Capstone 1 – Final Report

Developing a Machine Learning Model to Predict Airbnb Listing Prices

Author: Zach Palamara

Project Aim and Background

- Airbnb Founded in 2008
- It is a growing popular alternative to traditional lodging like hotels
- Currently there are no free resources to optimize listing prices for hosts
- My goal was to develop a model for hosts in Austin, TX to use for pricing their Airbnbs

Data Wrangling

- Data was sourced from Inside Airbnb, a free and independent source that scrapes publically available data from Airbnb
- Dataset contained 11,151 listings with 78 unique features.
- I read this source .csv file directly into a Pandas DataFrame to begin the data cleaning and preprocessing phase

Data Cleaning

- Wrote a function that accepts a column name as an argument and returns the following:
  - *percent of data missing, the name and count of each unique value, and the number of unique values for that feature*
- Missing Data
  - Dropped features with >90% missing data
  - Replaced “NaN” with “Unknown” for categorical features
  - Replaced some numerical features with median
- Binary Features
  - Replaced True “t” or False “f”strings with 1’s and 0’s

```
col_info('host_response_time')
```

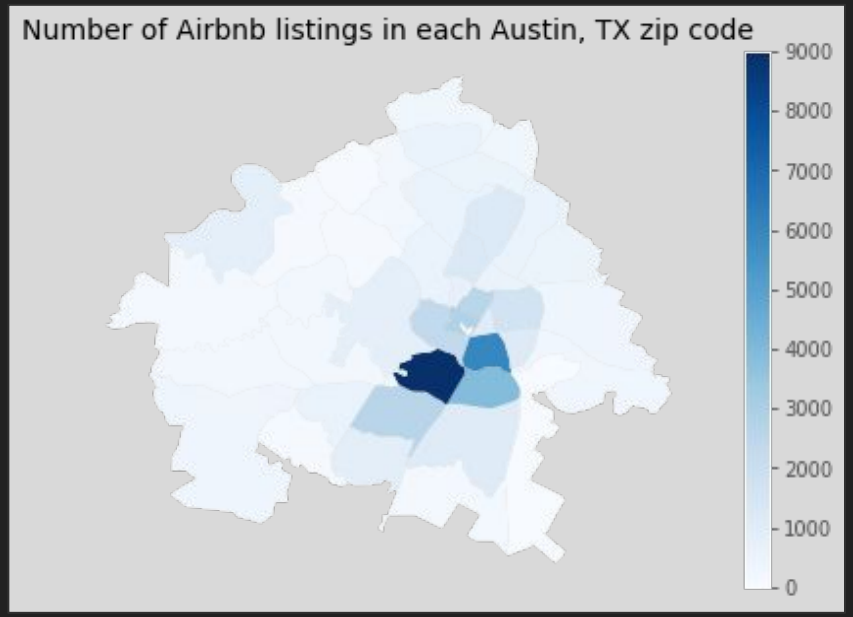
```
Column Name
host_response_time

Data Type: object
Number of Unique Values: 4
Missing Data: 31.0%

Value Counts
within an hour        6331
NaN                   3598
within a few hours     937
within a day           540
a few days or more     114
Name: host_response_time, dtype: int64
```

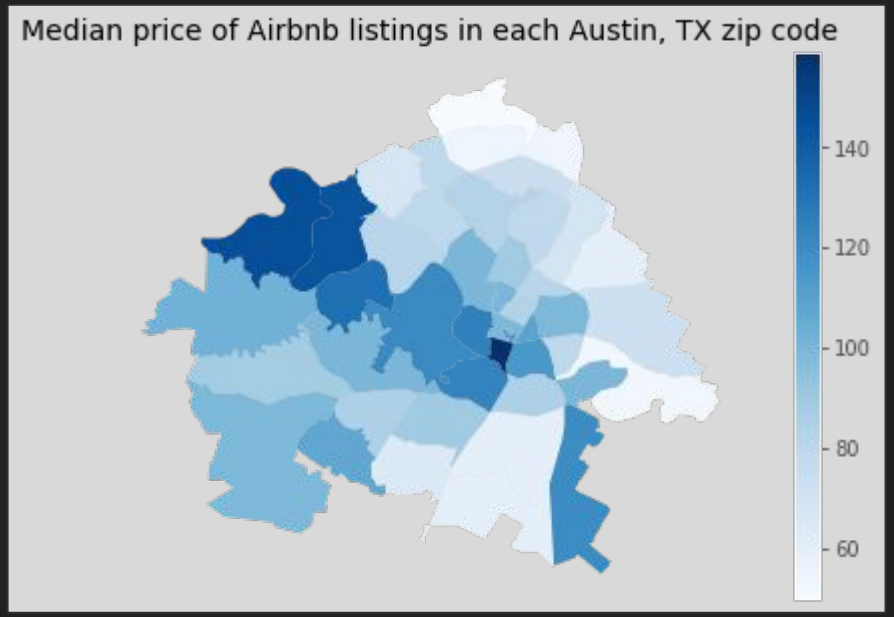
Data Cleaning

- Date-Time Features
  - Converted to DateTime objects
- Categorical Features
  - Binned the 31 unique listing types into 4 new bins: *“House, Apartment/Condo, Other and Hotel.”*
- Numerical Features
  - Mostly kept as continuous numbers, however some features were binned

Exploratory Data Analysis

Which areas/neighborhoods in Austin, TX are the most expensive and have the most listings?



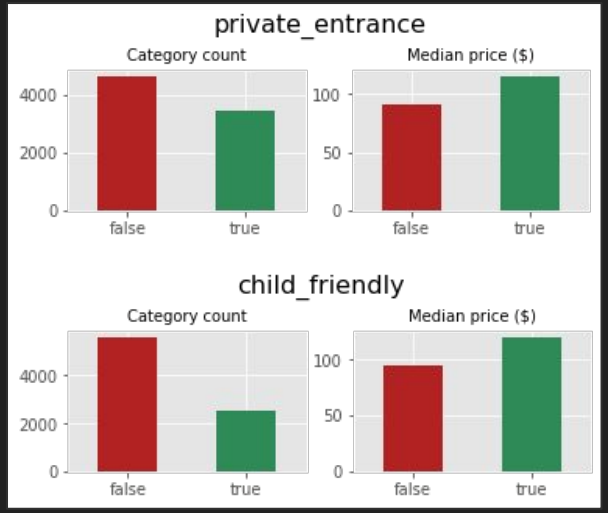


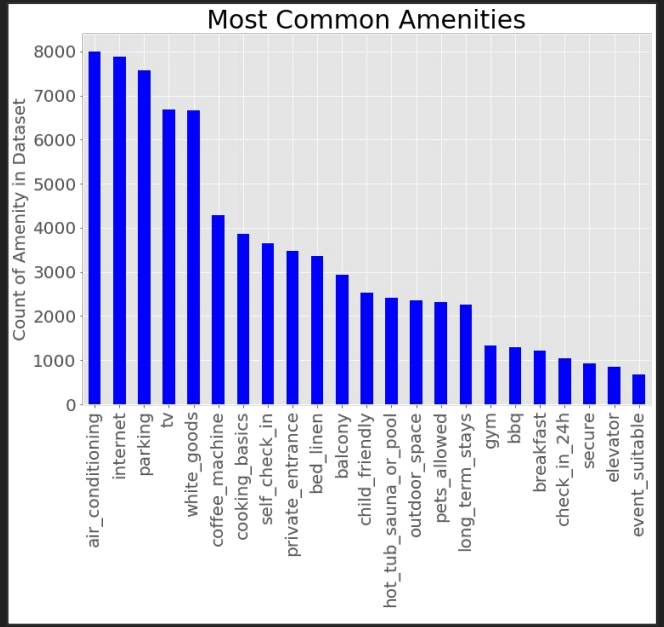
Exploratory Data Analysis

How do Airbnb prices correlate with the number of people a listing accommodates?

Exploratory Data Analysis

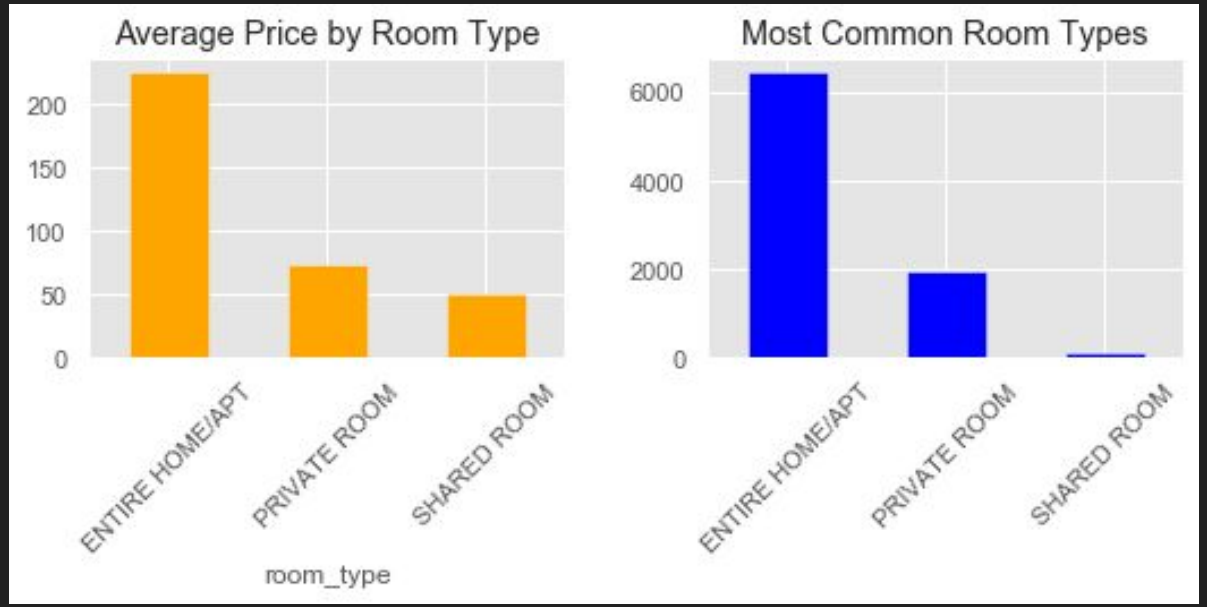
What are the most common amenities, and which amenities are likely to increase the listing price?





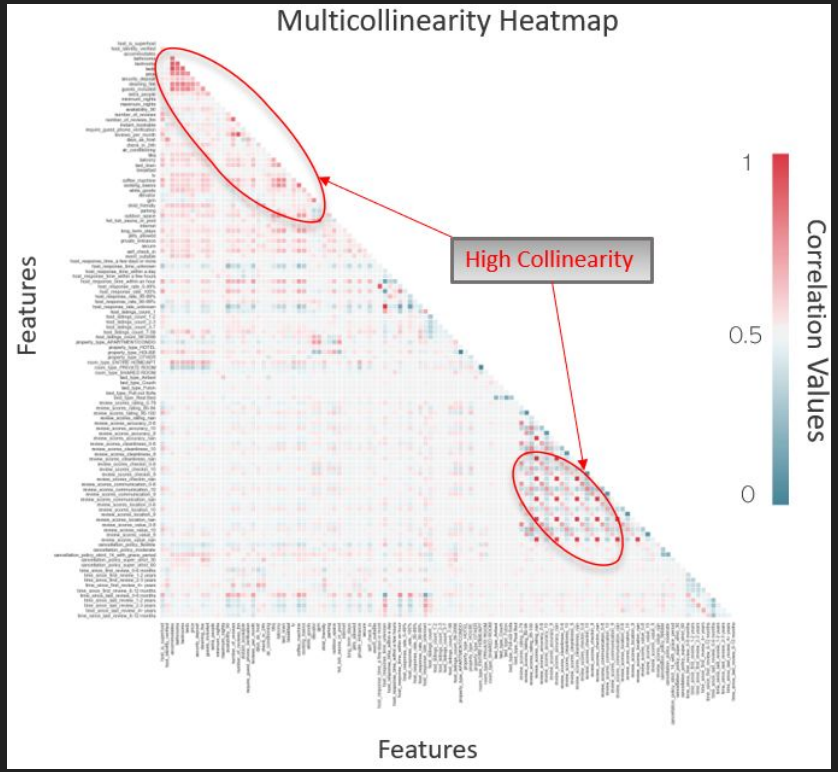
Exploratory Data Analysis

Property Type/Room Type



Model Preparation

- Normalization and Standardization
  - Log transformed several features to normalize their distributions
  - Standardized all numerical features using SKLearn Standard Scaler
- Multicollinearity
  - Assessed for highly correlated features and removed some for dimensionality reduction



Model Preparation

One-Hot Encoding

“One-Hot Encoding” - allows categorical features to be expressed as integers.

- For example, instead of having a single feature such as “room_type” that contains string values such as “private_room” or “entire_home/apt”, One-Hot Encoding transforms each one of those unique values into separate columns containing either a “1” or a “0”.

id	host_is_superhost	host_identity_verified	accommodates
2265	1.0	1.0	4
5245	1.0	1.0	2
5456	1.0	1.0	3
5769	1.0	1.0	2
6413	1.0	0.0	2

Machine Learning

Model Performance

Metric	Model	
	XGBoost Regression	SKLearn Gradient Boosted Regression
Train MSE	19.61%	19.28%
Test MSE	21.11%	21.64%
Train R^2	0.7228	0.7274
Test R^2	0.6943	0.6867
Computational Time (sec)	0.9	11.8

- Initial model was constructed using a gradient boosted regression classifier from the SKLearn ensemble.
- I also tested a gradient boosted regression model using the XGBoost framework.
- Similar results however, the SKLearn ensemble is significantly more computationally expensive

Machine Learning

Feature Importance

Feature Importance	Model	
	XGBoost Regression	SKLearn Gradient Boosted Regression
1	room_type_entire home/apt	accomodates
2	accomodates	property_type_other
3	bathrooms	security_deposit
4	cleaning_fee	bathrooms
5	air_conditioning	require_guest_verification
6	parking	extra_people
7	cancellation_policy_strict_14_ with_grace_period	reviews_per_month
8	host_listings_count_58-2056	child_friendly
9	security_deposit	cleaning_fee
10	property_type_other	white_goods

- Most feature across the board appears to be the number of people a listing and the number of bathrooms a listing accomodates.
- Important ancillary features that could help boost the price of the listing, such as having parking, air conditioning, the entire home to yourself, being child friendly or having linens supplied.

Conclusion

- Both models had nearly the same results in regards to accuracy and MSE.
- The SKLearn model had slightly better training results, while the XGBoost model had better results on the test set.
- This indicates that the SKLearn model might have been slightly overfitting the data.
- The XGBoost model was more than 13x faster in computational speed, which could have huge implications as the data set grows.

Considerations for Future Development

- One thing that I think will help the accuracy of this model is to use the geographic data to engineer a feature that represents the listing's proximity to popular spots in the city.

- Additionally, given more time I would have liked to conduct some Natural Language Processing (NLP) on listing reviews to engineer a feature that includes a user's sentiment about their stay.

- Lastly, pictures of the listing itself are something that my model is agnostic to. From experience, I think high quality photos are something that greatly influences a user in selecting an Airbnb for their stay.