Stochastic Coordinate Update Methods

Ernest K. Ryu
Seoul National University

Mathematical and Numerical Optimization
Fall 2020

Last edited: 12/01/2020

Outline

Stochastic coordinate fixed-point iteration

Coordinate and extended coordinate friendly operators

Coordinate-partitioning

Partition $x \in \mathbb{R}^n$ into m non-overlapping blocks of sizes $n_1, \dots, n_m$. Write $x = (x_1, \dots, x_m)$, so $x_i \in \mathbb{R}^{n_i}$. Partition $\mathbb{T}\colon \mathbb{R}^n \to \mathbb{R}^n$ into

$$\mathbb{T}(x) = \begin{bmatrix} (\mathbb{T}(x))_1 \\ \vdots \\ (\mathbb{T}(x))_m \end{bmatrix},$$

so $(\mathbb{T}(x))_i \in \mathbb{R}^{n_i}$. Define

$$\mathbb{T}_i(x) = \begin{bmatrix} x_1 \\ \vdots \\ x_{i-1} \\ (\mathbb{T}(x))_i \\ x_{i+1} \\ \vdots \\ x_m \end{bmatrix},$$

i.e., $\mathbb{T}_i$ is $\mathbb{T}$ on the i-th block and is identity on the other blocks. We say "block" and "coordinate" interchangeably.

Coordinate-update fixed-point iteration

For $\mathbb{T}\colon \mathbb{R}^n \to \mathbb{R}^n$, consider

$$\underset{x\in\mathbb{R}^n}{\text{find}} \quad x = \mathbb{T}x.$$

Coordinate-update fixed-point iteration (C-FPI) is

$$\begin{aligned} &\text{select } \ i(k) \in \{1,\dots,m\}, \\ &x^{k+1} = \mathbb{T}_{i(k)}(x^k). \end{aligned}$$

At the k-th iteration, C-FPI updates only the $i(k)$-th block. Specifying the selection rule for $i(k)$ fully specifies the method.

Block selection rules

There are many ways to select $i(k)$ with different advantages and disadvantages.

Common selection rules:

- Cyclic rule. Select the blocks in a cyclic order.
- Essential cyclic rule. Each block appears once or more in each "cycle".
- Greedy rule. Select block that leads to the most progress, measured in many different ways.
- Stochastic rule. Select blocks randomly.

Stochastic coordinate-update fixed-point iteration

We focus on the stochastic rule $i(k) \in \{1, \ldots, m\}$ independently uniformly at random as its analysis is simplest.

We get stochastic coordinate-update fixed-point iteration (SC-FPI):

$$\begin{aligned} i(k) &\sim \text{IID Uniform}\{1, \ldots, m\} \\ x^{k+1} &= \mathbb{T}_{i(k)}(x^k) \end{aligned}$$

Stochastic coordinate-update fixed-point iteration

Theorem 2.

Assume $\mathbb{T}\colon \mathbb{R}^n \to \mathbb{R}^n$ is θ-averaged with $\theta \in (0,1)$ and $\operatorname{Fix}\mathbb{T} \neq \emptyset$. Assume the random indices $i(0), i(1), \ldots \in \{1, \ldots, m\}$ are independent and identically distributed with uniform probability. Then $x^{k+1} = \mathbb{T}_{i(k)} x^k$ with any starting point $x^0 \in \mathbb{R}^n$ converges to one fixed point with probability 1, i.e.,

$$x^k \to x^\star$$

with probability 1 for some $x^\star \in \operatorname{Fix}\mathbb{T}$. The quantities $\mathbb{E}\operatorname{dist}^2(x^k, \operatorname{Fix}\mathbb{T})$ and $\mathbb{E}\|x^k - x^\star\|^2$ for any $x^\star \in \operatorname{Fix}\mathbb{T}$ decrease monotonically with k. Finally, we have

$$\operatorname{dist}(x^k, \operatorname{Fix}\mathbb{T}) \to 0$$

with probability 1.

Proof of Theorem 2

We use the following standard result from probability theory.

Theorem.

(Supermartingale convergence theorem.) Let V^k and S^k be $\mathcal{F}_k$-measurable random variables satisfying $V^k \geq 0$ and $S^k \geq 0$ almost surely for $k = 0, 1, \ldots$. Assume

$$\mathbb{E}\left[V^{k+1} \,|\, \mathcal{F}_k\right] \leq V^k - S^k$$

holds for $k = 0, 1, \ldots$. Then

1. $V^k \to V^\infty$
2. $\sum_{k=0}^{\infty} S^k < \infty$

almost surely. (Note that the limit V^∞ is a random variable.)

Proof of Theorem 2

Define $\mathbb{S}$ with $\mathbb{T} = \mathbb{I} - \theta\mathbb{S}$ and $\mathbb{S}_i$ with $\mathbb{T}_i = \mathbb{I} - \theta\mathbb{S}_i$. So we have

$$\mathbb{S}_i(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ (\mathbb{S}(x))_i \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

for $i = 1, \ldots, m$. Alternately express $x^{k+1} = \mathbb{T}_{i(k)}x^k$ as

$$x^{k+1} = x^k - \theta\mathbb{S}_{i(k)}x^k.$$

Proof of Theorem 2

$\mathbb{T}$ is θ-averaged if and only if $\mathbb{S}$ is $(1/2)$-cocoercive:

$$\begin{aligned}
\mathbb{T} \text{ is } \theta\text{-averaged} \quad &\Leftrightarrow \quad \frac{1}{\theta}\mathbb{T} - \left(\frac{1}{\theta} - 1\right)\mathbb{I} \text{ is nonexpansive} \\
&\Leftrightarrow \quad \mathbb{I} - \mathbb{S} \text{ is nonexpansive} \\
&\Leftrightarrow \quad \|x - \mathbb{S}x - y + \mathbb{S}y\|^2 \le \|x - y\|^2 \quad \forall\, x, y \in \mathbb{R}^n \\
&\Leftrightarrow \quad \frac{1}{2}\|\mathbb{S}x - \mathbb{S}y\|^2 \le \langle x - y, \mathbb{S}x - \mathbb{S}y\rangle \quad \forall\, x, y \in \mathbb{R}^n \\
&\Leftrightarrow \quad \mathbb{S} \text{ is } (1/2)\text{-cocoercive.}
\end{aligned}$$

Clearly, $\operatorname{Fix}\mathbb{T} = \operatorname{Zer}\mathbb{S}$. For any $x^\star \in \operatorname{Fix}\mathbb{T} = \operatorname{Zer}\mathbb{S}$ and $x \in \mathbb{R}^n$,

$$\frac{1}{2}\|\mathbb{S}x\|^2 \le \langle \mathbb{S}x, x - x^\star\rangle \tag{1}$$

Proof of Theorem 2

x^{k+1} is a random variable depending on $i(k), i(k-1), \ldots, i(0)$.
x^0 is not random. Write $\mathbb{E}$ for the full expectation. Write $\mathbb{E}_k$ for the conditional expectation with respect to $i(k)$ conditioned on the past random variables $i(k-1), i(k-2), \ldots, i(0)$.

Under these definitions, $\mathbb{E}_k[x^k] = x^k$ and

$$\mathbb{E}_k[\mathbb{S}_{i(k)} x^k] = \frac{1}{m} \mathbb{S} x^k, \tag{2}$$

$$\mathbb{E}_k \|\mathbb{S}_{i(k)} x^k\|^2 = \frac{1}{m} \|\mathbb{S} x^k\|^2. \tag{3}$$

Proof of Theorem 2

Stage 1. For any $x^\star \in \operatorname{Fix} \mathbb{T}$,

$$\begin{aligned}\|x^{k+1} - x^\star\|^2 &= \|x^k - \theta \mathbb{S}_{i(k)} x^k - x^\star\|^2 \\ &= \|x^k - x^\star\|^2 - 2\theta \langle \mathbb{S}_{i(k)} x^k, x^k - x^\star \rangle + \theta^2 \|\mathbb{S}_{i(k)} x^k\|^2.\end{aligned}$$

Take conditional expectation $\mathbb{E}_k$ and use (2) and (3):

$$\begin{aligned}\mathbb{E}_k \|x^{k+1} - x^\star\|^2 &= \|x^k - x^\star\|^2 - 2\theta \langle \mathbb{E}_k [\mathbb{S}_{i(k)} x^k], x^k - x^\star \rangle + \theta^2 \mathbb{E}_k \|\mathbb{S}_{i(k)} x^k\|^2 \\ &= \|x^k - x^\star\|^2 - \frac{2\theta}{m} \langle \mathbb{S} x^k, x^k - x^\star \rangle + \frac{\theta^2}{m} \|\mathbb{S} x^k\|^2 \\ &\leq \|x^k - x^\star\|^2 - (1-\theta) \frac{\theta}{m} \|\mathbb{S} x^k\|^2, \qquad (4)\end{aligned}$$

where the inequality follows from (1).

So $(\|x^k - x^\star\|^2)_{k=0,1,\ldots}$ a nonnegative supermartingale.

Proof of Theorem 2

Take the full expectation on both ends of (4):

$$\mathbb{E}\|x^{k+1} - x^\star\|^2 \le \mathbb{E}\|x^k - x^\star\|^2 - (1-\theta)\frac{\theta}{m}\mathbb{E}\|\mathbb{S}x^k\|^2.$$

Therefore, $\mathbb{E}\|x^k - x^\star\|^2$ decreases monotonically with k and, by minimizing over $x^\star \in \operatorname{Fix}\mathbb{T}$, so does $\mathbb{E}\operatorname{dist}^2(x^k, \operatorname{Fix}\mathbb{T})$.

Proof of Theorem 2

Stage 2. We prove convergence of the iterates. Apply the supermartingale convergence theorem to (4) to get

(i) $\sum_{k=0}^{\infty} \|\mathbb{S}x^k\|^2 < \infty$ and

(ii) $\lim_{k\to\infty} \|x^k - x^\star\|$ exists

with probability 1. Note (i) implies $\|\mathbb{S}x^k\|^2 \to 0$ and (ii) implies x^k is bounded with probability 1.

For all $x^\star \in \operatorname{Fix}\mathbb{T}$, $\lim_{k\to\infty} \|x^k - x^\star\|$ exists with probability 1. Apply Proposition 1, which we state and prove soon, to conclude with probability 1, $\lim_{k\to\infty} \|x^k - x^\star\|$ exists for all $x^\star \in \operatorname{Fix}\mathbb{T}$. Now $x^k \to x^\star$ with probability 1 follows from the same argument of Theorem 1. □

Measurability argument

Proposition 1 is subtle. We choose $x^\star \in \operatorname{Fix}\mathbb{T}$ and then apply the supermartingale convergence theorem, so [$\lim_{k\to\infty}\|x^k - x^\star\|$ exists with probability 1] applies to one fixed point $x^\star$. This is weaker than what we need when there are uncountably many fixed points.

Proposition 1.

Let $Y \subseteq \mathbb{R}^n$ and let $x^0, x^1, \dots$ be a random sequence. Then statement 1 implies statement 2.

1. *For all $y \in Y$ [with probability 1, $\lim_{k\to\infty}\|x^k - y\|$ exists].*
2. *With probability 1 [for all $y \in Y$, $\lim_{k\to\infty}\|x^k - y\|$ exists].*

Proof outline. (i) $Y \subseteq \mathbb{R}^n$ has a countable dense subset (is separable), (ii) sequence of functions $\{\|x^k - \cdot\|\}_{k\in\mathbb{N}}$ has a limit on the countable dense subset of Y, and (iii) the equicontinuous sequence of functions has a limit on the dense subset of Y, so limit exists on all of Y. $\square$

Outline

Stochastic coordinate fixed-point iteration

Coordinate and extended coordinate friendly operators

Coordinate friendly operators

SC-FPI is computationally useful when $\mathbb{T}$ is coordinate friendly or extended coordinate friendly.

Let $z = (z_1, \ldots, z_m) \in \mathbb{R}^n$. Then $x \mapsto z$ is coordinate friendly if

$$\max_{i=1,\ldots,m} \mathcal{F}[x \mapsto z_i] \ll \mathcal{F}[x \mapsto z].$$

(Meaning of $\ll$ depends on context.)

$\mathbb{T}$ is coordinate friendly if $x \mapsto \mathbb{T}x$ is coordinate friendly.

Coordinate friendly $\Rightarrow$ parallelizable

If $x \mapsto z$ is coordinate friendly,

$$\mathcal{F}_p[x \mapsto z] = \max_{i=1,\ldots,m} \mathcal{F}[x \mapsto z_i] \ll \mathcal{F}[x \mapsto z]$$

for $p \geq m$. So $x \mapsto z$ is parallelizable.

Affine operators

Affine operator $\mathbb{T}x = Ax + b$, where $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$, is coordinate friendly if $n_i \ll n$ for $i = 1, \ldots, m$, since

$$\mathcal{F}[x \mapsto \mathbb{T}_i x] \sim 2nn_i \quad \ll \quad \mathcal{F}[x \mapsto \mathbb{T}x] \sim 2n^2.$$

Separable operators

$\mathbb{T}\colon \mathbb{R}^n \to \mathbb{R}^n$ is a separable operator if

$$\mathbb{T}(x) = (\mathbb{U}_1(x_1), \ldots, \mathbb{U}_m(x_m)),$$

where $\mathbb{U}_i\colon \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ for $i = 1, \ldots, m$. Separable operators are coordinate friendly if $\max_{i=1,\ldots,m} \mathcal{F}[x_i \mapsto \mathbb{U}_i(x_i)] \ll \mathcal{F}[x \mapsto \mathbb{T}(x)]$.

Common example: multiplication by a (block) diagonal matrix.

$f\colon \mathbb{R}^n \to \overline{\mathbb{R}}$ is a separable function if

$$f(x) = \sum_{i=1}^{m} f_i(x_i),$$

where $f_i\colon \mathbb{R}^{n_i} \to \overline{\mathbb{R}}$ for $i = 1, \ldots, m$. If f is separable and differentiable, then ∇f is separable. If f is separable and CCP, then Prox_f is separable.

Separable operators

A separable constraint is of the form

$$x_i \in C_i \quad \text{for } i = 1, \ldots, m.$$

Projection onto a separable constraint is separable.

Common example: box constraint

$$a_i \leq x_i \leq b_i \quad \text{for } i = 1, \ldots, m.$$

Extended coordinate-friendly

$\mathbb{T}$ is extended coordinate-friendly if there is an auxiliary quantity $y(x)$ such that

$$\max_{i=1,\ldots,m} \mathcal{F}\left[\{x, y(x)\} \mapsto \{\mathbb{T}_i x, y(\mathbb{T}_i x)\}\right] \ll \mathcal{F}\left[x \mapsto \mathbb{T}x\right].$$

In other words, computing $\mathbb{T}_i(x)$ is efficient if $y(x)$ is maintained.

More coordinate notation

Use notation $x = (x_1, \dots, x_m)$ with $x_i \in \mathbb{R}^{n_i}$. For $A \in \mathbb{R}^{r \times n}$, write

$$A_{:,i} \in \mathbb{R}^{r \times n_i}$$

for the submatrix, i.e.,

$$A = \begin{bmatrix} A_{:,1} & \cdots & A_{:,m} \end{bmatrix}$$

and

$$Ax = A_{:,1} x_1 + \cdots + A_{:,m} x_m.$$

When f is differentiable, write

$$\nabla f(x) = \begin{bmatrix} \nabla_1 f(x) \\ \vdots \\ \nabla_m f(x) \end{bmatrix}.$$

Example: Gradient descent on least squares

Consider

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|Ax - b\|^2,$$

where $A \in \mathbb{R}^{r \times n}$ and $b \in \mathbb{R}^r$, and

$$\mathbb{T}(x) = x - \alpha A^\intercal (Ax - b).$$

When $r \ll n$, the method is parallelizable, *not* coordinate friendly, but extended coordinate friendly.

Example: Gradient descent on least squares

Evaluation of $\mathbb{T}$ costs

$$\mathcal{F}[x \mapsto \mathbb{T}x] = \mathcal{O}(rn).$$

Parallelizable (assuming $p < \min\{r, n\}$):

$$\begin{aligned}\mathcal{F}_p[x \mapsto \mathbb{T}x] &= \mathcal{F}_p[\{A, x\} \mapsto Ax] + \mathcal{F}_p[\{A^\intercal, Ax\} \mapsto A^\intercal(Ax)] \\ &= \mathcal{O}\left(rn/p\right)\end{aligned}$$

Not coordinate friendly:

$$\begin{aligned}\mathcal{F}[x \mapsto \mathbb{T}_i x] &= \mathcal{F}[x \mapsto Ax] + \mathcal{F}[Ax \mapsto \mathbb{T}_i x] \\ &= \mathcal{O}\left(rn\right) + \mathcal{O}\left(rn_i\right) \\ &= \mathcal{O}\left(rn\right)\end{aligned}$$

Example: Gradient descent on least squares

Extended coordinate friendly with auxiliary quantity Ax:

$$\mathcal{F}\left[\{x, Ax\} \mapsto \{\mathbb{T}_i x, A(\mathbb{T}_i x)\}\right] = \mathcal{O}(rn_i)$$

if we use the formula

$$A(\mathbb{T}_i x) = Ax + A_{:,i}((\mathbb{T}x)_i - x_i).$$

Therefore the C-FPI with $\mathbb{T}$

$$\begin{aligned} x_{i(k)}^{k+1} &= x_{i(k)}^k - \alpha A_{:,i(k)}^\intercal (y^k - b) \\ x_j^{k+1} &= x_j^k \qquad \text{for } j \neq i(k) \\ y^{k+1} &= y^k + A_{:,i(k)}(x_{i(k)}^{k+1} - x_{i(k)}^k) \end{aligned}$$

costs $\mathcal{O}(rn_{i(k)})$ flops per iteration. ($x_j^{k+1} = x_j^k$ costs no operations.)
Initialize $x^0 = 0$ and $y = Ax^0 = 0$.

Example: Gradient descent on least squares

Other approach of using $\mathbb{T}(x) = x - \alpha((A^\intercal A)x - A^\intercal b)$ is not effective.

Precomputing

$$\mathcal{F}[\{A, b\} \mapsto \{A^\intercal A, A^\intercal b\}] = \mathcal{O}\left(rn^2\right)$$

can be prohibitively expensive, and

$$\mathcal{F}[\{x^k, A^\intercal A, A^\intercal b\} \mapsto x_{i(k)}^{k+1}] = \mathcal{O}\left(nn_{i(k)}\right),$$

is larger than $\mathcal{O}\left(rn_{i(k)}\right)$. (Remember, $r \ll n$.)

Example: Coordinate gradient descent

Consider

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x),$$

where f is differentiable. SC-FPI applied to $\mathbb{I} - \alpha \nabla f$

$$x_{i(k)}^{k+1} = x_{i(k)}^{k} - \alpha \nabla_{i(k)} f(x^k),$$

is stochastic coordinate gradient method or stochastic coordinate gradient descent. Converges if a minimizer exists, f is L-smooth, and $\alpha \in (0, 2/L)$.

Example: Coordinate gradient descent

In general, $\mathbb{I} - \alpha\nabla f$ may not be extended coordinate friendly.

However, the following machine learning setup is extended coordinate friendly

$$f(x) = \sum_{j=1}^{r} \ell_j(a_j^\intercal x - b_j),$$

where $a_1, \dots, a_r \in \mathbb{R}^n$, $b_1, \dots, b_r \in \mathbb{R}$, and $\ell_1, \dots, \ell_r$ are differentiable CCP functions on $\mathbb{R}$.

Write

$$A = \begin{bmatrix} — & a_1^\intercal & — \\ & \vdots & \\ — & a_r^\intercal & — \end{bmatrix} \in \mathbb{R}^{r\times n}, \qquad \ell(y) = \sum_{j=1}^{r} \ell_j(y_j).$$

Then

$$\nabla \ell(x) = (\ell_1'(x_1), \dots, \ell_r'(x_r)).$$

Example: Coordinate gradient descent

Stochastic coordinate gradient descent with $y^k = Ax^k$

$$x^{k+1}_{i(k)} = x^k_{i(k)} - \alpha A^{\intercal}_{:,i(k)} \nabla \ell(y^k - b)$$
$$y^{k+1} = y^k + A_{:,i(k)}(x^{k+1}_{i(k)} - x^k_{i(k)})$$

has cost per iteration of $\mathcal{O}\left(rn_{i(k)}\right)$, if $\max_{j=1,\dots,r} \mathcal{F}[x \mapsto \ell_j'(x)] = \mathcal{O}(1)$.

Initialize $x^0 = 0$ and $y = Ax^0 = 0$.

Example: Coordinate GD with block-wise stepsize

Consider

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x),$$

where f is L-smooth. For any diagonal matrix

$$D = \begin{bmatrix} \beta_1 I_{n_1} & & & \\ & \beta_2 I_{n_2} & & \\ & & \ddots & \\ & & & \beta_m I_{n_m} \end{bmatrix}$$

where $\beta_i > 0$ and $I_{n_i} \in \mathbb{R}^{n_i \times n_i}$ is the $n_i \times n_i$ identity matrix, the problem is equivalent to

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(Dx).$$

Stochastic coordinate gradient method on equivalent problem is

$$x_{i(k)}^{k+1} = x_{i(k)}^{k} - \alpha_{i(k)} \nabla_{i(k)} f(x^k),$$

where $\alpha_{i(k)} = \alpha \beta_{i(k)}$. Non-uniform block-wise stepsize is often necessary for a speedup compared to the (full deterministic) gradient method.

Example: Coordinate proximal-gradient descent

Consider

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad f(x)+\sum_{i=1}^m g_i(x_i)$$

where f is differentiable. So we minimize sum of a differentiable function and a separable function. Write

$$g(x)=\sum_{i=1}^m g_i(x_i).$$

SC-FPI with FBS operator $\mathrm{Prox}_{\alpha g}(I-\alpha\nabla f)$

$$x_{i(k)}^{k+1}=\mathrm{Prox}_{\alpha g_{i(k)}}\big(x_{i(k)}^k-\alpha\nabla_{i(k)} f(x^k)\big),$$

is coordinate proximal-gradient (descent) method. Converges if a minimizer exists, f is L-smooth, and $\alpha\in(0,2/L)$.

Example: Coordinate proximal-gradient descent

With block-wise argument, we get

$$x_{i(k)}^{k+1} = \mathrm{Prox}_{\alpha_{i(k)} g_{i(k)}}\big(x_{i(k)}^{k} - \alpha_{i(k)} \nabla_{i(k)} f(x^k)\big).$$

Non-uniform block-wise stepsizes important for speedup.

When g is not separable, $\mathrm{Prox}_{\alpha g}(I - \alpha \nabla f)$ is in general not extended coordinate friendly and SC-FPI not efficient.

Example: Stochastic dual coordinate ascent

Consider

$$\underset{x \in \mathbb{R}^r}{\text{minimize}} \quad g(x) + \sum_{i=1}^{n} \ell_i(a_i^\intercal x - b_i),$$

where g is a strongly convex CCP function on $\mathbb{R}^r$ (so g^* is smooth) and ℓ_i is a CCP function on $\mathbb{R}$. Write

$$A = \begin{bmatrix} \text{—}\ a_1^\intercal\ \text{—} \\ \vdots \\ \text{—}\ a_n^\intercal\ \text{—} \end{bmatrix} \in \mathbb{R}^{n \times r}, \qquad b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \in \mathbb{R}^n.$$

Primal problem generated bu

$$\mathbf{L}(x, u) = g(x) + \langle u, Ax - b \rangle - \sum_{i=1}^{n} \ell_i^*(u_i)$$

and corresponding dual problem is

$$\underset{u \in \mathbb{R}^n}{\text{maximize}} \quad -g^*\left(-A^\intercal u\right) - b^\intercal u - \textstyle\sum_{i=1}^{n} \ell_i^*(u_i).$$

Example: Stochastic dual coordinate ascent

Stochastic coordinate proximal-gradient applied to dual

$$
\begin{aligned}
u_{i(k)}^{k+1} &= \mathrm{Prox}_{\alpha_{i(k)}\ell^*_{i(k)}}\left(u_{i(k)}^k + \alpha_{i(k)}\left(A_{i(k),:}\nabla g^*(y^k) - b_{i(k)}\right)\right)\\
y^{k+1} &= y^k - A_{i(k),:}^{\intercal}(u_{i(k)}^{k+1} - u_{i(k)}^k)
\end{aligned}
$$

is a variation of stochastic dual coordinate ascent. Assume $\mathcal{F}[y \mapsto \nabla g^*(y)] = \mathcal{O}(r)$ and $\max_{i=1,\dots,n}\mathcal{F}[u \mapsto \mathrm{Prox}_{\alpha_i \ell_i^*}(u)] = \mathcal{O}(1)$. Extended coordinate friendly with $y^k = -A^{\intercal}u^k$ maintained. We have

$$
\mathcal{F}[\{y^k, u^k\} \mapsto \{y^{k+1}, u^{k+1}\}] = \mathcal{O}\left(rn_{i(k)}\right).
$$

(One can recover the primal solution with $\nabla g^*(y^k)$.)

Note on splitting data

Iteration of primal coordinate GD accesses $A_{:,i(k)}$, a block of columns.
Iteration of dual coordinate GD accesses $A_{i(k),:}$, a block of rows.

In machine learning, a row of A is a training sample, and we may not want to split it into parts. In so, dual approach is preferred.

Example: MISO/Finito

Consider

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad r(x) + \frac{1}{m} \sum_{i=1}^{m} f_i(x),$$

where $f_1, \ldots, f_m$ are differentiable. Use consensus technique to get

$$\underset{\mathbf{x} \in \mathbb{R}^{nm}}{\text{minimize}} \quad \delta_C(\mathbf{x}) + \sum_{i=1}^{m} \big(r(x_i) + f_i(x_i) \big),$$

where $\mathbf{x} = (x_1, \ldots, x_m)$ and C is the consensus set.

Write $f(\mathbf{x}) = \sum_{i=1}^{m} f_i(x_i)$ and $g(\mathbf{x}) = \delta_C(\mathbf{x}) + \sum_{i=1}^{m} r(x_i)$, so

$$\mathrm{Prox}_{\alpha g}(y_1, \ldots, y_m) = (x, \ldots, x), \quad x = \mathrm{Prox}_{\alpha r}\left(\frac{1}{m} \sum_{i=1}^{m} y_i \right).$$

(See Exercise 2.28.)

Example: MISO/Finito

Both FBS and BFS are extended coordinate friendly with $\overline{z}^k$ maintained. SC-FPI with BFS operator $(\mathbb{I} - \alpha \nabla f)\mathrm{Prox}_{\alpha g}$ is

$$\begin{aligned} x^k &= \mathrm{Prox}_{\alpha r}\left(\overline{z}^k\right) \\ z_{i(k)}^{k+1} &= x^k - \alpha \nabla f_{i(k)}(x^k) \\ \overline{z}^{k+1} &= \overline{z}^k + \frac{1}{m}\left(z_{i(k)}^{k+1} - z_{i(k)}^k\right). \end{aligned}$$

SC-FPI with FBS operator $\mathrm{Prox}_{\alpha g}(\mathbb{I} - \alpha \nabla f)$ is

$$\begin{aligned} x_{i(k)}^{k+1} &= \mathrm{Prox}_{\alpha r}(\overline{z}^k) \\ \overline{z}^{k+1} &= \overline{z}^k + \frac{1}{m}\left(x_{i(k)}^{k+1} - x_{i(k)}^k - \alpha(\nabla f_i(x_{i(k)}^{k+1}) - \nabla f_i(x_{i(k)}^k))\right), \end{aligned}$$

where $\overline{z}^k = \frac{1}{m}\sum_{i=1}^m (x_i^k - \alpha \nabla f_{i(k)}(x_i^k))$. These two equivalent methods are called minimization by incremental surrogate optimization (MISO) or Finito. Converges if a solution exists and $\alpha \in (0, 2/L)$.

Example: MISO/Finito

Among the two, BFS has a minor and subtle advantage.

For BFS, one can use $(z_1^0, \ldots, z_m^0) = (0, \ldots, 0)$ and $\overline{z}^0 = 0$ as the starting point.

For FBS, the starting point $(x_1^0, \ldots, x_m^0) \in \mathbb{R}^{nm}$ can be arbitrary, but

$$\overline{z}^0 = \frac{1}{m} \sum_{i=1}^{m} \left(x_i^0 - \alpha \nabla f_i(x_i^0) \right)$$

needs to be computed before starting the iterations in order to establish convergence via Theorem 2.

Example: Conic programs with many small cones

Consider

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & c^\intercal x \\ \text{subject to} & Ax = b \\ & x \in Q_1 \times \cdots \times Q_m, \end{array}$$

where $Q_i \subseteq \mathbb{R}^{n_i}$ is a nonempty closed convex set, $A \in \mathbb{R}^{r\times n}$ has rank r, and $b \in \mathbb{R}^r$. Assume $\mathcal{F}[x_i \mapsto \Pi_{Q_i} x_i] = C_i$.

Note: $[x \in Q_1 \times \cdots \times Q_m] \Leftrightarrow [x_i \in Q_i$ for $i = 1, \ldots, m]$

When $Q_1, \ldots, Q_m$ are convex cones, problem called a conic program.

Example: Conic programs with many small cones

Naive SC-FPI with DRS applied to

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \underbrace{c^\intercal x + \delta_{\{x \mid Ax=b\}}(x)}_{=f(x)} + \underbrace{\delta_{Q_1\times\cdots\times Q_m}(x)}_{=g(x)}$$

becomes

$$\begin{aligned} x_i^{k+1/2} &= \Pi_{Q_i}(z_i^k) \qquad \text{for } i=1,\dots,m \\ z_{i(k)}^{k+1} &= z_{i(k)}^k + D_{i(k),:}(2x^{k+1/2} - z^k) + v_{i(k)} - x_{i(k)}^{k+1/2}, \end{aligned}$$

where $D = I - A^\intercal(AA^\intercal)^{-1}A$ and $v = A^\intercal(AA^\intercal)^{-1}b - \alpha Dc$. (Exercise 2.24.) Costs $\mathcal{O}\left(C_1+\cdots+C_n+nn_{i(k)}\right)$ per iteration.

Utilize the extended coordinate friendly structure with $y^k = D2x^{k+1/2} - z^k$:

$$\begin{aligned} x_{i(k)}^{k+1/2} &= \Pi_{Q_{i(k)}}(z_{i(k)}^k) \\ z_{i(k)}^{k+1} &= z_{i(k)}^k + y_{i(k)}^k + v_{i(k)} - x_{i(k)}^{k+1/2} \\ y^{k+1} &= D_{:,i(k)}\left(2\Pi_{Q_{i(k)}}(z_{i(k)}^{k+1}) - 2x_{i(k)}^{k+1/2} - z_{i(k)}^{k+1} + z_{i(k)}^k\right), \end{aligned}$$

which costs $\mathcal{O}\left(C_{i(k)} + nn_{i(k)}\right)$ per iteration.