Real Estate Appraisal

CS229 Machine Learning Final Project Writeup

David Chanin, Ian Christopher, and Roy Fejgin

December 10, 2010

Abstract

This is our final project for Machine Learning (CS229) during the Autumn 2010 quarter. Our group attempted to predict house prices using various real estate, census, and stock market data. As a side goal, we also attempted to predict a number of real estate market trends by area, including the projected inventory (number of homes for sale), average listing price, and average time on the market for homes in the area. Ultimately we were able to predict individual house prices with an average relative error of 24% which compares with industry leader Zillow.com. We were also able to predict market trends with between 5% and 25% error depending on the trend.

1 Introduction

The very first motivating example of the quarter was real estate appraisal in Portland. Using information on house size and the number of bedrooms, linear regression was developed to determine a relationship between these variables and the corresponding house's selling price. Our project is very much in the spirit of this elementary example though it employs more advanced techniques, makes use of more comprehensive features, and seeks to predict not just current housing prices but trends in the market over time using.

2 Data

Finding comprehensive data was, unfortunately, not a simple task. We partnered with a realtor in Georgia who provided us with a MLS (Multiple Listing Service - read 'actual real estate data') connection which covers current and historical housing listings throughout the state of Georgia since 2006. This source contained over 1 million listings, however only about 50 thousand or so were suitable for this study.

We are also taking data from several public US Government sources. In particular, we are using demographic, population, and geographic data from the US Census and unemployment data from the Bureau of Labor Statistics (BLS). We are also using education data from the National Center of Education Statistics (NCES) and crime rate data from the FBI. Additionally, we used stock market data from Yahoo Finance. After adding all of this data together, we ended up with a total of slightly over one hundred potential features ranging from house-specific data such as number of bedrooms to aggregate data such as population, crime rates, teacher/student ratios, and average house price in the area.

2.1 Issues

Despite the amount of data we collected, there are inherent limitations. For instance, the selling price of a house is highly influenced by the terms of payment between the buyer and seller (mortgage payments, cash down, etc.) [Eld06]. Furthermore sellers could also be under external pressure to sell quickly (such as a new job across the country) or buyers could be simply misinformed. There are a number of other factors, including highly subjective factors, which go into determining a house's selling price which are not captured by MLS data.

Our data is missing some more basic real estate features as well. In particular, square footage, one of the key determinants of house price, was not usable for our calculation since the majority of realtors simply didn't fill that data out. Lot size was also missing from our data. According to our sources, lot size could have anywhere between twenty and eighty percent impact on final price [Eld06].

Despite lacking these features, we are still able to make competitive predictions of housing price by making use of neighborhood statistics and spatial features of listings. While data is an issue for our predictions of individual house prices, the features which are listing-specific should not have a significant impact on our prediction of overall market trends.

2.2 SQL Database

As an intermediate step, we are imported all our data into a relational database and geocoded all of the listings in our dataset. This was necessary so that we could perform geographic queries and generate spatial features for use in our algorithm. After the data was properly formatted and assembled we imported the data into MATLAB to perform actual calculations.

3 Our process

This section describes our process as we moved from one stage of development to another. It is meant to illustrate what we did and the thought process behind it.

3.1 Initial setup

Real estate was a relatively new area to all of us when we first started the project. To compensate for this lack of experience, we researched the field online and in literature. From this, we gained basic real-estate know-how and also were able to create a list of features that we though would be most important to collect. We also realized that there would be inherent limitations on our accuracy due to the subjective nature of real estate.

We also met with Billy McNair, a real estate broker from the McNair Group in Menlo Park. He gave us advice on predicting housing prices and also suggested we look into predicting real estate market trends over an area. According to him, Zillow.com is the industry leader for predicting house prices but it has about an average twenty to thirty percent relative error so it seemed like there was room for us to improve on. Furthermore, he helped give us a sense of the overall state of real-estate market analysis and how our project could be a valuable tool for realtor estate professionals.

3.2 Raw linear regression

Our first stab at prediction was running linear regression using the normal equations on roughly seventy hand selected features from our raw data set. Not surprisingly this did not produce the most promising results. For a majority of our data points this regression did fairly well, but there was a group of outliers that our model had huge errors predicting.

In particular we discovered that there were several outliers/mistakes in the data; there were several houses listed at much lower prices than expected in the data set. Some of these house prices seemed like clear human errors (according the data a few houses sold for one dollar) and others just seemed like they were selling for too low of a price. We attribute this underselling to factors that cannot be determined from the data. For example, a wind storm might have severely damaged a house but this would not be represented in our data set. General house upkeep is yet another unknown feature that could have big effects on house price. To handle these outliers, in our preliminary stages we ignored houses with relative errors in the top one percent of the test examples.

3.3 Initial data cleaning

Based on the houses that generated huge prediction errors, we started to analyze our data for problems. As an initial way to get rid of these data points, we set logical limits on ranges and excluded data points outside of them. For example, we had a number of houses (100) that were supposedly built after the year 4000. Accordingly we set a hard limit and only analyzed houses that were reportedly built between 1850 and 2010. We had a number of other hard limits like this on other features.

Additionally, we experimented with analyzing each feature individually to determine its distribution, discarding rid of data points that had features that were more than four standard deviations from their mean. Eventually, though, we abandoned this method in favor of more sophisticated anomaly detection techniques (see below).

3.4 Weighted linear regression

Because real estate prices are so localized, we suspected that weighted linear regression would help improve our results. Of this course, this meant having to define a weight matrix and train on the entire training data set but ultimately we needed this to get better results. As for the weight matrix itself, we had to define custom similarity metrics between data points and also had to determine an optimal tau using trial and error.

3.5 Move to iterative solvers

The move to weighted linear regression helped our prediction accuracy but it also significantly increased the time it took for us to test. To deal with slow down and an ever expanding feature set, we started to look to iterative solvers for our normal equations. After a bit of experimentation, we determined that the tradeoff in accuracy was well worth the speed it took for these methods to work. In the end we settled on MATLAB's BiCGStab implementation.

3.6 Feature normalization and tweaking

In addition to all of this, we knew that linear regression would not be able to recognize all the patterns in our data. In particular certain features would be useless to linear regression without modification, but can result in a large performance gain when first preprocessed or combined with other features. For instance, stock price a month before the sale and 6 months before it might not mean much as features, but if we created a running average, their difference might indicate the markets strength. Another example would be using the log of the price instead of just price because of a pattern we saw in the data. Ultimately feature tweaking led to significant improvements though it also took significant time to do.

3.7 K-Means clustering

As our data set continually grew, we were once again faced with increasingly slow testing times. To deal with this, we decided to cluster our data entirely based on geographic location and then just train and test on individual clusters instead of the entire data set. Of course this did start to introduce edge cases within clusters, but for the most part our accuracy was unaffected (only eight total clusters).

3.8 Anomaly detection

As we continued to analyze houses, we found more and more unusual data. Whether it was finding issues in construction date or seemingly impossible prices (a house that was probably sold for $170,000 was entered as $17,000) it just seemed like our data was still not reliable enough. Over time we realized that our naive cleaning earlier was not sufficient so we opted for a more comprehensive anomaly detection method.

Due to its ease of use and power, we opted for a one class SVM classifier using libsvm's MATLAB interface. After playing around with a few SVM parameters we were able to produce much better results, though we could not necessarily prove that the data points that we removed were exceptions.

3.9 Bayesian regression

One main issue with the regression techniques we used to estimate housing prices is that they require all test and training data to have a value for each feature. However, since most fields from real-estate data from MLS are optional, this means the majority of fields and listings present in the raw MLS database had to be thrown out. Naive Bayes has the advantage of being able to work with data where not all features are fully filled out, and as such we were able to use Bayes with a much larger set of features and listings than we could using linear regression.

A major drawback to Bayes, however, is that it operates on discrete valued data and its output is discrete valued as well. Almost all of the fields in our database are continuously valued and had to be discretized to work with the Bayes algorithm. This introduced a large amount of discretization error and the predictions made by our Bayes algorithm were not nearly as accurate as those obtained using weighted linear regression.

3.10 SVM regression

Another algorithm we explored was SVM regression ('SVR'), once again using the libsvm implementation. Though it was simple to obtain reasonable initial results with this algorithm (roughly 80% error), improving on that proved difficult. Though we spent some time trying different kernels and algorithm parameters, the results never approached the accuracy of weighted linear regression. We believe this happened for several different reasons. First we couldn't use our custom distance model in SVM space so location was no more important than having a pool for instance. Secondly, these kernels may not have been complex enough to model the data that we were giving it. What ever the reason, SVM regression did not produce as good of results as we were hoping for.

3.11 Additional predictions

We also used weighed linear regression to predict market trends by school district. School districts were chosen since they are a fine-grained geographical area which is officially defined by US Census data, and it makes sense that listings within a school district would be priced similarly.

Within each district, we go month by month from 2006 until 2009 and create a separate input vector for each month which contains at the inventory in that district, the average listing price, and the average time on market centered on that month from 1 year into the past until 1 year in the future. A separate value is included for every 2-month period in that time span. For example, an input vector corresponding to district 01 centered on March 2007 will have entries for each market trend in district 01 from march 2006 until march 2008. The future values (values after March 2007) are, of course, what we're trying to predict. The input vector also includes rate of change information for each trend over the past year as well as US Census data for the district.

The weighting is based on geographic distance between districts as well as the overall number of listings in the district. Weighted linear regression is performed separately for each future prediction point and each trend type (e.g. inventory in the district two months from now, four months from now, six months from now, etc.). In general, this algorithm performed best when tau is small, so each district essentially performs its own linear regression with little input from how trends in other school districts perform.

3.12 Results and Discussion

We are able to achieve an average relative error of 24% for our test examples when predicting housing price, however this is with significant outlier removal and clustering. By comparison, Zillow.com, which is arguably the best automated mass housing price predictor, typically has a relative error of around 20% - 30%. As such, 20 - 30% error may be about as accurate as is possible given the limitations of what is expressed in MLS data. In order to improve the accuracy further, we may have to find better sources of data.

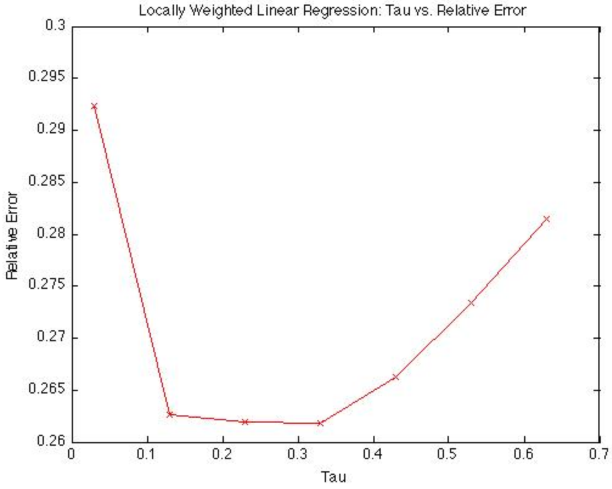
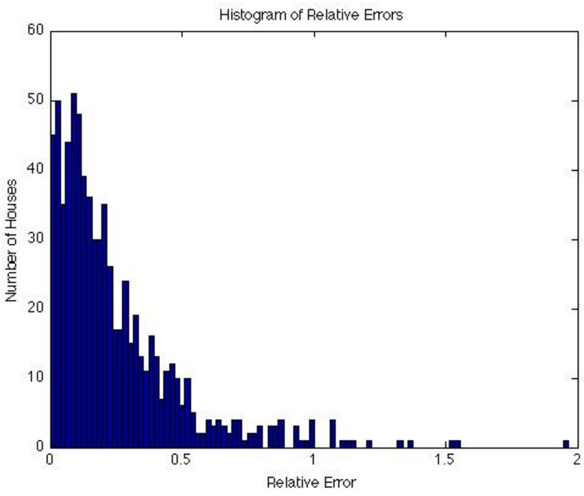


Figure 1: Weighted regression's tau.

As far as market trend predictions go, we are able to achieve an average relative error between 1% and 5% for future inventory, between 5% and 20% for future average price, and between 10% and 30% for future average time on market. In all cases, the lower error number corresponds to predictions for the near future, and the higher error number corresponds to predictions towards a year in the future. We also compared these relative error rates to error rates based on simply guessing that each trend will maintains its current value for the rest of the year, and in all cases our weighted linear regression prediction performed with between 3 and 5 times less error.

The market trends predictions are not fundamentally limited by the available data like the housing price prediction is, so it should be possible to get this prediction to be even more accurate. For instance, we are not taking into account the seasonally cyclic nature of the market at all in our prediction, and we can try incorporating better economic data as well. We could also try other regression algorithms, though it seems like linear regression may be our best bet.



4 Future work

We currently have reasonable predictions for individual housing prices and overall market trends. Even in their current state, our results could be useful to real estate professionals. Accordingly, there are a number of paths for us to continue this project after the conclusion of the CS229 class if we get positive feedback from realtors.

Speaking of which, we will continue working with Mr. Mcnair to improve our results and hopefully build it into a useful tool for real estate professionals. We plan on applying the techniques and algorithms we used for the Georgia MLS data to data for the Silicon Valley area as well. The state of the real estate market in Silicon valley also has a tight coupling with the economic state of technology companies in general so there is a lot of opportunity to integrate more financial data into our algorithms.

Regarding our methodology, there is still a lot we can do to try to improve the results for the market trend data predictions. For one, we believe we could bring our relative error down to about 20% if we just had square footage and lot size. We could also look into improving our data clustering algorithms to see if we can automatically detect and predict neighborhood bounds. Having an accurate estimate of such bounds, while useful in and of itself, should also allow us to better predict market trends and housing prices.

References

[Eld06] Gary W. Eldred. *Investing in Real Estate*. Wiley, 6th edition edition, 2006.

Algorithm	Init. Clean	Outliers	Stock Data	Relative Error	Absolute Error
Linear Reg.	No	None	No	1.08	$ 90,183
Linear Reg.	Yes	None	No	0.57	$ 48,447
Linear Reg.	Yes	One Class SVM	No	0.33	$ 46,968
Linear Reg.	Yes	KMeans	No	0.43	$ 48,300
Weighted Linear Reg.	Yes	KMeans	No	0.35	$ 38,961
Weighted Linear Reg.	Yes	One Class SVM	No	0.26	$ 37,197
Weighted Linear Reg.	Yes	One Class SVM	Yes	0.24	$ 37,124