Multi-dimensional Analysis On New York City's Airbnb Data
CS 396: Project Presentation

Jiwon Choi

Northwestern University

jiwonchoi2021@u.northwestern.edu

Dec 2, 2021

Overview

1. Motivation
2. SuperHost T/F Binary Classification
   - Host-Related Features
   - Property-Related Features
   - SuperHost Classification Insights
3. Price Prediction
   - Price Prediction Feature Set
   - Naive Price Prediction
   - Price Range (Bin) Prediction
     - Determine Price Range (Bin)
     - Price Range (Bin) Prediction
   - Price Prediction Insights

Motivation

Intuitive, concise, and informative data visualization is needed for efficiency and convenience, after aggregating all the necessary features:

- The era of post-COVID is coming, and many people started to plan their trip ahead.
- Needs of providing a meaningful data to both suppliers (Airbnb hosts) and demanders (Airbnb users).
- Users take a look into not only price, but also neighborhood, reviews, and other factors.
- Competitive hosts also perform manual lookup of what other hosts' have.

SuperHost T/F Binary Classification

Classifying whether one is a superhost or not would give an insight about how to become a superhost. This will allow current hosts to allure more customers and make more profit.

Two different dataset, with three different ML models have proposed:

1. Dataset with host-related features: kNN, Logistic Regression, Gradient Boosting Classifier
2. Dataset with property-related features: kNN, Logistic Regression, Gradient Boosting Classifier

Host-Related Features

About Dataset:

- **Features**: `host_response_rate`, `host_acceptance_rate`, `host_has_profile_pic`, `host_identity_verified`, `host_since`
- **Target:** `host_is_superhost`
- **Feature Engineering**: manual binarization required for 't'/'f' labels, mapped host year since into n^{th}-year value
- **Sample Size**: 20,141

Table 1: Superhost Classification Model with Host Features

	kNN	Logistic Regression	Gradient Boosting Classification
CV Accuracy	0.7720	0.7357	0.7517
Test Accuracy	0.7833	0.7223	0.7468

* **Model Parameter**: 5-fold cross validation used.

Property-Related Features

About Dataset:

- **Features**: `neighbourhood_group_cleansed`, `room_type`, `number_of_reviews`, `number_of_reviews_l30d`, `number_of_reviews_ltm`, `reviews_per_month`, `review_scores_rating`, `price`, `instant_bookable`
- **Target:** `host_is_superhost`
- **Feature Engineering**: manual binarization required for 't'/'f' labels, string values into one hot encoding
- **Sample Size**: 27,608

Table 2: Superhost Classification Model with Property Features

	kNN	Logistic Regression	Gradient Boosting Classification
CV Accuracy	0.7725	0.7733	0.8334
Test Accuracy	0.7680	0.7706	0.8392

* **Model Parameter**: 5-fold cross validation used.

SuperHost Classification Insights

- kNN classifier outperforms in case of the host-related features.
- Perhaps due to the small number/dimension of input features.
- Gradient Boosting classifier outperforms in case of property-related features.
- Gradient Boosting tree consists of many trees, and each of them gets trained on the residual of the previous tree. Thus, more robust.
- Determining a superhost relies more on property-related features than host-related features –accuracy is higher.

Price Prediction

Predicting property price would give an insight about getting a quote of the desired property. This will allow users to have an estimated price (range) based on the features of the property.

Two different dataset (only `price` target values are different!), with two different ML models have proposed:

1. Naive prediction with host and property-related features: Linear Regression, Gradient Boosting Regressor
2. Prediction with price range (bin) with host and property-related features: Linear Regression, Gradient Boosting Regressor

Price Prediction Feature Set

About Dataset:

- **Features**: `host_is_superhost`, `neighbourhood_group_cleansed`, `host_has_profile_pic`, `room_type`, `instant_bookable`, `host_since`, `host_identity_verified`, `reviews_per_month`, `number_of_reviews`, `number_of_reviews_l30d`, `number_of_reviews_ltm`, `review_scores_rating`
- **Target:** `price`
- **Feature Engineering**: manual binarization required for 't'/'f' labels, mapped host year since into n^{th}-year value, string values into one hot encoding
- **Sample Size**: 27,608 and 26,787, respectively

* Same features are used throughout price prediction. Only y target values are different.

Naive Price Prediction

In naive prediction, the model is trying to predict a precise price.

Table 3: Naive Price Prediction with Host & Property Features

	Linear Regression	Gradient Boosting Regressor
CV R^2	0.0442	0.0467
Test R^2	0.0335	0.0544

* **Model Parameter**: 5-fold cross validation used.

Determine Price Range (Bin)

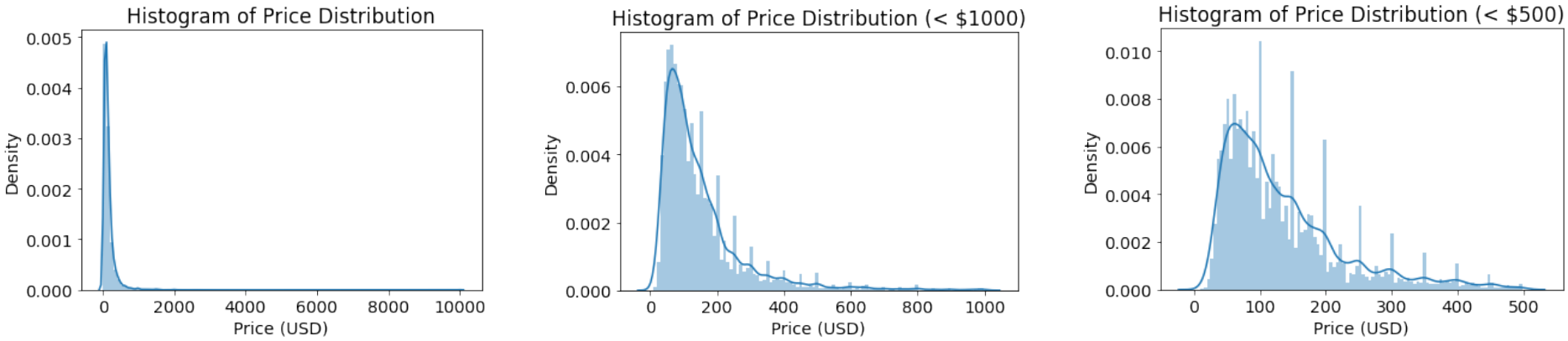

Figure 1: (left) Distribution of all price; (middle) Distribution of price under $1000; (right) Distribution of price under $500

1. Total distribution shows that it has a long tail at the right.
2. Distribution of prices under $1,000 still long tail.
3. Distribution of prices under $500 skewed to the right.
4. Shrink the dataset by using only the y target values under $500.
5. Bin step size is $100, so the total bin is 5, from 0 to 4.

Price Range (Bin) Prediction

In range prediction, the model is trying to predict a price range after dividing the y target values into bins.

Table 4: Price Range Prediction with Host & Property Features

	Linear Regression	Gradient Boosting Regressor
CV R^2	0.2313	0.3194
Test R^2	0.2511	0.3288

* **Model Parameter**: 5-fold cross validation used.

Price Prediction Insights

- Gradient Boosting regressor outperforms in both naive and range prediction.
- Gradient Boosting tree consists of many trees, and each of them gets trained on the residual of the previous tree. Thus, more robust.
- Yet, the R^2 scores are not promising, simply outperforms.
- Due to the nature of complexity, price prediction is more precise when there is a simpler or small number of targets to predict.