A Peek Inside the House Pricing Blackbox

Zhongyuan Ma, Liyun Wang, Hanyu Li, Bo Zhang

2016.12.05

TABLE OF CONTENTS

A Peek Inside the House Pricing Blackbox

Zhongyuan Ma, Liyun Wang, Hanyu Li, Bo Zhang

Abstract:

The city of Ames, Iowa provides the dataset of 1460 house sale prices with 79 attributes involved in assessing the house values. This project employed big data techniques to establish several models to identify how these 79 factors potentially impact house sale prices. Quadratic and Huber loss models without regularizer and with l_1, l_2 regularizers were explored. The results show that quadratic loss with l_1 regularizer performs the best and predicts sale prices with RMSE lower than 25,081.11. At the end, model limitations and improvements are discussed.

1. Data Analysis

1.1 Background

The purpose of this project is to train a model to predict sale price based on the dataset published by the researcher Dean De Cock in 2011 which contains the sales of individual residential property in Ames, Iowa from 2006 to 2010. Being able to predict house price is important in two ways. First, a good house price estimate can be a starting point for prospect buyers to assess a house's value and evaluate the seller's offer. Second, a prediction model can forecast the change in house price over the next few years, which can be informative for the buyers to determine the best time to make the purchase.

1.2 Data Description

The dataset we use for this project was published by the researcher Dean De Cock in 2011 who obtained the data directly from the City Assessor's Office at Ames, Iowa. The dataset has 1460 entries in total, with 79 explanatory variables involved in assessing house values as well as the prices houses are sold at. Some of the variables that required special knowledge or previous calculations for their use from the original data were removed by the publisher. The overall quality of the dataset is great, with only a few entries missing data.

Among the 79 variables, there are 23 nominal variables, 23 ordinal variables and 34 variables including 14 discrete variables and 20 continuous variables. These variables focus on the quality and quantity of physical attributes of the property like overall quality/condition, exterior quality/condition, type/style of dwelling, lot area, foundation, heating, electrical, central air etc. We randomly selected 80% entries as training set and 20% entries as test set. The variables presented in this data are a mix of nominal, ordinal, discrete or continuous values. Since there are so many variables, the first step is to visualize the data and get a sense of how it looks like.

1.3 Data Visualization

1.3.1 Correlation Visualization

Data visualization is a very useful tool to better explore which variables are important in determining the house price. We first split the dataset into two parts, one for numeric variables and another for nominal variables. For the numeric part, firstly, a correlation plot is plotted to visualize which variables could influence the house price distinctively and get better sense of the correlations between each pair of numeric variables.

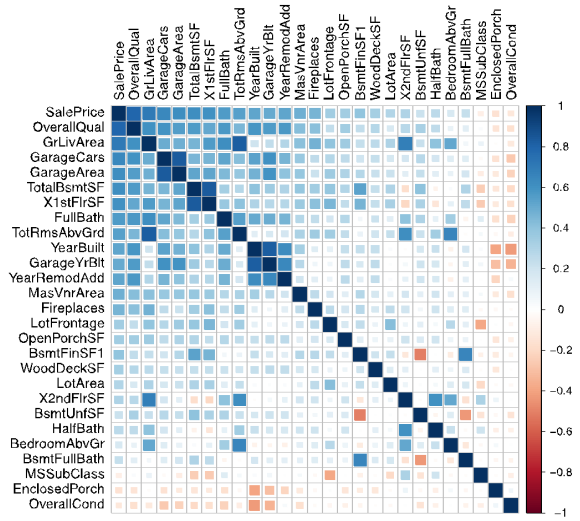

Figure 1.1 Correlation Visualization

As we can see from **Figure 1.1**, the dependent variable "SalePrice" (Sale Price) is highly related the feature "OveralQual" (Overall Quality), "GrLivArea", "GarageCars", "YearBuilt" and "YearRemodAdd". We first plot a boxplot to see how the overall overall quality influences the sale price of each house. By viewing the **Figure 1.2**, we can easily find that houses with higher ranking quality have higher sale prices. Then a boxplot about "GarageCar" and "SalePrice" is plotted showed by **Figure 1.3**. We can easily find that the higher the capacity of the garage, the higher the house price.

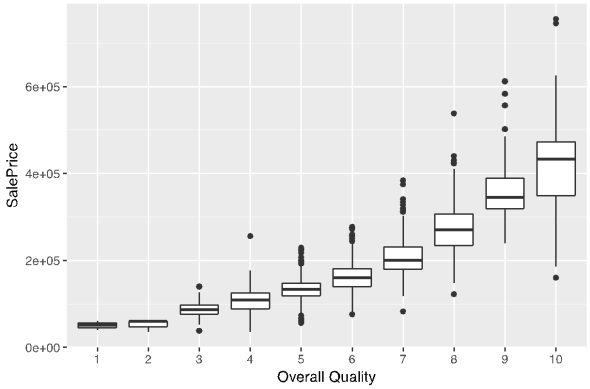

Figure 1.2 Overall Quality with Price

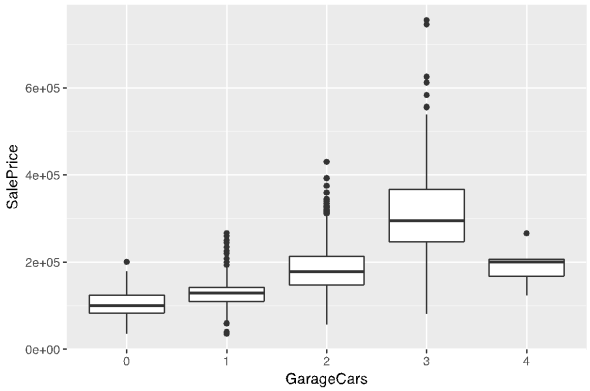

Figure 1.3 Number of Garage Cars with Price

1.3.2 House Quality Visualization

Intuitively, we consider two variables "YearBuilt" and "YearRemodAdd" are of great importance in determining the house quality. And as we can see from the correlation plot, there are highly related to the house quality. The plots of "OverallQuality" with "YearBuilt" and "YearRemodAdd" showed by **Figure 1.4** and **Figure 1.5**. We can easily find that the houses built and remodeled in most recent years are tend to have better qualities, thus with higher sale price.

1.3.2 House Price Visualization

We then plot the target variable "SalePrice", as shown in **Figure 1.6**, the original sale price distribution is skewed. In order to reduce the skewness of data and make it easier to work with, natural log transformation will be

applied on numerical house sale price at the part Data Cleaning.

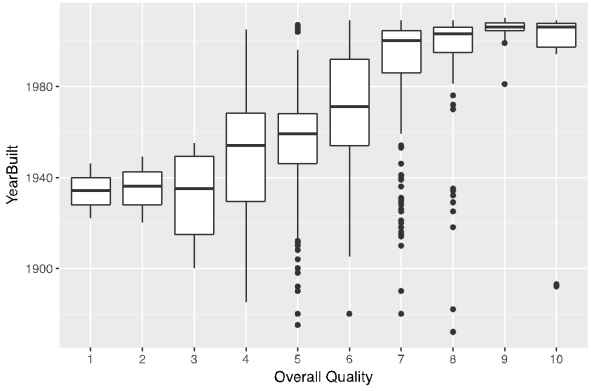

Figure 1.4 Overall Quality with YearBuilt

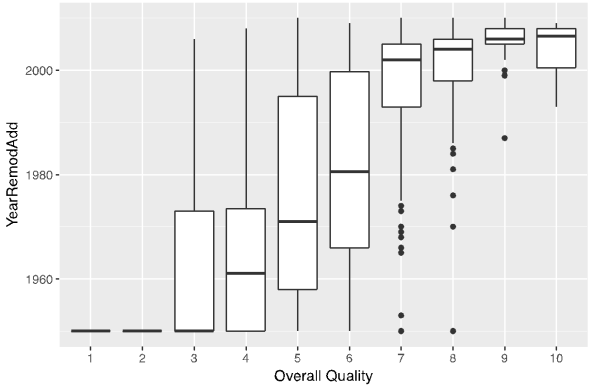

Figure 1.5 Overall Quality with Year Remodeled

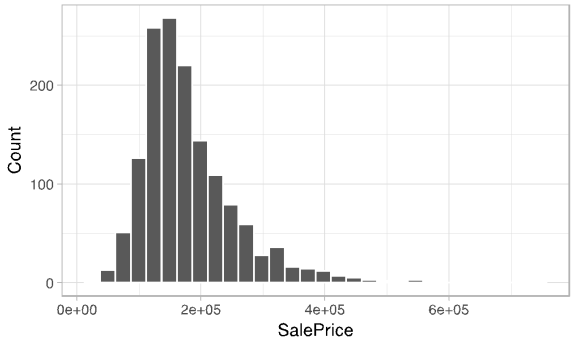

Figure 1.6 Sales Price Histogram

In order to further explore the relationship between sale price and each neighborhood, we plot a boxplot (**Figure 1.7**) to show the price range, median and outliers. Viewing the boxplot, we can find that "Northridge" and "Northridge Heights" are rich neighborhoods with several outliers in terms of price while the "BrookSide" and "South & West" of Iowa State University have cheap houses.

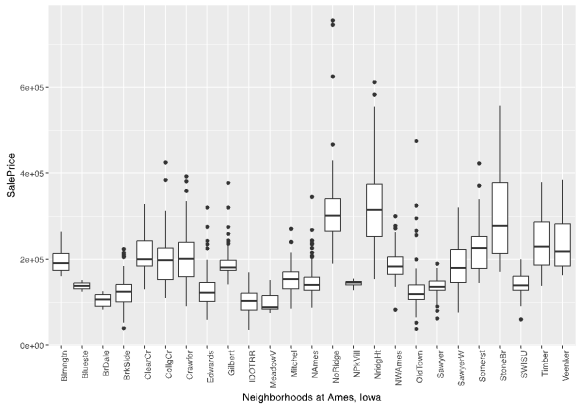

Figure 1.7 Boxplot for Sale Price and Neighborhoods

1.4 Data Cleaning

Several issues need to be addressed during data cleaning. First, there are a few missing values in the data set. The missing data may or may not have meanings, which will be handled in different ways. Second, there are many categorical data (ordinal and nominal) in the data set, which need to be transformed into calculable numerical values. Third, unbalanced data makes cross validation to be difficult, as some level of variable does not have enough data. Forth, the sale price distribution is highly skewed and the team will use log transformation to deal with it.

1.4.1 NA Values

The data set has excellent quality with only a few missing values. For most variables, "NA" values have real meanings. For example, the variable "Alley" indicates the type of access to property. It can have three values: "Grvl", "Paved", and "NA", where "NA" means no alley access. The team simply separate "Alley" variable into three independent variables: "AlleyGrvl", "AlleyNO", and "AlleyPave". The variables are assigned value of 1 whenever the corresponding condition is true. For those variables where "NA" simply indicates missing value, the team simply

delete the few data entries or used column average.

The data set has overall good quality with few NA values. Most of the NA values here do not represent missing values but with real meanings. The categorical variables with the largest number of NA values are: “Alley”, “FirePlaceQu”, “PoolQC”, “Fence”, and “MiscFeature”, however the “NA” values in these column have real meanings. For example, the variable “Alley” describes the type of access to property that contains three values: “Grvl”, “Paved”, and “NA”, where “NA” value indicates no Alley access to this house. In order to better fit our model, we separate “Alley” variable into three independent variables: “AlleyGrvl”, “AlleyNO”, and “AlleyPave”. The variables are assigned value of 1 whenever a condition is true and we will explain more details in part Data Clean.

The numeric variables do not have as many missing values but there are still some present. There are 259 missing values for the “LotFrontage”, 8 missing values for “MasVnrArea” and 81 missing values for GarageYrBlt. To better fit our model, we replace the numeric missing values (Na) with the mean of their respective columns.

1.4.2 Ordinal Values

The team convert ordinal values to categorical values. For example, the condition of the property can be good, fair or poor. This variable is separated into three variables: “Good”, “Fair”, and “Poor”. If the condition is good, the values would be “1, 1, 1”; if the condition is fair, the values would be “0, 1, 1”; if poor, the values would be “0, 0, 1”. Variables with ordinal numerical values are also handled in the same way. For example, the overall quality rating ranges from 10-1. This one variable is separated into three variables to avoid the pitfall of relative degree of difference.

1.4.3 Nominal Values

Variables with nominal values are transformed using one-hot coding technique. For example, the variable “Neighborhood” has 25 possible values. “Neighborhood” is separated into 25 variables, with only one variable being non-zero representing the neighborhood the property is located in.

1.4.4 Unbalanced Data

A few variables are extremely unbalanced, for example, the variable MSZoning identifies the general zoning classification of the sale. It can be agricultural, commercial, industrial, residential high density, residential medium density, residential low density, etc. However, over 90% of entries are residential. Commercial appear only on the test set but not on the training set, which is a big problem. To deal with the insufficient data, we combine low-occurrence levels, agricultural, industrial and commercial into one variable. Other variables with missing or insufficient data on levels are also handled in this manner.

1.4.5 Natural Log Transformation of Price

As shown in **Figure 1.6**, the original sale price distribution is skewed (skewness > 0.7). In order to reduce the skewness of data and make it easier to work with, natural log transformation is applied. As we can see from the **Figure 1.8,** log transformed data is more normally distributed, and the effects of outliers are also reduced.

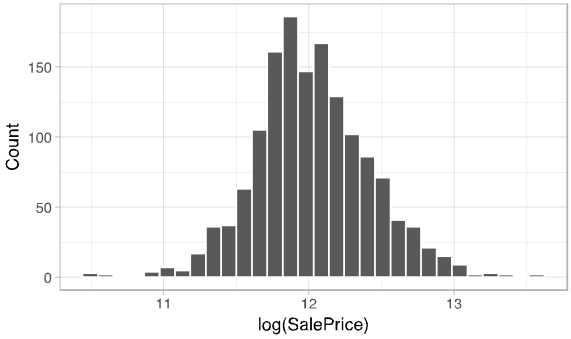

Figure 1.8 log(SalePrice)

2.Model Selection

2.1 Quadratic Loss

We first construct a linear model using only quadratic loss without any regularizations, and this model will be our basic model for comparisons with other potential models. The intuition in using a linear model with quadratic loss is simple. The explanatory variables in the data set are most likely to have a linear relationship with the housing price. The objective function of this linear model is defined as following:

$$minimize \sum_{i=1}^{n} (y_i - w^T x_i)^2$$

A. Quadratic Loss with l_1 Regularization

After fitting the linear model, we found out that the explanatory variables are not linearly independent, meaning x is not invertible, and thus there is not a unique solution to the linear model. This will be further discussed in "Model Results" section. Therefore, we will need regularization to guarantee that the model only produces a unique solution. l_1, as known as lasso regularization, is the first regularization we picked. The intuition is first to see if with regularization our model can perform better, and then hopefully that after eliminate some variables, it will be easier for us to understand the variables. Right now with more than 70 variables and more than 300 levels, it is very hard to explain variable importance.

The objective function of this linear model is defined as following:

$$minimize \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n} |w|$$

B. Quadratic Loss with l_2 Regularization

For the similar reason, we choose l_2 regularization, as known as ridge regularization to try to find a unique solution to this data set. In general, l_2 regularization does not produce a sparse solution like l_1 regularization, and it also will not eliminate certain unnecessary variables for us. However, ridge regularization might provide a better fitted model than lasso regularization.

The objective function of this linear model is defined as following:

$$minimize \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n} w^2$$

2.2 Huber Loss

We then try to fit a linear model on this data set using a different loss function. In a pool of housing price data, there are large outliers such as extravagant villas in wealthy regions. We don't want to skew our model too much in order to fit those outliers. Our intuition is that to punish less when the errors are too large, and punish the same as quadratic loss when the errors are within a reasonable range. Huber Loss is a combination of l_1 Loss and Quadratic Loss function. It punishes small error with Quadratic Loss (the Gaussian part) and large error with l_1 Loss (the robust part). Huber Loss function has one parameter k that controls where the linear part and quadratic part in the loss function intersects. By construction, Huber Loss does not penalize large outliers as much as Quadratic loss does so we thought Huber would produce a better solution for our data set.

The objective function of this linear model is defined as following:

$$minimize \frac{1}{n} \sum_{i=1}^{n} \mathbf{huber}\,(y_i - w^T x_i) + r(w)$$

$$\mathbf{huber}(z) = \begin{cases} \frac{1}{2} z^2 & |z| \le k \\ k(|z| - \frac{1}{2}k) & |z| > k \end{cases}$$

A. Huber Loss with l_1 Regularization

For the same intuition as that of 2.1, we will add L_1 regularization to the Huber loss. The objective function is defined as follows:

$$minimize\ \frac{1}{n}\sum_{i=1}^{n} \mathbf{huber}\ (y_i - w^T x_i) + \lambda\ \|w\|^2$$

In order to compare the result of one model with others, we need to find out the parameter value that provides the best solution for this model. To do this, we did cross validation where we computed the mean square error for each combination of λ and k.

B. Huber Loss with l_2 Regularization
We also apply l_2 regularization to Huber loss. The objective function is defined as follows:

$$minimize\ \frac{1}{n}\sum_{i=1}^{n} \mathbf{huber}\ (y_i - w^T x_i) + \lambda\ \|w\|$$

For the same reason as described in 3.1.2, we performed cross validation for this model to figure out the best parameter value for λ and k.

3. Model and Results

3.1 Quadratic Loss Model

3.1.1 Quadratic Loss Model 1

After data cleaning we have about 75 variables, which eventually are converted into 331 columns, and 1452 data entries. Initially, it seems that the number of data entries is large enough for us to fit a linear model.

Results: Due to the collinearity of the variables, there is not a unique solution for the quadratic loss linear model, and thus we are not able to obtain the result of such model.

3.1.2 Quadratic Loss Model 2

Even though we are not able to fit a unique quadratic loss linear model to the data set, we still need to build a base model for model comparison. Therefore, our solution is to manually select variables that seem important from the common knowledge of the real estate market, and delete other ones that do not seem to matter too much in predicting the housing price. By doing this, we eventually select 22 variables, totaling 97 columns. We conduct the similar data cleaning mechanisms as we do for the data set used in 4.0.1. We will use cross validation method to calculate the average error rate of this model.

Results: We are indeed able to fit a quadratic loss linear model on the original data set with only 22 variables selected. Using a 5-fold cross validation, we found an average RMSE of about 29,873.36, and the mean housing price from the data set is $180,921.2. R-squared is 0.863.

As the graph indicates, in general, the true sale price vs. predicted sale price is linearly correlated. However, the model shows that the divergence between predicted values and the true values is much larger when housing prices increase.

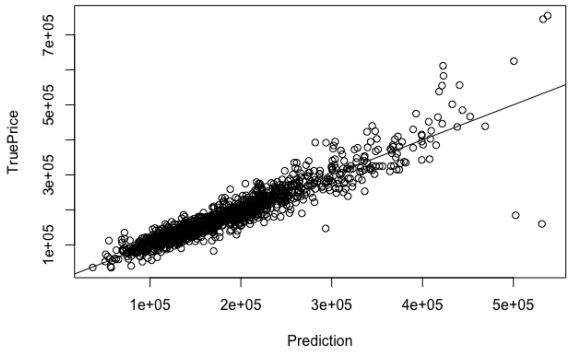

Figure 3.1 True Price vs. Predicted Price

3.1.3 Quadratic Loss + l_1 Model

In this model, we use the dataset with 75 variables, and hope that the lasso regularization can find a unique solution. We will also use cross validation to find the tuning parameter λ that gives a lowest MSE rate of this model.

Results: Quadratic loss and the lasso regularization model give us a unique solution, with a much lower error rate than

the quadratic loss model 2. Since the sale price is natural log transformed, the error rate is different from the actual MSE, but later this error rate will be expressed in the form of MSE again. The following graph shows this model's root mean squared error using 5-fold cross validation given different lambda values.

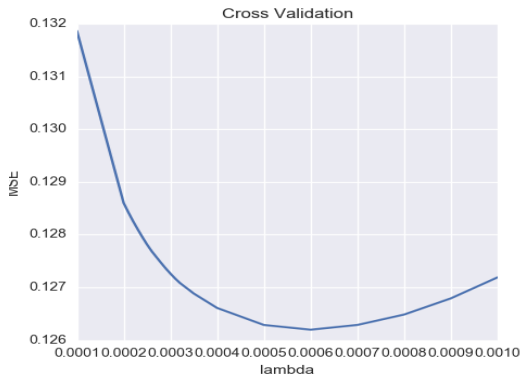

Figure 3.2 Cross Validation

As the graph indicates, the smallest lambda value of lasso regularization is about 0.00006. the corresponding RMSE is 0.1262, which is translated to be 25,081.11.

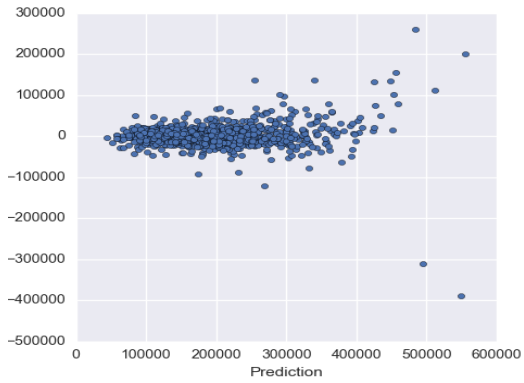

Figure 3.3 Residual vs. Predicted Price

As expected, the residual plot shows that this the explanatory variables are likely to be linearly correlated with the sale prices. However, even after natural log transformation of many variables, the model still has difficulty to capture the true value of high sale prices.

This model eventually eliminated 121 of the 331 variables. In the rest of the variable, general living area ("GrLivArea"), which is the total living area above ground, contributes most positively to the sale price. Those homes that are least functional ("Functional0") contribute most negatively to the sale price.

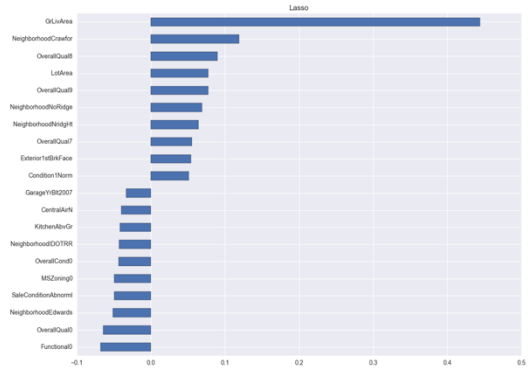

Figure 3.4 Coefficients Importance

3.1.4 Quadratic Loss + l_2 Model

We use the same data set that is used in 3.1.3 and fit a quadratic loss plus ridge regularization model to it. In this model, we also conduct a cross validation to find the best tuning parameter lambda that minimizes the overall error rate.

Results: We applied 5-fold cross validation on this model and found the best $\lambda = 15$ that gives us the smallest RMSE of 0.1289, which is only slightly higher than that of the previous lasso model. This RMSE is translated to be 25,357.62, which is also slightly higher than that of the lasso model.

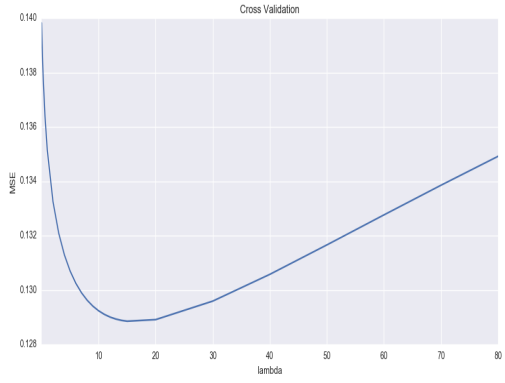

Figure 3.5 Cross Validation

The residual plot of ridge model also indicates the linear correlation between sale

price and explanatory variables. However, the outliers are still significant when price is large. Neither models seem to predict better in the high sale price range.

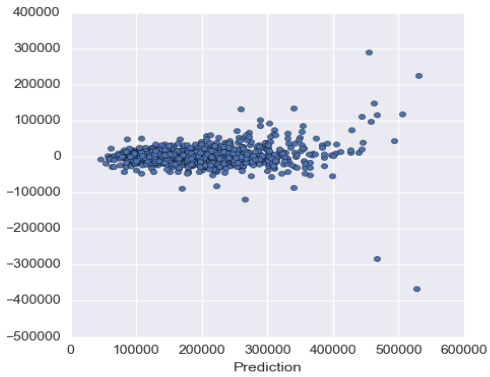

Figure 4.6 Residual vs. Predicted Price

Similar to the lasso model, general living area (GrLivArea) also contributes most positively to the housing sale price. However, the ridge model indicates that some of the neighborhood probably have the most negative impact to the sale price.

3.2 Huber Loss Model

3.2.1 Huber Loss l_1 Regularizer

This model uses Huber Loss function with l_1 regularizer. In the cross validation process, we computed the mean square error for each combination of λ and k with λ ranging from 0.0001 to 0.0009 and k from 1 to 9. We found that the model gives the smallest MSE when λ equals to 0.0001 and k equals to 3.5. It makes sense that we need a larger λ for our model because our dependent value y (natural log of sale prices) lies around 12 and we need to include more values into the Gaussian part of the equation.

Our computation shows this model can predict majority of housing prices with an error less than 30,000. RMSE of this model is 50,288.2. Given the limited data we have and simplicity of our model, we think the result is not bad.

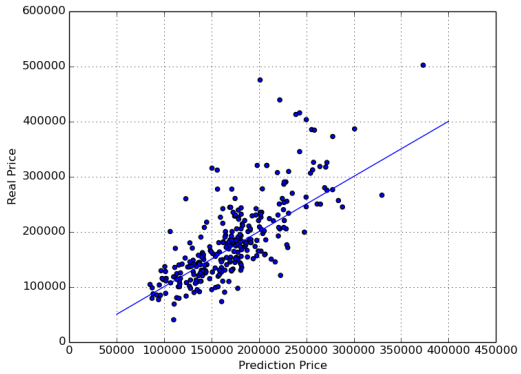

Figure 3.7 Prediction vs. Real Price

3.2.2 Huber Loss with l_2 Regularizer

This model uses Huber Loss function with l_2 regularizer. In the cross validation process, we find that best values for λ and k are 0.0001 and 9.

We have computed the RMSE of this model which is 48,348.9. Our computation shows that Huber Loss produces better prediction result with l_2 regularizer. l_2 regularizer penalized more on large w than l_1 loss, which gives a more stable solution.

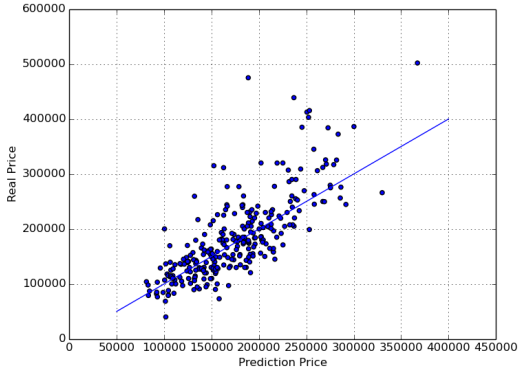

Figure 3.8 Prediction vs. Real Price

4. Future Improvement

4.1 Principal Component Analysis

All 79 variables are used to predict house prices in our model and are treated as being independent. However, there are some correlations amongst the 79 variables. **Figure**

1.1 shows that many variables may have positive or negative correlations. For example, "GarageYrBlt" has a high correlation with "YearBuilt". Therefore, one future work is to perform principal component analysis to convert the observed possibly correlated variables into a set of values of linearly uncorrelated components.

4.2 Overfitting

All the nominal and ordinal variables among the 79 features are transformed as categorical variables, bringing the total number of variables used to fit weights to around 200. With the 1168 data entries in the training set, we run into a considerable risk of overfitting. Using principle component analysis mentioned above to combine correlated variables is one way alleviate the problem. We also think about another way by combining neighborhoods that have limited data entries but with similar features and similar sale prices. For example, we can combine the two neighborhoods "Timber" and "Veenker" together for they have very limited data but share some similar features. Since the current data entries are partitioned by a total number of 25 neighborhoods, those neighborhoods short of data may impact the overall accuracy of the prediction model. The team thinks that this should be a valid approach, as these neighborhood pairs share similarities in geographical locations, in that they are about the same distance away from the downtown area of city Ames. Also, the pairs either both have no major outliers, or the pattern of their outliers are similar. The issue of overfitting will also be alleviated by obtaining more data; however, this was not possible within the scope of this project.

4.3 Smooth Regularizer

For variables such as neighborhood and type of dwelling, the team would like to use smooth regularizers to disallow dramatic changes of coefficients of adjacent features.

$$y_{N1} = W_1^{N1} X_2^{N1} + W_2^{N1} X_2^{N1}$$

$$y_{N2} = W_1^{N2} X_2^{N2} + W_2^{N2} X_2^{N2}$$

Coupling the coefficients of adjacent features to allow model to change over space or time. For example, the team would like to couple neighborhoods that are adjacent to each other and smooth the coefficients of their features. In this project, smooth regularization for neighborhoods was not possible because neighborhood levels in the data set do not correspond the actual neighborhood names in Ames, Iowa. Thus, there is not a good way to figure out actual location of the neighborhoods, nor their adjacency, leaving this to be a future work.

4.4 Neighborhood Model

For different neighborhoods, the most important features that affect house prices might be different. This is intuitive considering that Ames is a college town, location may weigh more for neighborhoods next to Iowa State University and downtown. Thus, ideally, the weights placed on features should vary from neighborhood to neighborhood. For future work, regression analysis on individual neighborhoods should be performed, given the premise that more data is available.

5. Conclusion

Accurate house price prediction can be used as a reference for prospect buyers to evaluate house value, and a time series prediction will assist them to determine the best time point to make the purchase. Among the few linear models explored in this project, quadratic loss with lasso regularizer appears to capture the price-determining factors well and offer the

best prediction. The predicted sale prices have RMSE lower than 25,081. Quadratic loss and lasso regularizer model also produces sparse solution, indicating which variables are the most significant in assessing house values. Interpretation of this model will simplify factors to consider when making a deal. However, for this model to be more informative and valuable for real estate market, further work needs to be performed. In this project, lack of data was a limitation. The team expects the model to perform better if the training set doubles its size. At the end, we would like to thank the City Assessor's Office at Ames, Iowa for gathering the data and Dr. Cock. for his effort to distribute the data set for higher-education use.