

# Airbnb - Bangkok

Aung Thura Htoo  
2024-08-13

```
#https://insidairbnb.com/bangkok/
```

Set the Directory, Load the data, and "tidyverse"

```
setwd("D:/ee/R-Language/Practice/Dataset")

options(repos = c(CRAN = "https://cran.rstudio.com/"))
install.packages("tidyverse")

## Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)

## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
## C:/Users/lenovo/AppData/Local/Temp/RtmpI247EY/downloaded_packages

library(tidyverse)

## --- Attaching core tidyverse packages --- tidyverse 2.0.0 ---
## ✓ dplyr 1.1.4 ✓ readr 2.1.5
## ✓ forcats 1.0.0 ✓ stringr 1.5.1
## ✓ ggplot2 3.5.1 ✓ tibble 3.2.1
## ✓ lubridate 1.9.3 ✓ tidyr 1.3.1
## ✓ purrr 1.0.2

## --- Conflicts --- tidyverse_conflicts() ---
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag() masks stats::lag()
## I use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

# Load the CSV file into a data frame
data <- read.csv("listings_airbnb_Aug2024.csv")
```

Check the data and Columns

```
# View the first few rows of the data
head(data)

##           id                               name host_id host_name
## 1  27924      Nice room with superb city view 120437  Nuttee
## 2  27979      Easy going landlord,easy place 120541  Emy
## 3  28745      modern-style apartment in Bangkok 123784 Familyroom
## 4  35780 Spacious one bedroom at The Kris Condo Bldg. 3 153780 Sirilak
## 5  48726 Condo with Chaopraya River view 222095 Athitaya
## 6  55681 Sathorn Terrace Apartment(61) 263949  Tor
## neighbourhood,group neighbourhood_latitude longitude room_type price
## 1 NA Ratchathewi 13.75983 106.5413 Entire home/apt 2026
## 2 NA Bang Na 13.66818 106.6167 Private room NA
## 3 NA Bang Kapi 13.75232 106.6246 Private room NA
## 4 NA Din Daeng 13.78823 106.5726 Private room 1286
## 5 NA Rat Burana 13.68556 106.4954 Private room 1653
## 6 Bang Rak 12.71934 106.5176 Private room 1150
## minimum_nights number_of_reviews last_review reviews_per_month
## 1 3 64 2020-01-06 0.43
## 2 1 0
## 3 60 0 NA
## 4 14 6 2024-05-22 0.06
## 5 3 1 2014-02-03 0.01
## 6 2 34 2024-04-17 0.21
## calculated_host_listings_count availability_365 number_of_reviews_ltm license
## 1 2 2 362 0
## 2 2 0 0
## 3 1 0 0
## 4 1 362 0 0 369 365 366 365 362 0 358 ...
## 5 1 365 0
## 6 7 356 5

str(data)
```

```
## 'data.frame': 23651 obs. of 18 variables:
## $ id : num 27924 27979 28745 35780 48736 ...
## $ name : chr "Nice room with superb city view" "Easy going landlord,easy place" "mo
## dern-style apartment in Bangkok" "Spacious one bedroom at The Kris Condo Bldg. 3" ...
## $ host_id : int 120437 120541 123784 153730 222095 263049 294896 302658 272478
## ...
## $ host_name : chr "Nuttee" "Emy" "Familyroom" "Sirilak" ...
## $ neighbourhood_group : logi NA NA NA NA NA ...
## $ neighbourhood : chr "Ratchathewi" "Bang Na" "Bang Kapi" "Din Daeng" ...
## $ latitude : num 13.8 13.7 13.8 13.8 13.7 ...
## $ longitude : num 101.101 101.101 100 ...
## $ room_type : chr "Entire home/apt" "Private room" "Private room" "Private room" ...
## $ price : int 2020 NA 1286 1653 1150 1384 1102 NA 1543 ...
## $ minimum_nights : int 3 1 60 3 2 2 30 1 90 ...
## $ number_of_reviews : int 64 0 0 0 1 34 210 2 0 10 ...
## $ last_review : chr "2020-01-06" "" "" "2024-05-22" ...
## $ reviews_per_month : num 0.43 NA NA 0.06 0.01 0.21 1.29 0.01 NA 0.11 ...
## $ calculated_host_listings_count: int 2 2 1 1 7 7 2 1 1 ...
## $ availability_365 : int 362 0 0 369 365 366 365 362 0 358 ...
## $ number_of_reviews_ltm : int 0 0 0 2 0 5 2 0 0 0 ...
## $ license : chr "" "" "" "" "" ...
```

```
unique_room_type <- unique(data$room_type)
unique_room_type
```

```
## [1] "Entire home/apt" "Private room" "Hotel room" "Shared room"
```

Distribution of the Room Type

```
data %>%
count(room_type) %>%
ggplot(aes(x = reorder(room_type, -n), y = n)) +
geom_bar(stat = "identity", fill = "skyblue", color = "black", width = 0.7) +
geom_text(aes(label = n), vjust = -0.5, color = "black", size = 3.5) +
labs(title = "Distribution of Room Types", x = "Room Type", y = NULL) +
theme_minimal() +
theme(plot.title = element_text(hjust = 0.5))
```



Hosts with multiple listings

```
top_hosts <- data %>%
group_by(host_name, room_type) %>%
summarize(listings_count = n(), groups = "drop") %>%
pivot_wider(
names_from = room_type,
values_from = listings_count,
values_fill = list(listings_count = 0) # Fill in 0 for missing values
) %>%
mutate(
`listings` = `Entire home/apt` + `Private room` + `Shared room` + `Hotel room`
) %>% arrange(desc(listings))

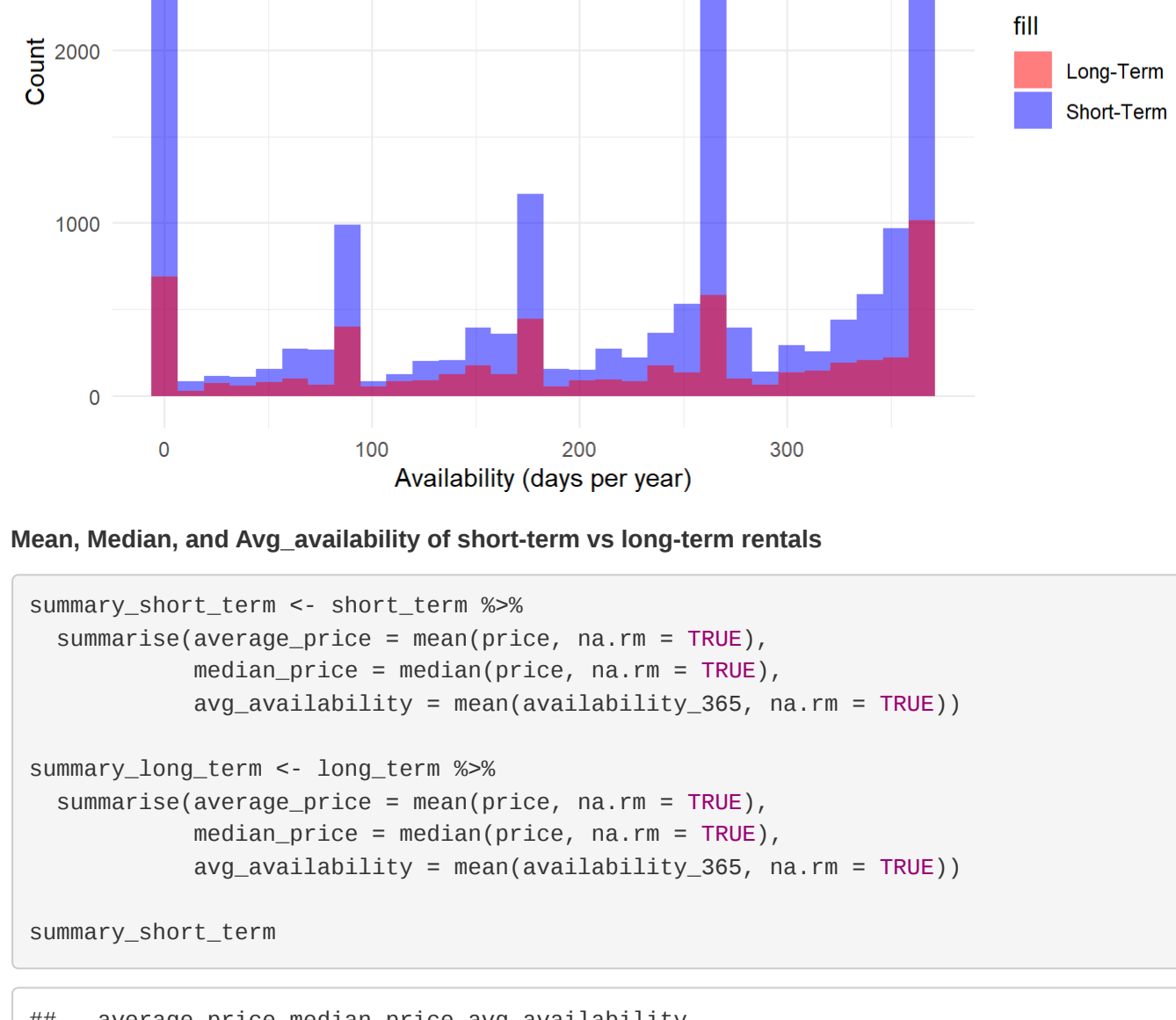
top_hosts

## # A tibble: 6,263 x 6
## host_name `Entire home/apt` `Private room` `Hotel room` `Shared room`
## <chr> <int> <int> <int> <int>
## 1 Alice 80 173 0 1
## 2 Curry 225 0 0 1
## 3 Krittika 175 10 0 0
## 4 Elner 0 153 0 8
## 5 Alex 151 5 1 0
## 6 Tony 145 0 0 0
## 7 Noons 123 0 0 2
## 8 Max 96 10 0 0
## 9 Yang 183 0 0 0
## 10 K 87 3 8 0
## # 16,253 more rows
## # 1 more variable: Listings <int>
```

Availability Distribution for Short-Term vs Long-Term Rentals

```
short_term <- data %>% filter(minimum_nights <= 7)
long_term <- data %>% filter(minimum_nights > 7)

ggplot() +
geom_histogram(data = short_term, aes(x = availability_365, fill = "Short-Term"), bins = 30, alpha = 0.5) +
geom_histogram(data = long_term, aes(x = availability_365, fill = "Long-Term"), bins = 30, alpha = 0.5) +
labs(title = "Availability Distribution for Short-Term vs Long-Term Rentals",
x = "Availability (days per year)",
y = "Count") +
scale_fill_manual(values = c("Short-Term" = "blue", "Long-Term" = "red")) +
theme_minimal()
```



Mean, Median, and Avg\_availability of short-term vs long-term rentals

```
summary_short_term <- short_term %>%
summarize(average_price = mean(price, na.rm = TRUE),
median_price = median(price, na.rm = TRUE),
avg_availability = mean(availability_365, na.rm = TRUE))

summary_long_term <- long_term %>%
summarize(average_price = mean(price, na.rm = TRUE),
median_price = median(price, na.rm = TRUE),
avg_availability = mean(availability_365, na.rm = TRUE))

summary_short_term

## average_price median_price avg_availability
## 1 2644.593 1472.5 221.0468

summary_long_term

## average_price median_price avg_availability
## 1 2074.552 1250 210.4231
```

Clean NA and infinite numbers

```
sum(!is.finite(data$price))

## [1] 4639

sum(is.na(data$price))

## [1] 4639

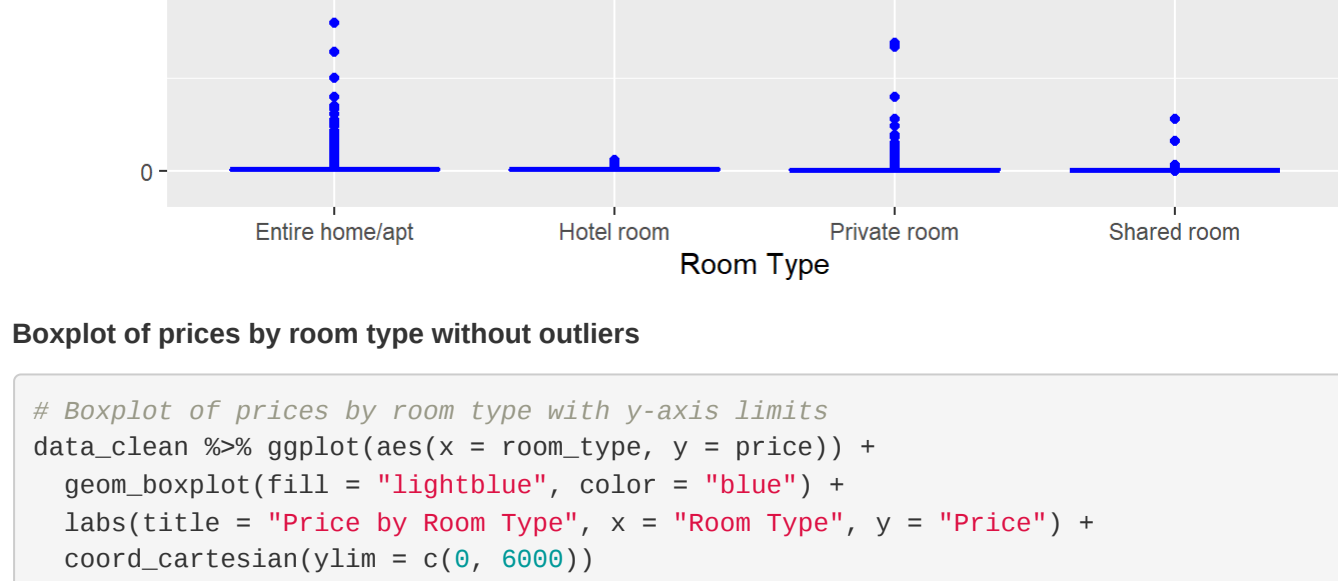
data_clean <- data %>%
filter(!is.finite(price) & !is.na(price))

str(data_clean)
```

```
## 'data.frame': 19012 obs. of 18 variables:
## $ id : num 27924 27979 28745 35780 48736 55681 55686 ...
## $ name : chr "Nice room with superb city view" "Spacious one bedroom at The Kris Co
## ndo Bldg. 3" "Condo with Chaopraya River View" "Sathorn Terrace Apartment(61)" ...
## $ host_id : int 120437 120541 123784 153730 222095 263049 263049 294896 272478 545896 578110 610315
## ...
## $ host_name : chr "Nuttee" "Sirilak" "Athitaya" "Tor" ...
## $ neighbourhood_group : logi NA NA NA NA NA ...
## $ neighbourhood : chr "Ratchathewi" "Din Daeng" "Rat Burana" "Bang Rak" ...
## $ latitude : num 13.8 13.8 13.7 13.7 13.7 ...
## $ longitude : num 101.101 100 101 101 ...
## $ room_type : chr "Entire home/apt" "Private room" "Private room" "Private room" ...
## $ price : int 2020 1286 1653 1150 1384 1102 1543 6024 1469 1190 ...
## $ minimum_nights : int 3 14 3 2 2 30 90 28 30 1 ...
## $ number_of_reviews : int 64 6 1 34 210 2 18 147 0 6 ...
## $ last_review : chr "2020-01-06" "2024-05-22" "2014-02-03" "2024-04-17" ...
## $ reviews_per_month : num 0.43 0.06 0.01 0.21 1.29 0.01 0.11 0.95 NA 0.35 ...
## $ calculated_host_listings_count: int 2 1 7 7 2 1 1 3 ...
## $ availability_365 : int 362 369 365 356 365 362 358 362 365 365 ...
## $ number_of_reviews_ltm : int 0 0 2 0 5 2 0 0 0 5 ...
## $ license : chr "" "" "" "" "" ...
```

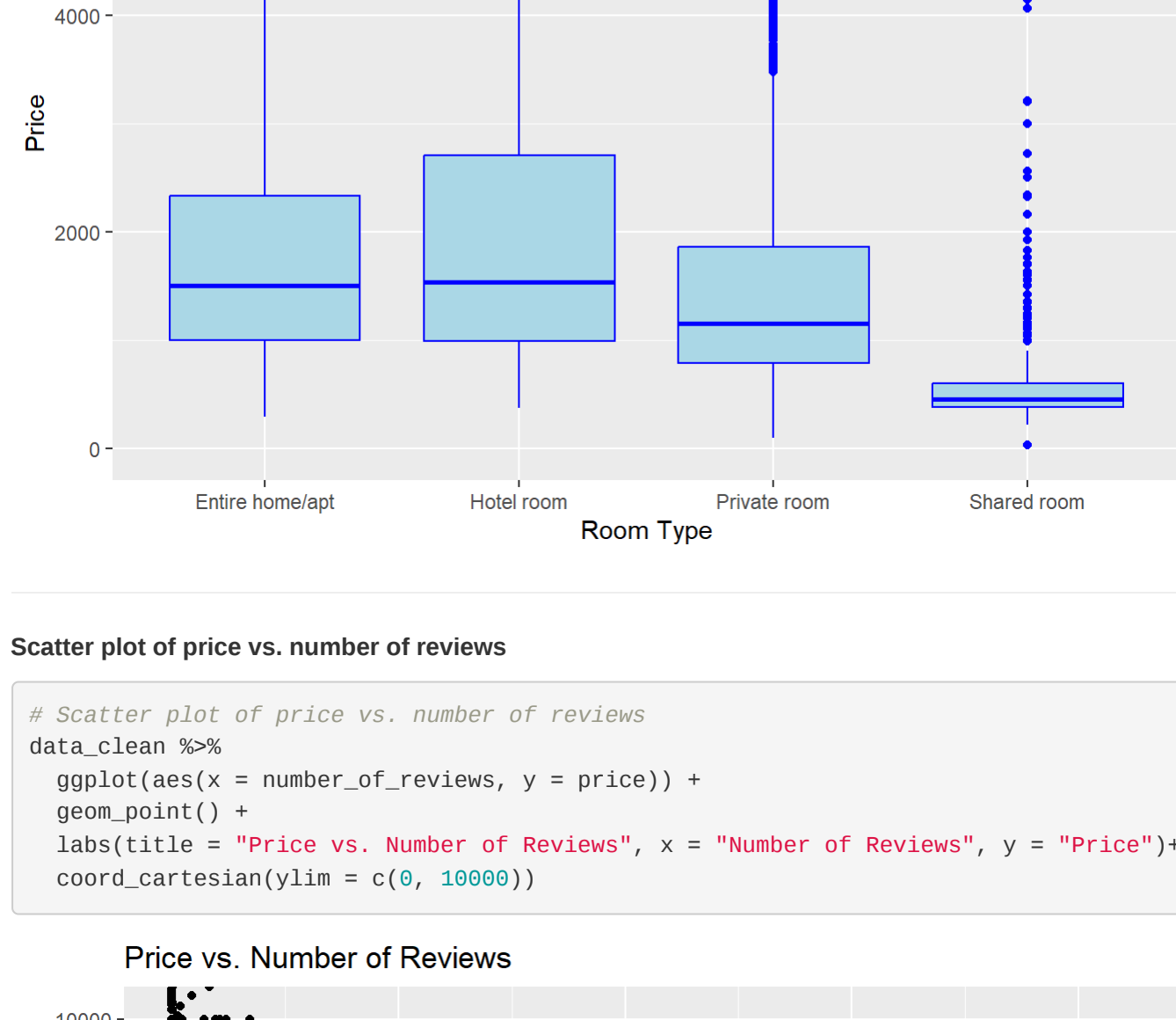
Boxplot of prices by room type

```
# Boxplot of prices by room type
data_clean %>% ggplot(aes(x = room_type, y = price)) +
geom_boxplot(fill = "lightblue", color = "blue") +
labs(title = "Price by Room Type", x = "Room Type", y = "Price")
```



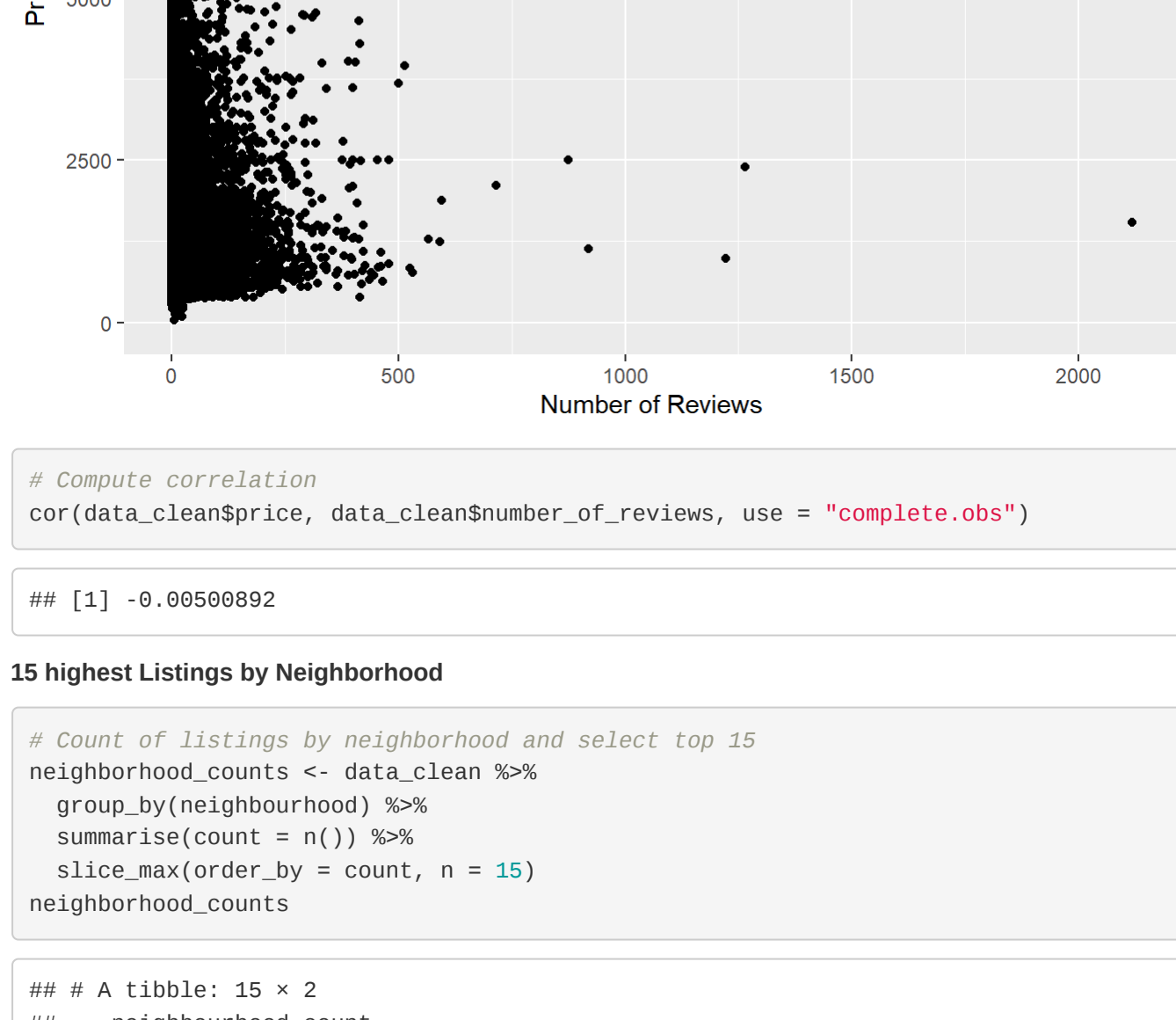
Boxplot of prices by room type without outliers

```
# Boxplot of prices by room type with y-axis limits
data_clean %>% ggplot(aes(x = room_type, y = price)) +
geom_boxplot(fill = "lightblue", color = "blue") +
labs(title = "Price by Room Type", x = "Room Type", y = "Price") +
coord_cartesian(ylim = c(0, 6000))
```



Scatter plot of price vs. number of reviews

```
# Scatter plot of price vs. number of reviews
data_clean %>%
ggplot(aes(x = number_of_reviews, y = price)) +
geom_point() +
labs(title = "Price vs. Number of Reviews", x = "Number of Reviews", y = "Price") +
coord_cartesian(ylim = c(0, 10000))
```



```
# Compute correlation
cor(data_clean$price, data_clean$number_of_reviews, use = "complete.obs")
```

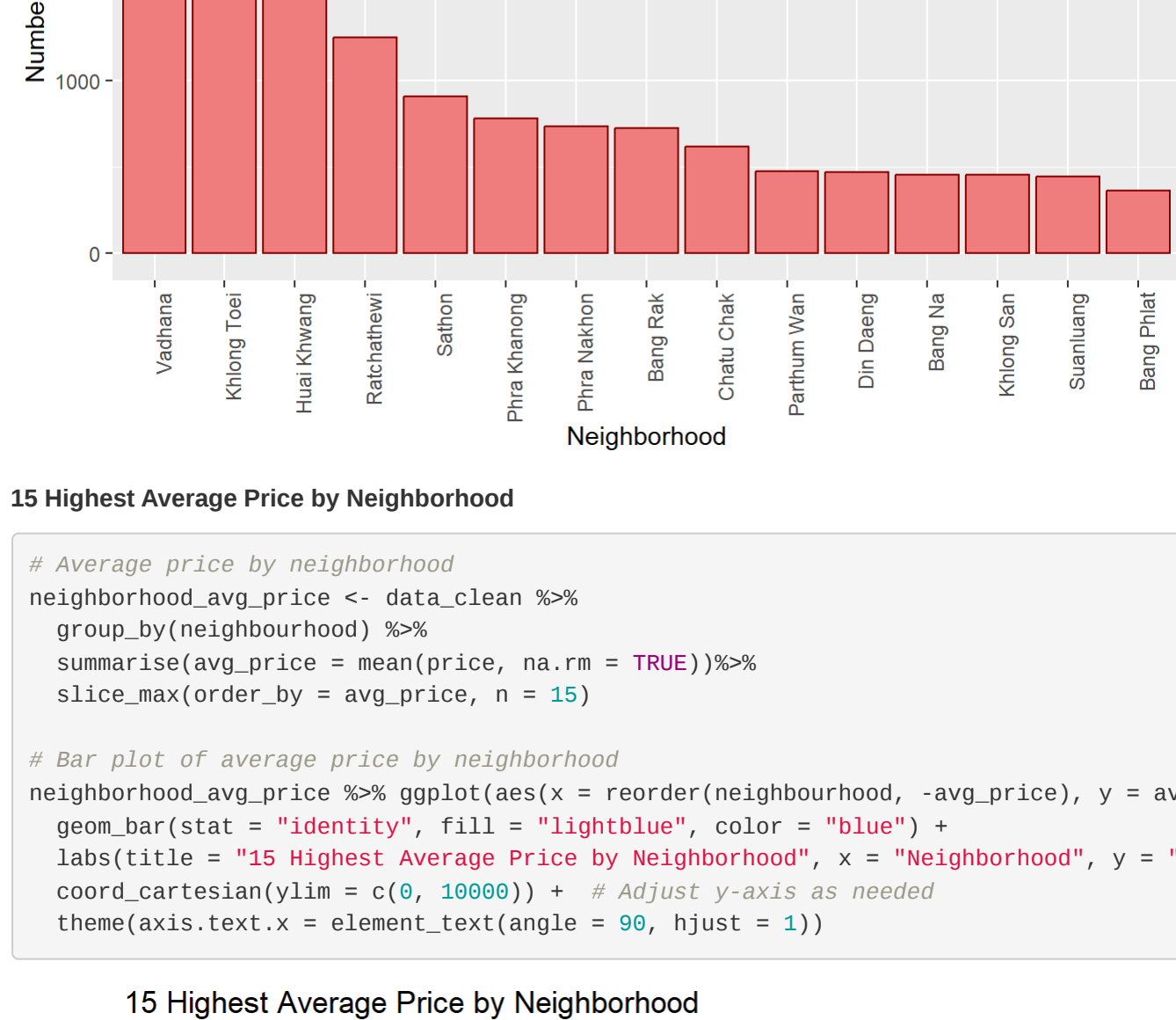
```
## [1] -0.00508892
```

15 highest Listings by Neighborhood

```
# Count of listings by neighborhood and select top 15
neighbourhood_counts <- data_clean %>%
group_by(neighbourhood) %>%
summarize(count = n()) %>%
slice_max(order_by = count, n = 15)
neighbourhood_counts
```

```
## # A tibble: 15 x 2
## neighbourhood count
## <chr> <int>
## 1 Vadhana 3295
## 2 Khlong Toei 2870
## 3 Huai Khwang 1892
## 4 Ratchathewi 1250
## 5 Sathon 996
## 6 Phra Khanong 781
## 7 Phra Nakhon 735
## 8 Bang Rak 725
## 9 Chatu Chak 617
## 10 Parthue Wan 473
## 11 Din Daeng 470
## 12 Bang Na 454
## 13 Khlong San 452
## 14 Suanluang 444
## 15 Bang Phlat 363

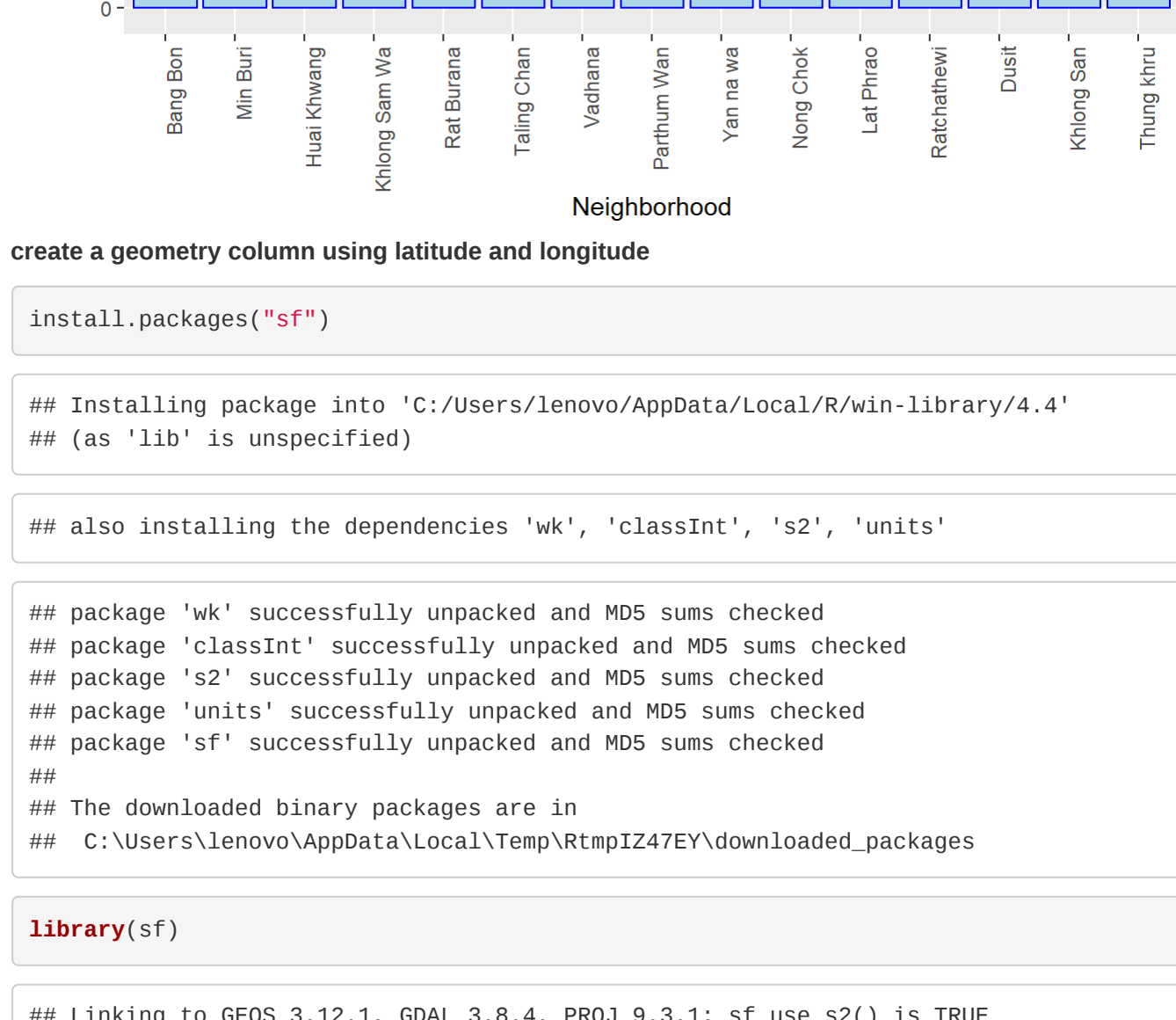
# Bar plot of listing counts by neighborhood, arranged in descending order
neighbourhood_counts %>% ggplot(aes(x = reorder(neighbourhood, -count), y = count)) +
geom_bar(stat = "identity", fill = "lightcoral", color = "darkred") +
labs(title = "15 highest Listings by Neighborhood", "neighbourhood", y = "Number of Listings") +
theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



15 Highest Average Price by Neighborhood

```
# Average price by neighborhood
neighbourhood_avg_price <- data_clean %>%
group_by(neighbourhood) %>%
summarize(avg_price = mean(price, na.rm = TRUE)) %>%
slice_max(order_by = avg_price, n = 15)
```

```
# Bar plot of average price by neighborhood
neighbourhood_avg_price %>% ggplot(aes(x = reorder(neighbourhood, -avg_price), y = avg_price)) +
geom_bar(stat = "identity", fill = "lightblue", color = "blue") +
labs(title = "15 Highest Average Price by Neighborhood", x = "Neighbourhood", y = "Average Price") +
coord_cartesian(ylim = c(0, 10000)) + # Adjust y-axis as needed
theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



create a geometry column using latitude and longitude

```
install.packages("sf")

## Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)

## also installing the dependencies 'wk', 'classInt', 's2', 'units'

## package 'wk' successfully unpacked and MD5 sums checked
## package 'classInt' successfully unpacked and MD5 sums checked
## package 's2' successfully unpacked and MD5 sums checked
## package 'units' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
## C:/Users/lenovo/AppData/Local/Temp/RtmpI247EY/downloaded_packages

library(sf)

## Linking to GEOS 3.12.1, GDAL 3.8.4, PROJ 9.3.1; sf_use_s2() is TRUE

data_clean_sf <- st_as_sf(data_clean, coords = c("longitude", "latitude"), crs = 4326)

install.packages("plotly")

## Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)

## also installing the dependencies 'later', 'htmlwidgets', 'lazyeval', 'crosstalk', 'promises'

## package 'later' successfully unpacked and MD5 sums checked
## package 'htmlwidgets' successfully unpacked and MD5 sums checked
## package 'lazyeval' successfully unpacked and MD5 sums checked
## package 'crosstalk' successfully unpacked and MD5 sums checked
## package 'promises' successfully unpacked and MD5 sums checked
## package 'plotly' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
## C:/Users/lenovo/AppData/Local/Temp/RtmpI247EY/downloaded_packages

library(plotly)

##
## Attaching package: 'plotly'

##
## The following object is masked from 'package:ggplot2':
##
## last_plot

##
## The following object is masked from 'package:stats':
##
## filter

##
## The following object is masked from 'package:graphics':
##
## layout
```

Geographical Distribution of Airbnb - Bangkok Listings along with their Price

```
# Creating the ggplot object
p <- ggplot(data_clean_sf) +
geom_sf(aes(color = neighbourhood, text = paste("Neighbourhood:", neighbourhood, "<br>Price (THB):", price))) +
theme_minimal() +
labs(title = "Geographical Distribution of Airbnb - Bangkok Listings",
subtitle = "Colored by Neighbourhood",
x = "Longitude",
y = "Latitude") +
theme(legend.position = "none") # Remove the legend

## Warning in layer_sf(geom = GeomSf, data = data, mapping = mapping, stat = stat,
## : Ignoring unknown aesthetics: text

# Converting to an interactive plot
p_interactive <- ggplotly(p, tooltip = "text")

# Displaying the interactive plot
p_interactive
```

