Springboard Data Science Intensive Course

Capstone Project 2

New York City Property Valuation

Marcus Bamberger

Table Of Contents

Abstract

New York City's property valuation data was used to create multiple Random Forest models predicting property valuations. As property valuations cost several hundred dollars per assessment, a robust property valuation model has the potential to save millions of dollars in valuation costs. The most robust and viable valuation models constructed were a random forest classification model, a random forest regression model, and an experimental composite random forest classification. The most influential factors in our models were land area in gross square feet, geographic location (expressed as ZIP code), and the year the property was built.

1. Introduction:

In 2020, the total market value of all properties in New York City was assessed at around 1.4 trillion dollars. That's the total GDP of the United States in 1973, or the total GDP of the United Kingdom in 1996, or China in 2002, or Australia today. Anyone who operates in or adjacent to the property market, whether they are homeowners, real estate developers, or stock market traders, has a stake in knowing how much a property is worth.

The issue is that properties, especially residential properties, are difficult to price because the market caters to a bewildering variety of needs. A three-bedroom, five-bathroom house would be perfect for a large family but a detriment to a bachelor. For this reason, a more consistent metric of property price is its assessed value, an appraisal done for the purpose of calculating property taxes. The Department of Finance values properties once per year, with the option to challenge the assessment if it is believed to be incorrect. This metric provides a more consistent means of tracking property value than the market price.

This project focuses on predicting New York City property assessed value from factors such as property size, presence of easements, year constructed, and borough of the city. The target audience includes anyone interested in property valuation, both homeowners and companies. It is also possible that this project could be used to identify anomalous data, such as someone deliberately underpaying their property taxes. While ideally such a model would be trained on data known to be falsified, it remains a potential application.

2. Data Collection:

Initial NYC property valuation data was taken from the NYC OpenData website, here. The provided spreadsheet tracks 5.74 million Class 1 property valuations from 2010 to 2017, with 117 columns of associated data such as house number, year of any alterations made, tentative and final valuations of land and property, and more. Tax class 1 includes 1-3 unit residential properties. Class 2 covers larger residentials, such as apartments, 3 denotes utility company property, and 4 is everything else. This project was restricted to Class 1 to remain at a manageable size and because valuation metrics may vary between property classes.

Where necessary, the data were verified from other property – focused websites such as Zillow. Additional information on the mechanisms of property valuation and the New York City property market were taken from sites such as the New York City Department of Finance website, here.

2.1. Data Cleaning:

The dataset contained a large number of superfluous variables where most or all of the data was a single fixed type. These columns were removed to reduce dataset size and make loading and handling the dataset easier: When handling the full dataset, Jupyter notebook would regularly crash with memory errors.

It was immediately clear that the YRB column, indicating the year a property was built, was only an estimate.

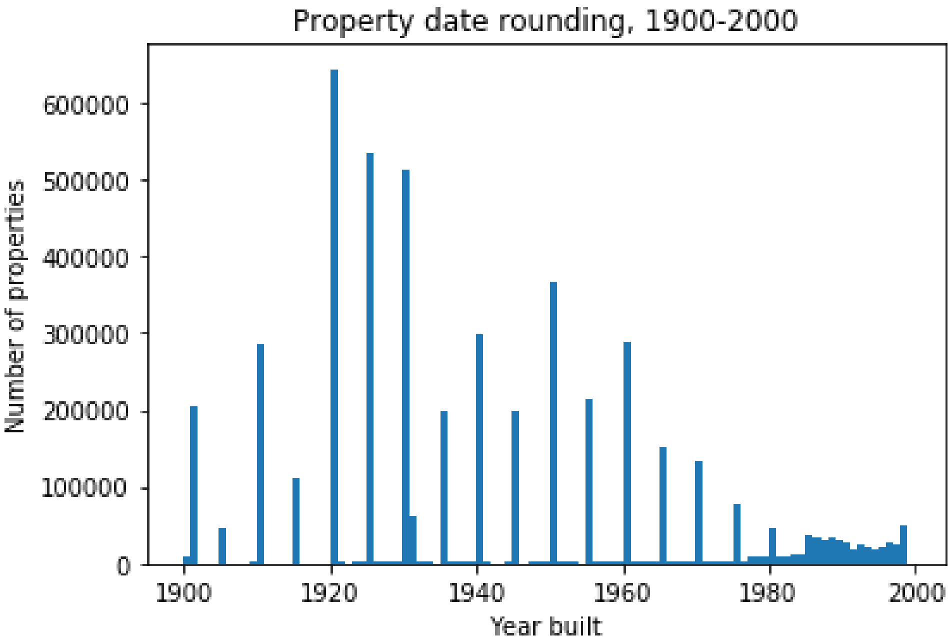

Figure One: Histogram of reported property build dates in the nineteenth century

Properties have been resoundingly reported as being built in years divisible by five up until the late 1980s, when the pattern stops. Until recently, the year-built column was only an estimate, not a precise number. Not only does this introduce error into the data, it also obfuscates other trends in valuation over time. For this reason, for visualization purposes only, we wrote a program to subtract a random number, 0-4, from the provided year, introducing more variation and spreading out the data. There was no point in making this modification to the modeling data, as this does not improve accuracy.

Although it is not viable to manually examine every column of the data, one cluster of outliers was immediately, obviously apparent:

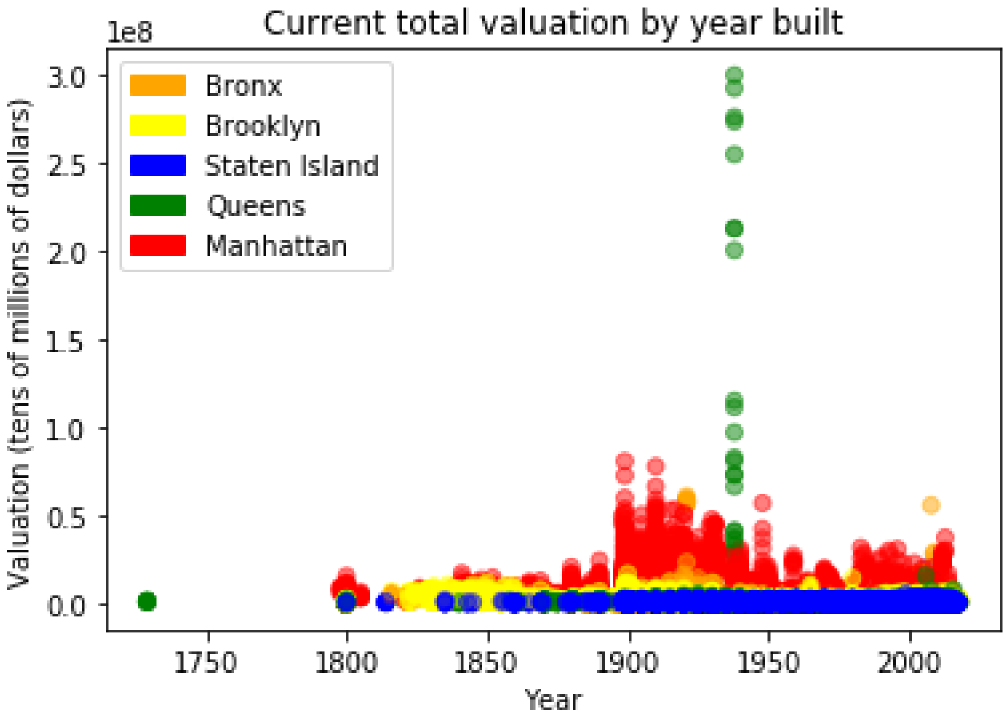

Figure Two: Scatter plot of property valuation by year and borough. Zero-year properties have been omitted from this graph.

The 1930 Queens properties, the large green spike in the middle of the graph, are anomalous in two respects: First, every other property in Queens is valued at a fraction of these nineteen properties. Second, every one of them was ostensibly built in 1930. Checking Zillow's listings for these properties indicates that several of them were not, in fact, built in 1930. This anomaly was difficult to explain until we encountered another error in the data.

Around 5% of the properties report 0 as their year of construction. As New York City was first settled in 1624, we may infer that these properties do not have a known data of construction. To correct this, we replace each zero value with the mean of the borough. It is possible that the properties with missing values are on average older than other properties – note that properties built after 1985 have not had their dates rounded, indicating better record-keeping in the modern era. For this reason, we will also examine models where the zero values have been replaced with years significantly below the mean.

Modeling this data made the Queens spike clearer:

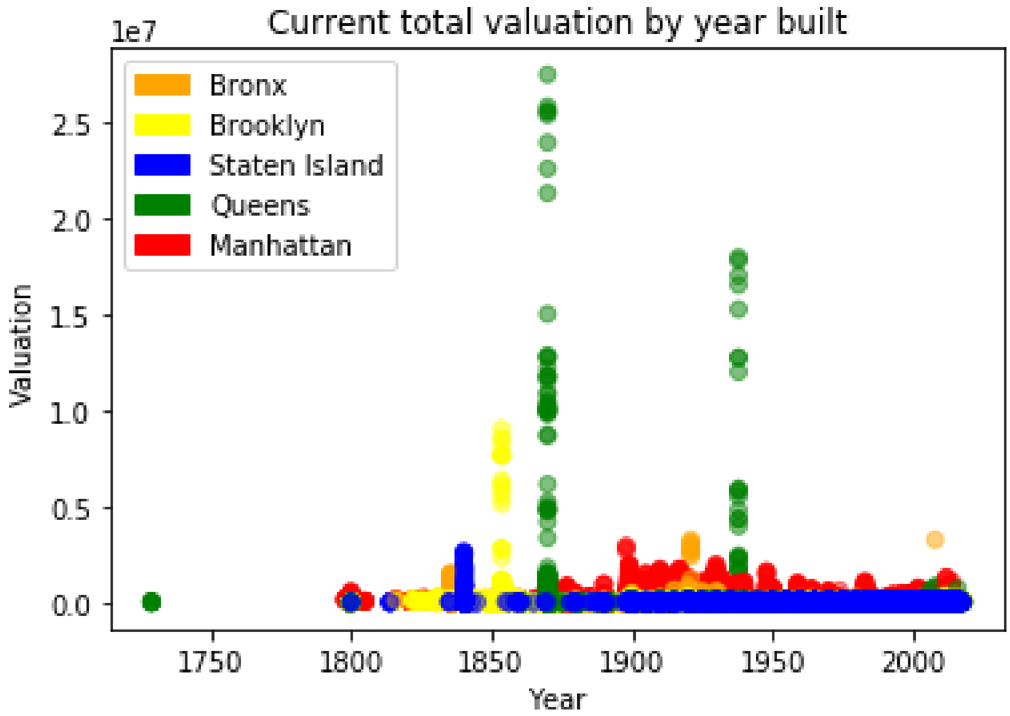

Figure Three: Scatter plot of property valuation by year and borough. Zero-year properties have been listed as being built in the mean year of their borough.

We can see that each borough has a number of valuable properties that were listed as being built in 0. It is likely that the 1930 Queens spike was due to someone entering 1930 as the year for a number of properties with unknown years of construction. It is unclear why only a few properties from one borough would be altered in this manner, but this does seem to be the most likely explanation. These older properties are likely being valued more highly due to a historical factor. Although this data is unusual, it does not appear to be false, and it was left in the dataset.

Finally, several categorical variables such as type of easement class (A, B, E-M, etc.) needed to be broken out into separate binary columns (Class A, True/False) in order to prevent modeling from creating inadvertent correlation between them.

3. Statistical Data Analysis:

3.1. Geographical Location

The single most important factor in the dataset is the borough. Borough alone, however, does not fully capture the geographic factors that drive property valuation. Targeting Manhattan data block by block, for example, significant variation can be observed:

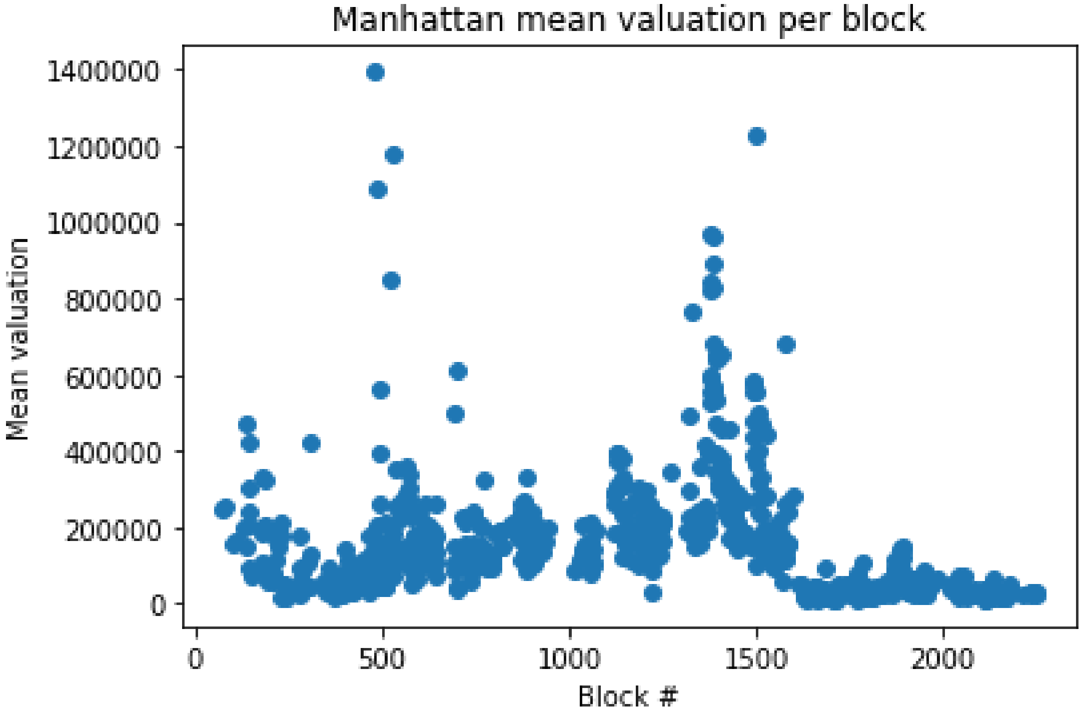

Figure Four: Scatter plot of Manhattan property valuation per block

Block # is clearly an important feature in the dataset, but mapping data by block poses difficulties. Firstly, with thousands of blocks, mapping the data is computationally expensive. Secondly, it is difficult to find a geoJSON map of New York by block. Finally, block data is correlated with borough data: Manhattan block numbers only run to ~ 2000, but all other boroughs reach at least three times that. For this reason, we selected zip code as a compromise: A variable that is easy to plot, contains more specific geographical information than plotting by borough, and can stand on its own.

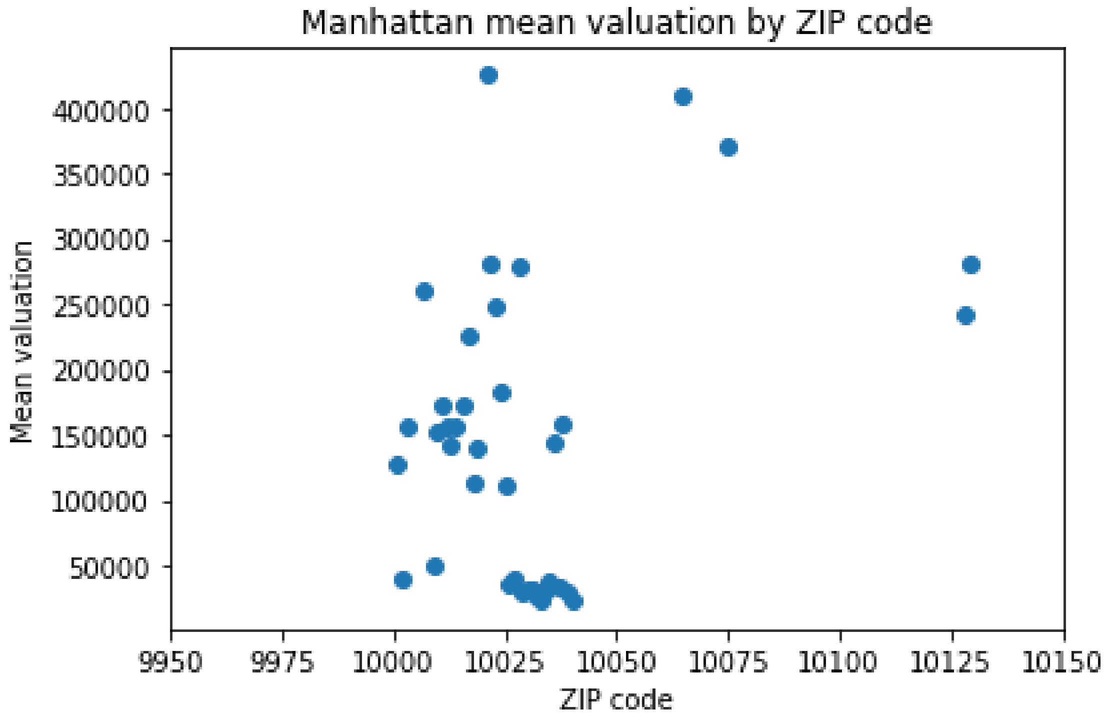

Figure Five: Scatter plot of Manhattan zip codes against mean valuation.

As the graph indicates, significant variation between zip codes does exist, with the cheapest Manhattan zip codes being valued at around one-tenth the cost of the most expensive ZIP codes. Still, some value has been lost: the dataset has been reduced from 758 distinct entries to 40.

3.2. Mapping

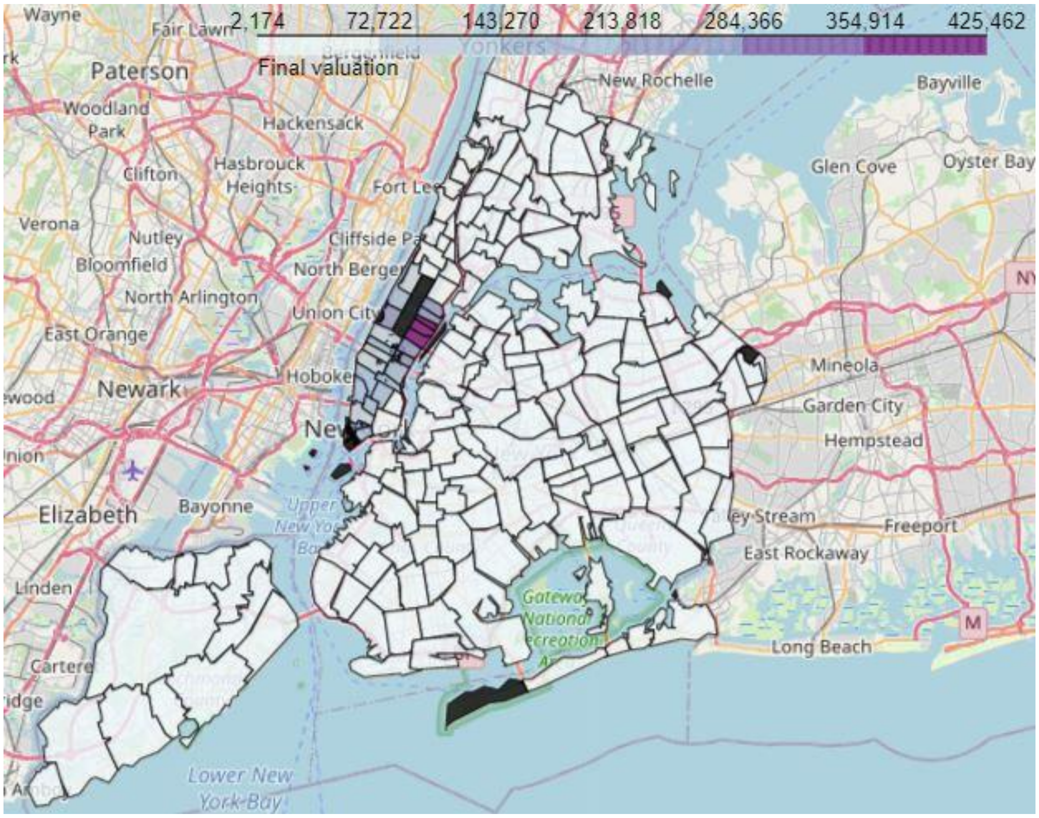

Figure Six: Choropleth map of New York City mean valuations by zip code.

In this unweighted map of New York property valuations, Manhattan’s zip codes are valued so far above the rest of the city that they aren’t colored in at all. This is clearly difficult to extract much meaningful data from, save that the Upper East Side is the most expensive part of New York City to live in.

With some wrangling, it is possible to produce interesting maps from this data.

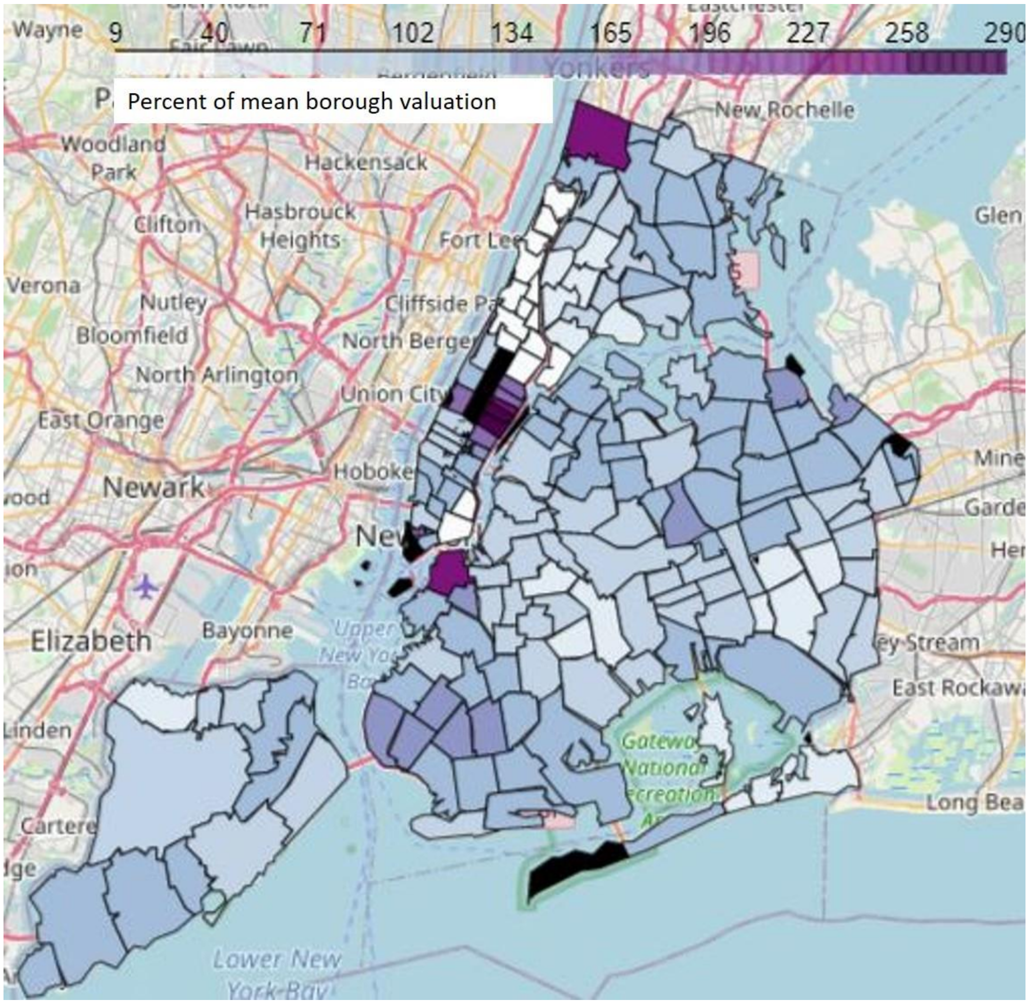

Figure Seven: Choropleth map of New York City zip codes, percent of mean borough valuation.

This choropleth map shows how far each zip code is from the mean valuation of its borough. We will experiment with other valuations as we continue working with this mapping module, Folium. Already, though, we can see that the Upper East Side is the wealthiest part of New York City, North Staten Island properties are valued less than south Staten Island properties, North Riverdale is highly valued, et cetera.

3.3. Final Factors

One especially interesting valuation pattern that emerged when examining the data was the number of stories the building had.

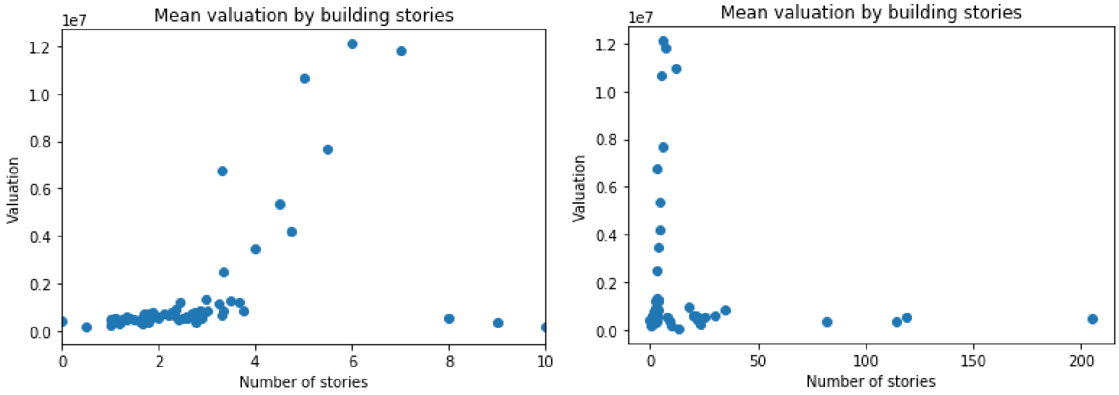

Figure Eight: Mean valuation by number of stories, 0-10 and 0-200.

First of all, there are a surprising number of properties with non-integer stories: over a million total, or almost 20% of the dataset. Half-stories like 2.5 and 1.5 are the most common, then 2.75, 1.67, 1.75, etc. It's unclear who would mark down a house as having 2.85 stories, or 1.60, 1.99, or various other improbable decimals, but we will assume that these are unusually specific and punctilious housing records. Second, property valuation increases exponentially until around 6 stories, then abruptly falls back to a consistent level. It's our hope that a simple random forest model will be able to accurately model this.

In about 30,000 properties, the valuation was protested. Some of these, most notably protest code 6, 1, and 6E were valued significantly higher than average. Others, like codes 9 and 5, were valued below the mean on average. Due to the very small total number of protested valuations (less than one percent of the total dataset), there was no reason to include this data in the random forest model.

Property area was consistently positively correlated to land area. The last factor that we expected to be strongly correlated with price was the presence of easements, but there were almost no Class 1 properties with easements (< 2000), so this was mostly irrelevant.

New York City caps the amount that a property can increase in price in a year to 6%, and in five years to 20%, so tracking the price of a property in 2010, where known, would be a simple way to add this factor to the model. However, this factor would actually be too good for our model to handle. Adding a variable that is always so closely correlated to the final valuation would eclipse all other variables in the model. It is for this reason that we will proceed assuming that no records on previous valuations are present.

In another example of punctiliousness, the dataset includes columns for current/transitional/final, transitional/actual, assessed/exempt, and land/total value, for a total of

24 permutations of value assessment. For simplicity, we will attempt to model Final Actual Assessed Total Value. For our next steps, we will begin working with random forest modeling to predict ‘Final Actual Assessed Total Value’ from all other factors. We considered including tentative valuations as a variable in the model, but a close reading of New York City property valuation law indicates that tentative valuations become final unless challenged, and as discussed above, very few property owners challenge their valuation. For this reason, the feature was discarded.

4. Machine Learning

4.1. Random Forest Classifiers

Initially, the data was modeled with a random forest classifier. The Random Forest model was selected for several reasons: Its speed and ability to handle large numbers of input variables made it an ideal choice for a dataset 2.2 gigabytes in size (5.3 before cleaning to remove superfluous data), the scikitlearn package provided verbose modeling feedback, and the classification model performed well for simple assessments.

Choosing the classification target proved difficult, however. The first classifier tested was evaluating if the property was valued above or below the mean valuation in the dataset. This proved to be a poor choice because, again, the Manhattan data dominated the model. The only input variable of serious interest was ‘Is this property in Manhattan?’ For this reason, another classifier needed to be selected.

The classification best suited to the dataset was determined to be ‘Is this property valued above the mean valuation for its borough?’ From borough to borough, this covered between 47-33% of the dataset. Classifying by borough ensured that properties in each borough would be both positive and negative, preventing the borough data from dominating the model. The data also divided well into above/below categories, meaning there was no need to sample the dataset for an even variable split.

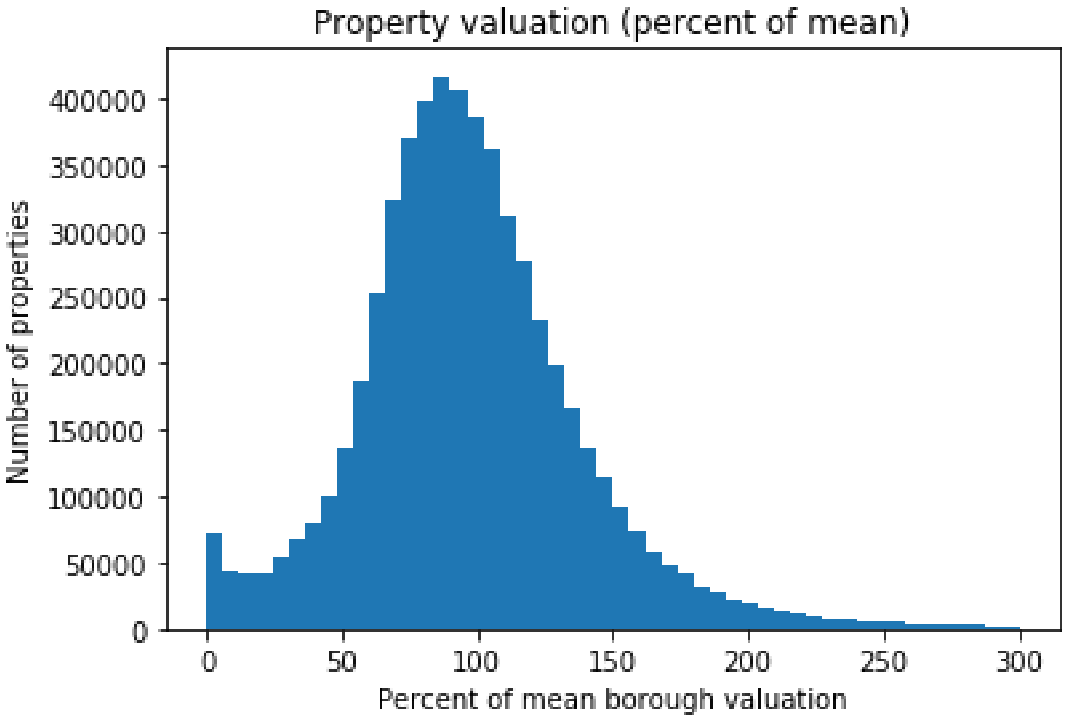

Figure Nine: Histogram of property valuation as percent of mean..

The dataset was too large to realistically model in its entirety. A simple three-variable random forest classification model took over six hours to train on the entire dataset. Initially, this was solved by splitting the dataset up by borough, but this had several issues. Firstly, a model trained on one borough performed significantly worse on the other boroughs, and secondly, it created issues when trying to draw conclusions about the New York City housing market as a whole.

The issue, then, was how to best model the dataset at an acceptable speed? Modeling even 40% of the entire dataset with an abbreviated feature list and reduced tree count took over 21 hours to complete. Clearly, significant sampling was required.

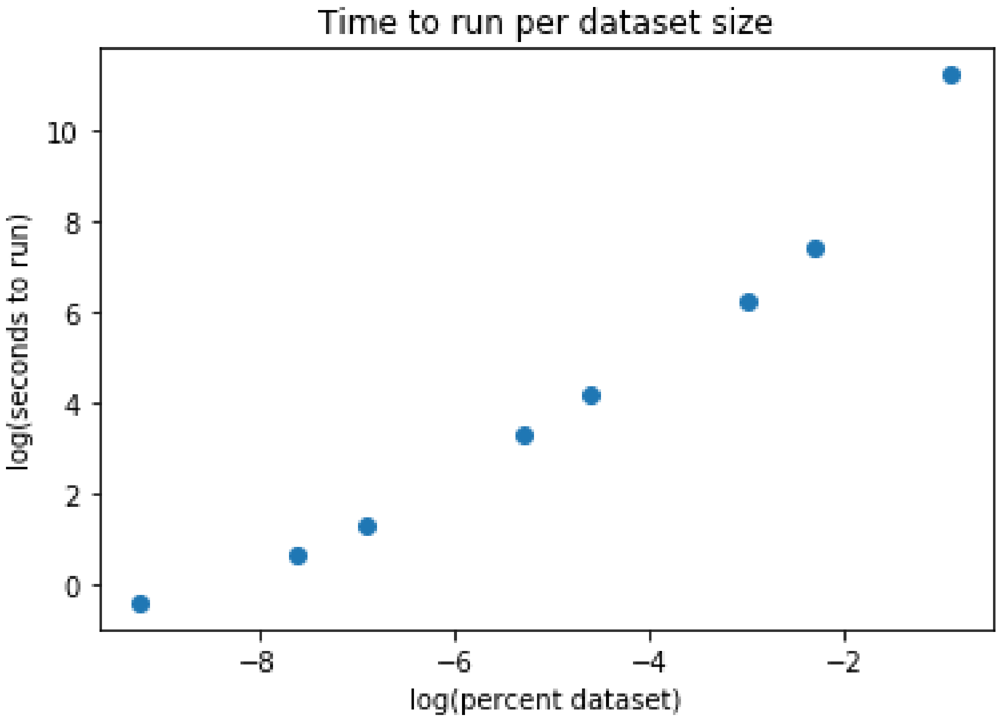

Figure Ten: Time to run random forest models of varying size. Due to the broad range of times and sample sizes, logarithms have been applied to the data.

For this reason, the second round of models were trained on a random sampling of 1% of the total dataset. These models were run ten times with different samplings to verify that key features of the dataset were preserved by sampling down to 1%. The standard deviations of the feature weights, from .008 to .0008, indicated that there was almost no difference in the models produced by this sampling and all key features were preserved.

We strove to avoid any data correlation when training the models and included only one column for each factor. ZIP code, land area, year built, current year, and total units in the property were all factored into the model – We incorporated any feature that had relevant feature-weight, another advantage of the random forest model. If a variable was not improving the model, it could be quickly identified and discarded.

Our final parameters for the model were, in order of feature importance:

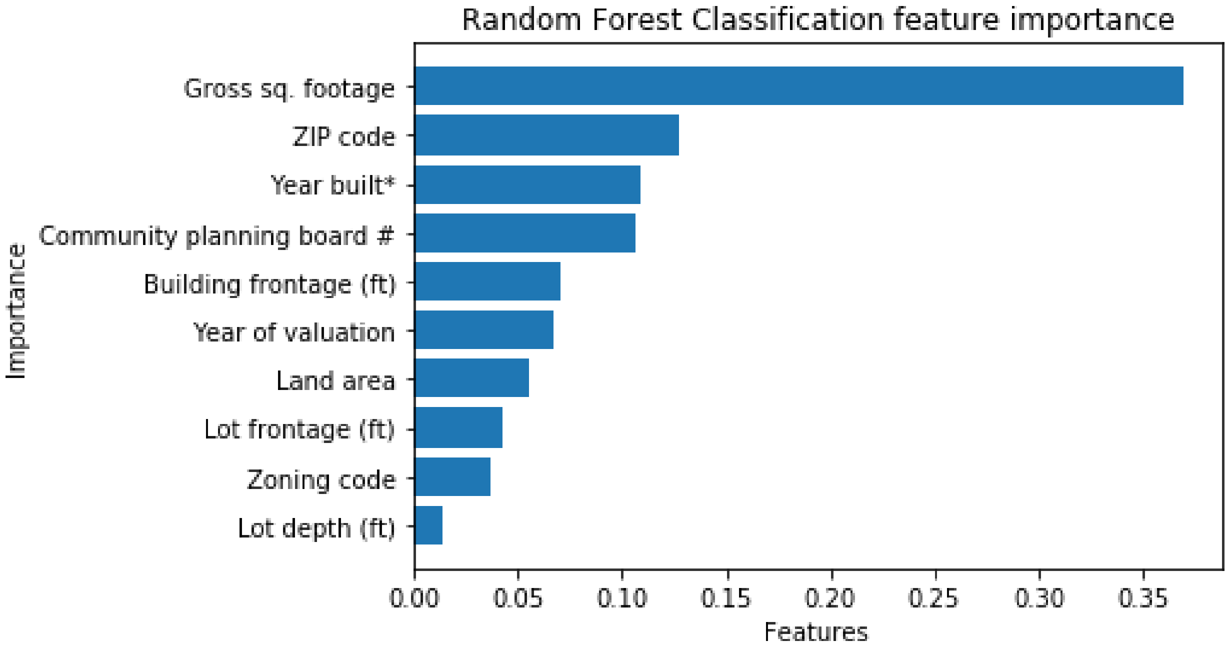

Figure Eleven: Feature importance for random forest classification

*: As mentioned in Data Cleaning, models were tested with both year-built variations. No significant variation was observed.

The data indicate that the most important feature when predicting valuation was the gross square footage of the property, followed by ZIP code and year built. Hyperparameters were set to 1000 estimators, max depth 10, minimum 1000 samples per split. The default criterion for node splitting, gini impurity, was retained, and the default maximum node size was also preserved. This produced a model with train score 0.856, test score 0.841, indicating almost no model overfit. It took approximately 1.5 minutes to train a 1000-tree forest on this data. Precision and recall were calculated to be 0.817 and 0.759. We then calculated the best depth for the model and found the following pattern:

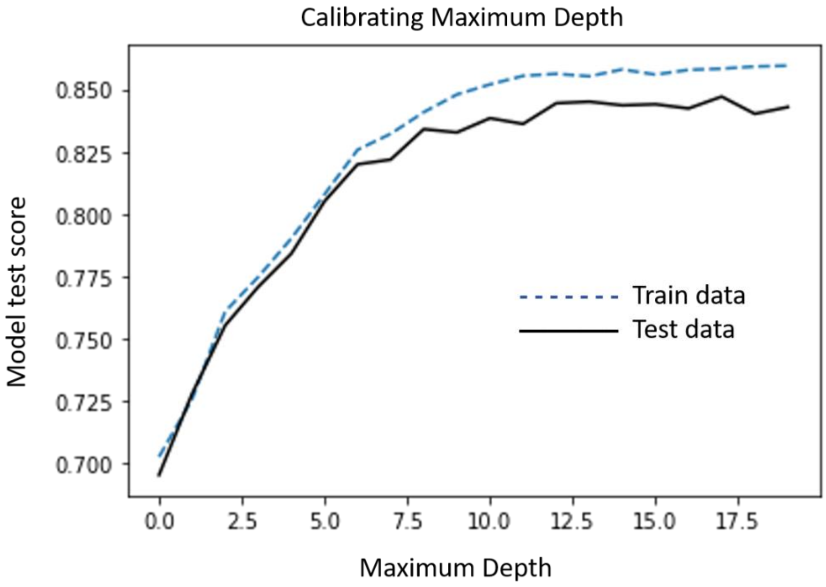

Figure Twelve: Model performance at varying depths, train and test score.

Maximum depth of around 8 appears to produce the best results with minimal overfit. We then attempted to produce the optimal results by running a robust random forest classifier sampled on 10% of the total dataset. This returned a train score of 0.845 and test score of 0.844, indicating that increasing dataset size beyond 1% did not improve the model.

At this point, we had reached the limit of what a simple classification model could accomplish. ‘Is this property likely to be above the mean valuation of its borough’ is not enough information. We next turned our attention to compositing several random forest models together.

4.2. Composite Random Forests

Random Forest Classification models proved to be highly accurate, but very imprecise: They only predicted if a valuation was above or below a certain amount. To remove this weakness while preserving the model’s strengths, we wrote new code to train seven random forest models on a single train/test dataset split, with the same hyperparameters as the basic Random Forest Classifier. Each model would classify properties as being at or above a different valuation level: the mean valuation in their borough plus or minus a fraction of the standard deviation of the mean.

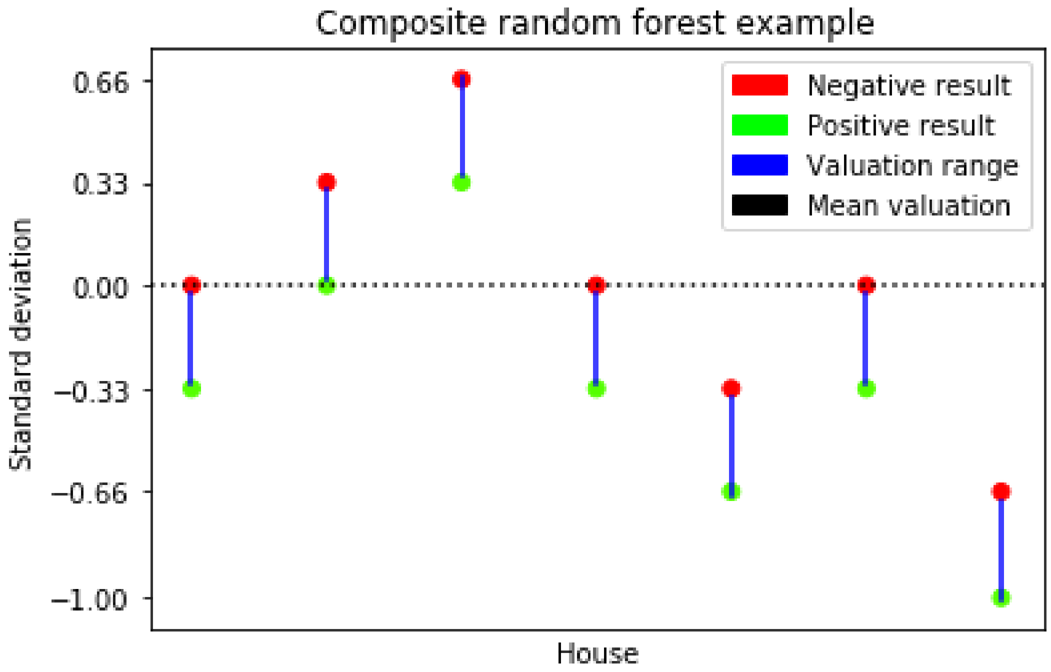

Figure Thirteen: Example of composite Random Forest model

Here's an example of this model in action. We've fed in seven properties, and for each one, the model has determined that its valuation most likely lies between two values. For the first house, that is between the mean valuation in this borough, and the mean valuation minus one-third the standard deviation from the mean.

This model was able to correctly bucket 67.7% of all test data and placed 96.1% of test data within one bucket of the true value. This represents a four-fold increase in precision over a conventional random forest model with acceptable loss of accuracy. Training time was approximately four minutes for 1000-tree forests, an acceptable timespan. However, this model does have flaws that made further exploration with it impractical.

Firstly, expanding the model further is difficult to automate: Each new classification tier needs to be built out and added to the model. Secondly, the model is difficult to score, as neither a conventional r-squared or accuracy score applies. Thirdly, the model is difficult to explain. For these reasons, we moved to regression modeling.

In the event that this model was revisited, it could be improved by increasing the number of classification models it composites. There is little reason to expand the range of prediction beyond one standard deviation, +-, as very few properties are valued so far above or below the mean. Instead, the additional models should be used to better predict property valuation within the existing range: move from training models on mean +- 1/3 standard deviation to mean +- 1/5, for example.

Initially, we were concerned that this model would contradict itself – for example, a property would be predicted to be valued less than the mean valuation and also equal to the mean valuation plus one-third the standard deviation from the mean, but

4.3. SGD Regression

SGD regression, or Stochastic Gradient Descent, is a regression model that can quickly and efficiently build regression models. Stochastic gradient descent proved to be orders of magnitude faster than random forest modeling, capable of building a model using the entire dataset in ~30 seconds. The input data needed to be scaled, but this was accomplished quickly. This increased speed came with reduced accuracy, however: With hyperparamters maximum iterations of 1000 and the squared-loss function, SGD regression returned a mean test score of 0.14 with standard deviation of 0.38. Note: it would be .26 and .11 if not for a notable outlier. For this reason, we chose to continue modeling with a more robust, if slower, regression model.

4.4. Random Forest Regression

Having reached the limits of classification models, we turned our attention to random forest regression models. The goal here was to predict precise valuations for each property in the dataset: More difficult than bucketing or classifying the data, but also much more valuable for interested parties.

Random Forest Regression proved more difficult to set up and run. More of the parameters were found to be irrelevant, and sampling on 1% of the dataset as with classification modeling proved to be significantly more volatile. Although the models would train and run in around one minute, their mean adjusted r-squared scores were -0.055, with a standard deviation of 0.908. In other words, completely unusable.

We were able to reduce volatility by increasing the dataset sample size from 1% to 10%, demonstrating that random forest regression benefits more from a larger sample size. Unfortunately, the model still performed too poorly to be used.

We suspected that these issues could partly be attribute to the very broad range of property valuations in the dataset. Again, the Manhattan dataset acted as a major outlier, distorting the rest of the data. Reducing sample size back to 1% and shifting the target variable to percent variation from mean slightly improved performance, but we found our best results when testing the regression model on a log(valuation) target. This improved the model significantly, bringing the mean r-squared up to 0.63, with standard deviation down to 0.025.

Surprisingly, the feature weighting of this model was largely keyed to the gross square feet variable alone. After removing all unimportant features, these were the feature weights:

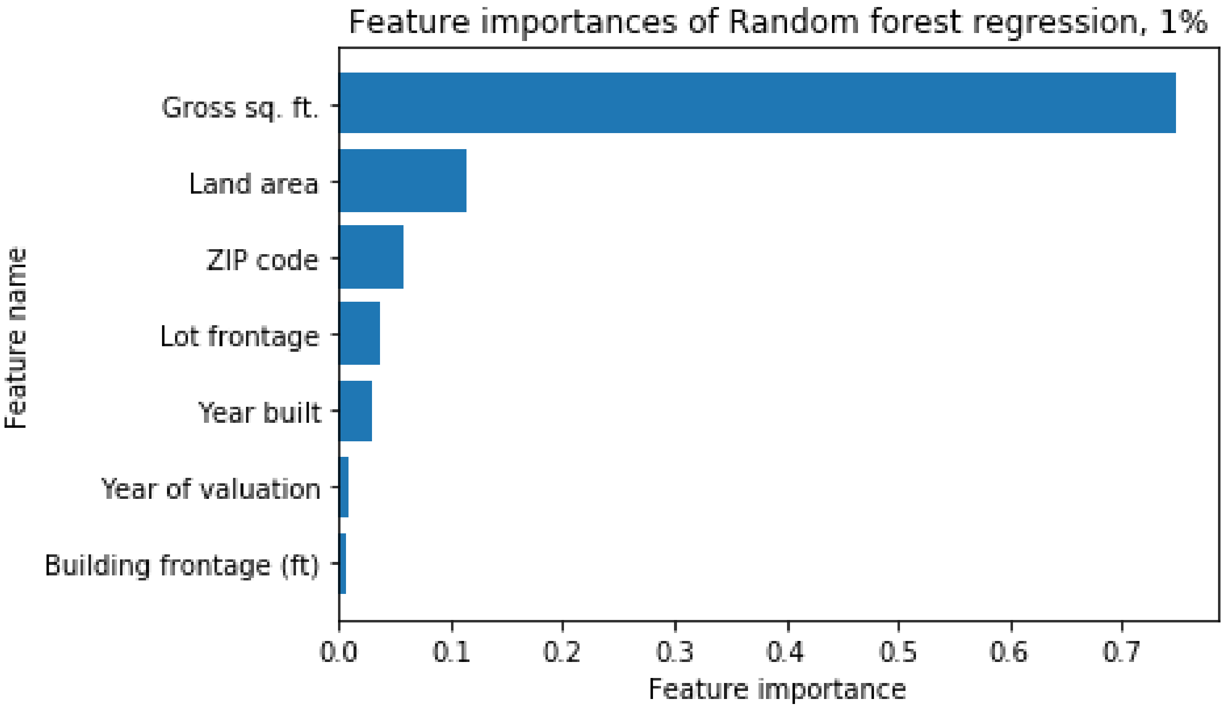

Figure Fourteen: Feature weights for Random Forest Regression

*: As before, models were tested with both year-built variations. No significant variation in final scoring was observed.

Unlike a random forest classifier, improving the sample size helped significantly. Running the model on 10% of the dataset with a 1000-tree, 10-deep forest improved the model performance to 0.694 on the train data and .701 on the test data, indicating almost no overfit.

This also reduced the feature importance of the gross square feet variable in favor of land area, ZIP code, and lot frontage.

As an experiment, we trained a 100-tree random forest on 100% of the dataset. This improved the model further, to a train score of 0.719 and test score of 0.716, with RMSE of 0.405.

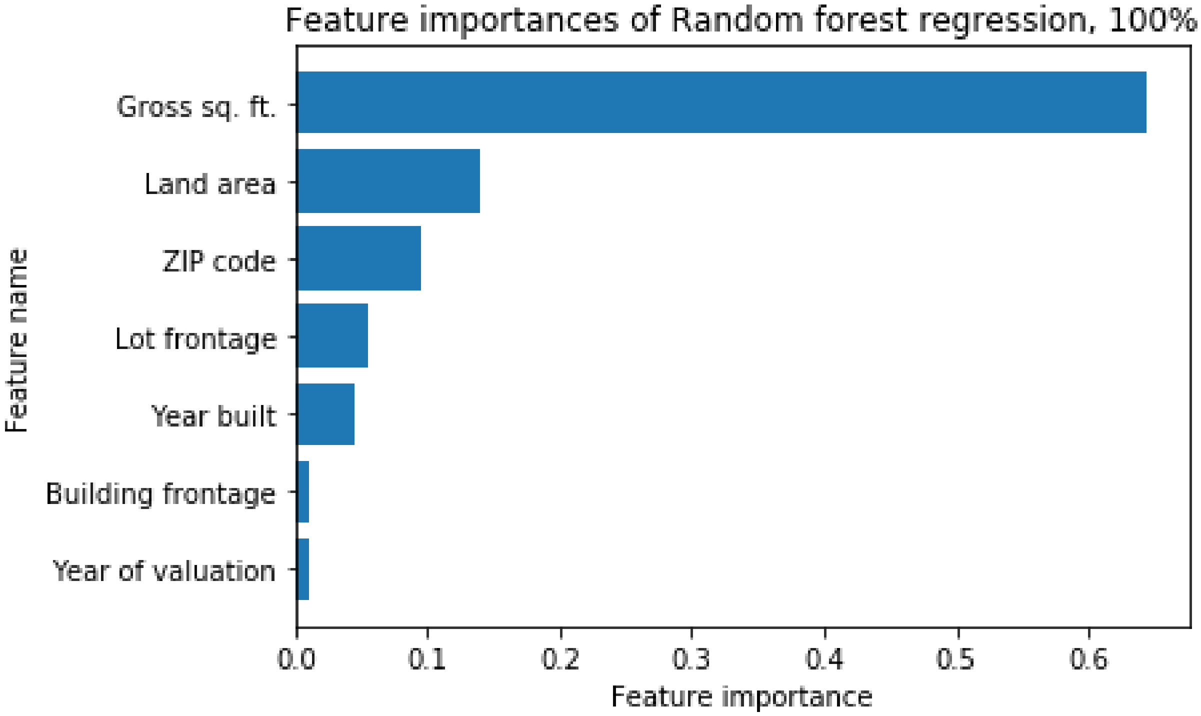

Figure Fifteen: Feature weights for random forest regression trained on 100% of the dataset.

With an adjusted r-squared score of 0.716 on the test data and RMSE of 0.405, this model was able to account for over 70% of the variance in property valuations, with an average error of 0.405. We can also visualize the valuation error to determine where the model fails.

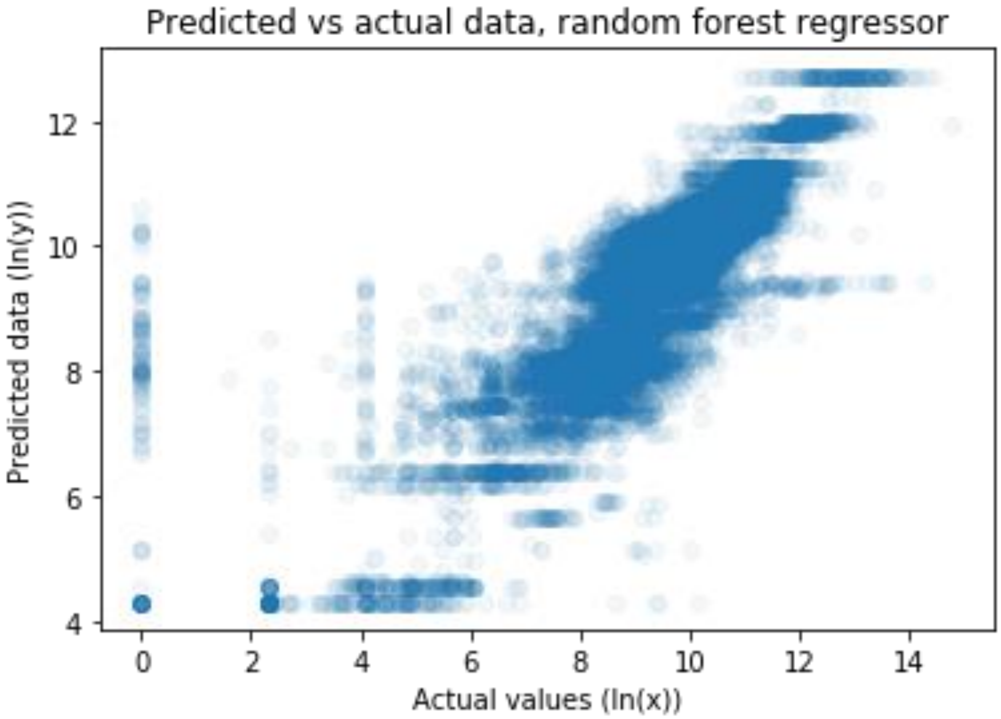

Figure Sixteen: Scatter plot of actual property valuation against predicted valuation for the final regression model.

We can see that the model encounters difficulty predicting low-valuation properties, which is likely due to the input dataset not including factors such as property damage, wear and tear, and other factors that would drive the price down. Without those factors, the model will tend to overshoot. We can also visualize error by zip code:

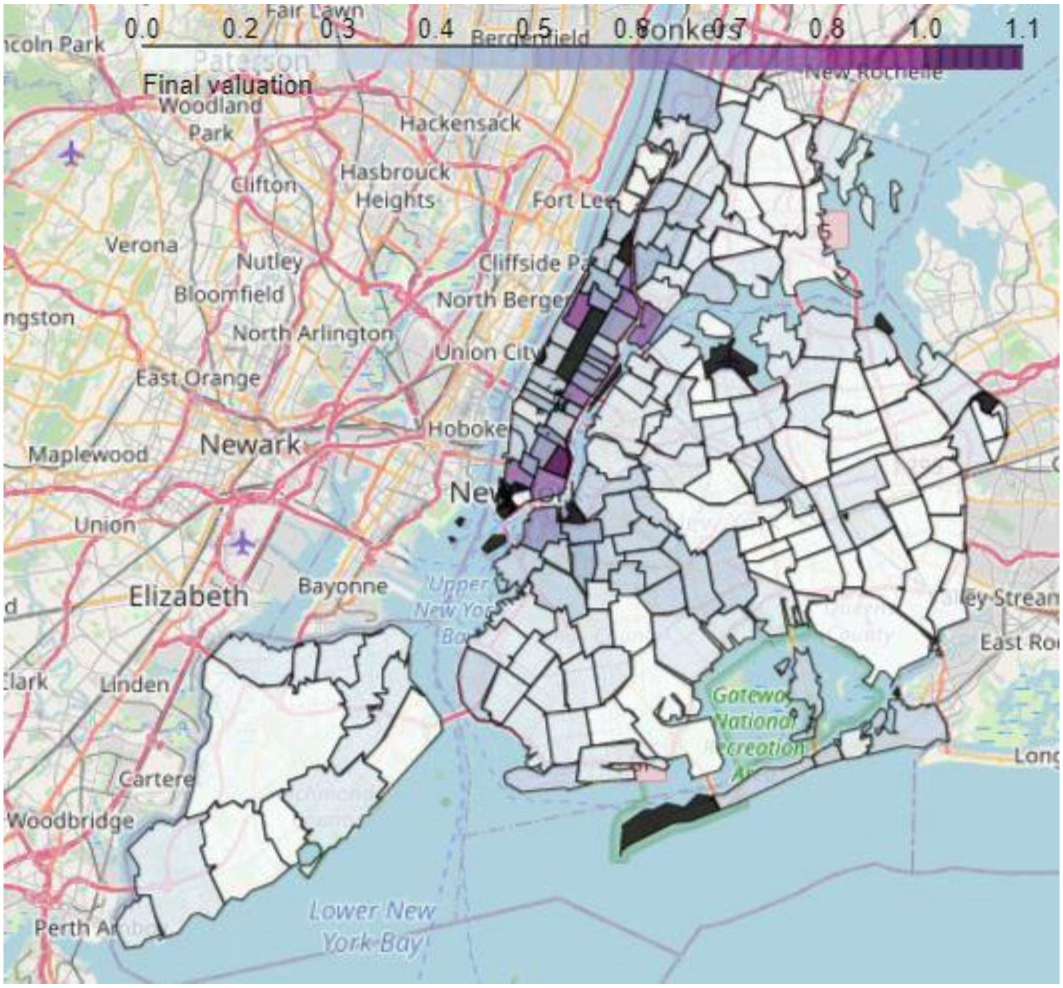

Figure Seventeen: Mean error by zip code for final random forest model.

We can see that Manhattan tends to perform worse in this model. Taking the log of the data smoothed it out enough for the random forest model to process it, but it still remains significantly higher in value than the other boroughs, and the model struggles to accommodate that.

5. Interested Parties, Usage, and Visualization

5.1. Interested Parties

Although we have discussed a potential application as fraud detection, we believe that a better application would be general valuation for real estate companies/purchasers. We do not have access to datasets outlining known cases of fraud, and the classification models would need to be reworked to accept final valuations as an input. For this reason, we will focus on general valuation. This is of interest to anyone in the real estate market in New York City, but we believe it will be of specific interest to groups performing/requesting a large volume of property valuations.

5.2. Usage

The LendingHome company estimates the typical cost of a housing valuation to be '$300-400'. Given that New York City performed 5.7 million residential valuations between 2010 and 2017, this represents a significant expenditure. Although these models are not sufficiently robust to completely replace human valuation, they can supplement it to reduce costs.

The random forest classification model is significantly more specialized than the random forest regression model. Although there are circumstances where it might only be necessary to predict valuation as being above or below a single price level, in most cases the random forest regression will be preferable, as it will predict an actual valuation, not a category. The composite random forest classifier was an attempt to reach a middle ground between the two models, but ultimately is too cumbersome for dedicated use.

5.3. Visualization

Random forest models can be visualized as decision trees, but given that this will be presented to a non-technical audience, it may be best to focus on results and not get bogged down in the details. For this reason, we do not recommend focusing on decision trees or other methods of visualizing the model's process and instead emphasize results.

A method to compare the property to other properties with similar traits (geographical location, gross square feet, tax class, etc) or similar valuations might help give perspective.

6. Failure Points, Limitations, and Inconsistencies

6.1. Failure Points

Outlier management is always a critical part of data handling, and in this case the entire dataset covers such a broad range of property valuations that the models employed struggled to handle the target data. We saw that moving to a log(valuation) target significantly improved the model, at the cost of compressing the most expensive properties. The regression models will make more accurate predictions for less expensive properties.

It should also be noted that economics in general and housing markets in particular are subject to rapid change without notice. Especially in the current climate of COVID-19, it is possible that housing valuations will shift rapidly and without relation to previous valuation metrics. In order to preserve the usefulness of this model, it will need to be regularly updated with current housing valuation data.

6.2. Limitations

These models struggle to handle the full dataset. Most of our models were trained on 1% of the dataset in order to build models in minutes, instead of hours. Although the random forest classifier demonstrated that the model is not improved by scaling up to 10%, the more volatile regression models benefit more from a larger share of the dataset.

Some variables and flags only occur in a tiny fraction of the total dataset, such as properties with protested valuations. These variables were omitted to save on computational costs and because they would, by definition, only improve edge cases, but they could theoretically be added back to the model for marginal gains.

Some factors that affect valuation were not found in the dataset. For example, the physical deterioration of the house, condition of systems, structures, and interiors, were not tracked in this dataset. Additionally, although limiting geographical data to zip code helped simplify and visualize the data, additional factors within zip codes still affect the final results.

Finally, the extreme differences between different boroughs and their average valuation level makes training one model on the entire dataset difficult. The decision was made to pursue one unified model for consistency, ease of use and ease of training, but specialized, targeted models may perform better overall.

6.3. Inconsistencies

As discussed elsewhere, most of Manhattan is a significant outlier compared to the other four boroughs, but taking the log of the valuation helps suppress this. Another difficult set of data points to handle are the zero-year properties: properties listed as being built in the year 0 that are often significantly more valuable than the rest of the dataset. They represent a smaller share of the dataset and disrupt the results less, but it might be worthwhile to flag them in some manner nonetheless.

7. Further Development and Refinement

As discussed above, it is possible that training models on the individual boroughs would produce better results than training them all on the whole dataset. This would also reduce the amount of data being fed into a given model, improving speed.

Another way to improve the model would be to target it to fraud detection: Identifying specific, suspicious outliers that are valued well below the norm. This would require rebuilding the classification model with final valuation as an input variable, and for best performance would require training it with known cases of fraud. This would require careful calibration and weighting, as only a tiny fraction of all properties will have committed fraud, but it is an interesting avenue for future research.

Finally, the model could be improved by continuing to add more recent valuation data. This is especially relevant in light of the current economic uncertainty caused by the COVID-19 pandemic: Past economic performance may be a poor indicator of future results.

8. Conclusion

We’ve been able to construct several random forest models to forecast the New York City housing valuation system. We’ve determined that the most predictive factors in this model are the gross square footage of the property, the location, which can be best expressed as a zip code, 25 and the year the property was built. This is intuitively reasonable: it makes logical sense that these factors would do the most to affect price.

Our best model is the random forest regression model: Training on 10% of the dataset is computationally expensive, but not impossible, and an adjusted R-squared of 70% indicates a strong model. Although not fully capable of capturing the complexities of the housing market, it can make reasonably accurate predictions on its own.

The most likely application of this model would be to supplement current housing valuation systems. Because these manual housing valuations can cost $300-400 dollars, this has significant value. It should be noted that this model is unlikely to be able to completely substitute for current valuations, but can supplement them to improve speed and reduce costs.