculated_host_listings_count               availability_365
##                              0                              0
```

I also noticed that the name column had some empty observations, which led me to try and understand which columns are necessary and which columns should I omit from my analysis. Columns like "host_name" are irrelevant to my analysis, also, columns such as "last_review" and "reviews_per_month" need some minor adjustments. Since "last_review" is a date, the missing value just means that it wasn't reviewed yet, but since this column won't be affecting my analysis, I will go ahead and remove it along with id and host_name. In "reviews_per_month" I can simply append it with 0.0 where the missing values are, meaning that it also was not reviewed yet.

```
# Dropping columns that are not significant or could be unethical (like the host's name).
colnames(dataset)
```

```
##  [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"
```

```
dataset <- dataset %>%
  select(-c(last_review, id, host_name))

# Checking the dataset once again.

head(dataset)
```

```
##                                  name host_id neighbourhood_group neighbourhood
## 1   COZICOMFORT LONG TERM STAY ROOM 2  266763        North Region     Woodlands
## 2     Pleasant Room along Bukit Timah  227796      Central Region   Bukit Timah
## 3                         COZICOMFORT  266763        North Region     Woodlands
## 4 Ensuite Room (Room 1 & 2) near EXPO  367042         East Region      Tampines
## 5     B&B  Room 1 near Airport & EXPO  367042         East Region      Tampines
## 6          Room 2-near Airport & EXPO  367042         East Region      Tampines
##   latitude longitude    room_type price minimum_nights number_of_reviews
## 1  1.44255  103.7958 Private room    83            180                 1
## 2  1.33235  103.7852 Private room    81             90                18
## 3  1.44246  103.7967 Private room    69              6                20
## 4  1.34541  103.9571 Private room   206              1                14
## 5  1.34567  103.9596 Private room    94              1                22
## 6  1.34702  103.9610 Private room   104              1                39
##   reviews_per_month calculated_host_listings_count availability_365
## 1              0.01                              2              365
## 2              0.28                              1              365
## 3              0.20                              2              365
## 4              0.15                              9              353
## 5              0.22                              9              355
## 6              0.38                              9              346
```

```
# Replacing NA's with 0's.

dataset[is.na(dataset)] <- 0

# Examining changes.

sapply(dataset, function(x) sum(is.na(x)))
```

```
##                           name                        host_id
##                              0                              0
##            neighbourhood_group                  neighbourhood
##                              0                              0
##                       latitude                      longitude
##                              0                              0
##                      room_type                          price
##                              0                              0
##                 minimum_nights              number_of_reviews
##                              0                              0
##              reviews_per_month calculated_host_listings_count
##                              0                              0
##               availability_365
##                              0
```

```
# Checking the different Categorical Values present in the column "neighbourhood_group".

unique(dataset$neighbourhood_group)
```

```
## [1] "North Region"      "Central Region"    "East Region"
## [4] "West Region"       "North-East Region"
```

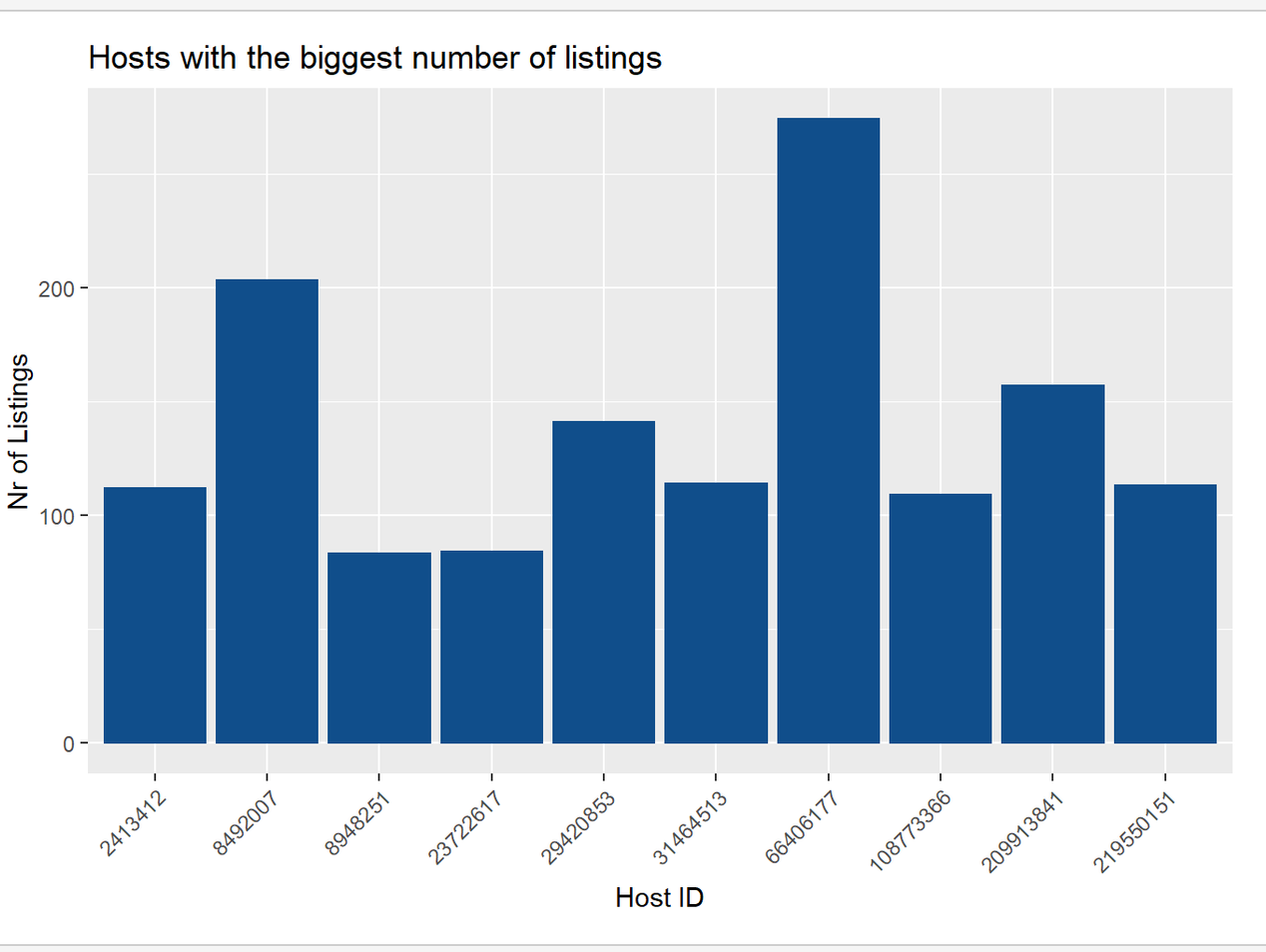
```
# Checking how many unique values are inside the column "neighbourhood"

n_distinct(dataset$neighbourhood)
```

```
## [1] 43
```

Exploring and Visualizing Data

Starting with the "host_id" column, we will first check which hosts have the most listings in the dataset.

```
# Creating a new table for the host_id and the listings count.

top_10_hosts <- table(dataset$host_id) %>%
  as.data.frame() %>%
  arrange(desc(Freq))

# Selecting the top 10 hosts.

top_10_hosts <- top_10_hosts[1:10,]
```



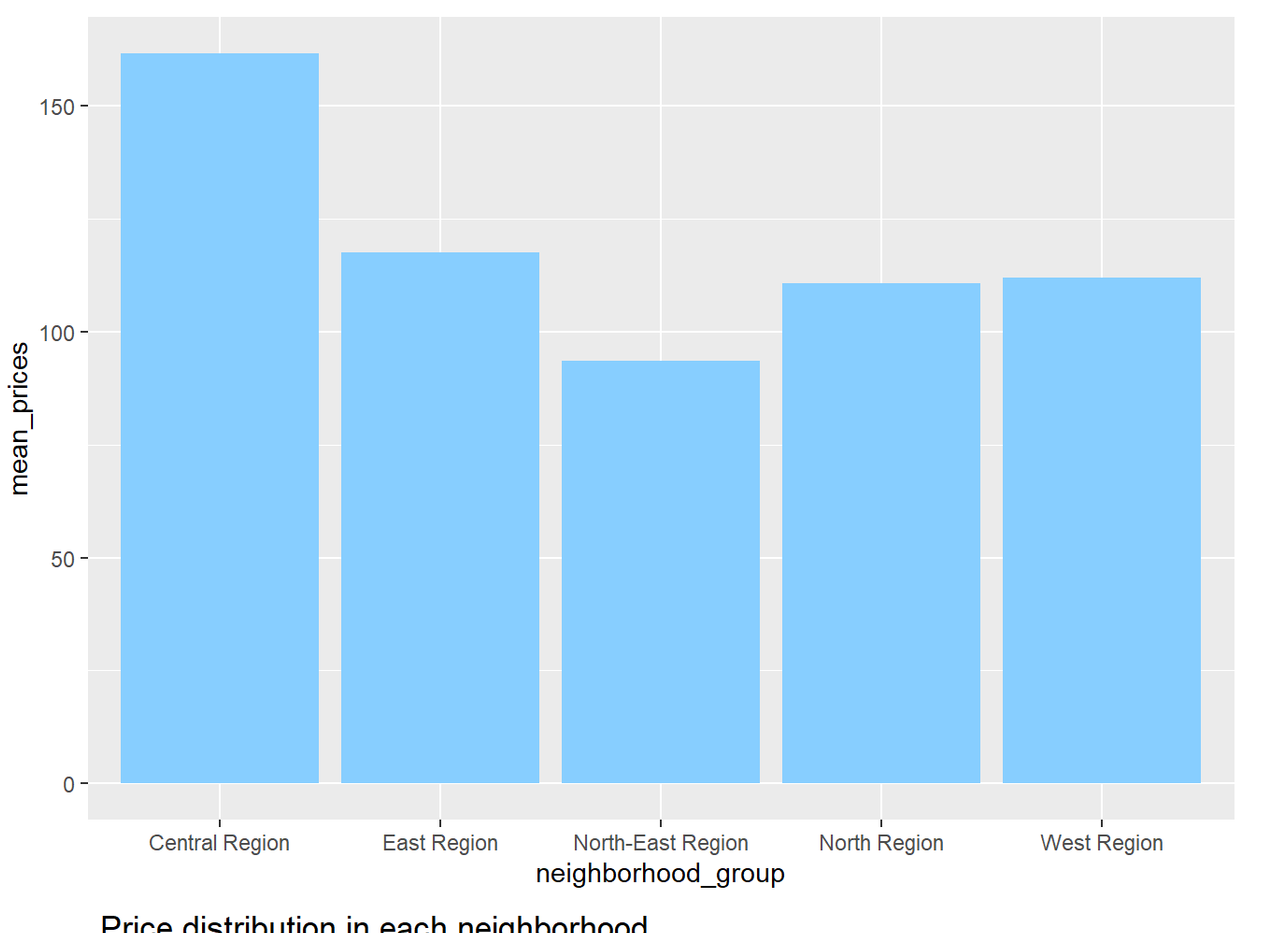

```
# Checking max prices to see if we have extreme values affecting our means.

max(dataset$price, na.rm=TRUE)
```

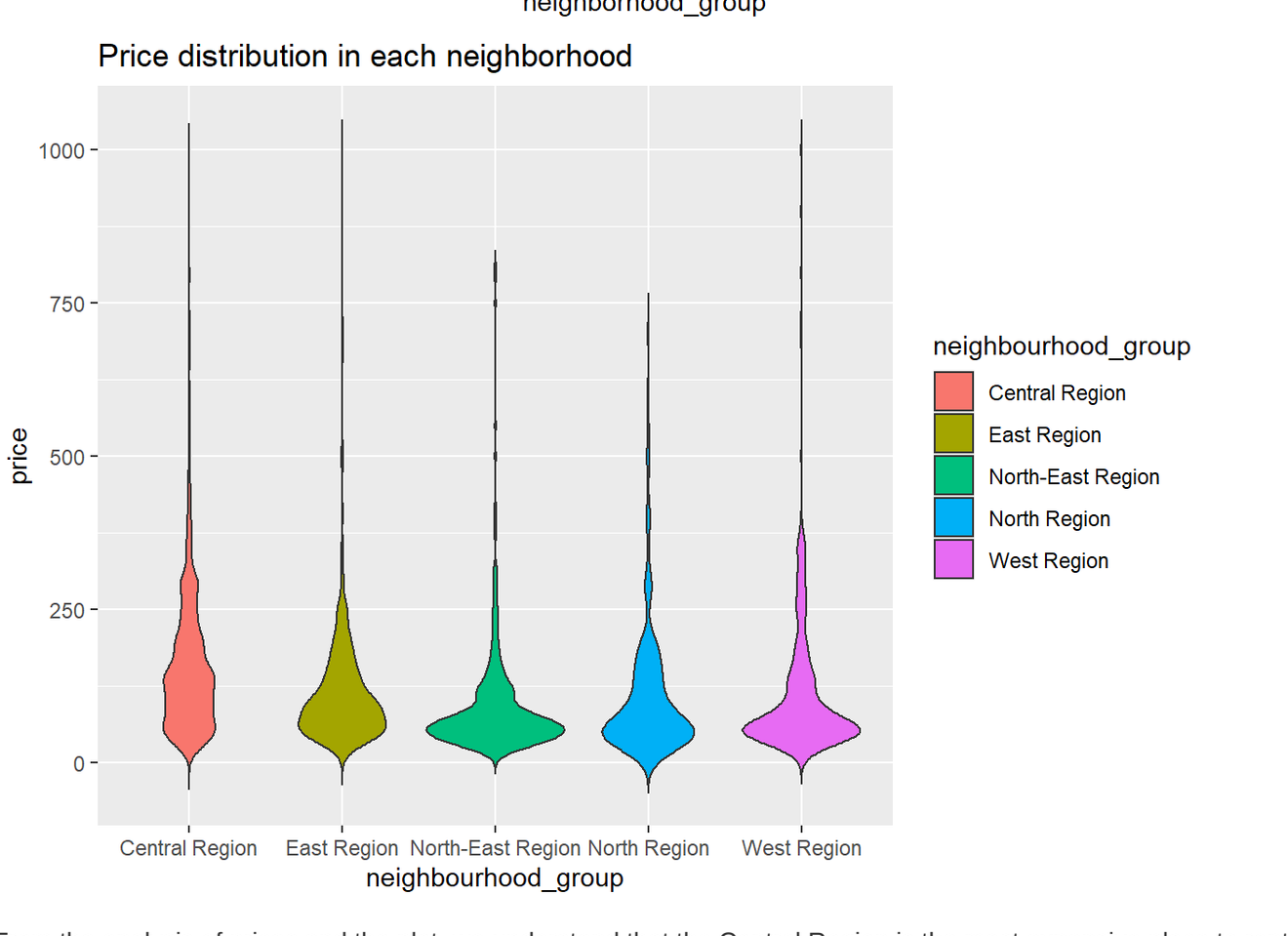
```
## [1] 10000
```

```
# Creating a new data frame without these extreme values.

sub_dataset <- dataset[dataset$price <= 1000,]

# Checking the mean values for each neighborhood group.

unique(sub_dataset$neighbourhood_group)
```

```
## [1] "North Region"      "Central Region"    "East Region"
## [4] "West Region"       "North-East Region"
```

```
north_set <- sub_dataset[sub_dataset$neighbourhood_group == "North Region",]
mean_north <- mean(north_set$price, na.rm=TRUE)

central_set <- sub_dataset[sub_dataset$neighbourhood_group == "Central Region",]
mean_central <- mean(central_set$price, na.rm=TRUE)

east_set <- sub_dataset[sub_dataset$neighbourhood_group == "East Region",]
mean_east <- mean(east_set$price, na.rm=TRUE)

west_set <- sub_dataset[sub_dataset$neighbourhood_group == "West Region",]
mean_west <- mean(west_set$price, na.rm=TRUE)

north_east_set <- sub_dataset[sub_dataset$neighbourhood_group == "North-East Region",]
mean_north_east <- mean(north_east_set$price, na.rm=TRUE)

neighborhood_group <- c(unique(sub_dataset$neighbourhood_group)[1], unique(sub_dataset$neighbourhood_group)[2], u
nique(sub_dataset$neighbourhood_group)[3], unique(sub_dataset$neighbourhood_group)[4], unique(sub_dataset$neighbo
urhood_group)[5])
mean_prices <- c(mean_north, mean_central, mean_east, mean_west, mean_north_east)

mean_prices_df <- data.frame(neighborhood_group, mean_prices)

ggplot(mean_prices_df, aes(x=neighborhood_group, y=mean_prices)) +
  geom_col(fill="skyblue1")
```





From the analysis of prices and the plot, we understand that the Central Region is the most expensive place to rent on average, with a mean price of ~161, the East Region stands in second with a mean price of ~117. West and North Regions have similar mean prices, between 110-112, and the West Region is the least expensive place, with a mean price of ~93.

I will now proceed to analyze the "neighbourhood" column.

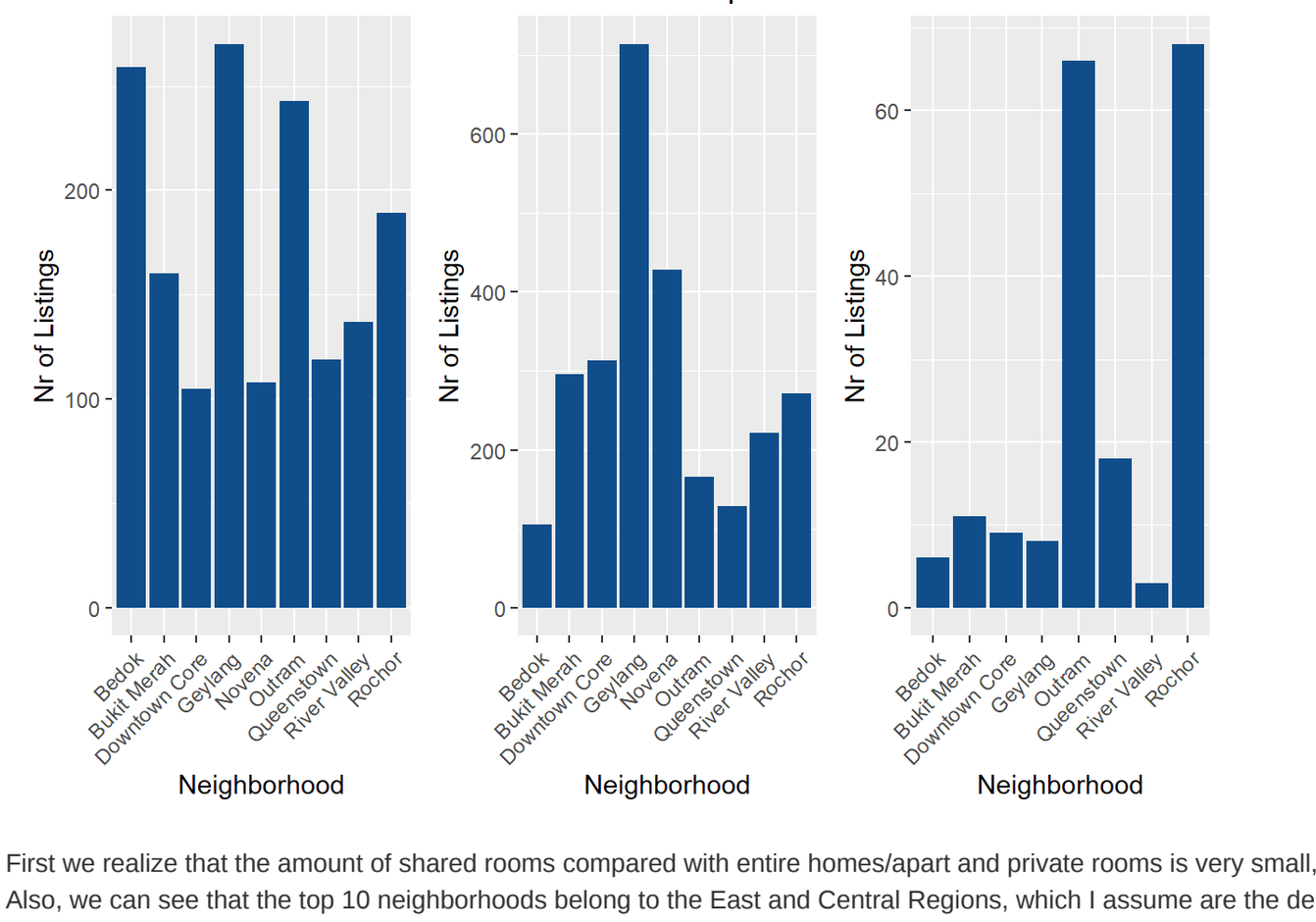
```
# Creating a new table for the neighborhoods and the listings count.

top_10_neighborhoods <- table(sub_dataset$neighbourhood) %>%
  as.data.frame() %>%
  arrange(desc(Freq))

# Selecting the top 10 neighborhoods.

top_10_neighborhoods <- top_10_neighborhoods[1:10,]
```

```
# Filtering the dataset to show only the top 10 neighborhoods.

target <- c("   Kallang", "Geylang", "Novena", "Rochor", "Outram", "Bukit Merah", "Downtown Core", "Bedok", "Rive
r Valley", "Queenstown")

neighborhood_set <- filter(sub_dataset, neighbourhood %in% target)
```

```
# Ploting neighborhoods with the room type and listing count.

private_room_subset <- neighborhood_set[neighborhood_set$room_type == "Private room",]

priv_room_viz <- ggplot(private_room_subset, aes(x=neighbourhood)) +
  geom_bar(stat = "count", position=position_dodge(), fill="dodgerblue4") +
  labs(title = "Private rooms") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  xlab("Neighborhood") + ylab("Nr of Listings")
```

```
entire_home_subset <- neighborhood_set[neighborhood_set$room_type == "Entire home/apt",]

entire_home_viz <- ggplot(entire_home_subset, aes(x=neighbourhood)) +
  geom_bar(stat = "count", position=position_dodge(), fill="dodgerblue4") +
  labs(title = "Entire homes/apartments") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  xlab("Neighborhood") + ylab("Nr of Listings")
```

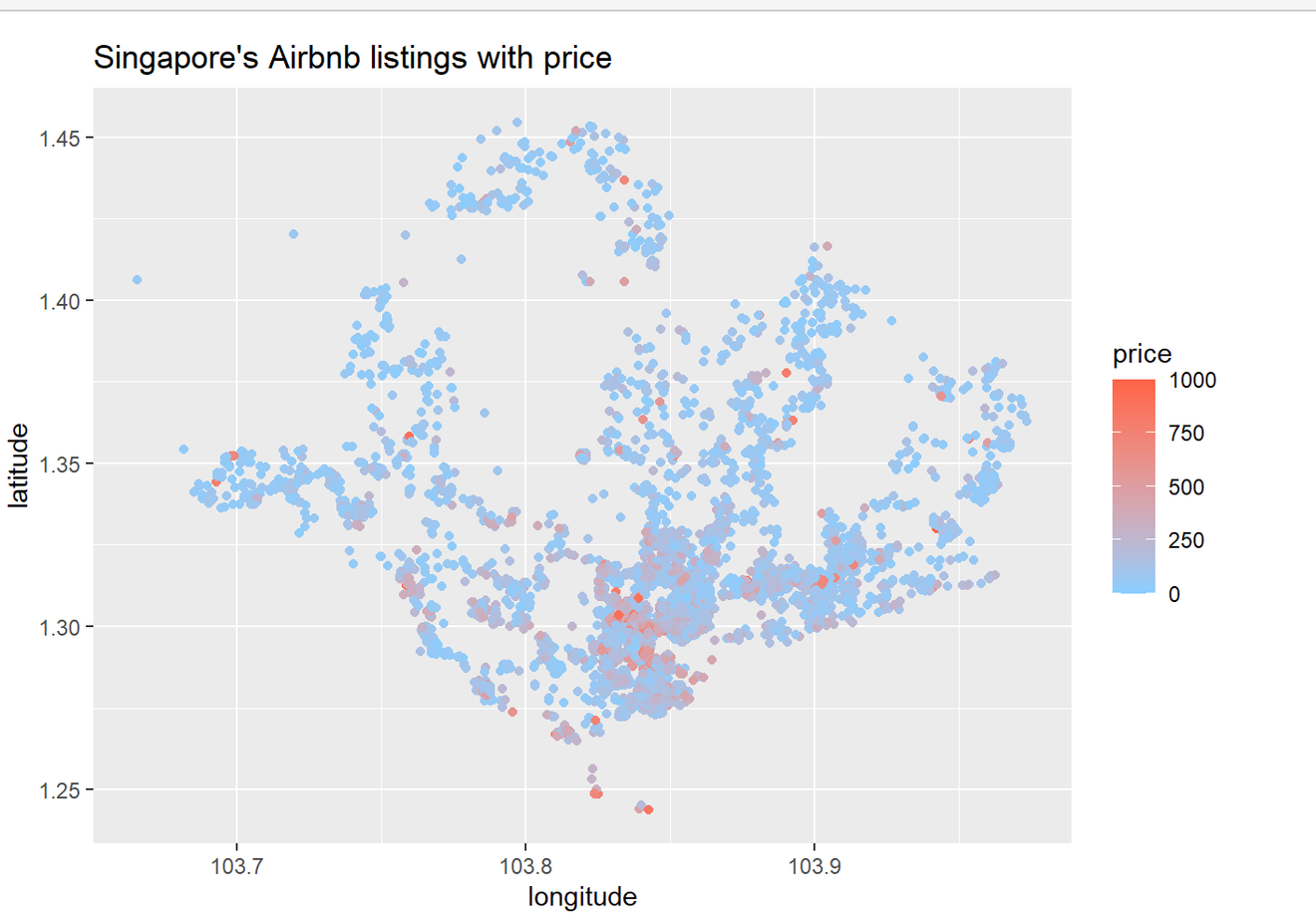
```
shared_room_subset <- neighborhood_set[neighborhood_set$room_type == "Shared room",]

shared_room_viz <- ggplot(shared_room_subset, aes(x=neighbourhood)) +
  geom_bar(stat = "count", position=position_dodge(), fill="dodgerblue4") +
  labs(title = "Shared rooms") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  xlab("Neighborhood") + ylab("Nr of Listings")

priv_room_viz | entire_home_viz | shared_room_viz
```



First we realize that the amount of shared rooms compared with entire homes/apart and private rooms is very small, meaning it is barely available. Also, we can see that the top 10 neighborhoods belong to the East and Central Regions, which I assume are the destinations most picked by people visiting Singapore. The most popular neighborhood for the Central Region is Geylang and for the East Region is Bedok.

Next, I will present a scatter plot with the latitude and longitude, and using price as color to quickly identify the most expensive areas and also the most dense areas in terms of listings.

```
singapore_scatter <- ggplot(sub_dataset, aes(y=latitude, x=longitude, color=price)) +
  geom_point() +
  labs(title="Singapore's Airbnb listings with price")
singapore_scatter + scale_color_gradient(low="skyblue1", high="tomato1")
```



Without the map underneath the scatter plot we cannot really know exactly where these points are, unless you are already familiar with Singapore's map, so it would be much easier to read and understand the plot if we had an image as background for easier interpretation. Lets fix that.

```
# Import the image
my_image <- readPNG("C:\\Users\\Dacs\\Documents\\R Working Directory\\Singapore Airbnb Analysis\\flag-of-singapor
e-map-png-favpng-gfJsSW96Ax0E4x8ZGayMc07ZL.png")

ggplot(sub_dataset, aes(y=latitude, x=longitude, color=price)) +
  annotation_custom(rasterGrob(my_image,
                               width = unit(1,"npc"),
                               height = unit(1,"npc")),
                               -Inf, Inf, -Inf, Inf) +
  scale_color_gradient(low="skyblue1", high="tomato1") +
  geom_point() +
  labs(title="Singapore's Airbnb listings with price") +
  xlim(103.58, 104.05) +
  ylim(1.16, 1.54)
```

Some points might not be on top of the map and this is due to the fact that I had to adjust the scatter plot scale to fit the map and because of that, this is not an 100% accurate representation of a heat map, but enough for us to understand where most of the pints are located and their price. At first I tried using the "maps" and "viridis" package but the map presented for Singapore was not as accurate and it just felt off, so I went with the background image instead even though I could not present the grid.

I will now proceed to analyze the "number_of_reviews" column.

```
reviewed_listings <- sub_dataset %>%
  arrange(desc(number_of_reviews))

top_10_reviewed_listings <- reviewed_listings[1:10,]
```

Now I will check the mean price per night in the top 10 reviewed listings.

```
mean(top_10_reviewed_listings$price)
```

```
## [1] 124.7
```

As you can see, the mean price per night in the top 10 reviewed listings is an average of 124.7 per night, with only 3 listings above 100 per night and 7 of those being private rooms and only 3 entire home/apartment (which are the most expensive ones of the 10). The most reviewed listing has 323 reviews and is located in Bedok, East Region.

Conclusion

Summary of my findings

The data set present in this notebook is from 2019 and has a variety of columns that let's us explore the data, column by column in various ways. I started by cleaning and preparing the data for our analysis, which involved checking column names, how many observations we had, removing some columns and fixing some NA's, either by removing or setting to 0. Then I checked the hosts with the most listings, which led us to realize that out top host had about 274 listings. After checking the host with most listings, we proceed to analyze the neighborhood_group and neighborhood columns for listings density and location to understand the most popular places in Singapore are and their prices, and for this we started off with a violin plot to get the price distributions of each of the neighborhood groups, followed by 3 bar plots to differentiate between Private room, Shared room and Entire home/apt in the top 10 neighborhoods. Afterwards I decided it would be a good idea to make use of the latitude and longitude to understand listing locations in a scatter plot along with a color bar for price so we can quickly identify the most popular areas and prices, for this scatter plot I also added an image of Singapore's map as background of the scatter plot for easier reading. To end the analysis, I checked the top 10 reviewed listings and their mean price.