Sparse Linear Programming via Primal and Dual Augmented Coordinate Descent

Ian E.H. Yen[1], Kai Zhong[1], Cho-Jui Hsieh[2], Pradeep Ravikumar[1] and Inderjit S. Dhillon[1]
[1]University of Texas at Austin. [2]University of California at Davis

Abstract

- Linear Programming (LP) has been widely used in different areas.
- State-of-the-art algorithms, e.g. interior-point method and primal, dual simplex methods, have complexity at least quadratic in the number of variables or constraints.
- We investigate a general LP algorithm based on the combination of Augmented Lagrangian and Coordinate Descent (AL-CD), giving an iteration complexity of $O((log(1/\epsilon))^2)$ with $O(nnz(A))$ cost per iteration.
- Experiment on large-scale LP shows that AL-CD is orders of magnitude faster than state-of-the-art implementations of IPM and Simplex Methods.

Sparse Linear Program

Given vectors $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $m \times n$ matrix

$$A = \begin{bmatrix} A_I \\ A_E \end{bmatrix} = [A_b\ A_f],$$

the *primal* and *dual* forms of Linear Program (LP) are

$$\min_{x \in \mathbb{R}^n} c^T x \quad s.t.\ \ A_I x \le b_I,\ A_E x = b_E,\ x_j \ge 0,\ j \in [n_b]$$

$$\min_{y \in \mathbb{R}^m} b^T y \quad s.t.\ \ -A_b^T y \le c_b,\ -A_f^T y = c_f,\ y_i \ge 0,\ i \in [m_I].$$

We say a LP is sparse in the sense that
(i) $nnz(A) \ll m \times n$.
(ii) $nnz(y^*) \ll m$ (dual sparsity) or $nnz(x^*) \ll n$ (primal sparsity).

Sparse Linear Program: Examples

- L1-regularized Multiclass SVM ($\{x_i, y_i\}_{i=1,2,\dots,l}$ is the given dataset)

$$\min_{w_m, \xi_i} \lambda \sum_{m=1}^{k} \|w_m\|_1 + \sum_{i=1}^{l} \xi_i \quad s.t.\ \ w_{y_i}^T x_i - w_m^T x_i \ge e_i^m - \xi_i,\ \forall (i,m)$$

- Sparse Inverse Covariance Estimation (S is the covariance matrix)

$$\min_{\Omega \in \mathbb{R}^{d \times d}} \|\Omega\|_1 \quad s.t. \quad \|S\Omega - I_d\|_{max} \le \lambda$$

- Non-negative Matrix Factorization (M is the given matrix and p is any positive vector of size n)

$$\min_{X \in \mathbb{R}_+^{n \times n}} p^T \mathrm{diag}(X)$$
$$\text{s.t. } \|M - MX\|_1 \le \epsilon,\ X(i,i) \le 1,\ X(i,j) \le X(i,i),\ \forall i,j$$

Algorithms for Linear Program

- **Simplex Method [1]**: Moving between corner points requires solutions of two $m \times m$ linear system ($O(m^3)$ for m iterations). #iter $\approx O(m)$ typically.
- **Interior Point Method (IPM) [1]**: Reduce LP to series of (asymptotically ill-conditioned) *unconstrained problems* via Barrier function. Newton step requires solution of $m \times m$ linear system ($O(m^3)$). #iter=$O(\log(1/\epsilon))$.
- **Augmented Lagrangian Method (ALM):** Reduce LP to series of *bound-constrained Quadratic Problem*. #iter=$O(\log(1/\epsilon))$.
  Previous works employ Proj-Newton / Proj-GD for sub-problems. However, Proj-Newton has no guarantee on #iter, while Proj-GD converges very slow.
  This paper $\Rightarrow$ Randomized Coordinate Descent (RCD) with ALM gives $O(nnz(A)\log^2(1/\epsilon))$ overall complexity, and more efficient for primal, dual-sparse problems via an Active-Set strategy.

Augmented Lagrangian Method (ALM)

Let $g(y)$ denote the objective of dual LP (taking ∞ for infeasible points). Then a primal ALM is eqivalent to a dual Proximal-Point iterates

$$y^{t+1} = \underset{y}{argmin}\ g(y) + \frac{1}{2\eta_t}\|y - y^t\|^2, \tag{1}$$

where we find y^{t+1} by solving the dual of (1)

$$\min_{x,\xi} c^T x + \frac{\eta_t}{2}\left\| \begin{matrix} A_I x - b_I + \xi \\ A_E x - b_E \end{matrix} + \frac{1}{\eta_t}\begin{bmatrix} y_I^t \\ y_E^t \end{bmatrix} \right\|^2 \quad s.t.\ \ x_b \ge 0,\ \xi \ge 0 \tag{2}$$

to obtain

$$y^{t+1} = y(x^*, \xi^*) = \eta_t \begin{bmatrix} A_I x^* - b_I + \xi^* \\ A_E x^* - b_E \end{bmatrix} + \begin{bmatrix} y_I^t \\ y_E^t \end{bmatrix}.$$

ALM with Coordinate Descent (AL-CD)

The quadratic program (with ξ eliminated)

$$\min_x F(x) = c^T x + \frac{\eta_t}{2}\left\| \begin{matrix} [A_I x - b_I + y_I^t/\eta_t]_+ \\ A_E x - b_E + y_E^t/\eta_t \end{matrix} \right\|^2 \quad s.t.\ \ x_b \ge 0.$$

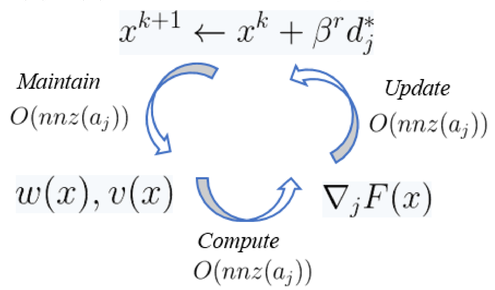
has

$$\nabla F(x) = c + \eta_t A_I^T [w(x)]_+ + \eta_t A_E^T v(x)$$

where

$$w(x) = A_I x - b_I + y_I^t/\eta_t$$
$$v(x) = A_E x - b_E + y_E^t/\eta_t$$

- Given $w(x)$, $v(x)$, gradient $\nabla_j F(x)$ of coordinate can be evaluated in $O(nnz(a_j))$.
- Maintaining $w(x)$, $v(x)$ after each coordinate j update needs $O(nnz(a_j))$.



Randomized Coordinate Descent for Subproblem

INPUT: $\eta_t > 0$ and $(x^{t,0}, w^{t,0}, v^{t,0})$
OUTPUT: $(x^{t,k}, w^{t,k}, v^{t,k})$
repeat
1. Pick a coordinate j uniformly at random
2. Compute $\nabla_j F(x)$, $\nabla_j^2 F(x)$.
3. Obtain Newton direction d_j^*, by solving a single-variable QP:
$$d_j^* = \left[x_j - \nabla_j F(x)/\nabla_j^2 F(x)\right]_+ - x_j,$$
4. Do line search to find step size β^r.
$$F(x^{t,k} + \beta^r d_j^* e_j) - F(x^{t,k}) \le \sigma\beta^r(\nabla_j F(x^{t,k}) d_j^*). \tag{3}$$
5. Update $x^{t,k+1} \leftarrow x^{t,k} + \beta^r d_j^*$.
6. Maintain relation $w(x)$, $v(x)$.
7. $k \leftarrow k + 1$.

until $\|d^*(x)\|_\infty \le \epsilon_t$.

Convergence Analysis

- Subproblem is strongly convex when restricted to $N(\bar{A})^\perp$ (CNSC[2]), where $\bar{A} = [A_I, I; A_E, O]$. $\Rightarrow$ We show RCD converges to $F(x) - F(x^*) \le \epsilon$ in $O(n\log(1/\epsilon))$ number of iterations w.h.p., and the cost for sub-problem is $O(nnz(A)\log(1/\epsilon))$
- The ALM does not amplify error when solving subproblem inexactly. $\Rightarrow$ If each subproblem is approximated to ϵ then after t iterations we have approximation error at most $t\epsilon$.
- Overall #iterations=$O(tn\log(t/\epsilon)) = O(n\log^2(1/\epsilon))$.

Implementation Details

- Maintain an active set $\mathcal{A}^t$ when solving subproblem. Remove coordinate $j \in [n_b]$ from $\mathcal{A}^t$ if $x_j = 0$ and $\nabla_j F(x) > \epsilon_A$.
- Two-Phase strategy:
  Phase-I: Spend constant cost on each sub-problem. ($|\mathcal{A}^t|$ is large)
  Phase-II: Solve each sub-problem to precision ϵ_t, with η_t, ϵ_t dynamically adjusted. ($|\mathcal{A}^t|$ is small)
- For ill-conditioned problem, use Projected-Newton-CG to achieve asymptotic fast convergence (when active set is almost fixed).

Experiments on L1 Multiclass SVM

Data	n_b	m_I
rcv1	4,833,738	778,200
news	2,498,415	302,765
sector	11,597,992	666,848
mnist	75,620	540,000
cod-rna.rf	69,537	59,535
vehicle	79,429	157,646
real-sim	114,227	72,309

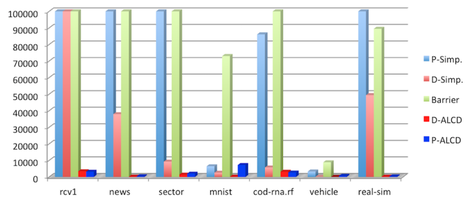


Figure: Timing Results (in sec., 100,000s denotes >100,000s) on L1-SVM

L1-regularized Multiclass SVM: $nnz(A) \ll mn$, $nnz(y^*) \ll m$, and $nnz(x^*) \ll n$.

Experiments on Sparse Inverse Covariance Estimation

Data	n_b	m_I	m_E	n_f	P-Simp	D-Simp	Barrier	D-ALCD	P-ALCD
textmine	60,876	60,876	43,038	43,038	> 48hr	> 48hr	> 48hr	43,096	**18,507**
E2006	55,834	55,834	32,174	32,174	> 48hr	> 48hr	94623	> 48hr	**4,207**
dorothea	47,232	47,232	1,600	1,600	3,980	103	82	47	**38**

Figure: Timing Results (in sec.) on Sparse Inverse Covariance Estimation

- Sparse Inverse Covariance Estimation:
$$\min_{\Omega \in \mathbb{R}^{n \times n}} \|\Omega\|_1,\ s.t. \|S\Omega - I_n\|_{max} \le \lambda$$
- $S = Z^T Z$, Z is $l \times n \ll n^2$ $\Rightarrow$ Transform to:
$$\min_{\Omega \in \mathbb{R}^{n \times n}, Y \in \mathbb{R}^{l \times n}} \|\Omega\|_1,\ s.t.\ \|Z^T Y - I_n\|_{max} \le \lambda,\ Y = Z\Omega$$
- $nnz(A) \ll mn$ and $nnz(\Omega^*) \ll n$.

Experiments on Non-negative Matrix Factorization

Data	n_b	m_I	P-Simp.	D-Simp.	Barrier	D-ALCD	P-ALCD
micromass	2,896,770	4,107,438	> 96hr	> 96hr	280,230	12,966	**12,119**
ocr	6,639,433	13,262,864	> 96hr	> 96hr	284,530	**40,242**	> 96hr

Figure: Timing Results (in sec.) on non-negative matrix factorization

NMF: $nnz(A) \ll mn$.

References

1. Jorge Nocedal and Stephen Wright. Numerical optimization. Springer Science & Business Media, 2006.
2. I. E.H. Yen, C.J. Hsieh, P. Ravikumar, and I. S. Dhillon. "Constant nullspace strong convexity and fast convergence of proximal methods under high-dimensional settings." NIPS 2014.

Mail: {ianyen,pradeepr,inderjit}@cs.utexas.edu, zhongkai@ices.utexas.edu,chohsieh@ucdavis.edu