The Real Deal with Real Estate: Holistically Predicting Housing Prices

Hamzah Chaudhry HAZASLAM@LIVE.UNC.EDU
Joshua Kennedy JOSHUAWK@LIVE.UNC.EDU
Surya Poddutoori SUNISC@LIVE.UNC.EDU
Jonah Soberano SOBERANO@LIVE.UNC.EDU

Abstract

For future homeowners and home sellers, the buying and selling process can be stressful. Homeowners want to know how much their house typically would sell for and home buyers want to know what kind of house they can buy given their budget. Knowing this can bring vital financial perspective to sellers and a realistic goal for prospective buyers. In this paper, we used Linear, Ridge and LASSO Regressions, SVM, and PCA to determine what features of a house are a factor in the eventual sale price. We did this with a data set of houses and their features in Ames, Iowa, and these models can be used to predict housing prices of houses in markets similar to that of the Iowan college town.

1. Introduction

1.1. Motivation

Can you describe you dream house? You probably wouldn't begin with the slope of the property or the proximity to a railroad. Also how much would this house approximately cost? There is so much more to housing price negotiations than the square footage or number of bathrooms. In this paper, we examine what factors into the housing prices, and with 79 potential predicting variables, we are able to provide a comprehensive analysis of how much the home you described would cost. Conversely, how much would your current house sell for? Are there any features that you could add that would drastically increase your predicted sales price and by how much? Homeowners and home buyers face similar problems when entering the housing market and it is clear that there needs to be an easier solution for these questions.

1.2. The Data

Our main dataset consists of real world housing data from Ames, Iowa from 2006 to 2010. The data was found and cleaned by Dean de Cock of Truman State University, and is available on Kaggle as a competition. The data set has 2930 observations (representing individual houses), which has been split into equal sized training and test sets by Kaggle. Each observation consists of 79 predictors, an ID and a field for the sales price of the house.

1.3. The Plan

In Section 1, we outlined our motivations behind this data analysis and we described how this data set was compiled and the basic dimensionality of the data. In Section 2, we further analyzed the data and began the pre-processing process. Section 3 presents and discusses our findings in Linear, Ridge, and Lasso Regression models. In Section 4, we use SVR and and a neural network to attempt to improve our ability to predict sale price. Finally Section 5 is the Conclusion and summary of our results.

University of North Carolina-Chapel Hill, COMP 562 Spring 2019 Course Project.

Table 1. Percent of missing observations by category.

COLUMN	% TRAIN	% TEST
POOLQC	99.52%	99.73%
MISCFEATURE	96.30%	96.44%
ALLEY	93.77%	92.60%
FENCE	80.75%	80.07%
FIREPLACEQU	47.26%	50.00%
LOTFRONTAGE	17.74%	15.55%

2. Pre-Processing the Data

2.1. A Closer Look at the Data

After examining the data, it became apparent that some features were missing in over 99% of the observations in the training sets. We decided to drop all columns that were missing in at least 10% of the observations. These columns are listed in Table 1.

For all remaining categories with missing data, we followed the following procedure: we filled in numerical data with the median for that column, and filled in categorical data with the label ”Missing”. After performing this pre-processing, we proceeded to find the 10 features most correlated with SalePrice based on the training set. The results of this analysis are summarized in the heatmap of correlation coefficients on Figure 1.

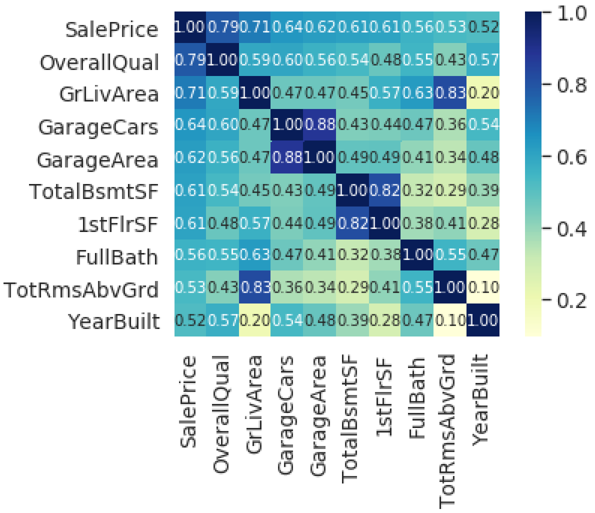

Figure 1. Heatmap of Variables Most Correlated with SalePrice

Additionally, we center (sustract feature mean the data points for each feature) and standardize (divide values by standard deviation of feature data) the numerical data using a scikit-learn package. This has the effect of giving each feature the same scale for use in a model.

2.2. Feature Engineering/Eliminating Redundancies

Looking at Figure 1, we notice that GarageCars and GarageArea are highly correlated with one another. Including both of these features in our model would be redundant, so we drop the GarageArea column, which has weaker correlation with SalePrice.

TotalBasementSF and 1stFlrSF are similarly correlated with one another. When looking at the descriptions of data fields in this set, we note that GrLivArea is the total square footage above ground for the real estate. Thus, the 1stFlrSF feature is a component of GrLivArea, and TotalBasementSF is a natural complement. Thus, we remove 1stFlrSF, and use the technique of feature engineering to add together GrLivArea and TotalBasementSF to create a new feature, TotalSF. Running a correlation analysis on this new feature shows a correlation of 0.78 with SalePrice, higher than that of GrLivArea or TotalBasementSF.

3. Regression

3.1. Linear Regression

We started with a relatively simple linear regression model. As discussed earlier, we standardized the numerical features, and we used a one-hot encoding representation of categorical features. We then ran a linear regression using scikit-learn using all of the features. This naive methodology has some issues we need to address, most notably overfitting. By utilizing features that have low correlation with SalePrice, this linear model is likely to be over-optimizing to the training set. We seek to develop a better model that reduces the risk of overfitting.

3.2. Ridge Regression

Ridge regression is known for being able to minimize the weight of a variable based on its usefulness. Because of this we decided to produce a ridge regression model at different alpha values to determine which values were useful in predicting SalePrice and which are not. We tried λ values 0.05, 0.10, 0.30, 1, 3, 5, 10, 15, 30 and determined that the best λ value is 5. Figure 2 shows how we chose the λ values; using the elbow method, we selected 5. The corresponding MSE is 0.1266.

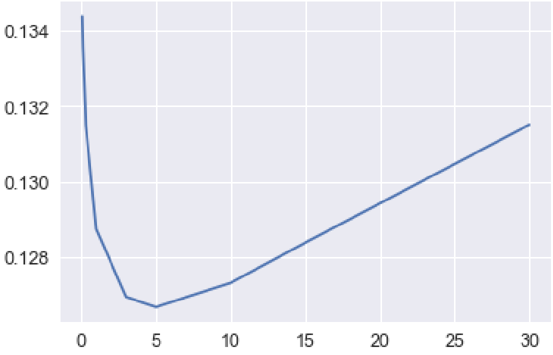

Figure 2. Line graph plotting the RMSE of the λ values in Ridge Regression

3.3. LASSO Regression

LASSO Regression is known for eliminating useless features in the data set by setting their weight equal to 0. Because we were unsure which regression would work better in accurately predicting sale price, we decided to do both and compare the accuracy. Again we tried multiple λ values and obtained an optimal MSE of 0.1236, a whole .003 better than ridge. As seen in Figure 3, total square footage and a clay roof are strong factors in the selling price of the house.

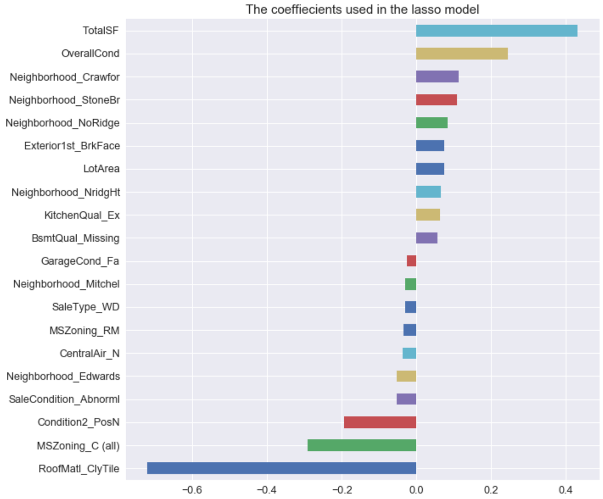

Figure 3. Bar graph showing coefficients used in LASSO model

Table 2. Regression accuracy for our models on the task of SalePrice prediction.

Model	Hyperparameters	Test Accuracy
Lin. Reg.	N/A	80%
Ridge Reg.	$\lambda = 5$	82%
LASSO Reg.	$\lambda = 0.005$	82%
RF Reg.	numtrees = 100	87%
SVR	N/A	87%
NeuralNet	Optimizer = Adam	80%

3.4. Random Forest Regression

Random Forest incorporates randomness to the decision tree model by generating many different decision trees. The final output is based on the majority output of each of the trees and can further be used to run either a classification or a regression model. We ran a Random Forest Regressor with 100 trees to get our results.

4. Modeling

4.1. SVR

SVR is a variant of the Support Vector Machine model which can be used for regression. This model fits our data because Sale Price is a continuous variable.

4.2. Neural Nets

There is a package for a Multi-layer Perceptron Regressor available in Scikit-learn. The perceptron network adds on to a linear regression model (a single layer network) by adding a hidden layer with 100 nodes and uses Rectified Linear Units (ReLU) as the activation function. It uses weights of multiple features to figure out the weights of the nodes in the next layer. The network also uses the ADAM stochastic gradient-based optimizer.

5. Summary

5.1. Best Models and Accuracy

Table 2 is a table of the models we used and their accuracy.

5.2. Conclusion

In conclusion, we used machine learning algorithms to answer the basic question of ”What contributes to the price of a house?” Our best predictive model were those of Random Forest Regression and SVR. From our LASSO regression, we are able to identify variables that impact the sale price of the house like total square footage and clay roofs. These factors tell us that if you are trying to sell your house for more money and you have a clay roof, it may be beneficial to re-roof your house in order for a higher sale price.

5.3. Future Work

We obtained our data through Kaggle, but we did not submit our models to the competition to test our respective models against the hidden portion of the test set; we only used the visible component of the test set. Doing this additional testing would likely be useful for obtaining more confidence in the respective accuracies of the models that we developed.

Furthermore, we could perform further analysis to attempt to improve the neural network model, which is surprisingly less accurate than simpler models. For the purposes of this project, we used the default hyperparameters in sklearn for the model. However, these hyperparamenters may not be optimal for this particular application; further research could attempt to tune these hyperparameters.

5.4. Github Code

For this project, we used Python, Jupyter Notebook, and Google Collab for our code, and we used Latex and Overleaf to write and compile this paper. The link to all of our code can be found at the following link: `https://github.com/hybrezz54/comp562-final-proj`

Acknowledgments

We thank Jorge Silva for teaching the course on Machine Learning for which this paper is for. We now understand the topics of Machine Learning in much more depth and we intend on continuing future research with machine learning in data analysis in regional data competitions, future research and our careers.