AirBnB Price Prediction using Machine Learning Techniques

Amrutha Ukkalam[1], Ananth Rastogi[2], Gurusankar Kasivinyagam[3], Mahesh D[4]
[1, 2, 3, 4] Computer Science and Engineering, PES University, Bangalore

***Abstract*—the ability to predict prices and features affecting the appraisal of property can be a powerful tool in such a cash intensive market for a lessor. Moreover, a predictor which predicts the number of reviews a particular listing will receive would be useful in exploring factors which determine the popularity of a particular property. The problem consists of 2 aspects - identify factors determining the cost of a listing and customer satisfaction of around 50000 Airbnb properties listed in the city of New York and to explore the factors which determine the number of reviews and customer popularity of the properties. The problem requires us to build models that correctly predict the cost and number of reviews of a listed property with high confidence. In this paper, we discuss the methods explored to solve the problem and our findings on the effectiveness of each of the models in solving this particular problem.**

Keywords*— Linear regression, Stochastic Gradient Regression (SGD), Bayesian Ridge Regression, Automatic Relevance Determination regression (ARD), Passive-Aggressive regression, Theil-Sen Regressor, Lasso Regression and Random Forest Regression

I. Introduction

Airbnb is a peer-peer online website which allows hosts with spare homes to rent them to the guests who are seeking accommodation in that location. It's vital for hosts to set an optimal price for their listings on AirBnb because it attracts more customers, improves customer satisfaction, increases the flow of customers and improves the revenue for the hosts. All of these are a direct consequence of better matching of demand with supply and thereby, directly impacts the host's turnover.

The number of reviews attracted by a property listing can be another factor that can not only be used to determine the potential popularity of a property, but can also be used to explore and analyze the factors which result in fewer reviews for certain properties. This can be useful in determining how to improve the average number of reviews received on property listings and thereby helping the hosts improve their revenue and making the platform more competitive against its rivals.

II. Previous Work

Airbnb is an online lodging service for sharing homes and experiences, which help guests, find an accommodation according to the features they are looking for. It is a two-sided market place where price of the housing is an important factor which ensures the balance of the market's supply and demand.

There has been a lot of work in exploring the various factors that help in determining the price of a listing on the website accurately as it determines the popularity of the place.[1] Airbnb doesn't control the pricing of listings of their hosts. This is a major drawback because if the price for a property isn't set optimally, the properties will not be occupied and the hosts will incur losses leading to them choosing other means to rent out their property. This could cause the company to lose its customers and the company's revenue will take a direct hit as the consequence. There are various factors that affect a listing price like the location, seasonality, type of listing, distance from the landmarks, and the popularity of the nearest landmarks.[1] It gives an insight into the dynamism involved in the responsible factors and the challenges that it poses in estimating the accurate demand curve which is required for estimating the optimal price of the listings that will attract people and maximizing the revenue for the hosts.

Airbnb not only aims at creating profits for the hosts but also at providing the customers a comfortable experience. Customer satisfaction plays a vital role in determining the reviews for a property and in the validation of the goodness of a property. If the customers are not happy with their experience, they might give bad reviews which lower the listing's demand and in-turn affects the host income from that listing. For example, the importance of having pictures of the listings is underestimated in gaining the trust of the customers.[2] It states that photos posted by the hosts of their properties has had a high effect on travelers' decision-making process even more than price as an important factor in this process.

The experimental results of a range of regression models were tried on the Airbnb dataset to develop a reasonable pricing model. [3] It emphasizes on the feature selection, using neural networks for prediction and leveraging the customer reviews through sentimental analysis in improving the prediction model.

Our literature review involved looking at potential regressors that have been used in attempt of solving a similar problem – important findings are listed below.

1. A simple linear regression model optimizes the linear combination of the predictors, but most real world problems are unfortunately not so simple because of the highly interconnected nature of the observed variables.
2. Bayesian Ridge regression (Linear Regression with L2 regularization) adds a penalizing term to the squared error cost function. It has been noted to perform well in datasets which were observed to have high variance and reduce overfitting, but is known to converge well only for linearly separable data [2].
3. Although Random Forest Regression has a high accuracy, and the ability to handle large number of features, tuning the hyperparameters has always been a challenge. It has been noted that the maximum number of features chosen randomly at each node is optimal when it is equal to the log base two of the total number of features. Although increasing the number of features reduces overfitting, it increases computation [4].

In addition to the above methods, we are also implementing a few more regression models to compare the overall effectiveness of each model against the others, and improve on the aspects of previous findings:

1. Stochastic Gradient Descent (SGD) – SGD linear regression uses SGD to determine the coefficients. It is a very simple yet efficient approach to minimizing the cost functions but is limited to capturing relationships ranging from simple to moderate complexities.
2. Automatic Relevance Determination (ARD) Regression - ARD is also known as Sparse Bayesian Learning and uses Bayesian interpolation to minimize overfitting by using the idea that 'high precision features are to have weights close to 0, and thus pruned'. It works closely in accordance with the idea postulated by Occam's razor.
3. Passive-Aggressive regression - Passive-aggressive algorithms are family of machine learning algorithms similar to the perceptron but usually used for large-scale learning (updates model step-by-step as opposed to batch learning). They do not require a learning rate, but do include a regularization parameter. If the prediction by the model is correct, the model remains unchanged (passive) but if the prediction is incorrect, the model is penalized in accordance with the regularization parameter.

   This model is compared to observe its effectiveness in hopes of scaling this particular model to the entire AirBnb dataset.
4. Theil-Sen Regressor - A more-accurate estimator than simple linear regression for skewed and heteroscedastic data, this regression model chooses the median of the slopes of all lines through different pairs of points. This particular model is equally good in comparison to least squares method even for normally distributed data.
5. Lasso Regression – It is a type of linear regression that uses shrinkage, i.e., where the values are shrunk towards a central point. It encourages simple models determined by fewer parameters by placing penalties on attributes with low correlation.

These particular models were chosen while keeping in mind the family of algorithms each of these algorithms belong in to expand our search for the best prediction algorithm for this particular problem. Linear regression is used as a yardstick, as a base measure, L1 and L2 regularizations as good estimates of linear regression predictors, Passive-aggressive belong to a family of online algorithms, stochastic gradient descent is a batch processing algorithm, Theil-Sen regression as the "most popular nonparametric technique for estimating a linear trend" and the random forest regression from the family of decision trees.

We are assuming that the data need not be processed real-time.

III. Proposed Solution and Implementation

The general overview of the proposed is as follows:

1. Data is pre-processed using insights from EDA.
2. Each of the above discussed models is trained for price prediction.
3. The models are tested and compared with each other.

A. Data Pre-processing

Primary observations made during EDA that will be useful in this stage:

1. Large difference between mean and max values in the following columns:
   a. *Minimum_nights*
   b. *Number_reviews*
   c. *Price*
   d. *Listing_count*

   Upon sampling such records, it was concluded that these were consistent and reliable data records and were not outliers.

2. One-fifth of the records had NAN values in the columns of *last_review* and *number_of_reviews*. These property listings did not have any reviews.

Very few records had NAN values in the columns *name* and *host_name*. The other columns did not have any NAN values. It is observed that the column *number_of_reviews* is symmetric without considering the NAN values. This allows us to fill the NAN values with the mean without skewing the variation.

Initially, there were 15 columns. The following columns were dropped as they do not contribute to the analysis: *Host_name*, *Last_review*, *Latitude* and *Longitude*. Then, the column *name*, the string used to introduce the property listing to the customer, was replaced by *name_length*, the length of the same string. Then, there were 11 columns.

Following this, the data is categorized into categorical and numerical columns. The categorical columns, *neighbourhood_group*, *neighbourhood* and *room_type*, encoded using one-hot encoding. One-hot encoding is used to prevent the model from capturing any ordinal relationships between the categories in the case they were ranked, thereby improving accuracy [5]. Finally, we are left with 232 columns.

Other worthwhile observations:

1. Number of reviews is inversely correlated to the price of the property listing.
2. Fewer reviews on shared rooms compared to having the whole property.
3. Number of reviews is inversely correlated to the minimum number of nights to be booked.

B. *Model Training*

First, we train a price-prediction model. The data set is split into 70-30 for training and testing using sklearn. Each model's effectiveness is compared using their R^2 values and Mean Square Errors (MSE). The results are shown in table 1.1.

Similarly, the model for prediction of the number of reviews are created and is shown in table 1.3.

C. *Experimental results and Explanation*

	Regression Model	R^2 values	MSE
1.	Linear	0.5614	35.8559
2.	Lasso (L1)	0.5698	35.8175
3.	Ridge (L2)	0.5624	35.8143
4.	SGD	-8.3107	4.9358
5.	Passive-Aggressive	-3.2140	111.144
6.	Theil-Sen	0.5592	35.9442
7.	Random Forest*	0.9429	35.2313
8.	RF2 model	0.8197	35.0626
9.	Random Forest$^\$$	0.879	34.6929

Table 1.1 – Comparing R^2 values and MSE for price prediction

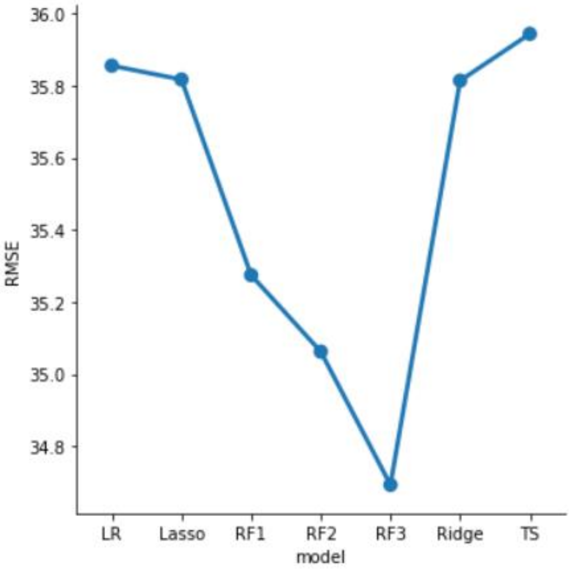

Fig 1.1 – RMSE vs model for Price Prediction

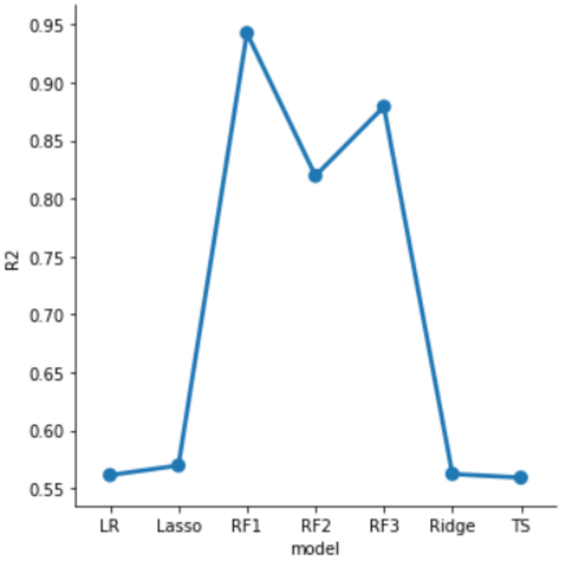

Fig 1.2 – R^2 vs model for Price Prediction

It's observed that L1 regularization has a slighlty higher R^2 value than Simple linear regression and L2, regularization has a significant fall in R^2 value. As L1 regularization penalies the less important features towards removing their influence and promotes feature selection, this suggests that our data shows high correlation among certian features.

This is further supported by the drop in L2 regulariations R^2 value as this model is better equipped to handling a large number of parameters exerting an equal influence on the result.

From the values we observe that Random Forests have the best R^2 values among all the models being compared. The two two random forests are different in terms of the hyperparameters passed to them and are shown in Table 1.2.

Random Forest* was observed to over-fit the data as it showed a very high R^2 value. So, for the model built, the hyperparameters were tuned using Randomized Grid Search. Random Forest$^\$$ parameters are the final set of hyperparameters used. The initial and final set of parameters that were used is shown in Table 1.2.

	Random Forest*	Random Forest$^\$$
N_estimators	300	600
Max_depth	None	90
Min_samples_split	2	10
Min_samples_leaf	1	1
Max_features	auto	sqrt
Bootstrap	True	False

Table 1.2 – Random Forest Hyperparameters before and after Randomized Grid Search

In the case of SGD regression and Passive-aggressive Regression, we observe negative R^2 values. A negative value is not mathematically impossible. It simply shows that the constructed model, specifically the best fit line constructed by the model, fits worse than a horizontal line. A representation of this is shown in Figure 1.1. SGD and Passive-Aggressive regression perform poorly on this dataset. It's likely that the hyperparameters set for the former are required to be tweaked and for the latter that the dataset is quite small.

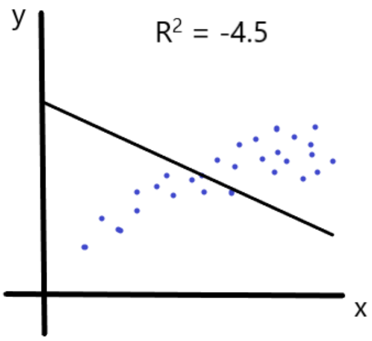

Figure 1.1 – Illustrating a negative R^2 value

The relatively low R^2 value of Theil-Sen regression is likely to be because of improper hyperparameter tuning as it usually achieves at least the R^2 values achieved by Simple Linear Regression. Irrespective, in terms of statistical power (probability of avoiding a type II error) it will perform quite similar to Simple Linear Regression.

Nevertheless, the best performing model is the random forest by a significant margin. It would prove to be futile to improve the R^2 values of the other models unless any of the assumptions change.

Similarly, the number of reviews prediction is done similarly and the results are shown in Table 1.3. The hyperparameter tuning for the Random Forests are kept the same as described earlier.

	Regression Model	R^2 values	MSE
1.	Linear	0.4079	34.4418
2.	Lasso (L1)	0.4674	34.3734
3.	Ridge (L2)	0.4107	
4.	SGD	-4.7203	9.7246
5.	Passive-Aggressive	-0.0539	45.9505
6.	Theil-Sen	0.4033	34.5756
7.	Random Forest* (RF1)	0.9555	26.4313
8.	Random Forest$^\$$ (RF3)	0.8707	26.7455

Table 1.3 – Comparing R^2 values and MSE for number of reviews.

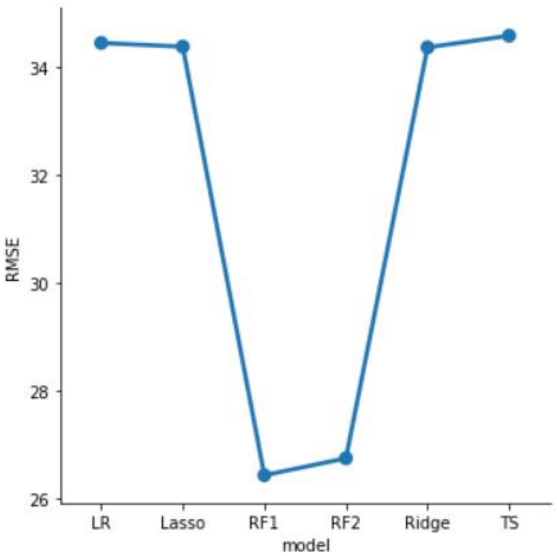

Fig 1.3 – RMSE vs model for prediction of number of Reviews

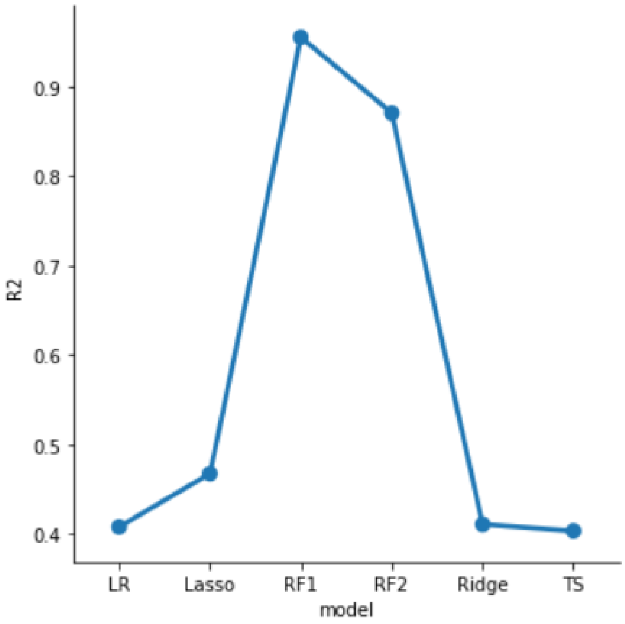

Fig 1.4 – R^2 vs model for prediction of number of Reviews

Unlike R^2 error, MSE is an absolute error. You can observe that despite the large change in R-square values between the Random Forests before tuning and after tuning, the MSE is small. But it is to be noted that the MSE has indeed changed.

D. Evaluation

RF2 in Figure 1.1 and 1.2 uses the following hyperparameters which were estimated through observation.

	Random Forest*
N_estimators	200
Max_depth	50
Min_samples_split	5
Min_samples_leaf	4
Max_features	Auto
Bootstrap	True

Table 1.4 – Random Forest (RF2) Hyperparameters estimate through observation

	R^2	RMSE
RF2 model	0.8197	35.0626

Table 1.5 - RF2 R^2 values and RMSE values MSE for price prediction

	R^2	RMSE
RF2 model	0.8707	26.7455

Table 1.6 - RF2 R^2 values and RMSE values MSE for number of reviews.

From the Pre-processing of data and the relative performances of the different regression models, we can say that random forests show the best performance owing to its ability to handle a large dataset with higher dimensionality and cross-validation which provides higher accuracy. It’s worth noting that care must be taken to avoid over-fitting of the data as was observed in our experiments.

One drawback would be the dimensionality of the dataset, which increased exponentially from 16 attributes to 232 attributes due to one-hot encoding. As it stands, the space complexity of the given dataset is 232*48,000, which is already very large despite the relatively small dataset. As we scale the model, it will become infeasible to work in a similar manner due to the curse of dimensionality. One possible approach to achieve dimensionality reduction would be employing Principal Component Analysis (PCA) at the preprocessing stage.

IV. Summary and Conclusions

The chosen problem of price-prediction and the number of reviews can be effectively modelled through regression. Upon deeper inspection, the random forest regressor produced the best results which initially overfit the data but with hyper parameter tuning and post pruning methods, the model achieved an R^2 value of 0.879.

The data pre-processing stage involved detecting outliers, sampling outliers to manually verify the correctness of data, replacing NAN values and correcting skewness. Redundant and unnecessary columns were removed or modified to extract usable data from them.

Eight different regression models were trained and tested using a 70:30 training-testing split. Their R^2 values and MSE were used to compare each one's effectiveness against the others. Among the tested models the most effective approach is observed to be random forests with our current assumptions.

The various models judged different predictors to be the most important feature under feature analysis. But overall, we can see that tree-based learning models perform better in real-world scenarios, and in this case of price-prediction, Random Forest Regression showed the best performance.

V. Future work

In the case of larger, real-time datasets it is likely that the current model will still be able to accommodate for the changes, but may not be the best performer. There is room for further experimentation in this area. We could come up with a dynamic pricing model.

It is likely that Passive-aggressive models would perform better and with a smaller space-time complexity in such a scenario and that its performance was limited by the assumption and the size of the dataset. Polynomial Regression model can be used for price prediction as not all relationships between feature variables are linear.

References

[1] Ye, Peng & Qian, Julian & Chen, Jieying & Wu, Chen-hung & Zhou, Yitong & Mars, Spencer & Yang, Frank & Zhang, Li. (2018). Customized Regression Model for Airbnb Dynamic Pricing. 932-940. 10.1145/3219819.3219830.

[2] E. Ert, A. Fleischer, and N. Magen, “Trust and reputation in the sharing economy: The role of personal photos in Airbnb,” Tourism Management, 55, pp.62-73, 2016.

[3] Rezazadeh, Pouya & Nikolenko, Liubov & Rezaei, Hoormazd. (2019). Airbnb Price Prediction Using Machine Learning and Sentiment Analysis.

[4] Ma, Yixuan & Zhang, Zhenji & Ihler, Alexander & Pan, Baoxiang. (2018). Estimating Warehouse Rental Price using Machine Learning Techniques. International Journal of Computers Communications & Control. 13. 235-250. 10.15837/ijccc.2018.2.3034.

[5] Potdar, Kedar & Pardawala, Taher & Pai, Chinmay. (2017). A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. International Journal of Computer Applications. 175. 7-9. 10.5120/ijca2017915495.

https://github.com/maheshd20/Da_project_sem5