4741 Midterm Report

Jenny Li (jl2872), Yanyun Chen(yc2295), Jiwon Kim(jk2332)

November 2019

1 Introduction

The data set that we are investigating is the New York City Airbnb data from InsideAirbnb.com. We plan to use the listings and reviews data to predict the price of a given listing. Our data have similar features to homework_3. However, we do have a twice bigger size of raw examples and more features. We will build on the structure from that homework but with criticism.

2 Data Preparation

2.1 Data Cleaning

The listing.csv file has 48,377 rows and 106 columns. After a preliminary investigation into each of the columns, we decided to discard the following:

Name of Column Discarded	Reason of Discarding
”thumbnail_url”	Over 80 percent of data missing
”medium_url”	
“xl_picture_url”	
“square_feet”	
“weekly_price”	
“monthly_price”	
“license”	
“jurisdiction_names”	
“experiences_offered”	Columns have very low variance
“host_acceptance_rate”	
“host_id”,“host_url”,“host_name”	Column probably has low correlation with price
“host_thumbernail-url”, “host_picure_url”	Column contains url
“city” “state” “market”	Already narrowed location down to NYC
“has_availability”, “calendar_last_scraped”,	Value that depended on the time that Data was collected
“first_review”,“last_review”	Date probably has low relevance in predicting price
“host_neighbourhood”	Current location of host not relevant

Table 1: Discarded Columns

Then, we investigated the specific rows in our data. We found some examples with missing data: There are 21 examples having 0 information regarding the host, and there are also examples that are missing values in the columns ”bathrooms”,”bedrooms”, and ”beds”. We believe that these three columns would be very important in determining the price of the listing, so any data without values in those columns are discarded
After discarding the above examples, we are left with 48,249 rows.

2.2 Categorizing Data Types

Next, we grouped our data based on its data type.

Real-valued data:

- ”accommodates” “bathrooms”, “bedrooms”, “beds”,”guests_included”, “extra_people”,“number _of_reviews”, “number_of_review_Itm” “availability_30”, “availability_60”, “availability_90”, “availability_365”,“calculated_host_listings_count”, “calculated_host_listings_count_entire_homes”, “calculated_host_listings_count_private_rooms”, “calculated_host_listings_count_shared_rooms”

- “security_deposit”, “cleaning_fee”, “reviews_per_month” : treat empty value as 0, which is the correct interpretation under this setting.
- “minimum_nights” “maximum_nights” “maximum_maximum_nights”... : We should also consider which combination of data to use, either “minimum_nights_avg_ntm” and “maximum_nights_avg_ntm” or “minimum_nights” and “maximum_nights”.
- ”review_scores_rating”, ”review_scores_accuracy”, ”review_scores_cleanliness”, ”review_scores_checkin”, ”review_scores_communication”, ”review_scores_location”, ”review_scores_value”: A large amount of data are missing values for these columns (about 10600 examples). Furthermore, they have a very low variance (most review scores are about 8, 9 , or 10). As a result, we decide to exclude these features in our preliminary model. A possible further consideration could be to develop a separate system for listings that have values in these columns.

Text data:

- “name”, “summary”, “space”, “description” “neighborhood_overview”, “notes”, “transit”, “access”, “interaction”, “house_rules”, “host_location”, and “host_about”

  The text fields in this setting are filled out voluntarily by the host. So it does not make sense to interpret empty cells as missing data and directly discard them. Instead, we should treat them as a unique type of data (e.g. “no information provided”) when fitting our model.

Categorical data:

- “host_response_time” (5 categories), “host_response_rate” “property_type” (36 categories), “room_type” (4 categories), “bed_type” (5 categories) “cancellation_policy” (6 categories)

Set Data: “amenities“. We are considering encoding the amenities as a many-hot vector with weights

Location Data:

- “neighbourhood” (195), “street” (318), “neighbourhood_cleansed” (224), “neighbourhood_group_cleansed” (5), “smart_location” (319, no missing values), “latitude” “longitude”. A potential problem with latitude and longitude is that there is low variation in these two values since they are all located in the NYC area.

Boolean Data:

- “host_is_superhost”,“instant_bookable”, “is_business_travel_ready”, “requires_license”, “require_guest_profile_picture”, “require_guest_phone_verification”. In particular, the field “is_location_exact” could potentially be combined with the location data as well when we are making predictions.

3 Preliminary Analyses

We first performed a simple least-squared error model on only real-valued data. And we ended up with very large train/test MSEs: 54842 and 66676.
After plotting out true y and predicted y, we realize there exist some “outliers“ in our dataset (Figure 1). After doing some research on listings that are priced over $2,500, we found out that most of the hosts were doing so to avoid people from booking. So we excluded data points that have price over $2,500 and ran again (Figure 2). The resulting MSEs improved significantly: 13589 and 14497.

4 Testing Effectiveness of Model

We plan to split the data into training and testing data. Performing cross validation (for example, k-fold cross validation) on the data set could also provide insight on the performance of our model.

5 Preventing: Overfitting/Underfitting

To prevent overfitting, when we use a regression to fit our data, we could include regularization on the model. In the case of underfitting, combining features (e.g. using feature engineering), or using models with higher-order and complexity could potentially alleviate the problem. One potential options we could explore are decision trees.

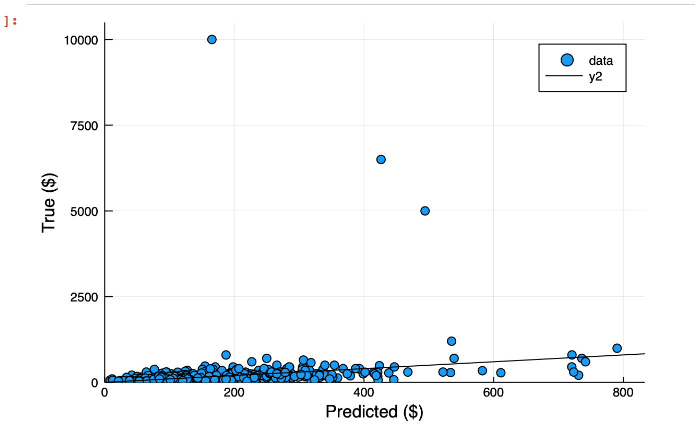

Figure 1: preliminary fitting with only real values

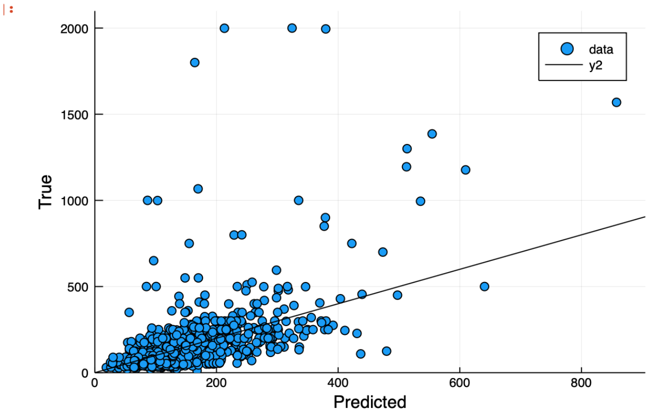

Figure 2: preliminary fitting with only real values and outliers excluded

6 Further development to the project

- We can see from Figure 2 that our linear model does not work well on houses that have a higher price. We could explore feature engineering to fit higher-order models to our data.
- For the problem with missing values of review scores, we could use data imputations. Specifically, we could use other features with non-missing values to first fit a model that predicts the review scores. Then, using this model, we could fill the missing review scores with the predicted review score values. After all values are filled in, we can then use this model to predict the price.
- So far we are not using the reviews and the summary data from listings.csv and reviews.csv. A potential way we could utilize these data is to use a pre-trained model on those data to provide more inputs to our model. (Perhaps detect positive words in the reviews and descriptions versus negative words)
- In the case of transit, many hosts put down descriptions such as ”10 minutes walk from...”, ”1 mile away from ...”, ”in the vicinity of...”. We could filter out examples with these flags in them, and use this as another input to train the model on.