Solving Statistical Optimization Problems: Second Order Methods with First Order Complexities or Less

Defeng Sun

Department of Applied Mathematics

The Hong Kong Polytechnic University

Shanghai Jiaotong University
June 9, 2020
Second Order Methods

with

First Order Costs or Less per Step
Our approach — Newton’s methods (second order methods) with low costs [can even be lower than first order methods]

Figure: Sir Isaac Newton (Niu Dun) (4 January 1643 - 31 March 1727)
Two classical conditions for Newton’s methods

1. Let $\mathcal{X}$ be a finite-dimensional real Euclidean space
2. $F : \mathcal{X} \to \mathcal{X}$ is continuously differentiable, e.g., $F(\cdot) = \nabla f(\cdot)$, (Condition 1)

Assume that $F(\bar{x}) = 0$.

Given $x^0 \in \mathcal{X}$. For $k = 0, 1, \ldots$

Main Step  Let $V_k := F'(x^k)$. Solve

$$F(x^k) + V_k(x^{k+1} - x^k) = 0$$

Rates of Convergence: Assume that $F'(\bar{x})$ is nonsingular (Condition 2) and that $x^0$ is sufficiently close to $\bar{x}$. Then

$$\|x^{k+1} - \bar{x}\| = \|V_k^{-1} \left[ F(x^k) - F(\bar{x}) - V_k(x^k - \bar{x}) \right]\| = o(\|x^k - \bar{x}\|).$$

The right hand side takes $O(\|x^k - \bar{x}\|^2)$ if $F'(\cdot)$ is Lipschitz continuous at $\bar{x}$. 
The statistical optimization problem setting

Let \( X \in \mathbb{R}^{m \times n} \) be the input data and \( b \) be the response variables with a noise vector \( \varepsilon = b - X\beta \). Let \( \lambda > 0 \). One of the most commonly used models to control the overfitting and/or variable selection is the Lasso model

\[
\min_{\beta \in \mathbb{R}^n} \left\{ \frac{1}{2} \|X\beta - b\|^2 + \lambda \|\beta\|_1 \right\}
\]

which relies on knowing the standard deviation of the noise. Here in the convex case we are interested in the more general model

\[
\min_{\beta \in \mathbb{R}^n} \left\{ \underbrace{h(X\beta) + p(\beta)}_{f(\beta)} \right\}
\]

where both \( h(\cdot) \) and \( p(\cdot) \) are proper and closed convex functions (i.e., their respective epi-graphs are non-empty closed convex sets), which can be nonsmooth or non-Lipschitzian. Here, \( h \) is not assumed to be differentiable!!!
One interesting example is the square-root Lasso (srLasso) model (Alex Belloni et al. (2011))

$$\min_{\beta \in \mathbb{R}^n} \{ \| X \beta - b \| + \lambda \| \beta \|_1 \}$$

which is equivalent to the robust least regression (Huan Xu et al. 2010)

$$\min_{\beta \in \mathbb{R}^n} \left\{ \max_{\Delta X \in U} \| b - (X + \Delta X)\beta \| \right\}$$

with the uncertainty set

$$U := \{(\Delta_1, \ldots, \Delta_n) \mid \| \Delta_i \| \leq \lambda, \ i = 1, \ldots, n\}$$

Another commonly used example is the constrained Lasso model

$$\min_{\beta \in \mathbb{R}^n} \{ \| \beta \|_1 \mid \| X \beta - b \| \leq \tau \}.$$  

Here $h(\cdot)$ is the indicator function

$$h(y) := \delta_{B^\tau}(y) \quad \forall \ y \in \mathbb{R}^m$$

over the ball $B^\tau := \{ y \in \mathbb{R}^m \mid \| y \| \leq \tau \}$ centered at 0 with radius $\tau > 0$.
There are many more convex Lasso-type models:

\[
\text{(LASSO)}
\quad \min \left\{ \frac{1}{2} \| X \beta - b \|^2 + \lambda \| \beta \|_1 \mid \beta \in \mathbb{R}^n \right\}
\]

where \( \lambda > 0 \).

\[
\text{(Fused LASSO)}
\quad \min \left\{ \frac{1}{2} \| X \beta - b \|^2 + \lambda \| \beta \|_1 + \lambda_2 \| B \beta \|_1 \right\}
\]

\[
B = \begin{pmatrix}
1 & -1 & -1 \\
1 & -1 & \ddots & \ddots \\
& \ddots & \ddots & 1 & -1
\end{pmatrix}
\]
More Lasso-type problems (continued)

(Clustered LASSO)

$$\min \left\{ \frac{1}{2}\|X\beta - b\|^2 + \lambda \|\beta\|_1 + \lambda_2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} |\beta_i - \beta_j| \right\}$$

Note that the above problem is not numerically solvable if $n$ is large as the objective function value computation itself would cost $O(n^2)$ flops.

Fortunately, [Lin-Liu-S.-Toh, SIOPT 2019] showed

$$\sum_{1 \leq i < j \leq n} |\beta_i - \beta_j| = \langle w, \beta^{\downarrow} \rangle,$$

where $\beta^{\downarrow}$ is the vector whose components are those of $\beta$ sorted in a non-increasing order, i.e. $\beta_1^{\downarrow} \geq \beta_2^{\downarrow} \geq \cdots \geq \beta_n^{\downarrow}$ [costs $O(n \log n)$] and the weight vector $w \in \mathbb{R}^n$ is defined by

$$w_k = n - 2k + 1, \ k = 1, \cdots, n.$$
(Gaussian graphical models with hidden clustering structure)

\[
\min_{S^n \supseteq \Omega \succeq 0} \left\{ \langle C, \Omega \rangle - \log \det(\Omega) + \rho \sum_{i<j} |\Omega_{ij}| + \lambda \sum_{i<j} \sum_{s<t} |\Omega_{ij} - \Omega_{st}| \right. \\
\left. \quad |\Omega_{ij} = 0, (i, j) \in J \right\},
\]

where \( J \) is the set of pairs of nodes \((i, j)\) such that \( z_i \) and \( z_j \) are known to be conditionally independent and parameters \( \lambda > 0 \) and \( \rho > 0 \).

Note that the last term in the objective function would cost \( O(n^4) \) flops. Using the above sorting idea, one would only need \( O(n^2 \log n) \) flops, which are vital for solving large scale problems.
(SLOPE, Ordered Lasso)

\[
\min \left\{ \frac{1}{2} \| X \beta - b \|_2^2 + \sum_{i=1}^{n} \lambda_i | \beta |_{i}^{+} \right\}
\]

with parameters \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \) and \( \lambda_1 > 0 \).

We are interested in \( n \) (number of features) large and/or \( m \) (number of samples) large. Note that the regularization term in SLOPE (ordered Lasso) is not separable.

The term \( \| X \beta - b \|_2^2 \) in various models can be replaced by \( \| X \beta - b \| \).
More on the loss functions

In the loss function part, $f$ can be the logistic regression function, defined as below: given $b \in \mathbb{R}^m$ and $X \in \mathbb{R}^{m \times n}$,

$$f(\beta) = \sum_{i=1}^{m} \log(1 + \exp(-b_i(X\beta)_i)) \quad (1)$$

- Define $h : \mathbb{R}^m \rightarrow \mathbb{R}$ as follows

$$h(z) = \sum_{i=1}^{m} \log(1 + \exp(-b_iz_i)) \quad \forall z \in \mathbb{R}^m$$

The function $f$ defined by (1) can be written as

$$f(\beta) = h(X\beta)$$
In general, the loss function $f : \mathbb{R}^{pK} \rightarrow \mathbb{R}$ can take as the multinomial logistic regression function: given $A := (A_1, \ldots, A_N)^T \in \mathbb{R}^{N \times p}$,

$$f(\beta) = -\sum_{i=1}^{N} \left( \sum_{k=1}^{K} y_{ik} A_i^T \beta_k - \log \sum_{k=1}^{K} \exp(A_i^T \beta_k) \right)$$  \hspace{2cm} (2)$$

Define $h : \mathbb{R}^{NK} \rightarrow \mathbb{R}$ as follows:

$$h(z) := -\sum_{i=1}^{N} \left( \sum_{k=1}^{K} y_{ik} Z_{ik} - \log \sum_{k=1}^{K} \exp(Z_{ik}) \right), \text{ with } Z := \text{mat}(z)$$

Then the function $f$ defined by (2) can be written as

$$f(\beta) := h(X\beta), \text{ with } X := I \otimes A \in \mathbb{R}^{NK \times pK}$$
A proper nonconvex regularization can achieve a sparse estimation with fewer measurements, faster convergence and more robust against noises.

In this talk, we aim to develop an efficient and robust algorithm for solving the following nonconvex problem \((P)\):

\[
\min_{\beta \in \mathbb{R}^n} \left\{ g(\beta) := h(X\beta) + p(\beta) - q(\beta) \right\}
\]

Here \(p : \mathbb{R}^n \to (-\infty, +\infty]\) is a proper closed convex function and \(q : \mathbb{R}^n \to \mathbb{R}\) is a finite-valued (smooth, not essential) convex function. Moreover, we require the proximal functions of \(h\) and \(p\) to be (strongly) semismooth (to be explained shortly).
For \( \lambda > 0 \), the SCAD regularization is defined by \( r(\beta) = p(\beta) - q(\beta) \) with

\[
p(\beta) = \lambda \| \beta \|_1
\]

\[
q(\beta) = \sum_{i=1}^{n} \begin{cases} 
0, & \text{if } |\beta_i| < \lambda \\
\frac{(\|\beta_i\| - \lambda)^2}{2(a_s - 1)}, & \text{if } \lambda \leq |\beta_i| \leq a_s \lambda \\
\lambda |\beta_i| - \frac{a_s + 1}{2} \lambda^2, & \text{if } |\beta_i| > a_s \lambda
\end{cases}
\]

Note that \( q(\cdot) \) is continuously differentiable. In our numerical experiments, we take \( a_s = 3.7 \).

There are many other non-convex regularizers. Almost all good ones have a kind of “DC” (difference of convex functions) structure.
A basic tool: The Moreau-Yosida regularization

Recall that the objective function in our setting is not differentiable or even not continuous. So one even cannot use gradient methods directly, let alone Newton’s method [one basic conditions of Newton’s method for solving unconstrained optimization problems is that the gradient of the objective function is $C^1$].

We need several key steps to use Newton’s method. The first important tool is the Moreau-Yosida regularization of a convex function. Recall that for any $t > 0$, $e_t f(\cdot)$ is the Moreau envelope of a closed proper convex function $f : \mathcal{X} \to (-\infty, \infty]$, associated with $t$, given by

$$
e_t f(x) := \min_{z \in \mathbb{R}^n} \left\{ f(z) + \frac{t}{2} \|z - x\|^2 \right\}, \quad \forall \ x \in \mathbb{R}^n. \tag{4}$$

Here $e_t f(\cdot)$ is continuously differentiable with

$$\nabla e_t f(x) = t[x - P_t f(x)], \quad \forall \ x \in \mathbb{R}^n,$$

where $P_t f(x)$ is the unique optimal solution to problem (8). $P_t f(\cdot)$, called the proximal mapping of $f$, is globally Lipschitz continuous with modulus 1.
Semismooth functions

The second key tool is nonsmooth Newton’s method.

1. Let $\mathcal{X}, \mathcal{Y}$ be two finite-dimensional real Euclidean spaces
2. $F : \mathcal{X} \to \mathcal{Y}$ a locally Lipschitz continuous function.

Since $F$ is almost everywhere differentiable [Rademacher, 1912], we can define

$$\partial_B F(x) := \left\{ \lim F'(x^k) : x^k \to x, x^k \in D_F \right\}.$$

Here $D_F$ is the set of points where $F$ is differentiable. Hence, Clarke’s generalized Jacobian of $F$ at $x$ is given by

$$\partial F(x) = \text{conv} \partial_B F(x).$$

Note that if $f : \mathcal{X} \to (-\infty, \infty]$ is a closed proper convex function, then for any $t > 0$, one can define Clarke’s generalized Jacobian for the proximal mapping $P_t f(\cdot)$ of $f$ even if $f$ itself is not continuous – What a beauty!
**Definition**

Let $\mathcal{K} : \mathcal{X} \ni \mathcal{L}(\mathcal{X}, \mathcal{Y})$ be a nonempty, compact valued and upper-semicontinuous multifunction. We say that $F$ is **semismooth** $x \in \mathcal{X}$ with respect to $\mathcal{K}$ if (i) $F$ is directionally differentiable at $x$; and (ii) for any $\Delta x \in \mathcal{X}$ and $V \in \mathcal{K}(x + \Delta x)$ with $\Delta x \to 0$,

$$F(x + \Delta x) - F(x) - V(\Delta x) = o(\|\Delta x\|) \quad (g-\text{semismooth}).$$

(5)

Furthermore, if (5) is replaced by

$$F(x + \Delta x) - F(x) - V(\Delta x) = O(\|\Delta x\|^{1+\gamma}),$$

(6)

where $\gamma > 0$ is a constant, then $F$ is said to be $\gamma$-order (strongly if $\gamma = 1$) semismooth at $x$ with respect to $\mathcal{K}$. 

17
Nonsmooth (local) Newton’s method

Assume that \( F(\bar{x}) = 0 \).

Given \( x^0 \in \mathcal{X} \). For \( k = 0, 1, \ldots \)

**Main Step**  Choose an arbitrary \( V_k \in \mathcal{K}(x^k) \). Solve

\[
F(x^k) + V_k(x^{k+1} - x^k) = 0
\]

**Rates of Convergence:** Assume that \( \mathcal{K}(\bar{x}) \) is nonsingular and that \( x^0 \) is sufficiently close to \( \bar{x} \). If \( F \) is \( g \)-semismooth at \( \bar{x} \), then

\[
\|x^{k+1} - \bar{x}\| = \| V_k^{-1} \left[ F(x^k) - F(\bar{x}) - V_k(x^k - \bar{x}) \right] \| = o(\|x^k - \bar{x}\|) .
\]

It takes \( o(\|x^k - \bar{x}\|^{1+\gamma}) \) if \( F \) is \( \gamma \)-order \( g \)-semismooth at \( \bar{x} \) [the directional differentiability of \( F \) is not needed in the above local convergence analysis]

The condition that \( \mathcal{K}(\bar{x}) \) is nonsingular implies that \( \bar{x} \) is an isolated solution of \( F(\cdot) = 0 \) – A big caution!
A brief history

1. The nonsmooth equation approach is popular in the complementarity and variational inequalities (nonsmooth equations) community (Robinson, Pang, ...).


5. L. Qi and J. Sun (1993) proved that semismooth functions satisfy Kummer’s condition — a big leap in the extension from piecewise smoothness to potentially infinitely many smooth pieces.

6. Since then, many exciting developments, in particular in the large-scale settings ...

Why are nonsmooth Newton methods important in solving large-scale statistical optimization problems?
The third important tool is the majorized proximal point dual Newton algorithm (mPPDNA).

We consider the following problem [e.g., the srLasso problem for example]

$$\min_{\beta \in \mathbb{R}^n} \left\{ h(X\beta) + p(\beta) - q(\beta) \right\}$$

- In Stage 1, replace $q$ by its linear approximation at the origin [when $q(0) = 0$ and $0 \in \partial q(0)$, which hold true for many interesting cases of $q$, we just delete $q$ from the original problem] and add “proper” proximal terms to obtain an initial point for the second stage.

- In Stage 2, a series of majorized proximal subproblems are solved to obtain an approximate solution point.
The subproblem

Given $\sigma > 0$, $\tau > 0$, $\tilde{\beta} \in \mathbb{R}^n$, $\tilde{v} \in \mathbb{R}^n$, $\tilde{b} \in \mathbb{R}^m$, in our main algorithm (mPPDNA) we need to solve the following minimization subproblem

$$
\min_{\beta \in \mathbb{R}^n} \left\{ g(\beta; \sigma, \tau, \tilde{\beta}, \tilde{v}, \tilde{b}) := \begin{cases} h(X\beta) + p(\beta) - (q(\tilde{\beta}) + \langle \tilde{v}, \beta - \tilde{\beta} \rangle) \\ \text{convex} \\ \text{linear} \end{cases} + \frac{\sigma}{2} \| \beta - \tilde{\beta} \|^2 + \frac{\tau}{2} \| X\beta - \tilde{b} \|^2 \right\} \tag{7}
$$

Here, $\tilde{v} \in \partial q(\tilde{\beta})$. Obviously, $g(\cdot; \sigma, \tau, \tilde{\beta}, \tilde{v}, \tilde{b})$ is a strongly convex function albeit nonsmooth or non-Lipschitzian.

- The big question is how one can solve (7) in a fast and robust way!!!

- For the convex case: $q \equiv 0$

- If $h$ is strongly convex, we can take $\tau = 0$ though not necessary [The algorithm is called SSNAL if $\tau = 0$ and $q(\cdot) \equiv 0$ due to its equivalence to the semismooth Newton augmented Lagrangian method]
The dual of the subproblem

The dual of (7), after converting it into the minimization form and ignoring the constant term, is

$$\min_{u \in \mathbb{R}^m} \left\{ \phi(u; \sigma, \tau) := \frac{\tau}{2} \| \tilde{b} + \tau^{-1} u \|^2 - e_\tau h(\tilde{b} + \tau^{-1} u) + \frac{\sigma}{2} \| \tilde{\beta} + \sigma^{-1} (\tilde{v} - X^* u) \|^2 - e_\sigma p(\tilde{\beta} + \sigma^{-1} (\tilde{v} - X^* u)) \right\}. $$

Recall that for any $t > 0$, $e_t f(\cdot)$ is the Moreau envelope of a closed proper convex function $f$, associated with $t$, given by

$$e_t f(x) := \min_{z \in \mathbb{R}^n} \left\{ f(z) + \frac{t}{2} \| z - x \|^2 \right\}, \quad \forall \ x \in \mathbb{R}^n. \quad (8)$$

Here $e_t f(\cdot)$ is continuously differentiable with

$$\nabla e_t f(x) = t[x - P_t f(x)], \quad \forall \ x \in \mathbb{R}^n,$$

where $P_t f(x)$ is the unique optimal solution to problem (8).
A semismooth Newton method for the subproblem

We shall apply the superlinearly (quadratically) convergent sparse semismooth Newton method to find the solution $\bar{u}$ of the nonsmooth equations

$$\nabla \phi(u; \sigma, \tau) = P_{\tau} h(\tilde{b} + \tau^{-1} u) - XP_{\sigma} p(\tilde{\beta} + \sigma^{-1}(\tilde{v} - X^* u)) = 0.$$ 

Then the unique optimal solution $\bar{\beta}$ to problem (7) is

$$\bar{\beta} = P_{\sigma} p(\tilde{\beta} + \sigma^{-1}(\tilde{v} - X^* \bar{u})).$$

Proposition

*Suppose that problem (7) is nondegenerate, which holds true if $f(\cdot) \equiv h(X \cdot)$ is continuously differentiable near $\bar{\beta}$ (this is the no-overfitting assumption for the squared root Lasso problem). Then all the elements in Clarke’s generalized Hessian $\partial^2 \phi(\bar{u})$ are self-adjoint and positive definite.*
A globalized SSN algorithm for the subproblem (7)

Algorithm SSN (SSN(\(\sigma, \tau\))): Given \(\mu \in (0, \frac{1}{2})\), \(\bar{\eta} \in (0, 1)\), \(\bar{\tau} \in (0, 1]\), \(\nu_1\), \(\nu_2 \in (0, 1]\), and \(\delta \in (0, 1]\), choose \(u^0 \in \mathbb{R}^m\). Set \(j = 0\) and iterate the following steps.

1. Choose \(V^j \in \partial P_{\tau h}(\bar{b} + \tau^{-1}u^j)\) and \(U^j \in \partial P_{\sigma p}(\bar{\beta} + \sigma^{-1}(\bar{v} - X^*u^j))\).

   Let \(H^j = \tau^{-1}V^j + \sigma^{-1}XU^jX^*\) and find the exact solution \(\Delta u^j\) or apply the PCG method to find an approximate solution \(\Delta u^j\) to

   \[
   (H^j + \varepsilon_j I)\Delta u = -\nabla \phi(u^j; \sigma, \tau)
   \]

   such that

   \[
   \|H^j \Delta u^j + \nabla \phi(u^j; \sigma, \tau)\| \leq \eta_j := \min(\bar{\eta}, \|\nabla \phi(u^j; \sigma, \tau)\|^{1+\bar{\tau}})
   \]

   where \(\varepsilon_j := \nu_1 \min \{\nu_2, \|\nabla \phi(u^j; \sigma, \tau)\|\}\)

2. Set \(\alpha_j = \delta^{l_j}\), where \(l_j\) is the first nonnegative integer \(l\) for which

   \[
   \phi(u^j + \delta^l \Delta u^j; \sigma, \tau) \leq \phi(u^j; \sigma, \tau) + \mu \delta^l \langle \nabla \phi(u^j; \sigma, \tau), (\Delta u^j) \rangle
   \]

3. Set \(u^{j+1} = u^j + \alpha_j \Delta u^j\)
Theorem

Assume that $P_\tau h(\cdot)$ and $P_\sigma p(\cdot)$ are strongly semismooth. If problem (7) is nondegenerate, in particular if $f(\cdot) \equiv h(X\cdot)$ is continuously differentiable near $\overline{\beta}$, then $\{u^j\}$ converges to the unique optimal solution $\overline{u}$ and

$$\|u^{j+1} - \overline{u}\| = O(\|u^j - \overline{u}\|^{1+\bar{\tau}}).$$

Note that if $\bar{\tau} = 1$, we get the quadratic convergence.
Algorithm. Let $\sigma^0, \sigma^1 > 0, \tau^0, \tau^1 > 0$ be given parameters

1. Compute

$$\beta^1 \approx \arg\min_{\beta \in \mathbb{R}^n} \{g(\beta; \sigma^0, \tau^0, 0, 0, b)\}$$

via solving its dual problem such that a prescribed stopping criterion is satisfied. Let $k = 1$ and go to Step 2.1.

2.1 Choose $v^k \in \partial q(\beta^k)$ and compute

$$\beta^{k+1} = \arg\min_{\beta \in \mathbb{R}^n} \{g(\beta; \sigma^k, \tau^k, \beta^k, v^k, X\beta^k) + \langle \delta^k, \beta - \beta^k \rangle\}$$

via solving its dual problem such that the vector $\delta^k$ satisfies a prescribed accuracy condition.

2.2. If $\beta^{k+1}$ satisfies a prescribed stopping condition, terminate; otherwise update $\sigma^{k+1} = \rho_k \sigma^k$, $\tau^{k+1} = \rho_k \tau^k$ with $\rho_k \in (0, 1)$ and return to Step 2.1 with $k = k + 1$. 
Algorithm. Let $\sigma^0, \sigma^1 > 0, \tau^0, \tau^1 > 0$ be given parameters. $\beta^0 \in \text{dom}(p)$.

1. Compute

$$\beta^1 \approx \arg\min_{\beta \in \mathbb{R}^n} \left\{ f(\beta) + p(\beta) + \frac{\sigma^0}{2} \| \beta - \beta^0 \|^2 + \frac{\tau^0}{2} \| X\beta - b \|^2 \right\}$$

via solving its dual problem such that a prescribed stopping criterion is satisfied. Let $k = 1$ and go to Step 2.1.

2.1 Compute

$$\beta^{k+1} \approx \arg\min_{\beta \in \mathbb{R}^n} \left\{ f(\beta) + p(\beta) + \frac{1}{2} \| \beta - \beta^k \|^2_{\sigma^k I + \tau^k X^*X} \right\}$$

via solving its dual problem satisfying a prescribed accuracy condition.

2.2. If $\beta^{k+1}$ satisfies a prescribed stopping condition, terminate; otherwise update $\sigma^{k+1} = \rho_k \sigma^k$, $\tau^{k+1} = \rho_k \tau^k$ with $\rho_k \in (0, 1)$ and return to Step 2.1 with $k = k + 1$. 
For simplicity, assume that we take for some constant $c > 0$ that

$$\tau_k \equiv c\sigma_k \quad \forall k.$$ 

Then the $k$-th subproblem of PPDNA can be written as

$$\beta^{k+1} \approx \arg\min_{\beta \in \mathbb{R}^n} \left\{ g_k(\beta) := f(\beta) + p(\beta) + \frac{\sigma_k}{2} \| \beta - \beta^k \|_M^2 \right\},$$

where

$$M := I + cX^*X \succ 0.$$
The stopping criterion for inner subproblems

\[(A) \quad g_k(\beta_{k+1}) - \inf g_k \leq \sigma_k \varepsilon_k^2 / 2, \quad \sum \varepsilon_k < \infty.\]

**Theorem (Global convergence)**

Suppose that the solution set to \((P)\) is nonempty. Then, \(\{\beta^k\}\) is bounded and converges to an optimal solution \(\beta^*\) of \((P)\).
Assumption (Error bound)

For a maximal monotone operator $\mathcal{T}(\cdot)$ with $\mathcal{T}^{-1}(0) \neq \emptyset$, $\exists \varepsilon > 0$ and $a > 0$ s.t.

$$\forall \eta \in B(0, \varepsilon) \quad \text{and} \quad \forall \xi \in \mathcal{T}^{-1}(\eta), \quad \text{dist}_M(\xi, \mathcal{T}^{-1}(0)) \leq a \|\eta\|_M,$$

where $B(0, \varepsilon) = \{y \in \mathcal{Y} \mid \|y\| \leq \varepsilon\}$. The constant $a$ is called the error bound modulus associated with $\mathcal{T}$.

1. In many cases, $\mathcal{T}$ is a polyhedral multifunction [Robinson, 1981].
2. $\mathcal{T}_g(\partial g)$ of LASSO, fused LASSO and elastic net regularized LS problems (piecewise linear-quadratic programming problems [J. Sun, PhD thesis, 1986] \(\Rightarrow\) error bound).
3. $\mathcal{T}_g$ of $\ell_1$ or elastic net regularized logistic regression [Luo and Tseng, 1992; Tseng and Yun, 2009].
4. Many more error bound results ...
PPDNA: Fast linear local convergence

Stopping criterion for the local convergence analysis

(B) \( g_k(\beta^{k+1}) - \inf g_k \)

\( \leq \min \{1, (\sigma_k \delta_k^2 / 2)\} \|\beta^{k+1} - \beta^k\|_M^2, \sum \delta_k < \infty. \)

Theorem

Assume that the solution set \( \Omega \) to (P) is nonempty. Assume that error bound condition holds for \( T_g \) with modulus \( l_g \). Then, \{\( \beta^k \)\} is convergent and, for all \( k \) sufficiently large,

\[ \text{dist}_M(\beta^{k+1}, \Omega) \leq \theta_k \text{dist}_M(\beta^k, \Omega), \]

where

\[ \theta_k \approx (l_g(l_g^2 + \sigma_k^{-2})^{-1/2} + 2\delta_k) \rightarrow \theta_\infty = (l_g\sigma_\infty) / \sqrt{1 + (l_g\sigma_\infty)^2} < 1 \]

as \( k \rightarrow \infty. \)

Note that \( \theta_\infty \ll 1 \) when \( l_g\sigma_\infty \) is close to zero. Thus, PPDNA can be treated as an approximate Newton’s method!!! (arbitrary linear convergence rate, a name coined by M.J.D. Powell in 1969).
So far we have

1. Outer iterations (PPA): asymptotically superlinear (arbitrary rate of linear convergence)
2. Inner iterations (nonsmooth Newton): superlinear + cheap

Essentially, we have a "fast$^2$" algorithm.
A convex example: the Fused LASSO problem

Before we talk about the non-convex case, let us look at the following convex problem

\[
\text{(Fused LASSO)}
\]

\[
\min \left\{ \frac{1}{2} \|X\beta - b\|^2 + \lambda \|\beta\|_1 + \lambda_2 \|B\beta\|_1 \right\}
\]

\[
B = \begin{pmatrix}
1 & -1 & & \\
1 & 1 & -1 & \\
& & \ddots & \ddots \\
& & 1 & -1
\end{pmatrix}
\]

In this case PPDNA \equiv SSNAL.
Performance profiles on biomedical datasets.
Numerical results for fused LASSO

Performance profiles on UCI datasets.
Why each nonsmooth Newton step cheap

For housing7 in the UCI data, the computational costs in our SSNAL are as follows:

1. costs for $X\beta$: 66 times, 0.11s in total;
2. costs for $X^T u$: 43 times, 2s in total;
3. costs for solving the inner linear systems: 43 times, 1.2s in total.

SSNAL has the ability to maintain the sparsity of $\beta$, the computational costs for calculating $X\beta$ are negligible comparing to other costs. In fact, each step of SSNAL is cheaper than many first order methods which need at least both $X\beta$ ($\beta$ may be dense) and $X^T u$. Note that SSNAL does not need to compute $XX^T$ (direct solvers) or $XX^T u$ (indirect methods) – many first order methods such as ADMM need (linearized ADMM can avoid it, but it is almost always inefficient).

**SOS is important for designing robust solvers!**

**SS-Newton equation can be solved very efficiently by exploiting the SOS property in solutions!**
For the illustrative purpose, consider a simpler example

\[
\min \left\{ \frac{1}{2} \|Ax - b\|^2 \mid x \geq 0 \right\}
\]

and its dual

\[
\max \left\{ -\frac{1}{2} \|\xi\|^2 + \langle b, \xi \rangle \mid A^T \xi \leq 0 \right\}
\]

**Interior-point based solver I: an** \( n \times n \) **linear system**

\[
(D + A^T A)x = \text{rhs}_1
\]

**D**: A **Diagonal matrix** with **positive** diagonal elements

Using PCG solver (e.g., matrix free interior point methods [K. Fountoulakis, J. Gondzio and P. Zhlobich, 2014])

Costly when \( n \) is large
Interior-point based solver II: an \( m \times m \) linear system

\[(I_m + AD^{-1}A^T)\xi = \text{rhs}_2\]

\[AD^{-1}A^T = O(m^2n \times \text{sparsity})\]
Why is sparse nonsmooth Newton’s method cheap?

Our nonsmooth Newton’s method: an \( m \times m \) linear system

\[
(I_m + APA^T)\xi = \text{rhs}_2
\]

\( P \): A **Diagonal matrix** with 0 or 1 diagonal elements (**generalized Jacobian** of \((\cdot)_+\))

\( r \): number of nonzero diagonal elements of \( P \) (**solution sparsity**)

\[
(AP)(AP)^T = \begin{bmatrix}
\color{blue}r \\
\color{white}m
\end{bmatrix} = O(m^2r * \text{sparsity})
\]
Usually, the solution can be very sparse, e.g, \( r = O(\log(n)) \). If \( r < m \), we can use the Sherman-Morrison-Woodbury formula:

\[
(AP)^T (AP) = O(r^2 m \ast \text{sparsity})
\]

which means solving our linear system meeds [assume that \( r = O(\log(n)) \)]

\[
O(r^2 m \ast \text{sparsity}) + O(r^3) = O((\log(n))^2 m \ast \text{sparsity}) + O((\log(n))^3).
\]

The above cost is negligible for large \( m \) and \( n \) comparing to others. This explains why sparse Newton’s method can be very cheap.
Let $\eta > 0$ and $\Phi_\eta$ be a set of all concave functions $\phi : [0, \eta) \to \mathbb{R}_+$ such that $\phi(0) = 0$, $\phi$ is continuous at $0$ and continuously differentiable on $(0, \eta)$ and $\phi'(x) > 0$, for $\forall \ x \in (0, \eta)$.

The function $f$ is said to have the Kurdyka-Łojasiewicz (KL) property at $\bar{x}$ if there exists $\eta > 0$, a neighbourhood $\mathcal{U}$ of $\bar{x}$ and a concave function $\phi \in \Phi_\eta$ such that

$$\phi'(f(x) - f(\bar{x})) \text{dist}(0, \partial f) \geq 1, \ \forall x \in \mathcal{U} \text{ and } f(\bar{x}) < f(x) < f(\bar{x}) + \eta,$$

where $\text{dist}(x, C) := \min_{y \in C} \| y - x \|$ is the distance from a point $x$ to a nonempty closed set $C$.

A function is said to have the KL property at $\bar{x}$ with an exponent $\alpha$ if the function $\phi$ in the definition of the KL property takes the form as $\phi(x) = \gamma x^{1-\alpha}$ with $\gamma > 0$ and $\alpha \in [0, 1)$. 
Convergence analysis

Theorem

Suppose that the function $g(\cdot)$ is bounded below and that $q$ is continuously differentiable near $B^\infty$, the set of all cluster points of the sequence $\{\beta_k\}$ generated by mPPDNA. Then every cluster point in $B^\infty$, if exists, is a d-stationary point of (3).

Theorem

Suppose that the function $g(\cdot)$ is bounded below and that $q$ is continuously differentiable near $B^\infty$, the set of all cluster points of the sequence $\{\beta_k\}$ generated by mPPDNA. If either one of the following two conditions holds,

(a) $B^\infty$ contains an isolated element;

(b) The sequence $\{\beta_k\}$ is bounded; for all $\beta \in B^\infty$, $\nabla q(\cdot)$ is locally Lipschitz continuous near $\beta$; and the function $g$ has the KL property at all $\beta \in B^\infty$;

then the whole sequence $\{\beta_k\}$ converges to a unique element of $B^\infty$. 
Theorem

Moreover, if the condition (b) is satisfied and \( \{ \beta^k \} \) converges to \( \beta^\infty \in \mathcal{B}^\infty \), the function \( g \) has the KL property at \( \beta^\infty \) with an exponent \( \alpha \in [0,1) \), then we have

(i) if \( \alpha = 0 \), then the sequence \( \{ \beta^k \} \) converges in a finite number of steps;

(ii) if \( \alpha \in (0, \frac{1}{2}] \), then the sequence \( \{ \beta^k \} \) converges R-linearly, that is, for all \( k \geq 1 \) there exist \( \nu > 0 \) and \( \eta \in [0,1) \) such that
\[
\| \beta^k - \beta^\infty \| \leq \nu \eta^k;
\]

(iii) if \( \alpha \in (\frac{1}{2}, 1) \), then the sequence \( \{ \beta^k \} \) converges R-sublinearly, that is, for all \( k \geq 1 \) there exists \( \nu > 0 \) such that
\[
\| \beta^k - \beta^\infty \| \leq \nu k^{-\frac{1-\alpha}{2\alpha-1}}.
\]
Our numerical experiments are implemented on a PC (Intel Core 2 Duo 2.6 GHz with 4 GB RAM).

The parameter $\lambda$ is defined by $\lambda = \lambda_c \Lambda$, $\Lambda = 1.1 \Phi^{-1}(1 - 0.05/(2n))$ with $\Phi$ the cumulative normal distribution function.

The number of nonzero elements of a vector is defined by the minimal $k$ such that

$$\sum_{i=1}^{k} |\tilde{\beta}_i| \geq 0.9999 \|\beta\|_1$$

where $\tilde{\beta}$ is obtained by sorting $\beta$ such that $|\tilde{\beta}_1| \geq |\tilde{\beta}_2| \geq \ldots \geq |\tilde{\beta}_n|$.
The step 1 of the mPPNDA algorithm will be terminated if the relative KKT residual\(^1\) satisfies

\[
\eta_{kkt} := \frac{\|\beta - P_L \lambda p \left( \beta - \frac{X^*(X \beta - b)}{\|X \beta - b\|} \right)\|}{1 + \|\beta\| + \frac{\|X^*(X \beta - b)\|}{\|X \beta - b\|}} < 10^{-6},
\]

or the number of iterations reaches the maximum 200 while the ADMMs will be terminated if (9) is satisfied or the number of iterations reaches the maximum 5000.

\(^1\)Whenever possible, try to avoid using the “fast convergence criteria” such as the relative distance of two consecutive iterates. Instead, try to design fast convergent algorithms independent of the “fast convergence criteria”, which may only indicate that the employed algorithm is slow for an earlier termination.
Table: The performances of ADMMs and PPDNA on UCI datasets for the srLasso problem. In the table, “a”=PPDNA, “b”=pADMM, “c”=dADMM.

<table>
<thead>
<tr>
<th>probname</th>
<th>$\lambda_c$</th>
<th>nnz</th>
<th>$\eta_{kkt}$</th>
<th>$\eta_G$</th>
<th>pobj</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$a$</td>
<td>$b$</td>
<td>$c$</td>
<td>$a$</td>
</tr>
<tr>
<td>log1p.E2006.test</td>
<td>0.260</td>
<td>201</td>
<td>7.9-7</td>
<td>1.2-4</td>
<td>1.2-3</td>
<td>2.8-6</td>
</tr>
<tr>
<td>3308;1771946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pyrim.scale.expanded5</td>
<td>0.109</td>
<td>70</td>
<td>5.9-7</td>
<td>2.0-5</td>
<td>3.7-3</td>
<td>6.9-7</td>
</tr>
<tr>
<td>74;201376</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>abalone.scale.expanded7</td>
<td>0.004</td>
<td>82</td>
<td>9.6-7</td>
<td>9.9-7</td>
<td>8.8-7</td>
<td>1.0-9</td>
</tr>
<tr>
<td>4177;6435</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bodyfat.scale.expanded7</td>
<td>0.012</td>
<td>51</td>
<td>7.4-7</td>
<td>1.1-6</td>
<td>9.9-7</td>
<td>2.2-8</td>
</tr>
<tr>
<td>252;116280</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
In our mPPDNA, the step 1 is used to generate an initial point for the step 2 and is stopped if $\eta_{kkt} < 10^{-4}$. The algorithms will be terminated if the relative KKT residual satisfies

$$\tilde{\eta}_{kkt} := \frac{\|\beta - P_1(p - q) \left( \beta - \frac{X^*(X\beta - b)}{\|X\beta - b\|} \right)\|}{1 + \|\beta\| + \frac{\|X^*(X\beta - b)\|}{\|X\beta - b\|}} < 10^{-6}$$

Besides, the algorithms will also be stopped when they reach the pre-set maximum number of iterations (200 for the second step of mPPDNA and 5000 for ADMM).
For $\lambda > 0$, the SCAD regularization is defined by $r(\beta) = p(\beta) - q(\beta)$ with

$$p(\beta) = \lambda \|\beta\|_1$$

$$q(\beta) = \sum_{i=1}^{n} \begin{cases} 
0, & \text{if } |\beta_i| < \lambda \\
\frac{(|\beta_i| - \lambda)^2}{2(a_s - 1)}, & \text{if } \lambda \leq |\beta_i| \leq a_s \lambda \\
\lambda |\beta_i| - \frac{a_s + 1}{2} \lambda^2, & \text{if } |\beta_i| > a_s \lambda
\end{cases}$$

In our numerical experiments, we take $a_s = 3.7$. 
Table: The performances of ADMM and mPPDNA on UCI datasets for the SCAD regularization. In the table, “a” = mPPDNA, “b” = ADMM.

<table>
<thead>
<tr>
<th>probname</th>
<th>$\lambda_c$</th>
<th>nnz</th>
<th>$\eta_{kkt}$</th>
<th>pobj</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$a$</td>
<td>$b$</td>
<td>$a$</td>
</tr>
<tr>
<td>log1p.E2006.test</td>
<td>0.257</td>
<td>207</td>
<td>2.1-7</td>
<td>5.9-3</td>
<td>2.1613+1</td>
</tr>
<tr>
<td>3308;1771946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pyrim.scale.expanded5</td>
<td>0.109</td>
<td>70</td>
<td>1.4-7</td>
<td>4.3-3</td>
<td>6.8301-1</td>
</tr>
<tr>
<td>74;201376</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>abalone.scale.expanded7</td>
<td>0.011</td>
<td>49</td>
<td>9.9-7</td>
<td>6.9-1</td>
<td>1.3292+2</td>
</tr>
<tr>
<td>4177;6435</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bodyfat.scale.expanded7</td>
<td>0.201</td>
<td>2</td>
<td>3.9-8</td>
<td>7.6-2</td>
<td>9.4125-1</td>
</tr>
<tr>
<td>252;116280</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Collaborators

- NUS: Kim-Chuan Toh; Yangjing Zhang; Meixia Lin; Yancheng Yuan
- Fudan University: Xudong Li
- Zhejiang University City College: Peipei Tang
- Fuzhou University: Yongjin Liu
- Dongguan University of Technology: Ning Zhang
- Beijing Jiaotong University: Ziyan Luo; Naihua Xiu
- Hong Kong PolyU: Ying Sun
- There are many more ...


Yangjing Zhang, Ning Zhang, Defeng Sun, and Kim Chuan Toh, An efficient Hessian based algorithm for solving large-scale sparse group Lasso problems, Mathematical Programming 179 (2020) 223–263.


Y. Sun, X.D. Li, D.F. Sun, and K.-C. Toh, A Proximal Point Dual Newton Algorithm for Solving Constrained Fused Lasso Problems, in preparation.