Predicting Airbnb prices in San Francisco

Shabnam Attaran

shabnam.attaran@gmail.com

Overview:

Airbnb(Air bed and breakfast)

Airbnb is an online marketplace for arranging or offering lodging, primarily homestays, or tourism experiences. The company does not own any of the real estate listings, nor does it host events; it acts as a broker, receiving commissions from each booking.

Since Airbnb has started more people are using the platform to list their house/apartment/room… for short term rent. The host can rent the place from one night to months. Sometimes the host will rent their house when they are away for a week.

Business Objective:

Who is your client and why do they care about this problem? In other words, what will your client do or decide based on your analysis that they wouldn't have done otherwise?

The challenge that the host face is finding the right price for their listing. One way they can do it is by searching similar places and see what are those listed as and then estimate a price for their listing but this is not accurate and it does not include all the information. (for example the user may think only number of bedrooms is important whereas there are many factors in predicting the price)

So having a tool for predicting the listing price of an Airbnb that takes into consideration several factor (location, number of bedrooms, time of the year for listing,..) would help the host estimate their listing.

Data:

What data are you using? How will you acquire the data?

I am using the data from http://insideairbnb.com/get-the-data.html and it is free. They data is as of November 30 ,2019. It contains about 8533 record for San Francisco.

Briefly outline how you'll solve this problem. Your approach may change later, but this is a good first step to get you thinking about a method and solution.

I will use the listing.csv file for SanFrancisco that has n features. I will apply some data wraggling on this dataset to see how the data is and possibly eliminate features that are not relevant(those that will have only one value for example "experinces_offered has only none value", or those)

-Collecting data

-Apply data wrangling method

-Story telling

-Use machine learning techniques to predict the price of listing

Data Cleaning and Exploratory data analysis(EDA): :

1. What kind of cleaning steps did you perform?
2. How did you deal with missing values, if any?
3. Were there outliers, and how did you handle them?

Basically I did a lot of data cleaning in here. Also removed many features that were not useful.

Initially there were 106 columns:

- columns that have only one unique value were dropped('country_code', 'scrape_id', 'country', 'has_availability', 'experiences_offered', 'is_business_travel_ready')
- URL columns are not needed , so dropped those columns as well.('listing_url','thumbnail_url','medium_url','picture_url','xl_picture_url','host_thumbnail_url','host_picture_url','host_url')
- When the data was scraped in is not needed. They were from 2019-12-04 and 2019-12-05. So I dropped those columns as well.(last_scraped','calendar_last_scraped)
- There were 2 columns that had all null values so dropped those columns('host_acceptance_rate', 'neighbourhood_group_cleansed')
- Dropping more columns that are more text since I will not be using NLP ('name', 'summary', 'space', 'description', 'neighborhood_overview', 'notes', 'transit', 'access', 'interaction', 'house_rules','host_id','host_name', 'host_location', 'host_about', 'host_neighbourhood', 'host_verifications')
- some of the columns seems to have mostly one value(from looking at the histogram) so will drop those ('maximum_nights_avg_ntm' , 'minimum_nights_avg_ntm' , 'maximum_maximum_nights', 'minimum_maximum_nights', 'maximum_minimum_nights', 'minimum_minimum_nights','requires_license','require_guest_profile_picture','is_location_exact','host_listings_count','require_guest_phone_verification')

- There were 2 columns that had all null values so dropped those columns('host_acceptance_rate', 'neighbourhood_group_cleansed')- they were all missing so just dropped the whole column
- There are different columns for city('state', 'market', 'jurisdiction_names',’smart_location’, market,street,state,city) and since we are looking at SanFrancisco it didn’t make sense to keep all these.
  I wanted to confirm and see we don’t have any outliers before dropping these columns.
  There were a few outliers:
    - There was one with the city = San Jose and after looking at the data it seems wrong so I dropped that row
    - While the primary ‘market’ is San Francisco I saw one listing where the market was “D.C.”. That looked like an outlier to me but after further investigation it turned out the value for this column was not correct and the rest of the columns showed San Francisco, so I kept this row.

- After these checks I dropped the 'state', 'market', 'jurisdiction_names',’smart_location’, market,street,state,city
- #We can see all these columns are Null on the 54 rows where host_since is null .so I will drop these 54 rows.(host_since,host_response_time,host_response_rate,host_is_superhost,host_total_listings_count, host_identity_verified)

- dropping more columns regarding Host since our focus is to predict the price and we will not consider the relationship to the host for this problem('number_of_reviews','number_of_reviews_ltm','first_review','last_review','review_scores_rating','review_scores_accuracy','review_scores_cleanliness','review_scores_checkin','review_scores_communication','review_scores_location','review_scores_value','instant_bookable','cancellation_policy','calculated_host_listings_count','calculated_host_listings_count_entire_homes','calculated_host_listings_count_private_rooms', 'calculated_host_listings_count_shared_rooms', 'reviews_per_month','host_response_time','host_response_rate')
- There is one column called amenities which give information about different amenities that are provided in the listing , this had 186 different values in different listings so I parsed this and grouped some of them together and checked for each entry if they had that amenity and added new columns for those.Added 84 columns . Then went through them and if for example one was happening in less than 10, I removed that column. ('Mudroom','Ski_In_out','shared_amenities','Standing_valet','Warming_drawer','Fax_machine','Hammock','Air_purifier','Tennis_court')
- After parsing amenities column I dropped it.
- There was a column named “other” which does not give enough information so I dropped that.
- Availability

  There are different columns for availability and they seem to be correlated so I will keep only one of them and drop the rest. After doing further analysis it was not clear what this feature really do, so I dropped it.

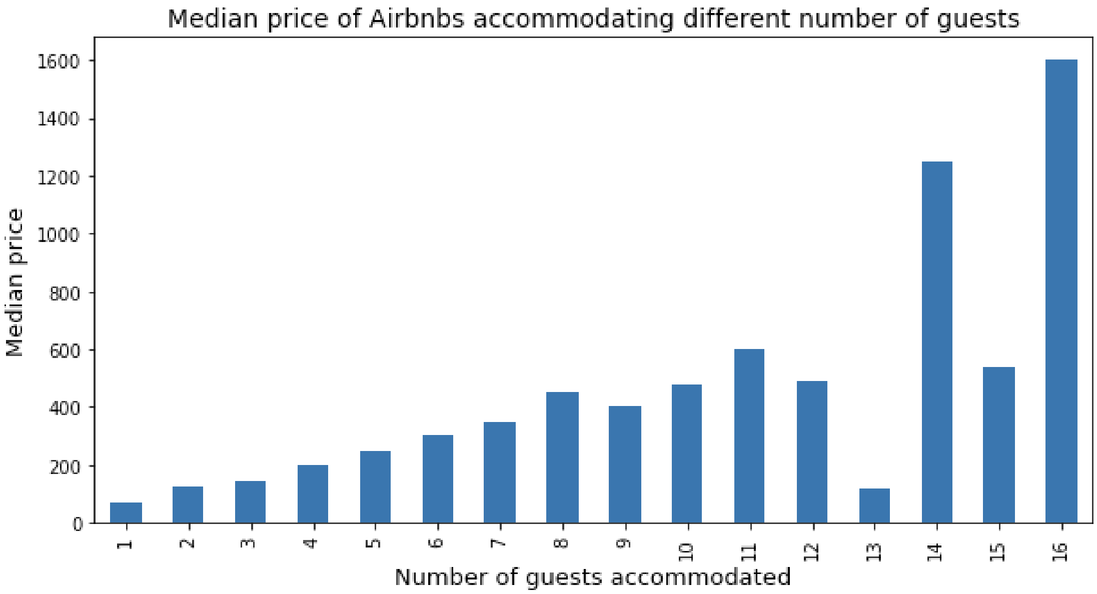
- cleaning Null values on Bedrooms,bathrooms and bed
  I will be putting Median for the missing values.(this was done during the model building)

- Cleaning price, security_deposit , cleaning_fee and extra_people

  Removing $ from Price and making it into int

-I did some EDA to see how different features are and how they related to the price.

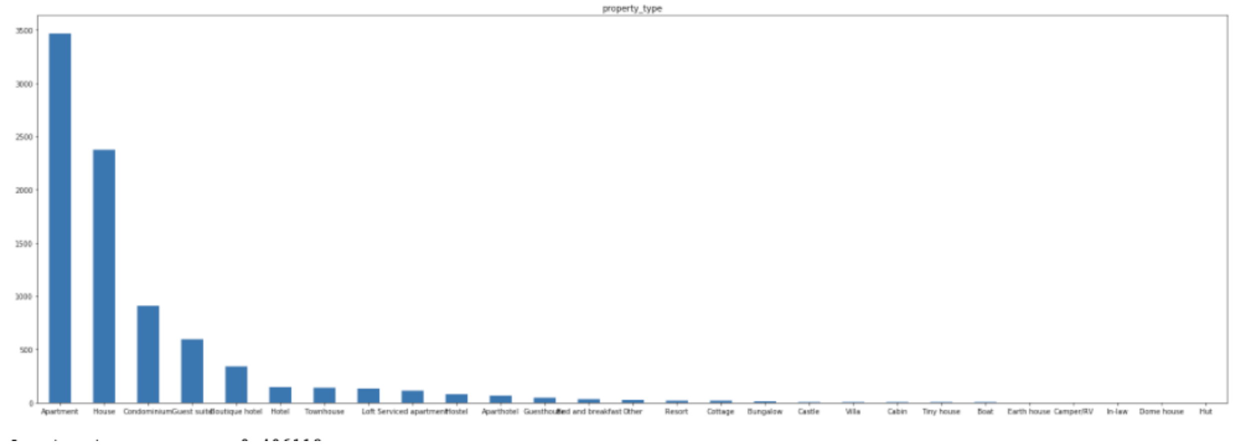
Here is how the number of guests a place accommodates related to the price:



I noticed when the accommodate is 13 the price suddenly drops so I wanted to check what made that.

There was only one record with accommodate =13 with 4 bedrooms and 3 bathroom so it it possible and I kept it.

-What are the most common properties that are listed in SF?



3 types make the majority of the properties and apartments are most common with almost half of the listings , following with house and condominion.

Apartment	0.406118
House	0.278364
Condominium	0.106657

Building the Model:

After a lot of Data Preparation (cleaning data, log transformation, one hot encoding and using standardScaler) I used Pipeline for my model building.

Definition of `pipeline` class according to scikit-learn is

Sequentially apply a list of transforms and a final estimator. Intermediate steps of pipeline must implement fit and transform methods and the final estimator only needs to implement fit.

As the name suggests, pipeline class allows sticking multiple processes into a single scikit-learn estimator. pipeline class has fit, predict and score method just like any other estimator

Then I split the data into training and test set and applied Lasso regression.

I used Gridsearch with cross validation of 10 fold and used R2 for evaluating the model.

After experimenting with the model, Some features seems to be highly related (zipcode, neighborhood and latitude/longtitude). So when choosing features I tried to only use one of them to compare the model. When only one of them were used the model performance was better compared to when all were used at the same time. But surprisingly none of these features were in the top features for predicting the price.

The features that had the highest coefficient are

Accommodates	22.870323
security_deposit	0.0423644
Cleaning_fee	0.3148890

Conclusion and recommendation:

Based on the model and the output I got this is not giving the most accurate result.

One of the thing I did not use in my model is anything related to the host since I was assuming if someone wants to put their property in Airbnb for the first time there wouldn’t be any previous knowledge of the host but it may be related for the hosts that are already in the platform and the reviews that have received on other properties. Those reviews and the information about the host can be used to see if the result will change.

This dataset may have not been enough to give the first time host the best possible model.

Also I was highly expecting the location to be an important feature in predicting the price which didn't seem so. That can be because of multiple reasons. We may have different prices in the same neighborhood depending on whether if it is the whole apartment or if it is only one bedroom that is being rented.

So for a future work , I recommend using features related to host and reviews and see the result based on that.