A statistical learning approach to predict the final sale price of houses

By

Nam Phuong Nguyen (40011040)

Tony Yuan (40029336)

Spyros Orfanos (40032280)

Concordia University

April 18, 2018

Contents

Introduction

The price at which a house is sold is the value both parties are willing to agree it is worth; however, knowing the actual value of the home gives one party the advantage to generate the most utility. Many factors such as lot area, house size, construction material, location, and year renovated influence the sale price of a house. By using this information in the context of a supervised statistical learning problem, one can attempt to predict the value of a house based on these different factors. Training a model that accurately predicts house prices is a very useful tool that would allow buyers and sellers to know if a house is priced at the fair market value based on the features of the house. By analyzing variance, correlation, residuals and other statistical measures, we will train several models to find the ideal model which provides accurate predictions of the sale price for a house in Ames, Iowa.

Our study uses methods such as Regression for a more structured and parametric model, and Tree based models to provide a more flexible model. Ensemble model stacking finally aggregates the results of each of the previous sections into one model to improve our predictive ability.

Dataset Description

The data set is the sale price of houses in the Ames, Iowa region between 2006 and 2010, and was found on Kaggle[1]. This data set contains the final sale prices of 1460 houses and 80 features were recorded varying between nominal and continuous types of variable. Each of these represent an aspect of the house that has sold.

Pre-processing

Data pre-processing was performed since there were many NA values. For categorical variables, the NA's were not an indication of missing data, but equivalent to not having a given feature, such as a fireplace. For numerical variables, the NA's did indicate missing data, and for simplicity, they were replaced with the mean of that numerical variable.

Excluded Predictors

Certain categorical variables such as PoolQC were excluded as predictors since there was one group with over ninety-nine percent of the data while all other groups combined had nearly zero exposure. Other variables like Street were also dropped due to very low exposure across many category levels. Due to these low exposures, the predictors are deemed insignificant to the modelling process. The other excluded predictors are LowQualFinSF, BsmtFinsSF2, MiscVal, PoolArea, EnclosedPorch, X2ssnPorch.

Categorical predictors with few small groups

There were certain categories that had good predictive ability but had certain group levels with extremely low exposure. This would cause issues in linear regression and would also cause issues in cross validation since the training model would not be aware of these groups if the data was split. We solved this by simply regrouping variables with less than one percent exposure into the level with the highest exposure for simplicity.

Categorical predictors appearing as Numerical

There were some instances where categorical predictors appear as numeric predictors. For example, MSSubClass is a variable that identifies the type of dwelling involved in the sale. This is a categorical

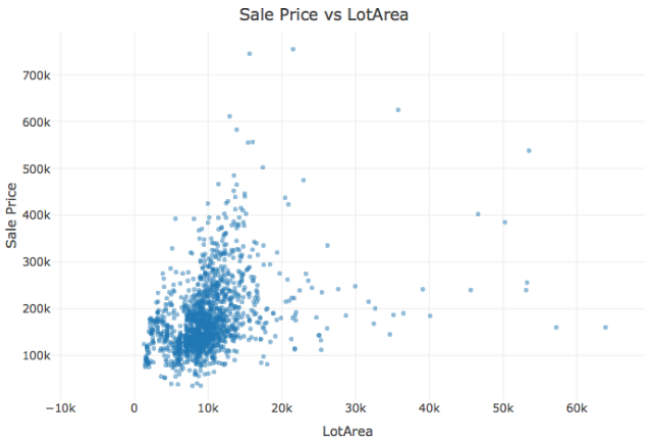
[1] https://www.kaggle.com/c/house-prices-advanced-regression-techniques

predictor, but different numbers are assigned for different types of dwellings. This means that the predictor should treated as a categorical variable instead of a numeric.

Some nice trends

By analyzing the scatterplots of Sale Price versus each of the numerical predictors, one can determine if certain predictor variables should be modeled using linear regression, polynomial regression, or another function or technique. Figures such as residual plots are also necessary to justify the validity of our linear modeling assumptions.

Figure 1: Scatter plot of SalePrice vs LotArea

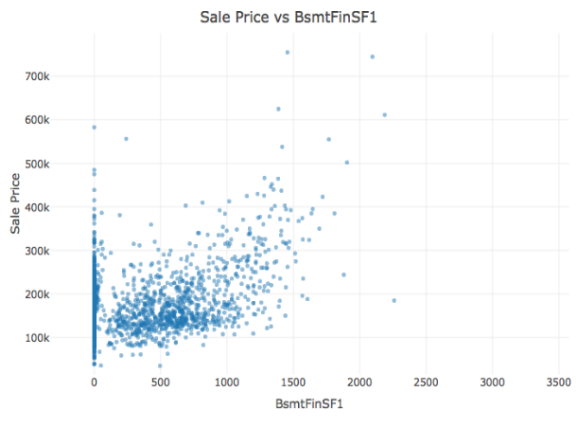


For example, Lot Area has a strong, positive correlation with Sale Price (Figure 1), but non-constant increasing variance. Therefore, a linear model may not be appropriate due to its assumption of constant variance, so we can instead train the model using other methods which have different or more relaxed assumptions.

Zeros

Certain variables in our data follow a clear trend for values greater than zero but have many repeated observations at zero which vary significantly. For example, the predictor "Basement Square Feet" is a continuous numerical variable; however, many houses do not have a basement, so the value of the predictor is often zero. The scatterplot (Figure 2) indicates that houses without basements do not necessarily have a lower sale price than those with a basement, but for the houses that do have a basement, there is a positive relationship between basement area and sale price. If we simply fit a linear regression model using all the data points for predictor variables of this nature, the model will not be flexible enough to capture the true trend since the points at zero will distort the regression curve. Alternative approaches which account for this situation will be

Figure 2: Scatter plot of Sale Price vs BsmtFinSF1



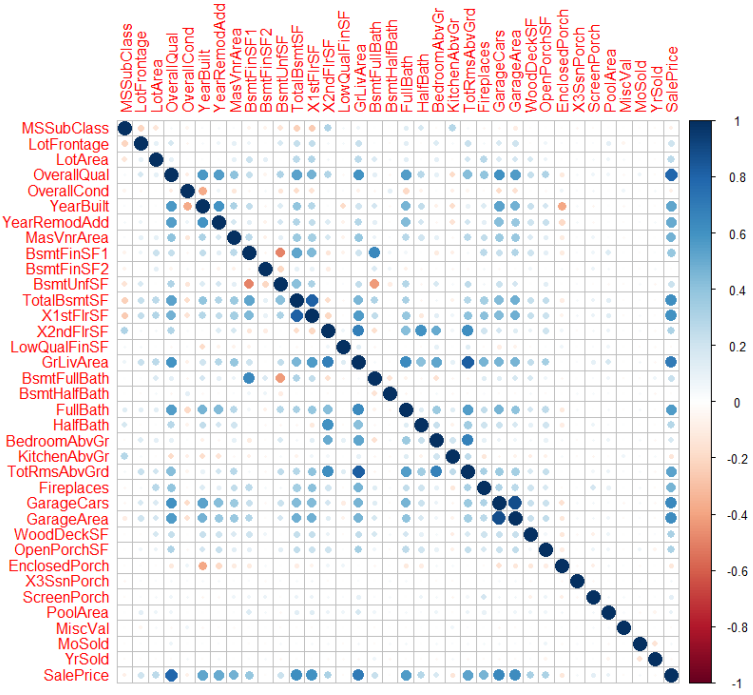
discussed further in the analysis component of the report. Other variables of this nature are MasVnrArea, BsmtFinSF1, BsmtUnfSF, TotalBsmtSF X2ndFlrSF, Garage Area, WoodDeckOpenPorch, SfScreenPorch.

Highly Correlated Predictors

There are also some predictors which are highly correlated with other predictors. For example, there is a variable for the area of a garage and another for how many cars fit in the garage. Another example is basement area versus ground floor area: given that the house has a basement, the basement is usually the same size as the ground floor (correlation of 0.89). Since these variables are so highly correlated, excluding them will reduce the complexity and parsimony of the model while still giving accurate predictions. Using parameter selection methods like AIC and BIC, we can reduce the number of predictors to include in the model.

By looking at Figure 3, we can see the quantitative predictors which have the most impact on the sale price are OverallQual, GrLivingArea, GarageCars, GarageArea, TotalBsmtSF, X1stFlrSF, TotRmsAbvGrd, Fireplaces, FullBath, YearBuilt, YearRemodAdd. These are the quantitative features that we will concentrate on the most to build our model.

Figure 3: Correlation between Numerical Variables



Analysis

There are many methods that were used to predict the sale price, but there is a tradeoff for these different models. Some techniques are less flexible and have lower variance, but a higher bias such as linear regression while some techniques being more flexible, were harder to interpret. The techniques used for the regression problem include, stepwise regression, ridge, polynomial regression, generalized additive models, smoothing splines, trees, boosting and stacking.

Performance Measure

After data pre-processing, we decided to separate our data into two sets where 70% of the data is used to train the models and perform cross-validation and 30% to test our model. After modeling our data with the training set, we use the out-of-sample mean-squared error (MSE) on our test set to assess the predictive ability of each statistical learning method. Throughout the report, the calculated MSE for each method is refered to qualitatively, but the actual values are summarized in Section 4 at the end of the analysis.

The MSE is calculated by predicting a sale price for each observation in the test set using the given model and the following formula

$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2$ where y_i is the actual sale price of the ith observation in the test set and

$\widehat{y_i}$ is predicted sale price of the ith observation in the test set and n is the number of observations in the test set.

1. Regression Approach

1.1. Linear regression

The first and most simple model considered is the linear regression model which accounts for the linear relationship between all the predictors and the response variable. This provides us with a good, non-flexible starting point to evaluate our predictors. Linear regression relies on many assumptions, so the first step is to check whether our data follows those assumptions such as the normality of the response variable. Plotting the histogram of the sale price shows a positively skewed dataset, hence we use the log-transformed data as its distribution is approximately normal.

Figure 4: Histogram of Sale Prices

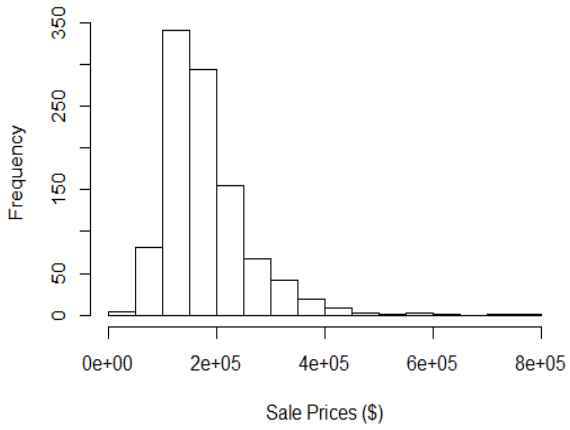
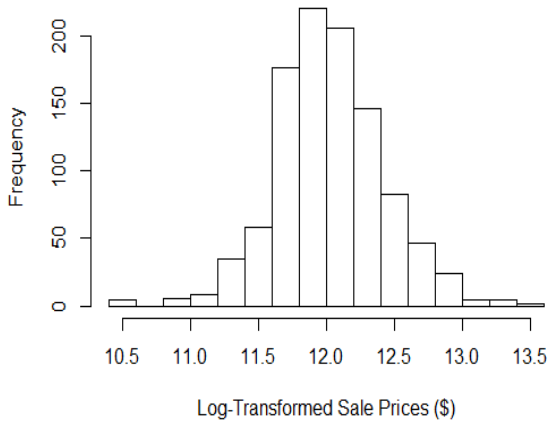


Figure 5: Histogram of Log of Sale Prices



After transforming the data, a hybrid stepwise approach was performed to select the variables most important in our model for the sake of parsimony. This method successively adds and removes predictors as long as our performance criterion improves. In this case, we use the AIC and BIC as performance criterion. Table 1 helps us keep track of the important predictor variables included in the model, which we will compare with more flexible methods later on. Note that the AIC performed slightly better than the BIC when it comes to testing the out-of-sample MSE with our test set.

When performing residual analysis regarding this model, we do not observe patterns in the residual versus fitted values plot, which can be seen in Figures 6 and 7. This signifies that our residuals may be normally distributed and independent and identically distributed meaning that linear regression was an appropriate model to use thus far.

However, as we noticed previously during our dataset analysis, some features have non-linear relationships with the response variable. We take this into consideration in our later models which use different approaches like polynomial regression to better fit our data.

Figure 6: Residuals vs Fitted Values with BIC

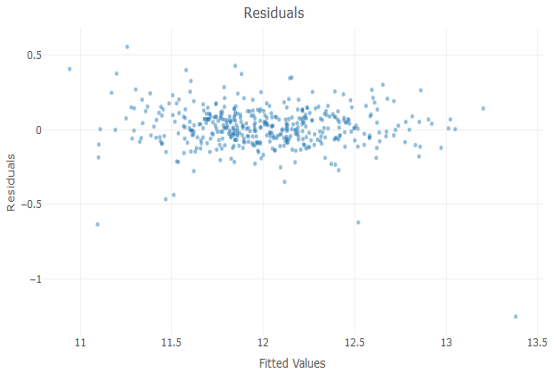


Figure 7: Residuals vs Fitted Values with AIC

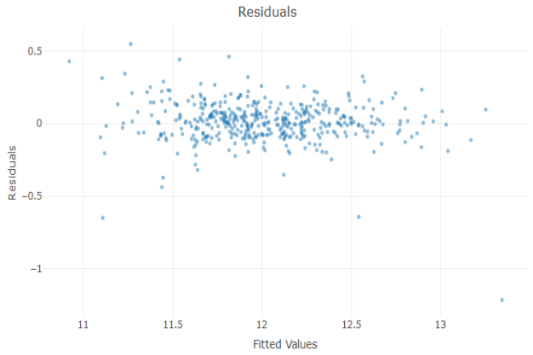


Table 1: Summary of Results from Linear Regression

		Features Included
Linear Regression	AIC	OverallQual, Neighborhood, GrLivArea, MSSubClass, BsmtFinType1, GarageCars, OverallCond, BsmtExposure,YearBuilt, CentralAir, SaleCondition, KitchenQual, FullBath, BsmtFullBath, Fireplaces, Condition1, HalfBath, Foundation, Exterior1st, WoodDeckSF, ScreenPorch, LandContour, OpenPorchSF, Heating, HeatingQC, LotFrontage, YrSold, BsmtQual, BsmtFinSF1, LotArea, MiscFeature
	BIC	OverallQual, GrLivArea, Neighborhood, BsmtFullBath, OverallCond, GarageCars, YearBuilt, BldgType, BsmtExposure, CentralAir, SaleCondition, FullBath, Fireplaces, ScreenPorch, WoodDeckSF, X1stFlrSF, Heating, OpenPorchSF

Categorical Variables

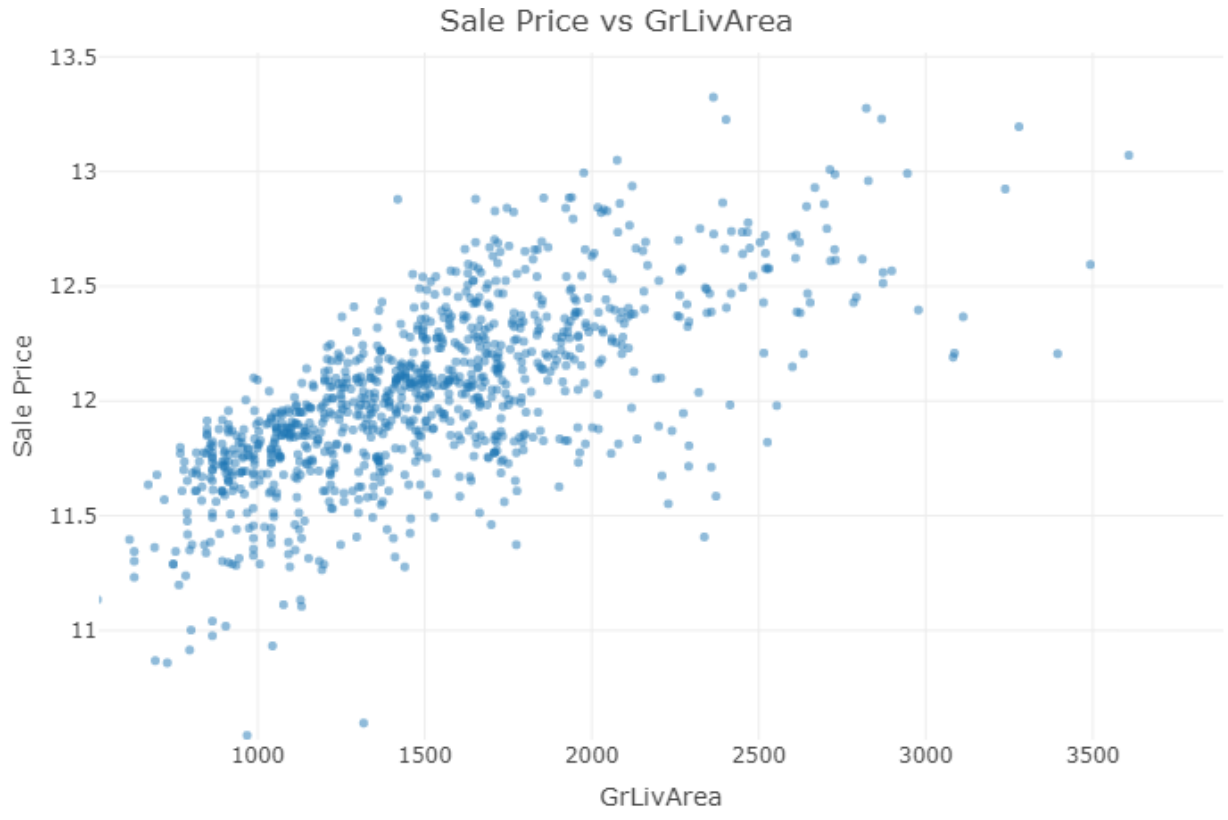
Important categorical values were identified by using the hybrid stepwise approach with BIC as our step criteria and only considering categorical variables in our steps. The variables kept were Neighborhood, KitchenQual, GarageFinish, BsmtExposure, MSSubClass, FireplaceQu, CentralAir, BsmtQual , ExterQual, Heating, RoofStyle, HeatingQC. Due to the fact that adding categorical variable has a high penalty in our BIC criteria since n-1 betas are added for n groups, having a list of these categorical variables is beneficial for when we build our full linear model later on.

1.2. Polynomial Regression

The next appraoch was to use polynomial regression for the numerical predictors which have more than ten unique predictor values. Polynomial regression was considered because it provides a model that is much more flexible than the previous linear model that was trained. Due to the stucture of our

data and having many repeated observations at zero, using a flexible model is appropriate. Also looking at the scatter plots of our logged(SalePrice) vs important predictors, we can notice non-linear patterns such at the one with GrLivArea. We found the polynomial of optimal degree by minimizing the out of sample MSE through k-fold cross validation. Note that the highest order polynomial conisdered was of fifth order. Analyzing the resulting curves and residuals plots allowed us to check if the polynomial regression is reasonable and fits the data properly. The results of the k-fold validation indicated that for many predictors, polynomials of degree five minimized out of sample MSE. However, this is the polynomial of maximal order that was tested, which indicates that polynomials may not fit all the predictors appropriately and is still not flexible enough. There were other polynomials whose optimal order was much more reasonable. For example, the predictors Greater Living Area, Year Built and Lot Area were fit using cubic polynomials. In our final model, we used cubic polynomials for these three predictors; however, other methods such as piecewise regresssion and smoothing splines will be used for the remaining predictors where polynomial regression is inappropriate.

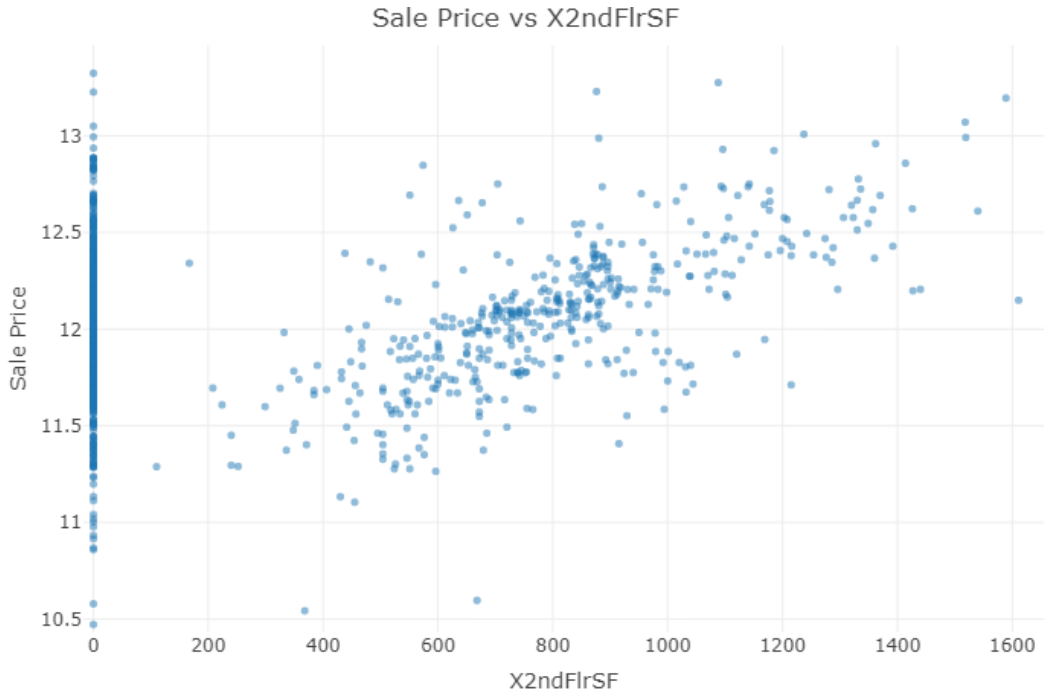
Figure 8: Scatterplot of Sale Price vs GrLivArea



1.3. Piecewise Regression for 'Censored' Predictors

As discussed earlier in the data analysis, there were several predictor variables which followed clear trends for all non-zero points. This type of data will be referred to as 'censored', though it is not truly censored – there are just many points at zero because the feature does not exist for the house. The previous models could not sufficiently fit curves to these censored predictors as they were not flexible enough around the zero-valued points. To remediate this issue, piecewise regression models or regression splines were used to split the predictor into two parts: where it equals zero and where it is strictly greater that zero. This method, compared to polynomial regression, allows us to capture the true trend more accurately without needing as many parameter estimates.

Figure 9: Scatterplot of Sale Price vs X2ndFlrSF



By placing the breakpoint at the minimum value for this censored data we were able to partition our data and perform a segmented regression. We remedied this by adding an indicator variable on whether it was at the breakpoint or not.

Using the techniques of Polynomials and Piecewise regression we were able to build our best parametric model by considering all our scatter plots and boxplots for variable selection. The next step was to compare this model with a regression splines.

1.4. Generalized Additive Models

The generalized additive model was used without a link function since we were satisfied with the normality of our log-transformed response variable. The same variables that were used in our best polynomial and segmented regression were used in the GAM. The main purpose of using a GAM was to see if it could use smoothing splines to deal with our censored predictor variables.

For the GAM package, smoothing splines require a value input for the degrees of freedom (df) of the spline depending on how flexible we wanted that variable to be. Out of sample MSE was used to determine the optimal degrees of freedom for the splines. This approach was more manual as the scatter plots for each of the censored variables included in our model were examined, and then a starting degree of freedom was decided upon. As seen previously, the X2ndFlrSF looked like it could be modelled using a polynomial fit, so we iterated through 1 and 20 degrees of freedom, as seen in the table below.

Table 2: Degrees of Freedom vs out of sample MSE:

1	522045815
2	511594687
3	495170722
4	478483449
5	465052281
6	454011692
7	445125348
8	438397944
9	433709626
10	430705462
11	429038157
12	428359850
13	428399291
14	428954223
15	429872164
16	431052585
17	432426875
18	433940563
19	435566071
20	437276751

2. Tree Models

2.1. Tree Models Advantages over Linear Models

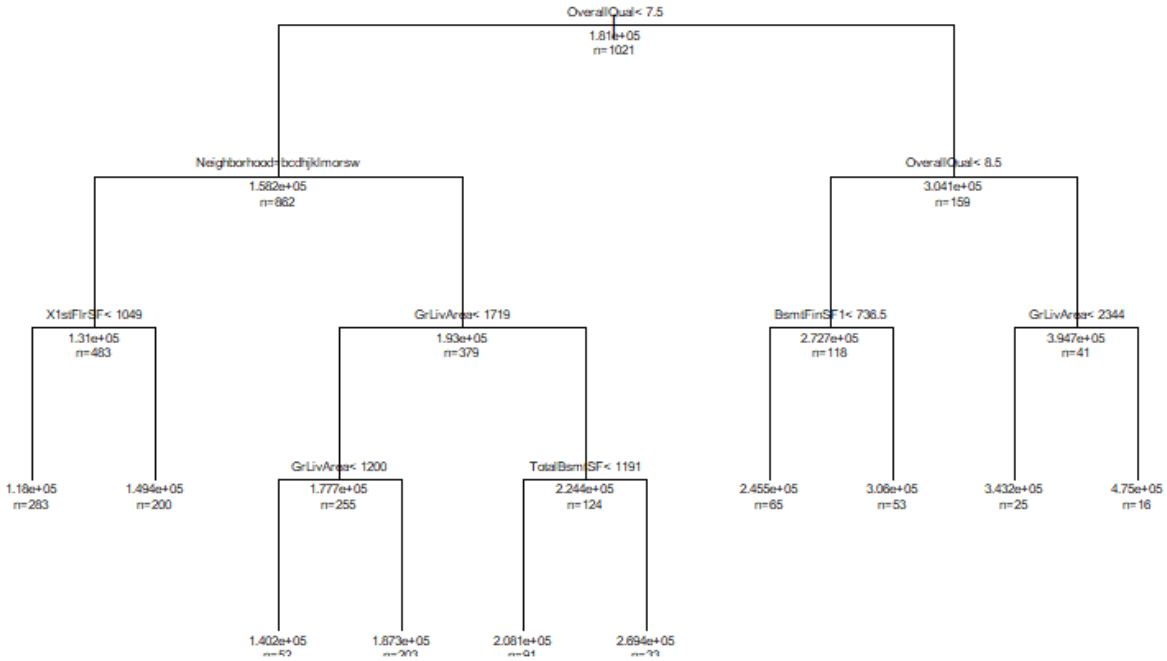
Due to the constraints on linear models, replicating non-linear patterns becomes difficult, and while polynomials, piecewise, interactions and splines can be used, a model involving trees does not require assumptions on the data, which allows the model to replicate patterns in the data more naturally and accurately. These trees work by dividing the p-dimensional data space into p-dimensional rectangles. These techniques will be explored to improve our predictions and variable analysis.

2.2. Greedy Rpart Tree

Our most simple tree model confirms our hypothesis of which predictors were correlated with Sale Price. The most important predictors, which are the main branches of this tree, are also the main quantitative predictors we wanted to prioritize. Furthermore, when comparing with regression, the predictors kept by variable selection are also seen in the tree.

As we can observe, Overall Quality is the predominant feature for predicting the sale price and was the first split chosen by the greedy algorithm. Due to our small data set of only 1000 lines, this method does not offer many distinct sale prices, but it does provide a range in which the sale price of a property would be based on only considering a few variables. Hence, it can be applicable in real life since a seller would be able to estimate roughly how much his house would sell for in the end. Since the prediction is not very flexible, the MSE is very high as it only predicts ranges of house prices.

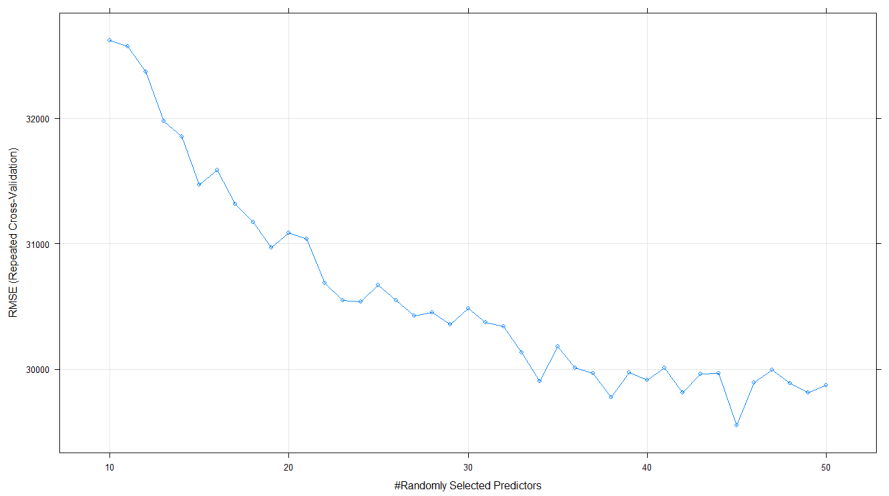
Figure 10: Regression Tree for Sale Prices



2.3. Random Forest

The Random Forest approach bootstraps different variables to grow many uncorrelated trees. In our case, 500 different trees were grown and the average of these trees was taken as the prediction values for this method. The optimal number of variables to be included for each random tree was optimized at 45 using k-fold Cross validation, as seen in Figure 11. When looking at the out of sample MSE, this model provides a significant improvement over the basic tree model.

Figure 11: K-fold Cross Validation on Random Tree

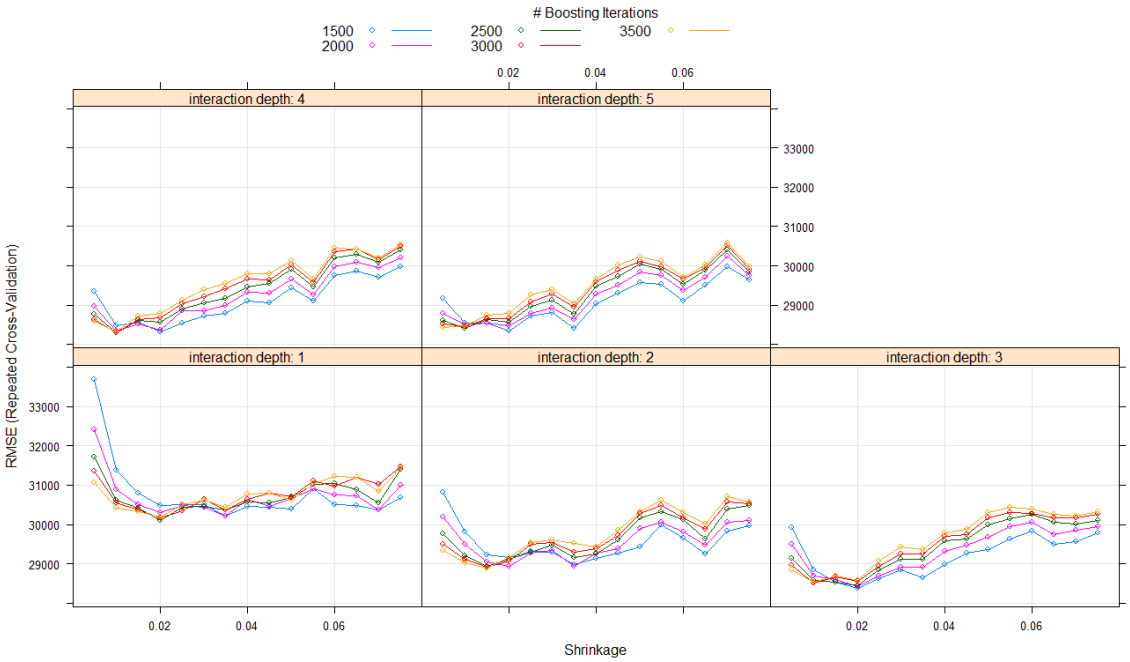


2.4. Gradient Boosting

Gradient Boosting is a tree based algorithm that builds off trying to minimize the loss function by using many "weak" trees. In our approach, we are using the MSE as the Loss Function. The algorithm is set such that we have a starting function as constant. We then set a maximum tree length and we commence our iterations. The trees are built sequentially such that the function is "learning" on each iteration based on the gradient of the previous function.[2]

Using the gradient descent method and our assumption of a Gaussian disribution we are able to find the Gradients as $Grad$: $z_i = y_i - f(x_i)$. So with a normal assumption, we can sequentially improve our model by adding new trees, which will be modelled on the residuals. This method could lead to overfitting so it is important to find optimal parameters such as the learning rate, number of trees and tree depth which can all be done through cross validation. Our optimal values that we found for our graident boosted machine was having a learning rate of 0.01 as the learning rate, 3000 trees grown, an interaction depth of 4 and a minimum number of observations of 15 at a terminal node as seen in the figure below.

[2]Paper written by Greg Ridgeway: http://www.saedsayad.com/docs/gbm2.pdf


Boosting Iterations
1500
2000
2500
3000
3500
interaction depth: 4
interaction depth: 5
interaction depth: 1
interaction depth: 2
interaction depth: 3
RMSE (Repeated Cross-Validation)
Shrinkage
0.02
0.04
0.06
29000
30000
31000
32000
33000

3. Ensemble Model Stacking

Model stacking is a technique we use to gain better results from the models we previously developped. The technique requires us to have a first level of models which are of different types, in our case, the GAM and Gradient Boosting Method. First, we make predictions using our training set. These predictions become the features for our model's second and final level. This technique is useful since different models have different strengths. Tree models tend to be more flexible than linear regression models, but tree models perform poorly in extrapolation compared to linear regression.

We selected our best linear regression model and our gradient boosted machine for the two predictions that will be used for our second layer model. Our second layer model was chosen as a Ridge Regression and we did not standardize the variables since all predictors and response variable were scaled the same. We performed cross validation and received various values of lambda, but after multiple runs all values were between 0 and 0.05. We decided to use 0.04 as the optimal penalty term since it was the model chosen most consistenly.

Comparing the out of sample MSEs of our Gradient Boosted Machine, GAM and Second Level Ridge, we are able to see that this model stacking does provide us with significant improvement. The beta values we obtained for our Ridge Regression are seen below.

Table 3: Summary of Ensemble Stacking MSE

Model	Out-of-sample MSE (in millions)
GBM	478
GAM	521
Second Level Ridge Regression	455

Table 4: Beta Coefficients for Ensemble Stacking Model

Model	Beta Coefficients
GBM	0.426
GAM	0.227
GBM:GAM (interaction)	0.0136

4. Summary of out of sample MSE

Model Name	Out of Sample MSE
AIC Stepwise	976290016
BIC Stepwise	1081796100
AIC Stepwise w/ manual variable selection	897591174
Polynomial + Piecewise	521363257
GAM & Splines	428359850
Simple Tree	1845585187
Random Forest	799182390
Gradient Boosting	478551285
Ensemble Stacked	419932368

Conclusion

In order to predict the final sale price of houses in Ames, Iowa, many methods were considered in our study. The first step was to determine which of the 80 initial features were the most relevant in order to create a parsimonious and comprehensive model. Linear regression allowed us to reduce the dimensionality of our model and check where homebuyers put the most importance when looking at a property. Tree models allowed for a less structured model to describe each predictor. Stacked model permitted to combine models to build the strongest model possible.

The best model was the Ensemble Model as it had the lowest out of sample MSE. The lowest MSE for an individual model however was a GAM due to its flexibility. Unfortunately for us to achieve a high performance with a GAM, it required a lot of variable selection, verification of assumptions, dealing with collinearity and curve fitting. While we were able to achieve a higher result with the GAM the difference in how to model made using a GBM much easier since most of the work was done by the computer.

The model stacking was the most interesting considering that despite having two models that we were no longer able to improve upon individually, we were able to combine the results to further our predictive power. It would be interesting to explore how many levels of model stacking could be achieved to maximize performance and which different types of models perform best together as a stack.

Although the predictive ability of this model may be limited since it is based on a specific region, the method could be used anywhere in a developed country since most people put importance on the same qualities before buying a property.