BUSINESS REPORT

Jason Carpenter, Nishan Madawanarachchi, Jose A. Rodilla, Kaya Tollas

DATA PRE-PROCESSING

In order to explain what factors are most relevant when explaining housing prices in Ames, Iowa, we built an OLS linear regression model using the *housing.txt* dataset.

Before jumping into modeling we needed to handle for several issues:
- NA handling
- Solve perfect collinearity issues
- Removal of influential points

NA HANDLING

In order to handle for NA values we previously carried out some Exploratory Data Analysis. We noticed that in the case of categorical variables, most NA values should not be interpreted as data being "not available". This is easily seen through the following example from the data description:

GarageType: Garage location

2Types	More than one type of garage
Attchd	Attached to home
Basment	Basement Garage
BuiltIn	Built-In (Garage part of house - typically has room above garage)
CarPort	Car Port
Detchd	Detached from home
NA	**No Garage**

For this variable --*GarageType*-- the NA value means that the house doesn't have a Garage. Therefore, this value has a clear meaning and should not be imputed or removed from the data. This was the case for many other categorical variables, where a value of "NA" would refer to the absence of a particular feature. For all these variables we decided to change the name of the category "NA" to "ABSENT". This way R would understand that data was in fact available for these cases.

For the remainder of NA values we used the mice library, which imputes missing values by taking into consideration similar data points.

PERFECT COLLINEARITY ISSUES

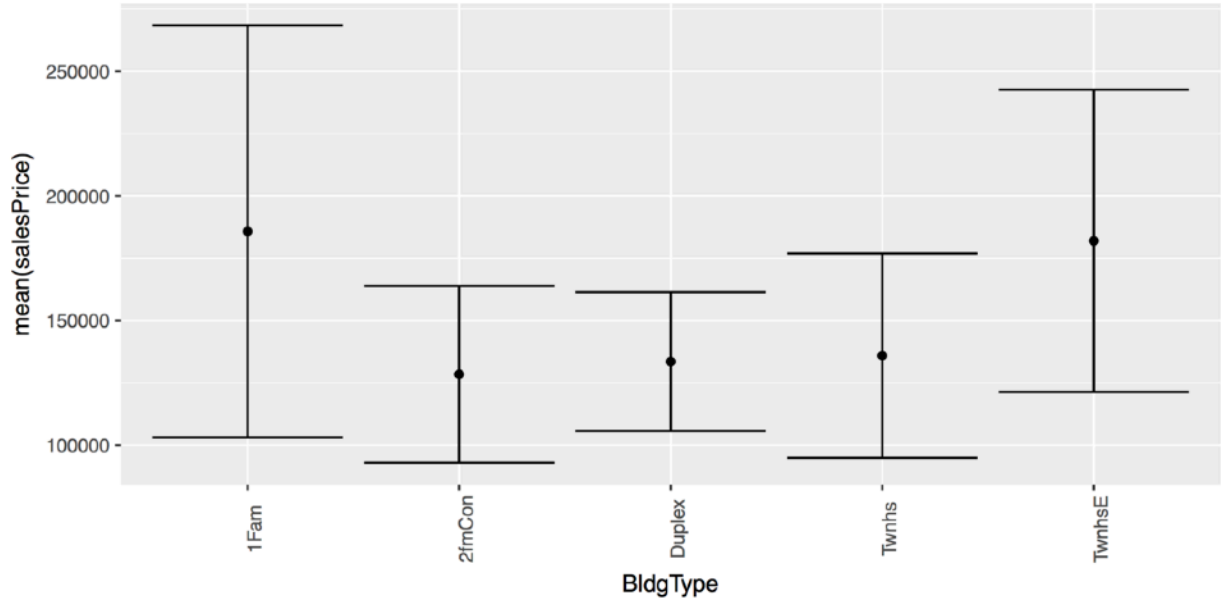
After running a first preliminary model --all variables included-- we realized that some of the coefficients for this fit returned a value of NA. This signals a presence of exact multicollinearity amongst variables. We explored each of these cases using our previous EDA analysis.

Categorical variables
A lot of these issues occurred only for specific categories within categorical variables. Especially, categorical variables that presented many categories, which would thus present very sparse columns of ones and zeros for certain categories. This sparseness was making it

possible for two or more categories belonging to different variables to present linear dependence. We decided to go through each of these cases; when reasonable, we grouped together several categories within a categorical variable. This resulted in more dense columns for these new categories, which in turn reduced the possibility of a linear dependence between categories.

The following example illustrates one of these cases:

Effects of BldgType categories on the response variable



For this case, the categories *2fmCon* and *Duplex* and have a similar effect on the response variable. Also, from a logical standpoint they are both referring to two-family houses. Thus we decided to group these into a new category *2fmCon_Duplex*.
Similar actions where taken for other categorical variables.

Numerical variables
We also eliminated numerical variables that were perfect linear combinations of other variables. More specifically, *TotalBsmtSF* —Total square feet of basement area— was a perfect linear combination of *BsmtFinSF1*, *BsmtFinSF2* and *BsmtUnfSF* and GrLivArea —GrLivArea: Above grade (ground) living area square feet— was a perfect linear combination of *X1stFlrSF*, *X2ndFlrSF* and *LowQualFinSF*.

REMOVAL OF INFLUENTIAL POINTS

The removal of influential points was carried out by calculating hat and DFFITS values in order to detect suspect points. Formal tests where then performed on these points.

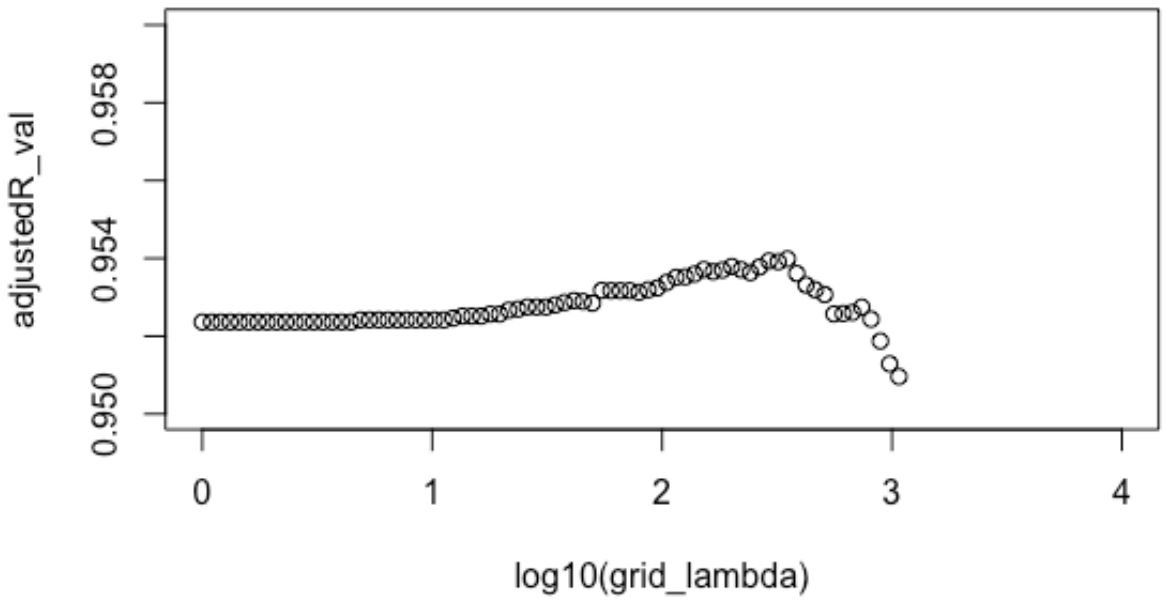
EXPLANATORY MODELING

TASK 1

In order to fit our explanatory model we used the transformed data from the previous steps. As a first step we partitioned the data into training and hold-out sets.

Variable selection
Using the training data only we ran a Lasso model in order to identify the variables that should be included in our OLS explanatory model. We used values of lambda that ranged from 0.01 to 10,000,000,000. For each lambda, we calculated the adjusted R-squared that resulted from running an OLS model which included those variables for which Lasso yielded a non-zero coefficient.

Adjusted R-Squared as a function of lambda

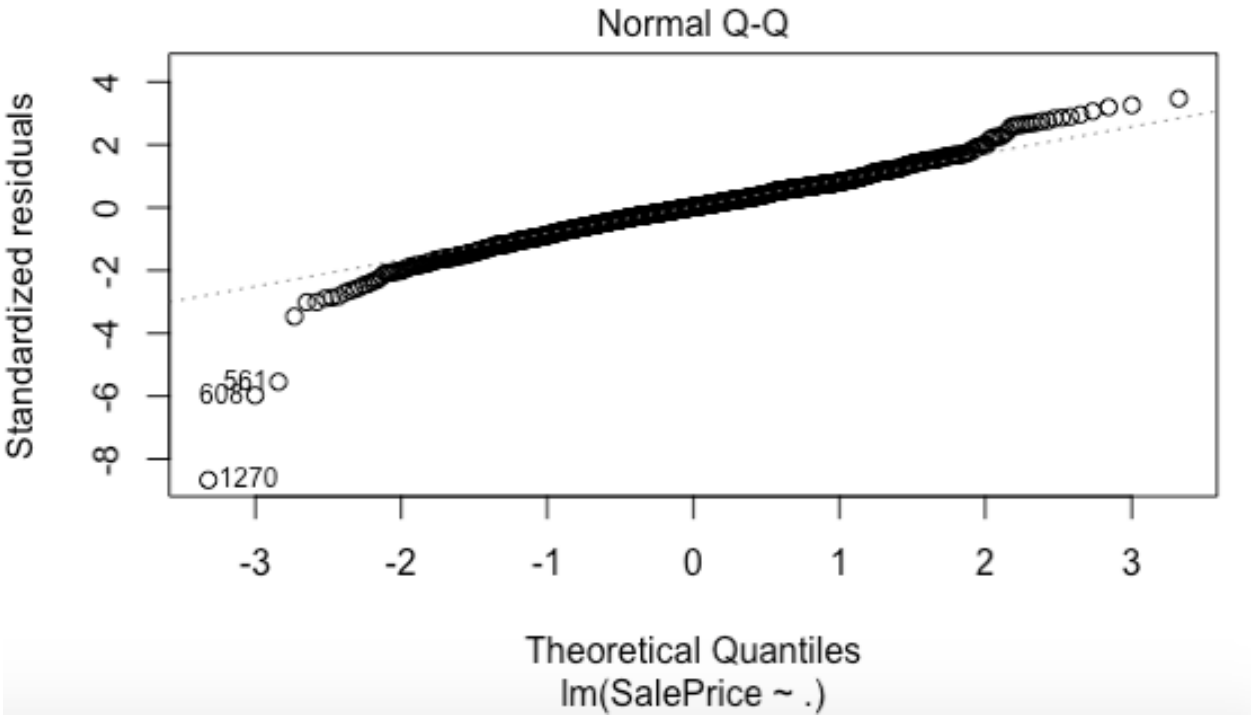


From these models, we selected the one that presented the highest adjusted R-squared (0.954) which was a associated to a lambda value of 351.

Multicollinearity
Our first VIF test strongly suggested collinearity. We thus decided to perform singular value decomposition on our design matrix, in order to build the π_{kj} statistics. As a rule of thumb, we decided to remove variables (k) that presented at least two π_{kj} values above 0.8. As a result, we ended up removing the variables *SaleTypeCOD* and *SaleConditionAdjLand* from our data.

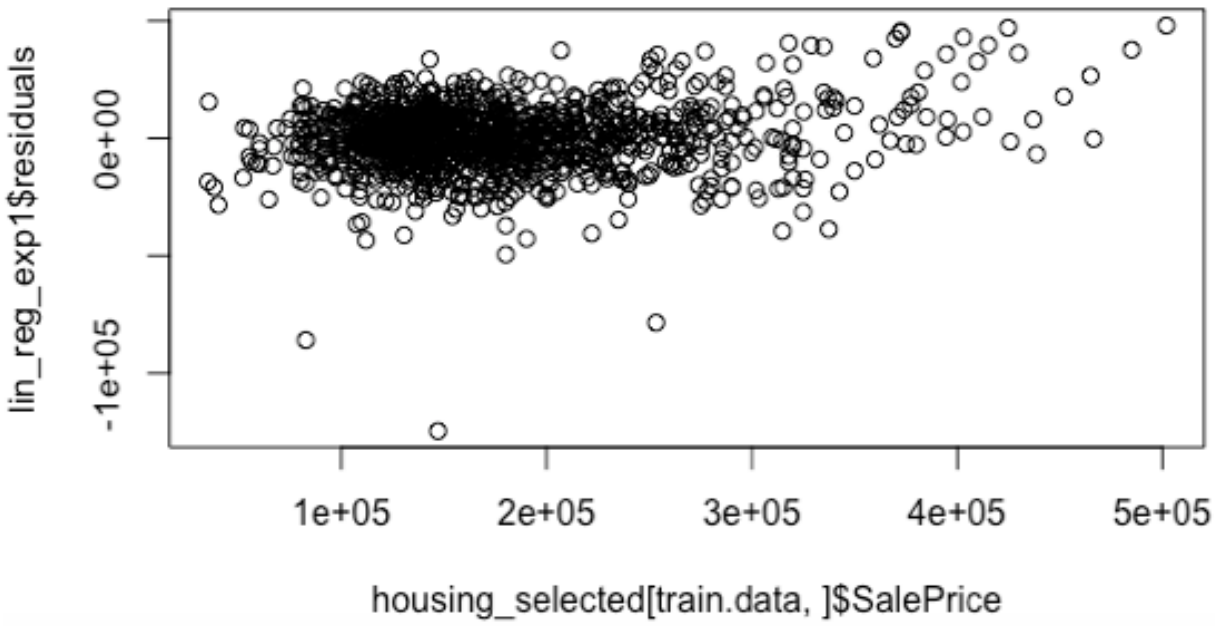
Preliminary check of assumptions
Using the resulting data from the previous test, we plotted a QQ plot of standard normals and standardized residuals.

The fat tails indicate that normality cannot be assumed:



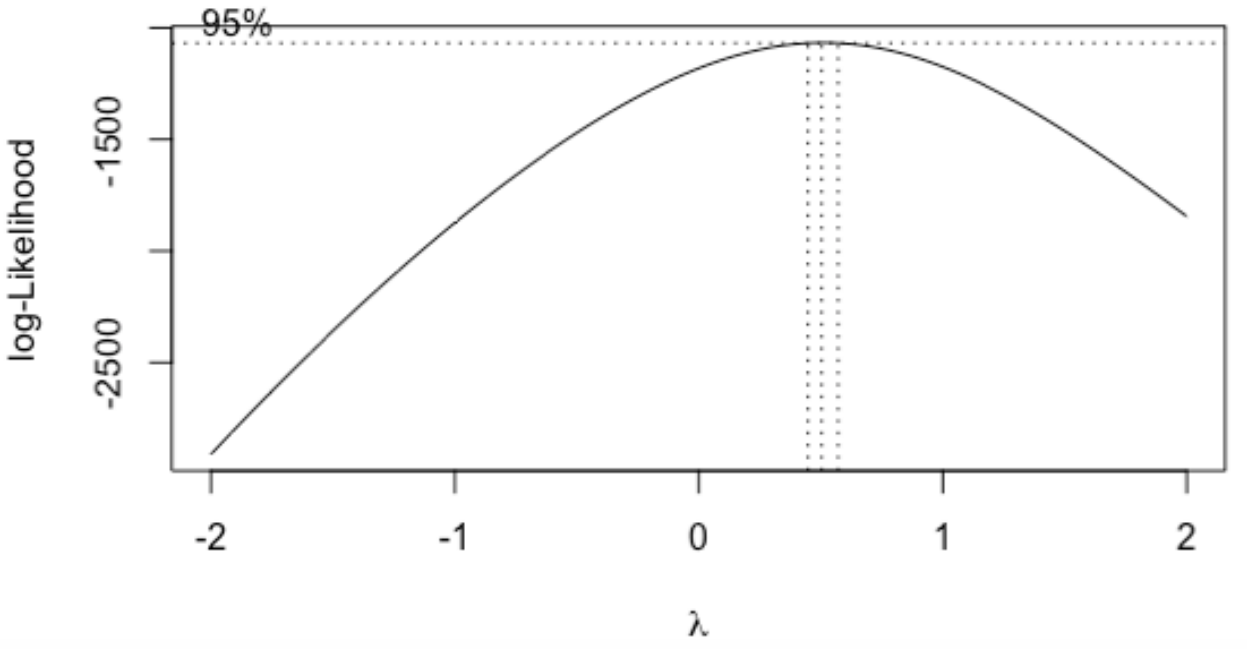
On the other hand the linearity plot gave the expected outcome:

Response values against residuals



Box-Cox transformation
In order to improve our normality issues we perform a Box-Cox transformation. The maximum likelihood estimator for our Box-Cox parameter lambda was 0.505.

Maximum likelihood estimation of lambda



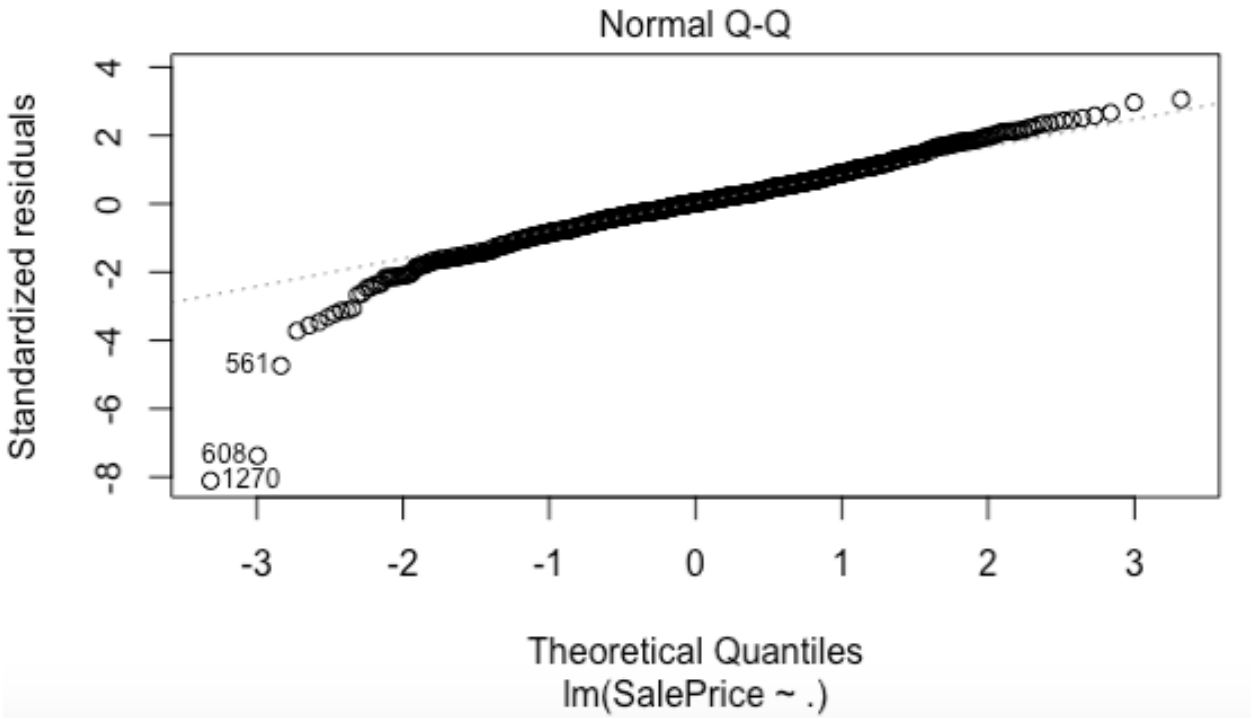
We transformed our response variable accordingly.
After performing this transformation, the resulting OLS model yielded the following QQ plot:

The Kolmogorov–Smirnov test confirmed that normality still didn't hold in our data.

One-sample Kolmogorov-Smirnov test

data: Std_residuals
D = 0.057815, p-value = 0.001143
alternative hypothesis: two-sided

Removal of influential points
In order to improve this issue we decided to perform a new check for influential points.
After removing these points the resulting model saw some improvement:

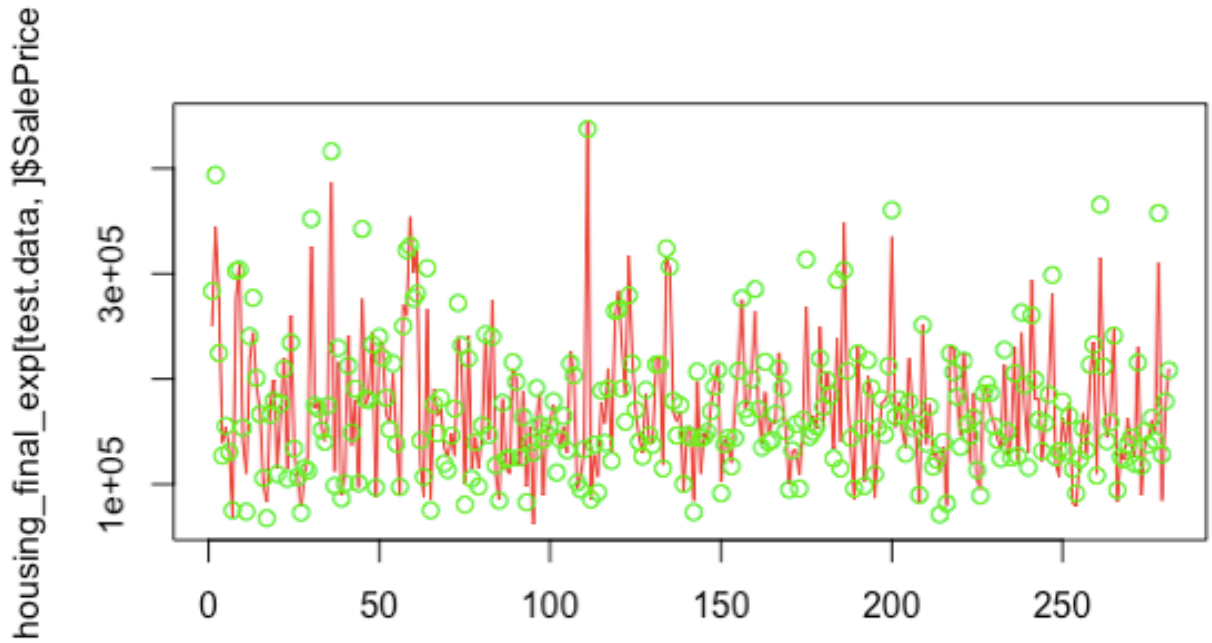


One-sample Kolmogorov-Smirnov test

data: Std_residuals
D = 0.056547, p-value = 0.001807
alternative hypothesis: two-sided

While the KS test was better than before, we still rejected the null hypothesis based on the p-value. However, we decided to go forward with this model.
At this point the adjusted R-Squared of our model was of 0.955.

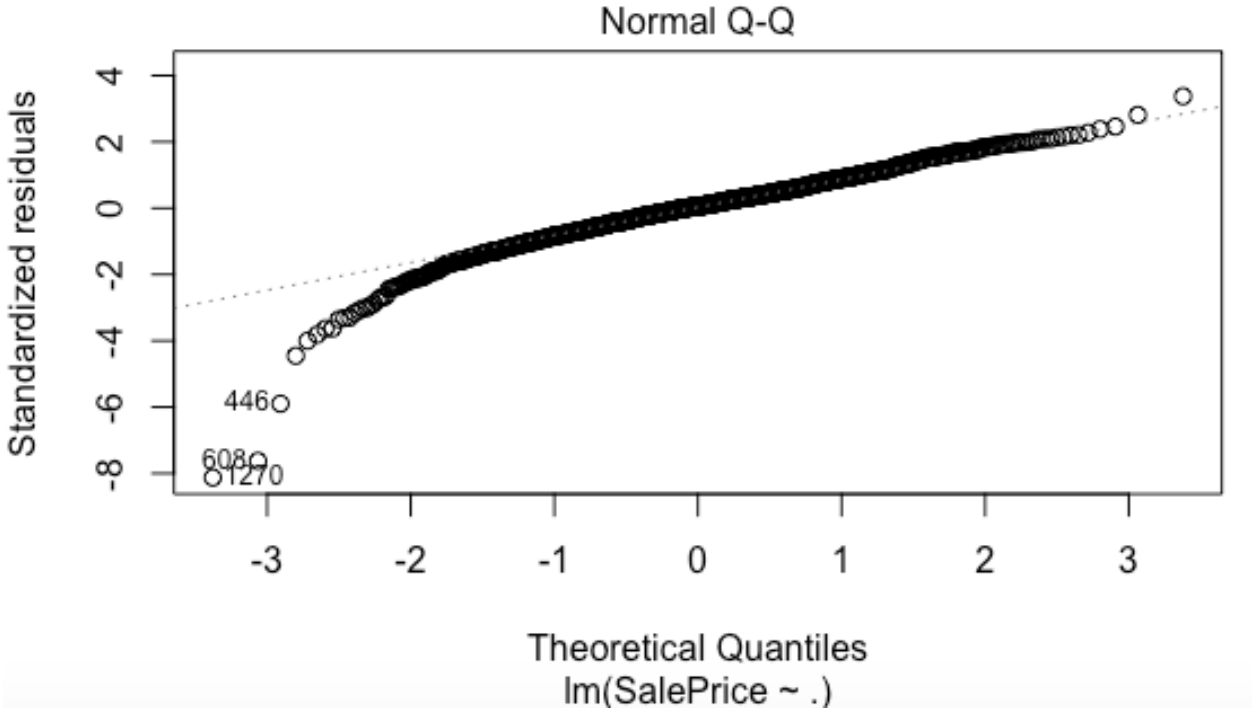
Prediction accuracy of the explanatory model
(Green = predicted values, Red = SalePrice in the original training dataset)



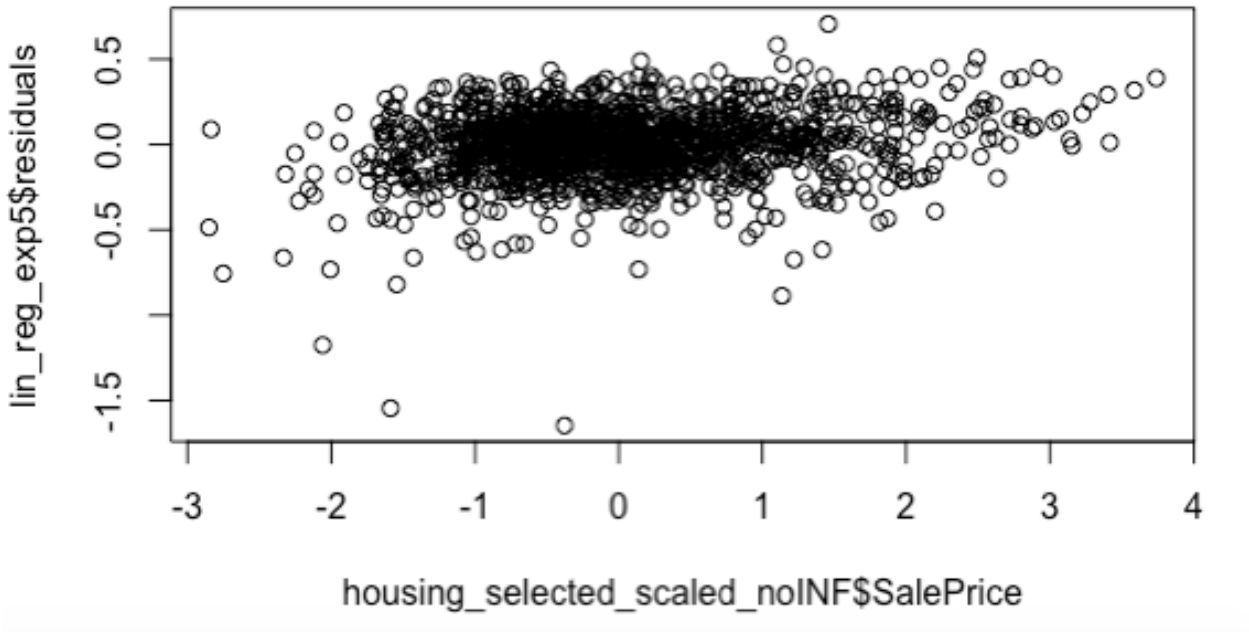
Final model and standardization

After making sure our model was able to perform appropriately both on explanation and prediction tasks we fit our final model using the entirety of the data (training plus hold-out sets). We also standardized the the explanatory variables in order to obtain comparable coefficients.

Final model results



Linearity



One-sample Kolmogorov-Smirnov test

data: Std_residuals
D = 0.053175, p-value = 0.0008587
alternative hypothesis: two-sided
———————————————————————
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 18.00591 Df = 1 p = 2.202201e-05
———————————————————————
Adjusted R-Squared: 0.9537

Summary of most relevant variables

Estimate	Std. Error	t value	Pr(>|t|)	variablenames
0.3555384	0.01910438	18.610305	3.848515e-68	X2ndFlrSF
0.2359422	0.01879874	12.550964	4.792476e-34	X1stFlrSF
0.2050371	0.02095609	9.784131	8.341327e-22	BsmtFinSF1
0.1889325	0.02222790	8.499793	5.544076e-17	YearBuilt
0.1766260	0.03141350	5.622613	2.336159e-08	LandSlopeGtl

TASK 2

In order to get an estimate for Morty's house selling price we first applied the pertinent transformations to his data in order for it to mimic our own. Once the transformation was in place we predicted the selling price using our final model. The predicted price and the 95% confidence interval where as follows:

fit	*lwr*	*upr*
139690	*110093*	*172766.7*

Based on this, our recommendation for maximum price would be 172,765usd.

Recommendations for Morty

According to our model the most important variables in determining the selling price of a house are the square feet of the second floor, first floor and basement. The fourth most relevant variable is the year the house was built. In light of these results, we recommend Morty to demolish his house and build a bigger one :)

If Morty is risk-averse and wants to go for a safer option we would recommend him to remodel his house and add some extra wings, as well as taking measures to reduce the slope of his property (5th most relevant factor).

Business Report

Jason Carpenter, Nishan Madawanarachchi, Jose Rodilla, Kaya Tollas

October 05, 2017

Predictive Modeling

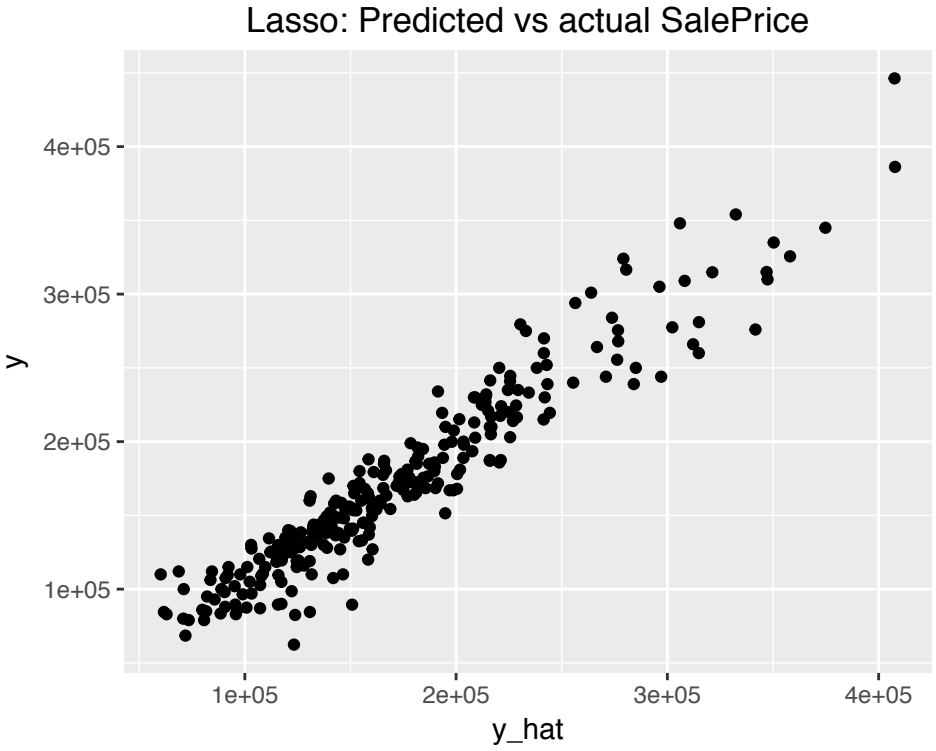
First, we transformed the training data to address NAs, solve problems of multicollinearity, and remove influential observations. Then we used the transformed data to train the predictive models using cross validation. Cross validation obtains the optimal parameters and mean squared prediction error (MSPE) for each model.

Our candidate models were Ordinary Least Squares (OLS), LASSO, Ridge, and Elastic Net. By cross validating our models over a grid of values for lambda, we selected the optimal lamdba that minimizes the mean cross-validated error.

Upon running cross validation, we found that the best predictive model (chosen by lowest MSPE) was: **lasso**

LASSO

Figure



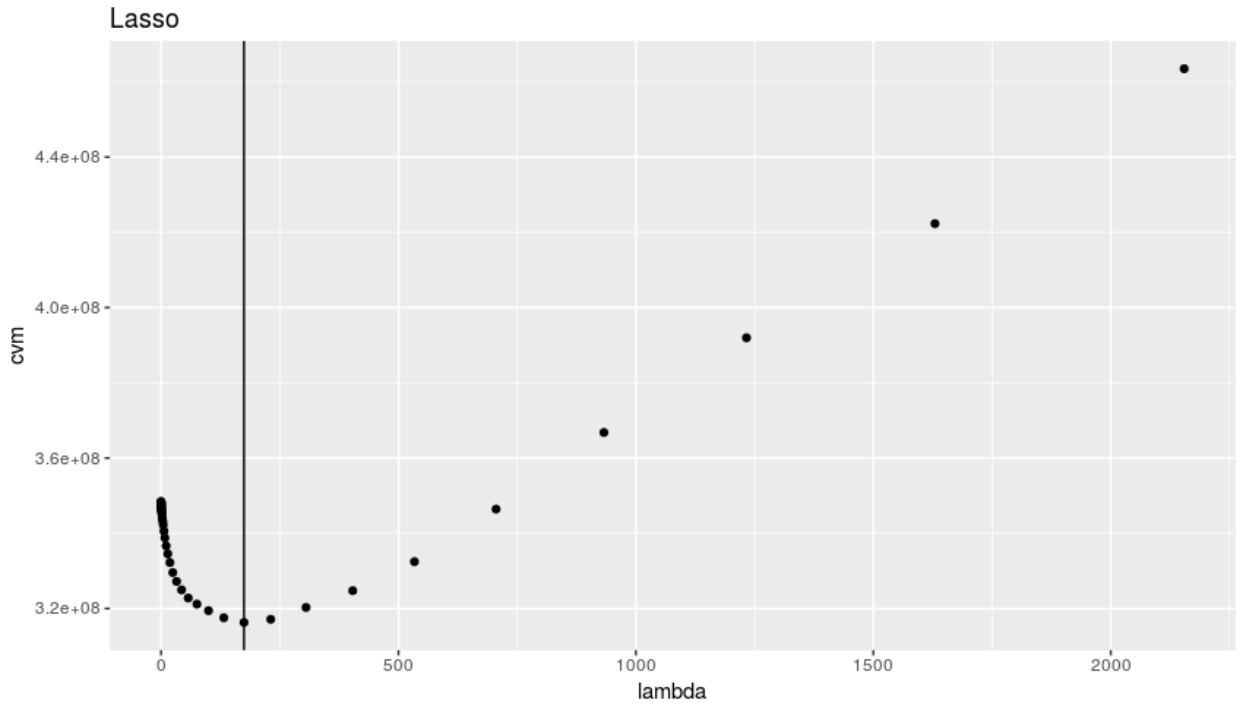


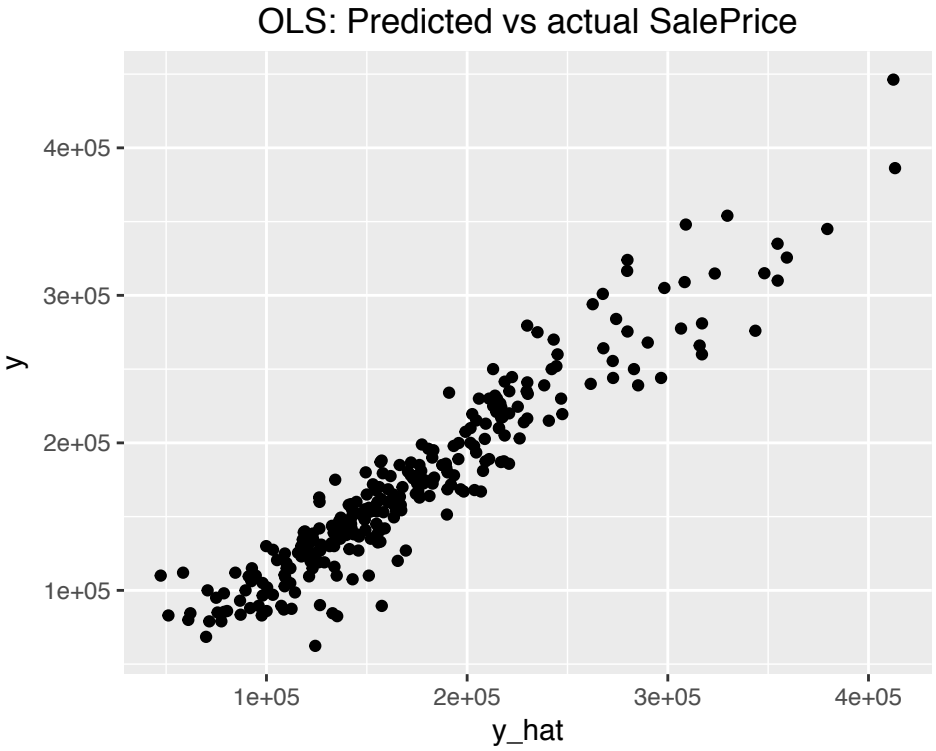
Figure 1: "Validation over lambda"

The MSPE for lasso was 3.9101493×10^8

The adjusted R squared for predicted versus actual values for lasso was **0.9030289**

OLS

For OLS, we first used LASSO to select the variables that we input to the model. We used the variables with nonzero coefficients returned from the optimal LASSO model, since LASSO sets less relevant variables to zero.



The MSPE for ordinary least squares was 4.4087412×10^8

The adjusted R squared for predicted versus actual values for ordinary least squares was **0.7511048**

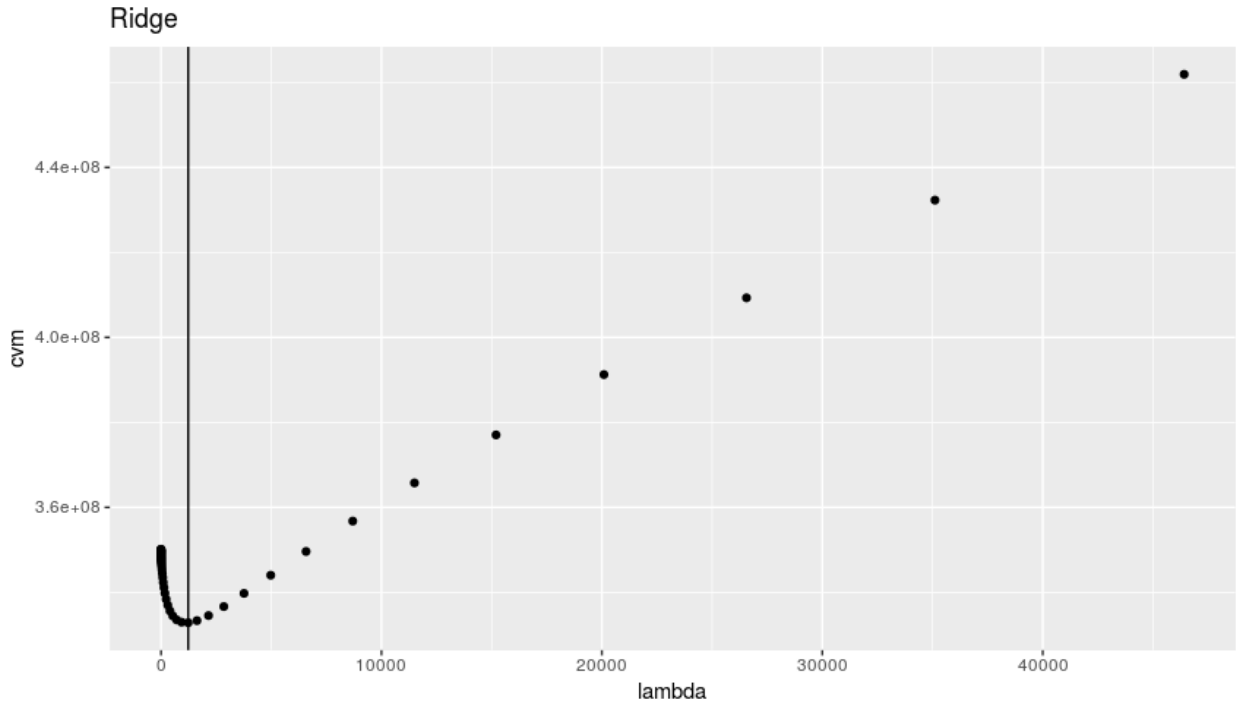
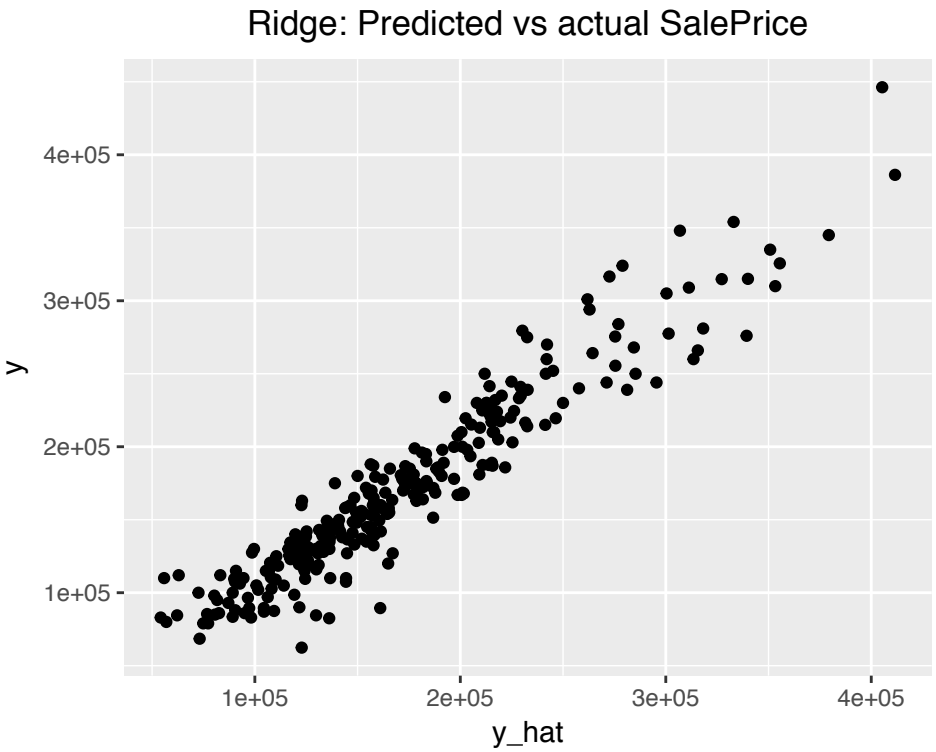


Figure 2: "Validation over lambda"

Ridge



The MSPE for ridge was 4.2531253×10^8

The adjusted R squared for predicted versus actual values for ridge was **0.8945231**

Elastic Net

The cross validation over lambda was done under alpha = .404 (calculated using cva.glmmnet, which cross-validates over lambda and alpha)

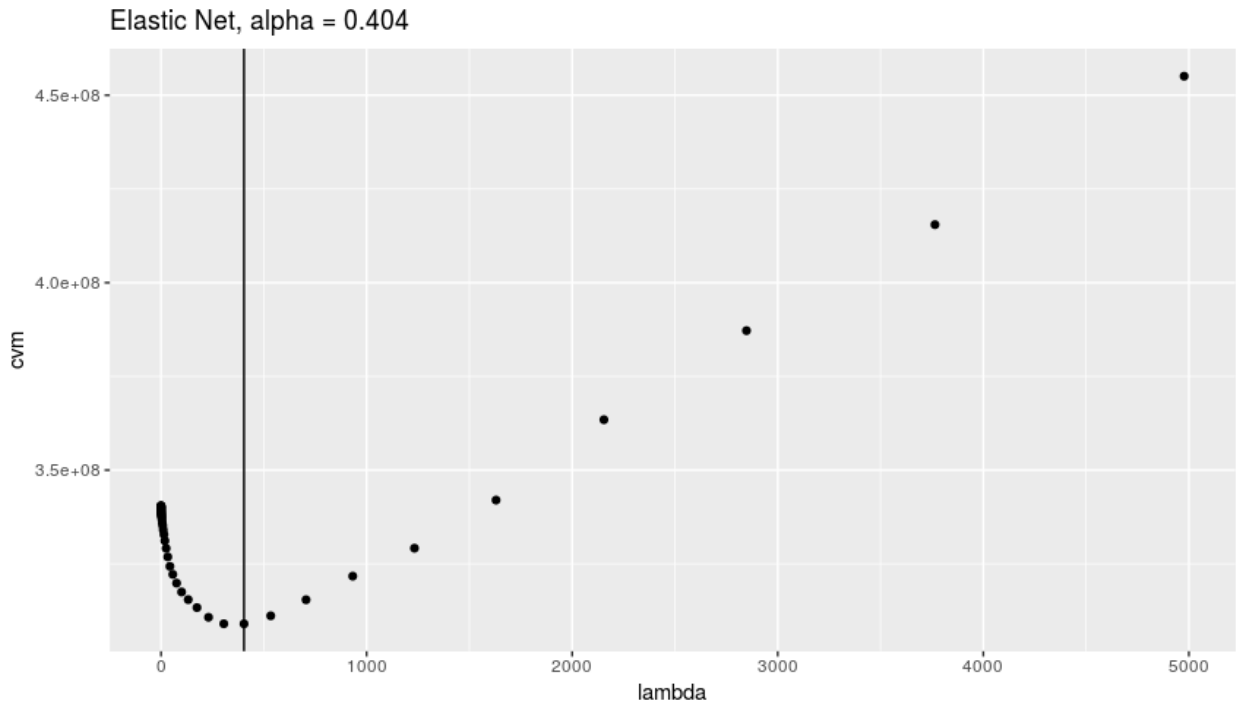
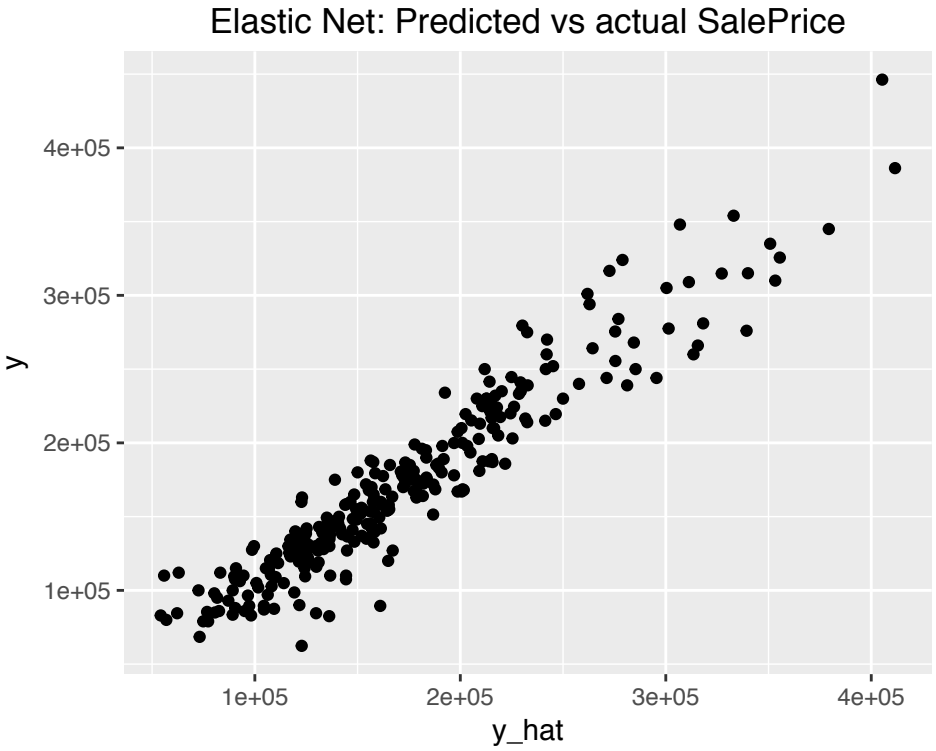


Figure 3: "Validation over lambda"



The MSPE for elastic net was 3.9389028×10^8

The adjusted R squared for predicted versus actual values for elastic net: **0.9333947**

Summary table of each model and its MSPE

model	MSPE
lasso	391014932
elastic net	393890280
ridge	425312528
ols	440874120