A hybrid quasi-Newton projected-gradient method with application to Lasso and basis-pursuit denoise

Ewout van den Berg

Human Language Technologies Group
IBM T.J. Watson Research Center

Work done at the Department of Statistics
Stanford University

October 10, 2014

This work was partially supported by National Science Foundation Grant DMS 0906812
(American Reinvestment and Recovery Act).
Basis pursuit denoise

$$\minimize_x \|x\|_1 \quad \text{subject to} \quad \|Ax - b\|_2 \leq \sigma$$

SPGL1 reduces this by to a series of Lasso problems [B, Friedlander, 2008]

$$\minimize_x \frac{1}{2}\|Ax - b\|_2^2 \quad \text{subject to} \quad \|x\|_1 \leq \tau$$

Root finding with $$\tau^+ = \tau + (\|r\|_2^2 - \sigma\|r\|)/\|A^Tr\|_\infty$$
Basis pursuit denoise

\[
\begin{align*}
\text{minimize} \quad & \|x\|_1 \quad \text{subject to} \quad \|Ax - b\|_2 \leq \sigma \\
\end{align*}
\]

SPGL1 reduces this by to a series of Lasso problems \[\text{[B, Friedlander, 2008]}\]

\[
\begin{align*}
\text{minimize} \quad & \frac{1}{2}\|Ax - b\|_2^2 \quad \text{subject to} \quad \|x\|_1 \leq \tau \\
\end{align*}
\]

Root finding with \(\tau^+ = \tau + \left(\|r\|_2^2 - \sigma\|r\|\right)/\|A^T r\|_\infty\)
Background

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

General form

\[
\text{minimize } f(x) \text{ subject to } x \in \mathcal{C}
\]

Solved using spectral projected-gradient (SPG) method:

\[
d = -\nabla f(x) \cdot \beta \quad \text{or} \quad d = \mathcal{P}(x - \nabla f(x) \cdot \beta) - x
\]

\[
x^+ = \mathcal{P}(x + \alpha d) \quad \text{or} \quad x^+ = x + \alpha d
\]

With

- \( \beta \): Barzilai-Borwein scaling parameter \[\text{[Barzilai,Borwein,1988]}\]
- \( \alpha \): Step length from non-monotone line search \[\text{[Birgin et al., 2000]}\]
- \( \mathcal{P} \): Orthogonal projection onto \( \mathcal{C} \)

\[
\mathcal{P}(x) := \arg\min_v \| x - v \|_2 \text{ subject to } v \in \mathcal{C}
\]
Motivation

Observation

- (Sometimes) difficult to get a highly accurate solution
- Iterates remain on the same face of $C$ (same sign pattern)
- Very little progress
**Motivation**

**Observation**
- (Sometimes) difficult to get a highly accurate solution
- Iterates remain on the same face of $C$ (same sign pattern)
- Very little progress

**Typical solution**
- Detect stagnation on a fixed face
- Solve problem constrained to the given face
- Check optimality for global problem
- Resume if not optimal
Motivation

Observation
- (Sometimes) difficult to get a highly accurate solution
- Iterates remain on the same face of \( C \) (same sign pattern)
- Very little progress

Typical solution
- Detect stagnation on a fixed face
- Solve problem constrained to the given face
- Check optimality for global problem
- Resume if not optimal

Difficulties
- When to initiate this procedure?
- Solving subproblem on incorrect face is wasteful
- Waiting too long defeats the purpose
1 Propose a new hybrid method for polyhedral $C$
   (Practical only for simple $C$: $\ell_1$, bound constrained, simplex)

2 Convergence of the method

3 Application to Lasso and basis pursuit
Hybrid method

Basic idea

- Take regular SPG steps by default
- After each iteration, check whether $\mathcal{F}(x^+) = \mathcal{F}(x) (\neq C)$
- Initialize or update L-BFGS model
- Use quasi-Newton search direction in next iteration
Hybrid method

Basic idea
- Take regular SPG steps by default
- After each iteration, check whether $\mathcal{F}(x^+) = \mathcal{F}(x) \neq C$
- Initialize or update L-BFGS model
- Use quasi-Newton search direction in next iteration

Some issues
- Quasi-Newton direction cannot simply be projected onto $C$
- Naive implementation ignores problem structure
Hybrid method

**Basic idea**
- Take regular SPG steps by default
- After each iteration, check whether $\mathcal{F}(x^+) = \mathcal{F}(x) \neq C$
- Initialize or update L-BFGS model
- Use quasi-Newton search direction in next iteration

**Some issues**
- Quasi-Newton direction cannot simply be projected onto $C$
- Naive implementation ignores problem structure

**Solution**
- Form an L-BFGS model restricted to the current face
- Capture only relevant information
Reduced L-BFGS model

Local function

- We only want to model $f(x)$ over the current $d$-face $F$
- Find an orthonormal basis $B \in \mathbb{R}^{n \times d}$ for $\text{lin}(F - F)$
- Define $\bar{f}(c) : \mathbb{R}^d \to \mathbb{R}$ for some fixed $x_0 \in F$
  \[
  \bar{f}(c) = f(x_0 + Bc)
  \]
- Choosing $c = B^T(x - x_0)$ gives $\bar{f}(c) = f(x)$ for $x \in F$
Reduced L-BFGS model

**Local function**

- We only want to model $f(x)$ over the current $d$-face $F$
- Find an orthonormal basis $B \in \mathbb{R}^{n \times d}$ for $\text{lin}(F - F)$
- Define $\tilde{f}(c) : \mathbb{R}^d \to \mathbb{R}$ for some fixed $x_0 \in F$

$$\tilde{f}(c) = f(x_0 + Bc)$$

- Choosing $c = B^T(x - x_0)$ gives $\tilde{f}(c) = f(x)$ for $x \in F$

**Model updates**

- Standard L-BFGS uses $s = x^+ - x$ and $y = \nabla f(x^+) - \nabla f(x)$
- We use $s = c^+ - c$, and $y = \nabla \tilde{f}(c^+) - \nabla \tilde{f}(c)$:

$$s = B^T(x^+ - x), \quad y = B^T(\nabla f(x^+) - \nabla f(x))$$

- Never need to choose $x_0$
Computing the search direction

▸ Want to compute search direction at current $x$
▸ Denote by $H^{-1}$ the inverse approximate Hessian ($\mathbb{R}^{d \times d}$)
▸ In the reduced space we compute the search direction

\[ \bar{d} = -H^{-1} \nabla \bar{f}(c) = -H^{-1} B^T \nabla f(x) \]

▸ Project back to ambient space using $B\bar{d}$:

\[ d = -BH^{-1} B^T \nabla f(x) \]

Properties

▸ Search direction along the face: $(x + \alpha d) \in \mathcal{F}$ for $0 \leq \alpha \leq \alpha_{\text{max}}$
▸ Guaranteed descent direction
Self-projection cone

Remaining issues

- Quasi-Newton step must be restricted to the face ($\alpha \leq \alpha_{\text{max}}$)
- Fall back to SPG step if line search fails (reset Hessian, history)
- Misses mechanism to avoid local minimum on relint($\mathcal{F}$)

\[ S(\mathcal{F}(x)) := \{ d \in \mathbb{R}^n | \exists \alpha > 0 : \mathcal{F}(P(x + \alpha d)) = \mathcal{F}(x) \} = N(x) + \text{lin}(\mathcal{F}(x) - \mathcal{F}(x)) \]
Self-projection cone

Remaining issues

- Quasi-Newton step must be restricted to the face ($\alpha \leq \alpha_{\text{max}}$)
- Fall back to SPG step if line search fails (reset Hessian, history)
- Misses mechanism to avoid local minimum on relint($\mathcal{F}$)

Self-projection cone

- Update and use L-BFGS model only if $-\nabla f(x^+) \in S(\mathcal{F}(x))$
- Where $S(\mathcal{F}(x))$ is the self-projection cone of $\mathcal{F}(x)$:

$$S(\mathcal{F}(x)) := \{ d \in \mathbb{R}^n \mid \exists \alpha > 0 : \mathcal{F}(\mathcal{P}(x + \alpha d)) = \mathcal{F}(x) \}$$

$$= \mathcal{N}(x) + \text{lin}(\mathcal{F}(x) - \mathcal{F}(x))$$
Theorem

Let \( f(x) \) be a twice continuously differentiable convex function that is bounded below and for which there exist constants \( 0 < \mu_1 \leq \mu_2 < \infty \) such that for all \( x, v \in \mathbb{R}^n \)

\[
\mu_1 \|v\|_2^2 \leq v^T \nabla^2 f(x)v \leq \mu_2 \|v\|_2^2.
\]

Then for any starting point \( x_0 \in \mathcal{C} \), the sequence \( \{x_k\} \) generated by the hybrid algorithm converges to the minimizer of \( f(x) \) over \( \mathcal{C} \).

Proof sketch:

- Finitely many quasi-Newton steps: done or SPG converges
- Infinitely many quasi-Newton steps:
  - Successful quasi-Newton (L-BFGS) step (Liu and Nocedal):
    \[
    f(x^+) - f(x^*) \leq (1 - c)(f(x) - f(x^*))
    \]
  - Finite number of quasi-Newton steps on incorrect faces
Application to general problems

Challenges for general problems

- Projection in SPG is difficult for general $C$
- Facial structure is often unknown
- Finding orthonormal basis for face may be expensive
- Even true for weighted $\ell_1$ ball

Well suited for simple problems

- Cross polytope ($\ell_1$-norm)
- Box constrained problems
- Simplex
Application to Lasso

Additional conditions

- Typically $A \in \mathbb{R}^{m \times n}$ with $m < n$
- Hessian not full rank for $d$-faces with $d > m$
- Use quasi-Newton steps only when $d \leq m$
Application to Lasso

Additional conditions

- Typically $A \in \mathbb{R}^{m \times n}$ with $m < n$
- Hessian not full rank for $d$-faces with $d > m$
- Use quasi-Newton steps only when $d \leq m$

Orthogonal projection

- Reduces to soft-thresholding, $\mathcal{O}(n \log n)$ complexity
Additional conditions

- Typically \( A \in \mathbb{R}^{m \times n} \) with \( m < n \)
- Hessian not full rank for \( d \)-faces with \( d > m \)
- Use quasi-Newton steps only when \( d \leq m \)

Orthogonal projection

- Reduces to soft-thresholding, \( O(n \log n) \) complexity

Orthonormal basis

- Normalize signs and permute indices: \( \mathcal{F} = \text{conv}\{e_1, \ldots, e_{d+1}\} \)
- Compute QR factorization of \([e_2 - e_1, \ldots, e_{d+1} - e_1]:\)

\[
Q_{i,j} = \begin{cases} 
-\sqrt{1/(j^2 + j)} & i \leq j \\
\sqrt{j/(j+1)} & i = j + 1 \\
0 & \text{otherwise.}
\end{cases}
\]

- Implicit \( B \) and \( B^T \), can apply in \( O(n) \) time
Let $d = -\nabla f(x)$ and define
\[
\mathcal{I}_1 = \{ i \in [n] \mid (x_i > 0 \text{ and } d_i < 0) \text{ or } (x_i < 0 \text{ and } d_i > 0) \}, \\
\mathcal{I}_2 = \{ i \in [n] \mid (x_i > 0 \text{ and } d_i \geq 0) \text{ or } (x_i < 0 \text{ and } d_i \leq 0) \}, \\
\mathcal{I}_3 = (\mathcal{I}_1 \cup \mathcal{I}_2)^c,
\]

Set $s_j := \sum_{i \in \mathcal{I}_j} |d_i|$ and assume that $x \not\in \text{relint}(C)$, then
\[
d \in S(\mathcal{F}(x)) \iff \begin{cases} 
  s_1 = s_2 + s_3 \text{ and } s_3 = 0, \text{ or } \\
  s_1 < s_2 + s_3 \text{ and } \max_{i \in \mathcal{I}_3} |d_i| \leq \frac{s_2 - s_1}{|\mathcal{I}_1 \cup \mathcal{I}_2|} 
\end{cases}
\]
Self-projection cone

Let \( d = -\nabla f(x) \) and define

\[
\mathcal{I}_1 = \{ i \in [n] \mid (x_i > 0 \text{ and } d_i < 0) \text{ or } (x_i < 0 \text{ and } d_i > 0) \}, \\
\mathcal{I}_2 = \{ i \in [n] \mid (x_i > 0 \text{ and } d_i \geq 0) \text{ or } (x_i < 0 \text{ and } d_i \leq 0) \}, \\
\mathcal{I}_3 = (\mathcal{I}_1 \cup \mathcal{I}_2)^c,
\]

Set \( s_j := \sum_{i \in \mathcal{I}_j} |d_i| \) and assume that \( x \notin \text{relint}(C) \), then

\[
d \in S(\mathcal{F}(x)) \iff \begin{cases} 
    s_1 = s_2 + s_3 \text{ and } s_3 = 0, \text{ or } \\
    s_1 < s_2 + s_3 \text{ and } \max_{i \in \mathcal{I}_3} |d_i| \leq \frac{s_2 - s_1}{|\mathcal{I}_1 \cup \mathcal{I}_2|}
\end{cases}
\]

Line search

- Can compute maximum step length \( \alpha_{\text{max}} \) to stay on face
- Objective is quadratic, can find minimum along search direction
- Can compute interval \([\alpha_{wmin}, \alpha_{wmax}]\) satisfying Wolfe conditions
Numerical experiments

▶ 10 Sparco problems, each with three $\tau$ values
▶ Random problems: $A, A + c, b = Ax, b$
▶ Heaviside matrix, random $b$
Numerical experiments

Heaviside matrix, random $b$

![Graph showing runtime versus relative duality gap for different versions of MATLAB.](graph.png)
Numerical experiments

Random $300 \times 800$ A, random $b$
Numerical experiments

Random $300 \times 800$ A, random $b$
Numerical experiments

$300 \times 800$ random + offset $A$, $b = Ax_0$, 50-sparse $x_0$
Numerical experiments

Sparco blurspike, $\tau \rightarrow \sigma \approx 0.1 \| b \|_2$
Sparco p3poly, $\tau \rightarrow \sigma \approx 10^{-3} \|b\|_2$
Numerical experiments
Lasso

- Sometimes the procedure is never used, small overhead
- Does well on problems that take longer to solve

Basis pursuit denoise

- SPGL1 has enthusiastic (aggressive) update strategy
- Subproblem terminated before quasi-Newton steps are taken
- Update strategy can lead to run-away behavior
- In those cases accurate solves with hybrid method can help
Conclusions

- Hybrid method shows encouraging results
- Apply to box-constrained problems

Reference