Airbnb in the Big Apple:
Understanding Room Types, Prices & Location Trends in New York City

Megha Moncy, Mohith Surya Kiran Kasula, Nisha Thakur, Pallavi Singh, Pallavi Vaswani
[memoncy, mkasula, nithakur, singpall, pvaswan] @iu.edu
Indiana University-Purdue University, Indianapolis, USA

Abstract: This study investigates factors affecting pricing trends in the New York City Airbnb market using a comprehensive dataset of local listings. Focusing on two main research questions: (1) the significance of price differences across room types, and (2) the correlation between a listing's distance from the city center and its pricing, we applied statistical testing methods to reveal insights into the market. Our findings offer valuable guidance for hosts and guests in making informed decisions about pricing strategies and location preferences within the NYC Airbnb landscape.
Keywords: New York City; Airbnb; pricing patterns; room types; city center; statistical testing; Airbnb; Brooklyn; Manhattan; Bronx; Staten Island; Queens.

1 Project Scope

1.1 Introduction

The New York City Airbnb dataset provides valuable insights into the local Airbnb market by enabling analysis of pricing patterns, host behavior, accommodation types, and neighborhood distribution. Studying these variables can help identify factors influencing pricing, uncover patterns in listing management, and determine the prevalence of various accommodation types across neighborhoods. Additionally, the dataset allows users to investigate popular neighborhoods, market saturation, and characteristics that attract guests. This information empowers stakeholders to make informed decisions, develop strategies, and uncover trends and opportunities in the ever-evolving New York City Airbnb landscape.

1.2 Aim

To perform a comparative analysis of the average price among various room types and examine the relationship between the distance of Airbnb listings from the city center and their pricing in New York City.

1.3 Purpose

The purpose of this analysis is to understand the factors influencing pricing in the New York City Airbnb market, enabling hosts to optimize pricing strategies and guests to make informed accommodation choices. Additionally, the findings may contribute to a broader understanding of the sharing economy's impact on urban housing markets.

1.4 Hypothesis:

1. Null Hypothesis: The average price of reservations made available by Airbnb is the same across all room types.
   Alternate Hypothesis: The average price of reservations made available by Airbnb differs significantly among all room types.

2. Null Hypothesis: There is no significant association between the distance of an Airbnb from the city center and the price of Airbnb.
   Alternate Hypothesis: There is a significant association between the distance of an Airbnb from the city center and the price of the Airbnb.

2 Methodology

Data Description: Acquired an extensive dataset of NYC Airbnb listings with details like listing name, hostname, neighborhood, longitude, latitude, room type, price, availability, and reviews.
Data Cleaning & Outlier Treatment: Tried finding null values but there were none, identified and treated outliers using capping methods to ensure accurate analysis.
Distance Calculation: Calculated the distance of listings from Times Square using the 'distVincentySphere' function and created "close_to_center" and "far_from_center" sub-categories based on median distance.
Data Visualization: Created various visualizations to explore the distribution of prices and other variables and average prices for each distance category.
Normality Testing: Assessed data normality using Q-Q plots and histograms.
Non-parametric Testing: Conducted Kruskal-Wallis, Mann-Whitney U, and permutation tests to compare average prices among room types and distance categories and validated results using Dunn's test and bootstrapping techniques.
Interpretation: Analyzed test outcomes and visualizations to conclude pricing factors in the NYC Airbnb market and discussed implications for hosts and guests.

2.1 Data Description

The New York City Airbnb dataset contains detailed information about Airbnb listings in the city, covering various aspects such as listing details, host information, location, and reviews. Here is a description of the main features included in the dataset:

Variable Name	Variable Description	Type of Variable
id	Unique identifier assigned to each Airbnb listing	Nominal
name	Title or name provided by the host for the listing	Nominal
host_id	Unique identifier assigned to each host	Nominal
host_name	Name of the host managing the listing	Nominal
neighbourhood_group	Borough in which the listing is located	Nominal
neighbourhood	Specific neighborhood within the borough where the listing is	Nominal
latitude	Geographic latitude coordinate of the listing	Numerical
longitude	Geographic longitude coordinate of the listing	Numerical
room_type	Type of accommodation offered in the listing	Categorical
price	Price per night for the listing, in USD	Numerical
minimum_nights	Minimum number of nights a guest must stay for a booking	Numerical
number_of_reviews	Total number of reviews the listing has received from guests	Numerical
last_review	Date of the most recent review for the listing	Categorical
reviews_per_month	Average number of reviews the listing receives per month	Numerical
calculated_host_listings_count	Total number of listings that the host manages on Airbnb	Numerical
availability_365	Number of days in a year when the listing is available for booking	Numerical

2.2 Data Collection and Extraction

The New York City Airbnb dataset was collected using Airbnb's public API, which provides access to listing and review data. Dgomonov curated the dataset and is available on Kaggle. Although the data collection method is not explicitly mentioned, it is reasonable to assume that web scraping techniques and/or API calls were employed to gather the data. The dataset spans from 2010 to 2019, offering a rich source of information for analyzing trends, patterns, and growth in the New York City Airbnb market.

2.3 Data Cleaning and Capping the Outliers

The 'price', 'latitude', and 'longitude' columns of the New York City Airbnb dataset were checked for null values. The 'price, latitude' and ‘longitude’ column contained outliers. The Capping Method replaced outliers with the maximum or minimum value within a specific range to address this. After treating the outliers, it was found that some listings had a price of $0 per night, while the highest price in the dataset was $334 per night. The lower and upper bounds for detecting outliers were calculated using the 1.5 * IQR rule for the 'price', 'latitude', and 'longitude' columns. By capping the outliers at these bounds, the box plot more precisely represented the data's central tendency, spread, and non-outlier values.

3 Data Analysis

To view and comprehend our data, we used RStudio. Many built-in R packages were utilized, including ggridges, hrbrthemes, ggthemes, sos, tidyverse, and dunn.test. The ggridges package enables the creation of density plots for visualization of multiple overlapping distributions. The hrbrthemes and ggthemes packages provide additional themes and styling options for ggplot2 visualizations, thereby expanding the range of available visualizations. The sos package streamlines the search process for R functions across multiple packages, making finding relevant functions for specific tasks easier. The dunn.test package provides tools for conducting Dunn's test, a non-parametric pairwise comparison test, after performing a Kruskal-Wallis test. Finally, the tidyverse package is a suite of packages designed for data manipulation and visualization, including dplyr, ggplot2, tidyr, and others, which are essential tools for data analysis in R.

3.1 Descriptive Statistics.

The descriptive analysis of the New York City Airbnb dataset reveals a wide range of prices, with an average price per night of $152.70 and a maximum price of $10,000. The average minimum stay required by listings is 7.03 nights, with significant variation in minimum stay requirements. The listings have an

average availability of 112.8 days per year, ranging from completely unavailable to available year-round. The dataset includes three unique values for the 'room_type' variable: Entire home/apt, Private room, and Shared room. The 'price' column had a range of values from 0 to 334, while the 'longitude' column ranged from -74.05 to -73.86 and the 'latitude' column ranged from 40.58 to 40.87.

3.2 Statistical Testing for Hypothesis-1

Statistical Test	Assumptions	Results	Limitations
Bartlett's test	Normality, Independent samples	The test was performed on the price variable with room_type as the grouping variable. Bartlett's K-squared value, which is 4464.1. The degrees of freedom (df) are 2, which is the number of groups minus 1. The p-value is less than 2.2e-16, which is very small and indicates strong evidence against the null hypothesis of equal variances.	Assumes normality and independence of observations; sensitive to departures from normality
Kruskal-Wallis test	Independence of observations, Homogeneity of variance	The test yielded a chi-squared value of 22434 with 2 degrees of freedom and a p-value of less (2.2e-16), indicating strong evidence to reject the null hypothesis.	Assumes rough equality of sample sizes; less powerful than parametric tests; does not identify which group differences are significant
Dunn's test (Post Hoc test)	Independence	Entire home/apt vs. Private room: Z = 145.1443, adjusted p-value = 0.0000* Entire home/apt vs. Shared room: Z = 57.33895, adjusted p-value = 0.0000* Private room vs. Shared room: Z = 12.95394, adjusted p-value = 3.348003e-38* All three pairwise comparisons have adjusted p-values less than the significance level (alpha = 0.05), which means that there are significant differences in the variable "price" between all pairs of room types. In conclusion, the Kruskal-Wallis test and Dunn's post-hoc test reveals significant differences in the variable “price” among all three-room types: Entire home/apt, Private room, and Shared room.	Dunn's test is a conservative test, which means that it is less likely to find significant differences between groups if they truly exist. It does not control the overall type I error rate when making multiple comparisons, which can lead to an increased risk of false positives. It is only applicable for non-parametric tests, which can limit its usefulness in certain research scenarios where parametric tests may be more appropriate.

3.3 Statistical Testing for Hypothesis-2

Statistical Test	Assumptions	Results	Limitations
Normality Test using QQ Plots	1. Continuous data: The data under analysis should consist of continuous values, meaning it can take on any value within a given range or interval. 2. Independence of observations: The data points should be independent of each other. 3.The data points in the QQ plot should fall	The points in the QQ plots above deviate from the red reference line, it indicates that the 'price', 'latitude' & 'longitude' columns may not follow a normal distribution. The specific pattern of deviation can give insight into the nature of the data's distribution (e.g., skewness, kurtosis, or other non-normal characteristics). Fig. 4, 5 & 6	Visual interpretation; may not be conclusive

	approximately along the reference line if the data follows a normal distribution.		
Mann-Whitney U Test	1. Independence of observations 2. Identical distributions under the null hypothesis	1. There is no clear pattern or clustering of points within each distance category, therefore, it supports the assumption of independence between the observations. This assumption is met as indicated in Fig. 7. 2. The null hypothesis for the Mann-Whitney U test states that the two populations should have identical distributions. Since the two density curves largely overlap, it indicates that the price distributions are identical between the two categories. This assumption is met as indicated in Fig. 8. Interpretation of Mann-Whitney U Test: Based on the extremely small p-value (< 2.2 x 10^-16), we reject the null hypothesis in favor of the alternative hypothesis. This suggests that there is a significant association between the distance of an Airbnb listing from the city center and the price of the listing in New York City Airbnb listings.	Assumes ordinal data; may not be robust to violations of assumptions
Bootstrapping	1.Representativeness: The original sample should be representative of the population from which it was drawn. 2. Independence of observations: The data points in the original sample should be independent of each other. To validate the Mann-Whitney U Test, we will perform Bootstrapping. It is a resampling technique that can be used to validate the Mann-Whitney U test results. It involves repeatedly sampling from the original data with replacement to create multiple bootstrap samples and perform the Mann-Whitney U Test on each sub-sample.	The bootstrapped Mann-Whitney U test yields a p-value of 0.505. This suggests that there is no significant difference in 'capped_price' between the 'distance_category' groups. Interpretation- This means that Mann Whitney U test was incorrectly testing the hypothesis. The observed difference in the original Mann-Whitney U test statistic is likely due to random variation rather than a true difference between the groups.	Dependent on the quality of the original sample; may not be reliable if the sample is not representative
Permutation Test	1. Independence of observations: The data points within each group and between groups should be independent. 2. Large enough sample size: Although permutation tests can work with relatively small sample sizes, larger samples generally provide more accurate and reliable results. To further validate the hypothesis, we performed a Permutation test. It is a non-parametric test that can be used to compare	A significant difference between groups. Interpretation- The p-value of 0 suggests strong evidence against the null hypothesis. Therefore, the observed difference is not due to random variation, as shown by bootstrapping, but rather indicates a true difference in pricing between the two categories.	Computationally intensive; assumes exchangeability under the null hypothesis; may not be robust to violations of assumptions

	distributions of two groups. It involves randomly permuting the observations between the groups and calculating the test statistic many times to estimate the null distribution.		

3.6 Data Visualization

Fig 1. The entire distribution of "price" values & distribution of "price" variable for a subset of data, where the price <= to 300

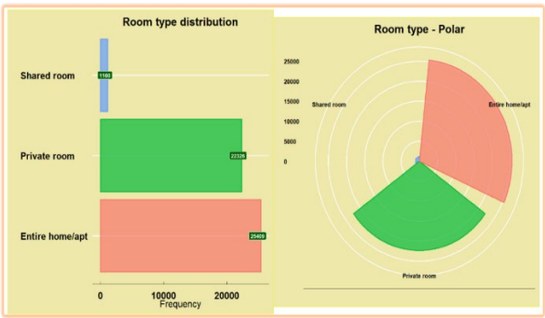


Fig 2. Bar chart and polar bar chart showing the majority of listings in dataset are for entire homes/apt followed by private and shared rooms

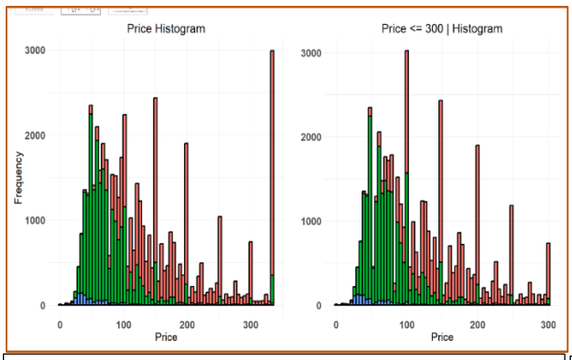


Fig 3. Histogram showing Price Vs Room_Type

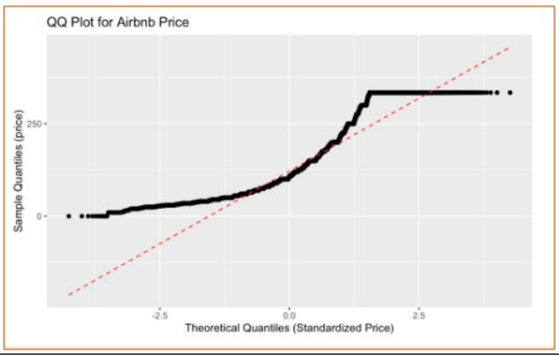


Fig 4. QQ Plot Deviations: Non-Normal Distribution of 'Price'

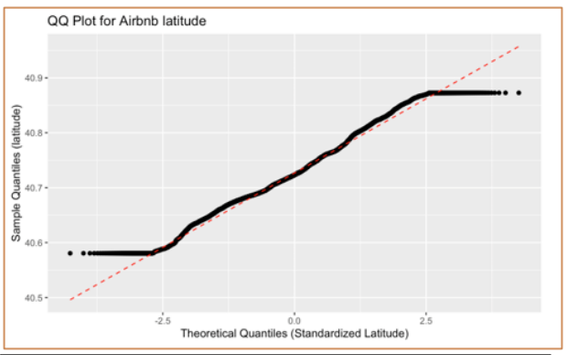


Fig 5. QQ Plot Deviations: Non-Normal Distribution of 'Latitude'

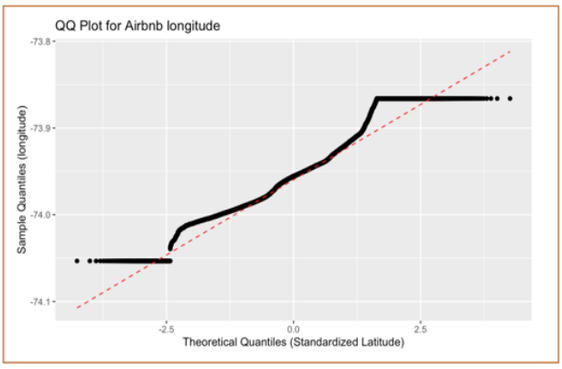


Fig 6. QQ Plot Deviations: Non-Normal Distribution of 'Longitude'

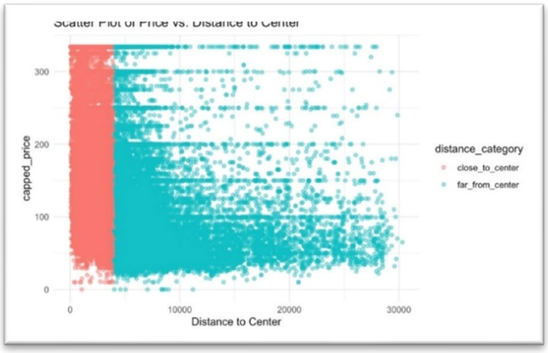


Fig 7. Scatter plot showing clustering of points within categories- 'close_to_center' and 'far_from_center'

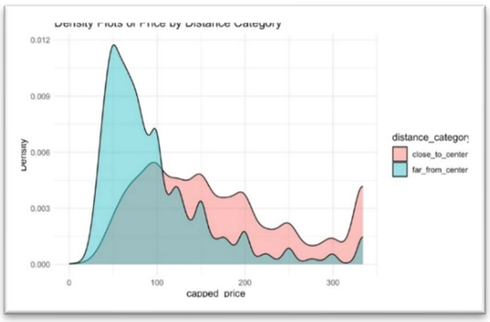


Fig 8. Density Curves showing overlapping distributions in 'Close_to_Center' and 'Far_from_Center' Categories

4 Results

Hypothesis 1: Room Types and Pricing

Our analysis revealed distinct price variations across room types in the New York City Airbnb market. The average price for entire home/apartment listings was found to be the highest, ranging from $160 to $180 per night. Private room listings followed, with an average price range of $70 to $83 per night. Shared room listings were the most affordable, ranging from $45 to $65 per night. The Kruskal-Wallis rank sum test demonstrated strong evidence to reject the null hypothesis, indicating that the average price of reservations differs significantly across room types.

Hypothesis 2: Proximity to the City Center and Pricing

Our findings indicated a substantial correlation between the distance of an Airbnb listing from the city center and its pricing. Properties closer to the city center exhibited higher average prices, with "close_to_center" listings having a mean price of $168.03 and a median price of $150. On the other hand, "far_from_center" listings had a lower average price of $97.92 and a median price of $79. The Mann-Whitney U test revealed an extremely small p-value, providing strong evidence to reject the null hypothesis. This suggests a significant association between the distance of a listing from the city center and its price.

These findings offer valuable insights for both hosts and guests in the New York City Airbnb market. Hosts can strategically determine the pricing based on room types and consider the impact of location. On the other hand, guests can make more informed decisions by considering their preferences, budget constraints, and desired proximity to the city center when selecting accommodations.

5 Conclusions

In conclusion, this study explored the factors influencing pricing trends in the New York City Airbnb market. Analyzing a comprehensive dataset of local listings and utilizing statistical testing techniques, we addressed two main research questions: the difference in average prices across distinct room types and the correlation between distance from the city center and pricing.

Our findings revealed valuable insights into the intricacies of the New York City Airbnb market. Firstly, we observed that the average price of Entire home/apartment listings was the highest, followed by Private and Shared room listings. This suggests that guests are willing to pay more for exclusive accommodation while shared spaces are more affordable. Secondly, our analysis demonstrated a substantial correlation between the distance of an Airbnb listing from the city center and its pricing. Listings closer to the city center exhibited higher prices than those farther away. This suggests that the convenience and proximity to key attractions and amenities in the city center are valued by guests, leading to higher price premiums for such locations.

Overall, this study contributes to understanding the New York City Airbnb market and provides a foundation for future research. By continuously analyzing market trends and factors influencing pricing, stakeholders can adapt and optimize their strategies to thrive in the dynamic and competitive landscape of the Airbnb marketplace.

6 References

1. New York City Airbnb Open Data. (2019). Kaggle. https://www.kaggle.com/datasets/dgomonov/new-york-city-airbnb-open-data
2. R code for descriptive statistics: Dimensions, summary, and min-max values

7 Appendix

```r
#No of rows and columns
dim(data)
```

```
[1] 48895    16
```

```r
# Generating a summary of all data attributes with the summary() function
summary(data)
```

RStudio: Notebook Output

```
'data.frame':   48895 obs. of  8 variables:
 $ neighbourhood_group: chr  "Brooklyn" "Manhattan" "Manhattan" "Brooklyn" ...
 $ neighbourhood      : chr  "Kensington" "Midtown" "Harlem" "Clinton Hill" ...
 $ latitude           : num  40.6 40.8 40.8 40.7 40.8 ...
 $ longitude          : num  -74 -74 -73.9 -74 -73.9 ...
 $ room_type          : chr  "Private room" "Entire home/apt" "Private room" "Entire home/apt" ...
 $ price              : int  149 225 150 89 80 200 60 79 79 150 ...
 $ minimum_nights     : int  1 1 3 1 10 3 45 2 2 1 ...
 $ availability_365   : int  365 355 365 194 0 129 0 220 0 188 ...
```

RStudio: Notebook Output

```
neighbourhood_group neighbourhood        latitude       longitude       room_type          price          minimum_nights    availability_365
Length:48895        Length:48895       Min.   :40.50   Min.   :-74.24   Length:48895       Min.   :    0.0   Min.   :  1.00   Min.   :  0.0
Class :character    Class :character   1st Qu.:40.69   1st Qu.:-73.98   Class :character   1st Qu.:   69.0   1st Qu.:  1.00   1st Qu.:  0.0
Mode  :character    Mode  :character   Median :40.72   Median :-73.96   Mode  :character   Median :  106.0   Median :  3.00   Median : 45.0
                                       Mean   :40.73   Mean   :-73.95                      Mean   :  152.7   Mean   :  7.03   Mean   :112.8
                                       3rd Qu.:40.76   3rd Qu.:-73.94                      3rd Qu.:  175.0   3rd Qu.:  5.00   3rd Qu.:227.0
                                       Max.   :40.91   Max.   :-73.71                      Max.   :10000.0   Max.   :1250.00   Max.   :365.0
```

```r
library(glue)

glue("Price Minimum: {min(data$price)} | Price Maximum: {max(data$price)}")
```

```
Price Minimum: 0 | Price Maximum: 10000
```

```r
c(unique(data["room_type"]))
```

```
$room_type
[1] "Private room"    "Entire home/apt" "Shared room"
```

```r
library(glue)

glue("Price Minimum: {min(data$price)} | Price Maximum: {max(data$price)}")
```

```
Price Minimum: 0 | Price Maximum: 10000
```

```r
glue("Minimum Longitude : {min(data$longitude)} | Maximum Latitude: {max(data$longitude)}")
```

```
Minimum Longitude : -74.24442 | Maximum Latitude: -73.71299
```

```r
glue("Minimum Latitude : {min(data$latitude)} | Maximum Latitude: {max(data$latitude)}")
```

```
Minimum Latitude : 40.49979 | Maximum Latitude: 40.91306
```

Data cleaning

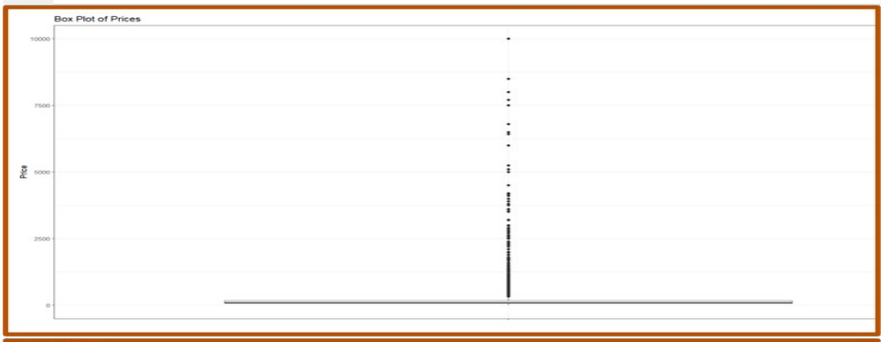
```r
#calculating upper and lower bound

price_IQR <- IQR(data$price, na.rm = TRUE)
price_Q1 <- quantile(data$price, 0.25, na.rm = TRUE)
price_Q3 <- quantile(data$price, 0.75, na.rm = TRUE)

lower_bound_price <- price_Q1 - 1.5 * price_IQR
upper_bound_price <- price_Q3 + 1.5*price_IQR
```

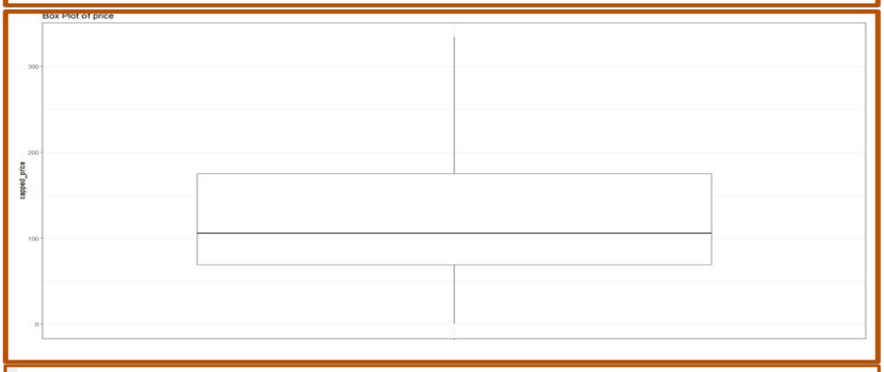
```r
#capping the price variable
data_capped <- data
data_capped$capped_price <- ifelse(data$price < lower_bound_price, lower_bound_price,
                                   ifelse(data$price > upper_bound_price, upper_bound_price,data$price))
```

```{r}
library(ggplot2)
ggplot(data, aes(x = "", y = price)) +
  geom_boxplot() +
  labs(title = "Box Plot of Prices", x = "", y = "Price") +
theme_bw()

```

```{r}
library(glue)

glue("Price Minimum: {min(data_capped$capped_price)} | Price Maximum: {max(data_capped$capped_price)}")

```

Price Minimum: 0 | Price Maximum: 334






```{r}
null_values_in_price <- sum(is.na(data$price))
print(null_values_in_price)
```

```
[1] 0
```

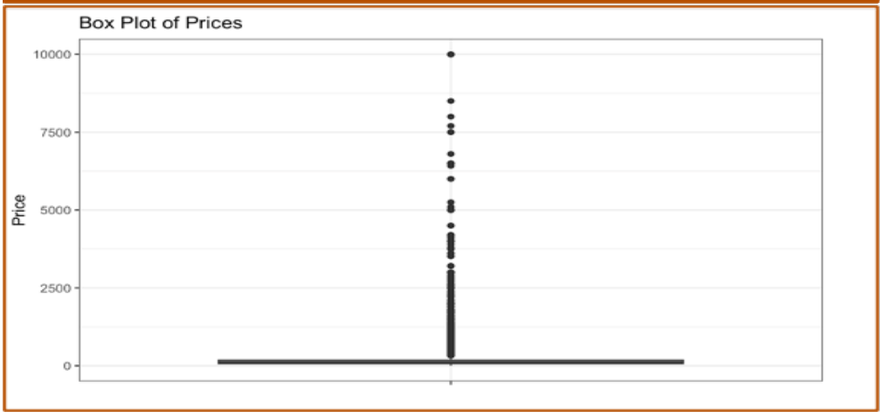
```{r}
null_values_in_latitude <- sum(is.na(data$latitude))
print(null_values_in_latitude)
```

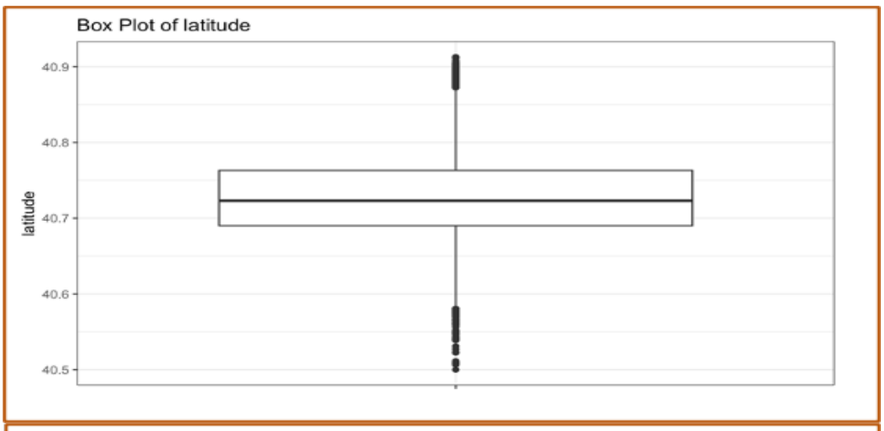
```
[1] 0
```

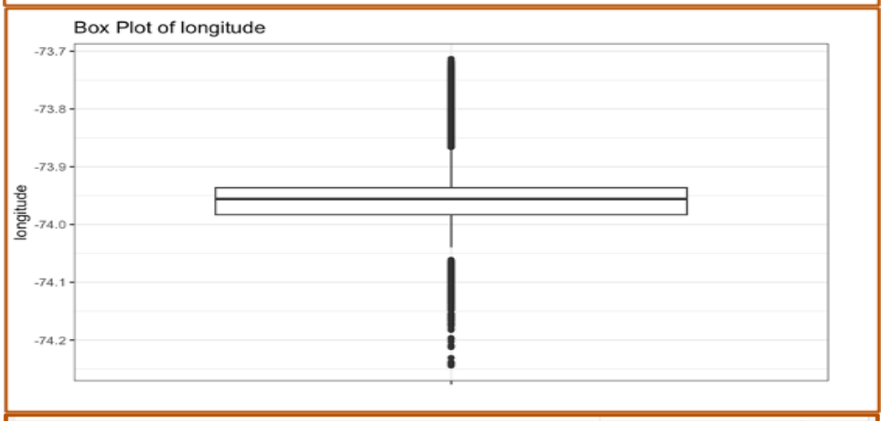
```{r}
null_values_in_longitude <- sum(is.na(data$longitude))
print(null_values_in_longitude)
```

```
[1] 0
```

```{r}
ggplot(data, aes(x = "", y = price)) +
  geom_boxplot() +
  labs(title = "Box Plot of Prices", x = "", y = "Price") +
  theme_bw()
```

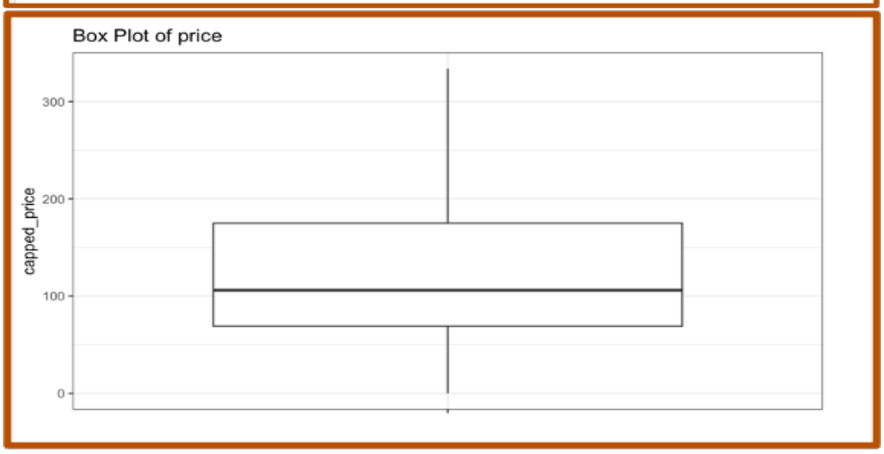


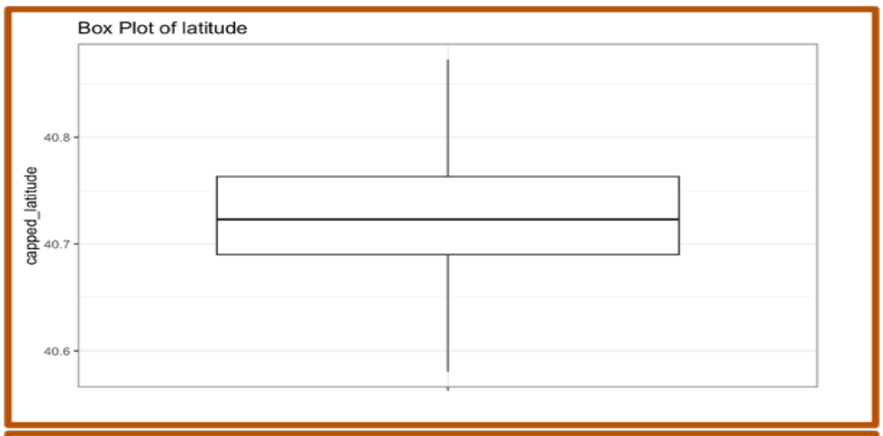


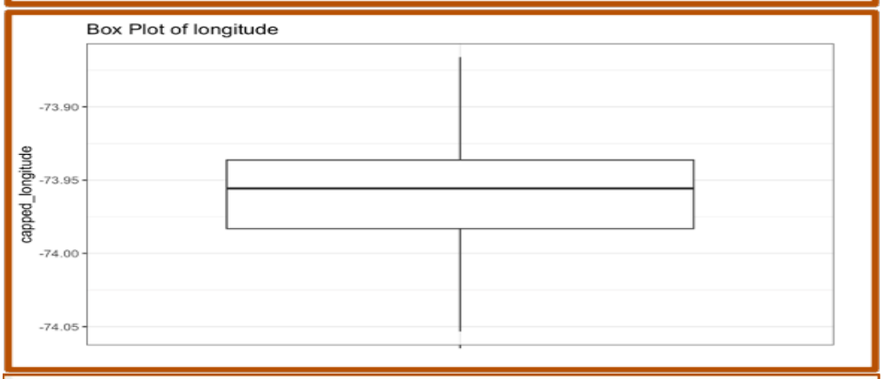



```r
install.packages("tidyverse")
library(tidyverse)
# Calculate the lower and upper bounds for detecting outliers using the 1.5 * IQR rule for 'price'
price_IQR <- IQR(data$price, na.rm = TRUE)
price_Q1 <- quantile(data$price, 0.25, na.rm = TRUE)
price_Q3 <- quantile(data$price, 0.75, na.rm = TRUE)

lower_bound_price <- price_Q1 - 1.5 * price_IQR
upper_bound_price <- price_Q3 + 1.5 * price_IQR

# Calculate the lower and upper bounds for detecting outliers using the 1.5 * IQR rule for 'latitude'
latitude_IQR <- IQR(data$latitude, na.rm = TRUE)
latitude_Q1 <- quantile(data$latitude, 0.25, na.rm = TRUE)
latitude_Q3 <- quantile(data$latitude, 0.75, na.rm = TRUE)

lower_bound_latitude <- latitude_Q1 - 1.5 * latitude_IQR
upper_bound_latitude <- latitude_Q3 + 1.5 * latitude_IQR

# Calculate the lower and upper bounds for detecting outliers using the 1.5 * IQR rule for 'longitude'
longitude_IQR <- IQR(data$longitude, na.rm = TRUE)
longitude_Q1 <- quantile(data$longitude, 0.25, na.rm = TRUE)
longitude_Q3 <- quantile(data$longitude, 0.75, na.rm = TRUE)

lower_bound_longitude <- longitude_Q1 - 1.5 * longitude_IQR
upper_bound_longitude <- longitude_Q3 + 1.5 * longitude_IQR

# Cap the outliers in the 'price', 'latitude', and 'longitude' columns based on the lower and upper bounds
data_capped <- data %>%
  mutate(
    capped_price = ifelse(price < lower_bound_price, lower_bound_price,
                          ifelse(price > upper_bound_price, upper_bound_price, price)),
    capped_latitude = ifelse(latitude < lower_bound_latitude, lower_bound_latitude,
                             ifelse(latitude > upper_bound_latitude, upper_bound_latitude, latitude)),
    capped_longitude = ifelse(longitude < lower_bound_longitude, lower_bound_longitude,
                              ifelse(longitude > upper_bound_longitude, upper_bound_longitude, longitude))
  )
```

```r
install.packages("ggplot2")
library(ggplot2)
ggplot(data_capped, aes(x = "", y = capped_price )) +
  geom_boxplot() +
  labs(title = "Box Plot of price", x = "", y = "capped_price") +
  theme_bw()
```








```{r}
city_center <- c(40.758896, -73.985130) # Times Square
```

```{r}
install.packages("geosphere")
library(geosphere)
data_capped$distance_to_center <- distVincentySphere(p1 = data_capped[, c("capped_latitude", "capped_longitude")], p2 = city_center)
```

```{r}
# Calculate the median distance
median_distance <- median(data_capped$distance_to_center)
data_capped$distance_category <- ifelse(data_capped$distance_to_center <= median_distance,
                                        "close_to_center", "far_from_center")
```

capped_price <dbl>	capped_latitude <dbl>	capped_longitude <dbl>	distance_to_center <dbl>	distance_category <chr>
149	40.64749	-73.97237	3705.8157	close_to_center
225	40.75362	-73.98377	221.7602	close_to_center
150	40.80902	-73.94190	5053.1746	far_from_center
89	40.68514	-73.95976	3621.4426	close_to_center
80	40.79851	-73.94399	4738.9179	far_from_center
200	40.74767	-73.97500	1179.2243	close_to_center
60	40.68688	-73.95596	3929.9674	close_to_center
79	40.76489	-73.98493	185.4278	close_to_center
79	40.80178	-73.96723	2388.9329	close_to_center
150	40.71344	-73.99037	1512.7861	close_to_center

Statistical Analysis:

```{r}
# Bartlett's test for homogeneity of variance
bartlett.test(capped_price ~ room_type, data = data_capped)
```

```
        Bartlett test of homogeneity of variances

data:  capped_price by room_type
Bartlett's K-squared = 4464.1, df = 2, p-value < 2.2e-16
```

```
#Using Kruskal Wallis Test to test the null hypothesis

```{r}
# Convert room_type to a factor variable
data_capped$room_type <- as.factor(data_capped$room_type)

# Perform Kruskal-Wallis test
kruskal_test= kruskal.test(capped_price ~ room_type, data = data_capped)
kruskal_test
```
```

```
Kruskal-Wallis rank sum test

data:  capped_price by room_type
Kruskal-Wallis chi-squared =
22434, df = 2, p-value < 2.2e-16
```

```
```{r}
# Install the dunn.test package if you haven't already
if (!requireNamespace("dunn.test", quietly = TRUE)) {
  install.packages("dunn.test")
}
# Load the package
library(dunn.test)


dunn_result <- dunn.test(data_capped[["capped_price"]], data_capped[["room_type"]], method = "bonferroni")
  print(dunn_result)
```
```

```
  Kruskal-Wallis rank sum test

data: x and group
Kruskal-Wallis chi-squared = 22434.2294, df = 2, p-value = 0


                           Comparison of x by group
                                 (Bonferroni)
Col Mean-|
Row Mean |   Entire h   Private
---------+----------------------
Private  |   145.1443
         |     0.0000*
         |
Shared r |   57.33895   12.95393
         |     0.0000*    0.0000*

alpha = 0.05
Reject Ho if p <= alpha/2
$chi2
[1] 22434.23

$Z
[1] 145.14439  57.33895  12.95394

$P
[1] 0.000000e+00 0.000000e+00 1.116001e-38

$P.adjusted
[1] 0.000000e+00 0.000000e+00 3.348003e-38

$comparisons
[1] "Entire home/apt - Private room" "Entire home/apt - Shared room"  "Private room - Shared room"
```

```
```{r}


# Calculate mean and median prices for each room type
library(dplyr)
summary_stats <- data_capped %>%
  group_by(room_type) %>%
  summarize(mean_capped_price = mean(capped_price), median_capped_price = median(capped_price))

# Print summary statistics
print(summary_stats)
```
```

A tibble: 3 × 3

room_type <chr>	mean_capped_price <dbl>	median_capped_price <dbl>
Entire home/apt	180.20819	160
Private room	82.78738	70
Shared room	64.50345	45

3 rows

```
```{r}
library(ggplot2)

ggplot(data_capped, aes(x = distance_to_center, y = capped_price, color = distance_category)) +
  geom_point(alpha = 0.5) +
  labs(title = "Scatter Plot of Price vs. Distance to Center",
       x = "Distance to Center",
       y = "capped_price") +
  theme_minimal()

```
```

```{r}
# Box plots
library(ggplot2)
ggplot(data_capped, aes(x = distance_category, y = capped_price)) +
  geom_boxplot() +
  labs(title = "Box Plots of Price by Distance Category",
       x = "Distance Category",
       y = "Price") +
  theme_minimal()

# Density plots
ggplot(data_capped, aes(x = capped_price, fill = distance_category)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plots of Price by Distance Category",
       x = "capped_price",
       y = "Density") +
  theme_minimal()
```

```{r}
median_distance <- median(data_capped$distance_to_center)
data_capped$distance_category <- ifelse(data_capped$distance_to_center <= median_distance,
                                        "close_to_center", "far_from_center")
```

```{r}
mann_whitney_test <- wilcox.test(capped_price ~ distance_category, data = data_capped)
print(mann_whitney_test)
```

```
        Wilcoxon rank sum test with continuity correction

data:  capped_price by distance_category
W = 456671250, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

```{r}
# Assuming you have a dataframe named "data_capped" with variables "capped_price" and "distance_category"

# Set the number of bootstrap iterations
n_bootstrap <- 1000

# Create an empty vector to store the bootstrap test statistics
bootstrap_stats <- numeric(n_bootstrap)

# Perform bootstrapping
for (i in 1:n_bootstrap) {
  # Create a bootstrap sample by resampling from the original data with replacement
  bootstrap_sample <- data_capped[sample(nrow(data_capped), replace = TRUE), ]

  # Calculate the Mann-Whitney U test statistic for the bootstrap sample
  bootstrap_stat <- wilcox.test(bootstrap_sample$capped_price ~ bootstrap_sample$distance_category)$statistic

  # Store the bootstrap test statistic
  bootstrap_stats[i] <- bootstrap_stat
}

# Calculate the original Mann-Whitney U test statistic
original_stat <- mann_whitney_test$statistic

# Calculate the p-value by comparing the original statistic to the bootstrap distribution
p_value <- sum(bootstrap_stats >= original_stat) / n_bootstrap

# Display the p-value
cat("p-value:", p_value, "\n")
```

```
# Define the observed test statistic (e.g., mean, median, etc.)
observed_stat <- median(close_to_center) - median(far_from_center)

# Set the number of permutations
n_permutations <- 1000

# Create an empty vector to store the permutation test statistics
perm_stats <- numeric(n_permutations)

# Perform the permutation test
for (i in 1:n_permutations) {
  # Combine the data and shuffle the distance category labels
  combined_data <- c(close_to_center, far_from_center)
  shuffled_labels <- sample(c(rep("close_to_center", length(close_to_center)),
                              rep("far_from_center", length(far_from_center))))

  # Calculate the test statistic for the permuted data
  perm_stat <- median(combined_data[shuffled_labels == "close_to_center"]) -
    median(combined_data[shuffled_labels == "far_from_center"])

  # Store the permutation test statistic
  perm_stats[i] <- perm_stat
}

# Calculate the p-value by comparing the observed statistic to the null distribution
p_value <- sum(abs(perm_stats) >= abs(observed_stat)) / n_permutations

# Display the p-value
cat("p-value:", p_value, "\n")
```

```r
# install.packages("dplyr")

# Load the dplyr library
library(dplyr)

# Calculate summary statistics for price variation by distance category
price_summary <- data_capped %>%
  group_by(distance_category) %>%
  summarize(
    min_price = min(capped_price),
    q1_price = quantile(capped_price, 0.25),
    median_price = median(capped_price),
    mean_price = mean(capped_price),
    q3_price = quantile(capped_price, 0.75),
    max_price = max(capped_price)
  )

# Print the summary statistics
print(price_summary)
```