Report

PROBLEM STATEMENT

The Gene Expression Omnibus (GEO) data series GSE4115 contains data from 192 human subjects, each with 22,283 profiled genes. Each subject can have one of three disease states: cancer, no cancer, or suspected cancer. Your task is to build a classifier for cancer vs. no cancer by using HDLSS techniques (such as elastic net).

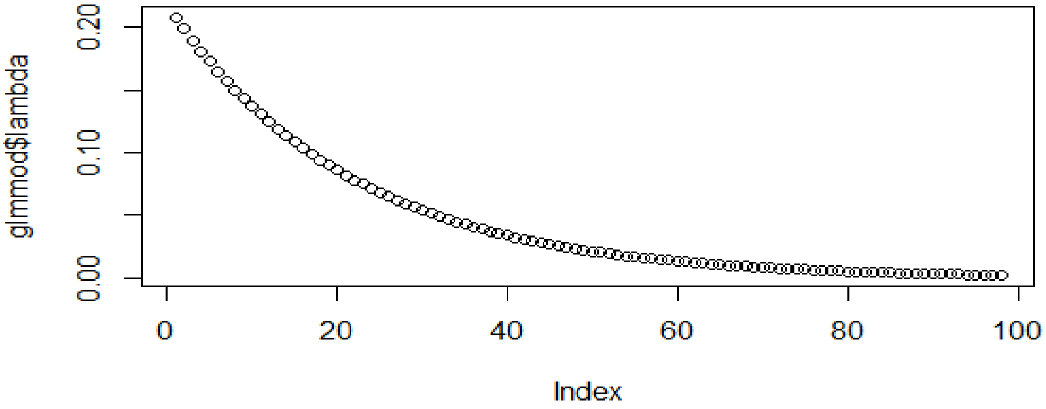
NOTES

For the purpose of classification into 2 classes a.k.a cancer or no cancer, the subjects numbered 188 to 192 are ignored in building the model. Upon analysing the data, it is found that in each of the last 67 features, more than 75% of the information is marked 'not available'. Thus, as a pre-processing step, those features are not considered in building the model. For the first three models discussed below, the penalty is defined as (lambda*((1 – α)*$|\beta|_2$ + α*$|\beta|_1$)) where alpha=1 corresponds to the lasso penalty, alpha=0 to the ridge penalty and the elastic net corresponding to $0 \leq \alpha \leq 1$. The number of lambda values considered for each of the above models is 100, the default value in *glmnet()* package.

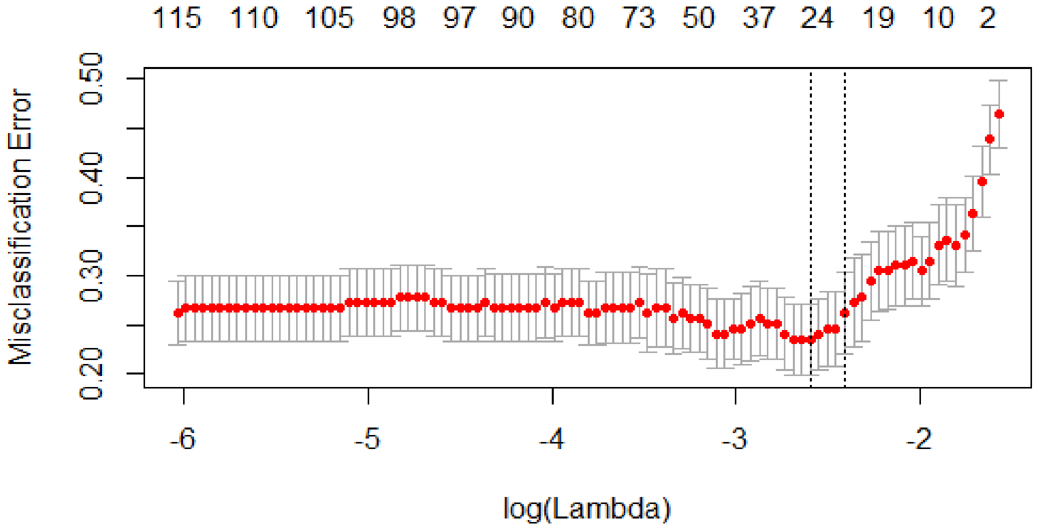
Cross-validation technique is used in building the models and one of the main reasons for using cross-validation instead of using the conventional validation (e.g. partitioning the data set into two sets of 70% for training and 30% for test) is that there is not enough data available to partition it into separate training and test sets without losing significant modelling or testing capability. Thus, in this case, a fair way to properly estimate model prediction performance is to use cross-validation as a powerful general technique. In summary, cross-validation combines (averages) measures of fit (prediction error) to derive a more accurate estimate of model prediction performance.

LASSO:

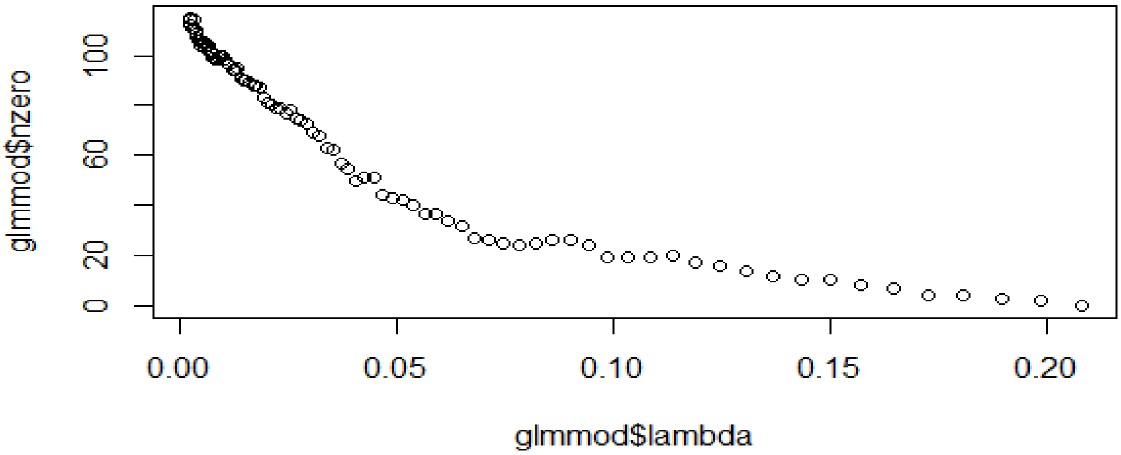
Lasso penalty corresponds to the value of alpha=1. Figure below shows the lambda sequence chosen to find the best hyper-parameter.



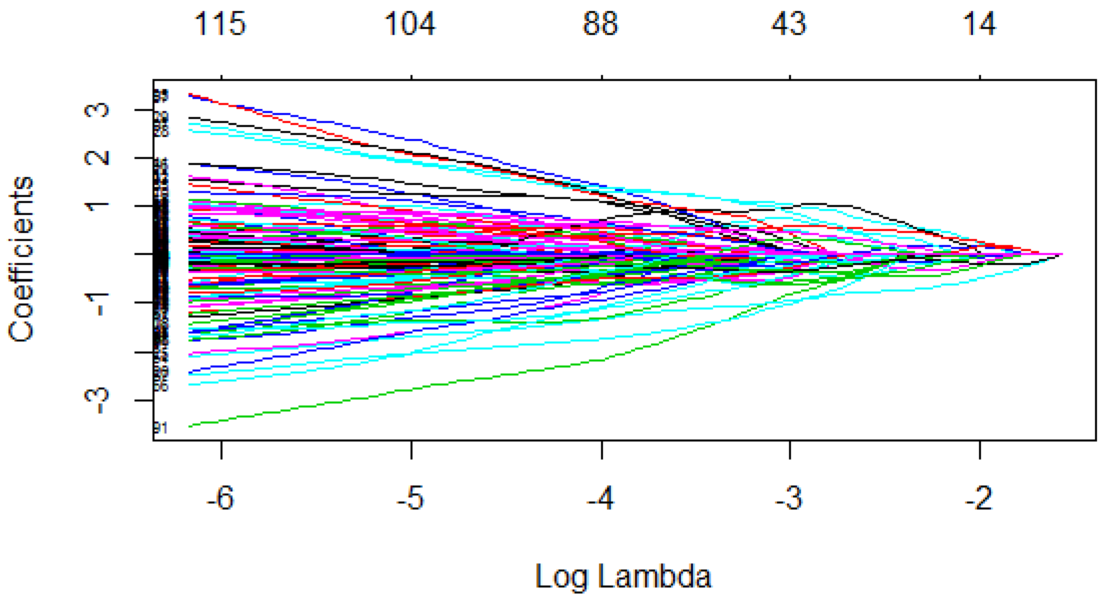
As it can be seen, the range starts from ~0.21 and decreases to ~0.01 in a non-linear fashion.
Figure below shows the misclassification error for each value of lambda. A 10-fold cross-validation is done and the value marked in red (in fig) is the average misclassification error over the 10 iterations for each lambda. The lines above and below these points determine the variance of the error over each iteration (for each lambda). For lambda=0.07464203, the minimum mean cross-validated error is obtained and is equal to 0.2352941.



Since LASSO penalization is L1-norm penalization and returns a sparse coefficient-vector, the no. of non-zero coefficients for at each value of lambda are evaluated and plotted. At lambda stated above, the number of non-zero coefficients is 26. It can be seen that with increase in lambda, the penalty increases and thus the number of decreases. This effect can be seen in the figure below.



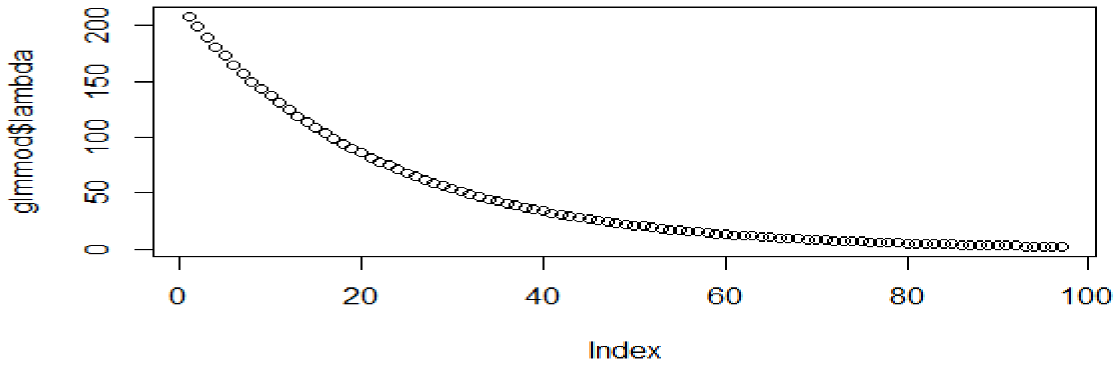
Adding to the above point, below is the figure showing the shrinkage of different coefficients as the penalty is increased. It can also be seen that as penalty increased, more and more coefficients become zero. It can be seen that for any particular coefficient, with increase in penalty, its magnitude decreases.



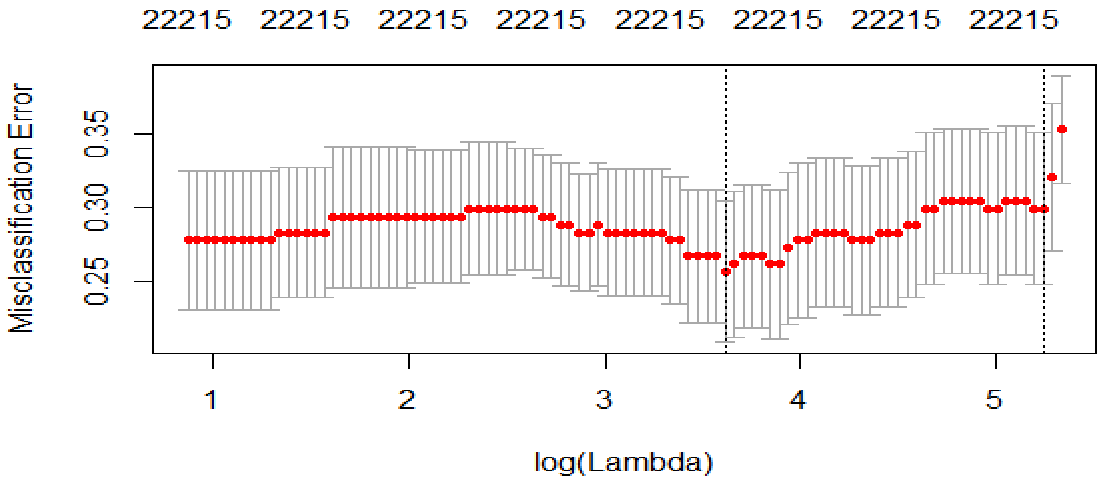
Thus the hyper-parameter lambda is chosen as that lambda for which the avg. cross-validation error is minimum. The validation accuracy for the chosen hyper-parameters is **76.47%** and the top 25 coefficients in the descending order of absolute magnitudes are: **ADAMTSL2 RAB1A MIR4640 USP12 SCRN3 POLR3B SH2D3A AK025072 ZKSCAN5 SGSM3 AGBL2 ASPH RRP7A KCTD17 MIR6875 ARSD HOXB6 DICER1 MBTPS2 CD52 EML2 PNN NDUFA7 SEC14L5 ARHGAP5.** The corresponding indices are 6155 12821 1 12708 18598 18823 21529 14782 3258 2543 19754 6809 2466 5088 13641 7305 4892 12611 5999 21802 3926 11422 2313 9653 17301. The indices are with reference to **Gene Expression Omnibus (GEO) data series GSE4115 with "MIR4640" being the index 1.** The higher the absolute magnitude of a coefficient, the higher its importance in the technique.

RIDGE:

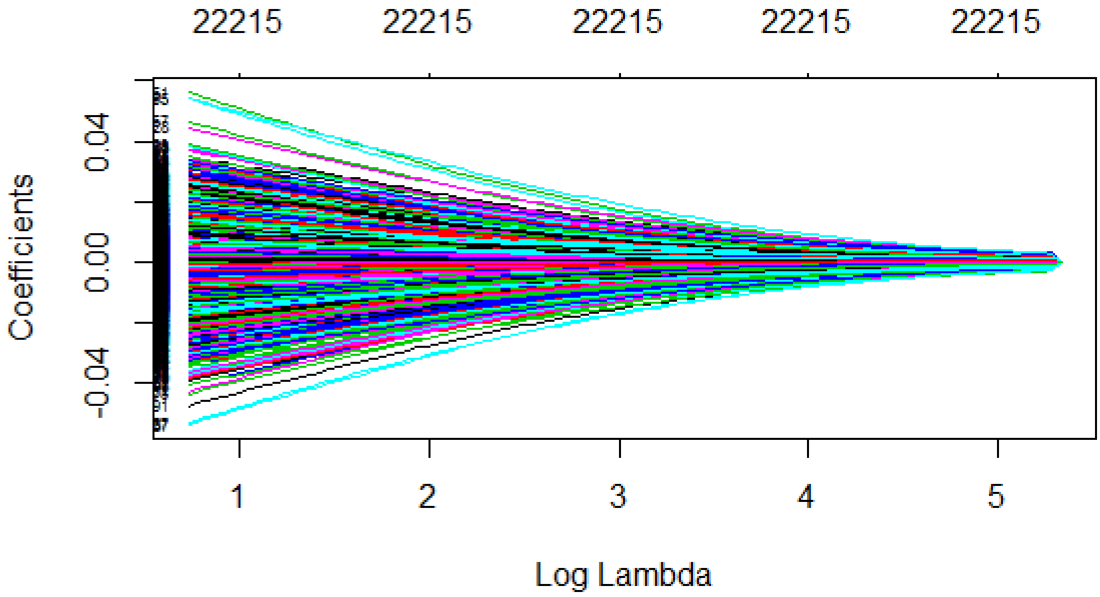
Ridge penalty corresponds to the value of alpha=0. Figures below show the lambda sequence chosen to find the best hyper-parameter and the mean and variance in the classification error for each value of lambda chosen for a 10-fold cross-validation. As it can be seen, penalizing too high or too low will not result in minimum average misclassification error. The minimum occurred for lambda=37.14951 with the average error at that value being 0.2566845.



glmmod$lambda
0
50
100
150
200
0
20
40
60
80
100
Index



22215 22215 22215 22215 22215 22215 22215 22215
Misclassification Error
0.25
0.30
0.35
1
2
3
4
5
log(Lambda)



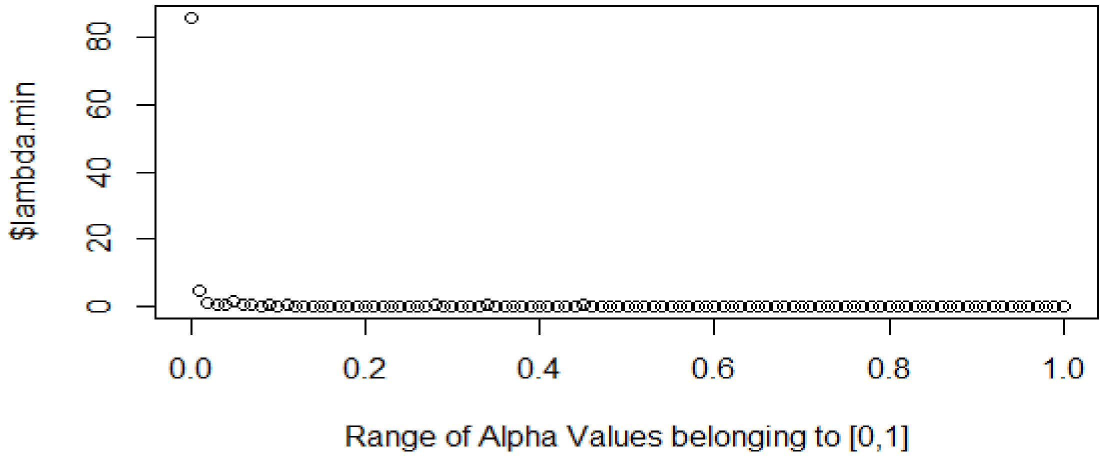
22215 22215 22215 22215 22215
Coefficients
-0.04
0.00
0.04
1
2
3
4
5
Log Lambda

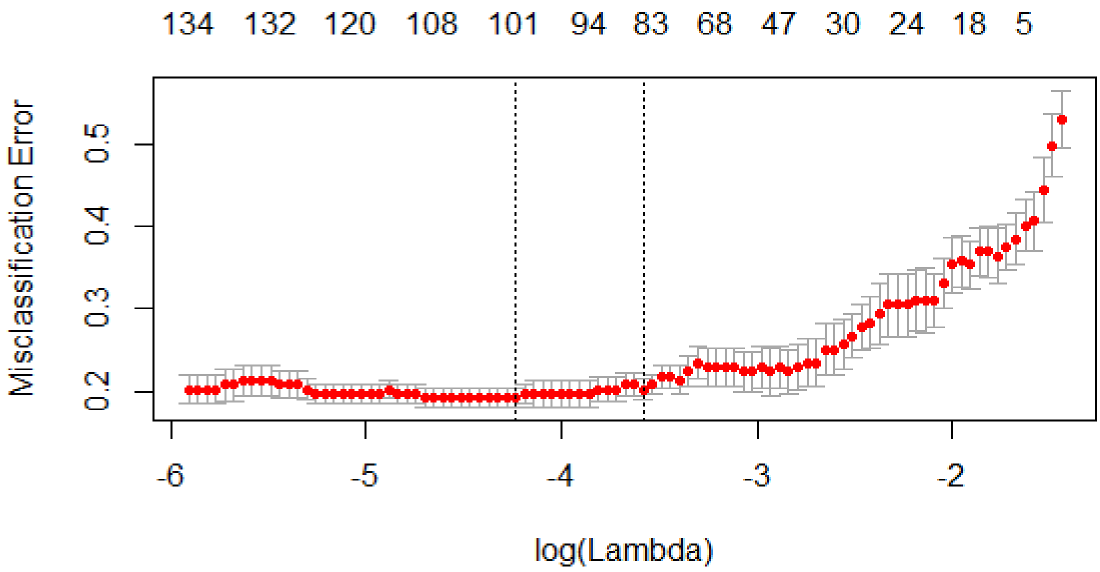
The figure above shows the shrinkage of coefficients with increase in penalty. A clear distinction can be seen the way the coefficients shrink in LASSO and in RIDGE.

Thus, the hyper-parameter lambda is chosen as that lambda for which the avg. cross-validation error is minimum. The validation accuracy for the chosen hyper-parameters is **74.34%** and the top 25 Coefficient s in the descending order of absolute magnitudes are: **C2orf68 DTX3 ZNF142 KIAA0586 CD52 SH2D3A UGT2B15 AL050026 216374_at LYPLA2 HNRNPM ACKR3 CENPB PNN RNF24 CD209 UGT1A1 EXOG MBTPS2 PTPA KCTD17 HOXB6 KIF20B CDK9 MIR4640.** The corresponding indices are: 22176 22049 21888 21874 21802 21529 16057 15996 15745 14941 14293 12360 11822 11422 10168 6802 6652 6429 5999 5978 5088 4892 4762 2726 and 1. The reference marking for these indices is stated in LASSO Se ction. The higher the absolute magnitude of a coefficient, the higher its importance in the technique.

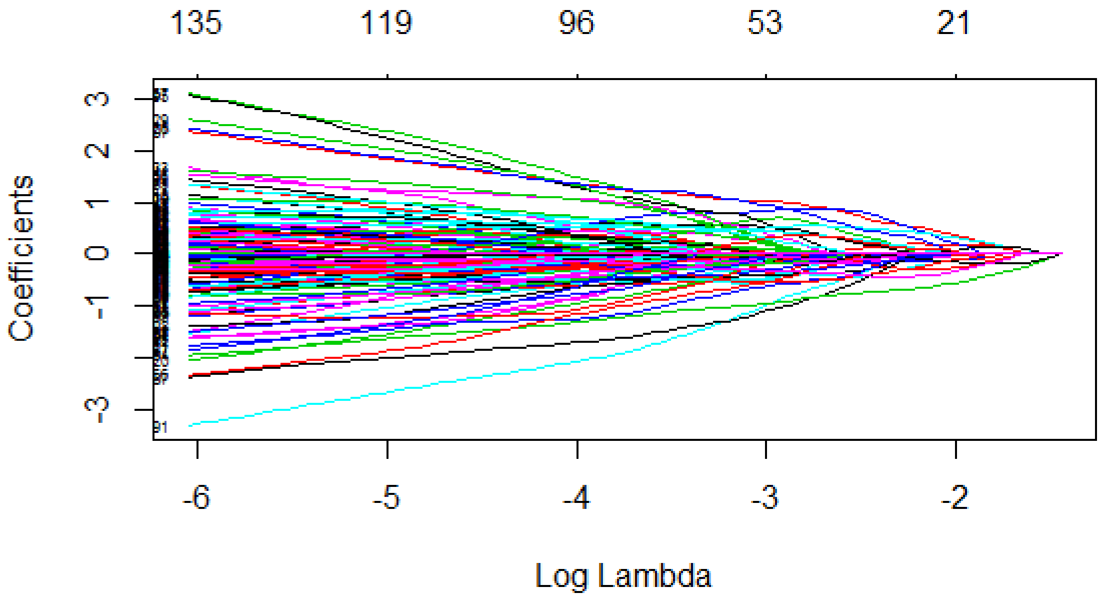
ELASTIC NET

To choose the best set of (alpha, lambda), a sequence of alpha from 0 to 1 is chosen with a step size of 0.01. At each value of alpha, a model is obtained with a corresponding lambda for which the cross-validation error is minimum. In the figure, for different values of alpha from [0,1], the corresponding lambda for min error is plotted.





Corresponding to alpha=0.88, the least average misclassification error is obtained and is equal to 0.19251337. This value is obtained at lambda=0.01448185. Thus, the set of hyper-parameters chosen are (0.88, 0.01448185). For alpha=0.88, the figure above depicts the variation in avg. misclassification error. The figure below depicts the different coefficients and their shrinkage with increase in lambda value for alpha=0.88. It is noteworthy that elastic net penalty is a combination of L1 and L2 norm.



For the (alpha, lambda) set obtained above, the cross-validation accuracy is **80.75%** and the top 25 Coefficients in the descending order of absolute magnitudes are: **MIR4640 HOXB6 KCTD17 SH2D3A USP12 ADAMTSL2 RNF24 PNN AL050026 MS4A3 KLHL18 MEAF6 IL1RN HSPB11 RAB1A ATP2C1 LOC101929219 EML2 HIVEP1 AGBL2 GGA1 CDK9 PGAP3 MBTPS2 TLE4** and the corresponding indices are 1 4892 5088 21529 12708 6155 10168 11422 15996 9734 12267 17530 12042 13542 12821 9421 21640 3926 4039 19754 16925 2726 21171 5999 4399. The reference marking for these indices is stated in LASSO Section.

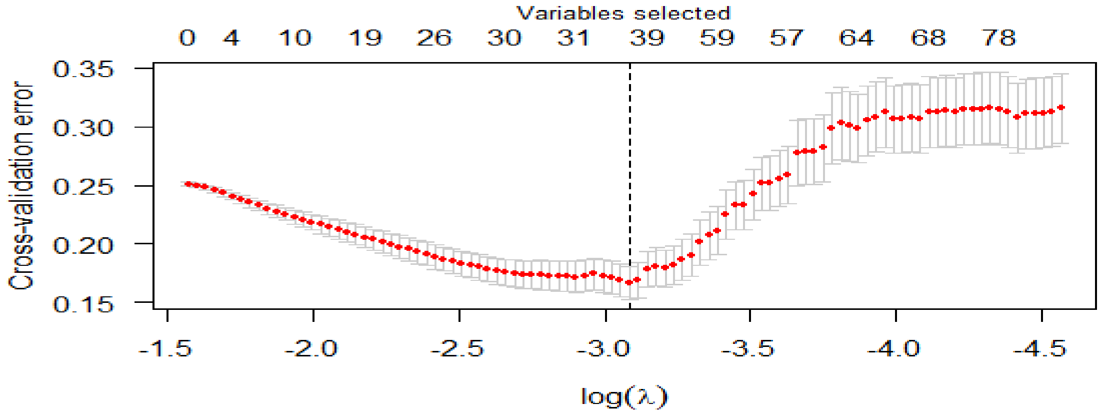
NOTES and INFERENCES

Although different genes are of importance in different techniques discussed above, few genes are consistent across the models. Some examples include the genes **MIR4640, SH2D3A, MBTPS2 and PNN** (indices: 1, 21529, 5999, 11422) which are found important in the three models. Also, some genes like **ADAMTSL2,RAB1A,USP12 and FAM110D** (indices:6155, 12821, 12708,19794) are found important in two of the three models.

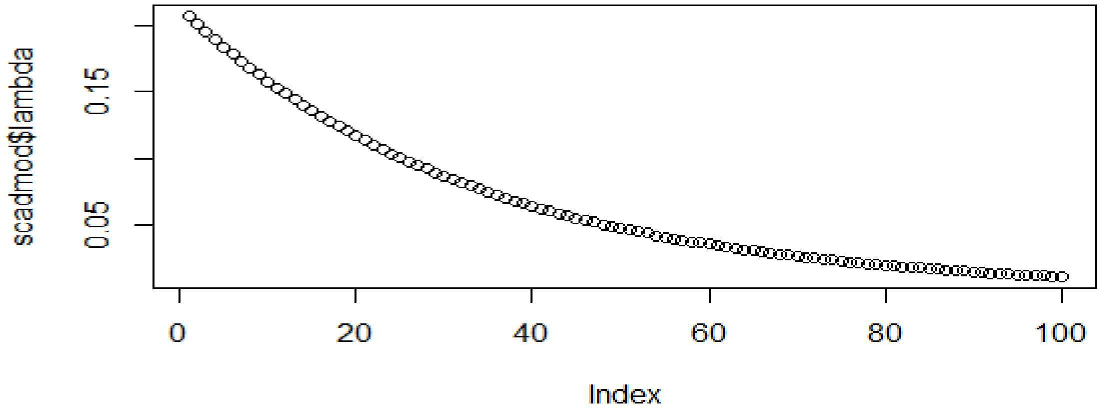
Also, it can be inferred from the trend in the coefficient magnitudes (from above) that ridge penalty is one possible way to deal with *multicollinearity* problem that arises when many predictors are highly correlated. Introducing ridge penalty effectively lowers these correlations. Also, a similarity between Ridge regression and Principal Component regression can be seen. The connection is that in PCR you effectively have a "step penalty" cutting off all the eigenvalues after a certain number, whereas in ridge regression, you apply a "soft penalty", penalizing all eigenvalues, with smaller ones getting penalized more.

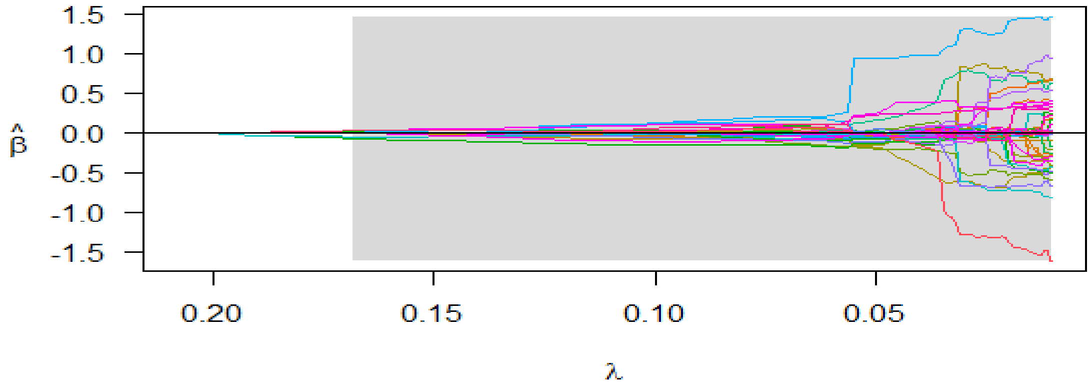
The next two models are SCAD and MCP penalties. Unlike the ones discussed above, these two penalized methods are non-convex penalties.

SCAD:



For selection of the best hyper-parameter set, a sequence of lambda is again chosen and the average misclassification error for each lambda is obtained. The figure below shows the different lambda values chosen and the figure above shows the corresponding misclassification errors with 10-fold cross-validation.



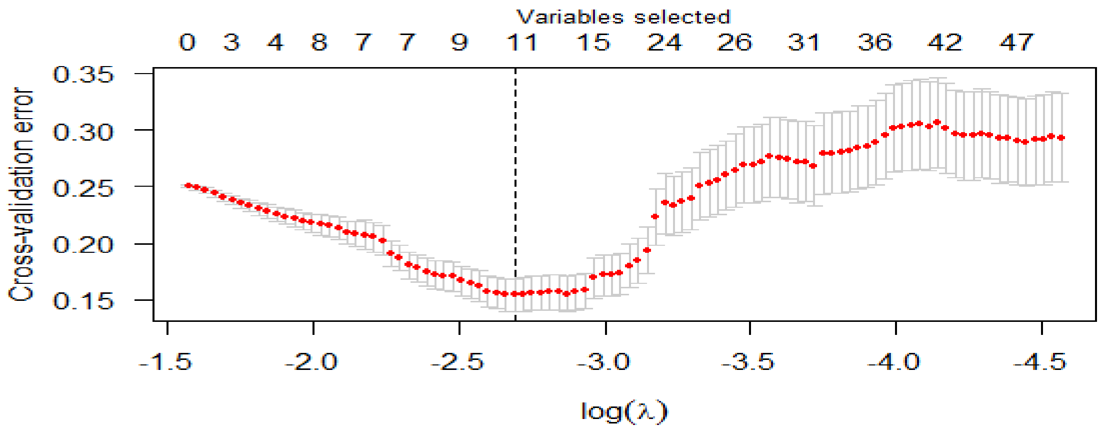


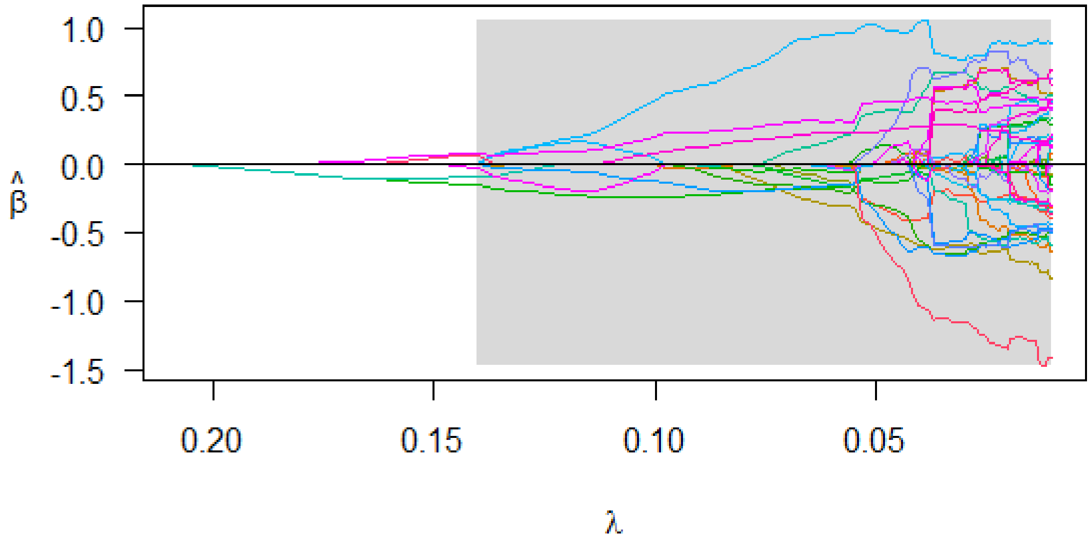
Corresponding to lambda=0.04574483, the minimum average misclassification error is 0.1664612. In the figure below, the variation of coefficients with change in penalty is plotted. Thus, for the hyper-parameters chosen the validation accuracy is **83.36%.**

MCP:

The LASSO is fast and continuous, but biased. The bias of the LASSO may prevent consistent variable selection. Subset selection is unbiased but computationally costly. The MC+ has two elements: a Minimax Concave Penalty (MCP) and a penalized linear unbiased selection (PLUS) algorithm.

The hyper parameters are selected in the same way as above. Corresponding to lambda =0.06779292, the minimum average misclassification error is obtained as 0.1542884.





Thus, for the hyper-parameters chosen the validation accuracy is **84.5%**.

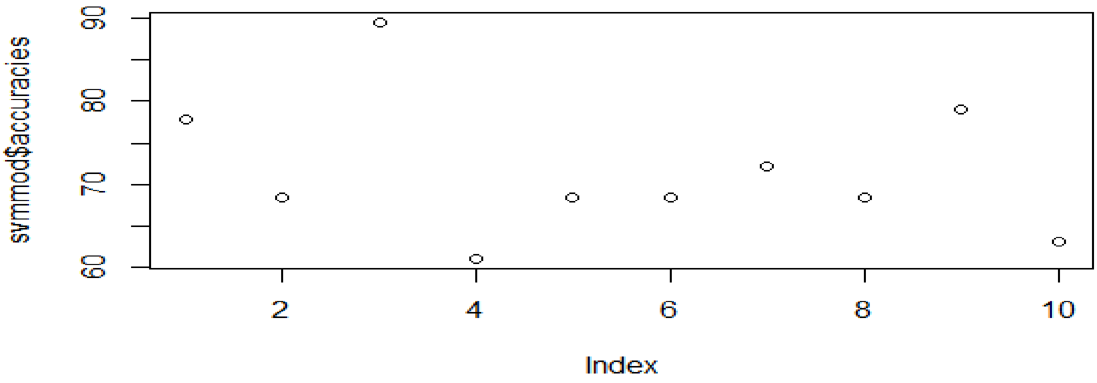
NOTES

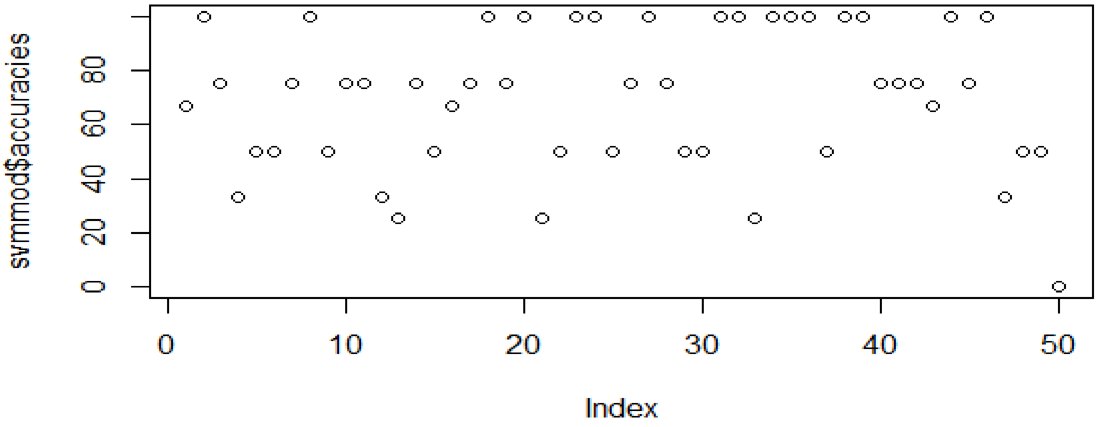
Next is SVM Classifier with k-fold cross validation and supervised Random Forest with details into the Confusion matrix obtained.

SVM:

The accuracies corresponding to each fold in 10-fold cross-validation and 50-fold cross-validation are shown in the figures below. It is seen that with increase in no. of folds, the average accuracy falls down with an increase in the accuracy variation.

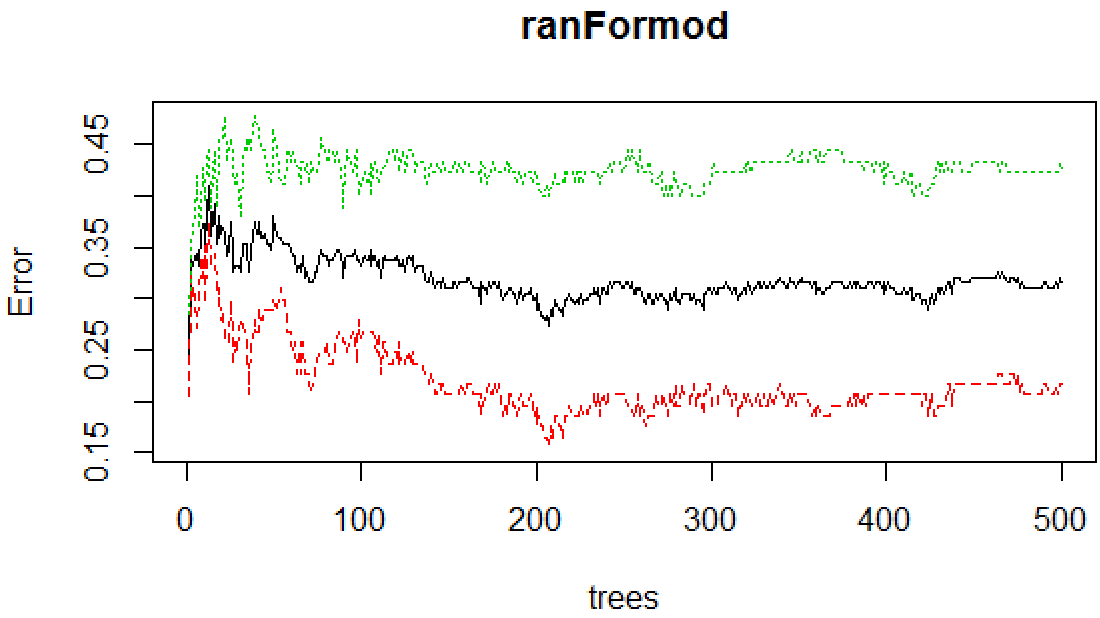
Average Accuracy for 10-fold cross-validation: 71.6 %
Average Accuracy for 50-fold cross-validation: 69.7 %





For more details on SVM, please refer to the code.

RANDOM FOREST:



The Class Error for Class 1 and 2 are(Confusion Matrix):

```
 [1] [2]  class.error
[1] 76 21   0.2164948
[2] 38 52   0.4222222
```

The top 25 important genes are:
SLC5A1 EEF1A1 217713_x_at C1QTNF3-AMACR ZNF473 ATP8B1 PTMA RABAC1 C6 P4HB ADH6 TNPO3 CD36 ARGLU1 RUNX1-IT1 210679_x_at METTL7A PRKCA NUDT4P1 CYB561 RPL23AP32 220856_x_at NFATC2IP PTGES CYR61

For more details on random Forest Model, please refer to the code.