Airbnb Price Prediction

Kailing Wang Luyu Jin Siyuan Xiang

Motivation and Problem

We predicted the optimal Airbnb price in New York City considering features such as room types, numbers of beds, ratings, and amenities. With better predicted price, the Airbnb hosts can provide profitable and satisfactory accommodations, while the guests can afford the accommodations expenses. We plan to reach the ultimate objective that both the hosts and guests are satisfactory with their experience of using Airbnb.

Data

The Airbnb dataset contains 40, 227 records and 95 attributes. First, we selected the following 14 attributes from the raw dataset:

- zipcode
- property_type
- room_type
- accommodates
- bathrooms
- bedrooms
- beds
- amenities
- price
- minimum_nights
- number_of_reviews
- review_scores_rating
- cancellation_policy
- calculated_host_listings_count

Next, we preprocessed these attributes:

- Create dummies for the categorical variables (i.e. property_type, room_type, and cancellation_policy)
- Convert the price format (e.g. convert a string '$1,200.00' to a float of 1200.0)
- Drop missing values in all columns except for 'review_scores_rating'
- Drop inconsistent entries (i.e. 'accommodates', 'bedrooms', 'beds', or 'price' with a value of 0)
- Create feature vectors for 'amenities'
- Move the target variable 'price' to the last column

Finally, we ended up with a cleaned dataset with 92 useful features.

Model

I. Linear Models

- Ridge
- Lasso
- Elastic Net
- Bayesian Ridge

II. Cluster + Linear Models

We stratified the input data into 4 clusters, according to zipcode and fit linear models with the data in each cluster separately. When predicting a new instance, we will first decide which cluster it belongs to, then the corresponding linear model will be used to predict its target value.

III. Non-linear Models

1. Random Forest

A random forest is a meta estimator that fits a number of decision trees on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. We used depth = 13, which gave the highest OOB score.

2. Kernel Ridge

Kernel ridge regression combines Ridge Regression with the kernel trick to deal with non-linear features, which are very common in our dummy variables.

3. Support Vector Regression

Support Vector Regression is another way to introduce non-linearity.

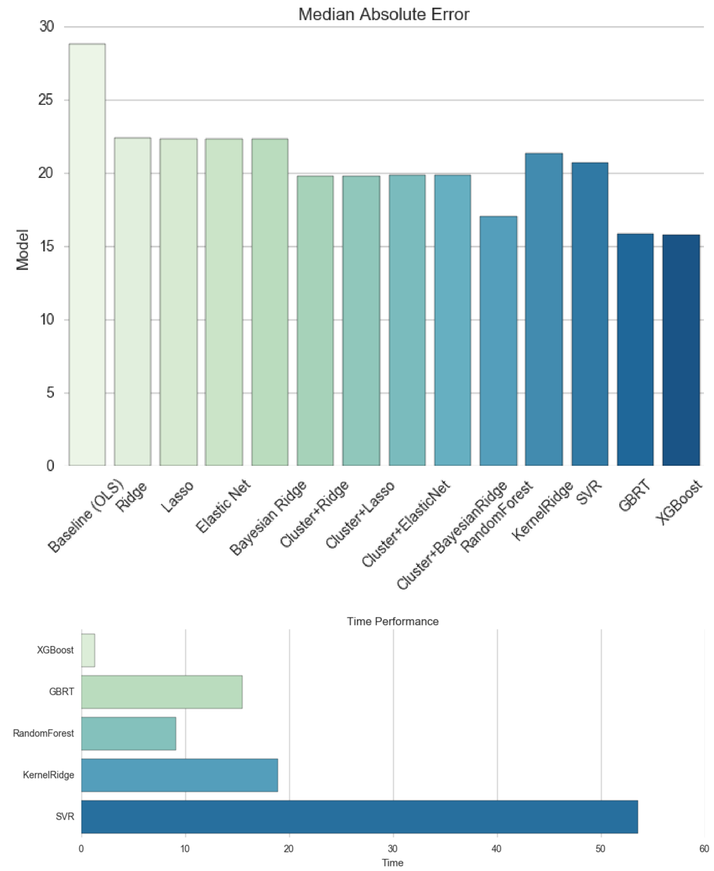
4. Gradient Boosted Regression Trees

We applied GBRT, which consists of 50 regression trees. With cross-validation, we got best depth = 9.

5. XGBoost

XGBoost is an implementation of gradient boosted trees designed for speed and performance that is dominative competitive machine learning. It also introduces features such as column sampling, GPU support and distributed computing.

Results



Further Steps

- Incorporate seasonality into our model
- Examine instances that deviate a lot from our prediction and revise the models
- Extract other features to improve performance
- Employ LightGBM to further reduce runtime

Reference

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794). ACM.

Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830. 2011.