Chapter 4. Linear regression Model with penalty or constraints (Part 3)

1 The logistic regression model

Suppose the response only takes two values or in two categories, obviously they can be denoted by $Y = 0$ and $Y = 1$ (or -1 and 1). For each X, its response is random and follows the **binomial distribution** with probability

$$P(Y = 1|X) = p(X, \tilde{\beta}) \quad \text{and} \quad P(Y = 0|X) = 1 - p(X, \tilde{\beta})$$

and

$$Y \sim B(p(X, \tilde{\beta}))$$

The problem is to model

$$p(X, \tilde{\beta}) = P(Y = 1|X) = E(Y|X) = ???$$

Note that it is not convenient to use a linear function of X to model $E(Y|X) = P(Y = 1|X)$ (Why?). Instead, it is popular to model it by

$$\log(\frac{p}{1-p}) = a + \beta_1 \mathbf{x}_1 + ... + \beta_p \mathbf{x}_p$$

which is equivalent to

$$P(Y = 1|X) = p(X, \tilde{\beta}) = \frac{\exp(a + \beta^\top X)}{1 + \exp(a + \beta^\top X)}$$

This model is also called the **logistic model**, with parameters

$$\tilde{\beta} = (a, \beta) = (a, \beta_1, ..., \beta_p).$$

Under this model, for a sample with predictors $X_i = (\mathbf{x}_{i1}, ..., \mathbf{x}_{ip})^\top$ the response Y_i, where $Y_i = 1$ when it is from group 1, and $Y_i = 0$ otherwise. we have

$$P(Y = 1|X = X_i) = p(X_i, \tilde{\beta})$$

and

$$P(Y = 0|X = X_i) = 1 - p(X_i, \tilde{\beta})$$

which can be unified as

$$P(Y = Y_i|X = X_i) = \{p(X_i, \tilde{\beta})\}^{Y_i}\{1-p(X_i, \tilde{\beta})\}^{1-Y_i}$$

When the samples are independent, the likelihood function is

$$L(\tilde{\beta}) = \Big\{ \prod_{i=1}^{n} p(X_i, \tilde{\beta})^{Y_i}\{1 - p(X_i, \tilde{\beta})\}^{1-Y_i} \Big\}$$

where $\tilde{\beta} = (a, \beta)$ and $\tilde{X}_i = (1, X_i^\top)^\top$. The log-likelihood can be written

$$\ell(\tilde{\beta}) \;=\; \log(L(\tilde{\beta})) = \sum_{i=1}^{n} \{Y_i \tilde{\beta}^\top \tilde{X}_i - \log(1 + e^{\tilde{\beta}^\top \tilde{X}_i})\}$$

The maximum likelihood estimator (MLE) of the model is the solution to maximize

$$\max \ell(\tilde{\beta}) \quad \text{or} \quad \min\{-\ell(\tilde{\beta})\}$$

Numerical calculation of the Estimator[1] The solution or estimator satisfies

$$\frac{\partial \ell(\tilde{\beta})}{\partial \tilde{\beta}} = \sum_{i=1}^{n} \tilde{X}_i \{Y_i - p(X_i, \tilde{\beta})\} = 0$$

[1] please skip this park

The numerical solution is calculated by the Newton-Raphson algorithm, which requires the second-derivative or Hessian matrix

$$\frac{\partial^2 \ell(\tilde{\beta})}{\partial \tilde{\beta} \partial \tilde{\beta}^\top} = -\sum_{i=1}^{n} \tilde{X}_i \tilde{X}_i^\top p(X_i, \tilde{\beta})\{1 - p(X_i, \tilde{\beta})\}$$

Starting with $\tilde{\beta}_{old}$, a single Newton update[2] is

$$\tilde{\beta}_{new} = \tilde{\beta}_{old} - \{\frac{\partial^2 \ell(\tilde{\beta})}{\partial \tilde{\beta} \partial \tilde{\beta}^\top}\}^{-1} \frac{\partial \ell(\tilde{\beta})}{\partial \tilde{\beta}}\Big|_{\tilde{\beta}=\tilde{\beta}_{old}}$$

where the derivatives are evaluated at $\tilde{\beta}_{old}$. In matrix, let

$$\mathbf{p} = \begin{pmatrix} p(X_1, \tilde{\beta}_{old}) \\ \vdots, \\ p(X_n, \tilde{\beta}_{old}) \end{pmatrix}, \quad \mathbb{X} = (\tilde{X}_1, ..., \tilde{X}_n)^\top, \quad \mathbb{Y} = \begin{pmatrix} Y_1 \\ \vdots, \\ Y_n \end{pmatrix}$$

$$W = \begin{pmatrix} p(X_1, \tilde{\beta}_{old})\{1 - p(X_1, \tilde{\beta}_{old})\} & 0 & ... & 0 \\ 0 & p(X_2, \tilde{\beta}_{old})\{1 - p(X_2, \tilde{\beta}_{old})\} & ... & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & ... & p(X_n, \tilde{\beta}_{old})\{1 - p(X_n, \tilde{\beta}_{old})\} \end{pmatrix}$$

[2]This is a general rule for general minimization problems

Then

$$\frac{\partial \ell(\tilde{\beta})}{\partial \tilde{\beta}} = \mathbb{X}^\top(\mathbb{Y} - \mathbf{p}), \qquad \frac{\partial^2 \ell(\tilde{\beta})}{\partial \tilde{\beta} \partial \tilde{\beta}^\top} = -\mathbb{X}^\top W \mathbb{X}.$$

The Newton step is thus

$$\begin{aligned}
\tilde{\beta}_{new} &= \tilde{\beta}_{old} - (\mathbb{X}^\top W \mathbb{X})^{-1} \mathbb{X}^\top (\mathbb{Y} - \mathbf{p}) \\
&= (\mathbb{X}^\top W \mathbb{X})^{-1} \mathbb{X}^\top W \{\mathbb{X}\tilde{\beta}_{old} + W^{-1}(\mathbb{Y} - \mathbf{p})\} \\
&= (\mathbb{X}^\top W \mathbb{X})^{-1} \mathbb{X}^\top W \mathbf{z}
\end{aligned}$$

where $\mathbf{z} = \mathbb{X}\tilde{\beta}_{old} + W^{-1}(\mathbb{Y} - \mathbf{p})$. Thus

$$\tilde{\beta}_{new} = arg \min_{\beta} (\mathbf{z} - \mathbb{X}\beta)^\top W (\mathbf{z} - \mathbb{X}\beta)$$

called iteratively re-weighted least squares, or IRLS[3]

[3] This idea can be extended to other cases, such as the ridge regression or LASSO

Classification

For a new sample X, the estimated model can predict its p

$$\hat{p} = \frac{\exp(\hat{a} + \hat{\beta}^\top X)}{1 + \exp(\hat{a} + \hat{\beta}^\top X)}$$

The classification is made as follows

- if $\hat{p} \geq 0.5$, classify $Y = 1$
- if $\hat{p} < 0.5$, classify $Y = 0$

2 The multilevel logistic regression model

Suppose there are K classes/groups, for convenience labeled $Y = 1, 2, ..., K$. We need to model the probability for any individual $X = (\mathbf{x}_1, ..., \mathbf{x}_p)^\top$

$$p_k = P(Y = k|X), \quad k = 1, ..., K$$

Obviously,

$$0 \leq P(Y = k|X) \leq 1, \quad \text{and} \quad p_1 + ... + p_K = 1 \tag{2.1}$$

We can model the probability by the **multilevel logistic regression model**

$$P(Y = k|X) = \frac{\exp(a_k + \beta_k^\top X)}{\exp(a_1 + \beta_1^\top X) + ... + \exp(a_K + \beta_K^\top X)}$$

Note that the notation $\beta_k = (\beta_{k1}, ..., \beta_{kp})^\top, k = 1, ..., K$.[4]

[4]note that in some books, there are only K-1 sets of parameters in stead of K

We can estimate the model similarly, and make prediction for sample X

$$\hat{p}_k = \frac{\exp(\hat{a}_k + \hat{\beta}_k^\top X)}{\exp(\hat{a}_1 + \hat{\beta}_1^\top X) + ... + \exp(\hat{a}_K + \hat{\beta}_K^\top X)}$$

with

$$\hat{p}_1 + ... + \hat{p}_K = 1.$$

We make classification,

with probability $\hat{p}_k$, the sample falls into class k.

or

the sample is classified as j if $p_j = max\{p_1, ..., p_K\}$.

R: glm(....), glmnet(....), predict(....)

- myreg = glm(y∼x, family="binomial")

  OUTPUT: ∗ myreg$coefficients ∗ summary(myreg)

- IMPORTANT FUNCTION: as.factor([a variable]) to change values to symbols "levels"

- myglm = glmnet(x, y, family="multinomial", lambda=0, alpha=0)

  OUTPUT: ∗ myglm$beta, ∗ myglm$a0

- PREDICTION: predict(myglm, newx = ??, type="response/class")

Example 2.1 For the heart diseases data ((**training set**) , (**validation set**)), we use logistic model to classify the data. The response variable is whether a man has coronary heart disease. The model is

$$P(y=1|X) = \frac{\exp(\alpha + \beta_1 \mathbf{x}_1 + ... + \beta_9 \mathbf{x}_9)}{1 + \exp(\alpha + \beta_1 \mathbf{x}_1 + ... + \beta_9 \mathbf{x}_9)}$$

The estimated model is

$$\begin{aligned}
\log(\frac{\hat{p}}{1-\hat{p}}) &= -6.7973 + 0.0079\mathbf{x}_1 + 0.0824\mathbf{x}_2 + 0.1866\mathbf{x}_3 + 0.0067\mathbf{x}_4 + 0.9435\mathbf{x}_5 \\
S.E. \quad & (1.389) \quad (0.006) \quad (0.027) \quad (0.062) \quad (0.031) \quad (0.2339) \\
& +0.0436\mathbf{x}_6 - 0.0513\mathbf{x}_7 - 0.0007\mathbf{x}_8 + 0.0488\mathbf{x}_9 \\
& 0.0127 \quad (0.0471) \quad (0.0046) \quad (0.01255)
\end{aligned}$$

```
> summary(mylogist)

Call:
glm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9,
    family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8314  -0.8041  -0.4221   0.8618   2.5923

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.7972945  1.3890372  -4.894 9.90e-07 ***
x1           0.0078531  0.0060083   1.307 0.191204
x2           0.0824016  0.0270630   3.045 0.002328 **
x3           0.1865903  0.0620688   3.006 0.002645 **
x4           0.0066980  0.0313151   0.214 0.830632
x5           0.9435008  0.2339728   4.033 5.52e-05 ***
x6           0.0435738  0.0127968   3.405 0.000662 ***
x7          -0.0512560  0.0471108  -1.088 0.276600
x8          -0.0007337  0.0046174  -0.159 0.873745
x9           0.0488264  0.0125550   3.889 0.000101 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 575.72  on 446  degrees of freedom
Residual deviance: 449.67  on 437  degrees of freedom
AIC: 469.67
```

The predicted probabilities for the new data set are respectively

$$0.09\ 0.09\ 0.52\ 0.12\ 0.07\ 0.48\ 0.33\ 0.19\ 0.06\ 0.52\ 0.60\ 0.34\ 0.13\ 0.57\ 0.62$$

and classified groups

$$0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1$$

The true response in the validation sets are

$$0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1$$

The rate of misclassification is

$$7/15$$

See code02Cx1.R

Example 2.2 In a popular simulated example, called waveform data, there are K=3 types response, with 21 predictors

$$Y = 1, 2, 3,$$

with $X = (\mathbf{x}_1, ..., \mathbf{x}_{21})$ Based on training data **(training set)**, our fitted model is

$$\hat{p}_1 = \frac{\exp(-2.30 + 0.17\mathbf{x}_1 + ... - 0.50\mathbf{x}_{21})}{\exp(-2.30 + 0.17\mathbf{x}_1 + ... - 0.50\mathbf{x}_{21}) + \exp(-1.68 - 0.18\mathbf{x}_1 + ... + 0.62\mathbf{x}_{21}) + \exp(3.98 + 0.01\mathbf{x}_1 + ... - 0.12\mathbf{x}_{21})}$$

$$\hat{p}_2 = \frac{\exp(-1.68 - 0.18\mathbf{x}_1 + ... + 0.62\mathbf{x}_{21})}{\exp(-2.30 + 0.17\mathbf{x}_1 + ... - 0.50\mathbf{x}_{21}) + \exp(-1.68 - 0.18\mathbf{x}_1 + ... + 0.62\mathbf{x}_{21}) + \exp(3.98 + 0.01\mathbf{x}_1 + ... - 0.12\mathbf{x}_{21})}$$

$$\hat{p}_3 = \frac{\exp(3.98 + 0.01\mathbf{x}_1 + ... - 0.12\mathbf{x}_{21})}{\exp(-2.30 + 0.17\mathbf{x}_1 + ... - 0.50\mathbf{x}_{21}) + \exp(-1.68 - 0.18\mathbf{x}_1 + ... + 0.62\mathbf{x}_{21}) + \exp(3.98 + 0.01\mathbf{x}_1 + ... - 0.12\mathbf{x}_{21})}$$

Next, we make classification for the **(test set)**. Below are the predicted

probabilities for the first few samples in the testing set

```
                1            2            3
 [1,] 7.757738e-06 9.999877e-01 4.528874e-06
 [2,] 9.855757e-01 1.442414e-02 1.218037e-07
 [3,] 8.780823e-05 9.872205e-01 1.269165e-02
 [4,] 1.913173e-05 5.169965e-01 4.829844e-01
 [5,] 9.162010e-01 8.368719e-02 1.118265e-04
 [6,] 1.654262e-01 9.876561e-08 8.345737e-01
 [7,] 1.929775e-01 5.834892e-04 8.064390e-01
 [8,] 8.468964e-02 9.153077e-01 2.632778e-06
 [9,] 2.745262e-01 7.254732e-01 6.432133e-07
[10,] 2.767948e-07 2.559017e-01 7.440980e-01
[11,] 9.928826e-01 2.161088e-09 7.117426e-03
```

and below are the classification for the testing data

"2" "1" "2" "2" "1" "3" "3" "2" "2" "3" "1" "2" "3" "1" "3" ...

The rate of misclassification is 0.178.

See code02Cb.R

3 Ridge regression and LASSO for the logistic regression

Similarly[5], we can estimate the model by imposing penalty

$$\beta_{ridge} = arg\max_{\beta}\{\ell(\beta) - \lambda\|\beta\|_2^2\} \qquad (ridge\ regression)$$

or

$$\beta_{lasso} = arg\max_{\beta}\{\ell(\beta) - \lambda\|\beta\|_1\} \qquad (lasso)$$

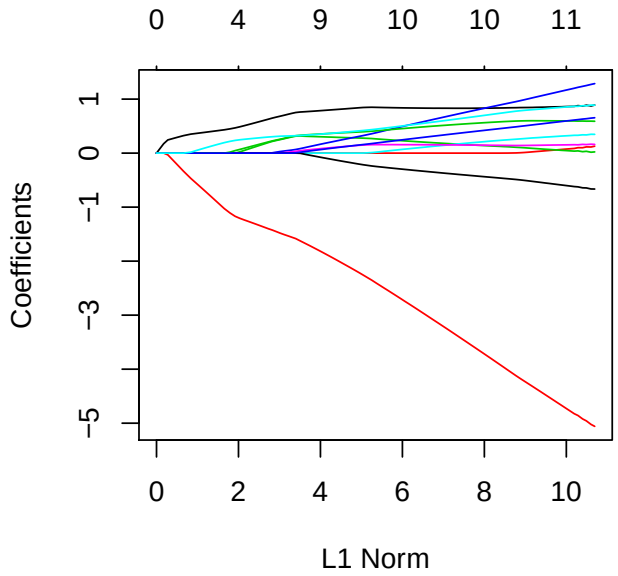
or

$$\beta_{net} = arg\max_{\beta}\left[\ell(\beta) - \lambda\{\alpha\|\beta\|_1^2 + (1-\alpha)\|\beta\|_2^2\}\right] \qquad (net)$$

[5] $\|\beta\|_2^2 = \beta_1^2 + ... + \beta_p^2, \quad ||\beta||_1 = |\beta_1| + ... + |\beta_p$

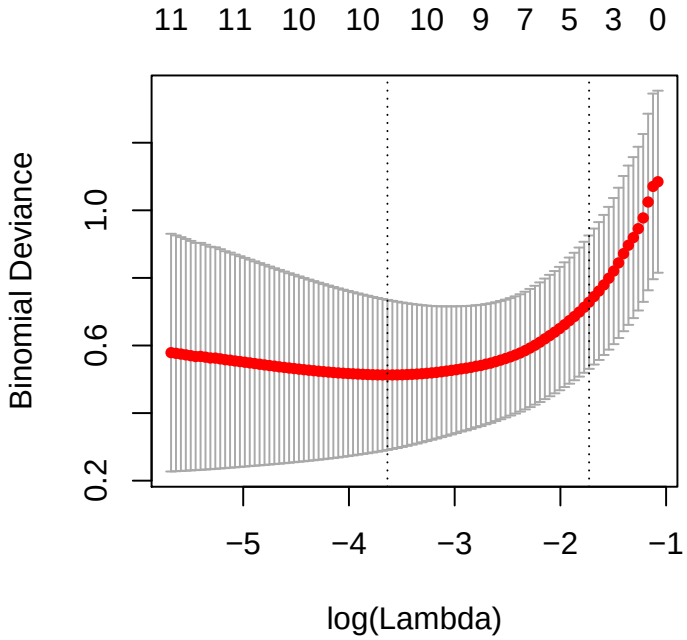
Example 3.1 (cell classification) For the leukemia gene expression data sets, **(training data)** and **(testing set)**[6] There are 25 cells with 250 genes in the training set. there are from two types of cells.

The path of selection is as follows.



[6]More statistician-friendly data can be found at http://stat.ethz.ch/∼dettling/bagboost.html

The values of λ and its CV-values



R code for the calculation **(code)**

Note that only a few variables(genes) are selected in the prediction, i.e. only those genes might be the cause of the disease.

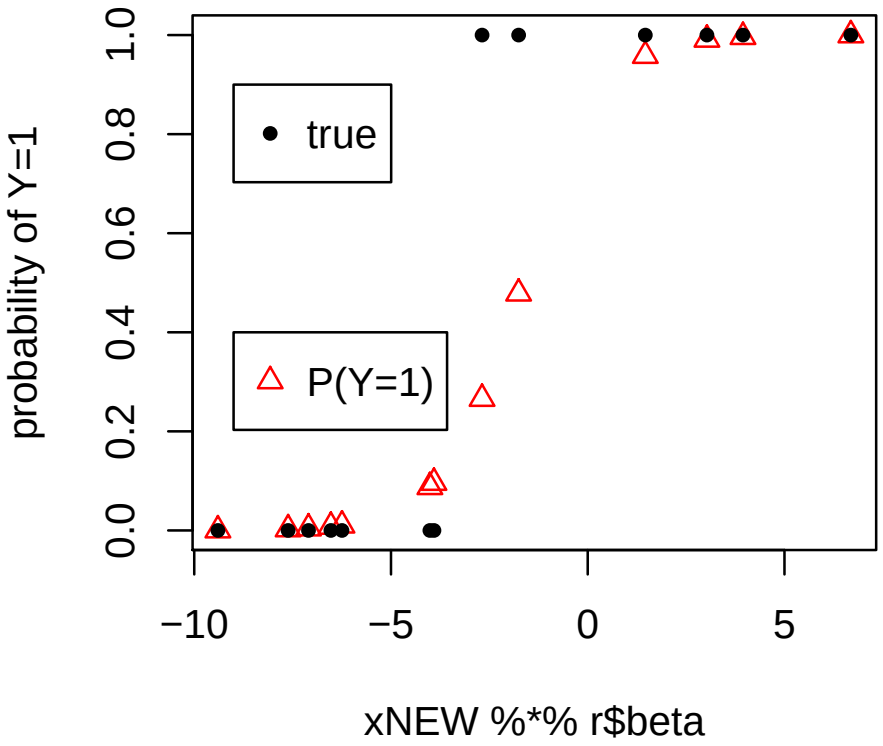
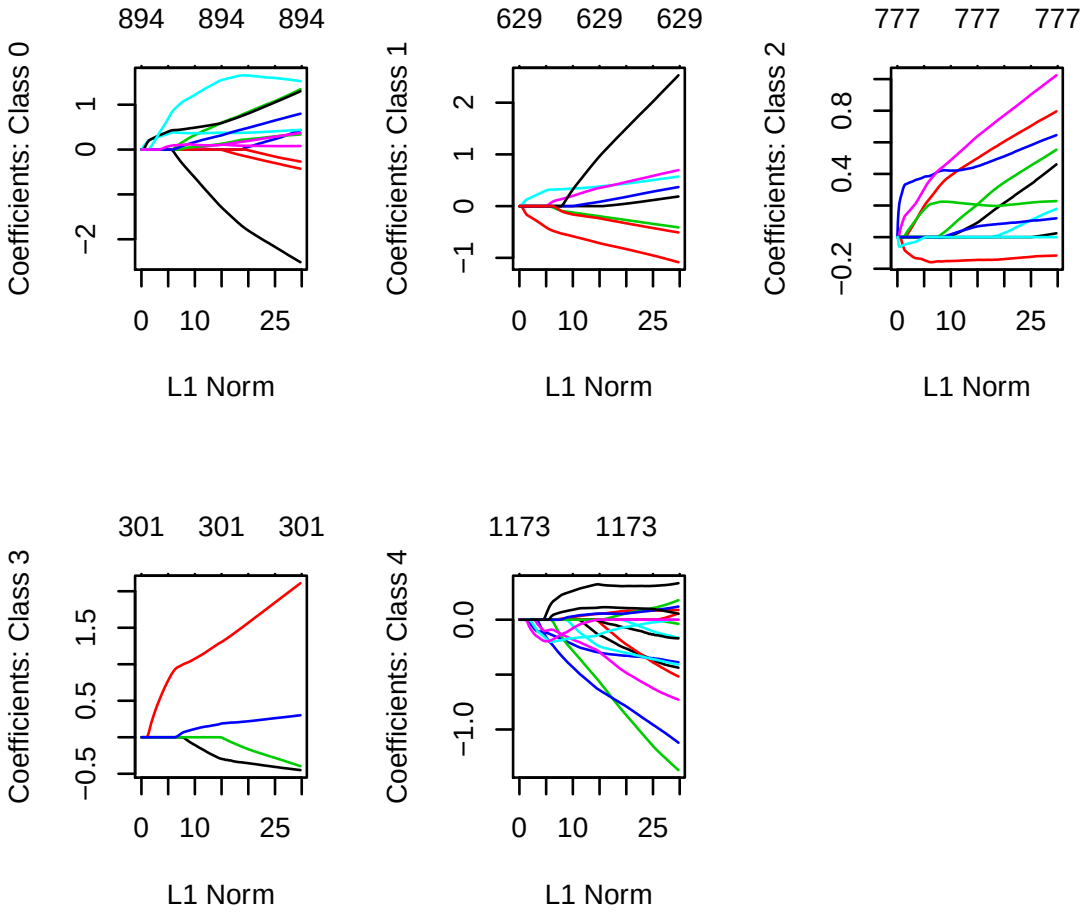


Figure 1: The classification of the the 13 cells in the validation set

Example 3.2 The brain data brain.rda, contains 42 cells each of which as 5597 genes. The cells are from 5 types, denoted by 0, 1, 2, 3, 4 respectively. We fit a logistic model with 5 levels. We apply the lasso to estimate the model. See **(code02Ca.R)**. The fitted probabilities are

	groups				
cell	0	1	2	3	4
1	0.6874	0.0538	0.1226	0.0262	0.1099
2	0.7986	0.0337	0.0650	0.0138	0.0889
⋮	⋮	⋮	⋮	⋮	⋮
12	0.0359	0.8464	0.0333	0.0307	0.0536
13	0.0502	0.8282	0.0340	0.0361	0.0515
⋮	⋮	⋮	⋮	⋮	⋮
41	0.0338	0.0845	0.0451	0.0553	0.7814
42	0.0739	0.0922	0.0536	0.0278	0.7526


894 894 894
Coefficients: Class 0
−2 0 1
0 10 25
L1 Norm
629 629 629
Coefficients: Class 1
−1 0 1 2
0 10 25
L1 Norm
777 777 777
Coefficients: Class 2
−0.2 0.4 0.8
0 10 25
L1 Norm
301 301 301
Coefficients: Class 3
−0.5 0.5 1.5
0 10 25
L1 Norm
1173 1173
Coefficients: Class 4
−1.0 0.0
0 10 25
L1 Norm

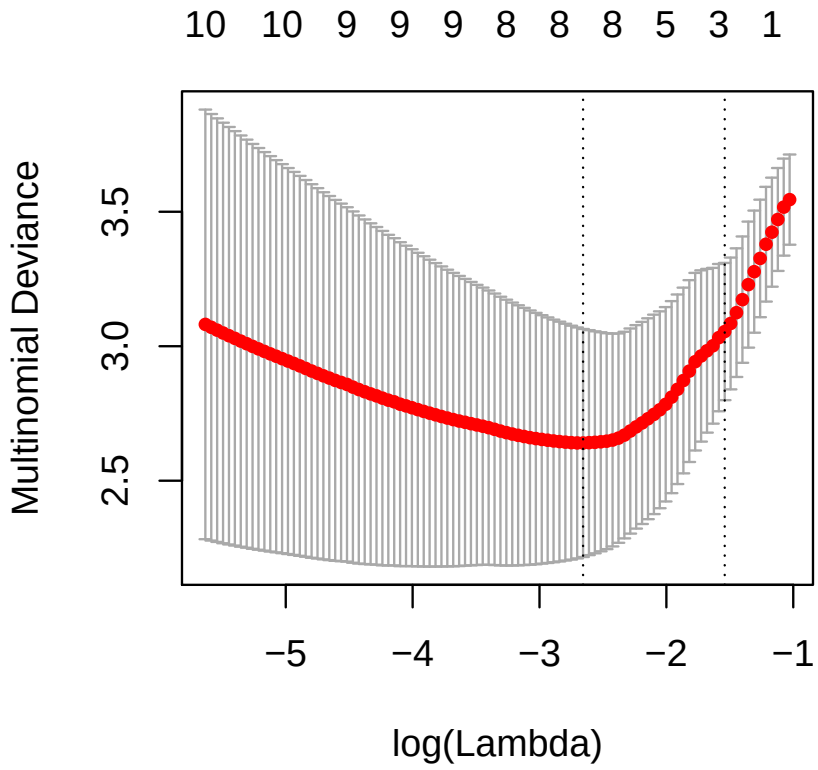


Figure 2: CV values and the tuning parameter values