Statistics & Machine Learning

Price Prediction Model of Airbnb NYC Listings

****

# Team Members:

Aakash Shetty, Pratik Patil, Saiprasad Bahulekar, Saket Tulsan, Vaibhavi Mulay

Table of Contents

[Team Members 1](#_Toc39245357)

[Abstract 3](#_Toc39245358)

[Introduction 3](#_Toc39245359)

[Methodology 3](#_Toc39245360)

[Results 5](#_Toc39245363)

[Conclusion 5](#_Toc39245364)

[References 5](#_Toc39245365)

# Abstract

Airbnb is a paid community platform for renting and booking private accommodation founded in 2008. Airbnb allows individuals to rent all or part of their own home as extra accommodation.

The goal of our project is to analyze the listings of New York City and create a prediction model for price depending on the dependent features. The audience of our project will be the hosts and travelers who want to take maximum benefits from Airbnb platform.

We have used the New York City Airbnb Open Data which is available from Kaggle. We performed data cleaning, exploratory data analysis, statistics and machine learning on the dataset to achieve our goal. In cleaning, null and missing values were checked and data wrangling was performed. In Exploratory Data Analysis, we checked for different variables, how they were distributed and created visualizations to come up with the solution. In Statistics and machine learning, we developed a model which will be used for price prediction of the Airbnb listings.

# Introduction

Airbnb is an e-commerce platform for people to rent houses for either completely or partially from a host. Since 2008, guests and hosts have used Airbnb to expand on traveling possibilities and present a more unique, personalized way of experiencing the world. Today, Airbnb became one of a kind services that is used and recognized by the whole world. Data analysis on millions of listings provided through Airbnb are a vital factor for the company.

These millions of listings generate a lot of data - data that can be analyzed and used for business decisions, understanding of customers' and providers' (hosts) behavior and performance on the platform, guiding marketing initiatives, implementation of innovative additional services and much more. Our dataset comprised various features that an Airbnb would offer to the customer and it had the minimum nights booked feature which told us what the minimum number of nights was the listing was booked. The focus of our project was to build predictive models which can predict the price and help the customer to understand what the right price for a specific Airbnb listing was. Thus, as Airbnb is one of the best platforms for hosts to rent their space and customers to have an affordable place to stay when out of town, along with the trust of the community, this was the motivation behind our project choice.

# Methodology

**Data Cleaning**

Administratively incorrect, inconsistent data can lead to false conclusions, predictions and misdirect investments on both public and private scales. In order to avoid the mishaps, we follow the process of data cleaning.

Data cleansing or data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate [records](https://en.wikipedia.org/wiki/Storage_record) from a record set, [table](https://en.wikipedia.org/wiki/Table_(database)), or [database](https://en.wikipedia.org/wiki/Database) and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the [dirty](https://en.wikipedia.org/wiki/Dirty_data) or coarse data Data cleansing may be performed [interactively](https://en.wikipedia.org/wiki/Interactively) with [data wrangling](https://en.wikipedia.org/wiki/Data_wrangling) tools, or as [batch processing](https://en.wikipedia.org/wiki/Batch_processing) through [scripting](https://en.wikipedia.org/wiki/Script_(computing)). After cleansing, a [data set](https://en.wikipedia.org/wiki/Data_set) should be consistent with other similar data sets in the system

For our dataset, we will do operations such as getting our data into a standard format, handling null values, removing unnecessary columns or values etc.

**Data Exploratory**

Exploratory Data analysis or EDA is an approach to analyzing your dataset to summarize their characteristics often with visual methods. For the above given the dataset we have explored the attributes using appropriate graphical model. This will help us to understand the nature of our data, its behavior and so on. In the below sections we will analyze our data that with try to answer question like why, where and how the factors affect the Airbnb ratings and prices.

The following are the plots and libraries we used for EDA,

* from plotly.offline import init\_notebook\_mode, iplot to create maps according to different

prices

* import plotly.express to create scatterplot for Airbnb by Borough in NYC
* import seaborn as sns and import matplotlib.pyplot as plt for graph for prices according to different

room type, another plot for price according to different room type and

neighborhood

* Borough wise price probability distribution for adjusted price
* Violinplot to showcase density and distribution of prices for different neighborhood
* from scipy.stats import norm for price distribution plot
* from scipy import stats for stat.probplot of price log
* Correlation Matrix to show that there is no strong relationship between price and other features.
* Residual Plot is strong method to detect outliers, non-linear data and detecting data for

regression models.

### **Multicollinearity**

Multicollinearity will help to measure the relationship between explanatory variables in multiple regression. If multicollinearity occurs, these highly related input variables should be eliminated from the model. In our kernel, multicollinearity will be control with Eigen vector values results. For this dataset, eigenvalues of the correlation matrix are close to zero. It means that there is no multicollinearity in the data.

**Classification Models**

First, Standard Scaler technique will be used to normalize the data set. For this dataset, each feature has 0 mean and 1 standard deviation.

Secondly, data will be split in a 70–30 ratio for applying classification models and prediction along with calculating accuracy and other related parameters.

**Extra Trees Classifier**

This class implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

To build a feature importance graph for our dataset, we have also used E.Tree classifier. And according to the analysis, neighborhood group and room type have the lowest importance on the model. Under this result, **the model building will be made in 2 phases.** In the first phase, models will be built using all features and in the second phase, models will be built without using neighborhood group and room type features.

**Model Building**

**Phase 1 - With All Features:**

Correlation matrix, Residual Plots and Multicollinearity results show that underfitting occurs on the model and there is no multicollinearity on the independent variables. Avoiding underfitting will be made with Polynomial Transformation since no new features cannot be added or replaced with the existing ones.

In model building section, Linear Regression, Ridge Regression, Lasso Regression, and ElasticNet Regression models will be built. These models will be used to avoiding plain Linear Regression and show the results with a little of regularization.

**Phase 2 - Without All Features:**

All steps from Phase 1, will be repeated in this Phase. The difference is, neighbourhood\_group and room\_type features will be eliminated.

**K-Fold Cross Validation**: It is any of various similar model validation techniques for assessing how the results of a statistical analysis will generalize to an independent data set. Before model building, 5-Fold Cross Validation will be implemented for validation of this dataset.

**Polynomial Transformation:** It consists of computing the polynomial whose roots are a given function of the roots of polynomial. Polynomial regression models are usually fit using the method of least squares. For this dataset, the polynomial transformation will be made with a second degree which adding the square of each feature.

**Model Prediction**

**Linear Regression**: In linear regression, the relationships are modeled using [linear predictor functions](https://en.wikipedia.org/wiki/Linear_predictor_function) whose unknown model [parameters](https://en.wikipedia.org/wiki/Parameters) are [estimated](https://en.wikipedia.org/wiki/Estimation_theory) from the [data](https://en.wikipedia.org/wiki/Data). In [statistics](https://en.wikipedia.org/wiki/Statistics), linear regression is a [linear](https://en.wikipedia.org/wiki/Linearity) approach to modeling the relationship between a scalar response (or [dependent variable](https://en.wikipedia.org/wiki/Dependent_variable)) and one or more [explanatory variables](https://en.wikipedia.org/wiki/Explanatory_variable) (or [independent variables](https://en.wikipedia.org/wiki/Independent_variable)). The case of one explanatory variable is called [simple linear regression](https://en.wikipedia.org/wiki/Simple_linear_regression). For more than one explanatory variable, the process is called multiple linear regression.

**Ridge Regression**: It is particularly useful to mitigate the problem of [multicollinearity](https://en.wikipedia.org/wiki/Multicollinearity) in [linear regression](https://en.wikipedia.org/wiki/Linear_regression), which commonly occurs in models with large numbers of parameters.

**Lasso Regression**: In statistics and machine learning, lasso is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The lasso procedure encourages simple, sparse models (i.e. models with fewer parameters).

**ElasticNet Regression**: In [statistics](https://en.wikipedia.org/wiki/Statistics) and, in particular, in the fitting of [linear](https://en.wikipedia.org/wiki/Linear_regression) or [logistic regression](https://en.wikipedia.org/wiki/Logistic_regression) models, the elastic net is a [regularized](https://en.wikipedia.org/wiki/Regularization_(mathematics)) regression method that [linearly combines](https://en.wikipedia.org/wiki/Linear_combination) the [L1](https://en.wikipedia.org/wiki/Taxicab_geometry) and [L2](https://en.wikipedia.org/wiki/Norm_(mathematics)#Euclidean_norm) penalties of the [lasso](https://en.wikipedia.org/wiki/Lasso_(statistics)) and [ridge](https://en.wikipedia.org/wiki/Tikhonov_regularization) methods.

# **Results**

|  |  |  |  |
| --- | --- | --- | --- |
| **Linear Regression**  **Phase 1:**  MAE: 0.377923  RMSE: 0.522021  R2 0.527663  **Phase 2:**  MAE: 0.531963  RMSE: 0.685894  R2 0.184227 | **Ridge Regression**  **Phase 1:**  MAE: 0.377915  RMSE: 0.522038  R2 0.527631  **Phase 2:**  MAE: 0.529255  RMSE: 0.679340  R2 0.199742 | **Lasso Regression**  **Phase 1:**  MAE: 0.375922  RMSE: 0.520400  R2 0.530591  **Phase 2:**  MAE: 0.523562  RMSE: 0.671290  R2 0.218595 | **ElasticNet Regression**  **Phase 1:**  MAE: 0.371707  RMSE: 0.518862  R2 0.533362  **Phase 2:**  MAE: 0.524883  RMSE: 0.670878  R2 0.219553 |

We calculated the 3 metrics of our models (Linear Regression, Ridge, Lasso and Elastic Net) for evaluating the predicting. The 3 metrics are: Mean Absolute Error (MAE) shows the difference between predictions and actual values. Root Mean Square Error (RMSE) shows how accurately the model predicts the response.  
R^2 will be calculated to find the goodness of fit measure.

The results show that all models have similar prediction results. All metric values are increased in Phase 2 it means, the prediction error value is higher in that Phase and model explain ability are very low the variability of the response data around mean.

The MAE value of 0 indicates no error on the model. In other words, there is a perfect prediction. The above results show that all predictions have great error especially in phase 2.

RMSE gives an idea of how much error the system typically makes in its predictions. The above results show that all models with each phase have significant errors.

R2 represents the proportion of the variance for a dependent variable that is explained by an independent variable. The above results show that, in phase 1, 52% of data fit the regression model while in phase 2, 20% of data fit the regression model.

# Conclusion

* This Airbnb dataset for the 2019 year appeared to be a very rich dataset with a variety of columns that allowed us to do deep data exploration on each significant column presented.
* By creating a map which shows adjusted price of each listing, we saw how the pricing was distributed for each listing over the New York. Also, how listings are distributed according to borough, number of listings belonging to each borough, how pricing was distributed among each borough was examined.
* From this, we saw the mean price of listings for each and every borough which will help the customer to not overpay in specific area and not get fooled by the hosts.  
  A model was fitted to predict price v/s every feature and price v/s all features expect neighbourhood\_group, room\_type. We saw that price dependent on each and every feature as the error for them was less than phase 2.
* Finally, ElasticNet was the best model among others to predict the price.

# References

<https://en.wikipedia.org/wiki>

<https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data>