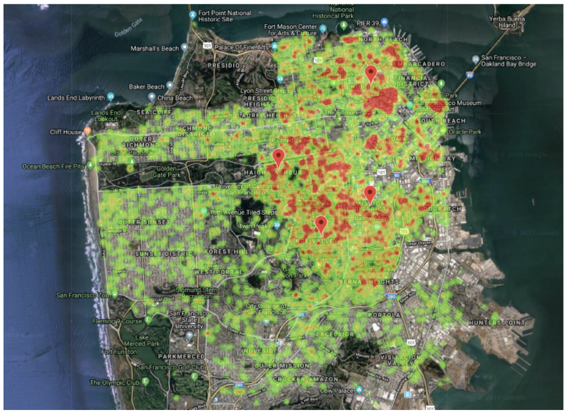
CP 101 FINAL PROJECT
AIRBNB HOME PRICE
PREDICTION AND RECOMMENDATION

Keqin Cao

1 Introduction

1.1 Background

Airbnb is an online marketplace for accommodations sharing, standing for “Air Bed and Breakfast”. It allows people to rent out their homes to guests looking for a place to stay in an area. It is becoming more and more popular because guests can get a taste of the local lifestyle and recommendations, and hosts can get extra income by sharing sparse spaces. If new hosts want to select a listing price for their houses, they would have to research the housing prices in their neighborhood and make the decisions correspondingly. This takes effort and time and it is difficult for a host to find a suitable price totally by himself. Currently, Airbnb offers a pricing tool called ’Smart Pricing’ that can help hosts to determine the price based on the demand. However, it turns out this tool sometimes underestimates the demand and set the prices too high, which negatively affects the profitability of hosts. Also, as the number of hosts increases, a better pricing algorithm is in need to help set a competitive and suitable price.



1.2 Purpose and goal

The goal is to start from the standpoint of Airbnb hosts and help them select an optimal price for new listings competitively based on similar listings to attract more guests using a machine learning technique. The data is obtained from Inside Airbnb 12, where monthly data is available for different cities all around the world. In this project, I focus on the 2016-2018 San Francisco data. Although it would be super interesting to analyze Airbnb pricing across the current COVID-19 situation, I do not trust the data accuracy in different sorts of websites. The fluctuation of the pricing aligns with the public knowledge but the sudden hit to the marketing; however, the company does not publicly release the data so I would have to wait till next quarter or even next year for the "actual (with modification)" released data.

2 Data description and Pre-processing

The data is a snapshot of listings available on a particular day. That is, I only have access to the data for a representative day in each month for several years. Each city has different ranges of available and some data are missing for particular cities. In each representative day, there are four data sets available, and they are listings, calendars, reviews, and neighborhoods. The following is a rough description of each data set with its corresponding size. In this project, I will focus on the Listings data due to various reasons. Firstly, Reviews will not be a feasible feature for a new host simply because the host does not have any previous record of hosting. Also, the Technique of NLP analysis is out of the scope of this class. I have also encountered some hosts during my stay that offers to leave a review of what they want in trade for five dollars Venmo. Secondly, Calendar can be analyzed separately for recommending a better time to rent a house, but it won't very helpful to use it here. Thirdly, the information in Neighborhoods can also be found in the 'neighborhood' column in Listings, so it is somehow redundant.

Name	Size	Discription
Listings	(7k, 96)	home information (location, price, amenities, etc.)
Calendar	(1700k, 4)	availability information (listing id, date, if that date is available, price)
Reviews	(300k, 6)	review content (listing id, date, reviewer id, comments)
Neighborhoods	(37, 2)	region/area names in a city

Figure 1: Data Description

It is not reasonable to use all 96 features, so I first fit a Lasso regression and a random forest to extract features that could explain more variances. I am able to reduce the features.

2.1 Linear assumptions

In the phase of data exploration, I aim to find more interesting features that are correlated with price. Before that I first explore the distribution for our response variable: price.

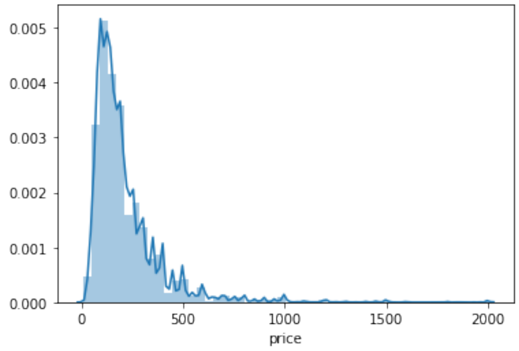


Figure 2: RF tuned parameters

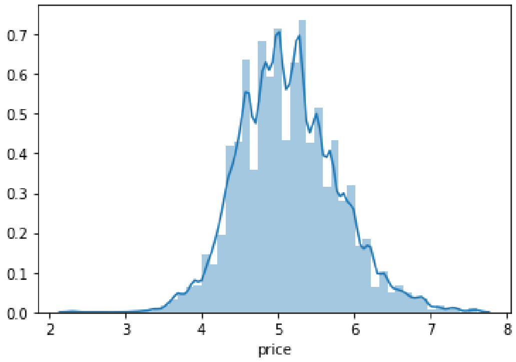


Figure 3: Log Price Distribution

The distribution of the listing price is highly skewed to the right, so to better detect its characteristic, we did a log transformation. This can also help make its distribution more normal, so we can apply models like linear regression on it.

However, when I use normal QQ-plot and fitted vs residual plot to check normality, the points do not fall about a straight line but with an uprising tail. The fitted vs residual plot indicates that the linear model will not be a good fit as the residuals do not "bounce randomly" around the 0 lines. This suggests that the assumption that the relationship is linear is not reasonable. The variance also shows a systemically decreasing pattern as the fitted value increases and I do see outliers from the plot.

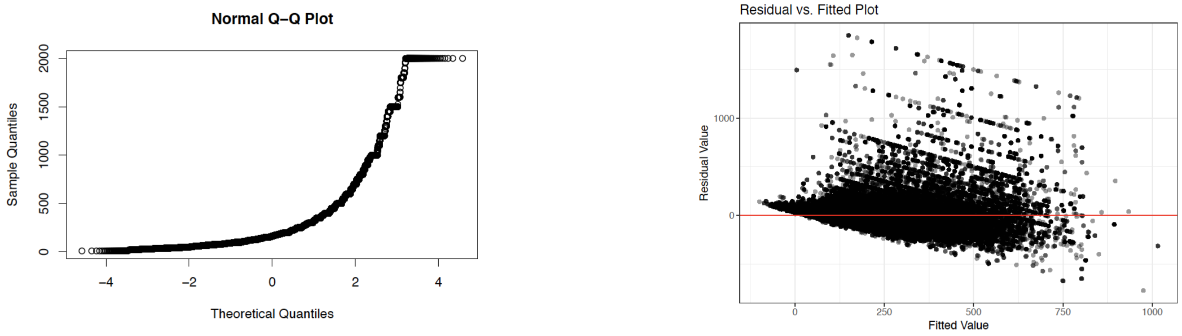


Figure 4: Linear model assumption check

2.2 Remove Outliers and Data Cleaning

Firstly, I use regex(regular expression) to removed "$" signs in the price column to make the column pure numerical. Then I use regex to remove commas in integers values more than a thousand such as 3,000 > 3000. One of the most time-consuming parts to handle missing values and outliers. I remove all the outliers in a few selected features. For example, I remove all the observations with minimum nights larger than 2000, with host total listings count larger than 86, with bathrooms larger than 3.5, with bedrooms larger than 6, with beds larger than 12. I also remove all the observations with a number of reviews equal to 0 because these might be outliers and some people may just open an account and list one of their houses and never log in again. Removing these people will help us focus on the observations that are actually more active on the platform. I also encode all the categorical variables into dummy variables. Moreover, from the Heatmap, I have four locations with the highest prices, I decided to also include distance features such as distance to Nob Hill, distance to Haight Ashbury, distance to Noe Valley, distance to Mission District. In addition to the distance to these four locations, since I assume location-wise information is very important to determine the price, the transportation facility will also be very important. Therefore, I decided to include the distance to the nearest BART station. I have talked to some of my local friends from the Bay area to confirm their knowledge of the pricey area in SF.

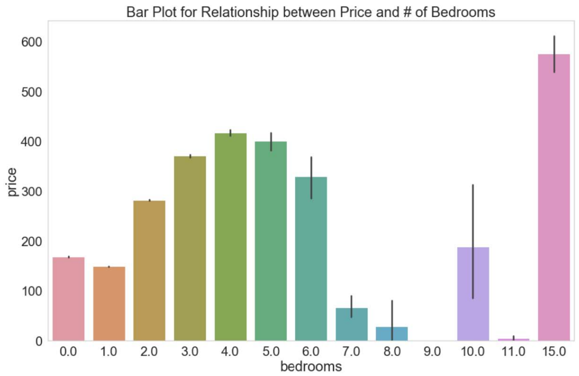


Figure 5: Number of of Bedrooms vs Price

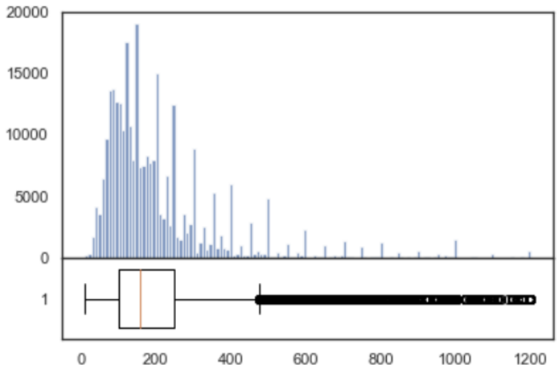


Figure 6: Outliers Cut

2.3 Features choice

Finally, after a thorough investigation of the variables in the data set, I am interested in host total listings count, host identity verified, neighborhood cleansed, latitude, longitude, property type, room type, accommodates, bathrooms, bedrooms, beds, bed type, amenities, minimum nights, cancellation policy, is business travel-ready, cleaning fee, number of reviews. Location information and property information is mostly related to price amongst all others. I conducted a small survey around my friends asking what are the most important features they care about when they are booking Airbnb for personal travel and group gathering(Students from BCSSA and CPU club)

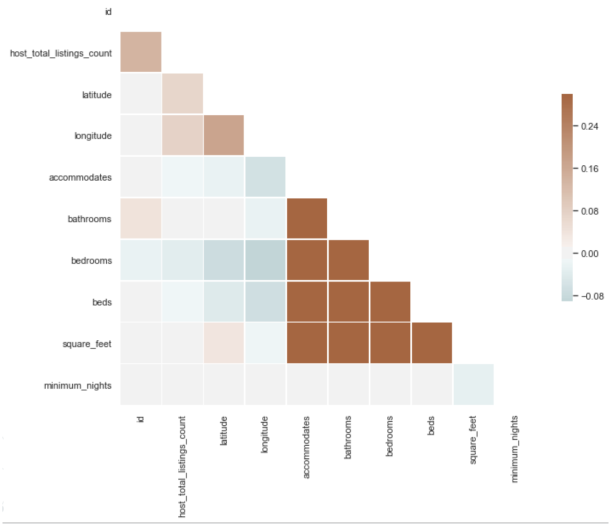


Figure 7: Features correlation plot

One aspect I believe is very important to look at is the correlation between the features. The col-linearity. Col-linearity is a condition in which some of the independent variables are highly correlated. From the correlation plot, we see that there is no need to omit features as they are not highly correlated with each other.

3 Methodology

3.1 Time series modeling

To model this time series, I start with the data itself. I observe that data in 2016-01, 2016-03, 2018-06 are missing, so I impute them by taking the mean of the surrounding 3 data points. As shown in Fig. 8, the data itself is so noisy. Therefore, I decide to filter the data using the Moving Average filter with window size equal to 5 because window size 7 seems to be too smooth and window size 3 seems to be too wiggly. Also, since all the variances for different months are almost the same, I don't need to scale back the difference by its variance if I use a simple linear trend to model this time series. I have tried many different linear (in parameters) models to fit this trend, such as Ridge/Lasso Regression with higher-order polynomial terms, with Gaussian features, with Sinusoid features. All of these do not differ much from a simple linear trend if we don't want to overfit in terms of backtesting cross-validation. I have also tried to fit a SARIMA model on this data but it turns out that SARIMA is too complex for this time series with only 36 data points, and even though we have monthly data in three years, each year still behaves quite differently so that it is too hard to capture an obvious pattern. Therefore, I simply just use a simple linear model over time to model this time series because of the reasons I state above.

To approach this goal, I combine 2016 to 2018 San Francisco listing data into a single data frame with around 272k rows and 17 columns. The variables of interest have both categorical and numerical variables, and some of them have outliers. Since some of the variables are correlated, as shown in Fig. 6, I believe that Random Forest[4] will give us better performance than Linear Regression. Specifically, our modeling procedure is that I first model the average price over time as a time series, I then apply Random Forest to do feature selection for price difference model, and then I apply Random Forest and Linear Regression using these selected features.

3.2 Obstacles

There are two biggest obstacles I have encountered in this project. The first one is the train/validation/test split. As this is a time-series data set, there are hosts of multiple listing but with the same content. This results in very high test accuracy in both training and test data. The second question is to find a suitable error metric: MAE, MSE, or R-square? Unlike the classification question where we could simply get accuracy to verify the model, the regression problem is harder since we have to determine our metrics.

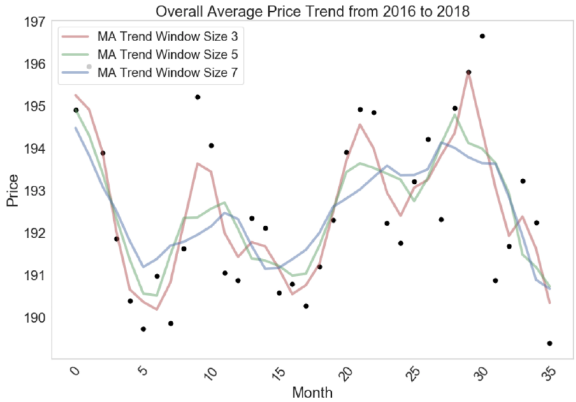


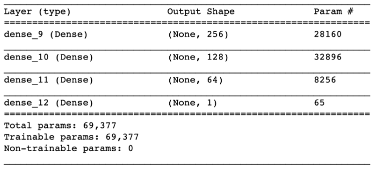
Figure 8: Time series modeling

3.3 Solution

There are a couple of methods I tried. Firstly, I use the train/validation/test sets by UserID. This ensures no duplication in the UserID but it might cause selection bias. Then, I tried to divide San Francisco into 16 grids and randomly sample unique ids in each grid to avoid the problem that the same entry appears in both train and test set along with stratifying by neighborhoods (This turns out to be complicated as I am not very familiar with the neighborhoods and there are neighborhoods with same names). Another aspect of the approach is to train on data points from within one month: Hosts are very unlikely to consistently modify their pricing list within a month. Lastly, I Separate validation/test sets into two price ranges(<=300 and >300) for more accurate error metrics measurement. I also decide to use MAE instead of MES(too much variation when you square the error).

4 Finding and Discussions

4.1 Neural Network

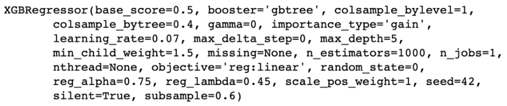


```
Layer (type)                 Output Shape              Param #
=================================================================
dense_9 (Dense)              (None, 256)               28160
_________________________________________________________________
dense_10 (Dense)             (None, 128)               32896
_________________________________________________________________
dense_11 (Dense)             (None, 64)                8256
_________________________________________________________________
dense_12 (Dense)             (None, 1)                 65
=================================================================
Total params: 69,377
Trainable params: 69,377
Non-trainable params: 0
_________________________________________________________________
```

Figure 9: NN Epochs = 8

The neural network is considered as one of the most "successful" deep learning methods for training classifications and regression problems. However, the complicated layers and hyper-parameters made it hard for individuals to operate on a personal computer. Therefore, I did not process in tuning hidden layers because the run time is too long. I set this as a baseline(naive approach) that the training MAE is 105.3914 and validation MAE is 107.92.

4.2 Gradient Boosting



```
XGBRegressor(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bytree=0.4, gamma=0, importance_type='gain',
       learning_rate=0.07, max_delta_step=0, max_depth=5,
       min_child_weight=1.5, missing=None, n_estimators=1000, n_jobs=1,
       nthread=None, objective='reg:linear', random_state=0,
       reg_alpha=0.75, reg_lambda=0.45, scale_pos_weight=1, seed=42,
       silent=True, subsample=0.6)
```

Figure 10: Gradient boosting tuning parameter

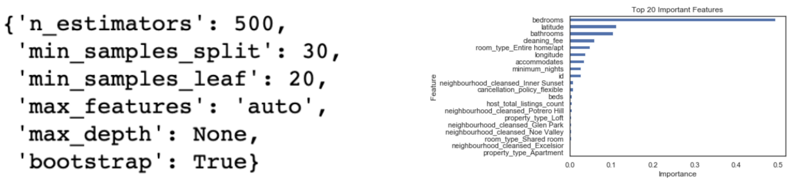
Reasons for choosing: It can benefit from regularization methods that penalize the algorithm and generally improve the performance of the algorithm by reducing over-fitting. The validation MAE for the combined is 61.40 and for For prices <= 300: 42.37 and for prices > 300: 168.26. The results supports my theory of the higher the price, the higher the variation.

4.3 Random Forest Without Feature Selections

Reasons for choosing: Random forest algorithm provides higher accuracy compare to the decision tree especially in the complex dataset. It will handle the missing values and maintain the accuracy of a large proportion of data. If there are more trees, it will reduce overfitting. Since some features are correlated as showed in the correlation plot, I believe that this should be the best model. After tuned hyperparameters using RandomizedSearchCV, I get validation MAE for Combined: 61.69, For prices <=300: 41.58 and for price >300: 174.55.

4.4 Random Forest With Feature Selections

Validation MAE: Combined: 61.72; For prices <=300: 41.54; For price >300: 175.03 which is a minor improvement for price less than 300 but a worse model for price larger than 300.



```
{'n_estimators': 500,
 'min_samples_split': 30,
 'min_samples_leaf': 20,
 'max_features': 'auto',
 'max_depth': None,
 'bootstrap': True}
```

(a) Hyperparameters (b) Feature importance

Figure 11: RF feature selections

5 Ethical considerations

Firstly, the Airbnb website does state that the data was after some sort of "modification" before publishing, but how much of the modification is enough? There is no such threshold for saying that. Also, when hosts make a listing on the Airbnb website, they have to click the "accept the privacy" to proceed as there is no way for them to deny and continue. With the easily accessed property records and geographic profiling, would that also make data revealing not only in the platform of Airbnb but also anyone who wants to access regardless of doing good or harm? Secondly, even with the modification, there is some data that cannot be re-identified. While Airbnb has to censor some data, it still needs to maintain certain information accurately to serve the purpose of research. If the information is not correctly used, for example, the combination of housing size, the listing of housings and telephone numbers could be metrics for promotion workers to target specifically on the products and pricing. Started two years ago, Airbnb uses has complained that their accounts being hacked, and unauthorized bookings charge and cancellations have been operated. This was extremely common for non-frequent Airbnb users. There are lots of ads on social media on "promotion for Airbnb booking" which in the end they steal the account and credit card information from the users. In the CSV file that Airbnb provided, there is a column userid which is a primary key (unique identification). By accessing the account and cross-reference with the public data, hackers can decode the blurring and blocking technique Airbnb uses for modification of the data and retrieve the original format. This is extremely dangerous since the user ID could be identified with name, gender, age, payment and a lot more. That is also saying, hackers do not need to access the actual Airbnb database to steal information. The public released data made everything easier for them. Overall speaking, nowadays, technical companies devoted lots of money hiring SWE to the security group to protect the data breach and information leaking. However, the data ethics problem remains in debate as the once a while "data privacy" topic being brought up on the table. We are all aware of the Facebook data privacy scandal two years ago and I have also believed that if you have used an App or have registered your personal information at some website, no matter it is their intention or not, you are just at risk of getting data privacy invasion at some point. This is unavoidable in the big data world and all we can do is to "trust" the platform and continue life. The model I am currently building is based on the validity of the data. If the data itself is not representative in the first place, then it will not be useful for hosts.

6 Conclusion and Future improvements

To summarize this project I want to say that it is very interesting to approach pricing prediction from a different perspective. We, usually as customers, do not take into accounts the issue of data privacy when we are traveling with

Airbnb. The amenities and price are what customers care the most about. On the other hand, as a host, we release everything about our property to the Airbnb database and it is on their integrity team to keep our data safe. In a word, the random forest is the best model for the Airbnb price prediction problem. This is a real-world data set and I do see lots of messy data (outliers, missing value, duplicates). When I am doing machine learning, most of the time there is no simple solution to resolve those problems. As for this project, if I have time, I would like to have more data to see possible cyclical trends. The model I fit is limited to be used in SF only. SF is considered a developed, compact urban city which means the model would not work in suburban areas. For future research, I would also like to add the NLP technique into the study for analyzing the positivity/negativity of the comments from customers. In the meantime, I would need a computer with better RAM and processor if I want to run more data. The current data is around 2.5GB and it took me more than 10 hours to tune the parameter by search grid command and it would not be realistic if I want to expand my areas outside SF.

7 Reference

[1] Available at http://insideairbnb.com/get-the-data.html.

[2] Available at http://insideairbnb.com/behind.html.

[3] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, Oct 2001. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[4] 3.17 Lecture slides: Introduction to Artificial Intelligence Machine Learning