Predicting Sale Price of Homes in Ames, Iowa

Kyle Jones (gtid3: 449, email: [kjones374@gatech.edu](mailto:kjones374@gatech.edu))

*November 28th, 2021*

*Data Mining and Statistical Learning (ISYE-7406), Fall 2021*



# Abstract

# Introduction

Accurate projections of a home’s sale price are relevant to those interested in buying or selling a home and to those aiding in the process (e.g., real estate agents, brokers). Some simple features related to a home’s sale price that are available in almost any housing dataset include square footage, location, number of bedrooms, number of bathrooms, etc. These variables alone often explain much of the variation in sale price, especially if you control for the year the house sold.1 However, there are many other attributes of a home that may be relevant to the sale price, that often aren’t available in housing datasets (e.g., proximity to park, foreclosure or normal sale, zoning classification of sale).

The Ames housing dataset contains 80 variables (23 nominal, 23 ordinal, 14 discrete, and 20 continuous) related to the quality and quantity of physical attributes of 2,919 home in Ames, Iowa that sold between 2006-2010.2 Most of the categorical variables contain many unique values (e.g., 25 unique neighborhoods), which when combined with the possibility of feature engineering (e.g., interactions between variables, transformations) makes it possible to create hundreds (maybe even thousands) of predictor variables. As a result, the Ames housing dataset was the perfect dataset to explore less commonly available predictors and assess their relevance to sale price predictions.

The primary goal of this project was to construct the best possible model to predict sale price using techniques taught in ISYE-7406. Due to time restrictions, I was not able to explore all possible predictor variables, so the models created used information from only 25 of the 80 variables available. Given unlimited time, I would have explored many more variables and most likely developed better models. This report will walk you through the methods I used to predict sale price, my strategies to create the best model, estimated sale price errors on unseen data, and actual errors on an unlabeled test set. Additionally, the relative importance of home attributes that I examined will be discussed.

# Data Source

This was a Kaggle competition dataset composed of a training set (N = 1,460) and a test set (N = 1,459), each containing 80 predictor columns:

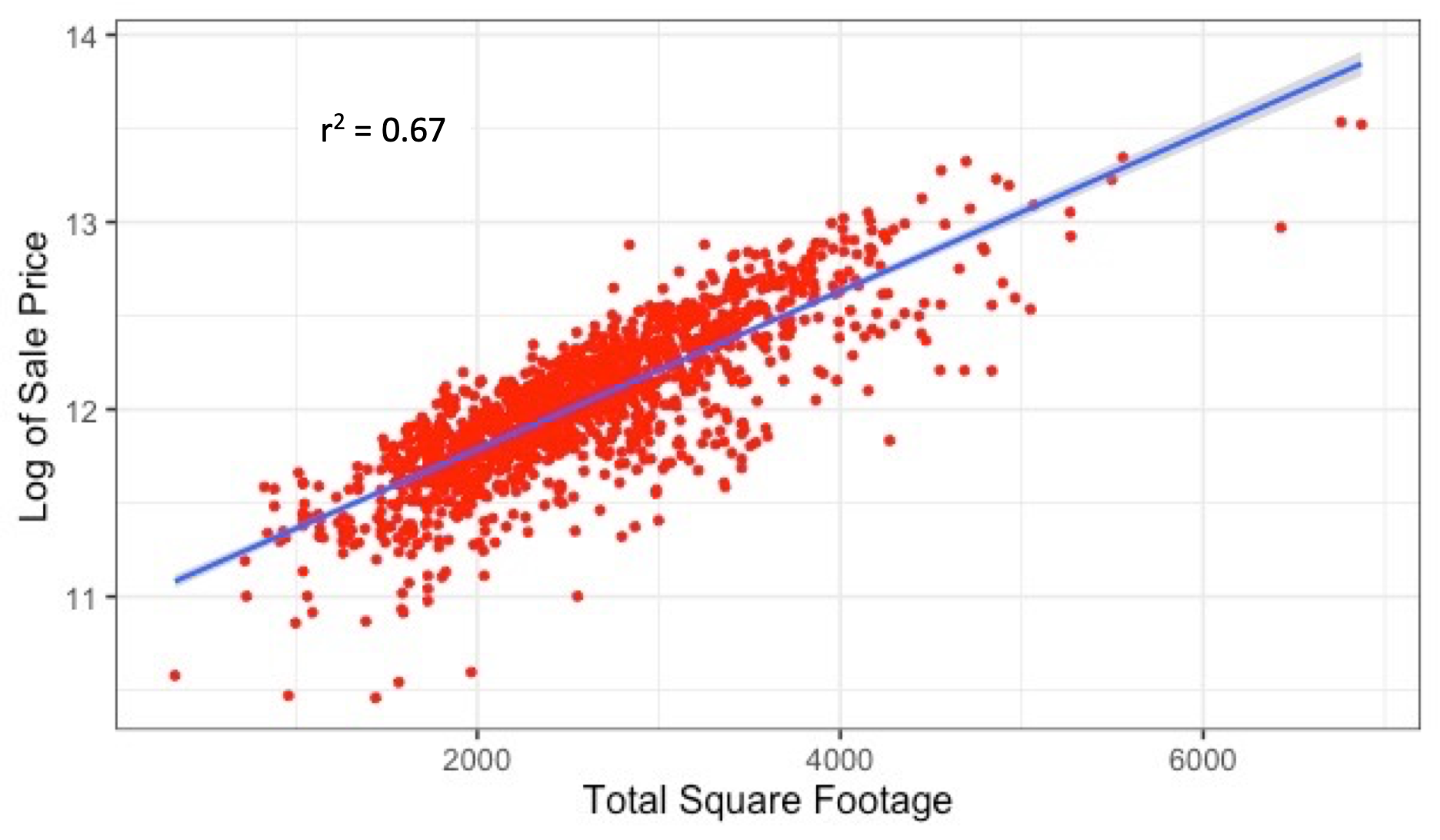
<https://www.kaggle.com/c/house-prices-advanced-regression-techniques>

A Kaggle competition dataset means that sale prices were only available for the training set. After development, the best model was used to make predictions on the unlabeled test set. Predictions were uploaded to Kaggle, which subsequently computed test error.

# Proposed Methodology

In the train set, square footage alone accounted for 67% of the variation in sale prices in a linear regression model (Figure 1).

**Figure 1.** Sale price as a function of square footage



Accounting for the location of the home increased the r2 value to 0.80. Other variables found to be significant predictors of sale price when included with square footage and location were:

* boundary between house and the nearest road (lot frontage)
* year of remodel or year built (whichever was more recent)
* exterior, heating system, and kitchen quality (5 level ordinal variables)
* number of bathrooms
* condition of sale (normal, foreclosure, between family members, etc.)
* basement in good condition
* number of cars that can fit in garage
* has a fireplace
* central air conditioning
* zoning classification of sale (commercial, agriculture, industrial, etc.)

Non-significant variables included in the models:

* has a fence with good privacy
* has a porch
* near a railroad
* near a park

Non-significant variables were included in the models because they had associations with sale price when tested individually. Additionally, non-significant variables in a regression model may hold more weight in non-linear models.

There were a multitude of other variables in the dataset not considered (e.g., type of road access, type of alley, shape of property) due to time constraints. With unlimited time, all variables should have been considered to maximize model performance.

The final train set included 41 predictor variables that showed some association with sale price during exploratory analysis. All numeric predictor variables were standardized before model testing since some models were sensitive to differing scales between variables. Sale price had a log transformation applied since Kaggle submissions were evaluated on Root-Mean-Square-Error (RMSE) between the logarithms of the predicted and actual sale price to ensure errors for expensive houses and cheap houses had equal weight.

Seven models were tested on the train set:

* Linear Regression
* Linear Regression Subset (using only square footage and location)
* K Nearest Neighbor (KNN)
* Lasso Regression
* Ridge Regression
* Random Forest
* A stacking ensemble method (average of Lasso and Random Forest predictions)

100-fold Monte Carlo cross validation (80% train, 20% validation) was performed on the train set to estimate which model would generalize best to unseen data. In each of the 100 Monte Carlo iterations, tuning parameters (e.g., lambda for Lasso and Ridge, K for KNN, number of trees for Random Forest) were optimized using 5-fold cross validation before making predictions. The model that had the lowest median test error among the 100 iterations was chosen as the best model, which ended up being the stacking ensemble method. Since the stacking ensemble consisted of Lasso and Random Forest, these two methods were retuned using 5-fold cross validation on the full train set before making predictions on the test set. Figure 2 shows the results of 5-fold cross validation on the full train set with Random Forest

**Figure 2.** Random Forest cross validated training error

Chart, line chart

Description automatically generated

# Analysis and Results

Figure 3 displays the error results across the 100-fold Monte Carlo cross validation experiment for each model.

**Figure 3.** Box Plot distributions of errors for 100-fold Monte Carlo cross validation, by model

Chart, box and whisker chart

Description automatically generated

The stacking ensemble method had the lowest median test error (0.130), so it was chosen as the method to make predictions on the test set. This method used the average predicted log sale price between the Lasso and the Random Forest methods. Lasso was included in the ensemble because it had the lowest median test error (0.134) among the linear methods and Random Forest was included because it is a nonlinear method that had a similar skill level to Lasso (0.136).

Table 1 displays the RMSE values on the train and test sets using the optimized ensemble method. It also includes the average deviation in $ between the predicted and actual sale price of a $160,000 home based on the RMSE value.

**Table 1.** Ensemble results on full train and test sets

|  |  |  |
| --- | --- | --- |
|  | **RMSE of log sale price** | **Average error on $160,000 house** |
| **Train Set (N=1,460)** | 0.09714 | +/- $14,811\* |
| **Test Set (N=1,459)** | 0.151 | +/- $22,424 |

\* exp(11.98) – exp(11.98+0.09714), where a log sale price of 11.98 corresponds to a $160,000 house, which was the average sale price of a home in the train data.

# Conclusions

# Appendix

# References