Second Order Sparsity and Big Data Optimization

Defeng Sun

Department of Applied Mathematics, The Hong Kong Polytechnic University

October 31, 2017/Southwest Jiaotong University

Based on joint works with: Houduo Qi, Kim-Chuan Toh, Xinyuan Zhao, Liuqin Yang, Xudong Li, et al.
Consider the nearest correlation matrix (NCM) problem:

$$\min \left\{ \frac{1}{2} \| X - G \|_F^2 \mid X \succeq 0, X_{ii} = 1, i = 1, \ldots, n \right\}.$$ 

The dual of the above problem can be written as

$$\min \frac{1}{2} \| \Xi \|_2^2 - \langle b, y \rangle - \frac{1}{2} \| G \|_2^2$$

s.t. \( S - \Xi + A^* y = -G, \quad S \succeq 0 \)

or via eliminating \( \Xi \) and \( S \succeq 0 \), the following

$$\min \left\{ \varphi(y) := \frac{1}{2} \| \Pi_{\geq 0}(A^* y + G) \|_2^2 - \langle b, y \rangle - \frac{1}{2} \| G \|_2^2 \right\}.$$
Numerical results for the NCM

Test the second order nonsmooth Newton-CG method [H.-D. Qi & Sun 06] and two popular first order methods (FOMs) [APG of Nesterov; ADMM of Glowinski (steplength 1.618)] all to the dual forms for the NCM with real financial data: 

$G$: Cor3120, $n = 3, 120$, obtained from [N. J. Higham & N. Strabić, SIMAX, 2016] [Optimal sol. rank = 3, 025]

<table>
<thead>
<tr>
<th>$n = 3, 120$</th>
<th>SSNCG</th>
<th>ADMM</th>
<th>APG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rel. KKT Res.</td>
<td>2.7-8</td>
<td>2.9-7</td>
<td>9.2-7</td>
</tr>
<tr>
<td>time (s)</td>
<td>26.8</td>
<td>246.4</td>
<td>459.1</td>
</tr>
<tr>
<td>iters</td>
<td>4</td>
<td>58</td>
<td>111</td>
</tr>
<tr>
<td>avg-time/iter</td>
<td>6.7</td>
<td>4.3</td>
<td>4.1</td>
</tr>
</tbody>
</table>

Newton method only takes at most 40% time more than ADMM & APG per iteration. How is it possible?
Lasso-type problems

We shall use simple vector cases to explain why:

\[(\text{LASSO})\]

\[
\min \left\{ \frac{1}{2} \| Ax - b \|_2^2 + \lambda \| x \|_1 \mid x \in \mathbb{R}^n \right\}
\]

where \( \lambda > 0, \ A \in \mathbb{R}^{m \times n}, \) and \( b \in \mathbb{R}^m.\)

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

\[
\min \left\{ \frac{1}{2} \| Ax - b \|_2^2 + \lambda \| x \|_1 + \lambda_2 \| Bx \|_1 \right\}
\]

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

\[ \min \left\{ \frac{1}{2} \|Ax - b\|^2 + \lambda \|x\|_1 + \lambda_2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} |x_i - x_j| \right\} \]  

(Clustered LASSO)

\[ \min \left\{ \frac{1}{2} \|Ax - b\|^2 + \lambda \|x\|_1 + \lambda_2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} |x_i + x_j| + |x_i - x_j| \right\} \]  

(OSCAR)

We are interested in \( n \) (number of features) large and/or \( m \) (number of samples) large
Sparse regression:

\[ A^{m \times n} \approx b^{1 \times n} \]

# of features \( n \gg \) # of samples \( m \)

searching for a sparse solution
Example: Support vector machine

Class +1

Class -1

Support Vectors

Residuals, $r_i$

Normal Vector

Separating Hyperplane
Figure: Sir Isaac Newton (Niu Dun) (4 January 1643 - 31 March 1727)
Newton’s method
Interior point methods

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. Foutoulakis, J. Gondzio and P. Zhlobich, 2014])

Costly when \(n\) is large
Interior point methods

Interior-point based solver II: an $m \times m$ linear system

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

$$AA^T = O(m^2 n \ast \text{sparsity})$$
Our nonsmooth Newton’s method

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

$r$: number of nonzero diagonal elements of $P$ (*second order sparsity*)

$$(AP)(AP)^T = \begin{bmatrix} m & \frac{r}{m} \\ \frac{r}{m} & \frac{r}{m} \end{bmatrix} = O(m^2r \cdot \text{sparsity})$$

Sherman-Morrison-Woodbury formula:

$$(AP)^T(AP) = \begin{bmatrix} \frac{r}{m} & \frac{r}{m} \\ \frac{r}{m} & \frac{r}{m} \end{bmatrix} = O(r^2m \cdot \text{sparsity})$$
Convex composite programming

\[ (P) \quad \min \{ f(x) := h(Ax) + p(x) \} , \]

Real finite dimensional Euclidean spaces \( \mathcal{X}, \mathcal{Y} \)

Closed proper convex function \( p : \mathcal{X} \to (-\infty, +\infty] \)

Convex differentiable function \( h : \mathcal{Y} \to \mathbb{R} \)

Linear map \( A : \mathcal{X} \to \mathcal{Y} \)

Dual problem

\[ (D) \quad \min \{ h^*(\xi) + p^*(u) \mid A^*\xi + u = 0 \} \]

\( p^* \) and \( h^* \): the Fenchel conjugate functions of \( p \) and \( h \).

\( p^*(z) = \sup \{ \langle z, x \rangle - p(x) \} . \)
Examples in machine learning

Examples of smooth loss function $h$:
- Linear regression $h(y) = \|y - b\|^2$
- Logistic regression $h(y) = \log(1 + \exp(-yb))$
- many more ...

Examples of regularizer $p$:
- LASSO $p(x) = \|x\|_1$
- Fused LASSO $p(x) = \|x\|_1 + \sum_{i=1}^{n-1} |x_i - x_{i+1}|$
- Ridge $p(x) = \|x\|_2^2$
- Elastic net $p(x) = \|x\|_1 + \|x\|_2^2$
- Group LASSO
- Fused Group LASSO
- Clustered LASSO, OSCAR
- etc
Assumptions on the loss function

Assumption 1 (Assumptions on $h$)

1. $h : \mathcal{Y} \to \mathbb{R}$ has a $1/\alpha_h$-Lipschitz continuous gradient:

$$\|\nabla h(y_1) - \nabla h(y_2)\| \leq \left(\frac{1}{\alpha_h}\right)\|y_1 - y_2\|, \quad \forall y_1, y_2 \in \mathcal{Y}$$

2. $h$ is essentially locally strongly convex [Goebel and Rockafellar, 2008]: for any compact and convex set $K \subset \text{dom } \partial h$, $\exists \beta_K > 0$ s.t.

$$(1 - \lambda)h(y_1) + \lambda h(y_2) \geq h((1 - \lambda)y_1 + \lambda y_2) + \frac{1}{2}\beta_K \lambda(1 - \lambda)\|y_1 - y_2\|^2$$

for all $\lambda \in [0, 1]$, $y_1, y_2 \in K$
Properties on $h^*$

Under the assumptions on $h$, we know

a. $h^*$: strongly convex with constant $\alpha_h$

b. $h^*$: essentially smooth$^1$

c. $\nabla h^*$: locally Lipschitz continuous on $\mathcal{D}_{h^*} := \text{int} \ (\text{dom} \ h^*)$

d. $\partial h^*(y) = \emptyset$ when $y \notin \mathcal{D}_{h^*}$.

Only need to focus on $\mathcal{D}_{h^*}$

---

$^1$ $h^*$ is differentiable on $\text{int} \ (\text{dom} \ h^*)$ $\neq \emptyset$ and $\lim_{i \to \infty} \|\nabla h^*(y_i)\| = +\infty$ whenever $\{y_i\} \subset \text{int} \ (\text{dom} \ h^*) \to y \in \text{bdry}(\text{int} \ (\text{dom} \ h^*))$. 
An augmented Lagrangian method for (D)

The Lagrangian function for (D):

\[ l(\xi, u; x) = h^*(\xi) + p^*(u) - \langle x, A^*\xi + u \rangle, \quad \forall (\xi, u, x) \in \mathcal{Y} \times \mathcal{X} \times \mathcal{X}. \]

Given \( \sigma > 0 \), the augmented Lagrangian function for (D):

\[ \mathcal{L}_{\sigma}(\xi, u; x) = l(\xi, u; x) + \frac{\sigma}{2} \| A^*\xi + u \|^2, \quad \forall (\xi, u, x) \in \mathcal{Y} \times \mathcal{X} \times \mathcal{X}. \]

The proximal mapping Prox_p(x):

\[ \text{Prox}_p(x) = \arg \min_{u \in \mathcal{X}} \left\{ p(u) + \frac{1}{2} \| u - x \|^2 \right\}. \]

Assumption: \( \text{Prox}_{\sigma p}(x) \) is easy to compute given any \( x \)

Advantage of using (D): \( h^* \) is strongly convex; \( \min_u \{ \mathcal{L}_{\sigma}(\xi, u; x) \} \) is easy.
An inexact augmented Lagrangian method of multipliers.

Given $\sum \varepsilon_k < +\infty$, $\sigma_0 > 0$, choose $(\xi^0, u^0, x^0) \in \text{int}(\text{dom } h^*) \times \text{dom } p^* \times X$. For $k = 0, 1, \ldots$, iterate

**Step 1.** Compute

$$(\xi^{k+1}, u^{k+1}) \approx \arg \min \{\Psi_k(\xi, u) := \mathcal{L}_{\sigma_k}(\xi, u; x^k)\}.$$  

To be solved via a nonsmooth Newton method.

**Step 2.** Compute $x^{k+1} = x^k - \sigma_k(A^*\xi^{k+1} + u^{k+1})$ and update $\sigma_{k+1} \uparrow \sigma_\infty \leq \infty$.  

The stopping criterion for inner subproblem

\[ (A) \quad \Psi_k(\xi^{k+1}, u^{k+1}) - \inf \Psi_k \leq \frac{\varepsilon_k^2}{2\sigma_k}, \quad \sum \varepsilon_k < \infty. \]

Theorem 1 (Global convergence)

Suppose that the solution set to \((P)\) is nonempty. Then, \(\{x^k\}\) is bounded and converges to an optimal solution \(x^*\) of \((P)\). In addition, \(\{(\xi^k, u^k)\}\) is also bounded and converges to the unique optimal solution \((\xi^*, u^*)\in \text{int(dom } h^*) \times \text{dom } p^*\) of \((D)\).
Assumption 2 (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.

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

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. $\mathcal{T}$ is a polyhedral multifunction [Robinson, 1981].
3. $\mathcal{T}_f$ of $\ell_1$ or elastic net regularized logistic regression [Luo and Tseng, 1992; Tseng and Yun, 2009].
Fast linear local convergence

Stopping criterion for the local convergence analysis

\[(B) \quad \Psi_k(\xi^{k+1},u^{k+1}) - \inf \Psi_k \leq \min\{1,(\delta^2_k/2\sigma_k)\}\|x^{k+1} - x^k\|^2, \quad \sum \delta_k < \infty.\]

**Theorem 2**

Assume that the solution set $\Omega$ to $(P)$ is nonempty. Suppose that Assumption 2 holds for $T_f$ with modulus $a_f$. Then, $\{x^k\}$ is convergent and, for all $k$ sufficiently large,

$$\text{dist}(x^{k+1},\Omega) \leq \theta_k \text{dist}(x^k,\Omega),$$

where $\theta_k \approx (a_f(a_f^2 + \sigma_k^2)^{-1/2} + 2\delta_k) \rightarrow \theta_\infty = a_f/\sqrt{a_f^2 + \sigma_\infty^2} < 1$ as $k \rightarrow \infty$. Moreover, the conclusions of Theorem 1 about $\{((\xi^k,y^k))\}$ are valid.

ALM is an approximate Newton’s method!!!
Fix $\sigma > 0$ and $\tilde{x}$, denote

$$
\psi(\xi) := \inf_u \mathcal{L}_\sigma(\xi, u, \tilde{x})
$$

$$
= h^*(\xi) + p^*(\text{Prox}_{p^*/\sigma}(\tilde{x}/\sigma - A^*\xi)) + \frac{1}{2\sigma} \|\text{Prox}_{\sigma p}(\tilde{x} - \sigma A^*\xi)\|^2.
$$

$\psi(\cdot)$: strongly convex and continuously differentiable on $D_{h^*}$ with

$$
\nabla \psi(\xi) = \nabla h^*(\xi) - A \text{Prox}_{\sigma p}(\tilde{x} - \sigma A^*\xi), \quad \forall \xi \in D_{h^*}
$$

Solving nonsmooth equation:

$$
\nabla \psi(\xi) = 0, \quad \xi \in D_{h^*}.
$$
Denote for $\xi \in D_{h^*}$:

$$\hat{\partial}^2 \psi(\xi) := \partial^2 h^*(\xi) + \sigma A \partial \text{Prox}_{\sigma p}(\tilde{x} - \sigma A^* \xi) A^*$$

$\partial^2 h^*(\xi)$: Clarke subdifferential of $\nabla h^*$ at $\xi$

$\partial \text{Prox}_{\sigma p}(\tilde{x} - \sigma A^* \xi)$: Clarke subdifferential of $\text{Prox}_{\sigma p}(\cdot)$ at $\tilde{x} - \sigma A^* \xi$

Lipschitz continuous mapping: $\nabla h^*$, $\text{Prox}_{\sigma p}(\cdot)$

From [Hiriart-Urruty et al., 1984],

$$\hat{\partial}^2 \psi(\xi) (d) = \partial^2 \psi(\xi) (d), \quad \forall d \in \mathcal{V}$$

$\partial^2 \psi(\xi)$: the generalized Hessian of $\psi$ at $\xi$. Define

$$V^0 := H^0 + \sigma A U^0 A^*$$

with $H^0 \in \partial^2 h^*(\xi)$ and $U^0 \in \partial \text{Prox}_{\sigma p}(\tilde{x} - \sigma A^* \xi)$

$V^0 \succ 0$ and $V^0 \in \hat{\partial}^2 \psi(\xi)$
SSN($\xi^0, u^0, \tilde{x}, \sigma$). Given $\mu \in (0, 1/2)$, $\bar{\eta} \in (0, 1)$, $\tau \in (0, 1]$, and $\delta \in (0, 1)$. Choose $\xi^0 \in D_{h^*}$. Iterate

**Step 1.** Find an approximate solution $d^j \in \mathcal{Y}$ to

$$V_j(d) = -\nabla \psi(\xi^j)$$

with $V_j \in \partial^2 \psi(\xi^j)$ s.t.

$$\|V_j(d^j) + \nabla \psi(\xi^j)\| \leq \min(\bar{\eta}, \|\nabla \psi(\xi^j)\|^{1+\tau}).$$

**Step 2.** (Line search) Set $\alpha_j = \delta^{m_j}$, where $m_j$ is the first nonnegative integer $m$ for which

$$\xi^j + \delta^m d^j \in D_{h^*}$$

$$\psi(\xi^j + \delta^m d^j) \leq \psi(\xi^j) + \mu \delta^m \langle \nabla \psi(\xi^j), d^j \rangle.$$

**Step 3.** Set $\xi^{j+1} = \xi^j + \alpha_j d^j$. 
Theorem 3

Assume that $\nabla h^*(\cdot)$ and $\text{Prox}_{\sigma p}(\cdot)$ are strongly semismooth on $D_{h^*}$ and $\mathcal{X}$. Then $\{\xi^j\}$ converges to the unique optimal solution $\bar{\xi} \in D_{h^*}$ and

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

Implementable stopping criteria: the stopping criteria (A) and (B) can be achieved via:

\[(A') \quad \|\nabla \psi_k(\xi^{k+1})\| \leq \sqrt{\frac{\alpha h}{\sigma_k}} \varepsilon_k\]

\[(B') \quad \|\nabla \psi_k(\xi^{k+1})\| \leq \sqrt{\frac{\alpha h}{\sigma_k}} \delta_k \min\{1, \sigma_k \|A^* \xi^{k+1} + u^{k+1}\|\}\]

\[(A') \Rightarrow (A) \quad \& \quad (B') \Rightarrow (B)\]
So far we have

- Outer iterations (ALM): asymptotically superlinear (truly fast linear)
- Inner iterations (nonsmooth Newton): superlinear + cheap

Essentially, we have a "fast + fast" algorithm.
LASSO: \( \min \left\{ \frac{1}{2} \| Ax - b \|_2^2 + \lambda_1 \| x \|_1 \right\} \)

\( h(y) = \frac{1}{2} \| y - b \|_2^2, \quad p(x) = \lambda_1 \| x \|_1 \)

\( \text{Prox}_{\sigma p}(x) \): easy to compute = \( \text{sgn}(x) \circ \max\{|x| - \sigma \lambda_1, 0\} \)

Newton System:

\[
(I + \sigma AP A^*) \xi = \text{rhs}
\]

\( P \in \partial \text{Prox}_{\sigma p}(x^k - \sigma A^* \xi) \): diagonal matrix with 0, 1 entries. Most of these entries are 0 if the optimal solution \( x^{opt} \) is sparse.

**Message:** Nonsmooth Newton can fully exploit the second order sparsity (SOS) of solutions to solve the Newton system very efficiently!
Fused LASSO: $\min \left\{ \frac{1}{2} \|Ax - b\|^2 + \lambda_1 \|x\|_1 + \lambda_2 \|Bx\|_1 \right\}$

$$B = \begin{pmatrix} 1 & -1 \\ 1 & -1 \\ \vdots & \vdots \\ 1 & -1 \end{pmatrix}$$

$h(y) = \frac{1}{2} \|y - b\|^2$, $p(x) = \lambda_1 \|x\|_1 + \lambda_2 \|Bx\|_1$

Let $x_{\lambda_2}(v) := \arg \min_x \frac{1}{2} \|x - v\|^2 + \lambda_2 \|Bx\|_1$.

Proximal mapping of $p$ [Friedman et al., 2007]:

$$\text{Prox}_p(v) = \text{sign}(x_{\lambda_2}(v)) \circ \max(\text{abs}(x_{\lambda_2}(v)) - \lambda_1, 0).$$

Efficient algorithms to obtain $x_{\lambda_2}(v)$: taut-string [Davies and Kovac, 2001], direct algorithm [Condat, 2013], dynamic programming [Johnson, 2013]
Newton system for fused LASSO

Dual approach to obtain $x_{\lambda_2}(v)$: denote

$$z(v) := \arg\min_z \left\{ \frac{1}{2}\|B^*z\|^2 - \langle Bv, z \rangle | \|z\|_\infty \leq \lambda_2 \right\}$$

$$\Rightarrow x(v) = v - B^*z(v).$$

Let $C = \{z | \|z\|_\infty \leq \lambda_2\}$, from optimality condition

$$z = \Pi_C (z - (BB^*z - Bv))$$

and the implicit function theorem $\Rightarrow$ Newton system for fused Lasso:

$$(I + \sigma \hat{P} A^*) \xi = \text{rhs}$$

$$\hat{P} = P(I - B^*(I - \Sigma + \Sigma BB^*)^{-1} \Sigma B) \quad \text{(positive semidefinite)}$$

$$\Sigma \in \partial \Pi_C (z - (BB^*z - Bv))$$

$P, \Sigma$: diagonal matrices with 0, 1 entries. Most diagonal entries of $P$ are 0 if $x^{opt}$ is sparse. The red part is diagonal + low rank

Again, can use sparsity and the structure of red part to solve the system efficiently
Numerical results for LASSO

KKT residual:

\[ \eta_{\text{KKT}} := \frac{\|\tilde{x} - \text{Prox}_p[\tilde{x} - (A\tilde{x} - b)]\|}{1 + \|\tilde{x}\| + \|A\tilde{x} - b\|} \leq 10^{-6}. \]

Compare SSNAL with state-of-the-art solvers: mfIPM, ... [Fountoulakis et al., 2014] and APG [Liu et al. 2011]

\((A, b)\) taken from 11 Sparco collections (all easy problems) [Van Den Berg et al, 2009]

\[ \lambda = \lambda_c \|A^*b\|_\infty \text{ with } \lambda_c = 10^{-3} \text{ and } 10^{-4} \]

Add 60dB noise to \(b\) in MATLAB: \(b = \text{awgn}(b, 60, \text{'measured'})\)

max. iteration number: 20,000 for APG
max. computation time: 7 hours
Numerical results for LASSO arising from compressed sensing

(a) **our SSNAL**  
(b) mflPM  
(c) APG: Nesterov’s accelerated proximal gradient method

<table>
<thead>
<tr>
<th>probname</th>
<th>$m; n$</th>
<th>$\eta_{\text{KKT}}$</th>
<th>time (hh:mm:ss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>srcsep1</td>
<td>29166;57344</td>
<td>1.6-7</td>
<td>7.3-7</td>
</tr>
<tr>
<td>soccer1</td>
<td>3200;4096</td>
<td>1.8-7</td>
<td>6.3-7</td>
</tr>
<tr>
<td>blurrycam</td>
<td>65536;65536</td>
<td>1.9-7</td>
<td>6.5-7</td>
</tr>
<tr>
<td>blurrycam</td>
<td>16384;16384</td>
<td>3.1-7</td>
<td>9.5-7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>probname</th>
<th>$m; n$</th>
<th>$\eta_{\text{KKT}}$</th>
<th>time (hh:mm:ss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>srcsep1</td>
<td>29166;57344</td>
<td>9.8-7</td>
<td>9.5-7</td>
</tr>
<tr>
<td>soccer1</td>
<td>3200;4096</td>
<td>8.7-7</td>
<td>4.3-7</td>
</tr>
<tr>
<td>blurrycam</td>
<td>65536;65536</td>
<td>1.0-7</td>
<td>9.7-7</td>
</tr>
<tr>
<td>blurrycam</td>
<td>16384;16384</td>
<td>3.5-7</td>
<td>7.4-7</td>
</tr>
</tbody>
</table>
Numerical results for LASSO arising from sparse regression

11 large scale instances \((A, b)\) from LIBSVM [Chang and Lin, 2011]

\(A\): data normalized (with at most unit norm columns)

\[ \lambda_c = 10^{-3} \]

<table>
<thead>
<tr>
<th>probname</th>
<th>(m; n)</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>time (hh:mm:ss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2006.train</td>
<td>16087; 150360</td>
<td>1.6-7</td>
<td>4.1-7</td>
<td>9.1-7</td>
<td>01</td>
</tr>
<tr>
<td>log1p.E2006.train</td>
<td>16087; 4272227</td>
<td>2.6-7</td>
<td>4.9-7</td>
<td>1.7-4</td>
<td>35</td>
</tr>
<tr>
<td>E2006.test</td>
<td>3308; 150358</td>
<td>1.6-7</td>
<td>1.3-7</td>
<td>3.9-7</td>
<td>01</td>
</tr>
<tr>
<td>log1p.E2006.test</td>
<td>3308; 4272226</td>
<td>1.4-7</td>
<td>9.2-8</td>
<td>1.6-2</td>
<td>27</td>
</tr>
<tr>
<td>pyrim5</td>
<td>74; 201376</td>
<td>2.5-7</td>
<td>4.2-7</td>
<td>3.6-3</td>
<td>05</td>
</tr>
<tr>
<td>triazines4</td>
<td>186; 635376</td>
<td>8.5-7</td>
<td>7.7-1</td>
<td>1.8-3</td>
<td>29</td>
</tr>
<tr>
<td>abalone7</td>
<td>4177; 6435</td>
<td>8.4-7</td>
<td>1.6-7</td>
<td>1.3-3</td>
<td>02</td>
</tr>
<tr>
<td>bodyfat7</td>
<td>252; 116280</td>
<td>1.2-8</td>
<td>5.2-7</td>
<td>1.4-2</td>
<td>02</td>
</tr>
<tr>
<td>housing7</td>
<td>506; 77520</td>
<td>8.8-7</td>
<td>6.6-7</td>
<td>4.1-4</td>
<td>03</td>
</tr>
</tbody>
</table>
Why each nonsmooth Newton step cheap

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

- costs for $Ax$: 66 times, 0.11s in total;
- costs for $A^T\xi$: 43 times, 2s in total;
- costs for solving the inner linear systems: 43 times, 1.2s in total.

SSNAL has the ability to maintain the sparsity of $x$, the computational costs for calculating $Ax$ are negligible comparing to other costs. In fact, each step of SSNAL is cheaper than many first order methods which need at least both $Ax$ ($x$ may be dense) and $A^T\xi$.

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

**SS-Newotn equation can be solved very efficiently by exploiting the SOS property in solutions!**
Numerical results for fused LASSO

(a) our SSNAL
(b) APG based solver [Liu et al., 2011] (enhanced...)
(c1) ADMM (classical) (c2) ADMM (linearized)

Parameters: $\lambda_1 = \lambda_c \|A^*y\|_\infty$, $\lambda_2 = 2\lambda_1$, $\text{tol} = 10^{-4}$

Problem: triazines 4, $m = 186$, $n = 635376$

<table>
<thead>
<tr>
<th>Fused Lasso P.</th>
<th>iter</th>
<th>time (hh:mm:ss)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_c$</td>
<td>nnz</td>
<td>$\eta_C$</td>
</tr>
<tr>
<td>10^{-1}</td>
<td>164</td>
<td>2.4-2</td>
</tr>
<tr>
<td>10^{-2}</td>
<td>1004</td>
<td>1.7-2</td>
</tr>
<tr>
<td>10^{-3}</td>
<td>1509</td>
<td>1.2-3</td>
</tr>
<tr>
<td>10^{-5}</td>
<td>2420</td>
<td>6.4-5</td>
</tr>
</tbody>
</table>

SSNAL is vastly superior to first-order methods: APG, ADMM (classical), ADMM (linearized)

ADMM (linearized) needs many more iterations than ADMM (classical)
When to choose SSNAL?

When $\text{Prox}_p$ and its generalized Jacobian $\partial \text{Prox}_p$ are easy to compute

Almost all of the LASSO models are suitable for SSNAL

When the problems are very easy, one may also consider APG or ADMM

Very complicated problems, in particular with many constraints, consider 2-phase approaches
Big Data Optimization Models Provide Many Opportunities to Test New Ideas. *SOS* is just one of them.


[X.Y. Zhao, D.F. Sun, and K-C. Toh], A Newton-CG augmented Lagrangian method for semidefinite programming, SIOPT, 2010
