Airbnb Analysis &

Price Prediction



Team Members:

**Graphical user interface, application

Description automatically generated**

Table of Contents

[Introduction 3](#_Toc121231948)

[Data Source 3](#_Toc121231949)

[Data Cleaning 4](#_Toc121231950)

[Removing Extra Data 4](#_Toc121231951)

[Filling Null Values 5](#_Toc121231952)

[Transforming Data 5](#_Toc121231953)

[Insights from Data 6](#_Toc121231954)

[Model Selection 9](#_Toc121231955)

[Decision Tree Regression 9](#_Toc121231956)

[Linear Regression   10](#_Toc121231957)

[Feature Selection 10](#_Toc121231958)

[Selection\_5 10](#_Toc121231959)

[Selection\_10 11](#_Toc121231960)

[Conclusion  12](#_Toc121231961)

[Appendix 12](#_Toc121231962)

# Introduction

Airbnb is an online marketplace for property owners to list their properties primarily for tourist/business homestays. Established in 2008, the company acts as a broker and charges a commission from each booking which allows property owners to list their properties for a larger audience in a streamlined manner. The property owners can develop trust with their customers under the brand name of Airbnb and are saved the hassle of creating their own customer base, while the customers have a choice of properties that are up to market standards.

This project aims at analyzing the properties listed on Airbnb and create a model that can predict the price of a new property to be listed so that it is as per market rates. Our model will also help the customers by informing them if a property is too expensive for the amenities it provides.  We plan to collect data by exploring the listing price data from the Inside Airbnb Database website and other sources. Once we have successfully collected and extracted data, we will clean our data to begin our analysis. We will visualize our cleaned data by plotting graphs to identify outliers and visualize the overall relationships and trends in a large amount of data. To enhance the accuracy of our model, we will train our data by splitting them into 80 and 20, and we will decide our predictor variables by performing feature engineering. Finally, we will run our regression model with the selected predictor variables to accurately predict the price of a property on Airbnb to help the new property homeowners.

Problem statement

New property owners generally don’t have a good understanding of the market and don’t have a clear idea about how they should price their property. New property owners who want to rent their properties on Airbnb can predict the price for their properties using our model and set the rental price accordingly, so that it appeals to most customers. This will allow property owners to price their property competitively based on relevant factors.

# Data Source

The data set used for this project is a publicly available data regarding the Airbnb listings. This dataset allows you to explore how Airbnb is really being used in cities around the world. For our project we have used data which has listings of all the Airbnb properties of Los Angeles. This dataset has a total of 89 attributes to describe a property, some of the attributes are Review ratings, Number of people the property can accommodate, Cleaning Fees, etc. We have information of 19427 listings, all the listing has unique features. On analyzing the data, we found that there are some missing values in the dataset which needs to be removed, we cleaned our dataset using various methods so that we can visualize data and create prediction models.

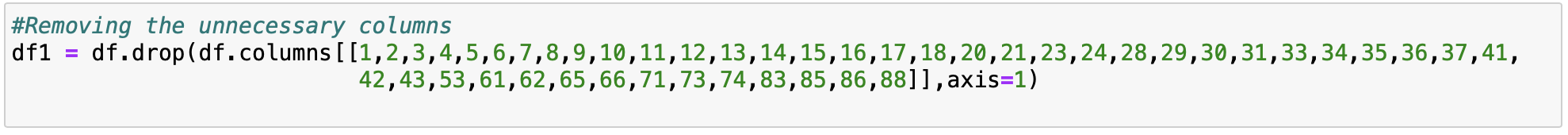
# Data Cleaning

The data cleaning step can be broadly bifurcated into 3 steps:

1. Removing Extra Data
2. Filling null values
3. Transforming Data

## Removing Extra Data

In our dataset we had a lot of columns which were not useful for our analysis like URLs for listing etc. We got rid of these kind of data by logically analysis each column and dropping the unnecessary columns. We identified that there are 47 such columns that are not useful for our analysis and we removed those columns.

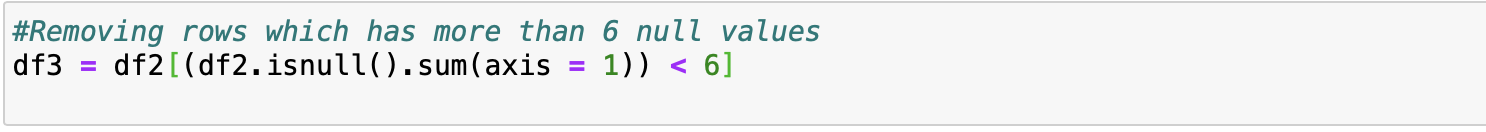


Post removal of extra columns we identified the columns which had more than 50% of its values as null. For such columns we don’t want to fill the null values because if we fill more that 50% of the values the data would be fabricated and not give us accurate results. We identified that there are 6 such columns that have more than 50% of its values as nulls, so we removed those 6 columns, which left us with 36 columns in our dataset.

Text

Description automatically generated with medium confidence

After cleaning the columns, we then moved to cleaning the rows, we identified the rows which have more than 6 null values and removed such rows. Next, we identified columns for which we cannot fill the null values and removed those null values as filling those nulls would lead to a bias in our analysis.



## Filling Null Values

In this step, after removing the extra data we still have null values which needs to be filled so that we can analyze the data and run machine learning algorithms. The nulls for each column were filled in a unique manner so that the data is not biased, first we filled the nulls in the zip code column, to fill the nulls in this column we used the Latitude and Longitude columns in our dataset. We created a function that converts the Latitude and Longitude into zip codes and used those zip code to fill the null values.

Graphical user interface, text, application

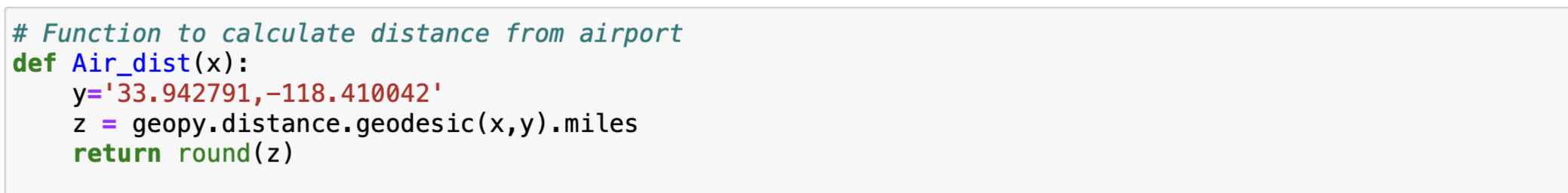
Description automatically generated

Next, we filled the null values for number of bathrooms and number of bedrooms a property has with the help of number of people it can accommodate column. The number of bathrooms and bedrooms will always be proportional to the number of people it can accommodate, so using a simple multiplication factor we filled the nulls in the number of bedrooms and bathrooms columns. We had null values in the cleaning fees column, the nulls were filled with the mean cleaning fees as it is a standard price which is charged for any Airbnb property. For nulls in review scores for specific fields we filled those nulls with the overall rating the property has, as the review score of a specific field will be proportional to the overall score. So, after filling all the null values we were left with 36 columns and 10942 rows.

## Transforming Data

After cleaning and filling all the null values our data set was ready to be used for analysis but there were a few columns which could give us more insights. To get more insights from such columns we had to transform them, first we converted the list of amenities to count of amenities. We transformed this column by creating a function which will count the number of strings in a cell. The count of amenities is a more usable information then list as this can quantified and used in machine learning models.

The next columns we used is the Latitude and Longitude columns, as for any property its location is very important, so one aspect of location which is very important is distance to airport. So, we used the Latitude and Longitude columns to calculate the property’s distance to Los Angeles airport. We calculated the distance by creating a function using geopy library that will take coordinate and calculated its distance from airport coordinates.



# Insights from Data

After all the cleaning and transforming data, we were ready to visualize the data to get more actionable insights. We used the interactive library plotly to plot all the visualization.

Chart, histogram

Description automatically generated

First, we displayed the price distribution which tells us that most of the properties are priced between $80 to $99. This visualization helped the new host to know how the properties are priced in the market.

Chart, bar chart

Description automatically generated

In the next visualization we displayed the correlation of the price of a property with different factors. From the graph it was clear that Price is highly correlated to the number of people the property can accommodate, the cleaning fees which is charged and the number of beds in the property.

Chart, bubble chart

Description automatically generated

In the bubble chart we can see a clear trend that as the number of people a property can accommodate, and the number of amenities increases the price of the property also increases and similarly the cleaning fees that is charged also increases. The pie chart shows on average what percentage of total fees is the cleaning fees & rental fees.

A map of a country

Description automatically generated with low confidence

The map helps the new host to gauge the competition by checking the price the competition is charging in the same locality. In general, we can see that the average price decreases as the distance from the airport increases.

Chart, bar chart

Description automatically generated

The next important factor for a host is ratings, there are two factor which we can plot to see how the rating are affected. From these graphs we can see that the host with moderately strict cancellation policy and who replies within a day has the highest ratings.

Chart, bar chart

Description automatically generated

These graphs will help you gauge the competition in terms of the type of property you have, the graphs show that ‘Apartment’ and ‘House’ are the property types which are mostly listed on Airbnb. Among these property types there are some proportions of properties that offer the entire house, some offer a private room and other offer shared rooms. Based on the room type and property type the new host can gauge the competition in the market.

# Model Selection

We used two machine learning models to predict our continuous output ‘Price.’ Because our goal is to predict the rental price for the new hosts, we wanted to work with a data frame that only contained numeric or binary values. Therefore, we converted the non-numeric type to integer and categorical values to binary using the panda’s get dummies function and were left with 59 attributes with numeric and binary values. We then moved on to running the algorithms. We explicitly used decision tree regression and linear regression. While these two algorithms are easy to manipulate, they also give the highest accuracy and are the best at predicting the continuous output. In addition, to measure our accuracy, we used the r-squared to evaluate the performance of our models.

## Decision Tree Regression

First, we ran a decision tree regressor with all attributes as our baseline/benchmark. Our variable X will store 58 attributes excluding the 'Price' and the variable y will store the 'Price'. Then, we split the dataset into the training and test set by 80 to 20, and we imported DecisionTreeRegressor from the tree package to run a decision tree model to fit it with our split X train and y train values. We now want to see how well-trained our dataset is, so we used the predict function on our decision tree regression model to predict the y value. Finally, we printed out the results. Based on our first model's results, we got an r-squared value of 57.71% and a mean absolute error of 41. This model is arguably not the best with low accuracy and a high error rate. To visualize the results, we plotted a regression plot, and we can easily spot many noises around the linear lines.Chart, scatter chart

Description automatically generated

## Linear Regression

Moving on to the linear regression model, just like the previous model, we did the exact data procedure other than running a LinearRegression imported from the linear\_model package. The results came out to be relatively similar to the first model, we got a 51.4% accuracy with an error rate of 50.

Chart, scatter chart

Description automatically generated

# Feature Selection

## Selection\_5

With the hope of improving our models and getting the best performing model, we decided to use both regression models concurrently to compare while making valid modifications.

To improve our models, we decided to pick the most informative attributes towards to our models. We did so by using the backward sequential feature selection function imported from SequentialFeatureSelector. We first tested out and found the five most informative attributes scored by the r squared values. As a result, we found the following five attributes,

**[ 'Bathrooms, Bedrooms, Cleaning Fee, Room Type\_Private room, Room Type\_Shared room' ]**

Now that we found the five most informative attributes, we ran our two algorithms with those five. As a result, we have improved both of our models. We have increased the Decision tree model by 10%, and the linear regression model by 15% in increase.

Although both models improved, we were not happy with the overall results using the five attributes and led us to experiment furthermore by increasing the number of informative

Attributes to ten.



## Selection\_10

Again, using backward feature selection, we found the ten most informative attributes.

**[‘Accommodates’, ‘Bathrooms’, ‘Bedrooms’, ‘Cleaning Fee,’ ‘Availability 90’, ‘Review Scores Location’, ‘Dist\_to\_Airp’, ‘Property Type\_House’, ‘Room Type\_Private room’, ‘Room Type\_Shared room’].**

Using the ten attributes above, we first ran linear regression. As you can see from the results below, we got the best accuracy and error rate of all time, although we still see some noises around the linear line compared to our benchmark and other models. This model is performing the best. Contrary to the linear regression model, the decision tree model accuracy deteriorated by 13% from its benchmark.

# Conclusion

After running ten different models, we discovered the best performing model is a linear regression with ten attributes with an accuracy of 70%. With this model, we want to be able to optimize to predict the Airbnb rental price range for the new tenant. To demonstrate, we have manually inputted a new datapoint that contains numeric or binary values of each ten attributes. Then, we ran our best model and predicted the outcome 'Price' using the predict function. As you can see below, we got the price predicted of $148.48. Text

Description automatically generated

To conclude, the linear regression model with ten attributes gave us the most accurate result when predicting a continuous value and answering our initial question. On the other hand, our decision tree regression models did poorly in predicting a continuous value. In addition, we learn that experimenting with the attributes such as leaving in and leaving out significant features are important when creating an accurate machine learning model.

# Appendix

**Data source**: https://data.opendatasoft.com/explore/dataset/airbnb-listings%40public/table/?disjunctive.host\_verifications&disjunctive.amenities&disjunctive.features