Can we predict the number of reviews of an Airbnb in NYC? | A simple analysis to help you make your Airbnb go VIRAL!

Contents of the Blog:

This blog will go over how I analyzed NYC's Airbnb prices datasets and used it to train a supervised machine learning model to better understand what contribute the most to the an Airbnb's reviews per month. Through this, we can explain what features are the most contributing factors for a Airbnb and how to optimize your property as a home/Airbnb owner.

The contents of the blog are as follows:

1. What is an Airbnb

First founded by Brian Chesky, Nathan Blecharczyk, and Joe Gebbia in 2008, Airbnb is an online marketplace platform that helps connect property owners (hosts) with travelers who are seeking short-term housing rentals to gain local traveling experience. Hosts are responsible for providing an appealing and comfortable stay for the travaler and the travaler in exchange will pay the host an agreeed upon amount. Through Airbnb, this allows hosts and travelers to connect more easily and provide more freedom for travel.

2. Problem Definition and Data Understanding

While keeping in mind with all that was mentioned above, I tackled a simple problem to predict the reviews per month of Airbnbs using simple dataset provided by Kraggle and some analyses.

Within this dataset, it include several measurements that will help in our analysis, these are as the following:

Feature	Description
id	Listing ID
name	Name of the listing
host_id	Host ID
host_name	Name of the host
neighbourhood_group	General area in NYC

Feature	Description
neighbourhood	Specific neighbourhood in NYC
latitude	Latitude coordinates
longitude	Longitude coordinates
room_type	Listing space type
price	Price in dollars
minimum_nights	Amount of nights minimum
number_of_reviews	Number of reviews
last_review	Latest review date
calculated_host_listings_count	Amount of listings per host
availability_365 Analysis	Number of days when listing is available for booking
reviews_per_month	Number of reviews per month

The reponse / target feature in the dataset is `reviews_per_month` which we will try to predict using different supervised learning models.

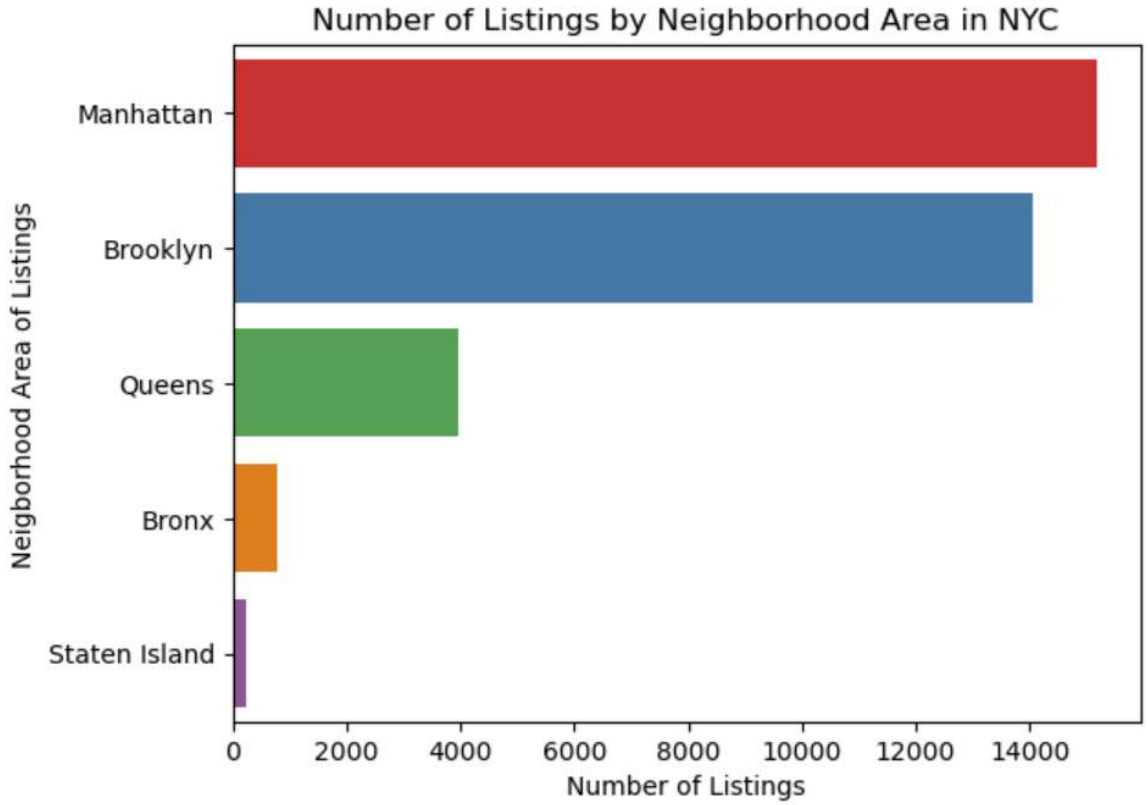
By analyzing and finding a model that accurately predicts `reviews_per_month` hosts can use it to see how popular their listings may be before posting or potentially help guide hosts in creating more appealing listings.

3. Exploratory Data Analysis

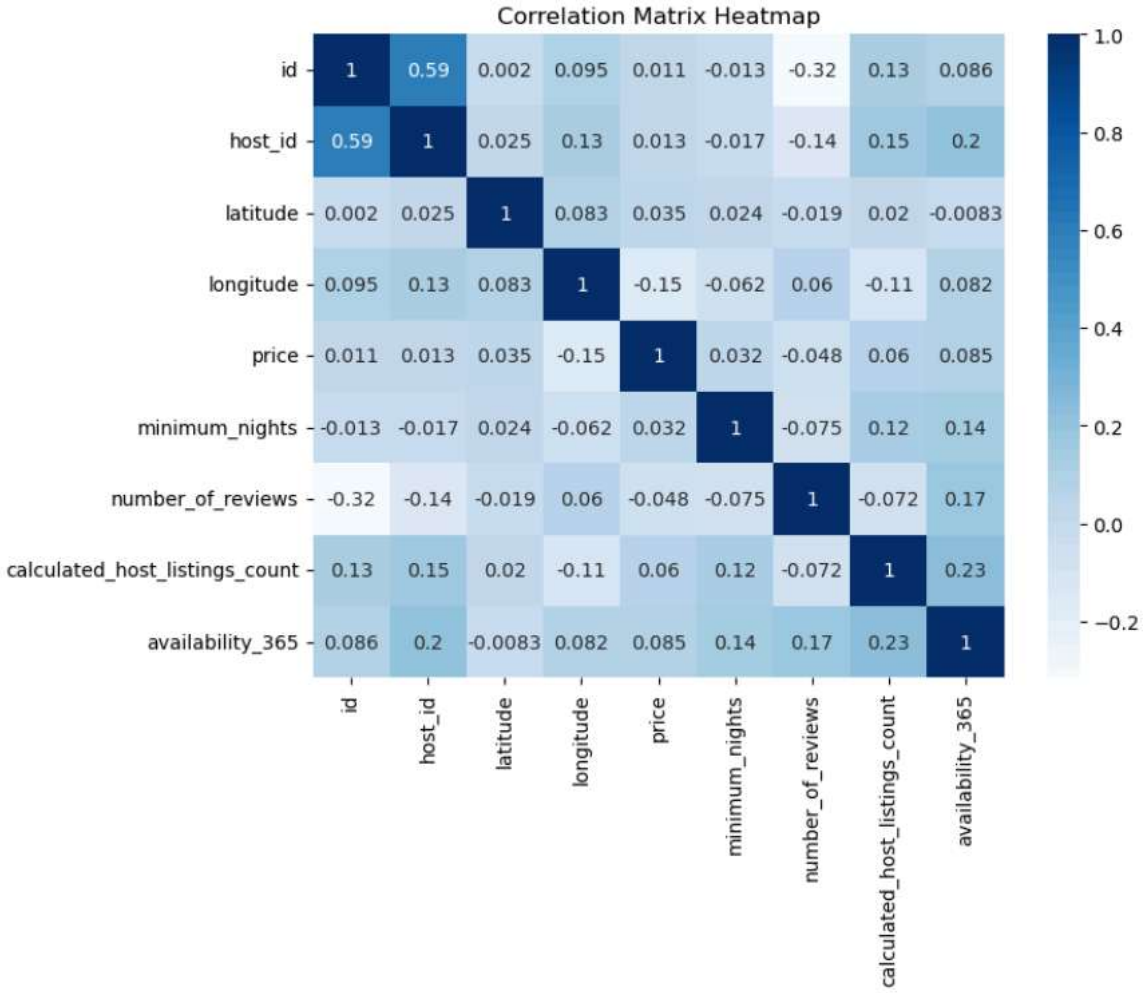
I did some Exploratory Data Analysis (EDA) and constructed some visualization to gain a better understading of the data and features. A few snippets of the results of the EDA are as follow:

```
<class 'pandas.core.frame.DataFrame'>
Index: 34226 entries, 36150 to 15725
Data columns (total 15 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   id                              34226 non-null  int64
 1   name                            34216 non-null  object
 2   host_id                         34226 non-null  int64
 3   host_name                       34209 non-null  object
 4   neighbourhood_group             34226 non-null  object
 5   neighbourhood                   34226 non-null  object
 6   latitude                        34226 non-null  float64
 7   longitude                       34226 non-null  float64
 8   room_type                       34226 non-null  object
 9   price                           34226 non-null  int64
 10  minimum_nights                  34226 non-null  int64
 11  number_of_reviews               34226 non-null  int64
 12  last_review                     27236 non-null  object
 13  calculated_host_listings_count  34226 non-null  int64
 14  availability_365                34226 non-null  int64
dtypes: float64(2), int64(7), object(6)
memory usage: 4.2+ MB
```

(^ Fig. 1 above)


Number of Listings by Neighborhood Area in NYC
Manhattan
Brooklyn
Queens
Bronx
Staten Island
Neigborhood Area of Listings
0
2000
4000
6000
8000
10000
12000
14000
Number of Listings

(^ Fig. 2 above)



(^ Fig. 3 above)

As seen from the EDA preformed above we can see a few key features that outlines the dataset.

Firstly, as seen from Fig. 1, we see that there are 34226 entries in total and each feature mostly contains same amount of entries. This suggests that there is a good chance most of the feature have no missing data and only a few features needs to be imputed.

Secondly, the bar chart in Fig. 2 shows that most of the listings in the dataset comes from the Manhattan area of NYC, indicating that this dataset is heavily skewed towards Manhattan listings and should be considered as a potential issue when generalizing any model to the wider population of NYC.

Lastly, to read the correlation heatmap as seen in Fig 3., one needs to look at the number corresponding to two variables. This grid like table has computed all the numerical representation of how "associated" or "corralated" two variables are to each other. The stronger the association, the higher the absolute correlation value. Conversly, the weaker the association, the lower the absolute correlation value. The negative and positive signs shows what the direction of change in value of one variable causes when its corresponding variable

changes. Hence, intuively, we can see on the correlation tables that all diagonal element is 1, which makes sense since an element should have the strongest correlation with itself.

As a result, through the heatmap, we can see none of the features are highly correlated to each other which we should also consider when preforming our future analysis.

4. Model Creation

To compare different models, I first settled on a metric in which I wanted use to measure the preformance of each model. After identifying that this problem is a regression problem that exhibits outlier and skewed data, I picked Mean Absolute Percent Error (MAPE) and Root Mean Squared Error (RMSE) as my metrics so I can provide the most coverage over detecting outlier issues (since RMSE punishes more on large errors) and also provides a focus on relative size errors (MAPE), ultimately showing if the model is over or under predicting.

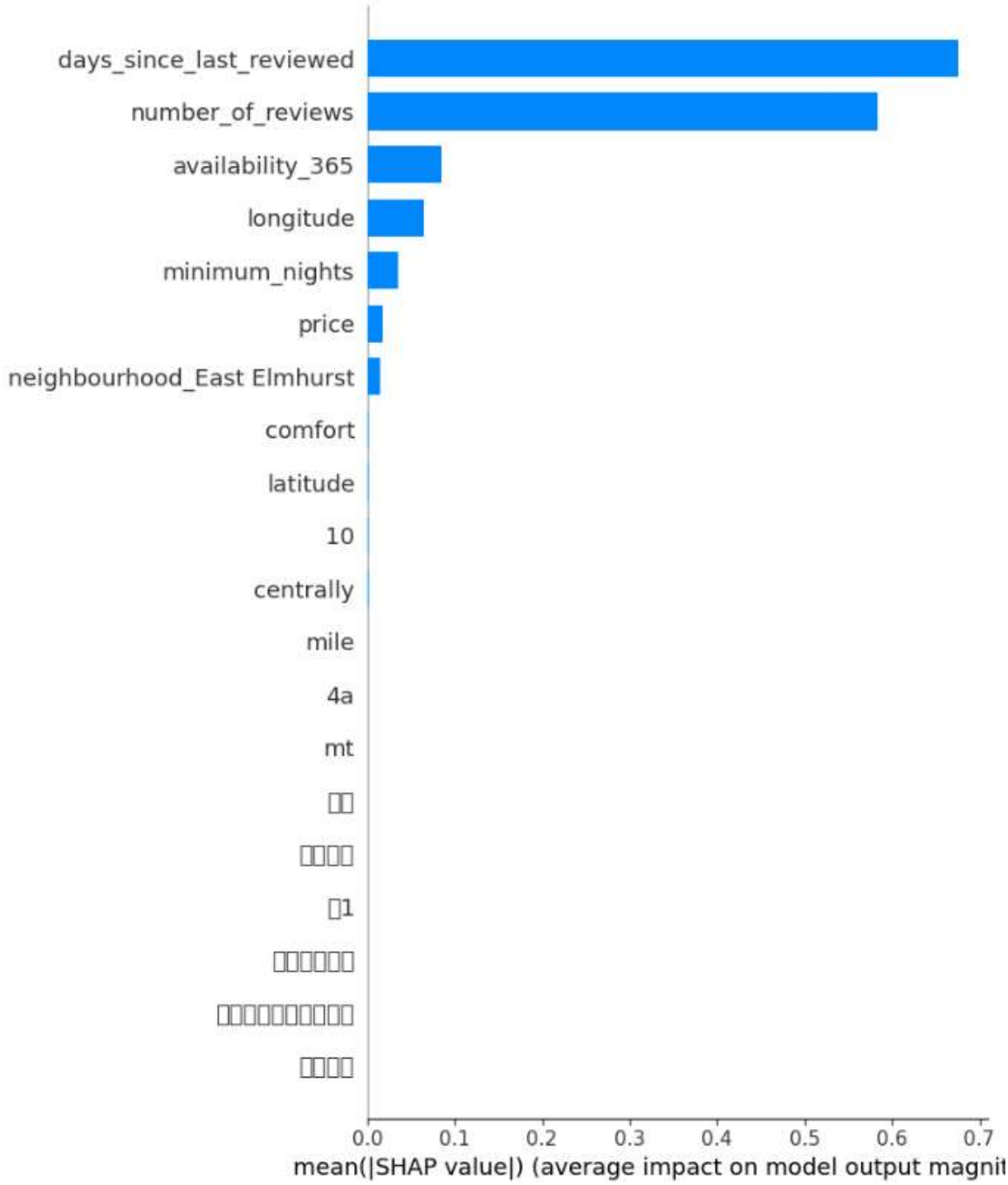
After preprocessing and transforming the data into a usable version, I trained and tested various models. After several iterations, I found that the RandomForestRegressor preformed the best out of all the models I tested.

The RandomForestRegressor models is an ensemble model that creates many different 'mini' decision trees (you can think of them as decision flowcharts)that has been tuned to predict the target given random samples of data and random feature. After feeding each mini decision tree their required information, we take the average of the predictions of all mini-trees and output it as our reseult.

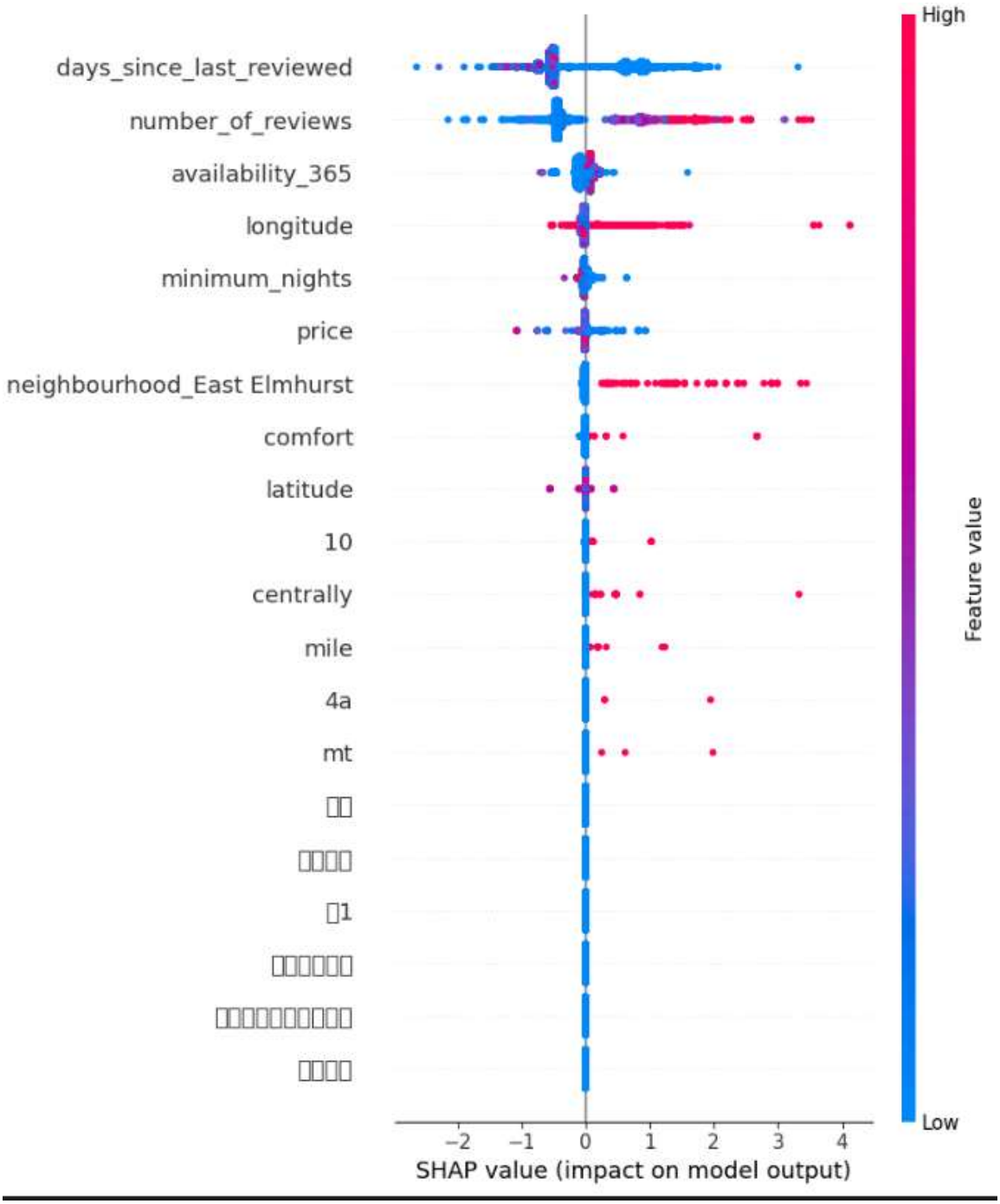
5. Feature Importance

To find out which feature contributed the most to the prediction of our target `reviews_per_month`, we can use SHAP plots as seen in the following:

Note that these SHAP plots are the representation of a regular `DecisionTreeRegressor` since `RandomForestRegressor` fitting and training time takes too long.



(^ above is Fig. 4)



(^ above is Fig. 5)

SHAP values tells how much a feature contributes to a model's prediction. If the shap value is positive, it will increase the prediction values. Inversly, a negative shap value will decrease the prediction value. And by looking at the SHAP plots from Fig 4. and Fig. 5, we can see that the most impactful feature seems to be the feature `days_since_last_reviewed` with the lower the value, the higher the prediction would be. On the other hand, the second most common feature was 'number_of_reviews' where the higher the value, the higher the prediction would be.

Therefore, through the analysis of the feature importance of this model, I can confidently say that if Airbnb hosts of NYC focus on decreasing their `days_since_last_reviewed` and increasing the `number_of_reviews` in total, their listing will recieve more listing a months.

6. Reflection

Some sources of errors can stem from several asepects. Firstly, the dataset as mentioned before had many listings regarding on the area of Manhattan, making the resulting fitted models unable to be generalize to the overall area of NYC. Secondly, the SHAP value shown does not reflect the best model I explored, hence, the features that may be important in the `DecisionTreeRegressor` model may not apply to the `RandomForestRegressor` model. Lastly, I think one a component that could be misleading was the fact I chose `RandomForestRegressor` as my "best model" even though there are multiple other models I have not tested. If I would to do this analysis again, will defintely dive deeper into other ensemble models such as `LightGBM` or `CatBoost`.
