STA 141A Final Project

Investigation of the AirBnb NYC dataset

Johann Piedras: jpiedras@ucdavis.edu
Alvin Hui: abhui@ucdavis.edu
Marianne Adamian: mkadamian@ucdavis.edu
John Tran: johtran@ucdavis.edu

Date: 2019-12-08

Contents

Introduction **3**

Background . . . 3

Question 1 . . . 3

Question 2 . . . 3

Question 3 . . . 3

EDA - Exploratory Data Analysis **4**

A) Reading Data . . . 4

1. Importing Data And Including Libraries . . . 4

2. Checking Structure . . . 4

3. Neighborhood Denisty . . . 4

4. Property Types . . . 5

5. Room Types . . . 5

6. Externalities (Random Findings) . . . 5

B) Visual Exploration . . . 5

1. Price and Location . . . 5

2. Review Score and Number of Reviews . . . 6

Question 1: Price And Location **6**

Expectations . . . 6

Average Prices Per Neighborhood . . . 6

Table For Neighborhood Density . . . 7

Simple Linear Regression Price and Neighbourhood . . . 7

Narrowing Data To Manhattan **8**

Model Creation Using AIC Methods . . . 8

Linear Regression Using Best AIC Model For Manhattan . . . 8

Expanding the optimal Model to the whole dataset(Not just Manhattan) . . . 8

Creating Dummies For optimal categorical variables . . . 9

Linear Regression With Optimal Variables As Dummies . . . 9

Question 2 Reviews And Score Affect On Price **9**

Linear regression with number of reviews and review scores rating . . . 9

Transformation Of Price . . . 10

AIC Using Interaction Terms **10**

Question 3: Predicting Property Type Using KNN **10**

Expectations . . . 10

Training Data and Test Data . . . 11

KNN Prediction Confusion Table . . . 11

Misclassification Error . . . 11

Engineering A More Appropiate K . . . 11

Plotting KNN Results of different K Values . . . 11

KNN with engineered K and Misclassification . . . 12

Conclusion **12**

Q1 . . . 12

Q2 . . . 13

Q3 . . . 13

Overall Conclusion **13**

Introduction

Tasks:

Johann Piedras created the RMD file, inputted the exploratory analysis, helped on Question 1, and worked on Question 3. Marianne wrote the introduction, the conclusion for each question and the overall conclusion of the project. Marianne and Johann helped each other in organizing the RMD file. Alvin and John consecutively worked on Question 2 as well as helped in cleaning the final report by identifying irrelevant codes. As a group, we explored the data, created lines of codes to produce plots and relationships to assess the relationships between variables in our data and came up with the three questions.

Background

This project utilizes the Airbnb data recorded during the time span of seven years (June 2008 - August 2015). Airbnb is an online marketplace that connects people who want to rent out their homes to people who are looking for accommodations in that locale. This data focuses on the five boroughs in New York, such as Manhattan, Brooklyn, and Queens, which is listed under the Neighborhood variable and looks at the Property Type, Room Type and Price for each listed Airbnb. The Neighborhood variable consists of the five boroughs in New York that were used in the dataset. The data also lists the Number of Reviews for each Airbnb as well. The Property Type identifies the type of Airbnb it is, such as a home, apartment, or townhouse. The specific type of rooms, for example private room or shared room, for the Airbnb is listed under the Room Type variable. The Price variable records the prices of Airbnb, and the Number of Reviews are the number of reviews written by Airbnb guests. The primary goal of this project was to determine whether a regulatory relationship exists between price and the neighborhood of the Airbnb. As a secondary question, we attempted to examine the relationship between the price and the reviews given for an Airbnb. Lastly, our third question involved predicting the property type of a Manhattan Airbnb listing. To understand the research questions of this project, we identified the coefficient of determination by using multiple linear regression models and used k-nearest neighbors. We also created visualizations by creating histograms and boxplots to recognize the relationship between price and neighborhood of the Airbnb. Answering these questions would provide insight into which Airbnb is a better value for him or herself. By clarifying these relationships, one can establish a better foundation for further research into choosing a sufficient Airbnb for oneself.

Question 1

What is the relationship between the price of an Airbnb and the location of the Airbnb?

Question 2

Do the reviews for the listing affect the price of the Airbnb?

Question 3

Can we predict the type of property for an Airbnb listing in Manhattan?

EDA - Exploratory Data Analysis

In the exploratory data analyst section, we briefly outline the steps in our journey to answer the 3 questions. Feel free to move forward to the first question since the initial steps are just procedure.

A) Reading Data

1. Importing Data And Including Libraries

The libraries we used to conduct are analysis include, readxl to read the AirBnb Data, ggplot2 for visualizations, Tidyverse/Stringer indexing, dummies for categorical data analysis, and more.

```
airbnb <- read_excel("airbnb.xlsx")
data<- data.frame(airbnb, na.rm= TRUE)
```

2. Checking Structure

Now the data exploration really starts. We start by using the str() function. With the structure function, we are given the dimensions of the dataset, a list of each column, along with the class type of each.

```
## 'data.frame':    30478 obs. of  14 variables:
##  $ Host.Id                   : num  5162530 33134899 39608626 500 500 ...
##  $ Host.Since                : POSIXct, format: NA NA ...
##  $ Name                      : chr  "1 Bedroom in Prime Williamsburg" "Sunny, Private room in Bushwi
##  $ Neighbourhood             : chr  "Brooklyn" "Brooklyn" "Manhattan" "Manhattan" ...
##  $ Property.Type             : chr  "Apartment" "Apartment" "Apartment" "Apartment" ...
##  $ Review.Scores.Rating..bin.: num  NA NA NA NA 95 100 100 90 90 95 ...
##  $ Room.Type                 : chr  "Entire home/apt" "Private room" "Private room" "Entire home/apt
##  $ Zipcode                   : num  11249 11206 10032 10024 10036 ...
##  $ Beds                      : num  1 1 1 3 3 1 1 1 2 2 ...
##  $ Number.of.Records         : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Number.Of.Reviews         : num  0 1 1 0 39 4 9 80 95 23 ...
##  $ Price                     : num  145 37 28 199 549 149 250 90 270 290 ...
##  $ Review.Scores.Rating      : num  NA NA NA NA 96 100 100 94 90 96 ...
##  $ na.rm                     : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
```

3. Neighborhood Denisty

From the str() function, we see that there are 4 or more neighborhoods we can focus on in this data set. To observe the number of Airbnb's in each neighborhood we use the table function. The table() output clearly show that Mannhattan and Brooklyn have a lot more listing than the other cities. Due to the high concentration of airbnb's in two cities, we might have to exclusively focus on Manhattan or Brooklyn.

```
##
##         Bronx      Brooklyn     Manhattan        Queens Staten Island
##           345         11675         16033          2278           147
```

4. Property Types

Next we investigate the number of property types and discover that there are 20 types of properties; one which is a NA option that we may have to remove.

```
##
##       Apartment Bed & Breakfast           House            Loft
##           27102             180            2090             753
##       Townhouse
##             136
```

5. Room Types

Finally, room types are observed using table() to grasp their impact in the dataset. The results show that there are two out of three room types that dominate the listings.

```
##
## Entire home/apt    Private room     Shared room
##           17024           12609             845
```

6. Externalities (Random Findings)

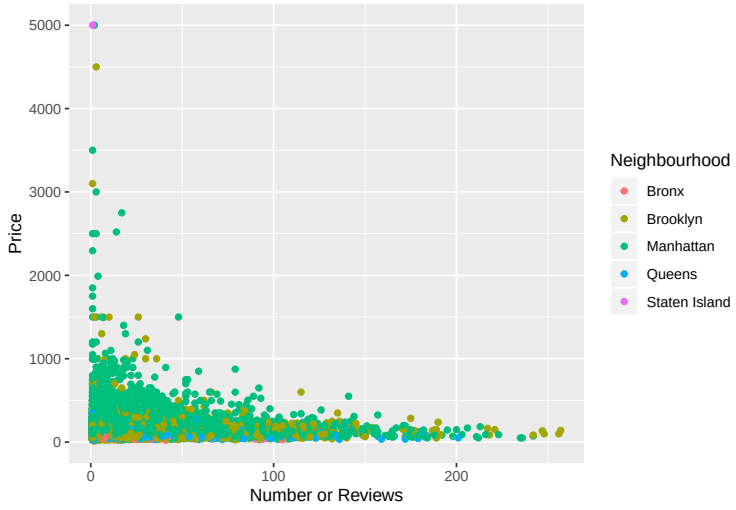
To answer our 3 questions, we hypothesised that earlier findings would be useful, but we also looked at other variables for fun. There are 189 zipcodes we can work with that span 5 neighborhoods in NY. There are 24,421 unique host, but 30,478 total listings

B) Visual Exploration

1. Price and Location

We are interested in detecting the relationship between the price of a location and the number of reviews for a listing. Interpreting from the dot plot, it can be seen that there are more reviews for Airbnbs that are lower in price. We conclude this to be reasonable because lower priced Airbnbs are accessible to more people, therefore it is more likely to have more reviews.

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



2. Review Score and Number of Reviews

Furthermore, we determined the relationship between the number of reviews and the review scores rating. From the dot plot, we identify that there are more reviews for a higher rating Airbnb compared to a lower rating airbnb.

Question 1: Price And Location

What is the relationship between the price of an Airbnb and the location of the Airbnb?

Expectations

The boroughs with a higher cost of living will consist of costlier Airbnbs. We used data from streeteasy (https://streeteasy.com/blog/cost-of-living-nyc-income-housing-all-5-boroughs/) to rank the boroughs from most to least expensive because our dataset does not include cost of living information.

1. Manhattan : $18,900 | $36,252
2. Queens : $16,812 | $29,256
3. Brooklyn : $15,144 | $31,908
4. Staten Island : $14,292 | $28,752
5. Bronx : $13,176 | $28,548

Therefore we expect, to see the highest prices in Manhattan, and the least expensive in Bronx

Average Prices Per Neighborhood

```
##   data1$Neighbourhood data1$Price
## 1               Bronx    75.19535
## 2            Brooklyn   127.77348
## 3           Manhattan   183.55538
## 4              Queens    96.81345
## 5       Staten Island   147.69149
```

Our expectations were met.Cities with a higher average have more expensive Airbnbs. To double check our plot, we wanted to observe the density of Airbnbs of each borough.

Table For Neighborhood Density

```
##
##         Bronx      Brooklyn     Manhattan        Queens Staten Island
##           215          8432         11693          1576            94
```

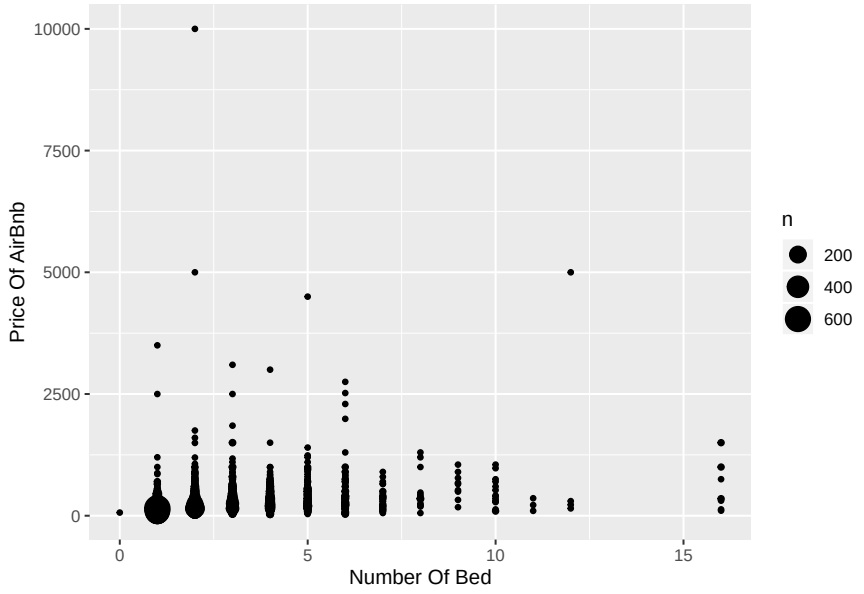
As suspected, there's a higher density of Airbnbs in Manhattan, Brooklyn, and Queens. Going forward, we will start taking their high density in mind. However, before we minimize the data, we will make a simple linear regression to find the relationship between price and borough.

Simple Linear Regression Price and Neighbourhood

After running a linear regression model just using price and neighborhood, we see that the R^2 value is really low (0.0460396). This calculation informs us that neighborhood alone is not a strong predictor of price. Though the p-values for all neighborhoods except for Queens were low, the R^2 value is not high enough to confirm that neighborhood solely is a predictor of price.

Linear Regression Price and Beds

Next we decide to observe the influence of price based on number of beds. We do this using a geom_count from ggplot to observe visually the density of beds based on price, and then we run a simple linear regression model using $lm(Price \sim Beds)$.



From the plot, we conclude that number of beds will have a strong bias towards 1 bed. It can also be ovserved that there a lot of outliers in the dataset.Just to double check, we do a simple linear regression to find the strength of beds as a predictor of price. The R^2 value is low(0.1351408), therefore we can pass on beds as an indicator.

Narrowing Data To Manhattan

Because we can't seem to find a strong predictor of price for the entire dataset, we will filter out rows that are not from Manhattan since it had the most AirBnb listings. Furthermore we will limit the dataframe to include room type, number of reviews, scores, property type, and beds.

Model Creation Using AIC Methods

With the narrow down data(just mannhattan), we will use the AIC method to help determine which linear regression model has the best fit to predict price.

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## Price ~ 1
##
## Final Model:
## Price ~ Beds + Room.Type + Review.Scores.Rating + Property.Type +
##     Number.Of.Reviews
##
##
##                      Step Df   Deviance Resid. Df Resid. Dev      AIC
## 1                                           11524  339167213 118591.1
## 2                  + Beds  1 41201077.2     11523  297966136 117100.5
## 3             + Room.Type  2 20410041.1     11521  277556095 116286.7
## 4 + Review.Scores.Rating  1   748006.2     11520  276808089 116257.6
## 5        + Property.Type 14  1199125.1     11506  275608964 116235.6
## 6    + Number.Of.Reviews  1   290045.3     11505  275318918 116225.4
```

After using forward selection, we have determined that the best model linear regression to determine price in manhattan is $Price \sim Beds + Room.Type + Review.Scores.Rating + Property.Type + Number.Of.Reviews$

Linear Regression Using Best AIC Model For Manhattan

To test the strength the new model based on AIC best fit, the R^2 value is now(0.1882502). Compared to the previous models (Beds and Price= 0.1351408), (Price and Nieghbourhood = 0.0460396), the AIC optimal model is a better predictor of price, however the R^2 is really low. To combat this, we decided to apply a transformation to price. The transformation of the LM is as follows $lm(\frac{1}{price} \sim Beds + Room.Type + Review.Scores.Rating + Property.Type + Number.Of.Reviews$. The transformed linear model had a R^2 of 0.5059959 which is almost twice as large than best AIC model without transformation. These results are more satsifactory, however the R^2 is still relatively low.

Expanding the optimal Model to the whole dataset(Not just Manhattan)

Just to experiment, we decided to run the best AIC model from Manhattan to the entire dataset. The R^2 value we got is 0.2252882. Surprisingly, the R^2 value is higher than our previous models that just emphasized Airbnb listings in Manhattan without transformations. Given this surprise, we decide to clean up the data further in the entire dataset, then run the AIC model on the cleaned dataset.

Creating Dummies For optimal categorical variables

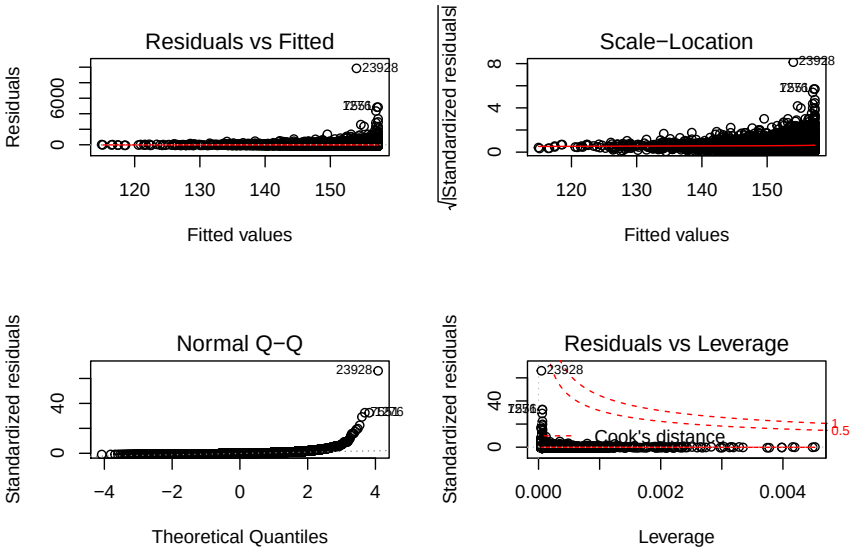
First and foremost we create dummies for the categorical data. This is important because room type and property.type are caterogrical variables. Following dummy creation, we combine the dummies the the original dataframe, and run the linear regression model afterwards.

Linear Regression With Optimal Variables As Dummies

As a result, the optimal AIC model with dummy variables, along with a transformation of price outputs a R^2 of (0.573407) which is the highest R^2 we've been able to obtain so far. Overall, we have determined that finding a relationship between price and variables in the Airbnb dataset show a vague relatiobship, which means that creating a model that predicts price would have to be more complex.

Question 2 Reviews And Score Affect On Price

Linear regression with number of reviews and review scores rating



Following the discoveries in question one, we decided to take a deeper dive into reviews of Airbnb listings.Here we are trying to answer the question whether reviews have any effect on price. By using linear regression, we decided to see if there is any significance that the Number.Of.Reviews or Review.Scores.Rating have in price. We started out with the most simplest model possible Price and just Review.Scores.Rating. Although Review.Scores.Rating was significant, the R-squared value was only (0.0007), which means that the regression did not explain much of the variability in Price.

Possible reasons for this poor fit include outliers and high leverage points, as shown in the Residual vs Leverage plot, the negative estimate for Number.Of.Reviews, which is impossible to get a negative number of reviews, unequal variance, as shown in the Residuals vs Fitted Plot, and violation of normality, seen in the normal QQ plot.

Transformation Of Price

Seeing the poor fit of the linear regression for the data, we tried to transform the data by taking the log of Price, square rooting Price, and taking 1/Price to see if the linear regression would improve. In order, the R-squared values for log transformation was (0.0135883), for square root transformation was (0.009), and for 1/Price transformation was (0.0169966). All of these transformations improved on the original linear regression, but are nowhere near high enough to properly explain Price using Review.Scores.Rating alone. However, the $\frac{1}{Price}$ transformation shouldn't be considered because it produced a negative estimate for Review.Scores.Rating, which is impossible in this context.

Next, we explored including interaction terms to see if it would improve the linear regression model. As an example, we included an interaction term between Property.Type and Neighbourhood with Review.Scores.Rating and got an R-squared value of (0.0684427). After applying the three transformations used before, the log transformation R-squared was (0.1644194), the square root transformation R-squared was (0.1644194), and the $\frac{1}{Price}$ transformation R-squared was (0.1738552). However, once again, the $\frac{1}{Price}$ transformation is unusable because it gives a negative estimate for Review.Scores.Rating, so from this point on we will not use the $\frac{1}{Price}$ transformation. The R-squared values after transformation with the inclusion of an interaction term has greatly increased from the original value of (0.0007) to (0.1644194). A possible reason for this increase is the normality assumption being met again for log transformation, as shown in the normal QQ plot.

AIC Using Interaction Terms

```
##                                                 Step Resid. Dev
## 1                                                    486793940
## 2                                  + Neighbourhood  464098240
## 3              + Review.Scores.Rating:Neighbourhood  455666213
## 4                                  + Property.Type  451108037
## 5                     + Property.Type:Neighbourhood  446737053
## 6                                + Number.Of.Reviews  446237501
## 7          + Review.Scores.Rating:Number.Of.Reviews  445837963
## 8              + Review.Scores.Rating:Property.Type  445226787
## 9 + Review.Scores.Rating:Property.Type:Neighbourhood  442413738
```

In an attempt to further increase the fit of the linear regression, we used the step AIC function in R to try to find the best model with interaction terms. Taking the model the step AIC function gave us, we fitted a linear regression with a log transformation to it and got a R-squared value of (0.1722728), which is slightly better than the (0.1644194) value from before.

Based on the low R-squared value of (0.1722728), we can say that although review scores play a small part in predicting price, it is not the only factor that does so. It is the combination of many factors that influence the price of an Airbnb. This makes sense because humans take many factors into account when deciding on the price for an Airbnb, and this decision making process is not something that can be easily defined using linear regression.

Question 3: Predicting Property Type Using KNN

Expectations

Property type includes 3 factors; Entire home/apartment , Private Room , Shared Room. After reviewing the first two questions, we expect that predicting property type is also difficult. Given that linear regression models have not produced satisfactory results, we are shifting to KNN models to make preditions. In order to

reduce the computational power to process all 30,000 rows, we are limiting the prediction to neighbourhood of Manhattan. ## Starting the KNN

```
##          Room.Type Beds Number.Of.Reviews Price Review.Scores.Rating
## 5     Private room    3                39   549                    96
## 7  Entire home/apt    1                 9   250                   100
## 9  Entire home/apt    2                95   270                    90
## 10 Entire home/apt    2                23   290                    96
## 12    Private room    2               120    59                    93
## 13    Private room    2                81    49                    91
```

Training Data and Test Data

```
train_Manhattan <- Manhattan[1:8185,2:5]
test_Manhattan <- Manhattan[8186:11693,2:5]
```

KNN Prediction Confusion Table

Now that Training data and Test data are prepared, we need to choose the an appropiate K for the KNN test. It is typical is use $\sqrt{Number\ of\ rows\ In\ Test}$ to choose an inital K. In this case $\sqrt{8185} = 90$, so K for the first KNN test is 90. After choosing an intial K, we procede to use the knn() function to conduct the test, then we create a confusion table to observe our success.

```
##
## predict_Manhattan Entire home/apt Private room Shared room
##   Entire home/apt            1899          316          11
##   Private room                238          934         110
##   Shared room                   0            0           0
```

Misclassification Error

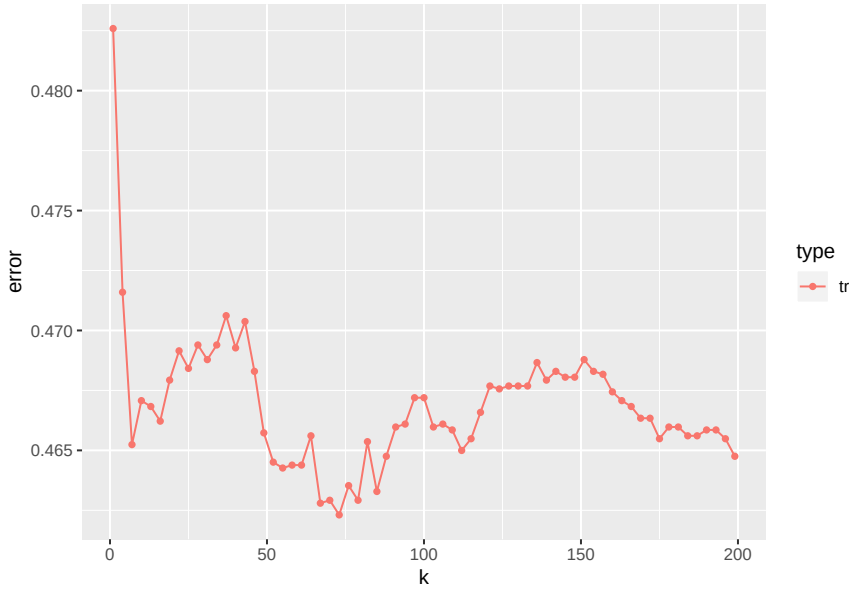
Finally we find the missclassifcation of the KNN test, which was 0.1924173. Interpretting the confusion table and error rate, KNN seems to be a relatively strong model to predict Room Type in Manhattan.

Engineering A More Appropiate K

However, we decide to conduct a more intensive test to choose an optimal K. $\sqrt{Number\ of\ rows\ In\ Test}$ gave a K of 90, but now we we test multiple K's ranging from 1 through 200 using a function written by Johann.

Plotting KNN Results of different K Values

After testing different K's ranging from 1-200, we plot their errors to visualize the outputs of each K.



KNN with engineered K and Misclassification

```
##
## predict_Manhattan Entire home/apt Private room Shared room
##   Entire home/apt            1909          317          12
##   Private room                228          933         109
##   Shared room                   0            0           0
```

Now that we know of an even more optimal K(73), we run another knn test, and output the confusion matrix. The results seem slightly better which a misclassification rate of 0.1898518. This is a slight improvement from 0.1924173.

Conclusion

Q1

For the first question, we were interested in discovering the relationship between price and the location of an Airbnb listing in New York. After ranking the boroughs from least to most expensive, we expected to see the highest prices for Airbnbs in Manhattan, which was ranked the most expensive borough. After running linear regression to find the relationship between price and borough, we discovered that neighborhood alone is not a significant predictor of price due to the low R^2. Though the p-values for all the boroughs were less than alpha .05, meaning a borough is a significant factor, R^2 is not a high enough to claim that neighborhood solely predicts the price of an Airbnb. To test which variable is a predictor of price, we narrow down the dataset to the borough of Manhattan and used the AIC method to determine which linear regression model will best predict price. A lower AIC indicates a more parsimonious model, relative to a model fit; therefore, we determine that predictor variables Beds, Room Type, Review Scores Rating, Property Type, and Number of Reviews optimally determine price in Manhattan. To continue, we expand the favorable model to the rest of the dataset. Finally, we decided to transform the price using $\frac{1}{Price}$ to both Manhattan and the entire dataset for the linear model. Using $\frac{1}{Price}$ gave satisfactory results, however, R^2 was still too low.

Q2

We are primarily answering the question whether reviews have any effect on price. Through linear regression, Review.Scores.Rating was not significant in explaining the variability in Price due to the low R-squared value of 0.003654. Some explanations for this poor fit include outliers and high leverage points, the negative estimate for Number.Of.Reviews, unequal variance, and violation of normality. As a result, we transformed the data by taking thelog$_{10}$ $(price)$,$\sqrt{Price}$, and taking $\frac{1}{Price}$ to see if the linear regression would improve; however, the R-squared values were still relatively low and one transformation produced a negative estimate, which does not make sense in this example. In conclusion, we used the step AIC function in R to fit a linear regression with a log transformation to the interaction terms (Number.Of.Reviews and Review.Scores.Rating) to get a R-squared value of 0.1723, which increased from 0.003654. Although review scores play a small part in predicting price, it is not the only factor that does so due to the low R squared value.

Q3

For the last question, we were interested in whether we could determine the property type for an Airbnb listing in Manhattan by using the k-Nearest Neighbors method since linear regression was not sufficient in producing satisfactory results. KNN works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label. As we increase the value of K, our predictions become more stable due to averaging, and thus, more likely to make more accurate predictions up to a certain point. We choose the correct K for our data by evaluating different Ks and picking the one that works best and that provides the most minimum misclassification error. We can safely conclude that K = (73) is the most optimal K for a KNN test, allowing us to predict Room Type in Manhattan with an error of 0.1898518.

Overall Conclusion

We hoped to gain useful insight from the parametric regression approach (linear regression) to determine the relationship between price and neighborhood as well as reviews and price. Since reduced model has higher accuracy than full model, we concluded that predictor variables Beds, Room Type, Review Scores Rating, Property Type, and Number of Reviews optimally determine price while taking 1/Price to improve the linear regression when testing for the significance between reviews and price. On the other hand, a non-parametric model (KNN) was used to obtain a model to predict the property type for an Airbnb listing in Manhattan. It is expected that KNN will have higher classification accuracy, since the size of the data set is sufficiently large, so we used it as a "database" in KNN. Thus, the above two models have their own strengths and weaknesses. For example, the linear regression provided the best linear regression with the lowest AIC value for price and neighborhood, but R^2 value was still relatively low for those same predictor variables, coming at a 0.2245. Although, KNN is useful for recognizing patterns and for estimating the likelihood of a specific property type, the user cannot gain any insight from the KNN model. We hope to find a model that combines the strengths of the two models that will include high classification accuracy and interpretability.

R Appendix

```
library(readxl)
library(ggplot2)
library(MASS)
library(dummies)
library(class)
library(mvtnorm)
library(tidyverse)
library(stringr)
library(plotly)
airbnb <- read_excel("airbnb.xlsx")
data<- data.frame(airbnb, na.rm= TRUE)
data1 = na.omit(data) #  <- data without NA's
# Our function is used to help Question 3.
# The function gets the KNN error rates and makes a table of the results.
get_knn_error_rates <- function(data, train_data, test_data, k){
    train_data_x <- train_data %>% select(-y)
    train_data_y <- train_data$y # turn into a vector
    # get predictions on training data
    knn_train_prediction <- knn(train=train_data,test=test_data,
                                cl=data[1:nrow(train_data),1], k=k) # set k

    tr_err <- mean(data[1:nrow(train_data),1] != knn_train_prediction)
    list(tr=tr_err) # training error rate
}
str(data)
table(data$Neighbourhood)
property.type <- table(data$Property.Type)
property.type[which(property.type>100)]
table(data$Room.Type)
length(unique(data$Zipcode)) # 189 unique zipcodes
dim(data)[1] # 30478
length(unique(data$Host.Id)) # 24421 unique host
ggplot(data = data1) +
  geom_point(mapping =aes(x = Number.Of.Reviews, y = Price,
                          color = Neighbourhood)) +
  ylim(c(0,5000)) + labs(x = "Number or Reviews", y = "Price")
ggplot(data = data1,mapping =aes(x = Review.Scores.Rating,
                                 y= Number.Of.Reviews,
                                 color = Neighbourhood)) +
  geom_point() +
  labs( x = "Review Scores Rating", y = "Number Of Reviews")
averages <- aggregate(data1$Price ~ data1$Neighbourhood, FUN=mean)
averages
table(data1$Neighbourhood)
lreg <- lm(Price ~ Neighbourhood, data = data1)
r_SLR_PandN<-summary(lreg)$r.squared  # r squared
ggplot(data = data1) +
 geom_count(aes(x = Beds, y = Price))+
    labs(x = "Number Of Bed", y = "Price Of AirBnb")

r_squared_LM_BandP <-summary(lm(data1$Price ~ data1$Beds))$r.squared  # r squared
Manhattan <- data1[which(data$Neighbourhood == "Manhattan"),]
```

```
Manhattan <- Manhattan[c(5,7,9,11,12,13)]
# Creating Model With No Point Estimaters
model_null = lm(Price~1, data=Manhattan)
# Creating a full model for the data
model_full = lm(Price~., data=Manhattan)
# AIC_Manhattan uses the Forward Selection Procedure
AIC_Manhhattan <- stepAIC(model_null,
                          scope=list(lower = model_null,
                                     upper = model_full),direction="both",k=2)
AIC_Manhhattan$anova
Manhattan_lm <- lm(Price ~ Beds + Room.Type +
                     Review.Scores.Rating +
                     Property.Type +
                     Number.Of.Reviews, data = Manhattan)
r_SLR_AIC<-summary(Manhattan_lm)$r.squared  # r squared

Manhattan1trans_lm <-lm(1/Price ~ Beds +
                          Room.Type +
                          Review.Scores.Rating +
                          Property.Type +
                          Number.Of.Reviews, data = Manhattan)
R_transformed_Manhattan <- summary(Manhattan1trans_lm)$r.squared
optimal_lm <- lm(Price ~ Beds + Room.Type +
                   Review.Scores.Rating +
                   Property.Type +
                   Number.Of.Reviews, data = data1)
r_whole_model_AIC <-summary(optimal_lm)$r.squared
data1_with_dummies <- data1[c(4,5,7)]
data1_with_dummies<-dummy.data.frame(data1_with_dummies)

# Merging dummy variables and regular variables into the same dataframe
columns_to_keep <- c(9,11,12,13)
data1_with_original_columns_to_keep <- data1[columns_to_keep]

total_data <- cbind.data.frame(data1_with_original_columns_to_keep,data1_with_dummies)

total_data_lm <- lm(1/Price ~., data = total_data)
r_total_data_lm <-summary(total_data_lm)$r.squared
lreg_02 = lm(Price ~ Number.Of.Reviews, data = data1)
r2_lreg_02 <-summary(lreg_02)$r.squared
lreg2 = lm(Price ~ Review.Scores.Rating, data = data1)
lreg3 = lm(Price ~ Review.Scores.Rating +
             Number.Of.Reviews, data = data1)
r2_lreg_02 <-summary(lreg_02)$r.squared
par(mfcol = c(2, 2))
plot(lreg_02)
lreg4 = lm(log(Price) ~ Review.Scores.Rating, data = data1)
r2_lreg4 <-summary(lreg4)$r.squared

lreg5 = lm(sqrt(Price) ~ Review.Scores.Rating, data = data1)
r2_lreg5 <-summary(lreg5)$r.squared

lreg6 = lm((1/Price) ~ Review.Scores.Rating, data = data1)
```

```
r2_lreg6 <-summary(lreg6)$r.squared
lreg7 = lm(Price ~ Review.Scores.Rating +
             Property.Type*Neighbourhood, data = data1)
r2_lreg7 <-summary(lreg7)$r.squared

lreg8 = lm((1/Price) ~ Review.Scores.Rating +
             Property.Type*Neighbourhood, data = data1)
r2_lreg8 <-summary(lreg8)$r.squared

lreg9 = lm(sqrt(Price) ~ Review.Scores.Rating +
             Property.Type*Neighbourhood, data = data1)
r2_lreg9 <-summary(lreg9)$r.squared

lreg10 = lm(log(Price) ~ Review.Scores.Rating +
              Property.Type*Neighbourhood, data = data1)
r2_lreg10 <-summary(lreg10)$r.squared
#used stepAIC to find the best model with interaction terms
AIC_Interactions <- stepAIC(lreg2,
      scope=~Review.Scores.Rating*Property.Type*Neighbourhood*Number.Of.Reviews,
      direction = "forward")
AIC_Interactions$anova[,c(1,5)]
#with transformation of price, R-squared goes up to 0.1456
lreg11<-lm(log(Price) ~ Review.Scores.Rating + Neighbourhood +
  Property.Type +
    Number.Of.Reviews +
    Review.Scores.Rating:Neighbourhood +
    Neighbourhood:Property.Type +
    Review.Scores.Rating:Number.Of.Reviews +
  Review.Scores.Rating:Property.Type +
    Review.Scores.Rating:Neighbourhood:Property.Type,
    data = data1)
r2_lreg11 <-summary(lreg11)$r.squared
data <- na.omit(data.frame(airbnb, na.rm= TRUE))
Manhattan <- data[which(data$Neighbourhood == "Manhattan"),]
Manhattan$Room.Type <- as.factor(Manhattan$Room.Type)
Manhattan <- Manhattan[c(7,9,11,12,13)]
head(Manhattan)
train_Manhattan <- Manhattan[1:8185,2:5]
test_Manhattan <- Manhattan[8186:11693,2:5]
predict_Manhattan <- knn(train_Manhattan,test_Manhattan,Manhattan[1:8185,1],k = 90)
predict_Manhattan_table <- table(predict_Manhattan,Manhattan[8186:11693,1])
predict_Manhattan_table
KNN_classification_error_01 <-
  1- sum(diag(predict_Manhattan_table))/sum(predict_Manhattan_table)
y=1
k_values <- c( seq(from=1, to=200, by=3))# values of K to use
num_k <- length(k_values)# number of k values to check

# initialize data frame to save error rates in
error_df <- tibble(k=rep(0, num_k),tr=rep(0, num_k))

# evaluate knn for a bunch of values of k
for(i in 1:num_k){
```

```
    k <- k_values[i]# fix k for this loop iteration
    errs <-
      get_knn_error_rates(Manhattan, train_Manhattan,test_Manhattan, k)
    error_df[i, 'k'] <- k  # store values in the data frame
    error_df[i, 'tr'] <- errs[['tr']]
}
error_df %>%
    gather(key='type', value='error', tr) %>%
    ggplot() +
    geom_point(aes(x=k, y=error, color=type, shape=type)) +
    geom_line(aes(x=k, y=error, color=type, linetype=type))
train_Manhattan <- Manhattan[1:8185,2:5]
test_Manhattan <- Manhattan[8186:11693,2:5]

k <- error_df$k[which.min(error_df$tr)]

predict_Manhattan <-
  knn(train_Manhattan,test_Manhattan, Manhattan[1:8185,1], k)
predict_Manhattan_Engineered_K <-
  table(predict_Manhattan, Manhattan[8186:11693,1])
predict_Manhattan_Engineered_K
# table(predict_Manhattan) <- Original Solutions

KNN_classification_error_02 <-
  1 -sum(diag(predict_Manhattan_Engineered_K))/sum(predict_Manhattan_Engineered_K)
# 0.8081528 <- original at K = 90
```