Airbnb Price & Superhost Prediction for London, UK

Divyank Rahoria
University of California, San Diego
drahoria@eng.ucsd.edu

Sai Jadhav
University of California, San Diego
smjadhav@eng.ucsd.edu

Sharath Dharmaji
University of California, San Diego
sdharmaj@eng.ucsd.edu

***Abstract*— Airbnb is a sharing platform that links suppliers of living space with those requiring short-term accommodation. However, it's not easy for home owners to decide the price for their accommodation. To be in demand as well as earn appropriate profit, price is the main criteria. In this project, we considered solving the problem regarding setting the price for normal days. Apart from that, we also considered another problem regarding finding out the qualities required to be an Airbnb Superhost. In this project we used the Airbnb dataset of London and filtered out required data. Once we got the required data, we extracted different features responsible for price prediction. These features are trained using machine learning techniques such as Linear regression, Gradient boosting regression and Random Forest regression for predicting the price. Among all of these, Gradient boosting regression gave the best performance result. We used different classifier such as Decision Tree classifier and Random Forest classifier and XGBoost for extracting features for Superhost classification task. All these features and machine learning techniques gave quite satisfying results. For the price prediction task, we calculated the mean square error (MSE) and R^2 statistics over refined data. Similarly we used accuracy and precision for determine the quality of prediction of a Superhost.**

Keywords—Linear Regression, Gradient Boosting Regression, Random Forest Regression, Decision Tree Classifier, Random Forest Classifier, Mean Square Error, Precision Matrix, Data Normalization, R^2 Statistics, Precision

I. Introduction

At the peak of connecting via internet and social media, the concept of sharing is now getting a makeover. Airbnb is one such service that allows users to share their most intimate setting: the home. It's a fast growing shared economy since its first appearance in 2008. This company do not own any home in real, it acts as a broker agent between the owners and customers. However Airbnb regularize by some set of rules for the properties and it makes it more reliable for the customer end. Airbnb has around 4 million lodging listings world wide with around USD 2.6 billion annual revenue. It offers various type of lodging properties including Apartment, Villa and Private home etc. But one of the major issue is regarding the price for the new listing for the user. As the pricing depends on various factors like number of bedrooms, neighborhood area, property type, number of guests allowed etc. So it's not easy for the user to decide the price for their listing. This could be easy if they could take account all these features and compare with other listings. So we did that in this project, we analyzed the data off older listings and tried to predict the ideal price for a new listing. We trained the same using different machine learning techniques listed as follows:

1.Linear Regression

2.Gradient Booster Regression

3. Ridge Regression

4. Lasso Regression

5. Random Forest Regression

We compared all the above mentioned regression method on our dataset and reasoning is provided in subsequent sections.

Apart from that, Airbnb also give a special title to the user who hosts very well based on user feedback. This approach has multifold advantages. It encourages the host to provide better services and facilities to the users. Apart from that, it makes the host more reliable at the customer end and that assure them for better facilities than the listing without superhost. We realized that every host wants to be listed as superhost so in this project, we found out the qualities needed for being a superhost. Airbnb is growing fast and new user comes up with their properties regularly. Our project results will help them to realize what qualities they should maintain for being a Superhost.

II. Literature Review

The dataset we used here is from non commercial and independent website "Inside Airbnb". It provides a set of tools and data that allows to explore how Airbnb is really being used in cities around the world. By analyzing publicly available information about a city's Airbnb's listings, Inside Airbnb provides filters and key metrics so we can see how Airbnb is being used to compete with the residential housing market. Inside Airbnb provides the data for the research purposes open to all. Following figure shows the Airbnb property spread in London provided by Inside Airbnb.

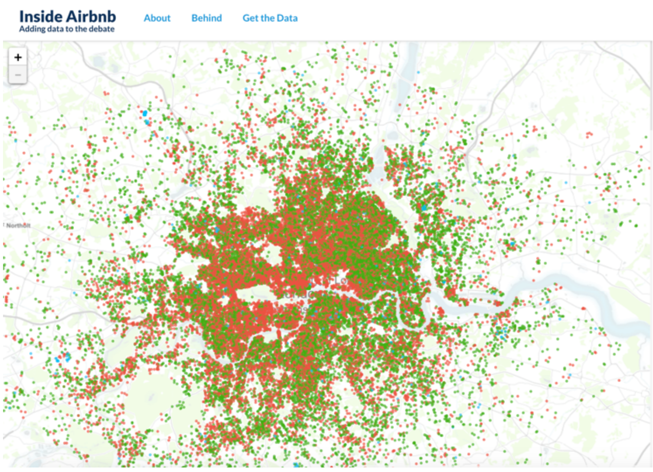


Fig. 1. Airbnb property listings in London

A. Previous Work

The dataset we are using here is very recently compiled i.e. November 2018 so there's no previous work on the same dataset has been found. But there are some other datasets that are similar to this dataset and been worked on previously. Airbnb Boston and Seattle listings are the similar dataset. The general problem statements for those datasets are as follows:

a) Modeling Price Prediction
b) Data cleaning
c) Superhost Prediction

The Boston and Seattle listings were quite less (around 4k) than the London listings (74k). The machine learning techniques used in the previous work was linear regression with some limited features like number of bedrooms, bathrooms etc. Hence the error rate was quite higher than we got for this project. Our project is more accurate and reliable because it is trained and tested on quite large dataset. The mean square error is quite better in this project rather than the previous work. The accuracy achieved in this project is a little higher than the other related work done in past. We used and compared all the different important machine learning techniques for the dataset. We performed Exploratory data analysis and tuned the dataset by removing outliers. The previous work was just on the dataset directly provided. There wasn't any further analysis and data cleaning used. This made our results better than the previous work
In the superhost prediction, we used two different classifiers and gave a comparative analysis over the result. That was something missing on the previous work done.

III. Dataset

The dataset we used for this project is sufficiently large dataset with 74k unique Airbnb listings in London. Following sections will provide the basic statistics and interesting findings of the data.

A. Basic Statistics and Properties

We used London Airbnb Open Dataset for our case project. This dataset provides the following information:

(1) Listings: A csv file which has listings across various properties in London.
(2) Calendar: A csv.file which has details on booking date and price for each listing for 365 days.
(3) Review: A csv.file which has review provided for a particular listing id by a particular reviewer.

The neat thing about this dataset is that we get all the public information as provided on the official Airbnb website, regarding each and every single listing/review, all properly stored in CSV format. This is a huge benefit as it significantly reduced the time needed for us to work on creating the dataset by collecting data from the source Airbnb website. We used the above mentioned datasets for our project.

B. Dataset Description

Inside Airbnb collects all the information provided publicly on the Airbnb website, it was able to provide us with unique features regarding each Airbnb listing. The feature list included various information ranging from property type, neighborhood of the place provided by the host, whether a host is superhost or not, facilities offered, review score of the place on different criteria, latitude - longitude of the house, all the way to the house's daily price.

We can quickly see that this dataset included a mixture of qualitative, quantitative, and repeated information. The qualitative data included the summary and descriptions the host provided. Some of the data had Nan value. This may pose a challenge later on when we decide to do data cleaning and analysis to seek further insight.

The data also has an abundant amount of quantitative information such as average review for the place in different categories, number of bedrooms, number of bathrooms owner's pricing for each night/week/month, availability for 30, 60 and 90 days, number of guests allowed, longitude/latitude, etc. This information allowed for quick analysis on the data set without much preprocessing. Then we also had a lot of redundant information. For example, the dataset included street names, latitude-longitude on which the housing is located, but it also provided the city, zip code, and general neighborhood name. All of this information can be derived from just the latitude-longitude. The description and the summary provided are also usually duplicates of each other. This meant that we can represent quite a few of these features by condensing them into one.There were some feature like if host accept the customers or not, so it was yes for all the listings except one. We removed that and similarly another outliers. Doing this gave a better dataset to work on.

This dataset provides answers of the various questions information about listings such as:

1. What was the price of the listing at a particular day of the year?
2. What is the neighborhood area of a particular listing?
3. Whether a host is superhost or not ?
4. What are the number of minimum nights to be booked?

C. Exploratory Data Analysis

To get a deep sense of the type of data we are working with and finding a few trends and similarity, we decided to check that by graphs. We began by analyzing how prices correlate with respect to different features such as room configurations (number of bedrooms, room type), day of the year, neighborhood etc. we carried out extensive exploratory analysis on our data. Few snapshots and interesting findings are given in brief in following sub sections. There were a total of 74k listings in the Airbnb London data.

After cleaning the data(redundant information, removing null rows, incomplete data, etc.) we were left with 54k listings. The finding on that cleaned data are as follows:

(1) Price Distribution
We started our data analysis by plotting the price distribution and found out that most of the listings are priced less than 100 dollars per night.
The price distribution graph is as follows:

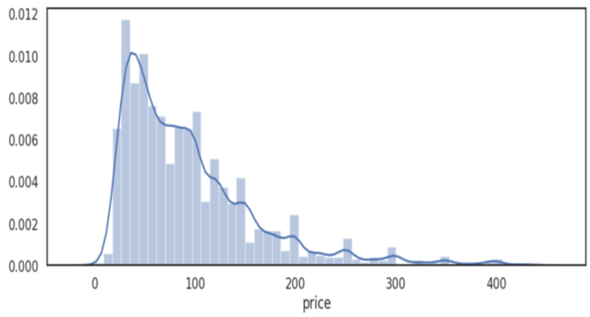


Fig. 2. Price Distribution of Airbnb listings in London

(2) Price/bedroom versus Number of bedrooms

For analysis of the data, we plotted Price (y axis) versus the Number of bedrooms (x axis) as shown in figure 2. We found that one bedroom houses were expensive than other houses. Intresting finding was the fact that, as the number of bedrooms increased, prices got lower.

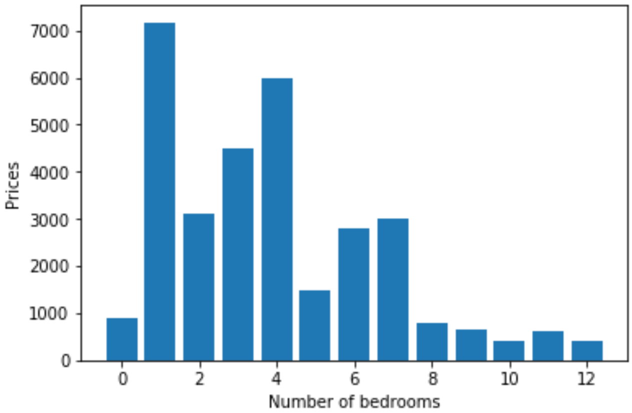


Fig. 3. Price versus Number of bedrooms

(3) Price versus Neighborhood

First of all, we took all centroid of all the neighborhoods with the nearby areas to get Neighborhood distribution. The distribution is as follows:

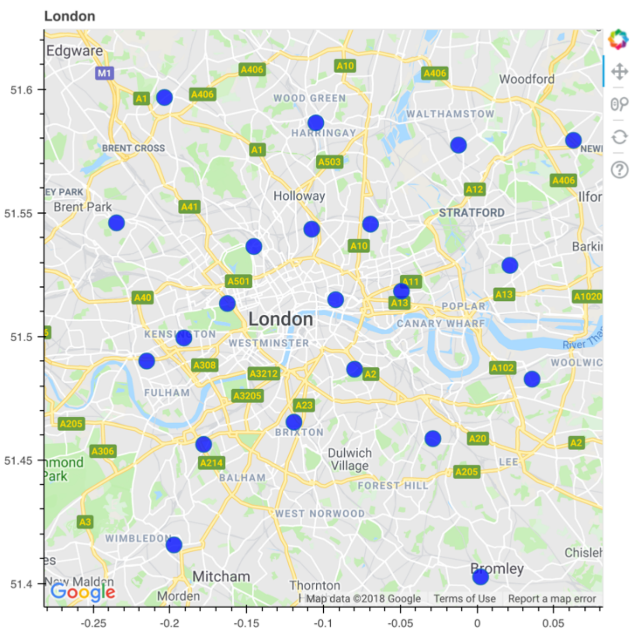


Fig. 4. Airbnb property distribtion in neighborhood area of London

Then we checked the price variation over the neighborhood area. The price that the customers pay for a property is proportional to the neighborhood. To find this correlation in our dataset, we plotted prices (y axis) versus the neighborhoods (x axis). Some area like London downtown had the prices higher than other areas.

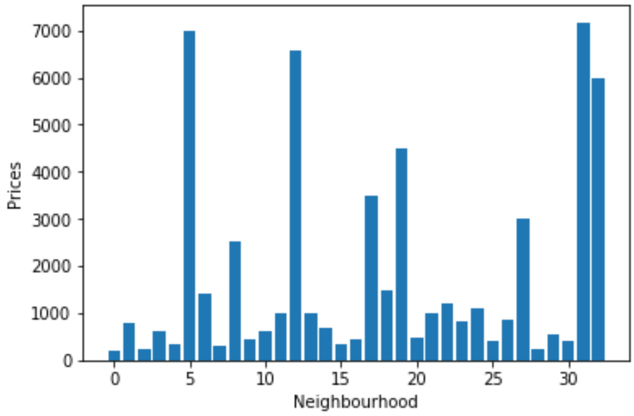


Fig. 5. Price versus Neighborhood

(4) Price versus Property Type

We had 12 different property type in our data like apartment, villa, bungalows etc. We again merged them in 7 different property type according to similarity. To find this correlation in our dataset, we plotted prices (y axis) versus the property type (x axis).

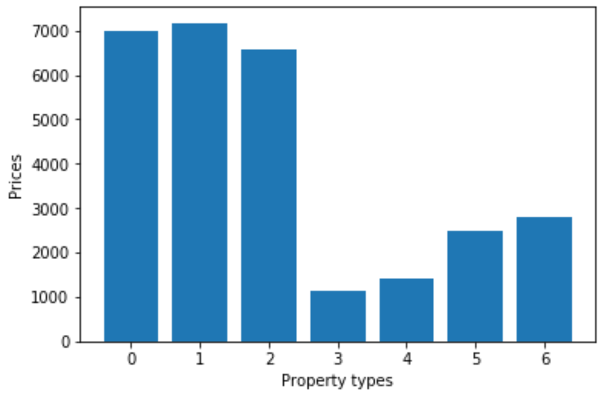


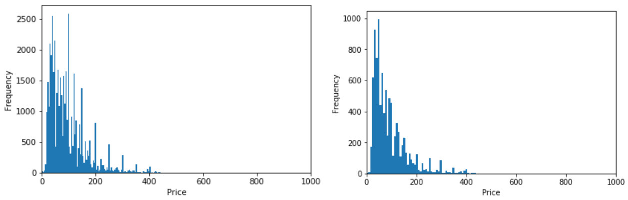
Fig. 6. Price versus Property type

We analyzed that house type of property had the highest price among other property type.

(4) Superhost versus Normal host

The another major model in this project apart from price prediction is superhost prediction so we did a comparative anaysis of superhost versus normal host. We segregated the hosts into super hosts and normal hosts and we made the following plots:

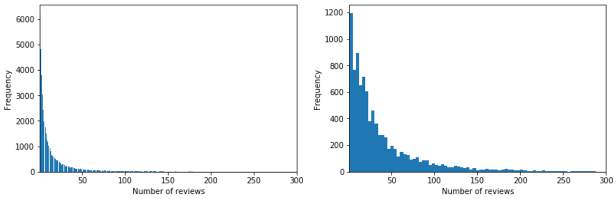
(i) (a) Number of normal hosts versus price and (b) Number of super hosts versus price (Figure 7). We found that the super hosts and normal hosts had similar type of graph. But the frequency for listings by normal hosts were quite higher (approx. 2.5 times).



(a) (b)

Fig. 7. Number of hosts versus price (a) Normal hosts, (b) Superhosts

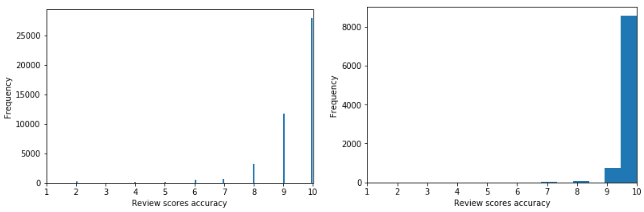
(ii) (a) Number of normal hosts versus Number of reviews and (b) Number of super hosts versus Number of reviews (Figure 8). We found that the super hosts were more likely to get reviews from their customers.



(a) (b)

Fig. 8. Number of hosts versus price (a) Normal hosts, (b) Superhosts

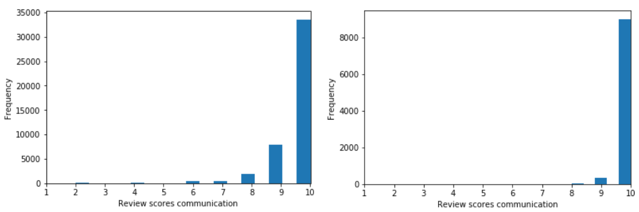
(iii) (a) Number of normal hosts versus Review score accuracy and (b) Number of super hosts versus Review score accuracy (Figure 9). We found that the super hosts had more review accuracy than normal host from their customers.



(a) (b)

Fig. 9. Number of hosts versus price (a) Normal hosts, (b) Superhosts

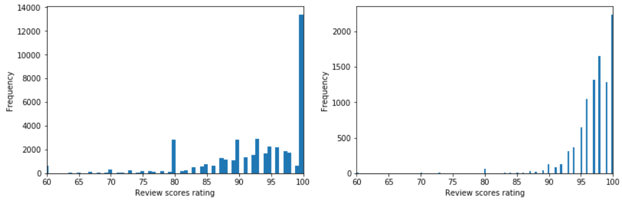
(iv) (a) Number of normal hosts versus Review score communication and (b) Number of super hosts versus Review score communication (Figure 10). We found that the super hosts had review score communication always above 8 while normal host did go below 8 ratings from their customers.



(a) (b)

Fig. 10. Number of hosts versus price (a) Normal hosts, (b) Superhosts

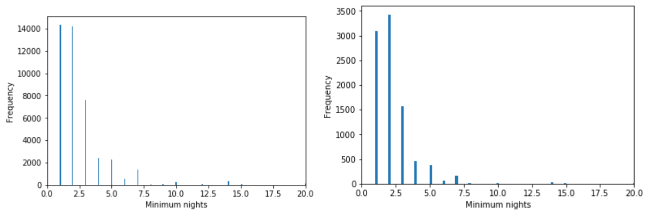
(v) (a) Number of normal hosts versus Review score rating and (b) Number of super hosts versus Review score rating (Figure 11). We found that the super hosts had majority of review score communication above 90 while normal hosts had more ratings distribution from 75 to 95.



(a) (b)

Fig. 11. Number of hosts versus price (a) Normal hosts, (b) Superhosts

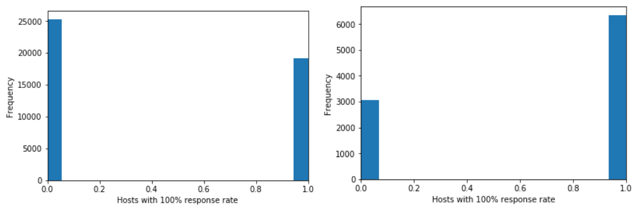
(vi) (a) Number of normal hosts versus Minimum nights and (b) Number of super hosts versus Minimum nights (Figure 12). We found that the both normal hosts and super hosts generally accept one night as minimum nights. However while looking closely, Superhosts demands booking of two nights rather than one night.



(a) (b)

Fig. 12. Number of hosts versus price (a) Normal hosts, (b) Superhosts

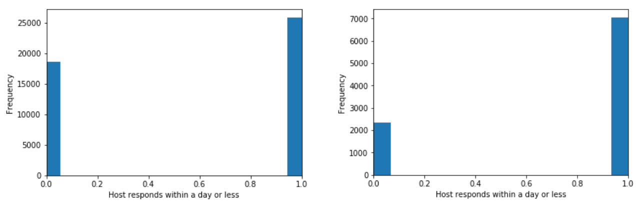
(vi) (a) Number of normal hosts versus Hosts with 100% response rate and (b) Number of super hosts versus Hosts with 100% response rate (Figure 13). We found that the normal hosts rejects more booking rather than a superhost.Superhost is more likely to accept the booking request.



(a) (b)

Fig. 13. Number of hosts versus price (a) Normal hosts, (b) Superhosts

(vii) (a) Number of normal hosts versus Hosts respond within a day or less and (b) Number of super hosts versus Hosts respond within a day or less (Figure 14). We found that the super hosts more likely to respond in a day or less rather than normal hosts.



(a) (b)

Fig. 14. Number of hosts versus price (a) Normal hosts, (b) Superhosts

IV. Predictive Tasks

We identified two predictive tasks that we could perform using the dataset. One was to model the prices based on the configuration of the property, location and other important price depending features. The second task is to predict the characteristics needed for a host to promote to a super host. Before that we divided the dataset into three sections that is Train data(50%) , Test data(25%) and Validation data(25%).

A. Modeling Price

The dataset provided rich insights into the features that had a strong influence on how the property would be priced. We built a model to predict the prices with respect to these features. A few notable features that could be used are, room type, number of bedrooms, number of bathrooms, neighborhood area, no of extra guests and cancellation policy. We excluded the top 1 percentile of prices from our prediction tasks as they were clear outliers.

B. Modeling Superhosts

Airbnb gives certain hosts a special status called Superhost. title. These are the hosts, who provide a shining example to other hosts and extraordinary experiences for their guests. A Superhost title provides a number of benefits to both Airbnb and the host. We did an in-depth study of Airbnb reviews and listings data to gain more insights into this problem. Though there are a set of guidelines provided by Airbnb on how to become a super host, we found there are many more latent factors that goes into determining if a host would become a super host. In our study, we visualized various features to determine their relationship with making the user a super host. At the end of the study, we also ran a XGBoost Classifier and Random Forest Classifier to determine if the new user is a super host and determined it's qualities for being a superhost (feature importances).

C. Relevant Features

(i) Relevant features for modeling prices:

From the exploratory data analysis of the dataset, we found that the features related to neighborhood, room type, accommodates, number of bedrooms and number of bathrooms etc. influence the prices the most. Some of the features are as listed below:
(a) Property type: 12 types of property, ex. Townhouse, Villa, Guesthouse. We merged them into 7 types and that feature was converted to a categorical feature ranging from 0 to 6.
(b) Room type: 3 types of rooms, Private Room, Entire home / apartment, Shared room.
(c) Bathrooms: The number of bathrooms that a property has.
(d) Bedrooms: The number of bathrooms that a property has.
(e) Beds: The number of beds in the property.
(f) Bed type: different types of beds, ex. Pull-out Sofa, Couch etc.
(g) Security deposit: amount of money paid as deposit.
(h)Neighborhood: neighborhood of the property.
(i) Accommodates
(j) Review ratings
(k)Minimum number of nights
(l) Is location exact

(ii)Modeling super host:

For predicting whether a user can be a super host, we explored
various features and found the followings as important:
(a) number of reviews: Number of reviews written by the guests for a host.
(b) Property type: 26 types of property, ex. Townhouse, Villa, Guesthouse. We merged them into 7 types and that feature was converted to a categorical feature ranging from 0 to 6.
(c) Room type: 3 types of rooms, Private Room, Entire home / apartment, Shared room.
(d) Bathrooms: The number of bathrooms that a property has.
(e) Bedrooms: The number of bathrooms that a property has.
(f) Beds: The number of beds in the property.
(g) Bed type: different types of beds, ex. Pull-out Sofa, Couch etc.
(h) Reviews scores communication: Rating of how easy was it to communicate with the hosts
(i) Host Response Time: How quickly the hosts responded.
(j)Neighborhood: neighborhood of the property.
(k) Accommodates
(l) Review ratings
Apart from that we added another dataset we had for the guests review.
We applied different classifiers to get the best results we could achieve. Results are in the following sections.

V. Models

Here are our experiments with different models for both our predictive tasks.

A. Modeling prices

(a)Baseline:We predicted the mean of all the prices(excluding top 1 percentile).
(b) Linear Regression: We did this to analyze the linear relationship between the test and train data.
(b) Ridge Regression: We needed to improve on the previous model. We used Ridge Regression which has L2 regularization in it.
(c) Lasso Regression: In cases where relevant information is smeared over large parts of the spectrum asking the regularization to drop variates will low co-efficient is not a particularly sensible approach. Two parameters which are very well correlated maybe dropped by Lasso Regression.
(d) Random Forest: Random forests are an ensemble learning method for regression, that operate by constructing a multiple decision trees at training time and outputting the mean prediction of the individual trees as the final prediction. Random decision forests prevent decision trees' overfitting by optimizing the tuning parameters that governs the number of features that are randomly chosen to grow each tree.
(e) Gradient Boosting: Gradient Boosting is an ensemble technique in which weak predictors are combined in building

a better model.These weak predictors learn from the misclassifications from the previous steps and better in the next steps by boosting the importance of incorrectly predicted data points. The aggregate forecast got from each of the weak learners will be much better than each of the learners alone. We got the best performance in the case of Gradient boost regressor.

B. Modeling Superhost

We used two different models to predict if the new user is a super host or not.
Apart from general classification, we used different classifier for modeling superhost that are as follows:

(a) Baseline models: we implemented two models which we act as baseline for accuracy and precision.
(i)Baseline1: Hosts with reviews/ratings=100 were predicted as superhosts. This gave us our baseline precision
(ii)Baseline2:All hosts were predicted as not a super host . This gave us our baseline accuracy.

(b) Decision Tree Classifier: Decision tree learning uses a decision tree as a predictive model which maps observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modeling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. A decision tree is a simple representation for classifying examples.
In the task of classifying super hosts, there is a single target feature which is whether the host is a super host or not. Each internal node in a decision tree is labeled with an input feature. Each leaf of the tree is labeled with a class.

(c) Logistic Regression: Similar to linear regression, but instead of predicting the actual value we used logistic functions to predict a probability of 0 or 1.

(d) Random Forest classifier: We used Random forest classifier instead of decision tree as random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

(e) XGBoost Classifier: XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting that solved problems in a fast and accurate way.

VI. RESULTS

A. Modeling prices

As the first step towards building the models, we built the models based on Baseline modeling, Linear Regression, Ridge Regression, Lasso Regression, Random forest and Gradient Boosting. These models were initially built without cross-validation.
Mean Square Error (MSE) was used as a metric to compare

Model	Train Error (MSE)	Test Error(MSE)	R^2 Statistic
Baseline	4602	5031	0.061
Linear Regression	1621	1686	0.639
Ridge Regression	1608	1660	0.645
Lasso Regression	2103	2145	0.532
Gradient Boosting Regression	1443	1521	0.669
Random Forest Regression	440	1620	0.668

Table 1 : Regression Models with MSE and R^2

the models. As shown in the Table 1, Linear Regression and Ridge Regression gave comparable errors. The Ridge regression was done with different regularization parameter. As the errors of the two models are comparable, the regularization added in Ridge did not impact the model a lot in terms of error rate.
Lasso Regression performed the worst on our model as shown in the Table 1. Reason is model would have dropped many features that could add value to the model as the L1 norm used in Lasso drops the features with lesser significance. Although some features may not be as significant as the others, they added value to the model and it is not suitable to drop them and hence Lasso Regression was not useful. The ensemble techniques like Random Forest and

Model	Hyperparameter Tuning	Test Error (MSE)	R^2 Statistic
Gradient Boosting Regression	n_estimators=300 learning_rate=0.1 max_features='sqrt'	1360	0.711
Random Forest Regression	n_estimators=500 depth=20 max_features='sqrt'	1279	0.718

Table 2 : Regression Models with MSE and R^2 after hypertuning

Gradient Boosting performed well and were comparable in performance as shown in Table 1.
As seen from the literature review, these methods which are state-of-the-art performed well on our predictive task as well. We did this regression on the Training as well as Test dataset and that is listed in the Table 1. As a next step, we checked R^2 statistics for all these regression model. As it is a coefficient of determination, that will give some information about the goodness of fit of a mode. The comparison between the models before and after cross-validation (Train and test) are shown in the Table 1.

GBR and Random forest gave better results, we hyper tuned data with these parameter:
For GBR: Number of estimators = 300, learning rate = 0.1 and maximum features = sqrt.
For Random Forest: Number of estimators = 500, depth = 20 and maximum features = sqrt.
This gave us better results as shown in Table 2

Apart from that, we checked the features of importance for the price prediction as shown in figure 15. Some important features are:

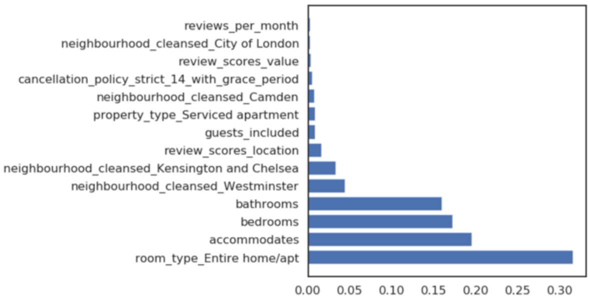


Fig. 15 : Feature of importance for Price predicrion task

(a)Room type
(b)Accommodates
(c) bedrooms
(d) bathrooms
(e) Neighborhood etc.

A. Modeling Superhost

The problem of modeling a super host is rather challenging. The major challenge being that there are not many distinguishing parameters between the super hosts and the normal hosts. To deal with this challenge, we performed exploratory analysis and found the various features that were distributed differently between the two type of hosts. We also encountered the problem of imbalanced dataset as the number of super hosts was just 17% of the total hosts.

Model	**Train Accuracy (%)**	**Test Accuracy(%)**	**Precision**
Baseline1(rating =100)	0.51	0.33	0.2
Baseline 2(All zeros)	0.79	0.76	0.0
Logistic Regression	84.40	84.04	0.65
Decision Tree Classifier	98.1	81.02	0.47
Random Forest Classifier	98.6	86.23	0.71
XGBoost Classifier	87.6	87.3	0.7

Table 3: Classifiers with Accuracy and Precision Results

To deal with the problem of imbalanced dataset we experimented with using Baseline with rating 100. The accuracy was very low in that case so checked for another baseline that is choosing all zeros. This was done just to get us baseline accuracy as its precision would obviously be zero. To improve on our accuracy and precision, we experimented with logistic regression and decision trees. We got a good improvement in the accuracy as well as precision. We got 98.1% training accuracy and 81.02% testing accuracy for decision tree. Which means decision trees were overfitting the train set. Then we tried random forest that gave a slight improvement on accuracy(98.6% for

Model	**Hyperparameter Tuning**	**Test Accuracy (%)**	**Precision**
Random Forest Classifier	n_estimators=100 max_features='auto' max_depth=8 criterion='gini'	90.3	0.77
XGBoost Classifier	n_estimators=100 max_depth=5 min_child_weight=5 gamma=0.01 min_leaves=2	91.2	0.78

train and 86.23 for test accuracy) and precision and that was

Fig. 15. Table 4: Best performing Classifiers with Accuracy and Precision Results

the best result we got. Then we tried using XGBoost, the accuracy was better in the case of test accuracy.

XGBoost and Random forest gave better results, we performed hyper parameter tuning on the validation set to and tested on the test set to confirm our hypothesis.
Tuning Metrics:

For XGBoost: Number of estimators = 100, gamma = 0.01 , max depth= 5, min child weight=6 and min leaves=2
For Random Forest: Number of estimators = 100, max depth = 20 ,maximum features = auto and criterion=gini

With regard to the features thatwere helpful in predicting whether a host is a super host or not. We found that there a number of features that come into play in determining the badge of a host. Airbnb lists a couple of attributes that are paramount for becoming a super host. Through our analysis we validated these attributes and found several other latent attributes that determines whether a host would become a super host. The important latent features
being:
_ Number of reviews written by the guest for a host
_ Level of cleanliness of the host
_ Responsiveness of the host
_ How accurate is the listing
_ Review score rating
_ Type of the room provided
It is shown in the figure below

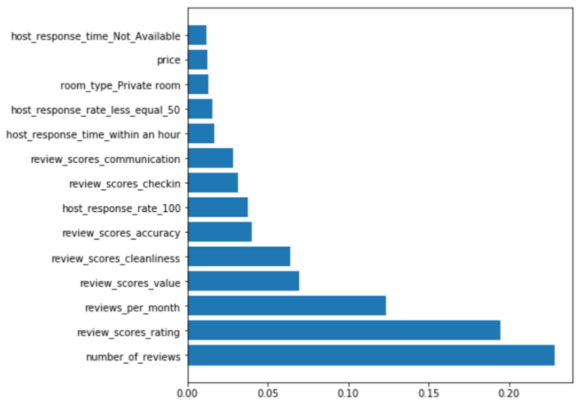


Fig. 16 : Feature of importance for Superhost predicrion task

VII. Conclusion

In this project, we performed two predictive tasks. First we studied the features that influenced price of the listings. Second, we studied what attributes make a super host.
For modeling prices, we studied various models. Machine learning techniques like Random Forest and Gradient Boosting proved to be better than most other models.
To derive characteristics associated with a super host, we performed exploratory data analysis to find that cleanliness, location, communication, number of reviews were major characteristics through which we can distinguish a super host. We verified our analysis, by building classifiers such as XGBoost and trees, and found our results were in line with the features we derived from our analysis.
The attempt to extract topics from review comments which differentiate between normal/super host did not work as expected. They didn't influence the results as much. However we beat our baselines and similar work performed by others by a margin mainly due to spending a lot of time in tuning our parameters to get optimum results.

VIII. Refrences

[1] https://www.kaggle.com/airbnb/boston

[2] Visit Limsombunchai. 2004. House Price Prediction: Hedonic Price Model vs. Artificial Neural Network. Commerce Division, Lincoln University (June. 2004), 1-15. https://ageconsearch.umn.edu/bitstream/97781/2/2004-9-house%20price%20prediction.pdf

[3] https://en.wikipedia.org/wiki/Airbnb

[4] https://en.wikipedia.org/wiki/Decision tree learning

[5] http://insideairbnb.com/get-the-data.html

[6] https://www.kaggle.com/residentmario/d/airbnb/boston/modelingprices/

[7] Winkler, RL. and Makridakis, S. (1983). .e Combination of Forecasts. J. R. Statis. Soc. A. 146(2), 150-157.

[8] Makridakis, S. and Winkler, R.L. (1983). Averages of Forecasts: Some Empirical Results.Management Science, 29(9) 987-996.

[9] https://www.cs.cornell.edu/home/cardie/papers/masa-sentire-2011.pdf

[10] http://res.cloudinary.com/general-assembly-pro.les/image/upload/v1474479496/geovpl78jbsx7dzqycde.pdf