Analysis of Airbnb 2019 Data

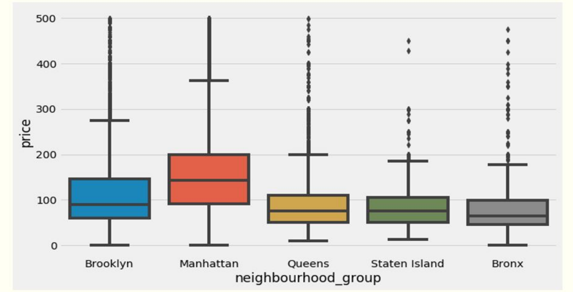
Chinta Gowtham Sai
Konduru Vivek Varma
Panchineni Prithvi
Kondrakunta Vivek and Michael Whalen

***Abstract*—Airbnb is an online marketplace for arranging or offering lodging, primarily home stays, or tourism experiences. It acts as a broker,receiving commissions from each booking. The data-set describes the listing activity and metrics of New York City on the Airbnb platform for the year 2019. New York City has been one of the hottest markets for Airbnb, with close to 50,000 listings as of December 2019. This means over 40 homes are being rented out per square km in New York City on Airbnb! One can perhaps attribute the success of Airbnb in NYC to the high rates charged by the hotels, which are primarily driven by the exorbitant rental prices in the city. This drives to do research and gain insights such as factors affecting the price of the property, the role of the property title in attracting customers and so on. This data-set has 16 features related to location, reviews, property details and prices.**

I. EXPLORATORY DATA ANALYSIS

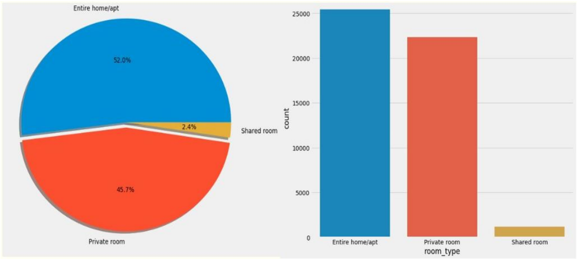
A. Boxplot

Boxplot of price vs neighborhood group is created to find the most expensive neighborhood group. For better visualization, instead of taking the whole price range, prices ranging from 0-500 are considered. From the boxplot, one could figure out that the most expensive neighborhood group is Manhattan, followed by Brooklyn and Queens.



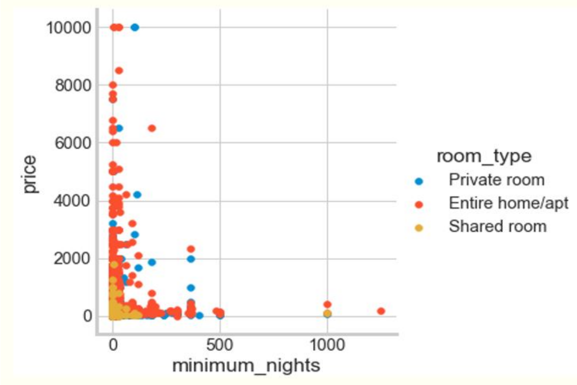
B. Pie chart and bar graph

Pie chart and Bar graph are created to find which room type has more number of listings and the count of each room type. They describe that the Entire home/apartment has more number of listings followed by private room type. Shared room type has minimal number of listings compared to the other two room types.



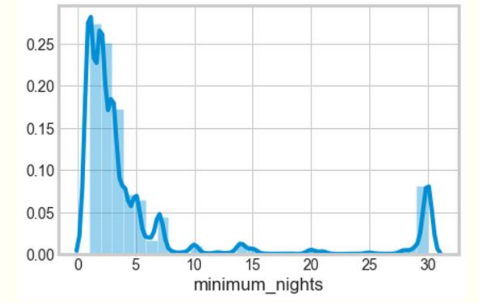
C. Scatter plot

A scatter plot of minimum nights vs price shows that property listings with less number of minimum nights have high prices.



D. Distribution plot

A distribution plot is created to find the exact range of minimum nights which have high prices. From the plot, it is found that the range is 1-3.



II. Regression

A. *Multiple Linear Regression*

The goal in applying multiple linear regression to the data was to see if there was a linear relationship between price , the response or dependent variable, and geographic location, room type, rating and review the explanatory variables or independent variables and to predict the price of a given property.Plotting the 3d surface plot between geographic location(latitude and longitude) and price can help visualize price variations across NYC. Top view of the plot helps us find out the price spikes can be found over a specific region(i.e over Manhattan). These plots indicate that there is no constant variation and causes some bias in the model. Initial fitted model has adjusted R^2 value 8.9. To overcome this problem, it was decided for this application to focus merely on the data where price is less than third quartile.

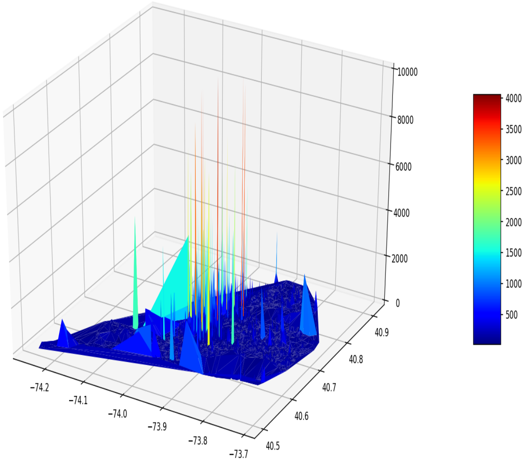


Fig. 1. 3D Surface Plot for price over NYC

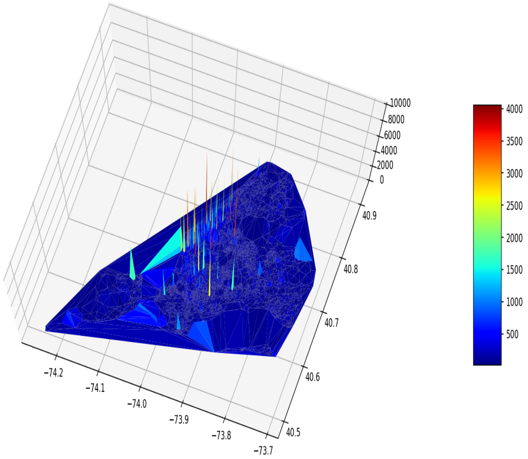


Fig. 2. Top View of 3D Surface Plot for price over NYC

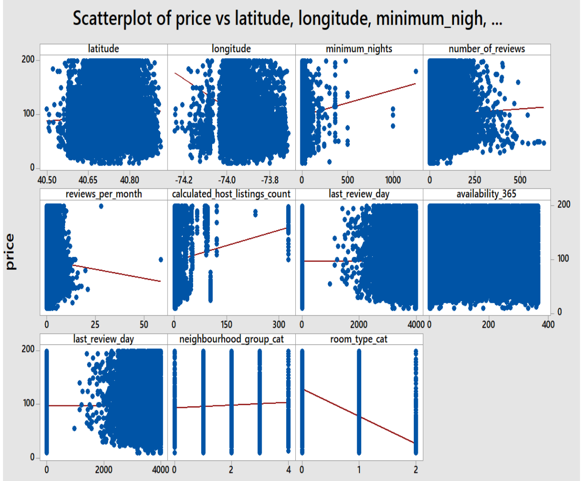


Fig. 3. Scatter plot for price variable against all independent variables

Before creating any models, it was necessary to inspect the variables and how they interact with the response variable. Scatter plots were created with price and all the independent variables.

Scatter plots with regression lines helps us in visualizing the strength of the relationship.Analyzing these graphs, it is clear that there is a linear relationship between independent and dependent variables. By conducting Global F-test we get p-value 0.000. This tells there is at least one independent variable that is linearly contributing to the price attribute.

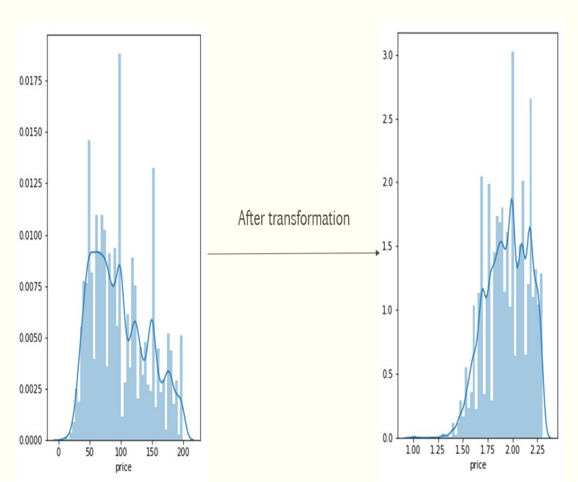


Fig. 4. Distribution plot before and after transformation

We applied log-transformation to our data to make it normally distributed.This transformation can help maintain constant variance among the residual errors.

Heat maps can help visualize the correlation or dependency among the attributes. Heat map is created between independent variables to check if there was any multicollinearity. It is

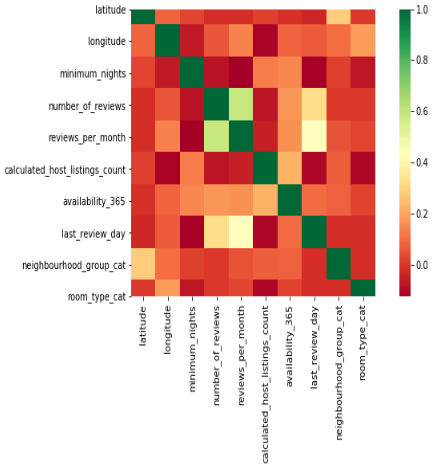


Fig. 5. Heat Map for independent variables

clear from the heat map that number-of-reviews and review-per-month are correlated. Reviews-per-month is number-of-reviews divided by total number of days for a given month.

It's obvious that number-of-reviews does not provide any additional information in presence of reviews-per-month. Further analysis shows p-value of reviews-per-month in presence of number-of-review is very high around 0.865. Using hypothesis testing we can then remove reviews-per-month if p-value is greater than 0.05 which in this case is true. So we removed reviews-per-month as this does not contribute much to the model.

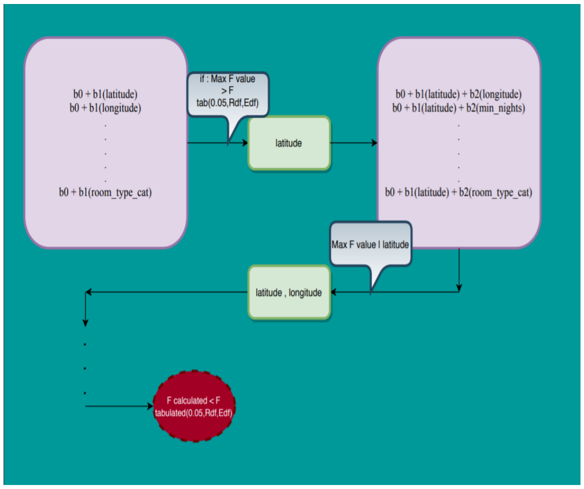

b0 + b1(latitude)
b0 + b1(longitude)
b0 + b1(room_type_cat)
if : Max F value > F tab(0.05,Rdf,Edf)
latitude
b0 + b1(latitude) + b2(longitude)
b0 + b1(latitude) + b2(min_nights)
b0 + b1(latitude) + b2(room_type_cat)
Max F value I latitude
latitude , longitude
F calculated < F tabulated(0.05,Rdf,Edf)

Fig. 6. Flow chart representing forward selection

To see if there was a better model, forward selection was performed on the most recent model. Forward selection begins with only an intercept. It performs nCk regressions, where k is the total number of regressors and chooses the regressor that best fits the model. As the model continues to improve we continue the process and add in one variable at a time and test at each step. Once the model no longer improves with adding more variables, the process stops. Using forward selection for our model we got to know calculated-host-listings-count has been removed. This attribute has p-value of 0.636 and does not contribute much to the model.

With the initial model we got an adjusted R^2 value of 8.9. But then considering the model with price less than third quartile we got R^2 value for train dataset as 71.27 and for test dataset adjusted R^2 value was 69.12.

From this analysis, it was hypothesized that perhaps better models would be created from considering this data where price was below third quartile.

III. Classification

Classification is supervised machine learning technique ,which categorizes some unknown items into a discrete set of categories or classes .It has a target attribute, the target variable in classification is a categorical variable with discrete values .by giving a set of training data points along with the target labels classification determines the class label for an unlabeled test case. Now fo this project I am using Random Forest classifier algorithm for better Results.

A. *Random Forest Classifier*

Random Forest classifier is an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees. Random forest is a meta estimator that fits a number of decision tree classifiers on various sub samples of the data set and uses averaging to improving the mid predictive accuracy and control overfitting.

Now in this project I have done classification on room prices in which category they fall into. For this I have taken the statistics of the data which appears as follows.

Feature	Price
count	48885
mean	152.72
std	240.17
min	0
25%	69
50%	106
75%	175
max	10000

Now from the derived statistics I categorized the room prices into three categories i.e. High,Low,Medium.

Range(in USD)	Category
0-69	Low
69-175	Medium
175-10000	High

Now I have divided the data into training data and testing data 1)Train data-80% 2)Test data-20%

Now after this the model is trained and the accuracy obtained on the first run is 69%.Now the main task is to improve the accuracy of model for that I have done hyper parameter tuning.

B. Hyper parameter tuning

In machine learning hyperparameter optimization or tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm .A hyperparameter is a parameter whose value user to control the learning process, by contrast the values of the other parameters .The same kind of machine learning model can require different constraints weights or learning rates to their generalize different data patterns. These measures are called hyper parameters and to be tuned so that the model can optimally solve the machine learning problem. The parameters tuned are – 1)Min samples leaf 2)n estimators 3)max features

C. Hyper Parameter Tuning for minimum samples leaf

The minimum number of samples required to be at a leaf node. A split point at any depth will only be considered if it leaves at least minimum samples live training samples in each of the left and right branches. This may have effect of smoothing the model.

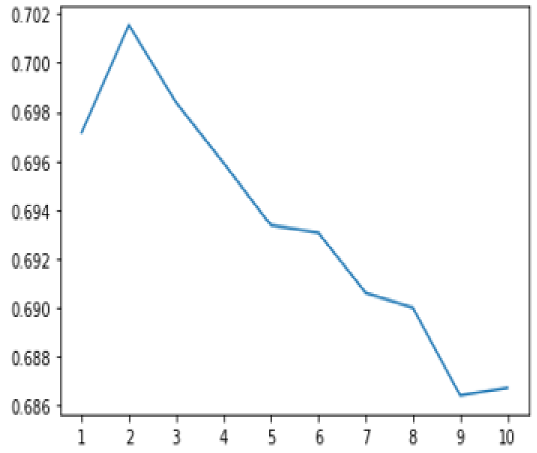


Fig. 7. Flow chart representing forward selection

min samples leaf = 2 is the optimum for this parameter.

D. Hyper Parameter Tuning for n estimators

The number of estimators to run in parallel fit, predict decision path and apply or all parallelized over the trees.

The optimum for ’n estimators’ is 1000.

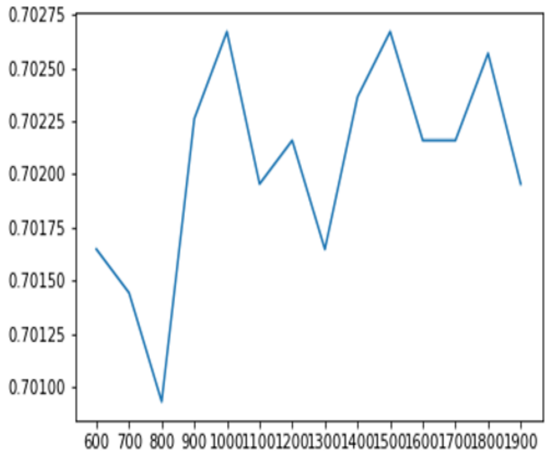


Fig. 8. Flow chart representing forward selection

E. Hyper Parameter Tuning for max $_{f}eatures$

The number of features to consider when looking for best split . The search for a split does not stop until at least one valid partition of the node samples is found, even if it requires to effectively inspect more than max features.

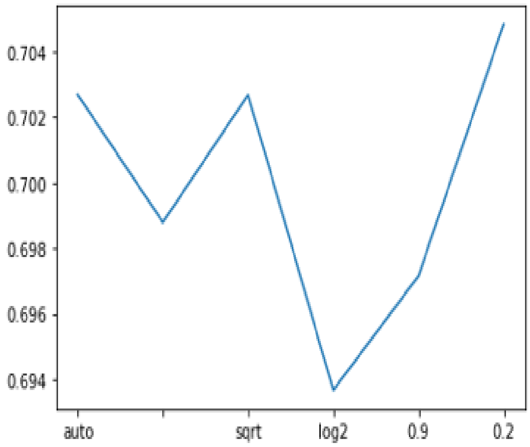


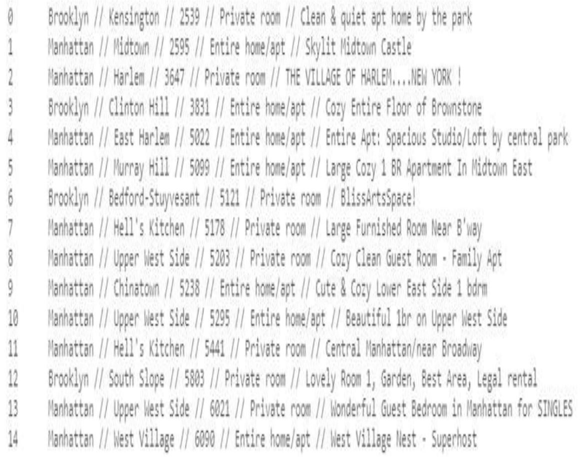
Fig. 9. Flow chart representing forward selection

The optimum for ’max features’ is 0.2.

Finally, after performing hyper parameter tuning on the model the accuracy is increased by 2% i.e 71%.

IV. Recommendation system

For the content based recommendation system we have used the main NYC data file and created a separate file similar to a .csv, except it is separated by “ // “. From left to right we have used: “id”, “neighborhood-group”, “neighborhood”, “room type”, and “description”. We can change what our dataset will focus on by simply adjusting column names within the code (for example: if we wanted to compare “prices”, “id” , “neighborhood”, and “room type” rather than what is listed above we would simply change variables within our code.



```
0     Brooklyn // Kensington // 2539 // Private room // Clean & quiet apt home by the park
1     Manhattan // Midtown // 2595 // Entire home/apt // Skylit Midtown Castle
2     Manhattan // Harlem // 3647 // Private room // THE VILLAGE OF HARLEM....NEW YORK !
3     Brooklyn // Clinton Hill // 3831 // Entire home/apt // Cozy Entire Floor of Brownstone
4     Manhattan // East Harlem // 5022 // Entire home/apt // Entire Apt: Spacious Studio/Loft by central park
5     Manhattan // Murray Hill // 5099 // Entire home/apt // Large Cozy 1 BR Apartment In Midtown East
6     Brooklyn // Bedford-Stuyvesant // 5121 // Private room // BlissArtsSpace!
7     Manhattan // Hell's Kitchen // 5178 // Private room // Large Furnished Room Near B'way
8     Manhattan // Upper West Side // 5203 // Private room // Cozy Clean Guest Room - Family Apt
9     Manhattan // Chinatown // 5238 // Entire home/apt // Cute & Cozy Lower East Side 1 bdrm
10    Manhattan // Upper West Side // 5295 // Entire home/apt // Beautiful 1br on Upper West Side
11    Manhattan // Hell's Kitchen // 5441 // Private room // Central Manhattan/near Broadway
12    Brooklyn // South Slope // 5803 // Private room // Lovely Room 1, Garden, Best Area, Legal rental
13    Manhattan // Upper West Side // 6021 // Private room // Wonderful Guest Bedroom in Manhattan for SINGLES
14    Manhattan // West Village // 6090 // Entire home/apt // West Village Nest - Superhost
```

Fig. 10. NYC data file seperated by //

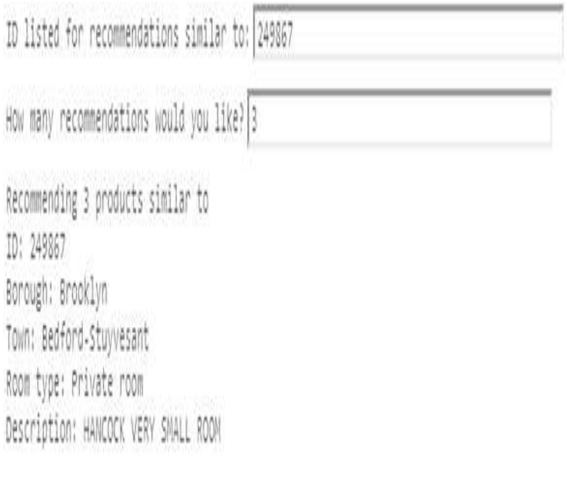
This separate dataset will be used to create a TfidfVectorizer, the vectorizer uses the formula:

$$w(i,j) = (tf)(i,j) * log(N/(df)(i))$$

in order to find similar contents based on phrases other than the stop-words. After creating the TfidfVectorizer we fit and transform the data in order to calculate its cosine similarities and its formula:

$$K(X,Y) = < X, Y > /(||X|| * ||Y||)$$

based on similar contents or based on the prior or in our case the user inputted “id”.



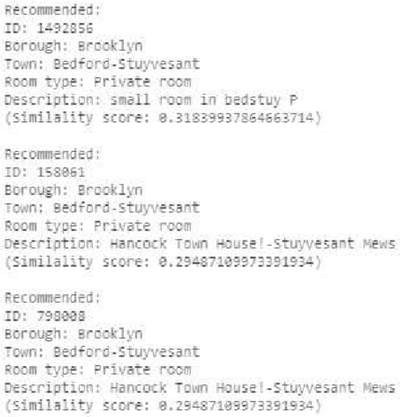
```
ID listed for recommendations similar to: 249867

How many recommendations would you like? 3

Recommending 3 products similar to
ID: 249867
Borough: Brooklyn
Town: Bedford-Stuyvesant
Room type: Private room
Description: HANCOCK VERY SMALL ROOM
```

Fig. 11. Recommendation system prompt

As seen above, our recommendation systems prompt the user to enter a listed “id” in the dataset (could use prior data for this if our dataset had this) and a prompt for the number of recommendations. Our recommendation system is limited to list up to 100 recommendations. The user has a choice to output a smaller size of recommendations in order to output its best result, but also has the option to output more results but with a lesser similarity score.



```
Recommended:
ID: 1492856
Borough: Brooklyn
Town: Bedford-Stuyvesant
Room type: Private room
Description: small room in bedstuy P
(Similality score: 0.31839937864663714)

Recommended:
ID: 158061
Borough: Brooklyn
Town: Bedford-Stuyvesant
Room type: Private room
Description: Hancock Town House!-Stuyvesant Mews
(Similality score: 0.29487109973391934)

Recommended:
ID: 798008
Borough: Brooklyn
Town: Bedford-Stuyvesant
Room type: Private room
Description: Hancock Town House!-Stuyvesant Mews
(Similality score: 0.29487109973391934)
```

Fig. 12. Output based on user input

Based on the input in the image prior to this our output is based on the (prior data) user input in this case. Similarity scores closer to “1” represents the most similar property based on the user input. When calculating the similarity score each column listed in our TfidfVectorizer dataset is taken into account. In our case we used: “id”, “neighborhood-group”, “neighborhood”, “room type”, and “description”.

V. Clustering

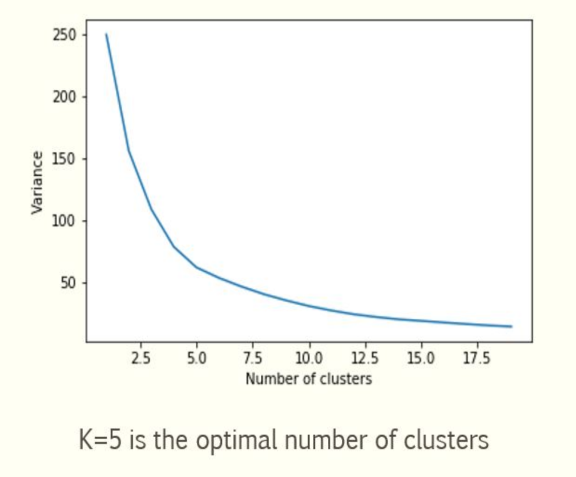
Clustering is an unsupervised machine learning technique that groups similar data. To find out which place has the higher number of listings I used clustering. To achieve this, latitude and longitude coordinates of the property listings are used. K-means clustering and hierarchical clustering are used to find out which one of them best suits the data. In this dataset, there are no ground truth values to verify the purity of the cluster. Therefore intrinsic cluster evaluation is used to find the performance of the clustering algorithms used. There are two parameters considered for the evaluation. The first is compactness. It is a measure of how close the data points are inside a cluster. It is calculated by measuring the distance of each data point to its centroid in a cluster. The second parameter is the inter-cluster distance. It is calculated by measuring the distance from the centroid of one cluster to its nearest cluster centroid.

Calinski-Harabasz index also known as the Variance Ratio Criterion is used to evaluate the model, where a higher Calinski-Harabasz score relates to a model with better-defined clusters. The index is the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters (where dispersion is defined as the sum of distances squared).

Davies-Bouldin index is another method used to evaluate the model.This index signifies the average similarity between clusters, where the similarity is a measure that compares the distance between clusters with the size of the clusters themselves.Zero is the lowest possible score. Values closer to zero indicate a better partition.

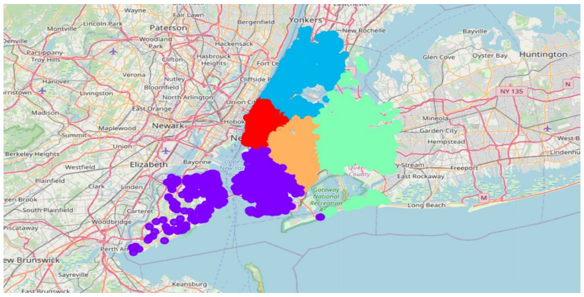
Algorithm	Davies-Bouldin index	Calinski-Harabasz index
K-means	0.80	36974
Hierarchical	0.85	33730

K means clustering outperformed hierarchical clustering in both the scores. Therefore K means is the best fit for this dataset. Using Elbow method it is found that k=5 is the optimal number of clusters.



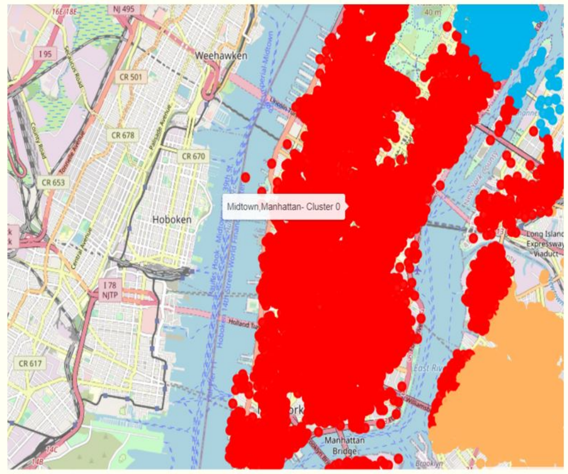
K=5 is the optimal number of clusters

With the help of latitude and longitude coordinates of the property listings, the clusters are visualized on the map.



It looks like the red cluster is the smallest of all. But when we see the count of each cluster it is clear that the red cluster has the highest number of listings. The reason it looks small is that all the property listings are close to each other when compared to that of properties in other clusters.

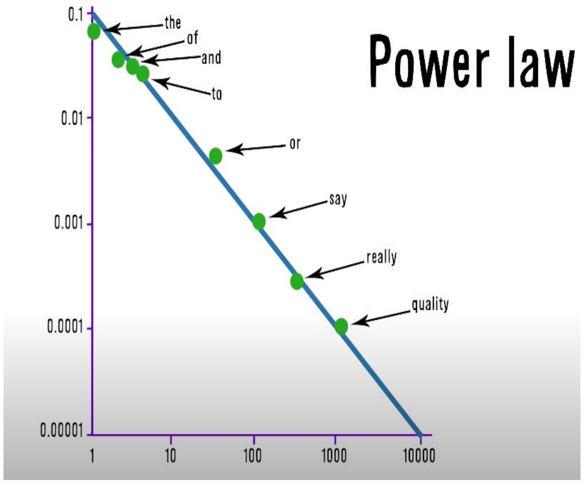
Colour	Count	Cluster
Red	14729	0
Purple	6373	1
Blue	10678	2
Green	2497	3
Orange	14618	4



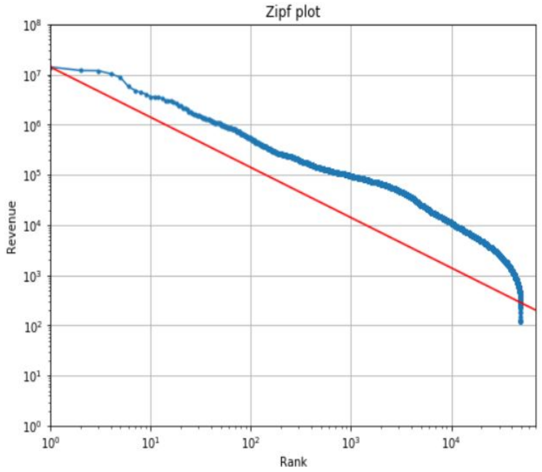
The majority of the property listings in the red cluster belong to Manhattan. By performing clustering on the location of the property listings we found that being the biggest tourist spot of New York City and Hudson river flowing across its border, Manhattan has the highest number of property listings. The next biggest cluster is orange and Brooklyn cements its position on the top.

VI. Zipf's Law

Zipf's law was proposed by George Zipf who was a linguist at Harvard University. It states that the frequency of events is inversely proportional to their rank. This law is best explained with the most common words used in the English language. 'The' is the most frequently used word in English language followed by 'of' in the second position. When the words are ranked based on the frequency, the second most used word will appear about half as often as the first most used word and the third most used word will appear one third as often as the first most used word and so on. Frequency and rank, when plotted on a log-log graph, follows a straight line. This phenomenon is called Zipf's law.



Zipf's law unknowingly exists in many things in this world. Some of the examples are City population, solar flare intensities and the rate at which we forget. All these events, when plotted on a log-log graph, follows a straight line. Even in this data, there is a parameter that satisfies Zipf's law.



Revenue generated from each listing is calculated by assuming each property has one booking per month. The properties are ranked based on the revenue. Revenue and rank, when plotted on a log-log graph, follows a path almost imitating a straight line. This proves that revenue generated by the properties satisfies Zipf's law.

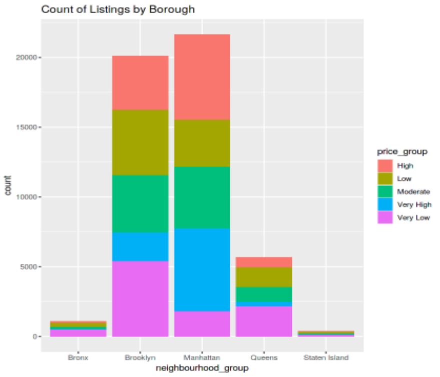
VII. Text Analysis

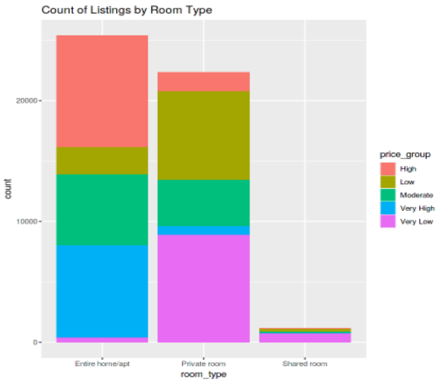
For doing text Analysis on the data we are using Natural Language Processing model

NLP: Natural Language Processing, or NLP for short, is broadly defined as the automatic manipulation of natural language, like speech and text, by software. NLP analyzes text and allows machines to understand how we speak. It considers the hierarchical structure of language and performs tasks like correcting the grammar, converting speech to text, and translating between languages. NLP help analysts to turn unstructured text into usable data and insights. In python it's built under the NLTK library. With Natural Language Processing, we carry out five different tasks they are Lexical Analysis, Syntactic Analysis, Semantic Analysis, Discourse Integration, and Pragmatic Analysis.

For doing text analysis on property names we have considered three Key Performance Indicators (KPIs) namely Number of reviews, price, and minimum nights. Using these KPIs we are calculating the Total Net revenue by taking the product of three KPIs.

From the property prices, we are grouping the properties into price group categorical variables by 20th percentile quantiles. They are very high, high, moderate, low, and very low. The below are for some Exploratory data analysis on the data with price groups.



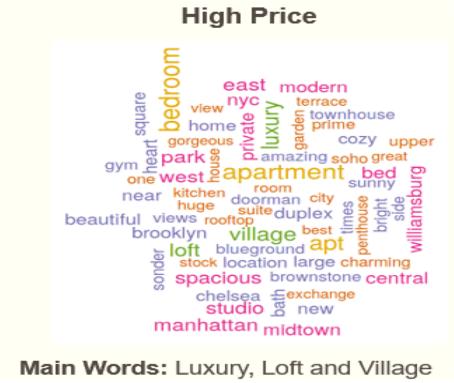


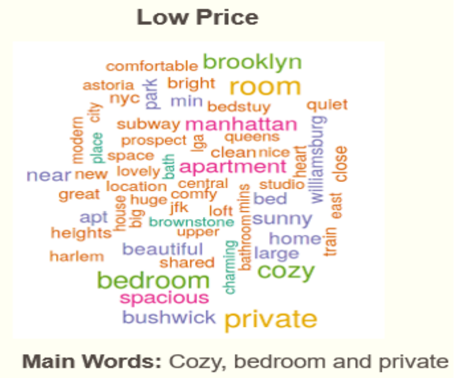
It seems that

- Manhattan contains the most amount of listings and it is the most expensive.
- Next, Brooklyn appears to be next popular; however, with a more reasonable distribution of price listings. High - Very High seem to only take up 20 percent of the population.
- Queens has only 5000 listings and appears very cheap.

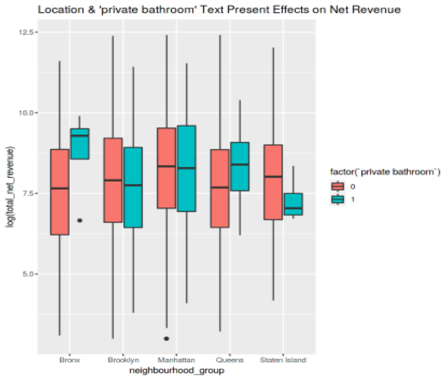
Lastly, the Bronx State Island do not appear very popular for Airbnb.

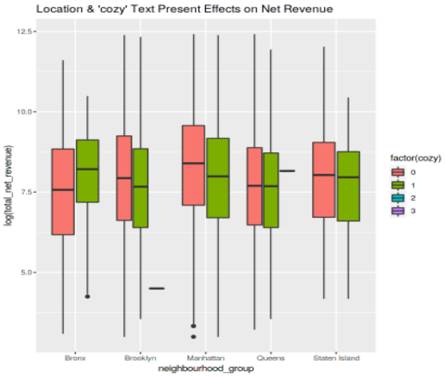
By applying NLP model on the property titles which are coming under very high and very low categories, we can do some of cleaning such as removing punctuations, Characters into lower case, removing numbers, removing English stop words, etc., The below are the word clouds for Very high and very low price group categories respectively.

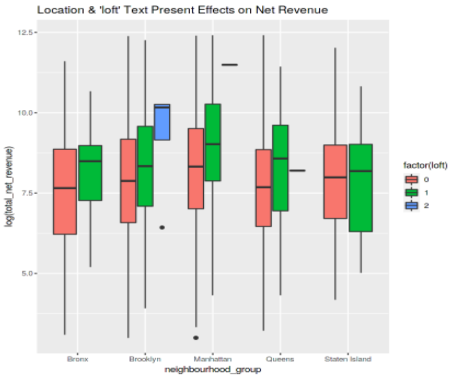




And the below plots suggests that how some of the key words in the titles have an impact on the net revenue in the neighborhoods.





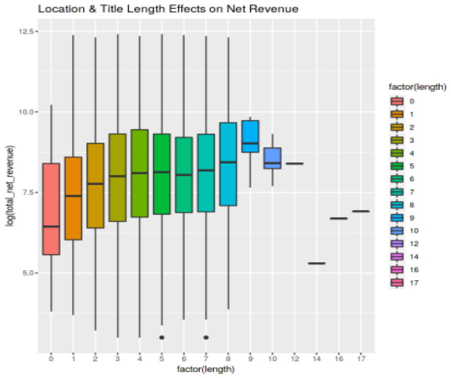


- These plots suggest that in Queens (discrediting the Bronx distribution due to sample size), those that list a title with the words ’private bathroom’ appear to bring in a higher Net Revenue.

- It also suggests that in the Bronx, those that list a title with the word ’cozy’ appear to bring in a higher Net Revenue; while in Manhattan, listings with ’cozy’ are not performing as well.

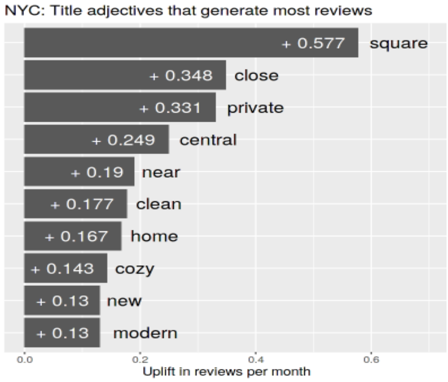
- Perhaps those that are staying in Manhattan don’t want a ’cozy’ spot - but a luxurious ’loft’.

From the above plot we can say,if a host lease a place on Airbnb in Manhattan or Brooklyn, it’s better to put ’loft’ in the title!



From the above plot, we can interpret that when listing your Airbnb property, make sure it contains at least 4 meaningful words!

Lastly, by applying the Generalized Linear Model on the titles, we can find which words in the titles are impacting to generate more reviews based on p-values. Looks like the words Square, Close, Private, and Central in the titles can generate more reviews.



These keywords can be neatly broken down into three subgroups:
1. Location: [Times] square, close [to a landmark, a prominent station, etc.], central and near (similar to close)
2. Objective qualities: clean, new and modern
3. Emotional qualities: private, home and cozy

Major Takeaways:
• The text that you put in your property title DOES matter. It varies across what region you are located in. It is important to include specific text, especially if your property contains it.

Ex: if you're near Central Park - put that in the title!

• If you're a high rated Host, and your property is also highly rated, you are most likely going to succeed in obtaining a larger net revenue!

VIII. Contributions

1) Chinta Gowtham Sai :
   - Exploratory Data Analysis
   - Clustering
   - Zipf's Law
2) Vivek Varma Konduru :
   - Exploratory Data Analysis
   - Text Analysis
3) Panchineni Prithvi :
   - Exploratory Data Analysis
   - Multiple Linear regression
4) Vivek Kondrakunta :
   - Exploratory Data Analysis
   - Classification
5) Michael Whalen :
   - Exploratory Data Analysis
   - Recommendation system

IX. References

1) Clustering :
   - https://scikit-learn.org/stable/modules/clustering.htmlclustering-performance-evaluation
2) Zipf's Law :
   - https://www.youtube.com/watch?v=fCn8zs912OEt=227s
3) Recommendation system :
   - https://scikit-learn.org/stable/modules/generated/sklearn.metrics. similarity.html
   - https://scikit-learn.org/stable/modules/generated/sklearn.feature-extraction.text.TfidfVectorizer.html
4) Classification :
   - https://scikit-learn.org/stable/modules/ensemble.htmlforests-of-randomized-trees
5) Regression :
   - https://online.stat.psu.edu/stat501/
   - http://home.iitk.ac.in/ shalab/course5.htm
6) Text Analysis :
   - https://machinelearningmastery.com/natural-language-processing/
   - https://medium.com/@rinu.gour123/nlp-tutorial-ai-with-python-natural-language-processing-ed81fdb3f0a3