Group 6 Final Report

Chun Yin Chan, Aodhan Hayter, Lindsay Fielden

House Price Prediction in Ames Iowa - An Analysis

Introduction

The goal of this analysis was to develop a statistical model that will enable the reliable prediction of property sale price based on a set of attributes unique to a particular property. This analysis was performed using a historical dataset which contains a list of property sales in Ames, Iowa between the years 1879 and 2010. The dataset provides 79 attributes for each property sale.

Kaggle, the host of the dataset provides a training and test dataset. The training dataset is for use in developing a model and the test dataset is used to test the performance of the trained model. To gauge model performance we submitted our model's predictions to Kaggle. Kaggle then scored the model predictions, returning an "log(rmse)" score. The lower the score the better the model is at predicting a property sale price. From this dataset we developed a regression model with the following statistical performance.

Model Performance

Training Set

- RMSE: 0.12
- R Squared: 0.91

Test set

- RMSE: 0.16
- R Squared: 0.84

Kaggle Score

- log(rmse): 0.1396
- Rank: 2213

The following sections specify our methods of analysis and address further description of the dataset and any required manipulations to make analysis possible or more reliable.

Data

The dataset for this analysis consists of various property types that were built anywhere between 1879 and 2010 in Ames, Iowa. The data set consists of 79 property attributes that can be used to help predict the final sale price of the house. All 79 variables can be broken down into four variable group types: continuous, discrete, nominal categorical, and ordinal categorical. The majority of these variables were categorical due to the amount of information collected regarding environmental factors about the property such as: street, neighborhood, zoning, utilities available, and a few other details future homeowners may want access to when making their final purchase. In addition to environmental factors, standard property information such as: number of bedrooms, number of bathrooms, square footage, garage access, etc. were also included in the dataset to help construct the property sale price prediction analysis.

Data Modeling

A large proportion of this data needed to be prepared to ensure our model would utilize it properly. We started the data cleaning process by inspecting the dataset for missing or unavailable entries for each variable. If the missing data clearly represented another category within that variable we converted it to a consistent format, converting “NA” to a “None” for example. When this process was complete we took all of the categorical variables and properly factored them.

We then examined the dataset for entries that were missing entirely. All the missing values of variables related to garage, basement and masonry are changed to zeros. This might potentially hurt the accuracy of the original data but we believe the missing values are related to the missing of features in this case. We then imputed the remaining numeric variables using the KNN method and the categorical variables with a random forest algorithm to have complete data entries. We also opted to derive two new entries to augment the dataset, TotalSF, representing the total square footage of the property, and TotalBath, representing the total number of bathrooms on the property.

Commented [1]: Is this a safe assumption to make? Would it be safer to impute the median value here? I don't actually know the answer, just wondering.

Commented [2]: That's true, but that's what I did when cleaning the data

Commented [3]: Yeah, no point in changing now. I guess at best we could mention the shortcomings of the approach taken.

We then converted several variables, including the target variable SalePrice, to a log scale. By converting SalePrice, LotArea, 1srFlrSF, GrlivArea, and TotalSF to a log scale, we further normalized the distribution of values, reduced skewness of values, and allowed a more linear fit for our model (see figure 2).

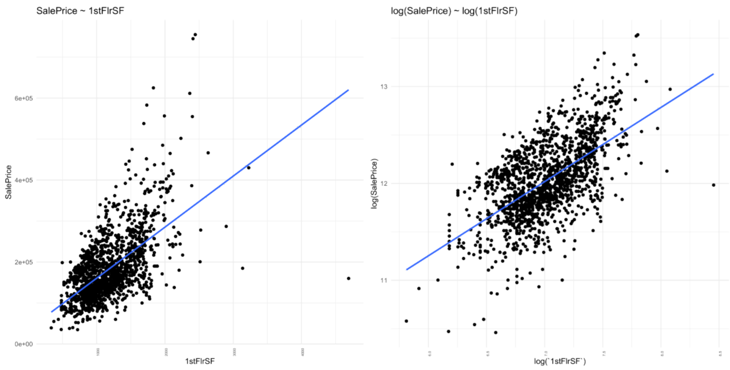


Figure 2: Relationship between SalePrice and 1stFlrSF before and after applying a log transformation to the data.

Methods and Analysis

We started our analysis by narrowing the variables of interest for inclusion in the model. We ran a linear regression model that used all of the available variables. We then inspected the summary output of this model to find variables that had a statistically significant effect on the target variable, SalePrice.

In addition to the previous method we also ran a simple correlation analysis on the dataset. We inspected the subset of variables only taking variables with a correlation of 30% and above to SalePrice (see figure 1).

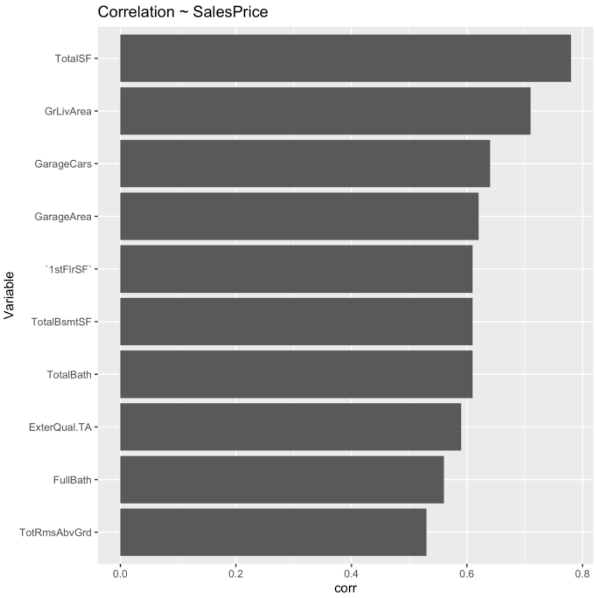


Figure 1: The top ten most correlated (negative/positive) variables to SalePrice

These two methods provided a good starting point from which to begin constructing a more focused linear regression mode. We then iteratively began this process again, running a model with this new set of variables derived from the full model output and correlation output. We then further inspected the model summary to determine which variables were statistically significant to predicting SalePrice, this allowed us to further remove variables that did not significantly contribute to SalePrice prediction. We then manually tuned the model with this more narrow set of variables until we were satisfied with the model's performance.

After carefully selecting variables to be used in the final model, we ended up using a 23 variables linear regression model to predict the sales prices of houses in the test set, building upon the 5 variables which we were already using in the interim report.

For numeric variables related to area, we used the log of LotArea, BsmtFinSF1, BsmtFinSF2, BsmtUnfSF, log of 1stFlrSF, 2ndFlrSF, log of GrLivArea, ScreenPorch, WoodDeckSF and log of TotalSF. The reason for us to pick these variables is because they represent different areas of a house, which includes the lot, the basement, first floor, second floor, deck, the porch and the total area. Including them all improved the accuracy of the prediction rather than just using the total area, and showed statistical significance when modeling.

For numeric variables related to place count, we picked Fireplaces, GarageCars and TotalBath. These variables represented some key features of a house including garage size and bathroom counts which are things we believe buyers would look into when buying a house. It is also notable that we used an interaction of GarageCars and GarageCond in the model.

For factor variables, we used MSZoning, Condition1, OverallQual, RoofMatl, Foundation, HeatingQC, KitchenQual, Functional, GarageCond, SaleCondition. These variables showed statistical significance when modeling and covered features including location, building materials, house quality and functionality. Variables related to year and month are not used in this model as we find them statistically insignificant when modeling house prices and can potentially lead to model overfit. The other features of houses are well covered by the variables we picked so the remaining variables are not chosen to avoid multicollinearity and overfitting.

Conclusion

Although we only utilized simple linear regression, our modeling approach was able to identify key attributes that gave our model relatively good predictive accuracy. This resulted in a respectable Kaggle score of 0.1396. We did not fully explore all of the possibilities of variable interactions and other attribute derivations, which possibly could have improved our model performance even further. Another approach we did not explore was the use of a regularized linear model such as Lasso and ridge, which are helpful with highly dimensional datasets.

Appendix

Figure 1: The top ten most correlated (negative/positive) variables to SalePrice

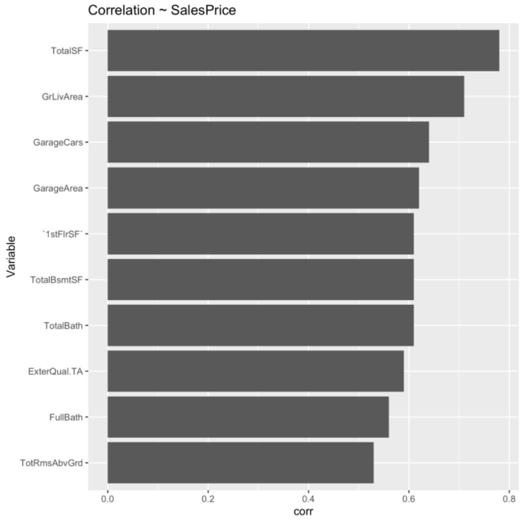
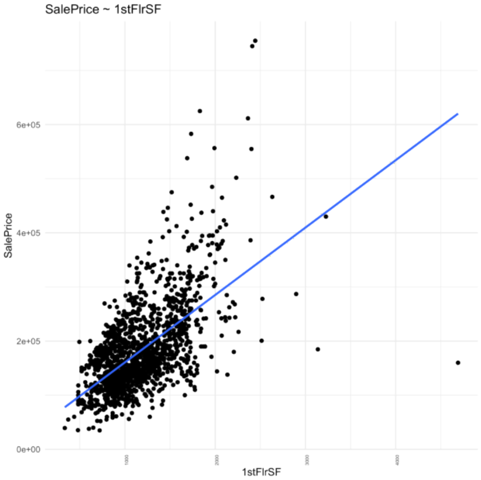
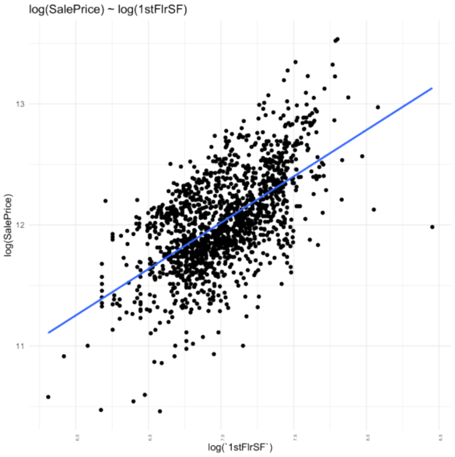


Figure 2: Relationship between SalePrice and 1stFlrSF before and after applying a log transformation to the data.

Before log transformation



After log transformation


log(SalePrice) ~ log(1stFlrSF)
log(SalePrice)
log(`1stFlrSF`)