Sparse Regression Without Using a Penalty Function

Kohei Adachi, Osaka University, Japan

Henk A. L. Kiers, University of Groningen, The Netherlands

1. Penalized vs. Unpenalized Sparse Regression

Sparse regression refers to the modified multiple regression which provides a coefficient vector $\boldsymbol{\beta}$ including a number of zeros. For n-observations × (p+1)-variables column-centered block data matrix [**X**, **y**] with **y** an $n \times 1$ dependent variable vector, the existing sparse regression procedures can be formulated as

$$\min_{\boldsymbol{\beta}} f(\boldsymbol{\beta}) + \lambda \,\mathrm{Pen}(\boldsymbol{\beta}) \,, \tag{1}$$

where $f(\boldsymbol{\beta}) = n^{-1}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}'\|_2^2$ is the regression loss function, while Pen($\boldsymbol{\beta}$) is a function penalizing for $\boldsymbol{\beta}$ to have nonzero elements with $\lambda \geq 0$ a penalty weight. A popular penalty function is $\mathrm{Pen}(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_1$ in Lasso. In contrast to such penalized approaches, we propose an unpenalized sparse regression procedure, which is formulated as

$$\min_{\boldsymbol{\beta}} f(\boldsymbol{\beta}) \text{ subject to } \boldsymbol{\beta} \text{ including } q \text{ zeros} \tag{2}$$

with q a pre-specified integer. We call (2) cardinality-constrained regression (CCREG).

An advantage of CCREG over (1) is that the tuning parameter q in (2) is restricted to an integer within the range [0, p–1], thus we can easily investigate suitability for all tuning parameter (q) values. On the other hand, that is difficult in (1), since λ can take any positive real value.

2. Cardinality-Constrained Regression

Using $s_{YY} = n^{-1}\mathbf{y}'\mathbf{y}$, $\mathbf{s}_{XY} = n^{-1}\mathbf{X}'\mathbf{y}$, and $\mathbf{S}_{XX} = n^{-1}\mathbf{X}'\mathbf{X}$, function $f(\boldsymbol{\beta})$ can be rewritten as $f(\boldsymbol{\beta}) = s_{YY} - 2\mathbf{s}_{XY}'\boldsymbol{\beta} + \boldsymbol{\beta}'\mathbf{S}_{XX}\boldsymbol{\beta}$. Followig (2002, (1), (11b)), we see that $f(\boldsymbol{\beta})$ is majorized by

$$m(\boldsymbol{\beta}) = c + \alpha\|\mathbf{b} - \boldsymbol{\beta}\|_2^2 \quad : \tag{3}$$

$f(\boldsymbol{\beta}) \leq m(\boldsymbol{\beta})$. Here, α is the maximum eigenvalue of $\mathbf{S}_{XX}$, $\mathbf{b} = \boldsymbol{\beta}^c - \alpha^{-1}(\mathbf{S}_{XX}\boldsymbol{\beta}^c - \mathbf{s}_{XY})$, with $\boldsymbol{\beta}^c$ the current $\boldsymbol{\beta}$, and c a constant with respect to $\boldsymbol{\beta}$ and defined so as to satisfy $m(\boldsymbol{\beta}^c) = f(\boldsymbol{\beta}^c)$. We can see that (3) is minimized when $\boldsymbol{\beta}$ is updated as

$$\boldsymbol{\beta}^u = \text{the vector } \mathbf{b} \text{ whose } q \text{ elements of the smallest absolute values are replaced by zeros,} \tag{4}$$

for given **b**. This fact, $f(\boldsymbol{\beta}) \leq m(\boldsymbol{\beta})$, and $m(\boldsymbol{\beta}^c) = f(\boldsymbol{\beta}^c)$ imply $f(\boldsymbol{\beta}^u) \leq m(\boldsymbol{\beta}^u) \leq m(\boldsymbol{\beta}^c) = f(\boldsymbol{\beta}^c)$. Hence updating $\boldsymbol{\beta}$ in this way will increase f or keep it equal. It leads to the CCREG algorithm, in which $\boldsymbol{\beta}^c$ is initialized, then setting $\mathbf{b} = \boldsymbol{\beta}^c - \alpha^{-1}(\mathbf{S}_{XX}\boldsymbol{\beta}^c - \mathbf{s}_{Xy})$ and updating $\boldsymbol{\beta}$ as (4) are alternately replicated, until convergence is reached.

Minimization (2) is performed for each of $q = 1, \ldots, p$–1. Among the resulting solutions $\hat{\boldsymbol{\beta}}$, we select the one with a suitable q. It can be given by $q^* = \mathrm{argmin}_K\, \mathrm{BIC}(q)$ with $\mathrm{BIC}(q) = n\log f(\hat{\boldsymbol{\beta}}) + (p - q + 1)\log n$.

3. Numerical Comparisons with Lasso

We synthesized 200 data matrices [**X**, **y**] (100 × 21) with $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_{\mathrm{true}} + \sigma\mathbf{e}$. Here, $\boldsymbol{\beta}_{\mathrm{true}}$ includes q_{true} zeros with $q_{\mathrm{true}} \in [5, 15]$ and each nonzero element of $\boldsymbol{\beta}_{\mathrm{true}}$ is drawn from the uniform distribution for [0.1, 0.9] or that for [–0.9, –0.1]. Each element of **e** and each row of **X** are sampled from normal distributions $N_1(0, 1)$ and $N_{20}(\mathbf{0}, \mathbf{V\Delta V}')$, respectively, where $\mathbf{V} \in 20 \times 20$ random orthonormal matrices and $\boldsymbol{\Delta} = \mathrm{diag}\{\delta_1, \ldots, \delta_{20}\}$, with $\delta_1 = 10$, $\delta_{20} = 2$, and the remaining δ_k chosen randomly from [2,10]. The error level σ is chosen so that $\|\sigma\mathbf{e}\|^2/\|\mathbf{y}\|^2 = 0.25$. The synthesized data were analyzed by CCREG and Lasso regression. Here, we also used BIC for Lasso: the solution with $\lambda = \mathrm{argmin}_{\lambda \in \Lambda} \mathrm{BIC}(\lambda)$ was selected, where $\Lambda = \{0.01, 0.02, \ldots, 9.99, 10\}$ thus covering a wide range of possible sparsenesses of $\boldsymbol{\beta}$.

As a result, CCREG was found to recover $\boldsymbol{\beta}_{\mathrm{true}}$ better than Lasso regression, with the averages of $p^{-1}\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_{\mathrm{true}}\|_1$ being .085 (sd = .045) for CCREG and .094 (sd = .043) for Lasso. Further, CCREG/BIC tended to overestimate q (by 1.2 on average), while Lasso/BIC tended to underestimate it (by –1.9 on average).

4. Conclusions

With CCREG it is easier to find the most suitable value of the tuning parameter, and it outperforms Lasso regression in the recovery of sparse coefficients.

Reference

Kiers, H. A. L. (2002). Setting up alternating least squares and iterative majorization algorithms for solving various matrix optimization problems. *Computational Statistics and Data Analysis*, **41**, 157-170.