NysADMM:faster composite convex optimization via low-rank approximation

Shipu Zhao, Zachary Frangella, Madeleine Udell

May 2, 2022

Outline

Background

NysADMM

Convergence analysis

Numerical experiments

Composite optimization

$$\text{minimize} \quad \ell(x) + r(x)$$

- $\ell : \mathbf{R}^n \to \mathbf{R}$ smooth
- $r : \mathbf{R}^n \to \mathbf{R}$ proxable
  - easy (often closed form) solution to $\mathbf{prox}_r(x) = \mathrm{argmin}_y \, r(y) + \frac{1}{2}\|x - y\|^2$
  - e.g., for $r(x) = \|x\|_1$, $\mathbf{prox}_r(x)$ is soft-thresholding operator

Example: Lasso

$$\text{minimize} \quad \frac{1}{2}\|Ax - b\|_2^2 + \gamma\|x\|_1$$

- $\ell(x) = \frac{1}{2}\|Ax - b\|_2^2$ smooth
- $r(x) = \gamma\|x\|_1$ proxable
- parameter $\gamma > 0$ controls strength of regularization

Example: ℓ_1-regularized logistic regression

$$\text{minimize} \quad \ell_{\text{logistic}}(Ax, b) + \gamma \|x\|_1$$

- $\ell(x) = \ell_{\text{logistic}}(Ax, b) = \sum_{i=1}^{n} \log(1 + \exp(-b_i(Ax)_i)$ smooth
- $r(x) = \gamma \|x\|_1$ proxable
- parameter $\gamma > 0$ controls strength of regularization

Example: SVM

$$
\begin{array}{ll}
\text{minimize} & \frac{1}{2}x^T\text{diag}(b)K\text{diag}(b)x - \mathbf{1}^T x \\
\text{subject to} & x^T b = 0 \\
& 0 \leq x \leq C.
\end{array}
$$

- $\ell(x) = \frac{1}{2}x^T\text{diag}(b)K\text{diag}(b)x - \mathbf{1}^T x$
- $r(x)$ is convex indicator of $\{x \mid x^T b = 0,\ 0 \leq x \leq C\}$

Outline

Background

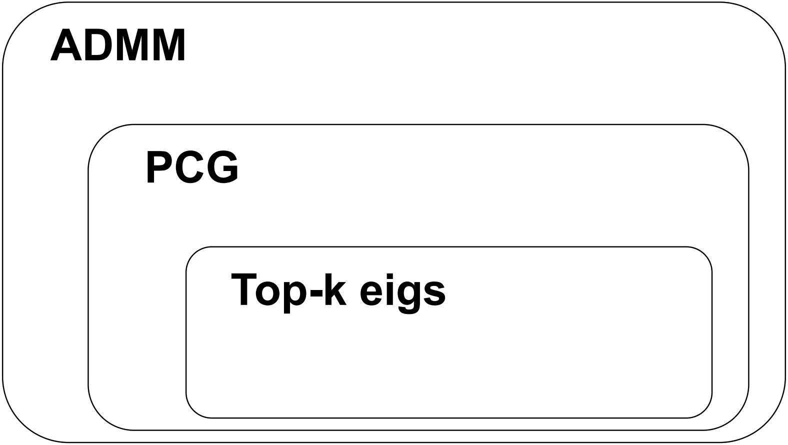
NysADMM

Convergence analysis

Numerical experiments

Our approach

approximate, approximate, approximate!



Alternating Directions Method of Multipliers

Algorithm ADMM

Input: loss function ℓ, regularization r, stepsize ρ,
initial z^0, $u^0 = 0$
for $k = 0, 1, \ldots$ **do**
  $x^{k+1} = \operatorname{argmin}_x \{\ell(x) + r(z) + \frac{\rho}{2}\|x - z^k + u^k\|_2^2\}$
  $z^{k+1} = \operatorname{argmin}_z \{\ell(x) + r(z) + \frac{\rho}{2}\|x^{k+1} - z + u^k\|_2^2\}$
  $u^{k+1} = u^k + x^{k+1} - z^{k+1}$
return $x_\star$ (nearly) minimizing $\ell(x) + r(x)$

problem: x-min involves the (large) data: not easy to solve!

solution: inexact ADMM

- solve x-min approximately with error ε^k
- converges if $\sum_k \varepsilon^k < \infty$ [Eckstein and Bertsekas (1992)]

implementation: use Nyström PCG to speed up x-min

Quadratic approximation

if ℓ is twice differentiable, approximate obj near prev iterate x^k

$$\ell(x) \approx \ell(x^k)+(x-x^k)^T A^T \nabla \ell(x^k)+\frac{1}{2}(x-x^k)^T A^T H_\ell(x^k) A(x-x^k)$$

where H_ℓ is the Hessian of ℓ.

with this approximation, x-min becomes linear system: find x so

$$(A^T H_\ell(x^k) A + \rho I)x = r^k$$

where $r^k = \rho z^k - \rho u^k + A^T H_\ell(x^k) A x^k - A^T \nabla \ell(x^k)$

Nyström PCG to solve ADMM subproblem

$$(A^T H_\ell(x^k) A + \rho I)x = r^k$$

- $A^T H_\ell(x^k) A$ has data in it $\implies$ fast spectral decay
- stepsize ρ regularizes linear system
- if ℓ is quadratic (e.g., lasso and SVM), $H_\ell(x^k) = H_\ell$ is constant, so only need to sketch $A^T H_\ell A$ once

in theory:

- solve to tolerance ε^k at iteration k, where $\sum_k \varepsilon^k < \infty$
- if sketch size $s \approx d_{\mathrm{eff}(\rho)}$, need $\leq O(\log(1/\varepsilon^k))$ CG steps

in practice:

- set $\varepsilon^k = \mathrm{geomean}(\text{primal resid}, \text{dual resid})$
- set sketch size $s = 50$

Outline

Background

NysADMM

Convergence analysis

Numerical experiments

Question

How many approximations can we make, while ADMM still converges?

- linearization
- inexact subproblem solve

General inexact linearized ADMM

$$\text{minimize}_{x \in \mathbb{R}^d} \; \ell(x) + r(Mx), \tag{1}$$

Algorithm General inexact linearized ADMM

Input: loss function ℓ, regularization r, stepsize ρ, psd matrix sequence $\{H^k\}_{k=0}^{\infty}$, positive inexactness sequences $\{\varepsilon_x^k\}_{k=0}^{\infty}$ and $\{\varepsilon_z^k\}_{k=0}^{\infty}$, positive parameter η

repeat

find $\tilde{x}^{k+1}$ that solves $\text{argmin}_x\{\langle x, \nabla \ell(\tilde{x}^k)\rangle + \frac{1}{2}(x - \tilde{x}^k)^T \eta H^k (x - \tilde{x}^k) + \frac{\rho}{2}\|Mx - \tilde{z}^k + \tilde{u}^k\|_2^2\}$ within tolerance ε_x^k

find $\tilde{z}^{k+1}$ that solves $\text{argmin}_z\{r(z) + \frac{\rho}{2}\|M\tilde{x}^{k+1} - z + \tilde{u}^k\|_2^2\}$ within tolerance ε_z^k

$\tilde{u}^{k+1} = \tilde{u}^k + M\tilde{x}^{k+1} - \tilde{z}^{k+1}$

until convergence

Output: solution $x_\star$ of problem (1)

Quadratic loss: convergence

Theorem

Consider the problem in (1) *with quadratic loss,* $\eta = 1$*, and* $\{H^k\}_{k=0}^{\infty}$ *is the Hessian sequence. Define initial iterates* $\tilde{x}^0$*,* $\tilde{z}^0$*, and* $\tilde{u}^0 \in \mathbb{R}^d$*, stepsize* $\rho > 0$*, and summable tolerance sequences* $\{\varepsilon_x^k\}_{k=0}^{\infty}$*,* $\{\varepsilon_z^k\}_{k=0}^{\infty} \subset \mathbb{R}_+$*. Assume for all* $k \geq 0$*, iterates* $\tilde{x}^{k+1}$ *and* $\tilde{z}^{k+1}$ *satisfy*

$$\|\tilde{x}^{k+1} - x^{k+1}\|_2 \leq \varepsilon_x^k \quad \text{and} \quad \|\tilde{z}^{k+1} - z^{k+1}\|_2 \leq \varepsilon_z^k,$$

where x^{k+1} *and* z^{k+1} *are the exact solutions of* x*-subproblem and* z*-subproblem respectively. Then as* $k \to \infty$*,* $\{\tilde{x}^k\}_{k=0}^{\infty}$ *converges to a solution of the primal* (1) *and* $\{\rho\tilde{u}^k\}_{k=0}^{\infty}$ *converges to a solution of the dual problem of* (1) *with rate of* $O(1/t)$ *where* t *is the* t*-th iteration.*

General non-quadratic loss: assumptions

- $\{H^k\}_{k=0}^{\infty}$ is a sequence of psd matrices that satisfies

  $$(1-\zeta^{k-1})H^{k-1} \preceq H^k \preceq (1+\zeta^{k-1})H^{k-1}, \ \forall k \geq 1,$$

  where $\{\zeta^k\}_{k=0}^{\infty}$ is a summable sequence, that is $\sum_{k=0}^{\infty} \zeta^k = A_1 < \infty$. Note this condition also implies $\prod_{k\geq 0}(1+\zeta^k) = A_2 < \infty$. Intuitively, this assumption requires H^k do not change too much between iterations.
- For all $k \geq 0$, iterates $\tilde{x}^{k+1}$ and $\tilde{z}^{k+1}$ satisfy

  $$\|\tilde{x}^{k+1} - x^{k+1}\|_2 \leq \varepsilon_x^k \quad \text{and} \quad \|\tilde{z}^{k+1} - z^{k+1}\|_2 \leq \varepsilon_z^k,$$

  where x^{k+1} and z^{k+1} are the exact solutions of x-subproblem and z-subproblem respectively. Further, the sequences $\{\varepsilon_x^k\}_{k=0}^{\infty}$ and $\{\varepsilon_z^k\}_{k=0}^{\infty}$ are summable, that is $\sum_{k=0}^{\infty} \varepsilon_x^k < A_3$ and $\sum_{k=0}^{\infty} \varepsilon_y^k < A_4$.

General non-quadratic loss: assumptions contd

- Functions ℓ and r are finitely valued, convex, and lower semi-continuous. In addition, ℓ satisfies the following L_ℓ-smoothness condition with respect to the H^k-(semi)norm locally for all k,

$$\ell(x) \leq \ell(x^k) + \langle \nabla \ell(x^k), x - x^k \rangle + \frac{L_\ell}{2} \|x - x^k\|_{H^k}^2.$$

  Function r is Lipschitz-continuous with constant L_r.
- For all k, iterates $\|\tilde{x}^k\|_2$, $\|\tilde{z}^k\|_2$, $\rho\|\tilde{u}^k\|_2$, $\|x^k\|_2$, and $\|z^k\|_2$ are bounded by a constant C_1, $\|H^k\|_2$ is bounded by constant C_2, and $\|\nabla \ell(x^k)\|_2$ is bounded by constant C_3.

General non-quadratic loss: convergence

Theorem

Let $\eta = L_\ell$, and $x^{t+1} = \frac{1}{t}\sum_{k=2}^{t+1} x^k$, where $\{x^k\}_{k\geq 1}$ are the iterates produced by the inexact linearized ADMM with forcing sequences $\{\varepsilon_x^k\}_{k=0}^{\infty}$ and $\{\varepsilon_z^k\}_{k=0}^{\infty}$. Then,

$$\ell(x^{t+1}) + r(Mx^{t+1}) - f_\star \leq$$
$$\frac{\frac{L_\ell}{2}(1 + A_1 A_2)\left(\|H^1\| C_1 + \|x_\star\|_{H^1}\right) + D_u + D_z + CA_3 + (C_1 + L_r)A_4}{t}.$$

Consequently, after $O(\frac{1}{\epsilon})$ iterations

$$\ell(x^{t+1}) + r(Mx^{t+1}) - f_\star \leq \epsilon.$$

Outline

Numerical experiments: settings

- pick datasets with $n > 10,000$ or $d > 10,000$ from LIBSVM, UCI, and OpenML.
- use random feature map to generate more features
- use same stopping criterion and parameter settings as the standard solver for each problem class
- constant sketch size $s = 50$

Numerical experiments: dataset statistics

Name	instances n	features d	nonzero %
STL-10	13000	27648	96.3
CIFAR-10	60000	3073	99.7
CIFAR-10-rf	60000	60000	100.0
smallNorb-rf	24300	30000	100.0
E2006.train	16087	150348	0.8
sector	6412	55197	0.3
p53-rf	16592	20000	100.0
connect-4-rf	16087	30000	100.0
realsim-rf	72309	50000	100.0
rcv1-rf	20242	30000	100.0
cod-rna-rf	59535	60000	100.0

The competition

lasso:

- SSNAL, a Newton augmented Lagrangian method [Li, Sun, and Toh (2018)]
- mfIPM, a matrix-free interior point method [Fountoulakis, Gondzio, and Zhlobich (2014)]
- glmnet, a coordinate-descent method [Friedman, Hastie, and Tibshirani (2010)]

logistic regression:

- SAGA, a stochastic average gradient method [Defazio, Bach, and Lacoste-Julien (2014)]

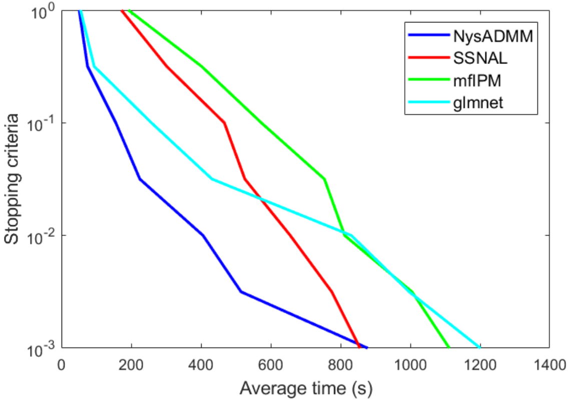
SVM:

- LIBSVM, a sequential minimal optimization (pairwise coordinate descent) method [Chang and Lin (2011)]

Lasso results

stl10 dataset. stop iteration when

$$\frac{\|x - \text{prox}_{\gamma\|\cdot\|_1}(x - A^T(Ax - b))\|}{1 + \|x\| + \|Ax - b\|} \leq \epsilon.$$



Lasso results

Task	Time for $\epsilon = 10^{-1}$ (s)			
	NysADMM	mfIPM	SSNAL	glmnet
STL-10	**165**	573	467	278
CIFAR-10-rf	**251**	655	692	391
smallNorb-rf	**219**	552	515	293
E2006.train	**313**	875	903	554
sector	**235**	678	608	396
realsim-rf	**193**	–	765	292
rcv1-rf	**226**	563	595	273
cod-rna-rf	**208**	976	865	324

ℓ_1-regularized logistic regression results

Table: Results for ℓ_1-regularized logistic regression experiment.

Task	NysADMM time (s)	SAGA (sklearn) time (s)
STL-10	**3012**	6083
CIFAR-10-rf	**7884**	21256
p53-rf	**528**	2116
connect-4-rf	**866**	4781
smallnorb-rf	**1808**	6381
rcv1-rf	**1237**	3988
con-rna-rf	**7528**	21513

Support vector machine results

NysADMM is $\geq 5\times$ faster, although code is pure python!

Table: Results of SVM experiment.

Task	NysADMM time (s)	LIBSVM time (s)
STL-10	**208**	11573
CIFAR-10	**1636**	8563
p53-rf	**291**	919
connect-4-rf	**7073**	42762
realsim-rf	**17045**	52397
rcv1-rf	**564**	32848
cod-rna-rf	**4942**	36791