Exploring Airbnb Of New York City

AUTHOR
Huiting Wu, Christian, Alexis

```
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.4.3
```

```
library(tidyverse)
library(dplyr)
library(maps)
library(gganimate)
library(gifski)
library(transformr)
library(av)
library(tidytext)
```

Import and Combine the datasets

```
AB_NYC <- read_csv("AB_NYC_2019.csv")
AB_NYC_2023 <- read_csv("AB_NYC_2023.csv")
combined_dataset <- bind_rows(AB_NYC, AB_NYC_2023)
combined_dataset <- within(combined_dataset,
                           rm(number_of_reviews_ltm, license))
```

Information about the data

```
dim(AB_NYC)
```

```
[1] 48895    16
```

```
dim(AB_NYC_2023)
```

```
[1] 42931    18
```

```
dim(combined_dataset)
```

```
[1] 91826    16
```

Missing Value

```
sum(is.na(combined_dataset))
```

```
[1] 40764
```

```
missing_values <- is.na(combined_dataset$host_name)
sum(missing_values)
```

```
[1] 26
```

```
combined_dataset_with_missing_values <- combined_dataset |> filter(if_any(ever
dim(combined_dataset_with_missing_values)
```

```
[1] 20385    16
```

Remove Missing Value

```
combined_dataset <- combined_dataset[!is.na(combined_dataset$host_name), ]
dim(combined_dataset)
```

```
[1] 91800    16
```

Duplicates

```
sum(duplicated(combined_dataset))
```

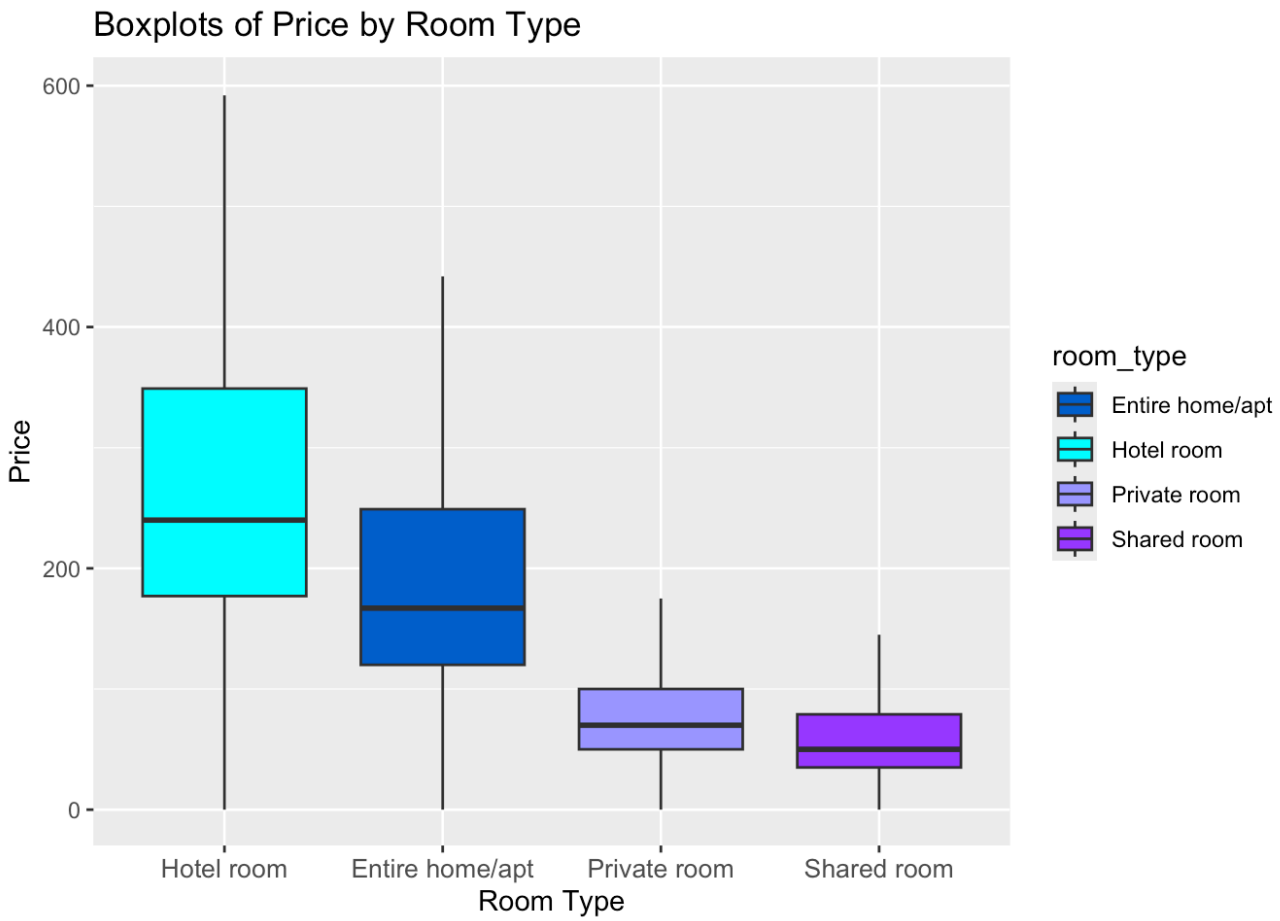
```
[1] 11
```

```
combined_dataset_unique <- distinct(combined_dataset)
dim(combined_dataset_unique)
```

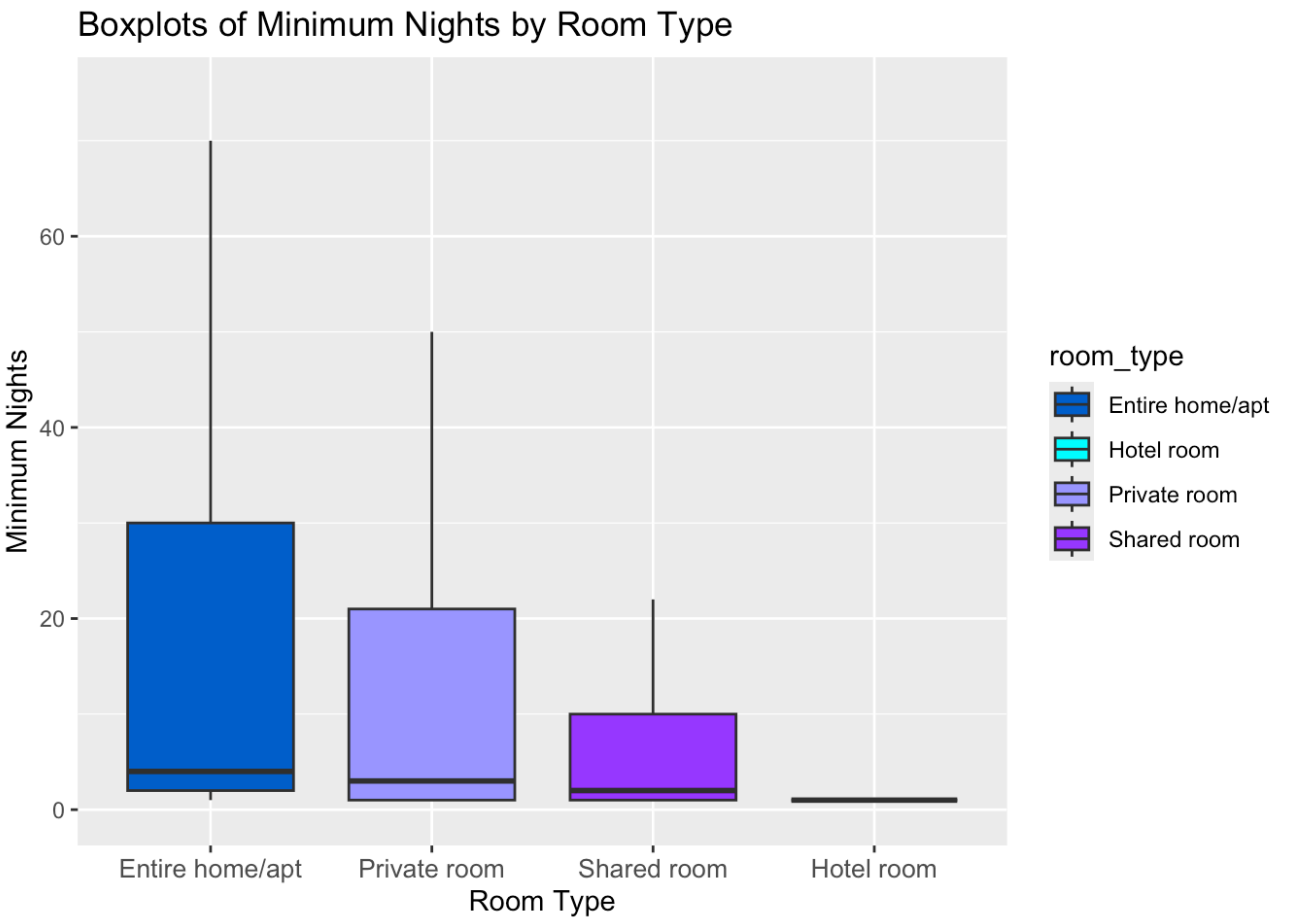
```
[1] 91789    16
```

Boxplot

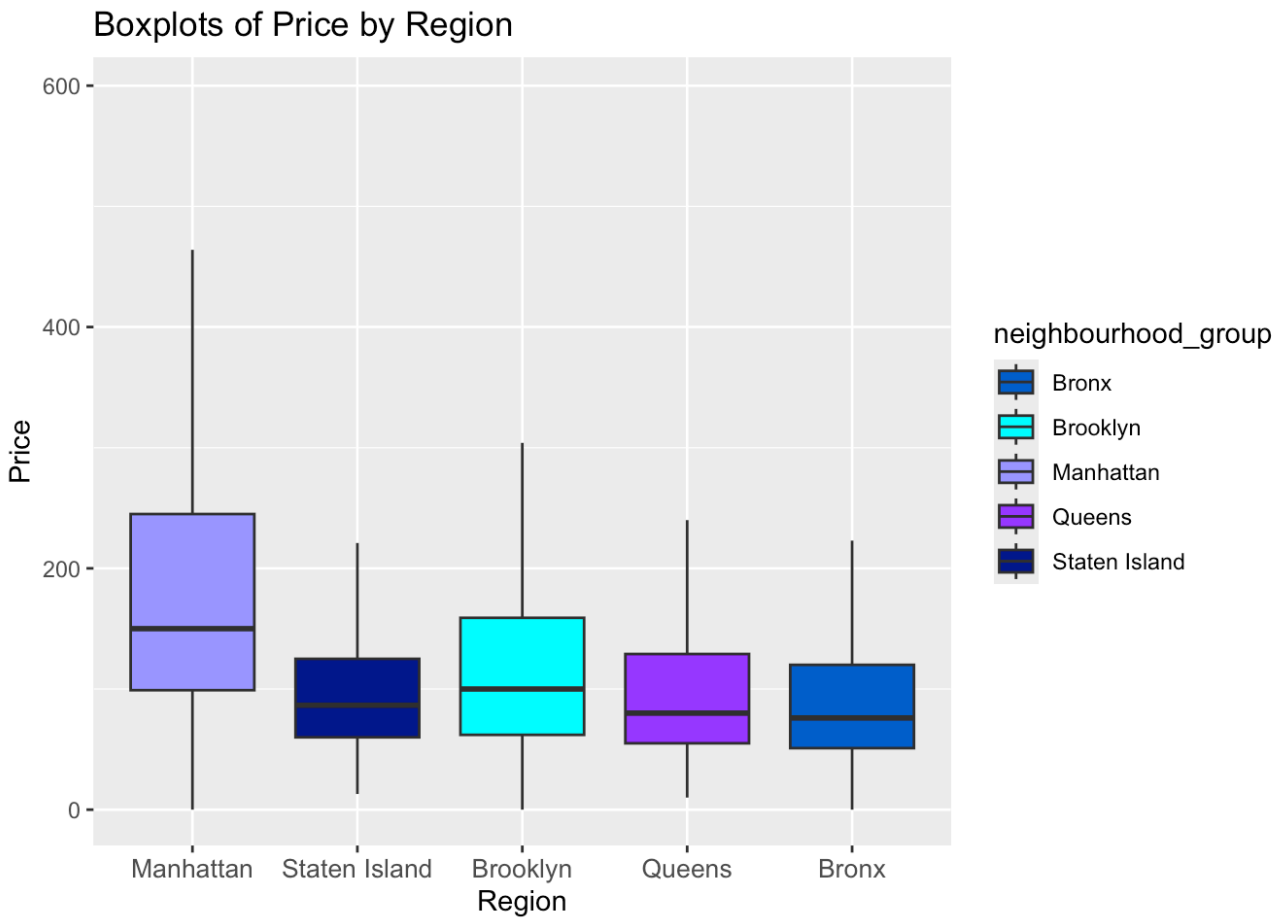
```
ggplot(combined_dataset_unique, aes(x = reorder(room_type, -price), y = price,
  scale_fill_manual(values = c("#0060CC","#00FFFF", "#9999FF", "#9933FF")) +
  geom_boxplot(outlier.shape = NA, outlier.colour = NA) + # no showing outlier
  coord_cartesian(ylim = c(0, max(combined_dataset_unique$price) * 0.006)) +
  labs(title = "Boxplots of Price by Room Type", x = "Room Type", y = "Price")
  theme(axis.text.x = element_text(size = 10))
```




```
ggplot(combined_dataset_unique, aes(x = reorder(room_type, -minimum_nights), y
  scale_fill_manual(values = c("#0060CC","#00FFFF", "#9999FF", "#9933FF")) +
  geom_boxplot(outlier.shape = NA, outlier.colour = NA) +
  coord_cartesian(ylim = c(0, max(combined_dataset_unique$minimum_nights) * 0.
  labs(title = "Boxplots of Minimum Nights by Room Type", x = "Room Type", y =
  theme(axis.text.x = element_text(size = 10))
```

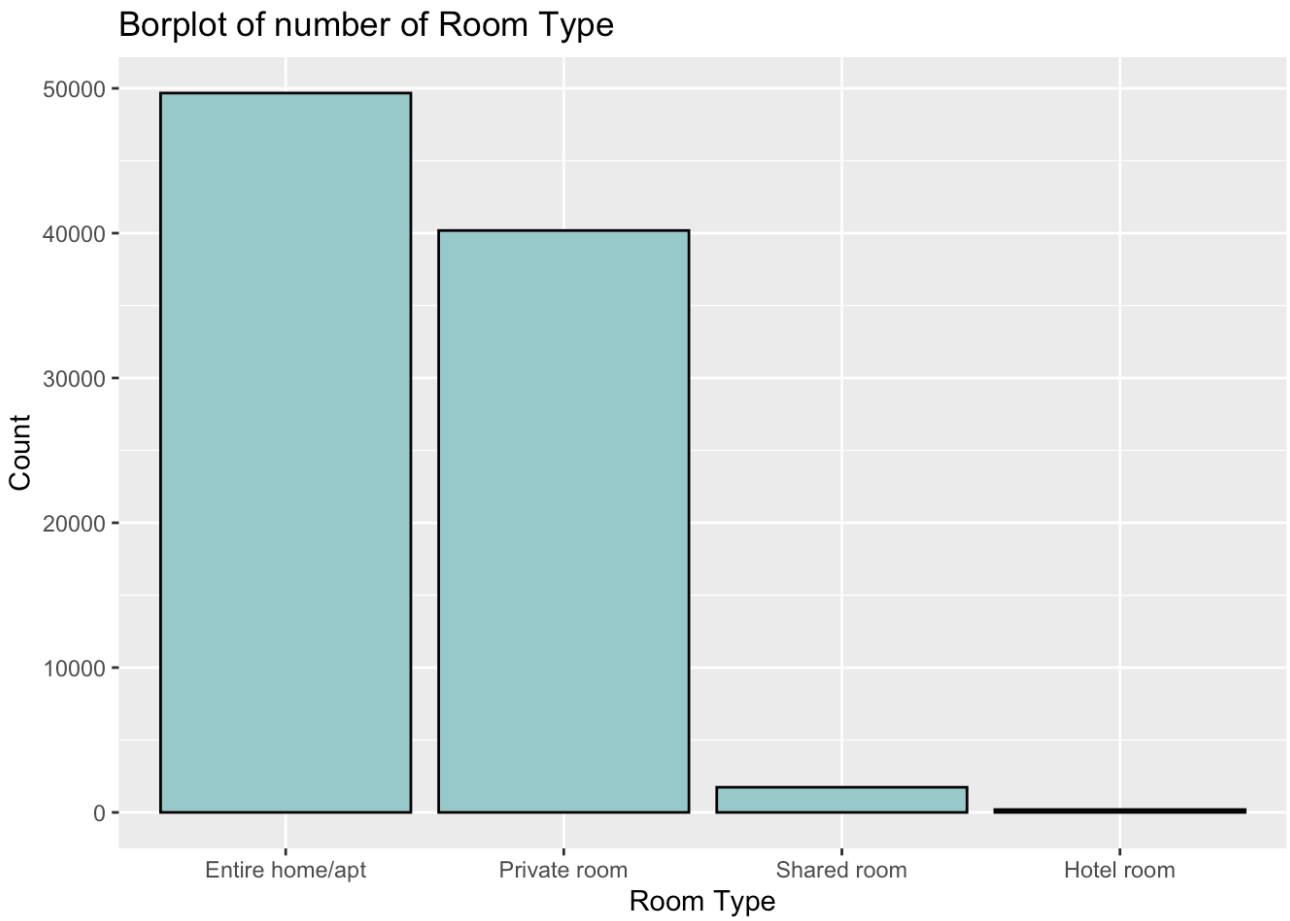



```
ggplot(combined_dataset_unique, aes(x = reorder(neighbourhood_group, -price),
                                    y = price, fill = neighbourhood_group)) +
  scale_fill_manual(values = c("#0060CC","#00FFFF", "#9999FF", "#9933FF","dark
  geom_boxplot(outlier.shape = NA, outlier.colour = NA) +
  coord_cartesian(ylim = c(0, max(combined_dataset_unique$price) * 0.006)) +
  labs(title = "Boxplots of Price by Region", x = "Region", y = "Price") +
  theme(axis.text.x = element_text(size = 10))
```



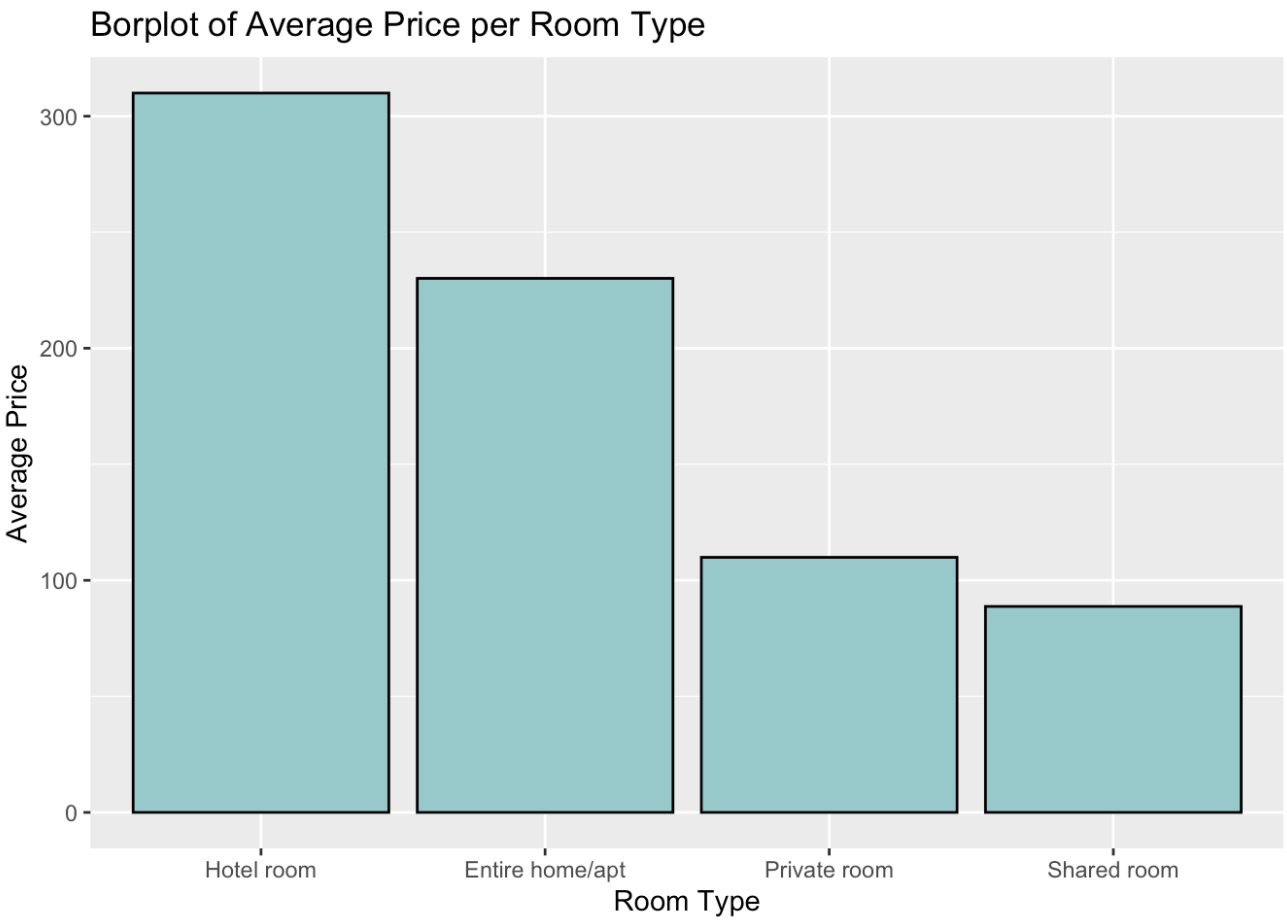
Barplot

```
combined_dataset_unique |> group_by(room_type) |>
  summarise(count = n()) |>
  ggplot(aes(x = reorder(room_type, -count), y = count)) +
  geom_bar(stat = "identity", fill =  "#99CCCC", color = "black") +
  labs(title = "Borplot of number of Room Type", x = "Room Type", y = "Count")
```



The entire home/department and private room are more popular in Airbnb.

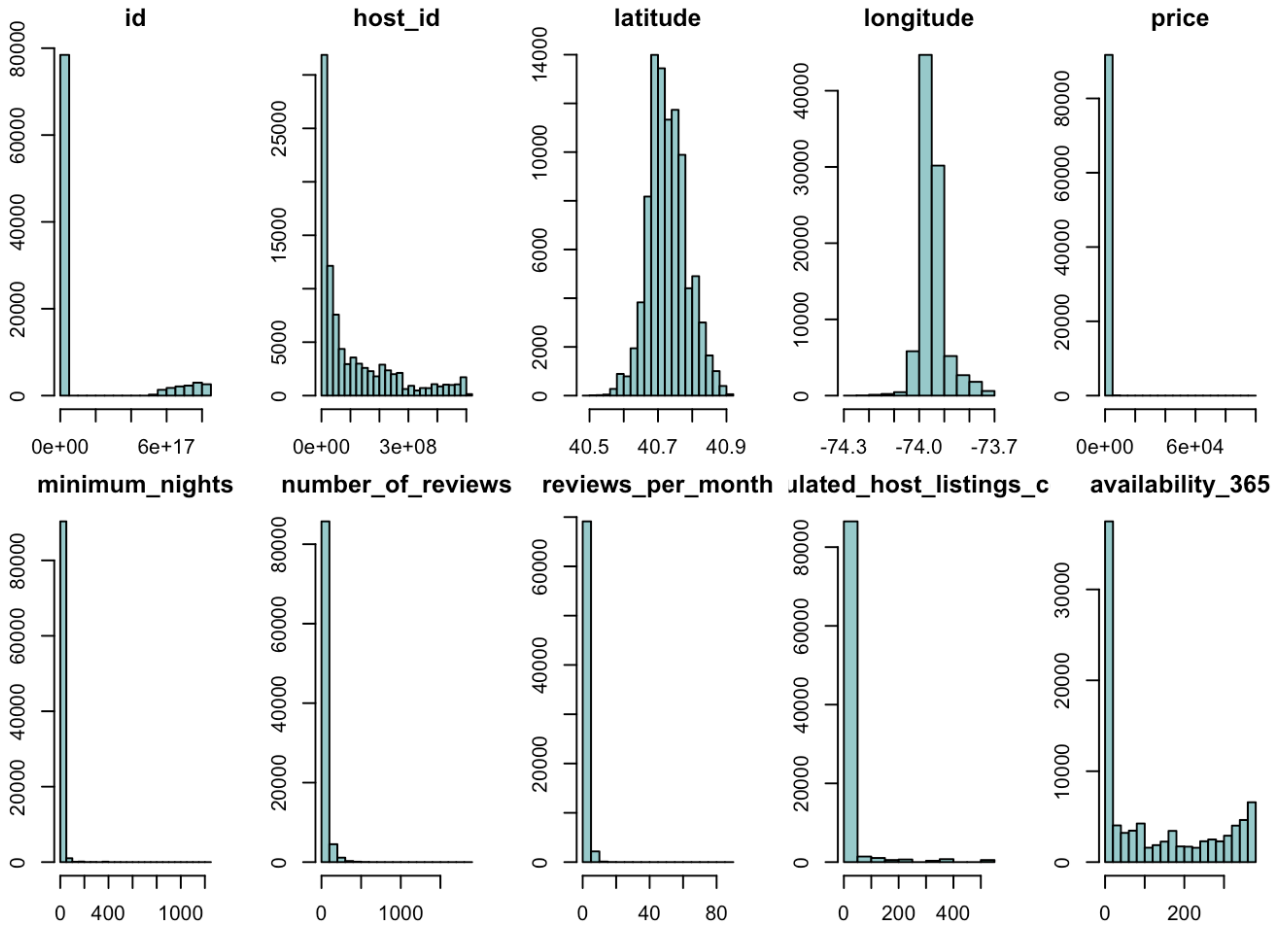
```
combined_dataset_unique |>
  group_by(room_type) |>
  summarise(count = n(),
            avg_price =  mean(price)) |>
  ggplot(aes(x = reorder(room_type, -avg_price), y = avg_price)) +
  geom_bar(stat = "identity", fill =  "#99CCCC", color = "black") +
  labs(title = "Borplot of Average Price per Room Type",
       x = "Room Type", y = "Average Price")
```



The hotel room are the most expensive room type compared to others. And the shared room are the cheapest.

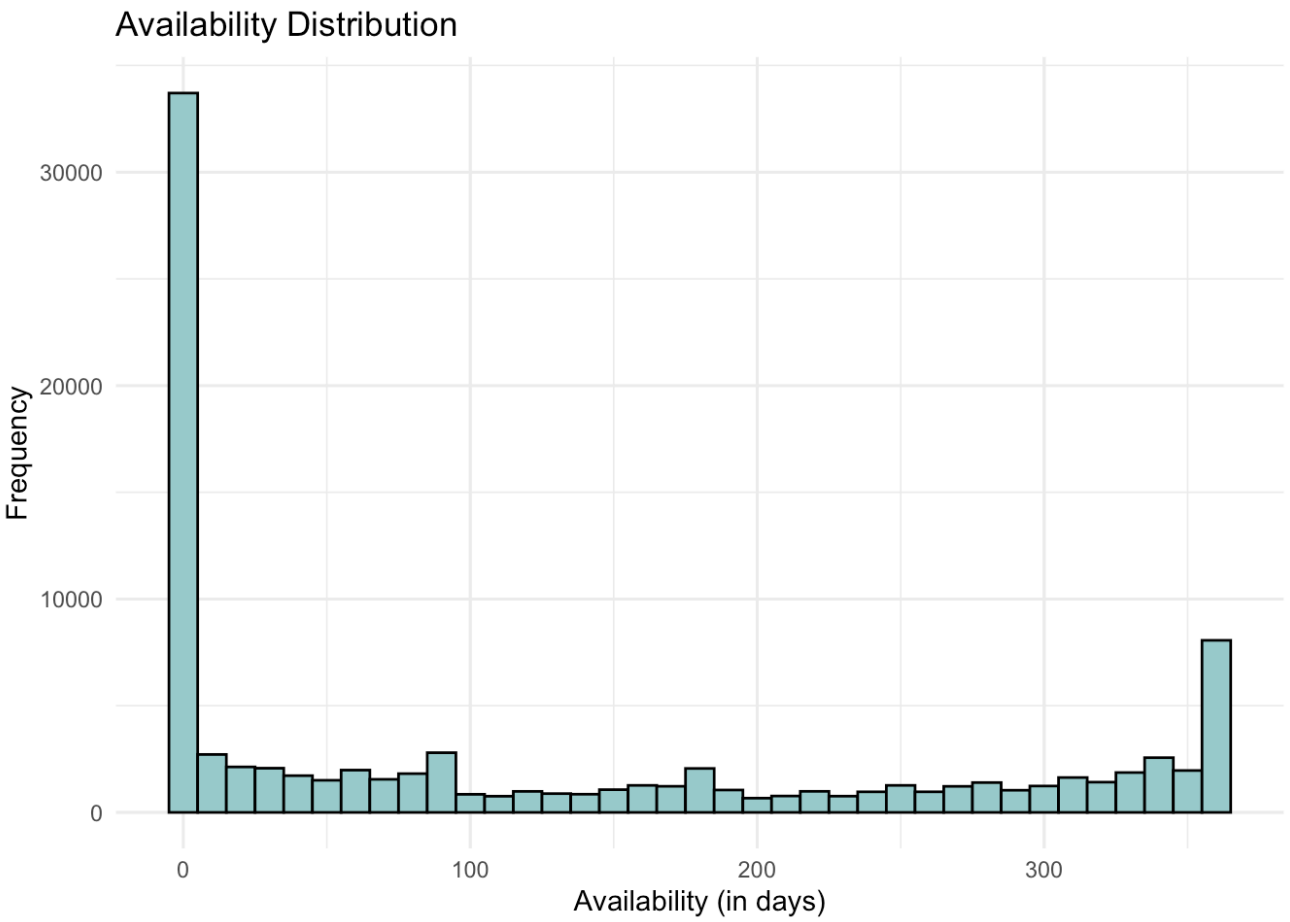
histogram

```
numeric_columns <- sapply(combined_dataset_unique, is.numeric)

par(mfrow = c(2, 5))
par(mar = c(2, 2, 2, 2))

for (col in names(combined_dataset_unique[, numeric_columns]))
{ hist(combined_dataset_unique[[col]],
       main = col,
       xlab=col,
       col= "#99CCCC",
       border="black") }
```



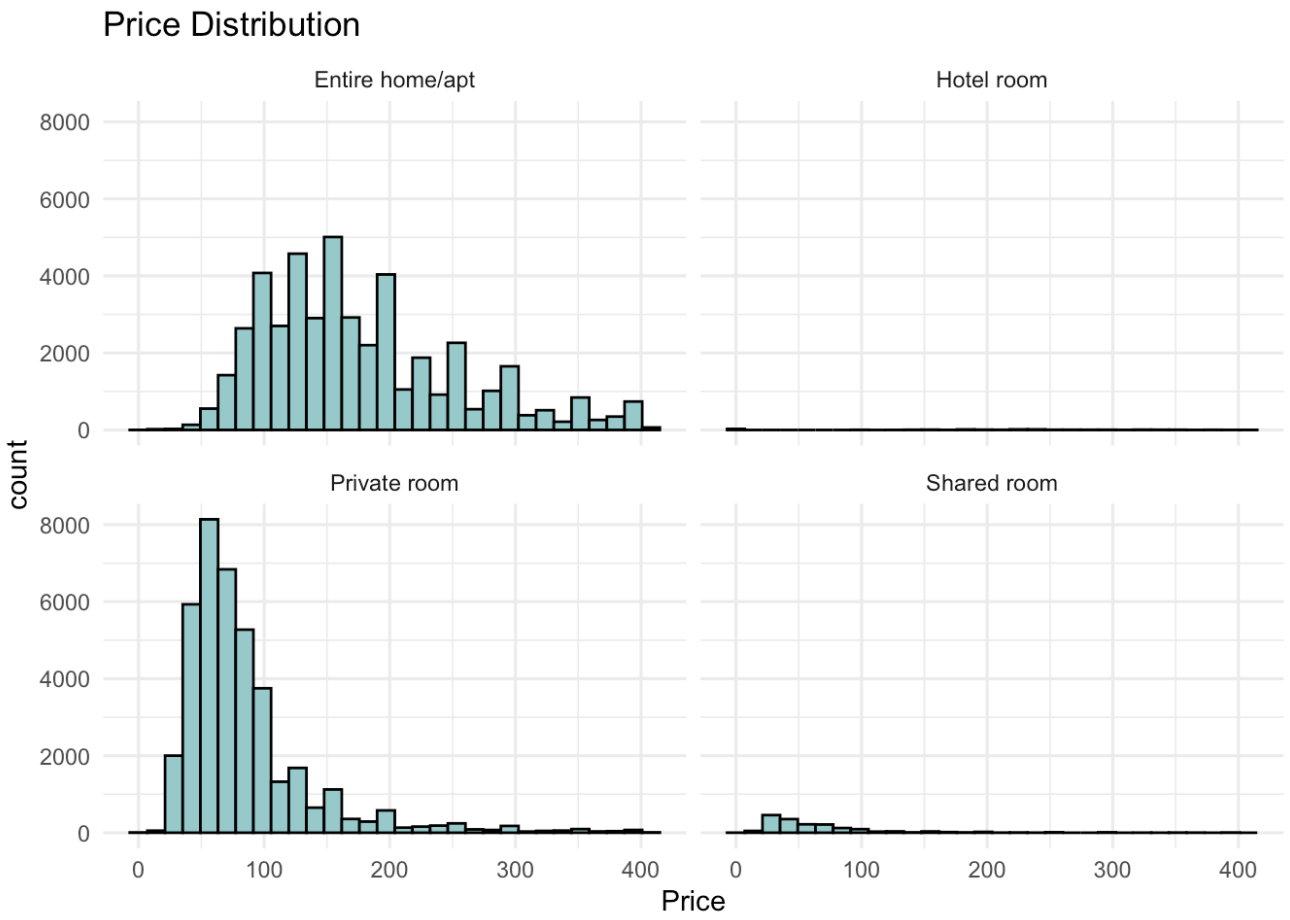
The histograms doesn't look good.

```
ggplot(combined_dataset_unique, aes(x = availability_365)) +
  geom_histogram(binwidth = 10, fill =  "#99CCCC", color = "black") +
  labs(title = "Availability Distribution", x = "Availability (in days)", y =
  theme_minimal()
```



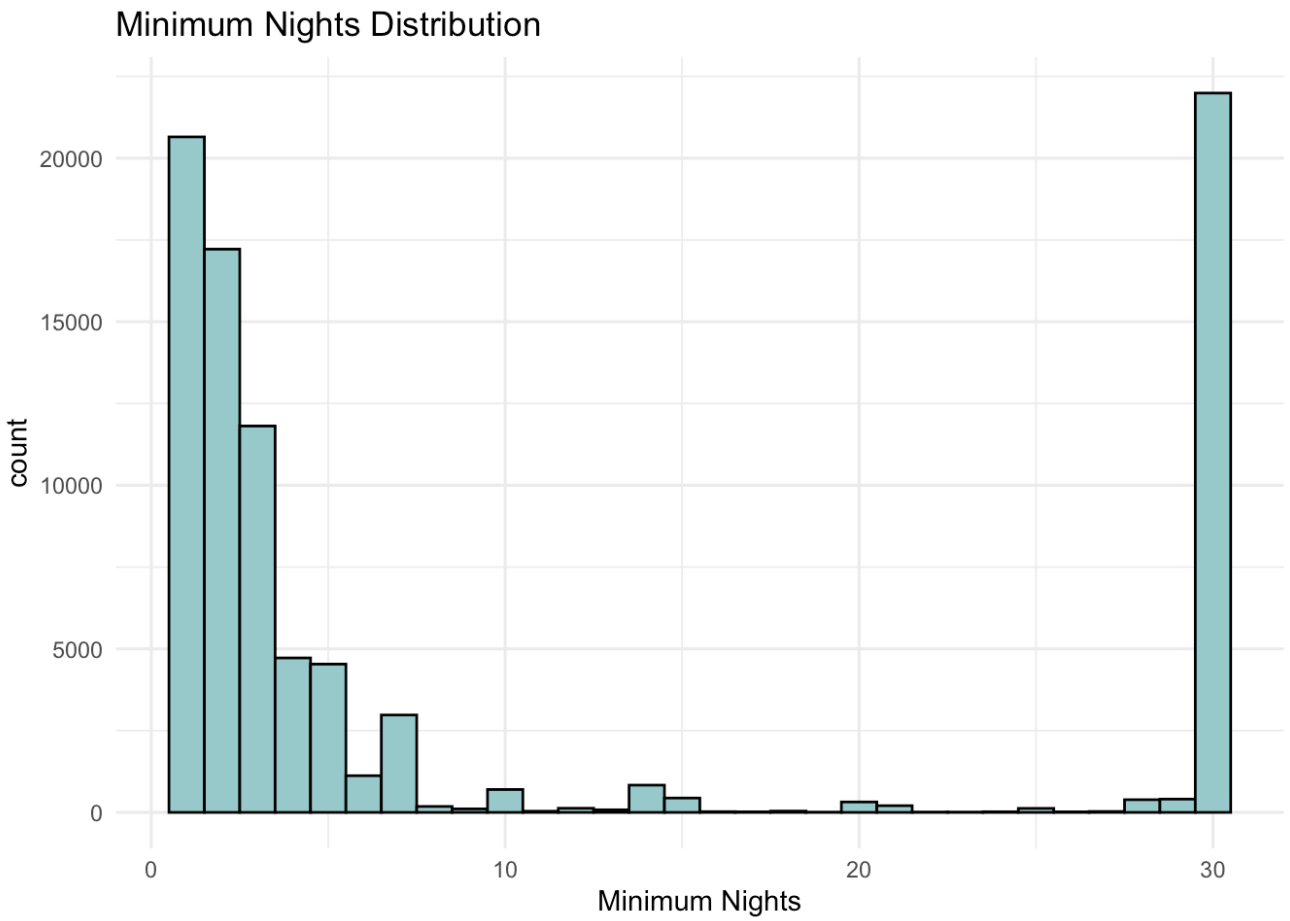
Most of the rooms in airbnb are being booked often.

```
price_upper_limit <- quantile(combined_dataset_unique$price, 0.95)

price_filtered <- combined_dataset_unique |>
  filter(price >= 0, price <= price_upper_limit)
# 95% of the data
ggplot(price_filtered, aes(x = price)) +
  geom_histogram(fill =  "#99CCCC", color = "black") +
  labs(title = "Price Distribution", x = "Price") +
  facet_wrap(~room_type) +
  theme_minimal()
```



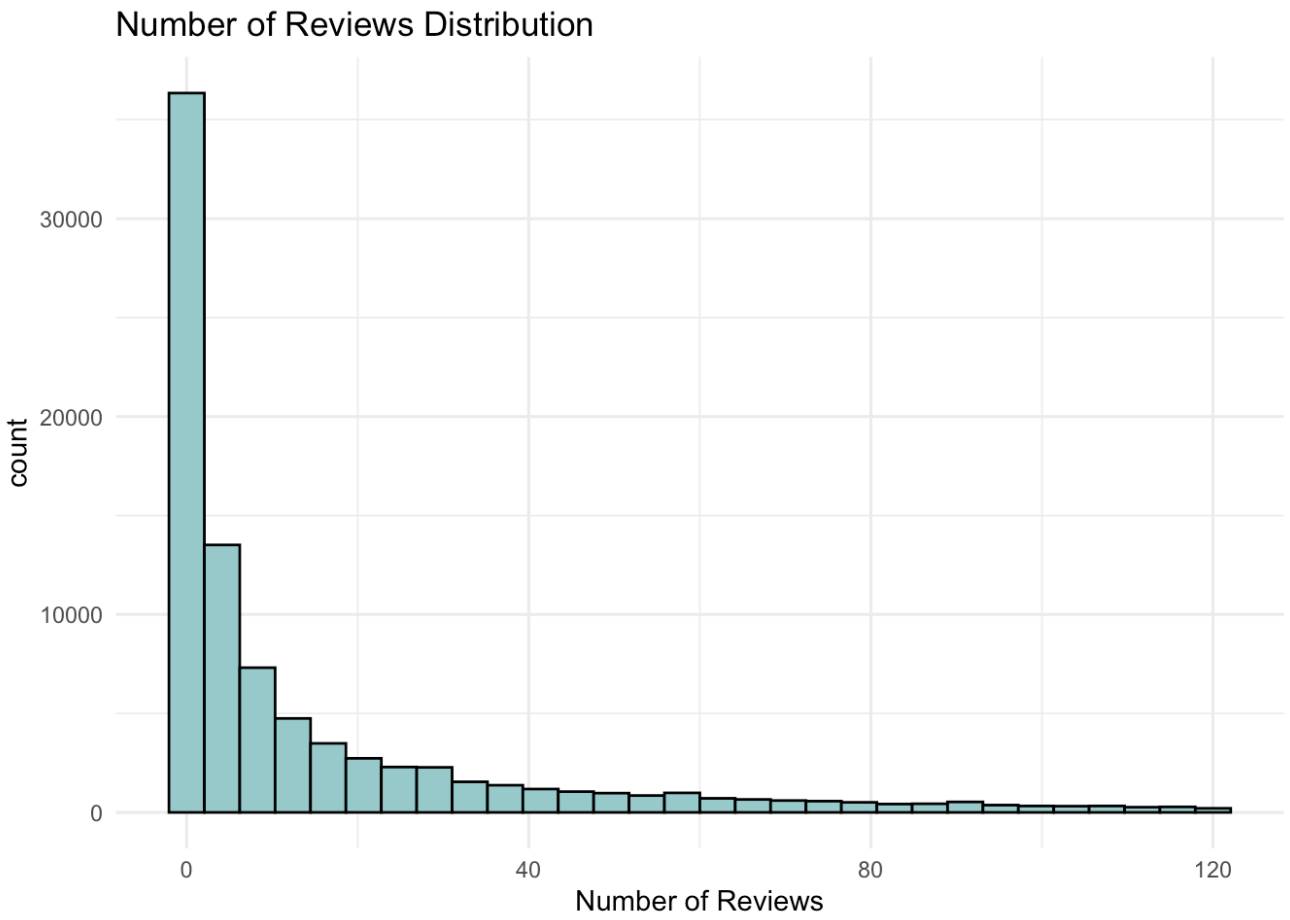
For the entire home/apartment, we can see the price distribution is slightly skewed to the left. Most of the entire home/apartment cost about the same price per night. Compare to the private room, most of the entire home/apartment cost higher than the private room. There is an obvious skewed to the left distribution for the price of private room per night. Most of the private rooms are cheaper.

```
nights_upper_limit <- quantile(combined_dataset_unique$minimum_nights, 0.95)
nights_filtered <- combined_dataset_unique |>
  filter(minimum_nights >= 0,
         minimum_nights <= nights_upper_limit)
# 95% of the data
ggplot(nights_filtered, aes(x = minimum_nights)) +
  geom_histogram(fill =  "#99CCCC", color = "black") +
  labs(title = "Minimum Nights Distribution", x = "Minimum Nights") +
  theme_minimal()
```



There is a short-term rental regulations in NYC airbnb. While waiting for approval, the host may set your calendar to 30 nights minimum so that the OSE may review and approve the listing's eligibility for short-term rental registration. That is why there are a lot of count at the minimum nights of 30. Most of the rooms with a under 10 minimum days to book.

```
reviews_upper_limit <- quantile(combined_dataset_unique$number_of_reviews, 0.9
reviews_filtered <- combined_dataset_unique |>
  filter(number_of_reviews >= 0, number_of_reviews <= reviews_upper_limit)
ggplot(reviews_filtered, aes(x = number_of_reviews)) +
  geom_histogram(fill =  "#99CCCC", color = "black") +
  labs(title = "Number of Reviews Distribution", x = "Number of Reviews") +
  theme_minimal()
```



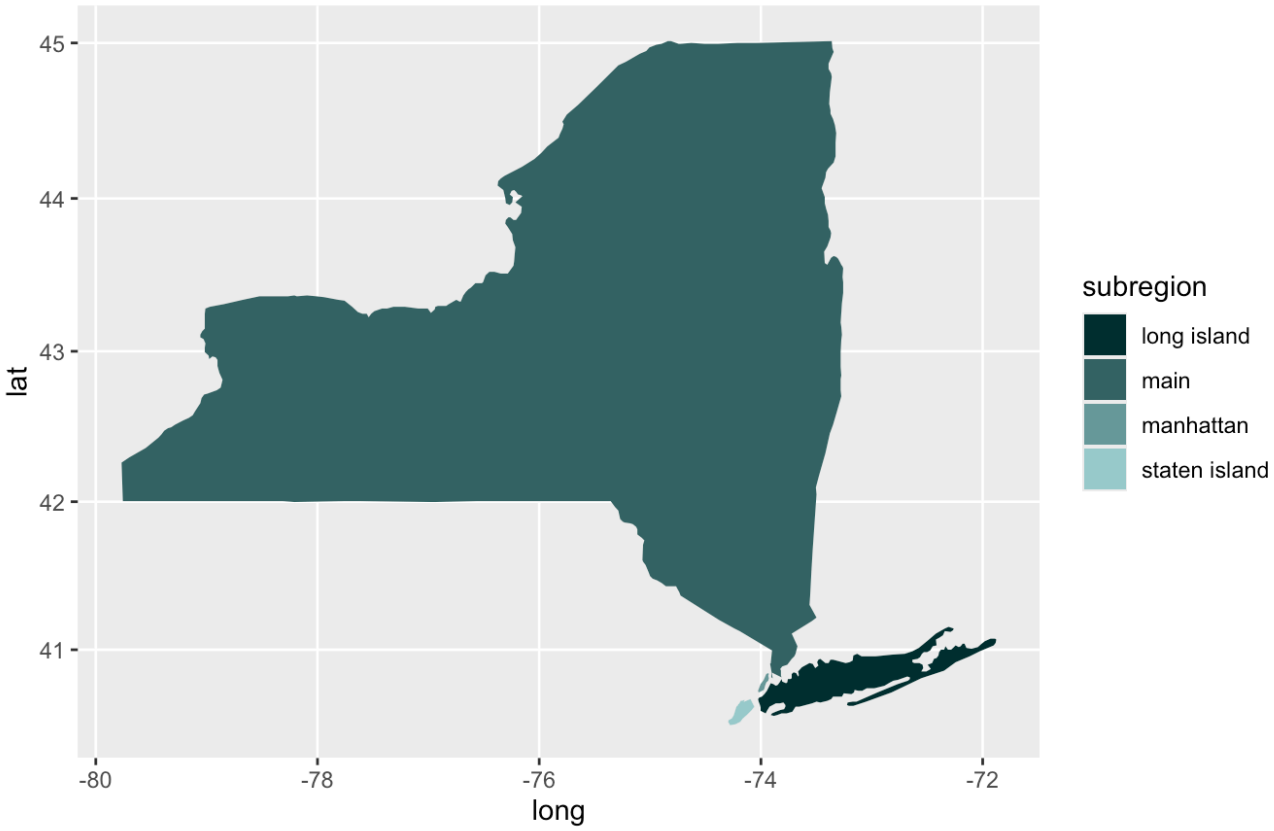
There is a obvious skewed to the right. most of the host recieved less reviews.

NY Map data

```
states <- map_data("state")
counties <- map_data("county")
NewYork <- subset(states, region == "new york")
head(NewYork)
```

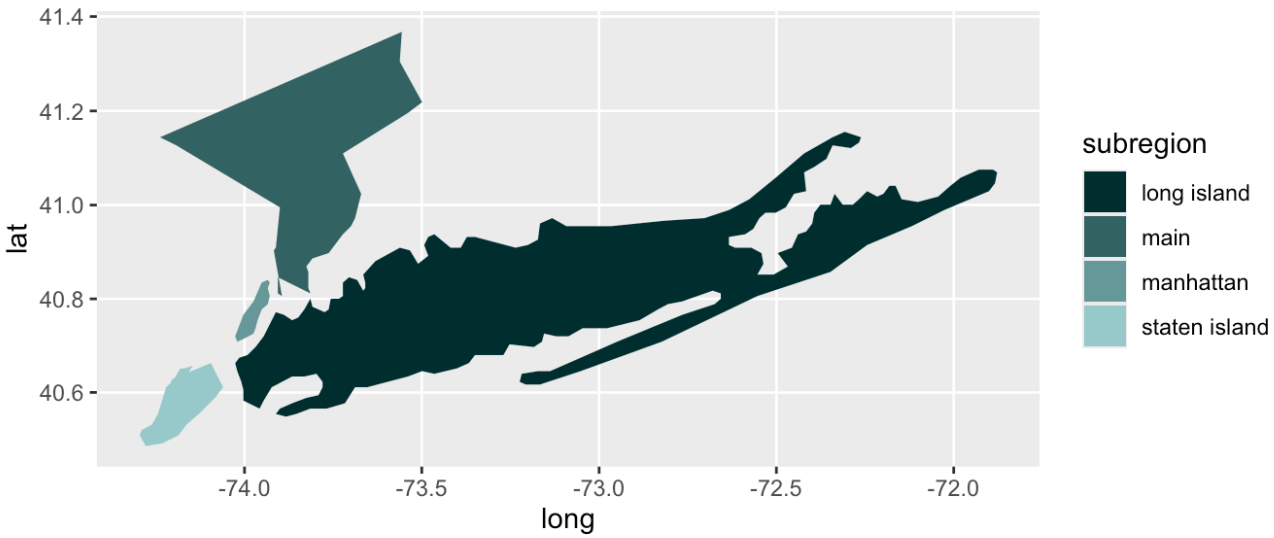
```
          long      lat group order   region subregion
9017 -73.92874 40.80605    34  9050 new york manhattan
9018 -73.93448 40.78886    34  9051 new york manhattan
9019 -73.95166 40.77741    34  9052 new york manhattan
9020 -73.96312 40.75449    34  9053 new york manhattan
9021 -73.96885 40.73730    34  9054 new york manhattan
9022 -73.97458 40.72584    34  9055 new york manhattan
```

```
ggplot(NewYork, aes(x = long, y = lat, group = group, fill = subregion)) +
  geom_polygon() +
  scale_fill_manual(values = c("#003333","#336666", "#669999", "#99CCCC")) +
  coord_map()
```




```
NYC_map <- NewYork |> filter(long >= -74.3 & long <= -71 & lat >= 40.4 & lat <
```

```
ggplot(NYC_map, aes(x = long, y = lat, group = group, fill = subregion)) + geo
  scale_fill_manual(values =c("#003333","#336666", "#669999", "#99CCCC"))
```



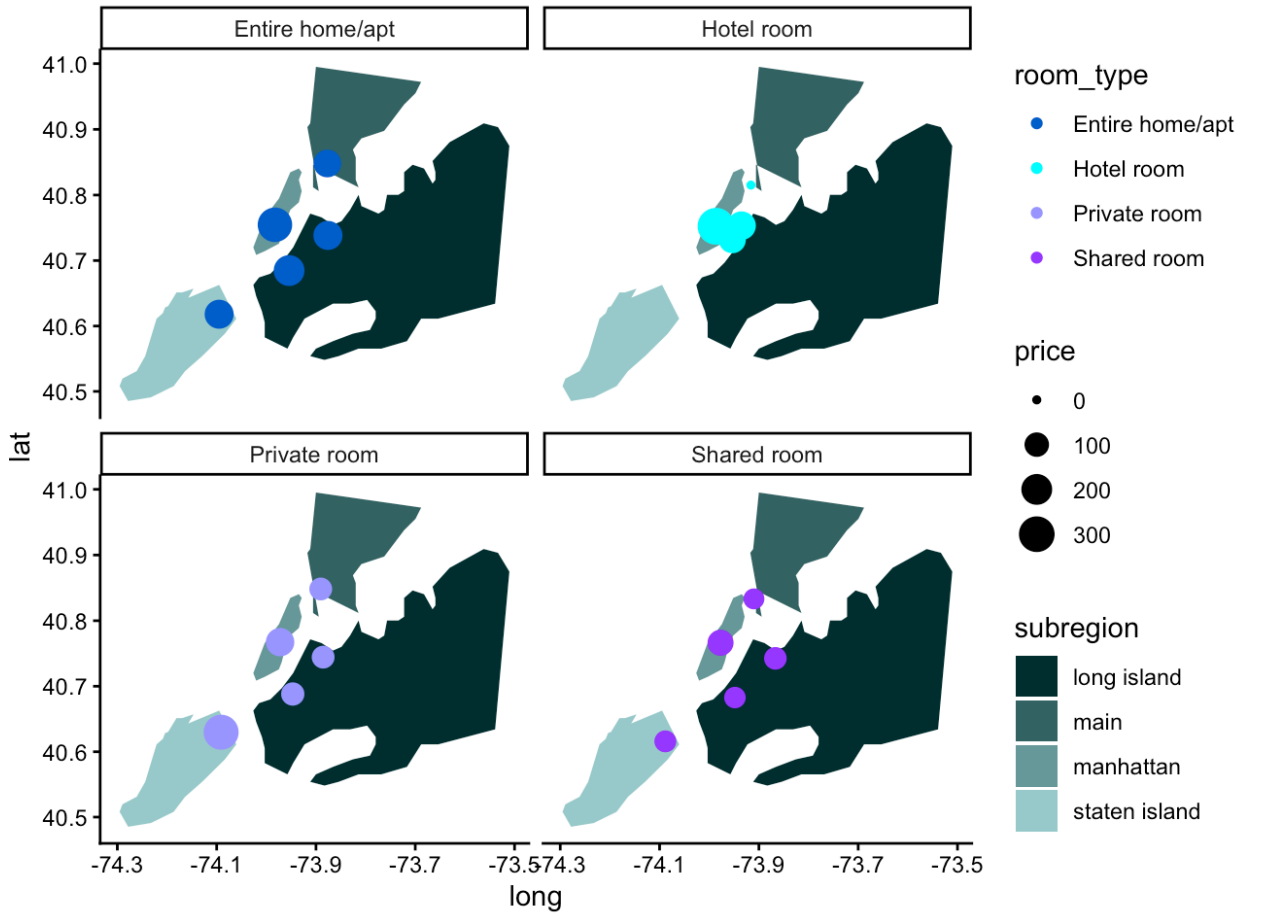
NYC Airbnb map

```
NYCAB_map <- NewYork |> filter(long >= -74.3 & long <= -73.5 & lat >= 40.4 & l

plot_data <-
  combined_dataset |>
  select(price, longitude, latitude, room_type, neighbourhood_group) |>
  group_by(neighbourhood_group, room_type) |>
  summarise(count = n(),
            price = mean(price),
            longitude = median(longitude),
            latitude = median(latitude))
```

```
`summarise()` has grouped output by 'neighbourhood_group'. You can override
using the `.groups` argument.
```

```
ggplot(NYCAB_map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = subregion)) +
  scale_fill_manual(values = c("#003333","#336666", "#669999", "#99CCCC")) +
  geom_point(plot_data,
             mapping = aes(x = longitude, y = latitude,
                           color = room_type, size = price,
                           group = NULL)) +
```

```
  scale_color_manual(values = c("#0060CC","#00FFFF", "#9999FF", "#9933FF")) +
  coord_map() +
  theme_classic() +
  facet_wrap(~room_type)
```

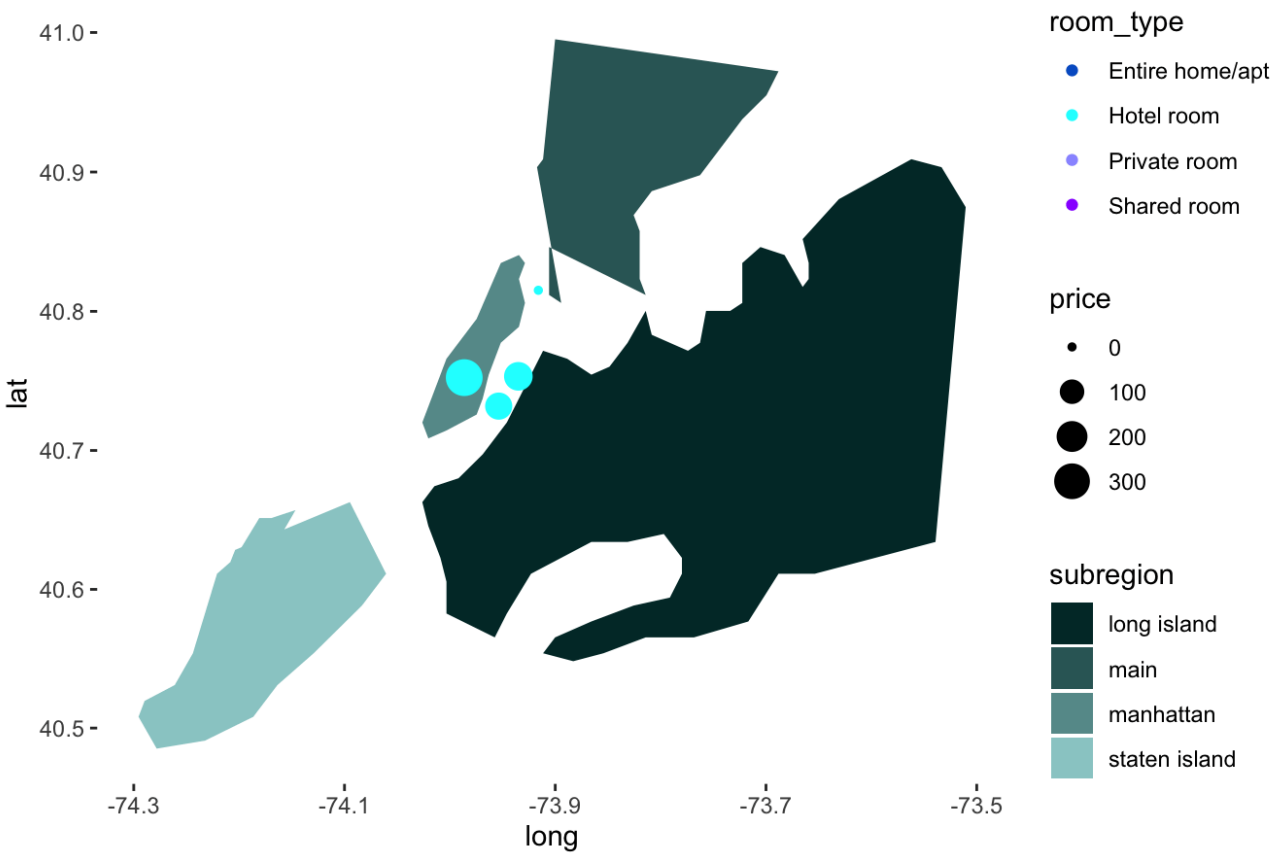


The color of the dots represents different regions of NYC. The hotel room type are mostly around the Manhattan region. The size of the dots are the price. Larger size represents higher price per night. The biggest dots are located in the Manhattan where cost the most expensive to stay in NYC.

- create an animation for the map

```
anim_room_type <- ggplot(NYCAB_map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = subregion)) +
  scale_fill_manual(values = c("#003333","#336666", "#669999", "#99CCCC")) +
  geom_point(plot_data,
             mapping = aes(x = longitude, y = latitude,
                           color = room_type, size = price,
                           group = NULL)) +
  scale_color_manual(values = c("#0060CC","#00FFFF", "#9999FF", "#9933FF")) +
  coord_map() +
  theme(panel.background = element_blank(),
        legend.key = element_blank()) +
  transition_states(states = room_type)
```

```
anim_room_type
```



- Save the animation

```
# anim_save("anim_room_type.gif", animation = anim_room_type)
```

Time series plot

- Table

```
combined_dataset |>
  filter(number_of_reviews > 0) |>
  group_by(Year = year(last_review)) |>
  summarise(Last_Reviews = n()) |>
  arrange(desc(Last_Reviews))
```

```
# A tibble: 13 × 2
    Year Last_Reviews
   <dbl>        <int>
 1  2019        27439
 2  2023        13291
 3  2022         8570
 4  2018         7478
```

```
 5  2017         4409
 6  2016         3982
 7  2015         2154
 8  2020         2109
 9  2021         1592
10  2014          301
11  2013           79
12  2012           37
13  2011           10
```

This data was collected for the activities of Airbnb in the year 2019 and 2023. However, the dataset contains the date of last review received from 2011 to 2023. The date of the last review in other years may be because no one has rented the room since that year so there are no reviews or the tenant did not leave a review.

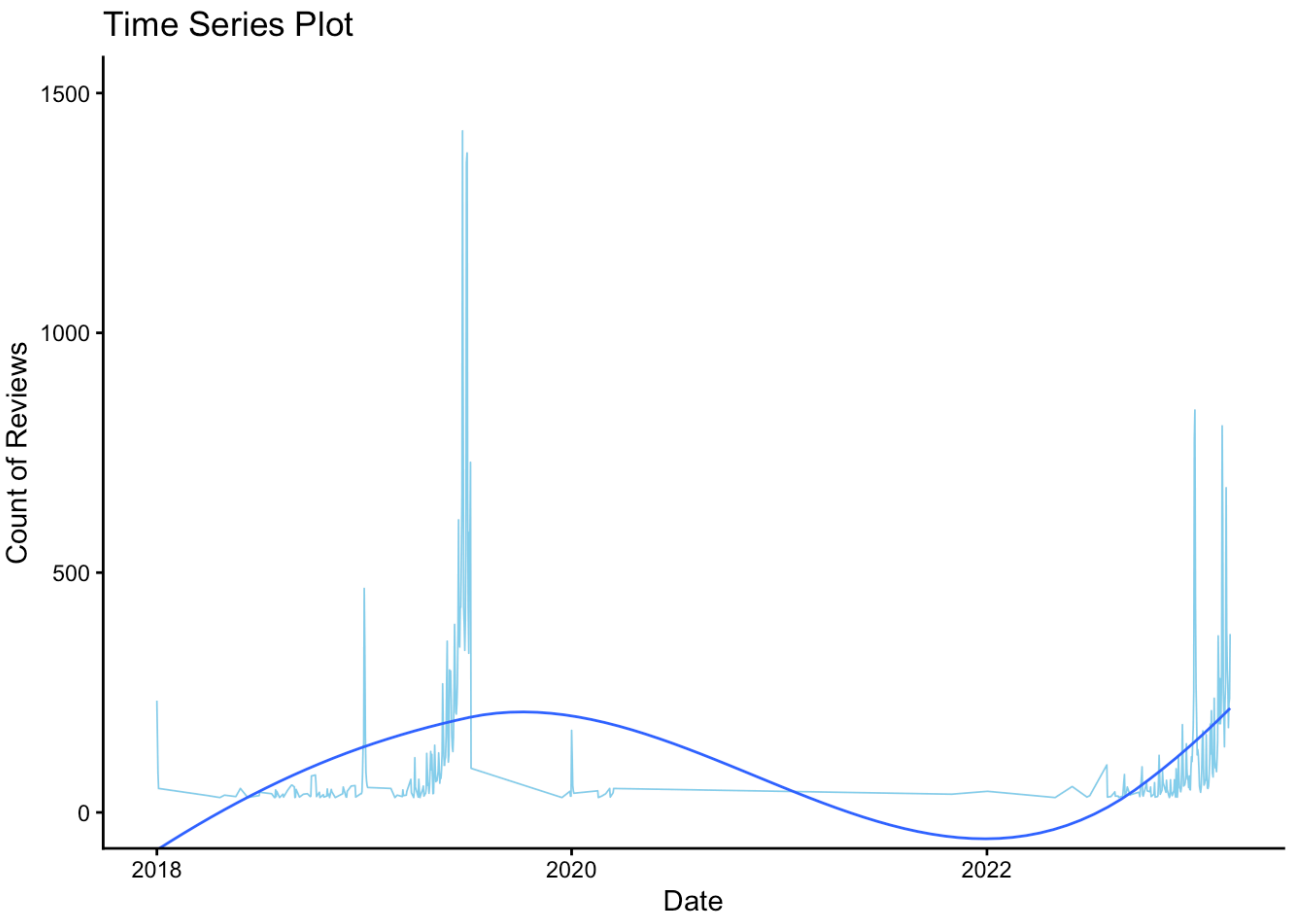
From the table, we can see that more reviews of Airbnb rooms have been updated to more recent years. More people keep booking rooms in Airbnb and wrote review about their experiments.

Although, the data in 2023 only contains up to March, the updated reviews in 2023 is almost half of the updated reviews in the whole year of 2019. The reason may because after covid, more people are excited to have fun out of their house.

```
class(combined_dataset_unique$last_review)
```

```
[1] "Date"
```

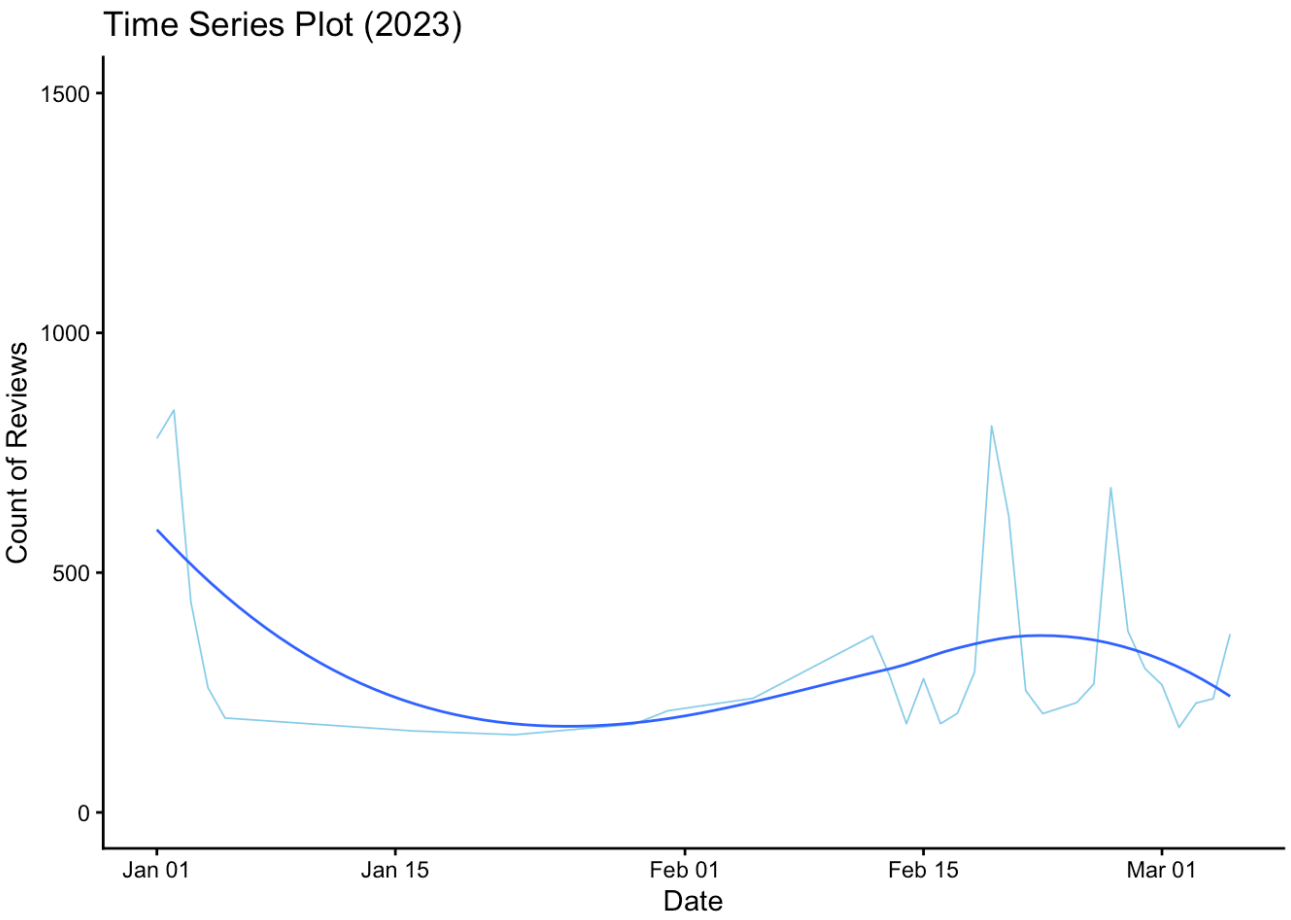
```
combined_dataset_unique |>
  group_by(date = last_review) |>
  summarise(count = n(),
            Avg_price = mean(price, na.rm = T)) |>
  filter(count >= mean(count),
         year(date) <= 2023,
         year(date) >= 2018) |>
  ggplot(aes(x = date, y = count)) +
  geom_line(linewidth = 0.3, color = "skyblue") +
  coord_cartesian(ylim = c(0, 1500)) +
  geom_smooth(se = F, linewidth = 0.5) +
  labs(x = "Date",
       y = "Count of Reviews",
       title = "Time Series Plot") +
  theme_classic()
```




```
class(combined_dataset_unique$last_review)
```

```
[1] "Date"
```

```
combined_dataset_unique |>
  group_by(date = last_review) |>
  summarise(count = n(),
            Avg_price = mean(price, na.rm = T)) |>
  filter(count > 150,
         year(date) <= 2023,
         year(date) >= 2023) |>
  ggplot(aes(x = date, y = count)) +
  geom_line(linewidth = 0.3, color = "skyblue") +
  coord_cartesian(ylim = c(0, 1500)) +
  geom_smooth(se = F, linewidth = 0.5) +
  labs(x = "Date",
       y = "Count of Reviews",
       title = "Time Series Plot (2023)") +
  theme_classic()
```

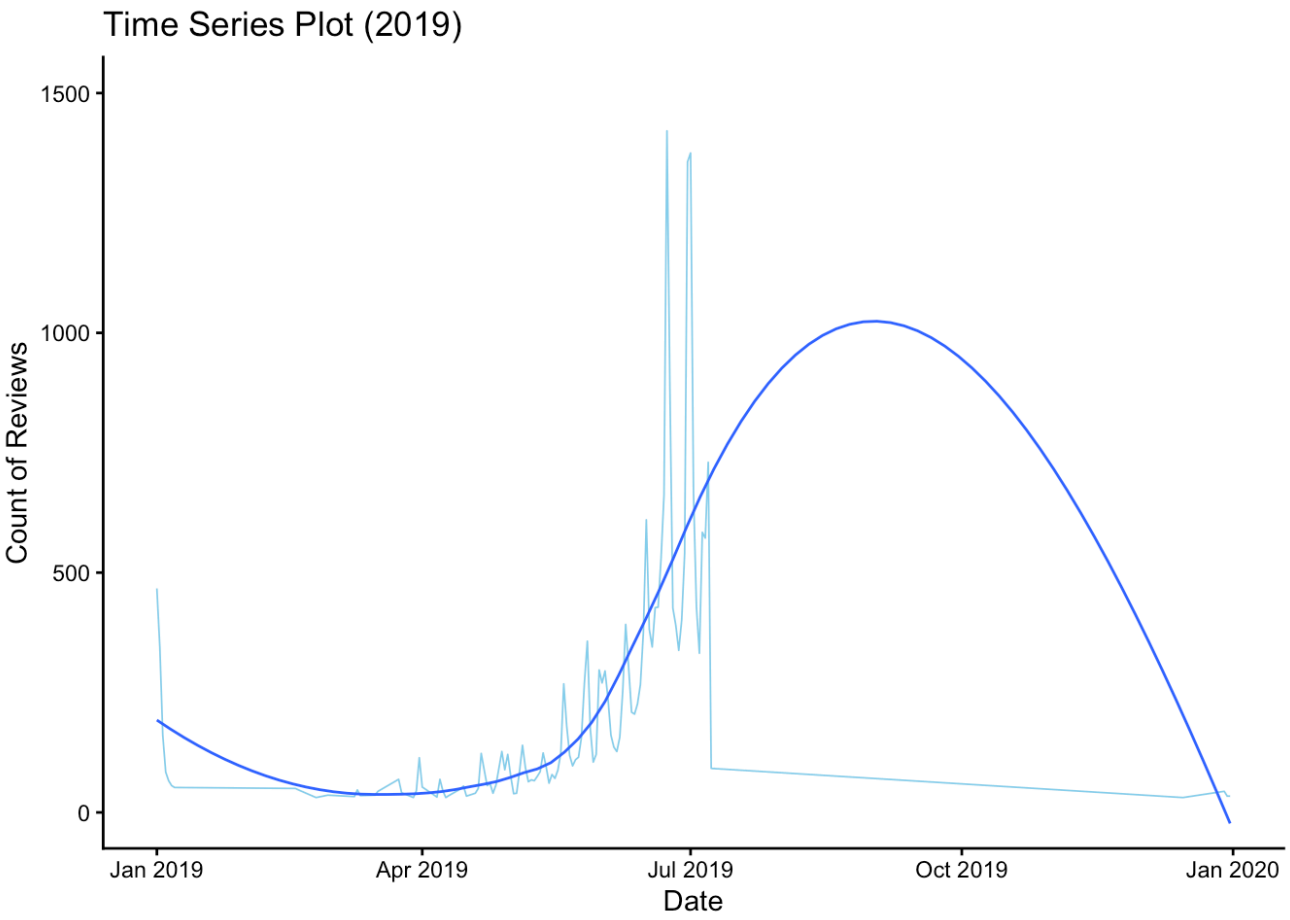


The data in 2023 only contains up to March.

```
class(combined_dataset_unique$last_review)
```

```
[1] "Date"
```

```
combined_dataset_unique |>
  group_by(date = last_review) |>
  summarise(count = n(),
            Avg_price = mean(price, na.rm = T)) |>
  filter(count >= mean(count),
         year(date) <= 2019,
         year(date) >= 2019) |>
  ggplot(aes(x = date, y = count)) +
  geom_line(linewidth = 0.3, color = "skyblue") +
  coord_cartesian(ylim = c(0, 1500)) +
  geom_smooth(se = F, linewidth = 0.5) +
  labs(x = "Date",
       y = "Count of Reviews",
       title = "Time Series Plot (2019)") +
  theme_classic()
```



In 2019, we can see there is a peak of date of last review be leaved around July 2019. The reason may because more people went for vacation during the summer holiday and then went back to work/school after that.

Most Frequently used word of description about the room

```
text_data = combined_dataset |>
unnest_tokens(word, name) |>
anti_join(stop_words) |>
count(word, sort = TRUE)
```

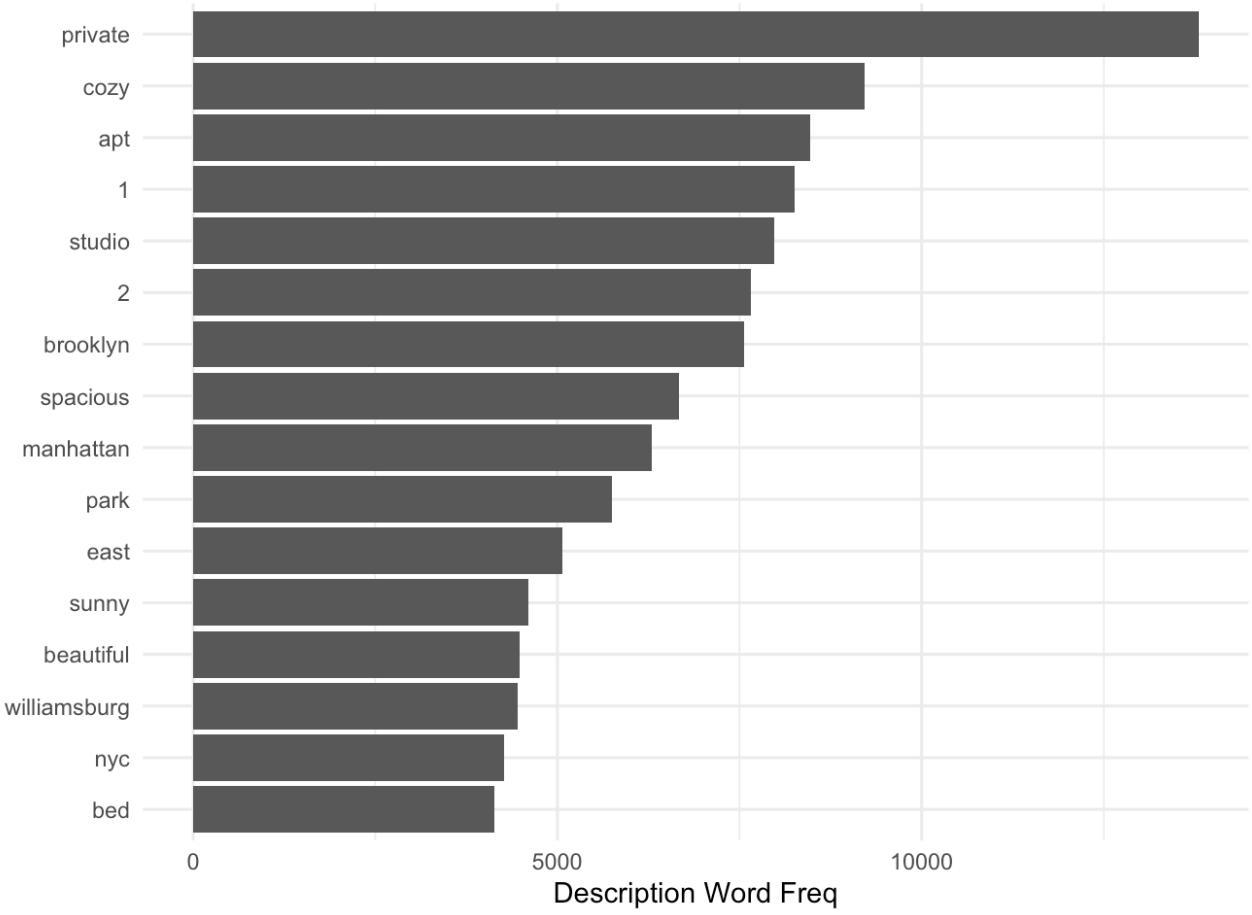
```
Joining with `by = join_by(word)`
```

```
text_data
```

```
# A tibble: 11,820 × 2
   word          n
   <chr>     <int>
 1 bedroom   16464
 2 private   13801
 3 apartment 12510
 4 cozy       9222
 5 apt        8476
```

```
 6 1          8253
 7 studio     7971
 8 2          7658
 9 brooklyn   7566
10 spacious   6665
# ℹ 11,810 more rows
```

```
text_data |>
  filter(n > 4000, word != "bedroom", word != "apartment") |>
  mutate(word = reorder(word, n)) |>
  ggplot(aes(n, word)) +
  geom_col() +
  ylab(" ") +
  xlab("Description Word Freq") +
  theme_minimal()
```



Most commonly used words to describe the listings are: cozy, spacious, and beautiful.