Springboard - DSC

Capstone Project 2 Final Report

Predicting market rent prices of rental properties from Craigslist in the Bay Area

Melissa Han

November 2022

Table of Contents

List of Figures

List of Tables

1 Introduction - the problem statement

Predicting rent prices from market listings is essential to making the most informed decision about people's biggest living expense, because it allows us to make the biggest savings on housing. Properties vary so much it is hard to determine how features influence rental price. It is difficult for renters and landlord's to identify properties with similar features in their vicinity for price reference.

This capstone project aims to

- Identify the magnitude of features that affect the market rent price of a property. It would be interesting to see how much of a premium individual features such as being pet friendly and having in unit laundry add to the rent price.

- Develop supervised machine learning models that use regression to predict the market rent price of a property based on relevant features common to a craigslist listing. The metros of interest are San Francisco, Peninsula, East Bay in the Bay Area.

2 About the data

2.1 Craigslist Data

Craigslist was chosen as the data source because it is one of the most common places to find a place to rent, especially in the Bay Area. There is no craigslist API, so the listing information was scraped from the Craigslist website every 3 days over the period Aug - Oct 2022. Beautiful Soup and Playwright were used to scrape the search result index and listing pages, and html was parsed to a dataframe. Because properties often stay on the market for several weeks, there are many duplicate rows as a result of data collection. It is also common for property management companies to post several craigslist ads for the same unit.

The features at the beginning of the data wrangling stage are:

- Title of the craigslist listing
- Address
- Neighborhood and city of the listing
- Rent price
- Number of bedrooms
- Number of bathrooms
- Area measured in Square foot
- The body of the craigslist post which contains text describing the property

Rent price prediction training and testing data follows the following assumptions:

- The property for rent is an entire property - that means not renting by room
- The property is located in the Bay Area metros - San Francisco, Peninsula, East Bay
- The rental price is quoted on a monthly basis

To make predictions, the following features must not be missing: Square footage, number of bedrooms, number of bathrooms, address.

Note: while rent controlled properties are included in the dataset, and is a feature, if something is rent controlled, the price of the unit will still be market rent because the entire place is being rented, which resets the rent controlled clock (if anything, landlord’s will charge slightly more than market rent because they know the rent will be locked in during the tenancy)

2.2 Bay Area Map

Figure 2.1 Map of Bay Area metros within the scope of this project

2.3 Walk score, Transit score, Bike Score data

The dataset is also supplemented by walk score, transit score, and bike score information from the website https://www.walkscore.com/ by looking up the address of the listing and web scraping. Scores range from 0 to 100, where 100 is the best.

Their scoring system is below:

Walk Score	Transit Score	Bike Score

90–100	**Walker's Paradise** Daily errands do not require a car
70–89	**Very Walkable** Most errands can be accomplished on foot
50–69	**Somewhat Walkable** Some errands can be accomplished on foot
25–49	**Car-Dependent** Most errands require a car
0–24	**Car-Dependent** Almost all errands require a car

Walk Score	Transit Score	Bike Score

90–100	**Rider's Paradise** World-class public transportation
70–89	**Excellent Transit** Transit is convenient for most trips
50–69	**Good Transit** Many nearby public transportation options
25–49	**Some Transit** A few nearby public transportation options
0–24	**Minimal Transit** It is possible to get on a bus

Walk Score	Transit Score	Bike Score

90–100	**Biker's Paradise** Daily errands can be accomplished on a bike
70–89	**Very Bikeable** Biking is convenient for most trips
50–69	**Bikeable** Some bike infrastructure
0–49	**Somewhat Bikeable** Minimal bike infrastructure

Knowledge of the walk score, transit score and bike scores are an important feature to the model in lieu of the address, especially for an urban area such as the Bay Area. If a renter was choosing between two similar properties in different neighborhoods, these scores could inform the level of car dependency without familiarity of the neighborhood.

These scores are currently not a required input on craigslist, although some places with high walk scores do include the information. The address to score conversion is done on https://www.walkscore.com/methodology.shtml

3 Data Cleaning and Wrangling

The purpose of data cleaning and wrangling are:

- To ensure all features are of the correct data type
- To identify missing data and what to do with them
- To identify and create useful features from the dataset
- To prepare the dataset for EDA and modeling

3.1 Removing duplicate listings

This was done considering several combinations of the following: the listing address, the listing city, the number of bedrooms, the number of bathrooms, the name of the listing's first image. At the beginning of data wrangling, the data frame has 1637948 rows where each row is a listing on craigslist. There are 12 features, with more created during the preprocessing stage of the project. After cleaning and filtering for duplicates, the data frame has 18338 rows.

3.2 Data Type Correction and Assumptions

Because the listing information is provided by Craigslist users, there were many discrepancies such as missing information and incorrect labels. Data cleaning was done to ensure the data was as accurate as possible and involved cross referencing what was provided both as drop down options (mandatory inputs in Craigslist) vs. free text in the title and body of the listings.

The following steps were taken:

- Remove rows with missing price data, and cap the minimum rental price to be $500 and max to be $15000/month. Listings with prices above this value will be excluded from further analysis.
- Use regex to parse important information from the title and body of the listing such as number of bedrooms, number of bathrooms, whether the place is shared or not (if it is shared, the listing is removed from the dataset), square footage of the property
- Cross check the listed metro, city, neighborhood, address were consistent
- Remove listings that are shared apartments/roommate wanted ads that were misclassified
- Remove listings that fell below a chosen square footage threshold of 150 (anything lower than this would be too small to be an entire property) or above 4000 square feet (likely an input error)
- Convert data type of number of bedrooms and bathrooms to numeric
- Extract relevant keywords from the body of the craigslist ad to create new binary features. Examples of this include: does the property have amenities such as a pool or gym, have a backyard, have a view, is remodeled, has premium finishes such as granite, marble or a fireplace.

3.3 Add Walk Score, Transit Score, Bike Score data

Lookup the walk, transit and bike scores by the address of the listing from the walkscore.com website. A SQL database of addresses corresponding to scores is created for ease of looking up past addresses before web scraping.

3.4 Missing values imputation

Missing values are identified and will be imputed in the preprocessing step, once the dataset has been split into training and testing sets. This is to avoid data leakage.

Table 3.1 Summary of missing values in the dataset

Feature	Count	%
listing_sqft	5267	32.9
transit_score	2805	17.5
bike_score	2049	12.8
walk_score	471	2.9
listing_address	396	2.5

Square foot values are missing the most, and will be imputed using the median square footage for a given bedroom/bathroom size. Missing walk score, transit score and bike score will be imputed using the median scores from each neighborhood. The missing addresses are not needed as they are only used to calculate the walk, transit and bike scores.

After data wrangling, the data has a relatively good distribution of apartments vs. houses, no imbalance classes, and the price feature has a long right tail as there are fewer more expensive properties.

4 Data Visualization and Analysis

4.1 Rental Price

The table and graph shows the distribution of listing prices. The majority are between the $2000-$4000/month range. There is a long right tail, which is to be expected for expensive rentals.

Table 4.1 Summary of listing price distribution

Mean	$3337.2
Standard Deviation	$1519.0
Minimum	$690.0
25% percentile	$2300.0
50% percentile	$2995.0
75% percentile	$3950.0
Maximum	$14995.0

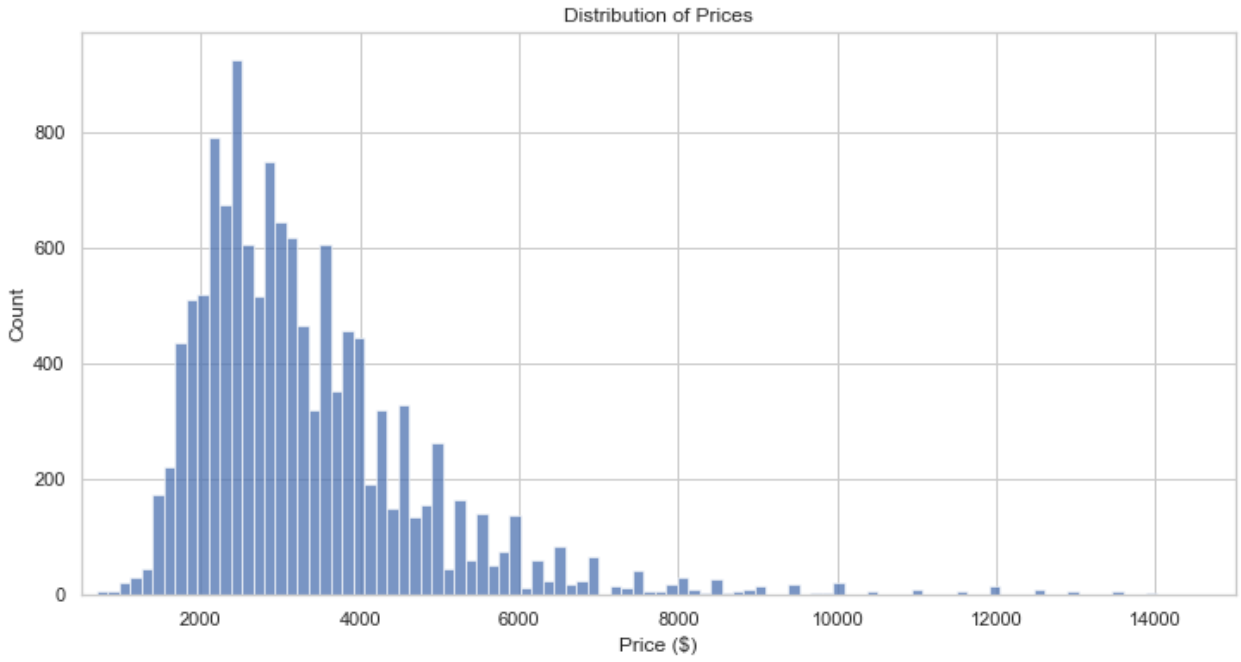

Figure 4.1 Bar Graph showing the distribution of rental prices in the dataset

There is greater variability in rental prices for the larger and notoriously more expensive cities: San Francisco, Berkeley, Oakland, Menlo Park, Palo Alto, Burlingame, San Mateo

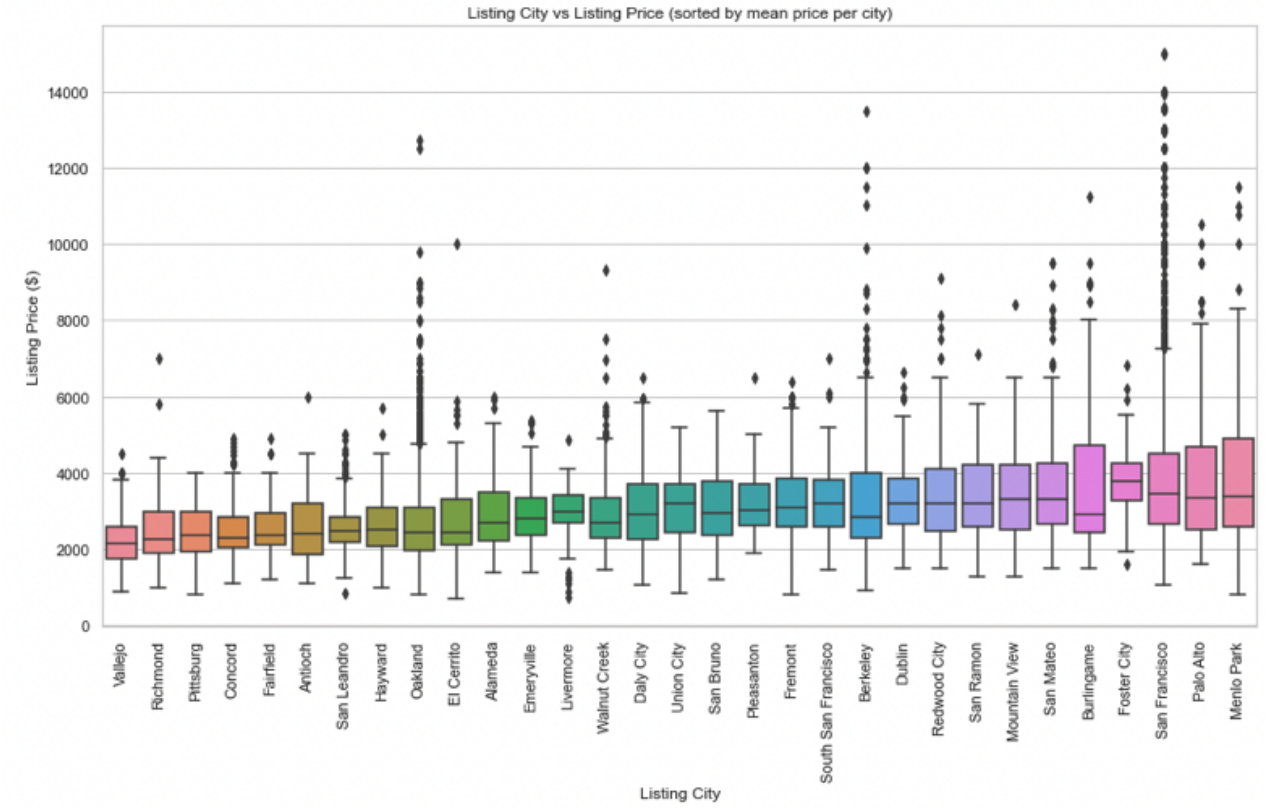

Figure 4.2 Box Plot showing the distribution of rental prices across listing cities

4.2 Correlation between features

The numeric features that seem most correlated with price are:

- Listing_sqft
- Number of bedrooms
- Number of bathrooms
- Laundry in unit
- Parking in an attached garage
- A house, apartment or condo

Other features that are noteworthy:

- Walk score, transit score and bike score are highly correlated
- The number of bedrooms and bathrooms is correlated to square footage
- Allowing cats and dogs are correlated but are negatively correlated with pets allowed

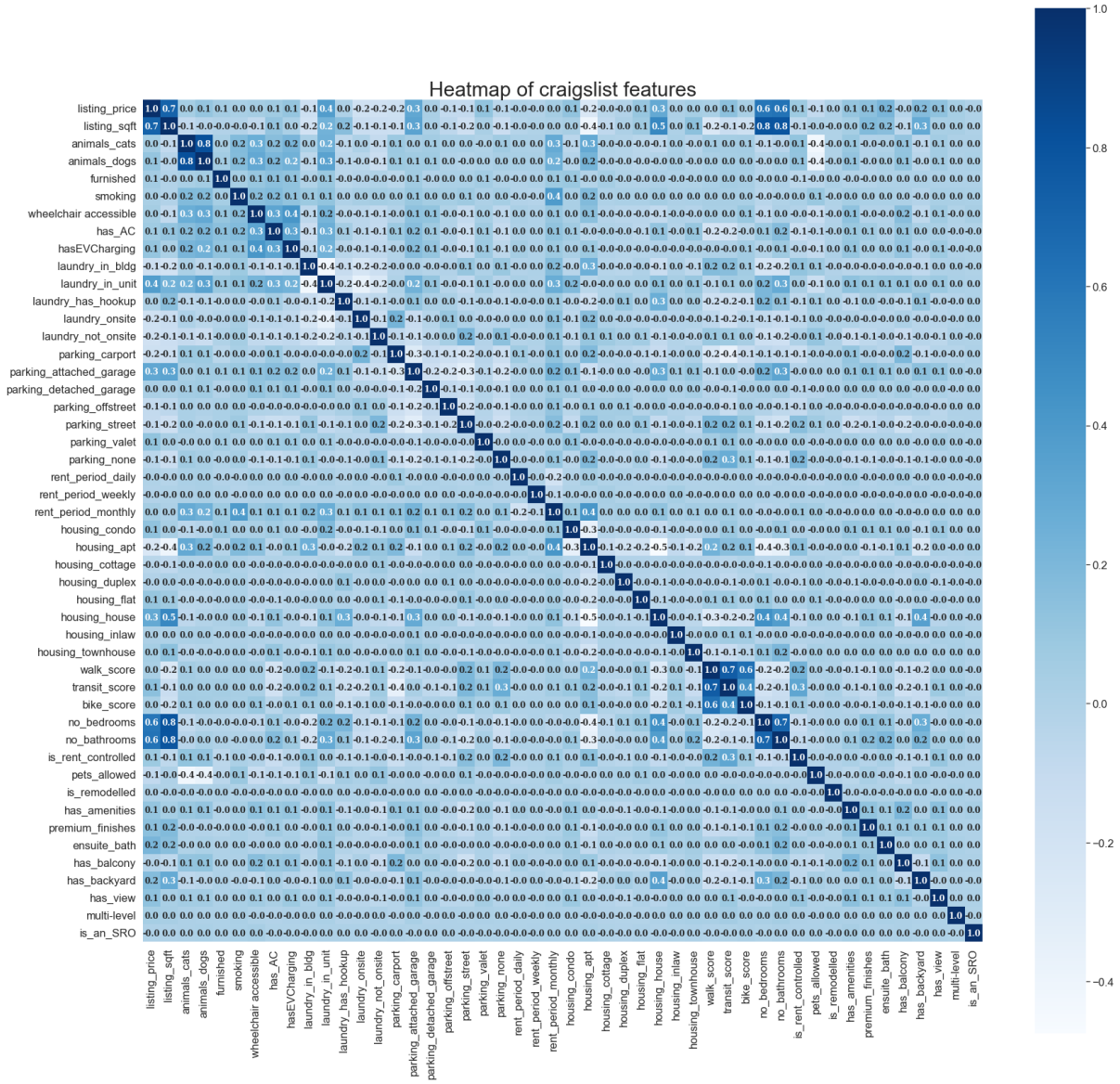

Figure 4.3 Pearson Correlation between listing price and features

4.1.2 Relationship between price and number of bedrooms/bathrooms

We expect the listing price to increase with the number of bedrooms/bathrooms. Shared bathroom rentals i.e. SROs seem to be an obvious low price category with few outliers.

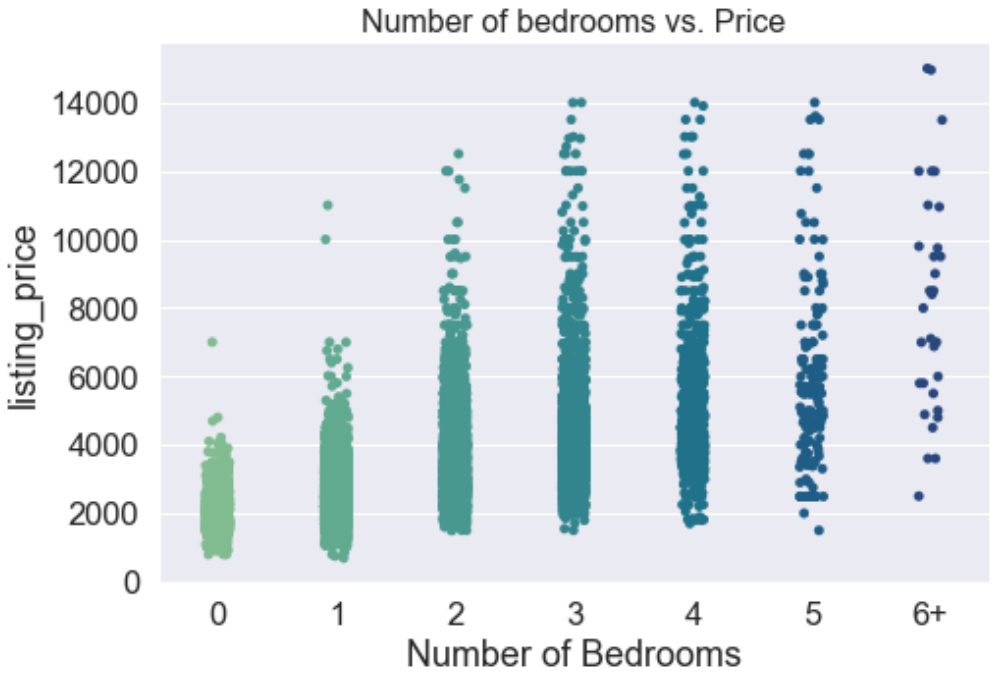

Figure 4.4 Scatterplot showing the distribution of prices for the number of bedrooms

Figure 4.5 Scatterplot showing the distribution of prices for the number of bathrooms

4.1.3 Square footage

Square footage has the highest correlation (0.7) to listing price out of all the features.

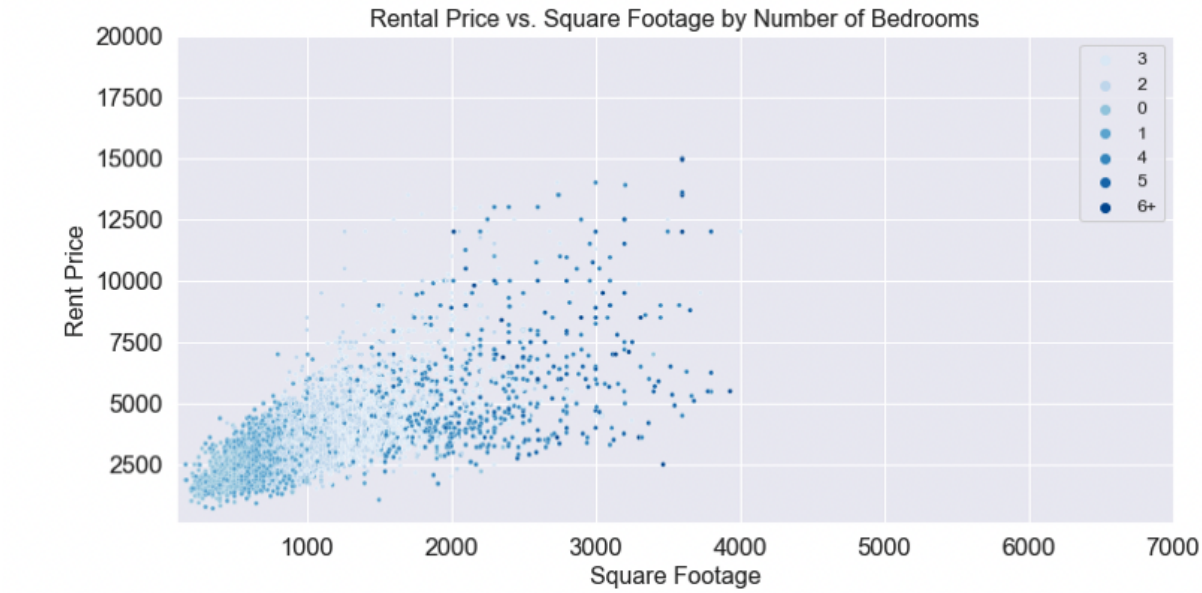

Figure 4.6 Scatterplot showing the relationship between price and square footage of properties, grouped by number of bedrooms.

It is clear there are many outliers in the square footage feature.

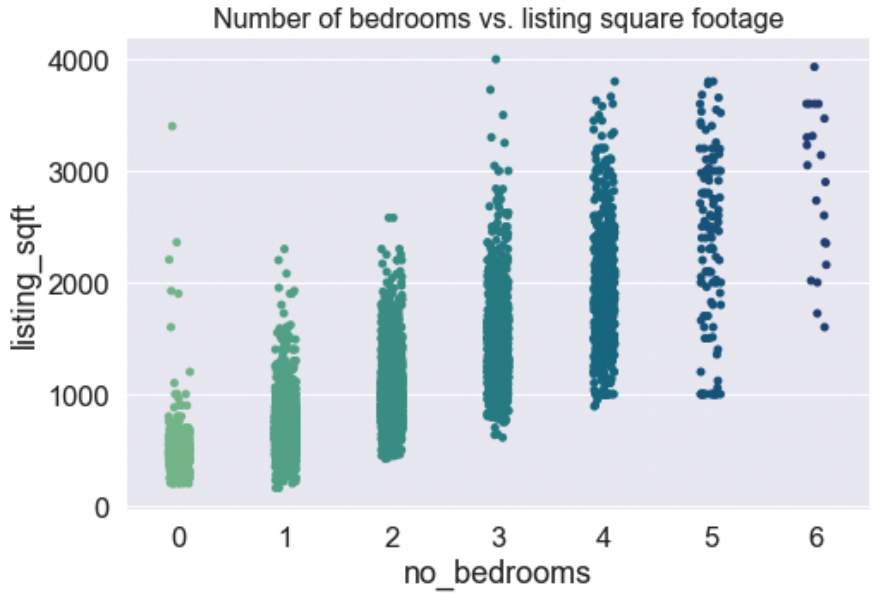

Figure 4.7 Scatterplot showing the relationship between square footage and the number of bedrooms

It is clear there are some outliers that are most likely data entered incorrectly, i.e. 20000 sqft instead of 2000 sqft. In the preprocessing step, outliers that are more than 3 standard deviations away from the median for 0 bedroom listings will be removed.

4.1.4 Relationship between price and neighborhood, a categorical feature - A San Francisco Example

Table 4.2 Summary of average price per square foot for the San Francisco Metro

Mean	$4.07
Standard Deviation	$0.56
Minimum	$3.1
25% percentile	$3.6
50% percentile	$4.1
75% percentile	$4.6
Maximum	$5.0

Note: Golden Gate Park is set to a price of $0 but is excluded from the summary statistics.

Plotting a heat map of San Francisco's average price per square foot for each of the 37 neighborhoods shows a large price difference between the neighborhoods in the top right corner of San Francisco near the financial district, and the cheaper neighborhoods in the west and south directions. Because there are too many neighborhoods to one hot encode (there are 81 in total), and what matters is only the relative difference between the neighborhoods, feature engineering will need to transform this feature into a numeric one without introducing too much dimensionality to the model.

In the preprocessing step, the categorical variable neighborhood will be replaced with a numeric feature average price per square foot.

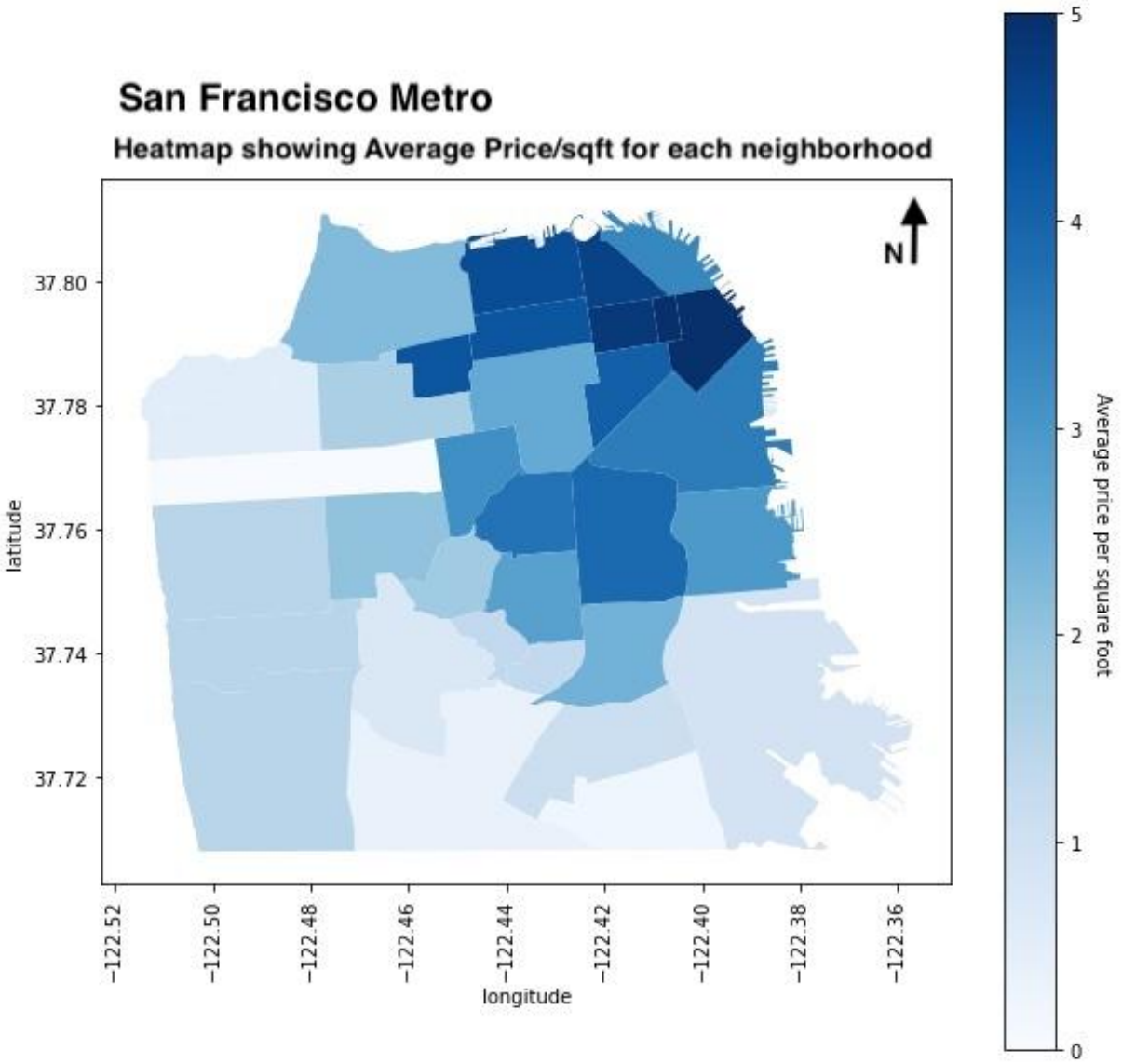

Figure 4.8 Choropleth graph showing the relational price difference for neighborhoods in San Francisco

4.2 Listing neighborhood, city and metro

There are 3 metros, 31 cities, and 81 neighborhoods. The East Bay is the largest metro, and the peninsula is the smallest.

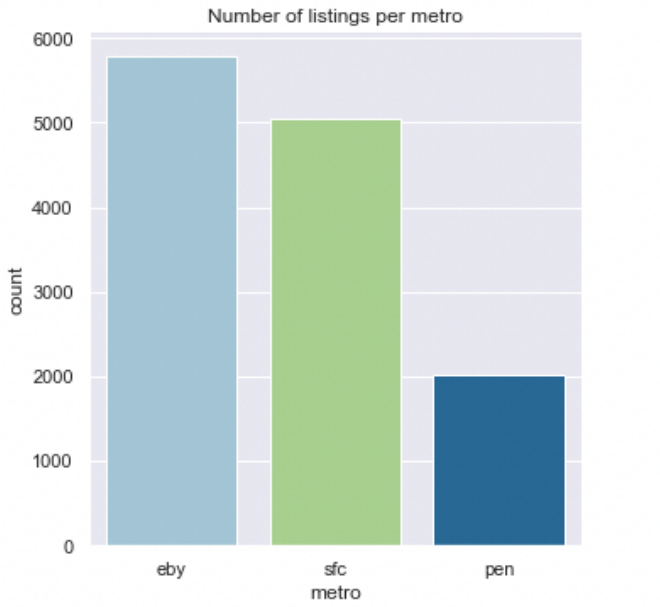

Figure 4.9 Bar Graph showing the number of listings in each metro

In theory, a city will contain multiple neighborhoods, but some neighborhoods are as big as a city, as shown below. The SF neighborhood SOMA/South beach is almost as big as the only neighborhood in the city of Berkeley.

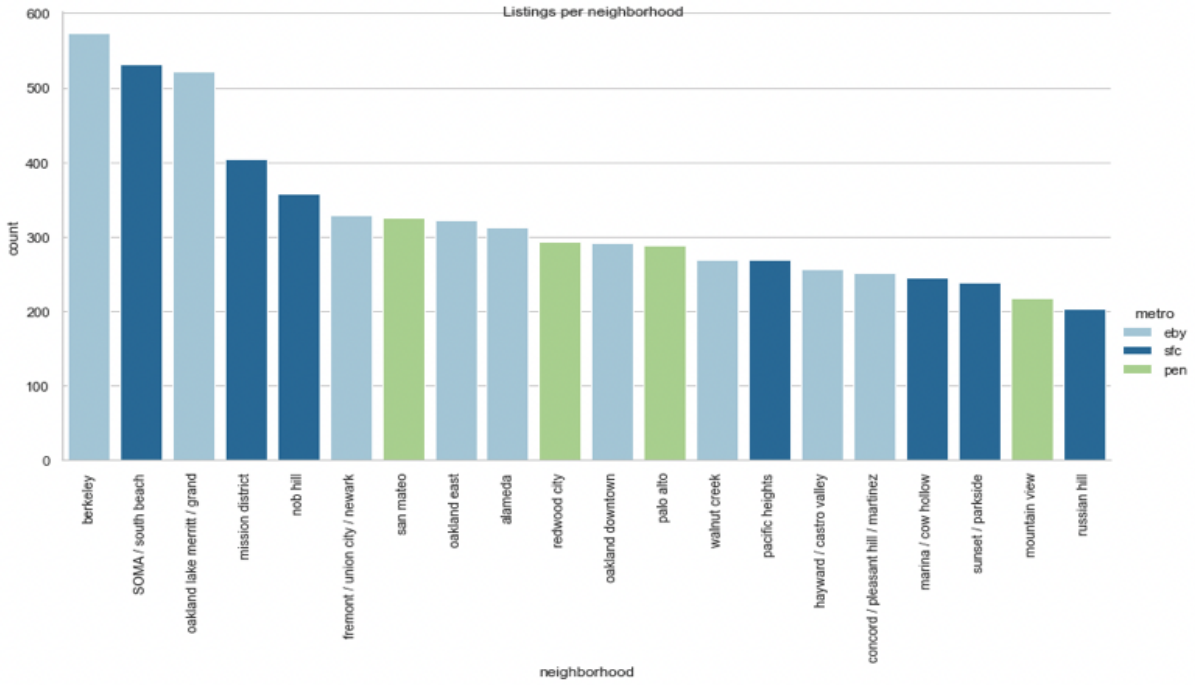

Figure 4.10 Bar graph of listings in each neighborhood across the 3 metros

4.3 Interesting distributions of listing features

4.3.1 What are the most common bedroom/bathroom combinations?

These bar graphs show the most common number of bedrooms are 1 and 2 bedroom places, with 1 bathroom.

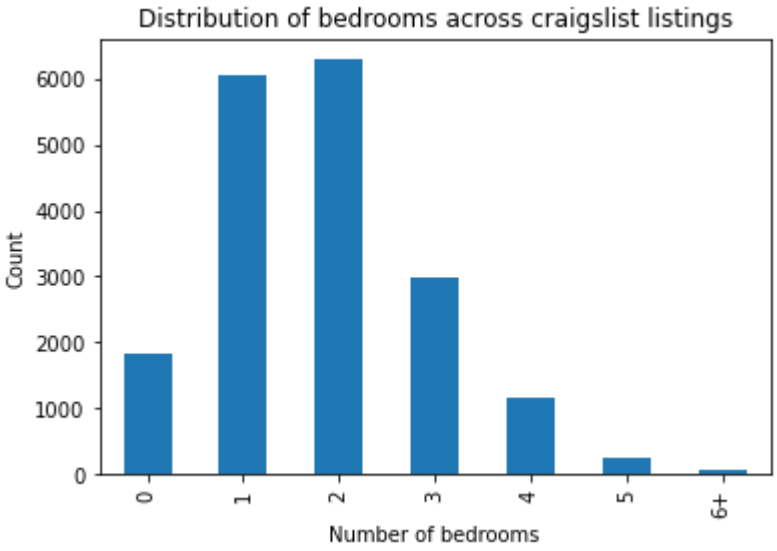

Figure 4.11 Bar graph showing the distribution of bedrooms in the dataset

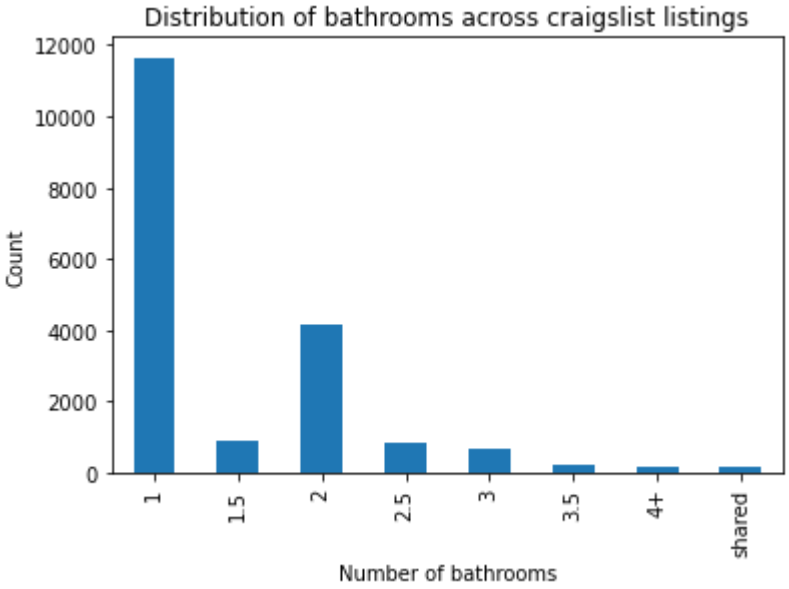

Figure 4.12 Bar graph showing the distribution of bathrooms in the dataset

Figure 4.13 Heat Map showing the most common bedroom/bathroom combinations in the dataset

The most common bedroom/bathroom combos are:

- studio with 1 bathroom
- 1 bedroom with 1 bathroom
- 2 bedroom with 1 bathroom
- 3 bedroom with 2 bathrooms
- 4 bedroom with 2 bathrooms
- 5 bedroom with 3 bathrooms
- 6+ bedroom with 3 bedrooms

4.3.2 What fraction of listings allow pets?

Surprisingly, only 43% of listings allow cats. More places allow cats than dogs, and only 20% of properties allow all animals (such as birds, reptiles, farm animals)

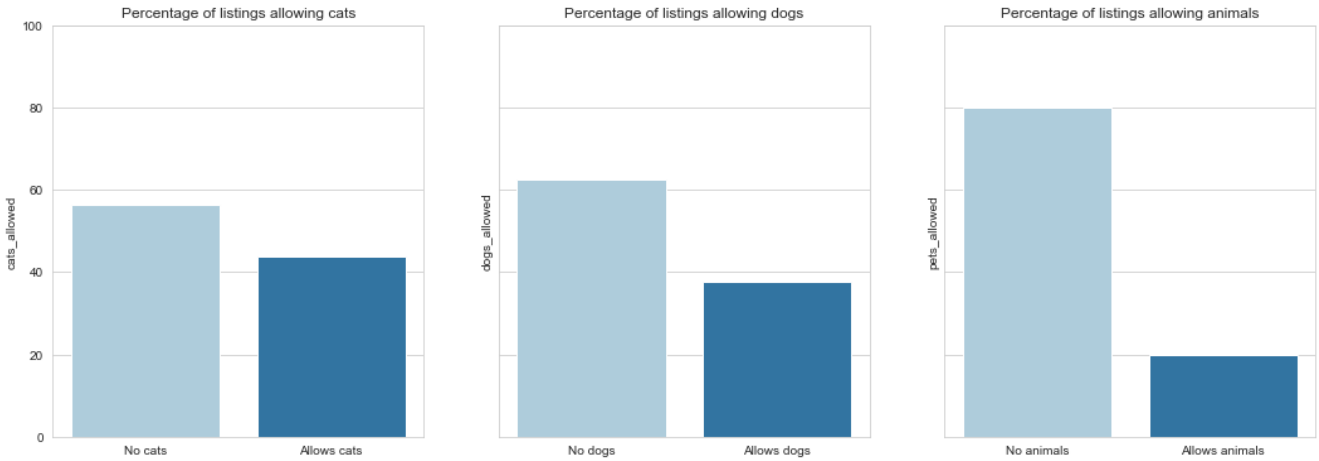

Figure 4.14 Bar Graphs showing the percentages of listings that allow cats, dogs, or all pets

4.3.3 Parking variants

The majority of places have some form of parking, although 25% only have street parking.

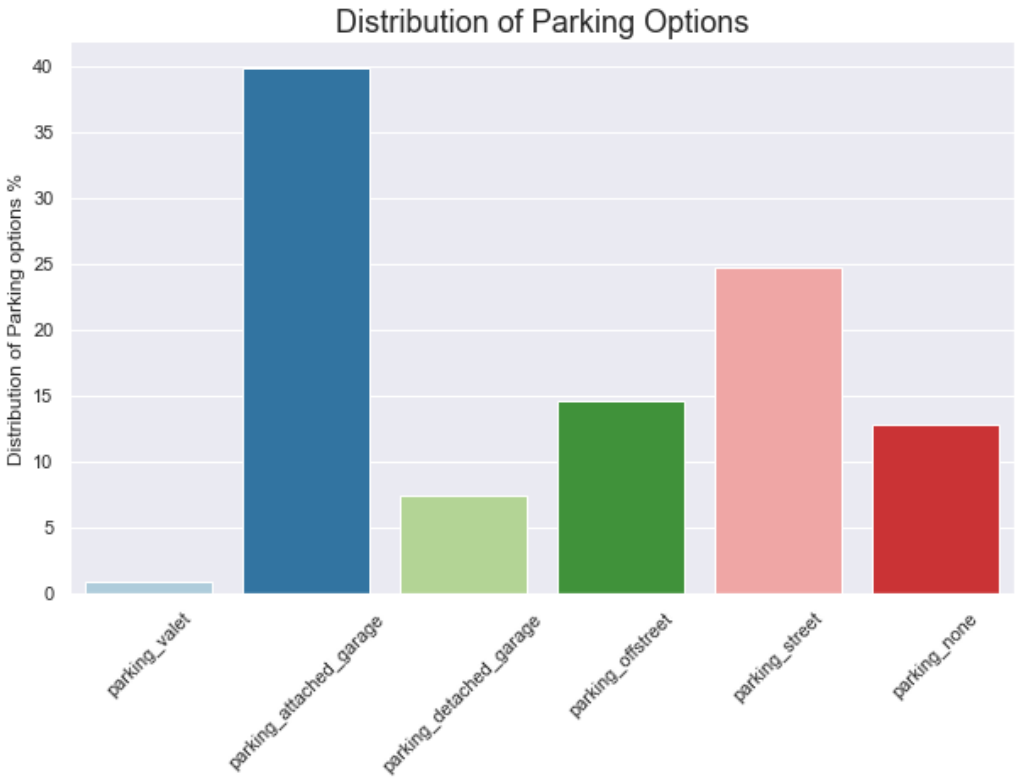

Figure 4.15 Bar Graph showing distribution of parking options in craigslist

4.4.4 Laundry variants

Approx. 85% of listings have some form of laundry in-unit or onsite which is pretty good.

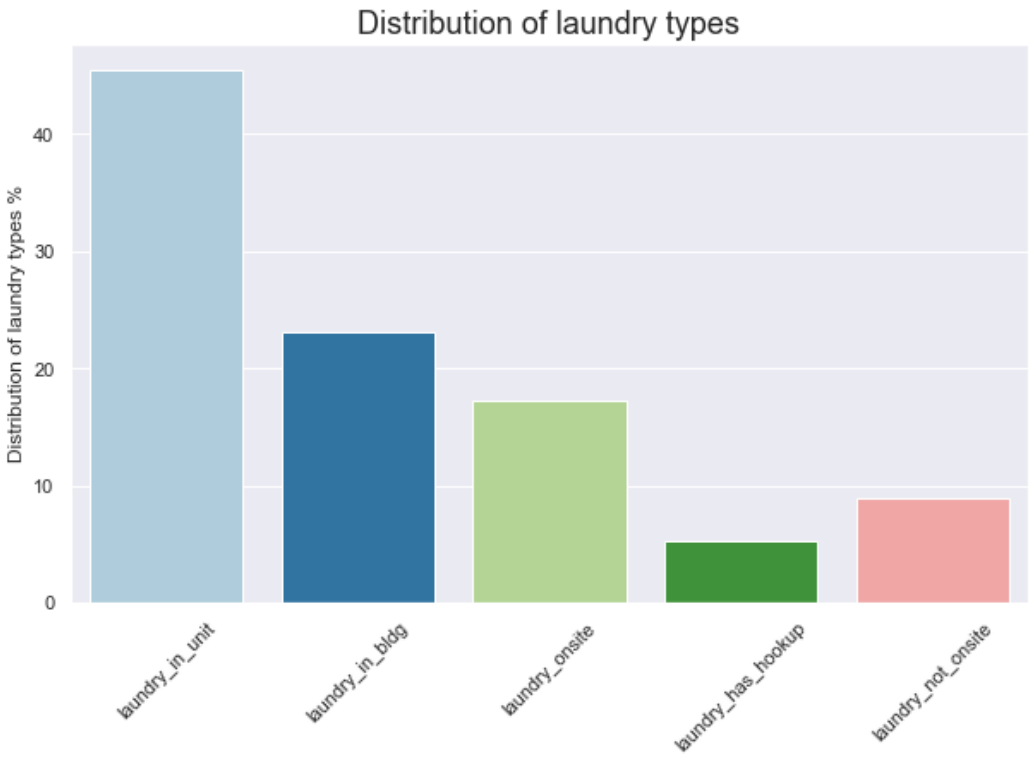

Figure 4.16 Bar Graph showing distribution of laundry options in craigslist

4.4.5 Walk Score, Transit Score, Bike Score distributions

Walk score, transit score, bike score: It is noteworthy that there are many properties that scored 100/100, and the average walk score is 80/100

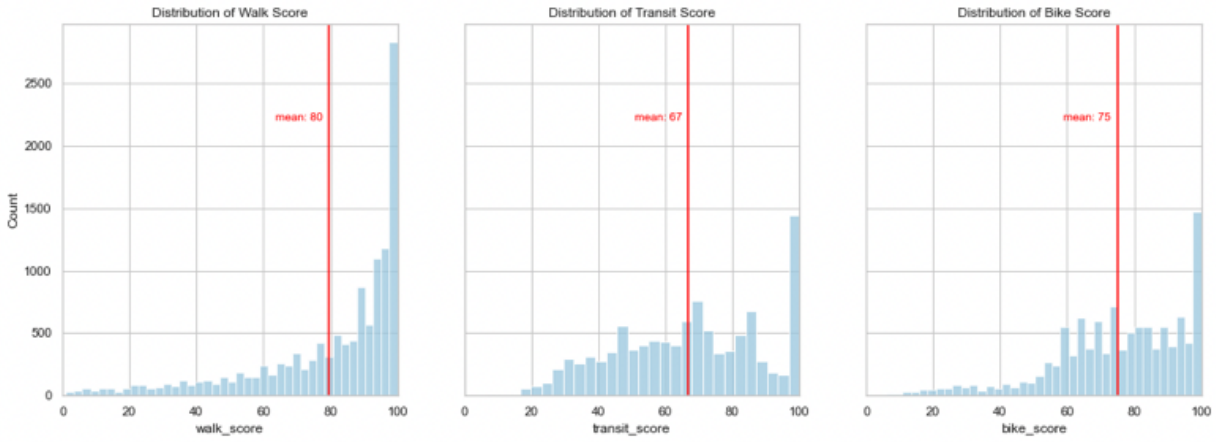

Figure 4.17 Bar Graphs showing the distribution of Walk score, transit score and bike score

5 Machine Learning

5.1 Preprocessing and feature engineering

The model is trained on 70% of the data, with the remaining 30% used to evaluate performance of the models. Before preprocessing, the data frame has 13020 rows of listings.

The numeric features that were not binary features were scaled using Standard Scaler from SKlearn. These features were: listing_sqft, walk_score, transit_score, bike_score, no_bedrooms, no_bathrooms, price_per_sqft, price_estimate_sqft_nh

In the data wrangling stage, the listing's address was converted into an equivalent walk score, transit score and bike score. The exact address doesn't matter as much as its proximity to shops, food, and transit options. Approx. 30% of addresses' walk scores were missing/not collected and were imputed in the preprocessing pipeline.

Many features are created in the preprocessing step after the training and testing set split to prevent data leakage:

- Replace neighborhood (81 in total) with the average price per square foot of that neighborhood
- Around 30% of listings did not provide a square footage value. Impute any missing square footage values based on the median square footage grouped by number of bedrooms and number of bathrooms
- Remove price outliers for one bedroom places - a bedroom of 1 was the default dropdown option so unsurprisingly, this had the most number of mislabeled units and the biggest range of prices. This was done by capping the top 1% percentile of prices for one bedroom listings.

5.2 Performance Metrics

The distribution of the price feature has a long right tail, so it made sense to use Mean Absolute Error (MAE) over Root Mean Squared Error (RMSE) because the squared term in RMSE penalized large errors too heavily without improving the model's performance. MAE is a better choice because the units are in dollars, so it is easier to interpret. However, the majority of the rent prices ranged between $100 and $6000, so a MAE of $300 is hard to compare across different listings.
Thus, the Mean Absolute Percentage Error (MAPE) was chosen as the accuracy metric as it could be easily compared between models and between listings.

To tune hyperparameters use SKLearn's Gridsearch with 5 fold Cross validation.

Machine Learning models:

1. Baseline Model
2. Stochastic Gradient Descent
3. Random Forest Regressor
4. XGBoost Regressor
5. LGBM Regressor
6. Piecewise SGDR by number of bedrooms
7. Piecewise SGDR by neighborhoods

5.3 Baseline Model

Create a baseline model with only 1 feature: the listing's square footage multiplied by the average price per square foot of each neighborhood. This captures the 'expensiveness' of the neighborhood and the size of the listing. The square footage was the most correlated feature with listing price with a pearson correlation of 0.71.

Table 5.1 Baseline metrics on the test data

Model	R^2 score	RMSE	MAE	MAPE %
Baseline Model	0.60	973.31	656.30	19.8%

5.4 Stochastic Gradient Descent Regressor

Train a Stochastic gradient descent model using Grid Search with 5 cross validation folds. The model is predicting lower prices for a listing than the actual price as prices increase. Figure 5.2 shows a longer right tail which reflects the original price distribution.

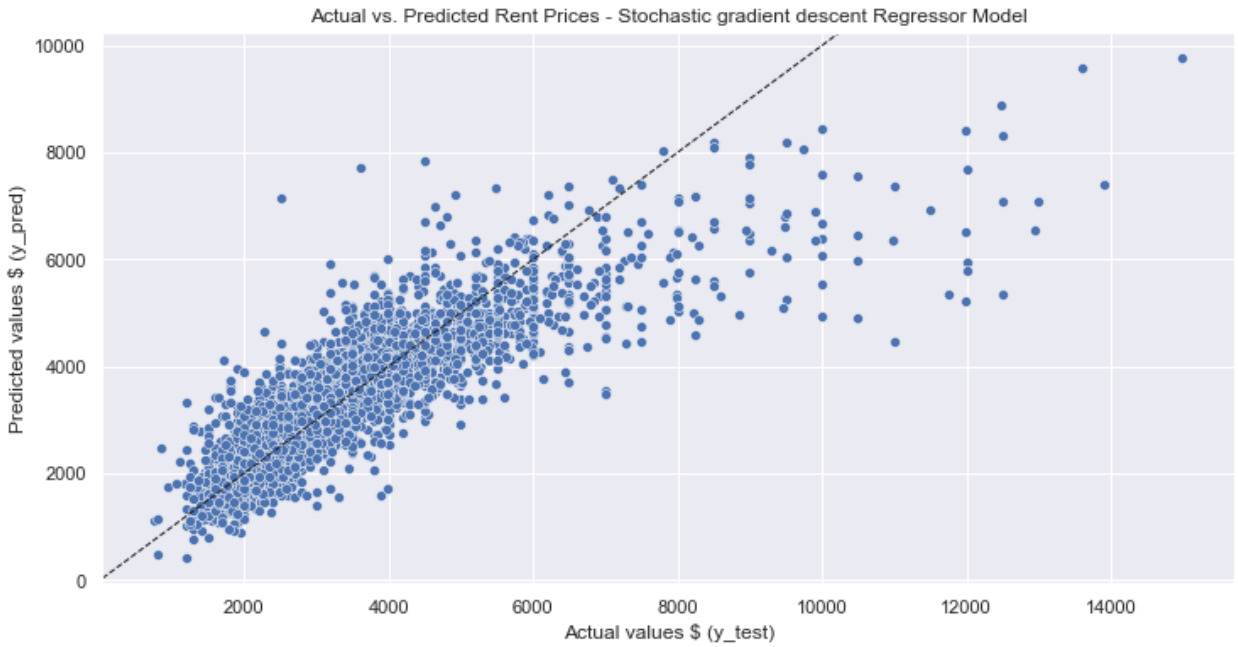

Figure 5.1 Scatter plot of predicted vs. actual listing price for the SDGR model

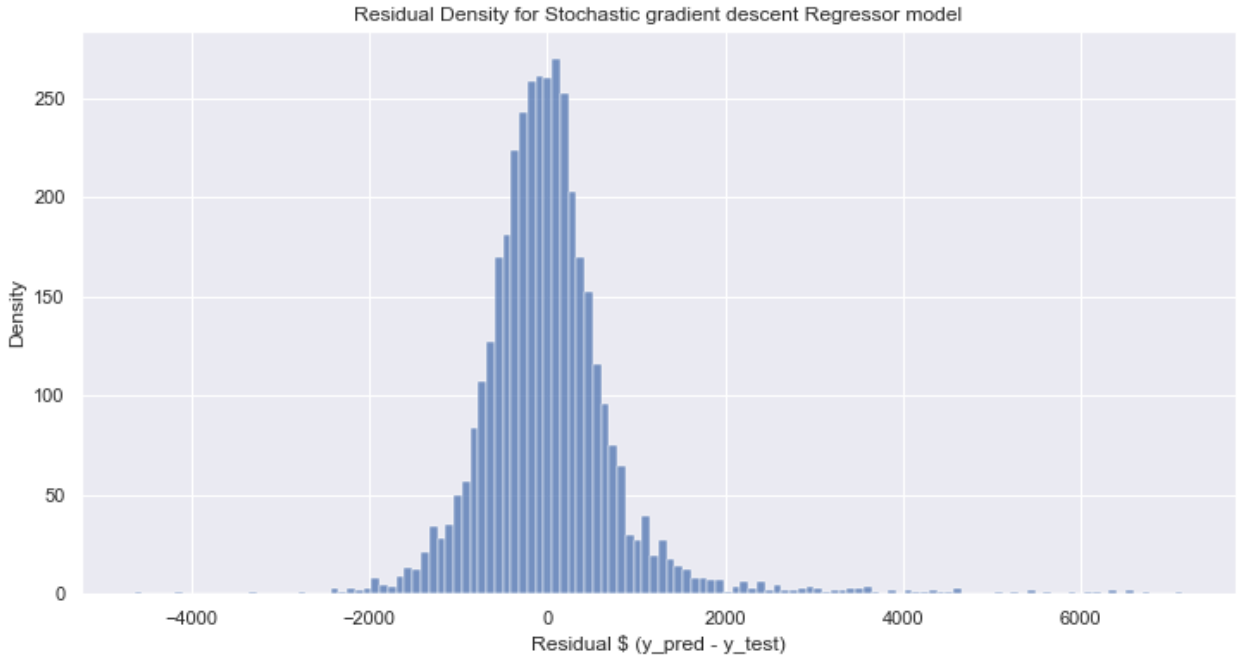

Figure 5.2 Bar plot showing the distribution of residuals (predicted - actual listing price) for the SDGR model

Plotting the residuals grouped by number of bedrooms and binned by square footage to see if there is any obvious clustering to explain where residuals occur.

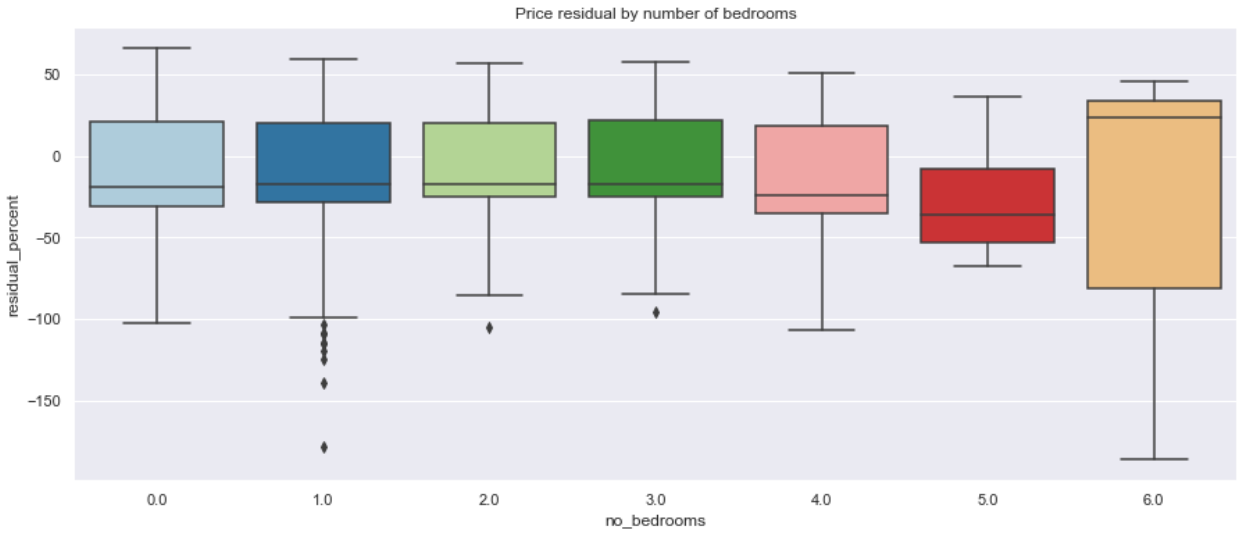

Figure 5.3 Boxplot of residuals distributed by number of bedrooms

The prediction accuracy is the worst for 1 bedroom places. This could be because they are the most common listing size. Another way to segment the listings would be by square footage. The average square footage sizes are:

- Studio apartments: 400-600 sqft
- 1 bedroom apartments: 700-800 sqft
- 2 bedroom apartments: 900- 1200 sqft
- 3+ bedroom apartments: 1300 sqft+

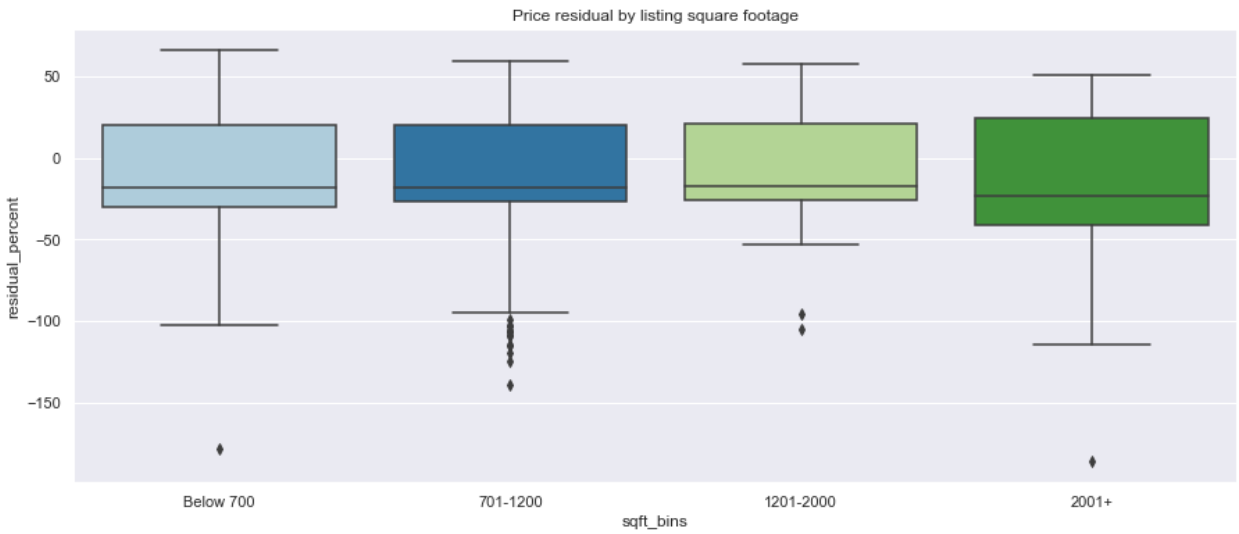

Figure 5.4 Boxplot of residuals distributed by square footage

5.5 Random Forest Regressor

Train a Random Forest Regressor model using Grid Search with 5 cross validation folds. These residual plots show that as listing price increases, the residuals get bigger, with the predicted value often being lower than the actual value.

Figure 5.5 Scatter plot of predicted vs. actual listing price for the Random Forest model

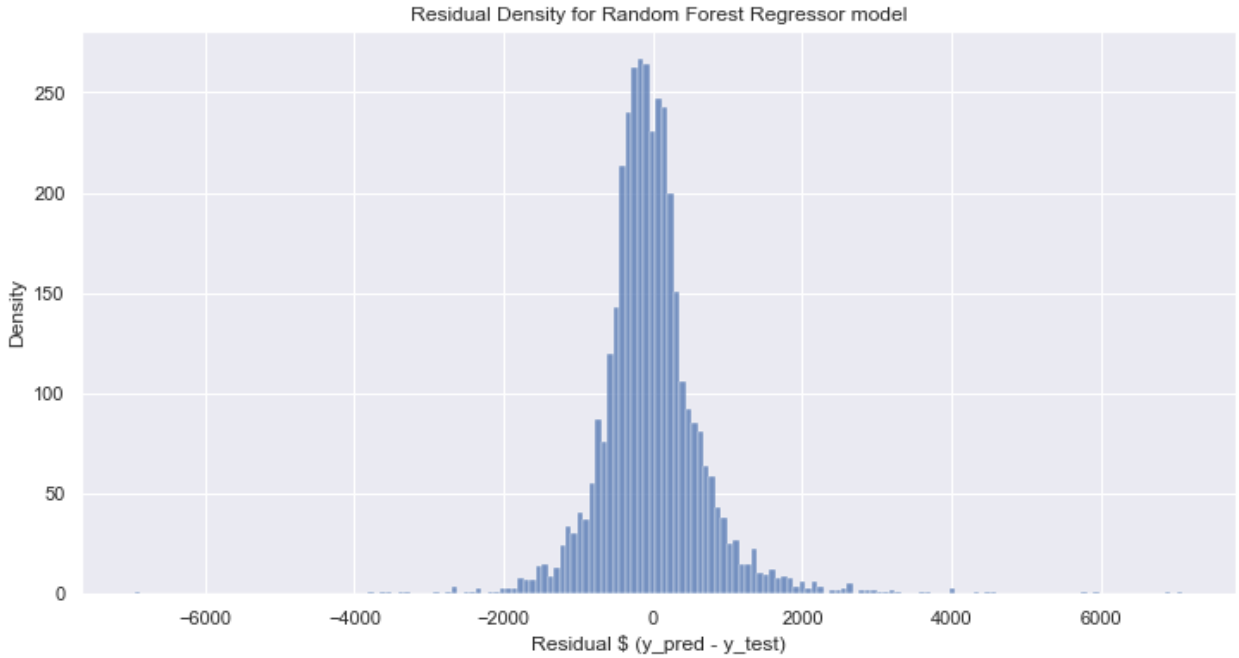

Figure 5.6 Bar plot showing the distribution of residuals (predicted - actual listing price) for the Random Forest model

5.6 XGBoost Regressor

Train a XGBoost model using Grid Search with 5 cross validation folds. Again, these residual plots show that as listing price increases, the residuals get bigger, with the predicted value often being lower than the actual value.

Figure 5.7 Scatter plot of predicted vs. actual listing price for the XGBoost model

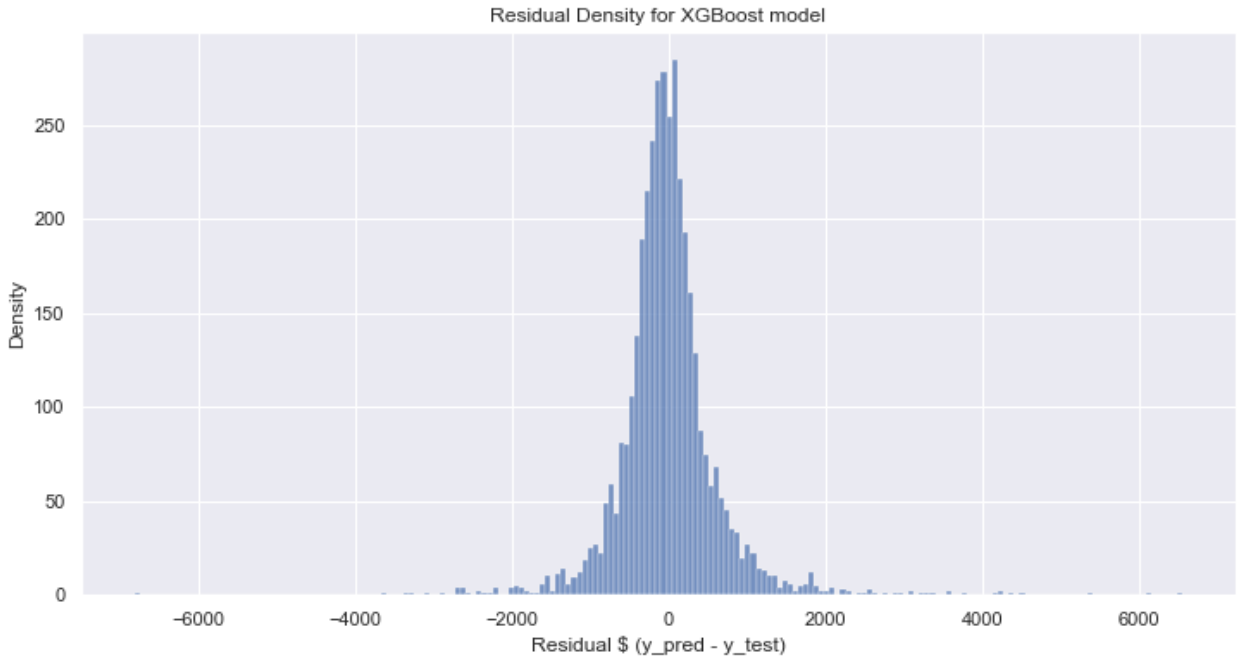

Figure 5.8 Bar plot showing the distribution of residuals (predicted - actual listing price) for the XGBoost model

5.7 LGBM Regressor

Train a LGBM model using Grid Search with 5 cross validation folds. Again, these residual plots show that as listing price increases, the residuals get bigger, with the predicted value often being lower than the actual value. There is a longer right tail as shown on the bar graph.

Figure 5.9 Scatter plot of predicted vs. actual listing price for the LGBM model

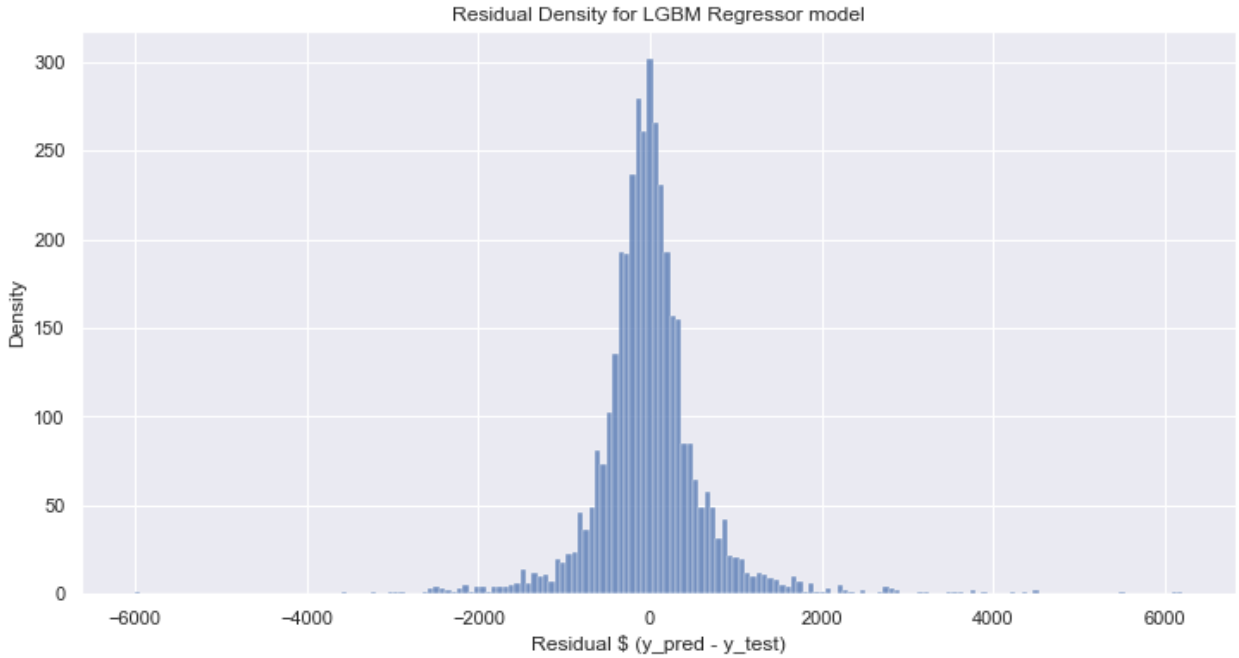

Figure 5.10 Bar plot showing the distribution of residuals (predicted - actual listing price) for the LGBM model

5.8 Piecewise Stochastic Gradient Descent Regressor by number of bedrooms

Train a piecewise model using Grid Search with 5 cross validation folds. Split the dataset into 5 subsets based on the number of bedrooms to introduce some nonlinearity to the SGDR model. Partition the data into groups where each has 0, 1, 2, 3, 4-6 bedrooms. These residual plots show that as listing price increases, the residuals get bigger, with the predicted value often being lower than the actual value. There is also a longer right tail as shown on the bar graph.

Figure 5.11 Scatter plot of predicted vs. actual listing price for the piecewise SGDR model

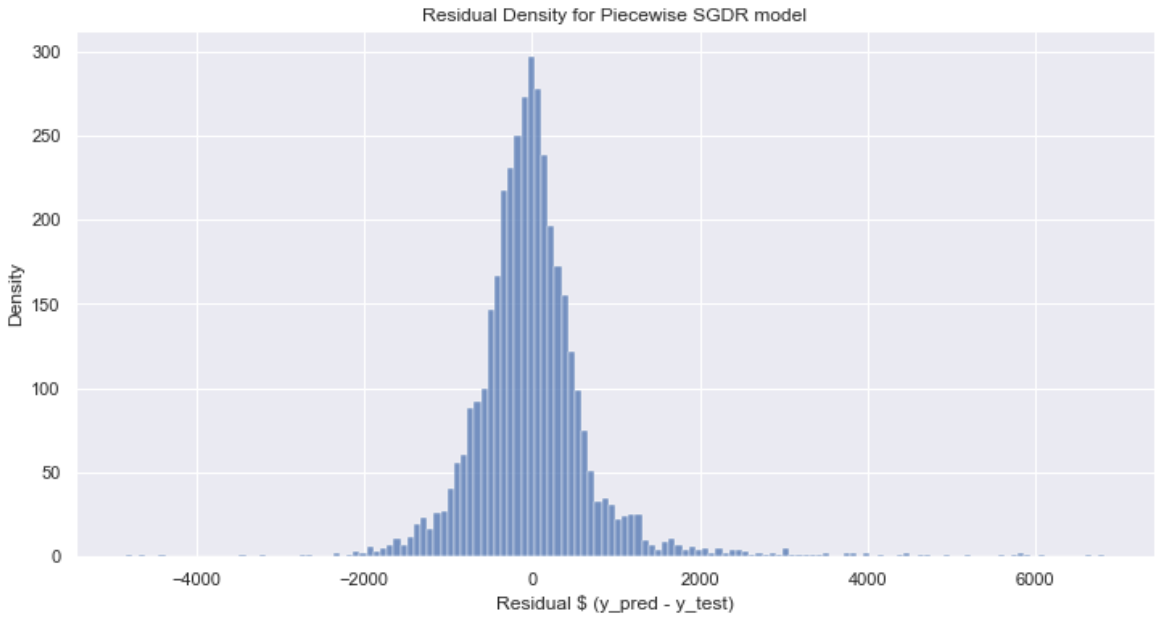

Figure 5.12 Bar plot showing the distribution of residuals (predicted - actual listing price) for the piecewise SGDR model

5.9 Piecewise Stochastic Gradient Descent Regressor by neighborhood

Train separate SGDR models using Grid Search with 5 cross validation folds for each neighborhood that has more than 100 listings. There are 36 models trained out of 81 neighborhoods in total, representing 76% of the data. Compare the results to the SDGR model for all neighborhoods in section 5.4

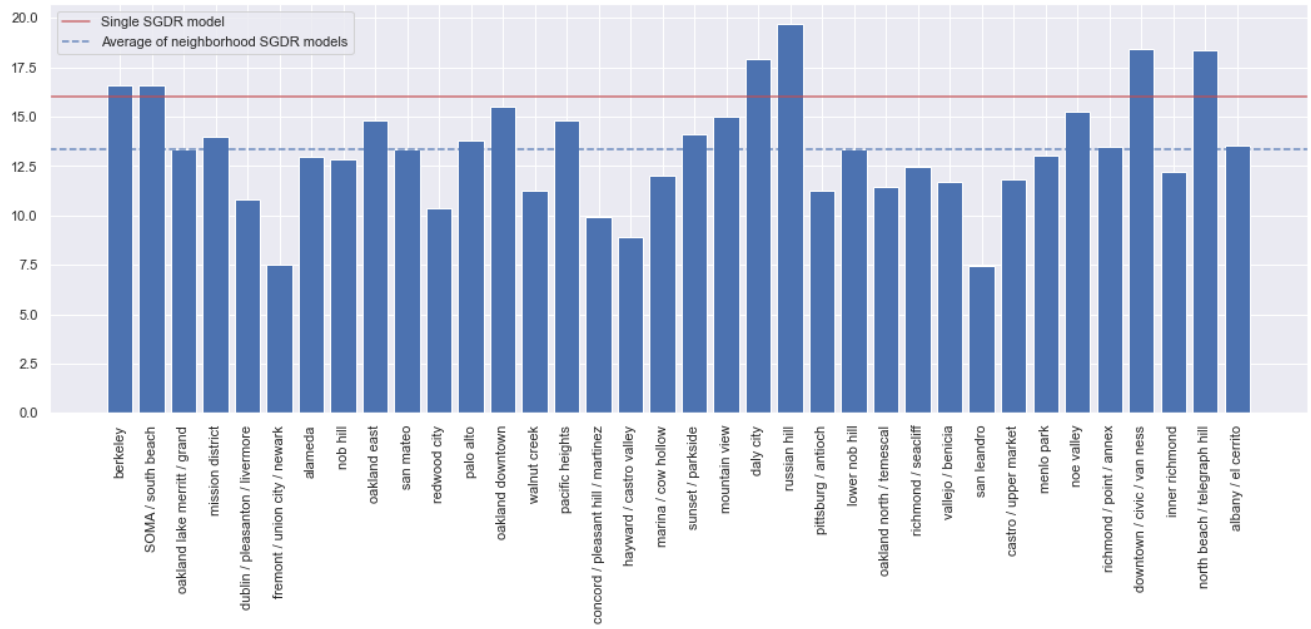

Figure 5.13 Bar plot showing each neighborhood's SGDR model performance against the single SGDR model for all 81 neighborhoods.

Overall, the individual neighborhood SGDR models produced better accuracy results than the single SGDR model. Some neighborhoods performed much better such as Fremont, Hayward, San Leandro. Other neighborhoods performed worse than the single SGDR model such as Russian Hill or North Beach.

Residual plots of the best performing neighborhood model: these show small residuals symmetrically distributed around a residual of $0.

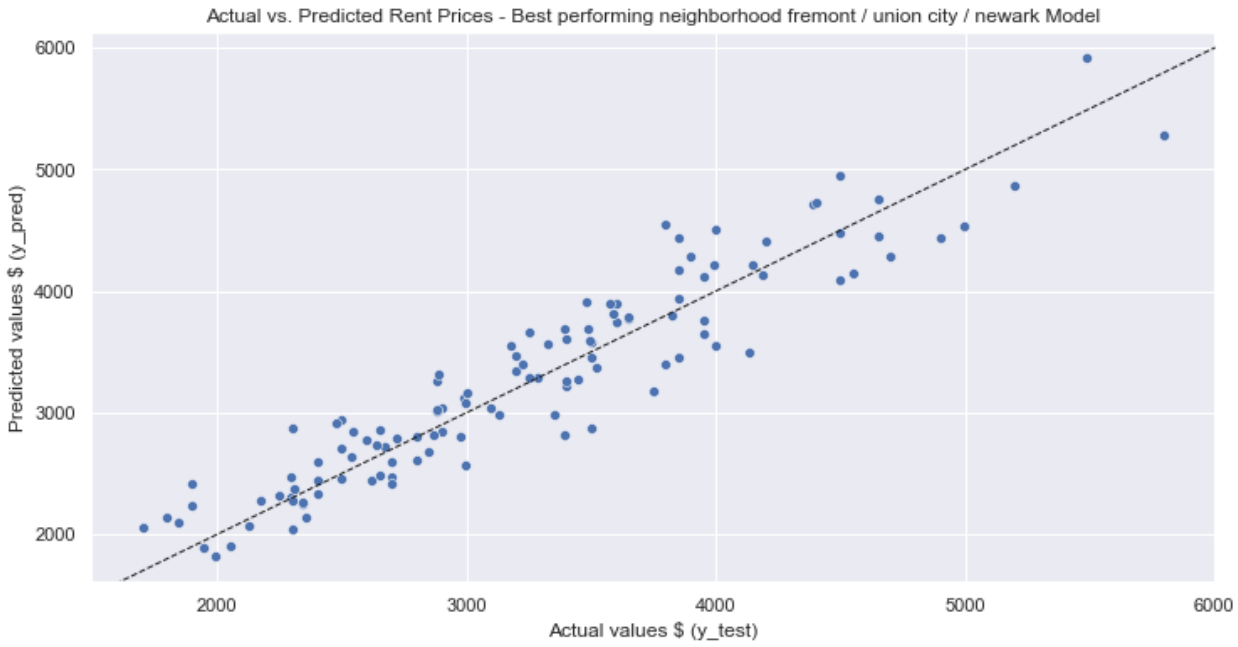

Figure 5.14 Scatter plot of predicted vs. actual listing price for the best performing piecewise SGDR neighborhood model

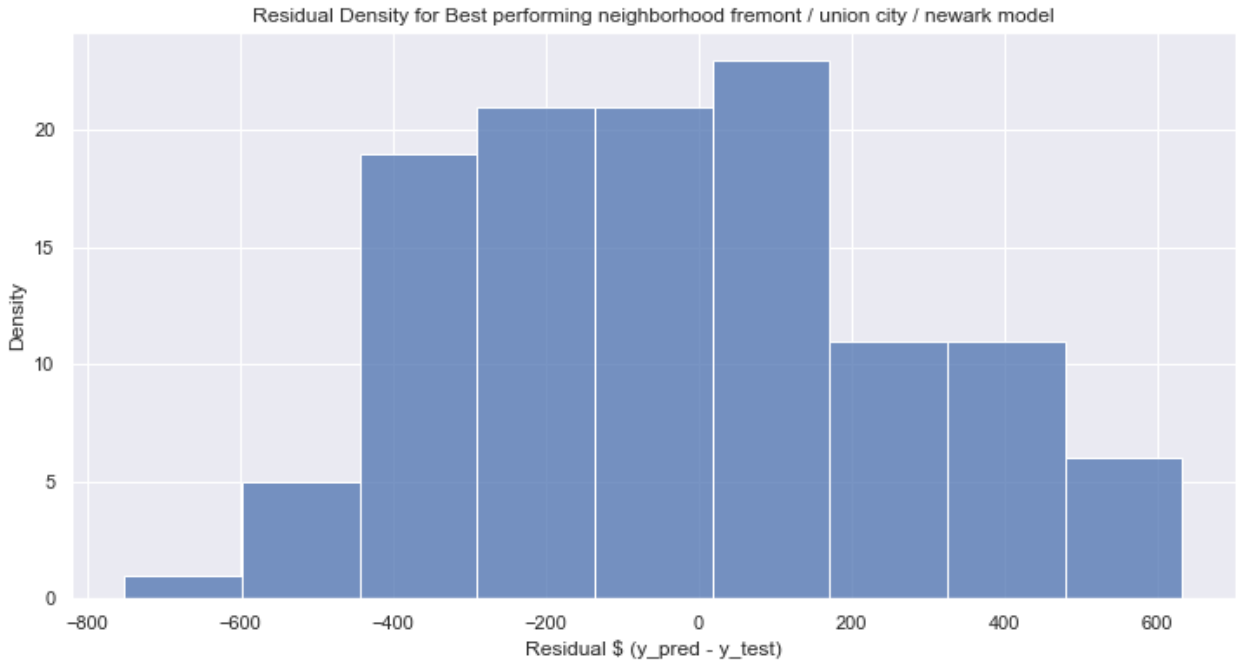

Figure 5.15 Bar plot showing the distribution of residuals (predicted - actual listing price) for the best performing piecewise SGDR neighborhood model

5.10 Conclusion/Interpretation of results

Unsurprisingly, the worst performing model was the baseline one, which was only predicting rent price based on the average price per square foot of each neighborhood multiplied by the square footage of the property.

XGBoost Regressor and LGBM Regressor performed the best, with a MAPE accuracy score of 12%. The XGBoost and LGBM Regressor both performed well as shown in the performance metrics table. A mean average prediction error of 12% is pretty good for one model on a wide range of listing sizes and locations. Since the ultimate objective of this model is to predict the rental price for the test data, the best model is the XGBoost Regressor.

The piecewise SGDR models showed an improvement in the linear model, especially training separate models by neighborhood. The piecewise SGDR models showed an improvement in the linear model, especially training separate models by neighborhood. If there was more data, the smaller neighborhoods (ones with less than 100 listings) could have been trained too, but defaulting to the overall model for smaller neighborhoods works ok too.

Table 5.2 Performance metrics on the test data

Model	R^2 score	RMSE	MAE	MAPE %
Baseline Model	0.60	973.3	656.3	19.8%
Stochastic Gradient Descent	0.69	854.8	547.0	16.6%
Random Forest Regressor	0.76	750.5	482.8	14.6%
XGBoost	0.81	670.5	416.9	12.3%
LGBM Regressor	0.81	671.6	419.7	12.4%
Piecewise SGDR by number of bedrooms	0.76	755.3	476.0	14.2%
Piecewise SGDR by neighborhood	0.73	664.5	445.1	13.4%

For each machine learning model, we are looking for a symmetric and uniform distribution of residuals. For all models, the residuals were slightly left skewed, as the model tends to overpredict at higher rent prices. This could be due to fewer data points at the higher price range, but it could also be that market rent for more expensive properties is driven less by the number of bedrooms or square footage in a linear manner and more to do with the fact that there is less demand for $10,000/month properties for rent.

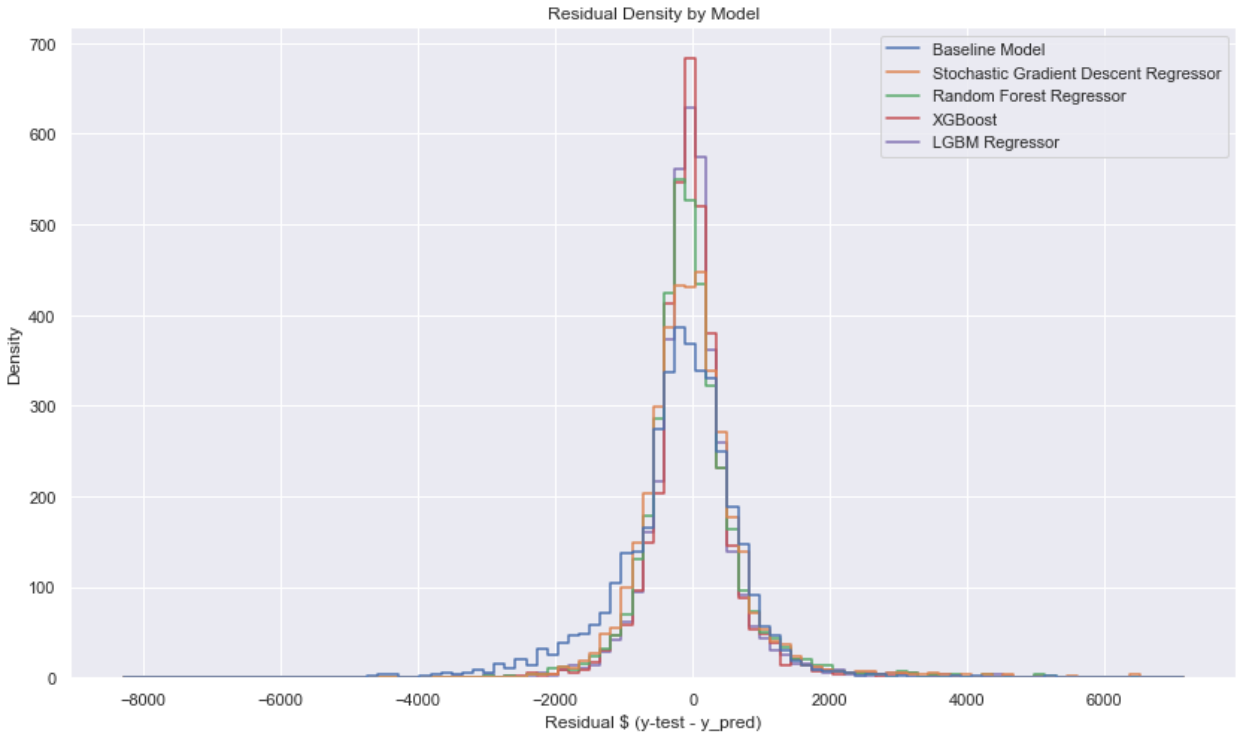

Figure 5.16 Residual distribution plot of all the machine learning models

The baseline model performed the worst, with the lowest density around residual = 0 and higher density in the left tail of the residual plot, indicating the model is predicting too high. The XGBoost and LGBM model performed the best, with the highest density around residual = 0 as shown in Figure 5.16

5.10 Feature Exploration

Feature importance of the best model XGBoost are plotted. The top features in the model are the size of the property i.e. square footage and number of bedrooms/bathrooms, and features to do with the location - the "expensiveness" of the neighborhood and the proximity to shops and transit.

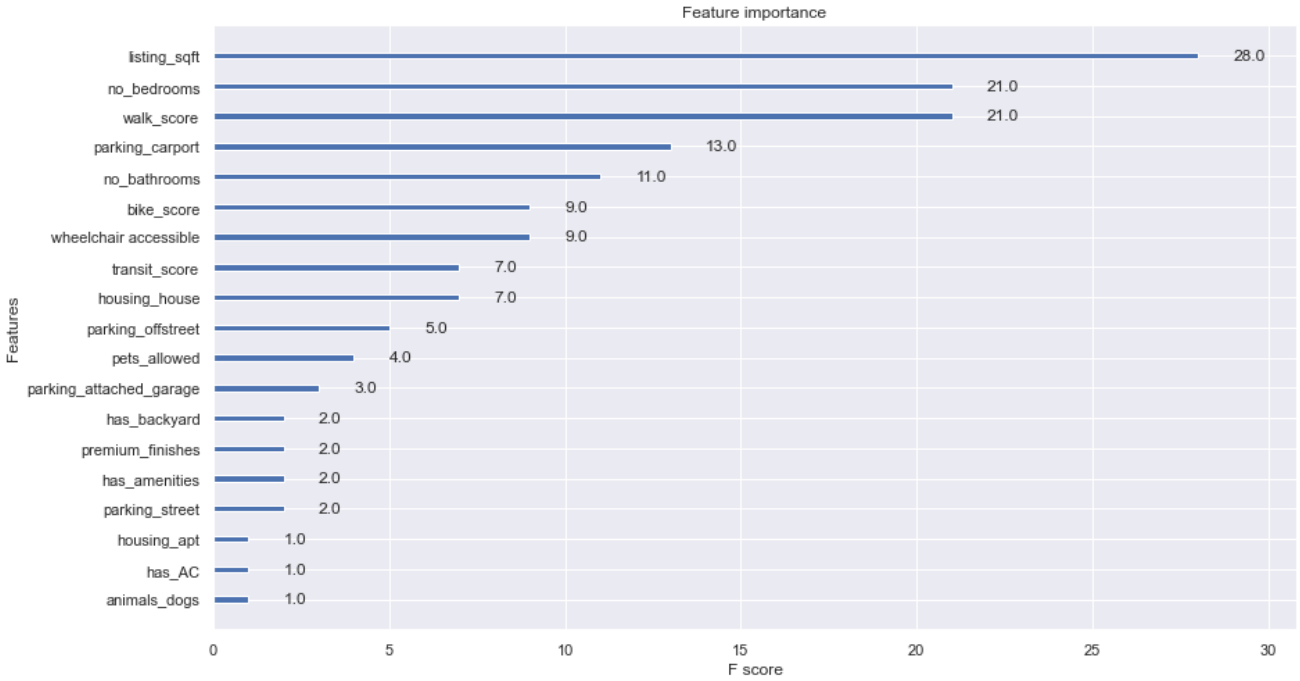

Figure 5.17 Feature importance graph of the XGBoost model

Plot the feature importances of the linear stochastic gradient descent regressor model using SHAP. The feature 'no_bedrooms' is the most important feature, with clusters shown at each 7 options. The next most important feature is 'listing_sqft', and has a high Shapley value range. The points are offset in the y-axis to show the distribution of the shapley values (there is a cluster of them in the blue range suggesting most are on the low end of the range, with a long right tail of red).

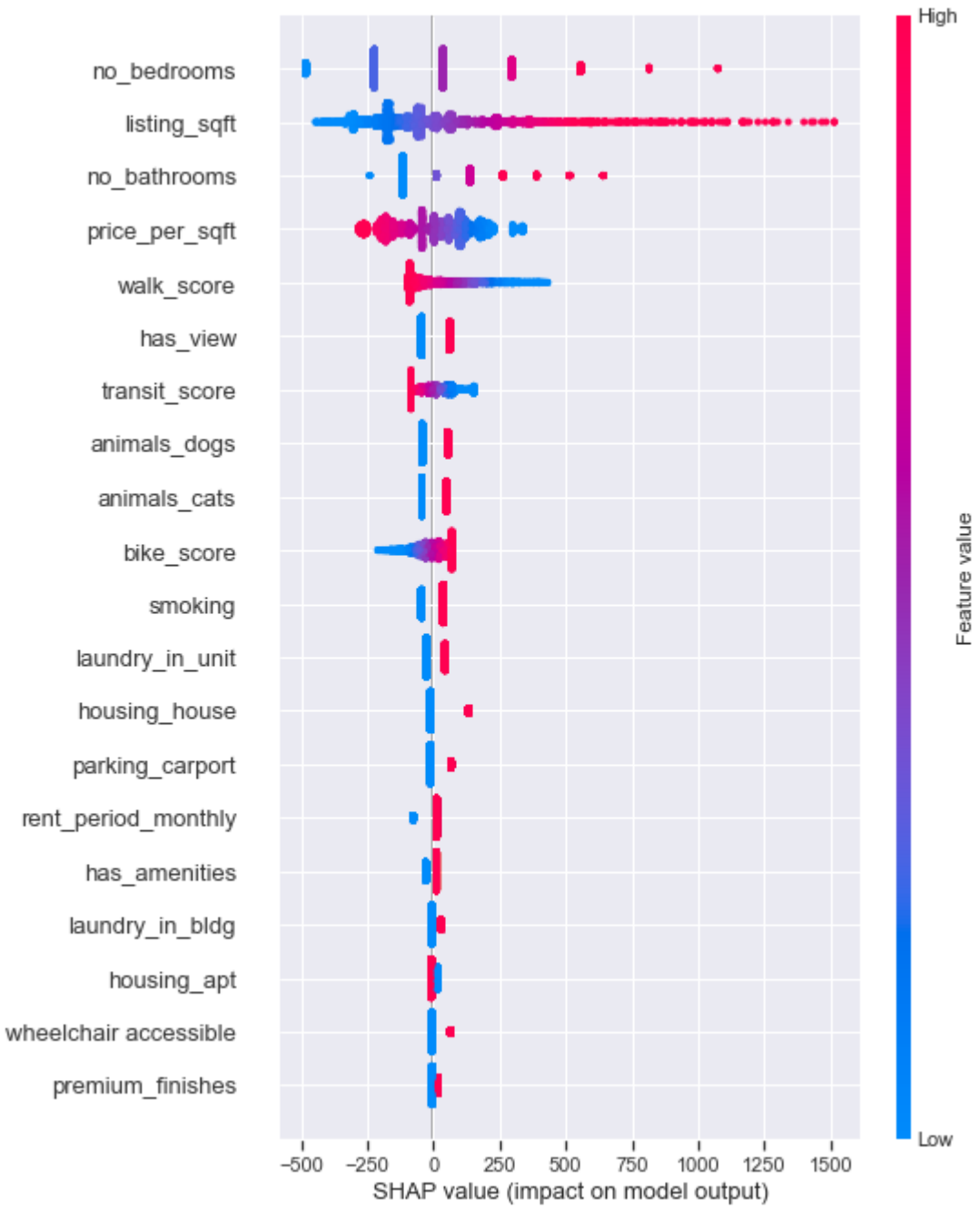

Figure 5.18 SHAP beeswarm plot showing the SHAP values of each feature

Plot the top positive and negative feature coefficients for the SDGR model. As expected, features that increase the size of the property all raise the rental price.
The top negative feature coefficients are also interesting: being an apartment vs. a house, not having any forms of parking, allowing pets, not having laundry at all or having laundry but inside the building and not the unit.

Figure 5.19 Bar chart showing the top positive and negative feature coefficients that impact listing price for the SGDR model

Look at the average contribution to the overall rental price of binary features:

Features of interest:

- Allows pets
- Is wheelchair accessible
- Allows smoking
- Has Air Conditioning
- Has EV charging
- Has laundry in-unit
- Has offstreet parking (this is a weighted average feature combining the features 'parking_carport', 'parking_attached_garage','parking_detached_garage', 'parking_offstreet')
- Has Valet parking
- Has a high walk score/transit score
- Has amenities
- Has premium finishes
- Has a backyard
- Has a view

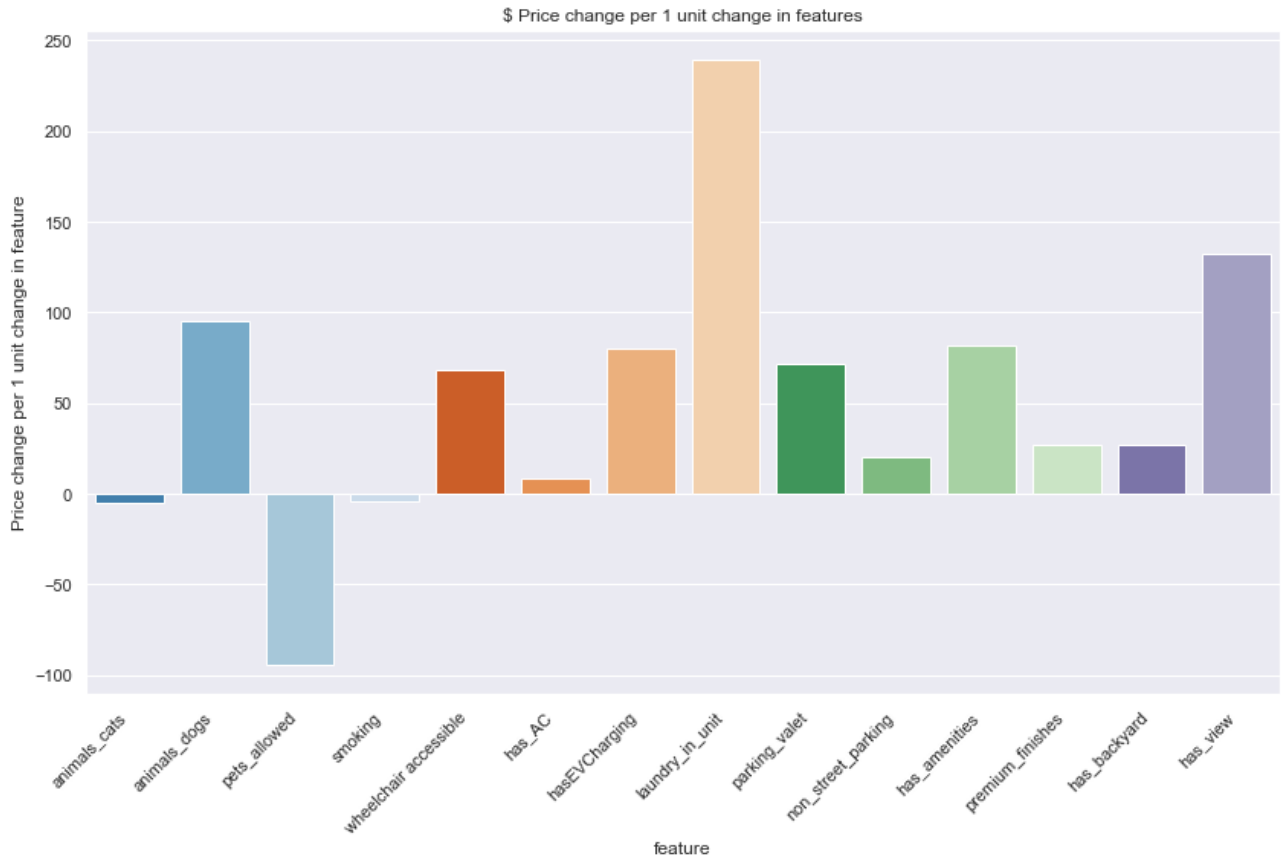

Figure 5.20 Bar chart showing the feature coefficients of features of interest of the SGDR model

Some interesting feature coefficients are in unit laundry which has a $240 premium, and listings with amenities have an $82 premium.

6 Applications of the price prediction model

Optimizing for market rent:

For renters looking to find a place to rent: they could use this model to predict the market rent for prospective properties, and prioritize the ones that are priced lower than the predicted market rent. This would be a good starting point that narrows down the housing search, optimizing for lowest rent.

For property owners looking to rent their place on craigslist: they could use this model to predict the market rent for their property and adjust the rent to increase revenue or increase competitiveness.

Cost Benefit analysis of feature coefficients

The business problem came about because properties vary so much it is hard to determine how features influence rental price. By looking at the feature coefficients, the linear model provides some interesting insights, since they can be used by renters and property managers to make informed decisions about pricing and how much features are worth.

For example, properties with in-unit laundry charged an average of 248 dollars/month more in rent than those without. Property owners can take these feature insights and do cost benefit analysis for areas of home improvement.

Renters may use the feature exploration results to decide if it is worth the money to choose a property with a higher walkscore (3.76 dollars/1 unit increase in walkscore) to cut down on driving/transportation costs or forgo a car entirely. Or it may be cheaper to rent a place with a gym or pool amenity worth an average of 92 dollars/month more which may be more economical than paying for a gym membership.

Flagging scams priced below market rent

A contextual piece of information to note when interpreting the model results is that the data comes from past and current Craigslist ads, and Craigslist rental scams are prevalent (accounting for 6% of listings per a study done on this subject), which would appear to bring down the average market rent.

The listings that are deemed scams do get flagged and are eventually removed, but they do appear on Craigslist and I would estimate that a similar fraction of the data used to build this machine learning model are scams. Typically, rental scams are posted using real property information copied off a real listing, and are hard to detect. The listed rental price is typically always lower than market value to attract people to click on the ad. Unfortunately, these artificially lower prices are not representative of the market rent of a property because it would not be available to rent. This model could be a starting point to flag the properties that are priced lower than market value.

Counterfactual analysis

Users could take this price prediction model to do counterfactual analysis using Bayes Optimization to predict the combinations of features to change the listing price of a property. For example, what are the changes a homeowner can make to their property (i.e. features to upgrade) to increase their property value by a certain amount e.g. 100K

Appendix

The Data:

Column name	Datatype	Description
listing_sqft	float	The unit's area measured in square foot
animals_cats	boolean	Allows cats
animals_dogs	boolean	Allows dogs
smoking	boolean	Allows smoking
wheelchair accessible	boolean	Is wheelchair accessible
has_AC	boolean	Has Air Conditioning
hasEVCharging	boolean	Has Electric Charging for vehicles
laundry_in_bldg	boolean	The unit has laundry in the building but not inside the unit
laundry_in_unit	boolean	The unit has laundry in the unit
laundry_has_hookup	boolean	The unit has no laundry but can be installed
laundry_onsite	boolean	The unit has laundry onsite but outside the building
laundry_not_onsite	boolean	The unit does not have any laundry options
parking_carport	boolean	Has parking in a carport
parking_attached_garage	boolean	Has parking in an attached garage
parking_detached_garage	boolean	Has parking in a detached garage
parking_offstreet	boolean	Has parking offstreet e.g. driveway
parking_street	boolean	Has street parking
parking_valet	boolean	Has valet parking
parking_none	boolean	Has no parking options
rent_period_monthly	boolean	The unit is rented on a monthly basis
housing_condo	boolean	The unit is a condominium
housing_apt	boolean	The unit is an apartment
housing_flat	boolean	The unit is a flat
housing_house	boolean	The unit is a house
housing_townhouse	boolean	The unit is a townhouse

walk_score	float	A measure of the walkability of the unit's surroundings - measured from 0 to 100, the higher the better
transit_score	float	A measure of how good the transit options are near the unit - measured from 0 to 100, the higher the better
bike_score	float	A measure of the bikeability of the unit's surroundings - measured from 0 to 100, the higher the better
no_bedrooms	float	The number of bedrooms in the unit
no_bathrooms	float	The number of bathrooms in the unit
is_rent_controlled	boolean	Is the unit rent controlled by local rental laws
pets_allowed	boolean	Does the unit allow pets - this could be animals other than cats or dogs
has_amenities	boolean	Does the unit have amenities (pool, gym, spa) for use
premium_finishes	boolean	Does the unit have any premium finishes: granite, marble, walnut, millwork, a fireplace, built-in cabinetry
ensuite_bath	boolean	Does the unit have an ensuite bathroom
has_balcony	boolean	Does the unit have balconies
has_backyard	boolean	Does the unit have a backyard
has_view	boolean	Does the unit have views/or is a penthouse
multi-level	boolean	Is the unit a multi-level 3, 4 or 5 story house
is_an_SRO	boolean	Is the unit an SRO. An SRO is a single room occupancy, generally room only, with access to shared bathroom and kitchen
price_per_sqft	float	The average price per square foot of listings in the unit's neighborhood