Airbnb Data Analysis Report

Let's save customers some money

BY:
ARPIT SHARMA
SANMAN YADAV

Contents

Introduction

Airbnb began in 2008 when two designers who had space to share hosted three travelers looking for a place to stay. Now, millions of hosts and travelers choose to create a free Airbnb account so they can list their space and book unique accommodations anywhere in the world. And Airbnb experience hosts share their passions and interests with both travelers and locals.

Airbnb helps make sharing easy, enjoyable, and safe. They verify personal profiles and listings, maintain a smart messaging system so hosts and guests can communicate with certainty, and manage a trusted platform to collect and transfer payments. Airbnb is an online marketplace that connects people who want to rent out their homes with people who are looking for accommodations in that locale. It currently covers more than 81,000 cities and 191 countries worldwide. The company's name comes from "air mattress B&B."

For hosts, participating in Airbnb is a way to earn some income from their property, but with the risk that the guest might do damage to it. For guests, the advantage can be relatively inexpensive accommodations, but with the risk that the property won't be as appealing as the listing made it seem.

There are many advantages and some disadvantages of Airbnb. We have listed some of the important factors below over which we will determine our business question and analyze the Airbnb data.

- Wide Selection

Airbnb hosts list many kinds of properties—single rooms, a suite of rooms, apartments, moored yachts, houseboats, entire houses, even a castle—on the Airbnb website.

- Free Listings

Hosts don't have to pay to list their properties. Listings can include written descriptions, photographs with captions, and a user profile where potential guests can get to know a bit about the hosts.

- Hosts Can Set Their Own Price

It's up to each host to decide how much to charge per night, per week or per month.

- Customizable Searches

Guests can search the Airbnb database—not only by date and location, but by price, type of property, amenities, and the language of the host. They can also add keywords (such as "close to the Louvre") to further narrow their search.

- Additional Services

In recent years Airbnb has expanded its offerings to include experiences and restaurants. Besides a listing of available accommodations for the dates they plan to travel, people searching by location will see a list of experiences, such as classes and sightseeing, offered by local Airbnb hosts. Restaurant listings also include reviews from Airbnb hosts.

- Protections for Guests and Hosts

As a protection for guests, Airbnb holds the guest's payment for 24 hours after check-in before releasing the funds to the host.

- What You See May Not Be What You Get

Booking accommodations with Airbnb is not like booking a room with a major hotel chain, where you have a reasonable assurance that the property will be as advertised. Individual hosts create their own listings, and some may be more honest than others. However, previous guests often post comments about their experiences, which can provide a more objective view.

- Potential Damage

Probably the biggest risk for hosts is that their property will be damaged. While most stays go without incident, there are stories of entire houses being trashed by dozens of party-goers when the Airbnb hosts thought they were renting to a quiet family. Airbnb's Host Guarantee program, described above, provides some assurance, but it may not cover everything, such as cash, rare artwork, jewelry, and pets. Hosts whose homes are damaged may also experience considerable inconvenience.

- Added Fees

Airbnb imposes a number of additional fees (as, of course, do hotels and other lodging providers). Guests pay a guest service fee of 0% to 20% on top of the reservation fee, to cover Airbnb's customer support and other services. Prices display in the currency the user selects, provided Airbnb supports it. Banks or credit card issuers may add fees if applicable.

Objective and Business Questions

The main aim of this project is to dig deep and analyze how we can benefit the consumers, host and even Airbnb to provide a better service.

- Consumers – after our analysis they can compare between different Airbnb rooms and nearby hotels to select the best available option.
- Airbnb – Competitive pricing is most important in today's business. We can analyze where Airbnb costumers are having the most issues with and they can focus to improve their service in that aspect
- Host – How can they improve their houses for a better customer service. Also beneficial for new hosts who can have a head start about the price point where their house might land
- Hotels – With growing popularity of Airbnb, Hotels can have competitive pricing to lure customers into their hotels and add more additional services if they feel they are lacking in some department.

We will use R and WEKA software to run data mining algorithms to understand the relationship between attributes and the target variable.

Data Description

In this project we take data of Airbnb listings in the US and try to predict the price of stay in that listing. The data includes 84411 listings and 29 columns - including log_price, what we are trying to predict.

Some columns will not be used as features, such as ID and thumbnail URL, so we are left with 26 columns to process and consider as features.

There was no information about this data so we assume that since all the listings are in the us, the price (or log_price) that we are trying to predict is the general pricing per 1 night stay of the listing, in USD, not for specific dates/seasons and not including additional fees, i.e. cleaning and Airbnb service fees.

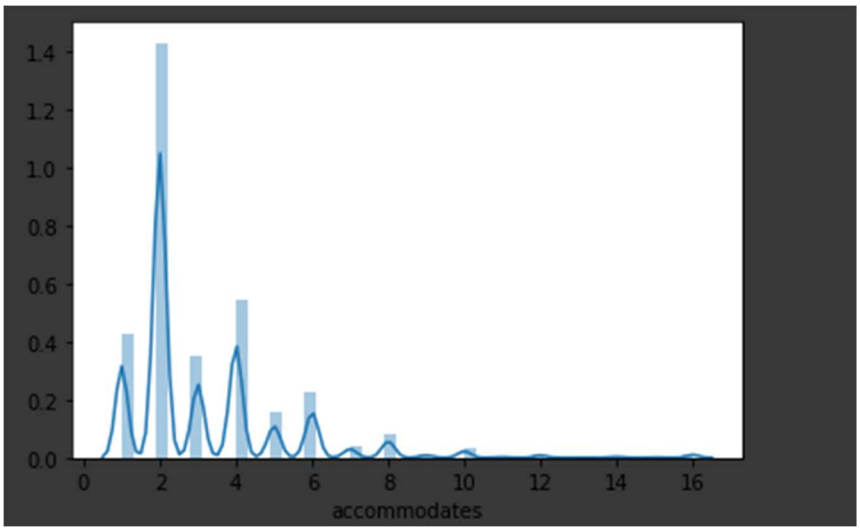
The data is for 6 cities across the US: NYC, LA, San Francisco, Washington DC, Boston and Chicago. The table below displays some of the columns that are present in the data and their description.

Name	Description
Log Price	Log price of each booking
Property type	Apartment/ House / condo
Room type	Entire room or entire house etc
Amenities	Wireless Internet, Air conditioning, Kitchen, Heating
Cancelation Policy	Strict, Moderate etc.
Cleaning Fee	Cleaning fee at time of booking
City	Different cities and areas
Host response rate	How fast does the host reply to issues
Number of reviews	How good is the review of the listed house

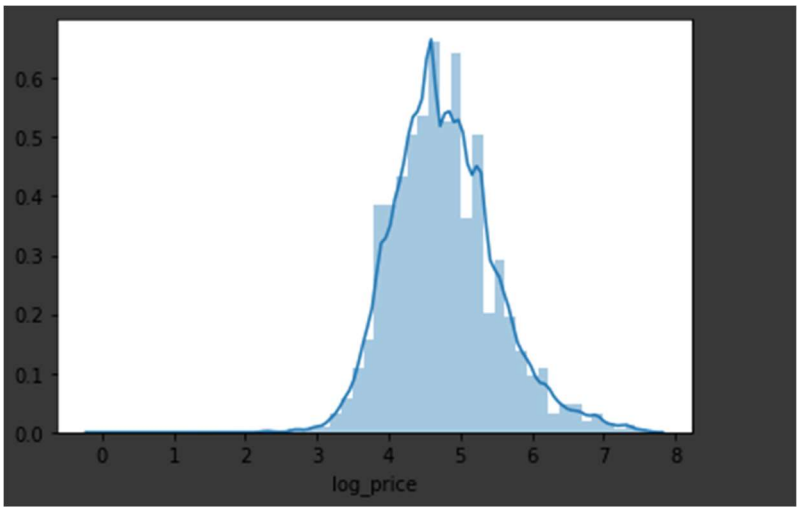
Data Preprocessing and Preparation

While looking at the data, one thing that catches eye is that, there are some columns that have missing data. Some of those are: Cleaning fee, number of beds, number of rooms etc. Some of the data that was missing was continuous and categorical. So, when it came to categorical data, we took the mean of the column (by dropping the NA's) and then used the mean value to fill the rows. For categorical data, we used the mode of the column of the column to fill the data. We also tried to fill the categorical data by looking at the cities and then taking the mean. In both the cases the accuracy wasn't affected much.

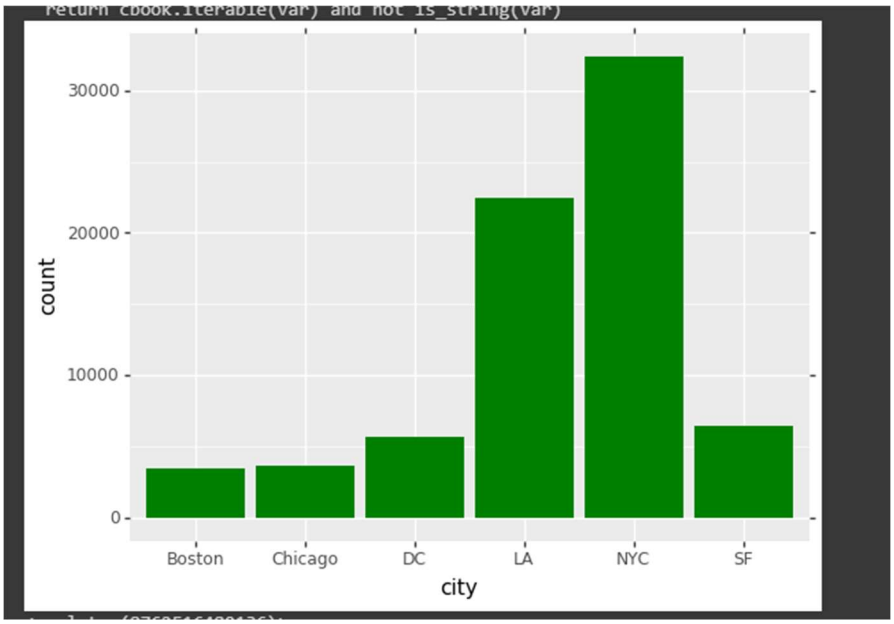
EDA:



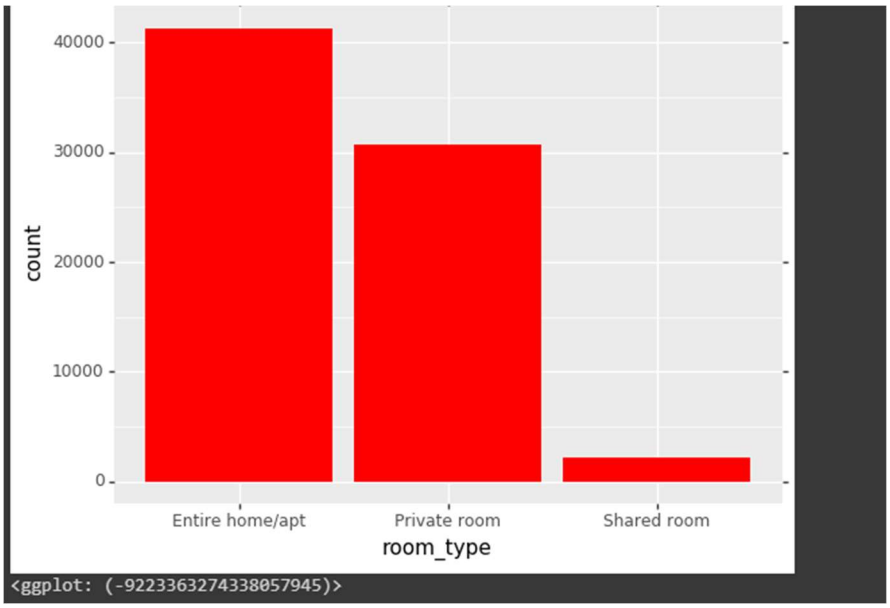
This figure shows, the distribution of accommodations across the bookings. 2 accommodations being the highest and followed by 4 and 1.



This figure shows, the distribution of log price of the bookings across the bookings. As seen in the distribution, most of the bookings range from 4 to 5.5.



This figure shows, the distribution of bookings across the different cities. NYC getting the crown of the most bookings and LA comes in second. This requires further analysis to see if number of listings is high in NYC or in LA. If there are more listings in NYC, then the distribution is true.



This figure shows, the distribution of different room type. It looks like, most of the people who book rooms using AirBnb, prefer getting an entire home or apartment.

property_type	[Apartment, House, Condominium, Loft, Townhouse, Hostel, Guest suite, Bed & Breakfast, Bungalow, Guesthouse, Dorm, Other, Camper/RV, Villa, Boutique hotel, Timeshare, In-law, Boat, Serviced apartment, Castle, Cabin, Treehouse, Tipi, Vacation home, Tent, Hut, Casa particular, Chalet, Yurt, Earth House, Parking Space, Train, Cave, Lighthouse, Island]
room_type	[Entire home/apt, Private room, Shared room]
bed_type	[Real Bed, Futon, Pull-out Sofa, Couch, Airbed]
cancellation_policy	[strict, moderate, flexible, super_strict_30, super_strict_60]
cleaning_fee	[True, False]
city	[NYC, SF, DC, LA, Chicago, Boston]
host_has_profile_pic	[t, nan, f]
host_identity_verified	[t, f, nan]
host_response_rate	[nan, 100%, 71%, 68%, 67%, 83%, 50%, 90%, 86%, 92%, 82%, 80%, 89%, 93%, 99%, 0%, 88%, 96%, 70%, 94%, 91%, 25%, 95%, 98%, 62%, 29%, 33%, 81%, 63%, 38%, 60%, 79%, 78%, 75%, 65%, 97%, 87%, 40%, 54%, 53%, 58%, 76%, 30%, 64%, 17%, 20%, 77%, 73%, 41%, 59%, 57%, 85%, 56%, 42%, 44%, 35%, 14%, 74%, 27%, 10%, 84%, 6%, 72%, 36%, 55%, 43%, 13%, 39%, 46%, 26%, 61%, 52%, 23%, 22%, 69%, 66%, 15%, 11%, 31%, 21%, 47%]
instant_bookable	[f, t]
neighbourhood	[Brooklyn Heights, Hell's Kitchen, Harlem, Lower Haight, Columbia Heights, Noe Valley, nan, Downtown, Richmond District, Alphabet City, Hermosa Beach, Torrance, U Street Corridor, Humboldt Park, Wicker Park, South Boston, Lower East Side, Flatbush, Sherman Oaks, East Flatbush, Valley Glen, Dupont Circle, Jamaica, Forest Hills, Murray Hill, Lefferts Garden, Mid-Wilshire, Venice, West Hollywood, Brownsville, Williamsburg, East Village, South Loop/Printers Row, Westlake, Hollywood Hills, Upper East Side, Bushwick, Bedford-Stuyvesant, Pilsen, Chelsea, Sunnyside, Greenwich Village, Washington Heights, Pasadena, Potrero Hill, Brookland, Los Feliz, Hollywood, Midtown East, Glendale, Park Slope, Arcadia, West Village, Astoria, Portola, Burbank, East Harlem, Silver Lake, Hillbrook, Shaw, Hillcrest, Morningside Heights, Tribeca, Studio City, Western Addition/NOPA, Echo Park, Financial District, Lakeview, Gramercy Park, Mission District, Kingman Park, Sunset Park, Upper West Side, Greenpoint, Highland Park, Cleveland Park, Prospect Heights, Glover Park, Gravesend, Jamaica Plain, Beacon Hill, SoMa, Flushing, Van Nuys, Del Rey, Midtown, Soho, Mission Hill, Roosevelt Island, Marina Del Rey, West Roxbury, Streeterville, Bernal Heights, Soundview, North Beach, Belmont Cragin, Crown Heights, Gowanus, Boerum Hill, Greenwood Heights, ...]

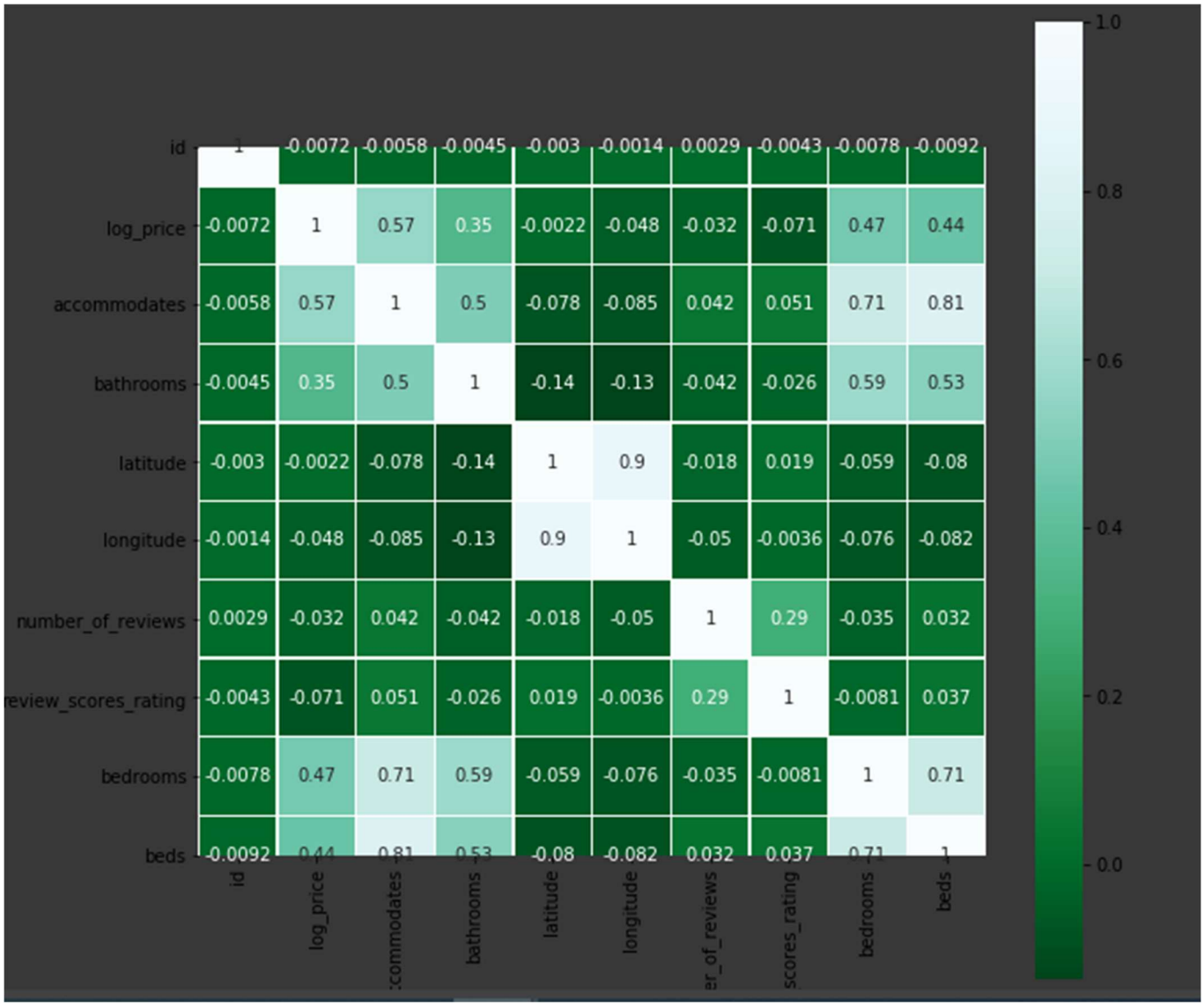
This figure shows different categorical variables that were used in the analysis.

Data Analysis

In data analysis, to answer the business questions we decided to use following methods.

a) Logistic Regression
b) Random Forest
c) Gradient Boosting
d) Clustering
e) Association Rule Mining

To start the data analysis, we decide to look into the correlation between the variables in the dataset.



As it can be seen for one of our target variable, log_price, the most correlated variables are accommodates, bathrooms, bedrooms and beads. Hence, these four variables were used in all the analysis methods, but we kept on changing other variables to find the mixture of variables that gave the lowest possible RMSE and highest possible accuracy.

LOG_PRICE

A> Random Forest

```
[ ]  regressor1=RandomForestRegressor(n_estimators = 100,max_depth = 80, min_samples_split = 3,
                                      min_samples_leaf=8, max_features=3,bootstrap=True)
```

We ran this model to predict the Log_Price. All the attributes were selected by running the different values of the attributes through a Grid Search. And the values used here give the lowest possible RMSE.

B> Gradient Boosting

```
gb_op = GradientBoostingRegressor(n_estimators=20, learning_rate = 1,
                                  max_features=2, max_depth = 2, random_state = 42)
gb_op.fit(X_train,y_train)
y_pred = gb_op.predict(X_test)
```

We ran this model to predict the Log_Price. All the attributes were selected by running the different values of the attributes through a Grid Search. And the values used here give the lowest possible RMSE.

THINKING OUTSIDE THE BOX:

To validate the models and make sure that the models were not over-fitting, we decide to use the Log_Price variable in a different way.

While handling the Log_Price variable, we took the mean of the Log_Price and created another column, a categorical one, which is a categorical column with values 0 and 1. We assigned 0 to the values lesser than mean and 1 to the values greater than or equal to the

mean. Now, to look at the accuracy of the model, we used this newly created column as the target variable.

For this analysis, we decided to only focus on the different categorical variables that the dataset offer. The different categorical variables that were used are as follows:

1> Room_Type:

```
#Handing Categorical Variables
filedata.room_type.value_counts()
```

```
Entire home/apt    41310
Private room       30638
Shared room         2163
Name: room_type, dtype: int64
```

2> Bed_Type

```
#creating dummy variable for column bed_type
filedata.bed_type.value_counts()
```

```
Real Bed         72028
Futon              753
Pull-out Sofa      585
Airbed             477
Couch              268
Name: bed_type, dtype: int64
```

3> Cancellation Policy

```
#creating dummy variable for column cancellation_policy
filedata.cancellation_policy.value_counts()
```

```
strict             32374
flexible           22545
moderate           19063
super_strict_30      112
super_strict_60       17
Name: cancellation_policy, dtype: int64
```

4> City

```
[ ]  #creating dummy variable for column city
     filedata.city.value_counts()
```

```
NYC        32349
LA         22453
SF          6434
DC          5688
Chicago     3719
Boston      3468
Name: city, dtype: int64
```

5> Instant Bookable

```
[ ]  #creating dummy variable for column instant_bookable
     filedata.instant_bookable.value_counts()
```

```
f    54660
t    19451
Name: instant_bookable, dtype: int64
```

6> Property Type

```
[ ]  #creating dummy variable for column property_type
     filedata.property_type.value_counts()
```

```
Apartment             49003
House                 16511
Condominium            2658
Townhouse              1692
Loft                   1244
Other                   607
Guesthouse              498
Bed & Breakfast         462
Bungalow                366
Villa                   179
Dorm                    142
Guest suite             123
Camper/RV                94
Timeshare                77
Cabin                    72
In-law                   71
Hostel                   70
Boutique hotel           69
Boat                     65
Serviced apartment       21
Tent                     18
Castle                   13
Vacation home            11
Yurt                      9
Hut                       8
Treehouse                 7
Chalet                    6
```

We also decided to use some interaction terms in the model, because we had a lot of fields which had numeric values and were dependent on each other to predict the Log_Price. Hence, we created different interaction terms to help our prediction. The interaction terms are as follows:

1> Bed * Bathrooms * Bedrooms

```
[ ]  interactionDF= pd.DataFrame()

[ ]  interactionDF['bedrooms']=filedata['bedrooms']
     interactionDF['beds']=filedata['beds']
     interactionDF['bathrooms']=filedata['bathrooms']

[ ]  interactionDF['bed*bathroom*bedrooms']=filedata['bedrooms']*filedata['beds']*filedata['bathrooms']
```

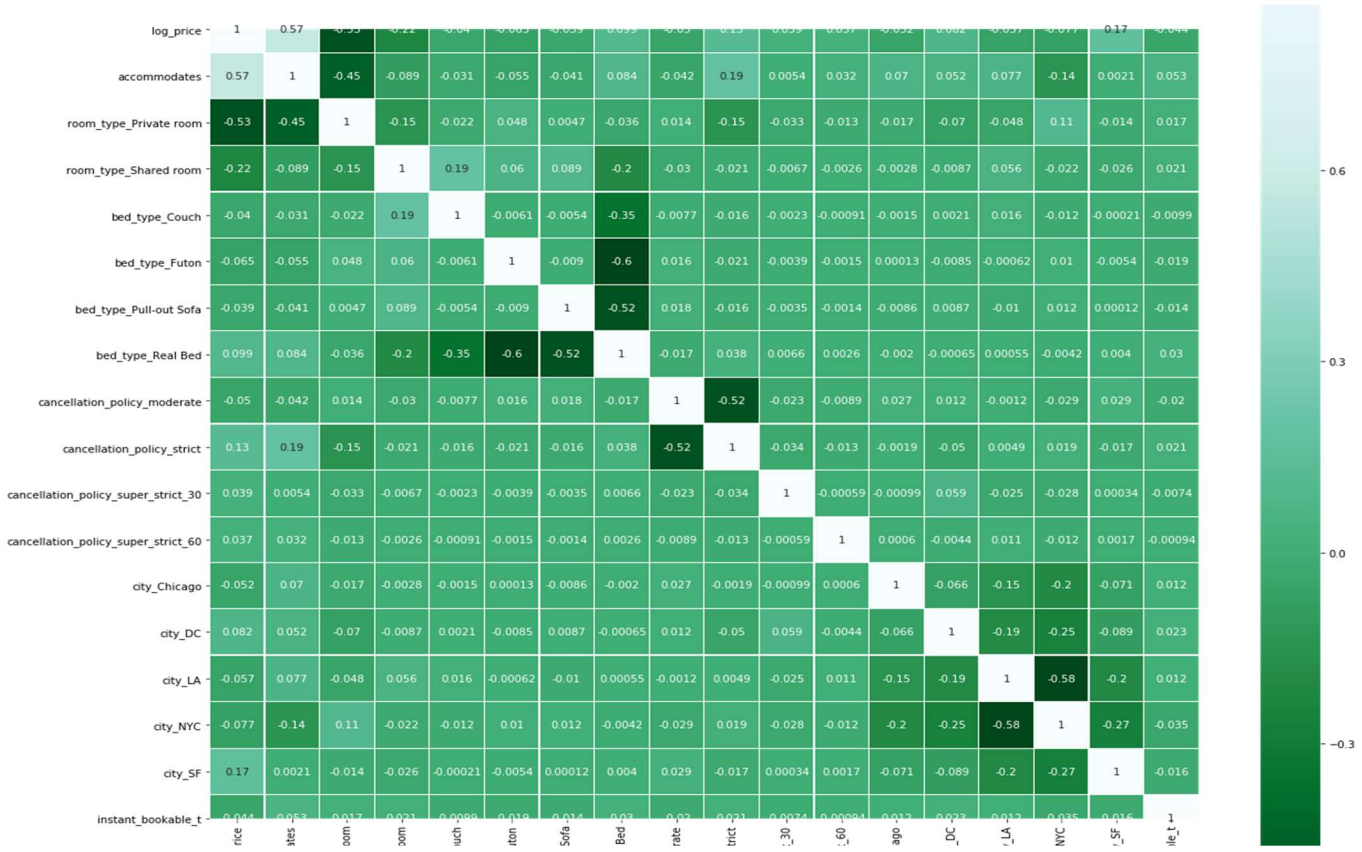
2> Review_Scores_Rating * Number_Scores_Ratings

```
[ ]  interactionDF1= pd.DataFrame()

[ ]  interactionDF1['review_scores_rating']=filedata['review_scores_rating']
     interactionDF1['number_of_reviews']=filedata['number_of_reviews']

[ ]  interactionDF1['reiew_score*Number']=filedata['review_scores_rating']*filedata['number_of_reviews']
```

After creating these variables, we also looked for the correlation between them:



As seen from correlation plot, the correlation values look good for log_prices, thus helping us choose the variables for the prediction.

MODELS FOR THE NEWLY CREATED LOG_PRICE:

A> Logistic Regression:

```
classifier = LogisticRegression()

[ ] classifier.fit(X_train,y_train)

/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/logistic.py:432: FutureWarning: Default
  FutureWarning)
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='warn', n_jobs=None, penalty='l2',
                   random_state=None, solver='warn', tol=0.0001, verbose=0,
                   warm_start=False)
```

We ran this model to predict the Log_Price as a categorical variable.

B> Random Forest:

```
[ ]  random1 = RandomForestClassifier(n_estimators = 200,max_depth = 90,
                                      min_samples_split = 8, min_samples_leaf = 3,
                                      max_features = 3,bootstrap = True)

[ ]  random1.fit(X_train,y_train)
     y_pred = random1.predict(X_test)
```

We ran this model to predict the Log_Price as a categorical variable. All the attributes were selected by running the different values of the attributes through a Grid Search.

C> Gradient Boosting:

```
[ ]  #Using learning rate = 1
     gb_op = GradientBoostingClassifier(n_estimators=20, learning_rate = 1.0,
                                        max_features=2, max_depth = 2, random_state = 0)
     gb_op.fit(X_train,y_train)

     GradientBoostingClassifier(criterion='friedman_mse', init=None,
                                learning_rate=1.0, loss='deviance', max_depth=2,
                                max_features=2, max_leaf_nodes=None,
                                min_impurity_decrease=0.0, min_impurity_split=None,
                                min_samples_leaf=1, min_samples_split=2,
                                min_weight_fraction_leaf=0.0, n_estimators=20,
                                n_iter_no_change=None, presort='auto',
                                random_state=0, subsample=1.0, tol=0.0001,
                                validation_fraction=0.1, verbose=0,
                                warm_start=False)
```

We ran this model to predict the Log_Price as a categorical variable. All the attributes were selected by running the different values of the attributes through a Grid Search.

D> Clustering

Clustering is one of the most common exploratory data analysis technique used to get an intuition about the structure of the data. In our analysis, after many combinations and iterations in Weka, we got four clusters with each cluster having almost equal instances.

The clusters were divided based on their location like LA or NYC and property type. Another insight was over the cancellation policy and it is having different values in different clusters. This was important to us as cancellation policy being one of our target variables.

```
Final cluster centroids:
                                                Cluster#
Attribute                            Full Data          0           1           2           3
                                     (44044.0)   (7341.0)   (12896.0)   (13774.0)   (10033.0)
=============================================================================================================
log_price                               4.7969     5.0969       4.852      4.4412      4.9948
property_type                         Apartment      House   Apartment   Apartment   Apartment
room_type                   Entire home/apt Entire home/apt Entire home/apt Private room Entire home/apt
accommodates                            3.1291     4.7494      2.8162       1.995      3.9025
bathrooms                                 1.23     1.6918      1.0976       1.123      1.2094
bed_type                             Real Bed   Real Bed    Real Bed    Real Bed    Real Bed
cancellation_policy                    strict     strict    moderate    flexible      strict
cleaning_fee                             TRUE       TRUE        TRUE       FALSE        TRUE
city                                      NYC         LA         NYC         NYC          LA
host_since                            3/30/2015  2/14/2014  10/14/2013   7/21/2014   3/30/2015
neighbourhood                       Williamsburg      Venice  Williamsburg Williamsburg  Hollywood
number_of_reviews                      21.4766    26.4776     23.1687     13.1698     27.0465
review_scores_rating                   94.0538    94.6676     94.6797     93.6248     93.3891
bedrooms                                1.2597      1.973      1.0483      1.0246      1.3321
beds                                    1.7011     2.6292      1.4757      1.1885      2.0155



Time taken to build model (percentage split) : 0.17 seconds

Clustered Instances

0       3832 ( 17%)
1       6573 ( 29%)
2       7044 ( 31%)
3       5241 ( 23%)
```

E> Association Rule Mining

With such a diverse data set, we received many accusation rules but had to narrow it down to some which truly affect and help us in solving our business problems. Listed below are the main rule that had the strongest confidence and lift

- Cancellation policy is flexible when city is NYC and property type is Apartment
- Cleaning fee is TRUE when bed type is Real Bed
- Cancellation policy is Strict when city is LA and property type is Apartment

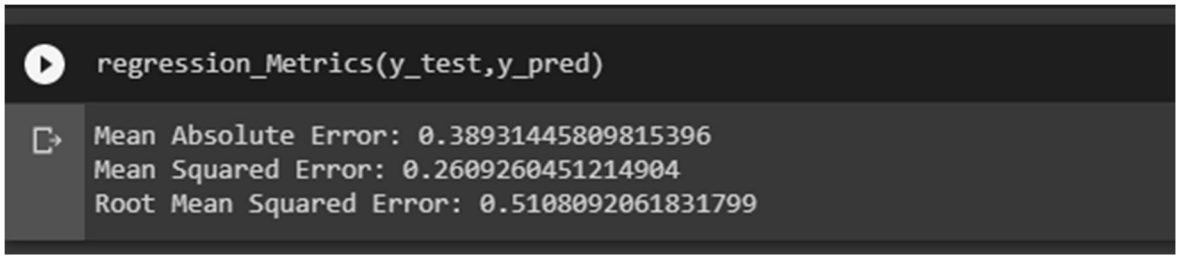
Model Evaluation

For model evaluation, for Log_Price we used Mean Absolute Error, Mean Squared Error and Root Mean Square Error. And for categorical variable we used accuracy.

LOG_PRICE

A> Random Forest

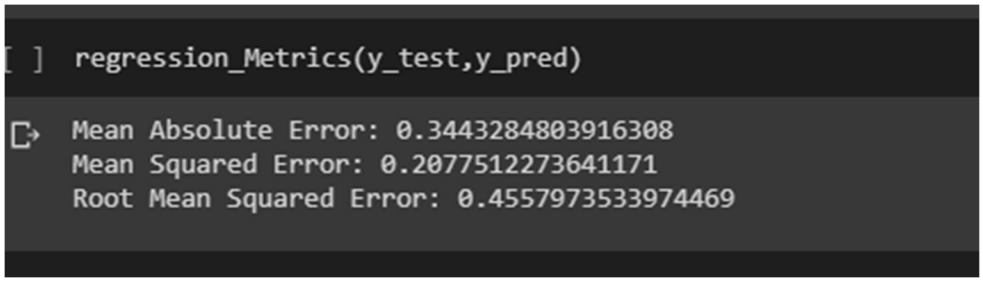
After running random forest for LOG_PRICE we got the following results:



These results we got for the most tuned model. We used Mean Squared Error as one of the important measures of how good the model is.

B> Gradient Boosting

After gradient boosting forest for LOG_PRICE we got the following results:



These results we got for the most tuned model. We used Mean Squared Error as one of the important measures of how good the model is.

From both the models used for prediction, the Gradient Boosting model performed the best as it can be seen form the Mean Squared Error. The Mean Squared Error is the lowest for the Gradient Boosting. In the beginning of the analysis, we used PCA and correlation plots to decide on the variables. So there is no over-fitting of the model.

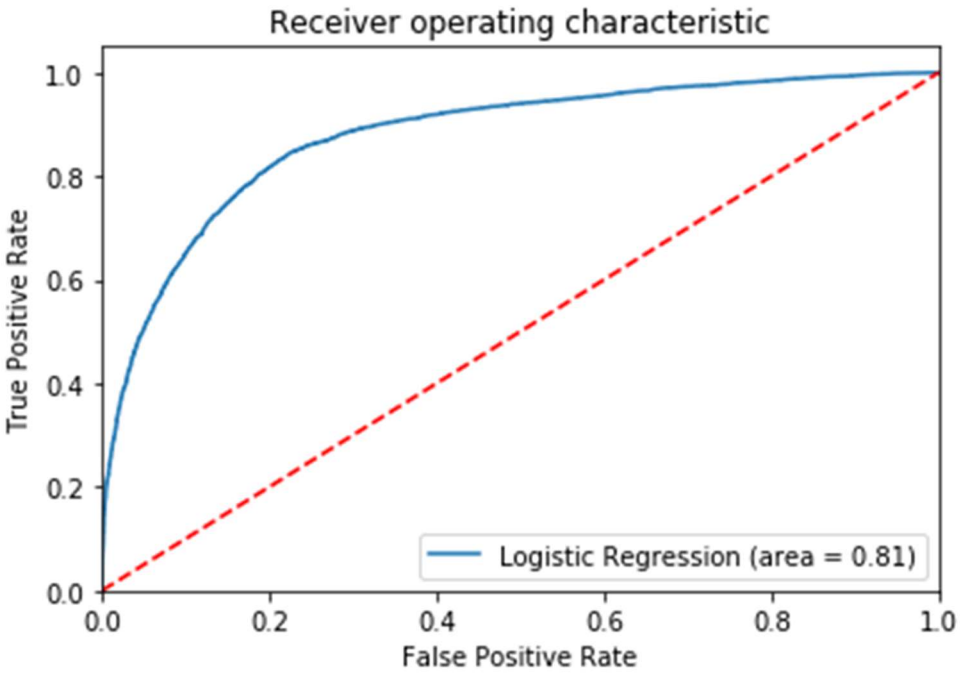
MODELS FOR THE NEWLY CREATED LOG_PRICE:

A> Logistics Regression.

After running logistics regression for LOG_PRICE we got the following results:

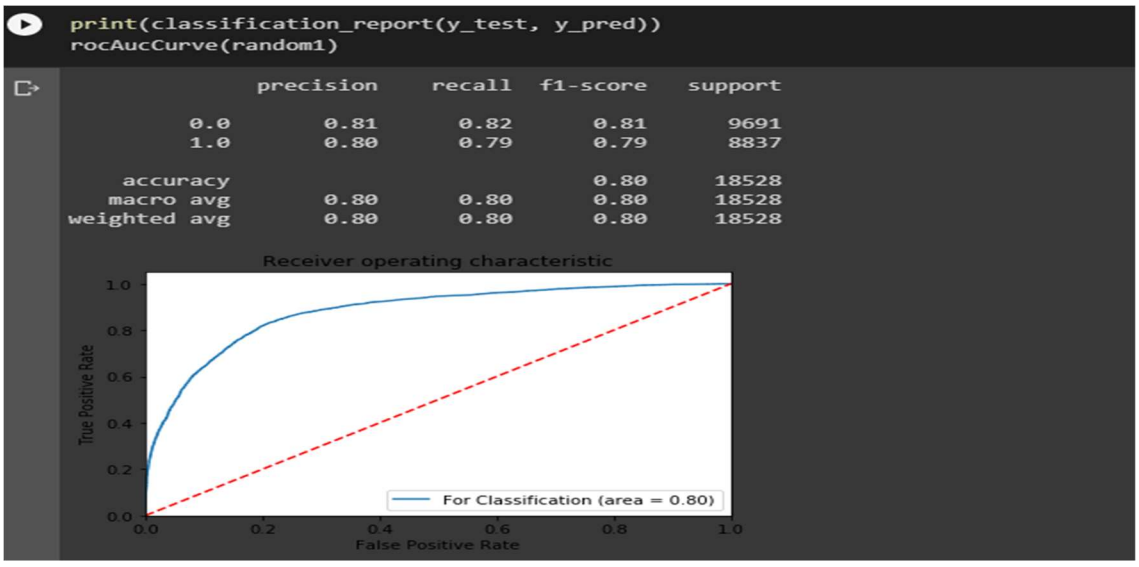
```
[ ] print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0.0	0.83	0.80	0.81	9691
1.0	0.79	0.82	0.80	8837
accuracy			0.81	18528
macro avg	0.81	0.81	0.81	18528
weighted avg	0.81	0.81	0.81	18528



The accuracy we got 81% accuracy after running the logistic regression.

B> Random Forest



The accuracy we got 80% accuracy after running the random forest.

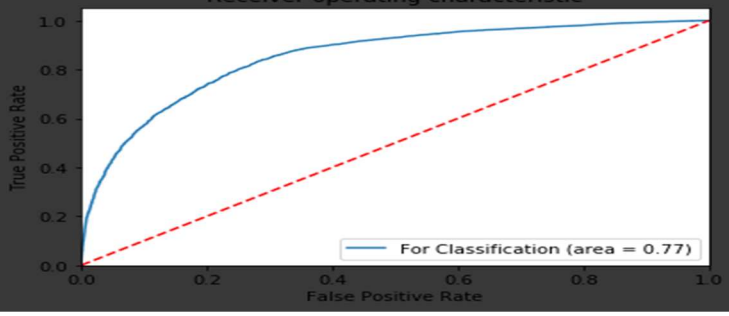
C> Gradient Boosting

```
[ ]  print(classification_report(y_test, y_pred))
     rocAucCurve(gb_op)
```

```
                   precision    recall  f1-score   support

              0.0       0.77      0.80      0.78      9691
              1.0       0.77      0.73      0.75      8837

         accuracy                           0.77     18528
        macro avg       0.77      0.77      0.77     18528
     weighted avg       0.77      0.77      0.77     18528
```



The accuracy we got 77% accuracy after running the gradient boosting.

As it can be seen from the accuracy we got, the accuracies for all the three models are high. So these models can be used to determine if the prices that are put on listings are around the mean of that area or no.

Challenges

With data sources multiplying and complexity rising, the most common challenge was getting the relevant data. The challenge is mining the seemingly endless data sets, sifting and sorting it to get data that is valuable and useful.

The challenge here for us was to understand the broader purpose of the data. Then use our expertise to analyses the datasets, and to piece together the insights for consumption. Which type of visual analytics to use and selecting the best ways to crunch enormous volumes of data, select and present the data for meaningful interpretation.

As the dataset was large and diverse it took time for cleaning and pre-processing more than the actual data analysis. It was time consuming to tune the parameters of algorithms in Weka and was difficult to keep the track of the parameters that have been set on the previously built models

Conclusion:

The Airbnb data that we had selected was diverse and had many possibilities. After analyzing the data rigorously, we can finally infer some business values that will be beneficial for the customers, host and to Airbnb itself.

When looking at the log price, with an error of 0.207 (i.e., $1.60 - $1.75), we can say that the given price of the house is over or below the present margin.

When a host wants to put listing and wants to see if the listing is below or above the mean price, with 81% accuracy, our model can predict if it is above or below the mean price.

Customers can get insights about areas having strict cancellation policy and factors deciding it. The cancellation policy factors heavily over the reviews and area where the Airbnb house is located.

There is usually no cleaning fee if the Bed type is anything other than a real bed (e.g. Couch, Futon, etc.) The mean price of a certain houses in San Francisco and DC generally have higher prices.

LA have the most lavish houses with costs of over $2000 per night and having 16 beds and 8 baths. Book sharing or guest room is a new feature of Airbnb comparatively and tend to have a flexible cancelation policy. This gives the customer an option to cancel the booking and not lose out on any money.