Study of key features influencing Airbnb listing price

Jae Young Kim, Kevin Tsang

• Problem definition

Nowadays, many people use Airbnb as a platform for traveling accommodations and most customers would love to see if they are paying for what it is worth. Some hosts would love to see which factors they should consider to decide the proper price. We know that many factors can affect the price, so in this project, we want to determine the best features to predict booking price per night by exploring the data. We plan to use several machine learning algorithms and find out the feature importance of the input variables. That way, customers and hosts can benefit from understanding which features drive up the price.

• Description of background (why is this meaningful? What to solve? Contribution?)

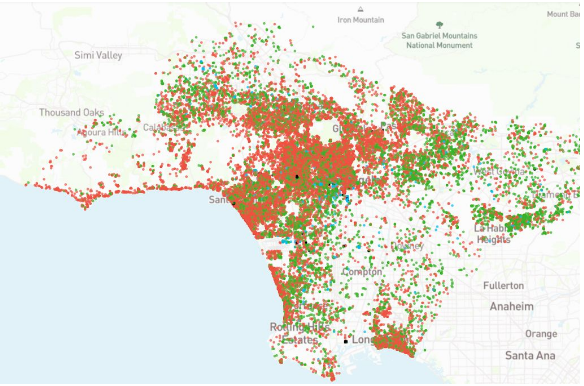
This project is beneficial for both the customers and the Airbnb hosts because we can see if the price is justified in a listing based on the features that affect the price the most. For example, if a host overcharges the price of the listing, customers can then use this data to decide whether it is worth it to stay at that specific location. For hosts, our result will help them decide the booking price of airbnb. Though in our dataset, we do not have the airbnb listing price that changes over time, our goal is not to predict the actual price, we want to see the feature importance of input variables, so even though the data is scrapped on a day, the data set can still prove meaningful for our study.

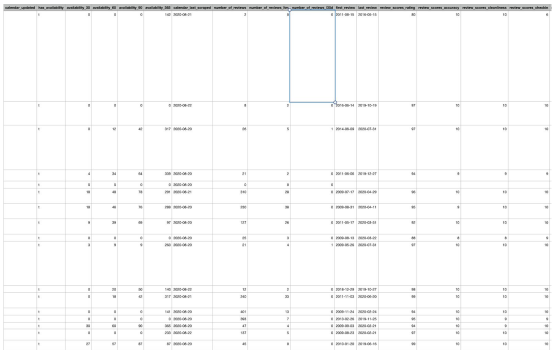
• Description of dataset (quantity, quality)

From "http://insideairbnb.com/get-the-data.html", we found a dataset of LA Airbnb listings. This dataset is under Common Creative CC0 Public Domain Dedication license. This dataset includes 33078 rows × 74 columns and this data was scraped on August 20, 2020, which are all the listings available for LA on that day (It is not airbnb listing which can be booked for that day).

Amenities, Bedrooms, Bathrooms and the number of reviews seem to be useful columns. Especially in the feature “Amenities”, there are lists of the amenities(eg. ["Smoke alarm", "Hot tub", "Indoor fireplace", "Elevator", "Free parking on premises"]) whose elements can be influential features deciding the quality of the rooms or homes. Finally after preprocessing dataset, we got 26 columns :

['host_since', 'host_is_superhost', 'host_listings_count',
'host_total_listings_count', 'host_verifications', 'latitude',
'longitude', 'distance_btw_lax', 'accommodates', 'bedrooms', 'beds',
'bathrooms', 'amenity_num', 'host_response_time', 'host_response_rate',
'host_acceptance_rate', 'neighborhood_matched', 'property_type',
'room_type', 'maximum_nights', 'minimum_nights', 'number_of_reviews',
'number_of_reviews_l30d', 'negiborhood_frequency',
'review_scores_rating', 'price']





calendar_updated	has_availability	availability_30	availability_60	availability_90	availability_365	calendar_last_scraped	number_of_reviews	number_of_reviews_ltm	number_of_reviews_l30d	first_review	last_review	review_scores_rating	review_scores_accuracy	review_scores_cleanliness	review_scores_checkin
	t	0	0	0	142	2020-08-21	2	0	0	2011-08-15	2016-05-15	80	10	10	6
	t	0	0	0	0	2020-08-22	8	2	0	2016-06-14	2019-10-19	97	10	10	10
	t	0	12	42	317	2020-08-20	26	5	1	2014-06-09	2020-07-31	97	10	10	10
	t	4	34	64	339	2020-08-20	21	2	0	2011-06-06	2019-12-27	94	9	9	9
	t	0	0	0	0	2020-08-20	0	0	0						
	t	18	48	78	291	2020-08-21	310	28	0	2009-07-17	2020-04-29	96	10	10	10
	t	18	46	76	289	2020-08-20	230	38	0	2009-08-31	2020-04-11	95	9	10	10
	t	9	39	69	97	2020-08-20	127	26	0	2011-05-17	2020-03-31	92	10	10	10
	t	0	0	0	0	2020-08-20	25	3	0	2009-08-13	2020-03-22	88	8	8	9
	t	3	9	9	263	2020-08-20	21	4	1	2009-05-26	2020-07-31	97	10	10	10
	t	0	20	50	140	2020-08-22	12	2	0	2018-12-29	2019-10-27	98	10	10	10
	t	0	18	42	317	2020-08-21	240	33	0	2011-11-03	2020-06-20	99	10	10	10
	t	0	0	0	141	2020-08-20	401	13	0	2009-11-24	2020-02-24	94	10	10	10
	t	0	0	0	0	2020-08-20	393	7	0	2013-02-26	2019-11-25	95	10	9	9
	t	30	60	90	365	2020-08-20	47	4	0	2009-09-03	2020-02-21	94	10	9	10
	t	0	0	0	233	2020-08-22	137	5	0	2009-08-23	2020-02-21	97	10	10	10
	t	27	57	87	87	2020-08-20	45	0	0	2010-01-20	2019-06-16	99	10	10	10

[Picture 1 The visualization of airbnb location in LA] [Picture 2 Dataset]

Preprocessing step

Since our dataset has 74 columns, our first step was collecting features which seems important. Our team removed some text features such as the name of the airbnb and description of the airbnb. Our team focused on numerical variables and categorical variables. With the selected variables, our team cleansed the values and created some new features and dummy variables for categorical variables.

For example, our team replaced null values with -1 and created distance from LAX airport by calculating distance with the latitude and longitude of LAX and the airbnb. In case of categorical variables like property type, room type and host response rate, after cleaning up the similar responses (e.g. private room in an apartment, private room in a house are both converted to private room), our team applied numerical encoding so we changed the categorical variables to numerical.

For feature “amenities”, the value of the feature was a list amenities such as “["Smoke alarm", "Hot tub", "Indoor fireplace", "Elevator", "Free parking on premises", "Air conditioning", "Heating", "Carbon monoxide alarm", "Pool"]”. At first, our team tried one-hot encoding for each of the amenities. However, the team found that there are 270 distinct types of amenities. Therefore, if the team applies one-hot encoding, it will produce 270 additional dimensions. It was definite that it will cause high dimension problem and it will cause deduction of the performance. Finally, the team decided to calculate the length of the list of the amenities.

For feature host_neighbourhood, it was the name of the city which airbnb is located like “Culver city”. To make this variable numerical, the team counted the frequencies of the appearance of the feature value in the whole dataset and replaced it with the count like “Culver city” → 316.

Finally, the team filtered out 25 input columns and 1 target column named price.

• Description of method used

Our team selected four algorithms to make a regression model of predicting the price of airbnb. Linear regression, Classification and Regression Trees (CART), Random Forest and XGBoost are those four algorithms. Since these algorithms are widely used as common regression models, we have decided to use these to study the feature importance of our input variables. For linear regression, CART and Random Forest, we used ScikitLearn packages to import the regression models and used root mean squared error in the metrics to see the performance of our models. For XGBoost, we used the XGBoost package to run the model and used ScikitLearn root mean squared error for metrics.

Our team decided to calculate the **Feature importance** of each input feature. Feature importance is a standard of measuring the relevance of the feature to the target variable. In algorithms which are based on decision trees like XGBoost, CART and

Random Forest, we can calculate the feature importance by measuring how much entropy decreased or how much Information gain the node has after passing the branch of the trees.

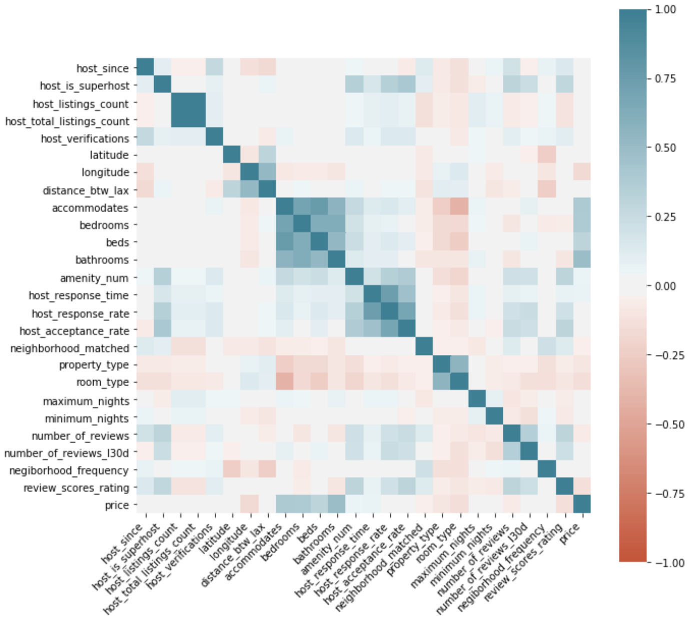
Background for Algorithms

Decision tree algorithms are an important type of machine learning algorithm that excels at predictive analysis. Classification and Regression Trees (CART), Random Forest and XGBoost are three prominent models used by data scientists. For CART, it adopts a binary tree representation which predicts the outcome by asking a set of if-else questions. Each tree has a root node that best divides the data and below that are internal nodes (have a parent node, and give two children nodes) and leaf nodes (have a parent node but do not have children nodes). Each branch indicates the values the node can assume, which is oftentimes the boolean answer to the if/else question. Node represents input variables and split points of the dataset while the leaf nodes are output that used to do prediction. CART determines feature importance scores based on the reduction in the criterion to select split points like Entropy.

Random Forest creates a large number of trees while decision trees build a simple single tree. It creates a large number of training samples and trains decision trees for every sample. After training, based on each trained model, it votes on the predicting results (We call this Bagging) After voting, the most voted result becomes the final prediction result. As it is an Ensemble Learning algorithm of different training samples, it can avoid overfitting.

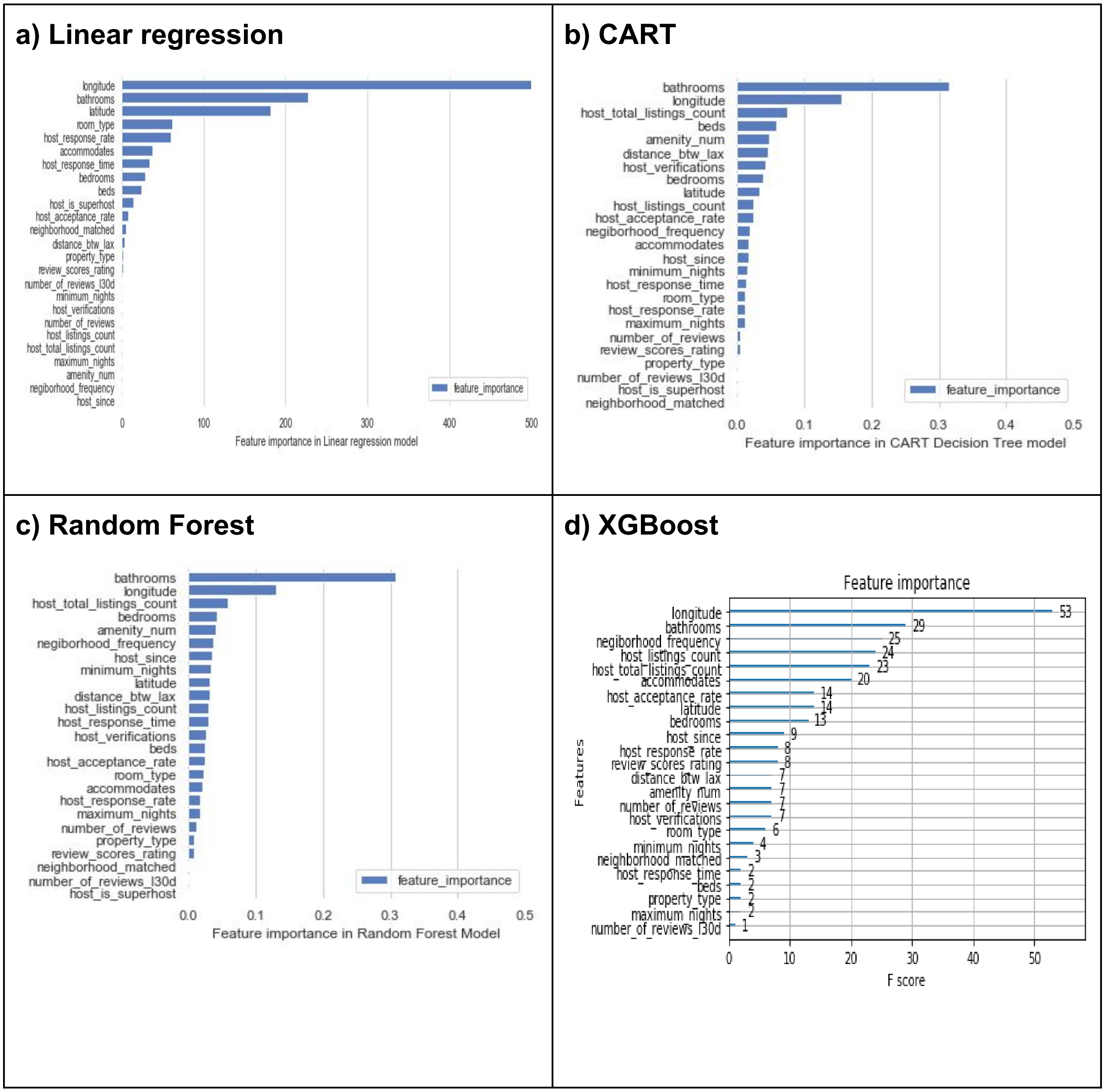
XGBoost is also an Ensemble Model based on decision tree. However, it is a kind of gradient boosting model. Similar to Random forests, xgboost also combines a large number of decision trees. XGboost combines the trees along the way, while Random forest combines the trees at the end of the process. Each of the trees in XGBoost is grown using information from the previously grown trees.(We call this Boosting) As XGBoost has the advantage of both Bagging and Boosting, it is widely used in classification and regression tasks.

• Experiment: analysis results



[Picture] Heatmap of correlation matrix of our input features and target feature(price)

Our team visualized the correlation matrix of the features. Some features had high correlation such beds, bathrooms, accommodates, bathrooms. All of them are related to the size of the airbnb. In addition, it had positive correlation with the price, the target feature. In addition, host_response_time, host_response_rate and host_acceptance_rate had positive correlation. Accommodates and room_type had weak negative correlation. As we encoded categorical feature room_type as {'Entire home/apt':1, 'Private room':2, 'Shared room':3, 'Hotel room':4}, we interpreted that the reason of this negative correlation is that the order of opposite order of encoding from the size of the airbnb.



[Table] Result of calculating Feature importance of each input features for all four algorithms chosen

After running feature importance functions in each algorithm, we find that for Linear regression, the top three input variables are Longitude, Bathrooms and Latitude, with feature importance scores of 500, 220 and 180 respectively. For CART, the top three input variables are bathrooms, longitude and host listings count, with feature importance scores of 0.31, 0.16 and 0.07 respectively. The Random Forest algorithm yields bathrooms, longitude and bedrooms as the top three variables (feature importance scores 0.30, 0.13, 0.06 respectively). Last but not least, the XGBoost algorithm results show longitude, bathrooms and neighbourhood frequency as the top three features with feature importance scores of 53, 29, and 25 respectively. For all of the models, we have calculated the root mean squared error, they were 461.75 for Linear Regression, 452.76 for CART, 388.82 for Random Forest and 457.77 for XGBoost.

• Observation and Conclusion

From Feature importance, we interpreted that the location of the airbnb and size of Airbnb are influential features for price selection. Longitude, Latitude and neighbourhood_frequency which scored the highest in most of the algorithms are the features about the location of the airbnb. In addition, room_type, accommodates, bathrooms, bedrooms and beds are features which are related to the size of the airbnb. We could also check that the features related to the size of the airbnb had correlation between price from the correlation matrix heatmap. Longitude ,latitude and neighbourhood_frequency did not show high correlation on correlation matrix heatmap. It means that Longitude and latitude do not have a simple linear relation between price.

From the RMSE scores, we see that the Random Forest model performed the best with a score of 388.82. The scores for the rest of the algorithms are more than 450, so it is significantly worse than Random Forest. Overall the RMSE scores are all very high and we attribute this to the complexity of predicting price with the given features. To have more accurate model results, we need more meaningful features in our input variables.

• References

Quattrone, G., Greatorex, A., Quercia, D. et al. Analyzing and predicting the spatial penetration of Airbnb in U.S. cities. EPJ Data Sci. 7, 31 (2018).
https://doi.org/10.1140/epjds/s13688-018-0156-6
https://epjdatascience.springeropen.com/articles/10.1140/epjds/s13688-018-0156-6

Villar, Brunna. A Financial Data Analysis of Airbnb. Towards Data Science. May 21, 2020.
https://towardsdatascience.com/the-future-of-airbnb-what-the-data-can-tell-us-178e2f227753

Laurae, Visiting: Categorical Features and Encoding in Decision Trees,Apr 24, 2017

https://medium.com/data-design/visiting-categorical-features-and-encoding-in-decision-trees-53400fa65931

Manish Pathak,Using XGBoost in Python,November 9th, 2019

https://www.datacamp.com/community/tutorials/xgboost-in-python

Harshit, Gradient boosted trees: Better than random forest?, February 23, 2018

https://kharshit.github.io/blog/2018/02/23/gradient-boosted-trees-better-than-random-forest