LASSO算法与广义贝叶斯信息准则

---毕业论文中期汇报

高明

2018年4月

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

LASSO (Least absolute shrinkage and selection operator) + **normal distribution**

Given a data set $\{y_i, x_{i1}, \ldots, x_{ip}\}_{i=1}^n$, suppose $y_i \sim N(\mathbf{x}_i^\top \beta, \sigma^2)$, then the model takes the form

$$y_i = \beta_0 1 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\top \beta + \varepsilon_i, \qquad i = 1, \ldots, n, \qquad (1)$$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

$$X = \begin{pmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix},$$

Often these n equations are stacked together and written in vector form as $\mathbf{y} = X\beta + \varepsilon$.

We can calculate the likelihood function

$$\sum_{i=1}^{n} f(y_i|\beta, \sigma^2) = (\frac{1}{\sqrt{2\pi}})^n \sigma^{-n} e^{-\frac{(y-X\beta)^T(y-X\beta)}{2\sigma^2}} \tag{2}$$

MLE: $\hat{\beta} = (X^TX)^{-1}X^Ty, \hat{\sigma^2} = \frac{1}{n}(y - X\hat{\beta})^T(y - X\hat{\beta})$.

When we look it up in the matrix space, it is the same to minimize (when fix σ^2)

$$l(\beta) = (y - X\beta)^T(y - X\beta) \tag{3}$$

OLS: same.

LASSO:

$$\tilde{\beta} = argmin\{\frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2\} \quad subject\ to|\beta|_1 \le t \tag{4}$$

Group Lasso and Sparse Group Lasso

$$\tilde{\beta} = argmin\{\frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda|\beta|_1\} \triangleq argmin\{Q(\beta) + \phi(\beta)\} \quad (5)$$

GRPL:

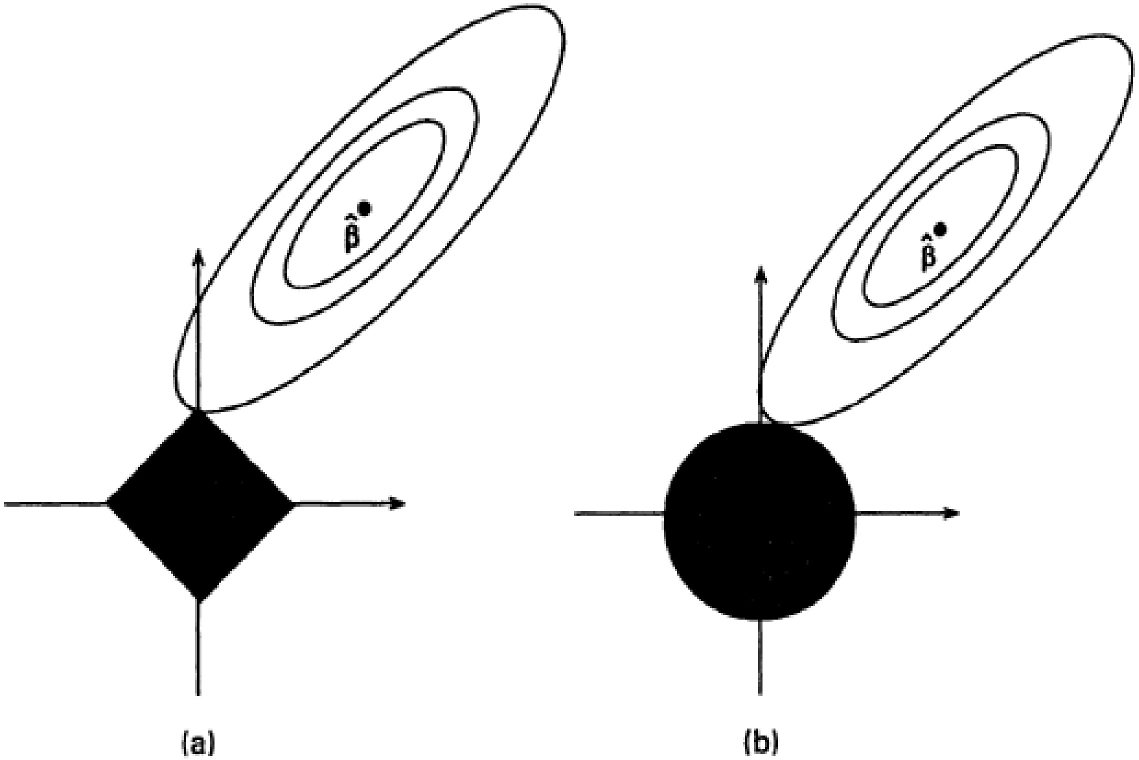
we divide {1...m} into J group $\{G_1, ..., G_J\}$. $\#\{G_j\} \triangleq p_j$. $\beta_{G_j} = (\beta_k)_{k\in G_j}$

$$\phi(\beta) = \lambda\sum_{j=1}^{J}\sqrt{p_j}|\beta_{G_j}|_2 \quad (6)$$

SGRPL:

$$\phi(\beta) = \lambda\{(1-\alpha)\sum_{j=1}^{J}\sqrt{p_j}|\beta_{G_j}|_2 + \alpha|\beta|_1]\} \quad (7)$$

Lasso vs Ridge



Estimation picture for (a) the lasso and (b) ridge regression

$$\tilde{\beta} = argmin\{\frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2\} \quad subject\ to|\beta|_1 \le t$$

Solve lasso with normal distribution

Algorithm 1 Coordinate Descent

1: **repeat**
2: Choose next index j.
3: Update β_j.
4:

$$\beta_j = arg\min_{\beta_j}\{\frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda|\beta|_1\}$$

5: **until** stopping condition is met.

If we standardize the design matrix X, we may get an explicit solution.

$$\beta_j = s[\hat{\beta}_j, \lambda] \triangleq sign(\hat{\beta}_j)(|\hat{\beta}_j| - \lambda)_+ \tag{8}$$

s is called soft-thresholding operator. $\hat{\beta}_j$ is the OLS of residual, fixing β_k, $k \neq j$.

Also, check whether the gradient=0 fall in to the non-differentiable point. Subgradient.

It can be shown that $\beta \to \tilde{\beta}$

Solve grplasso with normal distribution

Algorithm 2 Block Coordinate Descent

1: **repeat**
2: Choose next block index G_j.
3: Update β_{G_j}.
4:
$$\beta_{G_j} = arg\min_{\beta_{G_j}}\{\frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{J}\sqrt{p_j}|\beta_{G_j}|_2\}$$
5: **until** stopping condition is met.

$$\beta_{G_j} = s[\hat{\beta_{G_j}}, \lambda\sqrt{p_j}] \tag{9}$$

Solve sgrplasso with normal distribution

Algorithm 3 Block Coordinate Descent

1: **repeat**
2: Choose next block index G_j.
3: **repeat**
4: Choose next index k in G_j.
5: Update β_k.
6:

$$\beta_k = arg\min_{\beta_k}\{\frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda(1-\alpha)\sum_{j=1}^{J}\sqrt{p_j}|\beta_{G_j}|_2 + \lambda\alpha|\beta|_1\}$$

7: **until** stopping condition is met.
8: **until** stopping condition is met.

the coordinate-wise algorithms for the lasso, the grouped lasso and elastic net, etc. converge to their optimal solutions.

Lasso with binomial distribution

Suppose $y_i \sim Bin(1, p_i)$, then $\mu_i \triangleq E(y_i) = p_i$. Let $x_i^T \beta = g(\mu_i) \triangleq ln\frac{p_i}{1-p_i}$.
Hence,

$$p_i = P(y_i = 1 | x_i) = \frac{1}{1 + e^{-x_i^T \beta}} \tag{10}$$

$$\begin{aligned} Q(\beta) &= \frac{1}{n} \sum_{i=1}^{n} y_i ln p_i + (1 - y_i) ln(1 - p_i) \\ &= \frac{1}{n} \sum_{i=1}^{n} y_i (x_i^T \beta) - ln(1 + e^{x_i^T \beta}) \end{aligned}$$

Algorithm 4 Newton Method

1: **repeat**
2: Evaluate $g = \nabla Q(\beta) = X^T(\mu - y)$
3: Evaluate $H = \nabla g = X^T S X$, here $S = diag\{\mu_1(1-\mu_1), ..., \mu_n(1-\mu_n)\}$
4: Solve $d = -H^{-1}g$
5: Update β: $\beta = \beta + d$
6: $\beta^{t+1} = (X^T S^t X)^{-1} X^T S^t z^t$, here $z^t = X\beta^t + S^{-1t}(y - \mu^t)$
7: **until** stopping condition is met.

Algorithm 5 Iteratively Reweighted Least Squares

1: **repeat** *at* t^{th} iterate
2:

$$Q(\beta^t) = -\frac{1}{2n}\sum_{i=1}^{n} s_i^t (z_i^t - x_i^T \beta^t)^2 + CONSTANT$$

3: Update $\beta^{t+1} = argminQ(\beta^t)$
4: From OLS, $\beta^{t+1} = (X^T S^t X)^{-1} X^T S^t z^t$
5: **until** stopping condition is met.

Solve Lasso with binomial distribution

Algorithm 6 IWLS + Coordinate Descent

1: **repeat** *at* t^{th} iterate
2: Update the quadratic approximation $Q(\beta^t)$ at β^t
3: **repeat**
4: Choose next index j
5: Update $\beta_j^t = arg \min_{\beta_j^t} \{Q(\beta^t) + \lambda|\beta^t|\}$.
6: **until** stopping condition is met.
7: **until** stopping condition is met.

Solve grpLasso with binomial distribution

Algorithm 7 Block Coordinate Gradient Descent

repeat *at* t^{th} iterate
    Update the quadratic approximation $Q(\beta^t)$ at β^t
    repeat
        Choose next block index G_j
        Approximate $H^t_{G_j G_j} = h^t_{G_j} I$
        Find direction $d^t_{G_j}$

$$d^t_{G_j} = arg \min_{d^t_{G_j}} \{Q(\beta^t) + \lambda \sum_{j=1}^{J} \sqrt{p^t_{G_j}} |\beta^t_{G_j}|_2\}$$

        Line search τ using the Armijo rule.
        Update $\beta^t_{G_j} = \beta^t_{G_j} + \tau d^t_{G_j}$
    until stopping condition is met.
until stopping condition is met.

Solve sgrpLasso with binomial distribution

Algorithm 8 Block Coordinate Gradient Descent

repeat *at* t^{th} iterate
  Update the quadratic approximation $Q(\beta^t)$ at β^t
  repeat
    Choose next block index G_j
    Approximate $H^t_{G_jG_j} = h^t_{G_j} I$
    repeat
      Choose next index k in G_j
      Find direction d^t_k

$$d^t_k = arg\min_{d^t_k}\{Q(\beta^t) + \lambda(1-\alpha)\sum_{j=1}^{J}\sqrt{p^t_{G_j}}|\beta^t_{G_j}|_2 + \lambda\alpha|\beta^t|_1\}$$

      Line search τ using the Armijo rule.
      Update $\beta^t_k = \beta^t_k + \tau d^t_k$
    until stopping condition is met.
  until stopping condition is met.
until stopping condition is met.

Model selection: CV

In K-fold cross-validation, the original sample is randomly partitioned into K equal sized subsamples. Of the K subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $K-1$ subsamples are used as training data.

for each $k = 1, ..., K$, fit the model with parameter λ to the other $K-1$ parts, giving $\tilde{\beta}^{-k}(\lambda)$ and compute its loss $LOSS_k(\lambda)$ in predicting the k^{th} part. This gives the cross-validation error

$$CV(\lambda) = \frac{1}{K}\sum_{k=1}^{K} LOSS_k(\lambda) \tag{11}$$

$$\lambda^* = argminCV(\lambda) \tag{12}$$

Model selection: IC

7 AIC

$$AIC(\lambda) = -2lnl(\hat{\beta}(\lambda)) + 2\nu(\lambda) \tag{13}$$

8 BIC

$$BIC(\lambda) = -2lnl(\hat{\beta}(\lambda)) + \nu(\lambda)lnn \tag{14}$$

Here,$\nu(\lambda) = df(\lambda)$

9 EBIC

$$EBIC_\gamma(\lambda) = -2lnl(\hat{\beta}(\lambda)) + \nu(\lambda)lnn + 2\gamma\nu(\lambda)lnp \tag{15}$$

Theorem 1 *Suppose* λ_0 *is the true model. Under some mild conditions with* $n \to \infty$, *we have*

$$P\{minEBIC_\gamma(\lambda) \leq EBIC_\gamma(\lambda_0)\} \to 0 \tag{16}$$

谢谢！