Newton-type Methods for Support Vector Machines and Signal Reconstruction Problems

Kimon Fountoulakis

coauthors: I. Dassios, J. Gondzio and R. Tappenden

Numerical Analysis Group Internal Seminar, Oxford, 28 April 2015
Outline

Data fitting and binary classification
- Method: Flexible Coordinate Descent
- Coauthor: R. Tappenden

Signal reconstruction
- Method: Primal-dual Newton Conjugate Gradients
- Contribution: Provably efficient preconditioning technique
- Coauthors: I. Dassios and J. Gondzio
Standard examples in optimization

Data fitting

\[
\text{minimize } \tau \|x\|_1 + \frac{1}{2} \|Ax - b\|_2^2
\]

\[
\text{minimize } \frac{1}{2} \|Ax - b\|_2^2
\]
Standard examples in optimization

**Binary classification**

\[
\text{minimize } \tau \|x\|_1 + \sum_{i=1}^{m} \log(1 + e^{-b_i x^T a_i})
\]

\[
\text{minimize } \tau \|x\|_2^2 + \sum_{i=1}^{m} \log(1 + e^{-b_i x^T a_i})
\]
Problem formulation

\[
\text{minimize } F(x) := \Psi(x) + f(x)
\]

- \( x \in \mathbb{R}^N, f(x) : \mathbb{R}^N \to \mathbb{R}, \Psi(x) : \mathbb{R}^N \to \mathbb{R} \)

Assumptions

- \( f \) is smooth convex function
- \( \Psi \) is a (possibly) nonsmooth convex function

Plenty of data

- \( N \) is very large. i.e. of order millions or billions
Numerical methods in convex optimization

Build a convex function $Q$ that locally approximates $F$ at a point $x$:
- $Q(y; x) \approx F(y)$ for $y$ close to $x$
- $Q(x; x) = F(x)$

General framework
1: Given $x_0$ (an initial guess)
2: For $k = 0, 1, 2, \ldots$
3: Approximately solve the subproblem

$$x^{k+1} \approx \arg \min_y Q(y; x_k)$$
Examples of local convex approximations

\(_1 \text{Logistic Regression (ℓ}_1 \text{ LR): minimize } \tau \|x\|_1 + \sum_{i=1}^m \log(1 + e^{-b_i x^T a_i})\)

- Separable model (majority of modern algorithms)
  
  \(Q(y; x_k) := \tau \|y\|_1 + f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{L}{2} \langle y - x_k, y - x_k \rangle\)

- Non separable model

\(Q(y; x_k) := \tau \|y\|_1 + f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{1}{2} \langle y - x_k, H_k(y - x_k) \rangle\)
Examples of local convex approximations

$\ell_1$ Logistic Regression ($\ell_1$ LR): minimize $\tau \|x\|_1 + \sum_{i=1}^{m} \log(1 + e^{-b_i x^T a_i})$

- Separable model (majority of modern algorithms)
  \[ Q(y; x_k) := \tau \|y\|_1 + f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{L}{2} \langle y - x_k, y - x_k \rangle \]

- Non separable model
  \[ Q(y; x_k) := \tau \|y\|_1 + f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{1}{2} \langle y - x_k, H_k(y - x_k) \rangle \]

\[ L = \lambda_{\text{max}}(\nabla^2 f(y_k)) \]

\[ H_k = \nabla^2 f(y_k) \]
Trade-off between separable and non-separable approximations

<table>
<thead>
<tr>
<th>Type</th>
<th>Inexpensive step</th>
<th>Good approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Separable model</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Non sep. model</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Aim: control the trade off

Three ways in Flexible Coordinate Descent (FCD)

- Inexpensively choose $H_k$ such that it incorporates the structure of $f$
- Dimensionality reduction: update only a subset of coordinates
- Solve approximately the subproblem over the chosen coordinates
1) Construct any $H_k \succ 0$ such that $H_k \approx \nabla^2 f(x_k)$. No need to store $H_k$, we only need a process to perform matrix-vector products with it.

2) Construct a local convex model

$$Q(y; x_k) := \Psi(y) + f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{1}{2} \langle y - x_k, H_k(y - x_k) \rangle$$
Dimensionality reduction

Select a subset of indices/coordinates $S \subseteq [N]$
Dimensionality reduction

Select a subset of indices/coordinates $S \subseteq [N]$
Dimensionality reduction

Select a subset of indices/coordinates $S \subseteq [N]$
Dimensionality reduction: notation

\[ x \]

\[ \nabla_{Sf}(x) \]

\[ H^S \]

\[ x^S \]

\[ U_S x^S \text{ zeros} \]
Dimensionality reduction: notation

\[ x^S \]

\[ \nabla_S f(x) \]

\[ H^S \]

\[ x^S \quad U_S x^S \]

zeros
Dimensionality reduction: a smaller subproblem

Assumption: $\Psi$ is separable

$$\Psi(x) = x^T \Psi(x) = \Psi_1(x^{(1)}) + \Psi_2(x^{(2)}) + \Psi_3(x^{(3)}) + \Psi_4(x^{(4)}) + \Psi_5(x^{(5)})$$

Reformulation of local approximation and subproblem

$$Q_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle + \frac{1}{2} \langle t^S, H_k^S t^S \rangle$$

$$t_k^S \approx \arg \min_{t^S} Q_S(x_k^S + t^S; x_k)$$
Dimensionality reduction: a smaller subproblem

Assumption: \( \Psi \) is separable

\[
\Psi(x) = \Psi_1(x^{(1)}) + \Psi_2(x^{(2)}) + \Psi_3(x^{(3)}) + \Psi_4(x^{(4)}) + \Psi_5(x^{(5)})
\]

Reformulation of local approximation and subproblem

\[
Q_S(x^S_k + t^S; x_k) := \Psi_S(x^S_k + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle + \frac{1}{2} \langle t^S, H^S_k t^S \rangle
\]

\[
t^S_k \approx \arg \min_{t^S} Q_S(x^S_k + t^S; x_k)
\]
Dimensionality reduction: a smaller subproblem

Assumption: $\Psi$ is separable

$$\Psi(x) = \sum_{i=1}^{5} \Psi_i(x^{(i)})$$

Reformulation of local approximation and subproblem

$$Q_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle + \frac{1}{2} \langle t^S, H_k^S t^S \rangle$$

$$t_k^S \approx \arg \min_{t^S} Q_S(x_k^S + t^S; x_k)$$
Dimensionality reduction: a smaller subproblem

Assumption: $\Psi$ is separable

\[
\Psi(x) = x^T \Psi(x) = \Psi_1(x^{(1)}) + \Psi_2(x^{(2)}) + \Psi_3(x^{(3)}) + \Psi_4(x^{(4)}) + \Psi_5(x^{(5)})
\]

Reformulation of local approximation and subproblem

\[
Q_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle + \frac{1}{2} \langle t^S, H^S_k t^S \rangle
\]
\[
t_k^S \approx \arg\min_{t^S} Q_S(x_k^S + t^S; x_k)
\]
Dimensionality reduction: a smaller subproblem

Assumption: $\Psi$ is separable

$$
\Psi(x) = \begin{bmatrix}
\begin{array}{cccc}
\Psi_1(x^{(1)}) & & & \\
\Psi_2(x^{(2)}) & \Psi_3(x^{(3)}) & & \\
\Psi_4(x^{(4)}) & & \Psi_5(x^{(5)}) & \\
\end{array}
\end{bmatrix}
$$

Reformulation of local approximation and subproblem

$$
Q_S(x^S_k + t^S; x_k) := \Psi_S(x^S_k + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle + \frac{1}{2} \langle t^S, H^S_k t^S \rangle
$$

$$
t^S_k \approx \arg \min_{t^S} Q_S(x^S_k + t^S; x_k)
$$
Trade offs in FCD: inexact solution of subproblem

Interpretation: solve subproblem until
- $x^S_k + t^S_k$ reduces $Q_S$ compared to $x^S_k$, and
- $x^S_k + t^S_k$ is closer to optimality than $x^S_k$.

First condition: decrease of local model

$$Q_S(x^S_k + t^S_k; x_k) < Q_S(x^S_k; x_k)$$

Second condition: decrease distance from optimality for the local model

$$\|g_S(x^S_k + t^S_k)\|_2 \leq \eta^S_k \|g_S(x^S_k)\|_2, \quad \eta^S_k \in [0, 1)$$

Think $g_S(x^S_k + t^S)$ as an approximate subgradient of $Q_S$ at $t^S$
Flexible Coordinate Descent (FCD)

1: Input: Choose $x^0$

2: Loop: For $k = 1, 2, ...$, until termination criteria are met

3: Sample a subset of coordinates $S \subseteq [N]$ with probability $P(S)$

4: Calculate direction $t_k^S$ by approximately solving

$$t_k^S \approx \arg \min_{t^S} Q_S(x_k^S + t^S; x_k)$$

5: Backtracking line search along direction $t_k^S$. That is, find $\alpha \in (0, 1]$ such that a sufficient decrease condition is satisfied (explained in next slide).

6: Set $x_{k+1}^S := x_k^S + \alpha t_k^S$
Global convergence of FCD

Line search: find a step size $\alpha \in (0, 1]$ such that:
- the decrease in $F$ (objective function) is proportional to the decrease in its first order approximation with respect to the chosen coordinates.
- First order approximation of $F$ on $S$:

$$\ell_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle$$
Global convergence of FCD

Line search: find a step size \( \alpha \in (0, 1] \) such that:
- the decrease in \( F \) (objective function) is proportional to the decrease in its first order approximation with respect to the chosen coordinates.
- First order approximation of \( F \) on \( S \):

\[
\ell_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle
\]
Global convergence of FCD

Line search: find a step size $\alpha \in (0, 1]$ such that:
- the decrease in $F$ (objective function) is proportional to the decrease in its first order approximation with respect to the chosen coordinates.
- First order approximation of $F$ on $S$:

$$\ell_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle$$
Global convergence of FCD

Line search: find a step size $\alpha \in (0, 1]$ such that:
- the decrease in $F$ (objective function) is proportional to the decrease in its first order approximation with respect to the chosen coordinates.
- First order approximation of $F$ on $S$:

$$\ell_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle$$
Global convergence of FCD

Line search: find a step size $\alpha \in (0, 1]$ such that:
- the decrease in $F$ (objective function) is proportional to the decrease in its first order approximation with respect to the chosen coordinates.
- First order approximation of $F$ on $S$:

$$\ell_S(x_k^S + t^S; x_k) := \Psi_S(x_k^S + t^S) + f(x_k) + \langle \nabla_S f(x_k), t^S \rangle$$
Assumptions

Ψ is coordinate separable, for example:

\[ \Psi(x) = \|x\|_1 \]

Lipschitz continuity for all \( S \subseteq [N] \)

\[ \| \nabla_S f(x + U_S t^S) - \nabla_S f(x) \| \leq L_S \| t^S \|, \]

Bounded Hessian approximations

\[ 0 < \lambda_S \leq \lambda_{\text{min}}(H^S_k) \quad \text{and} \quad \lambda_{\text{max}}(H^S_k) \leq \Lambda_S, \quad \text{for all} \ S \subseteq [N] \text{ and } k. \]

Coordinate selection

\[ |S| = \xi > 0 \quad \text{and} \quad P(S) = 1/\binom{N}{\xi} \quad \forall S \]
High probability worst case iteration complexity

\[ F(x_K) - F^* \leq \epsilon \]

<table>
<thead>
<tr>
<th>( f )</th>
<th>subproblem</th>
<th>iterations K</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convex</td>
<td>exact</td>
<td>( \mathcal{O}\left( \frac{N}{\xi} \max_{S \in 2^\left[\mathcal{N}\right]} \frac{L^2_S}{\lambda^2_S} \right) )</td>
</tr>
<tr>
<td></td>
<td>inexact</td>
<td>( \mathcal{O}\left( \frac{N}{\xi} \max_{S \in 2^\left[\mathcal{N}\right]} \frac{L^2_S}{\lambda^3_S} + \frac{L^2_S}{\lambda^2_S} \right) )</td>
</tr>
<tr>
<td>Strongly convex</td>
<td>exact</td>
<td>( \mathcal{O}\left( \frac{N}{\xi} \log \frac{1}{\epsilon} \max_{S \in 2^\left[\mathcal{N}\right]} \frac{L^2_S}{\lambda^2_S} \right) )</td>
</tr>
<tr>
<td></td>
<td>inexact</td>
<td>( \mathcal{O}\left( \frac{N}{\xi} \log \frac{1}{\epsilon} \max_{S \in 2^\left[\mathcal{N}\right]} \frac{L^2_S}{\lambda^3_S} + \frac{L^2_S}{\lambda^2_S} \right) )</td>
</tr>
</tbody>
</table>
Numerical experiments: synthetic $\ell_1$ least squares

Solvers
- UCDC v.1: Single coordinate descent with separable model
- UCDC v.2: Block coordinate descent with separable model
- FCD v.1: $H_k^S := \text{diag}(\nabla^2_S f(x_k))$. FCD v.2: $H_k^S := \nabla^2_S f(x_k)$

Instance info: $N = 2^{21}$, $m = N/4$, $\text{nnz}(A) = 10^{-4} m N$ and $|S| \approx N/100$. 
Numerical experiments: real world binary classification

Info
Name: webspam
$N \approx 16$ million, $m \approx 0.02N$

Info
Name: kdd2010 (algebra)
$N \approx 20$ million, $m \approx 0.4N$
Signal Reconstruction
Example: undersampling and denoising
Example: undersampling
Problem formulation

\[
\text{minimize } \tau \|W^* x\|_1 + \frac{1}{2} \|Ax - b\|_2^2
\]

- \( b \in \mathbb{R}^m \)
- \( A \in \mathbb{R}^{m \times n}, \; m \ll n \)
- \( W \in \mathbb{E}^{n \times l}, \; \text{where } \mathbb{E} = \mathbb{R} \text{ or } \mathbb{C} \)
- \( \tau \in \mathbb{R}_+ \)

Assumption: Optimal \( x_\tau \) has a sparse image through \( W^* \), \( W^* x_\tau \) is sparse

For ease of presentation we will assume that \( W = I_n \)
Moreau-Yosida smoothing

Replace \( \|x\|_1 = \sup_{\|g\|_{\infty} \leq 1} g^T x \)

with \( \psi_\mu(x) = \sup_{\|g\|_{\infty} \leq 1} g^T x - \mu d(g) \),

where

\[
d(g) = n - \sum_{i=1}^{n} (1 - g_i^2)^{1/2}
\]

is the proximity function on \( \|g\|_{\infty} \leq 1 \)

Pseudo-Huber: \( \psi_\mu(x) = \sum_{i=1}^{n} \left( \sqrt{\mu^2 + x_i^2} - \mu \right) \)
Moreau-Yosida smoothing
Moreau-Yosida smoothing
Shortcomings of smoothing

$\nabla \psi_\mu(x)$ is highly nonlinear for $\mu \approx 0$

Inaccurate linearization of the first-order optimality conditions:

$$\tau \nabla \psi_\mu(x) + A^T(Ax - b) = 0$$
Shortcomings of smoothing

∇ψ_µ(x) is highly nonlinear for µ ≈ 0

Inaccurate linearization of the first-order optimality conditions:

τ∇ψ_µ(x) + A^T(Ax - b) = 0
Shortcomings of smoothing

$\nabla \psi_\mu(x)$ is highly nonlinear for $\mu \approx 0$

Inaccurate linearization of the first-order optimality conditions:

$\tau \nabla \psi_\mu(x) + A^T(Ax - b) = 0$
A better linearisation

\[ \tau \underbrace{Dx}_{\nabla \psi_\mu(x)} + A^T(Ax - b) = 0, \]

where \( D := \text{diag}(D_1, D_2, \cdots, D_n) \) with

\[ D_i := (\mu^2 + x_i^2)^{-\frac{1}{2}} \quad \forall i = 1, 2, \cdots, n \]

Set \( g = Dx \) and linearise the blue instead of the red equations.

\[ \tau g + A^T(Ax - b) = 0, \quad \tau g + A^T(Ax - b) = 0, \]

\[ g = Dx. \quad D^{-1}g = x. \]

Essentially we are solving the primal-dual problem:

\[ \min_x \frac{1}{2} \|Ax - b\|_2^2 + \tau \sup_y y^T x - \mu d(y) \]

A better linearisation

Example: $g_i = 0.99$

\[ g_i = D_i x_i \]

\[ D_i^{-1} g_i = x_i \]

Newton-type directions

Linearisation of the optimality conditions reduces to

\[ B(x, g) \Delta x = -\nabla f^\mu_T(x) \quad \text{where} \quad B := \tau \nabla^2 \psi(x, g) + A^T A. \quad (1) \]

\( \Delta g \) is inexpensive to calculate

- Replace nonsymmetric \( B \) with symmetric \( \text{sym}(B) := \tau/2(\nabla^2 \psi + \nabla^2 \psi^T) + A^T A \)
- \( \text{sym}(B) \succeq 0 \) if \( \|g\|_\infty \leq 1 \)
- Use PCG to solve (1) approximately
- Efficient preconditioner for \( \text{sym}(B) \) (main contribution)
Primal-dual Newton Conjugate Gradient (pdNCG)

1: **Input:** $x^0, g^0$, where $\|g^0\|_\infty \leq 1$

2: **Loop:** For $k = 1, 2, \ldots$, until termination criteria are met

3: Calculate primal-dual directions $\Delta x^k, \Delta g^k$ *approximately* with PCG

4: $g^{k+1} := P_{\|\cdot\|_\infty \leq 1}(g^k + \Delta g^k)$, $P_{\|\cdot\|_\infty \leq 1}(\cdot)$ is the projection on the $\ell_\infty$ ball

5: Perform backtracking line search for the direction $\Delta x^k$

6: Set $x^{k+1} := x^k + \alpha \Delta x^k$

Properties of $\text{sym}(B) := \frac{\tau}{2}(\nabla^2 \psi + \nabla^2 \psi^\top) + A^T A$

- Rows of $A$ are nearly orthogonal, i.e. $\|AA^T - I_n\|_2 \leq \delta$, where $\delta$ is small

- Subsets of columns of $A$ are nearly orthogonal

- For all at most $q$-sparse $x \in \mathbb{R}^n$ we have that $\frac{\tau}{2}\|\nabla^2 \psi + \nabla^2 \psi^\top\|_2 = O(\frac{\tau}{\mu})$
Properties of \( \text{sym}(B) := \tau/2(\nabla^2\psi + \nabla^2\psi^T) + A^TA \)

- Rows of \( A \) are nearly orthogonal, i.e. \( \|AA^T - I_n\|_2 \leq \delta \), where \( \delta \) is small

- Subsets of columns of \( A \) are nearly orthogonal

\[
\tilde{A} = \begin{bmatrix}
\end{bmatrix}
\]

- For all at most \( q \)-sparse \( x \in R^n \) we have that \( \tau/2\|\nabla^2\psi + \nabla^2\psi^T\|_2 = \mathcal{O}(\tau/\mu) \)
Properties of \( \text{sym}(B) := \frac{\tau}{2}(\nabla^2\psi + \nabla^2\psi^T) + A^T A \)

- Rows of \( A \) are nearly orthogonal, i.e. \( \|AA^T - I_n\|_2 \leq \delta \), where \( \delta \) is small

- Subsets of columns of \( A \) are nearly orthogonal

For all at most \( q \)-sparse \( x \in R^n \) we have that \( \frac{\tau}{2}\|\nabla^2\psi + \nabla^2\psi^T\|_2 = \mathcal{O}(\tau/\mu) \)
Properties of $\text{sym}(B) := \frac{\tau}{2}(\widetilde{\nabla^2}\psi + \widetilde{\nabla^2}\psi^\top) + A^\top A$

- Rows of $A$ are nearly orthogonal, i.e. $\|AA^\top - I_n\|_2 \leq \delta$, where $\delta$ is small.

- There exists $\delta_q < 1/2$ such that Restricted Isometry Property (RIP) holds:

$$
(1 - \delta_q)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_q)\|x\|_2^2,
$$

for all at most $q$-sparse $x \in \mathbb{R}^n$.

- For all at most $q$-sparse $x \in \mathbb{R}^n$ we have that $\tau/2\|\widetilde{\nabla^2}\psi + \widetilde{\nabla^2}\psi^\top\|_2 = \mathcal{O}(\tau/\mu)$.
Preconditioner

Approximate

\[
\text{sym}(B) := \tau/2(\widehat{\nabla^2\psi} + \widehat{\nabla^2\psi}^\top) + A^\top A
\]

with

\[
\tilde{\mathcal{N}} := \tau/2(\widehat{\nabla^2\psi} + \widehat{\nabla^2\psi}^\top) + \rho l_m, \quad \text{where} \quad \rho \in [\delta_q, 1/2]
\]

**Theorem (Brief description).** Let \( \lambda \in \text{spec}(\tilde{\mathcal{N}}^{-1}\text{sym}(B)) \), then close to \( x_\tau \) the following holds

\[
- |\lambda - 1| \leq \frac{1}{2}(\chi + 1 + (5\chi^2 - 2\chi + 1)^{\frac{1}{2}})O(\mu), \quad \chi := 1 + \delta - \rho
\]

Sketch of proof: 1) close to the \( q \)-sparse optimal solution and for \( \mu \approx 0 \)

\[
\tau/2\|\widehat{\nabla^2\psi} + \widehat{\nabla^2\psi}^\top\|_2 = O(\tau/\mu) \text{ dominates } \|A^\top A\|_2
\]

2) Near column and row orthogonality of \( A \) (RIP + \( \|AA^\top - I_n\|_2 \leq \delta, \ \delta \text{ is small} \)\)
Spectrum in practice ($\mu = 1.0e-5$)
Dependence of pdNCG on problem size: $W = I_n$

<table>
<thead>
<tr>
<th>$n$</th>
<th>Processors</th>
<th>Memory (terabytes)</th>
<th>Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2^{30} \approx 1$ billion</td>
<td>64</td>
<td>0.192</td>
<td>1,923</td>
</tr>
<tr>
<td>$2^{32}$</td>
<td>256</td>
<td>0.768</td>
<td>1,968</td>
</tr>
<tr>
<td>$2^{34}$</td>
<td>1024</td>
<td>3.072</td>
<td>1,986</td>
</tr>
<tr>
<td>$2^{36}$</td>
<td>4096</td>
<td>12.288</td>
<td>1,970</td>
</tr>
<tr>
<td>$2^{38}$</td>
<td>16384</td>
<td>49.152</td>
<td>1,990</td>
</tr>
<tr>
<td>$2^{40} \approx 1$ trillion</td>
<td>65536</td>
<td>196.608</td>
<td>2,006</td>
</tr>
</tbody>
</table>

**inner products are implemented in parallel**
Benchmarks for Total Variations ($W$: tridiagonal and rank deficient)
Dependence of pdNCG on problem size

The image Shepp-Logan has been used for this experiment, 25% of the measurements are used and SNR is fixed to 15 dB

<table>
<thead>
<tr>
<th>Solver</th>
<th>64 × 64</th>
<th>128 × 128</th>
<th>256 × 256</th>
<th>512 × 512</th>
<th>1024 × 1024</th>
</tr>
</thead>
<tbody>
<tr>
<td>TFOCS</td>
<td>17</td>
<td>23</td>
<td>56</td>
<td>260</td>
<td>1018</td>
</tr>
<tr>
<td>TVAL3</td>
<td>5</td>
<td>8</td>
<td>37</td>
<td>99</td>
<td>365</td>
</tr>
<tr>
<td>pdNCG</td>
<td>2</td>
<td>6</td>
<td>12</td>
<td>62</td>
<td>250</td>
</tr>
</tbody>
</table>
Dependence of pdNCG on $\mu$

25% of the measurements are used and SNR is fixed to 15 dB

<table>
<thead>
<tr>
<th>$\mu$</th>
<th>House</th>
<th>Peppers</th>
<th>Lena</th>
<th>Fingerprint</th>
<th>Boat</th>
<th>Barbara</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0e-07</td>
<td>11</td>
<td>11</td>
<td>49</td>
<td>282</td>
<td>68</td>
<td>102</td>
</tr>
<tr>
<td>1.0e-10</td>
<td>12</td>
<td>13</td>
<td>66</td>
<td>282</td>
<td>81</td>
<td>103</td>
</tr>
<tr>
<td>1.0e-13</td>
<td>12</td>
<td>13</td>
<td>66</td>
<td>310</td>
<td>71</td>
<td>100</td>
</tr>
</tbody>
</table>
### Dependence on the level of noise

SNR is measured in dB, 25% of the measurements are used.

<table>
<thead>
<tr>
<th>Solver</th>
<th>SNR</th>
<th>House</th>
<th>Peppers</th>
<th>Lena</th>
<th>Fingerprint</th>
<th>Boat</th>
<th>Barbara</th>
</tr>
</thead>
<tbody>
<tr>
<td>TFOCS</td>
<td>60</td>
<td>56</td>
<td>56</td>
<td>262</td>
<td>274</td>
<td>261</td>
<td>260</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>56</td>
<td>56</td>
<td>262</td>
<td>268</td>
<td>260</td>
<td>261</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>56</td>
<td>56</td>
<td>260</td>
<td>267</td>
<td>260</td>
<td>259</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>56</td>
<td>56</td>
<td>266</td>
<td>280</td>
<td>260</td>
<td>259</td>
</tr>
<tr>
<td>TVAL3</td>
<td>60</td>
<td>7</td>
<td>4</td>
<td>17</td>
<td>22</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>6</td>
<td>4</td>
<td>17</td>
<td>23</td>
<td>17</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>4</td>
<td>9</td>
<td>39</td>
<td>65</td>
<td>41</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>13</td>
<td>16</td>
<td>60</td>
<td>76</td>
<td>61</td>
<td>63</td>
</tr>
<tr>
<td>pdNCG</td>
<td>60</td>
<td>18</td>
<td>24</td>
<td>115</td>
<td>114</td>
<td>121</td>
<td>113</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>18</td>
<td>24</td>
<td>115</td>
<td>113</td>
<td>121</td>
<td>112</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>19</td>
<td>17</td>
<td>78</td>
<td>113</td>
<td>78</td>
<td>107</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td><strong>10</strong></td>
<td><strong>9</strong></td>
<td><strong>41</strong></td>
<td><strong>229</strong></td>
<td><strong>44</strong></td>
<td>111</td>
</tr>
</tbody>
</table>
Dependence on the number of measurements

SNR is fixed to 15 dB

<table>
<thead>
<tr>
<th>Solver</th>
<th>m</th>
<th>House</th>
<th>Peppers</th>
<th>Lena</th>
<th>Fingerprint</th>
<th>Boat</th>
<th>Barbara</th>
</tr>
</thead>
<tbody>
<tr>
<td>TFOCS</td>
<td>75%</td>
<td>61</td>
<td>61</td>
<td>255</td>
<td>262</td>
<td>260</td>
<td>273</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>60</td>
<td>57</td>
<td>260</td>
<td>256</td>
<td>263</td>
<td>268</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>57</td>
<td>60</td>
<td>253</td>
<td>250</td>
<td>273</td>
<td>262</td>
</tr>
<tr>
<td>TVAL3</td>
<td>75%</td>
<td>13</td>
<td>14</td>
<td>55</td>
<td>56</td>
<td>54</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>11</td>
<td>14</td>
<td>55</td>
<td>71</td>
<td>57</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>14</td>
<td>17</td>
<td>57</td>
<td>72</td>
<td>61</td>
<td>64</td>
</tr>
<tr>
<td>pdNCG</td>
<td>75%</td>
<td>9</td>
<td>9</td>
<td>74</td>
<td>196</td>
<td>62</td>
<td>108</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>8</td>
<td>9</td>
<td>57</td>
<td>217</td>
<td>91</td>
<td>106</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>10</td>
<td>9</td>
<td>40</td>
<td>222</td>
<td>42</td>
<td>113</td>
</tr>
</tbody>
</table>
Single pixel camera benchmarks

TFOCS, 25 sec.  
24 sec.  
37 sec.  
26 sec.  
49 sec.  
pdNCG, 7 sec.  
15 sec.  
15 sec.  
27 sec.  
33 sec.
Thank you!

**Software and experiments:** http://www.maths.ed.ac.uk/ERGO/pdNCG/

**Convergence and iteration complexity for pdNCG:** A second-order method for strongly-convex l1-regularization problems, Math. Prog. A (accepted), 2015

**Preconditioner for pdNCG:** A Preconditioner for a Primal-Dual Newton Conjugate Gradients Method for Compressed Sensing Problems, Technical Report ERGO-14-021

**Flexible Coordinate Descent:** Robust Block Coordinate Descent, Technical Report ERGO-14-010