Duke University
May 1st, 2019
Team 10: Caroline Breaux, David Kohler, Karen Ou, Maahin Puri, Siyi Xu
CS 216 - Final Report

Final Report: Airbnb and the Real Estate Market

Motivation:

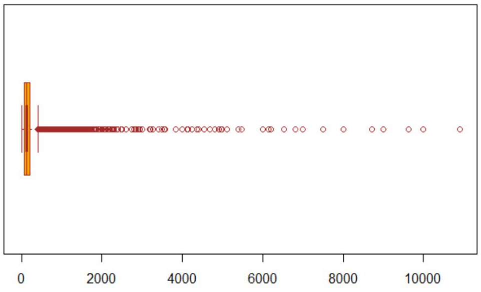
The rise of Airbnb since its founding in 2008 has fundamentally transformed the hospitality industry and the way in which people around the world are traveling. Its exponential growth in recent years relies heavily on its innovative digital solution to the traditional practices of home-renting. Curious about how the listing prices are determined, our team set out to develop a price prediction model that would not only shed light on which factors influence listing prices in different geographical locations, but also provide insight into the market potential of a "new" city if Airbnb were to expand into a city in which it had minimal host network. Furthermore, we brought in Zillow's real estate data to see if there was any correlation between Airbnb listing prices and home values across different cities.

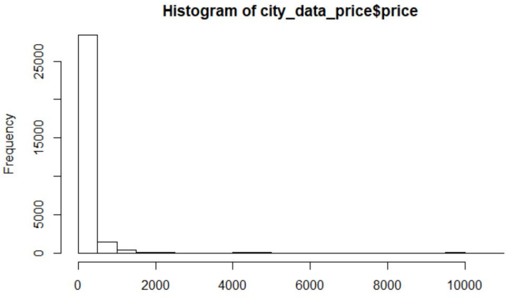
Methodology & Data Cleaning:

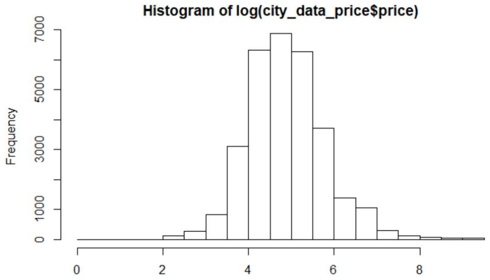
We used Airbnb datasets from four cities: Austin, Boston, Chicago, and Denver; and merged them all into one big dataset. We chose these cities at random, and believe that they give us a good variety of listings to allow for a somewhat robust model opposed to only using a single city. Initially, the datasets from Airbnb had almost 100 columns of data. After careful consideration, we eliminated most of the columns because they were either redundant or irrelevant (such as host name and max nights) and selected the following columns to use in our data exploration: listing id, whether the host is a super host, neighborhood, city, zip code, latitude, longitude, property type, room type, accommodates, bathrooms, bedrooms, beds, bed type, amenities, review score, num reviews, square feet, minimum nights, and price. Also, we brought in a Zillow dataset, containing over 15,000 zip codes from across the country and their corresponding Zillow Home Value Index (ZHVI), a smoothed, seasonally adjusted measure of the median estimated home value across a given region and housing type.

We started with some basic exploratory data analysis. For square feet, the dataset had 30,450 missing rows of data out of a total of 30,892 observations. While we believe this could be a very important variable in the future for determining housing prices, the lack of available data today meant we had to drop this column from our dataset. We also decided to drop the minimum night column because from some initial visualizations, we did not see a significant relationship between minimum nights and listing price. From there, we dropped all the rows that had missing (NaN) data. 'Is superhost' had 4 missing, 'zip code' had 258 missing, 'bathrooms' had 31 missings, 'bedrooms' had 12 missings and 'beds' had 25 missings. We decided to drop these missing observations rather than use a method such as mean imputation because after dropping all these missing rows, we still had 30,573 observations (which we believe is still plenty to build a more accurate model off of).

We then analyzed the predictor variable of price and found that there were some listings over $10,000 and some listings for $0. These obvious outliers were not very representative of the dataset as a whole, and so we decided to only use listings with a price within $10 and $2000 per night to build our models. Most of the outliers above $2000 come from Austin, specifically advertised for the SXSW music festival, which explains the very high listing prices. The distribution of price, however, was still skewed after removing our outliers, so we decided to use a log-transform to transform the price data. After removing our outliers, our final dataset contained 30,199 entries.







For property type, there were 39 property types listed in the original Airbnb data, so we decided to categorize them into just 6: Hotel (Hotel, Aparthotel, Boutique hotel, Hostel, Resort), Apartment (Apartment, Serviced apartment, Loft), Condominium, House (Bungalow, Cabin, Chalet, Townhouse, Cottage, Dome house, Earth house, Tiny house, Villa), Guest (Guest suite, GuestHouse), and Other (Barn, Boat, Bus, Camper/RV, Campsite, Casa particular (Cuba), Castle, Cave, Farm stay, Houseboat, Hut, Nature lodge, Others, Tent, Tipi, Treehouse, Yurt, Bed and Breakfast). After this, we used one-hot encoding to convert ‘property type’, ‘is superhost’, ‘room type’, and ‘bed type’ into a better form for our models .

Our next goal was to clean our ‘review score’ column, as there were more than 5,000 entries with missing data. The explanation of this was found in the ‘num reviews’ column, where we could see that these missing review scores were from listings that had only 0 or 1 reviews, and thus had not gained a review score yet. To deal with the missing data and outlier scores taking into account the influence of the number of reviews, we used the Wilson score confidence interval. We constructed a 95% confidence interval and used the mean of this interval to be our modified review score. We then assigned our entries with no review score to instead have the overall mean of this modified review score column (which turned out to be about 83). Below is the equation that we used to construct our intervals and modify our data:

$$\left(\hat{p}+\frac{z_{\alpha/2}^2}{2n}\pm z_{\alpha/2}\sqrt{[\hat{p}(1-\hat{p})+z_{\alpha/2}^2/4n]/n}\right)/(1+z_{\alpha/2}^2/n).$$

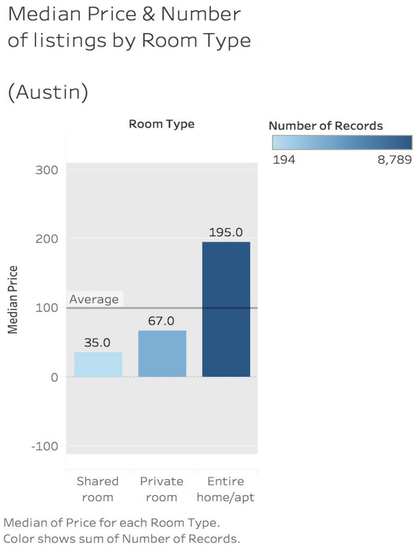
After dealing with the modified review scores, we moved on to the amenities columns, which originally was just a string for each row naming every amenities that the listing had. We cleaned these strings and made them into lists for each entry of our dataset and compiled a master set of all the amenities possible. From this, we built a dictionary to find the most common amenities in our data. There were a total of 195 unique amenities, so we chose to work with only the 20 most common ones. We then one-hot encoded the presence of these 20 most common amenities.

Lastly, we decided to use zip code as the key to merge our ZHVI data with our Airbnb dataset. This was the smallest and most precise metric that we could find housing data for using Zillow's research data. We brought in this Zillow data so that our model could be used on listings that did not solely come from the four cities we trained on. Although categorizing listings based on neighborhood would be even more precise, the Zillow dataset contained a limited list of neighborhoods, rendering neighborhood a less useful metric than zip code to link the Airbnb and Zillow datasets. Since not every zip code in the Airbnb dataset was present in the Zillow dataset, we also had to come up with a method to merge our data. To do this, we made a function that would check if the listing's zip code was in the Zillow dataset, and if so, would return the mean ZHVI for that zip code. If the zip code was not in the Zillow dataset, it would check if the listing's city was in the Zillow dataset, and if so, would return the mean ZHVI for that city. If the city was not listed in the Zillow data, it would then check the state, and if the state was found, would return the mean ZHVI for that state. If even the state was not found, it would return the mean ZHVI for the entire Zillow dataset. Once we had the datasets merged, we noticed a very large range of ZHVI values, and so decided to use min-max scaling to scale the data down.

Visualizations:

We did some exploratory data visualizations in Tableau to learn more about trends in our data and to guide us in answering the questions we were trying to answer: What factors have a significant contribution to price? Are there some neighbourhoods or zip-codes that are significantly more expensive than the average? If so, what factors influence those prices? In other words, do we need to find out a way to incorporate neighbourhood for our price prediction model?

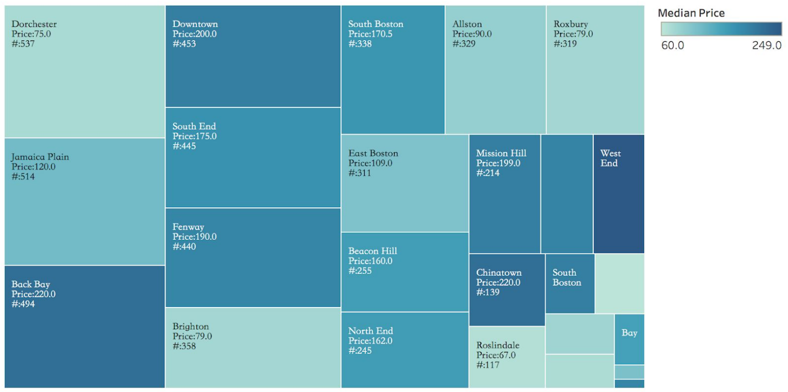
The visualization to the right shows the median price of listings by room type and the number of listings in Austin, and the same type of graph was made for the other three cities as well. The Airbnb dataset had three types of rooms: Shared room, Private Room and Entire home/apt. First thing to look at is that the number of records is significantly higher for Entire home/apt compared to private rooms and shared rooms. That is, there are more listings of entire homes or apartments, which holds true for all four cities. We initially thought that since it is easier to list a shared room or a private room, there would be more listings for those room types, but it makes sense for the number of listings of Entire home/apt to be higher since people are generally more comfortable in renting out entire homes and apartments. Furthermore, it can



be seen that the price of entire homes and apartments is significantly higher than those of private rooms and shared rooms, so whether or not a listing is an entire home/apartment might be a significant price determining factor. The grey line in each graph shows the average listing price for all three room types. The price of entire homes/apartments is around 95% higher than the average listing price in all four cities. In these graphs, the average price might vary across cities (Austin and Boston have an average price of over $100; the average price of Denver and Chicago is around $75). Nevertheless, the relationship between room type and price as well as the relationship between number of listings and price remain consistent across the four cities.

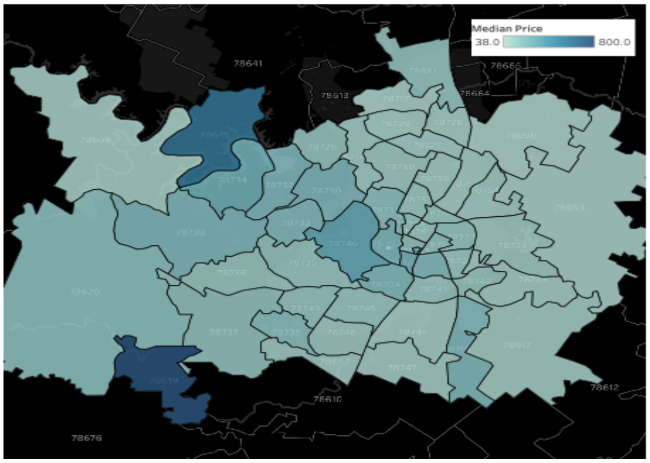
Another visualization that we made looks at the relationship between neighborhood and price. The visualization to the right shows the neighborhoods in Austin included in the Airbnb dataset. The size of each rectangle represents the number of listings in a particular neighborhood, so Dorchester has the highest number of listings followed by Jamaica Plain, with 537 and 514 listings respectively. The neighborhoods are color-coded such that a darker color indicates a higher average price. We also mapped the median price based on zip codes across the four cities to display the data in a geographical way as seen in the map titled “Median Airbnb Listing Price in Austin By Zip code”, however, it is a broader representation of the graph above.

Records and Median Price by Neighbourhood in Austin



Neighbourhood Cleansed, median of Price and sum of Number of Records. Color shows median of Price. Size shows sum of Number of Records. The marks are labeled by Neighbourhood Cleansed, median of Price and sum of Number of Records.

Median Airbnb Listing Price in Austin By Zip code



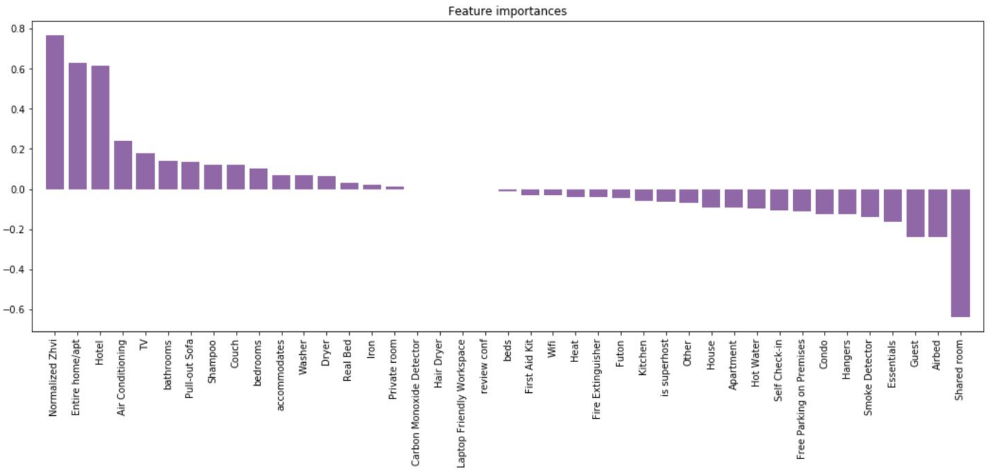
Another important conclusion is that the listing price varies significantly across neighborhoods and zip codes, so location is an important contributor to price. It can also be seen that price and number of listings in a particular neighbourhood do not seem to have a significant correlation. The neighborhoods that have more Airbnb listings do not necessarily have higher median prices. For instance, West End has a much higher median price compared to Dorchester but has significantly fewer listings. Although these conclusions are consistent across all four cities, it is important to note that our team ultimately opted to use zip-codes rather than neighbourhoods in our model due to missing neighbourhood data for a variety of listings. Also, the Zillow data used zip codes which made it easier to link the two datasets together.

Results & Modeling:

To model our data for the price prediction, we decided to use a variety of packages and approaches. Specifically, we used Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regressors, Random Forest Regressors, and Support Vector Regression (SVR). For all of these approaches, we used a k-fold Cross-Validation with a k value of 10 to train our models optimally. We also dropped some of the columns of our dataset that we didn't think were important for our models, so that our final dependent variables were 'is superhost', 'accommodates', 'bathrooms', 'bedrooms', 'beds', room type' (one-hot encoded into 3 columns), 'bed type' (one-hot encoded into 5 columns), 'property type' (one-hot encoded into 6 columns), 'normalized ZHVI', 'amenities' (one-hot encoded into 20 columns), and 'review conf' (which was our modified review score). Aside from the listing-specific data, we also included the guest reviews in our analysis to see if the most frequently occurring words would reveal anything about which attributes of a listing the guests cared most about.

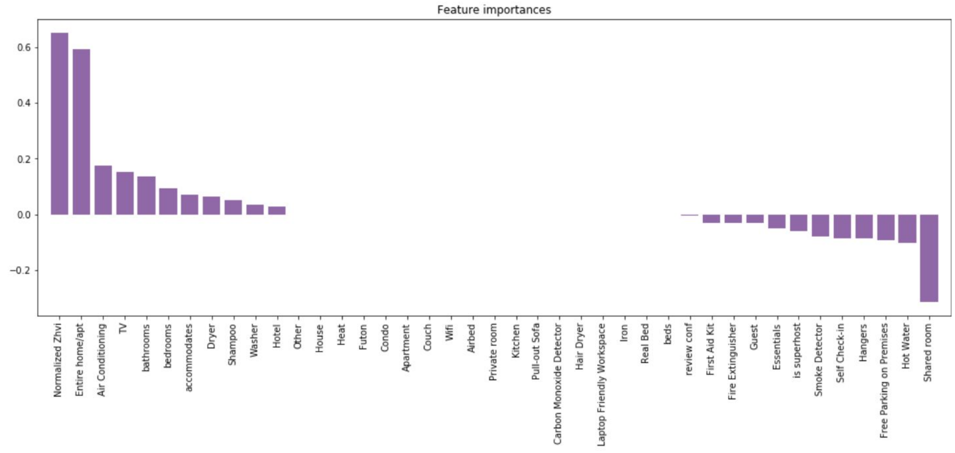
Linear Regression: using the LinearRegression package from the sklearn library, we were able to build a model resulting in a 0.56 r-squared and 0.31 mean squared error. The Airbnb listing price appeared to have strong, positive correlations with some of the variables, such as the Zillow House Value Index (ZHVI) and Entire home/apt, and strong, negative correlations with others, such as whether or not the room is a shared room. The model is shown below with the coefficients for all the variables which are also visually represented by a bar graph:

$$\begin{aligned}\text{logprice} = 4.05 &+ (-0.066725) \times \text{is. superhost} + 0.068651 \times \text{accommodates} + 0.139422 \times \text{bathrooms}\\ &+ 0.100926 \times \text{bedrooms} + (-0.009845) \times \text{beds} + 0.625996 \times \text{Entire home apt}\\ &+ 0.007420 \times \text{Private room} + (-0.633416) \times \text{Shared room} + (-0.224364) \times \text{Airbed}\\ &+ 0.097956 \times \text{Couch} + (-0.036596) \times \text{Couch} + 0.127272 \times \text{Pull out Sofa}\\ &+ 0.035733 \times \text{Real Bed} + (-0.087978) \times \text{Apartment} + (-0.121576) \times \text{Condo}\\ &+ (-0.234617) \times \text{Guest} + 0.598255 \times \text{Hotel} + (-0.088035) \times \text{House}\\ &+ (-0.066049) \times \text{Other} + 0.764261 \times \text{Normalized Zhvi} + (-0.029826) \times \text{Wifi}\\ &+ (-0.038886) \times \text{Heat} + (-0.161803) \times \text{Essentials} + (-0.137893) \times \text{Smoke Detector}\\ &+ (-0.059792) \times \text{Kitchen} + 0.238660 \times \text{Air Conditioning} + (-0.126428) \times \text{Hanger}\\ &+ 0.121606 \times \text{Shampoo} + 0.179469 \times \text{TV} + 0.066489 \times \text{Washer} + 0.063873 \times \text{Dryer}\\ &+ (-0.001695) \times \text{Laptop Friendly Workspace} + 0.023393 \times \text{Iron}\\ &+ (-0.040146) \times \text{Fire Extinguisher} + (-0.099524) \times \text{Hot Water}\\ &+ (-0.112362) \times \text{Free Parking on Premises} + (-0.105954) \times \text{Self Check in}\\ &+ (-0.029816) \times \text{First Aid Kit} + (-0.002957) \times \text{review conf}\end{aligned}$$



Ridge Regression: Since we had 42 features in the model (due to one-hot encoding), we tried to improve the previous linear regression model by using ridge regression. Using the GridSearchCV from the sklearn library, we calculated the best lambda score to be 7.087370814634009 (without normalization). The ridge regression model turned out to have a similar performance as our linear regression model, with 0.56 r-squared and 0.31 mean squared error. The coefficient for each variable in the ridge regression is basically the same as in our linear regression.

Lasso Regression: We also tried using lasso regression in order to build a better linear model. Using GridSearchCV from sklearn library again, we found an alpha of 0.005. Using the Lasso() function to fit the model, we ended up with 0.54 r-squared and 0.32 mean squared error. The r-squared and mean squared error were not improved, but it is interesting to note that using this method, we ended up with fewer variables in our model. A bar graph showing the feature importances is shown below:



Feature importances: 1. Normalized Zhvi (0.652426) 2. Entire home/apt (0.593729) 3. Air Conditioning (0.174082) 4. TV (0.153522) 5. bathrooms (0.135890) 6. bedrooms (0.093266) 7. accommodates (0.071250) 8. Dryer (0.062780) 9. Shampoo (0.050783) 10. Washer (0.034809) 11. Hotel (0.028261) 12. Other (0.000000) 13. House (-0.000000) 14. Heat (-0.000000) 15. Futon (-0.000000) 16. Condo (-0.000000) 17. Apartment (0.000000) 18. Couch (-0.000000) 19. Wifi (-0.000000) 20. Airbed (-0.000000) 21. Private room (-0.000000) 22. Kitchen (-0.000000) 23. Pull-out Sofa (0.000000) 24. Carbon Monoxide Detector (0.000000) 25. Hair Dryer (0.000000) 26. Laptop Friendly Workspace (-0.000000) 27. Iron (0.000000) 28. Real Bed (0.000000) 29. beds (-0.001838) 30. review conf (-0.003651) 31. First Aid Kit (-0.029830) 32. Fire Extinguisher (-0.030771) 33. Guest (-0.031215) 34. Essentials (-0.051118) 35. is superhost (-0.060285) 36. Smoke Detector (-0.080182) 37. Self Check-in (-0.085180) 38. Hangers (-0.087273) 39. Free Parking on Premises (-0.091698) 40. Hot Water (-0.100741) 41. Shared room (-0.315594)

Decision Trees: The next model that we tried to use was a decision tree regression model. Using the DecisionTreeRegressor package from the sklearn library, we ended up with 0.53 r-squared and 0.33 mean squared error. The r-squared and mean square error were not improved and in fact both turned out to be worse than those of our normal linear regression model from earlier.

Random Forest: In an attempt to improve the decision tree model by reducing error due to variance, we used the random forest regressor from the sklearn library, which fits multiple decision trees on various subsamples of the dataset and uses averaging to improve the predictive accuracy and control overfitting. 100 random decision trees of depth 5 were included in the random forest. The result, however, showed slightly deteriorated performance compared to the decision tree model - we obtained a r-squared value of 0.52 and a mean square error of 0.34.

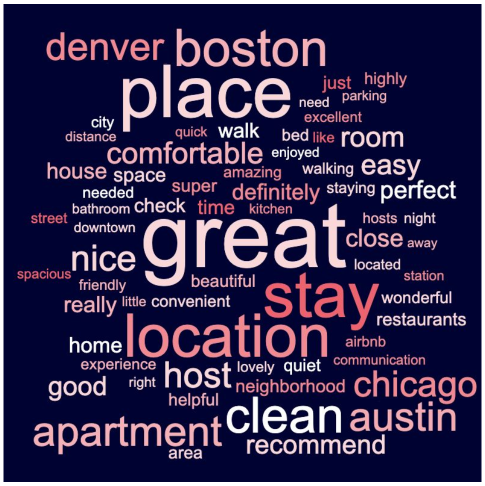
SVR: The last model we tried was the support vector regression model imported from the sklearn library. A linear kernel was used, and the rest of the parameters were left as default. The predictive accuracy turned out to be the same as that of the decision tree model, with a r-squared of 0.53 and a mean squared error of 0.33.

The r-squared values and mean squared errors of the 6 regression models are tabulated below:

Type of Model	**R-Squared Value**	**Mean Squared Error**
Linear regression	0.56	0.31
Ridge regression	0.56	0.31
Lasso regression	0.54	0.32
Decision trees	0.53	0.33
Random forest	0.52	0.34
Support vector regression (SVR)	0.53	0.33

Based on these r-squared values and mean squared errors, we concluded that our prediction model should be built using linear regression, although it was deemed the simplest among all the models. Chances are that other models would have yielded higher predictive accuracy had the appropriate model parameters been finely adjusted. We were unable to find the optimal combination of parameters for each model due to the limited computational capacity of the virtual machines on which these models were run.

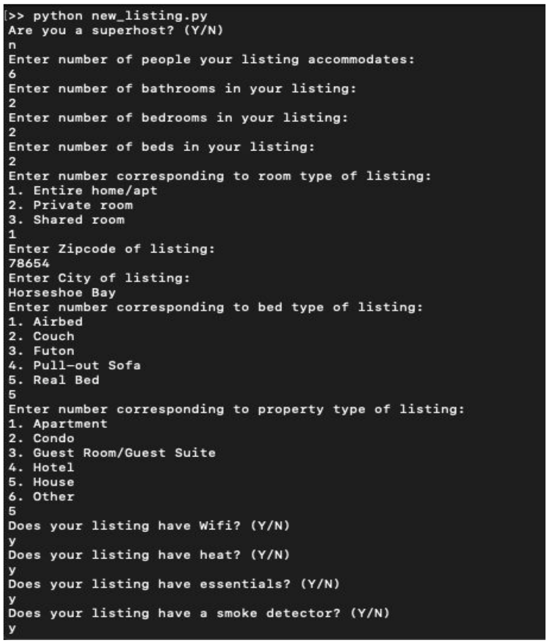
Guest Reviews: The TfidfVectorizer from the sklearn library was used to analyze the Airbnb guest reviews, and the most frequently occurring words were found based on term frequency, or the tf score, in the four chosen cities. In order to extract the key words, stop-words = ‘english’ was used to filter out the common stop words in the English language which have little meaning. Additionally, the minimum and maximum document frequencies were set to be 0.05 and 0.90, respectively, to filter out words that appeared in less than 5% or more than 90% of the reviews. After the reviews had been processed, the keywords were ranked from most to least frequently occurring based on the calculated term



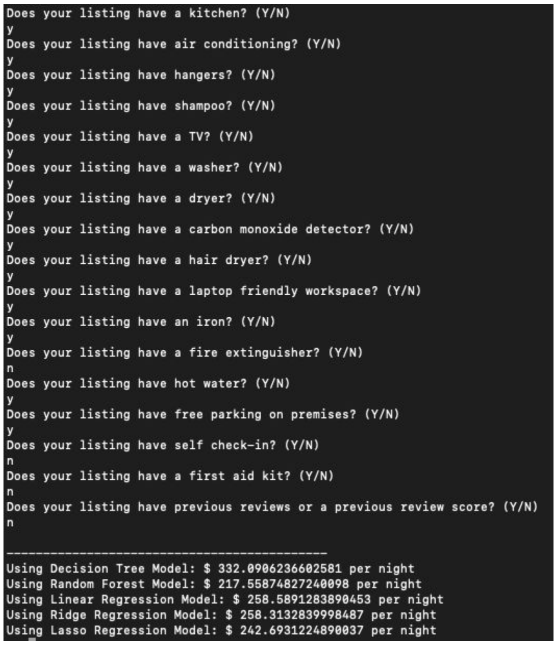
frequency. The results are visually represented in a word cloud included on the previous page, and it is evident that certain keywords coincide with the variables found in the regression models, including location and apartment. We can also infer that guests value cleanliness, comfort, and convenience since words like “clean”, “comfortable”, and “convenient” are frequently mentioned by reviewers.

Conclusions:

We found it surprising that the linear regression model outperforms other more sophisticated models in terms of prediction accuracy. The fact that the normalized Zillow Home Value Index has the largest, positive coefficient in the linear regression model indicates that the real estate value largely influences the Airbnb listing price. Also, contrary to our hypothesis, being a superhost is negatively correlated with the Airbnb listing price. We also created a python script that can be ran by our consumer to predict how much they should charge per night for their listing. We used the pickle library to save and export our models so that after the user has answered all of the prompts, it passes the gathered data into our models, returning the estimated price each of the models predict the consumer should charge per night. The following is what the user sees:



```
>> python new_listing.py
Are you a superhost? (Y/N)
n
Enter number of people your listing accommodates:
6
Enter number of bathrooms in your listing:
2
Enter number of bedrooms in your listing:
2
Enter number of beds in your listing:
2
Enter number corresponding to room type of listing:
1. Entire home/apt
2. Private room
3. Shared room
1
Enter Zipcode of listing:
78654
Enter City of listing:
Horseshoe Bay
Enter number corresponding to bed type of listing:
1. Airbed
2. Couch
3. Futon
4. Pull-out Sofa
5. Real Bed
5
Enter number corresponding to property type of listing:
1. Apartment
2. Condo
3. Guest Room/Guest Suite
4. Hotel
5. House
6. Other
5
Does your listing have Wifi? (Y/N)
y
Does your listing have heat? (Y/N)
y
Does your listing have essentials? (Y/N)
y
Does your listing have a smoke detector? (Y/N)
y
```



```
Does your listing have a kitchen? (Y/N)
y
Does your listing have air conditioning? (Y/N)
y
Does your listing have hangers? (Y/N)
y
Does your listing have shampoo? (Y/N)
y
Does your listing have a TV? (Y/N)
y
Does your listing have a washer? (Y/N)
y
Does your listing have a dryer? (Y/N)
y
Does your listing have a carbon monoxide detector? (Y/N)
y
Does your listing have a hair dryer? (Y/N)
y
Does your listing have a laptop friendly workspace? (Y/N)
y
Does your listing have an iron? (Y/N)
y
Does your listing have a fire extinguisher? (Y/N)
n
Does your listing have hot water? (Y/N)
y
Does your listing have free parking on premises? (Y/N)
y
Does your listing have self check-in? (Y/N)
n
Does your listing have a first aid kit? (Y/N)
n
Does your listing have previous reviews or a previous review score? (Y/N)
n

--------------------------------------------
Using Decision Tree Model: $ 332.0906236602581 per night
Using Random Forest Model: $ 217.5587482724098 per night
Using Linear Regression Model: $ 258.5891283890453 per night
Using Ridge Regression Model: $ 258.3132839998487 per night
Using Lasso Regression Model: $ 242.693122489037 per night
```

Initially, the goal of our project was to create a price prediction model that could be taken and adapted to be used for new listings in a city where Airbnb may not be popular or there is minimal data.   We did not test this directly, but believe our model would be able to perform under these circumstances considering it worked for our testing purposes of using a teammate’s house, which produced an estimated price per night that reflects that of an actual house listed on Airbnb.

Reference:

Evan Miller, “How Not To Sort By Average Rating”,
http://www.evanmiller.org/how-not-to-sort-by-average-rating.html