Stat425 - AirBnB Data Analysis

December 18, 2017

Team Members : 1. Ankush Agrawal (ankusha2) 2. Apurv Garg (apurvg2) 3. Xiao Liu (xiaol5)

About the data

The dataset used for analysis was sourced from Airbnb listings for Chicago from Inside Airbnb link. The data consists of 95 variables and 5207 observations where each observation represents a listing.

The data is available at the Inside Airbnb site and is scrapped from publicly available information from the Airbnb site. We are using the detailed listings data for Chicago. It was scrapped on 10th May 2017.

In this project, we have done the following:

1. Pre-processed the data for the purpose of EDA and modelling. We have treated the data for missing values, outliers, erroneous data (values which do not make sense). This also helped us understand important features of the data such as the price of a listing, number of reviews, and reviews per month(acts as proxy for booking rate), etc.

2. Used statistical modelling techniques to find interesting insights from the data.

3. Created an interactive dashboard so that the end user can get insights that we have collected in the form of charts, tables and maps.

Getting Started : Data Preprocessing

```r
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=55),tidy=TRUE)

Chicago <- read.csv("listings 2.csv", header = TRUE,stringsAsFactors = F)

a <- strsplit(Chicago$amenities,",")
```

By looking at the summary statistics generated from summary function, we saw the following things:

a. A few variables like scrape_id are either entirely null or have only one single value populated throughout (e.g. experiences_offered) and hence, they can be removed.

```r
# Removing variables that contain only a single value
Chicago <- Chicago
a <- apply(Chicago, 2, unique)
b <- NULL
for (i in 1:length(a)) {
    if (length(a[[i]]) == 1) {
        b <- rbind(b, names(a[i]))
    }
}

b <- as.vector(b)
Chicago[, b] <- NULL
```

This leads to a removal of 9 such variables.

b. All the url related variables can be removed as they contain weblinks which aren't useful for any kind of analysis

```
Chicago[, names(Chicago)[grep("url", names(Chicago))]] <- NULL
```

This leads to removal of 8 such variables.

c. Next we removed all those variables which have more than 80% missing values.

```
a <- apply(Chicago, 2, is.na)
b <- apply(a, 2, sum)
Chicago[, names(b[b/5207 >= 0.8])] <- NULL
```

This leads to a removal of 1 variable.

d. Based on intuition and further inspection we observed that variables such as security_deposit, weekly_price, monthly_price, first_review, last_review, jurisdiction_names, zipcode, street, market, cleaning_fee, transit, amenities, house_rules, host_about, neighborhood_overview, host_neighbourhood, name, interaction, access, space, notes, summary, description, host_name, host_has_profile_pic, host_verifications, host_neighborhood, require_guest_profile_picture, require_guest_phone_verification, calculated_host_listings_count, host_location, transit, neighborhood_overview, house_rules,host_about,license, and requires_license do not add any value to our analysis and should be removed. Hence, we have removed 37 such variables.

```
Chicago[, c("security_deposit", "weekly_price", "monthly_price",
    "first_review", "last_review", "jurisdiction_names",
    "zipcode", "street", "market", "cleaning_fee", "name",
    "interaction", "access", "space", "notes", "summary",
    "description", "host_name", "host_has_profile_pic",
    "host_verifications", "host_neighborhood", "require_guest_profile_picture",
    "require_guest_phone_verification", "calculated_host_listings_count",
    "host_location", "transit", "neighborhood_overview",
    "house_rules", "host_about", "license", "requires_license",
    "host_neighbourhood")] <- NULL
```

e. Removing variables with duplicated values

```
Chicago[, names(Chicago[(duplicated(t(Chicago)))])] <- NULL
```

f. We observed that different listings have different amenities and hence we created flag/dummy variables for the most important amenities we thought a user looks for while booking at Airbnb.

These amenities are TV, Internet, Air Conditioning, Breakfast, Kitchen, and Pets.

```
# adding amenities categories
a <- Chicago$amenities
Chicago$TV <- ifelse(grepl("TV", a, ignore.case = T) ==
    T, 1, 0)
Chicago$Internet <- ifelse(grepl("Internet", a, ignore.case = T) ==
    T, 1, 0)
Chicago$AirCondition <- ifelse(grepl("conditioning", a,
    ignore.case = T) == T, 1, 0)
Chicago$Pets <- ifelse(grepl("Pet", a, ignore.case = T) ==
    T, 1, 0)
Chicago$Pets <- ifelse(grepl("Dog", a, ignore.case = T) ==
    T, 1, Chicago$Pets)
Chicago$Pets <- ifelse(grepl("Cat", a, ignore.case = T) ==
    T, 1, Chicago$Pets)
Chicago$Kitchen <- ifelse(grepl("Kitchen", a, ignore.case = T) ==
```

```
    T, 1, 0)
Chicago$Breakfast <- ifelse(grepl("breakfast", a, ignore.case = T) ==
    T, 1, 0)
Chicago[, c("amenities")] <- NULL
```

The data types of variables which we thought might be significant were converted to the appropriate data types.

```
# Converting price variable to integer
Chicago$price <- sub("\\$", "", Chicago$price)
Chicago$price <- sub(",", "", Chicago$price)
Chicago$price <- as.integer(Chicago$price)

# Changing the default character data type of
# categorical variable to factor
Chicago$host_response_time <- as.factor(Chicago$host_response_time)
Chicago$host_is_superhost <- as.factor(Chicago$host_is_superhost)
Chicago$host_identity_verified <- as.factor(Chicago$host_identity_verified)
Chicago$neighbourhood_cleansed <- as.factor(Chicago$neighbourhood_cleansed)
Chicago$is_location_exact <- as.factor(Chicago$is_location_exact)
Chicago$property_type <- as.factor(Chicago$property_type)
Chicago$room_type <- as.factor(Chicago$room_type)
Chicago$bed_type <- as.factor(Chicago$bed_type)
Chicago$calendar_updated <- as.factor(Chicago$calendar_updated)
Chicago$instant_bookable <- as.factor(Chicago$instant_bookable)
Chicago$cancellation_policy <- as.factor(Chicago$cancellation_policy)

# Treating host_response_rate and extra_people from
# character to a numeric variable
Chicago$host_response_rate <- as.numeric(sub("%", "", Chicago$host_response_rate))
Chicago$host_response_rate <- Chicago$host_response_rate/100
Chicago$extra_people <- as.numeric(sub("\\$", "", Chicago$extra_people))
```

We saw that some listings have 0 as their price, which is not possible and is possibly an erroneous data. Hence, we have removed such listings.

```
# Removing listings which have price = 0
Chicago <- Chicago[-c(717, 1334, 3093), ]
```

We also created a variable about the number of days since the listing was on Airbnb. We leveraged the variable host_since which gives us the dates since the listings have been on Airbnb and we got the number of days by subtracting this date from the 10

May 2017, i.e. the date when the data was scrapped.

```
Chicago$host_since <- as.Date(Chicago$host_since)
Chicago$host_since <- as.Date("2017-05-10") - Chicago$host_since
```

In order to reduce the number of levels in factor variables with large number of levels, we clubbed the factors with low number of listings. In the dataset, we observed that neighbourhood_cleansed, which gives us an idea of the nighbourhood of the listing, has 72 levels.

```
library(dplyr)
a <- Chicago %>% group_by(neighbourhood_cleansed) %>% summarise(len = length(neighbourhood_cleansed))
# sum(a[(a$len<70),2])

Chicago <- merge(Chicago, a, by = "neighbourhood_cleansed")
```

```r
Chicago$neighbourhood_cleansed <- as.character(Chicago$neighbourhood_cleansed)
Chicago$neighbourhood_cleansed <- ifelse(Chicago$len < 150,
    "Others", Chicago$neighbourhood_cleansed)
Chicago$len <- NULL
```

Missing Value Treatment

We checked for all types of missing values in all variables of the final dataset. This includes is.na, "", and "N/A".

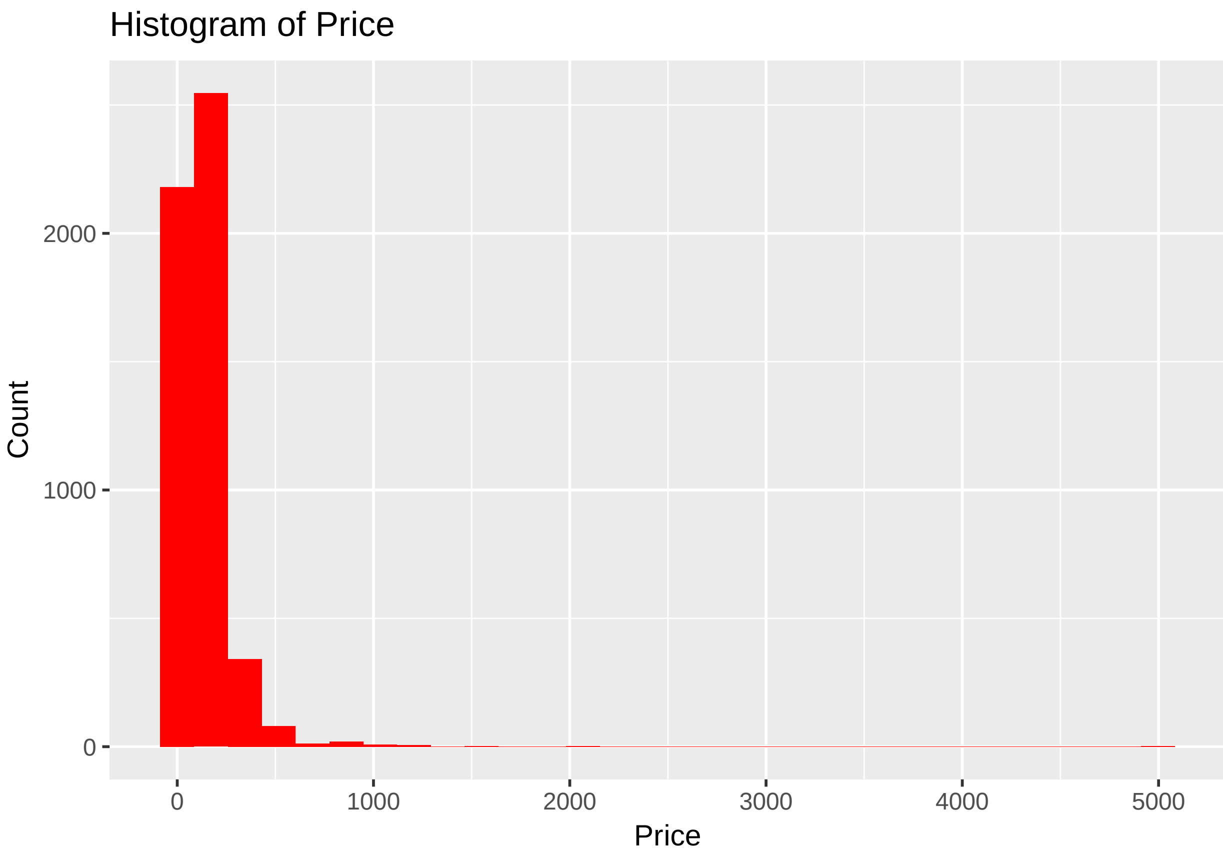
```r
# checking missing values in different columns
missing = data.frame(col = colnames(Chicago), type = sapply(Chicago,
    class), missing = sapply(Chicago, function(x) sum(is.na(x) |
    x == "" | x == "N/A"))/nrow(Chicago) * 100)
```

We treated the missing values as follows:

```r
# Replace missing values in host_response_time with the
# most common occurring category (its mode)
Chicago$host_response_time <- sub("N/A", "within an hour",
    Chicago$host_response_time)

# replace missing values in host_response rate with mean
# values observed for host_response_time when it is
# within an hour
Chicago$host_response_rate <- ifelse(is.na(Chicago$host_response_rate) ==
    T, 0.99, Chicago$host_response_rate)

# Replace missing values in bathrooms, bedrooms and beds
# with the median value
Chicago$bathrooms <- ifelse(is.na(Chicago$bathrooms) ==
    T, 1, Chicago$bathrooms)
Chicago$bedrooms <- ifelse(is.na(Chicago$bedrooms) == T,
    1, Chicago$bedrooms)
Chicago$beds <- ifelse(is.na(Chicago$beds) == T, 1, Chicago$beds)

# Replace the missing values in review scores with the
# median values as well
Chicago$review_scores_rating <- ifelse(is.na(Chicago$review_scores_rating) ==
    T, 97, Chicago$review_scores_rating)
Chicago$review_scores_accuracy <- ifelse(is.na(Chicago$review_scores_accuracy) ==
    T, 10, Chicago$review_scores_accuracy)
Chicago$review_scores_cleanliness <- ifelse(is.na(Chicago$review_scores_cleanliness) ==
    T, 10, Chicago$review_scores_cleanliness)
Chicago$review_scores_checkin <- ifelse(is.na(Chicago$review_scores_checkin) ==
    T, 10, Chicago$review_scores_checkin)
Chicago$review_scores_communication <- ifelse(is.na(Chicago$review_scores_communication) ==
    T, 10, Chicago$review_scores_communication)
Chicago$review_scores_location <- ifelse(is.na(Chicago$review_scores_location) ==
    T, 10, Chicago$review_scores_location)
Chicago$review_scores_value <- ifelse(is.na(Chicago$review_scores_value) ==
    T, 10, Chicago$review_scores_value)
Chicago$reviews_per_month <- ifelse(is.na(Chicago$reviews_per_month) ==
    T, 1.55, Chicago$reviews_per_month)
```

Outliers

```
# Histogram of price
library(ggplot2)
ggplot(data = Chicago, aes(price)) + geom_histogram(fill = "red") +
    labs(title = "Histogram of Price") + labs(x = "Price",
    y = "Count")
```




```
# Percentile of price
quantile(Chicago$price, c(0.9, 0.95, 0.97, 0.975, 0.98,
    0.99, 0.995, 0.999, 0.9999))
```

```
##     90%     95%     97%   97.5%     98%     99%   99.5%   99.9%  99.99%
##  250.00  350.00  400.00  450.00  500.00  649.04  900.00 1898.50 5000.00
```

As we can see from the histogram as well as the percentile distribution of Price, there are extreme values in Price. So we performed Winsorization at 99% level and captured the maximum value of price at 650 USD.

```
# Capture the extreme values of Price at 99 percentile
# level
Chicago$price <- ifelse(Chicago$price > 650, 650, Chicago$price)
```

Dashboard

http://apurvg2.shinyapps.io/airbnb/

Our UI should contain the following components:

A map which allows users to look at Airbnb listings on a map. There are also filters based on the Price range and Neighbourhoods.

The map on the 2nd tab shows the distribution of listings based on the prices for different room types.

Graphs provide the functionalities for users to have a look at box-plots for different prices and the number of booking values based on different categorical variables like room_type, bed_type, cancellation policy, and property type.

User can also look at the variation of prices and number of bookings vs host_since(a scatter plot). We have provided multiple filters like number of bathroom, bedrooms and guests so that user can see the metrics based on their requirements.

Note: We have used plotly to make graphs which is an interactive graph library. It allows user to subselect factors in the graph by clicking on the legends.

Analysis 1: Comparing the booking rate based on cancellation policy

We wish to know whether the booking rate is different for different cancellation policies.

H0: The cancellation policy has no effect on the average booking rate.

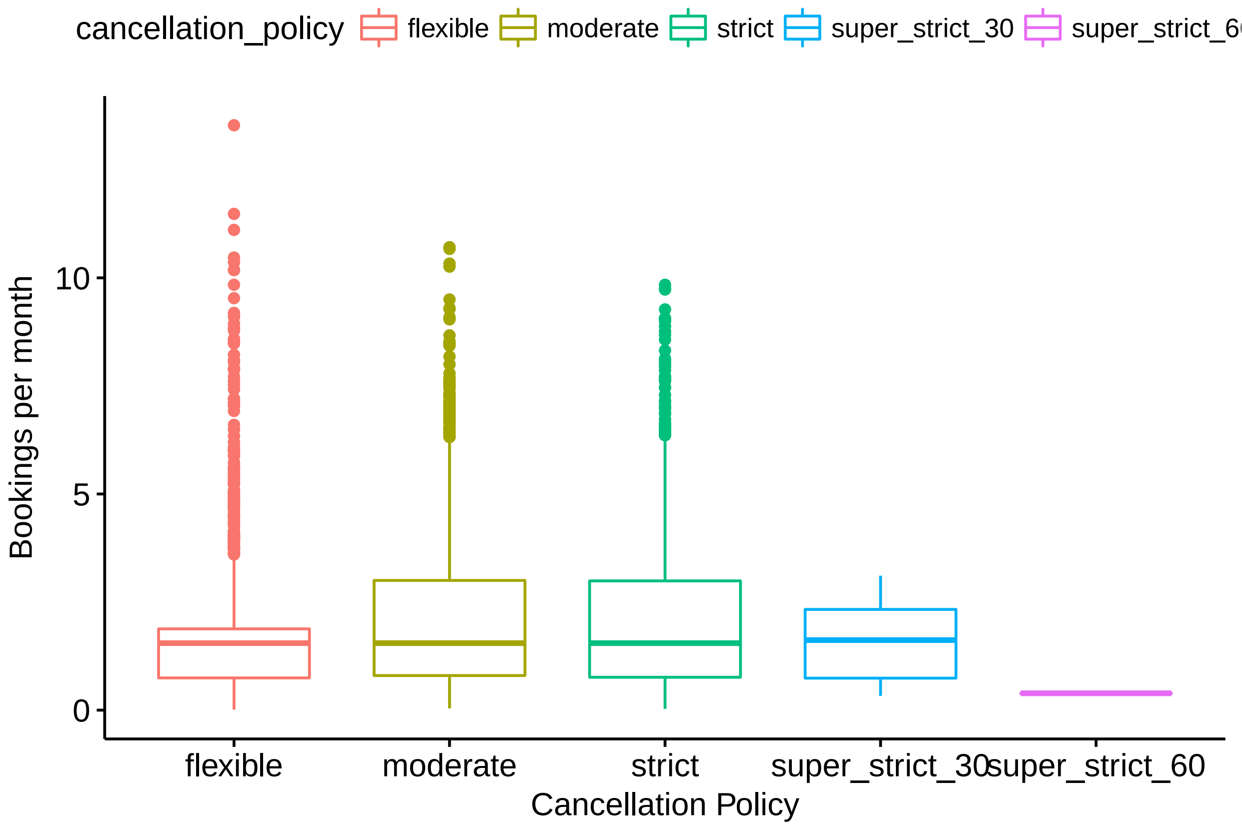
H1: The average booking rates are different with different cancellation policies.

Assumption: The monthly booking rate here is based on reviews per month. This implies that it is not actually the booking rate (including cancellation) but it is the booking rate where people have actually stayed at the accommodations(or the staying rate).

Note: We used variable reviews_per_month as a measure for the bookings rate.

We first got a box-plot for qualitative check.

```r
# Visualize the distribution of reviews_per_month for
# different cancellation policy using a boxplot
library(ggpubr)
ggboxplot(Chicago, x = "cancellation_policy", y = "reviews_per_month",
    color = "cancellation_policy", ylab = "Bookings per month",
    xlab = "Cancellation Policy")
```



Initial inspection of the data using the boxplot suggests that there are differences in the booking rates for different cancellation policies: the accommodations with Strict and Moderate cancellation policies have higher average booking rates (staying rate). To investigate for if there are significant differences among groups quantitatively, we then fited an one-way ANOVA model as below.

```
options(width = 90)
# Compute the analysis of variance
anova1 <- aov(reviews_per_month ~ cancellation_policy, data = Chicago)

# Summary of the analysis
summary(anova1)
```

```
##                       Df Sum Sq Mean Sq F value   Pr(>F)    
## cancellation_policy    4    136   33.93   11.31 3.89e-09 ***
## Residuals           5199  15600    3.00                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As the p-value is less than the significance level 0.05 in the model summary, we conclude that there are significant differences among the groups of cancellation policy.

In one-way ANOVA test, a significant p-value indicates that some of the group means are different, but we don't know which pairs of groups are different.

As the ANOVA test is significant, we can use Tukey's method for performing multiple pairwise-comparisons between the means of pairs of groups.

```
TukeyHSD(anova1)
```

```
##   Tukey multiple comparisons of means
```

```
##     95% family-wise confidence level
##
## Fit: aov(formula = reviews_per_month ~ cancellation_policy, data = Chicago)
##
## $cancellation_policy
##                                    diff        lwr        upr     p adj
## moderate-flexible                0.39114084  0.2169511 0.56533061 0.0000000
## strict-flexible                  0.33487110  0.1725756 0.49716656 0.0000002
## super_strict_30-flexible        -0.12762107 -2.2453254 1.99008327 0.9998363
## super_strict_60-flexible        -1.36362107 -6.0920441 3.36480193 0.9344815
## strict-moderate                 -0.05626973 -0.2111722 0.09863276 0.8594029
## super_strict_30-moderate        -0.51876190 -2.6359125 1.59838868 0.9631125
## super_strict_60-moderate        -1.75476190 -6.4829369 2.97341311 0.8496426
## super_strict_30-strict          -0.46249217 -2.5786973 1.65371301 0.9756672
## super_strict_60-strict          -1.69849217 -6.4262439 3.02925959 0.8642318
## super_strict_60-super_strict_30 -1.23600000 -6.4138344 3.94183435 0.9664290
```

We can see from the results that the average booking rate is significantly different for moderate and flexible cancellation policies as well as Strict and flexible cancellation policies.

Therefore, we rejected our null hypothesis, which implies, there are significant difference in average booking rate among different cancellation policies. As we can see from the box plot, the accomodations with Strict and Moderate cancellation policies have higher average booking rate(staying rate)

Analysis 2: Comparing the Review Scores based on bed type

This analysis is done to see if the type of bed provided by a host in Airbnb has any effects on the average review score for that listing

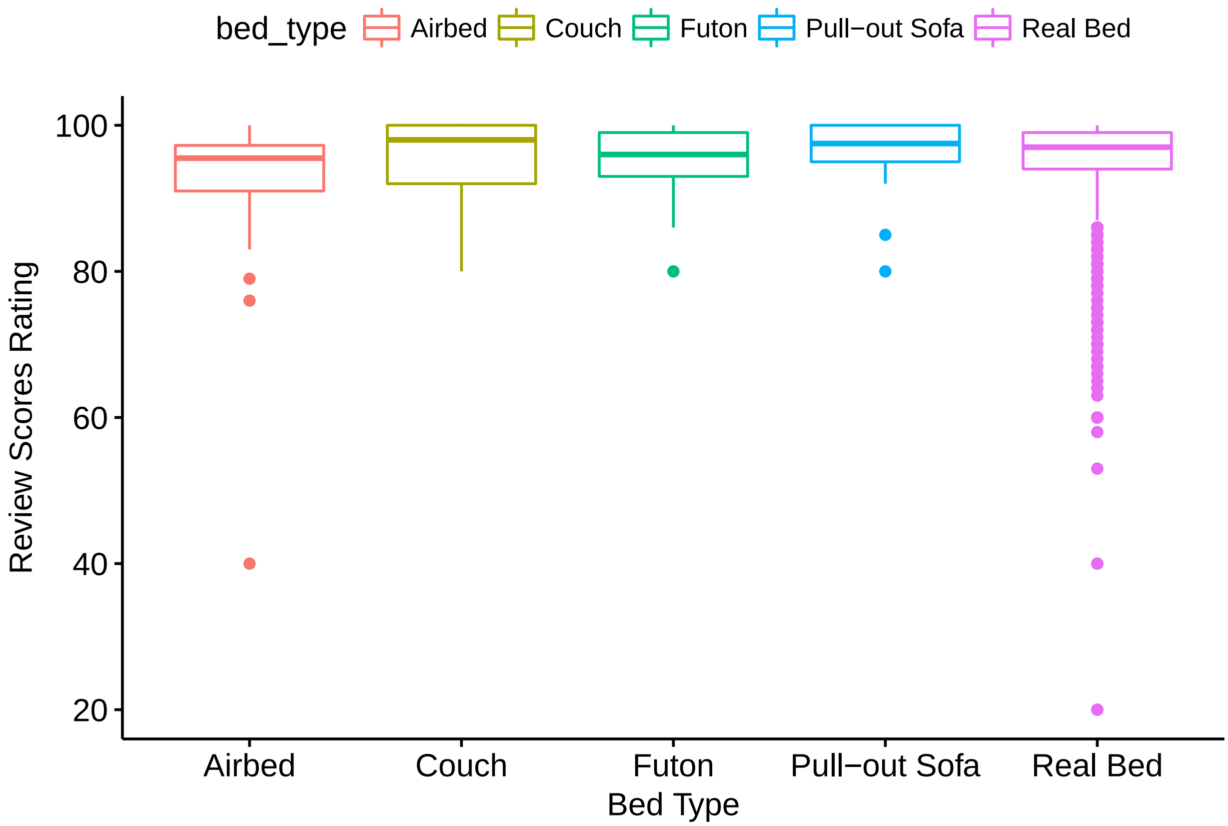
H0: Different bed types do not have any effects on average review scores.

H1: The average review score changes with different bed types.

Assumption: Different bed types are assumed to be providing different comfort levels(bed types are proportional to comfort), therefore, the average review scores should be different for different bed types.

We first got a box-plot for qualitative check.

```r
# Visualizing the distribution of review_scores_rating
# for different bed type using a boxplot
ggboxplot(Chicago, x = "bed_type", y = "review_scores_rating",
    color = "bed_type", ylab = "Review Scores Rating", xlab = "Bed Type")
```



Initial inspection of the data using the boxplot suggests that there are no differences in the review rating for different bed types. To investigate for if there are differences among different groups quantitatively, we fitted an one-way ANOVA model.

```
# Compute the analysis of variance
anova2 <- aov(review_scores_rating ~ bed_type, data = Chicago)

# Summary of the analysis
summary(anova2)
```

```
##               Df Sum Sq Mean Sq F value Pr(>F)
## bed_type       4    269   67.32   2.035 0.0867 .
## Residuals   5199 171951   33.07
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since P-value is larger than 0.05, we accpet the null hypothesis that the average review scores do not change with the bed types. That is not quite what we epxpected based on our intuition. Intuitively, people prefer comfort and hence would prefer "Real Bed" versus "Sofa" or "Couch".

However, our results imply that when it comes to high average review scores, bed type does not have any significant effects on it. Also, this shows that people tend to base their ratings not just on the comfort.

Analysis 3: Predicting the prices of listings & Finding factors impacting prices

```
##
## Call:
```

```
## lm(formula = price ~ as.factor(neighbourhood_cleansed) + host_since +
##     Breakfast + Kitchen + Pets + AirCondition + Internet + TV +
##     reviews_per_month + review_scores_rating + number_of_reviews +
##     review_scores_accuracy + review_scores_value + review_scores_cleanliness +
##     review_scores_checkin + review_scores_communication + review_scores_location +
##     as.factor(cancellation_policy) + guests_included + beds +
##     as.factor(bed_type) + bathrooms + as.factor(property_type) +
##     as.factor(room_type), data = Chicago)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -484.79  -37.19   -7.46   20.64  579.94
##
## Coefficients:
##                                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                      -8.501e+01  3.173e+01  -2.679 0.007410 **
## as.factor(neighbourhood_cleansed)Lake View        3.724e+01  6.864e+00   5.426 6.03e-08 ***
## as.factor(neighbourhood_cleansed)Lincoln Park     3.684e+01  7.534e+00   4.890 1.04e-06 ***
## as.factor(neighbourhood_cleansed)Logan Square     5.612e+00  7.098e+00   0.791 0.429202
## as.factor(neighbourhood_cleansed)Loop             3.583e+01  7.641e+00   4.689 2.82e-06 ***
## as.factor(neighbourhood_cleansed)Lower West Side -6.274e+00  8.373e+00  -0.749 0.453698
## as.factor(neighbourhood_cleansed)Near North Side  5.842e+01  7.088e+00   8.242  < 2e-16 ***
## as.factor(neighbourhood_cleansed)Near West Side   1.459e+01  7.894e+00   1.848 0.064694 .
## as.factor(neighbourhood_cleansed)Others          -7.084e+00  6.395e+00  -1.108 0.268011
## as.factor(neighbourhood_cleansed)Uptown           5.668e+00  8.020e+00   0.707 0.479779
## as.factor(neighbourhood_cleansed)West Town        9.093e+00  6.670e+00   1.363 0.172891
## host_since                                        3.005e-03  1.844e-03   1.630 0.103166
## Breakfast                                        -8.138e+00  2.841e+00  -2.865 0.004191 **
## Kitchen                                          -7.689e+00  4.740e+00  -1.622 0.104790
## Pets                                             -2.295e+00  2.364e+00  -0.971 0.331620
## AirCondition                                      9.085e+00  3.731e+00   2.435 0.014928 *
## Internet                                         -1.197e+01  6.244e+00  -1.917 0.055267 .
## TV                                                1.198e+01  2.876e+00   4.166 3.14e-05 ***
## reviews_per_month                                -9.103e+00  7.676e-01 -11.859  < 2e-16 ***
## review_scores_rating                              4.476e-01  3.405e-01   1.315 0.188736
## number_of_reviews                                -2.870e-02  3.540e-02  -0.811 0.417479
## review_scores_accuracy                           -2.036e+00  2.594e+00  -0.785 0.432482
## review_scores_value                              -6.184e+00  2.513e+00  -2.461 0.013901 *
## review_scores_cleanliness                         1.043e+01  2.018e+00   5.169 2.44e-07 ***
## review_scores_checkin                             3.520e+00  3.209e+00   1.097 0.272782
## review_scores_communication                      -7.741e+00  3.551e+00  -2.180 0.029336 *
## review_scores_location                            1.128e+01  1.949e+00   5.785 7.68e-09 ***
## as.factor(cancellation_policy)moderate           -1.203e+00  2.870e+00  -0.419 0.675155
## as.factor(cancellation_policy)strict             -6.120e+00  2.788e+00  -2.195 0.028198 *
## as.factor(cancellation_policy)super_strict_30     8.057e+01  3.389e+01   2.377 0.017474 *
## as.factor(cancellation_policy)super_strict_60    -7.042e+01  7.544e+01  -0.933 0.350688
## guests_included                                   5.926e+00  8.905e-01   6.655 3.12e-11 ***
## beds                                              1.684e+01  1.069e+00  15.750  < 2e-16 ***
## as.factor(bed_type)Couch                          1.693e+01  2.071e+01   0.818 0.413511
## as.factor(bed_type)Futon                          1.739e+01  1.412e+01   1.231 0.218233
## as.factor(bed_type)Pull-out Sofa                 -3.263e+00  1.636e+01  -0.199 0.841945
## as.factor(bed_type)Real Bed                       3.029e+00  9.678e+00   0.313 0.754324
## bathrooms                                         4.943e+01  2.194e+00  22.532  < 2e-16 ***
## as.factor(property_type)Bed & Breakfast           4.245e+01  1.807e+01   2.349 0.018863 *
```

```
## as.factor(property_type)Boat                     1.955e+02  1.899e+01  10.291  < 2e-16 ***
## as.factor(property_type)Boutique hotel           7.285e+01  2.125e+01   3.429 0.000611 ***
## as.factor(property_type)Bungalow                -3.184e+01  3.084e+01  -1.032 0.301929
## as.factor(property_type)Condominium              1.464e+01  3.436e+00   4.260 2.09e-05 ***
## as.factor(property_type)Dorm                    -1.083e+02  1.858e+01  -5.828 5.94e-09 ***
## as.factor(property_type)Guest suite             -2.243e+01  2.665e+01  -0.842 0.400061
## as.factor(property_type)Guesthouse               2.893e+01  1.738e+01   1.665 0.095937 .
## as.factor(property_type)Hostel                  -1.848e+01  3.183e+01  -0.581 0.561598
## as.factor(property_type)House                    1.296e+01  3.316e+00   3.908 9.43e-05 ***
## as.factor(property_type)In-law                  -5.453e+01  5.324e+01  -1.024 0.305817
## as.factor(property_type)Loft                     2.025e+01  7.833e+00   2.585 0.009770 **
## as.factor(property_type)Other                    5.473e+01  1.275e+01   4.294 1.79e-05 ***
## as.factor(property_type)Serviced apartment       8.585e+01  7.555e+01   1.136 0.255896
## as.factor(property_type)Timeshare                5.874e+01  7.545e+01   0.778 0.436342
## as.factor(property_type)Townhouse                1.741e+01  8.132e+00   2.141 0.032341 *
## as.factor(property_type)Vacation home            1.342e+02  7.526e+01   1.783 0.074578 .
## as.factor(property_type)Villa                    1.008e+01  4.352e+01   0.232 0.816797
## as.factor(room_type)Private room                -4.988e+01  2.699e+00 -18.478  < 2e-16 ***
## as.factor(room_type)Shared room                 -8.492e+01  6.730e+00 -12.618  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 75.12 on 5146 degrees of freedom
## Multiple R-squared:  0.516,  Adjusted R-squared:  0.5107
## F-statistic: 96.26 on 57 and 5146 DF,  p-value: < 2.2e-16
```

Following are the results that we obtained from the above linear regression model along with the hypothesis/explanations of behaviors of the variables:

Based on the p-values for the above regression model, amenities like TV and Air Conditioning significantly impact the prices. Since the Beta-values are positive, we can say that having TV and AC leads to increase in prices (considering other variables are fixed). This can be because AC and TV are comparatively expensive amenties.

Also, the booking rate is inversely proportional to the prices considering the other factors remain unchanged.

Number of guests, number of beds, and number of bathrooms directly affect the prices. As the number of guests, beds or bathrooms increases, prices increase (considering amenities/comfort and location is same). The hotels share the same theory.

The categorical variables: neighbourhood, property type, and room type have significant impact on prices. This is generally true as properties in luxurious localities will be more expensive than other areas. Also, a shared room will be cheaper than a whole house/apartment in the same locality.