Conjugate gradient iterative hard thresholding for compressed sensing and matrix completion

Jared Tanner

Sparse representations, numerical linear algebra, and optimization
Banff, 6th October 2014

University of Oxford
Joint with Blanchard & Wei

1Supported by: Leverhulme Trust, EPSRC, NVIDIA, & SELEX-Galileo
Explicit search for simple solution from \((y, A)\), NP-hard

- Compressed sensing combinatorial search:
  \[
  \min_{x} \|x\|_0 \quad \text{subject to} \quad \|y - Ax\|_2 \leq \tau
  \]
  where \(\| \cdot \|_0\) counts the number of non-zeros.

- Matrix completion minimum rank search:
  \[
  \min_{X} \text{rank}(X) \quad \text{subject to} \quad \|y - A(X)\|_2 \leq \tau
  \]
Explicit search for simple solution from \((y, A)\), NP-hard

- Compressed sensing combinatorial search:
  \[
  \min_x \|x\|_0 \quad \text{subject to} \quad \|y - Ax\|_2 \leq \tau
  \]
  where \(\|\cdot\|_0\) counts the number of non-zeros.

- Matrix completion minimum rank search:
  \[
  \min_X \text{rank}(X) \quad \text{subject to} \quad \|y - A(X)\|_2 \leq \tau
  \]

- There is a growing number of practical alternatives to the above, nearly all of which are “easily” proven to have an “optimal order.” (More details to come.)

- The most widely studied alternatives are convex relaxations.
Convex relaxations

- Replace compressed sensing combinatorial search

$$\min_x \|x\|_0 \quad \text{subject to} \quad \|y - Ax\|_2 \leq \tau$$

which can be reformulated as linear ($\tau = 0$) or quadratic ($\tau > 0$) programming.
Convex relaxations

- Replace compressed sensing combinatorial search
  \[
  \min_x \|x\|_0 \quad \text{subject to} \quad \|y - Ax\|_2 \leq \tau \quad \text{with}
  \]
  \[
  \min_x \|x\|_1 \quad \text{subject to} \quad \|y - Ax\|_2 \leq \tau
  \]
  which can be reformulated as linear ($\tau = 0$) or quadratic ($\tau > 0$) programming.

- Replace matrix completion minimum rank search
  \[
  \min_X \text{rank}(X) \quad \text{subject to} \quad \|y - A(X)\|_2 \leq \tau
  \]
  with
  \[
  \min_X \|X\|_* := \sum_i \sigma_i(X) \quad \text{subject to} \quad \|y - A(X)\|_2 \leq \tau
  \]
  which can be reformulated as semi-definite programming.
Optimal order recovery - sampling theorems

- CS characterised by three numbers: \( k \leq m \leq n \)
  - \( n \), Signal Length, ambient dimension
  - \( m \), number of inner product measurements
  - \( k \), signal complexity, sparsity

### MC has four defining numbers:
- \( r \leq m \leq n \)
- \( p \times m \times n \), Matrix size, ambient dimension
- \( p \), number of inner product or entry measurements
- \( r \), matrix complexity, rank, with \( r(m+n-r) \) d.o.f.
Optimal order recovery - sampling theorems

- CS characterised by three numbers: $k \leq m \leq n$
  - $n$, Signal Length, ambient dimension
  - $m$, number of inner product measurements
  - $k$, signal complexity, sparsity

- MC has four defining numbers: $r \leq m \leq n$ and $p$
  - $m \times n$, Matrix size, ambient dimension
  - $p$, number of inner product or entry measurements
  - $r$, matrix complexity, rank, with $r(m + n - r)$ d.o.f.
Optimal order recovery - sampling theorems

- CS characterised by three numbers: \( k \leq m \leq n \)
  - \( n \), Signal Length, ambient dimension
  - \( m \), number of inner product measurements
  - \( k \), signal complexity, sparsity

- MC has four defining numbers: \( r \leq m \leq n \) and \( p \)
  - \( m \times n \), Matrix size, ambient dimension
  - \( p \), number of inner product or entry measurements
  - \( r \), matrix complexity, rank, with \( r(m + n - r) \) d.o.f.

- Mixed under/over-sampling rates compared to naive/optimal

\[
\delta := \frac{\text{#measurements}}{\text{ambient dimension}}, \quad \rho := \frac{\text{degrees of freedom}}{\text{#measurements}}
\]

- For \( \delta \) fixed, recovery possible using polynomial complexity algorithms, for \( \rho \) bounded away from zero!
CS: $\ell^1$ decoder [Donoho & T 05, 07]

- With overwhelming probability on $A_{m,n}$ drawn Gaussian:
  for any $\epsilon > 0$, as $(k, m, n) \to \infty$
  - All $k$-sparse signals if $k/m \leq \rho_S(m/n, C)(1 - \epsilon)$
  - Most $k$-sparse signals if $k/m \leq \rho_W(m/n, C)(1 - \epsilon)$
  - Failure typical if $k/m \geq \rho_W(m/n, C)(1 + \epsilon)$

\[
\delta = \frac{m}{n}
\]

- Asymptotic behaviour $\delta \to 0$: $\rho(m/n) \sim [2(e \log(n/m))]^{-1}$
MC: Schatten-1 decoder [Amelunxen, Lotz, McCoy, Tropp]

- With overwhelming probability on $\mathcal{A}(\cdot)$ drawn Gaussian:
  for any $\epsilon > 0$, as $(r, m, n, p) \to \infty$,
  - Most matrices if $r(m + n - r)/p \leq \rho_W(p/mn, N)(1 - \epsilon)$
  - Failure typical if $r(m + n - r)/p \geq \rho_W(p/mn, N)(1 + \epsilon)$

\[ \frac{r(m + n - r)}{p} \]

\[ \delta = p/mn \]

- Many other decoders have been proposed. In particular, Iterative Hard Thresholding (IHT) decoders which are observed to be efficient and simple, but limited theory...
Three prototypical IHT algorithms for CS

Alternating projection approaches to
\[
\min_{x} \|y - Ax\|_2 \quad \text{subject to} \quad \|x\|_0 = k
\]

▶ Normalized Iterated HT (NIHT) [Blumensath & Davies 09]
\[
x_l = H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))
\]
Three prototypical IHT algorithms for CS

Alternating projection approaches to
\[
\min_x \|y - Ax\|_2 \quad \text{subject to} \quad \|x\|_0 = k
\]

- **Normalized Iterated HT (NIHT) [Blumensath & Davies 09]**
  \[
x_l = H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))
\]

- **Hard Thresholding Pursuit (HTP) [Maleki 09, Foucart 10]**
  \[
l_l = \text{supp}(H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))) \quad \text{Descent supp. sets}
  \]
  \[
x_l = (A_{l_l}^T A_{l_l})^{-1} A_{l_l}^T y \quad \text{Pseudo-inverse}
\]
Three prototypical IHT algorithms for CS

Alternating projection approaches to
\[
\min_x \|y - Ax\|_2 \quad \text{subject to} \quad \|x\|_0 = k
\]

- **Normalized Iterated HT (NIHT) [Blumensath & Davies 09]**
  
  \[
  x_l = H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))
  \]

- **Hard Thresholding Pursuit (HTP) [Maleki 09, Foucart 10]**
  
  \[
  l_l = \text{supp}(H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))) \quad \text{Descent supp. sets}
  \]
  
  \[
  x_l = (A_{l_l}^T A_{l_l})^{-1} A_{l_l}^T y \quad \text{Pseudo-inverse}
  \]

- **Two-Stage Thres. [Milenkovic & Dai, Needell & Tropp 08]**
  
  \[
  v_l = H_{\alpha k}(x_{l-1} + \kappa A^T(y - Ax_{l-1}))
  \]
  
  \[
  l_l = \text{supp}(v_l) \cup \text{supp}(x_{l-1}) \quad \text{Join supp. sets}
  \]
  
  \[
  w_l = (A_{l_l}^T A_{l_l})^{-1} A_{l_l}^T y \quad \text{Least squares fit}
  \]
  
  \[
  x_l = H_{\beta k}(w_l) \quad \text{Second threshold}
  \]

- All optimal order, but how effective on typical problems?
Recovery phase transitions:
Gaussian matrix, sign vector, $n = 2^{12}$

Similar recovery regions, especially for $\delta \ll 1$. Which is fastest?
Algorithm Selection map:
Gaussian matrix, sign vector, $n = 2^{12}$, relative residual $10^{-3}$

What goes into the design of a fast CS algorithm?
Three prototypical IHT algorithms for CS

- Normalized Iterated HT (NIHT) [Blumensath & Davies 09]
  \[ x_l = H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1})) \]

- Hard Thresholding Pursuit (HTP) [Foucart 10]
  \[ l_l = \text{supp}(H_k(x_{l-1} + \kappa A^T(y - Ax_{l-1}))) \]
  Descent supp. sets
  \[ x_l = (A_{I_l}^T A_{I_l})^{-1} A_{I_l}^T y \] Pseudo-inverse

- Two-Stage Thres. [Milenkovic & Dai, Needell & Tropp 08]
  \[ v_l = H_{\alpha_k}(x_{l-1} + \kappa A^T(y - Ax_{l-1})) \]
  \[ l_l = \text{supp}(v_l) \cup \text{supp}(x_{l-1}) \] Join supp. sets
  \[ w_l = (A_{I_l}^T A_{I_l})^{-1} A_{I_l}^T y \] Least squares fit
  \[ x_l = H_{\beta_k}(w_l) \] Second threshold

- Low per iteration complexity best at early exploration phase,
  higher order better at later coefficient value recovery phase
- Can we do better, low per iteration with fast asymptotics?
Balancing the iteration cost with fast asymptotic rate

Conjugate Gradient IHT (CGIHT) [Blanchard, T & Wei 2013]

Initialization: Set $T_{-1} = \{\}$, $p_{-1} = 0$, $\nu_0 = A^*y$, $T_0 = \text{DetectSupport}(\nu_0)$, $x_0 = P_{T_0}(\nu_0)$, and $l = 1$.

Iteration: During iteration $l$, do

1: $r_{l-1} = A^*(y - Ax_{l-1})$ (compute the residual)
2: if $T_{l-1} \neq T_{l-2}$
   $eta_{l-1} = 0$
else
   $eta_{l-1} = \frac{\|P_{T_{l-1}}r_{l-1}\|^2_2}{\|P_{T_{l-1}}r_{l-2}\|^2_2}$ (compute orthogonalization weight)
3: $p_{l-1} = r_{l-1} + \beta_{l-1}p_{l-2}$ (define the search direction)
4: $\alpha_{l-1} = \frac{\|P_{T_{l-1}}(r_{l-1})\|^2_2}{\|AP_{T_{l-1}}(p_{l-1})\|^2_2}$ (optimal step size if $T_{l-1} = T_{l-2}$)
5: $\nu_{l-1} = x_{l-1} + \alpha_{l-1}p_{l-1}$ (conjugate gradient step)
6: $T_l = \text{DetectSupport}(\nu_{l-1})$ (proxy to the support set)
7: $x_l = P_{T_l}((\nu_{l-1}))$ (restriction to proxy support set $T_l$)
1-ALPS(2) [Cevher & Kyrillidis 11] and a variant, FIHT [T & Wei]

**Initialization:** Set $T_{-1} = \emptyset$, $x_{-1} = 0$, $\nu_0 = A^*y$, $T_0 = \text{DetectSupport}(\nu_0)$, $x_0 = P_{T_0}(\nu_0)$, and $l = 1$.

**Iteration:** During iteration $l$, do

1. $\tau_{l-1} = \frac{\langle y - Ax_{l-1}, A(x_{l-1} - x_{l-2}) \rangle}{\|A(x_{l-1} - x_{l-2})\|_2^2}$ (momentum step size)
2. $\nu_{l-1} = x_{l-1} + \tau_{l-1}(x_{l-1} - x_{l-2})$ (new extrapolated point)
3. $r_{l-1}^\nu = A^*(y - A\nu_{l-1})$ (residual of extrapolated point)
4. $T_{l-1}^\nu = \text{DetectSupport}(P_{T_{l-1}^\nu}^c(r_{l-1}^\nu)) \cup T_{l-1}$ (1-ALPS(2)) or $T_{l-1}^\nu = \text{DetectSupport}(\nu_{l-1})$ (FIHT: descent support)
5. $\omega_l = \frac{\|P_{T_{l-1}^\nu}(r_{l-1}^\nu)\|_2^2}{\|AP_{T_{l-1}^\nu}(r_{l-1}^\nu)\|_2^2}$ (restricted step to support of $\nu_{l-1}$)
6. $x_l = \nu_{l-1}^\nu + \omega_l r_{l-1}^\nu$ (steepest descent step)
7. $T_l = \text{DetectSupport}(x_l)$, $x_l = P_{T_l}(x_l)$ (restrict to $k$ sparse)
8. $r_l = A^*(y - Ax_l)$ (residual of $x_l$)
9. $\alpha_l = \frac{\|P_{T_l}(r_l)\|_2^2}{\|AP_{T_l}(r_l)\|_2^2}$ (optimal step size restricted to $T_l$)
10. $x_l = x_l + \alpha_l P_{T_l}(r_l)$ (gradient descent in restricted direction)
Recovery phase transitions:
Gaussian matrix, sign vector, \( n = 2^{12} \)

![Graph showing 50% phase transition curve for ensemble gen with \( n = 2^{12} \)]

Similar recovery regions, especially for \( \delta \ll 1 \). Which is fastest?
Algorithm Selection map:
Gaussian matrix, sign vector, $n = 2^{12}$, relative residual $10^{-3}$

Algorithm selection map for $(N,B), n = 2^{12}$

Layering with CGIHT and FIHT (ALPS) typically fastest.
Moderate noise: \( n = 2^{13} \) Gaussian matrix, sign vector, \( y = Ax + e \) for \( e \) drawn \( \mathcal{N}(0, \frac{1}{10} \|Ax\|_2) \)

Algorithm selection map for \( (\mathcal{N}_B, \varepsilon) \) \( \varepsilon = 0.1, n = 2^{13} \)

CGIHT restarted: plus
CGIHT: diamond
NIHT: circle
CSMPS: square

CGIHT variants nearly uniformly fastest especially with additive noise.
Similar behaviour for DCT and sparse matrices, other vector distributions.
CGIHT recovery guarantee

Restricted Isometry Property: sparse near isometry

- Classical $\ell^2$ eigen-analysis [Candes & Tao 05]

\[(1 - L_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + U_k)\|x\|_2^2 \quad \text{for } x \text{ k-sparse}\]
CGIHT recovery guarantee

Restricted Isometry Property: sparse near isometry

- Classical $\ell^2$ eigen-analysis [Candes & Tao 05]

$$(1 - L_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + U_k)\|x\|_2^2$$ for $x$ $k$-sparse

Theorem

Let $A$ be an $m \times n$ matrix with $m < n$, and $y = Ax + e$ for any $x$ with at most $k$ nonzeros. If the RIC constants of $A$ satisfy

$$\frac{(L_{3k} + U_{3k})(5 - 2L_k + 3U_k)}{(1 - L_k)^2} < 1,$$

then there exists a $K > 0$ depending only on $\|x_0 - x\|_2$ such that

$$\|x_l - x\| \leq K \cdot \gamma^l + \frac{2\kappa_\alpha(1 + U_{2k})^{1/2}}{1 - \gamma} \|e\|_2$$

$x_l$ is the $l^{th}$ iteration of CGIHT and $\gamma < 1$ (formula available).

CGIHT extends to matrix completion with roughly same theorem
Between CS and MC: Multi-measurement CS

- Multi-measurement, measure $r$ vectors, each of which are $k$ sparse with shared support set but different nonzero values (e.g., chemical spectroscopy and video with slowly varying images)

$$\min_{Z \in \mathbb{R}^{n \times r}} \| Y - AZ \|_2 \quad \text{subject to} \quad \| Z \|_{R_0} \leq k.$$
Initialization: Set $W_{-1} = \mathcal{A}^*(y)$,
$U_0 = \text{PrincipalLeftSingularVectors}_r(W_{-1})$,
$X_0 = \text{Proj}_{U_0}(W_{-1})$, $R_0 = \mathcal{A}^*(y - \mathcal{A}(X_0))$, $P_0 = R_0$, Restart_flag = 1,
set restart parameter $\theta$, and $l = 1$.
Iteration: During iteration $l$, do
CGIHT projected for matrix completion

1: if \( \frac{\| R_{l-1} - \text{Proj}_{U_{l-1}}(P_{l-1}) \|}{\| \text{Proj}_{U_{l-1}}(R_{l-1}) \|} > \theta \)

\[ \text{Restart\_flag} = 1, \quad \alpha_{l-1} = \frac{\| \text{Proj}_{U_{l-1}}(R_{l-1}) \|^2}{\| A(\text{Proj}_{U_{l-1}}(R_{l-1})) \|^2} \]

\[ W_{l-1} = X_{l-1} + \alpha_{l-1} R_{l-1} \]

else

\[ \text{Restart\_flag} = 0, \quad \alpha_{l-1} = \frac{\| \text{Proj}_{U_{l-1}}(R_{l-1}) \|^2}{\| A(\text{Proj}_{U_{l-1}}(P_{l-1})) \|^2} \]

\[ W_{l-1} = X_{l-1} + \alpha_{l-1} \text{Proj}_{U_{l-1}}(P_{l-1}) \]

2: \( U_l = \text{PrincipalLeftSingularVectors}_r(W_{l-1}) \), \( X_l = \text{Proj}_{U_l}(W_{l-1}) \), \( R_l = A^* (y - A(X_l)) \)

3: if Restart\_flag = 1 set \( P_l = R_l \), else

\[ \beta_l = \frac{\| \text{Proj}_{U_l}(R_l) \|^2}{\| \text{Proj}_{U_l}(R_{l-1}) \|^2} \]

\[ P_l = R_l + \beta_l \text{Proj}_{U_l}(P_{l-1}) \]
NIHT, FIHT, CGIHT: entry sensing \((m = n = 2000)\)

- Phase transition substantial above Schatten-1 norm
- CGIHT convergence rate is fastest in its class.
- What is happening in extreme undersampling \(p \ll mn\)?
CGIHT: entry sensing with $\delta = p/mn = 1/20$

- CGIHT at small $\delta = p/mn = 1/20$, 100 tests per value of $r$
- Recovery in at least 95 times in each of 100 tests for $\rho \leq 0.9$, whereas Schatten-1 recovery requires $\rho < 0.41$.
- Convergence rate appears to be only limit to recovery in matrix completion, even in extreme undersampling $\delta \ll 1$
A few concluding observations

- CS and MC algorithms have two phases: subspace determination and subspace data fitting
- When confidence in the subspace estimate is low, it is best to quickly search the space without minimizing local objectives
- Higher order methods can both accelerate convergence and increase recovery region
- CGIHT balances these competing aspects
- Iterative hard thresholding algorithms have substantially better average case matrix completion recovery than do convex regularizations
References

- Normalized iterative hard thresholding for matrix completion: SIAM J. on Scientific Computing (2012), Tanner and Wei

- Conjugate Gradient iterative hard thresholding for compressed sensing and matrix completion, Blanchard, Tanner and Wei.

- GPU Accelerated Greedy Algorithms for compressed sensing; Mathematical Programming Computation (2013), Blanchard and Tanner.

- Counting faces of randomly-projected polytopes when projection lowers dimension: J. of the AMS (2009), Donoho and Tanner

Thank you for your time