Airbnb Pricing Prediction and Optimization
Project Report - 4/26/2018
Group 9

Relevant People & Report Objective

Team Members:
Sankarshan Acharya
Paul de Fusco

Advisor(s):
Julian McAuley

Objective
This report will contain information about the challenges and results of Step 7 and the contributions due to this step

Contributions

Modeling (Including Data Preparation)
- Sankarshan Acharya - Regressions
- Paul de Fusco - Text: Feature Creation, Clustering, Latent Dirichlet Allocation

Visualization
- Paul de Fusco

Modeling

Analytic Approach

The analytic approach can be summarized as an iterative process comprising feature selection and model implementation/evaluation.

This week we continued with the above approach while also including new features i.e. columns we had created based on data wrangling of the amenities feature and columns we created from the following text features: 'space', 'description', 'neighborhood_overview', 'notes', 'transit', 'access', 'interaction', 'house_rules'.

For each, we had created a set of methods that give counts of words, punctuation, etc. This time we added methods that do the following:
- Tag and count grammar tokens and create features accordingly
- Perform Latent Dirichelet Allocation (NLP) to label text features with one of ten topics. Additionally, create numeric features reflecting the level of belonging to the top topic assigned.
- Perform clustering on text and labeled each text feature into one of ten clusters.

As a result, our dataset now includes 424 columns while last week it included 90.

For feature selection, we had ranked features according to their contribution to R2. While we have still used that approach, we have also performed Lasso and Ridge Regressions to select features. The process has resulted in a reduction of the dataset to 110 columns.

Model Description

The type of learning that we are doing for the modeling is all supervised learning. We are training a set of predictors (the inputs given) with a target that we created for this purpose (averaged price).

All the creation of training sets and test sets are doing with the function train_test_split in scikit-learn in python. From this, we created a training set, a validation set, and a test set to assess the functionality of the model. The scoring consists of predicting the price of the listing given the inputs that are pre-created and fed in.

For our purposes, we have mostly used linear regression and quadratic regression. We are also currently running tests with neural_networks and ensemble methods (regression) to optimize the model as well. All of these methods listed are parametrized models. The only difference between neural_networks and other three is that neural networks have (non-linear) activation functions which act on the parameters to non-trivially optimize them as you go deeper into the network.

Linear regression works by taking a list of features with the data values associated with them and training them with a target variable to come up with a model that is completely first degree with respect to the fields, linear with respect to the parameters and accounting for bias by means of the y-intercept.

Quadratic regression works in a similar manner to linear regression as it is a polynomial model with there being slight differences. The most significant one is that there are strong interaction terms to account for when doing polynomial regression because these form the backbone of higher-order polynomial regression.

And finally, we began to explore modeling with neural networks. Neural networks work by, in general, having multiple layers to do computation on the data. Each layer first works by taking in the values of the previous data, multiplying a weight matrix to the data, adding a bias term to that data. We can also, in the same layer, put that previously computed value through a non-linear activation function to arrive at an even more non-trivial result.

And to top it off, for the linear regression we tried to optimize the model by looking into regularization. Specifically, we tried using Lasso (L1) and Ridge Regression (L2) for feature removal and anti-overfitting measures to see if there was a benefit to that.

Model Performance

For this report, we worked on many models for the purpose of this project. These models all had some kind of difference. They could be different by how many features were there, what features were there,

whether or not we used regularization in the model, etc. We attempt to list out some of the most important iterations of modeling in this report so as to emphasize the most important findings that happened this week.

Below are some of the most important statistics from our report because we cannot list them all. The results for all the models can be found in our project repository in the notebooks regressions_two through regressions_six.

Model 1

Holidays

Linear Model	*Quadratic Model*
R^2 for validation is 0.659122179548 MAE for validation is 62.1892286849 R^2 is -0.448279870262 MAE is 66.2050991985	R^2 for validation is 0.290799197092 MAE for validation is 78.0034678952 R^2 for is -0.000627194665553 MAE is 16714.6570329

Weekdays

Linear Model	*Quadratic Model*
R^2 for validation is 0.578179812619 MAE for validation is 67.3657262885 R^2 is 0.60095715516 MAE is 66.6547857159	R^2 for validation is 0.383522750873 MAE for validation is 78.3227111599 R^2 for is 0.448714712015 MAE is 69.989243112

This model iteration served as our base from which we built our work and worked to continually correct.

Model 2

Weekdays

Linear Model	*Quadratic Model*
R^2 for validation is 0.591220217989 MAE for validation is 66.5884535025 R^2 is 0.608245442315 MAE is 66.6659432419	R^2 for validation is -1.70205974208e-05 MAE for validation is 536657086394.0 R^2 for is -0.000674362271768 MAE is 533313123650.0

Model 3

Weekdays

Linear Model	*Quadratic Model*
R^2 for validation is 0.607959811208 MAE for validation is 66.427615843 R^2 is 0.640284864283 MAE is 65.821378261	R^2 for validation is 0.0259912623834 MAE for validation is 381.815474329 R^2 for is -0.000272081381259 MAE is 115709905216.0

Model 4

Holidays

Linear Model	*Quadratic Model*
R^2 for validation is 0.664258811108 MAE for validation is 63.6454962801 R^2 is 0.625398818979 MAE is 59.9938154053	R^2 for validation is 0.330959098884 MAE for validation is 107.613344245 R^2 for is 0.282177722496 MAE is 125.324642221

Weekdays

Linear Model	*Quadratic Model*
R^2 for validation is 0.576248397064 MAE for validation is 71.5033793243 R^2 is 0.617307937621 MAE is 66.1873838863	R^2 for validation is 0.181784716548 MAE for validation is 123.286425122 R^2 for is 0.176095000949 MAE is 146.53126753

Model 5

Holidays

Linear Model
R^2 for validation is 0.713408608734 MAE for validation is 61.3380258186 R^2 is 0.69986984258 MAE is 61.9210474179

Weekdays

Linear Model
R^2 for validation is 0.635215022373 MAE for validation is 70.7960988664 R^2 is 0.68157070758 MAE is 67.5007212331

Model 12

Whole dataset

Linear Model
R^2 for validation is 0.589810898218 MAE for validation is 71.6424214816 R^2 is 0.618834621915 MAE is 64.4209759928

Model 13

Whole dataset

Linear Model with L2 Regularization (Ridge Regression)
R^2 for validation is 0.596022273409 MAE for validation is 70.8508026488 R^2 is 0.618351043053 MAE is 64.2045590595

LDA Model for Listing Descriptions

20 topics with learning decay of 0.7
Log Likelihood: -3209523.9166136212 Perplexity: 850.69712423150474

K Means Clustering Model for Listing Descriptions

init='k-means++', max_iter=100
Inertia: 5211.983300342347

Evaluation tools used to interpret result given as part of the model (names and brief description of the tool)

For this report, we have two main evaluation metrics for our models: R^2 (which is the coefficient of determination) and MAE (mean absolute error, the average of the absolute errors of the predicted and the actual target)

Interpretations of the results as given above

Some insight derived from the work given:

- Using NLP, or more specifically LDA, to create features for prediction gives you many feature to work with to allow for fine-tuning of a model (Model 4). This is especially true in the case of the presence of amenities and the topics of the listing description
- Use of transformations on the data is very useful when it comes to increasing R^2, however one needs to understand what the nature of the predictor is to effectively use it (Model 5)
- Interactions are useful for data model construction even in linear regression models (Model 3). It allows us to account for any
- Regularization is a useful technique and both Lasso and Ridge regression should be tested to see which is the most effective (Models 8-13). This is due to the strong collinearity in the field given in the data.
- Ensemble methods are a great tool for the purpose of predicting price especially when your models requires you to have a smaller MAE.

Conclusions

From this iteration, we acquired many variables from which we can create iterations for predictive models. Most of the features (the topic features) we acquired from the string based variables from which LDA was done to get some information on which to predict price.

We also acquired relevant techniques and tools as part of this step. The first of which is transforming the target with the logarithm and predicting the log of the target. This is a useful tool when predicting targets that are necessarily positive. And then, we acquired ensemble methods like Gradient Boosted Regression which works by using bagging to minimize error.

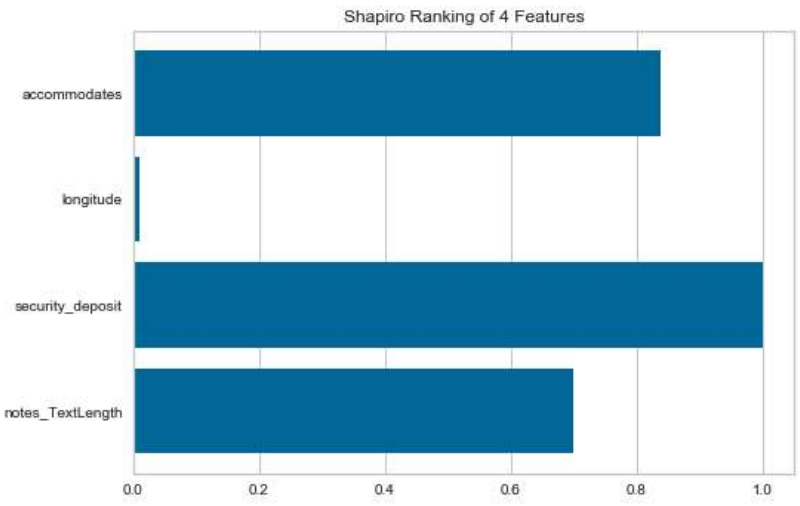
Most, if not all, of the listed techniques/tools listed will be used in the further three steps of this project.

Updates to the preceding 6 steps

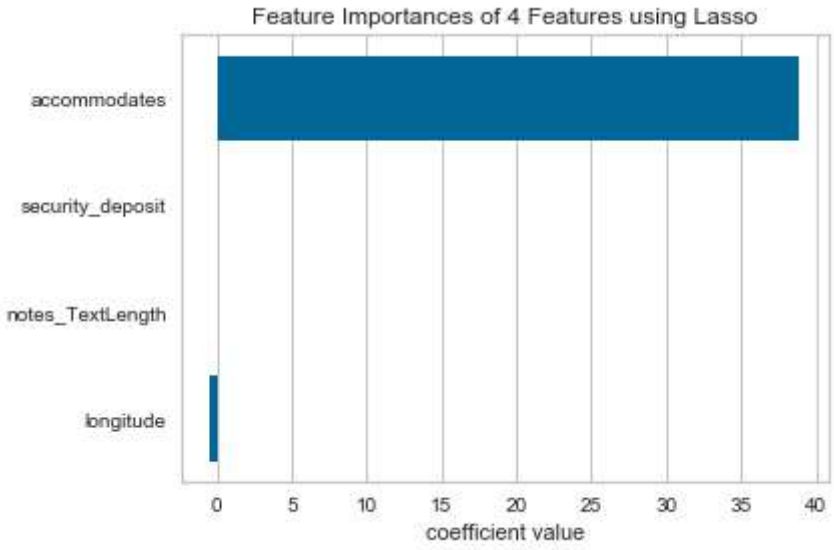
Due to this step, we have also made important updates to steps 3 and 6 at least, if not others. Contributions to step 6 are obvious in that it adds more modeling iterations to the ones that already occurred. And the contributions to step 3 were that it gave features on which we could do more exploration to find more patterns to make prediction of price easier and more efficient.

Visualizations

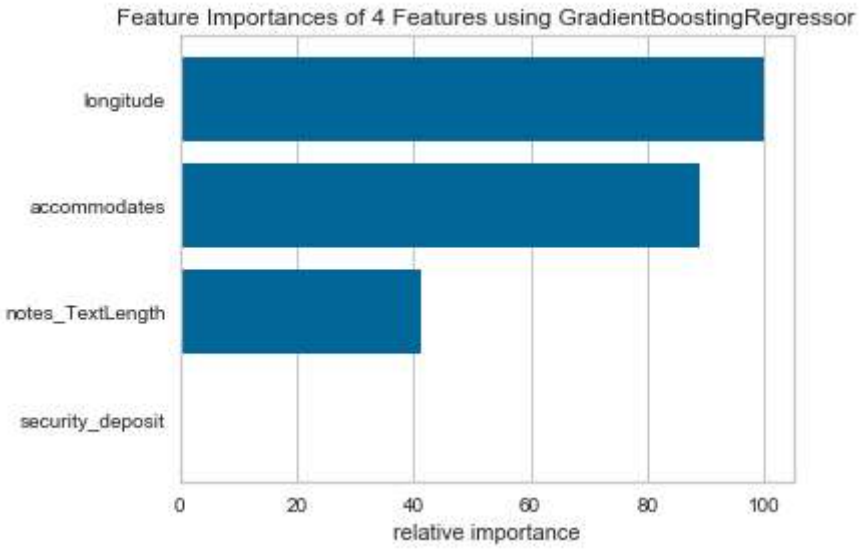
Shapiro Feature Ranking:



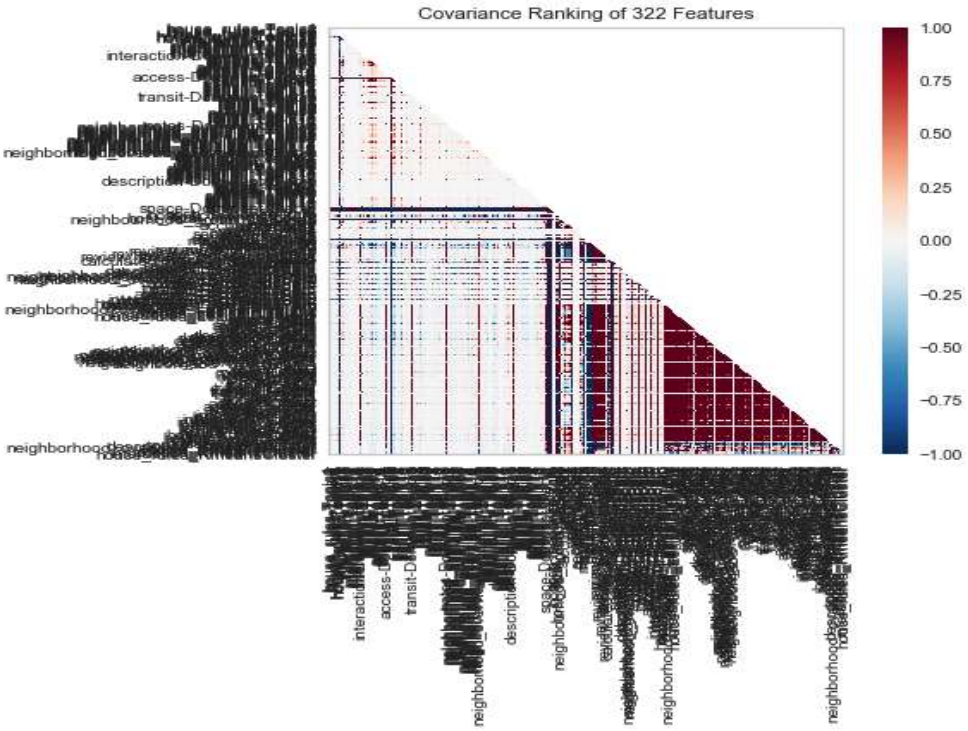
Feature Importance with Lasso



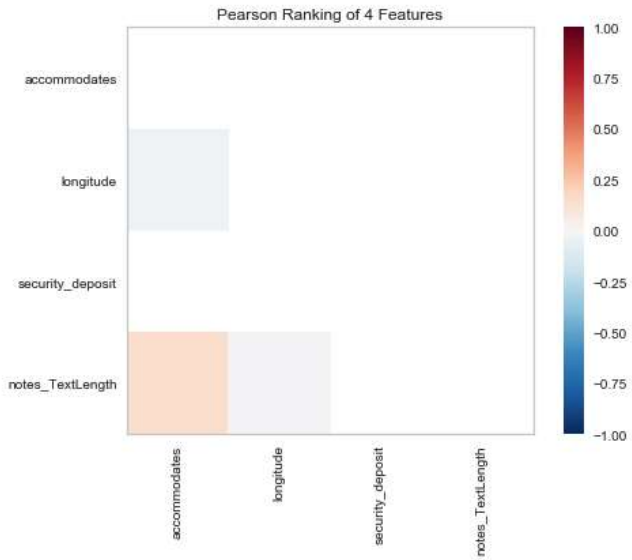
Feature Importance with GradientBoostRegressor



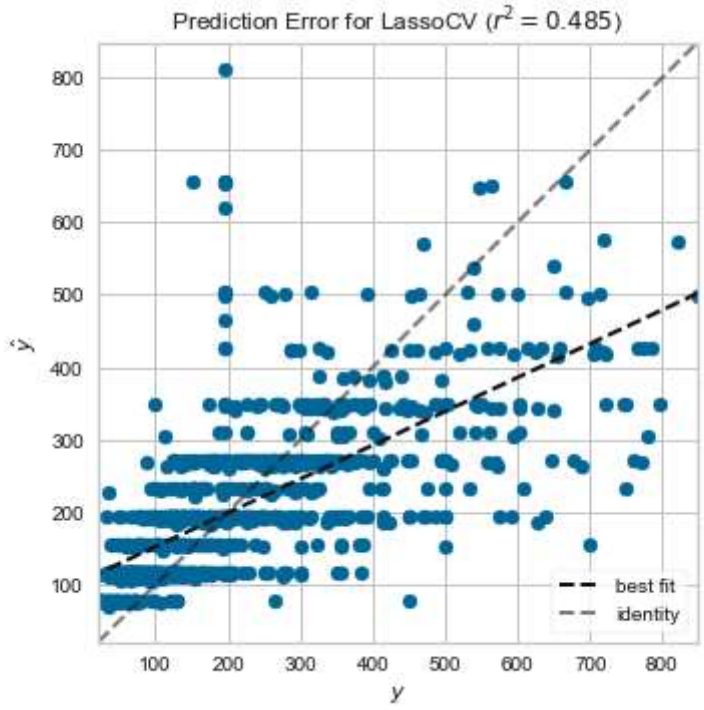
Feature Ranking with Covariance



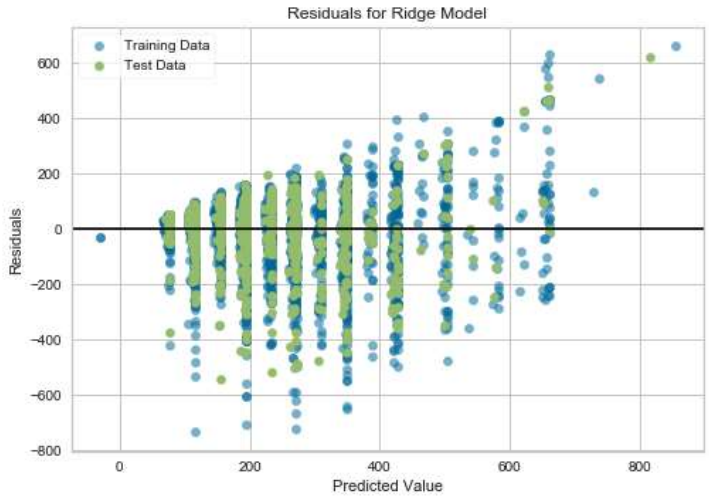
Feature Ranking with Pearson Correlation



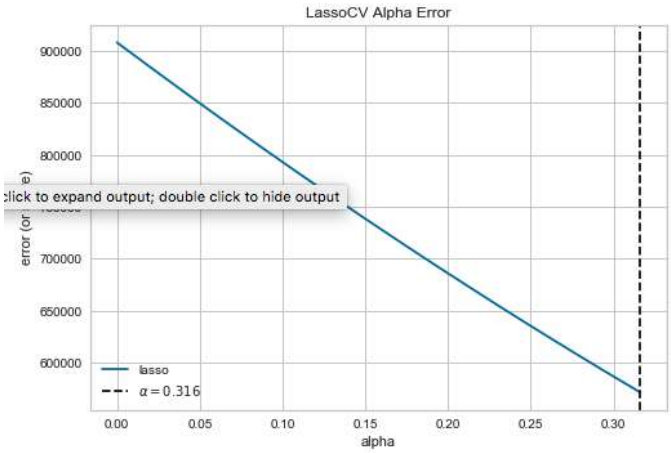
Visualization of Lasso Regression Residuals



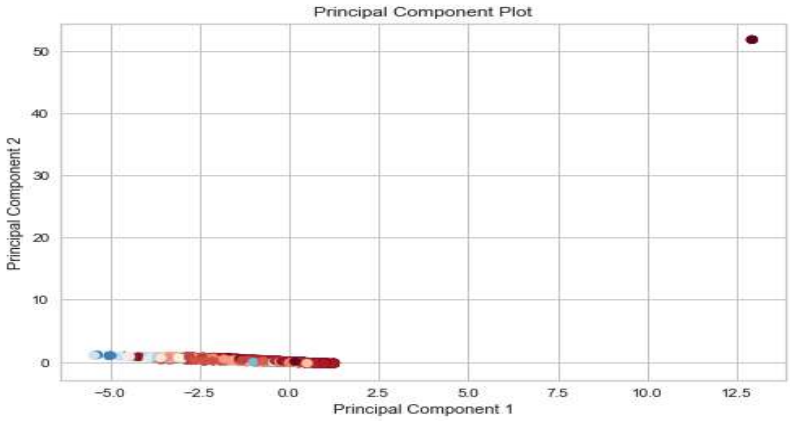
Visualization of Ridge Residuals

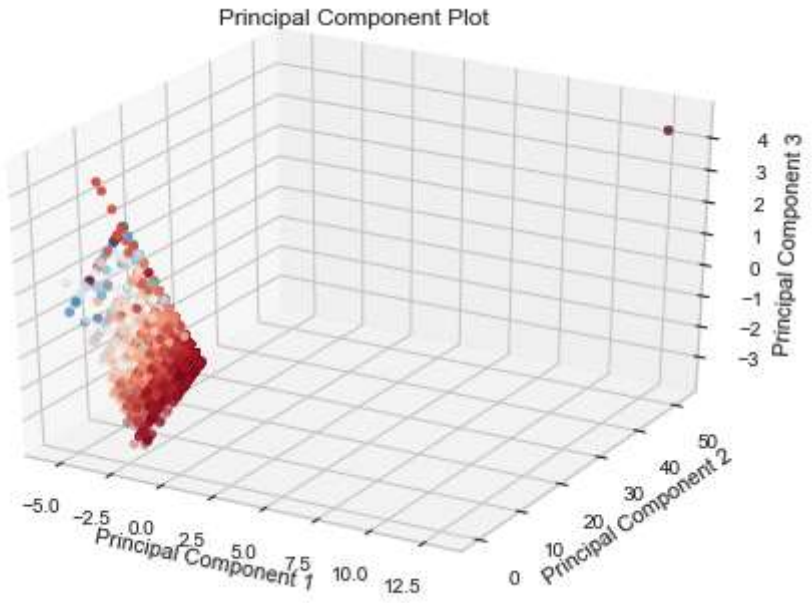


Visualization of Lasso Alpha Error

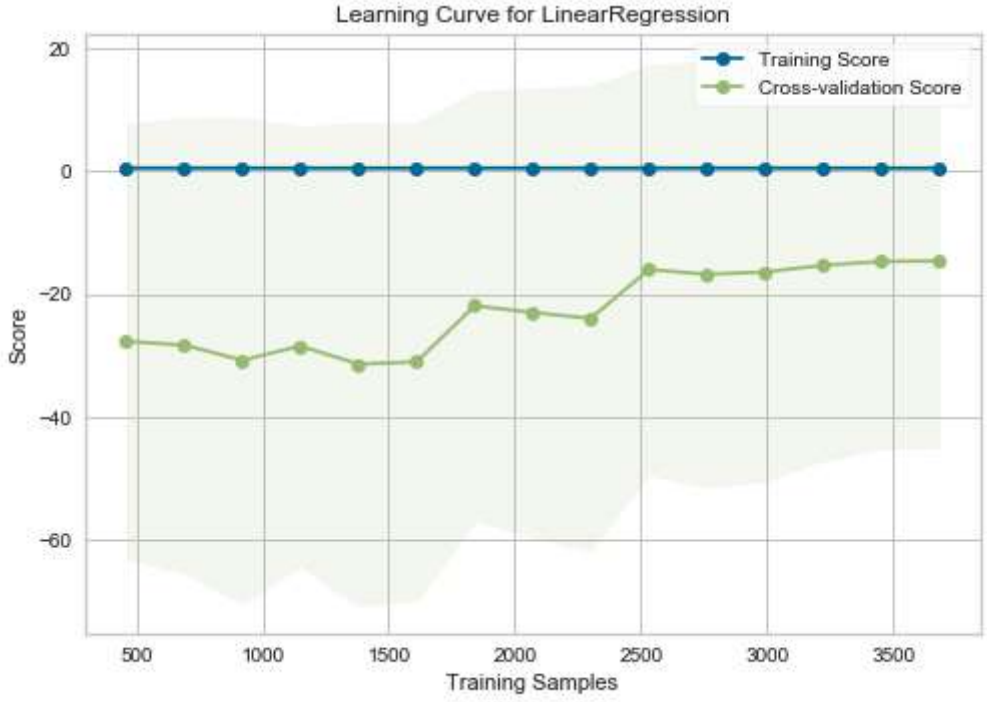


PCA Visualizations - 2D and 3D

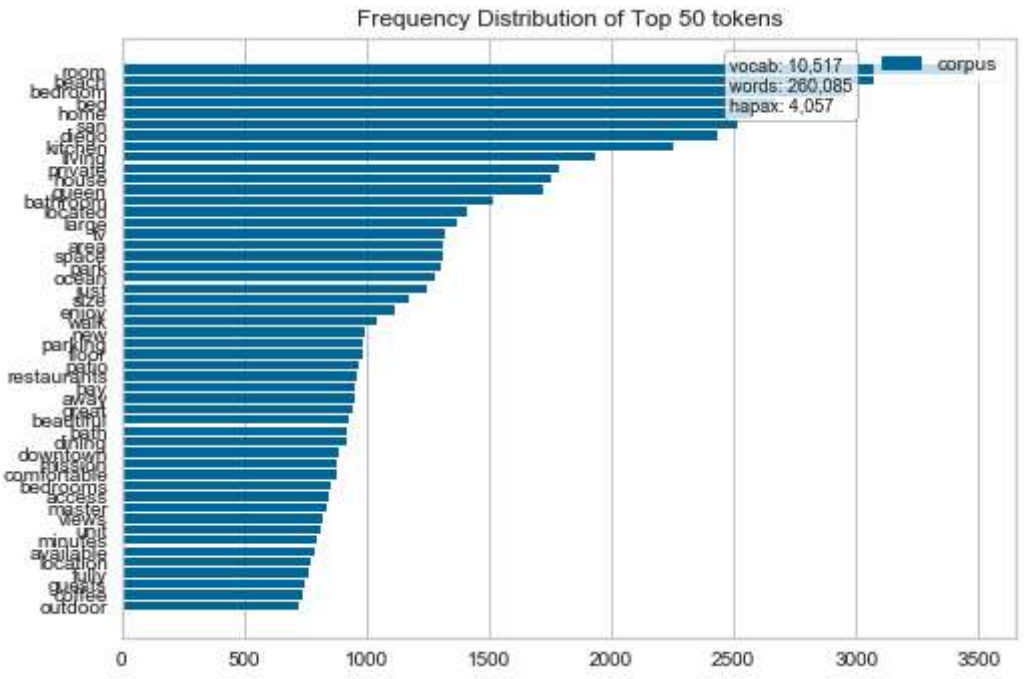




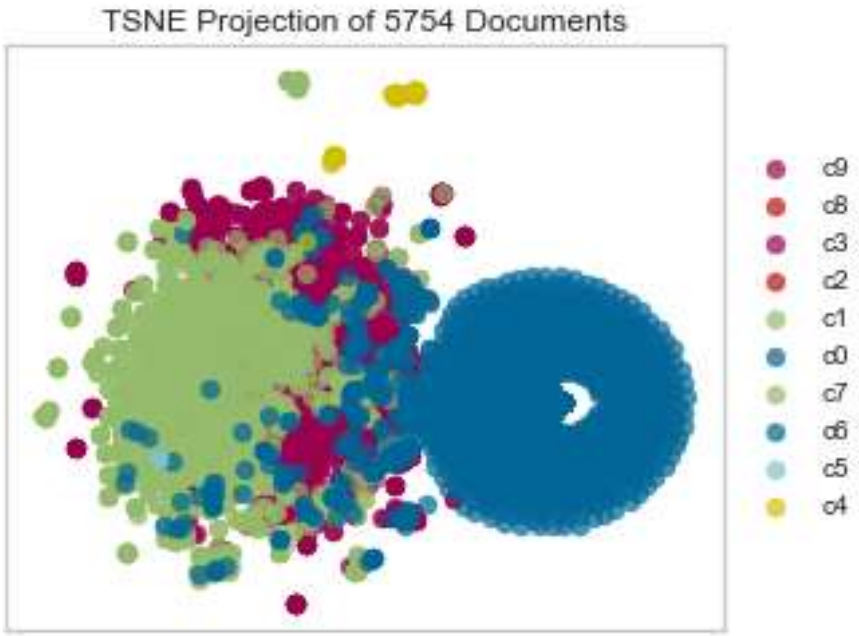
Cross Validation Learning Curve Visualization



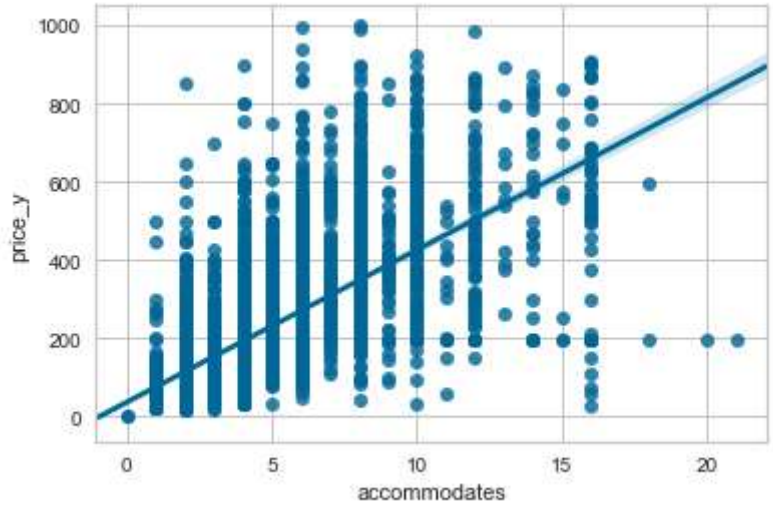
Top 50 Frequency Distribution for Tokens in feature ‘space’

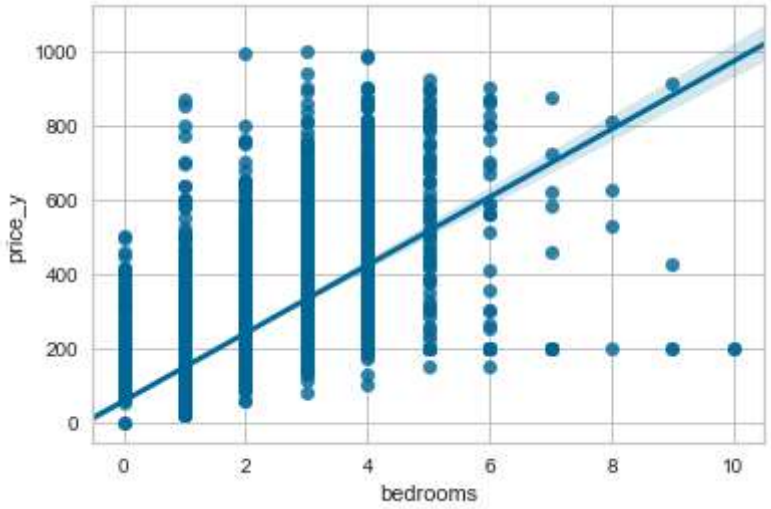


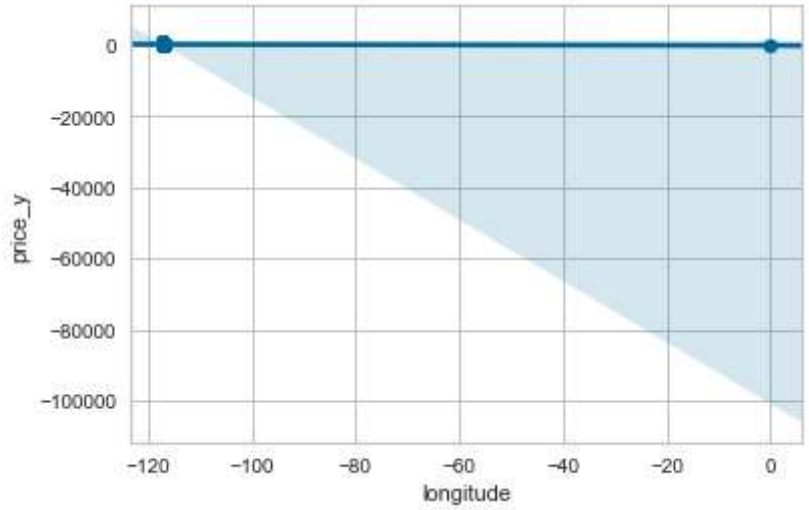
Visualization of the Kmeans Model (n=10) on Transit Column



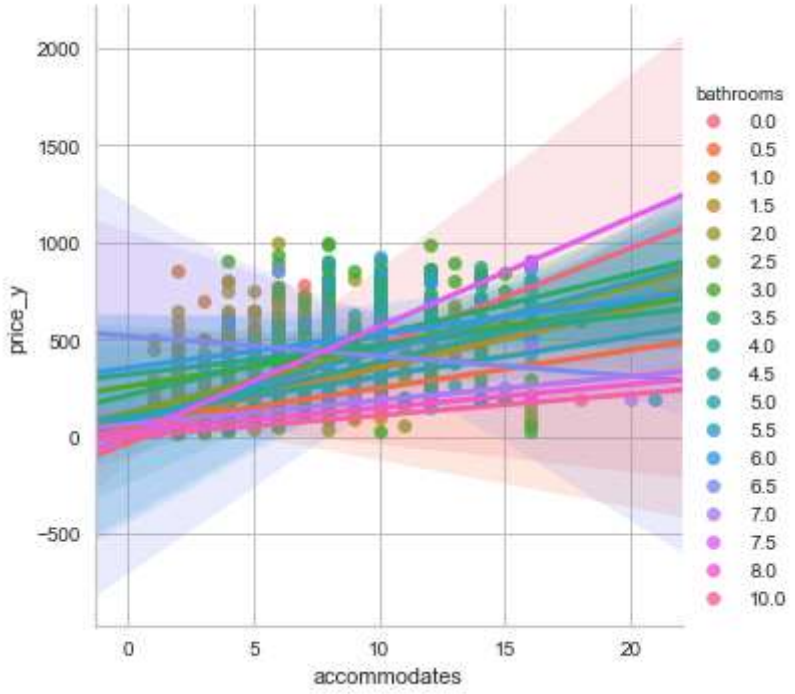
Visualizations of the top 3 R2 Variables with respect to price - Individually

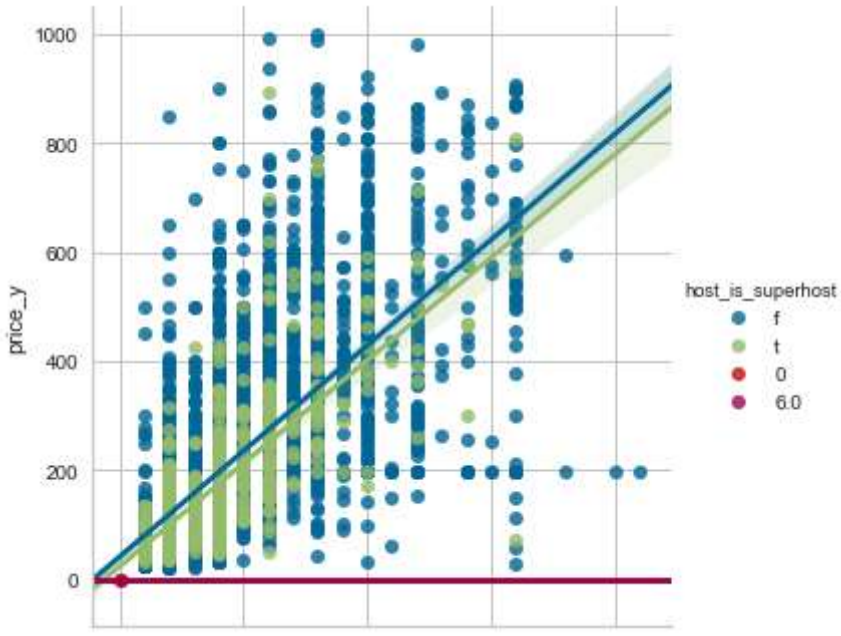

price_y
accommodates
0
200
400
600
800
1000
0
5
10
15
20


price_y
bedrooms
0
200
400
600
800
1000
0
2
4
6
8
10

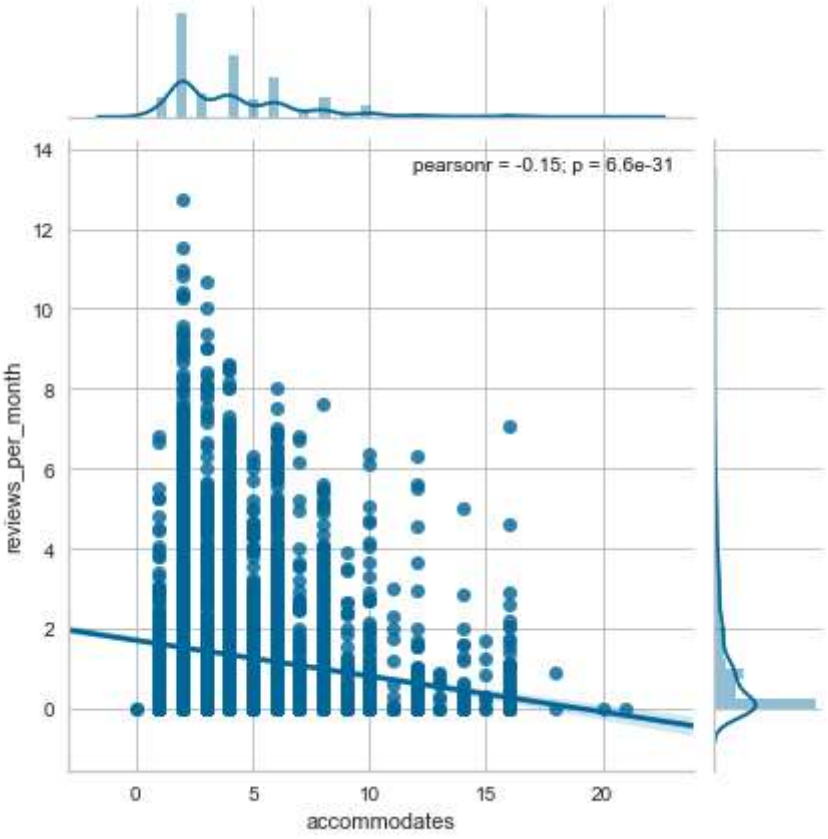

price_y
longitude
0
−20000
−40000
−60000
−80000
−100000
−120
−100
−80
−60
−40
−20
0

Visualization of the Influence of 3rd variables on relationship between independent and dependent variables





Visualization of relationship between two independent variables with relative frequency counts



Next Steps:

- **Modeling:**
  - Evaluate the possibility to remove outliers and rescale the target variable.
  - Manually edit models as a result of considering bias, outliers, multicollinearity, feature independence and correlations.
  - Test alternative regression techniques other than OLS and SGD (lower priority).
  - Expand on current feature selection methods and test more approaches (PCA?)
  - Optimize text models with GridsearchCV and create more features with LSA.
  - Explore models to score text richness in more advanced ways.
  - Perform similar and other text analysis on the reviews dataset and use results to augment the listings dataset.

- **Visualizations:**
    - Continue to build visualizations reflecting the above efforts
    - Do a visualization of the LDA with Gensim library to show topic distributions