First-Time Home Buyers Guide: Predicting How Much a House Costs with Certain Demographics

Brandon T. Shellenberger

Northwest Missouri State University, Maryville MO 64468, USA
S570782@nwmissouri.edu

Abstract. It is even harder in this generation for first-time home buyers to get a house. There is quite a bit of information about buying a home, it starts to become overwhelming. It can be difficult to understand what drives home prices. In this paper, a dataset from Kaggle will be used to analyze the main contributors to home prices using regression models. A few columns were dropped because of their low correlation to prices and one column will be split using one-hot encoding. After curating, the data is split 80% for training and 20% for testing. The models being used are multiple linear regression (MLP), polynomial regression, decision tree regressor, and a random forest regressor. The scores used to determine the best model are Root Mean Square Error (RSME) and R^2 values. The goal of this paper is to find the best-fit model and determine which variable drives the prices of houses. All source code is on the GitHub page.

Keywords: data analytics · housing prices · first-time home buyers · linear regression · predictive models

1 Introduction

It is always difficult to sell and buy a home, especially when there are many options, from the number of bedrooms/bathrooms to how big the entry property is. It can begin to confuse people who are buying their house for the first time. There are multiple variables associated when determining prices for a house. The data being used will come from Kaggle and has 13 columns and 545 records. [5]

I want to use this data set to find a model that will predict house prices then determine which variable will be closely correlated to the prices. After collecting the data from sources, the data will be cleaned and will only have valuable information. I will group certain columns for more precise analysis and use these columns in the prediction models.

A couple of key components will be to try different columns to figure out the best scores. I will be using the number of bedrooms/bathrooms, but other columns might have a greater impact on prices. Another key component will be the predictive aspect. This will be challenging since I will be using linear regression on multiple regions.

2 Methodology

This section will walk through the process of using the data, cleaning the data for proper analysis, and running the analytics to discover answers.

2.1 Dataset

The dataset is from the Kaggle page titled 'Housing Price Prediction'. [5] I will be using a package called 'Kagglehub' to extract the data from the webpage. Each time the program runs, this package will handle searching for the data and moving the data into the correct directory for analysis. This dataset has thirteen columns; some are continuous/discrete numerical values and others are boolean. Here are a few of the columns in the dataset:

- price - This column is the price of each home.
- bedrooms - Number of bedrooms
- bathrooms - Number of bathrooms
- basement - This is a boolean value of the house having a basement.

2.2 Data Cleaning

This section is where the data will be curated to fit the analysis process. Curating is the organization and management of datasets to meet the needs of a specific analysis. [3] I used Excel for simple curating and a notebook for more complex cleaning and structuring.

The Kaggle dataset was less curated with no duplicate/null values. The attributes I deleted were 'area', 'mainroad', 'guestroom', 'hotwaterheating', 'parking', and 'prefarea'. Because the models I want to use in the analysis only use numerical values, I had to split some attributes using a One-Hot Encoding method. This is where a categorical column can be split into separate columns, one for each of the categories found in the original column. For example, the attribute 'basement', which determines if there is a basement in the house, is either yes or no as in Table 2. This attribute is split using the Python package Pandas.get_dummies() [1], splitting the column into basement_yes and basement_no and inputting a 1 for yes and 0 for no as in Table 3. Other attributes that were curated this way are the 'airconditioning' and 'furnishingstatus'.

After cleaning the dataset, I went back and looked for outliers. Graphing the price frequency, shown in figure 1a, some prices are larger and might be considered outliers. To determine if these prices were outliers, two methods were used: Inner Quartile Range (IQR) and Z-scores. The IQR is the range above the smallest and below the largest quartiles, Q1 and Q4, respectively, also considered as the middle half of the data [4]. To be an outlier with the IQR method, the values will be smaller or larger than 1.5 times the IQR. After running this analysis, there were 15 outliers and it was determined that there were too many to be deleted from the dataset. The Z-score outliers are determined by the formula below, where x is the data point, μ is the mean of the dataset, and σ is the standard deviation of the dataset.

Table 1: One-Hot Encoding Example

Table 2: Original Table

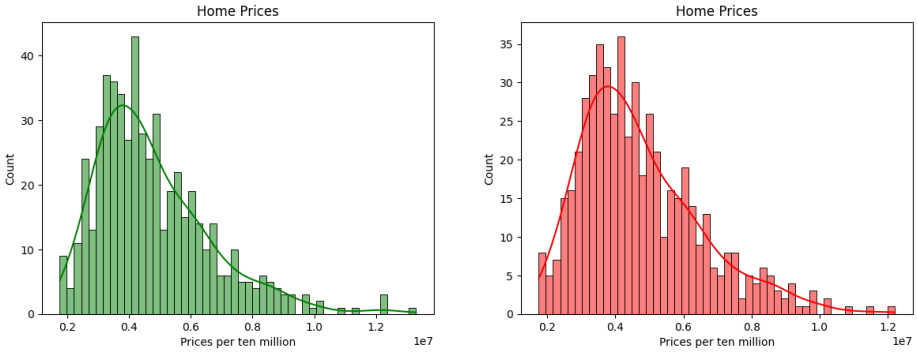
basement
yes
no
yes
no
no

Table 3: After One-Hot Encoding

basement_yes	basement_no
1	0
0	1
1	0
0	1
0	1

$$z = \frac{(x - \mu)}{\sigma} \tag{1}$$

Since this is a small dataset, if the Z-score for a data point is above 4 or below -4, it is considered an outlier. [6] With this process, there were 3 outliers. I deleted the outliers found by the Z-score and have shown the price frequencies in Figure 1b.



(a) Before removal of outliers (b) After removal of outliers

Fig. 1: With and without outliers

2.3 Analysis

The methods being used in the analysis consist of a spread of regression models: multiple linear regression, polynomial regression, decision tree regressor, and random forest regressor. The scores used during the analysis will be the Root Mean Squared Error (RSME) and the Coefficient of Determination R^2. Each model will first train 80% of the data and test the remaining 20% which is the recommended split. When running the models, there will be scores for training and testing.

Multiple Linear Regression The multiple linear regression will fit the best linear equation to the dataset. Because there is more than one feature being used, this model will have numerous coefficients, one for each of the features. Linear regressions are best for linearly scaled data sets. Looking at Table 4, the model does not help predict prices with a high average and low correlation.

Table 4: Multiple Linear Regression

Type	RSME	R^2
Train	1136595	0.5922
Test	1227494	0.5246

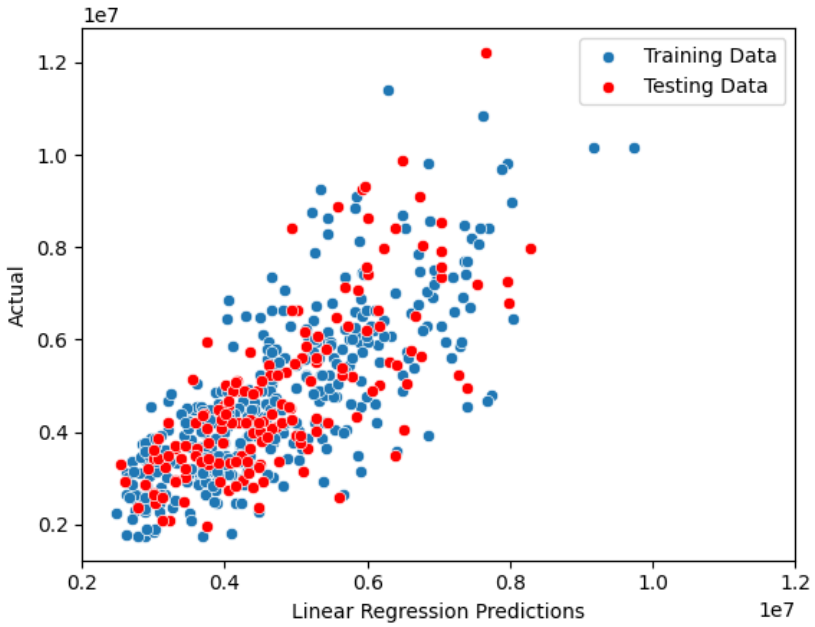


Fig. 2: Predicted vs. Actual

Polynomial Regression This model is similar to linear regression, but instead of a linear relationship, the relationships could be of a higher degree of freedom. In this analysis, the polynomial regression model used degrees of 2, 3, and 4 where the scores are in Table 5. The degree is the highest exponent found in the equation and will have degree−1 direction changes to account for possible

fluctuations in the data. Looking at the scores, R^2 from the training sets increases as the degree increases. This cannot be said for the testing sets where it decreases at degree 3, then begins to increase. Anything higher than a degree of 4, the models will overfit the data. Figure 3 shows the predicted vs. actual with a degree of 4.

Table 5: Polynomial Regression

Description	Type	RSME	R^2
Degree 2	Train	1067841	0.6401
Degree 2	Test	1194405	0.5498
Degree 3	Train	964012	0.7067
Degree 3	Test	1309440	0.4600
Degree 4	Train	952656	0.7135
Degree 4	Test	1172812	0.5660

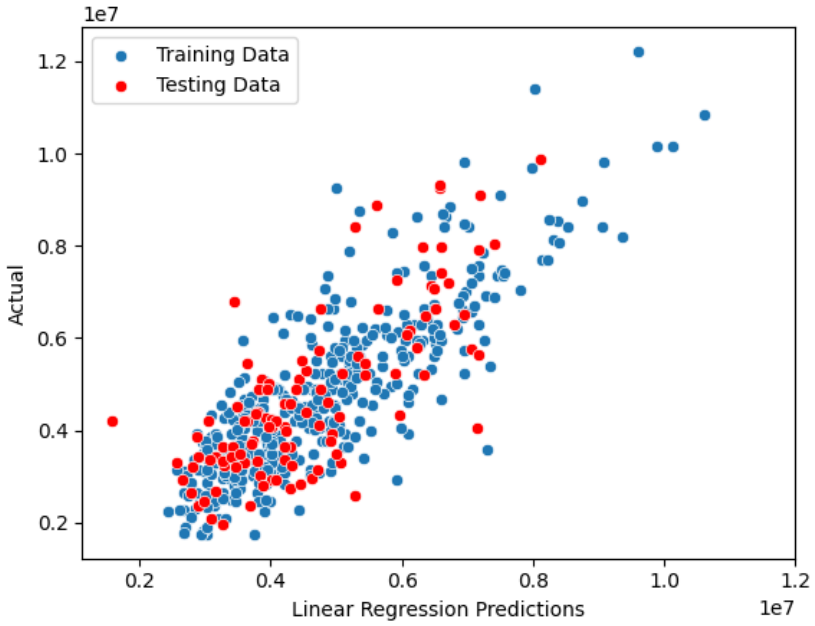


Fig. 3: Predicted vs. Actual

Decision Tree Regressor This is a model that will make a decision based on statistical probabilities. This model looks like a tree, splitting into leaf nodes

based on this probability. The model has an attribute for max depth which is the maximum number of rows/branches the model can go. In Table 6, models higher than a maximum depth of 5 start to overfit the data. Figure 4 shows the predicted and actual values with a maximum depth of 5 branches.

Table 6: Decision Tree Regressor

Maximum Depth	Type	RSME	R^2
3 branches	Train	1222539	0.5282
3 branches	Test	1256334	0.5020
4 branches	Train	1117445	0.6058
4 branches	Test	1239294	0.5154
5 branches	Train	1013031	0.6761
5 branches	Test	1216762	0.5328

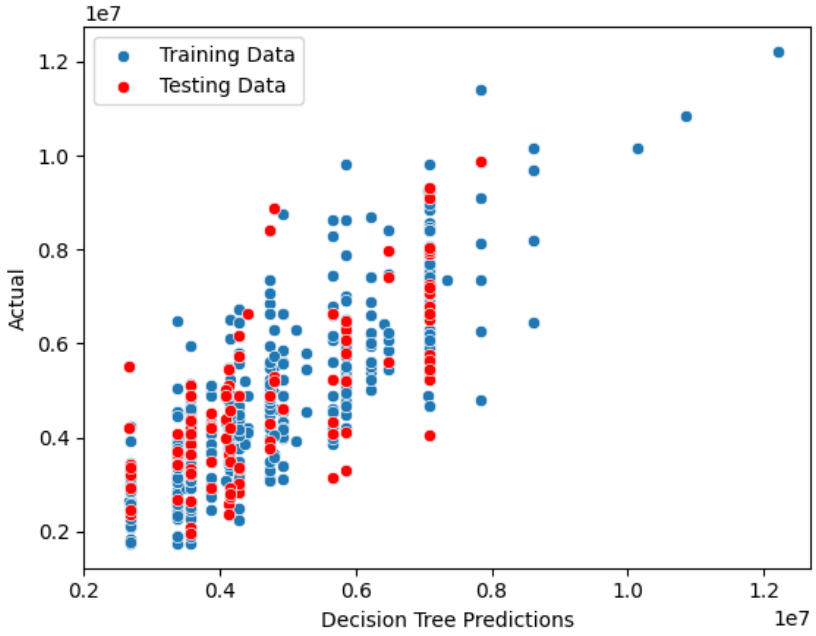


Fig. 4: Predicted vs. Actual

Random Forest Regressor This model uses multiple decision trees and averages the results to help control overfitting of the data. There are two main

attributes with this model, maximum depth (same as decision tree models) and estimators which determine the number of trees in the forest [2]. The difference in scores did not change much when the number of estimators changed. Knowing this, Table 7 will show the scores while keeping the estimators at 100 and changing the maximum depth. Figure 5

Table 7: Random Forest Regressor (est = 100)

Maximum Depth	Type	RSME	R^2
3 branches	Train	1139001	0.5902
3 branches	Test	1216965	0.5327
4 branches	Train	1027976	0.6664
4 branches	Test	1172175	0.5664
5 branches	Train	912367	0.7372
5 branches	Test	1159471	0.5758

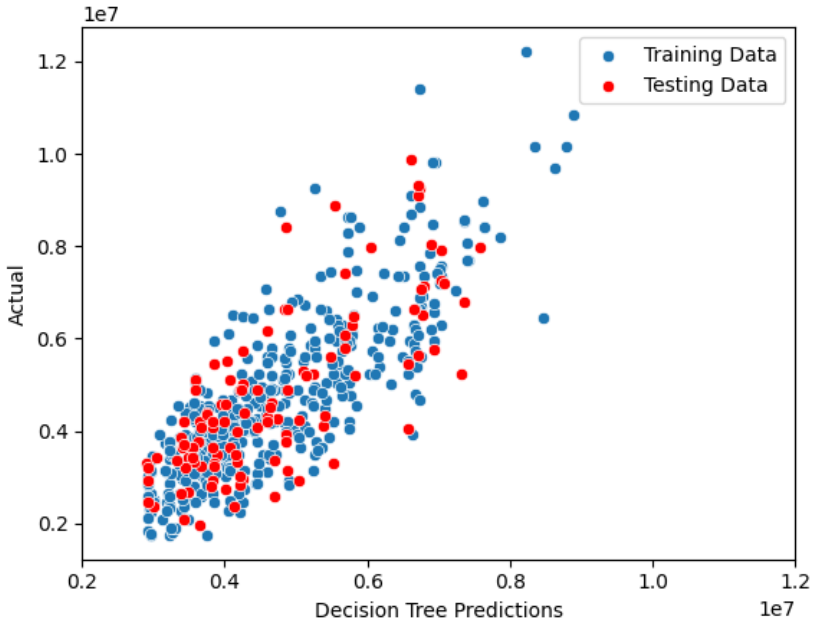


Fig. 5: Predicted vs. Actual

In the next section, we will compare the top results from each model.

3 Results

Even though most of the models show some overfitting, there are three models that would be best suited for predicting housing prices:

1. Decision Tree with a maximum depth of 5 branches
2. Random Forest with a maximum depth of 4 branches
3. Multiple Linear regression

In the end, the Random Forest showed a better score for the training and testing data without too much overfitting. Also, the RSME values for both datasets were lower than those of the other two models listed above. Looking back at the data, I ran a correlation matrix on each of the columns and found that the area of the house was the most closely correlated to the price and any other column. Figure 6 shows the scatter plot of the two variables a positive correlation near smaller houses, but as the houses grow, the correlation in price does not seem to be close.

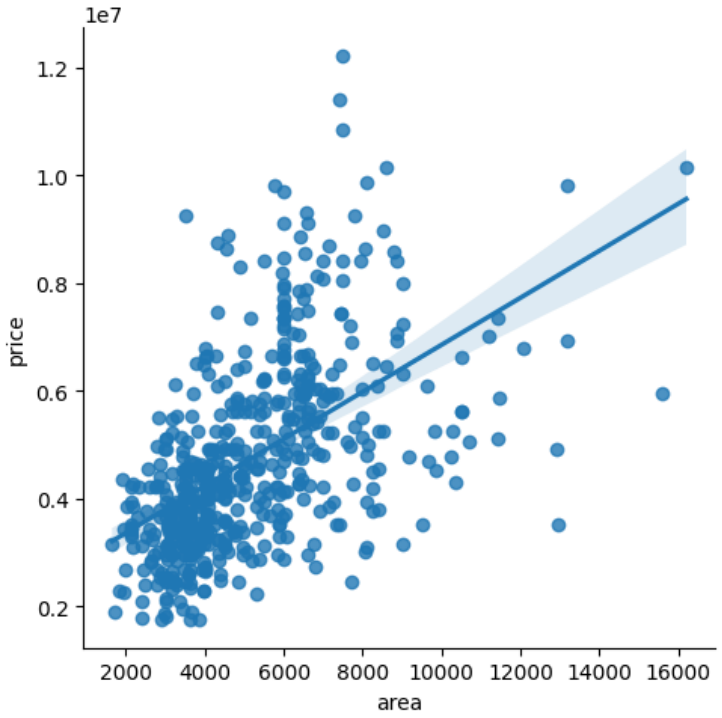


Fig. 6: Price vs. Area

Also, while looking back, the basement variables have hardly any correlation with price. I went back and dropped these columns and re-ran the models. The scores of each model improved slightly, but not significantly.

4 Limitations

One of the limitations I have encountered is that running each model with multiple details takes some time. The models also did not perform as expected with R^2 values less than 0.7.

5 Conclusion

With the data I used, there is not much of a correlation between the variables and the prices of the houses. After running the models, the linear regression is the better model, but it does not fit the data as expected.

References

1. pandas.get_dummies &x2014; pandas 2.2.3 documentation — pandas.pydata.org. https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html, [Accessed 31-03-2025]
2. RandomForestRegressor — scikit-learn.org. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html, [Accessed 18-04-2025]
3. What Is Data Curation? Why Is It Important? — Alation — alation.com. https://www.alation.com/blog/what-is-data-curation/, [Accessed 30-03-2025]
4. Frost, J.: Interquartile Range (IQR): How to Find and Use It — statisticsbyjim.com. https://statisticsbyjim.com/basics/interquartile-range/, [Accessed 06-04-2025]
5. KUMARdatalab, H.: Housing price prediction (Jul 2023), https://www.kaggle.com/datasets/harishkumardatalab/housing-price-prediction
6. Shiffler, R.E.: Maximum z scores and outliers. The American Statistician **42**(1), 79–80 (1988)