Capstone 1
Susanna Vinoba

Final Report: Suggested List Pricing - Nashville Airbnb

Contents

1. Problem Statement

Airbnb is launching a new app to simplify the journey for potential hosts. This app provides instant estimates of potential nightly earnings based on property features, aiming to lower the entry barrier for new hosts.

Prospective hosts input property details, such as rooms and , and receive a quick estimated nightly rate. This eliminates pricing guesswork and helps hosts make informed decisions.

The app starts its trial in Nashville, Tennessee, a vibrant location ideal for assessing its impact on host sign-ups.If successful, Airbnb plans to expand the app to other regions, creating a more diverse host network and enriching traveler experiences.

The success of this endeavor will be measured by Airbnb's ability to attract and onboard 10% more hosts in the next year than what was projected. If this goal is achieved, it would signify that the app has effectively addressed the entry barriers and empowered a significant number of new hosts to join the platform.

2. The Data

The data was pulled from a freely available website called …. The data was last scraped …. The raw 'listings' dataset includes an intensive list of the features of the house, the listing profile, and the host profile; the full list of columns is included in the appendix. The data in total included 8127 entries/ rows.

The condensed features dataset used for both EDA and the machine learning algorithm include the following features/ columns: room type, minimum nights, accommodates, bedrooms, beds, bathroom count, bathroom type, neighborhood. These features were chosen as they would provide a good base estimator for predicting price.

Data cleaning:

Price: If the price (the target variable) was null or zero the row was dropped. One row was dropped due to a zero price column. The price column was changed from an object to a float.

Bathrooms: There was one column for bathrooms called bathrooms_text. This was split into two groups one for bathroom count and the other for bathroom type. There are three groups for bathroom type - private bath, bath and shared bath. Some of these included the plural, so the categories were then changed to the singular. Bathroom count was changed to a float. There were two rows missing bathroom data. Missing bathroom type values were changed to "N/A" and missing count was changed to -1.

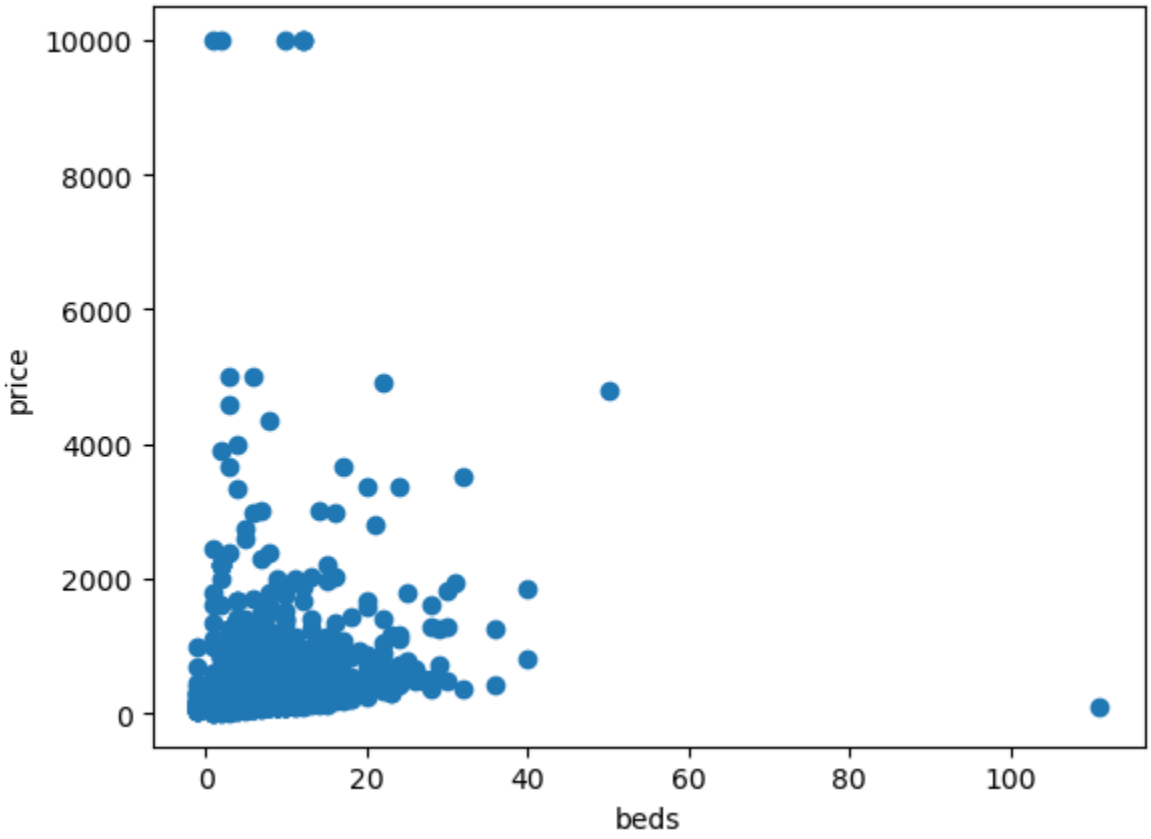
Beds & bedrooms: There were 46 missing bed values. There were 338 missing bedroom values. If both beds and bedrooms were null those values were replaced with -1 since nothing

could be assumed. This was the case for ten entries. If the bed value was 1 it was then assumed that the bedroom value was also 1.

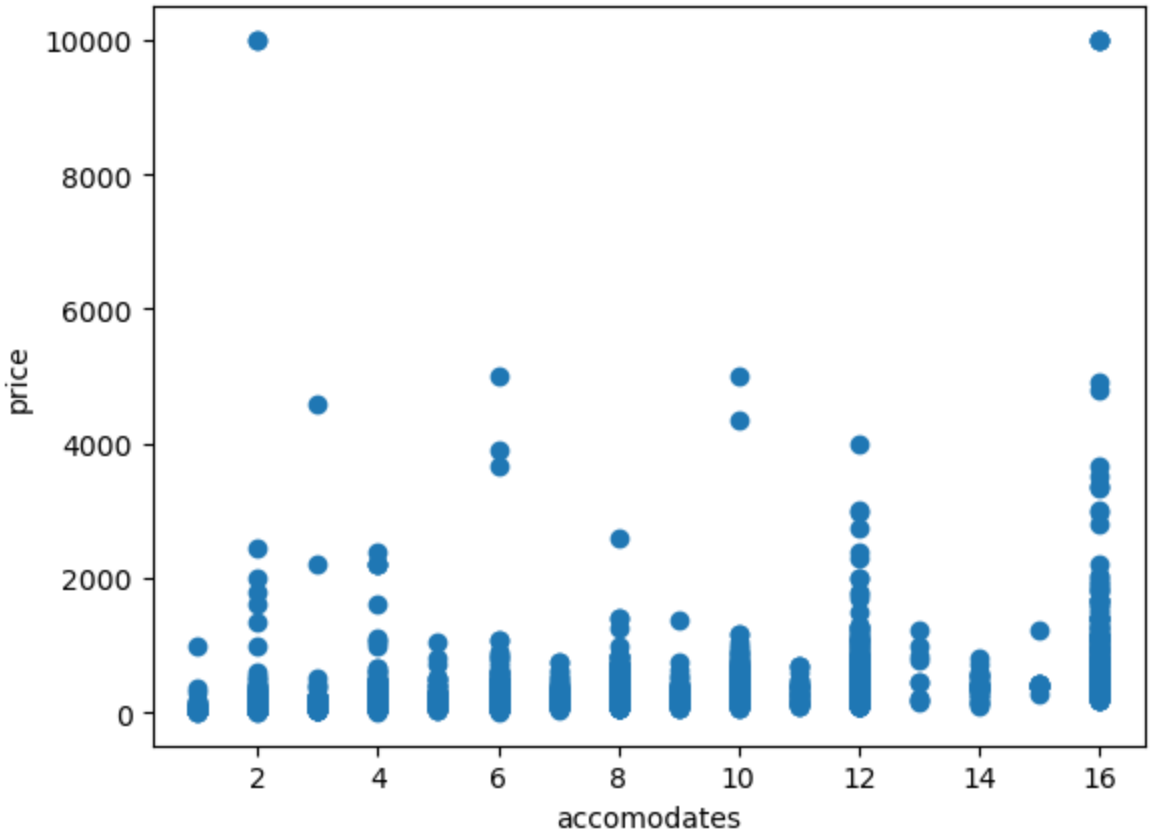
3. EDA and Statistical Analysis

Exploratory Data Analysis (EDA) was performed to gain insight into the dataset.

There was the most notable linear correlation between the number of beds and the price. This chart removed a price outlier of 80000 to get a better visual. One thing to mention is that much of the data is centered around a lower price and a lower number of beds. If all outliers are removed and it a zoomed loo was given a more notable trend could be estimated.



Many of the features including accommodates, bathroom count, neighborhood, bedrooms, have a more varied distribution. They are almost uniformly distributed wherever prices can be low or pretty high and it is not as dependent on the feature. For example, below is an image of the distribution for accommodates.

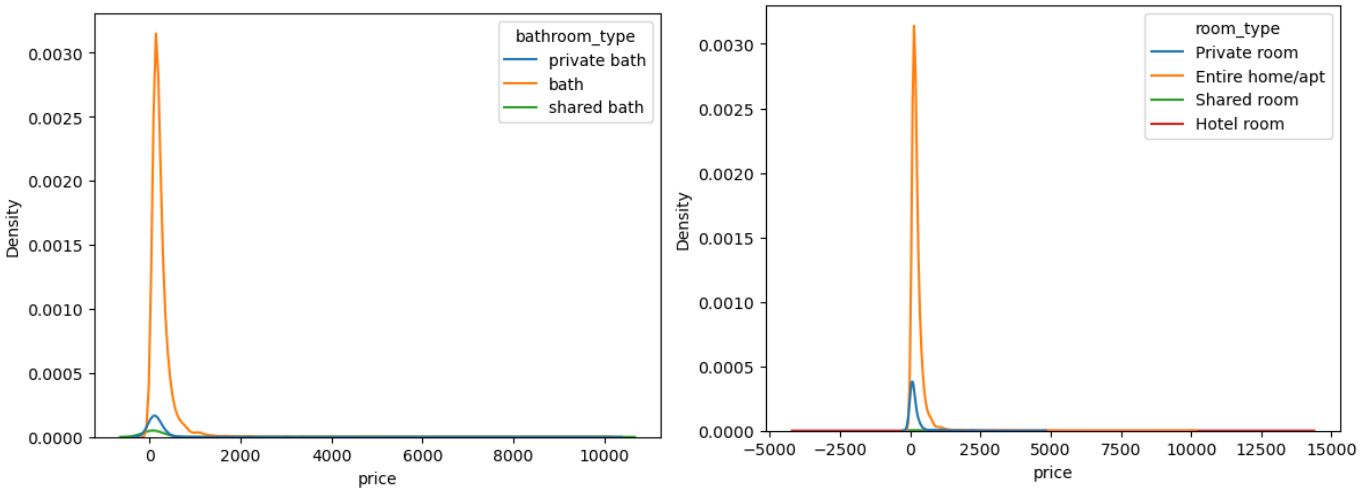


None of the features exhibit high negative or positive correlation with the price feature. There seems to be collinearity between bedrooms, beds, bathroom count, and accommodates. This is not good for the model as it will lead to computational complexity, overfitting, and unstable feature importance.

	price	minimum_nights	accommodates	bedrooms	beds	bathroom_count
price	1.000000	-0.009624	0.133185	0.157453	0.120949	0.165750
minimum_nights	-0.009624	1.000000	-0.132625	-0.061508	-0.092085	-0.063894
accommodates	0.133185	-0.132625	1.000000	0.830248	0.766874	0.784611
bedrooms	0.157453	-0.061508	0.830248	1.000000	0.771477	0.903611
beds	0.120949	-0.092085	0.766874	0.771477	1.000000	0.752387
bathroom_count	0.165750	-0.063894	0.784611	0.903611	0.752387	1.000000

For the categorical features, bathroom type and room type kdeplots were created. The price data has the most prices from 0 to around 1000. For bathroom type the category that is referenced the most is bath and for room type the entire home/ apt is the category with the most occurences.

This lowers the computational power of both of these variables in a tree based regressor because of the lack of opportunities for the model to encounter the other categories and predict price.



4. Application of Machine Learning

Model selection: Linear regression boosting was chosen as the model. By harnessing the power of ensembling, linear boosting models can effectively handle a mix of feature types, offering a robust solution for predicting prices accurately. These models excel at capturing complex relationships within categorical data like room type and neighborhood while also accommodating numerical features such as minimum nights and bedroom count.

Hyper parameter selection: The parameters n_estimators, learning_rate, max_depth, max_featres, and subsample were chosen. The selection of these parameters for our Gradient Boosting model was driven by the need to strike a balance between model complexity and generalization, ultimately aiming to enhance predictive performance.

- n_estimators: We explored a range of values to determine the optimal number of boosting stages. A higher number can capture more complex relationships but risks overfitting, while a lower number might lead to underfitting. We aimed to find the sweet spot.

- learning_rate: This parameter affects the step size during optimization. We experimented with various learning rates to control the convergence speed, with smaller rates often improving generalization at the expense of training time.

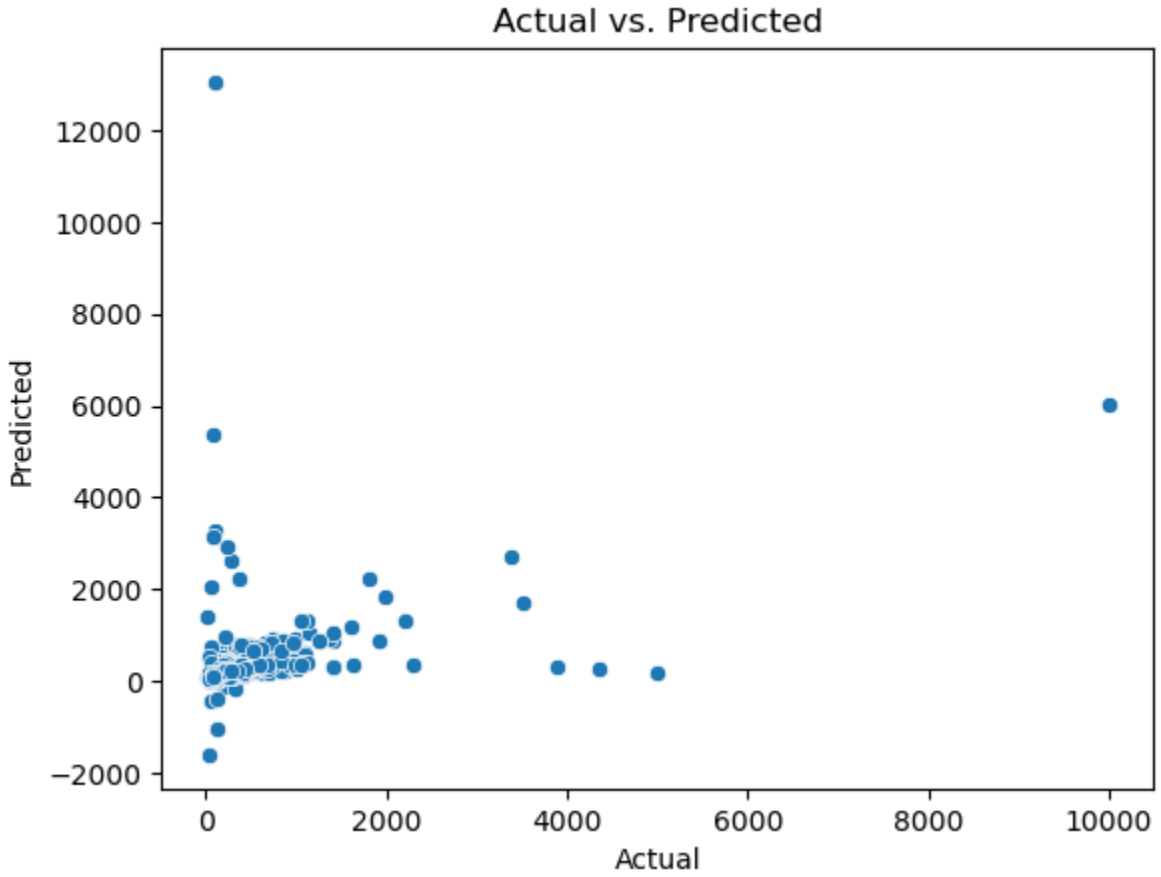
- max_depth: We considered different tree depths to manage the complexity of individual base learners. Deeper trees can capture intricate patterns but may lead to overfitting. Our goal was to find a depth that balances bias and variance.

- max_features: Varying the number of features considered at each split can mitigate overfitting. By trying different values, we aimed to identify the most informative subset of features.

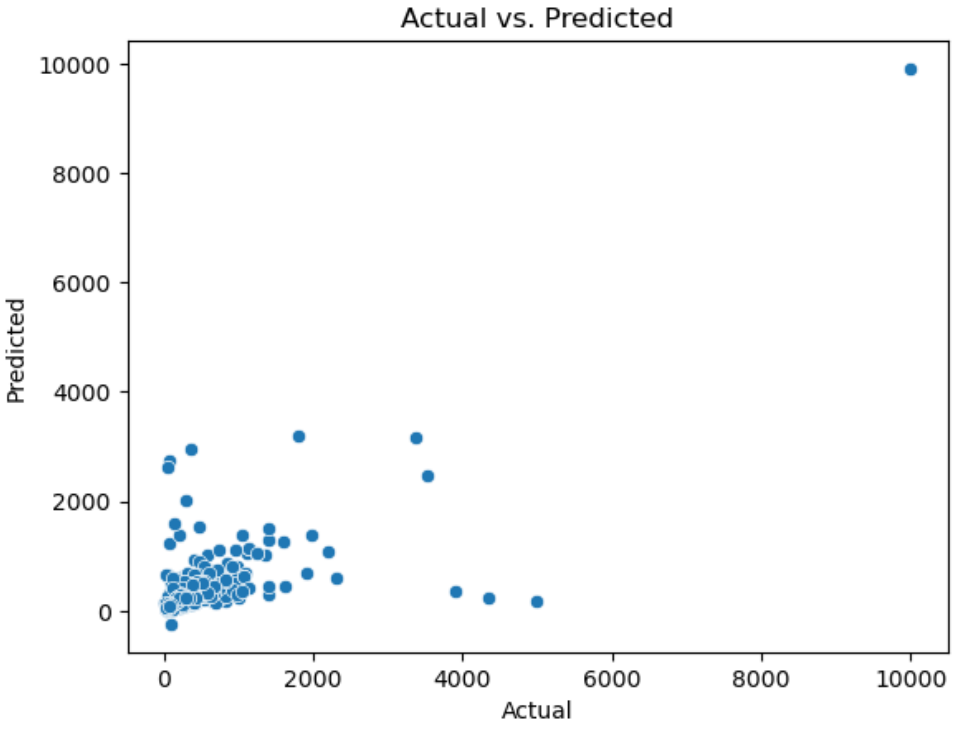
- subsample: Subsampling the training data can introduce randomness, potentially reducing overfitting. Our exploration of this parameter sought to strike the right balance between variance reduction and model performance.

Results:
Two models were run: (1) no hyper parameter tuning (2) hyper parameter tuning

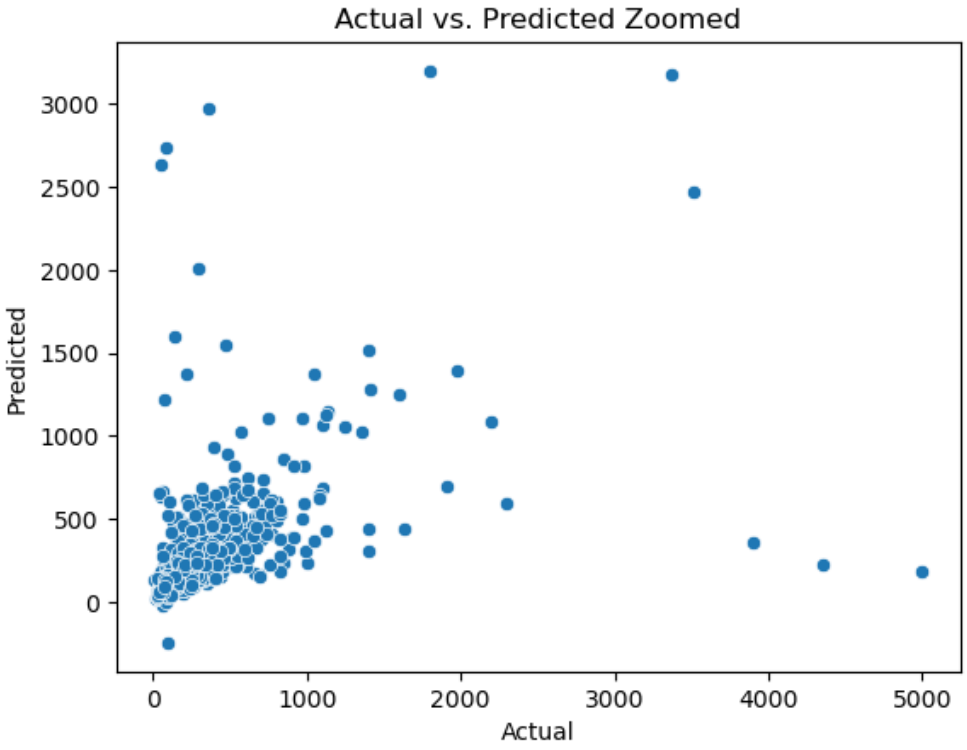
No hyper parameters:



Hyper parameter tuning:



Same model zoomed in… without the outlier:



Feature importance: The three features that have the most impact on the model are minimum nights (0.39), accommodates (0.28), and beds (0.21). The rest of the features have less than 0.05 weight on the model.

5. Conclusions and Future Work

Conclusions: A model was made to estimate how much a new AirBnb host in Nashville should charge per night just based off of the features of their house.

Future work: There is plenty of future work that can be done with this data. AirBnb can dig into host credentials to see if having other listings, having a good description or being a super host makes a difference on list price. Airbnb can also expand from just offering the app in Nashville to offering it in several cities.

6. Appendix

id,listing_url,scrape_id,last_scraped,source,name,description,neighborhood_overview,picture_url,host_id,host_url,host_name,host_since,host_location,host_about,host_response_time,host_response_rate,host_acceptance_rate,host_is_superhost,host_thumbnail_url,host_picture_url,host_neighbourhood,host_listings_count,host_total_listings_count,host_verifications,host_has_profile_pic,host_identity_verified,neighbourhood,neighbourhood_cleansed,neighbourhood_group_cleansed,latitude,longitude,property_type,room_type,accommodates,bathrooms,bathrooms_text,bedrooms,beds,amenities,price,minimum_nights,maximum_nights,minimum_minimum_nights,maximum_minimum_nights,minimum_maximum_nights,maximum_maximum_nights,minimum_nights_avg_ntm,maximum_nights_avg_ntm,calendar_updated,has_availability,availability_30,availability_60,availability_90,availability_365,calendar_last_scraped,number_of_reviews,number_of_reviews_ltm,number_of_reviews_l30d,first_review,last_review,review_scores_rating,review_scores_accuracy,review_scores_cleanliness,review_scores_checkin,review_scores_communication,review_scores_location,review_scores_value,license,instant_bookable,calculated_host_listings_count,calculated_host_listings_count_entire_homes,calculated_host_listings_count_private_rooms,calculated_host_listings_count_shared_rooms,reviews_per_month