Singapore Airbnb Analysis Capstone Project

KN

Contents

Importing Libraries

```
if(!require(readr)) install.packages("readr",
                                     repos = "http://cran.us.r-project.org")
if(!require(tidyverse)) install.packages("tidyverse",
                                         repos = "http://cran.us.r-project.org")
if(!require(ggplot2)) install.packages("ggplot2",
                                       repos = "http://cran.us.r-project.org")
if(!require(DataExplorer)) install.packages("DataExplorer",
                                            repos = "http://cran.us.r-project.org")
if(!require(corrplot)) install.packages("corrplot",
                                        repos = "http://cran.us.r-project.org")
if(!require(leaps)) install.packages("leaps",
                                     repos = "http://cran.us.r-project.org")
if(!require(glmnet)) install.packages("glmnet",
                                      repos = "http://cran.us.r-project.org")
if(!require(randomForest)) install.packages("randomForest",
                                            repos = "http://cran.us.r-project.org")
library(readr)
library(tidyverse)
library(DataExplorer)
library(ggplot2)
library(corrplot)
library(leaps)
library(glmnet)
library(randomForest)
```

Step 1: Data Methodology & Analysis

Overview: Step 1 - Import and split the raw data into 2 halves: One for training and another for testing.

Step 2 - Exploratory Data Analysis: In this step, a feeling can be established for the available cleaned dataset. A visualization of the data will help us build a better regression model.

Step 3 - Building a Regression Model: In order to build an optimal model, a suitable algorithm and certain features are necessary. Step 4 - Validate Prediction Quality: Different methods can be used to determine how good the model predicts a different set of listings.

Data Pre-processing

Importing Dataset

This project focuses on the gleaning patterns and other relevant information about Airbnb listings in Singapore To be more specifically, the goals of this projects are to answer questions such as: 1. how are rental properties distributed across the neighborhoods of Singapore 2. how do prices vary with respect neighborhoods, rental property types and rental amenities In this section we will take the first look at the loaded data frames. The necessary cleaning and transformations in dataset will be performed so that the data becomes more efficient.

We are using the dataset of February 2020, right before COVID-19 was declared a pandemic. The data is taken from http://insideairbnb.com/get-the-data.html. This site offers a dataset for every country.

```
## # A tibble: 6 x 16
##      id name  host_id host_name neighbourhood_g~ neighbourhood latitude
##   <dbl> <chr>   <dbl> <chr>     <chr>            <chr>            <dbl>
## 1 49091 COZI~  266763 Francesca North Region     Woodlands         1.44
```

```
## 2 50646 Plea~  227796 Sujatha   Central Region   Bukit Timah       1.33
## 3 56334 COZI~  266763 Francesca North Region     Woodlands         1.44
## 4 71609 Ensu~  367042 Belinda   East Region      Tampines          1.35
## 5 71896 B&B ~  367042 Belinda   East Region      Tampines          1.35
## 6 71903 Room~  367042 Belinda   East Region      Tampines          1.35
## # ... with 9 more variables: longitude <dbl>, room_type <chr>, price <dbl>,
## #   minimum_nights <dbl>, number_of_reviews <dbl>, last_review <date>,
## #   reviews_per_month <dbl>, calculated_host_listings_count <dbl>,
## #   availability_365 <dbl>
```

Quick glance at the data we have imported: 1. Over 8000 observations with 16 features 2. Over 2500 unique hosts offering rentals 3. The most expensive Airbnb charging $10,000 per night (Singapore dollars)

Summary

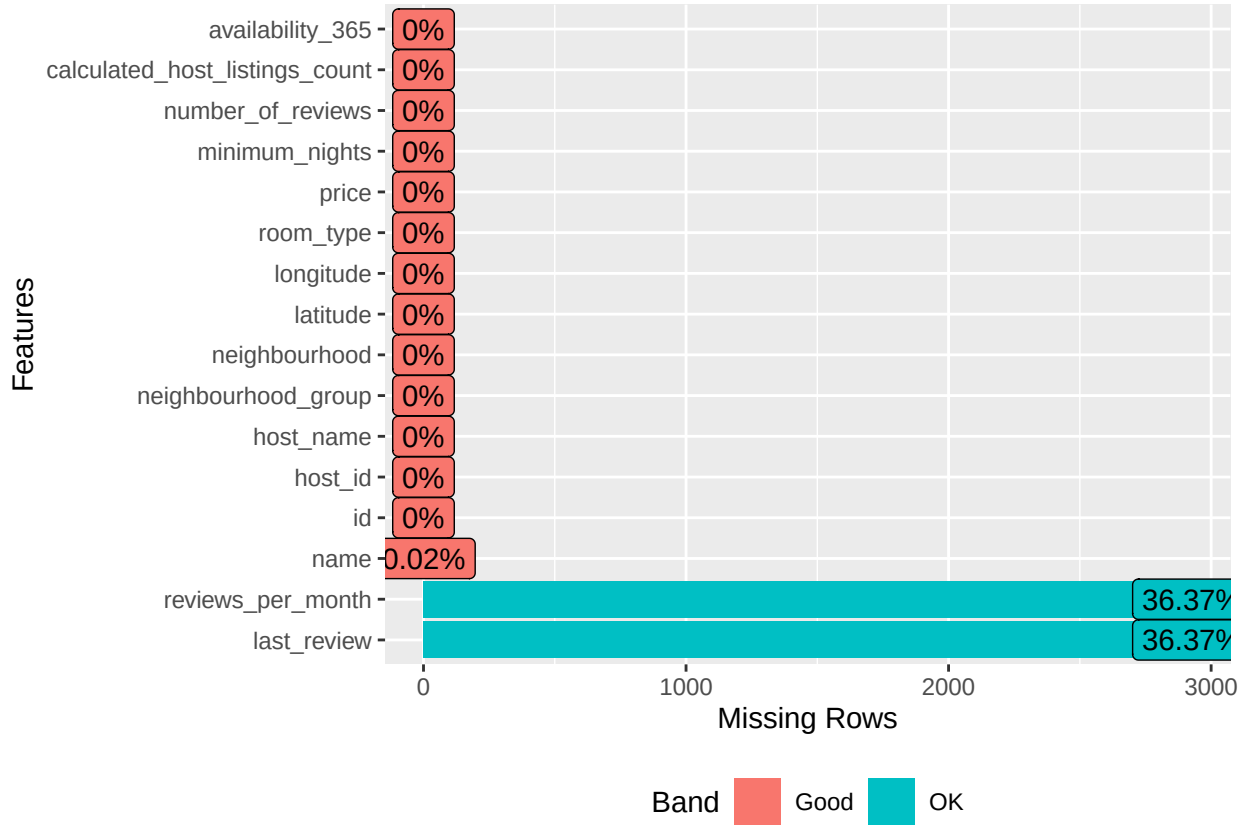
We check for any missing values and also create a plot to check how many columns have null values.

```
summary(airbnb)
```

```
##        id                 name               host_id            host_name
##  Min.   :   49091   Length:8047        Min.   :    23666   Length:8047
##  1st Qu.:16864566   Class :character   1st Qu.: 23722617   Class :character
##  Median :28116308   Mode  :character   Median : 66406177   Mode  :character
##  Mean   :26228320                      Mean   :102543782
##  3rd Qu.:36729204                      3rd Qu.:168257708
##  Max.   :42578709                      Max.   :338437839
##
##  neighbourhood_group neighbourhood         latitude       longitude
##  Length:8047         Length:8047        Min.   :1.244   Min.   :103.6
##  Class :character    Class :character   1st Qu.:1.296   1st Qu.:103.8
##  Mode  :character    Mode  :character   Median :1.311   Median :103.9
##                                         Mean   :1.313   Mean   :103.8
##                                         3rd Qu.:1.321   3rd Qu.:103.9
##                                         Max.   :1.455   Max.   :104.0
##
##   room_type             price         minimum_nights    number_of_reviews
##  Length:8047        Min.   :    0.0   Min.   :   1.00   Min.   :  0.00
##  Class :character   1st Qu.:   66.0   1st Qu.:   1.00   1st Qu.:  0.00
##  Mode  :character   Median :  126.0   Median :   3.00   Median :  1.00
##                     Mean   :  170.2   Mean   :  18.59   Mean   : 13.76
##                     3rd Qu.:  199.0   3rd Qu.:  14.00   3rd Qu.: 10.00
##                     Max.   :10000.0   Max.   :1000.00   Max.   :366.00
##
##   last_review         reviews_per_month calculated_host_listings_count
##  Min.   :2013-10-21   Min.   : 0.0100   Min.   :  1.00
##  1st Qu.:2019-03-15   1st Qu.: 0.1500   1st Qu.:  2.00
##  Median :2019-12-06   Median : 0.4600   Median : 11.00
##  Mean   :2019-06-01   Mean   : 0.9732   Mean   : 46.05
##  3rd Qu.:2020-01-30   3rd Qu.: 1.2000   3rd Qu.: 53.00
##  Max.   :2020-02-27   Max.   :24.4900   Max.   :340.00
##  NA's   :2927         NA's   :2927
##  availability_365
##  Min.   :  0.0
##  1st Qu.: 75.0
##  Median :297.0
##  Mean   :220.9
##  3rd Qu.:363.0
##  Max.   :365.0
##
```

Missing Data plot

```
plot_missing(airbnb)
```



There are two columns with missing values. We accordingly modify our data.

Data Modification - NULL values

Since reviews_per_month is a significant feature and used in our models, we modify the data by replacing all null values.

```
airbnb$reviews_per_month <- replace_na(airbnb$reviews_per_month, 0)
```

Split Data: Train and Test Sets

Dividing the training data into 80% of our actual data and 20% of test data for modelling, we will use the test dataset in the future for prediction purposes. Objects with price equal to 0 will be omitted since price cannot be 0 (faulty records). In order to remove the outliers, we are filtering the Airbnb data by removing the extreme values of price from both sides (10% from both the end). This will improve our models significantly.

```
airbnb_data_model <- airbnb %>% filter(price < quantile(airbnb$price, 0.9) &
  price > quantile(airbnb$price, 0.1)) %>% drop_na()

set.seed(1200)

airbnb_data_model <- airbnb_data_model %>% mutate(id = row_number())

#Dividing the data into 80% training data and 20% testing data
train_set <- airbnb_data_model %>% sample_frac(.8) %>% filter(price > 0)
test_set  <- anti_join(airbnb_data_model, train_set, by = 'id') %>% filter(price > 0)
```

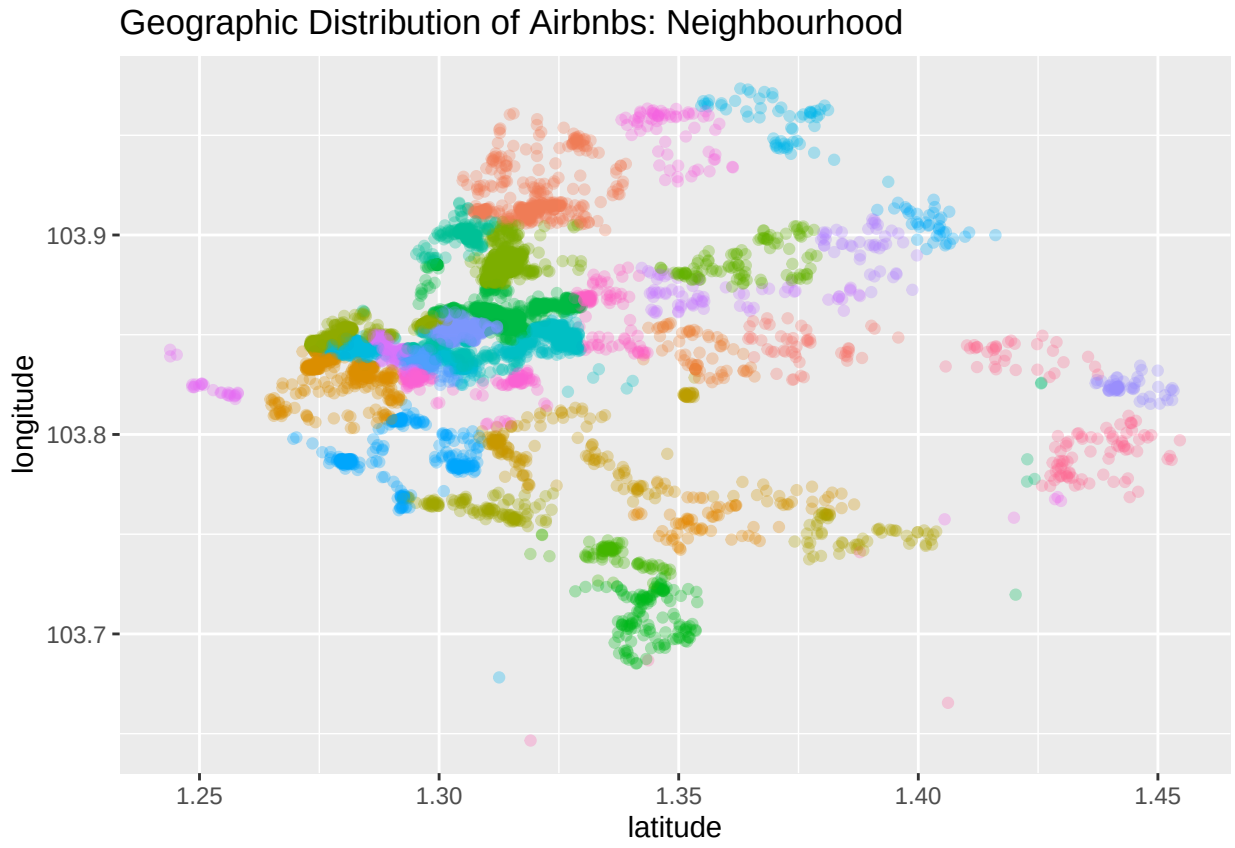
Procedure

Step 2: Exploratory Data Analysis

Geographic Distribution of Airbnbs in Singapore neighbourhoods

Here, we show a geographical plot of Airbnbs spread across multiple Singapore neighbourhoods. As observed, the dense portion is at central region of Singapore, also called the Beating Heart of Singapore. The frequency of Airbnb reduces on the outskirts of the country. Since, it is easier to stay closer to the attractions this country has to offer, people must prefer spending more to live in the heart of the country than travelling all the way. Central is a well-connected region with a thriving nightlife, plenty of job opportunities and some of Singapore's most exclusive neighbourhoods.

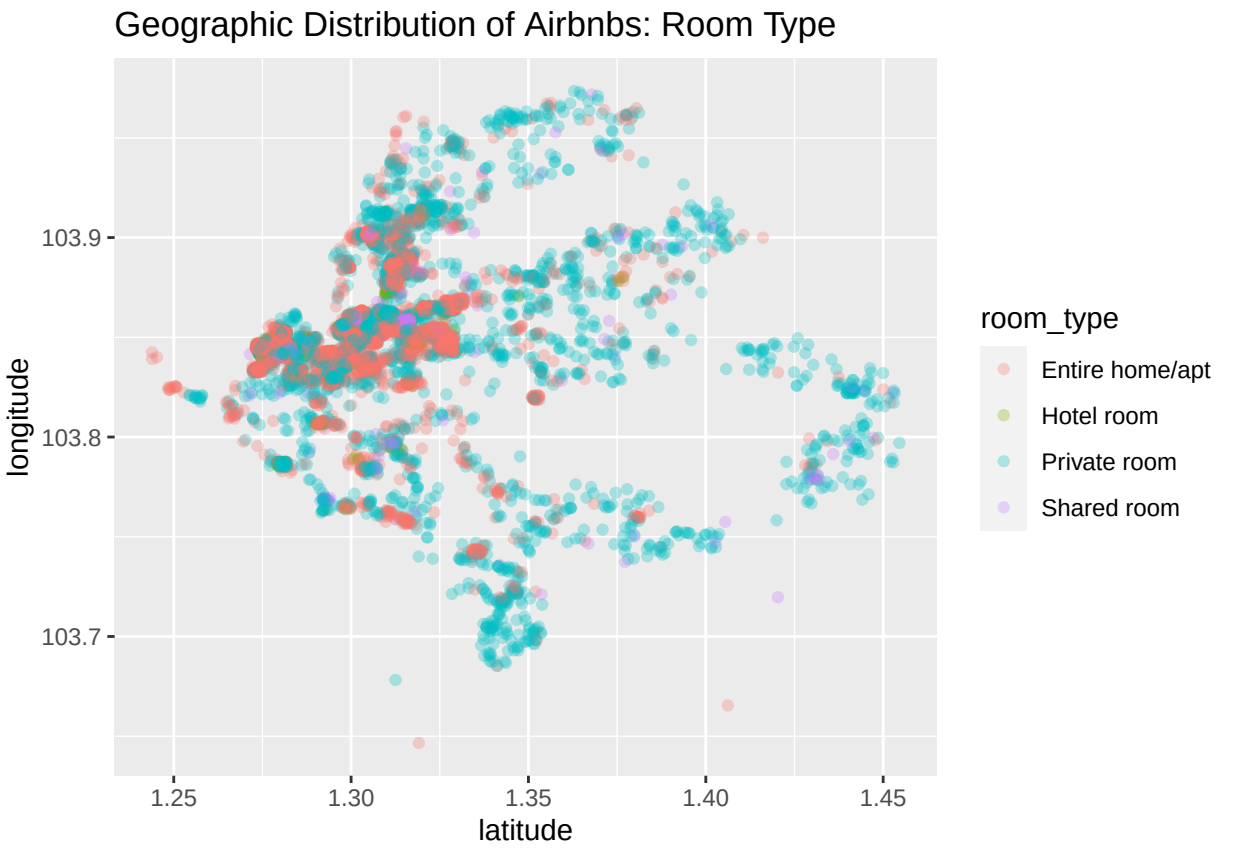
```
airbnb %>% ggplot(aes(x = latitude, y = longitude, color = neighbourhood)) +
  geom_point(alpha = 0.3) + theme(legend.position = "none") + labs(x = 'latitude',
  y = 'longitude', title = 'Geographic Distribution of Airbnbs: Neighbourhood')
```



Geographic Distribution of Airbnbs in Singapore neighbourhoods on basis of room types

Considering that most tourists are families and couples, renting private rooms is highly favored. Tourists refrain from renting an entire home/apartment because its expensive and tourists would rather spend money on exploring the country more, which indicates that they will spend very little time at the Airbnbs. College students or single immigrants prefer hiring shared rooms to cut extra and unnecessary costs. The layout of a private room is not much different from that of a hotel room, just cheaper than the rates of a hotel. Thus, there's a higher demand and availability of private rooms in the country.

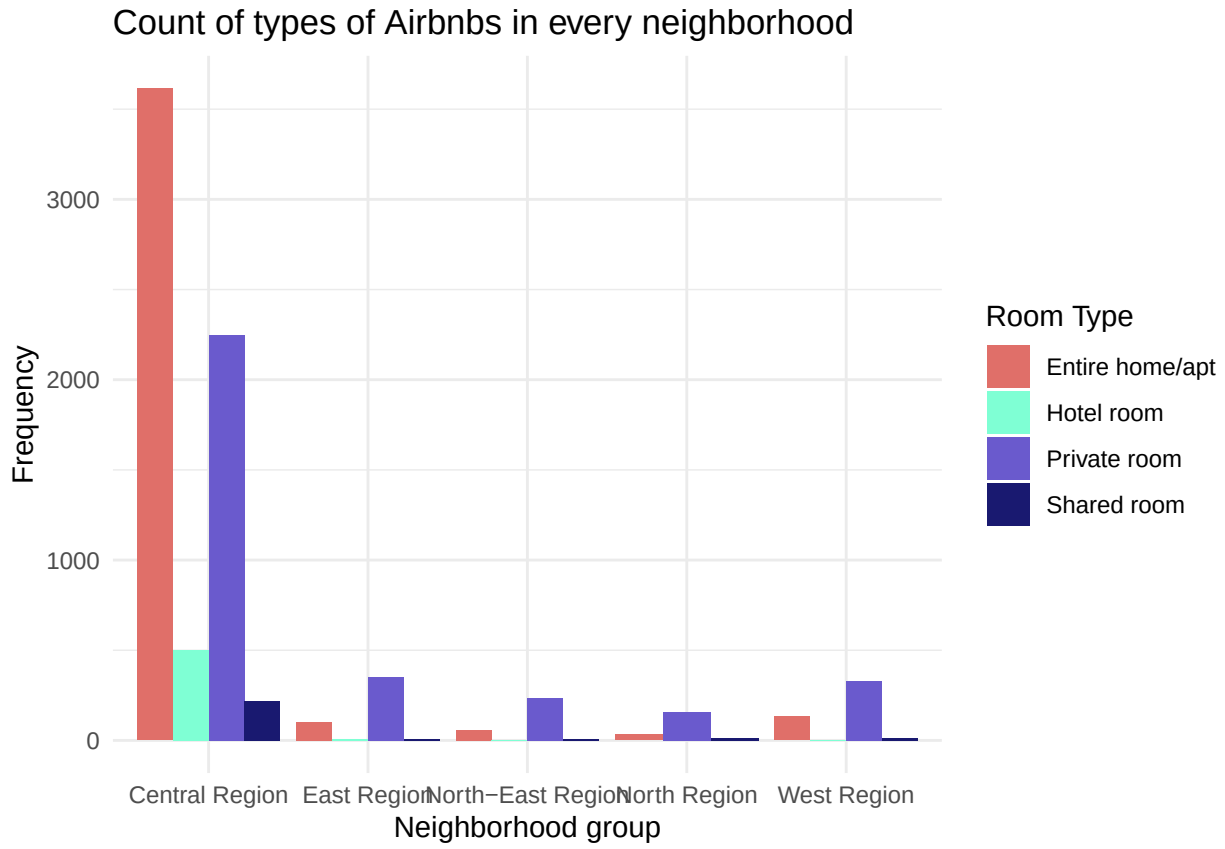
```
airbnb %>% ggplot(aes(x = latitude, y = longitude, color = room_type)) +
  geom_point(alpha = 0.3) + labs(x = 'latitude',
  y = 'longitude', title = 'Geographic Distribution of Airbnbs: Room Type')
```



Range of Airbnbs in every neighbourhood

Here we see a bar plot version of the above geographic distribution. Central Region being a densely populated borough constitutes majority of apartment style listings. There is a massive difference between the frequency of Airbnbs in Central Region and Eastern/Western Region. The number of Airbnbs in Central Region are ten to twenty times more than those in the remaining regions. This indicates that very few tourists/students visit the other regions and stick to the popular region only. For travellers and students, Central Region offers multiple restaurants and shops. Primarily, most of Singapore's major expressways wind their way to this region of Singapore. We have 4 types of Airbnbs, the new addition being hotel rooms, majorly used in Central Region as compared to the rest.

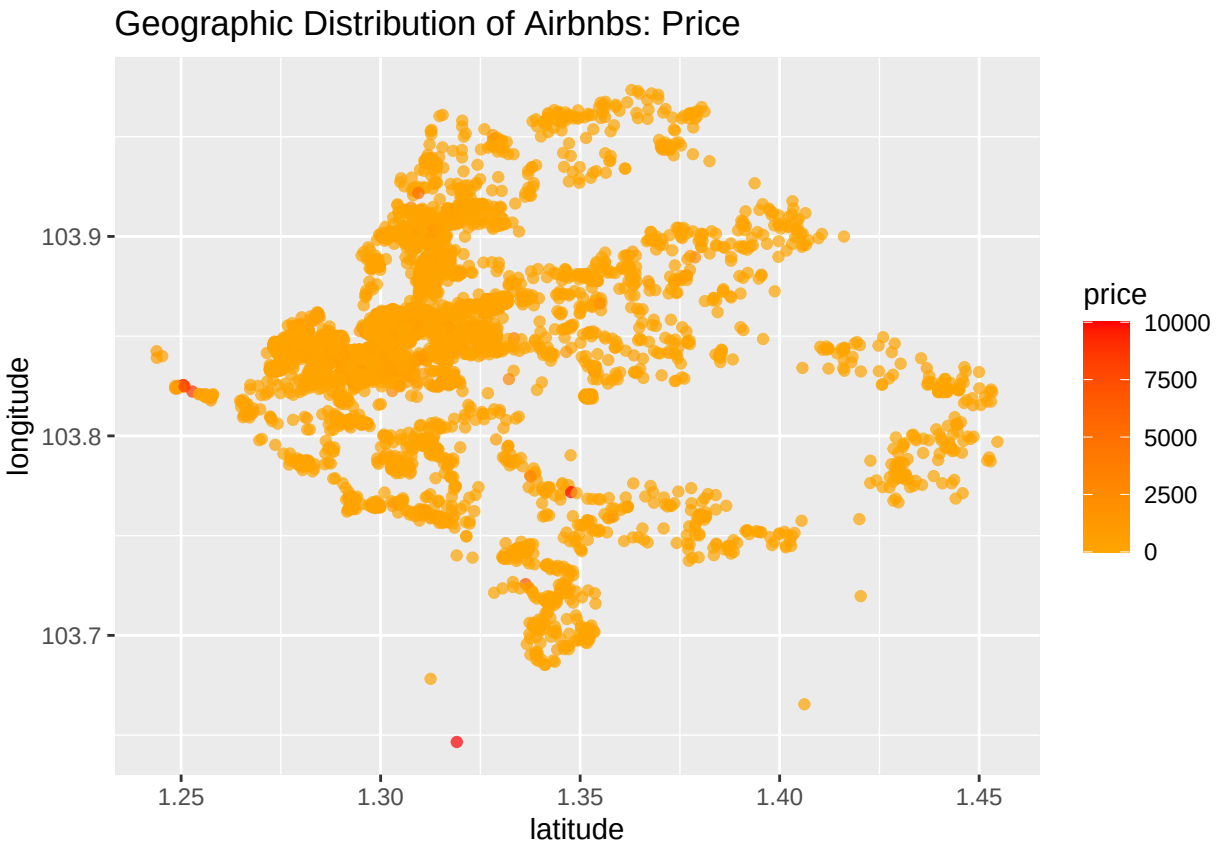
```
ggplot(data = airbnb, aes(x = neighbourhood_group)) +
  geom_bar(position = "dodge", aes(fill = room_type)) +
  scale_fill_manual("Room Type", values = c("#e06f69", "aquamarine", "slateblue", "midnightblue")) +
  labs(x = 'Neighborhood group', y = 'Frequency',
  title = 'Count of types of Airbnbs in every neighborhood') + theme_minimal()
```



Geographic Distribution of Airbnbs in Singapore neighbourhoods on basis of price

Most prices are below $5,000 and just a few above located at the heart of the country. Since people prefer Airbnbs over hotels, Airbnbs offer a decent price all over Singapore as seen in the plot below. With the lack of variation in prices, people have the luxury of renting an Airbnb in any neighbourhood. Tourists would rather walk till these top sights than spend more on travelling. Some might also wish to have Airbnbs with nice views from their balconies. Thus, this plot helped us understand that price is not a major factor considering all Airbnbs have similar prices.

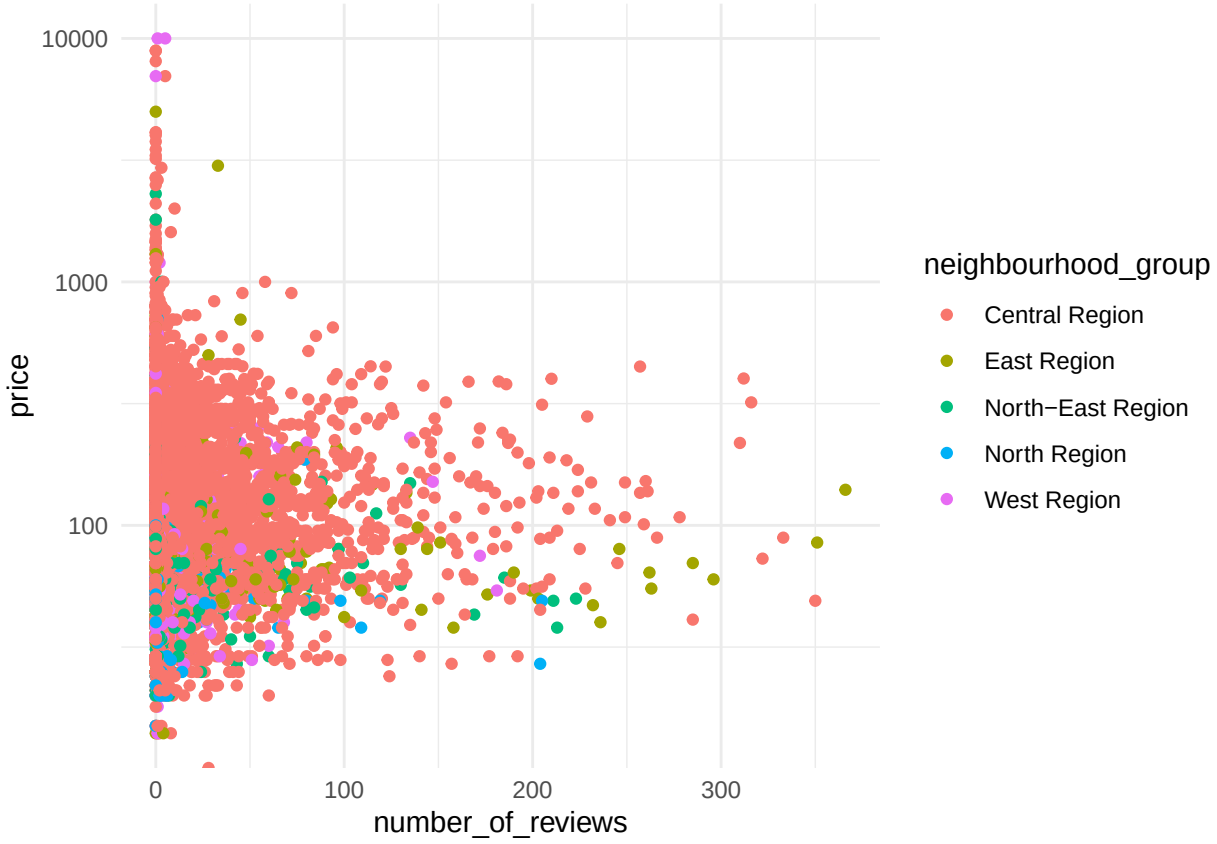
```
airbnb %>% ggplot(aes(x = latitude, y = longitude, color = price)) +
  geom_point(alpha = 0.7) + labs(x = 'latitude', y = 'longitude',
  title = 'Geographic Distribution of Airbnbs: Price') +
  scale_colour_gradient2(low = "yellow", mid = "orange", high = "red", midpoint = 50)
```



Number or reviews neighbourhood wise

Users have given around 300 reviews to rentals in the Central Region and East Region of Singapore The Airbnbs with the highest prices have the lowest reviews since it must be rented by wealthy people, most of whom did not give any feedback by writing a review on the website.

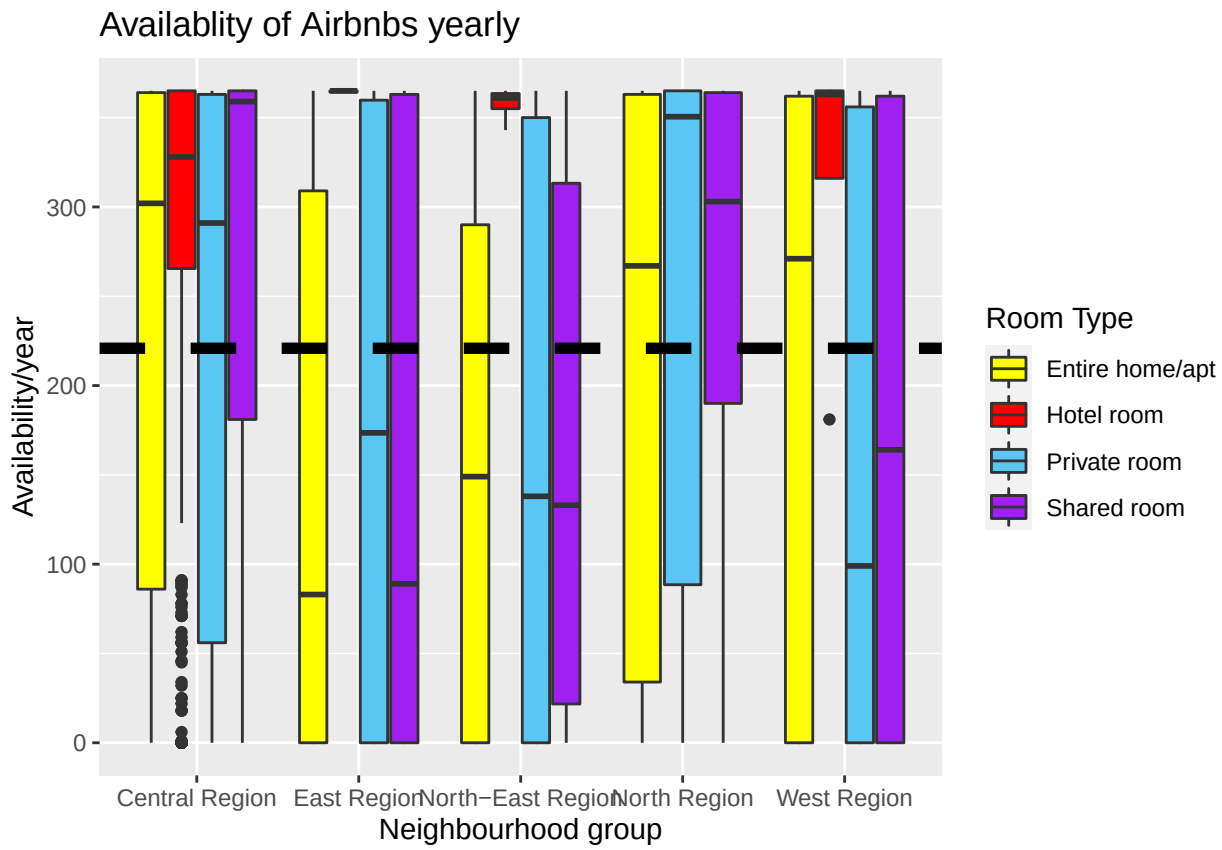
```
ggplot(data = airbnb) + scale_y_log10() + geom_point(aes(x = number_of_reviews,
          y = price, color = neighbourhood_group)) + theme_minimal()
```



Availability of Airbnbs yearly

Considering the number of tourists that visit the country every year, a particular Airbnb is available on a average of 200 days per year. Very few of these Airbnbs are available around 300 days a year which might be due to lack of good reviews. Thus, people might choose an Airbnb that's frequently used by many fellow tourists and owns a good rating.

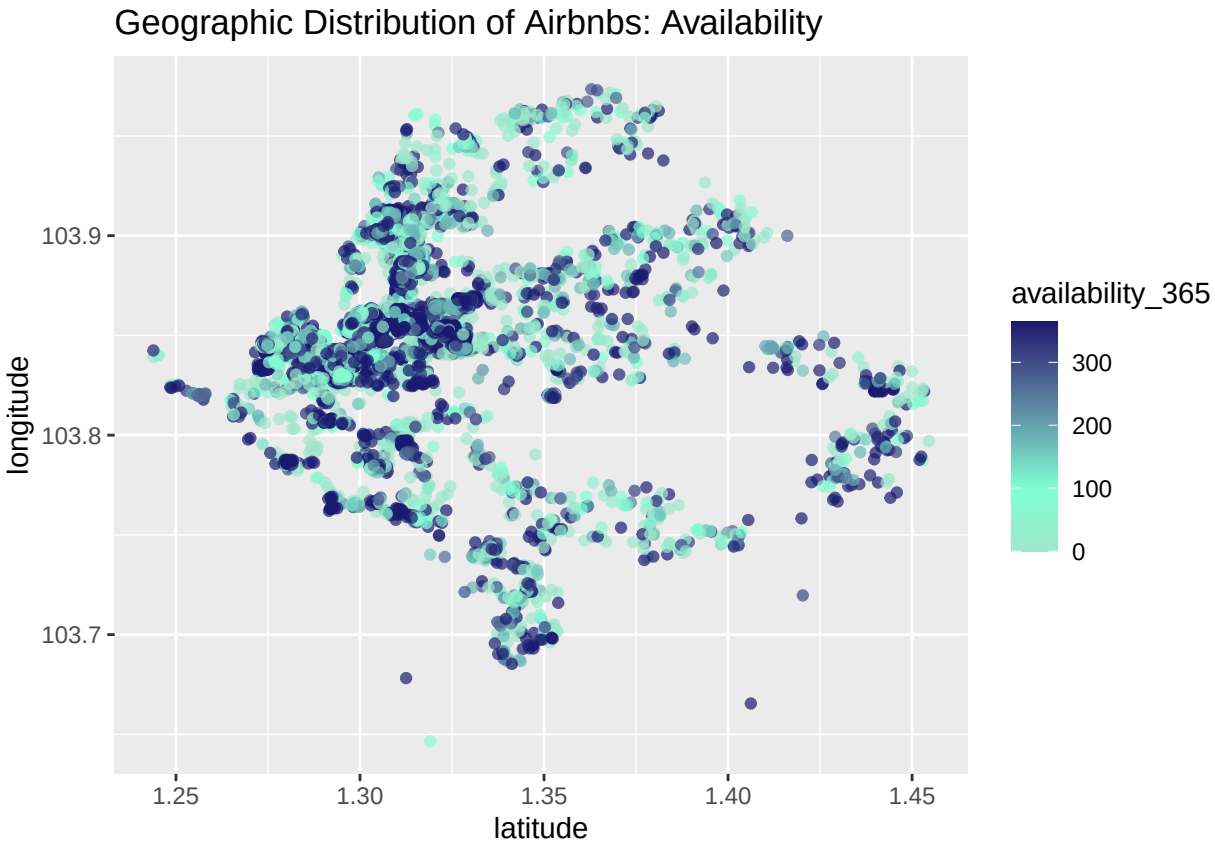
```
airbnb %>% ggplot(aes(y = availability_365, x = neighbourhood_group)) +
  geom_boxplot(aes(fill = room_type)) + geom_hline(yintercept =
  mean(airbnb$availability_365), color = "black", linetype = 2, size = 2) +
  scale_fill_manual("Room Type", values = c("yellow", "red", "#59c6f3", "purple")) +
  labs(x = 'Neighbourhood group', y = 'Availability/year',
       title = 'Availablity of Airbnbs yearly')
```



Geographic Distribution of Airbnbs in Singapore neighbourhoods on basis of availability

To get a better understanding, we'll see a geographic plot of Singapore which shows availability annually. The aquamarine colored points indicate that these Airbnbs have been used frequently on a yearly basis and are still likely to be used frequently. Some of these could be new rentals that haven't been used as much and thus, have a higher availability.

```
airbnb %>% ggplot(aes(x = latitude, y = longitude, color = availability_365)) +
  geom_point(alpha = 0.7) + labs(x = 'latitude', y = 'longitude',
  title = 'Geographic Distribution of Airbnbs: Availability') +
  scale_colour_gradient2(low = "grey", mid = "aquamarine", high = "midnightblue", midpoint = 100)
```



Correlation of 3 key characteristics

Since availability, number of reviews and calculated_host_listings_count are our key features in this dataset, we created a correlation table.

```
airbnb_correlation  <- airbnb  %>%
  select("availability_365","number_of_reviews", "calculated_host_listings_count")
cor(airbnb_correlation)
```

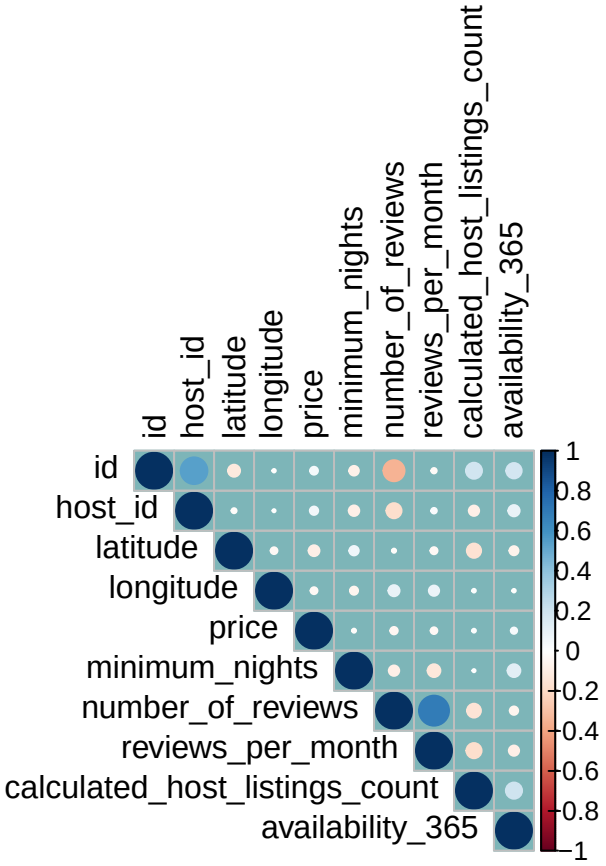
```
##                                availability_365 number_of_reviews
## availability_365                     1.00000000       -0.05382187
## number_of_reviews                   -0.05382187        1.00000000
## calculated_host_listings_count       0.20135295       -0.14745277
##                                calculated_host_listings_count
## availability_365                                    0.2013530
## number_of_reviews                                  -0.1474528
## calculated_host_listings_count                      1.0000000
```

None of these 3 key features have a strong correlation with each other.

Correlation of 8 features

Through this correlation matrix, we found that the calculated_host_listings_count is most correlated with availability_365. However, the correlations are not very strong.

```
airbnb_cor <- airbnb[, sapply(airbnb, is.numeric)]
airbnb_matrix <- cor(airbnb_cor, method = "pearson")
corrplot(airbnb_matrix, type="upper", bg = "#7db5b8", tl.col = "black")
```



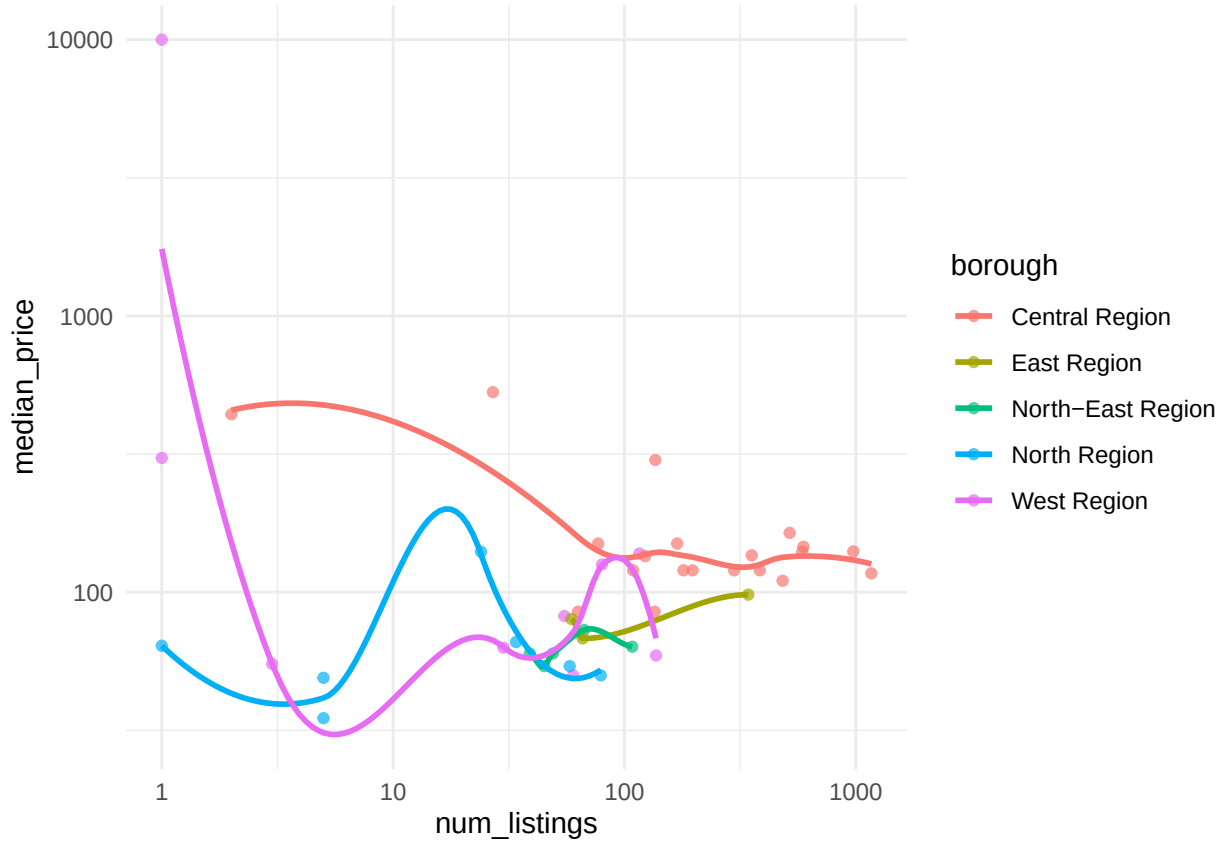
Median Price Plot

Here we explore a plot that shows the median price of all Airbnbs spreads across the neighbourhoods with Western Region soaring the highest ar 10000 Singapore dollars.

```
airbnb_median <- airbnb %>% group_by(neighbourhood) %>% summarize(num_listings = n(),
    median_price = median(price), long = median(longitude), lat = median(latitude),
    borough = unique(neighbourhood_group))

airbnb_median %>% ggplot(aes(x = num_listings, y = median_price, col = borough)) +
  geom_point(alpha = 0.7) + geom_smooth(se = FALSE) + scale_x_log10() +
  scale_y_log10() + theme_minimal()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Step 3: Model Building

I experimented with five models to find the most optimal model: first a Linear Regression Model without Reviews, then a Linear Regression Model with Reviews followed by a Log-Linear Regression Model, Subset Regression Model and finally a Random Forest Model. We iterated the variables in the models to increase R-squared, and reduce MAPE when running the models on the test data. We aimed to optimize the models with Subset Regression Model. Before running the data, we excluded the listings with 0 reviews as the price of these might not be tested by the market, i.e. irrelevant for the price estimation.

Linear Regression Model without reviews

Using this newly cleaned dataset, I instantiated the Liner Regression model, fit it on the training data, and then predicted the price for the test data. Using a linear regression model, we were able to get a R-squared less than 0.5. Since reviews is a key feature in this dataset, I explored two models, firstly, linear regression without reviews and secondly, linear regression with reviews.

```
lr_model <- lm(price ~ neighbourhood_group + latitude + longitude +
      room_type + minimum_nights + availability_365, data = train_set)
```

```
##
## Call:
## lm(formula = price ~ neighbourhood_group + latitude + longitude +
##     room_type + minimum_nights + availability_365, data = train_set)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -121.84  -32.31  -10.46   24.66  197.65
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       5.114e+03  3.972e+03   1.287 0.198027
## neighbourhood_groupEast Region    5.401e+00  4.678e+00   1.155 0.248378
## neighbourhood_groupNorth-East Region -7.492e+00  5.856e+00  -1.279 0.200840
## neighbourhood_groupNorth Region    4.385e+00  1.011e+01   0.434 0.664333
## neighbourhood_groupWest Region    -4.857e+00  6.742e+00  -0.720 0.471338
## latitude                          -1.950e+02  5.454e+01  -3.576 0.000354 ***
## longitude                         -4.526e+01  3.846e+01  -1.177 0.239366
## room_typeHotel room               -7.605e+00  3.859e+00  -1.971 0.048860 *
## room_typePrivate room             -7.962e+01  1.918e+00 -41.499  < 2e-16 ***
## room_typeShared room              -9.478e+01  8.547e+00 -11.089  < 2e-16 ***
## minimum_nights                    -8.444e-02  2.151e-02  -3.925 8.86e-05 ***
## availability_365                   4.191e-02  6.149e-03   6.815 1.12e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.68 on 3231 degrees of freedom
## Multiple R-squared:  0.4114, Adjusted R-squared:  0.4094
## F-statistic: 205.3 on 11 and 3231 DF,  p-value: < 2.2e-16
```

R squared and Adjusted R squared values for LR1

```
model_results <- tibble()
model_results <- tibble(method = "Linear Regression Model", MSE = lr_model_mse,
```

```
 R_squared = lr_model_rsquared, Adjusted_R_squared = lr_model_adj_rsquared)
model_results %>% knitr::kable()
```

method	MSE	R_squared	Adjusted_R_squared
Linear Regression Model	2468.079	0.4114382	0.4094344

Linear Regression Model with reviews

Here, I've added the three missing features from the previous model: 1. number_of_reviews 2. reviews_per_month 3. calculated_host_listings_count This model will undoubtedly be better than the first model since its got the three key features that are necessary to distinguish between Airbnbs across Singapore.

```
lr_model_2 <- lm(price ~ neighbourhood_group + latitude + longitude +
  reviews_per_month + room_type + minimum_nights  + number_of_reviews +
  calculated_host_listings_count + availability_365, data = train_set)
```

```
##
## Call:
## lm(formula = price ~ neighbourhood_group + latitude + longitude +
##     reviews_per_month + room_type + minimum_nights + number_of_reviews +
##     calculated_host_listings_count + availability_365, data = train_set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -123.772  -32.620   -9.664   24.734  196.350
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         2.933e+03  3.985e+03   0.736 0.461873
## neighbourhood_groupEast Region      4.393e+00  4.678e+00   0.939 0.347692
## neighbourhood_groupNorth-East Region -8.428e+00  5.836e+00  -1.444 0.148782
## neighbourhood_groupNorth Region      3.801e+00  1.007e+01   0.378 0.705777
## neighbourhood_groupWest Region      -3.468e+00  6.725e+00  -0.516 0.606132
## latitude                            -1.961e+02  5.443e+01  -3.603 0.000319 ***
## longitude                           -2.421e+01  3.859e+01  -0.627 0.530435
## reviews_per_month                    1.037e+00  8.398e-01   1.235 0.216911
## room_typeHotel room                 -9.239e+00  3.860e+00  -2.394 0.016740 *
## room_typePrivate room               -8.033e+01  1.991e+00 -40.343  < 2e-16 ***
## room_typeShared room                -9.678e+01  8.563e+00 -11.302  < 2e-16 ***
## minimum_nights                      -8.581e-02  2.154e-02  -3.984 6.94e-05 ***
## number_of_reviews                   -1.357e-01  2.823e-02  -4.807 1.60e-06 ***
## calculated_host_listings_count      -2.507e-02  1.303e-02  -1.923 0.054523 .
## availability_365                     4.512e-02  6.298e-03   7.164 9.68e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.48 on 3228 degrees of freedom
## Multiple R-squared:  0.4167, Adjusted R-squared:  0.4142
## F-statistic: 164.7 on 14 and 3228 DF,  p-value: < 2.2e-16
```

R squared and Adjusted R squared values for LR2

```
model_results <- bind_rows(model_results, tibble(method = "Linear Regression Model 2",
      MSE = lr_model_mse_2, R_squared = lr_model_rsquared_2,
      Adjusted_R_squared = lr_model_adj_rsquared_2))
model_results %>% knitr::kable()
```

method	MSE	R_squared	Adjusted_R_squared
Linear Regression Model	2468.079	0.4114382	0.4094344
Linear Regression Model 2	2448.112	0.4167419	0.4142123

Log Linear Regression Model

The log-linear regression model consistently gave an R-squared between 0.5 and 0.55. Percentage changes in price were driven by the following variables: room_type, neighborhood, reviews, log(calculated_host_listings_count) that is, we assumed that there is a marginally decreasing impact of calculated host listings count.

```
log_lr <- lm(formula = log(price) ~ room_type + neighbourhood + reviews_per_month +
  minimum_nights + number_of_reviews + number_of_reviews +
    log(calculated_host_listings_count), data = train_set)
```

```
##
## Call:
## lm(formula = log(price) ~ room_type + neighbourhood + reviews_per_month +
##     minimum_nights + number_of_reviews + number_of_reviews +
##     log(calculated_host_listings_count), data = train_set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.21979 -0.25903 -0.04239  0.23832  1.33963
##
## Coefficients:
##                                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           4.7852797  0.0848606  56.390  < 2e-16 ***
## room_typeHotel room                  -0.0521253  0.0282615  -1.844 0.065218 .
## room_typePrivate room                -0.6735312  0.0143384 -46.974  < 2e-16 ***
## room_typeShared room                 -0.8096116  0.0631972 -12.811  < 2e-16 ***
## neighbourhoodBedok                    0.3168050  0.0887991   3.568 0.000365 ***
## neighbourhoodBishan                   0.1456696  0.1133739   1.285 0.198934
## neighbourhoodBukit Batok              0.2248959  0.1171870   1.919 0.055058 .
## neighbourhoodBukit Merah              0.2433115  0.0882873   2.756 0.005886 **
## neighbourhoodBukit Panjang            0.2397106  0.1327199   1.806 0.070991 .
## neighbourhoodBukit Timah              0.2233785  0.1000438   2.233 0.025631 *
## neighbourhoodCentral Water Catchment -0.2044533  0.1969004  -1.038 0.299182
## neighbourhoodChoa Chu Kang            0.1519617  0.1144614   1.328 0.184397
## neighbourhoodClementi                 0.2718122  0.1070746   2.539 0.011179 *
## neighbourhoodDowntown Core            0.4791657  0.0888274   5.394 7.38e-08 ***
## neighbourhoodGeylang                  0.2059766  0.0863661   2.385 0.017141 *
## neighbourhoodHougang                  0.1114038  0.1023900   1.088 0.276662
## neighbourhoodJurong East              0.2657983  0.1051137   2.529 0.011497 *
## neighbourhoodJurong West              0.1151427  0.0962986   1.196 0.231909
## neighbourhoodKallang                  0.2769399  0.0861975   3.213 0.001327 **
## neighbourhoodMarine Parade            0.3071781  0.0925347   3.320 0.000912 ***
## neighbourhoodMuseum                   0.3831733  0.1048480   3.655 0.000262 ***
## neighbourhoodNewton                   0.3561017  0.0972429   3.662 0.000254 ***
## neighbourhoodNovena                   0.2987419  0.0882550   3.385 0.000720 ***
## neighbourhoodOrchard                  0.5257488  0.0999816   5.258 1.55e-07 ***
## neighbourhoodOutram                   0.2316862  0.0881789   2.627 0.008644 **
## neighbourhoodPasir Ris                0.2358330  0.1301312   1.812 0.070038 .
## neighbourhoodPunggol                  0.0918270  0.1224288   0.750 0.453283
## neighbourhoodQueenstown               0.3071868  0.0911765   3.369 0.000763 ***
## neighbourhoodRiver Valley             0.3027267  0.0885157   3.420 0.000634 ***
## neighbourhoodRochor                   0.3995027  0.0878211   4.549 5.59e-06 ***
## neighbourhoodSembawang                0.0452497  0.1173044   0.386 0.699711
## neighbourhoodSengkang                -0.0727094  0.1328618  -0.547 0.584241
```

```
## neighbourhoodSerangoon                0.0964571  0.1016542   0.949 0.342756
## neighbourhoodSingapore River          0.3174463  0.0966107   3.286 0.001028 **
## neighbourhoodSouthern Islands         1.2255290  0.2221187   5.517 3.71e-08 ***
## neighbourhoodSungei Kadut            -0.1417763  0.2658140  -0.533 0.593817
## neighbourhoodTampines                 0.2183801  0.1053756   2.072 0.038308 *
## neighbourhoodTanglin                  0.3229961  0.0940667   3.434 0.000603 ***
## neighbourhoodToa Payoh                0.1638450  0.0992605   1.651 0.098907 .
## neighbourhoodWestern Water Catchment -0.3075392  0.2668357  -1.153 0.249185
## neighbourhoodWoodlands                0.1643716  0.1111337   1.479 0.139227
## neighbourhoodYishun                   0.3230777  0.1513287   2.135 0.032842 *
## reviews_per_month                     0.0049909  0.0060766   0.821 0.411519
## minimum_nights                       -0.0004950  0.0001591  -3.112 0.001873 **
## number_of_reviews                    -0.0009075  0.0002056  -4.413 1.05e-05 ***
## log(calculated_host_listings_count)   0.0003870  0.0039606   0.098 0.922159
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3559 on 3197 degrees of freedom
## Multiple R-squared:  0.5145, Adjusted R-squared:  0.5077
## F-statistic: 75.29 on 45 and 3197 DF,  p-value: < 2.2e-16
```

R squared and Adjusted R squared values for Log LR

```
model_results <- bind_rows(model_results, tibble(method = "Log-Linear Regression Model",
  MSE = log_lr_mse, R_squared = log_lr_rsquared,
  Adjusted_R_squared = log_lr_adj_rsquared))
model_results %>% knitr::kable()
```

method	MSE	R_squared	Adjusted_R_squared
Linear Regression Model	2468.0792385	0.4114382	0.4094344
Linear Regression Model 2	2448.1116222	0.4167419	0.4142123
Log-Linear Regression Model	0.1266871	0.5145178	0.5076843

RMSE Prediction for Log LR model

```
predTest_log = predict(log_lr, newdata = test_set)
RMSE_1 = sqrt(mean((predTest_log - log(test_set$price))^2))
rmse_results = tibble()
rmse_results <- tibble(method = "Log Linear Regression Model", RMSE = RMSE_1)
```

Best Subset Regression Model

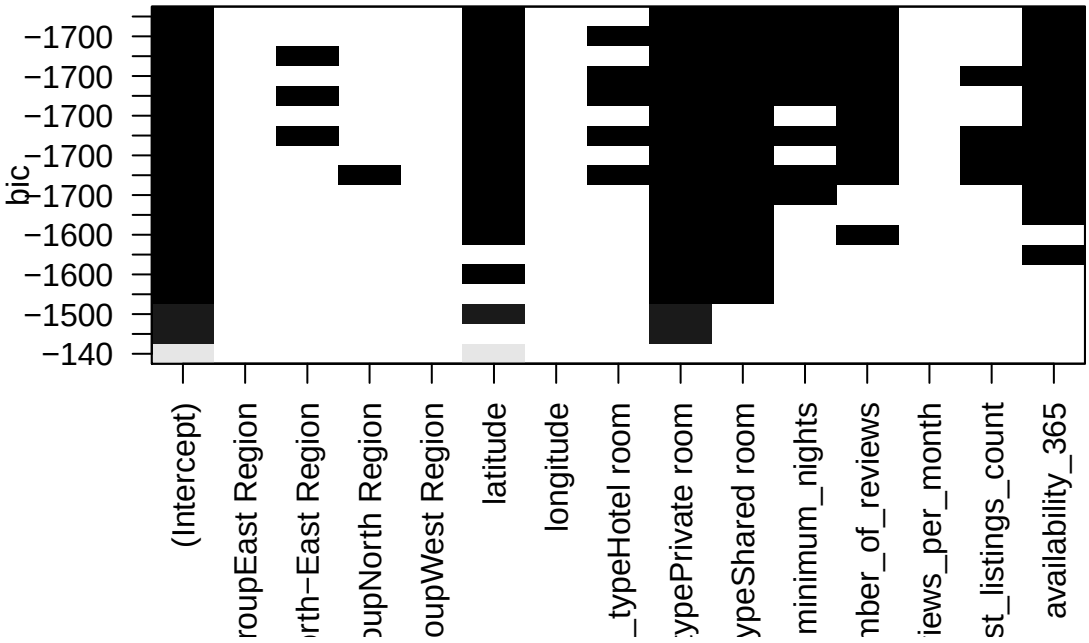
Subset regression offers a sequence of models which are the best subset for each size.

```
regression_model <- regsubsets(price ~ neighbourhood_group + latitude + longitude +
  room_type + minimum_nights  + number_of_reviews + reviews_per_month +
    calculated_host_listings_count + availability_365,
  data = train_set, nbest = 2, nvmax = 9)
```

```
## Subset selection object
## Call: regsubsets.formula(price ~ neighbourhood_group + latitude + longitude +
##     room_type + minimum_nights + number_of_reviews + reviews_per_month +
##     calculated_host_listings_count + availability_365, data = train_set,
##     nbest = 2, nvmax = 9)
## 14 Variables  (and intercept)
##                                       Forced in Forced out
## neighbourhood_groupEast Region            FALSE      FALSE
## neighbourhood_groupNorth-East Region      FALSE      FALSE
## neighbourhood_groupNorth Region           FALSE      FALSE
## neighbourhood_groupWest Region            FALSE      FALSE
## latitude                                  FALSE      FALSE
## longitude                                 FALSE      FALSE
## room_typeHotel room                       FALSE      FALSE
## room_typePrivate room                     FALSE      FALSE
## room_typeShared room                      FALSE      FALSE
## minimum_nights                            FALSE      FALSE
## number_of_reviews                         FALSE      FALSE
## reviews_per_month                         FALSE      FALSE
## calculated_host_listings_count            FALSE      FALSE
## availability_365                          FALSE      FALSE
## 2 subsets of each size up to 9
## Selection Algorithm: exhaustive
##          neighbourhood_groupEast Region neighbourhood_groupNorth-East Region
## 1  ( 1 ) " "                            " "
## 1  ( 2 ) " "                            " "
## 2  ( 1 ) " "                            " "
## 2  ( 2 ) " "                            " "
## 3  ( 1 ) " "                            " "
## 3  ( 2 ) " "                            " "
## 4  ( 1 ) " "                            " "
## 4  ( 2 ) " "                            " "
## 5  ( 1 ) " "                            " "
## 5  ( 2 ) " "                            " "
## 6  ( 1 ) " "                            " "
## 6  ( 2 ) " "                            " "
## 7  ( 1 ) " "                            " "
## 7  ( 2 ) " "                            "*"
## 8  ( 1 ) " "                            " "
## 8  ( 2 ) " "                            "*"
## 9  ( 1 ) " "                            "*"
## 9  ( 2 ) " "                            " "
##          neighbourhood_groupNorth Region neighbourhood_groupWest Region
## 1  ( 1 ) " "                             " "
## 1  ( 2 ) " "                             " "
## 2  ( 1 ) " "                             " "
```

```
## 2  ( 2 ) " "                            " "
## 3  ( 1 ) " "                            " "
## 3  ( 2 ) " "                            " "
## 4  ( 1 ) " "                            " "
## 4  ( 2 ) " "                            " "
## 5  ( 1 ) " "                            " "
## 5  ( 2 ) " "                            " "
## 6  ( 1 ) " "                            " "
## 6  ( 2 ) " "                            " "
## 7  ( 1 ) " "                            " "
## 7  ( 2 ) " "                            " "
## 8  ( 1 ) " "                            " "
## 8  ( 2 ) " "                            " "
## 9  ( 1 ) " "                            " "
## 9  ( 2 ) "*"                            " "
##          latitude longitude room_typeHotel room room_typePrivate room
## 1  ( 1 ) " "      " "       " "                   "*"
## 1  ( 2 ) "*"      " "       " "                   " "
## 2  ( 1 ) " "      " "       " "                   "*"
## 2  ( 2 ) "*"      " "       " "                   "*"
## 3  ( 1 ) " "      " "       " "                   "*"
## 3  ( 2 ) "*"      " "       " "                   "*"
## 4  ( 1 ) "*"      " "       " "                   "*"
## 4  ( 2 ) "*"      " "       " "                   "*"
## 5  ( 1 ) "*"      " "       " "                   "*"
## 5  ( 2 ) "*"      " "       " "                   "*"
## 6  ( 1 ) "*"      " "       " "                   "*"
## 6  ( 2 ) "*"      " "       " "                   "*"
## 7  ( 1 ) "*"      " "       "*"                   "*"
## 7  ( 2 ) "*"      " "       " "                   "*"
## 8  ( 1 ) "*"      " "       "*"                   "*"
## 8  ( 2 ) "*"      " "       "*"                   "*"
## 9  ( 1 ) "*"      " "       "*"                   "*"
## 9  ( 2 ) "*"      " "       "*"                   "*"
##          room_typeShared room minimum_nights number_of_reviews
## 1  ( 1 ) " "                    " "            " "
## 1  ( 2 ) " "                    " "            " "
## 2  ( 1 ) "*"                    " "            " "
## 2  ( 2 ) " "                    " "            " "
## 3  ( 1 ) "*"                    " "            " "
## 3  ( 2 ) "*"                    " "            " "
## 4  ( 1 ) "*"                    " "            " "
## 4  ( 2 ) "*"                    " "            "*"
## 5  ( 1 ) "*"                    " "            "*"
## 5  ( 2 ) "*"                    "*"            " "
## 6  ( 1 ) "*"                    "*"            "*"
## 6  ( 2 ) "*"                    " "            "*"
## 7  ( 1 ) "*"                    "*"            "*"
## 7  ( 2 ) "*"                    "*"            "*"
## 8  ( 1 ) "*"                    "*"            "*"
## 8  ( 2 ) "*"                    "*"            "*"
## 9  ( 1 ) "*"                    "*"            "*"
## 9  ( 2 ) "*"                    "*"            "*"
##          reviews_per_month calculated_host_listings_count availability_365
```

```
## 1  ( 1 ) " "                 " "                                  " "
## 1  ( 2 ) " "                 " "                                  " "
## 2  ( 1 ) " "                 " "                                  " "
## 2  ( 2 ) " "                 " "                                  " "
## 3  ( 1 ) " "                 " "                                  "*"
## 3  ( 2 ) " "                 " "                                  " "
## 4  ( 1 ) " "                 " "                                  "*"
## 4  ( 2 ) " "                 " "                                  " "
## 5  ( 1 ) " "                 " "                                  "*"
## 5  ( 2 ) " "                 " "                                  "*"
## 6  ( 1 ) " "                 " "                                  "*"
## 6  ( 2 ) " "                 "*"                                  "*"
## 7  ( 1 ) " "                 " "                                  "*"
## 7  ( 2 ) " "                 " "                                  "*"
## 8  ( 1 ) " "                 "*"                                  "*"
## 8  ( 2 ) " "                 " "                                  "*"
## 9  ( 1 ) " "                 "*"                                  "*"
## 9  ( 2 ) " "                 "*"                                  "*"
```



We made a model based on the plot above and added a new feature to our model - neighbourhood. This model could give us good values required for price predictions. The remaining reviews are same as the previous models since most of them are impactful considering the spread of Airbnbs across Singapore, most of them being in the Central Region.

```
subset_model <- lm(price ~ neighbourhood + latitude + longitude +
      room_type + minimum_nights  + number_of_reviews + reviews_per_month +
      calculated_host_listings_count + availability_365, data = train_set)
```

```
##
## Call:
## lm(formula = price ~ neighbourhood + latitude + longitude + room_type +
##     minimum_nights + number_of_reviews + reviews_per_month +
##     calculated_host_listings_count + availability_365, data = train_set)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -128.711  -32.322   -9.512   23.168  194.619
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        4.079e+04  1.285e+04   3.175 0.001515 **
## neighbourhoodBedok                 4.297e+01  1.740e+01   2.470 0.013565 *
## neighbourhoodBishan                8.015e+00  1.587e+01   0.505 0.613676
## neighbourhoodBukit Batok          -2.386e+01  1.974e+01  -1.208 0.226955
## neighbourhoodBukit Merah          -9.225e+00  1.852e+01  -0.498 0.618519
## neighbourhoodBukit Panjang        -2.291e+00  2.038e+01  -0.112 0.910522
## neighbourhoodBukit Timah          -1.517e+01  1.723e+01  -0.881 0.378547
## neighbourhoodCentral Water Catchment -3.599e+01  2.725e+01  -1.321 0.186685
## neighbourhoodChoa Chu Kang        -9.623e+00  1.932e+01  -0.498 0.618504
## neighbourhoodClementi             -1.300e+01  2.054e+01  -0.633 0.526860
## neighbourhoodDowntown Core         3.352e+01  1.839e+01   1.823 0.068434 .
## neighbourhoodGeylang               2.038e+01  1.562e+01   1.305 0.192110
## neighbourhoodHougang               2.773e+01  1.512e+01   1.834 0.066696 .
## neighbourhoodJurong East          -2.824e+01  2.034e+01  -1.388 0.165121
## neighbourhoodJurong West          -4.887e+01  2.172e+01  -2.250 0.024517 *
## neighbourhoodKallang               1.845e+01  1.497e+01   1.233 0.217851
## neighbourhoodMarine Parade         3.487e+01  1.754e+01   1.988 0.046937 *
## neighbourhoodMuseum                1.699e+01  1.847e+01   0.920 0.357797
## neighbourhoodNewton                1.155e+01  1.670e+01   0.692 0.489297
## neighbourhoodNovena                1.635e+01  1.427e+01   1.145 0.252127
## neighbourhoodOrchard               2.677e+01  1.740e+01   1.538 0.124141
## neighbourhoodOutram               -5.103e+00  1.819e+01  -0.281 0.779081
## neighbourhoodPasir Ris             6.484e+01  2.299e+01   2.820 0.004827 **
## neighbourhoodPunggol               3.983e+01  1.886e+01   2.111 0.034817 *
## neighbourhoodQueenstown           -1.302e+01  1.906e+01  -0.683 0.494443
## neighbourhoodRiver Valley         -2.168e+00  1.681e+01  -0.129 0.897395
## neighbourhoodRochor                2.544e+01  1.593e+01   1.597 0.110410
## neighbourhoodSembawang             1.168e+01  1.873e+01   0.623 0.533012
## neighbourhoodSengkang              1.455e+01  1.922e+01   0.757 0.449070
## neighbourhoodSerangoon             1.331e+01  1.444e+01   0.922 0.356474
## neighbourhoodSingapore River       9.152e+00  1.832e+01   0.500 0.617433
## neighbourhoodSouthern Islands      1.360e+02  3.531e+01   3.850 0.000120 ***
## neighbourhoodSungei Kadut         -2.315e+01  3.814e+01  -0.607 0.543950
## neighbourhoodTampines              5.890e+01  2.016e+01   2.921 0.003510 **
## neighbourhoodTanglin               9.476e+00  1.686e+01   0.562 0.574217
## neighbourhoodToa Payoh             1.488e+01  1.495e+01   0.996 0.319527
## neighbourhoodWestern Water Catchment -9.954e+01  4.189e+01  -2.376 0.017545 *
## neighbourhoodWoodlands             1.225e+01  1.866e+01   0.656 0.511579
## neighbourhoodYishun                4.452e+01  2.150e+01   2.071 0.038420 *
## latitude                          -2.795e+02  1.447e+02  -1.931 0.053521 .
## longitude                         -3.878e+02  1.236e+02  -3.137 0.001725 **
## room_typeHotel room               -7.667e+00  3.899e+00  -1.966 0.049356 *
```

```
## room_typePrivate room               -8.131e+01  1.995e+00 -40.748  < 2e-16 ***
## room_typeShared room                -9.945e+01  8.658e+00 -11.487  < 2e-16 ***
## minimum_nights                      -7.729e-02  2.185e-02  -3.537 0.000411 ***
## number_of_reviews                   -1.285e-01  2.810e-02  -4.574 4.97e-06 ***
## reviews_per_month                    5.637e-01  8.308e-01   0.679 0.497442
## calculated_host_listings_count      -4.764e-02  1.337e-02  -3.563 0.000372 ***
## availability_365                     4.614e-02  6.306e-03   7.316 3.21e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 48.58 on 3194 degrees of freedom
## Multiple R-squared:  0.4437, Adjusted R-squared:  0.4354
## F-statistic: 53.08 on 48 and 3194 DF,  p-value: < 2.2e-16
```

R squared and Adjusted R squared values for Log LR

```
model_results <- bind_rows(model_results, tibble(method = "Subset Regression Based Model",
MSE = subset_model_mse, R_squared = subset_model_rsquared,
Adjusted_R_squared = subset_model_adj_rsquared))
model_results %>% knitr::kable()
```

method	MSE	R_squared	Adjusted_R_squared
Linear Regression Model	2468.0792385	0.4114382	0.4094344
Linear Regression Model 2	2448.1116222	0.4167419	0.4142123
Log-Linear Regression Model	0.1266871	0.5145178	0.5076843
Subset Regression Based Model	2359.6814823	0.4437317	0.4353720

Random Forest Model

For further experimentation, I tried creating a Random Forest model. We will use RMSE to estimate how well our random forest was able to predict our test set outcomes for the Singapore Airbnb data and compare it with the RMSE of our log linear regression model.

```
RFmodel = randomForest(log(price) ~ neighbourhood_group + latitude + longitude +
  room_type + minimum_nights  + number_of_reviews + reviews_per_month +
    calculated_host_listings_count + availability_365, data = train_set,
  nodesize = 20, ntree = 200)

#RMSE Prediction
predTest = predict(RFmodel, newdata = test_set)
RMSE_2 = sqrt(mean((predTest - log(test_set$price))^2))
```

RMSE values for Log Regression and Random Forest Model

```
rmse_results <- bind_rows(rmse_results, tibble(method = "Random Forest Model",
        RMSE = RMSE_2))
rmse_results %>% knitr::kable()
```

method	RMSE
Log Linear Regression Model	0.3755732
Random Forest Model	0.3032733

As the square root of a variance, RMSE can be interpreted as the standard deviation of the unexplained variance, and has the useful property of being in the same units as the response variable. In this case though, the response variable is log-transformed, so the interpretation is not in the original unit, but more like a percentage deviation. Lower values of RMSE indicate better fit. RMSE is a good measure of how accurately the model predicts the response.

Choosing Optimal Model

A high R squared value means that the deviation of actual points from the fitted points, on average, is small. We prefer a higher adjusted R squared value since none of our models have a big difference between the values of R squared and adjusted R squared. We will rely on Adjusted R squared values to choose our optimal model. The model we created based on Best Subset Regression is most suitable for our prediction with just the right MSE value and R squared value. For our Random Forest model we secured a lower RMSE of 0.303.

Conclusion

The Airbnb dataset provides us with a fantastic source to better understand Singapore's bustling rental landscape. This data was scraped and manipulated accordingly. Between 2010 and 2014, as more and more people adopted the use of the internet as a service provider, the number of listings doubled every year. Through this exploratory data analysis and visualization project, we gained several interesting insights into the Airbnb rental market. Central Region was termed the most popular destination. Filled with the hustle and bustle of city life, the Central region of Singapore is the core area for tourists and locals to meet up and enjoy all that the island nation has to offer. Central consists of some of the main attractions of Singapore along with the downtown core which receives a lot of tourist crowds. Central & Western Region have the most expensive rentals compared to the other boroughs. Prices are higher for rentals closer to country hotspots.

Our purpose in this exercise was to be able to predict the price of an accommodation for a night in Singapore through AirBnB. This exercise was meant to entice private owners to list their property on the platform, given our routines ability to predict the price, the listing would sell for.

Due to COVID-19, people would have cut down their trips instantly resulting in higher availability of Airbnbs for a few months.

Future Scope

We want to expand our analysis to multiple countries and compare patterns and trends among these. From the insights we have derived, we would also like to build predictive models using different features from the dataset. Lastly, we hope to implement the visualizations and techniques used in this project to many other fields and datasets.