**Capstone 2: Consolidated Report**

***Chicago Crime and Housing Values***

Jared Hammernik

**Problem Statement**

How well can rates of crime predict housing prices? What kinds of crime have the strongest effect? This is a problem that has relevance to many real estate investors, homeowners, landlords, and neighborhood residents. In this analysis I sought to answer these questions.

**Data Sources**

To answer these questions, I obtained first crime data from the Chicago Police Department, using the “Crimes - 2001 to present” dataset (<https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2>). This dataset includes all crimes documented by the Chicago Police Department since January 2001. The dataset has over 7 million rows and 22 columns. Each crime includes information about its time, location, and category of crime type. Locations include the the community area, zip code, beat, ward, partially redacted address, and latitude/longitude.

For housing price information, I obtained values from Zillow’s research data (<https://www.zillow.com/research/data/>). Zillow provides housing price data in a number of different forms, however I chose to use the Median Home Value Per Sq Ft. Zillow provides this information as monthly values as columns. The rows can be a number of geographic representations such as zip code, state, or neighborhood.

**Data Wrangling**

The first and most obvious problem I faced was to determine how I would pair the home value data with the crime data. Since both data sources split the data by zip code, this seemed the obvious choice. However upon further inspection this proved more difficult. While the crime data set had a column for zip code, it was not labeled with its proper zip code and instead assigned a numerical value representing a zip code. Unfortunately, there was no clear way to identify which zip code each numerical value represented and any way to do so seemed extraordinarily tedious. In addition, I was wary about the trustworthiness of any zip code level population data, which I would need later to represent the crimes as rates. Instead, I opted to make the pairing based on Community Area and neighborhood. Chicago Community Areas were boundaries first established by the University of Chicago in the 1920’s and have the advantage of having rarely changed throughout the years. There are a total of 77 of them. In addition, they have also been incorporated with census data, making it easier to determine the population of each boundary. Each Community Area also has a designated number value associated with it (ie Rogers Park is community area 1, etc.)

Unfortunately while the crime data had values for Chicago Community Areas, the Zillow housing value data had values only for neighborhood, which do not exactly match up. The next challenge was to determine how to map the Zillow neighborhoods to Community Areas. In order to do this I took the liberty of using Wikipedia (<https://en.wikipedia.org/wiki/Community_areas_in_Chicago>), which gave nice tabular representations of each chicago community area and its corresponding neighborhoods. Fortunately, there were very few neighborhoods that were in multiple community areas. For those that were, I just opted to drop them from consideration rather than undertake the laborious task of determining how to geographically split the neighborhood and its home value between two community areas. I then used this mapping to create a csv file (Neighborhood\_dictionary.csv) which contained the dictionary values I would use to create a dictionary with Zillow neighborhoods as keys and Community Areas as values. I must note that there were roughly 12 community areas that had no Zillow neighborhoods associated with it and therefore housing value data associated with it.

I first had to take the 7 million row crime dataset and reshape it into a form compatible with the monthly zillow data and for modeling. To do this I reshaped it into a data frame with each row being a community-area-month (ie Rogers Park-January 2001), each column being a type of crime, and each value being a count of the number of crimes of that type in a given community area month. The code to make this dataframe took 4 hours to run. The resulting dataframe had 17,017 rows and 35 crime type variables.

Now that I had this dataframe, I needed to adjust the count of crime values to crime rates based on population. To do this I obtained population data on each community area from the Chicago Metropolitan Agency for Planning (<https://datahub.cmap.illinois.gov/dataset/2010-census-data-summarized-to-chicago-community-areas>). Fortunately since the community areas are incorporated in the Census, population values are able to be obtained from the 2010 census. Although the populations of these community areas likely do change over time, I was not able to obtain monthly population values for the Chicago community areas so that I could provide a unique denominator for each community-area-month. I figured the 2010 census population values were a decent estimate for any community-area-month value since the crime data goes from 2001 to 2019. After reading in the populations from CommunityArea\_dictionary.csv, I then divided each crime count value by the community area population and then multiplied by 100,000 to get a monthly crime rate per 100,000 people. Each crime type column name with this adjustment was given the “ADJ” suffix.

The next step was to add the Zillow data column for the housing values. I wrote a function that would take a community area and month, find the corresponding Zillow neighborhoods housing values for that month and average them. I’ll note that it is likely that some neighborhoods are bigger than others, and thus perhaps by simply taking the average, it may not be an accurate reflection of the true median housing value per sq. ft. of the community area. However, averaging it seemed to be the easiest method and is likely a decent estimate. Using this function, I added a column for the given month’s housing value (“MHV”). In addition, since I also hope to be able to use one months data to predict the prices of the next month, I made a similar function that would add the next months price as a column (“FUTURE MHV”).

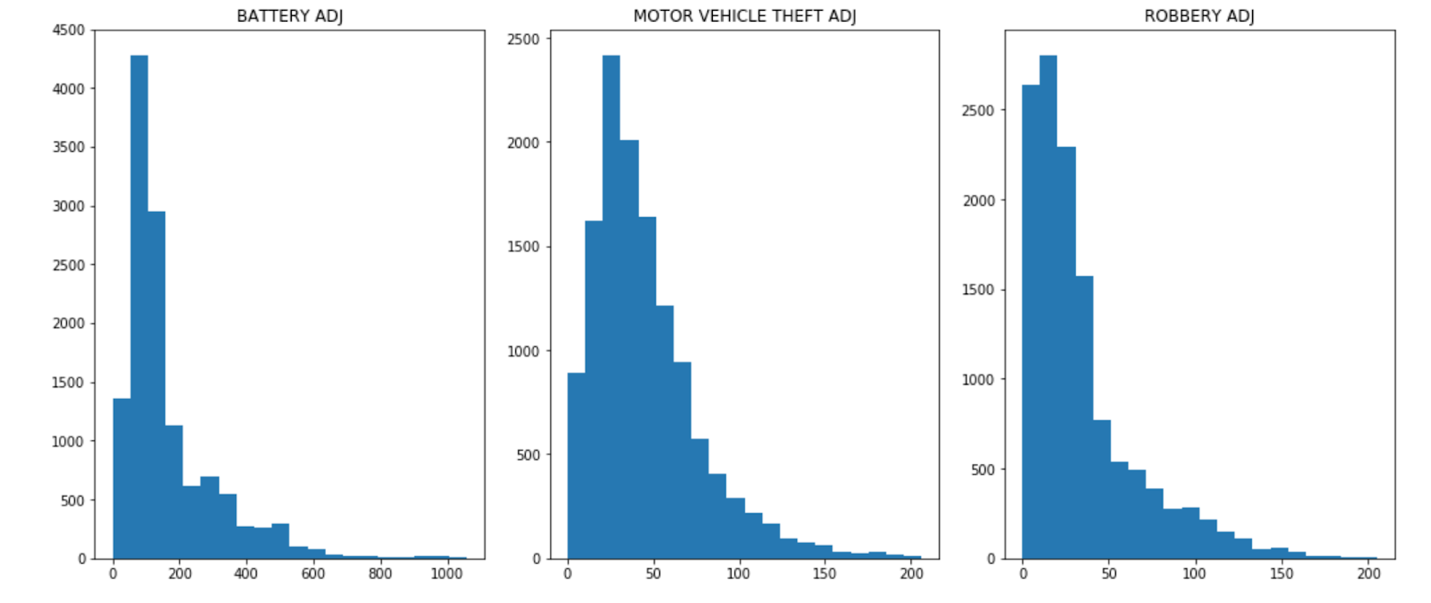
Upon simple examination of the housing prices in each community area over time, I noticed pretty quickly that they all followed a similar pattern of moving upward, suffering a large decrease, and then recovering. It soon occurred to me that the housing crisis likely played a big role in the housing values. I thought it wise to correct for this, as well as other broadly affecting factors such as inflation. I did this by creating an adjusted housing value that took the average of all the other community area housing values for that month and subtracting it from that community area-month’s value, therefore getting a value that was the difference between that areas housing value from the average, thereby correcting for these factors (“MHV ADJ”).

I also noticed that the crime data exhibited strong seasonality, i.e. crime tended to be much higher in the summer months and lower in the winter months. In order to correct for this, I created an adjusted crime value column for all crime types that was the 12 month rolling average of the crime rate of that type. This did however serve to reduce the amount of data, as I would have had no crime values for the first 12 months in each community area.

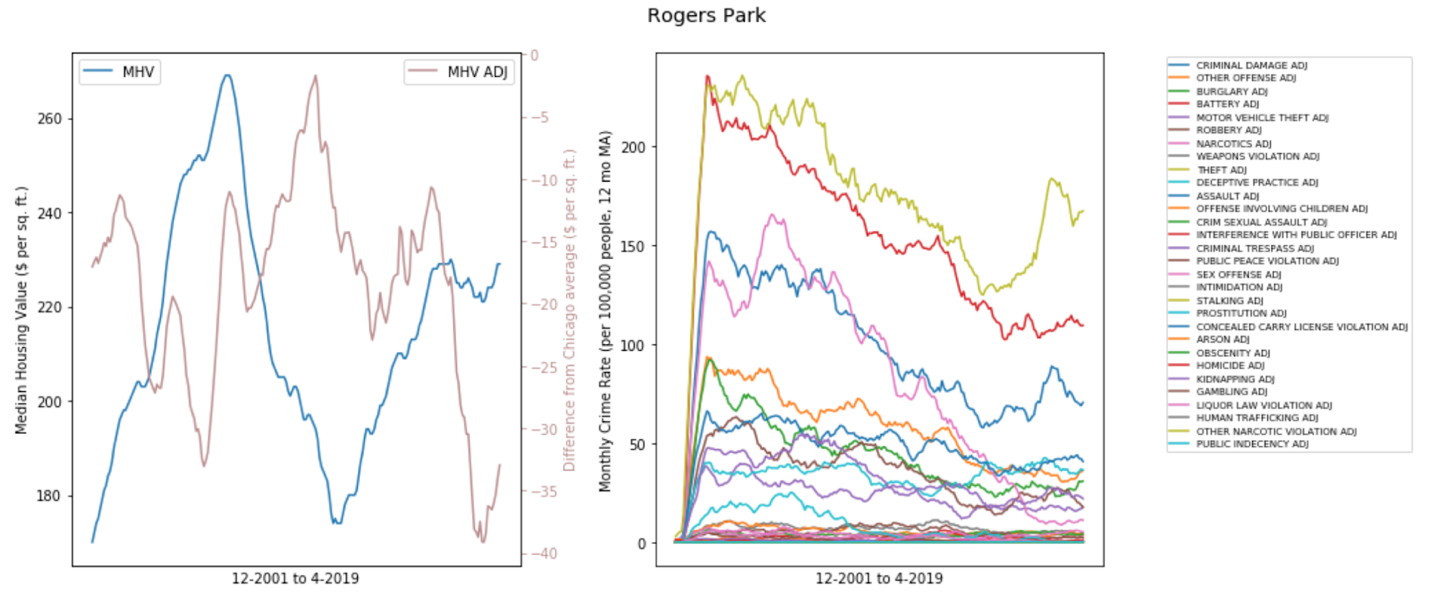
In addition, I had to delete all community areas in which there was no Zillow housing value data. Ultimately, this left me with a dataframe with 12,732 rows, the 35 adjusted crime value columns, and multiple potential housing value related target columns to predict on.

**EDA**

After applying the appropriate wrangling techniques, I then began to explore the data visually. I first sought out to explore the distributions of each crime type. I noticed that they all seemed to demonstrate truncated normal or exponential distributions. Here are some examples:

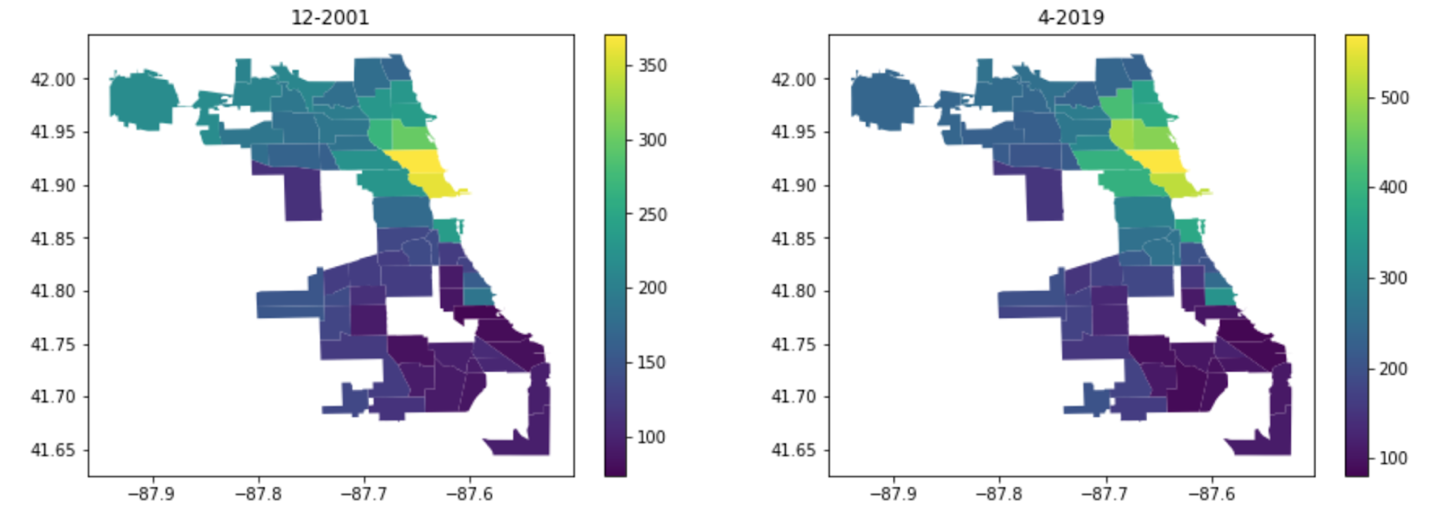


I then sought to visualize how the crime rates and housing prices have changed over time in each neighborhood. I did this by making two side-by-side graphs for each community area that show the rolling crime rates of each type over time on one side, and the median housing value and adjusted median housing value on the other side. For example, here’s my neighborhood of Rogers Park:



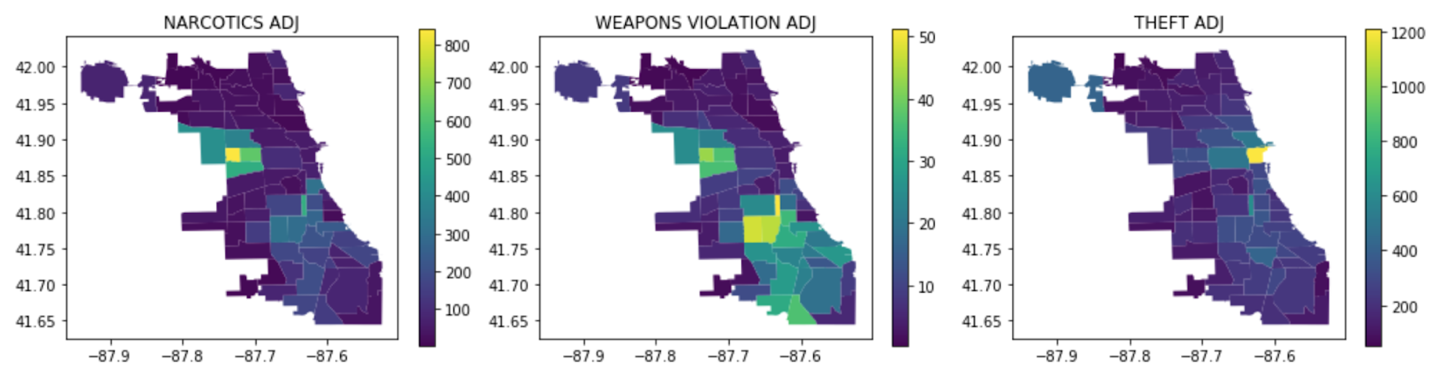
One can notice the peak, the dip, and the recovery in median housing value per square foot in this and all other community areas caused by the recession and housing crisis. In addition, we see there is a major slow runup in crime values for each community area. To me this suggests that the crime data appears to have not been as well collected in the first year, as I doubt crime went from such low values to such high values so quickly. Perhaps this suggests it is best to drop this early data from our models as well.

Lastly, I wanted to show geographic representations of the crime and housing values. To do this I used the GeoPandas library. I got the shapefile of community area boundaries from the Chicago Police department website. I then first plotted the median housing value in each neighborhood at the start of the crime data (December 2001) and compare that to the latest month (April 2019). The community areas with the highest housing values are shaded yellow and the ones with the lowest values are shaded purple.



One can also see that some of the community areas are blank as there was no Zillow neighborhoods data that corresponded with these community areas. One can also see that there has not been much change over the last 20 years in the *relative* housing values, although the overall housing values have certainly increased.

I was then curious to get an idea of which crime types might be more prevalent in different community areas. I therefore plotted the maps of the mean adjusted crime rate from 2001-2019 for each of crime types. Here are a couple examples below:



One can see that most crime seems to be concentrated in the south and west regions of Chicago, although there are certain crime types that seem to be concentrated in the loop. Theft for example, (not to be confused with robbery or burglary) seems to be concentrated in the loop, whereas both robbery and burglary seem to once again be most prevalent in the south and west sides. As I understand it theft includes pocket-picking, purse-snatching, and retail theft, whereas robbery involves the use of a weapon or intimidation, and burglary involves the unlawful entrance into property with intent to steal. In a way this makes sense as the loop is an area with plenty of opportunity for crime but is a highly visible area with lots of people and cameras, therefore stealthier tactics such as pick-pocketing would be more prevalent.

**Models**

Current Month

I then sought to predict the current month’s housing value based on the crime data for that month. The main intention of doing so was to see what were the most important features, but I was also curious to see how well they can be predicted.

Before doing any models I split the data between test and training sets. 20% of the data will be a test set and 80% will be the training data. I then scaled the training variables using StandardScaler, and then used this scaler to scale the test training variables.

*Random Forest*

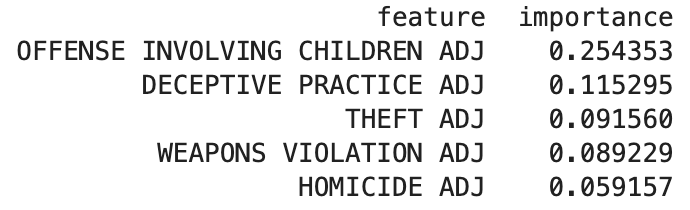
The first model I sought to try was Random Forest regression. In order to tune the hyperparameters I decided to use the Hyperopt library. I first defined a function I wished to minimize, which in this case was the cross validated random forest mean squared error. I then used hyperopt’s fmin function to find the best parameters for both the number of trees and number of features to consider at each split. To save time I only ran it for a maximum of 10 evaluations. I then used the best parameters it found to create a random forest regression model to predict the current month’s housing value.

R^2: 0.9710608240953853

Root Mean Squared Error: 15.019187291780348

I got an RMSE of 15 (dollars per sq ft), which is pretty good considering the range of values was 65 to 570. The R-squared was also very high suggesting much of the housing price variability is explained by the crime variables.

We can then look at the feature importances:



Interestingly, crimes against children seem to have the most significant effect on housing prices by a large degree, even more than homicide.

*Gradient Boosting*

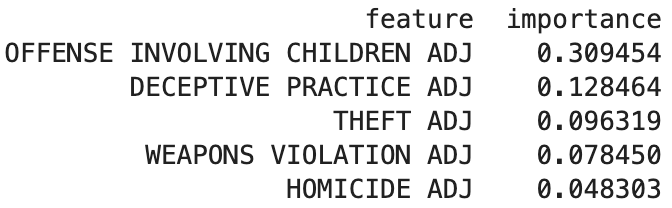
I then sought to try a Gradient Boosting regression model. Once again I used Hyperopt to tune the hyperparameters. The function to minimize was once again the cross validated mean squared error. After finding the best parameters for the number of trees, the max features, and the learning rate after 10 evals, I built a model to once again predict the current housing value.

R^2: 0.9745518028036164

Root Mean Squared Error: 14.084189635027736

This model performed slightly better than the Random Forest.

Once again, looking at the top feature importances, we see that it mimics that of the Random Forest:



*Other Models*

I also tried a couple other models including KNN, Linear Regression, and an ensemble Voting Regressor. Surprisingly, a KNN model with a single neighbor performed the best with an RMSE of 8. I believe this to be the case since the nearest neighbor in the training set would likely to be a nearby month in the same community area, which probably has very similar crime and housing values. Thus while very accurate, it probably wouldn’t generalize well onto data outside this dataset.

*Adjusted Housing Value*

I then tried to do everything above but instead predicting on the adjusted housing value (the difference of the housing value from the average housing value of the city). The results were all very similar except slightly more accurate. For instance:

Random Forest:

R^2: 0.9740762172521555

Root Mean Squared Error: 13.821385655337439

Gradient Boosting:

R^2: 0.9792239853808397

Root Mean Squared Error: 12.373242358449236

This suggests that broad effects like the housing crisis and inflation do play a role but perhaps not as significant of one as I expected.

Future Month

Since being able to predict future housing prices would be of the most business relevance, I decided to now predict the next months price based on all the crime values and the housing value of the current month.

After splitting the data into test and training sets and standardizing the data, I established a baseline RMSE value which was using the previous months price to predict the next month’s. Doing so gave me a baseline RMSE of 1.94.

However, this baseline does nothing to predict the direction the price will move in the future, which is of the most important business relevance. Therefore, in addition to RMSE, I decided to also look at the models ability to accurately predict the direction of the next months price as a percentage. I wrote a function that would give me this information. A quick test of the future prices compared to the current prices show that the prices go up 54.5% of the time in the dataset. Therefore a model that predicted it to go up 100% of the time would be 54.5% accurate. This provides a good baseline for the ability of the model to predict the direction of price movement.

Since I was using adding the current month’s price to the explanatory variables, and I assumed this would have a strong effect on the model, I decided to use Hyperopt once again to tune my parameters as I expected it might be different. After running the Random Forest regression model with the tuned parameters, I got the following results:

Random Forest:

R^2: 0.9996352833925544

Root Mean Squared Error: 1.689352278356195

Directionality %: 69.84687868080094

The RMSE was slightly better than the baseline at 1.68, however it was able to pick the correct direction of price movement 69.84% of the time, a solid improvement of the baseline at 54.5%.

I then once again used Gradient Boosting Regression. I used hyperopt again to tune the parameters, and with those parameters created a model. The models performance was as follows:

Gradient Boosting:

R^2: 0.9997370381029932

Root Mean Squared Error: 1.4344613930280252

Directionality %: 75.97173144876325

The RMSE performed much better than both the baseline and the Random Forest at 1.43. In addition, the ability to predict directionality improved to 75.97%.

I then once again tried the KNN, Linear, and ensemble Voting models:

KNN

R^2: 0.9979442257170397

Root Mean Squared Error: 4.0107922452317135

Directionality %: 66.70592854338437

Linear

R^2: 0.999600253445323

Root Mean Squared Error: 1.7686211605906303

Directionality %: 61.75893207695328

Voting

R^2: 0.9997059709212114

Root Mean Squared Error: 1.516832389521937

Directionality %: 78.955634079309

While the Gradient Boosting method had the best RMSE, the ensemble voting regressor had the best ability to predict the directionality at nearly 79%. These results suggest that with only the crime values of the current month (as a rolling average of the last 12 months), one can predict decently well whether the median housing value per sq ft will go up or down.

LSTM

Next, I tried building a multivariate LSTM model to predict the future prices. To do this, I first split the data into a test and training set. The training set was every crime rate before 2015, and the test set was everything after. I then standardized the data and constructed a function that would convert the data into a format to be used by the LSTM model. This function converted the data such that each data point was a 2 dimensional array containing the values of all the crime types for a given number of trailing months. In this case I converted use a 12 month trailing history to train the model. In addition, I dropped the first 14 months as my exploratory data analysis showed that these data points were likely untrustworthy. In addition, since a couple of community areas did not have a full set of data, the function removed these from consideration. Next I then converted the data into a TensorFlow dataset, and shuffled and batched the data. After some manual tuning of the parameters, I used a model with a single layer and 4 neurons. I ran the model for approximately 100 epochs until the validation loss stabilized and stopped decreasing. The mean squared error on the test set was 0.0063 in standardized units, which translated to 6.83 dollars per square foot, which is worse than any of the other models I tried. In addition, it predicted the correct direction of price movement 65% of the time compared to a baseline of 57.8%. All in all, the model did not perform any better than the other models, however with more time and more tuning of parameters it could perhaps be improved.

**Next Steps**

If I had more time I would do a couple things. First, I would further work on tuning the LSTM by trying a number of different architectures and optimization functions. Second, I would also be very curious to try the reverse and see if housing prices can predict crime. Lastly, I would have liked to run similar models on rental price data to see if it had the same effect.