Andrea Porelli
Thomas Hill
CS 109B
Home Price Prediction

Final Report

Problem Statement and Motivation

The problem of accurately predicting home prices is of great interest: to homeowners, home buyers, real estate brokers, and investors alike. Automated home price prediction has made enormous progress over the past fifteen years. Most famously, the listings service Zillow has lowered its prediction error to just 5%, down from 15% when the site first launched its "Zestimate" service in 2006.[1] However, many remain skeptical of the Zestimate's accuracy, and it is well-known that the service's performance varies widely by region.[2]

A quick browse through the sample kernels on the Zillow Prize's Kaggle site reveals the range of creative means that data scientists have employed to improve the algorithm's accuracy. Typically, such methods start by engineering new features from the data or collecting information on neighborhood features not available in the original MLS dataset. The problem with the latter approach is that there are countless neighborhood characteristics; it is simply not possible to account for all of them in a single model, and even if it were, such a model would not scale well.

In our project, we take a somewhat different approach, and attempt to use the predictive power of location itself to boost our model's performance. Our approach is partly inspired by the rich literature on "spatial autocorrelation", and the use of this phenomenon within predictive models in the fields of ecology and quantitative geography. According to a number of scholars and data scientists (such as Sergio Rey, the creator of Python's spatial analysis library PySAL), an attention to spatial autocorrelation can dramatically improve the explanatory power of statistical models.

In spite of its popularity in other fields, very few practitioners have attempted to exploit the power of spatial autocorrelation in real estate prediction.[3] While the three most important factors in real estate are "location, location, location", empirical attempts to predict real estate prices have typically employed relatively "spaceless" models, as a quick perusal of the sample kernels on the Zillow Prize shows.

The two underlying questions we attempt to answer in our research are:

[1] Zillow, Inc. "What Is a Zestimate? Zillow's Zestimate Accuracy." Zillow. Accessed May 1, 2018.
[2] "How Accurate Is Zillow's Zestimate? Not Very, Says One Washington-Area Agent. - The Washington Post." Accessed May 2, 2018.

[3] R. Kelley Pace, Ronald Barry, and C. F. Sirmans, "Spatial Statistics and Real Estate," The Journal of Real Estate Finance and Economics 17, no. 1 (1998): 5–13

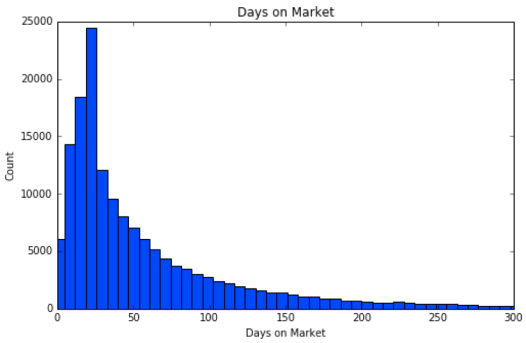
1. What is the best way to include location as a predictor when predicting home prices?
2. How important is location in predicting the sale price of a home?

Based on our results, the answer to the latter question is, in short, “Very!”.

Introduction and Description of Data

Description of the data

The MLS includes a wide range of features related to each individual home listed: from basic info like number of bedrooms and bathrooms, square footage, and date built, to locational information like nearby schools. A number of the fields, however, are not useful within a traditional regression context: for the purposes of our exploratory data analysis and the resulting baseline regression model, we disregarded such fields as: those pertaining to the agent assigned to the property, including office phone, agent name, showing instructions, etc.; those fields pertaining to the specific neighborhood, such as elementary school, junior high school, etc. (many of these were far too sparse to be useful); and those that are only known once the property is sold, such as “DOM” (Days on Market).



The MLS data also includes a wide range of qualitative and categorical features that we determined would be worthwhile to include in our analysis. “Style” and “OtherFeatures” were the most important of these. In order to include these variables in our statistical model, we created dummy variables for each distinct style and feature.

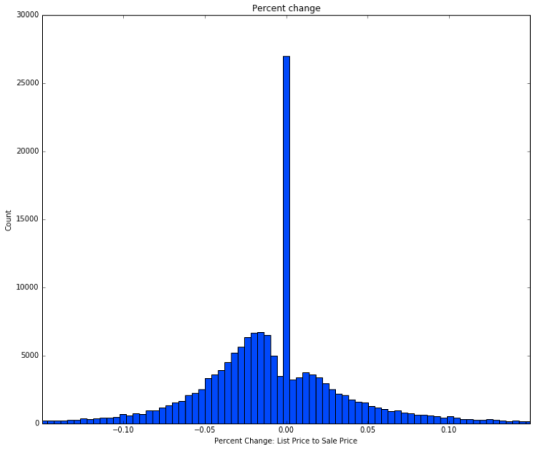
Crucially, we chose to disregard the “List Price” field, for four main reasons. First, the recorded List Price is often highly inaccurate: unlike the final sale price (which is recorded in the deed), the list price is not recorded in any public records. The list prices reported in the MLS are input by real estate agents at the time of sale. For obvious reasons, real estate agents have an interest in making it appear as if they listed the property for the same price that it sold for. After all, homeowners hire brokers to help them list their property at the right price.

Second, there is often no single “List Price”: homeowners (and their listing agents) often change their home’s listed price if it doesn’t yield offers immediately. In some cases, the list

price may change several times before the property sells. The recorded list price (in cases when it is truthfully recorded) typically reflects only the latest list price, not the initial one.

Third, since list prices are not recorded in public records, they are sometimes never input into the MLS. In situations where the list price is not provided, many MLS systems automatically fill in the "List Price" field with the property's final sale price.

The inaccuracy of the "List Price" field is obvious when one examines the distribution of price changes from list to sale. The histogram of Percent Change from list to sale below displays a highly unnatural spike at 0%.



Finally, we chose to exclude list price because we (and Zillow) would like our model to work for both listed and unlisted properties. Current homeowners should be able to use our statistical model to determine an estimated sale price for their own home, whether or not it is currently listed on the market.

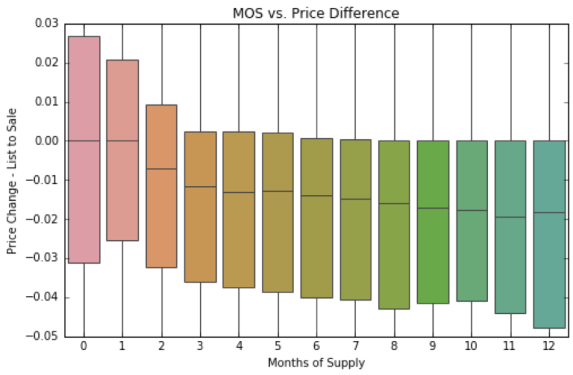
Feature Engineering

Using the data provided, we engineered several additional fields that enhance the performance of our baseline model.

Months of Supply

For every property listed, we calculated a "months of supply" figure, defined as the number of properties currently available in a given category divided by the number of properties in that category sold in the last twelve months. This ratio gives a back-of-the-envelope calculation of how many months it would take to sell a given type of property assuming sales patterns over the last twelve months hold. "Months of supply" is a useful figure as it gives a sense of the market's appetite for a given type of property. It is widely understood within the real estate industry that properties with lower supply relative to demand (which the "months of supply" figure captures) tend to sell faster than properties with more supply, and are also more likely to sell at or above asking price than properties that have more supply. As such, "months of supply" calculations are widely used within the real estate industry for estimating both the length of time it will take to sell a property, and also the final sale price of that property.

Our preliminary EDA indicated that months of supply explains some of the price change from a property's list price to final sale price: properties in categories that are "undersupplied" (i.e. 0 or 1 months of supply only) sell on average very close to their (reported) list price; properties with more than 1 month of supply tend to sell below asking price, and the average sale price declines steadily as the amount of supply (measured in months) increases. If accurate list prices were recorded, these trends would likely be more extreme.



Polarity

We also used sentiment analysis to determine the attitude of the realtor in respect to the properties description available in the dataset (i.e. the field named "REMARKS"). Sentiment Analysis is the domain of understanding emotions with software, and it is a component of Natural Language Processing (NLP). Sentiment analysis aims to determine the attitude of a speaker, writer, or other subject with respect to some topic or the overall contextual polarity or emotional reaction to a document, interaction, or event. Given some text the algorithm returns a result, Porality, that is float which lies in the range of [-1,1] where 1 means positive statement and -1 means a negative statement.

Despite the fact that sentiment analysis has shown promising results in predicting human appreciation for stock market assets and other commodities, including Polarity in our model actually worsened our predictions. Thus we chose to disregard polarity in favor of other approaches.

Distance Matrix

The workhorse of our model was a distance matrix: a matrix of distances between all properties sold, that allowed us to estimate each property's final sale price from the previous sale prices of its neighbors.

In order to undertake this approach, we first had to fetch the latitude and longitude coordinates for all properties in our dataset. While geocoding is possible in Python, we took advantage of the speed and accuracy of ArcGIS to do so.

For every property listed, we used ArcGIS to geocode the address and return latitude and longitude coordinates. Finding latitude and longitude coordinates made it possible to conduct spatial data analysis that is impossible with the original MLS

dataset. We used each address' latitude-longitude pairs to construct the distance matrix, and proceeded to explore a range of different methods for integrating this matrix into our model.

Our spatial data analysis relied heavily on PySAL (Python's Spatial Analysis Library), developed by the spatial statisticians Luc Anselin and Sergio Rey. Anselin and Rey developed PySAL to allow data scientists to integrate a greater attention to space into their models. PySAL allowed us to quickly and efficiently construct a distance matrix, and also gave us the ability to query this matrix in order to find the nearest neighbors of a given point (even if it was not included in that original matrix).

Computational Bottleneck

While PySAL's distance matrix algorithms are streamlined to work with large matrices, constructing and querying a distance matrix are O(n^2) problems: the computational intensity and time required for doing so increase exponentially with each additional data point. Constructing a distance matrix on a PC for just one year's worth of Massachusetts sales data takes several hours.

We tested a number of different packages and approaches to reduce the computational intensity of this spatial approach. We tried parallelizing our code using OpenACC, and also explored Amazon Web Services (AWS) cluster architectures.

However, as explained below, our model only required us to construct a distance matrix once: to compute the distances between all properties in the 2016 and 2017 sales data. Thus a calculation of several hours was not a major limitation for this project. However, if this algorithm were to work on sales data generated in real time, or for a larger dataset, a more efficient approach (or, alternatively, the use of additional servers) would likely be necessary.

Image processing

We have also explored integrating properties' locational features with image processing of Google Streetview images, including through the use of Convolutional Neural Networks (CNN)[4]. We deployed the ResNet50 algorithm in this case, but found that the features it could detect were limited.

Mainly due to time constraints, we decided to concentrate our efforts on understanding the effects of location, rather than on unstructured data detected from properties' images.

Literature Review

Since the 1970's, a rich literature on "spatial autocorrelation" has emerged

[4] You, Quazeng, Pang, Ran, Cao, Liangliang and Luo, Jiebo, "Image Based Appraisal of Real Estate Properties." Jul. 2017.

across a range of different disciplines, detailing the fact that observations that are nearby in space tend to be correlated.[5] This reality has led to a wealth of new insights in the fields of quantitative geography and ecology.

Spatial autocorrelation can be incredibly useful in developing predictive models. For many phenomena, one of the best ways to predict a particular quality or outcome for point n is to observe other data points in the vicinity, and to treat data points that are closer to point n as more relevant than those that are further away.

This approach is so intuitive as to be obvious. However, real estate practitioners - including data scientists working on property price prediction - have been slow to make use of the power of spatial autocorrelation within their statistical models.[6] We have located only a handful of academic articles that attempt to use spatial statistics and spatial autocorrelation for the purposes of real estate price estimation and analysis - and in nearly all cases, these articles note the virtual absence of spatial approaches from mainstream computational real estate appraisal. The 2016 paper "Spatial Modeling and Geovisualization of Rental Prices for Real Estate Portals", delivered at the *International Conference on Computational Science and Its Applications*, notes that geocoded address data is rarely used in real estate analysis and price estimation, and in most cases serves merely to "pin" addresses on a digital map.[7]

A 1998 article by the real estate scholar C.F. Sirmans notes that "real estate has historically employed statistical tools designed for independent observations", while simultaneously observing that the assumption of independence is violated by the fact that residuals tend to be spatially clustered: such as in certain neighborhoods, along particular roads or near facilities such as airports.[8]

One way to control for the spatial autocorrelation of residuals is to bring location directly into regression analysis. One of the most popular means of doing so is through the "distance matrix": X and Y coordinates are gathered for all n observations, and then an n x n matrix is generated from the distances between each of those points. The matrix is then used to gather relevant data from neighboring observations.[9]

---

[5] Pierre Legendre, "Spatial Autocorrelation: Trouble or New Paradigm?," Ecology 74, no. 6 (1993): 1659–1673.

[6] Pace, R. Kelley, Ronald Barry, and C. F. Sirmans. "Spatial Statistics and Real Estate." *The Journal of Real Estate Finance and Economics* 17, no. 1 (1998): 5–13.

[7] Schernthanner, Harald, Hartmut Asche, Julia Gonschorek, and Lasse Scheele. "Spatial Modeling and Geovisualization of Rental Prices for Real Estate Portals." In *International Conference on Computational Science and Its Applications*, 120–133. Springer, 2016.

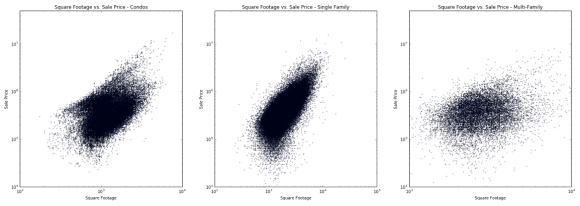
[8] Pace, R. Kelley, Ronald Barry, and C. F. Sirmans. "Spatial Statistics and Real Estate." *The Journal of Real Estate Finance and Economics* 17, no. 1 (1998): 5–13.

[9] Robin Dubin, Kelley Pace, and Thomas Thibodeau, "Spatial Autoregression Techniques for Real Estate

The writings of Luc Anselin and Sergio Rey, the two quantitative geographers who developed PySAL, have been integral in providing the methodological foundations for our research.[10,11] PySAL's extensive and detailed documentation has allowed us to develop a basic proficiency with the library's capabilities and applications over the course of this project.[12]

Modeling Approach

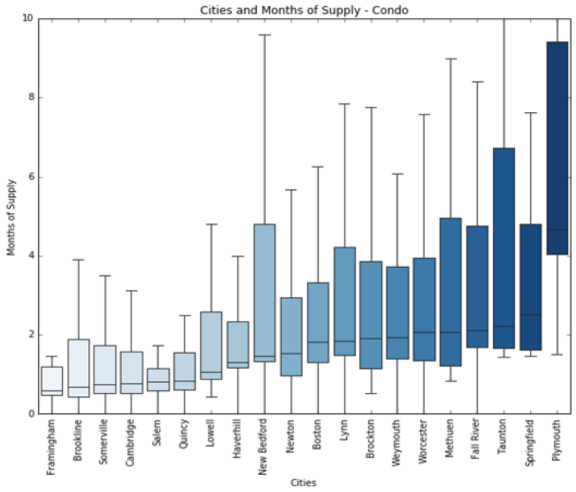
Our research began by trying to understand the different relationships that features have with sale price through EDA. Not surprisingly, square footage was one of most important physical features predicting sale price; the relationship between square footage and sale price varies by property type, as the following graphic shows.



Square footage has the most clearly linear relationship with sale price for single family homes and condos. The relationship between square footage and sale price is less obvious for multi-family properties.

As described above, before conducting serious regression analysis we engineered a number of additional features that we figured could be useful in predicting home price: months of supply, categorical variables for each home style and amenity, and polarity based on the textual descriptions of each property.

We calculated months of supply by categorizing properties into their respective property types, cities, and bedroom counts. The graphic below shows the way in which months of supply varies by city: as you would expect, Cambridge, Brookline, and Somerville are very undersupplied – the market is tight in these cities.



While these engineered features helped improve our baseline somewhat, they failed to raise the R^2 of our model

---

Data," Journal of Real Estate Literature 7, no. 1 (1999): 79–95.

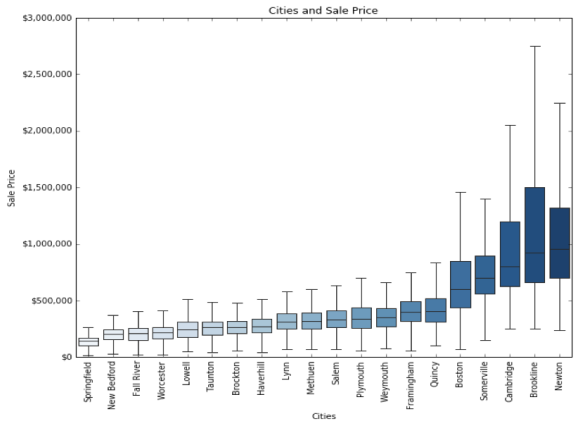
[10] Anselin, L. *Spatial Econometrics: Methods and Models*. Springer Science & Business Media, 2013.

[11] Rey, Sergio J., Luc Anselin, Xun Li, Robert Pahle, Jason Laura, Wenwen Li, and Julia Koschinsky. "Open Geospatial Analytics with PySAL." *ISPRS International Journal of Geo-Information* 4, no. 2 (May 13, 2015): 815–36.

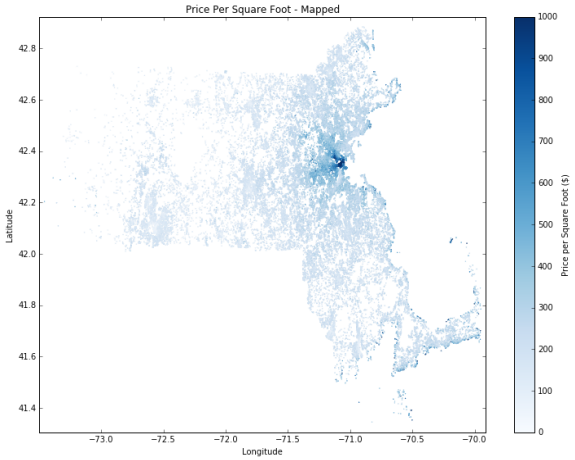
[12] Pysal Developers. "Pysal Documentation: Release 1.14.3". April 14, 2018.

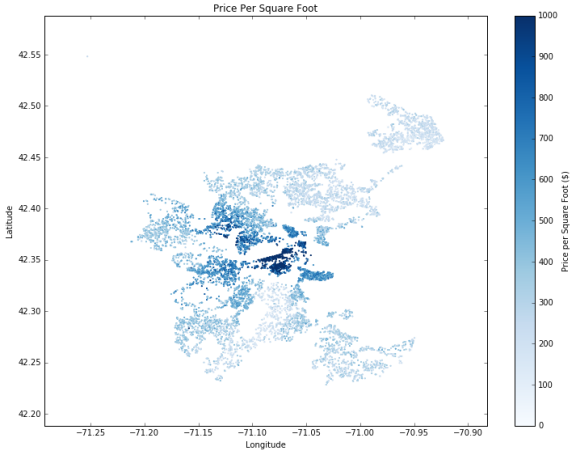
above 68%, which we felt was an inadequate level.

Given the limited usefulness of these engineered features, we decided to focus our attention on location. One of the most important insights our initial exploratory data analysis yielded was that unit price per square foot varies considerably across different geographical areas; this spatial pattern in home price per square foot (and therefore home prices) provided the inspiration for our decision to take a spatial approach to this project.



However, the spatial pattern in home prices is too fine-grained to be adequately captured by simply partitioning the data into different cities or zip codes. The following graphics map the variation in price per square foot across Massachusetts and also within the Boston Metropolitan Area. This spatial pattern is not adequately captured by county or city boundaries alone. Hence we chose to use an approach more sensitive to locational variation - the distance matrix.





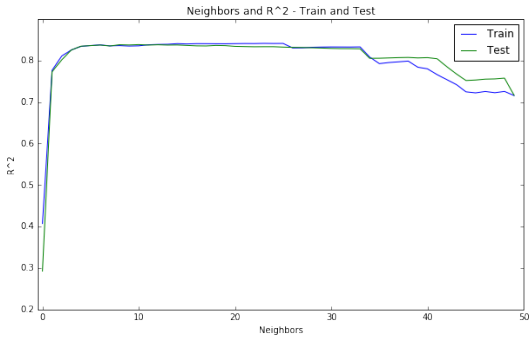
After exploring a variety of different means to employ the distance matrix in our analysis, we settled on one fairly intuitive method: use the distance matrix of 2016 sold properties to find the nearest neighbors of all 2017 sold properties (of the same property type), and attempt to use those 2016 nearest neighbors to predict 2017 sale prices. PySAL allows users to “query” existing distance matrices for the desired number of neighbors of a given point (even if that point is not contained within the original distance matrix), without re-constructing the entire matrix. Thus it is fairly quick to query even a large matrix for the k-nearest neighbors of a given point, and it is possible to loop through a

very large list of points to find the k-nearest neighbors of each of those points.

Our intuition was to create a predicted price for each 2017 sold property by using characteristics from these neighbors. For each k neighbor, we multiplied the sale price per square foot times the square footage of the corresponding 2017 property, yielding a list of k predicted prices for each 2017 property. We then averaged these k predicted prices, and added these n predicted prices as a field to our 2017 sold properties dataset. We then used this new, engineered field as a predictor alongside the other, baseline predictors such as “Beds”, “Baths”, and “Age”.

Since we were unsure how many neighbors would be optimal to construct this predicted price, we started high: we wrote a for-loop to fetch the 50-nearest neighbors from the 2016 dataset for each 2017 sold property. Our loop fetched both the price per square foot and the distance from the 2017 property for each of these 50 neighbors.

In order to determine the optimal number of neighbors, we wrote a series of algorithms to split the 2017 data into test and train and compute the train and test R^2 of this new regression model for each combination of neighbors between 0 and 50. The following graph shows the dramatic increase in R^2 that results from including just a handful of neighbors.



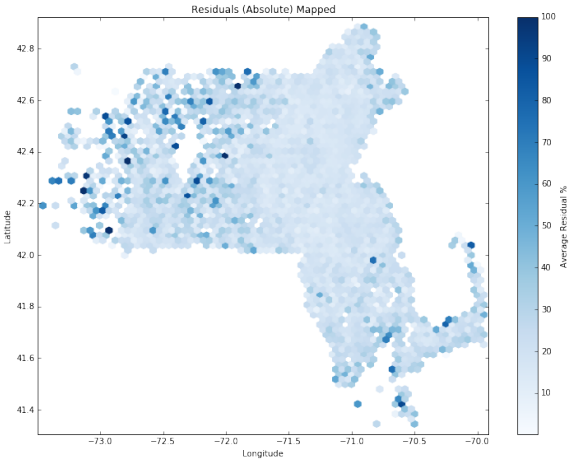
The R^2 of the model increases rapidly from 1 to 5 neighbors, and plateaus from 6 neighbors onwards. Beyond 30 neighbors, the accuracy of the model starts to degenerate: including increasingly distant neighbors drags down the accuracy of the model, since properties that are further away are typically less relevant for predicting a given property than those that are closer. Depending on the particular partition into train and test, we found that the optimal number of neighbors lay somewhere between 8 and 20.

The baseline model these predictions were added to was fairly simple: Beds, Baths, Age, SQFT, Garage, and Months of Supply. With just this set of predictors plus the “predicted price” field generated by the ~10 nearest neighbors, the resulting R^2 was approximately 85%.

We then spent days looking for ways to improve on this model, to little avail. We attempted to weight the average of neighboring predicted prices by their distance from the 2017 property. This failed to raise the R^2 at all. We attempted a generalized additive model,

since we suspected certain relationships to be non-linear. This yielded a low R^2. We tried building polynomial and interaction terms for all of our features - in each case, this either failed to raise the R^2 or lowered it on the test set.

Given our failure to raise R^2 through conventional means, we decided to analyze our residuals. What we found was revealing: the model appeared to perform well in the dense, populous region around Boston; high residuals were concentrated in the sparser, less dense areas of Western Massachusetts and Cape Cod.



Upon reflection, this is not surprising: using a method that finds ~10 nearest neighbors is likely to work better in dense places, since those 10 neighbors are likely to be very close to the property in question. In a sparsely-populated rural area, the 10 nearest neighbors might be over a mile away on average, and thus would be less relevant for predicting a given property's sale price. While we have not yet attempted this, having the number of neighbors included vary by location could possibly improve the accuracy of the model further.

Given the clear spatial pattern of the residuals, we also figured there might be other spatial factors at play. It is widely recognized that properties in many neighborhoods of the Boston metropolitan area are appreciating quickly. We figured including average year-over-year appreciation by property type and city could help improve our model's accuracy. We chose to calculate this figure internally using the data provided, yielding a % annual appreciation for each city and property type from 2016 to 2017. We then used this percentage to adjust the prices predicted by the k-nearest neighbors. This raised the R^2 slightly, to nearly 87%.

However, we recognize that there are problems with this method, since the limited number of sales in certain cities makes it impossible to accurately calculate appreciation, and also because the 2017 figure necessarily includes the 2017 properties we are trying to make predictions for.

Project Trajectory, Results and Interpretation

In summary, our initial exploratory data analysis made clear to us the importance of location in predicting home prices. This intuition was underscored by the fact that our early

feature engineering of months of supply and “Other Features” yielded a disappointingly low R^2.

Our choice of a distance matrix was motivated by a desire to integrate a more granular understanding of distance than simple dummy variables for City and Zip Code can provide.

Including the predicted prices generated by the ~10 nearest neighbors alone yielded an R^2 of around 80%. In other words, about 80% of the variation in home prices can be explained by the sale prices of neighboring properties alone.

When these predicted prices were combined with the simple baseline model mentioned above, the R^2 rose to 84%. And when predicted prices were adjusted by the 2016-2017 percent appreciation within each city, the R^2 rose to 87%.

Conclusions and Possible Future Work

There are definite ways this model could be improved. Obviously, our model is mostly designed to be sensitive to locational effects - to a degree, the use of neighbors can be seen as a proxy for a whole range of locational features. By contrast, our model is for the most part agnostic to specific, qualitative home features. While we did build dummy variables for all features and home styles, including these variables yielded no appreciable increase in R^2. Likely a better way to account for these qualitative, home-specific features would be image recognition algorithms - an approach we did not pursue in any serious way. Ensembling our locational approach with such a home-specific method would likely make up for much of this model’s current weaknesses.

In additional, some sort of hierarchical approach would likely also increase this model’s performance. The graph of neighbors and R^2 shown above shows the way in which R^2 increases rapidly with an increasing number of neighbors and declines beyond 30 neighbors. But there is no reason to assume that the optimal number of neighbors is uniform across all locations and housing categories. Allowing this optimal number to vary by location (and perhaps by other qualities) would quite possibly improve the model’s accuracy.

Bibliography

Anselin, L. *Spatial Econometrics: Methods and Models*. Springer Science & Business Media, 2013.

Basu, Sabyasachi, and Thomas G. Thibodeau. “Analysis of Spatial Autocorrelation in House Prices.” *The Journal of Real Estate Finance and Economics* 17, no. 1 (1998): 61–85.

Dubin, Robin A. “Spatial Autocorrelation and Neighborhood Quality.” *Regional Science and Urban Economics* 22, no. 3 (1992): 433–452.

Dubin, Robin, Kelley Pace, and Thomas Thibodeau. “Spatial Autoregression Techniques for Real Estate Data.” *Journal of Real Estate Literature* 7, no. 1 (1999): 79–95.

“How Accurate Is Zillow’s Zestimate? Not Very, Says One Washington-Area Agent. - The Washington Post.”

Zillow Inc. “What Is a Zestimate? Zillow’s Zestimate Accuracy.” Zillow. Accessed May 1, 2018.

Legendre, Pierre. “Spatial Autocorrelation: Trouble or New Paradigm?” *Ecology* 74, no. 6 (1993): 1659–1673.

Pace, R. Kelley, Ronald Barry, and C. F. Sirmans. “Spatial Statistics and Real Estate.” *The Journal of Real Estate Finance and Economics* 17, no. 1 (1998): 5–13.

Pysal Developers. “Pysal Documentation: Release 1.14.3”. April 14, 2018.

Rey, Sergio J., Luc Anselin, Xun Li, Robert Pahle, Jason Laura, Wenwen Li, and Julia Koschinsky. “Open Geospatial Analytics with PySAL.” *ISPRS International Journal of Geo-Information* 4, no. 2 (May 13, 2015): 815–36.

Schernthanner, Harald, Hartmut Asche, Julia Gonschorek, and Lasse Scheele. “Spatial Modeling and Geovisualization of Rental Prices for Real Estate Portals.” In *International Conference on Computational Science and Its Applications*, 120–133. Springer, 2016.