High Dimensional Data For Logistic Regression

Hongyuan Lu
Kansas State University

Abstract

This paper is mainly focus on how to use coordinate descent to solve high dimensional data for logistic regression when the response y from the data is binary. In this paper we only consider binary not binomial. The core idea is the that instead of using $||y - x'\beta||_2^2$, the log likelihood would be used for binary logistic regression, to achieve the penalized log likelihood. And we will try to maximize the penalized loglikelihood with respect to β. The penalty of lasso, elastic net, mcp will be given. And the formula for β_k of lasso, elastic net and mcp will be approached with proofs.

1 Introduction

The coordinate descent is a popular way for linear regression. It uses a penalty to achieve a sparse solution. Considering that we have a response variable $y \in R$ and a predictor vector $x \in R^p$, and we can approximate the regression model using $E(Y|X = x) = x'\beta$, where we have N observation pairs (x_i, y_i). Suppose if p is much larger than n, than we can use coordinate descent to solve the problem of $\min_{\beta \in R^p} \frac{1}{2N} \sum_{i=1}^{N} (y_i - x_i'\beta)^2 + \lambda P_a(\beta)$ where $P_a(\beta)$ could be either lasso, elastic net, MCP or something else. However in the above model we have continuous responses, and what if we have a logistic regression whose responses are not continuous.

For example let us consider the logistic regression model $log\frac{P(y=1|x)}{1-P(y=1|x)} = x'\beta$ then we have $P(y = 1|x) = \frac{1}{1+e^{-x'\beta}}$ and $P(y = 0|x) = \frac{1}{1+e^{x'\beta}} = 1 - P(y = 1|x)$. Here it raises the problem that we do not have continuous responses, and we can not directly use $\min_{\beta \in R^p} \frac{1}{2N} \sum_{i=1}^{N} (y_i - x_i'\beta)^2 + \lambda P_a(\beta)$ with coordinate descents. The paper Regularization Paths for Generalized Linear Models via Coordinate Descent (Jerome Friedman, Trevor Hastie , 2010) gives a nice solution.

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

2 Logistic Regression Approach

The basic idea in their paper is to maximize the penalized log likelihood, where the likelihood is $l(\beta) = \frac{1}{n} \sum_{i=1}^{N} y_i(x_i'\beta) - log(1 + e^{x'\beta})$. To maximize the penalized log likelihood is to find $\max_\beta \frac{1}{n} \sum_{i=1}^{N} y_i(x_i'\beta) - log(1+e^{x'\beta}) + \lambda P(\beta)$ with respect to β. Directly working on the log likelihood can be hard, but by Jerome Friedman, Trevor and Hastie Rob Tibshirani(2010), with taylor series, the quadratic approximation of the log likelihood is

$$l_Q(\beta) = -\frac{1}{2n} \sum_{i=1}^{n} w_i(z_i - x_i'\beta)^2 + c(\beta)^2$$

where

$$z_i = x_i'\beta + \frac{y_i - \hat{P}(x_i)}{\hat{P}(x_i)(1 - \hat{P}(x_i))}$$

$$w_i = \hat{P}(x_i)(1 - \hat{P}(x_i))$$

$c(\beta)$is constant. Now it is very easy to understand that to maximize the penalized log likelihood $\max_\beta \frac{1}{n} \sum_{i=1}^{N} y_i(x_i'\beta) - log(1 + e^{x'\beta}) + \lambda P(\beta)$, it is equivalent to minimize $-l(\beta) + \lambda P(\beta)$ with respect to β. Since we are doing the minimize, we do not consider the constant. Using the quadratic approximation of the log likelihood l_Q, we are going to minimize $\frac{1}{2n} \sum_{i=1}^{n} w_i(z_i - x_i'\beta)^2 + \lambda(\beta)$ with respect to β.
For the next, I will get the formula of β to minimize the penalized quadratic approximation of log likelihood, using the penalty of lasso, elastic net, and MCP.

2.1 Derive Formulas

2.1.1 Lasso Approach

For lasso the the penalty is $\lambda|\beta|$ and we have $l_Q(\beta) = -\frac{1}{2n} \sum_{i=1}^{n} w_i(z_i - x_i'\beta)^2 + c(\beta)^2$.
then
$-l_Q(\beta) + \lambda|\beta|$

$= \frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}|\beta_j|$

$$= \frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)} - x_{ik}\beta_k)^2 + \lambda\sum_{j=1}^{p}|\beta_j|$$

Taking derivatives with respect to β_k for the penalized log likelihood and set it equal to 0 we have

$$\frac{1}{2n}\sum_{i=1}^{n} 2w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)} - x_{ik}\beta_k)(-x_{ik}) + \lambda\frac{d|\beta_k|}{d\beta_k}$$

$= -\frac{1}{2n}\sum_{i=1}^{n} 2w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})x_{ik} + \frac{1}{2n}\sum_{i=1}^{n} 2w_i x_{ik}^2\beta_k + \lambda\frac{d|\beta_k|}{d\beta_k}$

$= 0$ then we have

$$\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2\beta_k = \frac{1}{n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})x_{ik} - \lambda\frac{d|\beta_k|}{d\beta_k}$$

let $S_k = \frac{d|\beta_k|}{d\beta_k}$, then

$$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}[\frac{1}{n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,\neq k}^{p} x_{ij}\beta_j^{(k)})x_{ik} - \lambda S_k]$$

let $f_k = \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})$ then
$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}[f_k - \lambda S_k]$
Then we have
if $\beta_k > 0$ then $f_k - \lambda S_k = f_k - \lambda \Rightarrow f_k > \lambda$

$$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}[f_k - \lambda]$$

if $\beta_k < 0$ then $f_k - \lambda S_k = f_k + \lambda \Rightarrow f_k < -\lambda$

$$\beta_k = -\frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}[-f_k - \lambda]$$

if $\beta_k = 0$ then $f_k = \lambda S_k$ and $S_k \in [-1, 1] \Rightarrow |f_k| \leq \lambda$
In Summary

$$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2} sign(f_k)[|f_k| - \lambda]_+$$

where

$$f_k = \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})$$

2.1.2 Elastic Net Approach

For elastic net the penalty is $\lambda\frac{1-\alpha}{2}||\beta||_{l2}^2 + \alpha\lambda|\beta|$ where $0 < \alpha < 1$ and

$$l_Q(\beta) = -\frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - x_i'\beta)^2$$

. So it is to minimze

$$\frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1}^{p} x_i\beta_j)^2 + \frac{1-\alpha}{2}\lambda\sum_{j=1}^{p}\beta_j^2 + \alpha\lambda\sum_{j=1}^{p}|\beta_j|$$

$= \frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,j\neq p}^{p} x_i\beta_j^{(k)} - x_{ik}\beta_k)^2 + \frac{1-\alpha}{2}\lambda\sum_{j=1}^{p}\beta_j^2 + \alpha\lambda\sum_{j=1}^{p}|\beta_j|$

take derivative with respect to β_k and set it qual to 0

$\frac{1}{2n}\sum_{i=1}^{n} 2w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)} - x_{ik}\beta_k)(-x_{ik}) + (1-\alpha)\lambda\beta_k + \alpha\lambda S_k = 0$

$\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2\beta_k + (1-\alpha)\lambda\beta_k = \frac{1}{n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})x_{ik} - \alpha\lambda S_k$

let

$$f_k = \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})$$

$$S_k = \frac{d|\beta_k|}{d\beta_k}$$

So

$$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 + (1-\alpha)\lambda}[f_k - \alpha\lambda S_k]$$

if $\beta_k > 0$ then $f_k - \alpha\lambda S_k = f_k - \alpha\lambda \Rightarrow f_k > \alpha\lambda$
if $\beta_k < 0$ then $f_k - \alpha\lambda S_k = f_k + \alpha\lambda \Rightarrow f_k < -\alpha\lambda$
if $\beta_k = 0$ then $f_k = \alpha\lambda S_k$ Since $S_K \in [-1, 1]$, $|f_k| \leq \alpha\lambda$
In Summary

$$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 + (1-\alpha)\lambda} sign(f_k)[|f_k| - \alpha\lambda]_+$$

2.1.3 MCP Approach

The penalty of MCP is the following(Cun-Hui Zhang 2010)

$$\text{p}(\beta_j) = \begin{cases} (|\beta_j| - \frac{\beta_j^2}{2\alpha\lambda}), & if|\beta_j| < \alpha\lambda \\ . \ \frac{\alpha\lambda}{2}, \text{ if } |\beta_j| \geq \alpha\lambda. \end{cases}$$

(1)

where $p(\beta) = \sum_{j=1}^{p} p(\beta_j)$
minimize

$-l_Q(\beta) + \lambda p(\beta)$

$= \frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p} P(\beta_j)$

$$= \frac{1}{2n}\sum_{i=1}^{n} w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j - x_{ik}\beta_k)^2 + \lambda\sum_{j=1}^{p} P(\beta_j)$$

take derivative with respect to β_k and set the derivative to 0.

$$\frac{1}{2n}\sum_{i=1}^{n} 2w_i(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)} - x_{ik}\beta_k)(-x_{ik}) + \frac{d\lambda p(\beta)}{d\beta} = 0$$

$$-\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)}) + \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2\beta_k + \frac{d\lambda p(\beta_k)}{d\beta_k}$$

$= 0$
let
$f_k = \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}(z_i - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j^{(k)})$
$S_k = \frac{d|\beta_k|}{d\beta_k}$
So
$-f_k + \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2\beta_k + \frac{d\lambda p(\beta_k)}{d\beta_k} = 0$
Condition 1
if $|\beta_k| < \alpha\lambda$
$-f_k + \frac{1}{n} w_i x_{ik}^2\beta_k + \lambda S_k - \frac{\beta_k}{\alpha} = 0$

$\beta_k = \frac{f_k - \lambda S_k}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}}$

sub condition 1.1
if $\alpha\lambda > \beta_k > 0$ then $\beta_k = \frac{f_k - \lambda}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}}$
and $f_k - \lambda > 0 \Rightarrow f_k > \lambda$
$\frac{f_k - \lambda}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik} - \frac{1}{\alpha}} < \alpha\lambda \Rightarrow f_k < \alpha\lambda\sum_{i=1}^{n} w_i x_{ik}^2$

sub condition 1.2
if $-\alpha\lambda < \beta_k < 0$ then $\beta_k = -\frac{-f_k - \lambda}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}}$
and $f_k + \lambda < 0 \Rightarrow f_k < -\lambda$
$\frac{-f_k - \lambda}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}} > -\alpha\lambda \Rightarrow f_k > -\alpha\lambda\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2$

sub condition 1.3 if $\beta_k = 0$ then $f_k = S_k\lambda \Rightarrow |f_k| \leq \lambda$
since $S_k \in [-1, 1]$
For condition 1 in summary
$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}} sign(f_k)[|f_k| - \lambda]_+$
if $|f_k| < \frac{\alpha\lambda}{n}\sum_{i=1}^{n} w_i x_{ik}^2$

Condition 2 if $|\beta_k| \geq \alpha\lambda$ then
$\frac{d\lambda p(\beta_k)}{d\beta_k} = 0$ then
$-f_k + \frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2\beta_k = 0$
$\beta_k = \frac{f_k}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}$ and
$|\beta_k| \geq \alpha\lambda \Rightarrow |f_k| \geq \frac{\alpha\lambda}{n}\sum_{i=1}^{n} w_i x_{ik}^2$
so for condition 2
$\beta_k = \frac{f_k}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}$ if $|f_k| \geq \frac{\alpha\lambda}{n}\sum_{i=1}^{n} w_i x_{ik}^2$

In summary
$\beta_k = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2 - \frac{1}{\alpha}} sign(f_k)[|f_k| - \lambda]_+$
if $|f_k| < \frac{\alpha\lambda}{n}\sum_{i=1}^{n} w_i x_{ik}^2$
$\beta_k = \frac{f_k}{\frac{1}{n}\sum_{i=1}^{n} w_i x_{ik}^2}$

if $|f_k| \geq \frac{\alpha\lambda}{n}\sum_{i=1}^{n} w_i x_{ik}^2$

2.2 Results and Discussion

2.2.1 Algorithm

According to the paper of Jerome Friedman and Trevor Hastie(2010), we have 3 loops for the algorithm. The largest loop is to iterate through the λs from the largest to the smallest. The middle loop is to update the quadratic approximation of the likelihood. The small loop is to do coordinate descent. In the R file, for lasso, the algorithm to get β for a single λ is in function LGRlasso, the algorithm to get the optimal λ, true positive, false positive and deviance is in function runlasso, for elastic net, the algorithm to get β for a single λ in function LGRelsnet, the algorithm to get the optimal λ, true positive, false positive and deviance is in function runelslasso, for mcp, the algorithm to get β for a single λ is in function LGReMCP, the algorithm to get the optimal λ, true positive, false positive and deviance is in function runMCP.

2.2.2 Deviance and Misclassification

Here raises another problem. Which λ is the best. If we have a continuous response we can use mean squared error to determine it. But here the response y is discrete for logistic regression. One solution is to use the misclassification. Another solution is to use deviance. The deviance function is
$d_i = \sqrt{2[log(1 + e^{x_i'\beta}) - x_i'\beta]}$ if $y_i = 1$
$d_i = -\sqrt{2log(1 + e^{x_i'\beta})}$ if $y_i = 0$
Now we can use misclassification or deviance. Our first problem is that does misclassification give the best λ. Set seed in R to be 29, and generate x from multi normal distributions.Each x_i is p dimensional and p=120. Let $\beta = (0, 0, 6, 5, -5, 0, 0, ...)$. Let our training data set to have size 80 and our testing data set to have size 30. Using elastic net we get the result in figure 1. From this result we see that minimum misclassification gives an interval of λs from 0.17 to 0.14. We can not tell which one is the best in the interval only according to misclassification. One way to solve this is to increase the testing data size. Once the testing data size is increased, the interval corresponding to the the minimum misclassification decreased dramatically. Figure 2 is the deviance vs λ and misclassification vs

λ with a testing data set of size 30 for lasso setting under seed 39. Figure 3 is the deviance vs λ and misclassification vs λ with a testing data set of size 200 for lasso and under seed 39.

| Lambda deviance. Misclassification true_positive false_positi

[1] 0.24000 25.61766 5.00000 3.00000 1.00000

[1] 0.23000 25.04252 4.00000 3.00000 1.00000

[1] 0.22000 24.43286 4.00000 3.00000 1.00000

[1] 0.21000 23.80689 3.00000 3.00000 1.00000

[1] 0.20000 23.14048 3.00000 3.00000 1.00000

[1] 0.19000 22.43728 3.00000 3.00000 2.00000

[1] 0.18000 21.67937 3.00000 3.00000 3.00000

[1] 0.17000 20.87811 2.00000 3.00000 3.00000

[1] 0.16000 20.06349 2.00000 3.00000 4.00000

[1] 0.15000 19.31687 2.00000 3.00000 6.00000

[1] 0.14000 18.56515 2.00000 3.00000 6.00000

[1] 0.13000 18.01588 4.00000 3.00000 10.00000

[1] 0.12000 17.85181 4.00000 3.00000 11.00000

[1] 0.1100 18.0088 4.0000 3.0000 16.0000

[1] 0.10000 18.38228 3.00000 3.00000 20.00000

Figure 1:

	Lasso	Elastic Net	MCP
Optimal Lambda	0.1216 (0.03513)	0.2258 (0.1286)	0.4416 (0.0508)
misclass	22.06 (10.6836)	28.02 (6.9941)	40.12 (10.7791)
True Positive	2.76 (0.4764)	2.56 (0.5406)	1.42 (0.8104)
False Positive	4.98 (6.0928)	6.12 (5.0654)	11.8 (35.4267)

Table 1: sample data size 200

Comparing figure 2 and figure 3, we can obviously see that the interval of λs with minimum misclassification decreased dramatically when testing data increased from 30 to 200. So misclassification is a good way to pick up optimal λ only when testing sample size is large. Then what if a large enough testing sample size is not available. Let us consider the deviance. For the result from figure 1, we will have $\lambda = 0.13$ with the smallest deviance. But this λ does not give the smallest misclassification, and it does not give the smallest

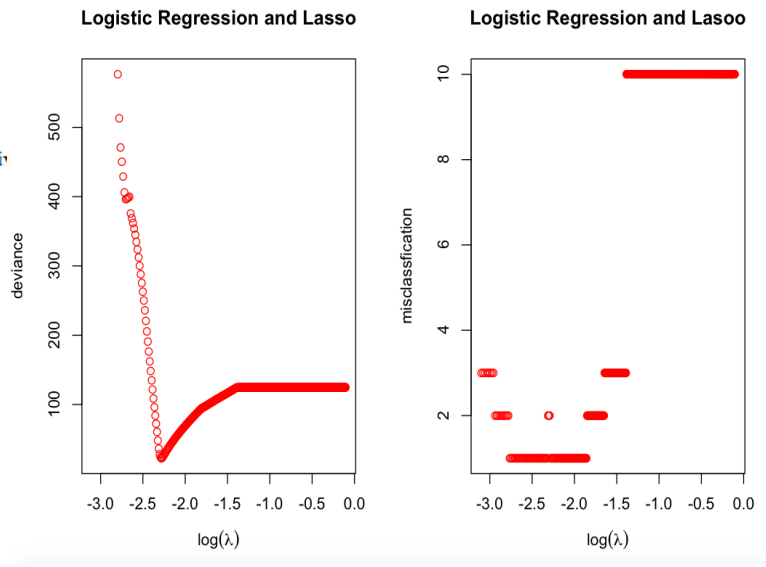

Figure 2:

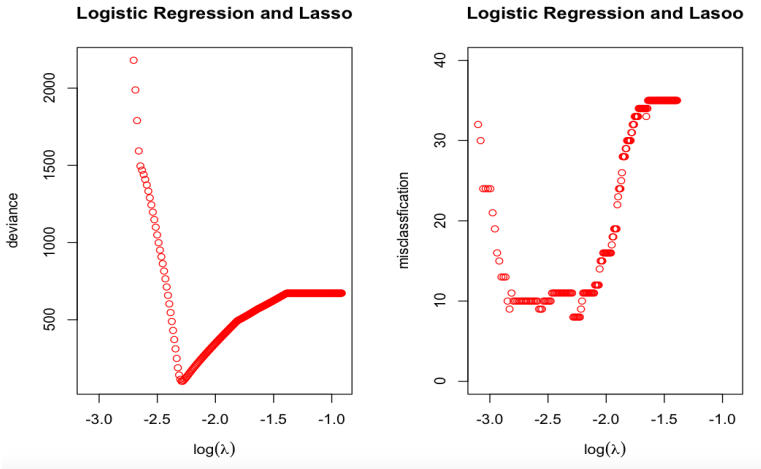

Figure 3:

false positive.

So determine the λ only on deviance may not be good enough. Here is my solution. We will first select the λ with the smallest misclassification and get an interval of λs. Then in this interval, I will select the optimal λ with the smallest deviance. Then in the result from figure1 the optimal λ is going to be 0.14. It gives exactly three true positives and gives 6 false positives which is not too bad. However according to true positive and false positive $\lambda = 0.14$ is still not the best. It is very likeliy that when your testing sample size is small, both deviance and misclassification may not give the best λ with the maximum true positves and minimum false positives. This is very disappointed to me. I think in the further research, we need to find another statistic other than misclassification or deviance

	Lasso	Elastic Net	MCP
Optimal Lambda	0.1514 (0.1129)	0.2052 (0.1589	0.4376 (0.0414)
misclass	3.7 (2.4432)	4.12 (1.8696)	5.74 (2.5056)
True Positive	2.5 (0.7354)	2.5 (0.6145)	1.56 (0.8843
False Positive	1.86 (2.4160)	8.04 (6.2334)	16.72 (40.8806)

Table 2: sample data size 30

to determine which λ is optimal for logistic regression, especially when we can not simply increase the testing data size. Given the criteria to find the optimal λ, I will use my solution to run 50 replicates to simulate the data and get result. For elastic net, $\alpha = 0.5$. For MCP $\alpha = 3$. let p=120 and training data set have size of 80, and testing data set have size of 200 for table 1 and 30 for table 2. Our X is form multi normal distribution. Another thing to notice is that the MCP algorithm is very slow, but its deviance increases super fast after reaching the smallest misclassification at some point. So in the MCP algorithm, when the deviance starts to increase, or somewhere the misclassification reaches the minimum and starts to increase, I will break the large loop, and save a lot of algorithm time. This small trick is in my code for algorithm of MCP. Table 1 and table 2 is the result of 50 replicates. The number in each cell are means for 50 replicates. And the number in the braces are the standard deviations.

2.2.3 Replicate Result and Discussion

Table 1 are the results running on a test data set of size 200.The result given by MCP is really bad. Considering the true positive, the result for MCP is 1 1 1 1 1 1 1 1 1 1 3 1 1 1 3 1 1 3 1 2 1 3 1 3 1 1 1 3 1 1 1 1 1 1 3 1 1 3 2 3 3 1 3 3 1 3 1 1 1 1. I double checked the formula and it is correct. There is a lot of thing I did trying to fix this problem. First I tried to change the weights from a vector of 0.25s to $w_i = \hat{p}(x_i)(1 - \hat{p}(x_i))$, but it improves nothing. Generally speaking, MCP does not perform very well for logistic regression. For true positive, lasso is slightly better than elastic net. Lasso gives slightly more maximum true positives. The true positives for lasso are 3 3 3 3 3 3 3 3 2 3 2 2 3 3 2 3 3 3 3 3 3 2 3 3 1 3 3 3 3 3 3 3 3 3 2 2 2 2 3 3 3 3 2 3 3 3 3 3 3 3 3 3, and the true positives for elastic net are 3 3 2 3 3 3 3 2 3 2 2 3 2 2 2 3 3 3 3 3 3 2 3 1 2 3 2 2 3 3 2 3 3 3 2 2 2 2 2 3 2 3 2 3 3 3 3 2 3 3 3. However, for false positive, lasso is much better than elastic net. The MCP is the worst of all. And further, in logistic regression, the performance of MCP, lasso, elastic is highly depended on the random generation of sample data. Regenerating the sample data are very likely to give a very different performance. The increase of sample size also increase the performance dramatically. I suggest that even we can run coordinate descent to solve high dimensional problem, we should collect sample with as large size as possible in experiment. **I think in the future research, I will focus on the following issues. First I will try to find a statistic better than deviance and misclassification to find the optimal λ. Second, I will try to research on MCP, such as the weights and likelihood.** The most important thing here is that I think that the quadratic approximation of the likelihood might be a bad idea especially for MCP. If we directly work on the real likelihood of logistic regression, then we might have a better result. Think about that, the discrete responses of y already losses some information, the approximation of the likelihood losses some further information. So I will try directly using real log likelihood in my next research.

References

Jerome Friedman, Trevor Hastie (2010).Regularization Paths for Generalized Linear Models via Coordinate Descent

Cun-Hui Zhang(2010).NEARLY UNBIASED VARIABLE SELECTION UNDER MINIMAX CONCAVE PENALTY

Patrick Breheny and Jian Huang(2011). COORDINATE DESCENT ALGORITHMS FOR NONCONVEX PENALIZED REGRESSION, WITH APPLICATIONS TO BIOLOGICAL FEATURE SELECTION