**Report**

This project gives insights into how different features of the hosted homes such as location, space, neighborhood, amenities, interaction and many other factors affect the prices of the households. The aim is to study the correlation between these parameters as well as to predict the prices based on different regression techniques and datasets used. We have selected data from two cities for this purpose- Boston and Seattle. The reason behind this is to show how different parameters for these cities compare with each other.

**Collection and Description Datasets**

**Boston Airbnb Open Data:**

<https://www.kaggle.com/airbnb/boston#listings.csv>

This dataset describes the listing activity of homestays in Boston, MA, since 2008.

It has about 95 columns.

**Seattle Listings**

[**https://www.kaggle.com/shanelev/seattle-airbnb-listings**](https://www.kaggle.com/shanelev/seattle-airbnb-listings)

The data was scraped on December 19th, 2018 and contains roughly 8000 listings of current Airbnb listings in Seattle. The data has the price, reviews, latitude, longitude, bedroom, bathroom, number of guests it accommodates, room type, and more.

**Preprocessing (Boston Dataset):**

All preprocessing was done in python, please refer to the colab: Boston.ipynb which was submitted for a more verbose representation of the features.

Initially, we were given 94 features that were web scraped from the Airbnb website.and looking over the following features we were able to remove 50 features:

* Features that were non numeric and non-categorical, for eg: Name,Decription have been removed. Also the ID columns were removed
* Features that were associated with location were removed as we had percise numerical values associated with this (Longitude & Latitude).

Following, for our data removal, we were left with 43 to work with; 13 of which were floats, 12 being ints, and 19 objects that we would have to clean/modify:

* Categorical columns(property\_type', 'room\_type','bed\_type', 'cancellation\_policy') to be label encoded then one hot encoded.
* Columns which had true false values have been converted to 0 & 1 value columns.
* Columns(Security\_deposit,cleaning\_fee), null values were replaced with 0 because there were 2000+ null values for these columns and if owners didn’t include this on the website it would be assumed there were no fees in the first place
  + For other features with null values, we replaced by mean.x

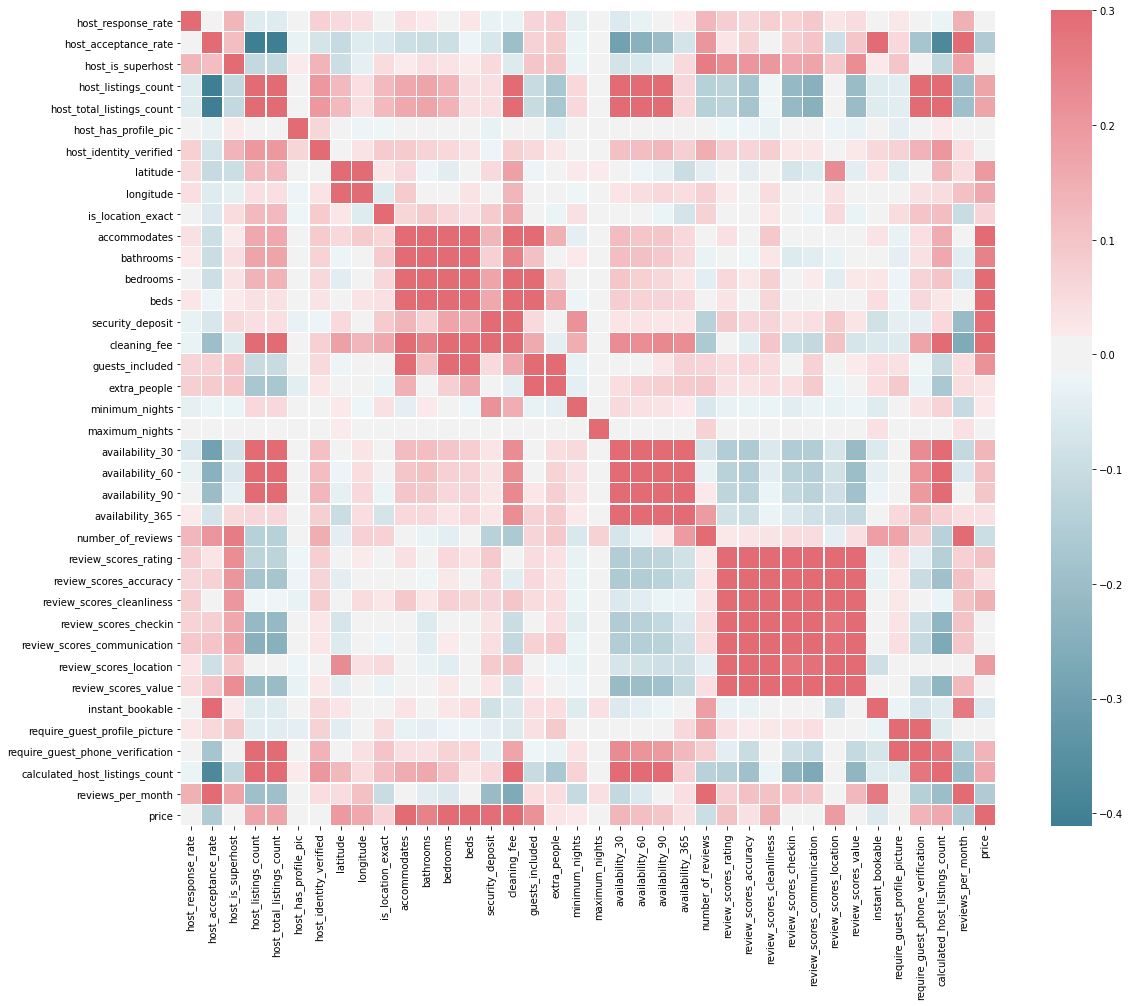
**Boston EDA**

**Boston HeatMap:**

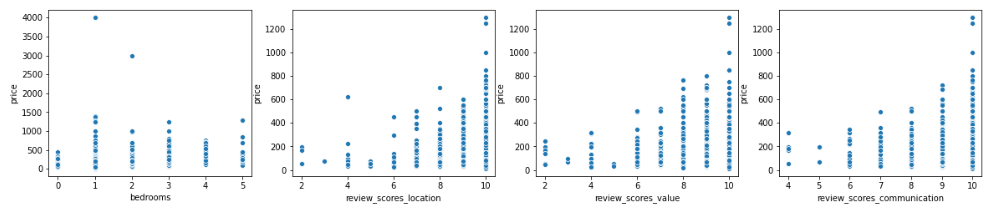
Columns that are “correlated” are those which are derived from each other and should be removed, ei: availability X Days, and Review Score columns, or ones that have correlation such as # bedrooms and bathrooms, and # accommodates.

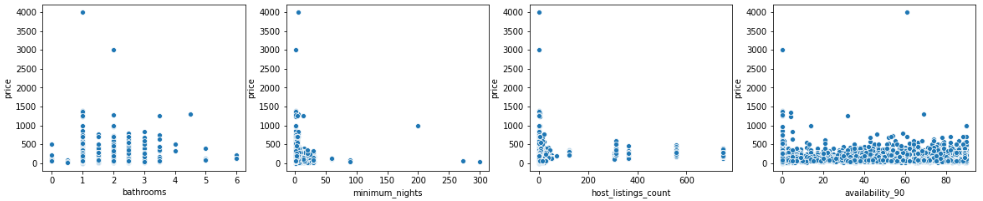
If anything, it is surprising that the correlation coefficient isn’t higher than .5 for these columns.

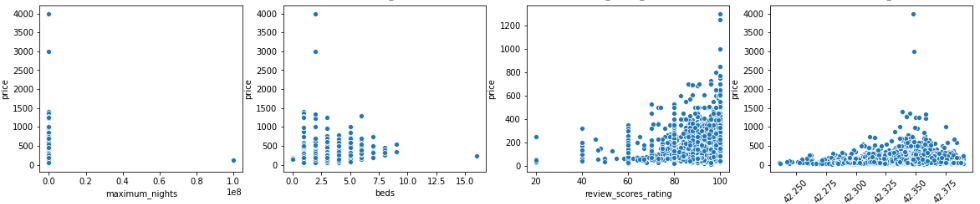
Price has highest correlation with #bathroom, #bedrooms, accommodates, is\_location \_exact, security\_deposit.

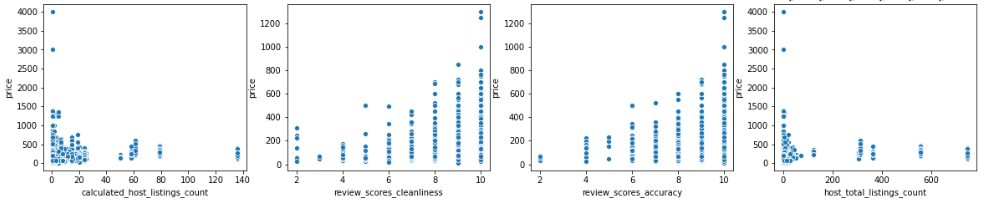


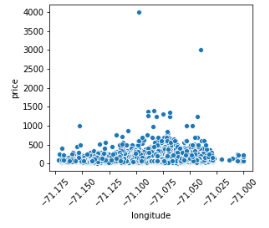
**Boston Nominal EDA**





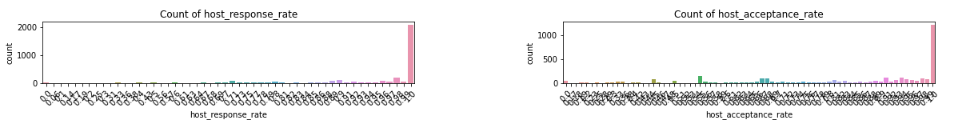






**Boston Categorical EDA**

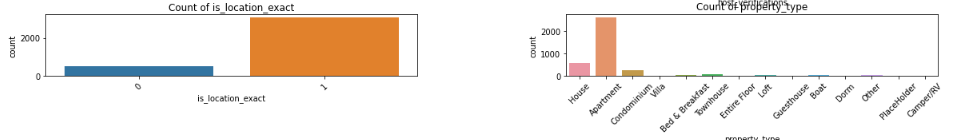
Host response is largely skewed to be 100% meaning hosts typically respond as we ll as count of host\_acceptance\_rate meaning they respond and typically will admit people. There are humps in Count of host\_acceptance\_rate meaning maybe those locations that deny people are on the higher ends or are places that are busy and have a high turn down rate because of events or such.



Count of host\_verifications is a list of lists, it was difficult to represent the x\_label because of this. but is very skewed right showing more obscure amenities that a location might have



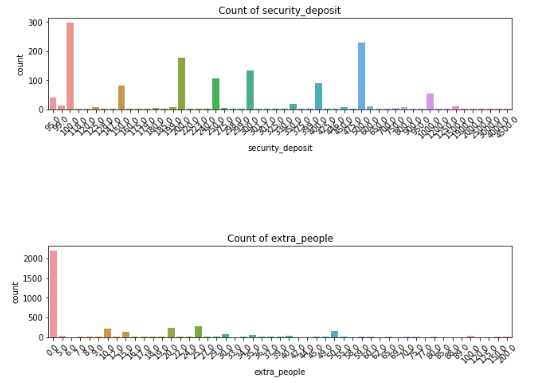
The location being exact is a good sign for guests, and most properties lie between a house, apartment, and condominium.



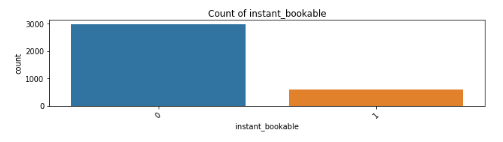
For the given property type, an entire property type would be for rent, second being the private room, and last being shared room; we can expect the pricing be extraordinarily different for each of these. It’s nice to say that people renting an Airbnb get a Real bed.



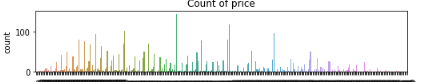
Count of Security Deposit shows that mode deposits are 95-100$ range with the next mode being $500 which could be an industry standard. Most airbnb listings don’t particularly care about the count of extra people, but if there are, there is a payment of maybe 20$.



Instant\_bookable skewed to be that most places are not, and you need to reserve it beforehand.

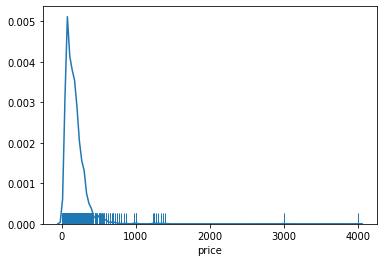


Count of price essentially is a worst histogram, but you can see the distribution a little better, it ranges from 0 to $4000 for a day of rent.



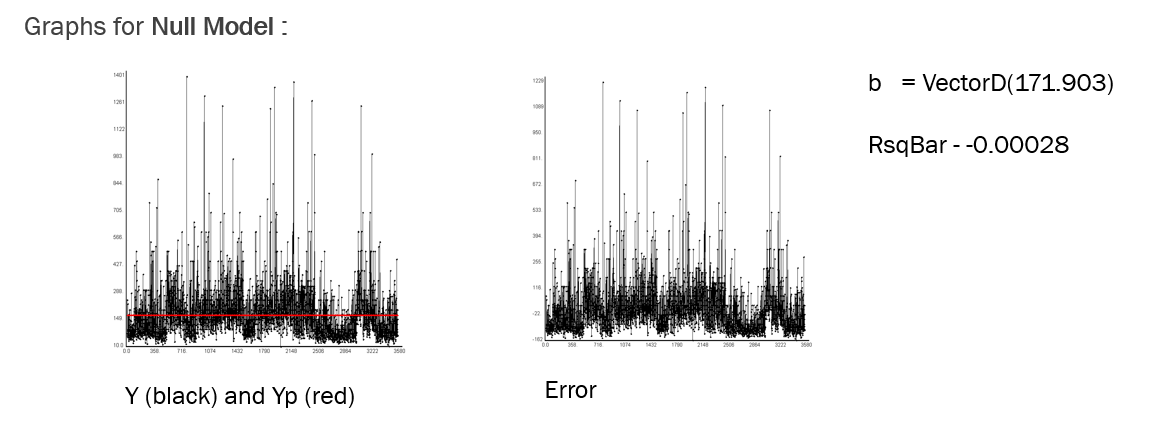
**Boston Histogram of Price (yp).**

While our response is skewed to the right, if we removed these values we would have ~500+ rows removed from our 3500+ row data set which is a substantial amount in terms of testing so we are keeping these rows in.



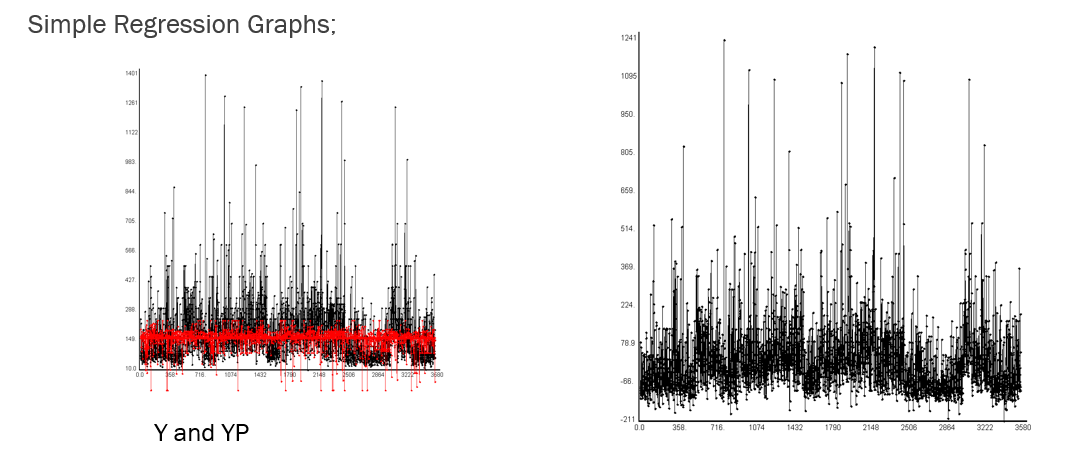
**Boston Dataset Results**

**Null Model:**



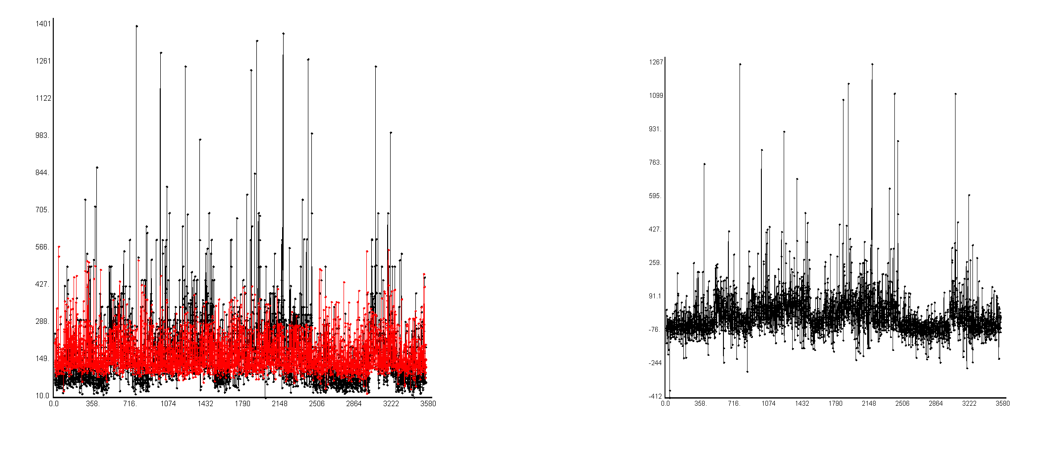
**Simple Regression:**

rSq -> -0.04557, rBarSq -> -0.07624



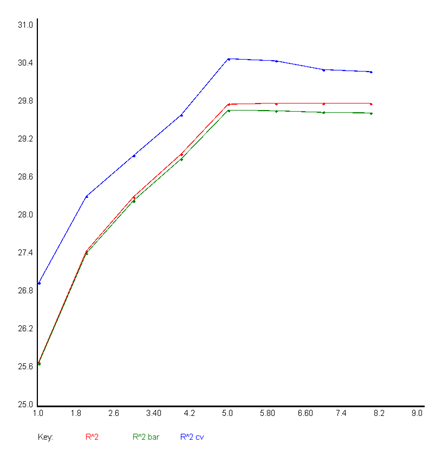
**Multiple Regression:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modelling Technique** | **RSQ** | **N\* for RSQ** | **AdjustedRSQ** | **N \* for RSQbar** | **RSQCV** | **N\* for rsqcv** |
| **Multiple Regression** | 29.7891 | 8 | 29.67529 | 5 | 30.48686 | 5 |

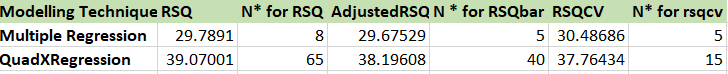


Y an Y predicted Error

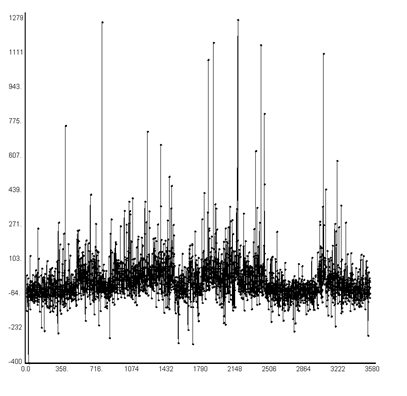
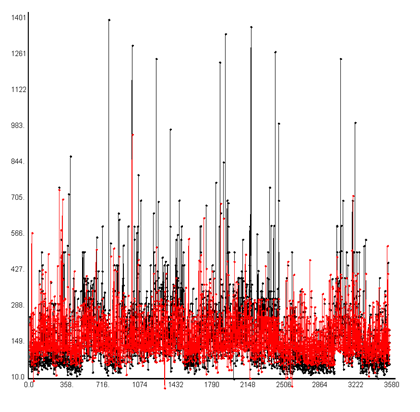
Graph of R squared values:



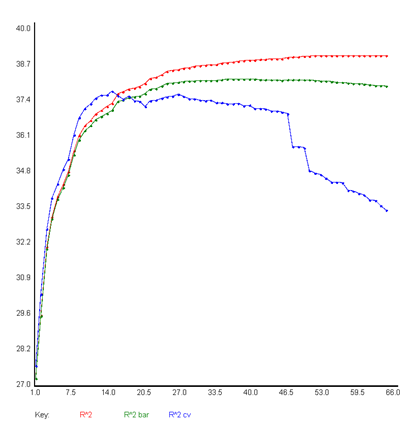
**QuadXRegression** :



Y and Y Predicted Error

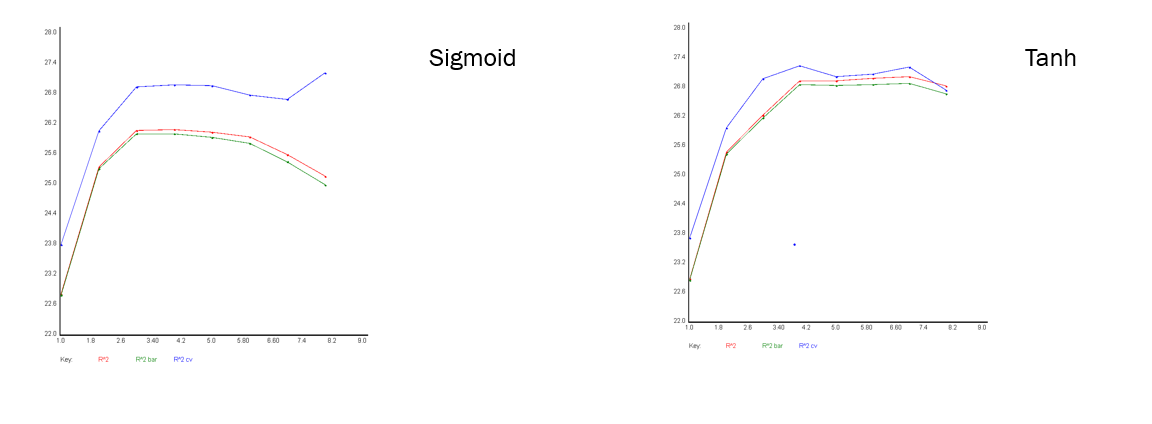


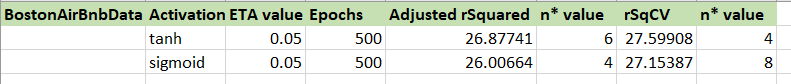
Graph of R Squared values :



**Boston Perceptron:**

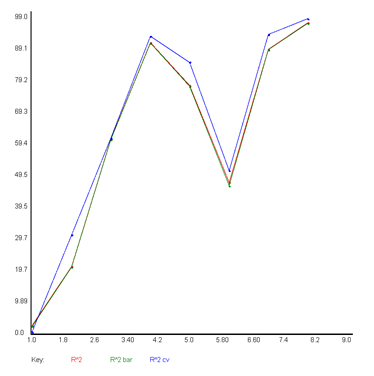
**R squared graph for perceptron**



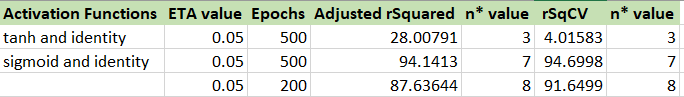


**Boston NNXL Scala:**

**RSquared graph**



Optimizer used : SGD



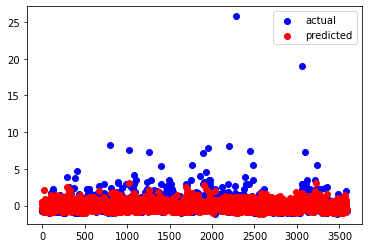
**Boston NNXL Python Best feature (Sigmoid/identity):**

We do not have the rBar graphs here because running the forward select method in python takes a very long time. However, the code to run and get the graphs are included within Boston.ipynb. The y and yp graph were made using the features that Scalation had selected as well as the R values.

R^2 -> .271

Radj -> .269

RCV -> .2754



**Boston Interpretations:**

The Boston dataset had a large amount of features, some of which were difficult to plot and glance over because of the categorical nature even with some of the numerical features. The scatter plots showed that several of the plots had a trend, while some were random variables so we used forward selection to weed the ones with least importance out of our model. To our surprise, on average 5-10 features were only selected (except for quadX where 40 were). QuadX provided the most stable output with having most variation explained by the model being at .3. While NNXL shows a HUGE promise with the R^2 being .91, we found this hard to believe and have a comparison values that were coded in keras (which are not at .99). We could not find the bug that would cause the values to be so high in Scalation.

**Preprocessing(Seattle Dataset):**

All preprocessing was done in python, please refer to the colab: Seattle.ipynb which was submitted for a more verbose representation of the features.

Seattle dataset initially had 17 columns, but were quickly reduced to 9 after glancing over the excel and removing:

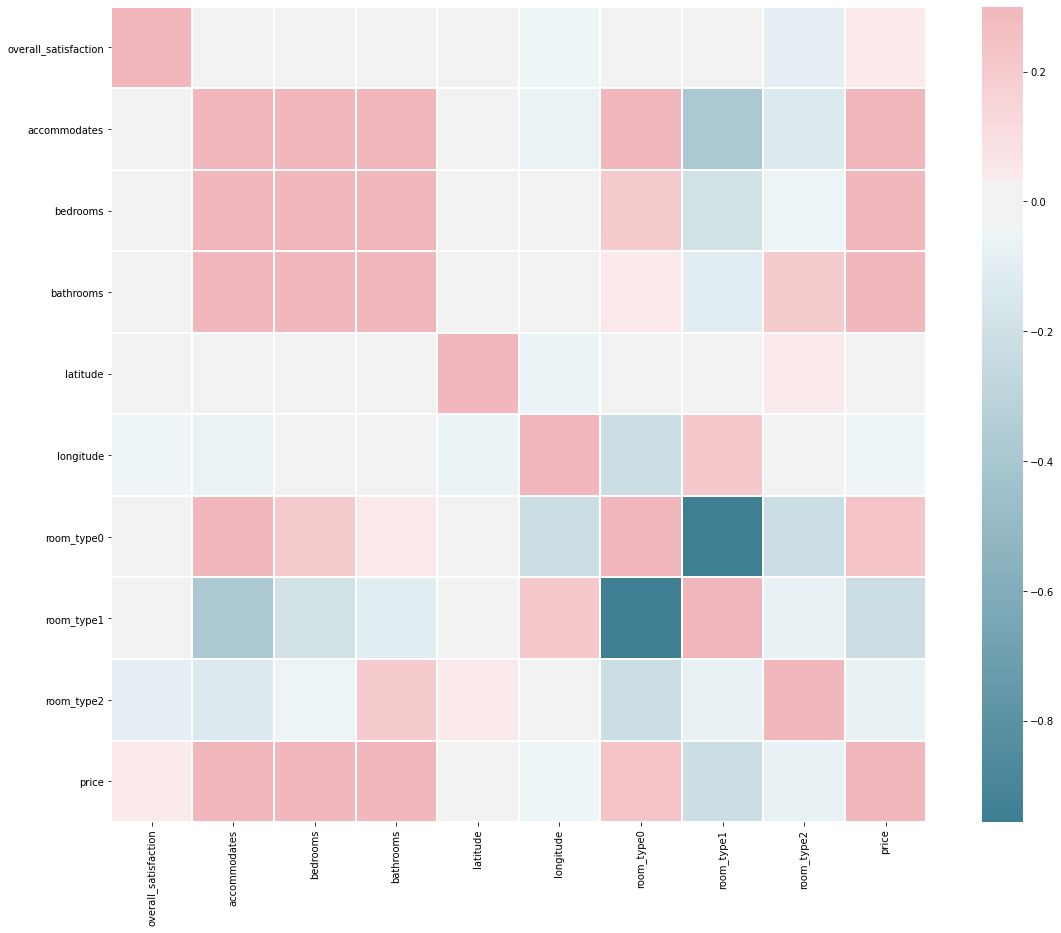
* Features that were non numeric and non-categorical, for eg: Name have been removed and ID.

Data Cleaning and modification included:

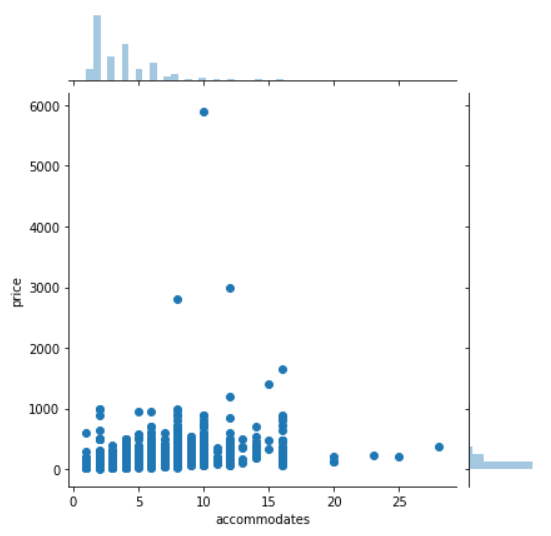
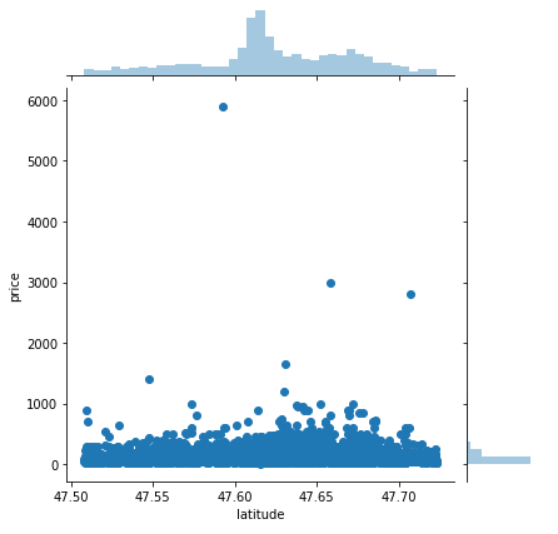
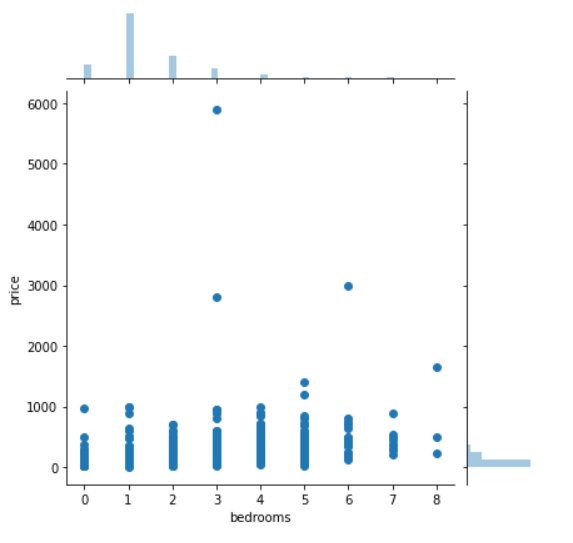
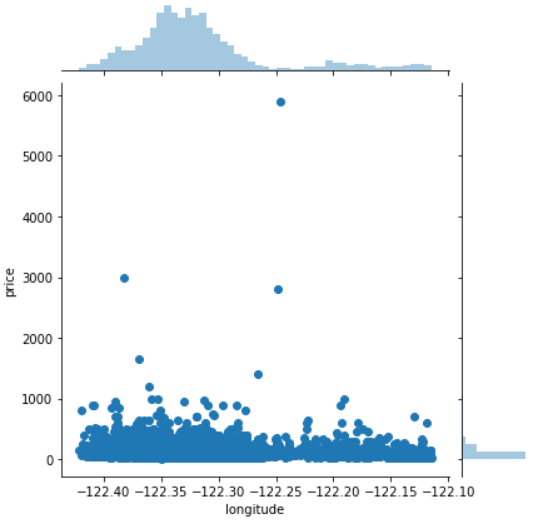
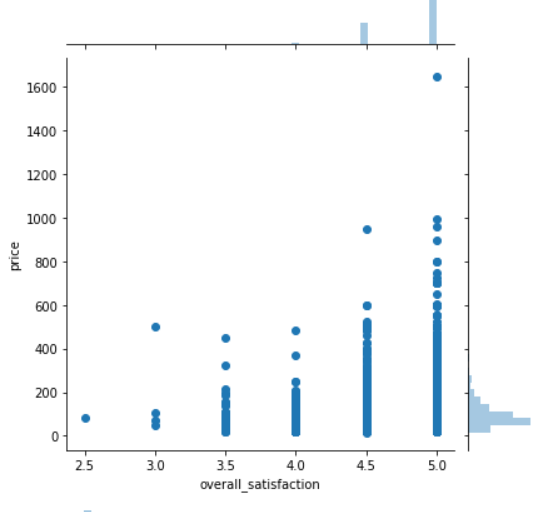
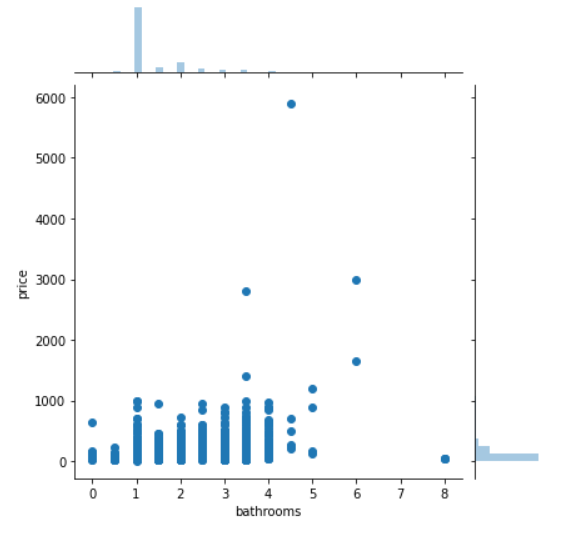
* Feature Room\_Type to be label encoded then one hot encoded
* The NAN values were replaced by mean

**Seattle Heatmap**

Same relation that was going on with Boston with Accommodates, # Bathrooms and bedrooms creates a stronger correlation than the rest. This specific correlation graph has the rooms one hot encoded because it could be represented on this heatmap on not the Seattle one. Without the one-hot encoding, the correlation of bedroom, bath, accommodates, and price shoot up to .3.

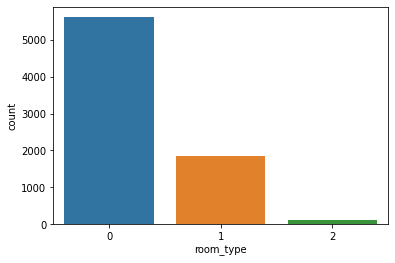


**Seattle Nominal EDA:**



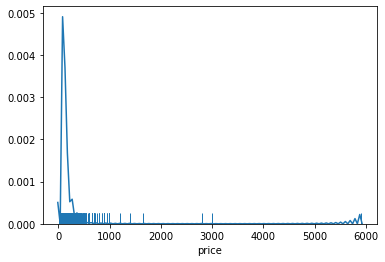
**Seattle Categorical EDA:**

0 = Entire Room/apt, 1 = private room, 2=shared just like Boston dataset and they both have very similar distribution.



**Seattle Histogram (yp)**

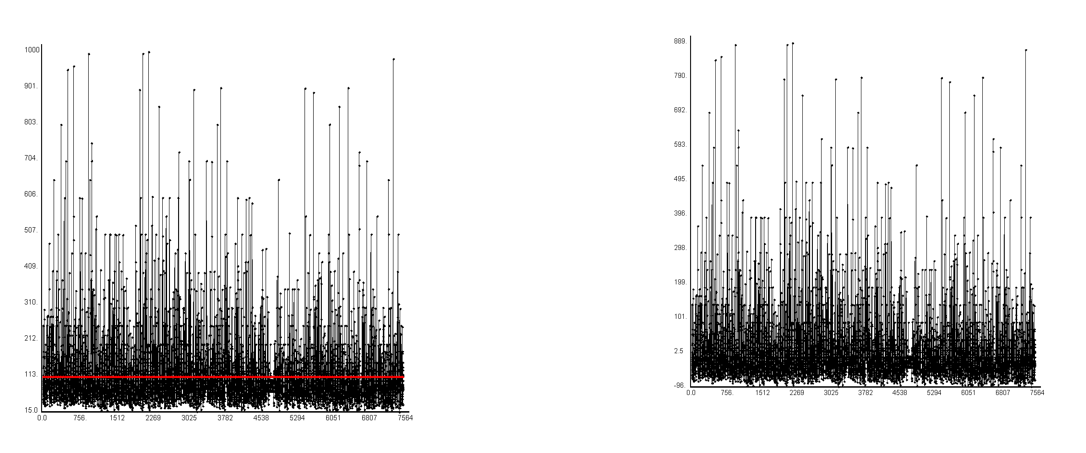
**Very similar shape to Boston’s. Skewed far right and we are keeping the data Boston data past the 3rd quartile because otherwise we would be losing 2000 instances.**



**Seattle Dataset Results**

**Null Model Graphs**

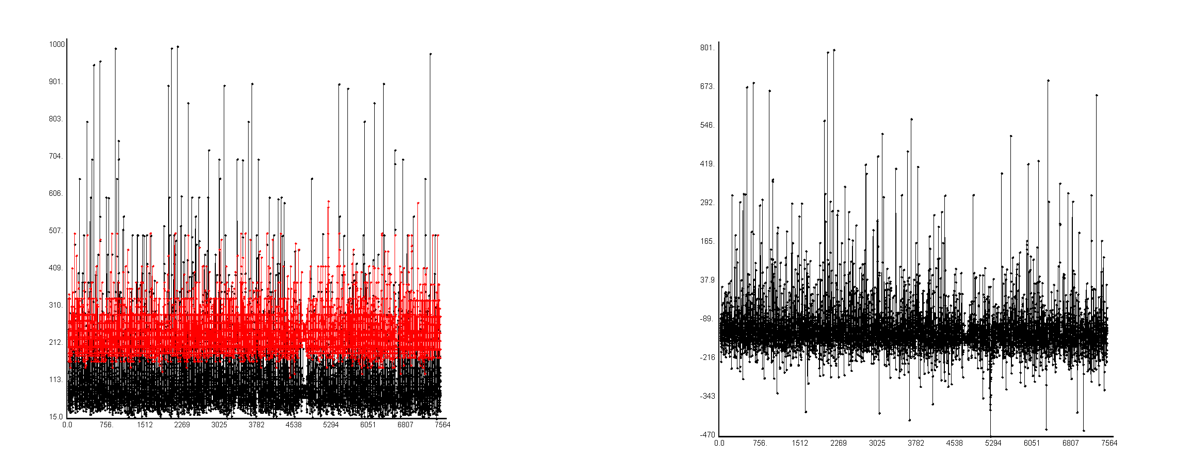
**rSq -> 0.00000, rBarSq -> -0.00013**



Y and Y Predicted Error

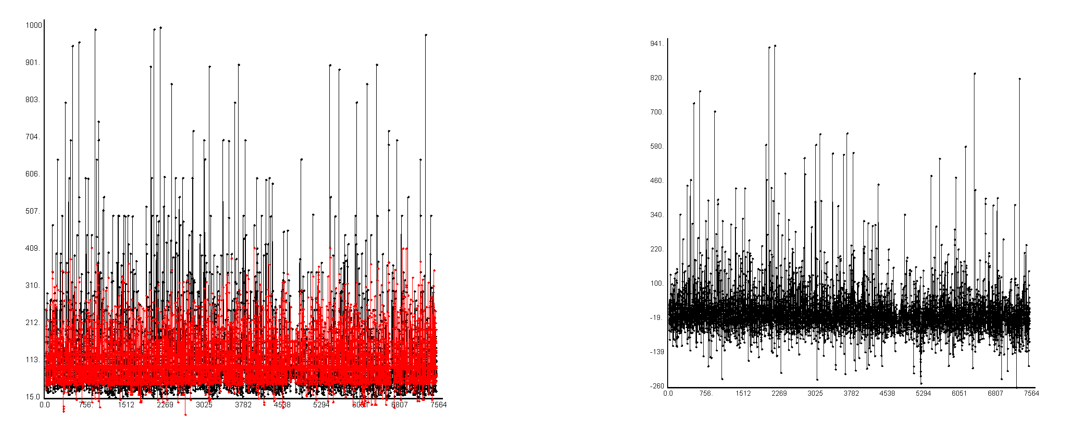
**Simple Regression**

**rSq -> -1.64217, rBarSq -> -1.64427**



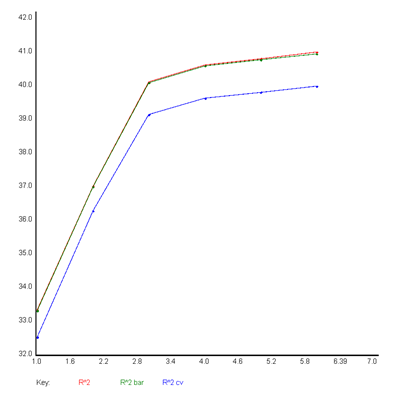
Y and Y predicted Error

**Multiple Regression**



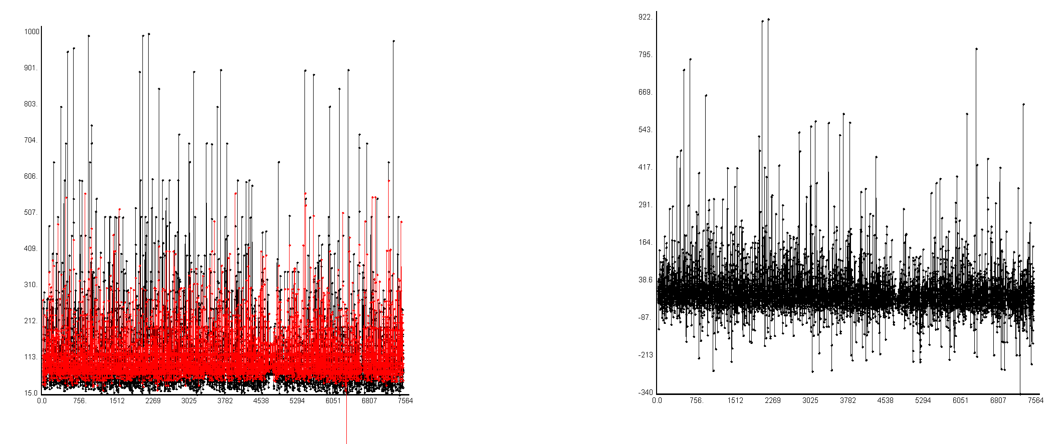
Y and Y predicted Error

**Graph for R squared values:**





**QuadXRegression Graphs**



Y and Y predicted Error

**NNXL Graphs Python**

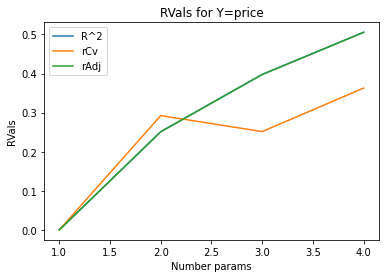
**Graph for R^2 values NNXL of Best Combo (S):**

N\* = 4 ('overall\_satisfaction', 'bedrooms', 'bathrooms', 'accommodates')

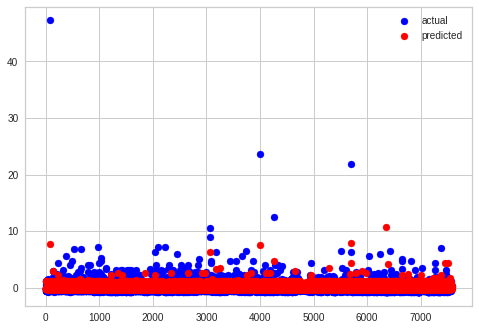
R^2-> .507

Radj -> .505

RCV -> .362



**Y and Y\_p of NNXL**



**Interpretations**

The Seattle dataset had a larger number of instances than the Boston data set (4000+), so the better R Values make sense, even if there were less attributes to choose from.

**Conclusion**

Our results find that with the provided datasets, we can explain the most variation within the Seattle Dataset through the four attributes 'overall\_satisfaction', 'bedrooms', 'bathrooms', 'accommodates' using the NNXL. With this we can somewhat predict price pf a house per day. When compared to the Boston data set's NNXL, our RValues are .20 above within the Seattle's model which can be tributed to 4000+ instances. This can be determined because Seattle's data set is essentially a subset of Boston's. This alone means it may be worthwhile for us to collect more information about Seattle to gain a substantial comparison. But with the information we had, we can conclude that overall\_satisfaction', 'bedrooms', 'bathrooms', 'accommodates' provides significant value to predicting price per day, because these features were also selected within the Boston Dataset. Boston's model's also included minumun\_nights. guests\_included, review\_score\_ratings, host\_response\_rate, and host\_acceptance\_rate which can be key components to helping airbnb hosts to find their perfect price listings.

**Improvements**

For preprocessing, there could have been a better way to one hot encode rather than replacing with 1 and 0 knowing we would be using QuadX in our analysis.

We could use a Natural Language Processer to convert several of the object ‘string’ columns that we had to drop to become an indicator of people’s reactions and feelings towards a certain Airbnb listing.

Season might have something to do with the pricing as people travel at different times, so converting this to a time series predictor would require an inner join with another .csv file and different ML models to be implemented but may be worth the consideration.