Capstone Project 2: Analysis of Airbnb Listings in Chicago

By Vikram Singh

Introduction

Data used in this project: http://tomslee.net/airbnb-data-collection-get-the-data

The purpose of this project to take a deep dive into many Airbnb listings in the city of Chicago in order to determine actionable insights to increase customer satisfaction with regard to future listings. Airbnb is a platform which has amassed extensive popularity with travelers as an alternative to hotels, by providing unique options for lodging. Airbnb is currently present in 191 countries, clearly demonstrating a worldwide presence in conjunction with its popularity. In order to gain insights into the data behind this platform, I have chosen to take a look at one of the most popular cities for Airbnb in the USA, the city of Chicago.

Although it is certainly possible to do an analysis of Airbnb data which extends beyond Chicago, there are a few reasons I have chosen against such an approach. The most prominent being that each individual city should be treated independently of one another when analyzing data respectively. Factors such as tourism, location, population of a city are influential factors in an Airbnb listing, and to generalize these factors across many cities across a continent such as the United States would involve losing much of the integrity that could be placed into a case-by-case analysis. This holds even more true when analyzing cities in different continents because the cities in question can become even more radically different in terms of customer response due to the individual cities. To gauge information across the entirety of Airbnb's listings, would need to involve an aggregate of many case by case analysis by city or country, and this project represents one such analysis in Chicago.

Cleaning and Observing the Data

The data used in this project contains information for Airbnb listings from December 23rd 2013 to July 11th 2017 and is restricted to listings in Chicago. The features of this dataset include:

- Accommodates: The number of guests a listing can accommodate
- Bathrooms: The number of bathrooms a listing offers
- Bedrooms: The number of bedrooms a listing offers
- Borough: A subregion of the city or search area for which the survey is carried out. The borough is taken from a shapefile of the city that is obtained independently of the Airbnb website. For many cities, there is no borough information.
- City: The city in which a listing is located
- Host_id: Unique identifier given to each host on Airbnb's website
- Last_modified: Date/Time which values were read from the Airbnb website
- Latitude: The latitude for a listing as posted on the Airbnb website
- Location: The location of a listing as posted on the Airbnb website
- Longitude: The longitude for a listing as posted on the Airbnb website
- Minstay: The minimum number of nights required for booking a listing, as posted by the host.
- Neighborhood: The neighborhood in which a listing is located
- Overall_satisfaction: Average rating out of 5 that the listing has received from those visitors who left a review
- Price: The nightly cost in $US for a listing
- Reviews: The number of reviews a particular listing has received
- Room_id: Unique identifier given to each listing
- Room_type: The category to which the listing most closely belongs. Can be 'Private Room', 'Entire home/apt', or 'Shared Room'.
- Survey_id: Unique identifier given to each instance of the survey given to a customer following their visit at a listing.

The data used was provided in 26 separate .csv files. Before I could begin my analysis, it was necessary to format the data in a way which it could be worked with properly. Luckily, this was a simple task using the pandas library, which allowed me to read each file in and append them to one another with ease. To determine which features of the dataset would be useful in my analysis, I began inspecting the data to get an understanding of it, and gauge the quality of data found in the files.

Amongst the 18 features, a few things became apparent immediately. Features such as “City” and “Country” were completely filled with NaN’s, which was completely fine because the data clearly represented listings in the city of Chicago in the United States. Additionally, the “Borough” feature had 0 values, and is likely so due to Chicago not being an area traditionally divided into boroughs. It was an easy decision to drop these three features.

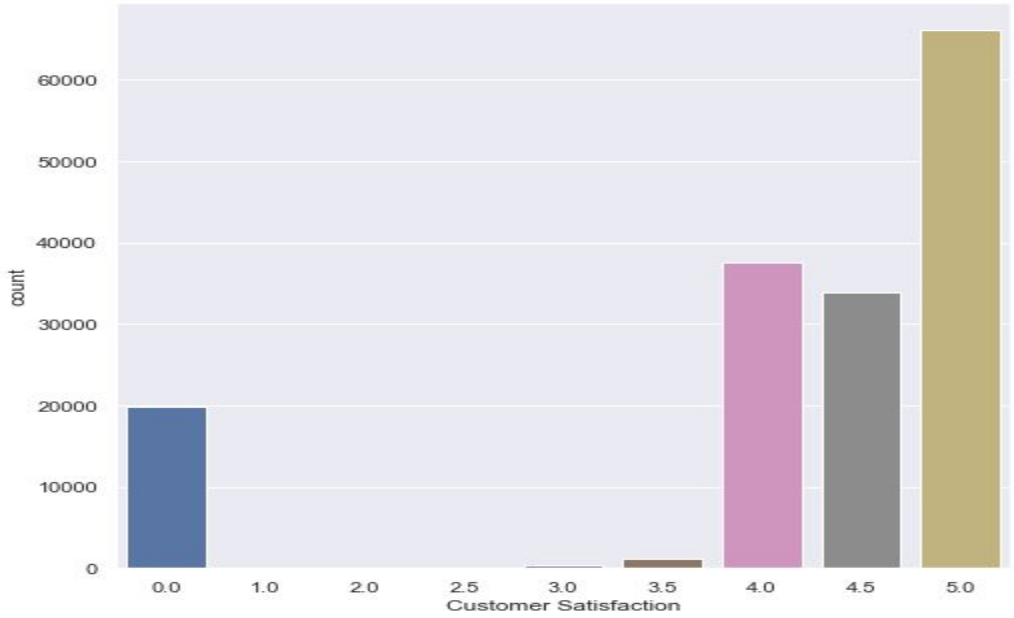
When inspecting other features, “Latitude”, “Longitude”, “Price”, “Reviews”, and “Room ID” had no missing values. The “Host ID” column only had 3 missing values, but this wasn’t worth looking into since my analysis wasn’t going to use such any ID value for anything other than identification purposes, if at all. However, there were four features which had missing values that required attention. For “Accommodates”, “Bedrooms”, and “Overall Satisfaction” which had 5668, 8804, and 31594 missing values respectively, I decided to fill the NaN values with the integer closest to their respective means. With “Accommodates” and “Bedrooms, this new value was equal to the median value for the feature, and provided insurance that the distribution of values would not be significantly altered following this change. Although overall satisfaction is the key metric of interest when determining customer response to a listing, using the mean seemed like the most fitting measure when gauging the whole of the data. As I would learn upon exploring the data, Airbnb listings are typically differentiated with regard to success mainly due to the presence of 0 star and 5 star ratings. Given that the rating for a listing is the calculated mean I decided to keep this trend consistent when examining this feature. For these reasons I filled in the many NaN’s of “Overall Satisfaction” with the value of 4, which is also very close to the pre-cleaned mean of 4.019. The “Minstay” feature had almost 40% of its data missing, with over 63000 NaN’s present. When dealing with this issue it became apparent that many listings likely not have a minimum required stay, and that this item is not required to be provided by a host upon creating a listing. With this assumption in mind, I filled in the missing

values for this feature with a value of 1, which would be the minimum number of possible nights somebody could stay at a listing at all. After making these changes, the data was finally ready for exploratory data analysis through application of several statistical methods and visualizations.

Exploratory Data Analysis

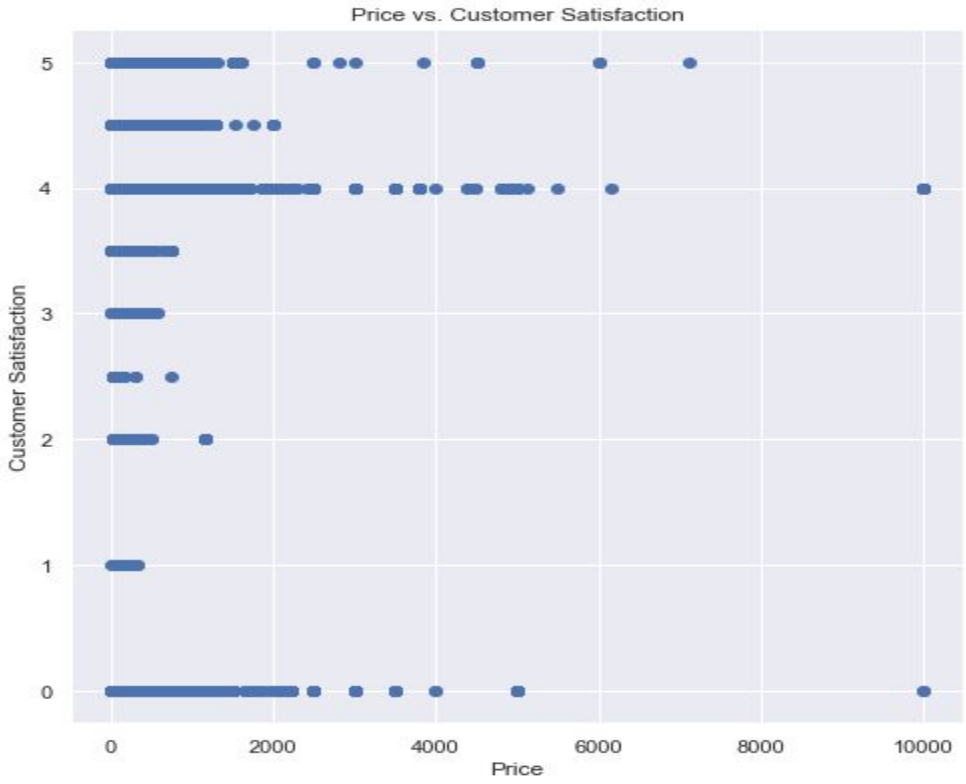
To begin exploring the data, I began taking a look at several relevant summary statistics for key variables, as well as analyzing distributions of the frequency of features I found interesting. Initially, I found that a listing most commonly accommodates two people by a massive margin, with 40.3% of the entries in the data accommodating two people. This proved to be no surprise when I observed that 66.3% of listings specified the number of bedrooms is 1, because the standard bedroom can reasonably be expected to host two people.

When observing the metric of interest, the overall satisfaction of a customer, I was surprised by the intense polarity I observed.



A simple glance at this count plot reveals that the region of customer ratings between 1.0 - 3.5 have very few entries. The most common rating is a 5 star rating with 41.5% of the data, followed by 4 star ratings with 23.6%. When considering this information, it is worth remembering that 84% of the values with a 4 star rating were originally NaN values which were given a value of 4 during the cleaning process. While there are a few thousand entries that fall in the 1.0 - 3.5 range, it is apparent that a strong majority of listings in Chicago are viewed by the customer as either an awful experience and are given a 0 star rating, or as an excellent experience and receive a 5 star rating. It would certainly be interesting to see how this trend holds across data from other cities in the USA, and across the world.

The price distribution is extremely right skewed, with 48.6% of listings having a nightly cost of less than $100, and 97% having a cost less than $500. There are a few listings which drive the range of values for this feature very high up, with the highest price per night being $10000. This type of listing is either extremely overpriced or a listing which is very luxurious and is not common by any means. With this in mind, I wanted to discern how customer satisfaction was related to the price of a listing.



This scatterplot reveals that generally, the distribution of price with respect to customer satisfaction contains more highly priced listings when looking at a 0 star rating. After we remind ourselves that 97% of listings in the data are below the price of $500, it becomes apparent from the graph that the 0 star ratings are saturated with prices above $500 a night. Further analysis reveals that 31% of the listings priced above $500 receive a 0 star rating. This provides credibility to the idea that many listings receive low ratings due to the customer feeling like they overpaid for the experience they had. It is also clear from the graph that there is a diverse spread of prices across all overall satisfaction ratings, which is to be expected for a metric on a platform with so many diverse listings.

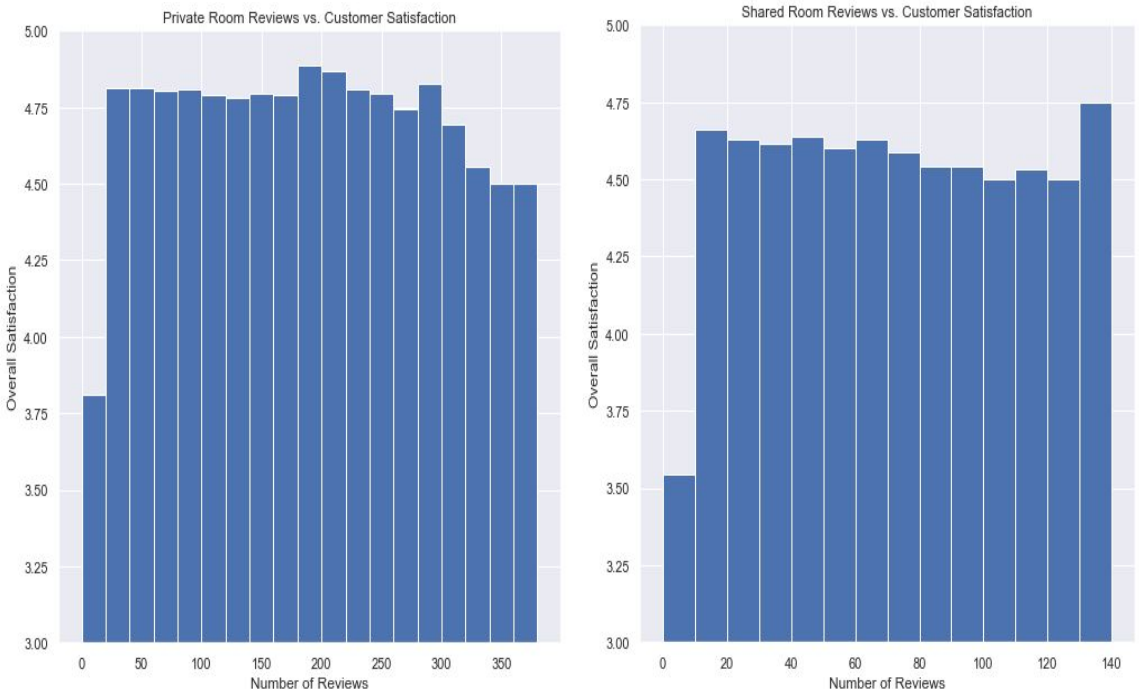
Following this preliminary analysis, I wanted to further compartmentalize the data based on the type of listing, or the “Room Type” feature. This is mainly because of the obvious differences in features, most notably price, I expected to observe across these subsets and I

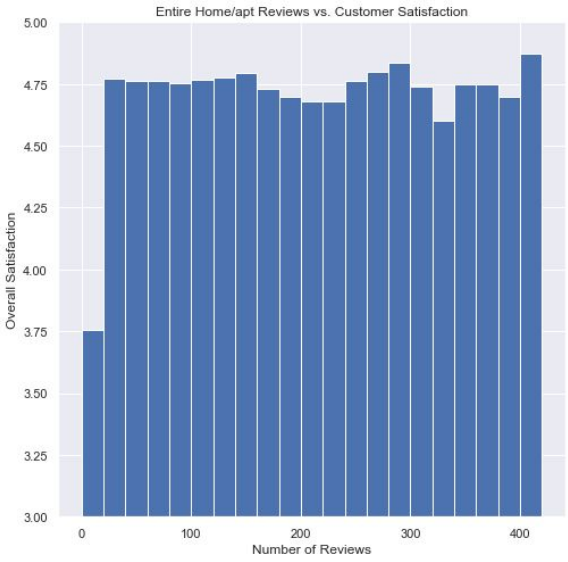
was interested to see how the overall satisfaction rating would subsequently be affected. I broke the data into three different sets corresponding to “Shared Room”, “Private Room”, and “Entire home/apt”, which had mean values for overall satisfaction of 3.89, 4.08, and 3.98 respectively. I then began searching for conditions which caused the overall satisfaction feature to reach extreme values. During this analysis I focused heavily on the “Reviews” feature, and noticed that the vast majority of “buckets” with numbers of reviews a listing has received, had very high customer satisfaction ratings of 4.5 - 4.8. This was very odd when compared to the means of these variables which were significantly lower. Further analysis revealed that a low number of reviews on a listing is highly correlated with a poor satisfaction rating across the board. Due to the nature of most buckets having very high average rating values, I was then able to determine that the distribution of reviews is also very right skewed, as shown with the following.

Descriptive/Relevant Statistics for Reviews

	Count	Mean # of Reviews	% of listings with 10 or less reviews	Mean Rating with more than 10 reviews	Mean Rating with 10 or less reviews
Private Room	58045	19.3	39.3%	4.8	3.58
Shared Room	6157	12	43.8%	4.62	3.57
Entire House/Apt	94969	16.3	43%	4.76	3.75

***NOTE* Graphs for Private Room and Entire House/Apt have xTicks of 20, instead of 10 as outlined in above table, for display purposes.**





It is apparent that having a low number of reviews is highly related to a low satisfaction rating, and although many of the buckets having ratings that are very high, the mean rating is dragged down by the high frequency of values which fall in the bucket of having 0 - 10 reviews.

This does not necessarily mean that more reviews implies a higher rating as we look at listings with many reviews. There is no clear upward trend of rating with respect to more reviews past the initial bucket, although it is worth noting that for “Shared Room” and “Entire house/apt”, the highest mean satisfaction rating does occur in the bucket containing the most reviews. From this analysis, we can definitively say that having very few reviews is bad for the rating of a listing, and that more reviews typically implies a rating above average but is not a determining factor.

Although reviews has proven to be an important variable, the price of a listing is undoubtedly one of the most important aspects of a listing. The next reasonable step was to observe the differences in rating within the different ranges and distributions of price across these subsets of data. As previously mentioned, 31% of 0 star ratings were amongst the 3% of listings priced above $500, which gave promise to the idea that price would be a highly influential variable.

Descriptive/Relevant Statistics for Price

	Mean Price	% of listings priced above the mean	% of listings priced below the mean	Average rating for listings priced above the mean	Average rating for listings priced below the mean	% of 5* ratings for listings priced above the mean	% of 5* ratings for listings priced below the mean
Private Room	$78.78	33.6%	66.4%	4.01	4.11	15.2%	29.1%
Shared Room	$57.95	30.2%	69.8%	3.63	4.01	8.59%	18.9%
Entire house/apt	$194.28	30.6%	69.4%	3.73	4.08	11.32%	29.3%

This analysis confirmed several hypothesis and assumptions that one would expect with Airbnb listings. A majority of the listings in every category are priced lower than the mean,

indicating a slightly right skewed distribution. There are also significant differences in the overall satisfaction when comparing these listings, with many more listings receiving 5 star ratings when priced below the mean. Although this might not sound surprising at first glance, this information implies that listings which may be priced higher to a more “high quality” listing are typically not as well received by the customer in many cases. This is confirmed by the frequency of 5 star ratings, and overall satisfaction rating being higher for listings across each type of room where the price is below the mean.

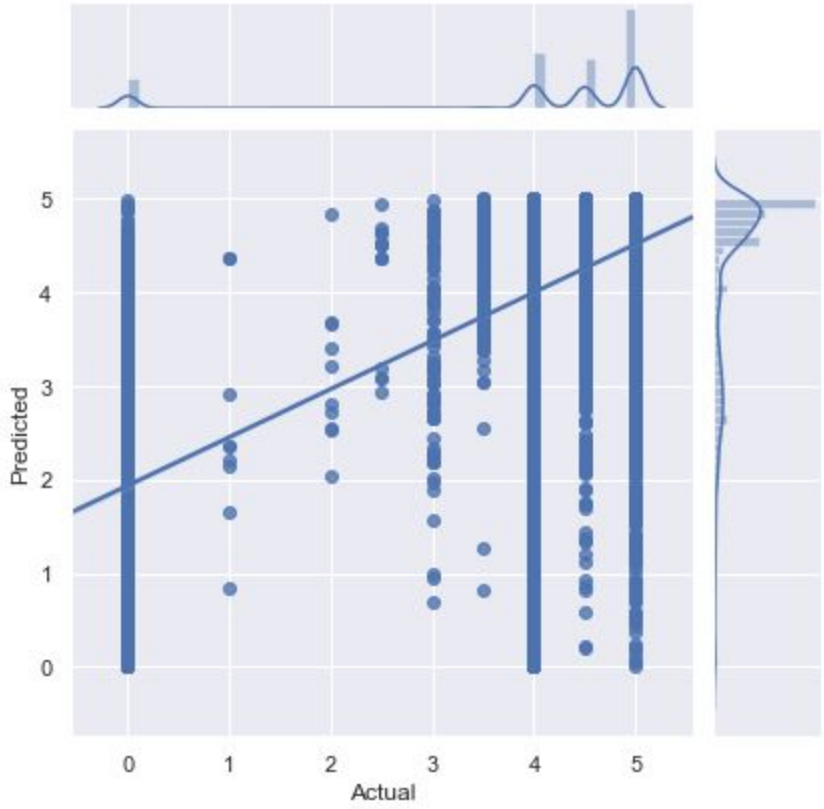
Machine Learning

Given my understanding of the data thus far, it was important to consider the most relevant quantitative variables with an appropriate model choice when applying machine learning. An analysis of the “minstay” variable did not reveal any significance, so I disregarded it. In addition, the latitude and longitude values were superfluous and did not provide any additional value. The “Room_Type” variable was clearly important based on my earlier analysis, so I included it after encoding this feature with dummy variables.

It was also important to consider what kinds of predictions I would be most interested in which were also feasible. I thought it would be interesting to see how well the model could predict an overall satisfaction rating, as well as the ability to classify if a given listing would receive a 5 star rating, since that is the most common and highest rating that can be given. With this in mind, the feature variables in my data are: “Accommodates”, “Bedrooms”, “Price”, “Reviews”, and “Room Type”, with a target variable of “Overall Satisfaction”.

Just out of curiosity, I fitted a Linear Regression model to the data and plotted the actual vs. predicted scores, and the results were awful. There were negative values, predicted values ranging from 5-8, and a correlation coefficient of 0.28. This served as a benchmark of

comparison for my next model, the Random Forest Regressor. After applying hyperparameter tuning to the number of estimators in the model, I found the best value was around 100. This new model was provided much stronger predictive capability, reflected in the correlation coefficient of 0.689 between the actual and predicted values.



As a scatter plot, it can be hard to discern much from the graph initially, since there are predicted values predicted across the spectrum of possible values. However, the density distribution reveals that the 5 star ratings are more correctly predicted with the presence of most values as 5 star, which reflects the actual data to an extent. This is also not a surprise, considering the MSE for the 5 star prediction mark is rather low when compared to the regression line. The issue with this model, is that there is a clear inability to accurately capture lower ratings, in particular the 0 star ratings. It was determined earlier that roughly 20,000 listings received 0 star ratings, but this does not reflect in the model due to the intercept of the

model being almost 2. This the main issue responsible for the R value not being able to improve much beyond 0.689, since lower rated listings will usually not be predicted correctly. This is mainly due to the heavy polarity in the dataset between ratings as a whole, and the addition of more ratings between 1.0 - 3.5, as well as more features would help improve the model performance.

When looking at my regression models, it was observed that lower rated models were less likely to have accurate predictions, and I was curious how this would hold up in the case of classification. When predicting if a listing would receive a 5 star rating, I started off with Logistic Regression. The model provided a True prediction if the listing was projected to receive 5 stars, and False otherwise.

Classification Report for Logistic Regression Model

	Precision	Recall	F1-Score	Support (Count)
False	0.61	0.91	0.73	23139
True	0.61	0.20	0.30	16657
avg/total	0.61	0.61	0.55	39796

Interestingly, the precision of the model is identical across both True and False predictions, sitting at 61%. A clear distinction can be made when observing False predictions being successfully recalled 91% of the time, compared to a small 20% for True predictions. The model clearly has a higher affinity to predicting listings which didn't receive 5 stars. Although the accuracy of True predictions is the same as False ones, many of the underlying True predictions aren't being captured and accurately predicted by the model, shown by the low True recall. This was a good benchmark for my next model.

The Random Forest Classifier has historically been a great model when I've used it, so it only made sense to attempt using it next. After building the model and tuning it appropriately, there was a clear improvement over the Logistic Regression model. Due to the higher predictive capability of my new model, I also chose to look into some key insights for feature importance as well. Note for feature importance that the various values of importance provided are with respect to one another and sum to 1, and the room type has been split into dummy variables.

Classification Report for Random Forest Classification Model

	Precision	Recall	F1-Score	Support (Count)
False	0.82	0.82	0.82	23139
True	0.76	0.76	0.76	16657
avg/total	0.80	0.80	0.80	39796

Feature Importance

Feature	Importance
Reviews	0.5951
Price	0.329
Accommodates	0.0493
Bedrooms	0.0196
Shared Room	0.0033
Private Room	0.00195
Entire home/apt	0.00153

This new model does a much better job of predicting new listings more accurately and comprehensively. Although it is still better at predicting when a listing does not have 5 stars, the discrepancy is much less severe and there is a much better recall present. During my

exploratory data analysis I placed the most emphasis on looking at reviews and price of listings, and many observable differences were found between the satisfaction of a customer with respect to these two variables. It is no surprise that these two variables have the most feature importance, although it is quite interesting that “Reviews” is higher than “Price”. After exploring our cleaned data and reviewing the results of the predictive models, there are a few recommendations to be made.