

*Toronto*

**Regularization: LASSO and Ridge**

Soni Manana

a *College of Professional Studies****,*** *Master of Professional Studies in Analytics.*

***Subject****: ALY6015* ***NUID****: 002982645*

Under the guidance of

**Dr. Prof. Alex Maizlish**

**Introduction**

Dataset was selected from library (ISLR) in R and dataset has data of colleges of United States. This was all regarding private and community colleges of United states in 1995, that is it provided.

**Regularization: LASSO and Ridge?**

* Regularization is the method which is used to adjust the model to increase accuracy. It used to calibrate model’s cost function by addition of a term.
* Terms in LASSO and Ridge are almost same which is shown below.

**LASSO: λΣ|βj|**

**RIDGE: λΣ βj2**

* Beta (β) is basically difference between value of sample and predictor of that sample. In Lasso mod had taken whereas in Ridge it squared.
* Usage of LASSO and RIDGE. In nutshell, when sample size is small or number of predictor variable is less go for LASSO because it reduces difference between performance and variance and that’s the difference which we identify through Lambda (λ). It’s basically penalty for making wrong choices That’s how it calibrates the model. For ridge its vice and versa.
* So from lambda you can guess and minimize the error.
* Mean Squared Error (MSE) is the way to calculate accuracy of model. So, these model helps to improve by introducing bias to the variance which leads to good MSE.
* The less MSE the less error but both methods can reduce MSE to certain value.
* Drawback of both methods that plots are difficult to interpret as multiple variables incepts with 0.

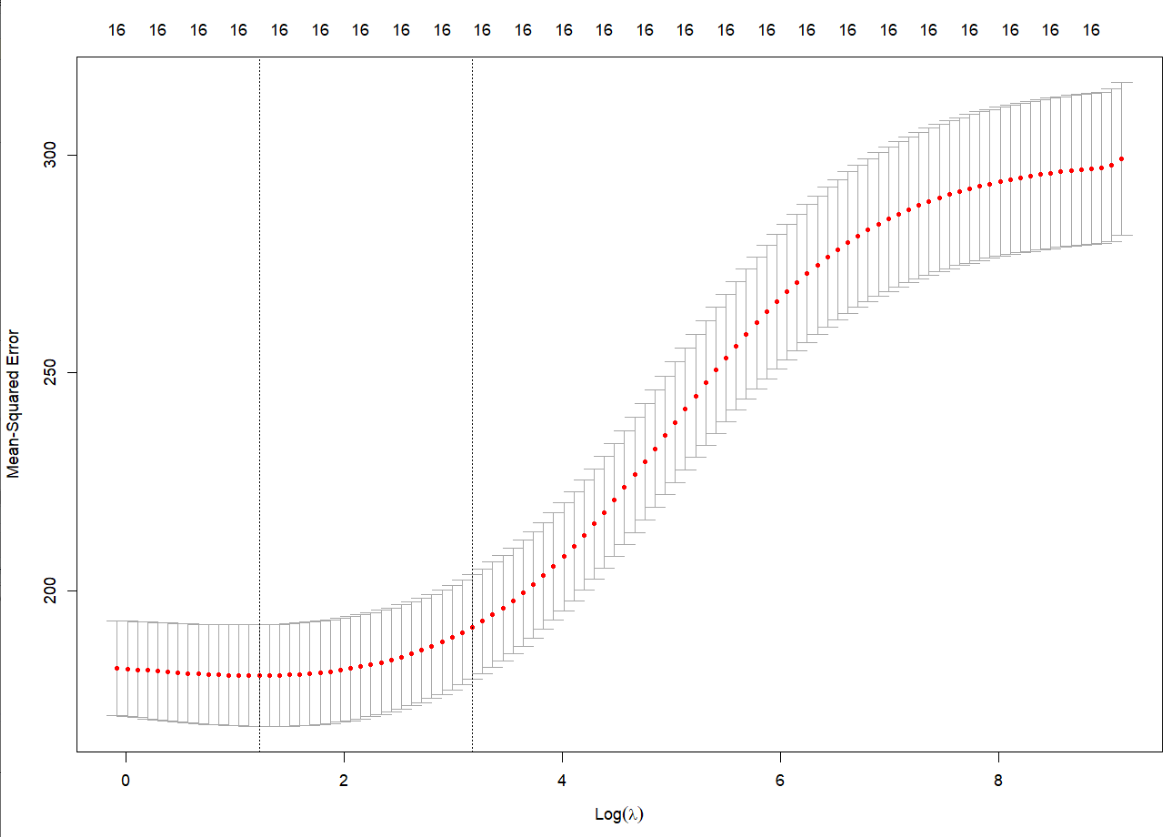
**Train-Test Split for Model**

We used to split the samples for training and testing the model. The threshold is 70% and 30% respectively.

Variables we have used for both the experiment are all of them expect graduation rate as we are trying to improve rate.

**Ridge Model**

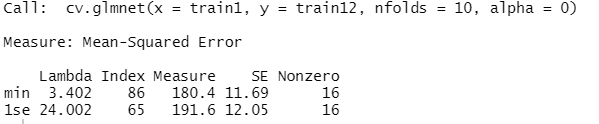
* Ridge model is to find that coefficient (Lambda) that lowers the RSS (Sum of squared residuals).
* In R, cv.glmnet() is used to perform this and It does standardization itself.
* This will return that lambda.
* And This will be the plot for the college dataset.
* Lambda min for this is 3.402
* It uses k-fold cross-validation and cv.glmnet() will take k=10 by default.
* We need to set a parameter alpha which is for choosing between LASSO and Ridge.
* For Ridge, its 0 and 1 is for LASSO in parameter.



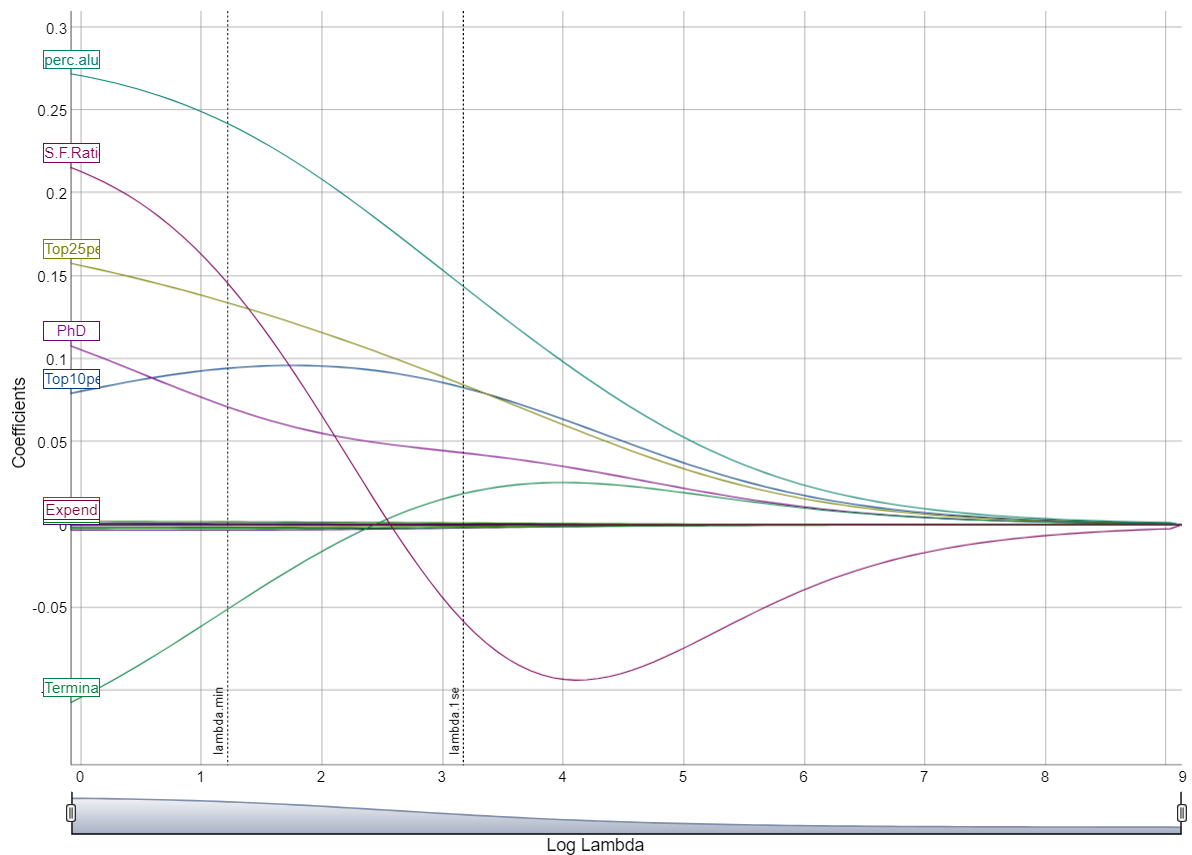
* This plot is hard to interpret but the second dotted line is around 3 as we can notice. So lowest point of the curve is the value of lambda.
* That red dotted line is of cross-validation with bounds of standard-deviation.
* Both dotted lines gives minimum mean cross-validation error.

> model$lambda.min

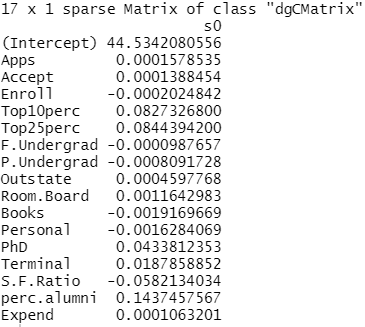
* This piece of code gives value of lambda.



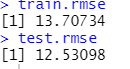
* Model performs or tests multiple lambda values and gives best of it. For lambda 0, model has no effect of methods and for some value > 0 its effective then for bigger value it goes to overfitting.



* This plot shows that at some point and by lambda, the variables are impact on variables. And at last they merge to 0.
* There is lambda.1se is should be calculated for most regulated model. To calculate this, glmnet() replace lambda.min to lambda.1se and for future scope lambda.1se is better.

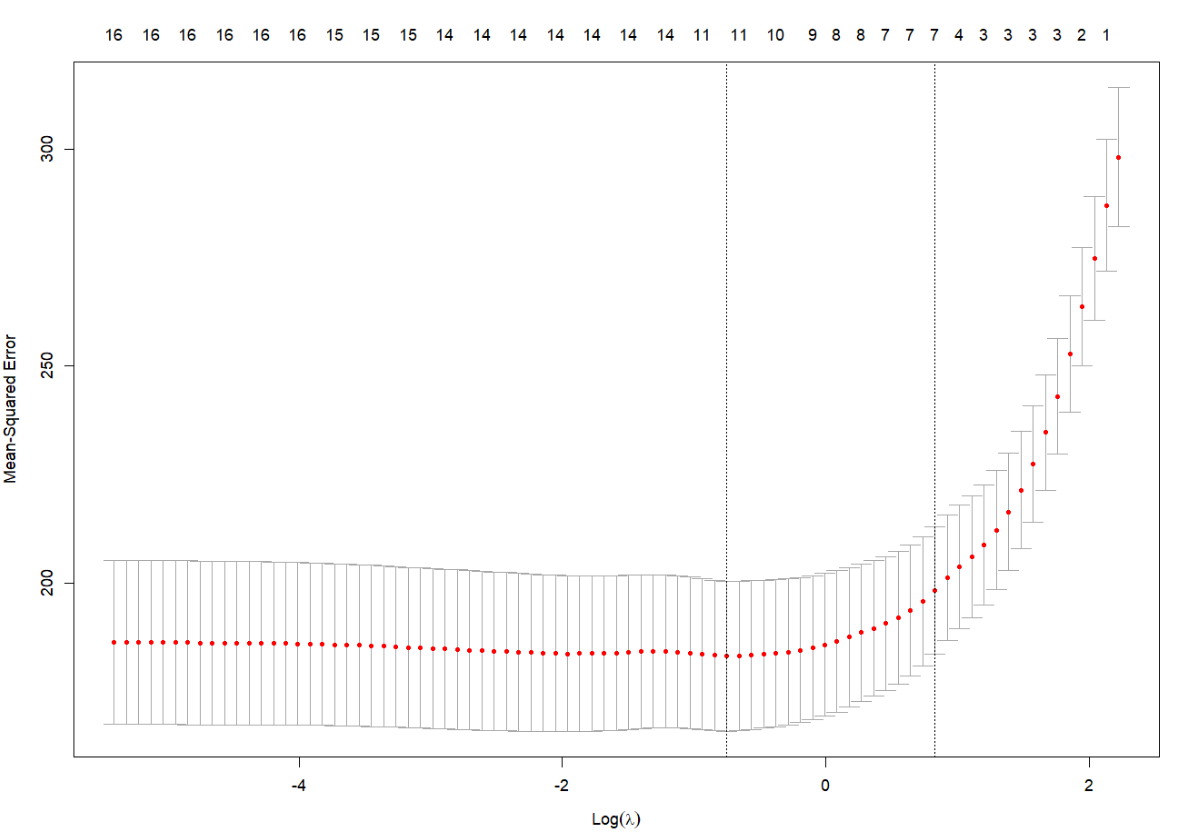


* This is the train RMSE value and test RMSE is respectively 13.707 and 12.530. Train RMSE is greater than Test RMSE. RMSE is basically Root mean square error, and its standard deviation of the residuals.

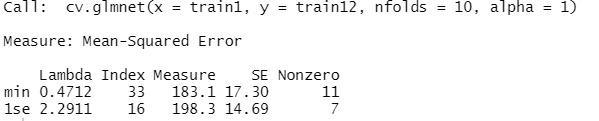
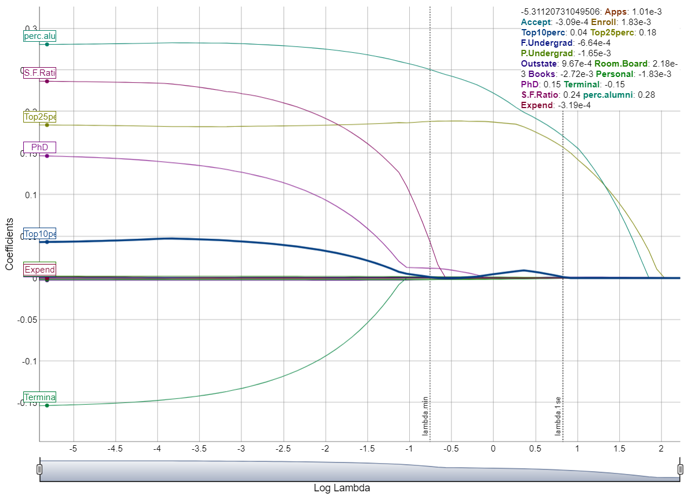


**LASSO Model**

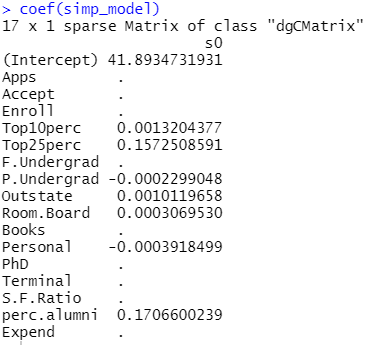
* LASSO is used when samples size is less as it used to eliminate variables unlike Ridge. We’ll see that later.
* Noticing as the two dotted lines, curve started to incline to 300, therefore, first line is first lambda.min and second is the value of lambda.se1.
* The indicators at the top of the plot, those number are of non-zero regression coefficients, which increasing from 1 to 16.



* As we discussed earlier, difference in this plot is , LASSO is tries to eliminate the errors.
* But its most effective when samples are limited or variables are less.
* Looking at value of lambda.min and lambda.1se. Accuracy actually improved at 1se not at min. Comparing both these plots, 1se seems to be good instead min.
* Alpha will be 1 for this test.



* The non-valued in this diagram are eliminated by LASSO. Intercept is reduced compared to RIDGE. To add to that it’s the main difference.



* The trained RMSE is 13.877 and test RMSE is 12.507. There is no particular threshold but RMSE of test < RMSE of train it could be underfitting for both methods.
* In this case both RMSE is almost similar.

**Conclusion**

By finishing this assignment, I understood the idea of regularisation. How to test if the version is under fitted or overfitting and what may be achieved in both of the cases. The essential distinction among Ridge and Lasso changed into additionally understood.

Both the methods, resulted same over here as RMSE of test was less than RMSE of train.

For not so large dataset one should use LASSO and for opposite one should use RIDGE to calibrate or improve the performance of model.

**References**

Utkarsh Kumar, 1st January 2020, Create Matrix and Data frame from lists in R Programming. *Source*: https://www.geeksforgeeks.org/create-matrix-and-data-frame-from-lists-in-r- programming

Zach, August 26 2021, When to Use Ridge & Lasso Regression *Source*: <https://www.statology.org/how-to-calculate-mse-in-r/>

Zach, November 11 2020, Introduction to Ridge Regression *Source*: https://www.statology.org/how-to-calculate-mse-in-r/

**Appendix**

pacman::p\_load(tidyverse, rio, party, Metrics, ggplot2, dplyr, vcd, webr, Matrics, ggridges, simex, Hmisc, plotrix, gginference,corrplot,ggplot2,ggpubr,BSDA,glmnet,coefplot,ISLR,caret)

#Importing Dataset

View(College)

#Split the data into a train and test set – refer to the Feature\_Selection\_R.pdf document for information on how to split a dataset.

set.seed(123)

IndexSet <- sample(2, nrow(College),

replace = T,

prob = c(0.7,0.3))

datatrain <- College[IndexSet == 1,]

setdata <- College[IndexSet == 2,]

#Removing private variable

College = College[,-1]

view(College)

datatrain = datatrain[,-1]

setdata = setdata[,-1]

view(datatrain)

train\_x <-model.matrix(Grad.Rate~.,datatrain)[,-1]

test\_x <- model.matrix(Grad.Rate~.,setdata)[,-1]

train\_x2 <- datatrain$Grad.Rate

test\_x2 <- setdata$Grad.Rate

#Ridge

set.seed(123)

glmRidge <- cv.glmnet(train\_x, train\_x2, nfolds = 10, alpha = 0)

glmRidge

plot(glmRidge)

coefpath(glmRidge)

best\_lambda <- log(glmRidge$lambda.min)

best\_lambda

simp\_lambda <- log(glmRidge$lambda.1se)

simp\_lambda

simp\_model <- glmnet(train\_x, train\_x2, alpha = 0, lambda = glmRidge$lambda.1se)

coef(simp\_model)

simp\_model

pred\_train <- predict(simp\_model1 , newx = train\_x)

train.rmse <-rmse(train\_x2, pred\_train)

train.rmse

pred\_test <- predict(simp\_model1 , newx = test\_x)

test.rmse <-rmse(test\_x2, pred\_test)

test.rmse

# Lasso

set.seed(123)

glmLasso <- cv.glmnet(train\_x, train\_x2, nfolds = 10, alpha = 1)

glmLasso

plot(glmLasso)

best\_lambda <- log(glmLasso$lambda.min)

best\_lambda

simp\_lambda <- log(glmLasso$lambda.1se)

simp\_lambda

simp\_model <- glmnet(train\_x, train\_x2, alpha = 1, lambda = glmLasso$lambda.1se)

coef(simp\_model)

simp\_model

coefpath(glmLasso)

pred\_train <- predict(simp\_model , newx = train\_x)

train.rmse <-rmse(train\_x2, pred\_train)

train.rmse

pred\_test <- predict(simp\_model , newx = test\_x)

test.rmse <-rmse(test\_x2, pred\_test)

test.rmse

step(lm(Grad.Rate ~ ., data = College), direction = 'both')

model\_step <- step(lm(Grad.Rate ~ ., data = College), direction = 'both')

summary(model\_step)