AirBnB Price Prediction

May 6, 2021

Contents

Introduction

Airbnb has become one of the essential elements of trips and vacation plans for over 150 million people. Since 2008, guests and hosts have used Airbnb for a unique and personalized experience of traveling with a wide range of travel possibilities. Before Airbnb, most consumers had to rely on hotels. As hotels aren't as widely available as Airbnb with their exceptional business model, they soon became the best vacation rental marketplace.
Because of the dramatic growth of Airbnb, price prediction becomes one of the essential elements for their platform. As hosts typically determine the price of the Airbnb; both the host and Airbnb need to provide a fair price to the consumer, as it is an essential element of their model. Determining the price is crucial for the new and existing host/Airbnb because the price cannot be too high that they lose popularity or not get any guest. As for customers, they have options to check and compare prices depending on their needs.
We plan to build a machine learning model using some Machine Learning algorithms like Linear Regression, Ridge Regression, Decision Tree Regression, and Random Forest Regression. Also, we would be using Machine learning techniques to tune our model and try to acheive a good predictive accuracy.

Data Preparation

The dataset used in the project has been accessed from Kaggle's database[1]. It is a public dataset of Airbnb that is accessible publicly on their original dataset website. This dataset describes the listing activity and metrics in NYC, NY for the year 2019. This data file includes all the needed information to find out more about hosts, geographical availability, necessary metrics to make predictions and draw conclusions. The dataset is in Comma-Separated Values(.csv) file format. After importing the datset, we performed some analysis on the dataset. The dataset contained 48894 rows and 16 columns.

Handling Missing Values

After importing the dataset. we noticed that column "last_review," "reviews_per_month" had 10052 missing values. Also, few columns had some missing values that were either imputed or dropped. This dataset did not contain many missing values or the rows that contained majority of missing values were not used to train or build the model.

Dealing with categorical inputs

Too many levels of a categorical variable are one of the most frequently occurring problems in predictive modeling. As for our dataset, we have three columns categories of the categorical variables they are: neighborhood_group, neighborhood and room_type are. In case of neighborhood_group and room_type, they do not have multiple levels so it is possible to use dummy coding. But in case of neighborhood, we have 219 levels which makes dummy coding overly complex. The number of dummy variables for the neighborhood variable would be one less than the number of levels in neighborhood. Since we have 219 levels in neighborhood variable it would lead to the problem of high dimensionality which will ultimately may cause over-fitting of the data. Thus, the model will not be able to generalize properly to a new dataset. Moreover, inclusion of categorical variables with too many inputs can lead to the problem of quasi-complete separation. The problem of quasi-complete separation occurs when a level of the categorical variable has a target response of either 0% or 100% and can cause the interpretation of the regression model. However, we did not include neighborhood in our final model.

Variable screening

In predictive modeling, carefully selected features can improve the accuary of the model while adding too many or too few variables can lead to overfitting or underfitting. The model can't generalize well and work on unseen dataset if the model is not "just right" or close to "just right." There are multiple methods for variable screening. We can perform many statistical tests such as AIC, BIC, F-test, Cross Validation to check what set of variables perform well on the model. There are many methods available such as Stepwise selection which uses either Forward or Backward selection method. But, we have a very small number of features, so we performed Best Subset selection method. Best subset selection method can be computationally expensive but in our case, it is possible to perform best subset.

```
Processed 792 models on 7 predictors in 10.57347297668457 seconds.
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared (uncentered):                   1.000
Model:                            OLS   Adj. R-squared (uncentered):              1.000
Method:                 Least Squares   F-statistic:                          6.721e+36
Date:                Sat, 17 Apr 2021   Prob (F-statistic):                        0.00
Time:                        00:42:23   Log-Likelihood:                     1.3582e+06
No. Observations:               42669   AIC:                                -2.716e+06
Df Residuals:                   42662   BIC:                                -2.716e+06
Df Model:                           7
Covariance Type:            nonrobust
==============================================================================
                         coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
price                  1.0000   3.13e-19    3.2e+18      0.000       1.000       1.000
minimum_nights      5.551e-16   5.64e-18     98.474      0.000    5.44e-16    5.66e-16
number_of_reviews  -5.551e-17   3.82e-19   -145.385      0.000   -5.63e-17   -5.48e-17
ng_Brooklyn                0   4.76e-17          0      1.000   -9.32e-17    9.32e-17
ng_Manhattan      -5.329e-15   5.34e-17    -99.717      0.000   -5.43e-15   -5.22e-15
ng_Staten Island   1.11e-14   1.96e-16     56.533      0.000    1.07e-14    1.15e-14
rt_Private room    2.665e-15   3.31e-17     80.446      0.000     2.6e-15    2.73e-15
==============================================================================
Omnibus:                    22258.986   Durbin-Watson:                   1.844
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           448010.773
Skew:                           2.064   Prob(JB):                         0.00
Kurtosis:                      18.328   Cond. No.                     1.38e+03
==============================================================================
```

Figure 1: Best Subset Regression Result

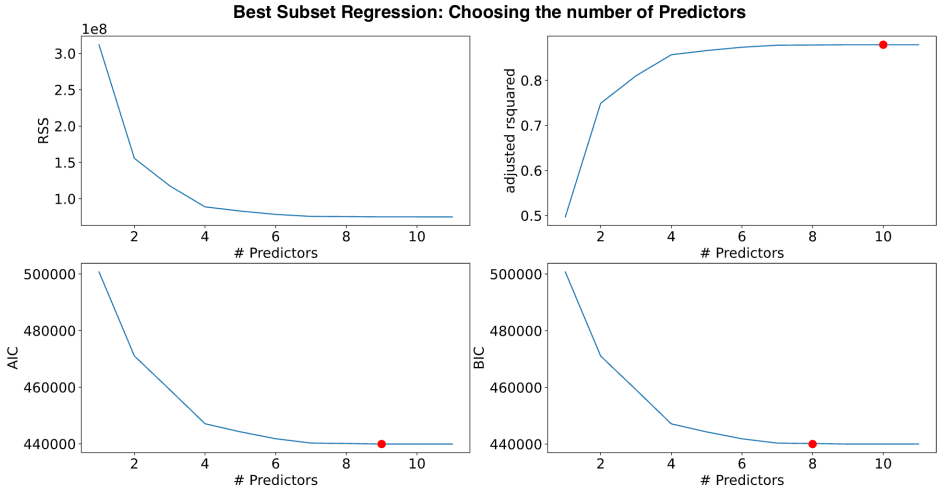

Figure 2: Best Subset Regression: Choosing number of predictors

Models Used

Linear Regression

Linear Regression is a technique that is widely used in Statistics, Statistical Learning, and also in Machine Learning. Linear Regression Model focused on modeling the relationship between observed variables. A simple linear regression model fit a line between the observation of two variables, that line is also called "line of best fit." The task is to draw "line of best fit" or "closest" to the points (x_i,y_i), where x_i and y_i are observations of the two variables which are expected to depend linearly on each other. [2]

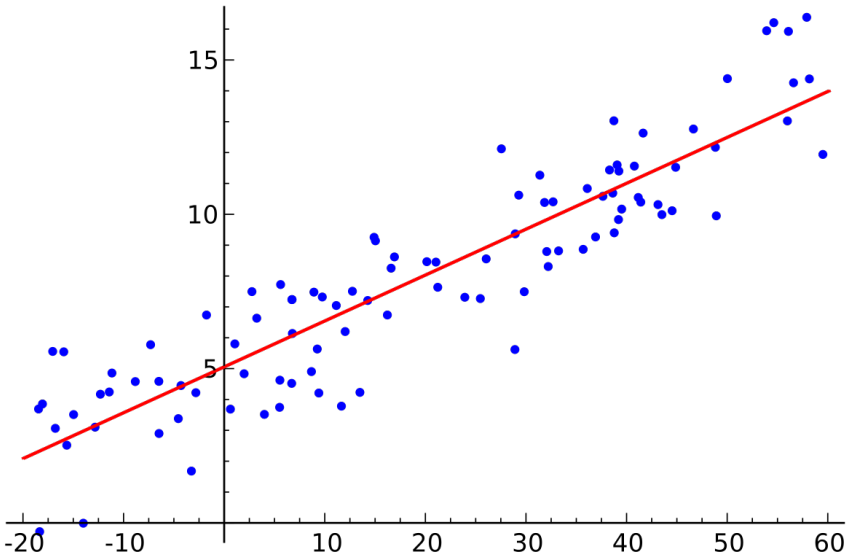

Figure 3: Simple Linear Regression from [3]

Ridge Regression

Ridge Regression is also a technique that is widely used in Statistics, Statistical Learning, and in Machine Learning. Ridge Regression is used when regression data suffer from multicollinearity. Ridge Regression introduces some degree of bias to the regression estimates, which may prevent overfitting of data. Introducing some bias may provide a better long term prediction and reduce the standard errors.
It is hoped that results are better as the model doesn't overfit the training set. [4]

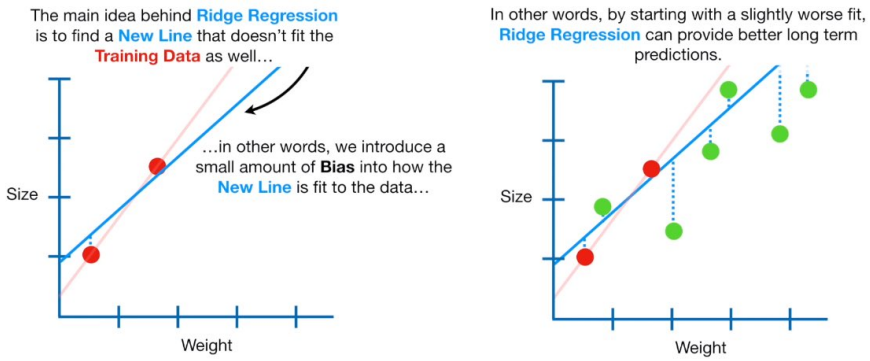

Figure 4: Ridge Regression figure from [5]

Decision Tree

Decision tree is also a model widely used in Machine Learning algorithms. Decision tree is used for both classification and Regression problems. Decision tree breaks down the dataset intro smaller subsets and simultaneously a decision tree is developed. Decision tree can be more like a flowchart, the final result is a tree with nodes and leafs. [6]

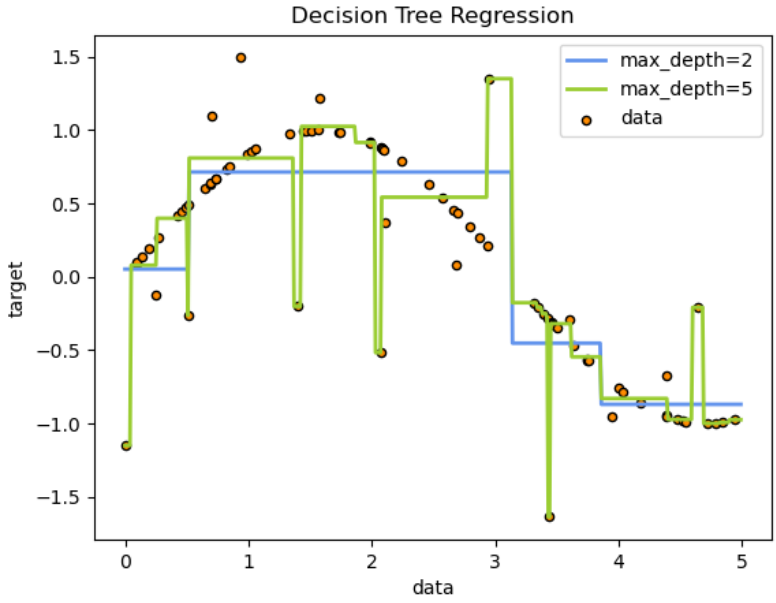

Figure 5: Decision Tree Regression from [6]

Random Forest Regression

A random forest is a model that uses multiple decision trees on various sub-samples of the dataset and uses the average of the all the decision tree results to improve the predictive accuracy while also controlling overfitiing. As the figure below shows, there are multiple deicison trees used and then the average is used as result of random forest. [7]

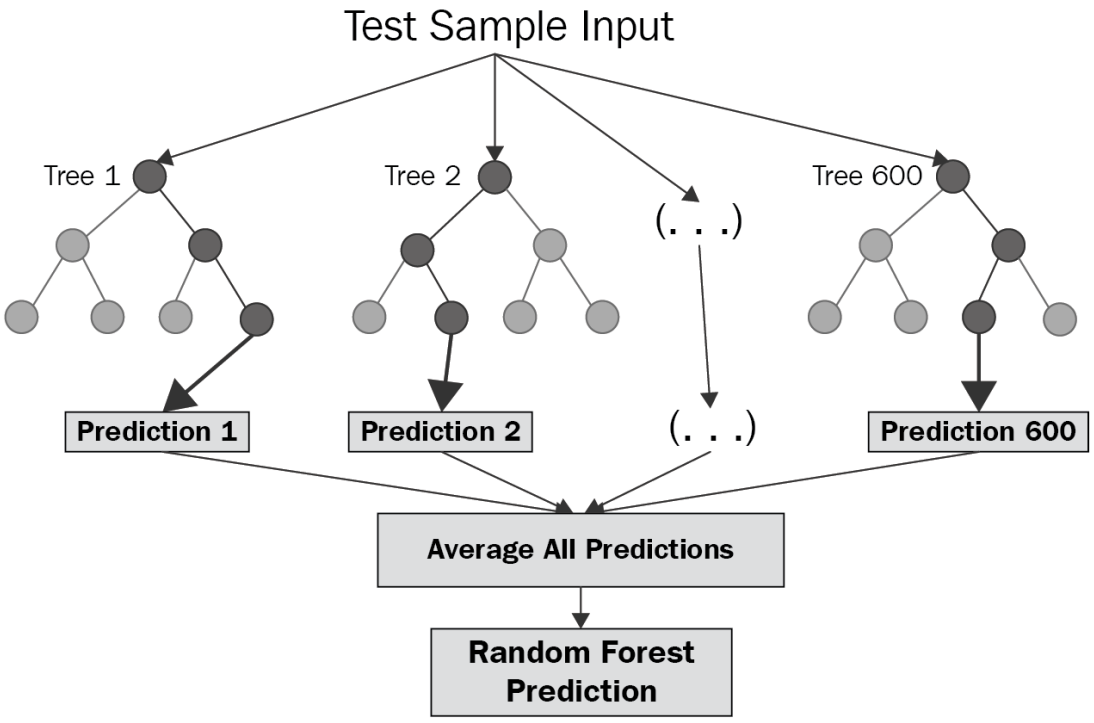

Figure 6: Random Forest Regression figure from [8]

Analysis

Exploratory Data Analysis

Correlation Analysis

A **correlation coefficient**is a value that tells if there's a linear association between two variables. The values can be anywhere from "-1" to "1." A value of "0" indicates no linear association between observed variables. A negative value indicates a negative linear association between observed variables and positive indicates a positive linear association. [9]

As, we can see from the Correlation Plot above, our features are correlated to each other. We can observe strong correlation between features. This can sometimes be an indicator for multicollinearity.

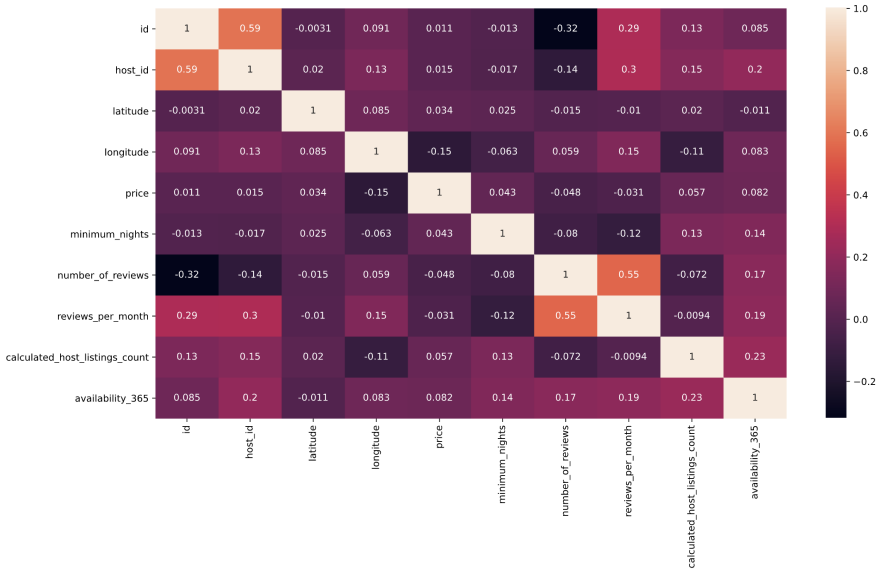

Figure 7: Results of Pearson Correlation Test on Dataset

Price Analysis

Machine Learning algorithms are typically sensitive to distribution of attribute values and range. Data outliers can mislead the training proccess and that would result us with a less accurate model and ultimately poor predictive accuracy. [10]
In order to build a good model, it is important to have a dataset that doesn't have too many outliers. So we try to detect outliers in our prices.

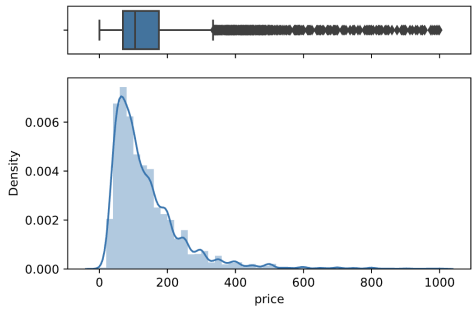

Figure 8: Density plot of price with outliers

As we can notice from the plot above, we have many outliers in our price distribution. So, we drop all the outliers from our dataset.

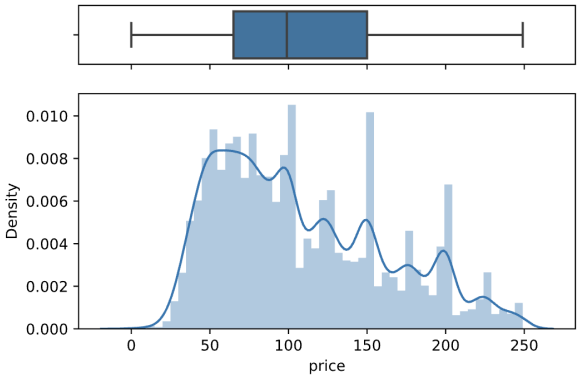

Figure 9: Density plot of price after removing outliers

Neighbourhood Group vs Price Analysis

Let's now look into the relationship between neighbourhood group and price. This would help us understand if the price variable is impacted by neighbourhood groups.

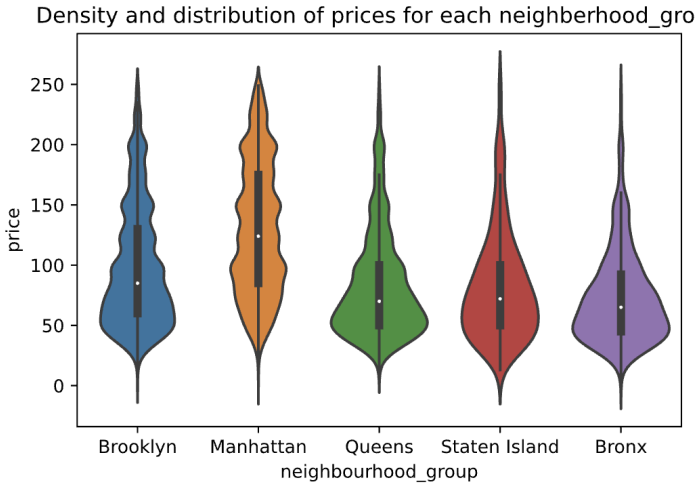

Figure 10: Violin Plot for Neighbourhood Distribution vs Price

We can notice from the violin plot above, Manhattan and Brooklyn are more expensive and distributed around the upper price ranges. Whereas Queens, Staten Island and Bronx is more distributed in the bottom price ranges. This indicates that the prices are being impacted by the loction of Airbnb.

Room Type vs Price Analysis
Now we look into the relationship between room type and price. This would help us understand if the price variable is impacted by room type.

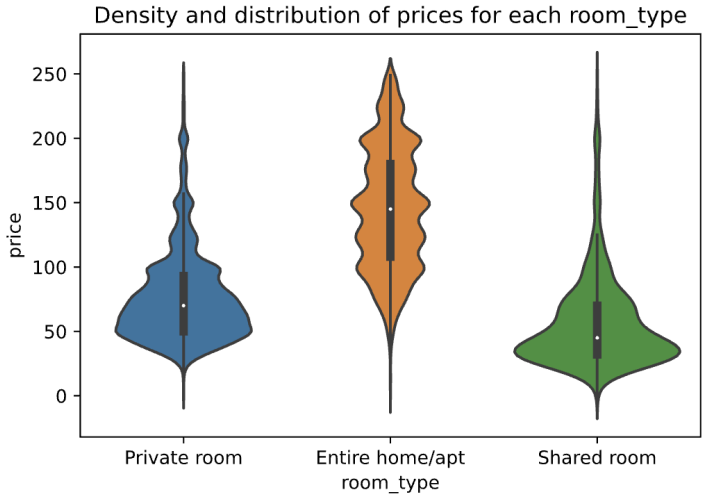

Figure 11: Violin Plot for room type vs price

As we can notice from our violin plot above, that getting an entire apartment is significantly more expensive whereas shared room is much cheaper. While Private rooms are bit more expensive than shared rooms but the major price ditribution is between shared rooms and entire apartment.

Model Optimization and Tuning Techniques

In our models, we have used multiple optimization and tuning techniques for Random Forest Regression. Before we analyze our models and results, it is important to understand these techniques. Here are the techniques that we have used to tune models using Python.

Cross Validation

“Cross validation is a model evaluation method that is better than residuals. The problem with residual evaluations is that they do not give an indication of how well the learner will do when it is asked to make new predictions for data it has not already seen. One way to overcome this problem is to not use the entire data set when training a learner. Some of the data is removed before training begins. Then when training is done, the data that was removed can be used to test the performance of the learned model on”new" data. This is the basic idea for a whole class of model evaluation methods called cross validation." [11]
There are many Cross Validation techniques, but we have used **K-fold Cross Validation**.

"**K-fold Cross Validation** is one way to improve over the holdout method. The data set is divided into k subsets, and the holdout method is repeated k times. Each time, one of the k subsets is used as the test set and the other k-1 subsets are put together to form a training set. Then the average error across all k trials is computed." [11]

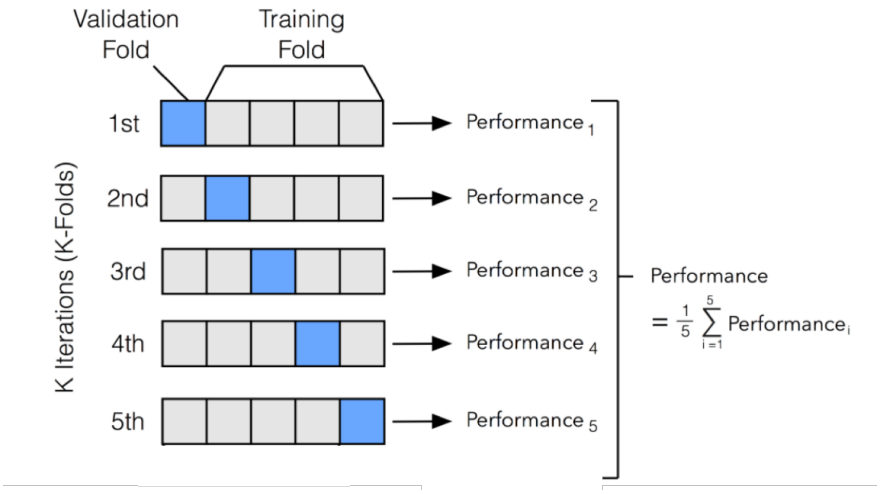

Figure 12: K-Fold Cross Validation Visualized from [12]

Bootstrapping

"Bootstrap aggregating, also called bagging (from bootstrap aggregating), is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also reduces variance and helps to avoid overfitting. Although it is usually applied to decision tree methods, it can be used with any type of method. Bagging is a special case of the model averaging approach." [13]

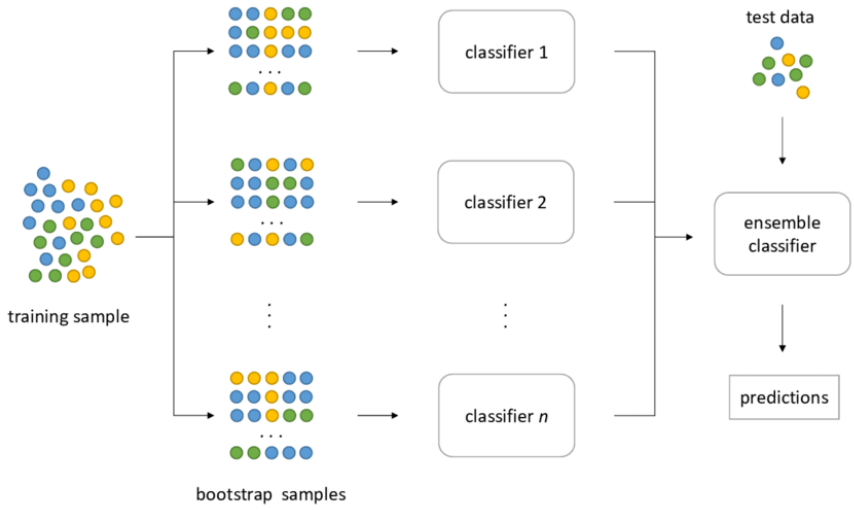

Figure 13: Bootstrap Visualized from [14]

Model Comparison and Results

I used four different Machine Learning models: Linear Regression, Ridge Regression, Decision Tree, and Random Forest Regression. And I also tuned the Random Forest Regressor model using cross-validation and bootstrap. For this model, the significant predictors were categorical variables such as neighborhood group and room type, and numeric variables like minimum nights, and host information. The results are not particularly good for predictive accuracy.

I trained my model with 70% of the dataset, and then I tested the model with the remaining 30%. I was able to achieve 55% model accuracy with a tuned random forest and the RMSE 36.40.

This dataset did not include features that are important for an Airbnb price prediction. In machine learning, the variables in the dataset must be significant predictors for the target variable. When we book an Airbnb, it is always a case that we look at the number of rooms, bathrooms, and other services included. Also, people generally case about service fees, cancellation policies. But since these features were not available in our

dataset, our model didn't perform well. In the figure below, we can see RMSE understand the performance of the model.

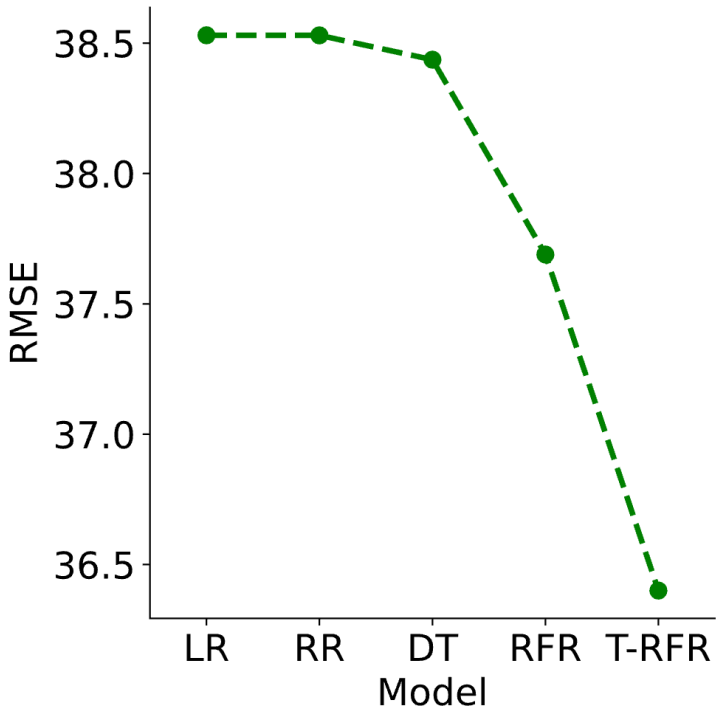

Figure 14: RMSE of all the models

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. [15]
Here is another plot that visualizes some datapoints in form of histogram to see how the Tuned-Random Forest Model performed.

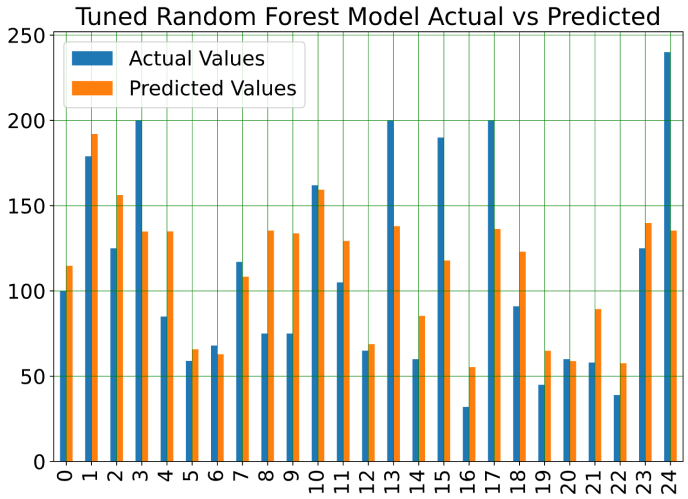

Figure 15: Tuned Random Forest Prediction Visualized

Conclusion

We worked on the Price Prediction problem and compared various Machine Learning models in this project. We did not achieve a significant predicting accuracy because of the lack of features in the dataset. While the model did not perform well, we were able to still compare multiple models and achieve a decent predictive accuracy. We also used frameworks provided by python: scikit-learn-RandomizedSearchCV to perform model tuning.

As we now know more about Airbnb, we can choose to do future work with a dataset with better features and build a model that would have better predictive accuracy. A better model can help hosts estimate their Airbnb prices. As we mentioned before, an expensive price can hurt the business of the host because people might not choose to stay if it is too expensive for what they offer, and a cheaper price might make the host lose money overall.

Bibliography

[1] Kaggle. *New York City Airbnb Open Data*, 2020.

[2] Linear regression.

[3] Wikipedia, the free encyclopedia. File:linear regression.svg, 2016. [Online; accessed April 27, 2021].

[4] *Ridge Regression*.

[5] Josh Starmer. Ridge and lasso regression. [Online; accessed May 06, 2021].

[6] *Decision Tree - Regression*.

[7] *sklearn.ensemble.RandomForestRegresso*.

[8] Chaya Bakshi. Random forest regression. [Online; accessed May 06, 2021].

[9] *Introduction to Correlation and Regression Analysis*.

[10] Mayank Tripathi. *Knowing all about Outliers in Machine Learning*, 2020.

[11] Jeff Schneider. *Cross Validation*, 1997.

[12] Github - scikit-learn. Model selection. [Online; accessed May 06, 2021].

[13] *Bootstrap aggregating*.

[14] Proskurin Oleksandr. Bagging in financial machine learning: Sequential bootstrapping. python example. [Online; accessed May 06, 2021].

[15] *RMSE: Root Mean Square Error*.