Approximate Leave-One-Out with `glmnet`

Linyun He Wanchao Qin Peng Xu Yuze Zhou

August 29, 2018

1 ALO for Linear Regression

Recall the objective function for the elastic net problem:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\sum_{j=1}^{n}(\boldsymbol{x}_j^\top \boldsymbol{\beta} - y_j)^2 + \lambda\left(\alpha\|\boldsymbol{\beta}\|_1 + \frac{1-\alpha}{2}\|\boldsymbol{\beta}\|_2^2\right).$$

Let $E = \{i : \beta_i \notin K, i = 1, \ldots, p\}$ be the active set, the ALO formula is

$$\boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}^{\setminus j} \approx \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}} + \frac{\boldsymbol{H}_{ii}\left(\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}} - y_j\right)}{1-\boldsymbol{H}_{ii}}, \qquad \boldsymbol{H} = \boldsymbol{X}_{\cdot,E}\left[\boldsymbol{X}_{\cdot,E}^\top \boldsymbol{X}_{\cdot,E} + (1-\alpha)\,\lambda \boldsymbol{I}_{E,E}\right]^{-1}\boldsymbol{X}_{\cdot,E}^\top.$$

2 ALO for Logistic Regression

For binomial logistic regression, the primal problem is:

$$\min_{\boldsymbol{\beta}} \sum_{j=1}^{n}\left[\ln\left(1+e^{\boldsymbol{x}_j^\top \boldsymbol{\beta}}\right) - y_j \boldsymbol{x}_j^\top \boldsymbol{\beta}\right] + \lambda\left(\alpha\|\boldsymbol{\beta}\|_1 + \frac{1-\alpha}{2}\|\boldsymbol{\beta}\|_2^2\right).$$

Let E be the active set, the ALO formula is

$$\boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}^{\setminus j} \approx \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}} + \frac{\boldsymbol{H}_{ii}\left(1+e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}\right)\left[e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}} - y_j\left(1+e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}\right)\right]}{\left(1+e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}\right)^2 - \boldsymbol{H}_{ii}e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}},$$

where

$$\boldsymbol{H} = \boldsymbol{X}_{\cdot,E}\left[\boldsymbol{X}_{\cdot,E}^\top \operatorname{diag}\left(\frac{e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}}{1+2e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}} + e^{2\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}}\right)\boldsymbol{X}_{\cdot,E} + (1-\alpha)\lambda \boldsymbol{I}_{A,A}\right]^{-1}\boldsymbol{X}_{\cdot,E}^\top.$$

3 ALO for Poisson Regression

For Poisson regression, the primal problem is:

$$\min_{\boldsymbol{\beta}} \sum_{j=1}^{n} \left(e^{\boldsymbol{x}_j^\top \boldsymbol{\beta}} - y_j \boldsymbol{x}_j^\top \boldsymbol{\beta} \right) + \lambda \left(\alpha \|\boldsymbol{\beta}\|_1 + \frac{1-\alpha}{2} \|\boldsymbol{\beta}\|_2^2 \right).$$

Let E be the active set, the ALO formula is

$$\boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}^{\setminus j} \approx \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}} + \frac{\boldsymbol{H}_{ii} \left(e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}} - y_j \right)}{1 - \boldsymbol{H}_{ii} e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}}},$$

where

$$\boldsymbol{H} = \boldsymbol{X}_{\cdot,E} \left[\boldsymbol{X}_{\cdot,E}^\top \operatorname{diag}\left(e^{\boldsymbol{x}_j^\top \hat{\boldsymbol{\beta}}} \right) \boldsymbol{X}_{\cdot,E} + (1-\alpha)\lambda \boldsymbol{I}_{A,A} \right]^{-1} \boldsymbol{X}_{\cdot,E}^\top.$$

4 ALO for Multinomial Regression

Assume that the response variable comes in as an $n \times K$ matrix indicator matrix, where K is the number of classes. We re-parametrize by considering $\boldsymbol{\mathcal{B}} = \operatorname{vec}(\boldsymbol{B})$:

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_K \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} y_{11} \\ y_{12} \\ \vdots \\ y_{1K} \end{bmatrix} \\ \begin{bmatrix} y_{21} \\ y_{22} \\ \vdots \\ y_{2K} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} y_{n1} \\ y_{n2} \\ \vdots \\ y_{nK} \end{bmatrix} \end{bmatrix}, \qquad \boldsymbol{\mathcal{X}} = \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \\ \vdots \\ \boldsymbol{X}_K \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \boldsymbol{x}_1^\top & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_1^\top & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}_1^\top \end{bmatrix} \\ \begin{bmatrix} \boldsymbol{x}_2^\top & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_2^\top & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}_2^\top \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \boldsymbol{x}_n^\top & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_n^\top & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}_n^\top \end{bmatrix} \end{bmatrix}, \qquad \boldsymbol{\mathcal{B}} = \begin{bmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \vdots \\ \boldsymbol{\beta}_K \end{bmatrix}.$$

Again let E denote the active set. Further, let

$$\mathcal{A}(\mathcal{B}) \coloneqq \begin{bmatrix} \boldsymbol{A}_1(\mathcal{B}) \\ \boldsymbol{A}_2(\mathcal{B}) \\ \vdots \\ \boldsymbol{A}_n(\mathcal{B}) \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \frac{\exp\left(\boldsymbol{X}_1^\top \boldsymbol{\beta}_1\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_1^\top \boldsymbol{\beta}_k\right)} \\ \vdots \\ \frac{\exp\left(\boldsymbol{X}_1^\top \boldsymbol{\beta}_K\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_1^\top \boldsymbol{\beta}_k\right)} \end{bmatrix} \\ \begin{bmatrix} \frac{\exp\left(\boldsymbol{X}_2^\top \boldsymbol{\beta}_1\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_2^\top \boldsymbol{\beta}_k\right)} \\ \vdots \\ \frac{\exp\left(\boldsymbol{X}_2^\top \boldsymbol{\beta}_K\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_2^\top \boldsymbol{\beta}_k\right)} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \frac{\exp\left(\boldsymbol{X}_n^\top \boldsymbol{\beta}_1\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_n^\top \boldsymbol{\beta}_k\right)} \\ \vdots \\ \frac{\exp\left(\boldsymbol{X}_n^\top \boldsymbol{\beta}_K\right)}{\sum_{k=1}^K \exp\left(\boldsymbol{X}_n^\top \boldsymbol{\beta}_k\right)} \end{bmatrix} \end{bmatrix},$$

and

$$\mathcal{D}(\mathcal{B}) \coloneqq \begin{bmatrix} \left[\operatorname{diag}\left(\boldsymbol{A}_1(\mathcal{B})\right) - \boldsymbol{A}_1(\mathcal{B})\boldsymbol{A}_1(\mathcal{B})^\top\right] & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \left[\operatorname{diag}\left(\boldsymbol{A}_n(\mathcal{B})\right) - \boldsymbol{A}_n(\mathcal{B})\boldsymbol{A}_n(\mathcal{B})^\top\right] \end{bmatrix}.$$

Finally, define

$$\mathcal{K}(\mathcal{X}, \mathcal{B}) \coloneqq \mathcal{X}^\top \mathcal{D}(\mathcal{B}) \mathcal{X} + \nabla^2 R(\mathcal{B}), \qquad \mathcal{G}_{i,E}(\mathcal{X}, \mathcal{B}) \coloneqq \boldsymbol{X}_{i,E} \mathcal{K}(\mathcal{X}_{\cdot,E}, \hat{\mathcal{B}})^+ \boldsymbol{X}_{i,E}^\top.$$

Then, with Newton's method, we can approximate the leave-i-out prediction as

$$\begin{aligned} \boldsymbol{X}_i \tilde{\mathcal{B}}^{\setminus i} &= \boldsymbol{X}_i \hat{\mathcal{B}} + \mathcal{G}_{i,E}(\mathcal{X}, \mathcal{B}) \left(\boldsymbol{A}_i(\hat{\mathcal{B}}) - \boldsymbol{y}_i\right) \\ &\quad - \mathcal{G}_{i,E}(\mathcal{X}, \hat{\mathcal{B}}) \left\{ \mathcal{G}_{i,E}(\mathcal{X}, \hat{\mathcal{B}}) - \left[\operatorname{diag}\left(\boldsymbol{A}_i(\hat{\mathcal{B}})\right) - \boldsymbol{A}_i(\hat{\mathcal{B}})\boldsymbol{A}_i(\hat{\mathcal{B}})^\top\right]^+ \right\}^+ \mathcal{G}_{i,E}(\mathcal{X}, \hat{\mathcal{B}}) \left(\boldsymbol{A}_i(\hat{\mathcal{B}}) - \boldsymbol{y}_i\right) \end{aligned}$$

5 ALO with Intercept

Including the intercept is straightforward. As we can augment $\boldsymbol{X}$ with an extra column of 1s, i.e. $\boldsymbol{X}^* = [\mathbf{1}_n, \boldsymbol{X}]$. Since the intercept is not reugularized, we need to change the corresponding second partial derivatives to 0, e.g.

$$\nabla^2 R\left(\hat{\beta}_0, \hat{\boldsymbol{\beta}}_A\right) = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & (1-\alpha)\lambda & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & (1-\alpha)\lambda \end{bmatrix}.$$

For multinomial model it can be a bit more complicated since there are now K intercepts. For programming convenience we augment $\boldsymbol{\mathcal{X}}$ by block of $\boldsymbol{I}_K$ and stack the intercepts on tops of $\boldsymbol{\mathcal{B}}$, i.e.

$$\boldsymbol{\mathcal{X}}^* = \begin{bmatrix} \begin{bmatrix} \boldsymbol{I}_K & \boldsymbol{X}_1 \end{bmatrix} \\ \begin{bmatrix} \boldsymbol{I}_K & \boldsymbol{X}_2 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \boldsymbol{I}_K & \boldsymbol{X}_K \end{bmatrix} \end{bmatrix}, \qquad \boldsymbol{\mathcal{B}}^* = \begin{bmatrix} \beta_{01} \\ \vdots \\ \beta_{0K} \\ \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_K \end{bmatrix}.$$

Accordingly, the first K diagonal elements of $\nabla^2 R$ will be set to 0.

6 Usage of ALO formulae with `glmnet` package

The `glmnet` package scales the elastic net loss function by a factor of $1/n$. Furthermore, for linear problems `glmnet` implicitly "standardizes y to have unit variance before computing its λ sequence (and then unstandardizes the resulting coefficients)". So it is necessary to rescale $\boldsymbol{y}$ by the MLE $\hat{\sigma}_y$ before fitting the model. For instance, `glmnet` is in fact optimizing the following problem for linear regression (assuming $\boldsymbol{X}$ is already standardized):

$$\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{j=1}^{n} \left(\frac{\boldsymbol{x}_j^\top \boldsymbol{\beta}}{\hat{\sigma}_y} - \frac{y_j}{\hat{\sigma}_y} \right)^2 + \frac{\lambda}{\hat{\sigma}_y} \alpha \|\boldsymbol{\beta}\|_1 + \frac{\lambda}{\hat{\sigma}_y^2} \frac{1-\alpha}{2} \|\boldsymbol{\beta}\|_2^2. \tag{1}$$

We thus have

$$\dot{\ell}(\boldsymbol{x}_j^\top \boldsymbol{\beta};\, y_j) = \frac{\boldsymbol{x}_j^\top \boldsymbol{\beta}}{n\hat{\sigma}_y} - \frac{y_j}{n\hat{\sigma}_y}, \qquad \ddot{\ell}(\boldsymbol{x}_j^\top \boldsymbol{\beta};\, y_j) = \frac{1}{n\hat{\sigma}_y}, \qquad \nabla^2 R(\hat{\boldsymbol{\beta}}_A) = \frac{(1-\alpha)\lambda}{\hat{\sigma}_y^2} \boldsymbol{I}_{A,A}.$$

Hence, for the linear elastic net problem, the primal ALO is:

$$\tilde{y}_j^{\setminus i} = \hat{y}_j + \frac{\boldsymbol{H}_{ii}(\hat{y}_j - y_j)}{n\hat{\sigma}_y - \boldsymbol{H}_{ii}}, \qquad \boldsymbol{H} = \boldsymbol{X}_{\cdot,E} \left[\frac{1}{n\hat{\sigma}_y} \boldsymbol{X}_{\cdot,E}^\top \boldsymbol{X}_{\cdot,E} + \frac{(1-\alpha)\,\lambda}{\hat{\sigma}_y^2} \boldsymbol{I}_{A,A} \right]^{-1} \boldsymbol{X}_{\cdot,E}^\top.$$

Further complications present when option `standardization = T` is given, in which case `glmnet` first standardize the data $\boldsymbol{X}$ using $\hat{\sigma}_{\boldsymbol{X}}$:

- If `intercept = F`, compute $\boldsymbol{X}^* = \mathrm{diag}[\hat{\sigma}_y \hat{\sigma}_{\boldsymbol{X}}]^{-1} \boldsymbol{X}$.
- If `intercept = T`, compute $\boldsymbol{X}^* = \mathrm{diag}[\hat{\sigma}_y \hat{\sigma}_{\boldsymbol{X}}]^{-1} (\boldsymbol{X} - \bar{\boldsymbol{X}} \mathbf{1}\mathbf{1}^\top)$.

Afterwards, the the coefficients are returned unstandardized, i.e. let $(\beta_0, \boldsymbol{\beta})$ denotes the original intercept and coefficients, `glmnet` reports

$$\boldsymbol{\beta}^* = \hat{\sigma}_y \,\mathrm{diag}[\hat{\sigma}_{\boldsymbol{X}}]^{-1} \boldsymbol{\beta}, \qquad \beta_0^* = \beta_0 - \bar{\boldsymbol{X}} \boldsymbol{\beta}^*.$$

For logistics and Poisson regression the standardization procedure is basically the same, except `glmnet` no longer standardize by $\hat{\sigma}_y$, which make sense since $\boldsymbol{y}$ is now either categorical or count data.

7 Benchmark

n	p	k	Average ALO	Average 5-fold CV	Relative	n full fit
300	100	60	0.016	0.053	3.313	1.2
500	800	500	0.251	0.533	2.124	36.5
1000	1200	800	0.489	1.200	2.454	211.0
2500	2000	1200	2.267	3.623	1.598	1577.5
5000	2500	2000	5.017	8.097	1.614	7740.0
10000	10000	2500	27.236	36.520	1.341	62530.0

Table 1: Averaged (over 10 runs) elapsed time (in seconds) comparison, 25 λs, $\alpha = 0.5$.

References

[1] Trevor Hastie & Junyang Qian, *Glmnet Vignette*.