Proximal LADMPSAP

- Even more general problem:

$$\min_{\mathbf{x}_1,\cdots,\mathbf{x}_n} \sum_{i=1}^{n} f_i(\mathbf{x}_i), \quad s.t. \quad \sum_{i=1}^{n} \mathcal{A}_i(\mathbf{x}_i) = \mathbf{b}.$$

$$f_i(\mathbf{x}_i) = g_i(\mathbf{x}_i) + h_i(\mathbf{x}_i),$$

where both g_i and h_i are convex, g_i is $C^{1,1}$:

$$\|\nabla g_i(\mathbf{x}) - \nabla g_i(\mathbf{y})\| \leq L_i \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^{d_i},$$

and h_i may not be differentiable but its proximal operation is easily solvable.

Proximal LADMPSAP

- Linearize the augmented term to obtain:

$$\mathbf{x}_i^{k+1} = \operatorname*{argmin}_{\mathbf{x}_i} h_i(\mathbf{x}_i) + g_i(\mathbf{x}_i) + \frac{\sigma_i^{(k)}}{2} \left\| \mathbf{x}_i - \mathbf{x}_i^k + \mathcal{A}_i^\dagger(\hat{\lambda}^k)/\sigma_i^{(k)} \right\|^2, \quad i = 1, \cdots, n,$$

- Further linearize g_i:

$$\begin{aligned} \mathbf{x}_i^{k+1} &= \operatorname*{argmin}_{\mathbf{x}_i} h_i(\mathbf{x}_i) + g_i(\mathbf{x}_i^k) + \frac{\sigma_i^{(k)}}{2} \left\| \mathcal{A}_i^\dagger(\hat{\lambda}^k)/\sigma_i^{(k)} \right\|^2 \\ &\quad + \langle \nabla g_i(\mathbf{x}_i^k) + \mathcal{A}_i^\dagger(\hat{\lambda}^k), \mathbf{x}_i - \mathbf{x}_i^k \rangle + \frac{\tau_i^{(k)}}{2} \left\| \mathbf{x}_i - \mathbf{x}_i^k \right\|^2 \\ &= \operatorname*{argmin}_{\mathbf{x}_i} h_i(\mathbf{x}_i) + \tfrac{\tau_i^{(k)}}{2} \left\| \mathbf{x}_i - \mathbf{x}_i^k + \tfrac{1}{\tau_i^{(k)}} [\mathcal{A}_i^\dagger(\hat{\lambda}^k) + \nabla g_i(\mathbf{x}_i^k)] \right\|^2. \end{aligned}$$

- Convergence condition:

$\tau_i^{(k)} = T_i + \beta_k \eta_i$, where $T_i \geq L_i$ and $\eta_i > n\|\mathcal{A}_i\|^2$ are both positive constants.

Experiment

- Group Sparse Logistic Regression with Overlap

$$\min_{\mathbf{w},b} \frac{1}{s}\sum_{i=1}^{s} \log\left(1+\exp\left(-y_i(\mathbf{w}^T\mathbf{x}_i+b)\right)\right) + \mu \sum_{j=1}^{t} \|\mathbf{S}_j\mathbf{w}\|, \tag{1}$$

where $\mathbf{x}_i$ and y_i, $i = 1, \cdots, s$, are the training data and labels, respectively, and $\mathbf{w}$ and b parameterize the linear classifier. $\mathbf{S}_j$, $j = 1, \cdots, t$, are the selection matrices, with only one 1 at each row and the rest entries are all zeros. The groups of entries, $\mathbf{S}_j\mathbf{w}$, $j = 1, \cdots, t$, may overlap each other.

Introducing $\bar{\mathbf{w}} = (\mathbf{w}^T, b)^T$, $\bar{\mathbf{x}}_i = (\mathbf{x}_i^T, 1)^T$, $\mathbf{z} = (\mathbf{z}_1^T, \mathbf{z}_2^T, \cdots, \mathbf{z}_t^T)^T$, and $\bar{\mathbf{S}} = (\mathbf{S}, \mathbf{0})$, where $\mathbf{S} = (\mathbf{S}_1^T, \cdots, \mathbf{S}_t^T)^T$, (1) can be rewritten as

$$\min_{\bar{\mathbf{w}},\mathbf{z}} \frac{1}{s}\sum_{i=1}^{s} \log\left(1+\exp\left(-y_i(\bar{\mathbf{w}}^T\bar{\mathbf{x}}_i)\right)\right) + \mu \sum_{j=1}^{t} \|\mathbf{z}_j\|, \quad s.t. \quad \mathbf{z} = \bar{\mathbf{S}}\bar{\mathbf{w}}, \tag{2}$$

The Lipschitz constant of the gradient of logistic function with respect to $\bar{\mathbf{w}}$ can be proven to be $L_{\bar{w}} \cdot \frac{1}{4s}\|\bar{\mathbf{X}}\|_2^2$, where $\bar{\mathbf{X}} = (\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, \cdots, \bar{\mathbf{x}}_s)$.

Experiment

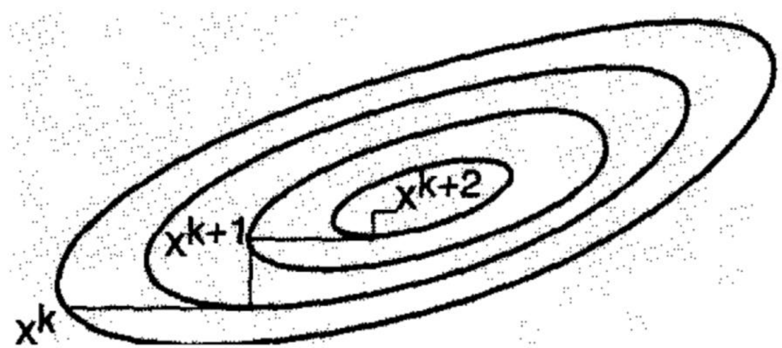
$(s,\, p,\, t,\, q)$	Method	Time	#Iter.	$\frac{\Vert\hat{\bar{\mathbf{w}}}-\bar{\mathbf{w}}^*\Vert}{\Vert\bar{\mathbf{w}}^*\Vert}$	$\frac{\Vert\hat{\mathbf{z}}-\mathbf{z}^*\Vert}{\Vert\mathbf{z}^*\Vert}$
(300, 901, 100, 10)	ADM	294.15	43	0.4800	0.4790
	LADM	229.03	43	0.5331	0.5320
	LADMPS	105.50	47	0.2088	0.2094
	LADMPSAP	57.46	39	0.0371	0.0368
	pLADMPSAP	**1.97**	141	**0.0112**	**0.0112**
(450, 1351, 150, 15)	ADM	450.96	33	0.4337	0.4343
	LADM	437.12	36	0.5126	0.5133
	LADMPS	201.30	39	0.1938	0.1937
	LADMPSAP	136.64	37	0.0321	0.0306
	pLADMPSAP	**4.16**	150	**0.0131**	**0.0131**
(600, 1801, 200, 20)	ADM	1617.09	62	1.4299	1.4365
	LADM	1486.23	63	1.5200	1.5279
	LADMPS	494.52	46	0.4915	0.4936
	LADMPSAP	216.45	32	0.0787	0.0783
	pLADMPSAP	**5.77**	127	**0.0276**	**0.0277**

Coordinate Descent and Block Coordinate Descent

- Coordinate Descent

The cost is minimized along one coordinate direction at each iteration. The order in which coordinates are chosen may vary in the course of the algorithm. In the case where this order is cyclical, given $\mathbf{x}^k$, the i-th coordinate of $\mathbf{x}^{k+1}$ is determined by

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}} f(x_1^{k+1}, x_2^{k+1}, \cdots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \cdots, x_n^k). \tag{1}$$



Coordinate Descent and Block Coordinate Descent

- Coordinate Descent - Parallel computation

Suppose that there is a subset of coordinates $x_{i_1}, x_{i_2}, \cdots, x_{i_m}$, which are not coupled through the cost function, that is, $f(\mathbf{x})$ can be written as $\sum_{i=1}^{m} f_{i_r}(\mathbf{x})$, where for each r, $fi_r(\mathbf{x})$ does not depend on the coordinates x_{i_s} for all $s \neq r$. Then one can perform the m coordinate descent iterations

$$x_{i_r}^{k+1} = \operatorname*{argmin}_{\xi} f_{i_r}(\mathbf{x}^k + \xi \mathbf{e}_{i_r}), \quad r = 1, \cdots, m,$$

independently and in parallel.

Coordinate Descent and Block Coordinate Descent

- Coordinate Descent - Convergence

The coordinate descent method generally has similar convergence properties to steepest descent. For continuously differentiable functions, it can be shown to generate sequences whose limit points are stationary, although the proof of this is sometimes complicated and requires some additional assumptions. The convergence rate of coordinate descent to nonsingular and singular local minima can be shown to be linear and sublinear, respectively, similar to steepest descent. Often, the choice between coordinate descent and steepest descent is dictated by the structure of the objective function. Both methods can be very slow, but for many practical contexts, they can be quite effective.

Coordinate Descent and Block Coordinate Descent

- Block Coordinate Descent

$$\begin{aligned} &\min_{\mathbf{x}} \ f(\mathbf{x}), \\ &s.t. \ \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{2}$$

where $\mathcal{X}$ is a Cartesian product of closed convex sets $\mathcal{X}_1, \cdots, \mathcal{X}_m$:

$$\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_m. \tag{3}$$

We assume that $\mathcal{X}_i$ is a closed convex subset of $\mathbb{R}^{n_i}$ and $n = n_1 + \cdots + n_m$. The vector $\mathbf{x}$ is partitioned as

$$\mathbf{x} = (\mathbf{x}_1^T, \mathbf{x}_2^T, \cdots, \mathbf{x}_m^T)^T,$$

where each $\mathbf{x}_i$ belong to $\mathbb{R}^{n_i}$, so the constraint $\mathbf{x} \in \mathcal{X}$ is equivalent to

$$\mathbf{x}_i \in \mathcal{X}_i, \quad i = 1 \cdots, m.$$

Coordinate Descent and Block Coordinate Descent

- Block Coordinate Descent

Let us assume that for every $\mathbf{x} \in \mathcal{X}$ and every $i = 1, \cdots, m$, the optimization problem:

$$\min_{\mathbf{x}} \ f(\mathbf{x}_1, \cdots, \mathbf{x}_{i-1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}, \cdots, \mathbf{x}_m),$$
$$s.t. \ \boldsymbol{\xi} \in \mathcal{X}_i,$$

has at least one solution. The following algorithm, known as *block coordinate descent* or *nonlinear Gauss-Seidel* method, generates the next iterate $\mathbf{x}^{k+1} = (\mathbf{x}_1^{k+1}, \mathbf{x}_2^{k+1}, \cdots, \mathbf{x}_m^{k+1})^T$, given the current iterate $\mathbf{x}^k = (\mathbf{x}_1^k, \mathbf{x}_2^k, \cdots, \mathbf{x}_m^k)^T$, according to the iteration

$$\mathbf{x}_i^{k+1} = \operatorname*{argmin}_{\boldsymbol{\xi} \in \mathbf{X}_i} \ f(\mathbf{x}_1^{k+1}, \cdots, \mathbf{x}_{i-1}^{k+1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}^k, \cdots, \mathbf{x}_m^k), \quad i = 1, \cdots, m. \tag{4}$$

Coordinate Descent and Block Coordinate Descent

- Block Coordinate Descent - Convergence

Proposition 1 (Convergence of Block Coordinate Descent)**.** *Suppose that f is continuously differentiable over the set $\mathcal{X}$ of equation (3). Furthermore, suppose that for each i and $\mathbf{x} \in \mathcal{X}$, the minimum below*

$$\min_{\boldsymbol{\xi} \in \mathcal{X}_i} f(\mathbf{x}_1, \cdots, \mathbf{x}_{i-1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}, \cdots, \mathbf{x}_m)$$

is uniquely attained. Let $\{\mathbf{x}^k\}$ be the sequence generated by the block coordinate descent method (4). Then every accumulate point of $\{\mathbf{x}^k\}$ is a stationary point.

Coordinate Descent and Block Coordinate Descent

- Block Coordinate Descent - Examples

Dictionary learning:

$$\min_{\mathbf{D},\mathbf{X}} \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2 + \lambda\|\mathbf{X}\|_1, \quad s.t. \quad \|\mathbf{d}_i\|_2 = 1, i = 1, \cdots, K.$$

Low-rank matrix completion:

$$\min_{\mathbf{U},\mathbf{V},\mathbf{A}} \frac{1}{2}\|\mathbf{U}\mathbf{V}^T - \mathbf{A}\|_F^2, \quad s.t. \quad \mathcal{P}_\Omega(\mathbf{A}) = \mathcal{P}_\Omega(\mathbf{D}).$$

Robust Matrix Factorization:

$$\min_{\mathbf{U},\mathbf{V}} \|\mathbf{W}\odot(\mathbf{U}\mathbf{V}^T - \mathbf{M})\|_1 + R_u(\mathbf{U}) + R_v(\mathbf{V}), \quad s.t. \quad \mathbf{U}\in\mathcal{U}, \mathbf{V}\in\mathcal{V}.$$

Truncated Nuclear Norm Minimization:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_r + f(\mathbf{X}), \text{ where } \|\mathbf{X}\|_r = \sum_{i=r+1}^{\min(m,n)} \sigma_i(\mathbf{X}).$$

Chapter 9. Acceleration Techniques

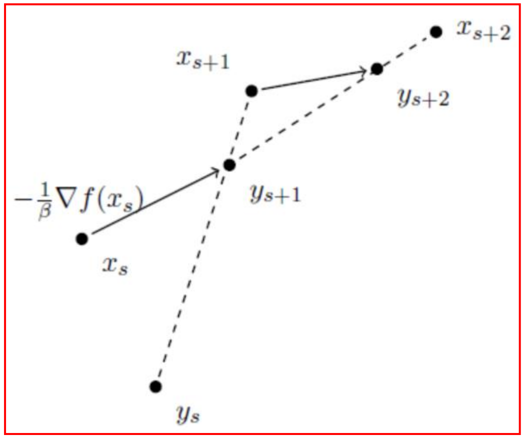
- Nesterov's accelerated gradient descent

The Smooth and Strongly Convex Case

Theorem 1. *Let f be α-strongly convex and L-smooth. Then gradient descent with $\eta = 2(\alpha + L)^{-1}$ satisfies for all $t \geq 0$,*

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*) \leq \frac{L}{2} \exp\left(-4t/(\kappa+1)\right) \|\mathbf{x}_1 - \mathbf{x}^*\|^2,$$

where $\kappa = L/\alpha$ is the condition number.



Nesterov's Accelerated algorithm: Start at an arbitrary initial point $\mathbf{x}_1 = \mathbf{y}_1$ and then iterate the following equations for $t \geq 1$,

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \frac{1}{\beta}\nabla f(\mathbf{x}_t),$$

$$\mathbf{x}_{t+1} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)\mathbf{y}_{t+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\mathbf{y}_t.$$

The Smooth and Strongly Convex Case

Theorem 2. *Let f be α-strongly convex and L-smooth. Then Nesterov's accelerated gradient descent satisfies*

$$f(\mathbf{y}_t) - f(\mathbf{x}^*) \leq \frac{\alpha + \beta}{2} \|\mathbf{x}_1 - \mathbf{x}^*\|^2 \exp\left(-\frac{t-1}{\sqrt{\kappa}}\right).$$

The Smooth and Convex Case

Theorem 3. *Let f be convex and L-smooth. Then gradient descent with $\eta = L^{-1}$ satisfies*

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{2L\|\mathbf{x}_1 - \mathbf{x}^*\|^2}{t-1}.$$

Nesterov's Accelerated algorithm:

$$\lambda_0 = 0, \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2}, \text{ and } \gamma_t = \frac{1-\lambda_t}{\lambda_{t+1}}.$$

(Note that $\gamma_t \leq 0$.) Now the algorithm is simply defined by the following equations, with $\mathbf{x}_1 = \mathbf{y}_1$ an arbitrary initial point,

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \frac{1}{\beta}\nabla f(\mathbf{x}_t),$$

$$\mathbf{x}_{t+1} = (1 - \gamma_s)\mathbf{y}_{t+1} + \gamma_t \mathbf{y}_t.$$

The Smooth and Convex Case

Theorem 4. *Let f be a convex and L-smooth function, then Nesterov's accelerated gradient descent satisfies*

$$f(\mathbf{y}_t) - f(\mathbf{x}^*) \leq \frac{2L\|\mathbf{x}_1 - \mathbf{x}^*\|^2}{t^2}.$$