Milestone Report: Airbnb, San Francisco listings

**I. Background**

When you want to go on a vacation or you have to move to a different city, you need to plan many things. One of the most important things you will need is a place to stay. If it's a short term stay, you can simply stay a few days at a hotel and if it's a long stay, you can stay at an apartment or buy a new house. There are many options but most people prefer that the cost is low and the place you stay at is homelike. In addition, there may not be any places to stay where you want to go. However, there will always be some type of housing, such as holiday cottages or houses/vacation houses. These will only be used by their owners when they are on a vacation so they will be free most of the time. There are also houses with unused rooms and other unoccupied housings that people ignore. Airbnb is allows people to lease or short term rent such lodgings.

**II. Problem and Customer**

There are listings in airbnb, especially in a big city such as San Fransisco. But how will we know which listing to choose? To figure this out, my project will focus on which neighborhoods and listings are the most popular, and what factors are associated with them. I use machine learning techniques to learn more about the listings.

**III. Data Cleaning**

To start cleaning the data set, I loaded the listings.csv file. The first thing I did to was delete columns that have only one possible value. This is because having a “True” for every listing would not be useful if I’m trying to compare the listings. Then I deleted columns that are useless. The second thing I did was delete rows that had a lot of NaNs. The third thing I did was turn categorical data into numerical data so that they would be useful for ML. Then I created new features such as review\_score\_totals\_ave, recent\_availability, license\_has, and transit\_has. Finally, I saved the file as listings\_cleaned.csv.

**IV. Questions**

1. Is there a linear relationship between the number of accommodates and the number of bedrooms?

**Null Hypothesis**: There is not a significant linear relationship between the number of accommodates and the number of bedrooms.

**Alternate Hypothesis**: There is a significant linear relationship between the number of accommodates and the number of bedrooms.

**Results**: coef = 0.7595, pval = 0

I used a Correlation Test so that I can learn if there is a linear relationship between the number of accommodates and the number of bedrooms. The results show that the Pearson correlation coefficient in the correlation test is .7595, which is closer to 1 and the p value is less than .05 so we reject the null hypothesis. This means that there is a linear relationship between accommodates and bedrooms. Knowing this shows that people tend to not care about the number of bedrooms. If there are a lot of people, they tend to get a house with more bedrooms rather than try to save money by getting the same about of bedrooms for different number of accommodates. This means that people in San Fransisco care about the overall experience rather than just using the house as a place to sleep.

1. Is there a linear relationship between the number of reviews per month and the number of reviews?

**Null Hypothesis**: There is not a significant linear relationship between the number of reviews per month and the number of reviews.

**Alternate Hypothesis**: There is a significant linear relationship between the number of reviews per month and the number of reviews.

**Results**: coef = 0.6388, pval = 0

I once again used a Correlation Test so I can learn if there is a significant linear relationship between the number of reviews per month and the number of reviews. The results show that the Pearson correlation coefficient in the correlation test is .6388, which is closer to 1 and the p value is less than .05 so we reject the null hypothesis. This means that there is a linear relationship between the number of reviews per month and the number of reviews. I tested this relationship because it is an indicator of popularity. Popular listings should have a larger number of reviews and a larger reviews per month whereas unpopular listings will have a lower number of both.

1. Does having lower prices make a listing more popular?

**Null hypothesis**: Popular listings have the same prices as non-popular listings  
**Alternate hypothesis**: Popular listings don't have the same prices as non-popular listing

**Results**: Ttest\_indResult(statistic=-4.2071077218452446, pvalue=2.9719290963239736e-05)

I used a 2 Sample T-test so I can learn if popular listings have the same prices as non popular listings. The p value is less than .05 so we can reject null hypothesis. This means that popular listings don’t have the same prices as non-popular listings. I then found that the average price of popular listings is $189 while the price of non popular listings is $496. This shows that cheaper listings are usually more popular.

1. Does having lower a larger summary length make a listing more popular?

**Null hypothesis**: Popular listings have the same summary lengths as non-popular listings  
**Alternate hypothesis**: Popular listings don't have the same summary lengths than non-popular listings

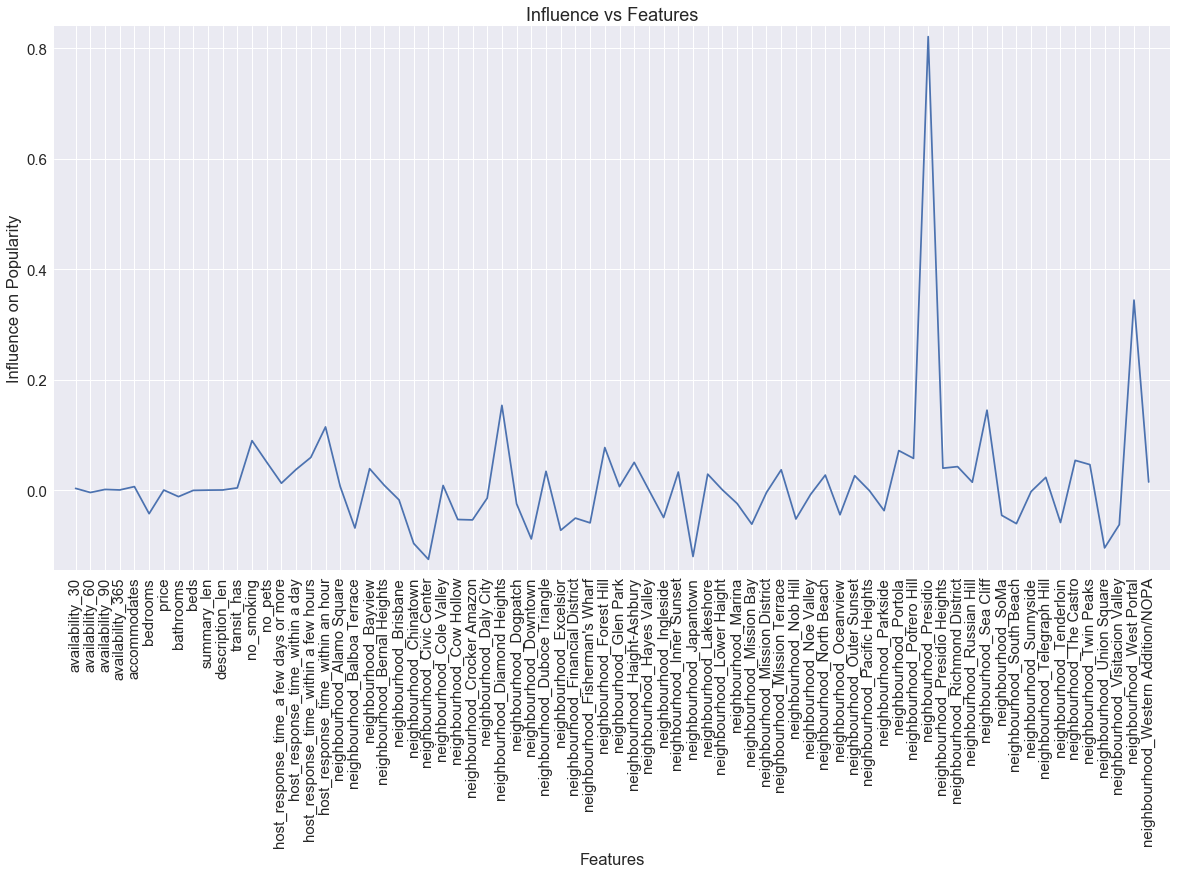
**Results**: Ttest\_indResult(statistic=-0.88781476250156488, pvalue=0.37492464455180641)

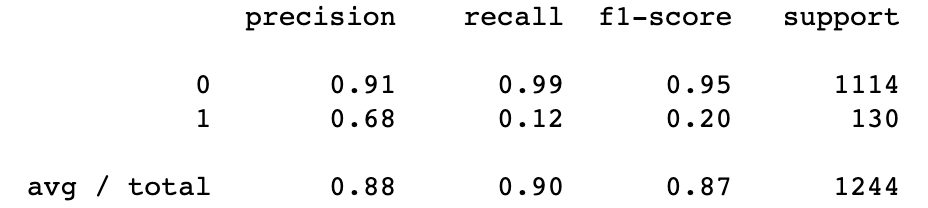
I used a 2 Sample T-test so I can learn if popular listings have the same summary length as non-popular listings. The p value is greater than .05 so the null hypothesis can’t be rejected. This means that popular listings have the same summary lengths as non-popular listings. The results show that summary length is not a factor in determining popularity of listings.

**III. Machine Learning process**

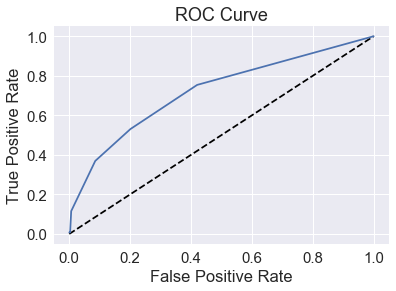
The first thing I did was clean up my data so that there were no missing values in the features I’m looking at. My goals in this section are as follows:

1. Train a KNN Classifier with high accuracy so we can accurately predict future trends.
2. Use lasso to find out which factors are associated with the most popular listings.
3. Make a classification matrix and find out the precision, recall and the f1-score
4. Find out the AUC scores using cross validation

First I trained a KNN Classifier on the following features: availability\_30, availability\_60, availability\_90, availability\_365, accommodates, bedrooms, neighbourhood, price, bathrooms, beds, summary\_len, description\_len, transit\_has, no\_smoking, no\_pets, host\_response\_time\_a few days or more, host\_response\_time\_within a day, host\_response\_time\_within a few hours, and host\_response\_time\_within an hour. I left out number\_of\_reviews since the target of the classifier is popularity, which is derived from the number\_of\_reviews. This would give reviews an unfair advantage in accuracy. Unless I am given the popular listings or I use a different predictor for popularity, I cannot use the number\_of\_reviews as an indicator for popularity. After creating a classifier with all of the above features with a train split of 80% and a test split of 20%, I got a .9019 for the knn score. This means a classifier with all of the features is about 90% accurate. This made me think how accurate the classifier would be if I only used the neighborhood, or only used the price, or different combinations of the features. However, all of these classifiers resulted in an accuracy of less than 90%. 

Second, I wanted to find out which of the above features affect the listings’ popularity the most. To do this, I used lasso regression analysis to see how much each feature influences the popularity. The results are shown in the figure above. The highest indicator of popularity is the neighborhood, Presidio, and the second highest indicator is the neighborhood, West Portal. The graph shows that most of the indicators that influence popularity are neighborhoods followed by pets and smoking policies.

Third, I wanted to know how much precision and recall the KNN Classifier has. To do this I used a classification report and a confusion matrix. The results showed that the TN = 1107, TP = 15, FN = 115, and the FP = 15. The classification report is shown above. The precision, recall, and the f1-score are above 90% so the KNN Classifier is very accurate in predicting popularity.



Fourth, I plotted the ROC Curve and calculated the AUC scores. The AUC scores computed using 5-fold cross-validation = [ 0.64186922 0.68629678 0.70105743 0.68076923 0.71992729]. The results of the cross validation are acceptable since they are mostly around 70%. The AUC is also closer to 1 than 0 so the model is is better at calculating True and False Positives.