Project Report: STAT 303-3 Spring 2022

Abstract

In our final project, we wanted to identify amenities and host qualities in Airbnb listings that lead to higher reviews. To do this, we decided to build a classification model in which we classify whether a listing is above or below the median review of 4.84 stars. We experimented with different models such as bagging, random forest, and ensemble modeling and tested each model's accuracy and precision values. We decided to focus on accuracy and precision to ensure overall reliability because of how high most Airbnb ratings are and to make sure the listings that were classified to be above the median were actually high quality. Through our models, we discovered a hard voting ensemble produced the highest values of accuracy and precision. Through each model's feature importances, we also gained insight into what factors are significant when classifying a listing to receive a higher-than-median rating. Amenities such as air conditioning and barbecue grills are the least important factors in high reviews. On the other hand, the most important features are superhost status, how many reviews a listing has, and availability throughout the year. Thus, these are the aspects that Airbnb hosts should focus on when trying to get a higher rating.

1. Background / Motivation / Problem statement / Data sources

Airbnb is an American company that operates an online marketplace for lodging, primarily homestays for vacation rentals, and tourism activities. The platform is accessible via website and mobile app.

As a group, we share a love for travel. Thus, being users of the vacation rental app, we wanted to explore the logistics of Airbnb rentals and discover what goes into making a stay great by identifying amenities and host qualities that are most closely associated with high reviews.

The dataset we used contains detailed information on 5,778 Chicago Airbnb listings such as host qualities, ratings, location, amenities, and price. The amenities for each rental are listed in strings unique for each cell.

For more details and access to the data:

http://data.insideairbnb.com/united-states/il/chicago/2021-12-15/data/listings.csv.gz

2. Metric of Interest

To measure the success of our models, we decided to focus on test accuracy and precision. High accuracy allowed us to obtain results close to the true values, and precision allowed us to minimize false positives. We wanted a small number of false positives because we believed that booking a low-quality Airbnb (that was predicted to have a high rating) would be a much worse scenario than booking a high-quality Airbnb (that was predicted to have a low rating). Once we began optimizing the models, we focused on accuracy over precision. The reasoning behind this decision was that most review scores were quite high (4+ stars), so even a lower-rated Airbnb would still be nice and well-maintained. In other words, the risk associated with a false positive rating was rather small. Fortunately, we obtained high precision scores throughout our analysis, so we did not have to make a big trade-off in terms of accuracy over precision.

3. Stakeholders

Our stakeholders include Chicago Airbnb owners who would want to implement the standards that are correlated with high Airbnb listing success rates in the form of positive reviews. Using our recommendations, they will be able to gain insight into what home characteristics are valued and lead to higher reviews, allowing them to improve their listings and receive better reviews as a result.

Airbnb guests could also benefit from this information by comparing their individual amenity preferences to the broader determinants of review scores. This would help them interpret a listing's reviews, allowing them to book a stay that is more likely to please them. For instance, if a user prefers places that accommodate many people and maximum occupancy is a large factor in review scores, they can book highly rated listings with more confidence.

4. Approach

We sought to achieve our goal of predicting review scores and identifying their most important factors by classifying them into low and high score categories. While we initially planned to classify reviews as above or below three stars, our exploratory data analysis revealed a very right-skewed distribution of scores. Because almost all the scores were above four stars, we decided to classify them as above or below the median rating (4.84).

Before developing our base models, we cleaned the dataset by splitting the lists of strings in the *amenities* column into separate columns. We then found the most common amenities and created dummy variables for them. Additionally, we merged the less

common categories of predictors such as *neighborhood* and *property_type* into an "other" category to reduce overfitting in our models. We also removed irrelevant columns such as listing descriptions and titles. Through additional analysis, we found that the review sub-scores were very correlated with the overall review scores and dropped them from the data. Finally, we split the data into test and train sets with an 80:20 ratio.

We created logistic regression and decision tree base models using all the viable predictors after cleaning to compare a linear approach with a non-linear one. The decision tree performed significantly better in all metrics on test data than the logistic model. As a result, we focused our final model development on exploring non-linear models such as bagging, random forest, and boosting.

5. Developing the models

As mentioned previously, our base decision tree model outperformed our logistic model, which was one reason we focused on optimizing a decision tree instead of a linear method. Another reason we opted for a tree-based model was that decision trees can handle multicollinearity of predictors better than linear models. We have many predictors in our dataset, so we have a higher potential for multicollinearity. Since multicollinearity violates the independent observations assumption of linear regression, we realized a decision tree would perform better than a logistic regression model in this regard.

After choosing our metric of success and base model type, we then improved and iterated upon our base model. We developed five "base estimator" models and later used those estimator models to create ensemble models. The first base estimator model we looked at was bagging. Bagging bootstraps the training data, reducing the variance of the model and increasing the test accuracy. Tuning the bagging model gave us a test accuracy of 75.45%, which was an improvement from our base model test accuracy of 65.32%. In addition to bagging, we also developed a random forest model. The tuned random forest had an accuracy of 74.81%. Although this accuracy was higher than the base model, it was lower than the bagging model. This was not unexpected, as random forests can perform worse when the number of predictors is large. For data with many predictors, the fraction of useful predictors is likely to be smaller, lowering the chance that the best predictors are chosen at each split of the random forest. This resulted in a slightly lower accuracy score than bagging, but still an improvement from the original 65.32%.

We then tuned three boosting methods: AdaBoost, gradient boost, and XGBoost. Since our data contained many predictors, there was the potential for overfitting issues. We chose to use these boosting methods because they learn trees slowly, thereby reducing the risk of overfitting on training data. After tuning the three boosting models (AdaBoost, gradient boost, and XGBoost), we achieved test accuracies of 75.06%, 75.45%, and 73.77%, respectively. These results were similar to the bagging and random forest models, with bagging and gradient boosting ending with the highest scores.

Since no model drastically outperformed the others, we decided to use ensemble models to further improve the test accuracy. We looked at four ensemble models: hard voting, soft voting, stacking with a logistic meta model, and stacking with a tuned random forest meta model. In the end, the model with the highest accuracy was the hard voting ensemble model, which had an accuracy of 76.49%. This small increase did not surprise us due to the nature of our base estimator models. The base estimators all performed similarly, so in most cases they "voted" the same, which is the reason the accuracy didn't increase more substantially. However, in the rare cases where the base models disagreed, the ensemble model chose the majority vote, therefore still slightly increasing the accuracy score.

Overall, our model development proved to be a success. Our five base estimator models had fairly high test accuracies, all in the 73.77% - 75.45% range. The hard voting ensemble model (our final model) had a test accuracy of 76.49%, a significant improvement from the 65.32% accuracy of the base decision tree model. In addition to accuracy, we were also able to improve precision. Our base model had a precision of 63.98%, but our final model had a precision of 73.75%, so we were happy that we didn't have to sacrifice precision to increase accuracy. Apart from accuracy and precision, our analysis was a success in terms of inference as well. Developing and optimizing our model allowed us to find the most important factors contributing to high ratings, as we will discuss in the following section.

6. Interpreting the models

After experimenting with different models, we used the feature importance function to develop with recommendations to our stakeholders and achieve our project objectives. For the bagging and random forest models, we calculated the feature importances of the model, which was important in our analysis because we had many predictors for each model. For both the bagging and random forest model, the most important features were similar - the most important factors were superhost status, number of reviews, reviews per month, price, availability throughout the year, and acceptance rate. Based on this information, we were able to curate our recommendations detailed in the next section.

7. Conclusions and recommendations to the stakeholders

We found that hard voting was the best model in terms of accuracy and precision values. This method produced the best accuracy (76.5%) and precision (73.8%) in our final model, while the other methods had similar but still less successful values falling in the lower 70% range.

Through our analysis of the most important features, we conclude that a few factors are especially helpful in receiving a higher rating for an Airbnb rental: Being a superhost, a high number of reviews, and availability throughout the year, price, and the rate at which the rental gets reviews per month. Making a listing open throughout the year, taking on more guests, being responsive, and avoiding cancellations is our recommended strategy to increase the number of satisfied customers, become a superhost, and ultimately receive higher-than-median ratings. Additionally, Airbnb property owners should incentivize guests to leave a review at the end of their trip through their own means as number of reviews was an important predictor for the models. Something we found particularly interesting was how low the different amenities, such as microwaves, air-conditioning, and barbecue grills were on the list of important features. This can also further give Airbnb hosts insight into what aspects of their listing to spend money and focus on when aiming for better review.

8. Future Work

In the future, the research community could potentially develop models for other cities such as Los Angeles or New York. It would also be interesting to analyze the differences and similarities in the models that work the best as well as their important features to gain a better understanding of which factors are generally most important for Airbnb hosts, regardless of which city it is aimed towards. We could also use a random sampling method between datasets for Airbnbs in other cities, not just Chicago, to ensure the external validity of the results and thus improve the viability of the recommendations to stakeholders. Finally, we could build a classification model using only amenities as predictors to determine which amenities lead to the best ratings. This could narrow down the scope of the project to recommend amenities for Airbnb hosts to purchase that are most valued by guests.