Using Geospatial Data to Find Investment Opportunities in the Real Estate Market

Chih Huang
huangcs
@seas.upenn.edu

Ignacio Navarro
inavarro
@seas.upenn.edu

Nikhil Ramesh
nramesh
@seas.upenn.edu

Abstract

Valuating real estate properties is driven by a series of direct factors (total square area, number of bedrooms, garage spaces, etc) and indirect factors (crime rate, quality of school district, etc). However, private sellers tend to focus more on the former when setting a listing price for their property. In particular, there are many geospatial features indirectly related to the property that sellers might easily overlook. Proximity to a church, a river, or a police station are all factors that play a role, albeit minor, in the valuation of a property. In this paper we determine by running a series of models on a dataset consiting on properties in South Philadelphia if geographical features do indeed play a decisive role in the valuation of a property.

1 Introduction

1.1 Motivation

Traditionally, real estate valuation has concentrated on assessing the features directly surrounding the property to come up with a price. Straightforward features such as number of bedrooms, whether the property has a pool or a garage space, or even the number of bathrooms play a major contribution on the price. And while that does play an important role, private sellers may not fully realize that the property's surroundings can also play a role. Points of interest like supermarkets, museums, parks, cafes, or police stations can affect house prices but many times these may be overlooked.

1.2 Problem Definition

With all of this in mind the natural question becomes: given all the geospatial data related to a property, can a machine learning model predict the real value of a property as a **regression** problem? And if so, can one use this to find investment opportunities in the real estate market? Throughout this paper we will answer the former question. The latter is a simple corollary of the former. Indeed, if there is a model that can accurately predict house prices given geospatial and direct features, one can use these results to find investments opportunities by comparing the listing price on properties currently on the market with what the best model predicts the value is and seeing if the property is really undervalued.

In other words, let $M : X \to \mathbb{R}^+$ be an accurate model (we will define what constitutes "accurate" to our use case in the next section), where X is the feature space that constitutes direct and indirect features related to a property. Given a property p with a listing price of s_p and a set of features x_p, we will buy the property if

$$M(x_p) > s_p \tag{1}$$

because p is undervalued.

1.3 Related Work

A lot of work has been done in the last few years on predicting house prices using machine learning. Traditional models have relied in hedonic regression, a type of regression which breaks the property apart into its direct factors in order to establish a relationship between the factors and the price of the property. A paper using this method have been presented by (Jiang et al., 2014) focusing on properties in Singapore with property sales between 1995 and 2014.

Our main source for our project is (Bergadano et al., 2019), in which they analyze a dataset consiting of properties in Madrid, Spain. There are similarities to what we're trying to investigate: the

paper models the question as a regression problem and uses similar models to the ones we train. The main difference, however, is that they only consider direct features for their models. The main novelty in our project is to combine these features with geospatial features. As far as we have researched, no work has investigated our proposal.

2 Data Acquisition and Exploration

2.1 Acquisition

The acquisition of the dataset for this project consisted on finding basic property information for Philadelphia that included direct features, and merging this dataset with all the geospatial information related to the property. The first part was relatively easy, as `https://www.opendataphilly.org/` provided basic information for over 250,000 properties in Philadelphia. The main challenge was in fact finding the geographical features. Our approach to solve this was to find APIs that offered nearest distance to points of interests. Once we found some reliable APIs for this task (e.g. GoogleMaps API, TomTom API, or FourSquare API), we began the process of merging the data to our property dataset.

The main obstacle of this part, and probably of the project, was the API rate limit each service offered. Conservatively speaking, we needed to make

$$\underbrace{250,000}_{\text{properties}} \times \underbrace{24}_{\text{geo features}} = 6,000,000 \text{ API calls}$$

and unfortunately most of the services offered 1,000 API calls limit per day. The solution to this part was to drastically reduce the size of the dataset to around 8,000 properties in South Philadelphia, and even this amount posed some architectural challenges: we distributed the API calls for each member of the group and remerged them later. We encourage the reader to look at the code dealing with this on the project's repo, in particular the `properties/data` package.

2.2 Features

The list of final features that we feed our model are in Table 1 for a total of 5 categorical and 28 numerical (nearest point of interest in meters).

Feature	Type
Fireplaces	categorical
Garage Spaces	categorical
Bathrooms	categorical
Bedrooms	categorical
Stories	categorical
Total Area	numerical
Total Livable Area	numerical
Latitude	numerical
Longitude	numerical
Nearest Museum	numerical
Nearest Gas Station	numerical
Nearest Coffee Shop	numerical
Nearest Stadium	numerical
Nearest Food	numerical
Nearest Bar	numerical
Nearest Gym	numerical
Nearest Bridge	numerical
Nearest Garden	numerical
Nearest Park	numerical
Nearest River	numerical
Nearest City Hall	numerical
Nearest Police Station	numerical
Nearest Hospital	numerical
Nearest Elementary School	numerical
Nearest Church	numerical
Nearest Bank	numerical
Nearest Supermarket	numerical
Nearest Pharmacy	numerical
Nearest Bus Stop	numerical
Nearest Metro Station	numerical
Nearest Train Station	numerical
Nearest University	numerical
Nearest Laundromat	numerical

Table 1: Features used

2.3 Exploration

Once we acquired the data the next step was to explore it to fully understand it. Some significant

results to show are the following.

The distribution of the properties by sale price in South Philadelphia can be seen in Figure 1. Clearly the distribution is skewed with a mean sale price of $324,662.

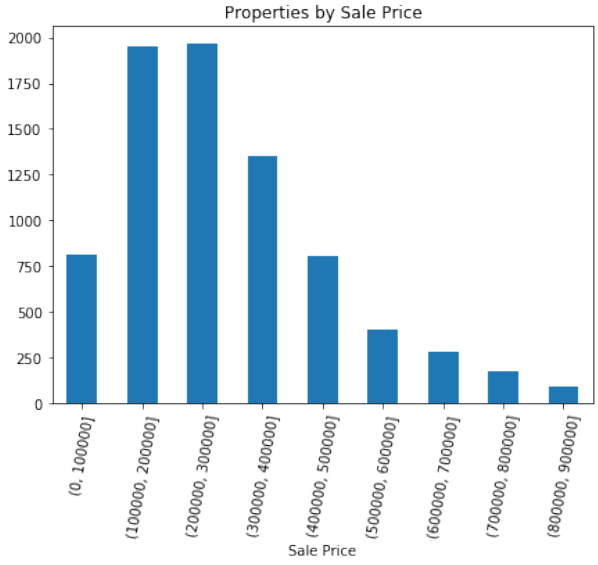


Figure 1: Distribution by sale price

We can also get a sense of where the properties are located by plotting a heatmap of South Philadelphia based on sale price (Figure 2).

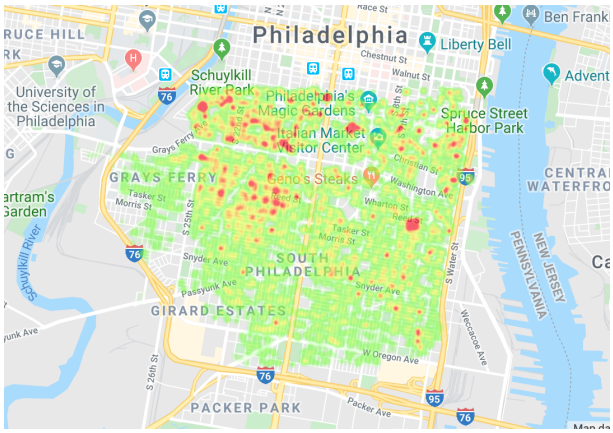


Figure 2: Heatmap of South Philadelphia (hotter means higher sale price)

Another important step was to see the relationship between the features. To do this we used a correlation matrix which can be seen in Figure 3. For instance, there is a strong correlation between the number of bathrooms and the total livable area. There's also some interesting relationships, e.g., there is negative correlation between the total livable area and the nearest laundromat, which makes (some) sense, as bigger houses in richer neighborhoods don't need laundromats.

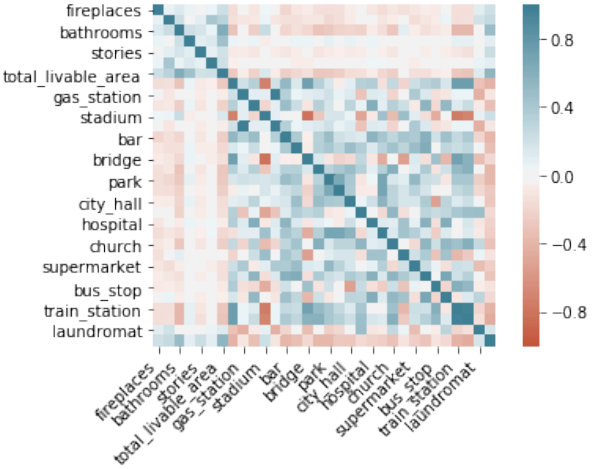


Figure 3: Correlation matrix

3 Models

3.1 Experimental Models Setup

We wanted to formulate this as a regression problem by using **sale price** as a target label and training on the features collected for our housing dataset. To do this, we decided to train the following 5 different regression models: Support Vector Regression, Linear Regression, Multi-Layer Perceptron, Regression Tree and KNeighbor Regression.

From here, we wanted to evaluate our model in a meaningful way without looking at an MSE Error value that is visually insignificant to us. Additionally, we needed a relative error metric since a $50,000 error on a $100,000 property is much worse than a $50000 error on a $1,000,000 property. We used our own error metric which classifies a prediction as good if a predicted sale price is within a given tolerance (e.g. 20%) of the ground-truth sale price and bad otherwise.

We initially trained one model on the entire training dataset but saw that this treated the higher priced houses as outliers. To account for this, we segmented the training data into three segments (cheap, moderate, expensive) and fit a separate model for each segment, resulting in 3 models to cover the entire dataset. This works because our project aims to predict how much a house on the market is actually worth; we should featurize a currently listed house and then based on its list

price set by the owner, we place it into the "cheap", "moderate" or "expensive" bucket; then, the corresponding model predicts its worth so we can see if it is being undervalued or overvalued. The key assumption we make here is that the predicted *Sale Price* should not deviate from the bucket that the *List Price* was placed into. We see now our experimental findings below.

3.2 Results and Findings

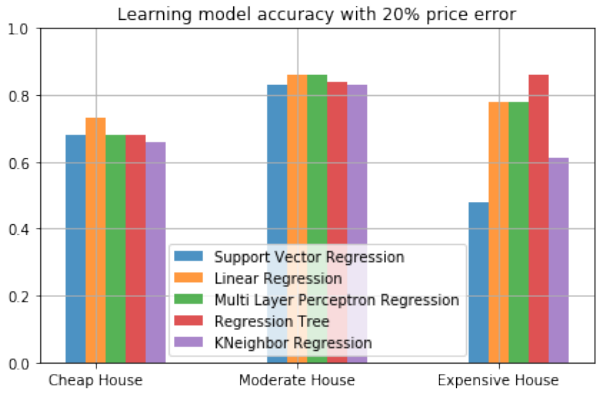


Figure 4: Accuracy comparison for five learning models which are used to predict the cheap, moderate, expensive properties.

Using the geospatial features and direct factors (number of bedrooms, spaces, etc) to train five learning models with a 20% tolerance for each model is listed in Figure 4. According to the result, the five models have almost the same accuracy to predict cheap properties and moderate properties. However, there we consistently see three models (Linear Regression, Multi-Layer Perceptron Regression and Regression Tree) that have better accuracy to predict expensive houses.

3.3 Finding the Best Model

We took these three models and retrained using a lower error tolerance (15%). The result is shown in Figure 5.

The accuracy decreased with a tolerance of 15% for three models, and by inspection we see that Linear Regression and Regression Tree model both have a generally better accuracy than Multi-Layer Perceptron Regression. Thus, we decided to look further into those models in particular and how they distributed their predictions to see if there were similarities to the ground truth test data. We analyzed the prediction data and test data distribution to visually see the resemblance between

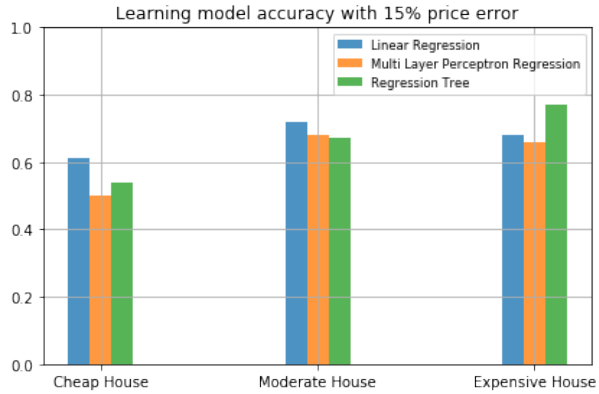
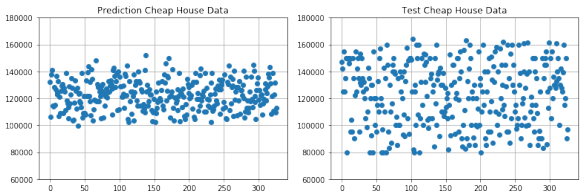


Figure 5: Accuracy comparison for three learning models which have better performance in previous prediction result.

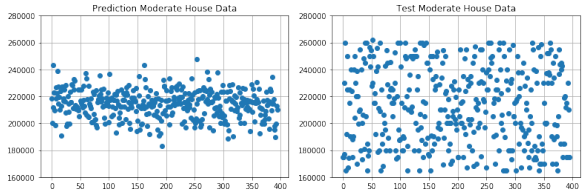


(a) Prediction data for cheap house (b) Test data for cheap house

Figure 6: Prediction and test data distribution comparison for cheap house by using linear regression model.

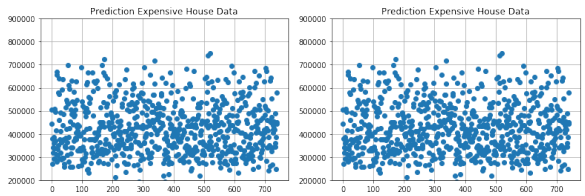
the prediction and ground truth test data given a model. For instance, the prediction data distribution are shown in Figure 6, Figure 7, Figure 8 for different models.

According the results shown in the figure, the Linear Regression model predicts the cheap and moderate properties prices into a smaller range than that of the test data, meaning the results don't necessarily reflect the test data well. If you see the results given in Figure 8, you can see that this Linear Regression model does give a more accurate representation/distribution of the test data.



(a) Prediction data for moderate houses (b) Test data for moderate houses

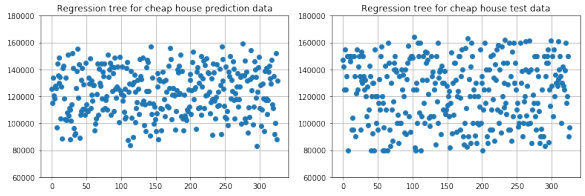
Figure 7: Prediction and test data distribution comparison for moderate houses by using Linear Regression model.



(a) Prediction data for expensive house (b) Test data for expensive house

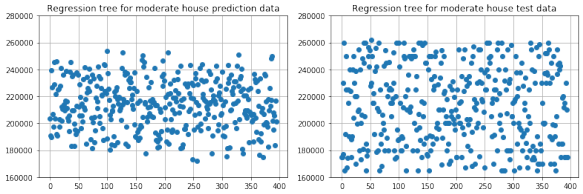
Figure 8: Prediction and test data distribution comparison for expensive house by using linear regression model.

After comparing the distribution for the training data and the test data by using Linear Regression model, we then examined the data distribution result by using Regression Tree model. The data distribution is listed in Figure 9, Figure 10, Figure 11. We can see that the distribution of the predicted house prices is pretty consistent with that of the ground truth data for all three buckets of property price (cheap, moderate and expensive).



(a) Prediction data for cheap house (b) Test data for cheap house

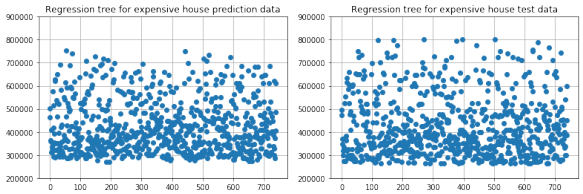
Figure 9: Prediction and test data distribution comparison for cheap house by using regression tree model.



(a) Prediction data for moderate house (b) Test data for moderate house

Figure 10: Prediction and test data distribution comparison for moderate house by using regression tree model.

By examining the models and analyzing the accuracy as well as the prediction distributions compared to the ground truth test data, we feel that the Regression Tree model most accurately captures the concepts in our data and will be the best to predict on houses on the market in the real world using our collected geospatial features.



(a) Prediction data for expensive house (b) Test data for expensive house

Figure 11: Prediction and test data distribution comparison for expensive house by using regression tree model.

4 Conclusion

From our models we were able to see that Regression Tree was able to perform the best from training on our features. Under the hood, we checked to see which features it prioritized and were able to find that the geospatial attributes that the most important were 'bus_stop', 'laundromat', 'university', 'park', 'river', 'train_station' and 'stadium'. These were in some cases even more important than the direct features. This shows that the collected geospatial features do have an impact in determining housing prices as our best models weighted them heavily.

Our most important finding here was the model's improvement in accuracy while using geospatial features over using only the direct features (such as livable area, bedrooms, etc.). We

found that the test accuracy increased on average by around 30% just by adding these new features. This gives us the result we were looking for: geospatial features do have a significant impact on property sale price.

Despite this finding, our models trained on the geospatial features did not perform well enough to accurately predict the true value of housing prices as the accuracy only reached about 70% on relatively high tolerance thresholds of 15-20%. This could be because of hidden variables that overpower our collected features, such as crime rate, quality of school district, etc. that we don't consider. As such, we conclude that geospatial features do have an impact on property sale prices but are not sufficient on their own to predict house prices, although it's a great initial feature space to investigate and adding additional types of features would only improve a resulting model's usability in real-world scenarios.

4.1 Future Work

Some work that can be done on this project includes adding these hidden factors such as crime rate and school district quality that may increase the effectiveness of our model as well as the usefulness of the geospatial features we've provided.

Additionally, we can try some more powerful regression models that may be able to capture the solution to our problem better; for this project, we used the same models as the ones in the paper for Madrid housing (Bergadano et al., 2019) but we may be able to do better.

Lastly, we can also compare our trained model with an existing industry model such as Zillow's and see how close we are to emulating their price prediction.

References

Francesco Bergadano, Roberto Bertilone, Daniela Paolotti, and Giancarlo Ruffo. 2019. Learning real estate automated valuation models from heterogeneous data sources.

Liang Jiang, Peter CB Phillips, and Jun Yu. 2014. A new hedonic regression for real estate prices applied to the singapore residential market.