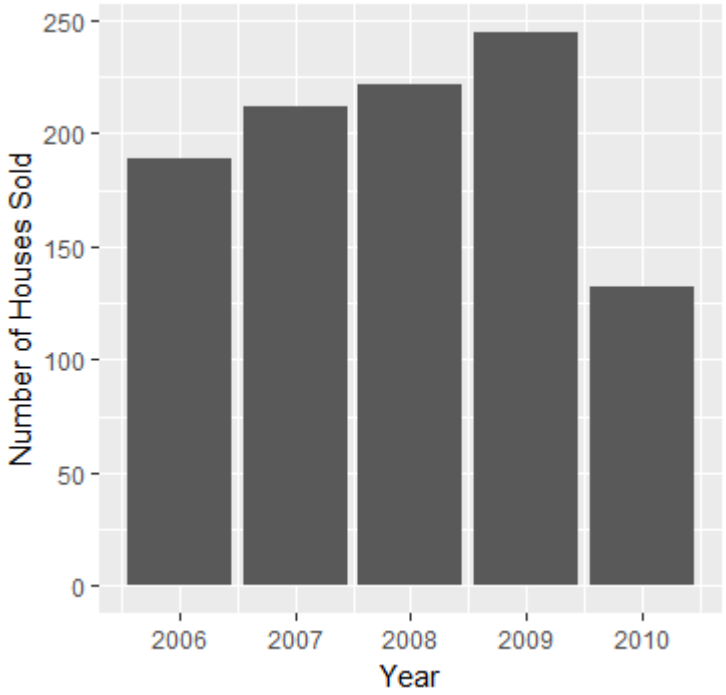
About the data

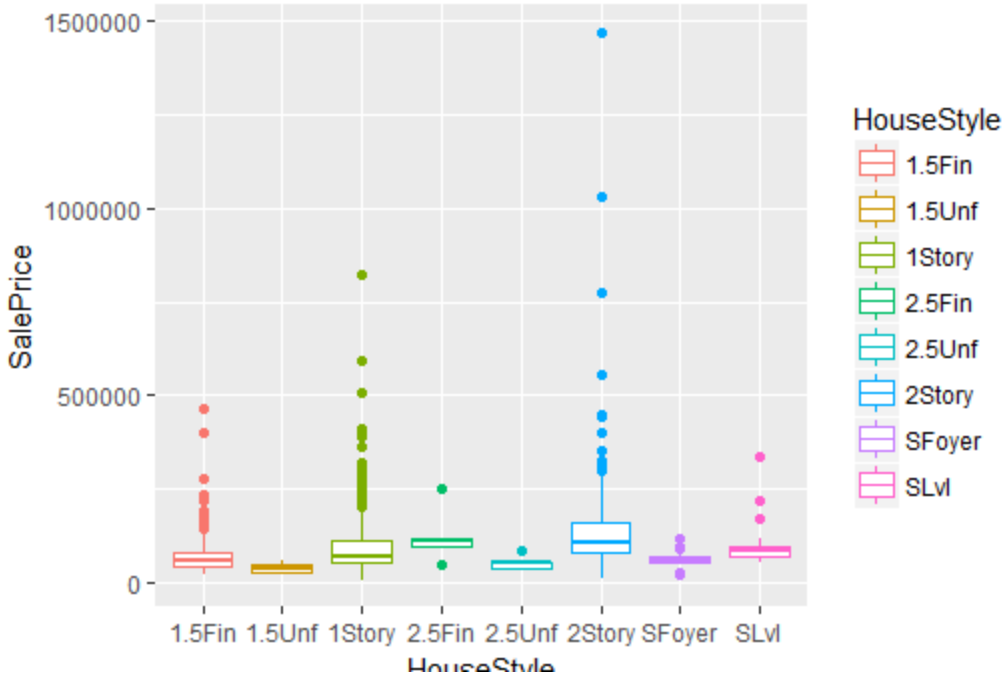
Data Visualization and Exploration

```
> dim(housing.data)
[1] 1000   70
```

Houses Sold Per Year

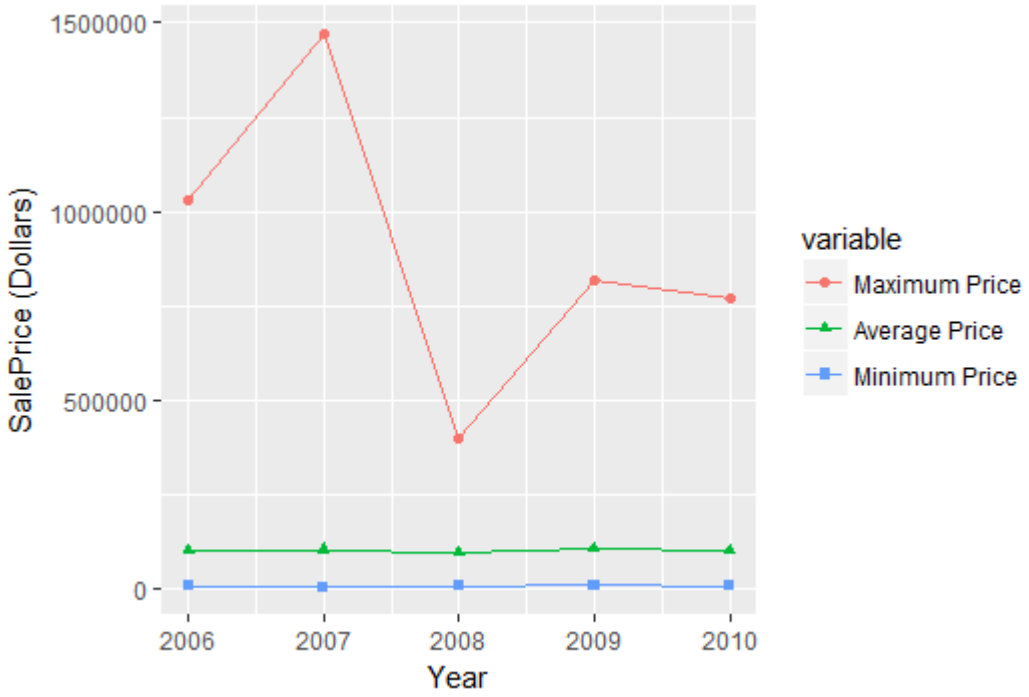


This plot indicates a steady increases in numbers of house sold per year from 2006 -2009 with a sharp decline in the year 2010 while 2009 was the year with highest amount of houses sold



This chart shows the 2 storey building commands the highest average price of all houses sold and while the split foyer and one and half storey building commands the least price all through the years

Price changes over the years.

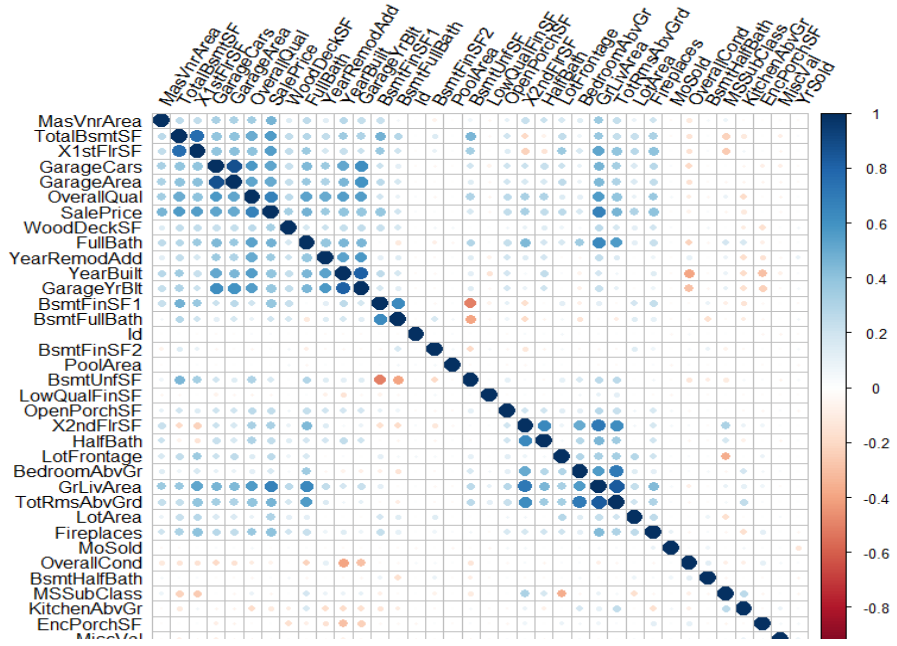


The plot shows the minimum and average price paid for a house over the years is pretty similar but 2007 is the year that the highest amount paid for a house was recorded.

Predictive Models

1. **Ordinary Least Square Model.**

   We tested the predictive strength of ordinary least regression on this complex housing data with presence of multicollinearity. We began by validating the presence of multicollinearity, testing for missing data and feature construction. The missing values indicate that majority of the houses do not have alley access, no pool, no fence and no elevator, 2nd garage, shed or tennis court that is covered by the MiscFeature. We carried out the following actions to prepare the data into form that will yield reasonable result in OLS regression without losing ability to interpret our model parameters.



- We deleted 5 predictors mentioned above with large percentage of missing data,
- Investigate multicollinearity between the predictors
- by using Boruta is an all relevant feature selection wrapper algorithm, capable of working with any classification method that output variable importance measure (VIM); by default, Boruta uses Random Forest. The method performs a top-down search for relevant features by comparing original attributes' importance with importance achievable at random, estimated using their permuted copies, and progressively eliminating irrelevant features to stabilize that test. Home values are influenced by many factors. Basically, there are two major aspects: The environmental information, including location, local economy, school district, air quality, etc. The characteristics information of the property, such as lot size, house size and age, the number of rooms, heating / AC systems, garage, and so on.
- 'LotArea', 'TotalBsmtSF', 'GrLivArea', 'GarageArea', 'BsmtUnfSF' all these variables are related to area type, so we can add all those values into single one.
- After using trying different approach and using stepwise predictor selection, the following variable were significant to predict saleprice of houses, MSSubClass, LotFrontage, LotArea,LandContour,LotConfig, LandSlope, Neighborhood, Condition1, HouseStyle, OverallQual, OverallCond, YearBuilt, MasVnrType, MasVnrArea, BsmtQual, BsmtExposure, BsmtFinSF1, BsmtFinSF2, BsmtUnfSF, CentralAir, X1stFlrSF, X2ndFlrSF, BsmtHalfBath, HalfBath, BedroomAbvGr, KitchenAbvGr, FireplaceQu, GarageType, GarageFinish, WoodDeckSF and MoSold
-

Final OLS model formula chosen:

```
Residual standard error: 45210 on 826 degrees of freedom
Multiple R-squared:  0.8051,    Adjusted R-squared:  0.7879
F-statistic: 46.75 on 73 and 826 DF,  p-value: < 2.2e-16
```

```
> RMSEstepfit
[1] 43315.24
> AIC(stepfit)
[1] 21921.36
> BIC(stepfit)
[1] 22281.54
> sum(vif(stepfit))
[1] 423.6303
>
```

#Over 78% of the variance of the response is explained by this model

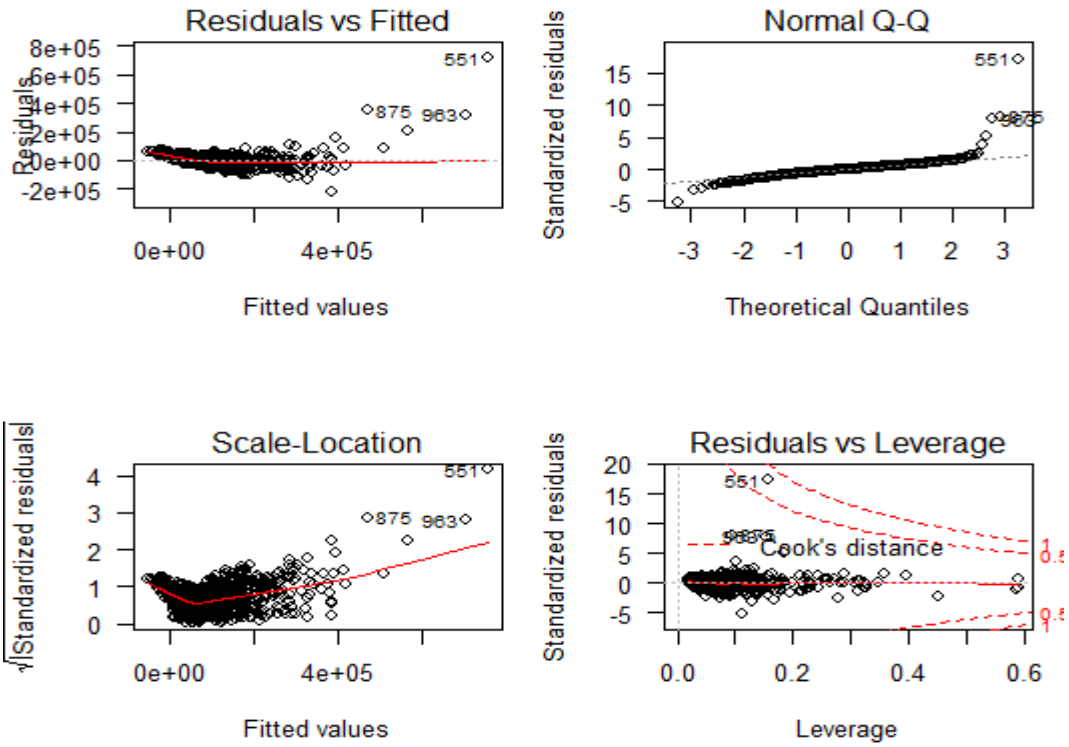
Summary of various OLS Models Tried

OLS	
Components = 70	Val_RMSE= 41160.03

OLS-Stepwise slelction	
components = 31	Val_RMSE= 43315.24

OLS - (eliminationg comp with VIF > 5)	
components = 28	Val_RMSE= 51510.18

Analysis of Residuals from chosen OLS model



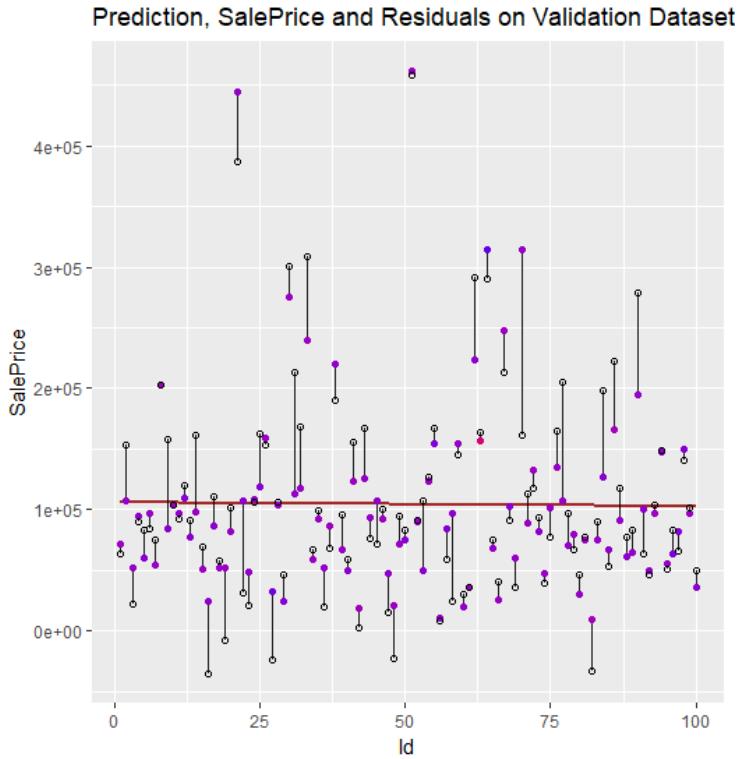
The Residual vs fitted plot shows the residuals of our models almost equal spread around a horizontal line, which indicates our model does not have a nonlinear relationship, but an improvement will spread the residuals around the line

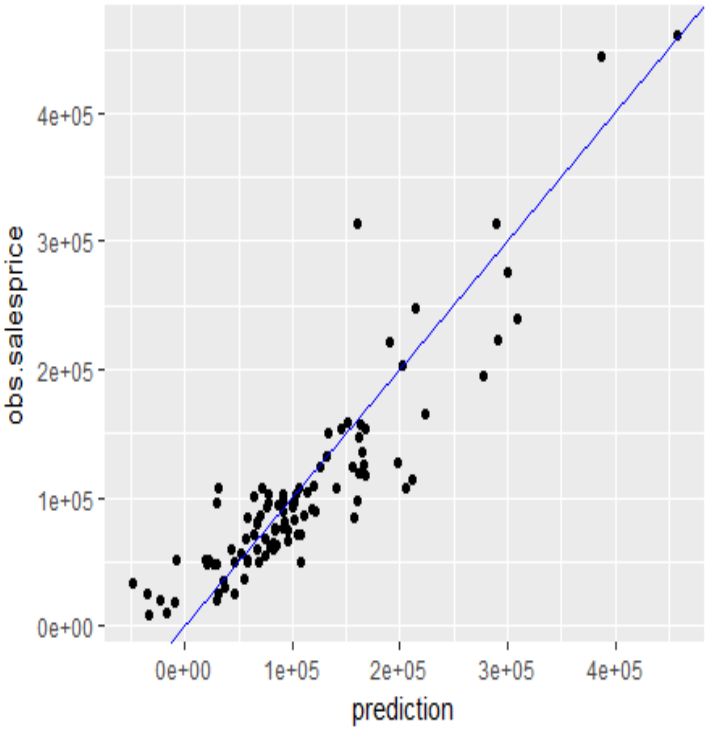
The Normal qq plot indicates our residuals are normally distributed as they're almost all lined well on the dashed straight line

The standardized residual vs fitted values plot indicates heteroscedasticity and we could improve our model by transforming the variables.

The residuals vs Leverage plot indicates there's no case outside the cook's distance hence, no case influential to the regression results

Testing our model on the validation data set,

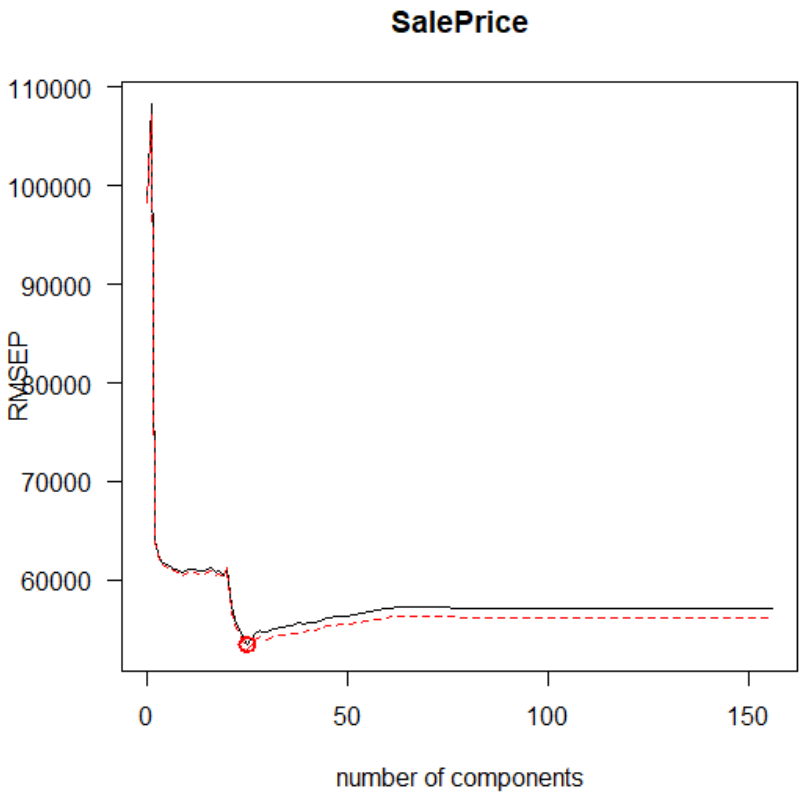




2. **Partial Least Square Model:**

The Second Machine Learning method we used is the Partial Least Square, from the Caret package. PLS finds a linear regression model by projecting the predicted variables and the observable variables to a new space. To avoid overfitting the model to the training dataset, we used in-built cross-validation features of the Caret package, in another attempt, we carried out recursive feature elimination (RFE). Recursive feature elimination is to select features by recursively considering smaller and smaller sets of features. First, the estimator is trained on the initial set of features and weights are assigned to each one of them. Then, features whose absolute weights are the smallest are pruned from the current set features. That procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached. after recursive training the above features are top features using Partial linear regression. According to RFE, the most important features are: OverallQual, GrLivArea, GarageCars, TotalBsmtSF, GarageArea

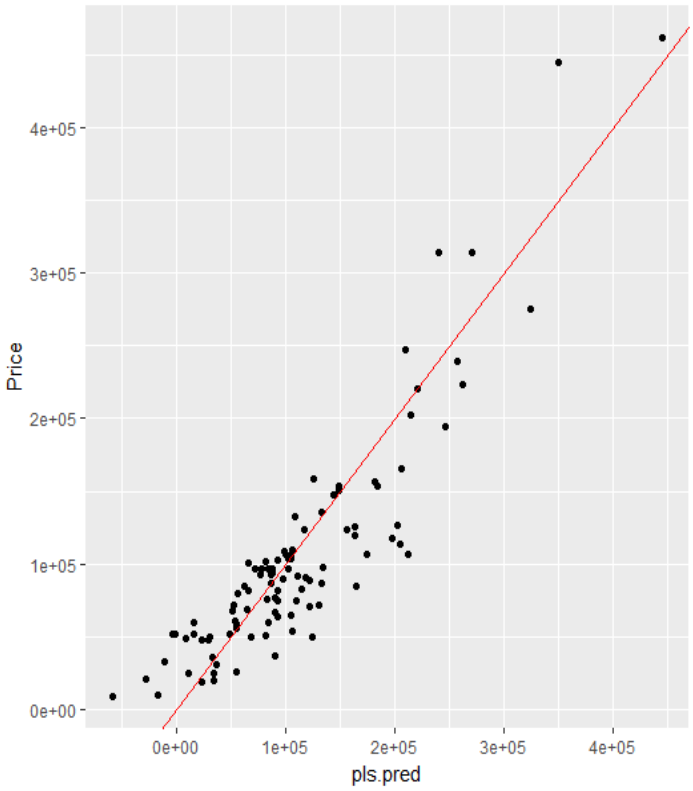
No of optimal components = 25

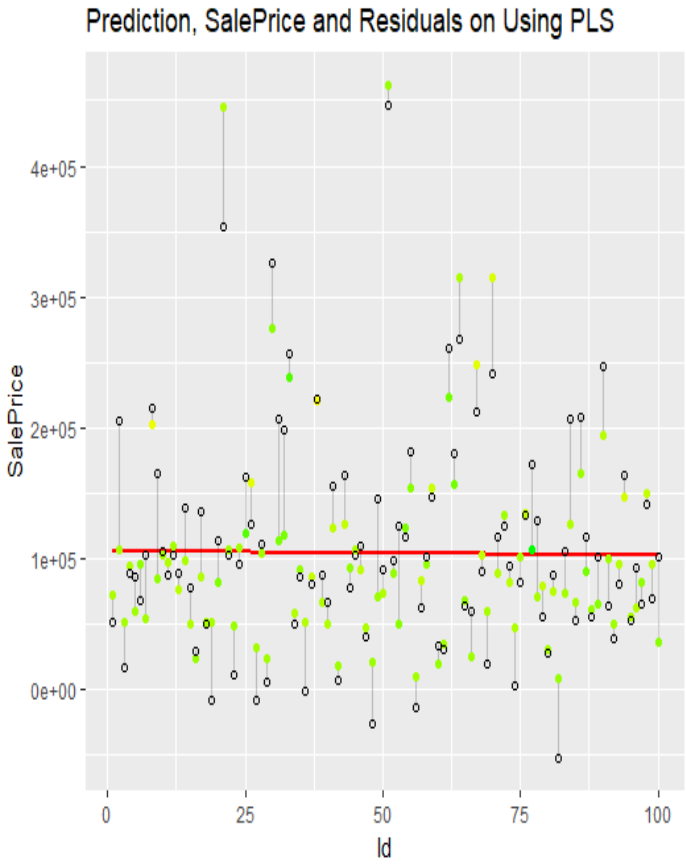


PLS Model-Rfe approach	
Component=5	Val_RMSE = 19142.5

PLS Model-RepeatedCV approach	
Component = 25	Val_RMSE = 44507.25

PLS Model-CV from Caret Package	
component = 30	Val_RMSE = 44165.91





The predicted and actual price are closer when compared to the plot of that from linear model, thereby indicating an improvement in prediction

```
> defaultSummary(data.frame(obs=obs.salesprice,pred=pls.pred))
        RMSE     Rsquared          MAE
1.918364e+04 9.405404e-01 1.357112e+04
```

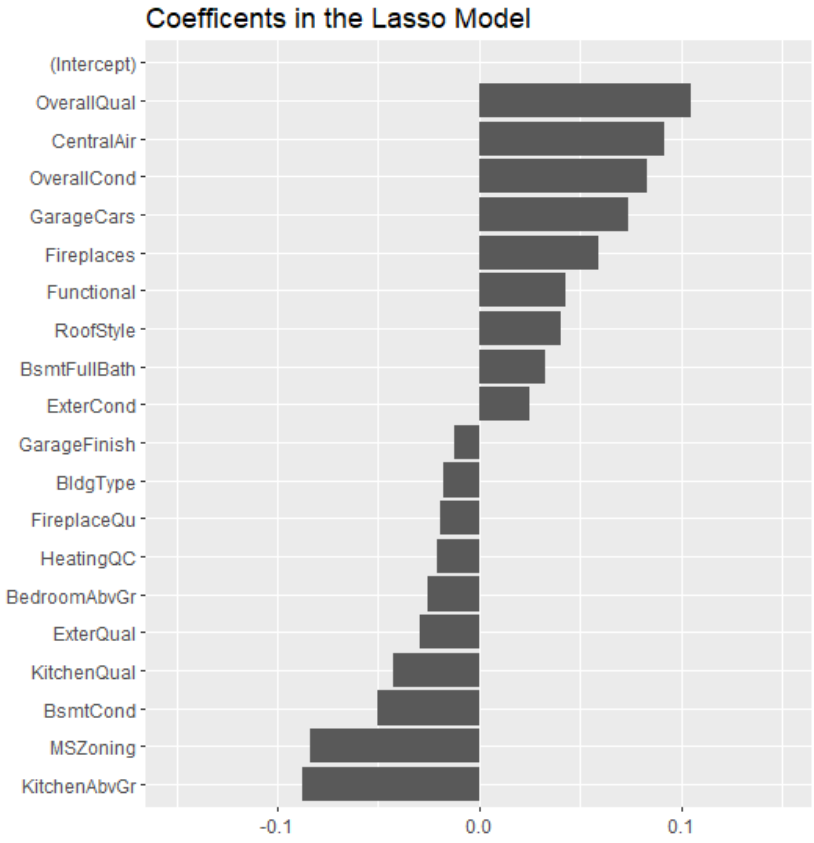
Lasso Model Approach

The regression models we tried is the lasso model, which is a regression analysis method that performs both variable selection and regularization in order to enhance prediction accuracy and interpretability of the statistical model it produces. One thing to note here however is that the features selected are not necessarily the ones to optimize your prediction - especially since there are a lot of collinear features in this dataset. One idea we used is to carry out a 5 repeat of 5-fold cross validation using the caret package to see how stable the feature selection is.

Lasso picked 58 variables and eliminated the other 12 variables

Hyperparameter tuning, best alpha and lambda value:

```
  alpha lambda
5     1  0.002
```

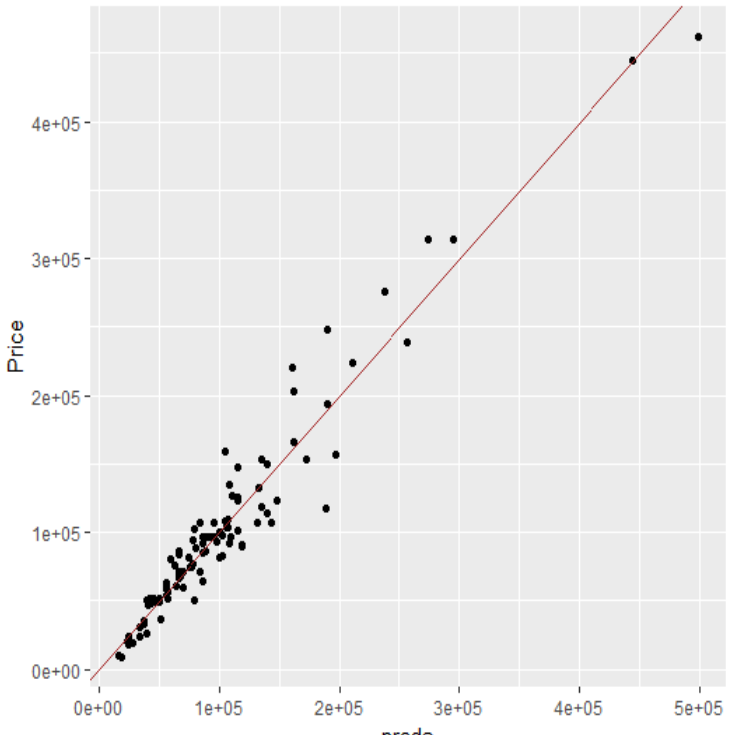


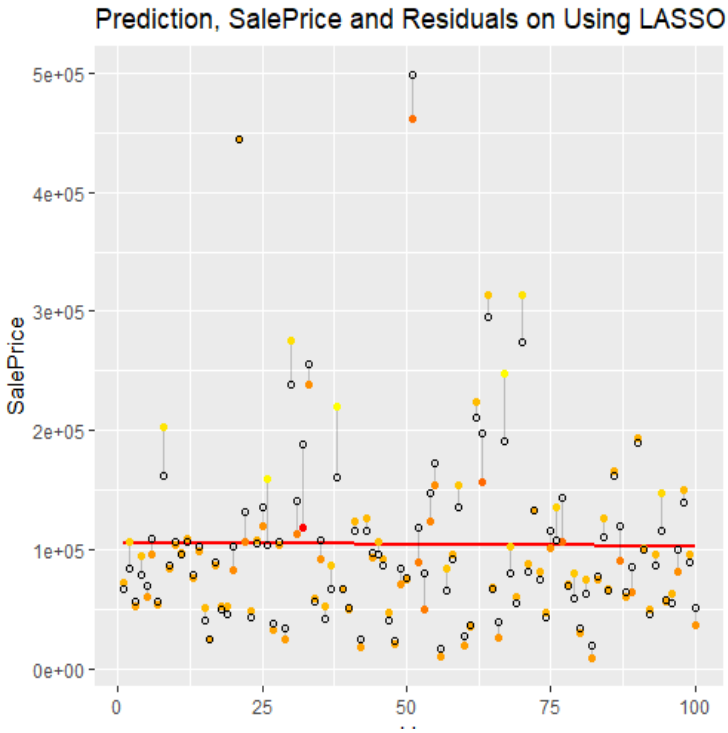
A plot of the coefficients of the top 10 and the bottom 10 nonzero features in our lasso model

We observed that the most important positive feature is GrLivArea - the above ground area by area square feet. This definitely make sense. Then a few other location and quality features contributed positively. Some of the negative features make less sense and would be worth looking into - it seems like they might come from unbalanced categorical variables.

RMSE of lasso model when fitted to the validation data. Lasso gives the best fit as shown below in the 2 plots below

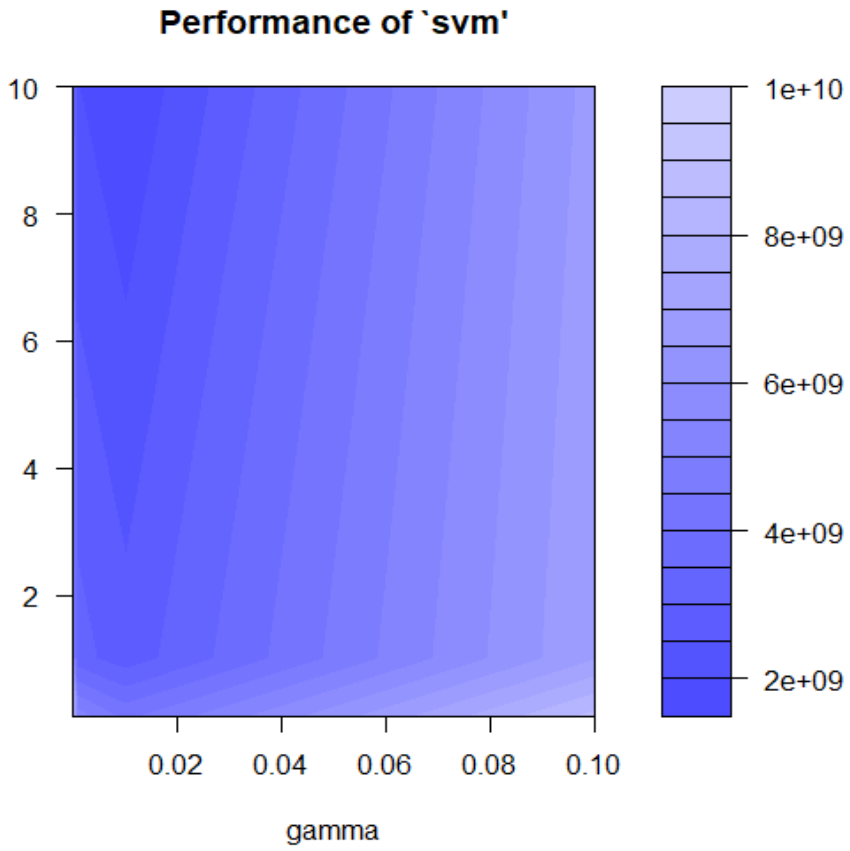
```
> RMSElasso
[1] 19724.23
```





Model of our Choosing- SVM

The next machine learning model we used in our prediction models is Support Vector Machines, are supervised learning algorithms that analyse data used for classification and regression analysis. We utilized the tune.svm function in e1027 package to select the optimal value of "cost" and "gamma", so we could have a good miscalculation trade-off of training data against simplicity of our decision surface.



This plot shows the performance of various models using color coding. Darker regions imply better accuracy. The use of this plot is to determine the possible range where we can narrow down our search for cost and gamma values to and try further tuning if required.

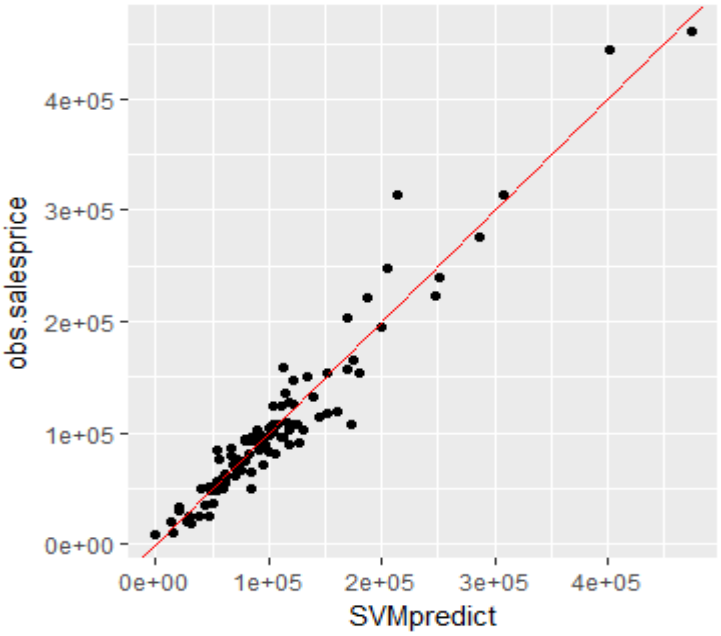
Best value of gamma and cost is summarized below

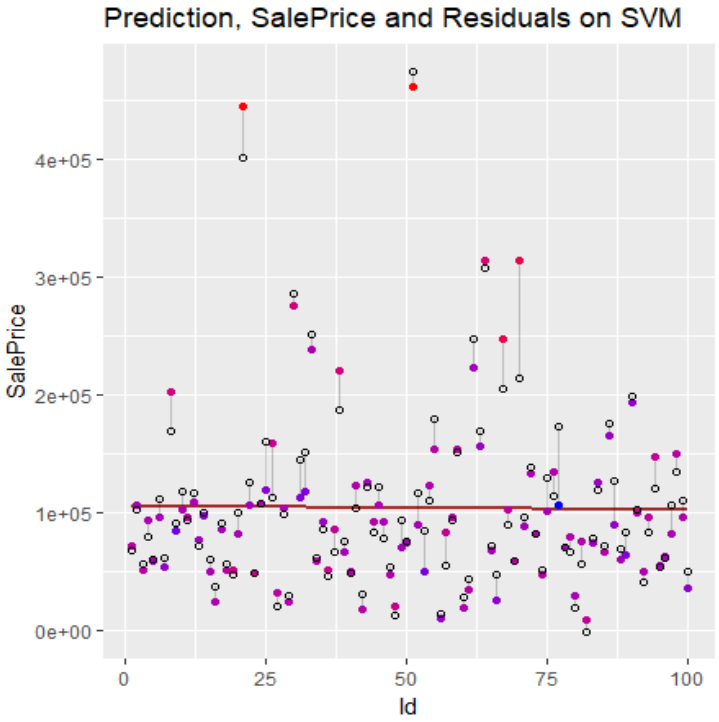
```
Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 gamma cost
  0.01   10

- best performance: 1572645969
```

We ran the regression using linear, radial and polynomial kernel, the polynomial to a power of

Some categorical variables were removed to prevent error in the svm model since their levels in the training and test data differs

```
Call:
svm(formula = SalePrice ~ ., data = housing.data.train[, -c(6, 20, 38, 51, 54)], kernel = "polynomial",
    gamma = 0.01, cost = 10)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  polynomial
       cost:  10
     degree:  3
      gamma:  0.01
     coef.0:  0
    epsilon:  0.1
```

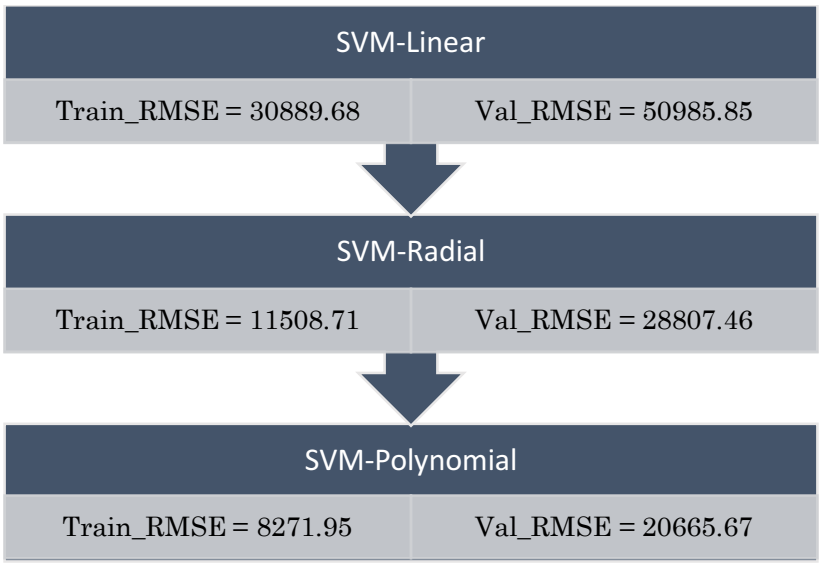




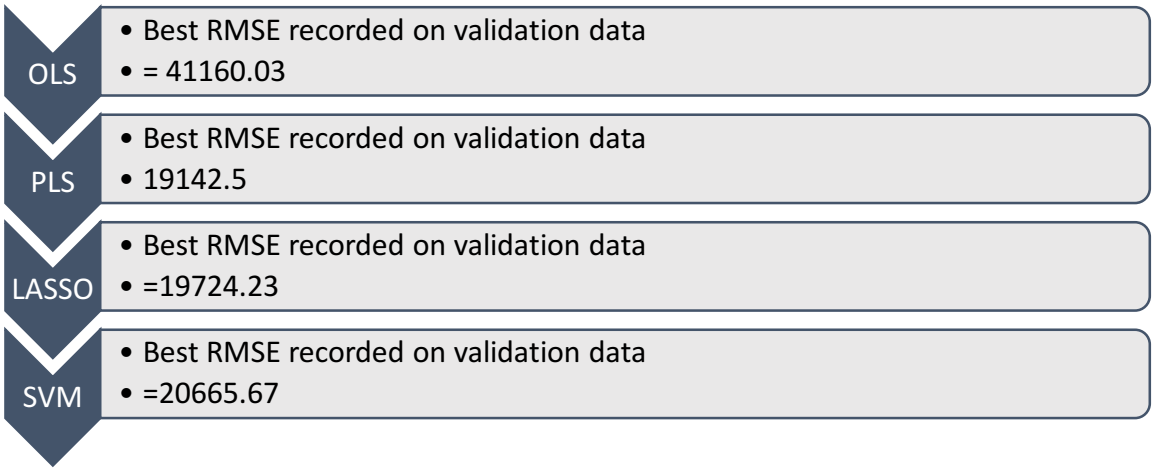
plot of actual Saleprice to predicted price on the validation dataset, using SVM model,polynomial kernel, cost = 10, gamma = 0.01

The plot showing deviation of actual SalePrice to the Predicted price for all Housing transaction covered in the validation data set

Summary of SVM regression carried out



SUMMARY OUR BEST MODELS IN EACH CASE (Based RMSE on the Validation data)



Future Directions

If given the luxury of additional time, we would dedicate more to engineering new features and checking interactions between features to improve our model predictions. A good understanding of the housing industry in US and more importantly in the state (or region) the data was collected would guide our decision in feature generation and interaction of features.