

Concomitant Huber

Leo

January 2020

A quick idea of concomitant huber numerical method

We can have a new formulation of concomitant huber.

Proposition 0.1.

$$\min_{\sigma > 0, \beta_1, \beta_2 \in \mathbb{R}^{d+m} \text{ } C \beta_1 = 0} \frac{\|X\beta_1 + \frac{\lambda}{2\rho}\beta_2 - y\|_2^2}{\sigma} + e\sigma + \lambda\|\beta_1, \beta_2\|_1 = \min_{\sigma > 0, \beta \in \mathbb{R}^d \text{ } C\beta = 0} h_\rho\left(\frac{X\beta - y}{\sigma}\right)\sigma + e.\sigma + \lambda\|\beta\|_1 \quad (1)$$

Consequence : Then to solve the concomitant huber problem, one only have to solve the problem :

$$Concomitant(\tilde{X}(\lambda, \rho), \tilde{y}, \lambda) \quad \text{with } \tilde{X}(\lambda, \rho) = (X, \frac{\lambda}{2\rho}I_n)$$

which is fast thanks to path algorithm.

Improvement : As the matrix depends on λ , it is not possible to compute the path for concomitant huber only with this method : it only gives the solution for a given λ . But maybe there is a way to do it. More specifically : firstly, it is possible that we can use KKT conditions to identify the dependance w.r.t. λ . But in a more easy way, it is possible that we can find a way to adapt this method in order to do warm start

Proof. We can use a known lemma that is true for any matrices X' and y' :

$$\min_{\beta, o \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \|X'\beta + o - y'\|_2^2 + \lambda \|\beta\|_1 + 2\rho \|o\|_1 = \min_{\beta \in \mathbb{R}^d} \min_{C\beta=0} h_\rho(y - A\beta) + \lambda \|\beta\|_1$$

Then we can take the formulation of the concomitant huber and manipulate it :

$$\begin{aligned} & \min_{\sigma > 0, \beta \in \mathbb{R}^d} \min_{C\beta=0} h_\rho\left(\frac{X\beta - y}{\sigma}\right) \sigma + e \cdot \sigma + \lambda \|\beta\|_1 \\ &= \min_{\sigma > 0} \sigma \left(e + \min_{\beta \in \mathbb{R}^d} \min_{C\beta=0} h_\rho\left(\frac{X\beta - y}{\sigma}\right) + \frac{\lambda}{\sigma} \|\beta\|_1 \right) \\ &= \min_{\sigma > 0} \sigma \left(e + \min_{\beta \in \mathbb{R}^d} \min_{C\beta=0} h_\rho\left(\frac{X}{\sigma}\beta - \frac{y}{\sigma}\right) + \frac{\lambda}{\sigma} \|\beta\|_1 \right) \\ &= \min_{\sigma > 0} \sigma \left(e + \min_{\beta, o \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \left\| \frac{X}{\sigma}\beta + o - \frac{y}{\sigma} \right\|_2^2 + \frac{\lambda}{\sigma} \|\beta\|_1 + 2\rho \|o\|_1 \right) \\ &= \min_{\sigma > 0} \left(\sigma e + \min_{\beta, o \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \sigma \left\| \frac{X}{\sigma}\beta + o - \frac{y}{\sigma} \right\|_2^2 + \lambda \|\beta\|_1 + 2\rho \sigma \|o\|_1 \right) \\ &= \min_{\sigma > 0} \left(\sigma e + \min_{\beta, o \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \frac{\|X\beta + \sigma o - y\|_2^2}{\sigma} + \lambda \|\beta\|_1 + 2\rho \sigma \|o\|_1 \right) \\ &= \min_{\sigma > 0} \left(\sigma e + \min_{\beta, \beta_2 \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \frac{\|X\beta + \frac{\lambda}{2\rho}\beta_2 - y\|_2^2}{\sigma} + \lambda \|\beta\|_1 + \lambda \|\beta_2\|_1 \right) \\ &= \min_{\sigma > 0, \beta, \beta_2 \in \mathbb{R}^d \times \mathbb{R}^m} \min_{C\beta=0} \left(\sigma e + \frac{\|X\beta + \frac{\lambda}{2\rho}\beta_2 - y\|_2^2}{\sigma} + \lambda \|\beta\|_1 + \lambda \|\beta_2\|_1 \right) \end{aligned}$$

□

Path algorithm :

Let

$$F(\beta_1, \beta_2, \sigma, \lambda) := \frac{\|X\beta_1 + \frac{\lambda}{2\rho}\beta_2 - y\|_2^2}{\sigma} + e\sigma + \lambda \|\beta_1, \beta_2\|_1$$

We now want to find

$$\hat{\beta}_1, \hat{\beta}_2, \hat{\sigma} := \min_{\sigma > 0, \beta_1, \beta_2 \in \mathbb{R}^{d+m}}^{-1} \min_{C\beta_1=0} F(\beta_1, \beta_2, \sigma, \lambda) \quad \forall \lambda \in PATH$$

We can then think about an algorithm :

$$\hat{\beta}_1^{k+1} = \min_{C\beta_1=0}^{-1} F(\beta_1, \hat{\beta}_2^k, \hat{\sigma}^k, \lambda) =: f(\hat{\beta}_2^k, \hat{\sigma}^k, \lambda) \quad (2)$$

$$\hat{\beta}_2^{k+1} = \min_{\beta_2}^{-1} F(\hat{\beta}_1^{k+1}, \beta_2, \hat{\sigma}^k, \lambda) =: g(\hat{\beta}_1^{k+1}, \hat{\sigma}^k, \lambda^{k+1}) \quad (3)$$

$$\hat{\sigma}^{k+1} = \min_{\sigma > 0}^{-1} F(\hat{\beta}_1^{k+1}, \hat{\beta}_2^{k+1}, \sigma, \lambda) =: h(\hat{\beta}_1^{k+1}, \hat{\beta}_2^{k+1}, \lambda) \quad (4)$$

With this algorithm one can then do warm start when we modify λ !
Now let's express f, g, h :

$$f(\beta_2, \sigma, \lambda) = \text{lasso}(X, y - \frac{\lambda}{2\rho}\beta_2, \sigma\lambda)$$

$$\begin{aligned} g(\beta_1, \sigma, \lambda) &= \text{lasso}(\frac{\lambda}{2\rho}I_n, y - X\beta_1, \sigma\lambda) \\ &= \text{lasso}(I_n, \frac{2\rho}{\lambda}(y - X\beta_1), \frac{4\rho^2\sigma}{\lambda}) \\ &= \text{prox}_{\gamma\|\cdot\|_1}(\frac{2\rho}{\lambda}(y - X\beta_1)) \quad \text{with } \gamma = \frac{2\rho^2\sigma}{\lambda} \end{aligned}$$

$$h(\beta_1, \beta_2, \lambda) = \frac{\|X\beta_1 + \frac{\lambda}{2\rho}\beta_2 - y\|_2}{\sqrt{e}}$$

Good news : g and h are easy to compute !!

Question : Which order is the best for optimization ? The question could be :

Doing for instance $(2) \rightarrow (3) \rightarrow (4)$ or $(2) \rightarrow (4) \rightarrow (3)$ OR even, depending on the and running time and improvement of each step, doing for instance : $(3) \rightarrow (4) \rightarrow (3) \rightarrow (4) \rightarrow (3) \rightarrow (4) \rightarrow (3) \rightarrow (4) \rightarrow (2)$