Regression Analysis

Kirtimaan Gogna
2019AIM1014@iitrpr.ac.in
Indian Institute of Technology
Ropar

Abstract

The purpose of this document is to report the observations and conclusions after experimenting with a given data set of Boston House prices. We use different features to predict the PRICE feature which is a dependent variable. We have done experiments by applying Linear Regression. Ridge Regression and Lasso Regression. We have performed visualizations on the dataset to better understand the data.

1 Introduction

Regression is a basic Machine learning algorithm that can be trained to predict real numbered outputs like temperature,stock ,price. Regression is based on a hypothesis that can be linear, quadratic, polynomial, non-linear ,etc. The hypothesis is a function that based on some hidden parameters and the input values. In the training phase, the hidden parameters are optimized with reference to the input values presented in the training. The process that does the optimization is the gradient descent algorithm. Once the hypothesis parameters got trained , then the same hypothesis with the trained parameters are used with new input values to predict outcomes that will again be real values. We are using Boston house data set from library on which we will be applying our regression algorithms.The following regression algorithms are applied on the dataset.

- Linear Regression
- Ridge Regression
- Lasso Regression

2 Data Analysis

After loading the Boston house price dataset from sklearn dataset. We split the dependent and the independent variables in separate dataframes. In this case dependent variable is price and the rest others are the independent variables. After separating the dependent variable and the independent variable we split the entire dataset in training and test data in the ratio of 70:30.

2.1 Linear Regression

After this we fit the training data in linear regressor. After training the data we obtain the coefficients corresponding to different features of the data. After plotting the bar chart for the

corresponding coefficients of the features, we observe that the linear regression coefficient corresponding to feature **RM** is the maximum and the coefficient value corresponding to feature **NOX** is the minimum. If we increase the values of RM feature the house prices will go up and if we increase NOX feature value the price will go down. Since our dataset is very small our linear regression model is prone to overfitting , to avoid this we use ridge regression which penalizes the coefficients and helps in fitting the data.

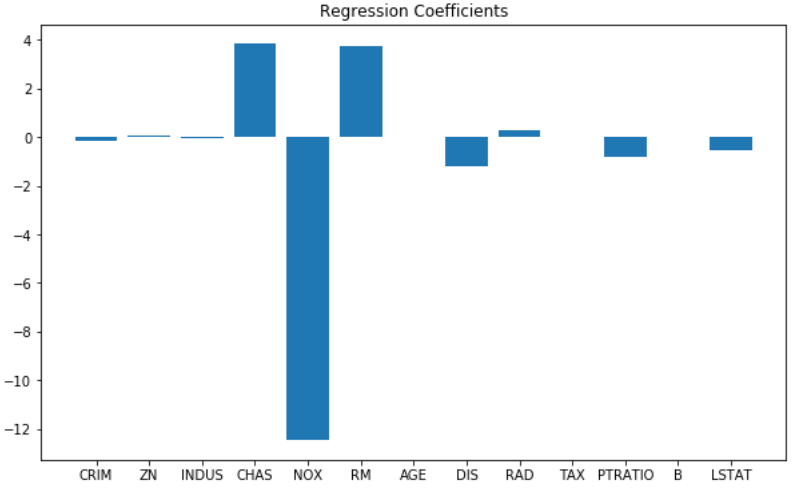

Figure 1: Linear Regression Coefficients

2.2 Ridge regression

The hyperparameter, λ, lets us control how much we penalize the coefficients, with higher values of λ creating simpler modelers. The ideal value of λ should be tuned like any other hyperparameter. In scikit-learn, λ is set using the alpha parameter. With 0 value of α the ridge regressor will act as linear regressor with the coefficient of feature RM beight the largest. We have changed the value of λ from 1 to 200 On increasing the value of λ we are penalizing the coefficient corresponding to RM as evident from the figure.

2.3 Lasso Regression

Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. . This particular type of regression is well-suited for models showing high levels of muticollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination. Lasso regression has a hyperparameter α which needs to be set. When the value of α is 0 it is acting as a linear regressor. It is not penalizing the coefficients. On increasing value of α the coefficients decreases and the model underfits. On increasing the value of λ more than 50, it makes all the coefficients zero.

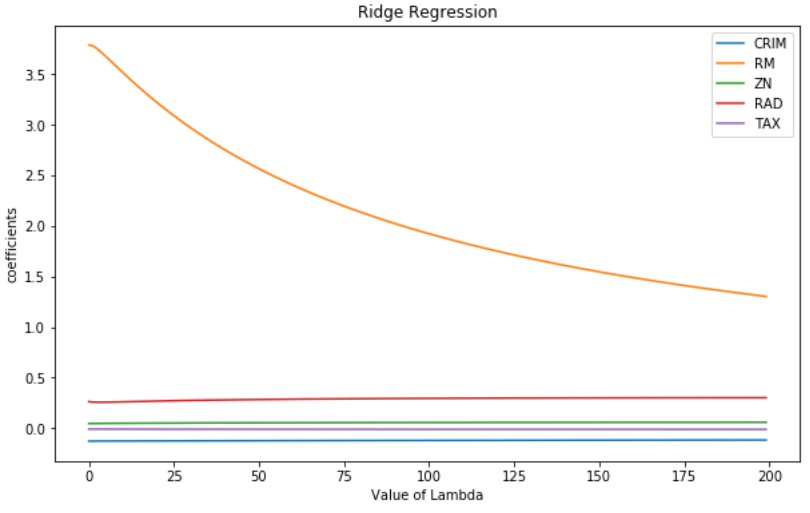

Figure 2: Ridge regression coefficients with different values of lambda

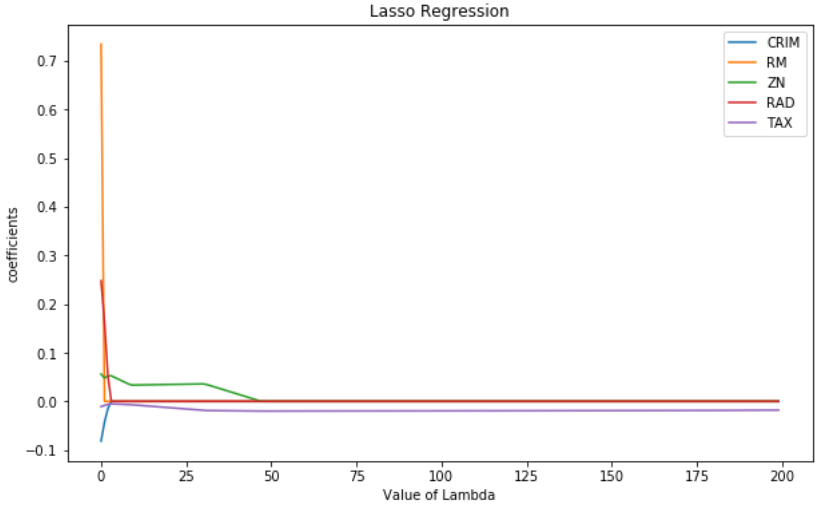

Figure 3: Lasso regression coefficients with different values of lambda

2.4 Visualizations

2.4.1 Linear regression

After applying linear regression on the training data, we plot a graph against the original price value and the predicted value given by our model. We can see that our predicted values are close to the original data.

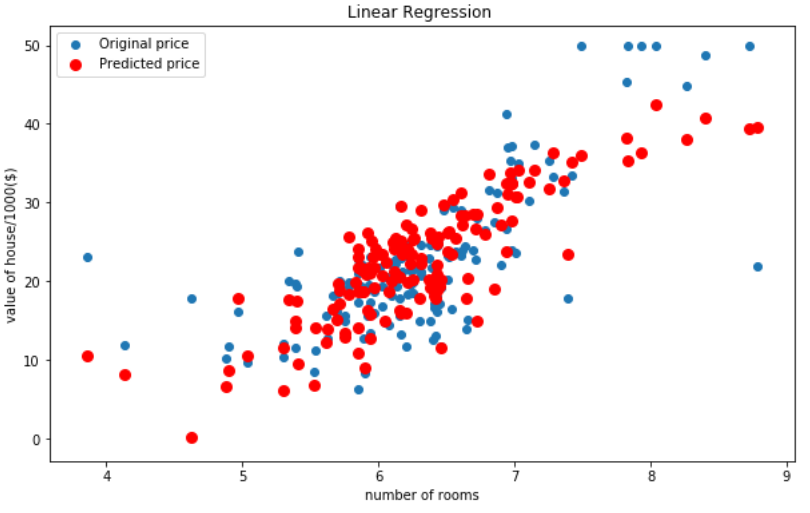

Figure 4: Linear regression scatter plot with test data and predicted data

2.4.2 Ridge regression

To avoid the overfitting on the model. We use Ridge regression. It is expected that after applying the ridge regression the overfitting should decrease and beyond certain limit of lambda the model will start to underfit. As we can see from the figure , this is what exactly is happening.

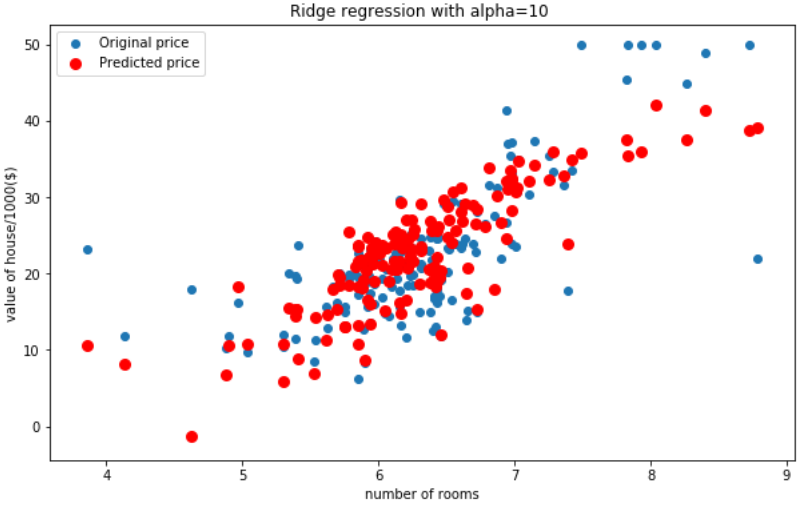

Figure 5: Ridge regression scatter plot with alpha=10

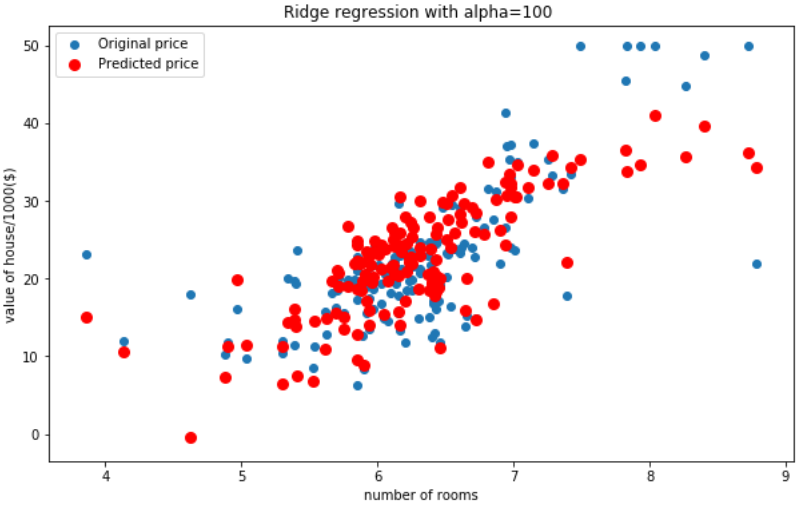

Figure 6: Ridge regression scatter plot with alpha=100

Figure 7: Ridge regression scatter plot with alpha=200

2.4.3 Lasso regression

Another method that we can try to reduce the overfitting is to use lasso regression. Lasso regression makes the unimportant features close to zero. While increasing the value of lambda, we can see that model begins to underfit and the accuracy for new data points decreases drastically.

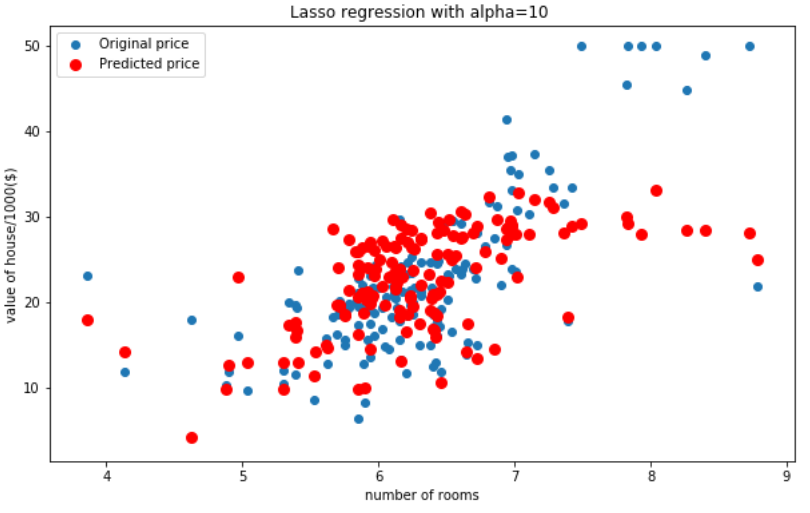

Figure 8: Lasso regression scatter plot with alpha=10

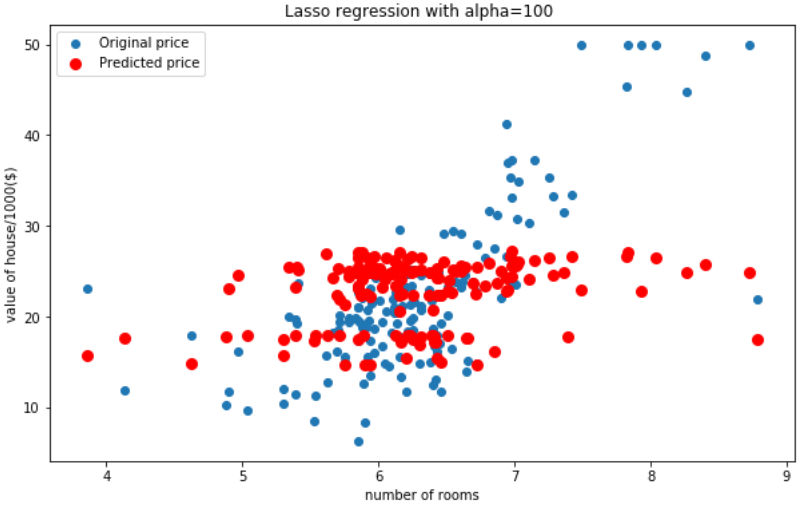

Figure 9: Lasso regression scatter plot with alpha=100

2.5 Errors

Finally after performing all the experiments, we calculate the training and the test errors for the various models.It is calculated as the mean squared errors. As we can observe from the table, When value of alpha is very small, the ridge regression is giving the error which is close to the linear regression training and test errors. Lasso regression is giving more training and test error compared to other models. Further increasing the value of alpha both the ridge and the lasso regression models are giving more error on the training and the test data.

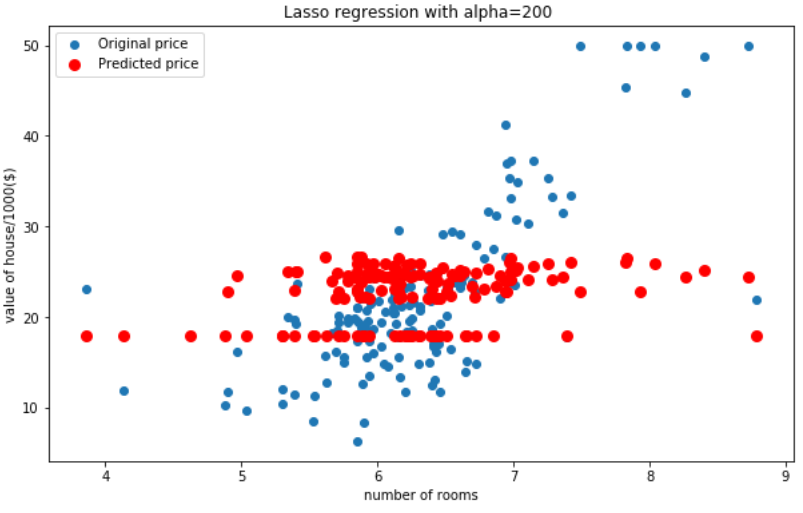

Figure 10: Lasso regression scatter plot with alpha=200

Model	Alpha	Train Error	Test Error
Linear	-	22.4197164	22.4319360
Ridge	10	22.9164874	23.8623139
Ridge	100	24.5585645	25.1227944
Ridge	200	25.5707444	26.3901162
Lasso	10	39.5484341	39.25655124
Lasso	100	67.6025045	59.1595356
Lasso	200	70.3572373	60.90278145

Table: Training and test errors on different models

3 Conclusions

- Linear Regression is prone to overfitting
- Ridge Regression on increasing the value of lambda beyond a certain point begins to undedrfit
- Lasso Regression on increasing the value of lambda beyond a certain point begins to underfit
- RM is the most important feature in the Boston House Data Set