Compressed Sensing: a Subgradient Descent Method for Missing Data Problems

ANZIAM, Jan 30 – Feb 3, 2011

Jonathan M. Borwein FRSC FAAAS FBAS FAA

Jointly with

D. Russell Luke, University of Goettingen

Director, CARMA, the University of Newcastle

Revised: 02/02/2011

Australian and New Zealand Industrial and Applied Mathematics

CARMA

A Central Problem: ℓ_0 minimization

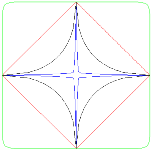
Given a linear map $A : \mathbb{R}^n \to \mathbb{R}^m$ full-rank with $0 < m < n$, solve

Program

$$(\mathcal{P}_0) \qquad \underset{x\in\mathbb{R}^n}{\textit{minimize}} \quad \|x\|_0 \qquad \textit{subject to} \quad Ax = b$$

where $\|x\|_0 := \sum_j |\mathit{sign}(x_j)|$ *with* $\mathrm{sign}\,(0) := 0$.

- $\|x\|_0 = \lim_{p\to 0^+} \sum_j |x_j|^p$ is a *metric* but not a norm.



p-balls for $1/5$, $1/2$, 1, 100

- *Combinatorial optimization problem* (hard to solve).

Central Problem: ℓ_0 minimization

Solve instead

Program

$$(\mathcal{P}_1) \qquad \underset{x\in\mathbb{R}^n}{\textit{minimize}} \quad \|x\|_1 \qquad \textit{subject to} \quad Ax = b$$

where $\|x\|_1$ is the usual ℓ_1 norm.

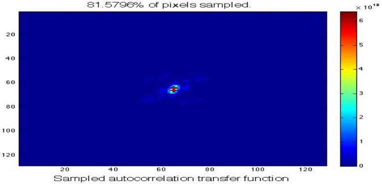
- ℓ_1 minimization now routine in statistics and elsewhere for "missing data" under-determined problems.

A nonsmooth convex, actually *linear, programming problem* ... easy to solve for small problems.

- Let's illustrate by trying to solve the problem for **x** a 512×512 image ...

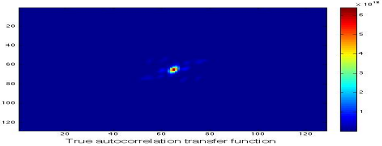
Application: Crystallography

Given data:



(autocorrelation transfer function — 'ATF' — missing pixels)

Desired reconstruction:



(The true ATF with all pixels)

Application: Crystallography

Formulate as: Solve

Program

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\textit{minimize}} & \|x\|_1 \\ \textit{subject to} & x \in C \end{array}$$

where $C := \{x \in \mathbb{R}^n \mid Ax = b\}$ *for a linear* $A : \mathbb{R}^n \to \mathbb{R}^m$ $(m < n)$*.*

- Could apply Douglas-Rachford iteration — originated in 1956 for convex heat transfer problems (Laplacian):

$$x_{n+1} := \frac{1}{2}\left(R_{f_1}R_{f_2} + I\right)(x_n)$$

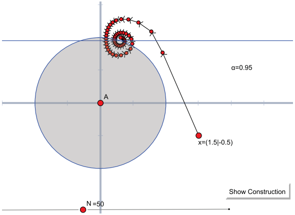
where

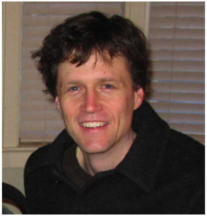
$$R_{f_j}x := 2\,\mathrm{prox}_{\alpha,f_j}\,x - x$$

for $f_1(x) := \|x\|_1$ and $f_2(x) := \iota_C(x)$, and $\alpha > 0$ fixed (a generalized best approximation or prox-mapping).

Application: Crystallography

- Great strategy for big problems, but convergence is *(arbitrarily) slow* and accuracy is likely to be poor ...





(Douglas-Rachford and Russell Luke)

It seemed to me that a better approach was to think about real dynamics and see where they go. Maybe they go to the [classical] equilibrium solution and maybe they don't.
— Peter Diamond (2010 Economics co-Nobel)

Motivation

A variational/geometrical *interpretation* of the *Candes-Tao* (2004) probabilistic criterion for the solution to $(\mathcal{P}_1)$ to be unique and exactly match the true signal x_*.

- *As a by-product,* better practical methods for solving the underlying problem.
- Aim to use entropy/penalty ideas and duality and also prove some rigorous theorems.

- The counterpart paper (largely successful):

  J. M. Borwein and D. R. Luke, "Entropic Regularization of the ℓ_0 function." In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, *Springer Optimization and Its Applications*. In press, 2011. Available at http://carma.newcastle.edu.au/jon/sensing.pdf.

Motivation

A variational/geometrical *interpretation* of the *Candes-Tao* (2004) probabilistic criterion for the solution to $(\mathcal{P}_1)$ to be unique and exactly match the true signal x_*.

- *As a by-product,* better practical methods for solving the underlying problem.
- Aim to use entropy/penalty ideas and duality and also prove some rigorous theorems.

- The counterpart paper (largely successful):

  J. M. Borwein and D. R. Luke, "Entropic Regularization of the ℓ_0 function." In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, *Springer Optimization and Its Applications*. In press, 2011. Available at http://carma.newcastle.edu.au/jon/sensing.pdf.

Motivation

A variational/geometrical *interpretation* of the *Candes-Tao* (2004) probabilistic criterion for the solution to $(\mathcal{P}_1)$ to be unique and exactly match the true signal x_*.

- *As a by-product,* better practical methods for solving the underlying problem.
- Aim to use entropy/penalty ideas and duality and also prove some rigorous theorems.

- The counterpart paper (largely successful):

J. M. Borwein and D. R. Luke, “Entropic Regularization of the ℓ_0 function.” In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, *Springer Optimization and Its Applications*. In press, 2011. Available at `http://carma.newcastle.edu.au/jon/sensing.pdf`.

Outline

Fenchel duality

The **Fenchel-Legendre conjugate** $f^* : X^* \to]-\infty, +\infty]$ of f is

$$f^*(x^*) := \sup_{x \in X} \{\langle x^*, x\rangle - f(x)\}.$$

- For the ℓ_1 problem, the norm is proper, convex, lsc and $b \in \mathrm{core}\,(A\,\mathrm{dom}\, f)$ so **strong Fenchel duality** holds.

That is:

Program

$$\inf_{x \in \mathbb{R}^n} \{\|x\|_1 : Ax = b\} = \sup_{y \in \mathbb{R}^m} \{\langle b, y\rangle - \|(A^* y)\|_1^*\}$$

where

$$\|x^*\|_1{}^* = \iota_{[-1,1]}(x^*)$$

is zero on the supremum ball and is infinite otherwise.

Elementary Observations

The **dual** to $(\mathcal{P}_1)$ is

Program

$$(\mathcal{D}_1) \qquad \underset{y\in\mathbb{R}^m}{\text{maximize}} \quad b^T y \qquad \text{subject to} \quad (A^* y)_j \in [-1,1] \quad j = 1,2,\ldots,n.$$

- The solution includes a vertex of the constraint polyhedron.
- Uniqueness of primal solutions depends on whether dual solutions live on edges or faces of the dual polyhedron.

- We deduce that if a solution $\bar{x}$ to $(\mathcal{P}_1)$ is unique, then

$$m \geq \{ \text{ number of active constraints in } (\mathcal{D}_1) \} = \|\bar{x}\|_0.$$

Elementary observations

The ℓ_0 function is proper, lsc but not convex, so only weak Fenchel duality holds:

Program

$$\inf_{x\in\mathbb{R}^n} \{\|x\|_0 \mid Ax = b\} \geq \sup_{y\in\mathbb{R}^m} \{\langle b,\ y\rangle - \|(A^*y)\|_0^*\}.$$

where

$$\|x^*\|_0^* := \begin{cases} 0 & x^* = 0 \\ +\infty & \textit{else} \end{cases}$$

Elementary observations

In other words, the dual to $(\mathcal{P}_0)$ is

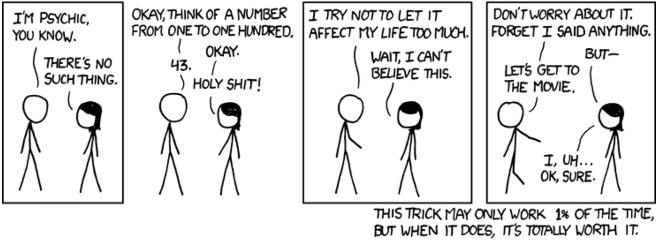
Program

$$(\mathcal{D}_0) \quad \begin{array}{ll} \underset{y \in \mathbb{R}^m}{\textit{maximize}} & b^T y \\ \textit{subject to} & A^* y = 0. \end{array}$$

- *primal problem* is a combinatorial optimization problem.
- *dual problem*, however, is a linear program, which is finitely terminating.
- The solution to the dual problem is trivial: $\bar{y} = 0$ ... which tells us *nothing* useful about the primal problem.

The Main Idea

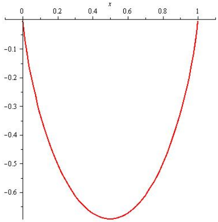
- **Relax and Regularize** the dual — and either solve this directly, or solve the corresponding regularized primal problem, or some mixture.
- Performance will be "**part art and part science**" and tied to the specific problem at hand.
- We do not expect a method which works all of the time ...



The Fermi-Dirac Entropy (1926)

The *Fermi-Dirac entropy* is ideal for $[0,1]$ problems:

$$\mathcal{FD}(x) := \sum_j x_j \log(x_j) + (1 - x_j)\log(1 - x_j).$$



Fermi-Dirac in 1-dim

- A Legendre barrier function with smooth finite conjugate

$$\mathcal{FD}^*(y) := \sum_j \log\left(1 + e^{y_j}\right).$$

Regularization/Relaxation: Fermi-Dirac Entropy

For $L, \varepsilon > 0$, define a shifted nonnegative entropy:

$$f^*_{\varepsilon,L}(x) := \sum_{j-1}^{n} \left[\varepsilon \left(\frac{(L+x_j)\ln(L+x_j) + (L-x_j)\ln(L-x_j)}{2L\ln(2)} - \frac{\ln(L)}{\ln(2)} \right) \right]$$

$$\text{for } x \in [-L, L]^n$$

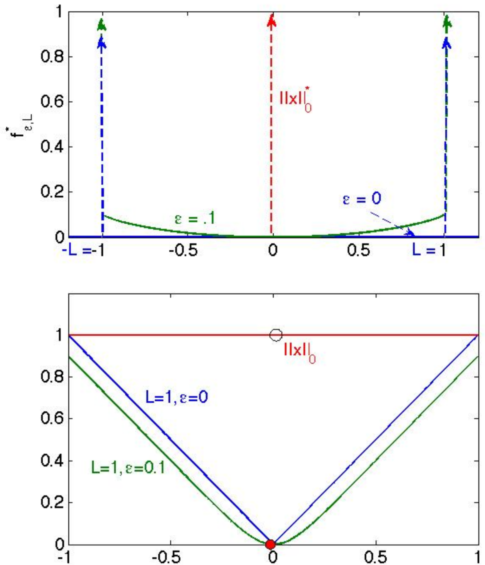
$$:= +\infty \qquad \text{for } \|x\|_\infty > L.$$

Then

$$f^{**}_{\varepsilon,L}(x) = \sum_{j=1}^{n} \left[\frac{\varepsilon}{\ln(2)} \ln\left(4^{x_j L/\varepsilon} + 1\right) - x_j L - \varepsilon \right]. \qquad (1)$$

- f^* is proper, convex and lsc, thus $f^{***} = f^*$. We set $f := f^{**}$.

Regularization/Relaxation: Fermi-Dirac Entropy



Regularization/Relaxation: Fermi-Dirac Entropy

For $L > 0$ fixed, in the limit as $\varepsilon \to 0$ we have

$$\lim_{\varepsilon\to 0} f^*_{\varepsilon,L}(y) = \begin{cases} 0 & y \in [-L, L] \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{\varepsilon\to 0} f_{\varepsilon,L}(x) = L|x|.$$

For $\varepsilon > 0$ fixed we have

$$\lim_{L\to 0} f^*_{\varepsilon,L}(y) = \begin{cases} 0 & y = 0 \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{L\to 0} f_{\varepsilon,L}(x) := 0.$$

- $\|\cdot\|_0$ and $f^*_{\varepsilon 0}$ have the same conjugate;
- $\|\cdot\|_0^{**} \neq \|\cdot\|_0$ while $f^{***}_{\varepsilon 0} = f^*_{\varepsilon 0}$;
- $f_{\varepsilon,L}$ and $f^*_{\varepsilon,L}$ are convex and smooth on the interior of their domains for all $\varepsilon, L > 0$.

This is in contrast to the *metrics* of the form $\left(\sum_j |x_j|^p\right)$ which are nonconvex for $p < 1$.

Regularization/Relaxation: Fermi-Dirac Entropy

For $L > 0$ fixed, in the limit as $\varepsilon \to 0$ we have

$$\lim_{\varepsilon \to 0} f^*_{\varepsilon,L}(y) = \begin{cases} 0 & y \in [-L, L] \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{\varepsilon \to 0} f_{\varepsilon,L}(x) = L|x|.$$

For $\varepsilon > 0$ fixed we have

$$\lim_{L \to 0} f^*_{\varepsilon,L}(y) = \begin{cases} 0 & y = 0 \\ +\infty & \text{else} \end{cases} \quad \text{and} \quad \lim_{L \to 0} f_{\varepsilon,L}(x) := 0.$$

- $\|\cdot\|_0$ and $f^*_{\varepsilon 0}$ have the same conjugate;
- $\|\cdot\|_0^{**} \neq \|\cdot\|_0$ while $f^{***}_{\varepsilon 0} = f^*_{\varepsilon 0}$;
- $f_{\varepsilon,L}$ and $f^*_{\varepsilon,L}$ are convex and smooth on the interior of their domains for all $\varepsilon, L > 0$.

This is in contrast to the *metrics* of the form $\left(\sum_j |x_j|^p\right)$ which are nonconvex for $p < 1$.

Regularization/Relaxation: FD Entropy

Hence we aim to solve

Program

$$(\mathcal{D}_{L,\varepsilon}) \qquad \underset{y \in \mathbb{R}^m}{\textit{minimize}}\, f^*_{L,\varepsilon}(A^*y) - \langle b,\ y\rangle$$

for appropriately updated L and ε.

- This is a **convex optimization problem**, so equivalently we solve the inclusion:

$$0 \in A\partial f^*_{L,\varepsilon}(A^*y) - b \qquad (\textbf{DI})$$

- We can also model more realistic relaxed inequality constraints such as $\|Ax - b\| \leq \delta$ (JMB-Lewis)

Regularization/Relaxation: FD Entropy

Hence we aim to solve

Program

$$(\mathcal{D}_{L,\varepsilon}) \qquad \underset{y\in\mathbb{R}^m}{\textit{minimize}}\, f^*_{L,\varepsilon}(A^*y) - \langle b,\ y\rangle$$

for appropriately updated L and ε.

- This is a **convex optimization problem**, so equivalently we solve the inclusion:

$$0 \in A\partial f^*_{L,\varepsilon}(A^*y) - b \qquad (\textbf{DI})$$

- We can also model more realistic relaxed inequality constraints such as $\|Ax - b\| \leq \delta$ (JMB-Lewis)

Regularization/Relaxation: FD Entropy

Hence we aim to solve

Program

$$(\mathcal{D}_{L,\varepsilon}) \qquad \underset{y\in\mathbb{R}^m}{\textit{minimize}}\, f^*_{L,\varepsilon}(A^*y) - \langle b,\ y\rangle$$

for appropriately updated L and ε.

- This is a **convex optimization problem**, so equivalently we solve the inclusion:

$$0 \in A\partial f^*_{L,\varepsilon}(A^*y) - b \qquad (\textbf{DI})$$

- We can also model more realistic relaxed inequality constraints such as $\|Ax - b\| \le \delta$ (JMB-Lewis)

Outline

$\varepsilon = 0$: $f^*_{L,0} = \iota_{[-L,L]^n}$

Solve

$$0 \in A\partial f^*_{L,0}(A^*y) - b$$

via **subgradient descent**:

Program

Given y_-, *choose* $v_- \in \partial f^*_{L,0}(A^*y_-)$, $\lambda_- \to 0$ *and construct* y_+ *as*

$$y_+ := y_- + \lambda_- (b - Av_-).$$

'Only' two issues remain: Direction and Step size

(a) how to choose direction $v_- \in \partial f^*_{L,0}(A^*y_-)$

(b) how to choose step length λ_-.

$\varepsilon = 0$: (a) Choose $v_- \in \partial f^*_{L,0}(A^* y_-)$

Recall that $f^*_{L,0} = \iota_{[-L,L]^n}$ so

$$\partial\iota_{[-L,L]^n}(x^*) = N_{[-L,L]}(x^*) \quad \text{(normal cone)}$$
$$= \{v \in \mathbb{R}^n \mid \pm v_j \le \ 0 \ (j \in \mathbb{J}_\pm), \ v_j = 0 \ (j \in \mathbb{J}_0)\}$$

where

$$\mathbb{J}_- := \{j \in \mathbb{N} \mid x_j = -L\}, \mathbb{J}_+ := \{j \in \mathbb{N} \mid x_j = L\}$$

and

$$\mathbb{J}_0 := \{j \in \mathbb{N} \mid x_j \in]-L, L[\}.$$

Program

Choose $v_- \in N_{[-L,L]}(A^* y_-)$ *to be the solution to*

$$\begin{array}{ll} \underset{v\in\mathbb{R}^n}{\text{minimize}} & \frac{1}{2}\|b - Av\|^2 \\ \text{subject to} & v \in N_{[-L,L]^n}(A^* y_-) \end{array}$$

$\varepsilon = 0$: Choose $v_- \in \partial f^*_{L,0}(A^* y_-)$

That is:

Program

$$(\mathcal{P}_{v_-}) \qquad \underset{v\in\mathbb{R}^n}{\textit{minimize}} \quad \frac{1}{2}\|b - Av\|^2 \qquad \textit{subject to} \quad v \in N_{[-L,L]^n}(A^* y_-)$$

Defining

$$B := \{v \in \mathbb{R}^n \mid Av = b\},$$

we reformulate as:

Program

$$(\mathcal{P}_{v_-}) \qquad \underset{v\in\mathbb{R}^n}{\textit{minimize}} \; \frac{\beta}{2(1-\beta)} \operatorname{dist}^2(v, B) + \iota_{N_{[-L,L]^n}(A^* y_-)}(v),$$

for given $\frac{1}{2} < \beta < 1$.

$\varepsilon = 0$: Choose $v_- \in \partial f^*_{L,0}(A^* y_-)$

Approximate (dynamic) averaged alternating reflections:

Program

We choose $v^{(0)} \in \mathbb{R}^n$. *For* $\nu \in \mathbb{N}$ *set*

$$v^{(\nu+1)} := \frac{1}{2}\left(R_1\left(R_2 v^{(\nu)} + \varepsilon_\nu\right) + \rho_\nu + v^{(\nu)}\right), \tag{2}$$

- where
  - $R_1 x := 2\,\mathrm{prox}_{\frac{\beta}{2(1-\beta)}\,\mathrm{dist}\,(v,B)^2}\, x - x$
  - $R_2 x := 2\,\mathrm{prox}_{\iota_{N_{[-L,L]^n}(A^* y_-)}}\, x - x$

- $\{\varepsilon_\nu\}$ and $\{\rho_\nu\}$ are the errors at each iteration, assumed summable.
- In the dynamic version we may adjust L as we go.

$\varepsilon = 0$: Choose $v_- \in \partial f^*_{L,0}(A^* y_-)$

- Luke (2005–08) shows (2) is equivalent to Inexact Relaxed Averaged Alternating Reflections:

Program

Choose $v^{(0)} \in \mathbb{R}^n$ and $\beta \in [1/2, 1[$. For $\nu \in \mathbb{N}$ set

$$v^{(\nu+1)} := \frac{\beta}{2}\left(R_B\left(R_{N_{[-L,L]^n}(A^*y_-)} v^{(\nu)} + \varepsilon_n\right) + \rho_n + v^{(\nu)}\right) + (1-\beta)\left(P_{N_{[-L,L]^n}(A^*y_-)} v^{(\nu)} + \frac{\varepsilon_n}{2}\right). \quad (3)$$

where $R_B := 2P_B - I$ also for $R_{N_{[-L,L]^n}(A^*y_-)}$. We can show:

Lemma (Luke 2008, Combettes 2004)

The sequence $\{v^{(\nu)}\}_{\nu=1}^{\infty}$ converges to $\bar{v}$ where $P_B\bar{v}$ solves $(\mathcal{P}_{v_-})$.

$\varepsilon = 0$: (b) Choose λ_-

Exact line search: choose largest λ_- that solves

Program

$$\underset{\lambda \in \mathbb{R}_+}{\textit{minimize}}\; f_{L,0}^*(A^*y_- + A^*\lambda(b - Av_-))$$

Note that $f_{L,0}^*(A^*y_- + A^*\lambda(b - Av_-)) = 0 = \min f$ for all $A^*y_- + A^*\lambda(b - Av_-) \in [-L, L]^n$. So we solve:

Program (Exact line-search)

$$(\mathcal{P}_\lambda) \quad \begin{array}{ll} \underset{\lambda \in \mathbb{R}_+}{\textit{minimize}} & -\lambda \\ \textit{subject to} & \begin{array}{c} \lambda(A^*(b - Av_-))_j \leq L - (A^*y_-)_j \\ \lambda(A^*(b - Av_-))_j \geq -L - (A^*y_-)_j \\ j = 1, \ldots, n \end{array} \end{array}$$

$\varepsilon = 0$: Choose λ_-

Exact line search is practicable: Define

$$\begin{aligned} \mathbb{J}_+ &:= \{j \mid (A^*(b - Av_-))_j > TOL\}, \\ \mathbb{J}_- &:= \{j \mid (A^*(b - Av_-))_j < -TOL\} \end{aligned}$$

and set

$$\lambda_- := \min \left\{ \begin{array}{c} \min_{j \in \mathbb{J}_+} \{(L - (A^*y_-)_j)/(A^*(b - Av_-))_j\}, \\ \min_{j \in \mathbb{J}_-} \{(-L - (A^*y_-)_j)/(A^*(b - Av_-))_j\} \end{array} \right\}$$

- Relies on 'simulating' exact arithmetic ... harder for $\varepsilon > 0$.
- Full algorithm *terminates* when current v_- is such that $\mathbb{J}_+ = \mathbb{J}_- = \emptyset$.

$\varepsilon = 0$: Choose λ_-

Exact line search is practicable: Define

$$\begin{aligned} \mathbb{J}_+ &:= \{j \mid (A^*(b - Av_-))_j > TOL\}, \\ \mathbb{J}_- &:= \{j \mid (A^*(b - Av_-))_j < -TOL\} \end{aligned}$$

and set

$$\lambda_- := \min \left\{ \begin{array}{c} \min_{j \in \mathbb{J}_+} \{(L - (A^* y_-)_j)/(A^*(b - Av_-))_j\}, \\ \min_{j \in \mathbb{J}_-} \{(-L - (A^* y_-)_j)/(A^*(b - Av_-))_j\} \end{array} \right\}$$

- Relies on 'simulating' exact arithmetic ... harder for $\varepsilon > 0$.
- Full algorithm *terminates* when current v_- is such that $\mathbb{J}_+ = \mathbb{J}_- = \emptyset$.

Choosing dynamically reweighted weights: details

The algorithm in our paper has a nice convex-analytic criterion for choosing the reweighting parameter L^k:

- L^k is chosen so the **projection of the data onto the normal cone** of the rescaled problem at the rescaled iterate y^k lies in the relative interior to said normal cone.
- This guarantees **orthogonality of the search directions** to the (rescaled) active constraints.
- What are optimal (in some sense) reweightings L^k (*dogmatic* or otherwise)?

Choosing dynamically reweighted weights: details

The algorithm in our paper has a nice convex-analytic criterion for choosing the reweighting parameter L^k:

- L^k is chosen so the **projection of the data onto the normal cone** of the rescaled problem at the rescaled iterate y^k lies in the relative interior to said normal cone.
- This guarantees **orthogonality of the search directions** to the (rescaled) active constraints.
- What are optimal (in some sense) reweightings L^k (*dogmatic* or otherwise)?

Choosing dynamically reweighted weights: details

The algorithm in our paper has a nice convex-analytic criterion for choosing the reweighting parameter L^k:

- L^k is chosen so the **projection of the data onto the normal cone** of the rescaled problem at the rescaled iterate y^k lies in the relative interior to said normal cone.
- This guarantees **orthogonality of the search directions** to the (rescaled) active constraints.
- What are optimal (in some sense) reweightings L^k (*dogmatic* or otherwise)?

Choosing dynamically reweighted weights: details

The algorithm in our paper has a nice convex-analytic criterion for choosing the reweighting parameter L^k:

- L^k is chosen so the **projection of the data onto the normal cone** of the rescaled problem at the rescaled iterate y^k lies in the relative interior to said normal cone.
- This guarantees **orthogonality of the search directions** to the (rescaled) active constraints.
- What are optimal (in some sense) reweightings L^k (*dogmatic* or otherwise)?

Outline

Sufficient sparsity

The **mutual coherence** of a matrix A is defined as

$$\mu(A) := \max_{1\le k,j\le n,\ k\ne j} \frac{|a_k^T a_j|}{\|a_k\|\|a_j\|}$$

where $0/0 := 1$ and a_j denotes the jth column of A.

- Mutual coherence measures the dependence between columns of A.
- The mutual coherence of unitary matrices, for instance, is zero; for matrices with columns of zeros, the mutual coherence is 1.
- What is a variational analytic/geometric interpretation (constraint qualification or the like) of the mutual coherence condition (4) below?

Sufficient sparsity

The **mutual coherence** of a matrix A is defined as

$$\mu(A) := \max_{1 \le k,j \le n,\ k \ne j} \frac{|a_k^T a_j|}{\|a_k\| \|a_j\|}$$

where $0/0 := 1$ and a_j denotes the jth column of A.

- Mutual coherence measures the dependence between columns of A.
- The mutual coherence of unitary matrices, for instance, is zero; for matrices with columns of zeros, the mutual coherence is 1.
- What is a variational analytic/geometric interpretation (constraint qualification or the like) of the mutual coherence condition (4) below?

Sufficient sparsity

Lemma (uniqueness of sparse representations (Donoho–Elad, 2003))

Let $A \in \mathbb{R}^{m\times n}$ $(m < n)$ be full rank. If there exists an element x^* such that $Ax^* = b$ and

$$\|x^*\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(A)}\right), \tag{4}$$

then it is unique and sparsest possible (has minimal support).

- In the case of matrices that are not full rank — and thus unitarily equivalent to matrices with columns of zeros — only the trivial equation $Ax = 0$ has a unique sparsest possible solution.

A Precise Result

Theorem (Recovery of sufficiently sparse solutions)

Let $A \in \mathbb{R}^{m\times n}$ $(m < n)$ be full rank (denote jth column of A by a_j). Initialize the algorithm above with y^0 and weight L^0 such that $y_j^0 = 0$ and $L_j^0 = \|a_j\|$ for $j = 1, 2, \ldots, n$. If $x^* \in \mathbb{R}^n$ with $Ax^* = b$ satisfies (4)*, then, with tolerance $\tau = 0$, we converge in finitely many steps to a point y^* and a weight L^* where,*

$$\operatorname{argmin}\left\{\|Aw - b\|^2 \mid w \in N_{R_{L^*}}(y^*)\right\} = x^*,$$

the unique sparsest solution to $Ax = b$.

- We showed a ‘greedy’ adaptive rescaling of our Algorithm is equivalent to a well-known *greedy algorithm*: **Orthogonal Matching Pursuit** (Bruckstein–Donoh-Elad 09).

Greedy or What?

“Orthogonality of the search directions is important for guaranteeing finite termination, but it is a strong condition to impose, and is really the mathematical manifestation of what it means to be a “greedy algorithm”.

I think “greed” is a misnomer because what really happens is that you forgo any RECOURSE once a decision has been made: the orthogonality condition means that once you’ve made your decision, you don’t have the option of throwing some candidates out of your active constraint set.

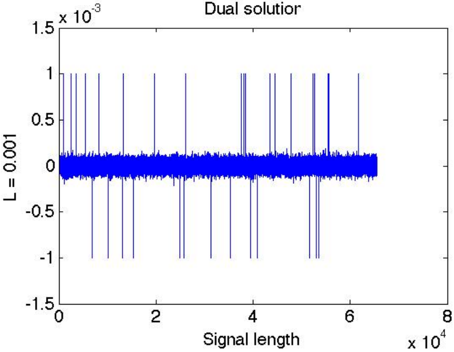
I’d call the strategy “dogmatic”, but as a late arrival to the scene I don’t get naming rights.”
— Russell Luke

Outline

1. Dual Convex (Entropic) Regularization
2. Subgradient Descent with Exact Line-search
3. Our Main Theorem
4. **Computational Results**
5. Conclusion and Questions

Computational Results

The image of the solution to the dual $\bar{y}$ under A^*



$A^*\bar{y}$

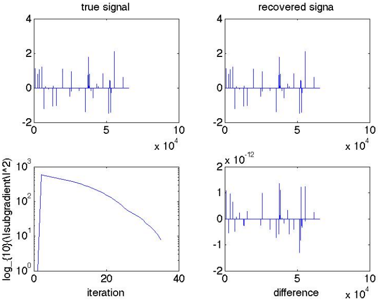
- Used $2 \cdot 2^7$ length real vectors with **70** non-zero entries.
- As often, this is a **qualitative solution** to the primal: yielding location and sign of nonzero signal elements.

Computational Results

The primal solution $\bar{x}$ as determined by the solution to

Program

$$(\mathcal{P}_{\bar{y}}) \qquad \underset{x\in\mathbb{R}^n}{\textit{minimize}} \quad \frac{1}{2}\|b - Ax\|^2 \qquad \textit{subject to} \quad x \in N_{[-L,L]^n}(A^*\bar{y})$$



where $\bar{y}$ solves the dual with $L := 0.001$ and ℓ_∞ error of 10^{-12}.

Computational Results

Observations:

- Inner iterations can be shown to be arbitrarily slow:
    - the solution sets to the subproblems are not metrically regular, and the indicator function $\iota_{N_{[-L,L]^n}}$ is not coercive in the sense of Lions.
- The **algorithm** fails when there are too few samples relative to the sparsity of the true solution.
- All-in-all the method seems highly competitive (and there is still much to tune).

It's generally the way with progress that it looks much greater than it really is.

— Ludwig Wittgenstein (1889–1951)

Outline

Conclusion

We have given a finitely terminating subgradient descent algorithm — one specialization of which yields a variational interpretation and proof of a known *greedy* algorithm.

Work in progress:

1. Characterize recoverability of true solution — in terms of the regularity of the **subproblem**

Program

$$(\mathcal{P}_{v_-}) \qquad \underset{v\in\mathbb{R}^n}{\textit{minimize}} \quad \frac{1}{2}\|b - Av\|^2 \qquad \textit{subject to} \quad v \in N_{[-L,L]^n}(A^*y_-)$$

2. Recovery of $\|.\|_0$ in the limit; not just its convex envelope, 0.
3. Robust code with parameters automatically adjusted (eventually) — appears insensitive to L but weighted norms seem useful; also for $\varepsilon > 0$ and non-dogmatically.

Conclusion

Thank you...

Australian Government

Australian Research Council