Predicting Sale Price of Homes in Ames, Iowa

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

November 28th, 2021

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

Ames, Iowa

Abstract

This analysis discusses statistical learning methods to predict the sale price of residential properties in Ames, Iowa between 2006 and 2010 using commonly known (e.g., square footage, neighborhood) and lesser-known predictors (e.g., zoning classification). Final models contained 41 predictor variables pertaining to roughly 25 housing attributes. The best model consisted of averaging the predictions of an optimized Random Forest model and an optimized Lasso model, referred to as the stacking ensemble in this paper. The stacking ensemble had a train set RMSE of 0.0971 between the predicted log sale price and actual log sale price, which corresponded to a median deviation of $8,167 in absolute price. For comparison, a Lasso model using only square footage and neighborhood as predictors had a median deviation of $15,473. The stacking ensemble had a test RMSE of 0.151 on an unlabeled test set, which ranked 2,752nd out of 4,895 among Kaggle submissions.

Introduction

Accurate projections of a home's sale price are relevant to individuals 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 and neighborhood. 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 house attributes 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, and discuss the errors of the best model on an unlabeled test set.

Data Source

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

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

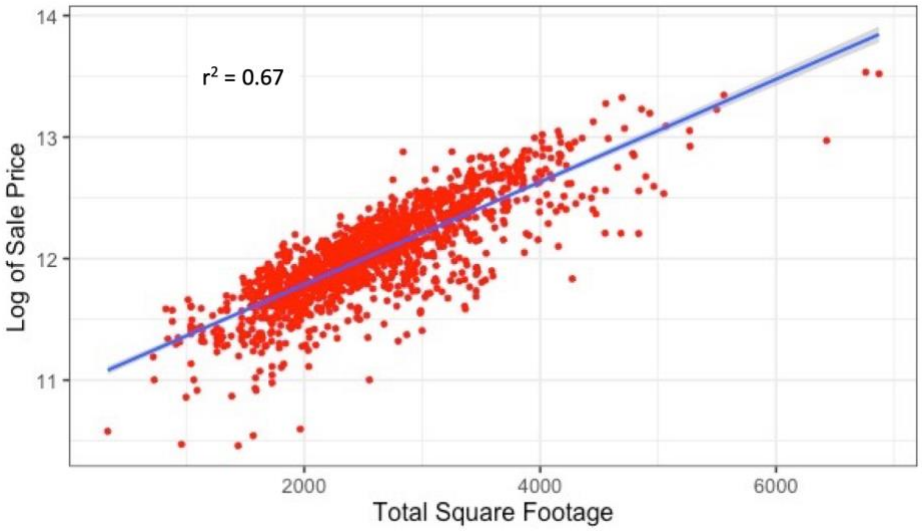
A Kaggle competition dataset means that sale prices were only available for the training set. After development, the best model was used to make predictions on the unlabeled test set. Then, predictions were uploaded to Kaggle so that test error could be computed.

Proposed Methodology

Variable Selection and Feature Engineering

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. Log sale price as a function of square footage



Accounting for the neighborhood of the home increased the r^2 value to 0.80. Other variables found to be significant predictors of sale price when included with square footage and neighborhood in a linear regression model 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 were:

- 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 from 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 would have been considered to maximize model performance.

Monte Carlo Cross Validation

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. This was done to ensure errors for expensive and cheap houses had equal weight.

Seven models were tested on the train set:

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

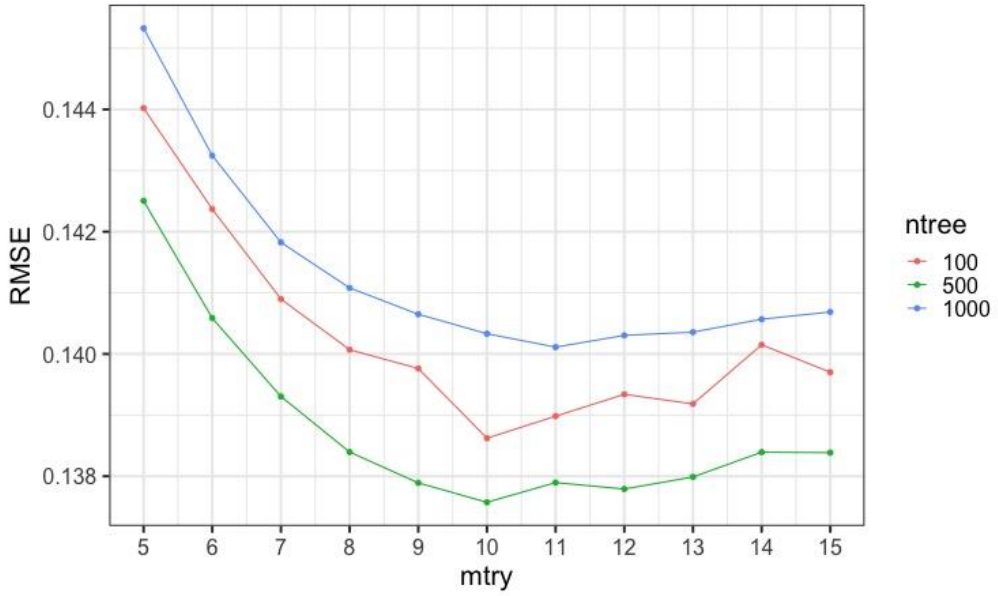
100-fold Monte Carlo cross validation (80% train, 20% validation) was performed on the train set to estimate which model would generalize best to unseen data. In each of the 100 Monte Carlo iterations, tuning parameters (e.g., lambda for Lasso and Ridge, K for KNN, number of trees for Random Forest) were optimized using 5-fold cross validation before making predictions.

Final Model Development

The stacking ensemble method, which consisted of the average predicted values between the Lasso and Random Forest methods, had the lowest Monte Carlo median test error among all models (Figure 3). As a result, the stacking ensemble method was used to make predictions on the unlabeled test set.

Both Lasso and Random Forest were retuned using 5-fold cross validation on the entire train set before making predictions on the unlabeled test set. Figure 2 shows the results of 5-fold cross validation with Random Forest.

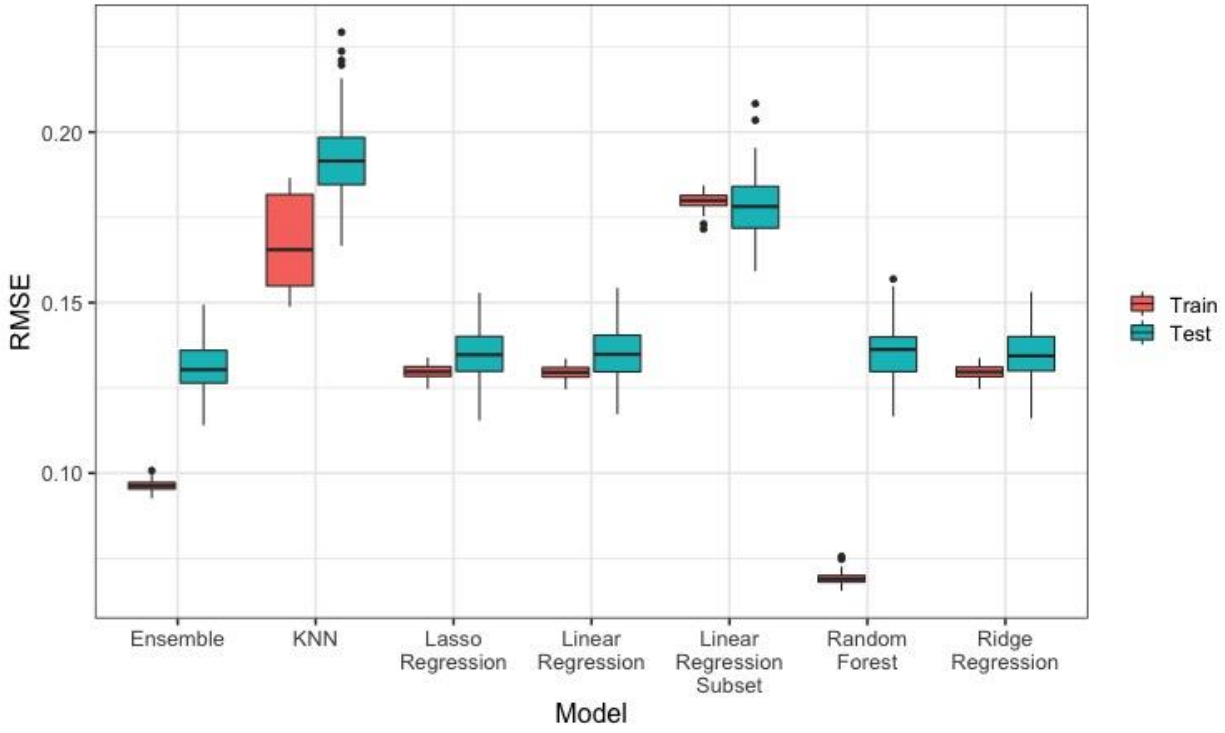
Figure 2. Random Forest cross validation errors



Analysis and Results

Figure 3 displays the error results across the 100-fold Monte Carlo cross validation experiment for each model. The stacking ensemble method had the lowest median test error (0.130).

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



* Linear Regression Subset: only used neighborhood and square footage as predictors

Table 1 displays the RMSE values on the train and test sets using the optimized stacking ensemble method and the linear regression subset method, which used square footage and neighborhood as predictors.

Table 1. RMSE result summary

	RMSE on Train Set (N=1,460)	**RMSE on Test Set (N=1,459)**
Stacking Ensemble	0.0971	0.151
Linear Regression Subset*	0.183	0.122

*optimized lasso regression model using square footage and neighborhood as predictors.

Discussion and Conclusions

Accounting for total square footage and neighborhood alone provided fairly accurate estimates of a home's sale price, with a median deviation between predicted and actual sale price of $15,473 (Figure A.1). This value corresponds to roughly a 10% difference in the actual sale price, considering the median value home in the Ames dataset was $160,000. Including multiple other home attributes improved prediction accuracy slightly, with the best model able to achieve a median deviation of $8,167 (~ 5% difference on $160,000 home). Some of the most relevant variables to sale price in Ames, Iowa besides square footage and neighborhood appear to be the number of bathrooms, year of remodel, kitchen quality, garage size, central air conditioning, fireplace, zoning classification of sale, and type of sale (Figures A.2 and A.3).

Monte Carlo cross validation identified linear regression, lasso regression, ridge regression, and random forest as similar methods in predicting home sale price based on RMSE between predicted and actual sale price (Figure 3). KNN performed particularly poorly with a median test error of 0.192, which was even worse than the linear regression model that used square footage and neighborhood as the only predictors. This poor performance can most likely be attributed to the large number of predictors relative to the small sample size, which is particularly problematic for non-parametric methods like KNN.[3] Spreading a relatively small dataset (train N = 1,460) across many predictor variables (p=41) often results in a phenomenon where a given observation has few neighbors, thus leading to a poor prediction. Prediction results for KNN could most likely have been improved by reducing the number of features to a handful of the most relevant.

Interestingly, averaging the predictions of the Lasso and Random Forest methods (stacking ensemble), resulted in slightly better predictions than any model by itself (Figure 3). This combination was the only one that resulted in better predictions, emphasizing that the models included in the stacking ensemble should be dissimilar in the manner they are created (e.g., linear vs. non-linear) but similar in skill level (e.g., RMSE). The stacking model could most likely have been improved further by including additional unique models with similar skill level. For example, support vector regression, boosting, and neural networks were not explored in this study due to time limitations. If any of these methods had resulted in similar or better RMSE relative to Lasso and Random Forest, their inclusion in the stacking ensemble may have improved prediction performance.

Variables included in the Ames housing dataset that were not explored in this analysis should be before further model development is attempted. The stacking ensemble resulted in an RMSE on the test set of 0.151, which ranked 2,752nd out of 4,895 Kaggle submissions (Figure A.4). Many of the models in the top 100 were using 100+ predicting variables whereas only 41 were used in this analysis. This highlights the need to spend substantial time creating and exploring all possible features in a dataset before building models. To guide feature exploration, observations with large deviations between actual and predicted sale price should be isolated and analyzed to determine if a missing predictor/s is the cause of the poor result (Figure A.1).

Lessons I have learned from course project

I have learned that most of your time as a data analyst/scientist will be spent cleaning and exploring the data, especially if the dataset is messy. Between log transformations and creating interaction terms, there are limitless feature engineering opportunities with a dataset that contains many predictors. Including all relevant features, or creating relevant feature from existing ones, seems to improve prediction accuracy to a greater degree than exploring alternative statistical learning methods.

I also learned that the difference between the most accurate model possible and a simplistic interpretable model is substantial with regards to time spent/effort, but marginal with regards to prediction accuracy. For example, models in the top 100 had a test RMSE of about 0.10, whereas my model ranked 2,752nd with a test RMSE of 0.151. This probably only corresponds to a median deviation in predicted sale price of a few thousand dollars.

Appendix

Code

All code and data files for this project can be found on my personal github account:

https://github.com/kylemjonesislanders/GT-7406-StatisticalLearning

The following R scripts are probably of most interest to the reader:

- Monte Carlo CV code: ./course_project/modeling/monte_carlo.R
- feature creation/cleaning: ./course_project/modeling/feature_engineering.R
- final model creation: . /course_project/modeling/final_model.R

Plots

Figure A.1 contains boxplots of the absolute deviations between predicted and actual sale price for the optimized stacking ensemble method and the linear regression subset, which only used square footage and neighborhood as predictors. Deviations were only calculated on the train set, since the test set was unlabeled.

Figure A.1. Absolute deviation between predicted and actual sale price on train set, by model

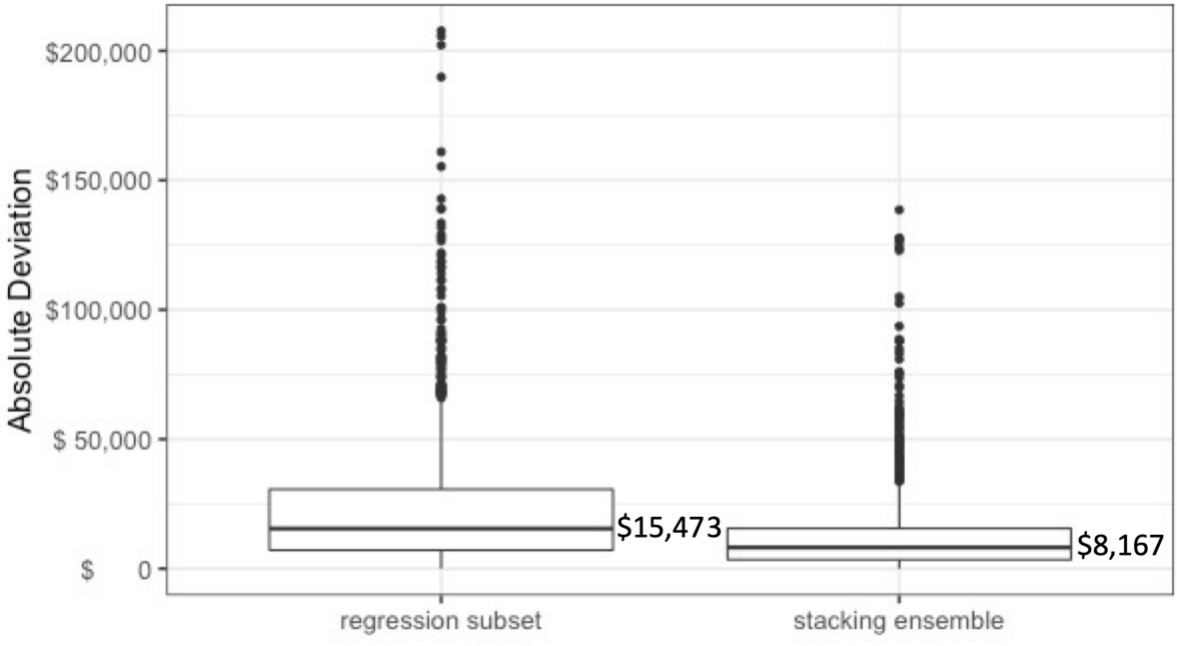


Figure A.2: Model coefficients for the final lasso model

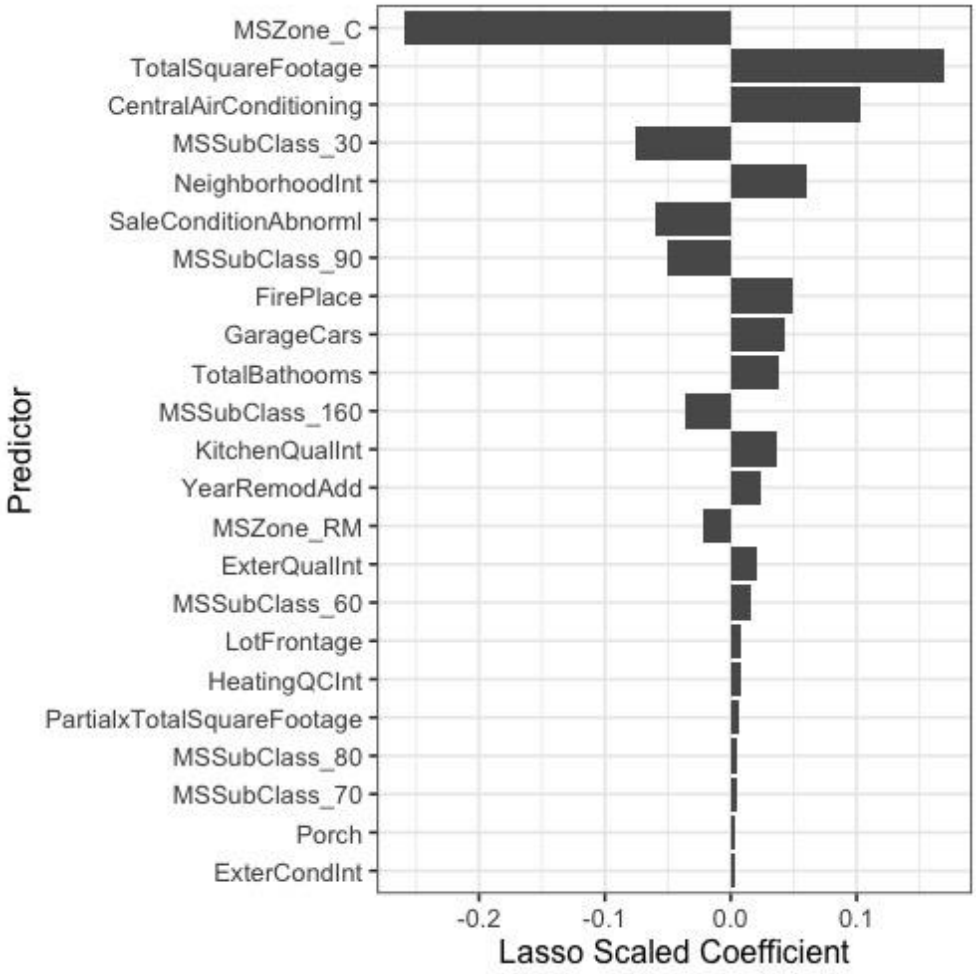


Figure A.3: Variable Importance of final Random Forest model

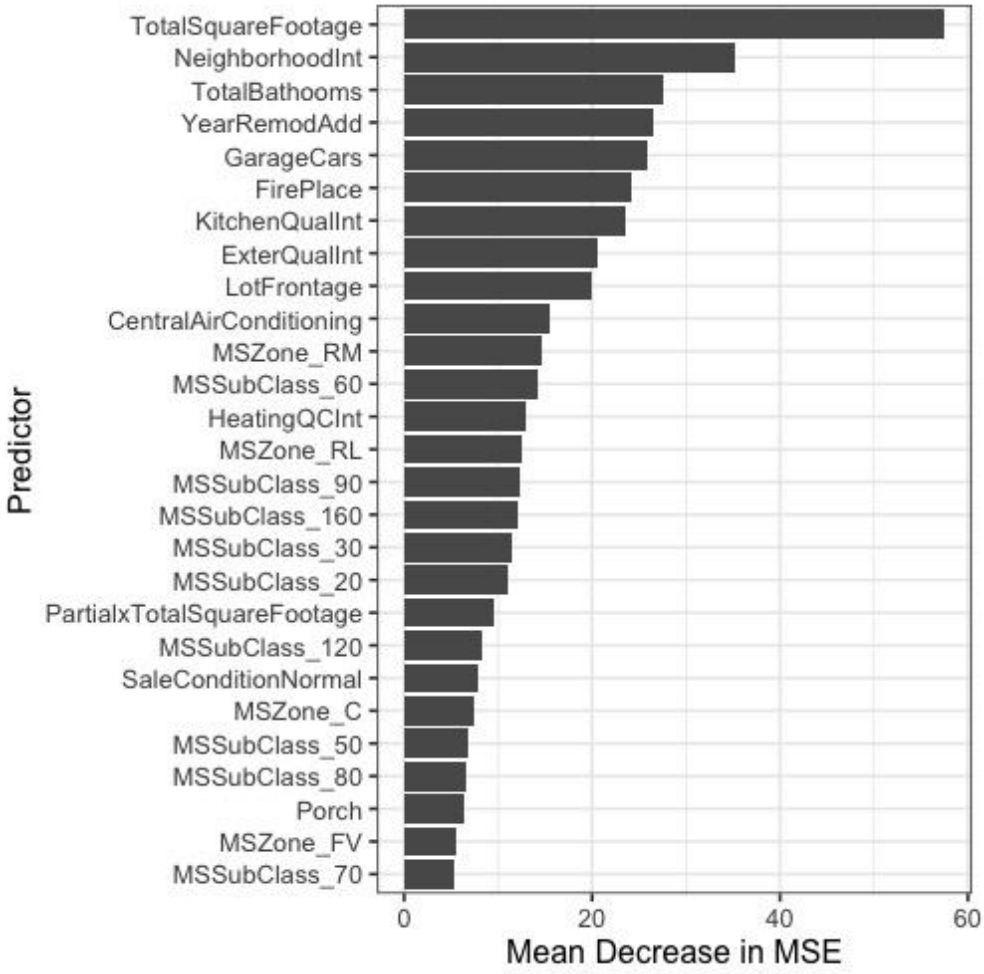
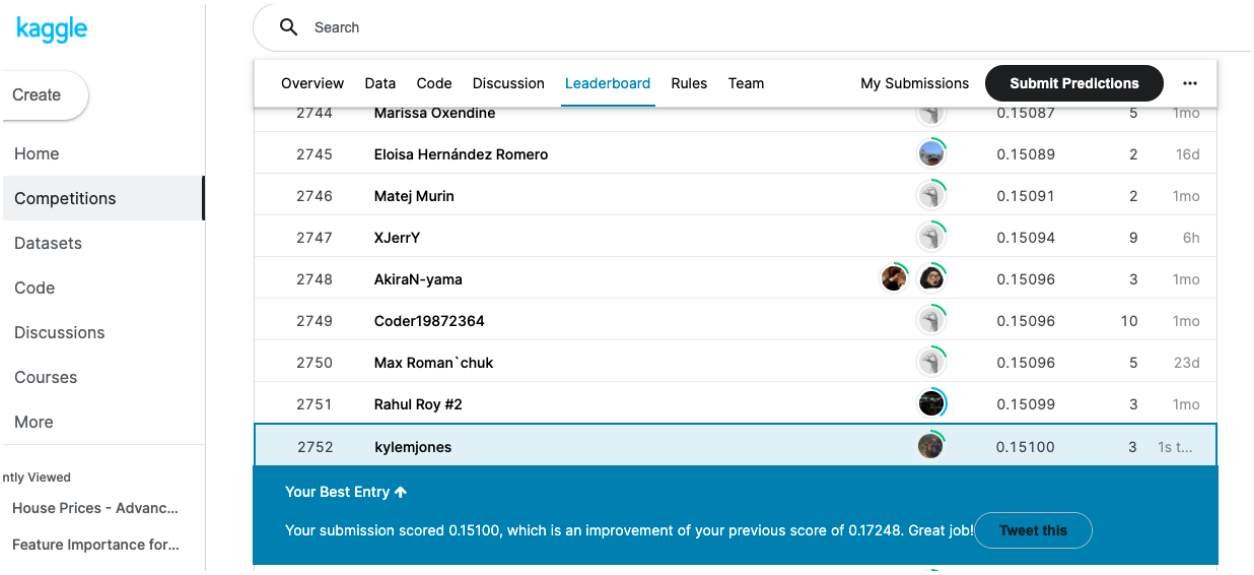


Figure A.4 Kaggle placement result



References

1. De Cock, Dean. “Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project.” *Journal of Statistics Education*, vol. 19, no. 3, 2011, https://doi.org/10.1080/10691898.2011.11889627.
2. https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data
3. James, Gareth. *An Introduction to Statistical Learning: With Applications in R*. Springer, 2017.