Lasso regression

konda varshith

1/10/2020

Lasso and linear regression

Splitting of data

Underlying probabilistic model for linear regression

$$y = N(w_0 + w_+, \sigma^2)$$

Training and testing error for the linear model

$$1/n \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

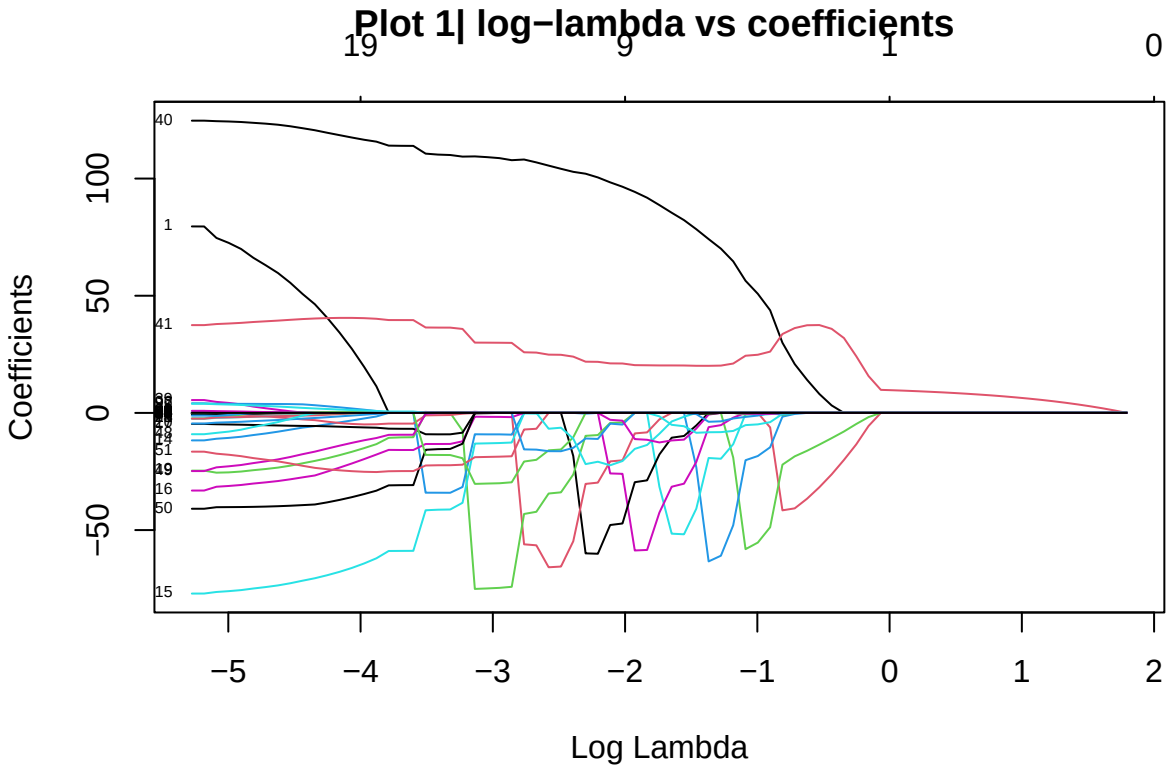
Train Error : 0.00570911701090834

Test Error : 722.429419336971

Underlying probabilistic model

$$\hat{w}^{lasso} = argmin\{\sum_{i=1}^{N} (y_i - w_0 - w_1 x_{1j} - ... - w_p x_{pj})^2 + \lambda \sum_{j=1}^{p} |w_i|\}$$

Model



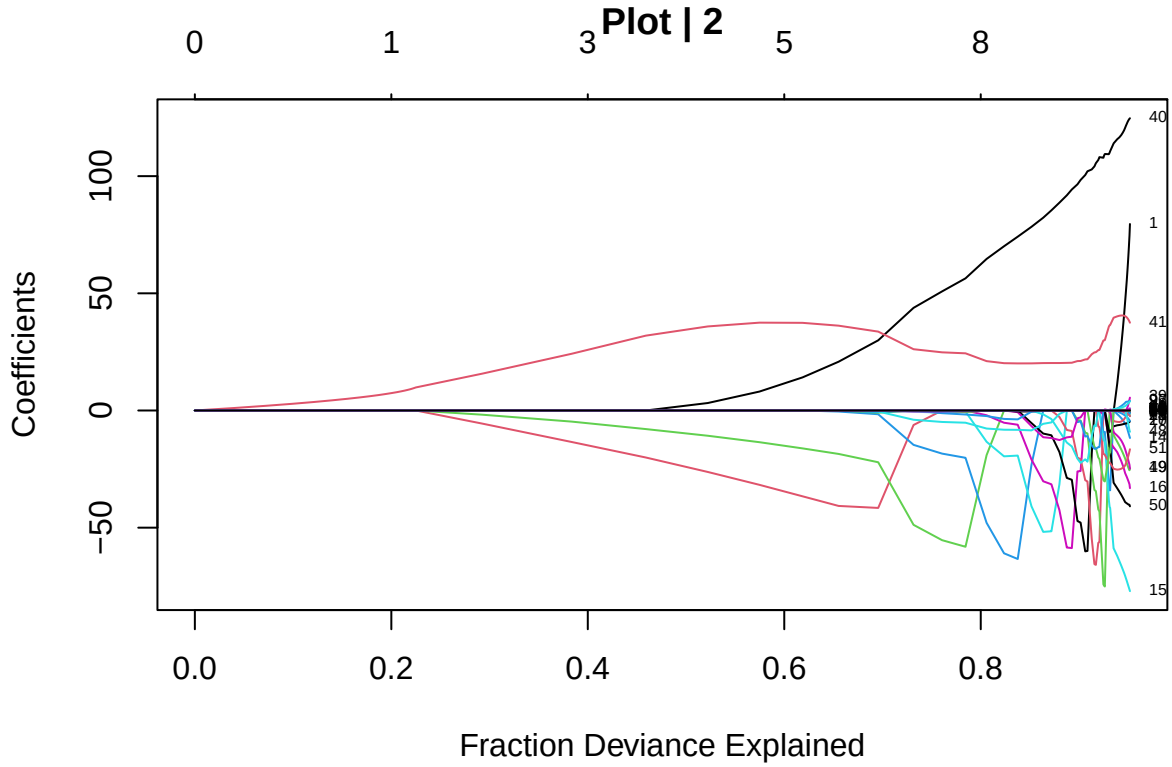
In the plot 1.1 we can interpret that coefficients of different channels were lowered to 0 by lasso regression.

Coefficient of channels 15 and are rapidly decreasing with increase in lambda values, where as channel 40 seems to decrease gradually.

Coefficients channel 41 is converging into 0 nearly at lambda value of 2,it seems to have more slow decline than channel 40,a slight irregularities can be observed from - 2 to 0.

Lambda values -3 to -1 can lead to overfitting of the data.

Plot to show dependence on "DEV"




```
##           Length Class     Mode
## a0            77 -none-    numeric
## beta        7700 dgCMatrix S4
## df            77 -none-    numeric
## dim            2 -none-    numeric
## lambda        77 -none-    numeric
## dev.ratio     77 -none-    numeric
## nulldev        1 -none-    numeric
## npasses        1 -none-    numeric
## jerr           1 -none-    numeric
## offset         1 -none-    logical
## call           5 -none-    call
## nobs           1 -none-    numeric
```

From the plot 2 we can notice that 8 channels can interpret 80 percent of the variable and 3 features can explain 40 percent of the variable.

It would be a better option to choose 8 or 5 features from the data as they can interpret sixty and eighty percentage of the variable.

Features above 80 percentage fall under over fitting region.

Choosing penalty factor from model

```
## [1] 0.8530452 0.7772630 0.7082131
```

Using VarImp function to crossvalidate the feature selection

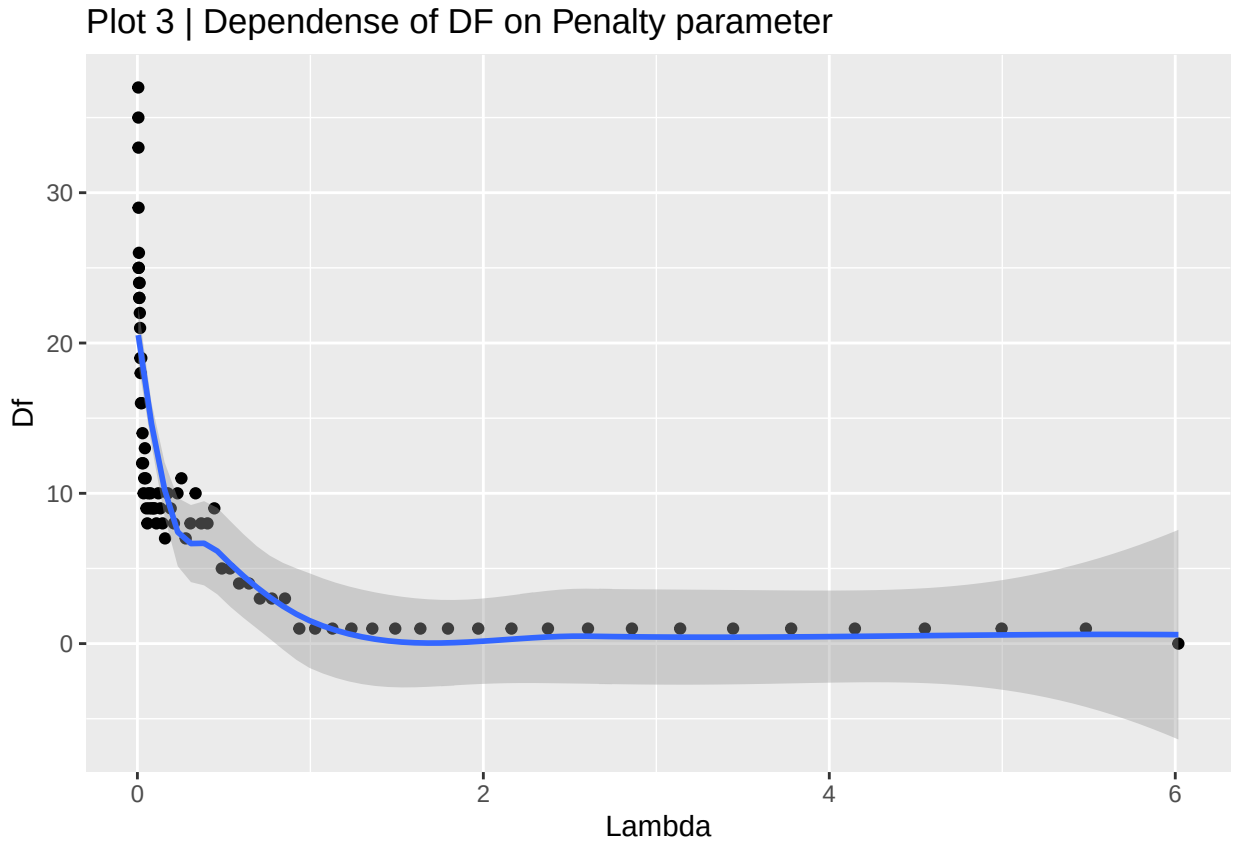
from the obtained lambda values and varImp function

As the lambda values lie between 0.7 and .85, I have chosen lambda as 0.8.

```
varimp <- varImp(lasso_model,lambda= 0.8)
```

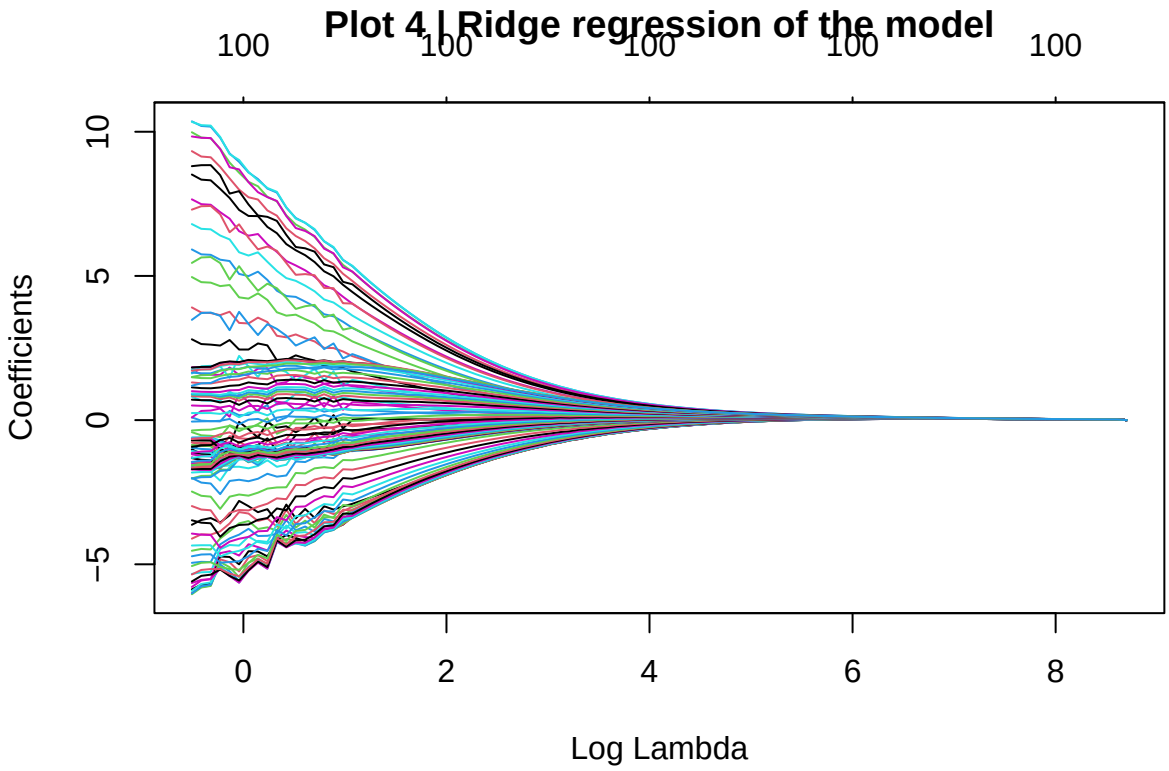
From the above steps we can prove that *lambda* can be 0.8 to obtain only three features

$$\lambda = 0.8$$



The degrees of freedom in gradually decreasing upto penalty factor of 1 and extended uniformly through remaining penalty factor values.

High concentration of points can be observed from 0 to 0.5 values.



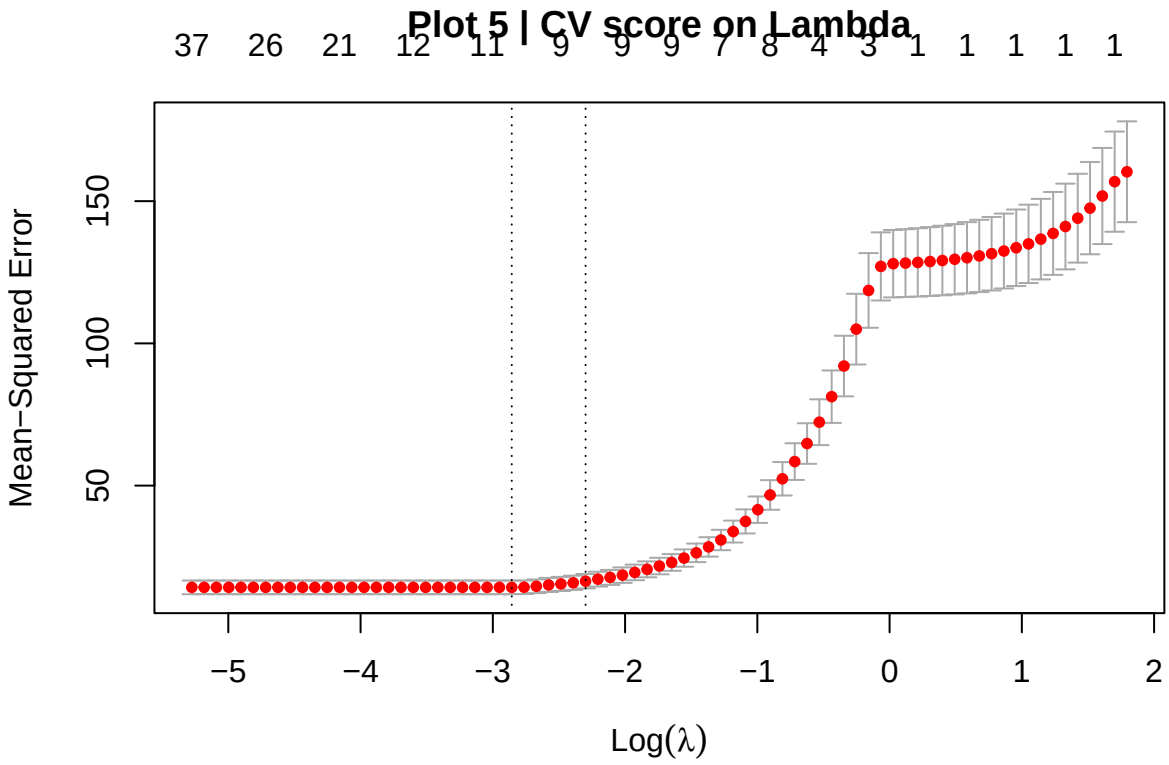
Ridge regression is using all the available features which intern makes model more complex.

From plot we can observe that ridge regression has shrunk coefficients to lower values but didn't turn them into zeros, this is due to ridge regression's penalty factor which penalizes high beta values thereby shrinking beta values.

Lasso has penalized the beta coefficients and enforced them to 0 , which excluded unnecessary features from model.

Cross validation model

```
##
## Call:  glmnet(x = x_train, y = y_train, alpha = 1, lambda = optimal_lambda)
##
##   Df  %Dev  Lambda
## 1  7 92.78 0.05745
```



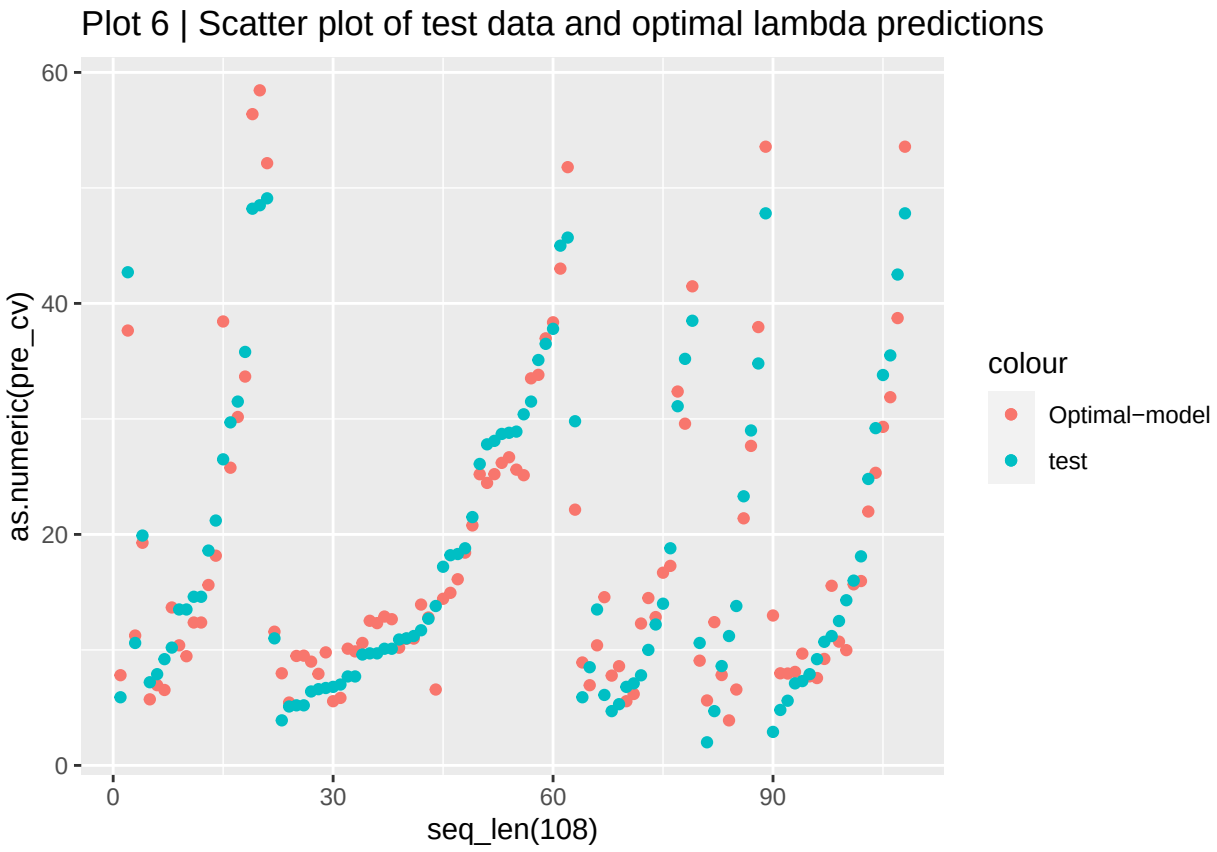
The optimal λ is 0.05744535 and 8 variables are chosen in this model.

The glmnet has highlighted two lambda values with one standard error apart, one closer to -3 gives minimum mean square error.

For large lambda values we observe minimal changes but as lambda decreases there's a rapid drop in mean squared error.

The region between these lambdas are best observed by the model , -2 as lambda has high MSE values,due to this model become biased.

Hence, choosing the value that increase the bias of a model is not recommended.



The model with optimal lambda predicted the test values with slight variations, it is not the best model but not a bad model.

Data generated does not fit the test data, but are nearer to the true values.

We can increase the quality of data generations with more tuning to the model, we should also cross check with other regression models for more insights.

References

1. https://beta.vu.nl/nl/Images/werkstuk-fonti_tcm235-836234.pdf
2. http://www.stat.cmu.edu/~ryantibs/statcomp-F16/lectures/train_test.html
3. https://bradleyboehmke.github.io/HOML/regularized-regression.html
4. https://hackernoon.com/practical-machine-learning-ridge-regression-vs-lasso-a00326371ece