

## Project 2

Andrew Ross

2023-11-30

### Introduction

Congratulations! You've had the thought of becoming a home owner in Ames, Iowa. We can imagine you have many questions as well as wants about your dream home. The biggest concern will be the price. The price will go up the more wants you have. However, the realtor may not be able to give a precise answer to the cost. Here we provide a scientific approach to studying houses in Ames, Iowa to predict the sales price of the house. We present a XBTre Boost, Random Forest, and a Multiple Adaptive Regression Splines model.

### Data/ Exploratory Data Analysis

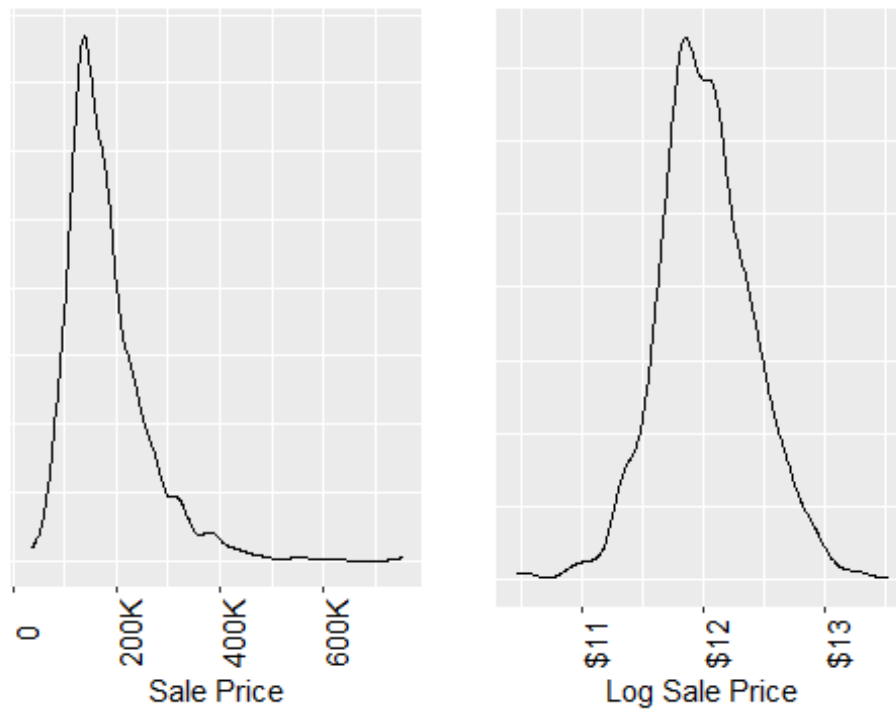
The housing data contains 1460 houses and 81 characteristics of the house. We analyzed each feature to see which feature is important.

First thing to note about this data set is that there are 6965 NA values (~6% of the data) that need to be addressed.

We assume if a feature is a continuous variable such as any feature related to 'area', then any NA value will be 0. We assume if a feature is a factor such as whether or not the house has a fence or not, then any NA value will be 'None' if it has more than 1 factor or 0 if the feature is binary. This can cause issues for some factor features because the features itself could have one NA value, so transforming it to 'None' implies another factor in the feature. This caused issues in the models because the 'None' factor had 0 variance. Therefore, when situations like this occurred, the features were removed because removing NA values was a problem for the Kaggle results. It required us to submit 1459 predictions.

The sale price of a house is the feature of interest. Below are two plots: one represents the feature itself and the second shows the feature with a log transformation.

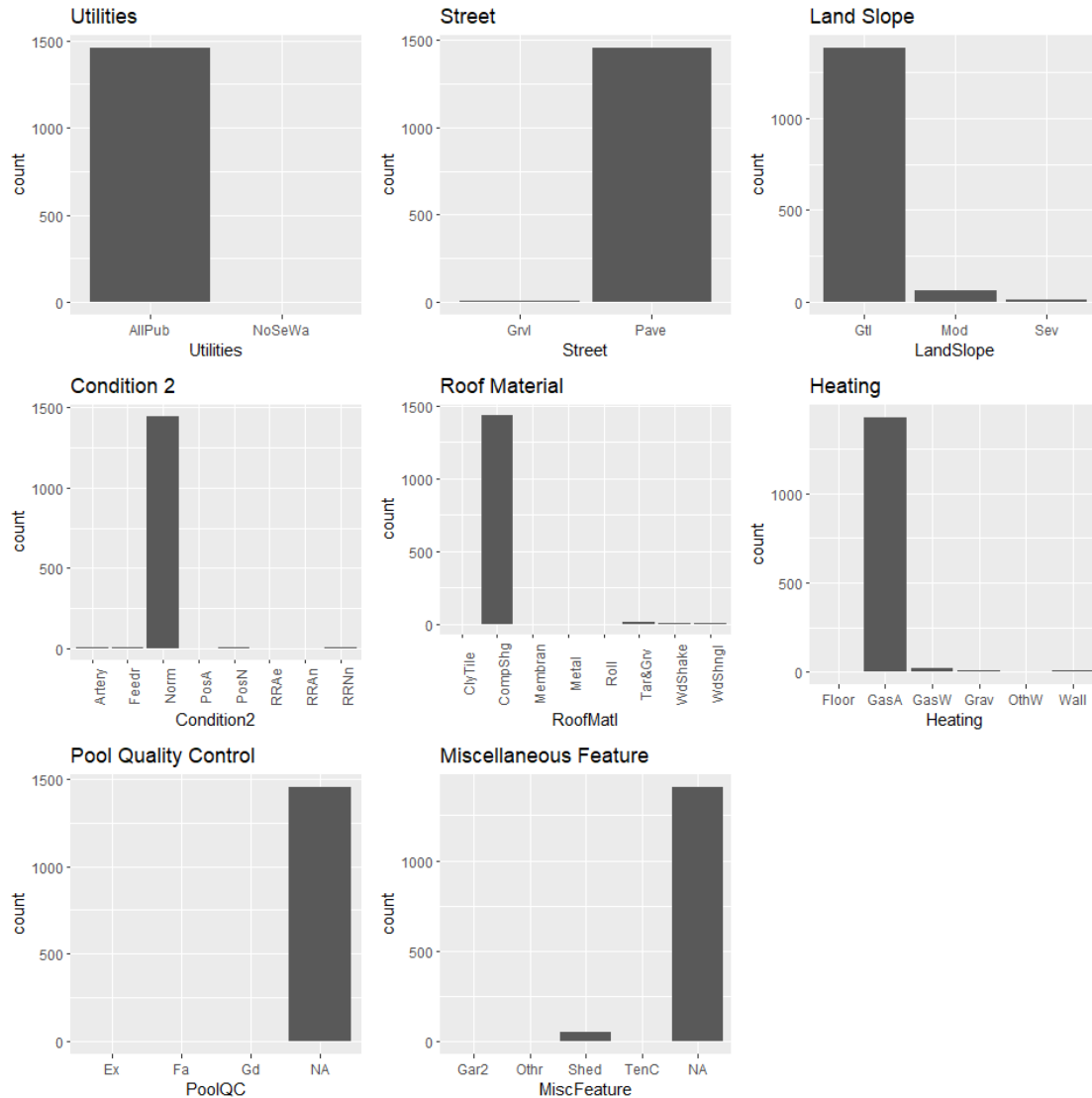
## Distribution of Sale Price



We can see a log transformation made the data more normally distributed. As such, Our model evaluation will be based on RSME on the log scale.

$$RMSE = \text{mean} \left( \left( \log(\text{predicted sale price}) - \log(\text{true sale price}) \right)^2 \right)$$

Second, some features were heavily skewed to one factor than another. Below is a plot of a few features:



We can see that each feature favors one factor compared to others. Therefore, Utilities, Miscfeature, poolqc, street, landslope, condition2, roofmatl, and heating were removed.

## Feature Engineering

Here is a look of the new features we created:

- **Total\_Bath:** The total number of baths in the house. This includes half baths.
- **fpyn:** whether or not the house has a fire place
- **remod:** whether or not the house had work done
- **Total\_area:** The total area of the house
- **Total\_Rooms:** the total bathrooms and rooms. Does not include basement rooms
- **HasPool:** whether or not the house has a pool
- **Total Porch Area:** The total area for all features related to porch.

- HasGarage: whether or not a house has a garage
- Overall: The overall total of overqual and overcond

The features that were used to create the new features were removed from the data set.

## Model Building

We split the training data into a train and test set to evaluate our models. Kaggle gave us a different test set without salesprice as a feature. The following are the models used: MARS, Random Forest, and XGTree Boosting. For each model, we use the 'caret' package to tune our models. The data was center, scaled, and we use 5-fold cross validation to help evaluate each model. The best random forest model we obtained used 40 features at each split yielding the lowest MSE to be 28474.90. The best MARS model we obtained uses 27 prunes and a degree of 2 yielding the lowest mse to be 33245. Finally, the best XGTree boost model we obtained uses minimum child weight to be 5, subsample to be 0.6, nrounds = 100, max\_depth = 5, eta = 0.1, gamma = 0, colsample\_bytree = 0.7. This resulted in the lowest MSE for all models to be 27644.56.

## Conclusion


Below is a table of the rmse for each model on the log scale:

Model	RMSE
Random Forest	0.137
XGTree	0.132
MARS	0.146

Here we see XGTree Boost had the best prediction accuracy.

For future work, it would be interesting to broaden this analysis to other cities, states, or countries. It would be nice to not need exactly the same number of observations in the Kaggle test set for submission, since individual observations could not be removed. Instead, a full column was removed. In the end, it may not matter as that feature was redundant.

Finally, a Kaggle screenshot is presented below:

1255	Philip Ross		0.13259	5	3h
------	-------------	---	---------	---	----