New York City Rent Price

Rosanna Luo – BrainStation

Executive Summary

Problem Statement

Being a landlord in New York City can be lucrative, but it can also be a lot of work. One huge decision that remains an issue for even savvy landlords is how much rent to charge. Charging under market prices may cost thousands or tens of thousands of dollars a year per unit, while charging over market prices may mean your unit is empty for months.

We seek to use machine learning to train a model based on real New York City rental data that can calculate what a good rental price for a given rental property. This can help landlords navigate the many factors involved in deciding how much to rent their unit.

This work is done for the BrainStation Capstone Project by Rosanna Luo

The Data

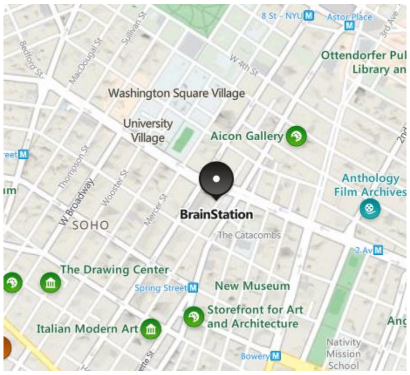
The primary source of data is derived from RentHop, a site that has rental listings. They shared a large set of rental property data from 2016 in a Kaggle competition. We seek to clean and then augment this data. Some fields of primary interest from the data include Description and Longitude and Latitude.

Description

The description is the listing information for a rental unit. This provides a free form text field that describes the rental properties. There is a lot of data in this field but it is also quite dirty. We can also explore this data to find out what qualities are useful in a marketing description for landlords.

Longitude and Latitude

Location is a primary factor in rental prices in NYC. We can use the longitude and latitude of a rental listing to derive further information about the location of a property. We will augment with the zip code which can serve as a proxy for the neighborhood, as well as the distance to a subway.



Methodology

1. Gather Data
2. Clean Data
    a. Remove bad columns
    b. Remove outliers/bad rows
3. Explore and Augment Data
    a. Description
    b. Distance to subway
    c. Zip Code
4. Inspect Data
    a. Inspect correlations
    b. Inspect a basic linear model coefficients
5. Further Clean Data – remove data based on outputs of (4)
6. Model Development – Create models and tune hyper parameters
7. Evaluation – Evaluate the models and analyze the best model

Data Cleaning

The first check is to inspect each column’s distribution of values and see if it makes logical sense for rental data in NYC and note any abnormalities. We uncover that some columns may have unreliable data (laundry in building) and that some rental units in the data are outside of NYC or have bad location data. Further, some data has outlier prices so we can remove these.

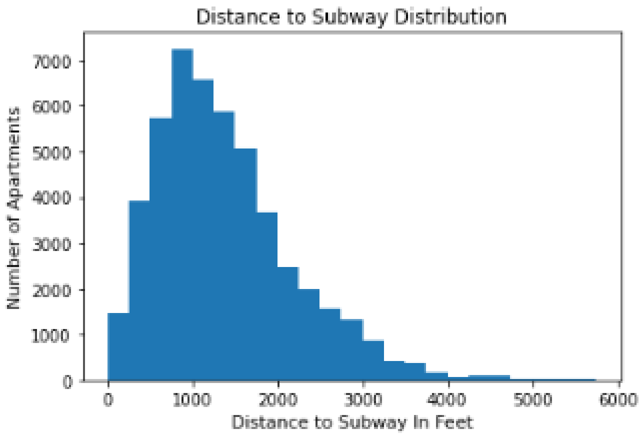
Explore And Augment Data

Zip Code

We use an external library to determine the zip code of a rental property. Rental prices in NYC are highly dependent on the neighborhood the property is in, so this can serve for those purposes. This data is a bit dirty, so we restrict to tracking only zip codes that have enough rental units in them to provide a signal.

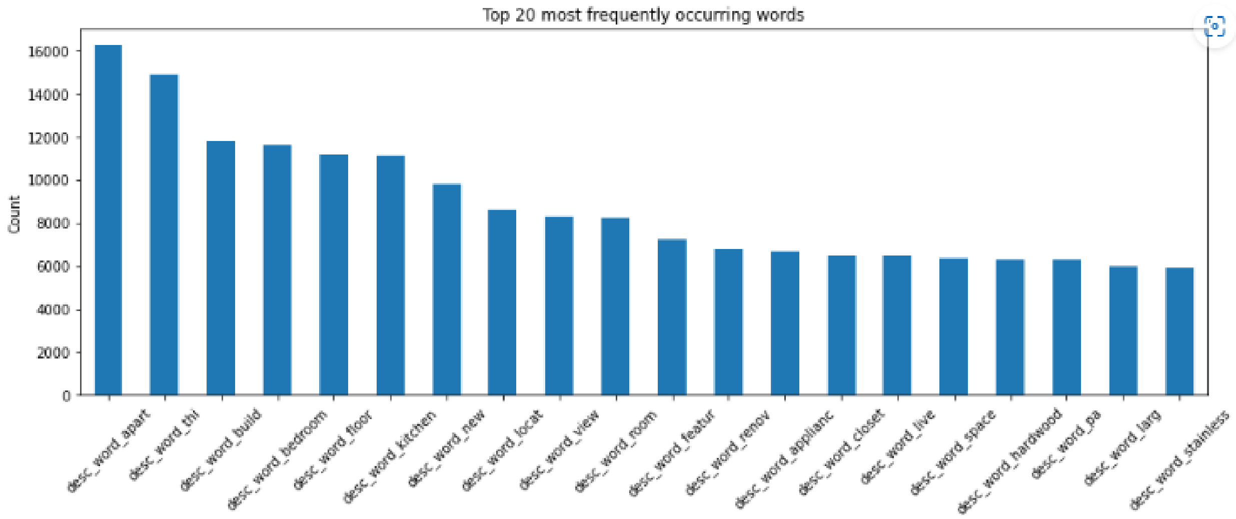
Distance To Subway

The subway is a popular form of transportation in NYC and hence the distance to one is highly impactful for rental prices. We calculate these using public data about subway stop locations. We further clean this by capping the value for outliers since some areas of NYC use buses or cars.



Description

We explore the popular words in descriptions. We must first clean up the data as some of it includes HTML tags and strange characters. We then see that popular words include terms like “renovated”, “hardwood”, “stainless”, or “kitchen”.



Inspect and Further Clean Data

Once the data is clean, we can begin to see how different fields may relate to the price of a unit.

Correlations

We look at the correlations of different fields to price. We see that bedrooms and bathroom counts have the most relation to price of a unit. This makes sense, as do most of the other correlations, so this gives us confidence the data is logical. We see some columns have correlations in the wrong direction and allow us to clean or remove such data.

Basic Linear Model

We run a linear regression and inspect the coefficients. We see that some coefficients are illogical and use this as a baseline on how to clean the data. Similar to tuning hyper parameters, we may try to reduce the amount of data derived from description or location in order to get to an amount of data that does not create overfitting.

We see that limiting the zip codes and description words to train on prevents overfitting in the model.

Model Results

We try three types of models. For each, we attempt to tune the hyper parameters in order to achieve the best fit against a validation set.

Linear Regression

Although this model does not perform very well, it provides us coefficients for each column in the data set which may help landlords understand the market. We can see how much adding a laundry unit into the apartment or allowing pets may increase the rental price. This can be used for a landlord to make decisions on how to run their unit. We can also look at the word description coefficients and see which words best improve the rental price. We see that some examples of the most impactful words to include in a listing are shown. They indicate that allowing a flex bedroom may be a desirable trait of a rental unit.

Good Words For Listing Description
support
estate
flex
convert
equal
separate
exposure

Random Forest and XGBoost

XGBoost is an improved version of Random Forest that combines weak outputs from a decision tree. These models make logical sense for determining rental price because renters make decisions on what apartments to rent similar to a decision tree.

We see the best results come from an XGBoost model that has tuned hyper parameters. Against the validation set, we have these results:

	Validation Score	Training Score	MSE	RMSE	MAE
Linear Regression	0.67	0.67	1430000	1195	701
Random Forest	0.83	0.975	750000	864	424
XGBoost	0.85	0.998	650000	806	404

We run the XGBoost against the test set to achieve a 0.804 score.

We see that the random forest and xgboost models perform better but have over-fitting. Likely we need more data points (such as square footage) or cleaner data to improve the model. We may also continue to tune the models.

The MAE indicates the model would on average be off from the actual price by around $400. This can ensure that a landlord is within the ballpark of the correct price to charge.

Wrapping Up and Conclusion

We have a model that can provide a rental price given the qualities of a rental property. This is trained on data from 2016, so we have to scale to 2022. We can multiply by the difference in average prices. From data on zumper.com, we can see that the median rental price in spring 2016 was around $3000 vs $3800 in fall 2022. Assuming that how contributing factors impact price remains constant (6 years is not long in the real estate space), we can multiple prices by 1.27x to get a modern price.

With this model, we can provide landlords a way to get within a ballpark of the market price for their rental property. We can also provide how different features may impact the price of their unit (including elements under their control like whether they allow pets) as well as popular words to include in their listing description.

To follow up on this model, we see that it is overfit, so it would be best to find more data points that are more directly related to price to include in the model, get more data points, and then to clean the data further. Some examples of data to find are: square footage, ceiling height, window type, utilities cost, and data from 2022 on beyond.

We may also deploy the model to a website to make being a landlord easier than ever.