House Prices: Advanced Regression Techniques

Andy Dingler
Dong Bing
Predict 413
Final Project

Problem Under Consideration

In this project, we are competing in the Kaggle.com Home Prices: Advanced Regression Techniques project. The goal is to evaluate home sale prices based on date sold, and 79 other explanatory variables that describe each aspect of a home. The dataset analyzed describes the sale of individual residential properties in Ames, Iowa from 2006 to 2010. There are different analytic techniques available to predict the sales price of a home. For example, the prediction of a house's sale price could be solved using time series forecasting, or a more unsupervised approach that uses complex decision trees. To evaluate the accuracy of the models, we split the data randomly into a train and test data set and used the Kaggle competition test data set as a hold out data split.

The ability to accurately predict the price of a home is significant for everyone in the world that owns a home or is interested in buying one in the future. The purchase of a home is usually the single biggest expense in someone's life, and dramatically overpaying can considerably change someone's financial stability. In this report, different features of the house are used to describe the overall cost of the house. In reality, these features are unique to each individual buying the home because everyone has specific preferences that are unique to them which can drastically change their perceived value of the house. The common conception in real estate, is that location, location, location is the single most important feature in a house's value. This report will put a magnifying glass on this concept and zoom into a much deeper level to investigate each and every aspect of the house. The understanding of how various characteristics impact the bottom line can arm home buyers with additional leverage throughout their home buying process, especially during the price negotiation stage.

Data and Data Preparation

The Ames housing dataset is compiled by Dean De Cock for use in data science education. The data set describes the sale of individual residential property in Ames, Iowa from 2006 and 2010 and can be aquired on Kaggle.com The data set has 80 different variables, including the predicted variable, SalePrice. Of the 80 variables, 23 are nominal variables, 23 are ordinal variables, 14 are discrete variables and 20 are continuous variables that can directly contribute to the sales price of a property by describing the quantity and quality of many physical attributes of the property. The Ames data set came with a lot of variables with a significant percent of the values missing. After carefully analyzing the missing value variables and the data dictionary, a majority of the variables can easily be imputed using the other variables in the data set. This is because a lot of the variables are missing values due to the houses not having the feature. For example, the five garage variables that have missing values can be imputed by the GarageArea variable. If the GarageArea variable is equal to zero, then the house does not have a garage, and the five garage variables should be zero. This same technique can be used to impute the five

basement variables using the TotalBsmtSF and the missing FireplaceQu using the Fireplace variable.

Unfortunately, the LotFrontage variable is missing 17% of its observations and does not have one variable that can easily be used to impute the missing values. We could assume that if the LotFrontage variable is missing, then the house does not have land area directly next to the street, but instead we used the MICE library in R. The MICE library stands for Multivariate Imputation by Chained Equations (MICE) algorithm and "imputes incomplete multivariate data by chained equations, which is a great method for complex incomplete data that have more than one variable with missing values." (Rubin 1987, 1996) The MICE algorithm is able to provide satisfactory performance with just 5 or 10 iterations due to the fast convergence in the monte carlo simulation model. (Brand 1999; van Buuren et al. 2006b) The primary assumption here is that the missing data is missing at random, so the probability that a value is missing depends only on observed value and can be predicted using them, which fits the bill for our dataset. The function we used will impute data on a variable by variable basis by specifying an imputation model by variable. (S. Buuren and K. Groothuis-Oudshoorn, 2011) Before imputing the LotFrontage variable and the remaining variables with a few missing variables, the train and test data sets were combined to provide the MICE algorithm with more observations to learn from. We were able to combine these two data sets because they were from the same time period and we are not predicting the value of house sales in the future.

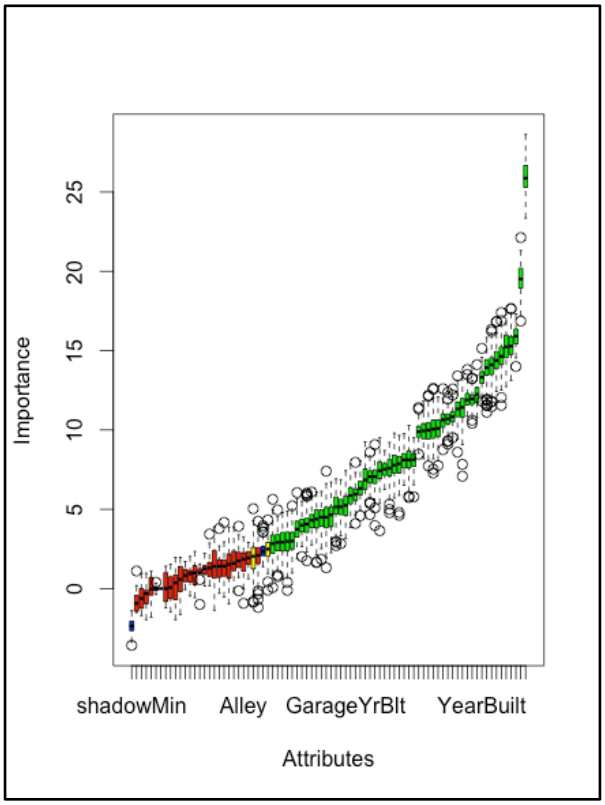
After reviewing the raw data set and its corresponding attributes, the month and year the house was sold was converted into factors to capture the seasonal trend. In addition, a seasonal categorical variable was created to help the model understand the seasons in a year and so the housing sold in the same season are analyzed the same way. All the data attributes, YearBuilt, GarageYrBuilt and YearRemod were converted into age variables by subtracting the YrSold from the corresponding variable. For example, to figure the age of the house when it was sold, YrSold minus YearBuilt, provided the house's age in years. There are 16 categorical quality variables in the data set. These variables were taken from a known finite set of possible values called levels. For our purpose, these variables were transformed into numeric variable values. For example, the ExterQuality variable has five levels PO (poor), FA (Fair), TA (Average/Typical), GD (Good) and Ex (Excellent) instead of keeping this variable as each factor was assigned a numeric value of one through five, where one represents the worst and five is the best. There are a number of advantages to converting categorical variables factor variables into numeric. The transformed variable can be used in statistical modeling where it will be implemented correctly by assigning the correct number of degrees of freedom. (Bruin, J. 2006)

Since there are 25 distinct neighborhoods in the Ames area, the Neighborhood variable was grouped into bins to model similar neighborhoods together. A cluster approach was used to analysis and assign each neighborhood to a designated bucket based on the neighborhood's median sales price tier throughout the four years. Normally, clustering is an unsupervised learning technique that finds natural grouping. For instance given labeled data in this report, a supervised clustering approach was used. Each neighborhood was partitioned into a set of meaningful sub-classes based on their median sales price. This approach reduced the neighborhood bins from 25 distinct levels to 10 distinct levels of similar trending neighborhoods. Refer to appendix one to see the neighborhood bin groupings.

Outliers are extreme values (either small or large) that can largely influence statistical analysis in our model; the data can be either measured wrong or recorded wrong. We have identified several outliers that need to be removed before implementing the model. One clear outlier are houses that exceede 4,000 square feet in GrLivArea. This criteria would remove 5 observations that are both partial sales as well as unusual sales which do not reflect the actual market value. Additionally, we also removed any house that exceeded the sales price of $700,000, have zero FullBath and LotArea greater than 60,000 square feet. We believe the error of inference will be significantly reduced as result, and that we can enhanced the accuracy of estimate in our model.

Models

Each observation in the data set has a MiscFeature variable and MiscValue variable. This variable is a catch-all and identifies any additional value the house has that is not captured in the other variables. For this reason, the complexity of the model can be reduced by directly subtracting the MiscValue from the SalesPrice. We can do this because in the provided test data set we know each house's MiscValue. Before running the different models, we had to figure out a way to reduce the number of variables the model would use. To do this, we used a Boruta feature selection process. Similar to the Ames housing data set, modern data sets are often described with far too many variables for practical model building. Usually, most of these variables are irrelevant to the model, and obviously their relevance is not known in advance. (Kursa and Rudnicki, 2010) This is the same situation with the Ames, Iowa housing data. There are 80 variables plus all of the additional feature variables that were created. To find the most meaningful variables in the data set, we used a Boruta feature selection where multiple RandomForest decision trees were created. The importance of an attribute was obtained as the loss of accuracy of classification caused by the random permutation of attribute values between objects, and each variable was converted into a Z-score by dividing the average loss by its standard deviation to determine the variable's importance measure. (Kursa and Rudnicki, 2010) The chart to the right shows each variable and their corresponding Z-score boxplot distributions. The chart clearly shows that there is a group of variables that are irrelevant and that there is a group of variables with more than a 10 importance level that stand out from the rest.



The Boruta feature selection found that 26 variables were irrelevant to predict the SalesPrice of a house and that there were 56 relevant features. Review appendix two to see which variables were accepted or rejected in the Boruta analysis. Multiple models were created to find the best models for the dataset. The best models were found using a 10-fold cross-validation and examining the correlation matrix of the predicted values from each model. The three we will talk about in this report are 1) multiple regression model 2) Gradient Boosting Machine 3) and the ensemble of the two models.

Multiple Regression Model

Linear regression is the oldest tool in the statistical tool book. This is because it is easy to understand and the coefficients can easily be interpreted. Using a stepwise greedy variable selection process with a 0.05 p-value acceptance threshold, the linear regression model decided to use 25 of the variables to predict each house's SalesPrice. The top five variables with the most influential power on the predicted SalesPrice were:

Variable	Coefficient	Std. Error	T-value	P-value
medPrice_Neighborhood	2.906e-01	3.296e-02	8.815	< 2e-16
OverallQual	6.137e-02	4.429e-03	13.857	< 2e-16
OverallCond	4.898e-02	3.351e-03	14.619	< 2e-16
KitchenQual_num	2.817e-02	7.642e-03	3.686	0.000236
HalfBath	2.815e-02	9.350e-03	3.011	0.002652

The assumption that location is the most important factor for a house's price holds true with this model. This is supported by the median SalesPrice per the neighborhood having the largest coefficient in the linear regression. The repetitive 10-fold statistic for the linear regression model is 0.13 RMSE and an R-square value of 0.89.

Gradient Boosting Machine (GBM)

A GBM model is a complex decision tree that additively learns from its past errors. The model was tuned using the caret library in R using an exhaustive parameter search to find the optimal values for interaction depth, number of trees, and shrinkage (the learning rate). The final values used in the model were 25 for interaction depth, number of trees was 4,000 and shrinkage was 0.01.

The top five variables the GBM felt were most informative in predicting the SalesPrice were:

Variable	**Relative Importance**

OverallQual	40.46
GrLivArea	12.93
GarageArea	5.01
TotalBsmtSF	4.75
medPrice_Neighborhood	4.67

The most important feature by far is the overallQual variable, which makes sense since the overallQual variable describes the quality of the property as a whole. This variable was also part of the linear regression top 5. The only other variable that was the same was the median SalesPrice per the neighborhood feature. The repetitive 10-fold statistic for the linear regression model is 0.12 RMSE and an R-square value of 0.90, which is slightly better than the linear regression model.

Ensemble of Multiple Regression and Gradient Boosting Machine

Both the multiple regression model and the GBM model did pretty well on both the cross validation split and the kaggle hold out data set. For the final model, we decided to use an ensemble model because the correlation between the predicted multiple regression and GBM values had a correlation of 0.94. The advantage of combining two models together that have roughly the same accuracy but a low correlation is that you can improve the accuracy of your prediction by reducing the standard deviation. The ensemble model achieved a cross validation RMSE of 0.11, which is higher than both the multiple regression (0.13 RMSE) and GBM (0.12 RMSE) models.

Real Estate agents can use this model to identify the correct price each listing should be listed at. This will help them gain knowledge about what drives the SalesPrice of a listing to ultimately help their clients make a smarter decision pertaining to buying their home.

Learning

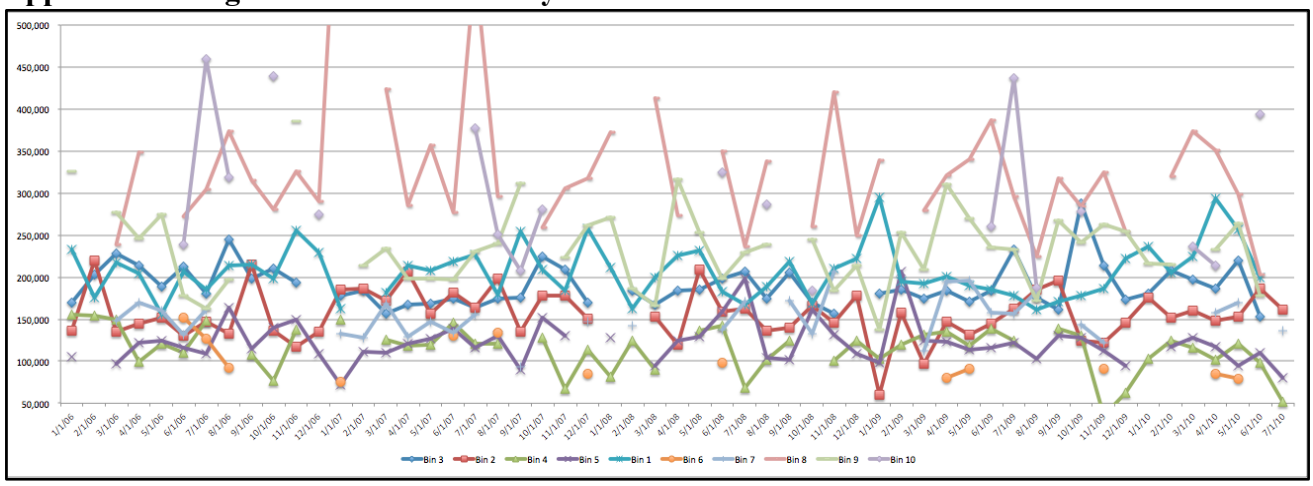
The model had a difficult time using the factor conditional variables, and when they were transformed into numerical equivalents, the models improved significantly. A prime example would be when we transformed various conditions and the quality of the houses' attributes into numeric variables. According to Leo Breiman, "One advantage of this approach is that it gets around the difficulty of what to do with categoricals that have many values. In the two-class problem, this can be avoided by using the device proposed in Breiman et al [1985] which reduces the search for the best categorical split to an O(I) computation. For more classes, the search for the best categorical split is an O(2I-1) computation." (Breiman, L. 2001) By converting these factor variables into numeric values, we allow the model to learn through an additive combination of variables from various different classes.

Future Work

With more time, we would dive into the variables available and create additional informative features. To further improve the model, we want to review the features we created and figure out

additional features that would be useful and are backed by industry research. When a feature is created with one or more of the variables, the old variables are removed from the data set. The feature creation will be an iterative process to eliminate the number of variables in the data set. In addition, traditional real estate analysis uses a comparable analysis model and adjusts house prices to the same time period. Since models do not use the year sold or month sold variables, we feel that adjusting all of the houses to the same time period, running the prediction model, and then re-adjusting the predicted SalesPrices back to their original time period should improve the models. We ran an initial analysis of this and it did not improve or make the model worse.

Appendix 1: Neighborhood Trend Analysis



Neighborhood Bin	Neighborhoods
1	CollgCr, ClearCr, Crawfor
2	Blueste, SWISU, Names, SawyerW
3	Gilbert, NWAmes, Blmngtn
4	IDOTRR, BrDale
5	OldTown, Edwards, BrkSide
6	MeadowV
7	NPkVill, Mitchel
8	NoRidge
9	Veenker, Somerst, Timber
10	StoneBr

Appendix 2: Boruta Feature Selection

Variable	Status	Variable	Status
MSZoning	Confirmed	RemodSaleYr	Confirmed
LotFrontage	Confirmed	no_GarageFlag	Confirmed
LotArea	Confirmed	Has2ndFloor	Confirmed
BldgType	Confirmed	HasMasVnrArea	Confirmed
HouseStyle	Confirmed	HasWoodDeckSF	Confirmed
OverallQual	Confirmed	HasOpenPorchSF	Confirmed
OverallCond	Confirmed	NewerDwelling	Confirmed
YearBuilt	Confirmed	LotShape_num	Confirmed
YearRemodAdd	Confirmed	GarageType_num	Confirmed
RoofStyle	Confirmed	PavedDrive_num	Confirmed
Exterior1st	Confirmed	ExterQual_num	Confirmed
Exterior2nd	Confirmed	BsmtQual_num	Confirmed
MasVnrType	Confirmed	BsmtCond_num	Confirmed
MasVnrArea	Confirmed	HeatingQC_num	Confirmed
Foundation	Confirmed	KitchenQual_num	Confirmed
TotalBsmtSF	Confirmed	FireplaceQu_num	Confirmed
CentralAir	Confirmed	BsmtExposure_num	Confirmed
X1stFlrSF	Confirmed	BsmtFinType1_num	Confirmed
X2ndFlrSF	Confirmed	BsmtFinType2_num	Confirmed
GrLivArea	Confirmed	Functional_num	Confirmed
FullBath	Confirmed	GarageFinish_num	Confirmed
HalfBath	Confirmed	Neighborhood_bin	Confirmed
BedroomAbvGr	Confirmed	total_porchSF	Confirmed
KitchenAbvGr	Confirmed	medPrice_Neighborhood	Confirmed
TotRmsAbvGrd	Confirmed	Log_price_new	Confirmed
Fireplaces	Confirmed	Neighborhood_bin	Confirmed
GarageYrBlt	Confirmed	total_porchSF	Confirmed
GarageArea	Confirmed	medPrice_Neighborhood	Confirmed

Variable	Status
Alley	Rejected
Utilities	Rejected
LotConfig	Rejected
Condition1	Rejected
Condition2	Rejected
RoofMatl	Rejected
Heating	Rejected
LowQualFinSF	Rejected
PoolArea	Rejected
MoSold	Rejected
YrSold	Rejected
SaleType	Rejected
new_house	Rejected
Peak_Season	Rejected
Sales_condition_priceDiff	Rejected
Pre_built	Tentative
HasEnclosedPorch	Tentative
HasScreenPorch	Rejected
HasX3SsnPorch	Rejected
LandContour_num	Rejected
LandSlope_num	Rejected
Electrical_num	Rejected
ExterCond_num	Rejected
PoolQC_num	Rejected
GarageCond_num	Rejected
Fence_num	Rejected

References

S. Buuren and K. Groothuis-Oudshoorn (2011), MICE: Multivariate Imputation by Chained Equations in R, Journal of Statistical Software

Bruin, J. (2006), newtest: command to compute new test, UCLA: Statistical Consulting Group. http://www.ats.ucla.edu/stat/stata/ado/analysis/

Rubin DB (1987), Multiple imputation for nonresponse in surveys, Wiley, New York

Rubin DB (1996), Multiple imputation after 18+ Years, Journal of the American Statistical Association, 91(434), 473–489

M. Kursa and W Rudnicki (2011), Feature Selection with the Boruta Package, Journal of Statistical Software, 36, 11

V. Buuren S and B. Groothuis-Oudshoorn CGM, Rubin DB (2006b). "Fully conditional specification in multivariate imputation." Journal of Statistical Computation and Simulation, 76(12), 1049–1064. URL http://www.stefvanbuuren.nl/publications/FCS%20in%20multivariate%20imputation%20-%20JSCS%202006.pdf