Predicting Housing Prices with Machine Learning Models

Antonio Flores

2024-08-06

# 1. Abstract

There are many different variables that affect housing prices which can vary drastically. In order to make the best decisions about housing prices, it is useful to be able to predict the sales price of a house given a set of descriptors. This project will seek to identify a model for predicting housing prices using a dataset provided by Kaggle. This will be achieved by comparing the accuracy and predictive power of different machine learning models.

# 2. Introduction

## 2.1 General Background Information

Machine learning models provide a massive opportunity for real estate investors to identify housing prices by using a set of predictors. This project will utilize Regression Prediction. Unlike Classification Prediction, which attempts to predict a classifier based on given predictors, Regression Prediction seeks to identify a continuous value (e.g., x amount of dollars, x amount of cells). In our case, our response variable is a continuous variable, the sales price of a house.

## 2.2 Description of data

There are 1,460 records, 81 columns/variables. There is an ID column, as well as a response value, meaning there are 79 predictors, which include descriptors of the house in question. These include both categorical predictors as well as numerical predictors. Areas described by the predictors include, Basements, Garages, Bathrooms, Location, Age, and many, many others. Additionally, as mentioned, the response variable is the sales price of the house.

## 2.3 Questions/Hypotheses to be addressed

The research question I plan to address with my analysis is: which features are the best predictors of housing prices? Additionally, the desired output of this analysis is a machine learning model which allows real estate investors or other interested stakeholders to better make decisions regarding future real estate purchases.

# 3. Methods

## 3.1 Data import and cleaning

### 3.1.1 Reading in the Data

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table 1: Data Snapshot of first 7 Variables   | Id | MSSubClass | MSZoning | LotFrontage | LotArea | Street | Alley | LotShape | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 60 | RL | 65 | 8450 | Pave | NA | Reg | | 2 | 20 | RL | 80 | 9600 | Pave | NA | Reg | | 3 | 60 | RL | 68 | 11250 | Pave | NA | IR1 | | 4 | 70 | RL | 60 | 9550 | Pave | NA | IR1 | | 5 | 60 | RL | 84 | 14260 | Pave | NA | IR1 | | 6 | 50 | RL | 85 | 14115 | Pave | NA | IR1 | |

### 3.1.2 Cleaning Data

There were several different data cleaning methods that were experimented with in order to find the best way to prepare the data for modeling. First I removed all variables that had more than 80% NA values. Then for those categorical predictors with some missing values, I replaced those with a “None” value. Next, I converted all the categorical variables to factors (they were initially read in as character values). Once the variables were recognized as factors, I could evaluate which predictors were victims of class imbalance. I removed the predictors with around 80% of observations in one class of a variable. Finally, I replaced the NA values in the numeric predictors with the median value of that variable. When I was finished with the data cleaning portion of this project, I had reduced the data set down from 79 to 63 predictors, 26 categorical and 37 numeric variables.

### 3.1.3 Background on Chosen Models

I chose to utilize the following models: the K-Nearest Neighbors model (KNN), Linear Regression model, and RandomForest models to determine the best prediction model for this project.

**KNN:**  
The KNN model predicts based on the closest samples or neighbors. Essentially, to predict a value, the data is broken up into samples/neighbors, and then the nearest samples (using Euclidean distance, typically)to the value of interest are examined to either classify or find a mean between the chosen samples. K represents the number of neighbors to utilize to come to this conclusion (Kuhn (2018)).

**Linear Regression:**  
Linear Regression focuses on minimizing the SSE (Sum of squared errors) between the predicted and original response value.

**RandomForest:**  
RandomForest models take advantage of decision trees. If we think about the scenario in our project (the price of a house), we could imagine a decision tree starting with, “does having two bathrooms cause a house to be priced at over 400K?” This would be our first node to split the data on. We could continue asking things like, “Are houses with larger basements priced at more or less than 400K?” Questions like these would represent more decision nodes for us to split the data on, getting us closer to the most accurate prediction model. The RandomForest algorithm uses different methods to create several uncorrelated “forests” of decision trees(IBM (n.d.)).

### 3.1.4 Background Regarding Performance Metrics

**R-Squared (R^2):**  
The R^2 value explains what percent (proportion) of the total variance in the data is explained by the model.  
**RMSE (Root Mean Squared Error):**  
Represents the average size of the difference between real values and the predicted values, then  
**MAE (Mean Absolute Error):**  
Similar to RMSE, but the absolute value of the difference between real and predicted is used.

# 4. Results

## 4.1 Basic statistical analysis

Before I began with the machine learning analysis, I sought to test the variables of interest in their significance of affecting the response variable.

Below are the results of the linear model fit with all variables as predictors

| term | estimate | std.error | statistic | p.value |
| --- | --- | --- | --- | --- |
| (Intercept) | 4.604292e+05 | 1.413535e+06 | 0.3257288 | 0.7446773 |
| MSSubClass | -1.820243e+02 | 2.767475e+01 | -6.5772690 | 0.0000000 |
| LotFrontage | -5.634362e+01 | 5.176798e+01 | -1.0883875 | 0.2766081 |
| LotArea | 4.304975e-01 | 1.021082e-01 | 4.2160918 | 0.0000264 |
| OverallQual | 1.733111e+04 | 1.187394e+03 | 14.5959170 | 0.0000000 |
| OverallCond | 4.674676e+03 | 1.032513e+03 | 4.5274745 | 0.0000065 |
| YearBuilt | 2.694912e+02 | 6.741315e+01 | 3.9976056 | 0.0000673 |
| YearRemodAdd | 1.344831e+02 | 6.858803e+01 | 1.9607366 | 0.0501043 |
| MasVnrArea | 3.134596e+01 | 5.932820e+00 | 5.2834831 | 0.0000001 |
| BsmtFinSF1 | 1.921300e+01 | 4.666827e+00 | 4.1169307 | 0.0000406 |
| BsmtFinSF2 | 8.273997e+00 | 7.057001e+00 | 1.1724524 | 0.2412114 |
| BsmtUnfSF | 9.297103e+00 | 4.193927e+00 | 2.2168013 | 0.0267940 |
| TotalBsmtSF | NA | NA | NA | NA |
| X1stFlrSF | 4.901342e+01 | 5.809589e+00 | 8.4366426 | 0.0000000 |
| X2ndFlrSF | 4.902978e+01 | 4.983306e+00 | 9.8388062 | 0.0000000 |
| LowQualFinSF | 2.534285e+01 | 1.996942e+01 | 1.2690830 | 0.2046187 |
| GrLivArea | NA | NA | NA | NA |
| BsmtFullBath | 9.369818e+03 | 2.611716e+03 | 3.5876105 | 0.0003450 |
| BsmtHalfBath | 2.051814e+03 | 4.091012e+03 | 0.5015420 | 0.6160672 |
| FullBath | 3.439741e+03 | 2.836717e+03 | 1.2125780 | 0.2254922 |
| HalfBath | -1.872747e+03 | 2.662817e+03 | -0.7032955 | 0.4819865 |
| BedroomAbvGr | -1.008636e+04 | 1.701690e+03 | -5.9272618 | 0.0000000 |
| KitchenAbvGr | -1.215840e+04 | 5.211646e+03 | -2.3329299 | 0.0197904 |
| TotRmsAbvGrd | 5.044090e+03 | 1.236951e+03 | 4.0778400 | 0.0000480 |
| Fireplaces | 3.984870e+03 | 1.776709e+03 | 2.2428376 | 0.0250605 |
| GarageYrBlt | 1.268380e+02 | 6.897832e+01 | 1.8388100 | 0.0661511 |
| GarageCars | 1.129285e+04 | 2.876386e+03 | 3.9260550 | 0.0000905 |
| GarageArea | -4.382456e+00 | 9.941118e+00 | -0.4408414 | 0.6593947 |
| WoodDeckSF | 2.388005e+01 | 8.011714e+00 | 2.9806414 | 0.0029252 |
| OpenPorchSF | -2.872111e+00 | 1.518148e+01 | -0.1891852 | 0.8499746 |
| EnclosedPorch | 1.193628e+01 | 1.686386e+01 | 0.7078021 | 0.4791839 |
| X3SsnPorch | 2.038201e+01 | 3.139056e+01 | 0.6493039 | 0.5162466 |
| ScreenPorch | 5.596076e+01 | 1.719053e+01 | 3.2553247 | 0.0011592 |
| PoolArea | -2.908211e+01 | 2.380658e+01 | -1.2215996 | 0.2220611 |
| MiscVal | -7.313279e-01 | 1.854773e+00 | -0.3942951 | 0.6934221 |
| MoSold | -4.856806e+01 | 3.447724e+02 | -0.1408699 | 0.8879926 |
| YrSold | -7.796538e+02 | 7.024882e+02 | -1.1098461 | 0.2672526 |

In both the numeric and categorical linear regression tests, not all variables were significant. I decided to keep all variables in to see if those variables had any affect on the final results.

## 4.2 Machine Learning Modeling

### 4.2.1 Model Performance Metrics

[Table 2](#tbl-resulttable1) displays the relevant metrics for the regression models runs.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table 2: Performance Metrics for Machine Learning Models   |  | LNM | KNN | RFM | | --- | --- | --- | --- | | RMSE | 54819.808 | 45321.318 | 37742.731 | | Rsquared | 0.612 | 0.646 | 0.754 | | MAE | 20989.778 | 24633.264 | 19650.826 | |

The RandomForest model performed the best of the three models, outputting the best results by a fairly considerable margin, in all three categories. The KNN model performed the second best of the three, but was closer to the Linear Model’s results than to the RandomForest model.

[Figure 1](#fig-result1) shows the distribution of predicted values for each model, overlayed with the real observed values (OBS).

|  |
| --- |
| Figure 1: Predicted Values Compared to Observed Values |

[Figure 2](#fig-result2) and [Figure 3](#fig-result3) provide additional views comparing RMSE and MAE across the different models

|  |
| --- |
| Figure 2: Root Mean Squared Error Barplot |

|  |
| --- |
| Figure 3: Mean Absolute Error Barplot |

### 4.2.2 Variables of Importance

Finally, we can examine which predictors specifically improved prediction the most. That is, which had the greatest weight on the final result.

[Figure 4](#fig-result4) shows the Predictors that had the most impact on the Linear Regression Model

|  |
| --- |
| Figure 4: Linear Model Variables of Importance |

X2ndFlrSF indicates the variable Second floor square feet (X is added prefix to align with R naming conventions). BsmtFinSF1 represents the variable for Basement finished area square feet. BsmtUnfSF represents the variable for Basement unfinished area square feet.

[Figure 5](#fig-result5) shows the Predictors that had the most impact on the KNN Model

|  |
| --- |
| Figure 5: KNN Model Variables of Importance |

OverallQual is a categorical variable that rates the overall material and finish of the house in terms of “Excellent”, “Poor”, etc. GrLivArea represents the total square feet of the above ground living area. TotalBsmtSF is the total basement square footage.

[Figure 6](#fig-result6) shows the Predictors that had the most impact on the RF Model

|  |
| --- |
| Figure 6: RF Variables of Importance |

We can observe similar values to those highlighted by the KNN model, with the Neighborhood value appearing higher for the RandomForest model.

# 5. Discussion

## 5.1 Summary and Interpretation

The results generated in this project indicate that there is some quality predictive power attributed to the chosen machine learning models. While the performance metrics were not quite as impressive as would be desired, given the relatively small size of the training data, these results are substantial. In addition to these findings, the models were able to produce which variables provided the most weight towards the prediction power. The Overall Quality and General Living Area Square Footage were the two predictors that were listed in all three models. Total square footage for Basements was also a variable common in both the RandomForest and KNN models.

## 5.2 Conclusions

I attempted to develop different regression models that accurately predicted the prices of houses with minimal error. The three models that I ran all showed substantial predictive power for determining the correct sales price of a house, however, the performance metrics showed some room for future improvement.

The other goal of this project was to identify which variables could best aid in identifying correct housing prices. While there was some variance among the predictors in terms of importance, some predictors clearly stood out above the rest. This information could be used as a items of focus in future analysis of housing prices as well as features of importance for real estate firms to prioritize.

Further research would include utilizing different datasets to determine if these results are reproducible. Specifically, using a larger dataset with more observations could provide additional insights as well as improve predictability. Using housing data for a different general location could also be useful in determining whether the results presented here are indicative of most areas or if they are subject to localized bias.

# 6. References

IBM. (n.d.). *What is random forest?*

Kuhn, &. J., M. (2018). *Applied predictive modeling*.