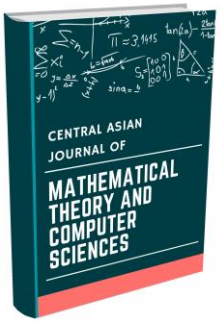


CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES

http://cajmtcs.centralasianstudies.org/index.php

Volume: 01 Issue: 01 | Sep 2020

Cross- Validation Method for Overfitting Research Data using R Programming

Yagyanath Rimal

[1]Faculty of Science and Technology, Pokhara University, Nepal
Corresponding E-mail: rimal.yagya@pu.edu.np

ABSTRACT

This review analytical paper clearly discusses the mathematically compression between various types of regression analysis using R programming when data sets of large dimension (Boston housing) of having overfitting and multicolinarity problems. Whose machine learning outputs were analysed with different log lambda and varimp plots were analysed and explained sufficiently for interpretation of data. Its primary purpose is to explain the different types of regression models (like as Ridge, Lasso and Elastic net) whose data structure were suffers from cross validation and overfitted data structure using R software, whose mixing percentage plot, log lambda plots were sufficiently explain with various intermediate output and graphical interpretation to reach the final conclusion. Beside that this paper clearly shows the steps for analysing ridge, lasso and elastic net regression with different sample data sets so that the research gap between which tools will be followed by the researcher after data collection will clearly explained when the data sets of different attributes on relationship. Therefore, this paper presents easiest way of regression analysis when data sets with multicolinarity and its strengths for data analysis using R programming.

ARTICLE INFO

Article history:
Received 10 July 2020
Received in revised form 24 July 2020
Accepted 31 August 2020

Keywords:
Data Analytics; Big Data; Regression Equation, Multicolinarity, Overfitting. Lasso, Ridge, Elastic Net.

1.INTRODUCTION

Regression analysis is a statistical process that allows a researcher to estimate the linear relationship between the search variables that is widely used for forecasting and forecasting research data considering some previous information values. Data analysis always requires adequate data analysis tools that predict future planning and, moreover, the most important challenge for modern data scientists is to process data, make decisions and forecast data. Linear regression summarizes the association of variables with related variables (Fiona, 2018). For example, the relationship between a company's sales and advertising, sales records depend, while the factors that influence sales are independent relationships that change with the dependent variable or vice versa. Statistical regression analysis would easily help solve the various relationships numerically. Therefore, regression analysis is an appropriate model for adjusting the relationship between dependent and independent datasets. However, the experimenter could set his priority to make independent and dependent decisions accordingly. In the regression model, the independent variable is labeled as variable X and the dependent variable is variable Y. The relationship between X and Y can be displayed on a graph, with the variable X the horizontal axis is independent and the dependent variable Y is on the vertical axis. The objective of the regression model is to determine the linear relationship that links X and Y (Astrid Schneider, 2010). The straight line connecting two of the variables X and Y can be explained mathematically as Y = a + bX, where it is called Y intercept b is the slope of the regression line. If the intersection and slope of the line can be completely determined by the relationship between the search variable. The intersection is the point where the regression line crosses the Y axis. Similarly, the slope of the line b concerns the slope of the line, if the line up or down sharply with respect to the search data. Therefore, the linear regression model of the forecast can be applied to single or multiple independent or dependent variables. The linear regression equation for multiple variable is Y = B1 + B2X2 + B3X3 + ... BK + € where Y is an estimated dependent variable and X are the independent variables with € is the error term and B1 are regression coefficients. Although, each regression technique has a default hypothesis that must satisfy the user's requirements before performing the regression. Once

the slope and intersection are determined, the line extends to infinity in any direction. Therefore, the simple linear regression formula Y = βo + β1X1 + € where Y is dependent and X1 are independent variables. B1 is the term coefficient is used to calculate the ratio of the independent variable, B0 is the intersection of two lines through when the regression line is located on the y axis, the € is error terms, whose analysis the regression concludes that for each independent variable the value of y increases. Therefore, it is concluded that every variation of 1 unit in an independent variable increases its value of the dependent variable if all the other coefficients were constant.

Ross Ihaka and Robert Gentleman develop the programming language R is an open source statistical programming language, it is free and is supported by a large community. The R software is both a software and a programming language in which the user can develop many programs. It is also a scripting language in which we can write many lines of code and generate results in the console command that is available for all platforms (Linux, Mac and Windows). Although "Now there were more than 10,000 R packages available" to download (Smith, 2018). (Machlis 2018) The Data Editor and Analytics Executive said that several large companies such as Microsoft, New York Times, Google Maps, Google, Amazon and Facebook that increase the use of the R programming language for server management due of their open source IDE. and is freely distributed under the term GNU, whose precompiled binary files are distributed on the CRC website (Comprehensive R Archive Network) (Paradis, 2005).

The R programming language supports many functions for statistical analysis and graphic display in the window. Some intermediate results, such as p-value, regression coefficient and residuals, can easily occur, so the data analyst can draw conclusions and predictions freely. The similar study package is available on https: //www.rstudio.com. products/rstudio/download. The object stores user-defined in the work area are automatically reloaded the next time it starts (Kopf, 2017) A New KDnuggets explained that there were only four languages domain for Analytics, which are R, SAS, Python and SQL used by 91% of data scientists and that the popularity of other languages decreases, except for Julia and Scala (Piatetsky, 2014). Similarly, the author (Jeevan, 2018) explained that there are R, Python, SAS, My SQL and Java are the most demanding for data analysis in modern century languages, and emphasized the analytical business decision for data science. Therefore, data scientists in the modern world must know the best solutions for particular tasks to meet the future needs of the organization.

However, atlas.it bloggers have stated that R is not so good at collecting data. Many new packages like reshape2 allow users to manipulate data frames to meet the criteria set for the requirements, although R is an exploratory analysis. Many models can be written with a few lines of codes. There are many advanced packages like ggvis, lattice and ggplot2 that support the ability to display R data programming and R, data is stored in data frames that can be used and reused throughout the program without hindering performance (Pedregosa, 2011) . Language developed by statisticians for statisticians in which Python is easier to learn the generic programming language (Nasridinov, 2013). The most important correlation method is the correlation coefficient of Karl Pearson or the Pearson correlation coefficient. R is between -1 and 1. Since the value of r, there is a certain correlation when r is -1, it means that there is a perfect negative correlation negative correlation, and r is 0 implies no correlation.

Crest regression is a technique for analyzing multiple regression data that suffer from multicolinarity. When multivariate occurs, the least squares estimates are unbiased, but it has large variations that can be far from the real value, reducing standard errors (Lane, 2018). The collinearity of linear relationships between independent variables (Eaton, 2018). However, the division of a very small quantity still manipulates the results. Therefore, one of the first steps in a regression analysis is to determine if multicollinearity is a problem. Therefore, collinearity can create inaccurate estimates of regression coefficients, inflate standard errors of regression coefficients, define partial t tests for regression coefficients, send false p values, not significant and degrade predictability regression coefficients. so detection of parity diagrams of multicolored scattered pairs of independent variables in search of almost perfect relationships. Also take a look at the correlation matrix for high correlations. Unfortunately, multicollinearity does not always appear when the variables are considered two by two. When the inflation factors of variance (VIF) exceed only the 10 collinear variables indicated.

The regularization helps to solve the adaptation problems, which implies that the model works well with the training data, but that it works poorly with the validation data (test). The regularization solves this problem by adding a penalty to the goal. The regularization is useful in the following situations: high number of variables, low ratio between number and number of variables and high multicollinearity. The regularization will try to minimize the function by adding a penalty to the sum of the absolute values of the coefficients. This regularization helps to resolve the adjustment problems in the training and validation data of the tests. The regularization of coefficient values. regression squares of the contraction model coefficients in the ridge regression Thus, in the crest regression model it

$$\text{Min}\,(\sum\varepsilon^2 + \lambda\sum\beta^2) = \text{Min}\,\sum(y - (\beta_1 + \beta_2 X_2 + \beta_3 X_3 + \ldots + \beta_k X_k))^2 + \lambda\sum\beta^2$$

i

Where, λ is the regularization term of non negative value. and do not assume normality in the error terms. The constraint is just on the sum of squares of regression coefficients of X's. On solving the above model can get the estimates of coefficient β as:

$$\hat{\beta} = (X'X + \lambda I)^{-1}X'y$$

If it use lambda = 0 then we get back to the usual OLS estimates. If lambda is chosen to be very large then it will lead to under fitting. Thus it is highly important to determine a desirable value of lambda. To tackle this issue, we plot the parameter estimates against different values of lambda and select the minimum value of λ after which the parameters tend to stabilize.

1.1 Ridge Regression

Here we are considering the BostonHousing data set, where there were dependent and independent variables. Where should we load the glmnet library to perform the crest regression as:
library (glmnet)
Similarly, the cv.glmnet () function performs a cross-validation with a predetermined alpha value = 0. Lambda is a sequence of different values with the application of cross-validation. model = cv.glmnet (as.matrix (X), y, alpha = 0, lambda = 10 ^ seq (4, -1, -0.1)). The best lambda is considering using lambda.min of the regression coefficients using the prediction function. What is known as the absolute minimum deviation method in the regression of the cycle is another regularization of the addition of a penalty term to the sum of the squares of the coefficients which is the contraction regression which is used in the regression of the ridge when one adds a restriction on the sum of the squares of the regression coefficients. So, in the regression of the crest, our function is: $Min(\Sigma 2\varepsilon 2+\lambda\Sigma 2) = Min\Sigma(y-(\beta 1+\beta 2X2+\beta 3X3+...))^2+ \lambda\Sigma\beta 2$
Where λ is the regularization parameter of a non-negative number. The restriction is only in the sum of the squares of the regression coefficients of Xs β like
$:\beta= (XX + \lambda)-1XY.$
If we choose lambda = 0 then the OLS estimates. If you choose lambda to be very large, insufficient adjustment will occur. Therefore, it is very important to determine a desirable lambda value. To reduce this problem, we need to calculate the parameter estimates with respect to different lambda values of the minimum value of.

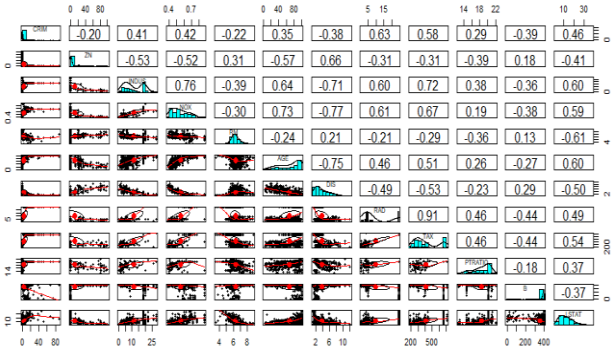
1. 2 R Code for Ridge Regression

```
> library(caret) #Classification and Regression Training
> library(glmnet)# Lasso and Elastic-Net Regularized Generalized
> library(psych) #Procedures for Psychological, Psychometric, Research
> library(mlbench)# Machine Learning Benchmark Problems
> library(readxl)
> BostonHousing <- read_excel("C:/Users/Yagya/Desktop/BostonHousing.xls")
> View(BostonHousing)
> data=BostonHousing
> str(data) #describes variables
Classes tbldf, tbl data.frame: 506 obs. of  14 variables:
 $CRIM : num .00632 .02731 .02729 .03237 .06905
 $ ZN    : num  18 0 0 0 0 0 12.5 12.5 12.5 12.5 ...
 $ INDUS : num 2.31 7.7 7.7 2.18 2.18 2.18 7.87 7.87 7.87
 $ CHAS  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ NOX   : num  .538 .469 .469 .458 .458 .458 .524 .524
 $ RM    : num  6.58 6.42 7.18 7 7.15 ...
 $ AGE   : num  65.2 78.9 61.1 45.8 54.2 58.7 66.6 96.1
 $ DIS   : num  4.09 4.97 4.97 6.06 6.06 ...
 $ RAD   : num  1 2 2 3 3 3 5 5 5 5 ...
 $ TAX   : num  296 242 242 222 222 222 311 311 311
 $ PTRATIO: num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 15.2
 $ B     : num  397 397 393 395 397 ...
 $ LSTAT  : num  4.98 9.14 4.03 2.94 5.33 ...
 $ MEDV   : num  24 21.6 34.7 33.4 36.2 28.7 22.9 27.1
> ?data
> ?BostonHousing# displays database variable description of data dictionary
```

The most variable in data set were in numeric type but CHAS is factor variable and the MEDV variable will be forecasted based on other 14 variable of 506 records. If you wish to exclude some variables from data sets.

```
> pairs.panels(data[c(-4,-14)],cex=2)# making pairs of 4th and 14th variable separation
> pairs.panels(data[c(-4,-14)],cex.cor=2) # creates scatter plot of every combination of two pair data when independent variable are numeric if they were highly correlated creates multicolinarity problem in such case multiple linear .
```

Figure No 1: Scatter plot of every possible combination of independent variables.



The table describes when numeric independent variables are highly correlated create multicolinarity if we apply multiple regression the output may not stable therefore we use ridge regression which shrinks coefficient to non zero values to prevent over fit, but keeps all variables. SSE ridge=$\Sigma(y-y-)^2+\lambda\Sigma B^2$

Data Partition

```
>set.seed(2222)# #Setting the seed to get similar results
>ind=sample(2,nrow(data),replace=T,prob=c(.7,.3))# splitting sample 70 30
>train=data[ind==1,] # 70%
>test=data[ind==2,]# 30% data
```

Two independent samples

Out of 506

test	149 obs.of 14 variables
train	357 obs.14 variables

Custom Control Method

custom control method apply cross validation of 10 fold train data of 10 split of train data with verboseIter method shows that we can see the process of iteration.

```
>custom=trainControl(method ="repeatedcv", number=10,repeats=5,verboseIter = T)
```

Linear Model

```
>set.seed(1234) # for consistency
>lm =train(MEDV ~ .,train,methods='lm', trControl=custom)
```

When taking regression of MEDV variable with all other variables with method lm produce outputs as – Fold 10.Rep5: mtry=13
Aggregating results

Selecting tuning parameters
Fitting mtry = 7 on full training set
Result
> lm$results
mtry RMSE Rsquared MAE RMSESD RsquaredSD MAESD
1 2 3.488882 0.8567224 2.447922 1.0574862 0.11960763 0.3940890
2 7 3.186282 0.8681266 2.238802 0.9325114 0.09890747 0.3549665
3 13 3.279558 0.8590564 2.296690 0.8358370 0.08735615 0.3417537

The model displays root mean squares R-squared coefficient of determinant 0.8567224 more than 85 percentage of variability can be seen to MEDV because of model MAE implies mean absolute error and root mean Rsquared standard deviation

> lm # displays detail of model
Random Forest
353 samples
13 predictor
No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 317, 318, 318, 318, 319, 317, ...
Resampling results across tuning parameters:

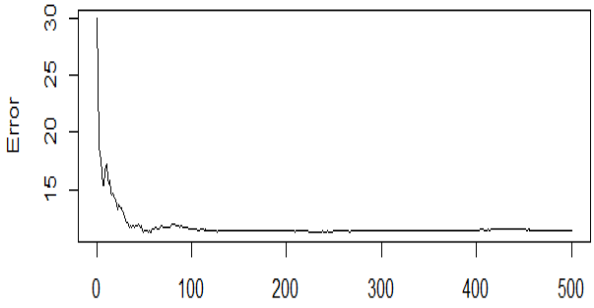
mtry	RMSE	Rsquared	MAE
2	3.488882	0.8567224	2.447922
7	3.186282	0.8681266	2.238802
13	3.279558	0.8590564	2.296690

RMSE used to select the optimal model using the smallest value.
The final value used for the model was mtry = 7.
The training data use cross validation of 10 fold with 5 reputation with 317 of each repetitions are used to create model and 10 part for errors and root mean squared shows the excluded variables of remaining variables.

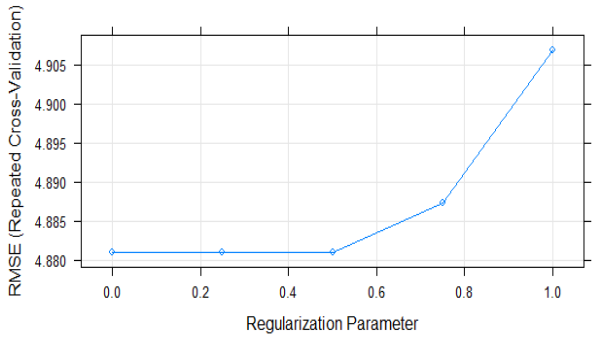
>summary(lm)

	Length	Class	Mode
call	5	-none-	call
type	1	-none-	character
predicted	359	-none-	numeric
mse	500	-none-	numeric
rsq	500	-none-	numeric
oob.times	359	-none-	numeric
importance	13	-none-	numeric
importanceSD	0	-none-	NULL
localImportance	0	-none-	NULL
proximity	0	-none-	NULL
ntree	1	-none-	numeric
mtry	1	-none-	numeric
forest	11	-none-	list
coefs	0	-none-	NULL
y	359	-none-	numeric
test	0	-none-	NULL
inbag	0	-none-	NULL
xNames	13	-none-	character
problemType	1	-none-	character
tuneValue	1	data.frame	list
obsLevels	1	-none-	logical
param	1	-none-	list

Those variables which do not have *(NULL) were not significant to the relation with MEDV.
>plot(lm$finalModel)
Figure2: Final Model



The above figure describes the relationship between MeDV values and ts error terms.
Ridge regression which use all variable
> set.seed(1234)
> ridge=train(MEDV ~.,train,method='glmnet',tuneGrid=expand.grid(alpha=0,lambda=seq(0.0001,1,length=5)),trControl=custom)
+ Fold10.Rep5: alpha=0, lambda=1
- Fold10.Rep5: alpha=0, lambda=1
Aggregating results
Selecting tuning parameters
Fitting alpha = 0, lambda = 0.5 on full training set
Here the best value of lambda is 0.5 is hyperparameter using of cross validation then as it is increase which ultimately decrease the coefficient
>plot(ridge)
Fig: 3 Lambda Plot



The figure describes the higher value of lambda increasing the penalty, in y axis the Root mean square regularization parameter predicts the best value of lambda is 0.5
>print(ridge) # which prints the best regression model
glmnet
359 samples 13 predictor
No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 324, 324, 324, 323, 323, 322, ...
Resampling results across tuning parameters:

lambda	RMSE	Rsquared	MAE
0.000100	4.880956	0.7203274	3.387177
0.250075	4.880956	0.7203274	3.387177
0.500050	4.880956	0.7203274	3.387177
0.750025	4.887344	0.7199132	3.385015
1.000000	4.906954	0.7185504	3.380644

Tuning parameter 'alpha' was held constant at a value of 0

RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 0 and lambda = 0.50005 is the best fits.
>plot(ridge$finalModel,xvar="lambda",label=T)
Figure 4: Lambda Log

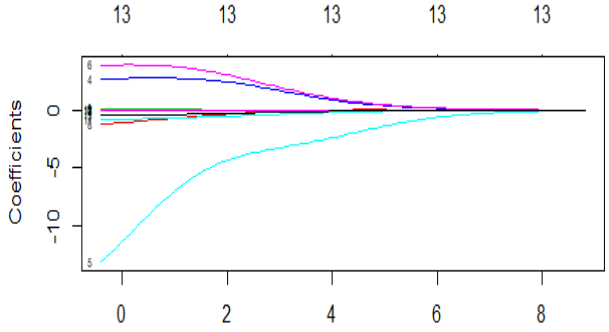
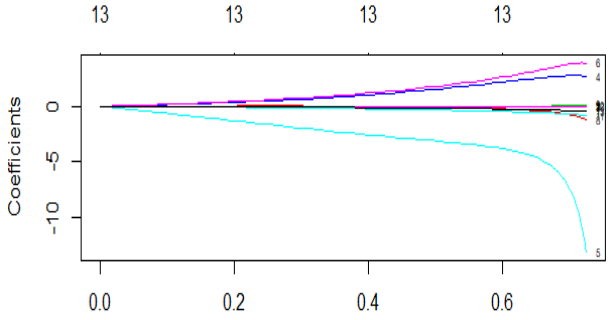
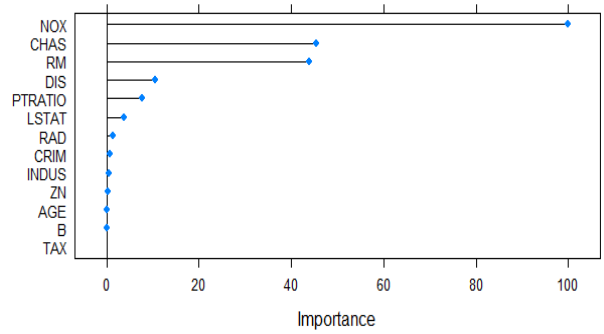


Figure describes when x axis value of Log Lambda and Coefficient ratio increases when increases by 9 increasing lambda decreases coefficients which are not contributing the model.
> plot(ridge$finalModel,xvar='dev',label=T)# describes fraction in deviance

Figure 5 Coefficient and fraction deviance explained



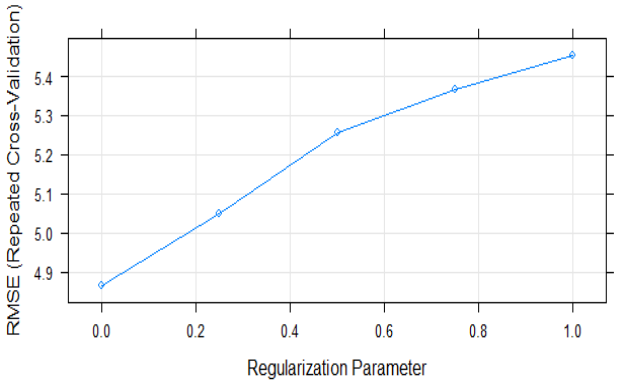
The figure fraction deviance explains the coefficient and deviance explained in every20 percentage of intervals when it reaches to 80% there the coefficient were highly influenced after 60% describes overfitting of data
> plot(varImp(ridge,scale=T))
Figure 6 Important variables in data sets



The variable NOX CHAS RM have most significant in the relation whereas TAV and B were least significant due to low coefficient in the model. When scale sets T the data shows 0 to hundred. Therefore Ridge regression has both shrinkage as well as feature selection

1.3 Lasso Regression

When the variables have a multiple collision, the cycle regression is selected to select a variable and ignore others. LASSO (operator of contraction selectors Least Absolute) is quite similar to the crest, but we understand the difference when we implement it in our big problem. LASSO is a regression method that involves penalizing the absolute size of the variable. The regression of the loop reduces the regression coefficients with some reduced to zero, so it helps with the selection of features. Loop SSE = Σ (y-y-) 2 + λΣ | B |. Therefore, the objective function in the Lasso regression becomes SSE cycle = (y−y−)2+|B|. Where λ is the intercept term of the non-regularized regularization parameter. The loop regression can perform the selection of variables as well as the contraction of the parameters. While using ridge regression, you can end up getting all the variables with reduced parameters. Where the best lambda value is lambda.min of the model and, therefore, obtaining the coefficients using the model prediction fits better. The regression of the cycle and the crest approaches with multicollinearity. However, ridge regression is computationally more efficient than cycle regression. So the best approach is to select the regression model that fits the test set..
>set.seed(1234)
>lasso=train(MEDV ~.,train,
method='glmnet',tuneGrid=expand.grid(alpha=1,lambda=seq(0.0001,1,length=5)),trControl=custom)
+ Fold10.Rep5: alpha=1, lambda=1
- Fold10.Rep5: alpha=1, lambda=1
Aggregating results
Selecting tuning parameters
Fitting alpha = 1, lambda = 1e-04 on full training set
>plot(lasso)
Figure 7: Regularization Parameter



Describes the higher value of Root mean square is higher when lambda increases parameter when we run this model with 0.02 range the lowest value is very close to zero which is the best value.
>lasso
glmnet
353 samples
13 predictor
No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 318, 317, 318, 317, 318, 318, ...
Resampling results across tuning parameters:

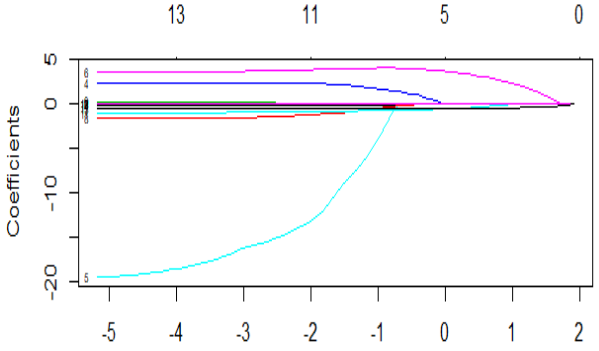
lambda	RMSE	Rsquared	MAE
0.000100	4.213231	0.7824496	3.027462

0.250075	4.430437	0.7632108	3.134944
0.500050	4.605248	0.7480758	3.290835
0.750025	4.685227	0.7445949	3.369183
1.000000	4.788167	0.7398345	3.451829

Tuning parameter alpha was held constant at a value of 1 RMSE was used to select the optimal model using the smallest value. The final values used for the model were alpha = 1 and lambda = 1e-04.

> plot(lasso$finalModel,xvar = 'lambda',label=T)

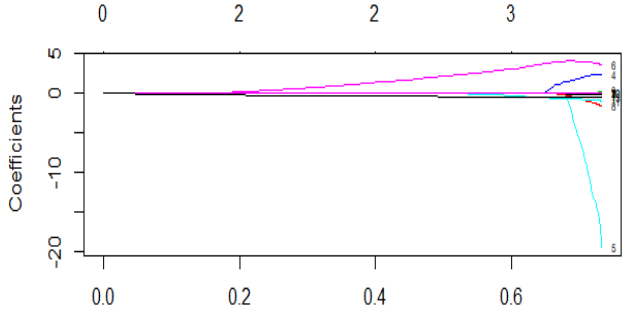
Figure:8 Relationship between Log Lambda



Describes the variable used in plot the coefficient is reduced when we reduced lambda. The above variable is better fitted than lower variable

> plot(lasso$finalModel,xvar = 'dev',label=T)

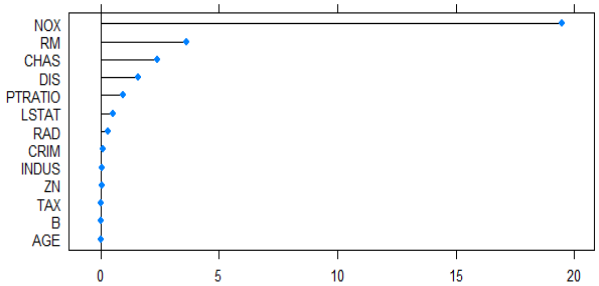
Figure 9 Coefficient fraction deviance relationship



Describes the similar confident whose value were dramatically change when deviance increases 60% and more the three variables could explained within 60% variability when coefficient increases more than 60% there were more overfitting data's. The lower variable has less important in model increase very rapidly compared others.

> plot(varImp(lasso,scale=F))

Fig 10 Variable Important graph



Describes the NOX, RM CHAS has highest important whereas AGE and B and TAX has least significant.

1.4 Elastic Net Regression

Elastic net regression is preferred over both ridge and lasso regression when one is dealing with highly correlated independent variables. It is a combination of both L1 and L2 regularization.

$$\text{Min}\left(\sum\varepsilon^2+\lambda_1\sum\beta^2+\lambda_2\sum|\beta|\right)=\text{Min}\sum(y-(\beta_1+\beta_2X_2+\beta_3X_3+\ldots+\beta_kX_k))^2\lambda_1\sum\beta^2+\lambda_2\sum|\beta|$$

Unlike ridge and lasso regression, elastic net regression assume normality. Elastic net regression is mix of ridge and lasso Σ(y-y-)2+λ[(1-α)ΣB2+αΣ|B| . When alpha=0 Elastic net regression is equal to ridge when alpha= 1 it becomes lasso. Setting some different value of alpha between 0 and 1 we can carry out elastic net regression.

>set.seed(1234)

>En=train(MEDV
~.,train,method='glmnet',tuneGrid=expand.grid(alpha=seq(0,1, length=10),lambda=seq(0.0001,0.2, length=5)),trControl=custom)

+ Fold10.Rep5: alpha=1.0000, lambda=0.2
- Fold10.Rep5: alpha=1.0000, lambda=0.2

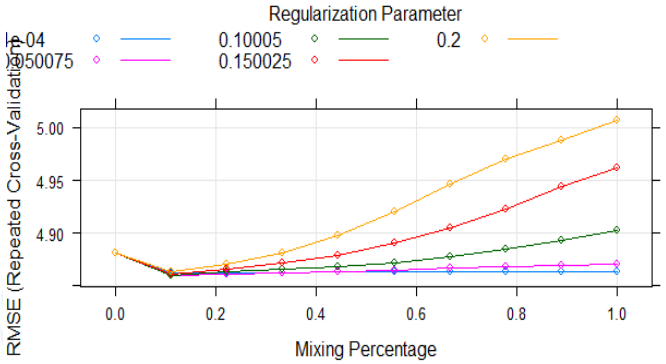
Aggregating results

Selecting tuning parameters

Fitting alpha = 0.111, lambda = 0.1 on full training set

> plot(En)

Figure11 Mixing percentage of alpha



Describes the optimal parameter of alpha and lambda 0001 mixing percentage of alpha between 0 and 1 the RMSQ is high. The best lambda is 0.1

> plot(En$finalModel,xvar='lambda',label=T)

Figure 12 Log Lambda and Coefficient Plot

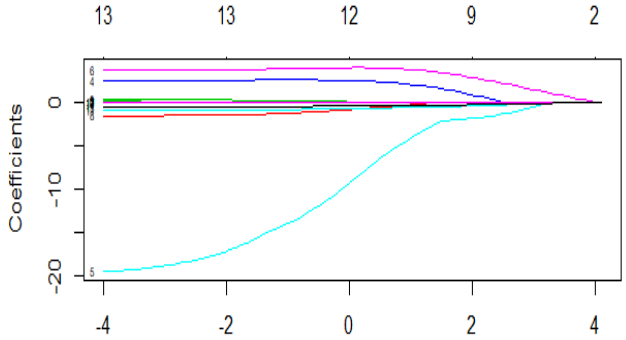
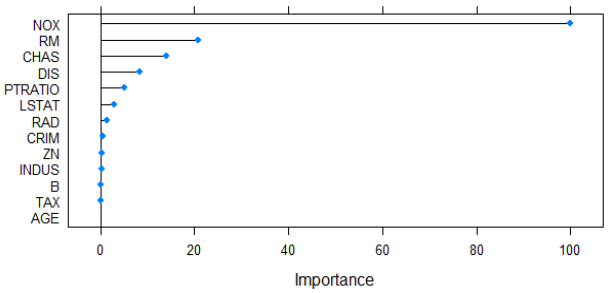


Figure describes the coefficient and log lambda when log lambda is 4 the coefficient is 0 when log lambda reaches at 2 some variables occurs its relationship there were 10 variables in model. When log lambda is 0 then there were few variables added its coefficient the upper line has greater significant then lower one variables.

> plot(En$finalModel,xvar='dev',label=T)

> plot(varImp(En))

Fig 13 Variable Importance



Like ridge regression the lasso has NOX, RM and CHAS variable significant as compared AGE, TAX and B variables.

Compare Models

> Modellist=list(linearModel=lm,Ridge=ridge,Lasso =lasso,ElasticNet=En)

> res=resamples(Modellist)

> summary(res)

Call:

summary.resamples(object = res)

Models: linearModel, Ridge, Lasso, ElasticNet

Number of resamples: 50

MAE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

linearModel 1.497610 2.029862 2.243868 2.238802

Ridge 2.005420 2.722421 2.966608 3.018479

Lasso 2.147462 2.677921 3.016945 3.027462

3.322270

ElasticNet 1.791466 2.756756 2.887369 3.026580

RMSE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

linearModel 1.959050 2.655978 3.093631 3.186282

Ridge 2.516846 3.625268 4.108764 4.245573

4.773495

Lasso 2.588672 3.484634 3.931738 4.213231

4.954492

ElasticNet 2.397171 3.570039 3.835377 4.235350

4.659404

Rsquared

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

linearModel 0.4339949 0.8659259 0.8944073 0.8681266

0.9126632 0.9663609 0

Ridge 0.5142722 0.7406188 0.7964940 0.7791615

0.8544076 0.9077201 0

Lasso 0.5482552 0.7362917 0.7997348 0.7824496

0.8541177 0.9116130 0

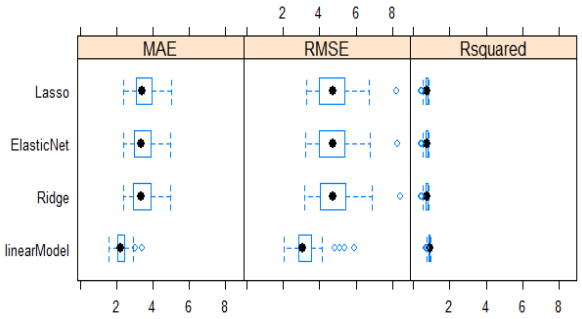
ElasticNet 0.2574850 0.7427887 0.8217334 0.7765588

0.8434995 0.8924685 0

This table describes min , max, median RMSE and maximum values of three regression model, the Root mean square 4.024 is mean value of Elastic net regression and Rsquared describe the variability in the response variables is explained by independent model the highest value of elastic net is 0.78433

> bwplot(res)

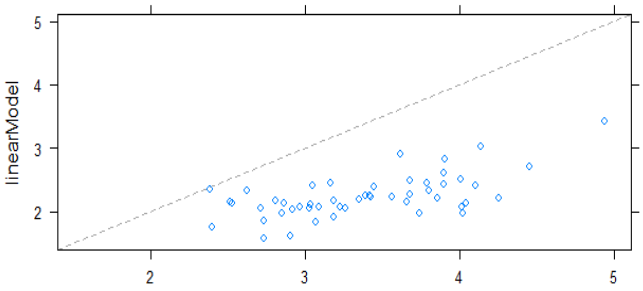
Figure 14 Box Plot



The value sown in three regression model of three model

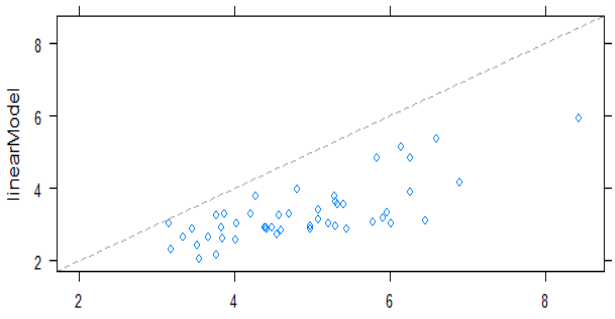
>xyplot(res)

Fig: 15 MAE and Ridge relationship



>xyplot(res,metric='RMSE')

Fig 16 scatter plot between ridge and linear regression model



The dotted line indicates the best when dotted line above means the data are best when linear regression rather than ridge regression. When there were more dots above the model best fitted with linear regression

Best model

> En$bestTune

alpha lambda

8 0.1111111 0.10005

The value of alpha indicates best for ridge as compared lasso model.

> coef(best,s=En$bestTune$lambda)

14 x 1 sparse Matrix of class "dgCMatrix" 1

(Intercept) 23.836156580

CRIM -0.155545721

ZN .042093859

INDUS -0.003887724

CHAS 1.430596788

NOX -11.717596093

RM 5.096463815

AGE -0.027710003

DIS -1.316292494

RAD 0.209120525

TAX -0.011052950

PTRATIO -0.889605615
B 0.008657381
LSTAT -0.334763519
The NOX and RM variables has highest values indicates significant values in the model
Save final model for later use
>saveRDS(En, "Final_model.rds") # saves in computer
Prediction
>fm=En
> p1=predict(fm,train)
> sqrt(mean((train$MEDV-p1)^2))
[1] 4.113352
> p2=predict(fm,test)
> sqrt(mean((test$MEDV-p2)^2))
[1] 6.154483

CONCLUSION

The relationship between the search variables is based on the correlation between them, which is between -1 and 1 and describes the research data of the association. If the higher correlation and the positive value of a riable correspond to the lower value, indicate a relation -ve. Research data could be easily analyzed using R programming instead of mathematical calculation. Cross-validation is a process for evaluating statistical analysis that will be generalized to independent data sets. The prediction problem, a model is usually established in a series of known data data on which the training (set of training data) is performed and a set of unknown data data with respect to which the model adapts. The goal of cross-validation is to measure the model's ability to predict new data that has not been used to estimate it, in order to overcome problems, such as excess data, it will be generalized to an independent data set. Excessive adjustment of a model is a real problem when performing a regression analysis. An excessive regulation mode involves misleading regression coefficients and squared R statistics. To regulate linear or logistic regression models, the elastic network is a method that linearly combines the two penalties of the loop and crest methods. Therefore, the elastic network is always preferable to the loop and crest regression because it solves the limits of both methods, in which we calculate different graphs that show the relationship of the variable variables to reach a conclusion. Therefore, the crest or loop model is as good as the elastic net method according to the good model selection criteria of the model selection process.

REFERENCES

[1] Cirillo, A. (2017). Data mining r for beginners. University of Muntiplier.

[2] Eaton, F. (2018). The collinearity problem in regression discontinuity model. University of Multiplier.

[3] Fiona, G. (2018). Overfitting regression analysis. www.bmj.com.

[4] Ihaka, R. (1996). Mining big data: Current status and forecast to the future.

[5] Jeevan, M. (2018). How i choose the right programming language for data science. Data Science.

[6] Kopf, D. (2017). Which programming should you learn. www.http://qz.com/python.

[7] Lane, D. (2018). Measures of central tendency, variability, introduction to sampling distributions, sampling distribution of the mean, introduction to estimation, degrees of freedom. Introduction to Estimation, Degrees of Freedom.

[8] Michis, S. (2018). Computer world data analysis. Applied Statistics.

[9] Nasridinov, A. (2013). The third international conference on ieee, visual analytics for big data using r. In Cloud and Green Computing (CGC).

[10] Piatetsky, G. (2014). Four machine language. Analytics data-mining.

[11] Rimal, Y. (2018). Cross- validation method for obverting research data using r programming. ICC.

[12] Smith, D. (2018). Revolutionary analytics. Blog, Revolutionary Analytics.