Lab 4: Regression in Biomedical Research

2022-04-12

Warm Up

We load the dataset using function “readDBS”.

```
data <- readRDS("/Users/kevinli/Downloads/Trp63.tf.rds")
```

As a background on the dataset, this is a relatively large dataset with 2177 observations and 226 variables. Each observation (each row) is a cell, whereas each variable (each column) represents a gene. In particular, the first column, Trp63, represents a very important gene for development. The rest 225 genes are transcription factors, which are thought to be the regulators for Trp63.Then, each entry Eij of the data matrix stands for the gene expression level of gene j for cell i.

The target here is the Trp63 gene. The regressors are the rest 225 genes, which we will use as features to predict our target.

Part 1: LASSO Regression

Next, we will use cross-validation to choose the best penalty parameter lambda.

```
#Load necesssary packages
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-3
```

```
library(DBI)
model <- cv.glmnet(as.matrix(data[,2:226]), data$Trp63)
lambda = model$lambda.1se # the value of lambda used by default
print(lambda)
```

```
## [1] 0.02353224
```

We will use our best lambda here to fit a LASSO model. We will print the regression results and calculate the MSE of the model.

```
##
## Call:  glmnet(x = as.matrix(data), y = data$Trp63, lambda = 0.023532)
##
##   Df  %Dev  Lambda
## 1  1 99.79 0.02353
```

```
## [1] 0.000553755
```

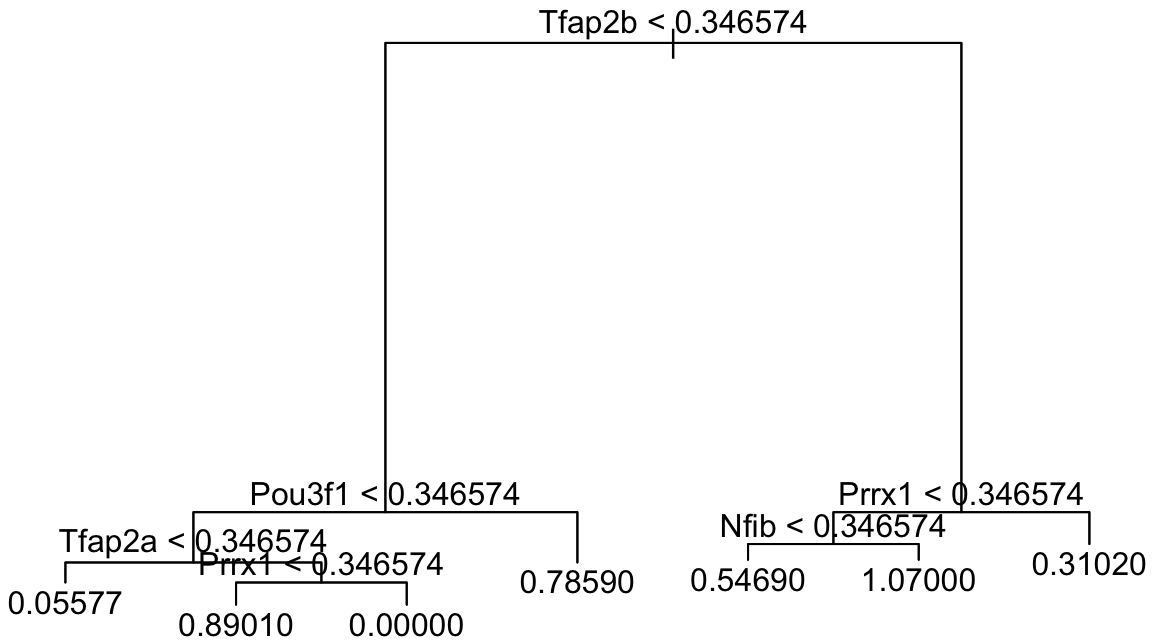
Part 2: Decision Trees

We use the 'tree' function to fit a regression tree to the Trp63.tf dataset. We plot the tree with the 'plot' and 'text' functions.

```
library(tree)
regression <- tree(data)
print(regression) #How do I see which genes are selected as important decision nodes?
```

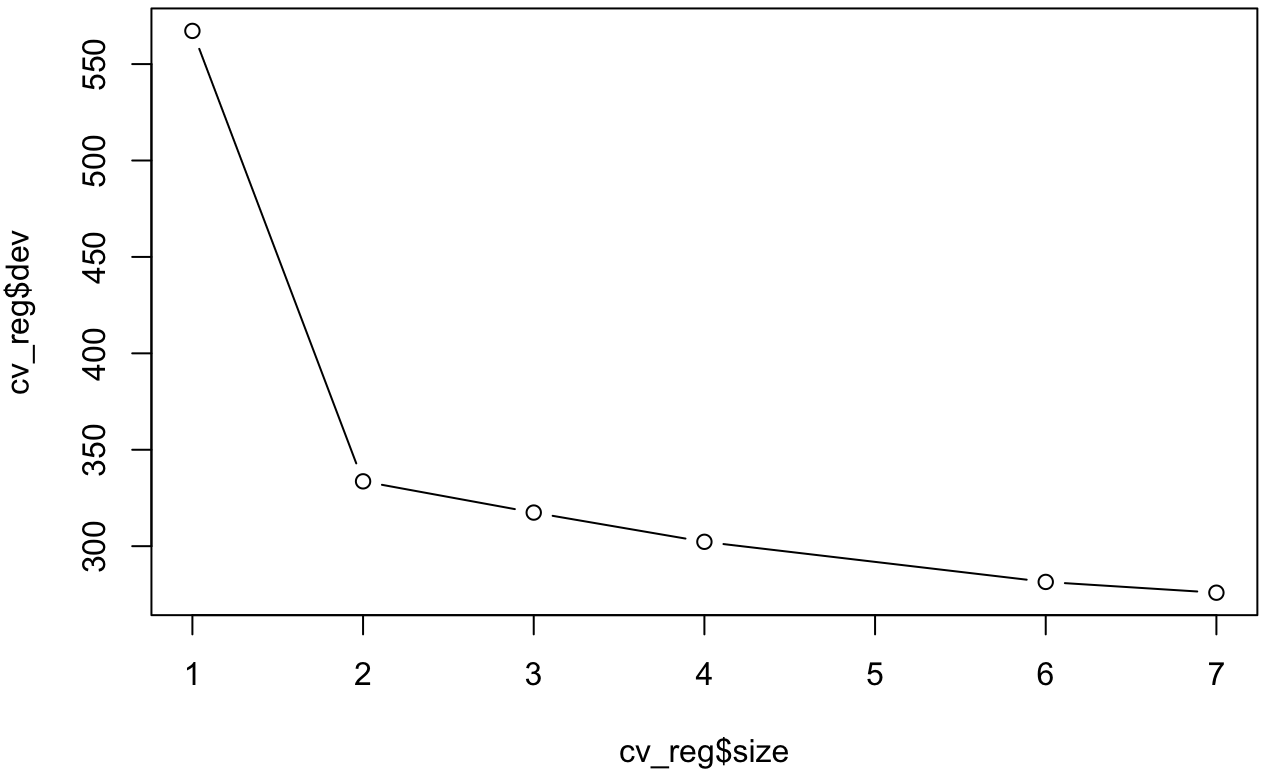
```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 2177 566.90 0.24260
##    2) Tfap2b < 0.346574 1794 176.10 0.09014
##      4) Pou3f1 < 0.346574 1743 124.90 0.06978
##        8) Tfap2a < 0.346574 1686  88.41 0.05577 *
##        9) Tfap2a > 0.346574 57  26.39 0.48410
##         18) Prrx1 < 0.346574 31  15.19 0.89010 *
##         19) Prrx1 > 0.346574 26   0.00 0.00000 *
##      5) Pou3f1 > 0.346574 51  25.74 0.78590 *
##    3) Tfap2b > 0.346574 383 153.80 0.95670
##      6) Prrx1 < 0.346574 348 127.40 1.02200
##       12) Nfib < 0.346574 32  14.02 0.54690 *
##       13) Nfib > 0.346574 316 105.40 1.07000 *
##      7) Prrx1 > 0.346574 35  10.30 0.31020 *
```

```
plot(regression)
text(regression)
```



Next, we first use the 'cv.tree' function to conduct cross validation on the data. The best parameter determined from the cross validation will be used when pruning the applied.

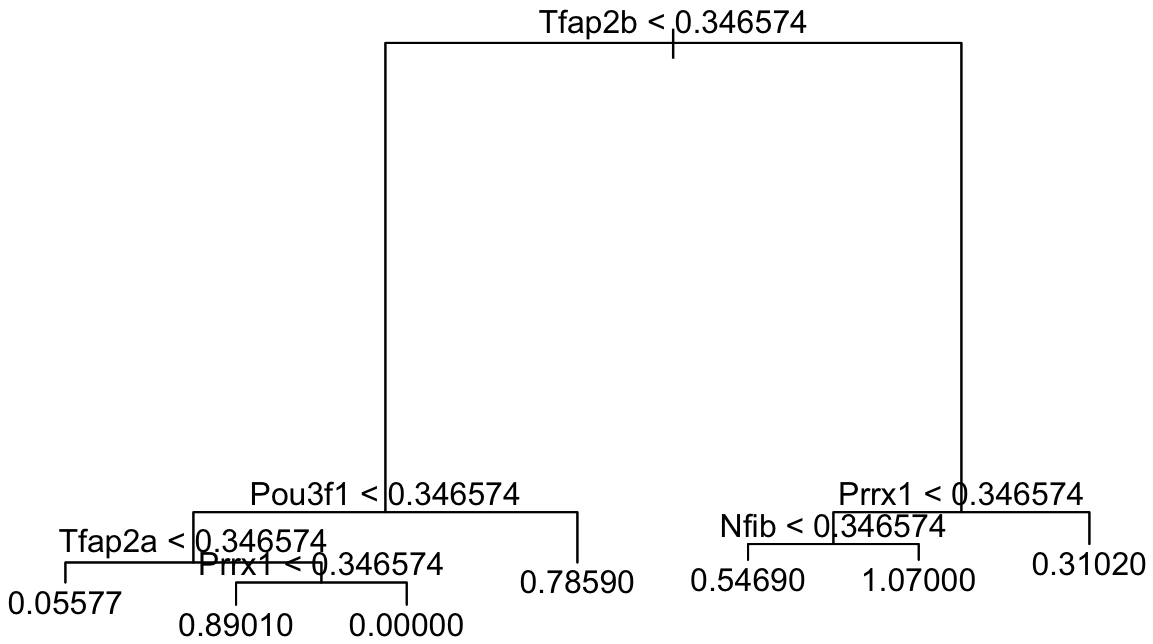
```
cv_reg <- cv.tree(regression)
plot(cv_reg$size, cv_reg$dev, type = "b")
```




```
pruned_reg <- prune.tree(regression, best = 7)
print(pruned_reg)
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 2177 566.90 0.24260
##    2) Tfap2b < 0.346574 1794 176.10 0.09014
##      4) Pou3f1 < 0.346574 1743 124.90 0.06978
##        8) Tfap2a < 0.346574 1686  88.41 0.05577 *
##        9) Tfap2a > 0.346574 57  26.39 0.48410
##         18) Prrx1 < 0.346574 31  15.19 0.89010 *
##         19) Prrx1 > 0.346574 26   0.00 0.00000 *
##      5) Pou3f1 > 0.346574 51  25.74 0.78590 *
##    3) Tfap2b > 0.346574 383 153.80 0.95670
##      6) Prrx1 < 0.346574 348 127.40 1.02200
##       12) Nfib < 0.346574 32  14.02 0.54690 *
##       13) Nfib > 0.346574 316 105.40 1.07000 *
##      7) Prrx1 > 0.346574 35  10.30 0.31020 *
```

```
plot(pruned_reg)
text(pruned_reg)
```



Lastly, we try to grow a random forest and calculate the MSE for the training and testing dataset.

```
library(randomForest)
```

```
## randomForest 4.7-1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
train=sample(1:nrow(data),1500)
data_test=data[-train, ]
rf_train <- randomForest(formula = Trp63 ~ ., data = data, subset = train)
rf_test <- randomForest(formula = Trp63 ~ ., data = data_test)

importance(rf_test)
```

```
##          IncNodePurity
## Msc       3.343332e-03
## Tfap2b    4.512149e+01
## Pou3f3    1.238742e-02
## Stat1     5.199434e-02
## Creb1     3.015526e-01
## Elk4      6.995798e-01
## Sox13     2.562936e-02
## Myog      6.006307e-17
## Elf3      2.410059e-02
## Prrx1     1.658895e+01
## Pou2f1    5.335715e-01
## Rxrg      3.173271e-04
## Lmx1a     5.784262e-17
## Pbx1      5.452621e-01
## Atf6      9.688999e-02
## Batf3     2.976202e-01
## Bmyc      2.672984e-01
## Rxra      7.213781e-01
## Lmx1b     3.431103e-01
## Lhx6      6.308587e-02
## Zbtb26    7.857557e-02
## Nr4a2     4.290419e-01
## Tbr1      9.903649e-02
## Sp3       3.725851e-01
## Atf2      3.678267e-01
## Hoxd9     6.122525e-03
## Hoxd8     1.253360e-01
## Spi1      2.189112e-02
## Ehf       2.209344e-17
## Meis2     4.988595e-01
## Mga       4.470894e-01
## Zscan29   2.098129e-01
## Zfp341    1.452757e-01
## Tgif2     1.310234e+00
## Snai1     3.169881e-01
## Cebpb     6.620244e-02
## Nfatc2    2.699834e-02
## Tfap2c    5.032396e+00
## Gata1     3.871998e-03
## Zfp449    2.413288e-01
## Zic3      3.622634e-02
## Foxo4     6.891194e-01
## Pou3f4    2.722567e-03
## Sox2      1.304169e-02
## Creb3l4   9.054331e-03
## Rfx5      4.345401e-02
## Arnt      1.077725e-01
## Tbx15     6.308467e-02
## Alx3      1.169842e-02
## Lef1      1.142442e+00
## Plag1     2.797774e-01
```

```
## Bach2    6.134959e-02
## Creb3    7.749746e-01
## Msantd3  2.450603e-01
## Klf4     9.059618e-01
## Nfib     3.560903e+00
## Jun      1.668751e+00
## Glis1    6.899936e-02
## Foxd2    1.776357e-18
## Foxo6    1.003226e-01
## Pou3f1   5.399666e+00
## Mtf1     3.053527e-01
## Tfap2e   3.457083e-01
## Trp73    2.322451e+00
## Fosl2    7.465904e-01
## Gfi1     8.434415e-03
## Tbx3     2.300768e-03
## Mafk     9.349895e-02
## Foxk1    9.859765e-02
## Zkscan5  6.680656e-02
## Dlx5     3.529114e-01
## Foxp2    2.168807e-02
## Zfp282   2.637147e-01
## Hoxa1    2.428150e-01
## Hoxa4    5.903674e-02
## Hoxa7    3.649808e+00
## Hoxa9    1.942764e-02
## Hoxa10   4.018326e-03
## Atoh1    4.632494e-02
## Tcf7l1   1.184488e-01
## Vax2     1.223052e-02
## Egr4     1.598721e-17
## Nr2c2    2.803808e-01
## Zfp384   1.148784e-01
## Tead4    1.506319e-01
## Etv6     4.861876e-02
## Zfp110   4.003344e-01
## Mzf1     8.536202e-03
## Meis3    1.039188e-01
## Fosb     4.674506e-01
## Relb     3.436656e-01
## Pou2f2   2.296868e-03
## Cebpg    4.285684e-01
## Cebpa    5.281643e-01
## Spib     1.026669e-02
## Nr1h2    4.629923e-01
## Irf3     7.494087e-01
## Tead2    1.252504e+00
## Dbp      3.059738e-01
## Myod1    4.440892e-19
## E2f8     4.797334e-01
## Nr2f2    5.631696e-01
## Tead1    3.436609e-01
```

```
## Maz      5.848944e-01
## Ebf3     9.607081e-02
## Irf7     2.705092e-02
## Esr1     2.739104e-02
## Foxo3    2.279246e-01
## Prdm1    2.609584e-01
## Zbtb7a   1.087200e+00
## Nfyb     7.660352e-01
## Prdm4    6.201281e-02
## Ascl1    6.284045e-03
## Nr2c1    6.286844e-02
## Stat2    1.436418e-01
## Irf2     2.693134e-01
## Jund     1.790561e+00
## Klf2     3.367149e-01
## Rfx1     4.185469e-02
## Nfix     2.814067e-01
## Junb     1.694759e+00
## Nfatc3   1.866114e-01
## Maf      2.199154e-01
## Irf8     8.428909e-03
## Foxc2    1.643130e-17
## Thrb     4.817201e-02
## Irf9     8.436529e-02
## Nfatc4   1.529074e-01
## Elf1     3.831187e+00
## Klf5     1.225199e+01
## Sox21    1.356159e+00
## Zfp317   4.597154e-02
## Rora     1.447058e-01
## Rfx7     2.741162e-01
## Zic1     1.825061e-01
## Ikzf1    2.441967e-02
## Meis1    2.604002e-01
## Rel      9.690174e-02
## Ebf1     2.099085e+00
## Tcf7     2.079521e-01
## Irf1     9.791099e-02
## Trp53    2.865034e+00
## Bcl6b    2.442491e-18
## Tbx2     1.521006e-17
## Vezf1    3.011890e-01
## Hlf      1.651344e-02
## Zfp652   2.378974e-01
## Hoxb9    1.394688e-01
## Hoxb8    2.686891e+00
## Hoxb7    1.911408e+00
## Hoxb6    4.244969e-01
## Hoxb5    1.366526e-01
## Hoxb3    4.223454e-01
## Nfe2l1   5.006861e-01
## Sp2      1.164598e-01
```

```
## Rara     2.631919e-01
## Stat3    7.400343e-01
## Etv4     3.037688e-01
## Meox1    3.560169e-02
## Sox9     5.511247e-01
## Foxk2    2.422107e-01
## Klf6     4.566158e-01
## E2f3     1.598337e-01
## Foxc1    2.737691e-01
## Rreb1    7.312539e-01
## Tfap2a   1.325199e+01
## Nfil3    1.981028e-01
## Zfp369   7.042377e-01
## Nr2f1    1.026212e-02
## Foxd1    4.590444e-02
## Osr1     2.614979e-01
## Grhl1    4.794892e-01
## Meox2    1.965578e-01
## Etv1     3.486204e-02
## Hif1a    6.374744e-01
## Max      4.838749e-01
## Zfp410   5.391273e-01
## Fos      4.266927e+00
## Batf     9.534852e-02
## Foxn3    9.024044e-01
## Gsc      3.838197e-01
## Yy1      6.531496e-01
## Osr2     2.870543e-01
## Grhl2    2.734867e+00
## Klf10    2.505473e-01
## Myc      4.240150e-01
## Hsf1     3.194221e-01
## Atf4     1.025050e+00
## Tef      1.737318e-01
## Vdr      1.932445e-01
## Pou6f1   1.156666e-01
## Nr4a1    1.105215e+00
## Zfp740   4.738740e-01
## Hoxc10   6.961358e-02
## Hoxc9    3.315988e-01
## Hoxc8    7.265665e-01
## Hoxc4    4.253503e-01
## Nfe2     1.927772e-02
## Tfap4    3.717639e-01
## Glis2    7.634561e-02
## Hic2     2.326999e-01
## Etv5     8.214583e-02
## Zfp148   5.181807e-01
## Gabpa    2.067801e-01
## Bach1    1.547843e-01
## Sox8     7.731366e-02
## Tead3    4.527104e-01
```

```
## Rxrb     2.308279e-01
## Pbx2     3.020161e-01
## Pou5f1   1.084317e-02
## Runx2    4.271851e-02
## Tgif1    6.909811e-01
## Zeb1     6.806312e-02
## Zfp24    6.139233e-02
## Pou4f3   4.042529e-02
## Fosl1    2.092730e-02
## Rela     2.544873e-01
## Esrra    1.532984e-01
## Rorb     1.091596e-02
## Klf9     2.668569e-01
## Glis3    2.224724e-02
## Nfkb2    1.188139e-01
## Emx2     3.057708e-01
## Pitx2    1.568922e-01
## Foxp1    1.468289e+00
```