Draft version July 8, 2020
Typeset using LaTeX default style in AASTeX63

Usefulness of Machine Learning with Airbnb data

Pouya Lajevardi[1]

[1] *University of Toronto Scarborough*
1265 Military Trail
Toronto, ON. Canada

ABSTRACT

The project attempted to find any correlation between features of the Airbnb listing and their respective prices. If machines can learn to find correlations and predict prices, many insights can be gained from the listing data sets available for scraping on Airbnb. These insights can be illuminating for both individuals as well as industry. The project mainly uses sklean packages and keras for data pre-processing, training and predicting. The study took a broad look into many popular methods of regressions such as Random Forest and Decision Tree as well as XGBoost and Artificial Neural Networks. Further an analysis was made on some classification methods.
Ultimately the results showed that with proper data pre-processing and tuning of the methods, many such correlations can be found a model with relatively accurate predications can be built. In turn for answering many questions, many other questions arose which requires further contemplation such as, whether or not better and more accurate results can be achieved and what are better ways of handling outliers.

1. INTRODUCTION

Airbnb data has always been readily available to be scraped, looked at and investigated. However, it has never been particularly useful and readable by humans. Machine learning methods excel in such area with plenty of data including many variables.

Here is where the machines are speculated to be useful. One of the main challenges when it comes to data sets scraped from Airbnb is to make them machine readable. Aside from the fact that much of what is provided includes empty fields,[1] there are many fields that require some changes to be useful to machines such as text based columns and columns including date format.

One of the reasons that gathering insights into the data set was deemed useful was due to the fact that without any insights pricing would merely be based on hunches and guesses. On the other hand finding which features are heavily correlated with prices, can help hosts optimize for a higher listing price while providing amenities that are presumably desirable for their guests.

Many such features such as location might not be modifiable for the host, nevertheless this could provide insight as to whether or not a property would be financially feasible for Airbnb listing consideration. One can see if these insights could be derived from the data using machine learning tools and methods, many benefits can be gained.

All of this hinges on whether there can be any useful insights derived from such data set. We have investigated the aforementioned question on a data set scraped on November 7th, 2018 by Murray Cox on Berlin listings.

2. METHODS

By taking a look at the data set it was immediately apparent that the data set required data pre-processing. This involved removing many columns including irrelevant data or columns that contained many NaN fields, replacing the few NaN fields with either the median or other relevant value. Another such data pre-processing involved text and

[1] referred to in machine learning packages such as pandas as NaN

date column parsing. Namely for host since and amenities columns.

In the beginning majority of the regression methods were widely considered considered. Including but not limited to linear regressions, random forest, decision tree, KNN and XGBoost. ANN was also considered as either a regressor or classifier.

The other matter seemed to be the choice of of features. For a good starting point, we took a look at others' work in the area. Granted they were not on the same data set and the differences in the data sets can lead to differences in results. For this purpose we took a look around and chose to two specific articles to delve deeper into. Namely LewisLewis (2019)' "Predicting Airbnb prices with machine learning and deep learning" article and Perez-SanchezPerez-Sanchez et al. (2018)'s "The What, Where, and Why of Airbnb Price Determinants".

In each case the choice of highly correlated features with the nightly price had many overlaps. It is noteworthy to mention that each of these papers evaluated various different data sets. Lewis' work mainly addressed data set based in London whereas Perez-Sanchez's work took a look at data gathered in various cities in the province of Valencia in Spain and some other data sets.

Since the features seemed highly correlated in both research we made the choice of using top 10 overlapping features. Some the highest correlated ones included maximum number of guests, cleaning fee, cost per extra people and availability in the next 90 days.

The main focus was put to evaluating the effect of clustering certain features into groups and add such groups to feature space. Then make use of the regressors to evaluate performance.

The choice of ANNs were highly contemplated and briefly implemented however due to less than mediocre result prior to tuning it was not pursued any further. Due to their justly reputable performance, ANNs do seem appropriate for further investigations.

2.1. *Decision Tree Regressor*

Decision Tree uses the tree structure to build the model which further breaks down the model into smaller subsets. For the purposes of this study sklearn's Decision Tree Regressor algorithm was used.

2.2. *Linear Regression*

Linear regression and multiple linear regression[2] uses a linear way of modeling the data. Given the set of independent variables and the dependent variable, this methods attempt to find the best hyper surface fit to the data and further for prediction depending on what the independent variable values are read at what value the dependent values is placed.

2.3. *Artificial Neural Networks*

Artificial Neural Networks have been introduced back in the 20th century but abandoned at the time due to limited computational resources. However in the early 21st century they were revisited.

The Artificial Neural Networks are loosely based on biological Neural Networks that are a vast sea of connected neurons. The Artificial Neural Networks consist of input layer, the independent variables, the hidden layers and the output layer, the dependent variables. The nodes in the input layer have output to the neurons in the hidden layer which in turn have outputs to deeper hidden layers and eventually to the output layer. The depth of the hidden layer is often determined by the complexity of the task at hand the number of independent variables (number of input neurons). It is worth nothing that Deep Learning comes from the fact that Artificial Neural Networks tend to have

[2] multiple linear regression uses multiple variables in the regression model

relatively deep hidden layer. The deeper the hidden layer the "deeper" the learning.

In this architecture each node connection starts with a low weight. Each node in the hidden layer is either activated at each moment, strengthening the connection by increasing its weight, or not activated. Ultimately this entire process constitutes training of the model. Lastly when called upon for prediction, the model, based on the trained weighted nodes, attempts to make a prediction about the dependent variable.

2.4. *K-Nearest-Neighbours*

KNN takes the inputs and identify K number of closest samples in the feature variables and the output would be the average of said neighbors. KNNs are generally based on Euclidean distances.

2.5. *Random Forest Regression*

Random Forest Regression (and classification) methods belong to the ensemble methods which use multiple methods to increase the predictive power of the method. The method was initially proposed in 1995.

Random Forest consists of Bootstrap Aggregation, commonly known as bagging, and Decision Trees. For the purposes of the this project the sklearn's Random Forest Regression Algorithm was used.

2.6. *XGBoost*

XGBoost is an open source project that aims to create a scalable and distributed gradient boosting algorithm.

Gradient boosting algorithm likewise is an ensemble of prediction methods such as decision trees and attempts to improve performance by optimizing a loss function.

3. RESULTS

As mentioned in the methods section, the training started with various regression models and set of about ten features initially. In this step based on performance of the best performing models, some features were selectively added that seemed to be positively affecting most relatively well performing models.

Initial choice of the features provided better than random results based on the consistency of the R^2 score remaining positive as a result of cross validation. However, these results were far from satisfying in the end which led us to expand our choice of features to many more including but not limited to features such as bathrooms, accommodates, cleaning fee, bedrooms, beds, minimum nights, security deposit, extra people, number of reviews, reviews score location, review score rating, 90 days availability, host since and location were ultimately chosen to be the most prominent ones.

Majority of the methods introduced in the method section did not have bright performance. Performances were measured using R^2 MSE and MAE scoring. However ultimately the MSE scoring was used to measure performance. With respect to all measures, majority of the models performed less than adequately. These involved Decision Tree algorithm, KNN and linear regression. Narrowing down the choice of methods to Artificial Nueral Networks, Random Forest Regression and XGBoost.

Moreover property type clustering and classifiers were required. After examination few clustering methods such as KMeans and HDBSCAN and visualizing on the location parameters the choice went to KMeans. Since the clustering was meant for creation a new type of feature, the possibility of noise was not desired which ruled out HDBSCAN very early on.

Further for the classifiers, Decision Tree Classifier, Random Forest Classifier and KNN were examined. Both Decision Tree and Random Forest had high accuracy for classification. However, Decision Tree had a slightly better

performance and computed quicker.

At this point with the right choice of feature space as well as few candidate methods, testing, tuning and evaluating started. The first glaring observation was the improvement in performance of all methods with the introduction of the cluster features.

More tuning of the models led to even better performance in all of our models. Fine tuning had a particularly drastic effect on the Artificial Neural Network model.

Orders of performance were as follows:

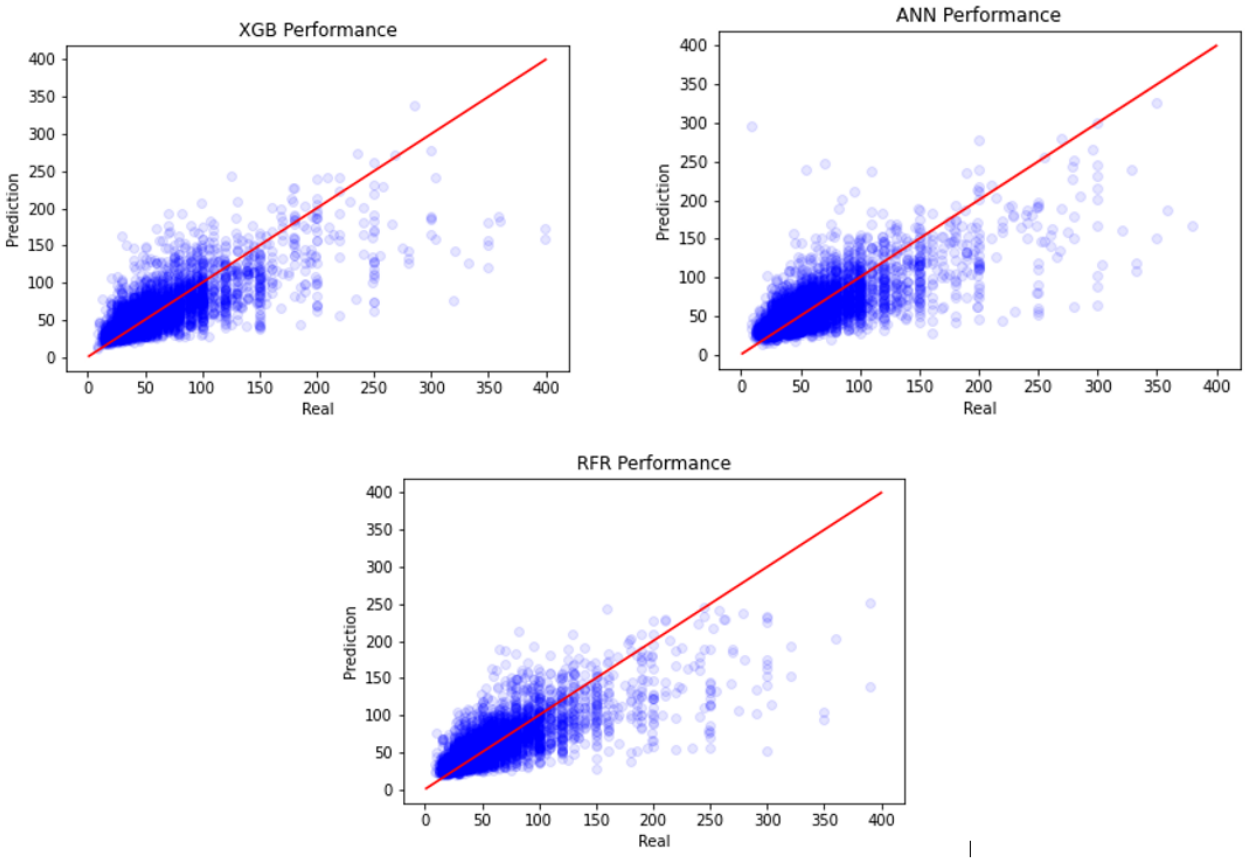

Figure 1. Model Performance with the red line being the prediction line. The farther from the red line the points are the higher the error for that prediction. The distance is based on Euclidean geodesic (The length of the shortest path from the point to the line)

$$ANN = XGBoost > RFR$$

The graphs seems to show very similar performances. And the performances indeed are close but, with MSE as a measure of performance, ANN and XGBoost perform at the same level within the margin of error and RFR's performance is slighter worse.

3.1. *Feature Importance*

The importance of our data with XGBoost and Random Forest Regression has been examined and the results were as followed:

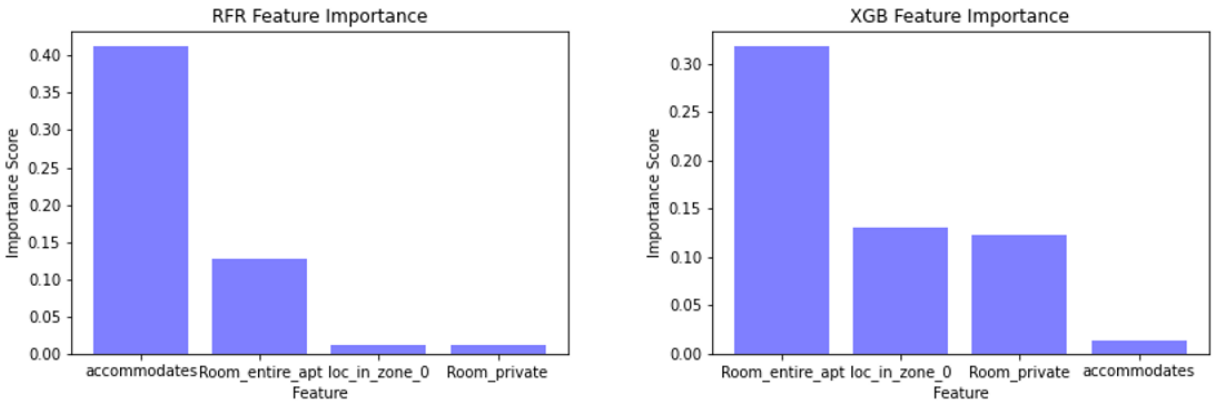

Figure 2. Importance score of the 4 most correlated features using XGB

As we can see in these graphs the top 4 features of importance in both models, XGB and RFR, were the same features. Namely, the entire room, Location in zone 0, private room and maximum number of guests. However, their respective scores and importance in different models are quite different.

On the other hand feature importance in an ANN model could not have been assessed as easily. Various elimination and trial and error methods were tried to narrow down the top most important features, however their relative importance remained unanswered. The top features as shown in the graph above, had high impacts on the ANN model as well.

3.2. *Outliers*

During the examination using cross validation for analyzing model performance, it became quite apparent that the performance of the model was abundantly dependent on how the data set is split into test and training data. The initial hypothesis was that there might be outliers and depending how they are distributed among the test and training sets.

With much trail and error it seemed that limiting the data to those with maximum of $400 price tag and the minimum of $8 price per night improves the performance and lowers the MSE. All the discussion around methods will be based on the aforementioned removal of the outliers.

4. DISCUSSION

The discussion was broken down to Features, feature engineering and importance and Methods and comparison drawn between them.

4.1. *Features*

When it comes to the features and their result, the outcomes are not too surprising. Especially when they are compared to the data assessed in other research work as discussed.

On the other hand when considering matters intuitively, features such as whether the property was the entire unit or a private room should have been an important feature which our model indeed confirmed. Other features such as the location of the property, perhaps its proximity to key landmarks, as well as the size of the property should also be determining factors. However, not considering maximum number of individuals accommodated as one of the features one might have considered the square feet or number of bathrooms as the determining factor. Although number of bathrooms and bedrooms proved important, they did not make the cut for the top 4 for either of the models.

On the other hand the square feet feature had logistical issues. Namely more than half of the entries (properties) did not have any information in their square feet feature column ergo leading to the column not being considered as one of the features. Since more than half of the entries needed to be guessed there were two options to consider:

- Attempt to predict the size of the unit in square feet and use the predicted column as a feature itself.
- Ignore the column and do not use it as a feature.

In this case the decision was made to not consider the square feet as a feature due to the fact that about 5% of the entries contained any information in that regard.

Moreover, the choice of introducing engineered features such as property type cluster and location clusters were contemplated given merely a hunch that a certain cluster of properties that share the same proximity to certain landmarks and attract more costumers ergo higher demands should have similarly higher than average price range for a property with otherwise very similar features. The property type clustering was mainly based on trying to narrow and cluster properties based on the features that seemed to put the listings very close to each other as a "type" such as how large they are and other factors. The choice for this engineered feature too was based on a hunch.

The features used to engineer the property type clusters were bathrooms, accommodates and cleaning fee. Initially the property type (apartment, house, etc.) was used among these features however with further investigation, it was realized that properties types are by enlarge apartments and for this data set and others similar, this feature was deemed unnecessary. The methods proved this assumption correct.

For the location cluster, merely longitude and latitude were used. One such clustering is displayed in the figure below.

4.2. *Methods*

As displayed in the results, Artificial Neural Networks and XGB regression models performed very well and quite close to each other[3].

For testing and training purposes, the data set was split into 30% test set and the rest training set. Both ANN and XGB's MSE averaged at 721 and 705 respectively. With standard deviation for ANN being lower at about 40 and XGB at 75. The ANN seems to have more consistent performance regardless of the split of the data and test sets.

Random Forest Regression model performed decent however as described in the results not as well as the other two. The average MSE score for RFR was 790 with standard deviation of 116. Not only RFR had worse performance than ANN and XGB, it clearly[4] was more dependent on the split of the test and training set.

With the taking all of that into account, the decision was made to put RFR aside and keep on fine tuning the ANN and XGB methods.

[3] For model performance comparison in this report MSE (Mean Squared Error) is considered and used

[4] Due to its higher spread

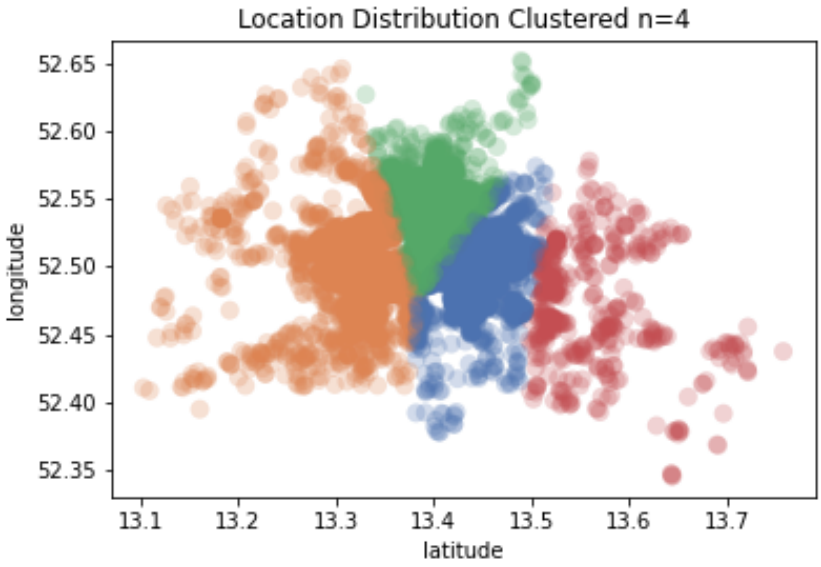

Figure 3. Location Distribution with 4 clusters

With further tuning of XGB and ANN we came to get very similar performance results and that concluded the performance analysis of the project. On the other hand however, there was another consideration to be made and that was complexity of the algorithm and how long it would take to train the model. With the tuning that were made both of them became a bit time consuming though not at all by the standards by which models are often trained in very large data sets these days. In either case it was a matter of minutes to train the model on the training set. However, if the same is done in a much larger data set[5], this can scale up.

It seemed that given our data set and parameters that were decided for each model after tuning, the ANN took an average of about 1.28 times as longer as XGB for training. This is can be a consideration to be made on a much larger data set (perhaps of multiple cities of countries).

5. CONCLUSION/SUMMARY

The methods and models are far from perfect but it became quite apparent that there are strong correlation between some of our features and the price. The fact that predictions could be done with such relatively low MSE score, suggest much can be done to find patterns of sorts in these data sets.

The data set is far from perfect. After all this is merely an snapshot of a certain moment in history of the Berlin market and is not alive. This paper was first drafted during the COVID-19 pandemic. One can only imagine what would the data represent if this snapshot was taken during this crisis and used without context.

The investigation itself was far from perfect and much more improvements can be made. From dealing with outliers better to engineering more features that can be useful the possibilities are plenty. Nevertheless the project managed to answer the very question it started to address, whether or not any strong correlation can be found between any or some of the features and the pricing of the listings. In short that answer was yes.

We hope that this work inspires others to take this work further. Perhaps with better choices of features, maybe instead of taking one snapshot, take many to move the model from a picture into a a motion picture describing the

[5] The entire data set at hand to work with was about 18000 entries

Airbnb market.

REFERENCES

Lewis, L. 2019, Predicting Airbnb prices with machine learning and deep learning

Perez-Sanchez, R., Serrano-Estrada , L., Marti, P., & Mora-Garcia, R. 2018, Sustainability, 10, 4596, doi: 10.3390/su10124596