# NYC Airbnb Price Prediction

Sichen Lyu, Shuwen Liang, Haoyang Li

May 2023

# 1 Abstract

In this project, we aimed to develop an effective predictive model for Airbnb rental prices in New York City by leveraging available features and factors. By using machine learning techniques such as Random Forest, XGBoost, etc, we analyzed various factors including host characteristics, location, property type and amenities. The questions we want to answer are: (1) How can we leverage the features and factors of Airbnb historical records in New York City to develop a pricing strategy that maximizes revenue for hosts and travelers? (2) Which predictive model is the most effective in utilizing the available features and factors of Airbnb historical records in New York City to provide optimal pricing strategies for hosts and for travelers to find the best deals? Our findings demonstrate the potential of machine learning algorithms in accurately predicting the prices, providing valuable insights for hosts and travelers in the competitive Airbnb market. This research contributes not only to understanding the pricing strategies in the hospitality industry but also offers practical implications for applications in the growing Airbnb landscape. The model that we trained can be used by users to see if the pricing of a property is reasonable, or by property owners to decide the price.

# 2 Introduction

Nowadays, Airbnb as an alternative to traditional accommodations becomes more and more popular and has significantly. New York City, as one of the world's most popular tourist destinations, has a complicated and dynamic Airbnb rental market. Therefore, accurate price prediction is crucial for both hosts and travelers, as it can optimize pricing strategies, increase occupancy rates, and maximize revenue. In this study, we investigate the applicability of several machine learning models to predict Airbnb prices in New York City using a dataset from Kaggle. The dataset contains valuable and fruitful information such as neighborhood, location, room type,

accommodation length of stay, and reviews and availability. We explain each variable in detail in the data preprocessing part..

There are two main questions we want to answer for the project: (1) How can the available features and factors of Airbnb historical rental records in New York City be effectively utilized by different machine learning models to predict the prices? (2) Which machine learning model provides the most accurate predictions based on the available features and factors of Airbnb historical rental records in New York City?

To address these questions, we first did some literature review to explore various machine learning models, such as Random Forest Regression, Extreme Gradient Boosting, Light Gradient Boost Machine Regression, etc. Meanwhile, we cleaned the data as preparation for model development. Then we trained predictive models by different machine learning methods. Afterwards, we compare the results to get the optimal model for our project.

## 3 Background

In recent years, machine learning techniques have been applied to analyze and predict pricing extensively, including Airbnb listing prices. We did literature review on some machine learning methods which are used as most: Random Forest Regression, Extreme Gradient Boosting, Light Gradient Boost Machine Regression, and random search cross validation, which have stated that the algorithms can be applied well in similar situations.

Dhillon, Jasleen, et al. (2021) implemented both logistic regression and Random Forest models for Airbnb price prediction. The result showed that the Random Forest method has the lowest RMSE values. Airbnb's unique pricing strategy was studied by Zhu, Li, et al. (2020) by using various machine learning methods. Among these methods, bagging, XGBoost, and random forest demonstrated the best prediction performance. Ahuja, Lahiri, et al. (2021) utilized features such as latitude, description score, longitude, review sentiments, and neighborhood popularity, the LightGBM model predicts the most accurate rental prices, with high R2 score achieved.

Finally, random search cross validation has been widely used for hyperparameter optimization in machine learning models (Bergstra \& Bengio, 2012), which indicates the best performance of the chosen algorithms.

Based on these studies, our project target to adapt the appropriate machine learning methods to predict Airbnb listing prices in New York City as well as find the impact of valuable factors on pricing.

# 4 Data Preprocessing

### *4.1 Variable Explanation*

-*id* is the primary key of the data set, each has a corresponding property.

-*name* is the name of the property.

-*host_name* is the name of the host.

-*host_id* is a primary key for hosts.

-*neighbourhood_group* indicates the district of New York City where the house is located.

-*neighbourhood* is a more accurate location of a house. A neighbourhood group contains several neighbourhoods.

-*longitude* and latitude gives the exact position of the house.

-*roomtype* is the floor plan of the house.

-*price* is our target variable, and it is the average price for a one-night stay.

-*minimum_nights* is the minimum length of stay when booking.

-*last_review* is a date time value that shows the latest time when a review was made.

-*reviews_per_month* is the monthly average review the house receives.

-*number_of_reviews* is the total amount of reviews.

-*calculated_host_listings_count* is the total number of available Airbnb houses of the property owner.

-*availability_365* indicates how many days a year the house is available to be rented.

### *4.2 Data Preprocessing Steps*

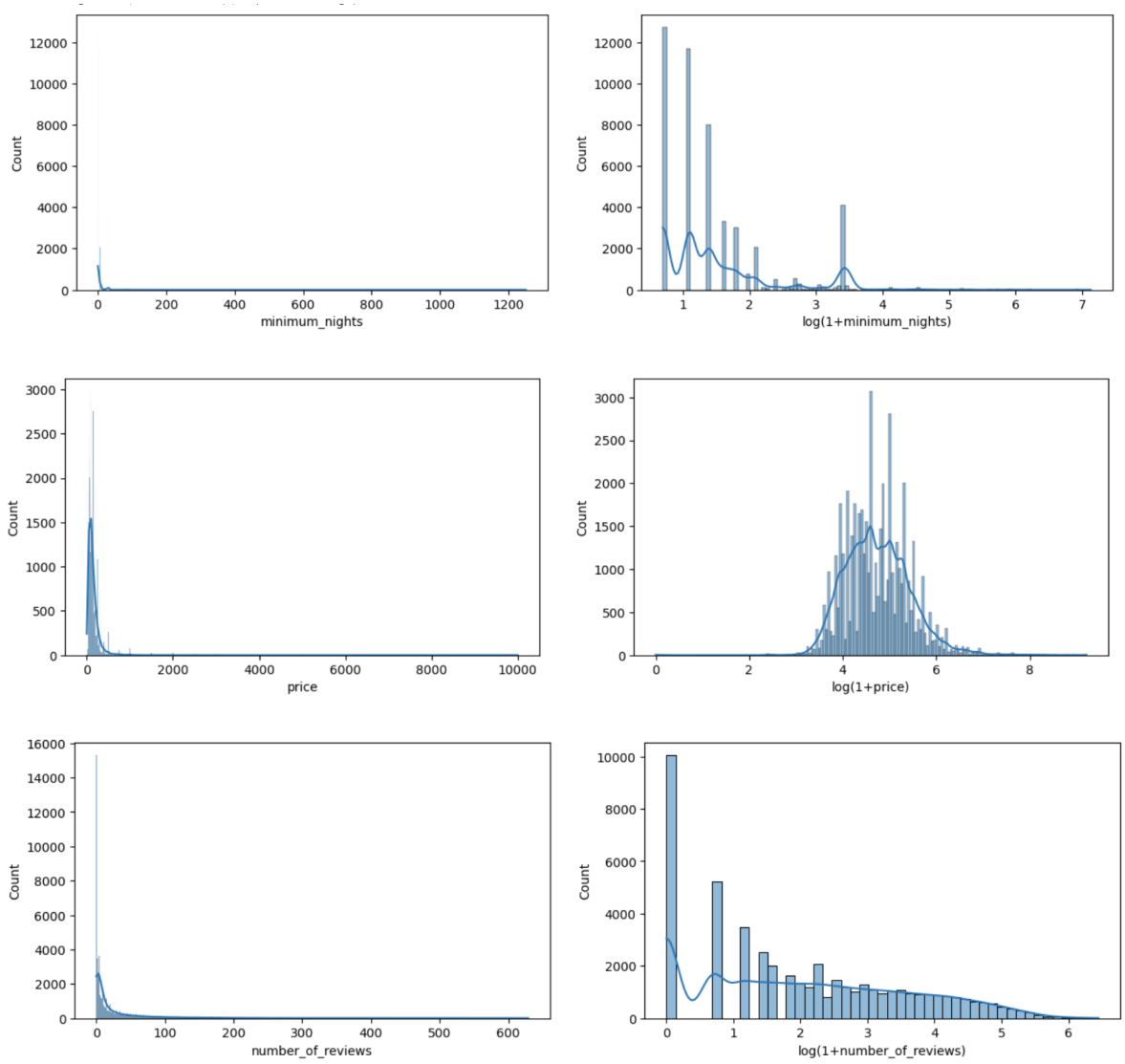
Columns 'id', 'host_id', 'name' and 'host_name' are used to identify properties and their users and are useless for our analysis, thus we dropped those columns.

Columns 'neighbourhood_group', 'neighbourhood' and 'room_type' and categorical data. We used one-hot encoding to convert them into numerical data. After the conversion, these 3 categorical variables were transformed into 229 Boolean variables.

Some properties are for large parties instead of a regular stay, for example, a 13 bedroom house with pool and garden. Since our model is designed to help users find a place to stay during a visit to NYC, we dropped rows with prices larger than 2000.

We converted 'last_review' from a date time value to an integer value, indicating the amount of days since last review.

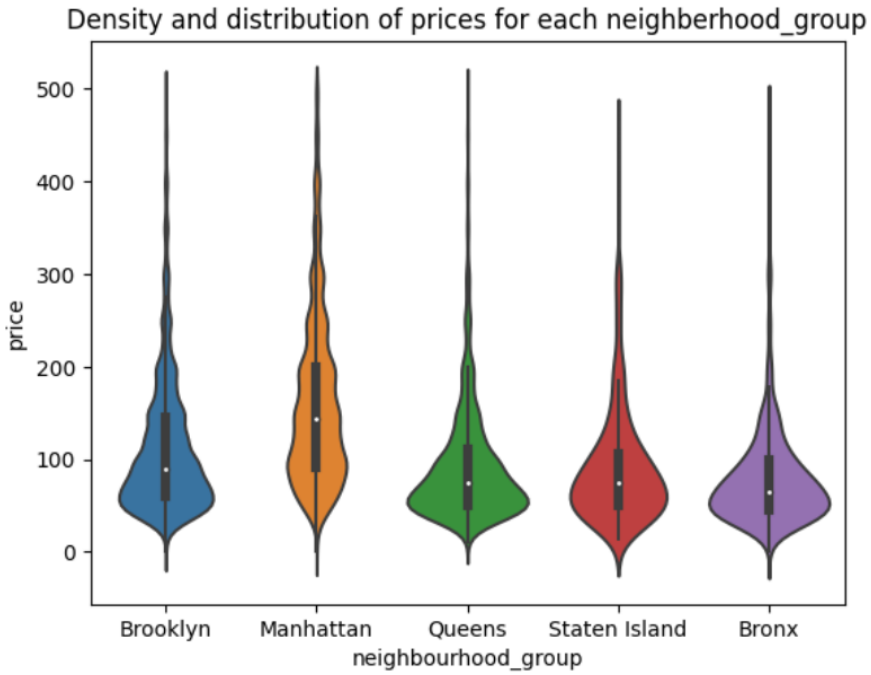
Columns 'number_of_reviews', 'minimum_nights' and 'price' are very unevenly distributed, with most values very low and dense. Thus we used a log transformation to take the log of the values to make them contribute more to the model.




### *4.3 Exploratory Data Analysis*

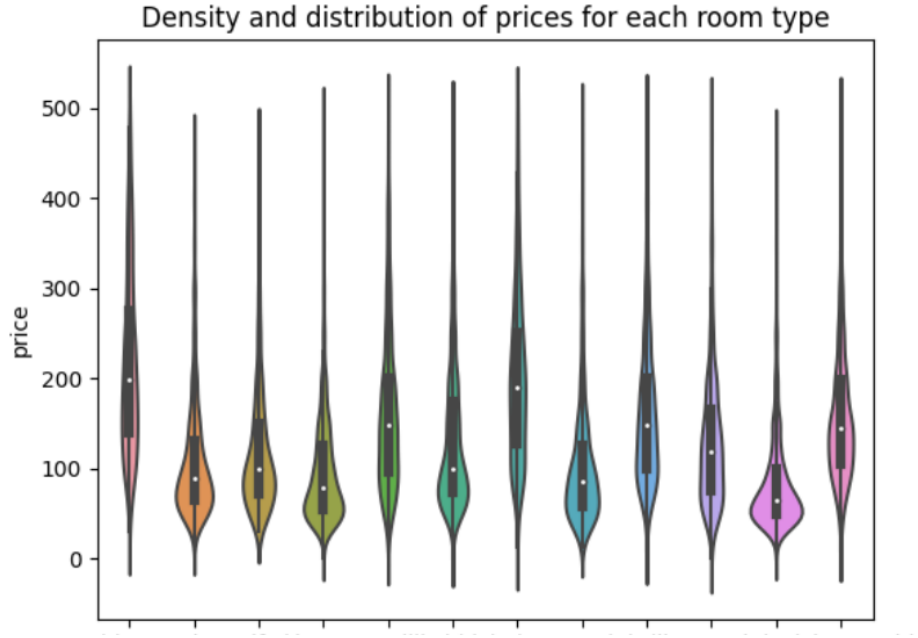
First we would like to see the distribution of price based on neighborhood area. This is a violin plot of the relation.

The outliers are properties with high prices. After we limit the highest price to 500 per night, the distribution becomes more readable.



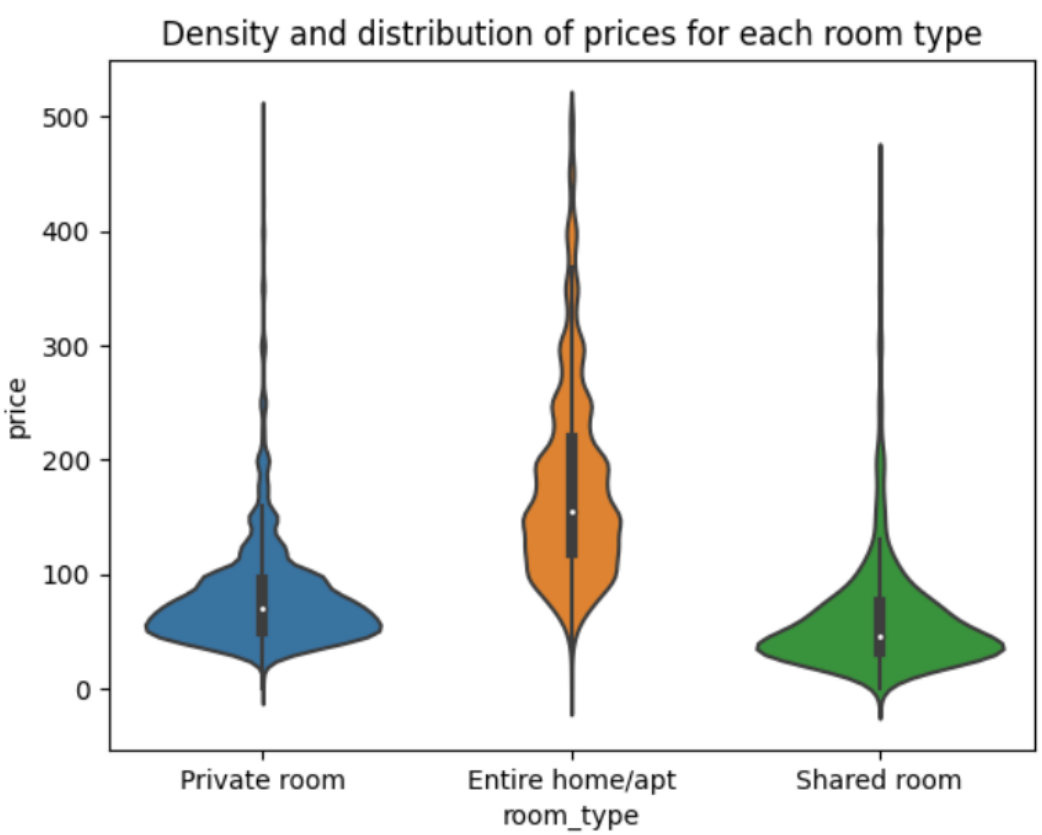

As this result suggests, price is related to neighbourhood_group and we shall add it as an attribute in our model.

This is a similar plot with different neighborhoods. Each of these neighbourhood belongs to a neighbourhood_group.




It shows that price and neighbourhood are also related.

Here is the violin plot of room type and price.




We can use one-hot encoding to transform categorical data like this one later when we build our models.

For the 3 columns related to reviews, their individual relation with price is hard to tell, but considering that keeping other variables same, rooms with better or more frequent reviews should charge more, we also put all data related to reviews into our model.

## 5 Methodology

### *5.1 Random Forest Regression*

Random Forest Regression constructs a large number of decision trees and combines their outputs to obtain an accurate prediction. Each tree in the forest is constructed by randomly selecting a subset of the input features and a subset of the training samples; this randomness helps reduce overfitting and build a more generalized model. Then the model summarizes the output of each tree to produce a

final prediction.Random Forest Regression is flexible, easy to use, and capable of handling large data sets with a large number of features.

### *5.2 Extreme Gradient Boost Regression*

Extreme Gradient Boosting (XGBoost) is an advanced machine learning algorithm that is used for prediction tasks.It constructs a set of weak prediction models (decision trees) under a gradient boosted framework.Each tree is constructed sequentially to improve based on the previous model.In XGBoost, each decision tree is constructed to minimize the residual error between the predicted and true values. The model summarizes output of each tree to produce a final prediction. XGBoost also uses regularization to prevent overfitting and build a more generalized model. It produces its result fast, and has a relatively high level of accuracy.

### *5.3 Light Gradient Boost Machine Regression*

LightGBM (Light Gradient Boosting Machine) is a gradient boosting framework that is designed to be highly efficient and scalable for large-scale machine learning tasks.

It is good at handling large data sets, since LightGBM uses Gradient-based One-Side Sampling (GOSS) that selects only the important data instances for gradient-based learning, resulting in much faster training times.

### *5.4 Random Search Cross Validation*

Random Search Cross Validation picks optimal random parameters from a data set and uses cross validation to verify performance. It picks the best parameters in the given iteration limit and saves more time than grid search cross validation.

## 6 Building the Model and Parameter Tuning

To begin with, we separated the data set to 80% training and 20% testing.

To compare the performance of the 3 models, we used max_depth = 8, learning_rate = 0.1 and n_estimators = 300 for all 3 models.

Here are the squared errors of each model:

```
from sklearn.ensemble import RandomForestRegressor
rf = RandomForestRegressor(max_depth=8, n_estimators=300)
rf.fit(all_train, y_all_train)
rf_all_pred = rf.predict(all_test)
print(mean_squared_error(rf_all_pred, y_all_test))
from xgboost.sklearn import XGBRegressor
xgb = XGBRegressor(max_depth=8, learning_rate=0.1, n_estimators=300)
xgb.fit(all_train, y_all_train)
xgb_all_pred = xgb.predict(all_test)
print(mean_squared_error(xgb_all_pred, y_all_test))
from lightgbm import LGBMRegressor
lgbm = LGBMRegressor(max_depth=8, learning_rate=0.1, n_estimators=300)
lgbm.fit(all_train, y_all_train)
lgbm_all_pred = lgbm.predict(all_test)
print(mean_squared_error(lgbm_all_pred, y_all_test))
```

```
0.17739512086139503
0.16588033326563123
0.1639004582110518
```

*Performance of the 3 Models Measured by Squared Error*

Here are differences between the testing y set and prediction of each model.

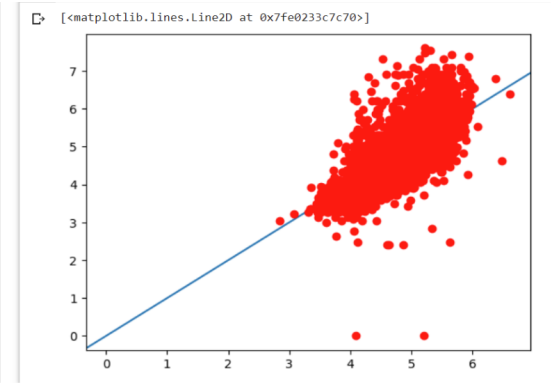



Figure 2: Performance of the Random Forest Regressor

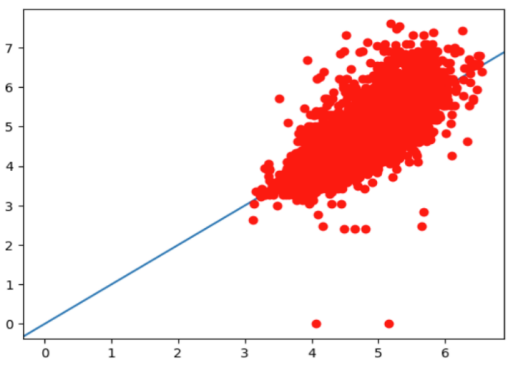


Figure 3: Performance of the Extreme Gradient Boost Regressor

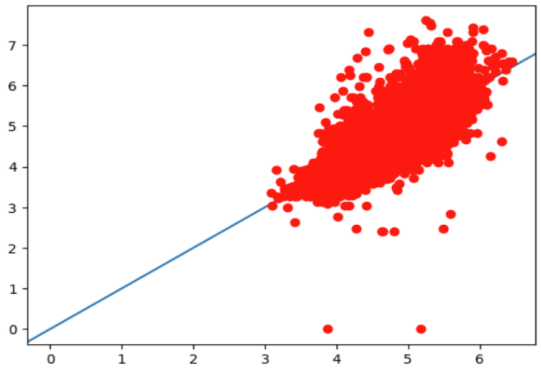


Figure 4: Performance of the Light Gradient Boost Regressor

The difference between predictions and true value are similar for the 3 models, but the mean squared error of the light gradient boost machine regressor is slightly lower. So we used a random search CV to pick the best parameters for a lgbm.

At last, we reached a mean squared error of 0.16319.

## 7 Discussion

We designed the model to help property owners decide their airbnb prices. After the user put their information in, the prediction of the model could offer a suggestion. It is not a weapon of math destruction because it wouldn't harm the society and it treats different data based on non-discriminating standards.

There are also some limitations about our model. As we were looking into the data set, we found that there were some property owners who offer half-year or one-year-long leases. Airbnb was designed to offer short-term leases for travelers and those long term leases seem to be like renting an apartment and shouldn't be posted on Airbnb. However, we chose not to drop that data.

Overall, our model managed to use given parameters to predict Airbnb prices, help users decide whether a price is reasonable and offer suggestions to property owners with acceptable mean squared error.

## 8 Reference


Ahuja, A., Lahiri, A., & Das, A. (2021). *Predicting Airbnb Rental Prices Using Multiple Feature Modalities*. arXiv preprint arXiv:2112.06430.

Bergstra, J.,\& Bengio, Y. (2012). *Random search for hyper-parameter optimization.* Journal of machine learning research, 13(2).

Dhillon, J., Eluri, N. P., Kaur, D., Chhipa, A., Gadupudi, A., Eravi, R. C., & Pirouz, M. (2021, January). *Analysis of Airbnb Prices using Machine Learning Techniques*. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference* (CCWC) (pp. 0297-0303). IEEE.

Zhu, A., Li, R., & Xie, Z. (2020, September). *Machine learning prediction of new york airbnb prices*. In *2020 Third International Conference on Artificial Intelligence for Industries (AI4I) (pp. 1-5). IEEE*.