Alternating Direction Augmented Lagrangian Algorithms for Convex Optimization

Donald Goldfarb

Joint work with Bo Huang, Shiqian Ma, Tony Qin, Katya Scheinberg, Zaiwen Wen and Wotao Yin

Department of IEOR, Columbia University

BIRS Workshop on Sparse Statistics, Optimization and Machine Learning
January 16-21 2011

Alternating direction augmented Lagrangian (ADAL) methods

- Alternating direction methods: go back to Peaceman, Rachford, Douglas, Gabay and Mercier, Glowinski and Marrocco, Lions and Mercier, and Passty etc.
- Augmented Lagrangian methods: Hestenes, Powell, Rockafellar

Motivation for Studying ADAL Methods

Motivation:

- Current optimization problems of interest in machine learning, data mining, medical imaging, etc., have enormous numbers of variables/constraints
- Only first-order methods are practical
- It is necessary to take advantage of the structure (e.g., sparsity) of the optimal solution

Part I: Introduction

SUM-K	SUM-2
$\min F(x) \equiv \sum_{i=1}^{K} f_i(x)$	$\min F(x) \equiv f(x) + g(x)$

- Minimize the sum of convex functions
- Assume the following problem is easy

$$\min_x \quad \tau f_i(x) + \frac{1}{2}\|x - y\|^2$$

- Examples of f_i: $\|x\|_1$, $\|x\|_2$, $\|Ax - b\|^2$, $\|X\|_*$, $-\log\det(X)$, $\|x\|_{1,2} \equiv \sum_{g \in G} \|x_g\|_2$ ......

Examples

- Compressed sensing (Lasso):

$$\min \quad \rho\|x\|_1 + \frac{1}{2}\|Ax - b\|_2^2$$

- Matrix Rank Min:

$$\min \quad \rho\|X\|_* + \frac{1}{2}\|\mathcal{A}(X) - b\|_2^2$$

- Robust PCA:

$$\min_{X,Y} \quad \|X\|_* + \rho\|Y\|_1 : X + Y = M$$

- Sparse Inverse Covariance Selection:

$$\min -\log\det(X) + \langle\Sigma, X\rangle + \rho\|X\|_1$$

- Group Lasso:

$$\min \quad \rho\|x\|_{1,2} + \frac{1}{2}\|Ax - b\|_2^2$$

Variable Splitting

$$(SUM-2) \quad \min \quad f(x)+g(x)$$

- Variable splitting

$$\begin{array}{ll} \min & f(x)+g(y) \\ \text{s.t.} & x=y \end{array}$$

- Augmented Lagrangian function:

$$\mathcal{L}(x,y;\lambda) := f(x)+g(y)-\langle \lambda, x-y\rangle + \frac{1}{2\mu}\|x-y\|^2$$

- Augmented Lagrangian Method:

$$\left\{ \begin{array}{rcl} (x^{k+1},y^{k+1}) & := & \arg\min_{(x,y)} \mathcal{L}(x,y;\lambda^k) \\ \lambda^{k+1} & := & \lambda^k - (x^{k+1}-y^{k+1})/\mu \end{array} \right.$$

Alternating Direction Augmented Lagrangian (ADAL)

- $\mathcal{L}(x, y; \lambda) := f(x) + g(y) - \langle \lambda, x - y \rangle + \frac{1}{2\mu}\|x - y\|^2$
- Solve augmented Lagrangian subproblem alternatingly

$$\begin{cases} x^{k+1} & := & \arg\min_x \mathcal{L}(x, y^k; \lambda^k) \\ y^{k+1} & := & \arg\min_y \mathcal{L}(x^{k+1}, y; \lambda^k) \\ \lambda^{k+1} & := & \lambda^k - (x^{k+1} - y^{k+1})/\mu \end{cases}$$

Symmetric ADAL

- $\mathcal{L}(x, y; \lambda) := f(x) + g(y) - \langle \lambda, x - y \rangle + \frac{1}{2\mu} \|x - y\|^2$
- Symmetric version

$$\begin{cases} x^{k+1} & := & \arg\min_x \mathcal{L}(x, y^k; \lambda^k) \\ \lambda^{k+\frac{1}{2}} & := & \lambda^k - (x^{k+1} - y^k)/\mu \\ y^{k+1} & := & \arg\min_y \mathcal{L}(x^{k+1}, y; \lambda^{k+\frac{1}{2}}) \\ \lambda^{k+1} & := & \lambda^{k+\frac{1}{2}} - (x^{k+1} - y^{k+1})/\mu \end{cases}$$

- Optimality conditions lead to (assuming f and g are smooth)

$$\lambda^{k+\frac{1}{2}} = \nabla f(x^{k+1}), \qquad \lambda^{k+1} = -\nabla g(y^{k+1})$$

Alternating Linearization Method

$$(SUM-2) \quad \min F(x) \equiv f(x) + g(x)$$

- Define

$$Q_g(u,v) := f(u) + g(v) + \langle \nabla g(v), u - v \rangle + \frac{1}{2\mu}\|u - v\|^2$$

$$Q_f(u,v) := f(u) + \langle \nabla f(u), v - u \rangle + \frac{1}{2\mu}\|u - v\|^2 + g(v)$$

- Alternating Linearization Method (ALM)

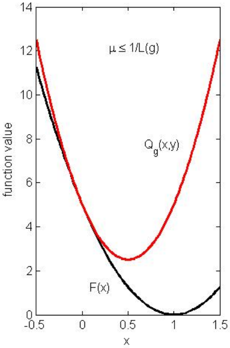
$$\begin{cases} x^{k+1} & := & \arg\min_x Q_g(x, y^k) \\ y^{k+1} & := & \arg\min_y Q_f(x^{k+1}, y) \end{cases}$$

- Gauss-Seidel like algorithm

Key Lemma for Proof

- $F(x) := f(x) + g(y)$: f and g are convex.
- $Q_g(x, y) :=$ $f(x) + g(y) + \langle \nabla g(y), x - y \rangle + \frac{1}{2\mu}\|x - y\|^2$
- $p_g(y) := \arg\min_y Q_g(x, y)$
- Key Lemma:

$$2\mu(F(x) - F(p_g(y))) \geq \|p_g(y) - x\|^2 - \|y - x\|^2$$



Complexity Bound for ALM

Theorem (Goldfarb, Ma and Scheinberg, 2009)

Assume ∇f and ∇g are Lipschitz continuous with constants $L(f)$ and $L(g)$. For $\mu \leq 1/\max\{L(f), L(g)\}$, ALM satisfies

$$F(y^k) - F(x^*) \leq \frac{\|x^0 - x^*\|^2}{4\mu k}$$

Therefore,

- convergence in objective value
- $O(1/\epsilon)$ iterations for an ϵ-optimal solution

- The first complexity result for splitting and alternating direction type methods
- Can we improve the complexity ?
- Can we extend this result to that ADAL method ?

Optimal Gradient Methods

$\min f(x)$ (assuming ∇f is Lipschitz continuous)

- ϵ-optimal solution $f(x) - f(x^*) \leq \epsilon$
- Classical gradient method

$$x^k = x^{k-1} - \tau_k \nabla f(x^{k-1})$$

Complexity $O(1/\epsilon)$

- Nesterov's acceleration technique (1983)

$$\begin{cases} x^k & := & y^{k-1} - \tau_k \nabla f(y^{k-1}) \\ y^k & := & x^k + \frac{k-1}{k+2}(x^k - x^{k-1}) \end{cases}$$

Complexity $O(1/\sqrt{\epsilon})$

- Optimal first-order method; best one can get

ISTA and FISTA (Beck and Teboulle, 2009)

- Assume g is smooth

$$\min \quad F(x) \equiv f(x) + g(x)$$

- Fixed Point Algorithm (Also called ISTA in compressed sensing)

$$x^{k+1} := \arg\min_x Q_g(x, x^k)$$

or equivalently

$$x^{k+1} := \arg\min_x \tau f(x) + \frac{1}{2}\|x - (x^k - \tau\nabla g(x^k))\|^2$$

- Never minimize g
- Iteration complexity: $O(1/\epsilon)$ for an ϵ-optimal solution $(F(x^k) - F(x^*) \leq \epsilon)$

ISTA and FISTA (Beck and Teboulle, 2009)

$$\min F(x) \equiv f(x) + g(x)$$

- Fast ISTA (FISTA)

$$\begin{cases} x^k & := \quad \arg\min_x \tau f(x) + \frac{1}{2}\|x - (y^k - \tau\nabla g(y^k))\|^2 \\ t_{k+1} & := \quad \left(1 + \sqrt{1 + 4t_k^2}\right)/2 \\ y^{k+1} & := \quad x^k + \frac{t_k - 1}{t_{k+1}}(x^k - x^{k-1}) \end{cases}$$

Complexity $O(1/\sqrt{\epsilon})$

Fast Alternating Linearization Method

- ALM

$$\begin{cases} x^{k+1} & := & \arg\min_x Q_g(x, y^k) \\ y^{k+1} & := & \arg\min_y Q_f(x^{k+1}, y) \end{cases}$$

- Accelerate ALM in the same way as FISTA
- Fast Alternating Linearization Method (FALM)

$$\begin{cases} x^k & := & \arg\min_x Q_g(x, z^k) \\ y^k & := & \arg\min_y Q_f(x^k, y) \\ w^k & := & (x^k + y^k)/2 \\ t_{k+1} & := & \left(1 + \sqrt{1 + 4t_k^2}\right)/2 \\ z^{k+1} & := & w^k + \frac{1}{t_{k+1}}(t_k(y^k - w^{k-1}) - (w^k - w^{k-1})) \end{cases}$$

- computational effort at each iteration is almost unchanged
- both f and g must be smooth; however, both are minimized

Complexity Bound for FALM

$$\min F(x) \equiv f(x) + g(x)$$

Theorem (Goldfarb, Ma and Scheinberg, 2009)

Assume ∇f and ∇g are Lipschitz continuous with constants $L(f)$ and $L(g)$. For $\mu \leq 1/\max\{L(f), L(g)\}$, FALM satisfies

$$F(y^k) - F(x^*) \leq \frac{\|x^0 - x^*\|^2}{\mu(k+1)^2}$$

Therefore,

- convergence in objective value
- $O(1/\sqrt{\epsilon})$ iterations for an ϵ-optimal solution
- Optimal first-order method

ALM with skipping steps

At k-th iteration of ALM-S:

- $x^{k+1} := \arg\min_x \mathcal{L}_\mu(x, y^k; \lambda^k)$
- If $F(x^{k+1}) > \mathcal{L}_\mu(x^{k+1}, y^k; \lambda^k)$, then $x^{k+1} := y^k$
- $y^{k+1} := \arg\min_y Q_f(y, x^{k+1})$
- $\lambda^{k+1} := \nabla f(x^{k+1}) - (x^{k+1} - y^{k+1})/\mu$

Note that only one function is required to be smooth.

Complexity Bound for ALM-S

Theorem (Goldfarb, Ma and Scheinberg, 2010)

Assume ∇f is Lipschitz continuous. For $\mu \leq 1/L(f)$, the iterates y^k in ALM-S satisfy:

$$F(y^k) - F(x^*) \leq \frac{\|x^0 - x^*\|^2}{2\mu(k + k_s)}, \forall k,$$

where k_s is the number of iterations until the k-th for which $F(x^{k+1}) \leq \mathcal{L}_\mu(x^{k+1}, y^k; \lambda^k)$. Thus $O(1/\epsilon)$ iterations to obtain an ϵ-optimal solution.

Similar algorithm can be designed for FALM with $O(1/\sqrt{\epsilon})$ complexity and only one function is required to be smooth.

Conjecture: Complexity result for ADAL and fast ADAL (FADAL)

Theorem (Conjectured)

Assume both ∇f and ∇g are Lipschitz continuous. For $\mu \leq 1/\max\{L(f), L(g)\}$, ADAL and FADAL need $O(1/\epsilon)$ and $O(1/\sqrt{\epsilon})$ iterations, respectively, to obtain an ϵ-optimal solution.

No proof currently known.

Basis for possible proof

- Let $A := \partial f$, $B := \partial g$ and the operator

$$S := (I - \mu A)(I + \mu A)^{-1}(I - \mu B)(I + \mu B)^{-1}$$

- The k-th iteration of ALM can be written as

$$v^{k+1} = S \circ v^k.$$

  where $v^k = (I + \mu B)y^k$, for all k.

- We can verify that at the k-th iteration for ADAL, the following relation holds

$$v^{k+1} = \frac{1}{2}(I + S) \circ v^k$$

Fast Generalized Alternating Direction Augmented Lagrangian (FGADAL)

Choose μ and a sequence $\theta_k = \min\{1, \frac{4/\rho}{k+2}\}$ and $0 < \rho \leq 2$.

$$\begin{cases} x^k = arg\min_x \mathcal{L}_\mu(x, y^k; \lambda^k) \\ \lambda^{k+\frac{1}{2}} = \lambda^k - \frac{\rho-1}{\mu}(x^k - y^k) \\ z^k = x^k + \theta_k(\frac{2}{\rho}\theta_{k-1}^{-1} - 1)[x^k - x^{k-1} - (1-\frac{\rho}{2})(y^k - y^{k-1}) \\ \quad +(\mu\lambda^{k+\frac{1}{2}} - \mu\lambda^{k-\frac{1}{2}}) - (1-\frac{\rho}{2})(\mu\lambda^k - \mu\lambda^{k-1})] \\ y^{k+1} = arg\min_y \mathcal{L}_\mu(x^{k+1}, y; \lambda^{k+\frac{1}{2}}) \\ \lambda^{k+1} = \lambda^{k+\frac{1}{2}} - \frac{1}{\mu}(x^{k+1} - y^{k+1}) \end{cases}$$

- If $\rho = 2$, FGADAL reduces to FALM.
- If $\rho = 1$, FGADAL is a fast version of ADAL.
- No proof of complexity currently known.

SUM-K

From P.L.Lions and B.Mercier's 1979 paper on operator splitting

- Generalization from 2 to K is possible, but
- Convergence proof for $K \geq 3$ is difficult

$$\min \quad F(x) \equiv f(x) + g(x) + h(x)$$

- Define

$$Q_{gh}(u, v, w) := f(u) + g(v) + \langle \nabla g(v), u - v \rangle + \|u - v\|^2 / 2\mu$$
$$+ h(w) + \langle \nabla h(w), u - w \rangle + \|u - w\|^2 / 2\mu.$$

$$\begin{cases} x^{k+1} & := & \arg\min Q_{gh}(x, y^k, z^k) \\ y^{k+1} & := & \arg\min Q_{fh}(x^{k+1}, y, z^k) \\ z^{k+1} & := & \arg\min Q_{fg}(x^{k+1}, y^{k+1}, z) \end{cases}$$

- However, no complexity results for Gauss-Seidel like algorithm!

MSA: A Jacobi Type Algorithm

$$\min \quad F(x) \equiv f(x) + g(x) + h(x)$$

- Multiple Splitting Algorithm (MSA)

$$\begin{cases} x^{k+1} & := & \arg\min Q_{gh}(x, w^k, w^k) \\ y^{k+1} & := & \arg\min Q_{fh}(w^k, y, w^k) \\ z^{k+1} & := & \arg\min Q_{fg}(w^k, w^k, z) \\ w^{k+1} & := & (x^{k+1} + y^{k+1} + z^{k+1})/3 \end{cases}$$

- Jacobi type algorithm
- Can be done in parallel
- We have a complexity result!

Complexity Bound for MSA

$$\min F(x) \equiv f(x) + g(x) + h(x)$$

Theorem (Goldfarb and Ma, 2009)

Assume ∇f, ∇g and ∇h are Lipschitz continuous with constants $L(f)$, $L(g)$ and $L(h)$. For $\mu \leq 1/\max\{L(f), L(g), L(h)\}$, MSA satisfies

$$\min\{F(x^k), F(y^k), F(z^k)\} - F(x^*) \leq \frac{\|x_0 - x^*\|^2}{\mu k}.$$

Therefore,

- convergence in objective value
- $O(1/\epsilon)$ iterations for an ϵ-optimal solution

Fast Multiple Splitting Algorithm

Fast Multiple Splitting Algorithm (FaMSA)

$$
\left\{
\begin{array}{lcl}
x^k & := & \arg\min Q_{gh}(x, w_x^k, w_x^k) \\
y^k & := & \arg\min Q_{fh}(w_y^k, y, w_y^k) \\
z^k & := & \arg\min Q_{fg}(w_z^k, w_z^k, z) \\
w^k & := & (x^k + y^k + z^k)/3 \\
t_{k+1} & := & \left(1 + \sqrt{1 + 4t_k^2}\right)/2 \\
w_x^{k+1} & := & w^k + \frac{1}{t_{k+1}}[t_k(x^k - w^k) - (w^k - w^{k-1})] \\
w_y^{k+1} & := & w^k + \frac{1}{t_{k+1}}[t_k(y^k - w^k) - (w^k - w^{k-1})] \\
w_z^{k+1} & := & w^k + \frac{1}{t_{k+1}}[t_k(z^k - w^k) - (w^k - w^{k-1})]
\end{array}
\right.
$$

Complexity Bound for FaMSA

$$\min F(x) \equiv f(x) + g(x) + h(x)$$

Theorem (Goldfarb and Ma, 2009)

Assume ∇f, ∇g and ∇h are Lipschitz continuous with constants $L(f)$, $L(g)$ and $L(h)$. For $\mu \leq 1/\max\{L(f), L(g), L(h)\}$, FaMSA satisfies

$$\min\{F(x^k), F(y^k), F(z^k)\} - F(x^*) \leq \frac{4\|x_0 - x^*\|^2}{\mu(k+1)^2}$$

Therefore,

- convergence in objective value
- $O(1/\sqrt{\epsilon})$ iterations for an ϵ-optimal solution
- optimal first-order method

Comparison of ALM/FALM and MSA/FaMSA

ALM/FALM

- Gauss-Seidel like algorithms
- expected to be faster than MSA/FaMSA since the information from current iteration is used
- complexity results for (SUM-2), no results for (SUM-K) when $K \geq 3$
- only one function needs to be smooth

MSA/FaMSA

- Jacobi like algorithms
- can be done in parallel
- complexity results for (SUM-K) for any $K \geq 2$

Comparison on compressed sensing model with $\rho = 0.01$

solver	obj in k-th iteration				cpu (iter)*
	200	500	800	1000	
FALM-S	9.726599e+4	9.341282e+4	9.182962e+4	9.121742e+4	24.3 (51)
FALM	9.516208e+4	9.186355e+4	9.073086e+4	9.028790e+4	23.1 (51)
FISTA	9.752858e+4	9.372093e+4	9.233719e+4	9.178455e+4	26.0 (69)
ALM-S	1.107103e+5	1.042869e+5	1.021905e+5	1.013128e+5	208.9 (531)
ALM	1.116683e+5	1.047410e+5	1.025611e+5	1.016589e+5	208.1 (581)
ISTA	1.079721e+5	1.040666e+5	1.025107e+5	1.018068e+5	196.8 (510)
SALSA	1.132676e+5	1.054600e+5	1.031346e+5	1.021898e+5	223.9 (663)
SADAL	1.068386e+5	1.021905e+5	1.004005e+5	9.961905e+4	113.5 (332)

* to achieve $F(x) \leq 1.04e + 5$

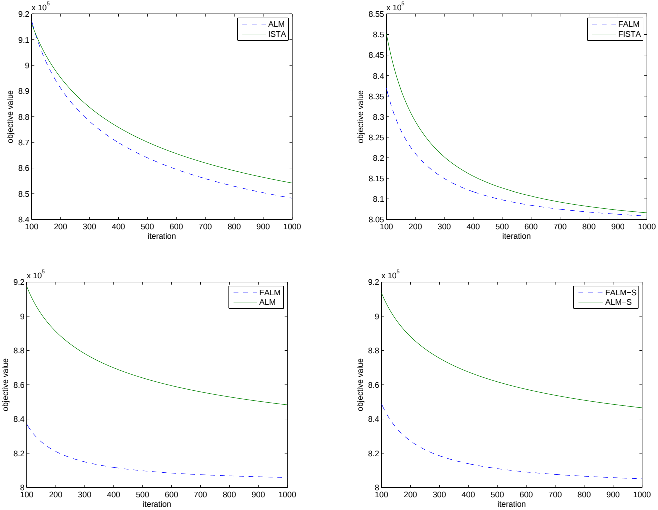


Figure: comparison of the algorithms

Application: Robust PCA

- Robust PCA:

$$\min_{X,Y \in \mathbb{R}^{m \times n}} \quad \{\text{rank}(X) + \rho \|Y\|_0 : X + Y = M\}$$

Recently it has been shown that under suitable conditions on the rank of X and the sparsity of Y, for ρ in a suitable range this generally NP-hard problem can be solved by solving the convex optimization problem

$$\min_{X,Y \in \mathbb{R}^{m \times n}} \quad \{\|X\|_* + \rho \|Y\|_1 : X + Y = M\}$$

ALM and FALM for Robust PCA

- Robust PCA: $f(X) = \|X\|_*, g(Y) = \rho\|Y\|_1$

$$\min_{X,Y \in \mathbb{R}^{m \times n}} \quad \{\|X\|_* + \rho\|Y\|_1 : X + Y = M\}$$

- Subproblem wrt X (a matrix shrinkage operator, corresponds to an SVD):

$$X^{k+1} := \arg\min_X f(X) + g(Y^k) + \langle \gamma_g(Y^k), M - X - Y^k \rangle + \|X + Y^k - M\|_F^2/2\mu$$

- Subproblem wrt Y (a vector shrinkage operator):

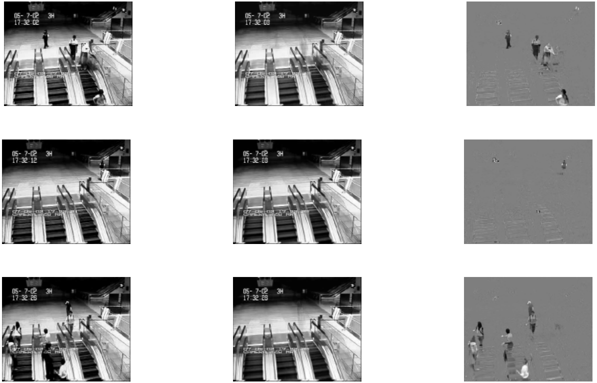
$$Y^{k+1} := \arg\min_Y f(X^{k+1}) + \langle \gamma_f(X^{k+1}), M - X^{k+1} - Y \rangle + \|X^{k+1} + Y - M\|_F^2/2\mu + g(Y)$$

- Smoothed $f(X)$ and $g(Y)$: subgradient $\gamma_f(X^k) = \nabla f(X^k)$ and subgradient $\gamma_g(Y^k) = \nabla g(Y^k)$

Surveillance video

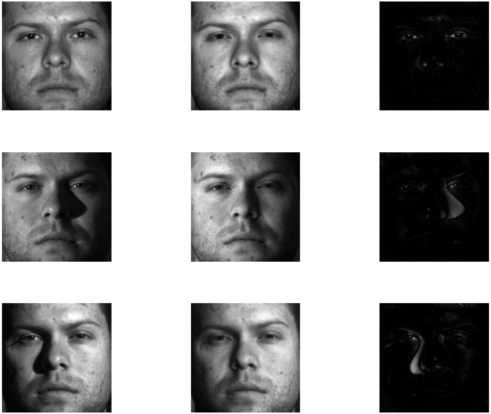
- 200 images with size 144×176, so $M \in \mathbb{R}^{25344 \times 200}$
- 43 SVDs, CPU time: 04:03.
- MATLAB code runs on a Dell Precision 670 workstation with an Intel Xeon(TM) 3.4GHZ CPU and 6GB of RAM.

Surveillance video



- 300 images with size 130×160, so $M \in \mathbb{R}^{20800\times 300}$
- 45 SVDs, CPU time: 05:53.

Shadow and specularities removal from face images



- 65 images with size 200×200, so $M \in \mathbb{R}^{40000 \times 65}$
- 42 SVDs, CPU time: 01:39

Video denoising

- 300 colored images with size 144×176, so $M \in \mathbb{R}^{25344\times 900}$
- 42 SVDs, CPU time: 01:00:18

ALM-S for Sparse Inverse Covariance Selection

- SICS: $f(X) = -\log\det(X) + \langle \Sigma, X \rangle, \quad g(Y) = \rho \|Y\|_1$
- Subproblem wrt X (corresponds to an eigenvalue decomposition):

$$X^{k+1} := \arg\min_X f(X) + g(Y^k) - \langle \Lambda^k, X - Y^k \rangle + \|X - Y^k\|_F^2 / 2\mu$$

- Subproblem wrt Y (a vector shrinkage operator):

$$Y^{k+1} := \arg\min_Y f(X^{k+1}) + \langle \nabla f(X^{k+1}), Y - X^{k+1} \rangle + \|Y - X^{k+1}\|_F^2 / 2\mu + g(Y)$$

Numerical Results: Synthetic Data

- Create data: create sparse matrix $U \in \mathbb{R}^{n \times n}$ with nonzero entries equal to -1 or 1 with equal probability.
- Compute $S := (U * U^\top)^{-1}$ as the true covariance matrix. Hence, S^{-1} is sparse.
- We then draw $p = 5n$ iid vectors, $Y_1, \ldots, Y_p$, from the Gaussian distribution $\mathcal{N}(\mathbf{0}, S)$ by using the *mvnrnd* function in MATLAB.
- $S := \frac{1}{p} \sum_{i=1}^{p} Y_i Y_i^\top$.
- We compare ALM with PSM (Duchi et.al.2008) and VSM (Lu 2009)
- Termination: $Dgap \leq 10^{-3}$

Numerical Results: Synthetic Data

	ALM			PSM			VSM		
n	iter	Dgap	CPU	iter	Dgap	CPU	iter	Dgap	CPU
$\rho = 0.1$									
200	300	8.70e-4	13	1682	9.99e-4	38	857	9.97e-4	37
500	220	5.55e-4	84	861	9.98e-4	205	946	9.98e-4	377
1000	180	9.92e-4	433	292	9.91e-4	446	741	9.97e-4	1928
2000	200	6.13e-5	3110	349	1.12e-3	3759	915	1.00e-3	16085
$\rho = 0.5$									
200	140	9.80e-4	6	6106	1.00e-3	137	1000	9.99e-4	43
500	100	1.69e-4	39	903	9.90e-4	212	1067	9.99e-4	425
1000	100	9.28e-4	247	489	9.80e-4	749	1039	9.95e-4	2709
2000	160	4.70e-4	2529	613	9.96e-4	6519	1640	9.99e-4	28779
$\rho = 1.0$									
200	180	4.63e-4	8	7536	1.00e-3	171	1296	9.96e-4	57
500	140	4.14e-4	55	2099	9.96e-4	495	1015	9.97e-4	406
1000	160	3.19e-4	394	774	9.83e-4	1172	1310	9.97e-4	3426
2000	240	9.58e-4	3794	1158	9.35e-4	12310	2132	9.99e-4	37406

Numerical Results: Real Data

Data on gene expression networks (Li and Toh, 2010): *(1) Lymph node status; (2) Estrogen receptor; (3) Arabidopsis thaliana; (4) Leukemia; (5) Hereditary breast cancer.*

	ALM			PSM			VSM		
n	iter	Dgap	CPU	iter	Dgap	CPU	iter	Dgap	CPU
587	60	9.41e-6	35	178	9.22e-4	64	467	9.78e-4	273
692	80	6.13e-5	73	969	9.94e-4	531	953	9.52e-4	884
834	100	7.26e-5	150	723	1.00e-3	662	1097	7.31e-4	1668
1255	120	6.69e-4	549	1405	9.89e-4	4041	1740	9.36e-4	8568
1869	160	5.59e-4	2158	1639	9.96e-4	14505	3587	9.93e-4	52978

Contributions and Future Work

Our contributions

- New alternating direction augmented Lagrangian, alternating linearization and multiple splitting methods
- Optimal first-order methods
- First complexity results for splitting and alternating direction methods (including Peaceman-Rachford method)

Current and Future Work

- Extension of ALM/FALM, MSA/FaMSA to constrained problems
- Line search variants
- Extension of MSA/FaMSA to nonsmooth problems
- Applications in many fields such as Medical Imaging, Machine Learning, Model Selection, Optimal acquisition basis selection (radar), etc.

Current Work: Constrained Problems

- Stable Robust PCA (SRPCA): Here the the elements of the matrix M are assumed to have noise.

$$\min_{X,Y \in \mathbb{R}^{m \times n}} \{\text{rank}(X) + \rho \|Y\|_0 : \|X + Y - M\|_F \leq \sigma, \}$$

  As in the RPCA problem, under suitable conditions on the rank of X and the sparsity of Y, for ρ in a suitable range solving (SRPCA) can be accomplished by solving

$$\min_{X,Y \in \mathbb{R}^{m \times n}} \{\|X\|_* + \rho \|Y\|_1 : \|X + Y - M\|_F \leq \sigma\}$$

  We have developed ISTA/FISTA and ALM/FALM algorithms for SRPCA that require only a modest increase in the work over that required to solve RPCA
- Overlapping Group Lasso: Here the groups are allowed to overlap,resulting in additional linear constraints. We have developed ISTA/FISTA and ALM/FALM algorithms for this problem.

Current Work: FISTA with line search

Given μ_0 and $0 < \beta < 1$. Cycle to find μ_k and t_k

$x^k = \arg\min_y Q_f(y^k, y)$
Find the smallest $i_k \geq 0$ such that
$\mu_k = \beta^{i_k}\mu_0$ and $F(x^k) \leq Q_f(y^k, x^k)$
$t_{k+1} := \frac{1+\sqrt{1+4\theta_k t_k^2}}{2}, \quad \theta_k := \frac{\mu_k}{\mu_{k+1}}$
$\mu_k t_k^2 \geq \mu_{k+1} t_{k+1}(t_{k+1} - 1)$
$y^{k+1} := x^k + \frac{t_k - 1}{t_{k+1}}(x^k - x^{k-1})$

$\Downarrow$

$$F(x^k) - F(x^*) \leq \frac{\|x^0 - x^*\|^2}{2\mu_k t_k^2}$$

$$\mu_k t_k^2 \geq \frac{\beta k^2}{4L} \Rightarrow F(x^k) - F(x^*) \leq \frac{2L\|x^0 - x^*\|^2}{\beta k^2}$$