Springboard Data Science - Career Track
Capstone Two - Final Report

Predicting Residential Real Estate Prices in Clark County, Nevada

Problem Statement

Based on residential real estate data from Redfin, what are the most important features when determining listing price for a home in Clark County, Nevada? Can listing price be predicted based on the various features?

Background

My husband and I are looking for an investment property to purchase in the Las Vegas, Nevada area (Clark County). Unfortunately, we don't know very much about real estate. I am interested to see what features most affect purchase price in our area. I would then like to use a predictive model to determine whether a listing is over/under-priced. This model could also help guide us on selecting a reasonable offer price for any properties that we hope to pursue.

Approach

I decided to use a regression model to predict real estate prices. The data that I used to train and test my model was downloaded from Redfin.com. A major constraint for this project was that Redfin limits their data downloads to 350 records, so I had to do individual searches on each zip code in the Las Vegas area. For the sake of time, as well as for consistency, my search was limited to properties sold in the 3 months preceding my data collection (properties sold February to April 2021). I also restricted my search to Single Family Homes, Condos, and Townhouses. My final dataset contained just over 11,500 records and the following features (listed in no particular order).

address	beds	year_built	latitude
city	baths	days_on_market	longitude
zipcode	square_feet	price_per_sqft	property_type
price	lot_size	hoa_per_mon	

Data Wrangling and Analysis

Missing Data

After dropping all duplicate rows from my dataset, I focused my attention on columns with missing data. In my initial dataset, there were missing values in the following columns: beds, baths, square_feet, lot_size, year_built, days_on_market, price_per_sqft, and hoa_per_mon. The records with missing "square_feet" values were dropped immediately because they were also missing a lot of other data. I also chose to drop the records with missing values in the "beds", "baths", and "lot_size" columns as I felt that these would likely be problematic later on. At that point, I was only left with missing values in the "days_on_market" and "hoa_per_mon" columns.

After some deliberation, I forward-filled the missing values in the "days_on_market" column, and due to some outliers, I decided to fill the missing values in the "hoa_per_mon" column with the median value. It is possible that some of the properties don't have HOA fees, in which case the null value might be correct, but based on the background research that I conducted, that is very unlikely in the Las Vegas area.

One-Hot Encoding (property_type and city)

The "property_type" column was very unbalanced, so after reviewing the definitions of each property type I chose to bin "condo" and "townhouse", leaving two possible values: "Single Family Residential" and "Condo/Townhouse". I then One-Hot Encoded the "property_type" column, resulting in a single column labelled "propertytype_singlefamily" with a "1" value denoting a Single Family Residence and a "0" denoting a Condo/Townhouse property. The "city" column was also One-Hot Encoded resulting in five columns that identify each record as being located in one of six possible cities in the Las Vegas area: Las Vegas, Henderson, North Las Vegas, Boulder City, Enterprise, Blue Diamond.

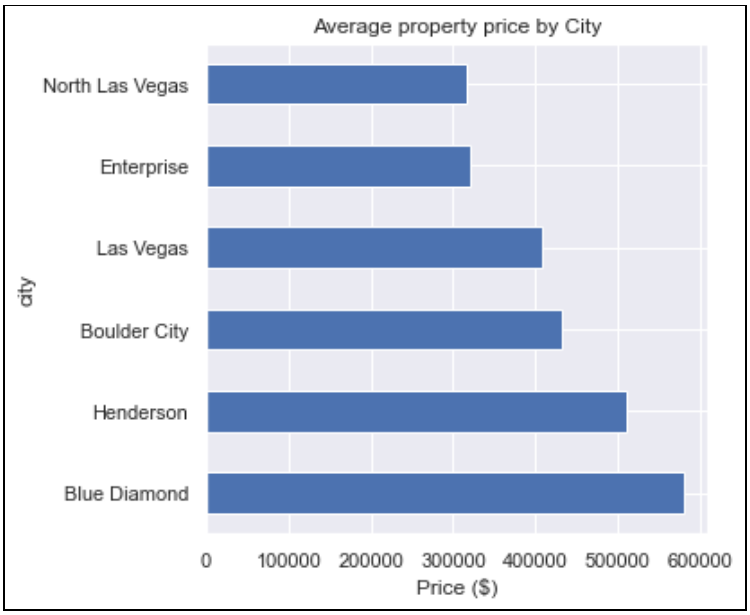
Outliers

While exploring the data, I noticed that there were outliers in a few of the columns. I decided to handle them in the following ways:

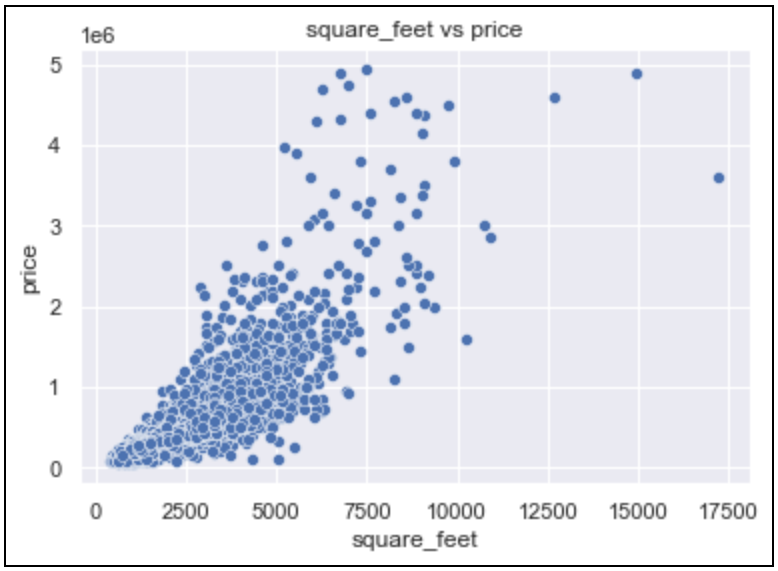
- price - dropped all records with price over $5 million (13 records total)
- lot_size - dropped 1 record because after a quick Google search it was clear that the property was vacant land mislabelled as a Single Family Residence

Exploratory Data Analysis

I created the bar graph below to visualize the average listing price per city. The Las Vegas area is made up of several smaller cities and it could be helpful to a buyer to limit their search to a specific area. This might be especially useful when looking for an investment property as most of Las Vegas' industry is centered around the Strip.



One of the most interesting findings during my exploratory data analysis was that the feature most highly correlated with my dependent variable (price) was square feet. I was a little surprised, because I would have thought that the number of bedrooms would be more related to price.



Modeling

Model Testing

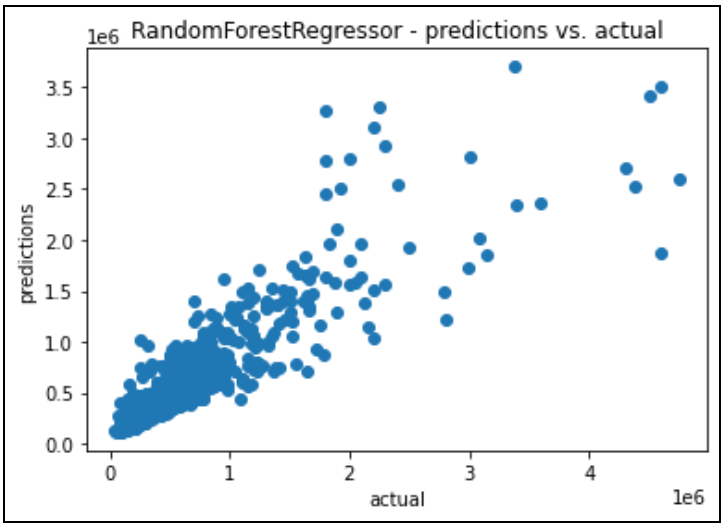
I split the data into training and test sets, and then tested several regression models. Based on the results (listed below), I decided to proceed with tuning the hyperparameters for both the RandomForestRegressor and the XGBRegressor models.

Model (default - no tuning)	**R2_score (training data)**	**R2_score (test data)**
LinearRegression	0.735	0.748
RandomForestRegressor	0.979	0.875
GradientBoostingRegressor	0.932	0.863
AdaBoostRegressor	0.483	0.464
XGBRegressor	0.991	0.873

Model Selection and Tuning

After tuning, it was clear that the RandomForestRegressor consistently outperformed the XGBRegressor model. See results below.

Model (after tuning)	Best Parameters	R2 Score	MAE
RandomForestRegressor	max_depth = 4 max_features = log2 n_estimators = 100	0.874	49339.39

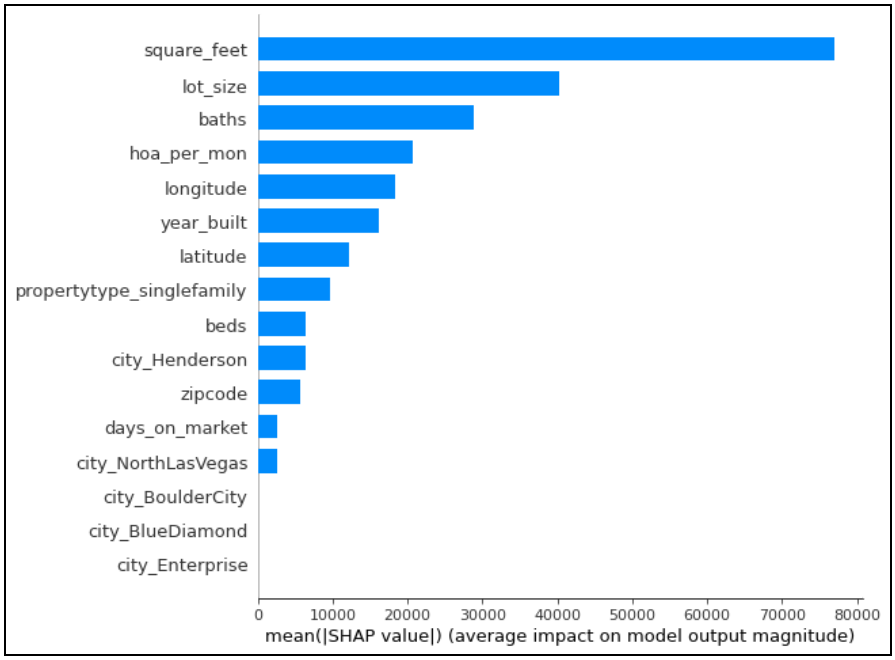


When using a regression model, it is common practice to reduce the effects of multicollinearity of your features by conducting a Variance Inflation Factor (VIF) analysis, and setting a VIF threshold. The feature with the highest VIF is removed, one at a time, and the VIFs then recalculated until all features have a VIF under the set threshold. After testing my model with and without VIF analysis, it was clear that my model performed much better without it.

Model	R2 Score	MAE	Parameters
RandomForestRegressor with VIF (threshold = 11)	0.802	63949.70	max_depth = 4 max_features = log2 n_estimators = 100
RandomForestRegressor without VIF	0.874	49339.39	

Feature Importance

I conducted a SHAP analysis on my Random Forest model, and the results were pretty close to what I expected. As mentioned previously, it was clear during my exploratory data analysis that square feet was the feature most highly correlated with price, so it's not a surprise that square_feet had the highest impact on the price prediction. The most interesting thing to me is that the number of bathrooms has a higher effect on the model than the number of bedrooms.



Future Recommendations

While my model performed well overall (R2 score = 0.874), I can think of two improvements that would be worth trying. First, I would like to include more features in my model. Some examples of important features that I did not have in my dataset include: school district, crime rate, noise level, distance to a grocery store, and walking distance to public transit. Second, the size of my dataset was limited by Redfin's download restrictions and by the amount of time I could spend on data collection. With unlimited time and a little creativity, I think I could have expanded my criteria to houses sold in the past year. Not only would this have substantially increased the size of my dataset, but it also would have given me the chance to visualize any seasonal changes in price.

Once these improvements are made, I think that this model has the potential to be useful in many ways. A seller could use this model to set a competitive listing price. A buyer could use this model to determine whether or not a property that they are interested in has a fair listing price, or they could use it to decide on a fair offer price for an investment property or even their dream home.