Advanced Topics in Data Sciences

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 06: Variance reduction and coordinate descent methods

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-731 (Spring 2016)

lions@epfl

MARIE CURIE ACTIONS

FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

erc

EPFL

Outline

This lecture:

1. Variance reduction.
2. Coordinate descent methods for smooth objectives.

Recommended reading materials

1. Y. Nesterov, Efficiency of coordinate descent methods on huge-scale optimization, *SIAM. J. Optim.*, vol. 22, pp. 341–362, 2012.
2. S. J. Wright, Coordinates descent algorithms, *Math. Program.*, vol. 151, pp. 3–34, 2015.

Recall: Stochastic proximal gradient method

Problem (Composite convex minimization)

Consider the following composite convex minimization problem:

$$\boxed{F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x}) := \mathbb{E}[h(\mathbf{x},\theta)] + g(\mathbf{x})\}} \tag{1}$$

- $f := \mathbb{E}[h(\mathbf{x},\theta)]$ *and* g *are both proper, closed, and convex.*
- ∇f *is* L*-Lipschitz continuous.*
- g *is possibly non-smooth.*
- θ *is a random vector whose distribution is supported on* Θ
- *The solution set* $\mathcal{S}^\star := \{\mathbf{x}^\star \in dom(F) : F(\mathbf{x}^\star) = F^\star\}$ *is nonempty.*

- Proximal gradient:

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\gamma_k g}\Big(\mathbf{x}^k - \gamma_k \nabla f(\mathbf{x}^k)\Big).$$

- Stochastic proximal gradient (SPGM):

$$\begin{cases} \text{Compute } G(\mathbf{x}^k,\theta_k) \text{ such that } \mathbb{E}[G(\mathbf{x}^k,\theta_k)] = \nabla f(\mathbf{x}^k), \\ \mathbf{x}^{k+1} = \mathrm{prox}_{\gamma_k g}\Big(\mathbf{x}^k - \gamma_k G(\mathbf{x}^k,\theta_k)\Big). \end{cases}$$

Recall: Stochastic proximal gradient method

Problem (Composite convex minimization: A simple example)

We consider the following simple example in the next few slides:

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := \underbrace{\frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x})}_{f(\mathbf{x})} + g(\mathbf{x}) \right\}$$

- f_i *and* g *are proper, closed, and convex.*
- ∇f_i *is* L_i*-Lipschitz continuous for* $i = 1, \dots, m$.
- g *is possibly non-smooth.*
- *The solution set* $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}(F) : F(\mathbf{x}^\star) = F^\star\}$ *is nonempty.*

- One prevalent choice is given by

$$G(\mathbf{x}^k, i_k) = \nabla f_{i_k}(\mathbf{x}^k).$$

- Computation of $\nabla f_{i_k}(\mathbf{x})$ is m times cheaper than $\nabla f(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} \nabla f_i(\mathbf{x})$.

Recall: Stochastic proximal gradient method

Problem (Composite convex minimization: A simple example)

We consider the following simple example in the next few slides:

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := \underbrace{\frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x})}_{f(\mathbf{x})} + g(\mathbf{x}) \right\}$$

- f_i *and* g *are proper, closed, and convex.*
- ∇f_i *is* L_i*-Lipschitz continuous for* $i = 1, \ldots, m$.
- g *is possibly non-smooth.*
- *The solution set* $\mathcal{S}^\star := \{\mathbf{x}^\star \in dom(F) : F(\mathbf{x}^\star) = F^\star\}$ *is nonempty.*

Variance reduction

To ensure the convergence of SPGM, we need the following assumption to hold:

$$\sum_{k\geq 0} \gamma_k^2 \, \mathbb{E}[\| G(\mathbf{x}^k, i_k) - \nabla f(\mathbf{x}^k)\|^2 | \{i_0, \ldots, i_{k-1}\}] < +\infty.$$

We decrease the learning rate γ_k to satisfy above condition $\implies$ Slow convergence!
Idea: We decrease the variance $\mathbb{E}[\| G(\mathbf{x}^k, i_k) - \nabla f(\mathbf{x}^k)\|^2 | \{i_0, \ldots, i_{k-1}\}]$ instead.

Variance reduction techniques: Simple variance reduction

Proximal stochastic variance reduction (SPGD-VR)

1. Choose $\overline{\mathbf{x}}^0 \in \mathbb{R}^p$, $0 \neq q \in \mathbb{N}$ and stepsize $\gamma > 0$.

2. For $k = 0, 1 \ldots$ perform:

2a. $\nabla f(\overline{\mathbf{x}}^k) = \frac{1}{m} \sum_{i=1}^m \nabla f_i(\overline{\mathbf{x}})$, $\mathbf{x}^0 = \overline{\mathbf{x}}^k$.

2b. For $l = 0, 1 \ldots, q-1$, perform:

$$\begin{cases} \text{pick } i_l \in \{1, \ldots, m\} \text{ uniformly at random,} \\ \overline{G(\mathbf{x}^l, i_l)} = \nabla f_{i_l}(\mathbf{x}^l) - \nabla f_{i_l}(\overline{\mathbf{x}}^k) + \nabla f(\overline{\mathbf{x}}^k), \\ \mathbf{x}^{l+1} = \mathrm{prox}_{\gamma g}(\mathbf{x}^l - \gamma \overline{G(\mathbf{x}^l, i_l)}). \end{cases}$$

3 Update $\overline{\mathbf{x}}^{k+1} = \frac{1}{q} \sum_{l=1}^q \mathbf{x}^l$.

Recipe:

In a cycle of q iterations:

- Set $\bar{\mathbf{x}}$ to be the previous iteration and compute the full-gradient at $\bar{\mathbf{x}}$.
- Perform q SPG-iterations with the following stochastic gradient

$$\overline{G(\mathbf{x}^l, i_l)} = \nabla f_{i_l}(\mathbf{x}^l) - \nabla f_{i_l}(\overline{\mathbf{x}}^k) + \nabla f(\overline{\mathbf{x}}^k).$$

- Update next iteration as average of q previous iterations.

Convergence of SPGD-VR

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := \frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x}) + g(\mathbf{x}) \right\}.$$

Theorem (Mean convergence of SPGD-VR [6])

Set $L_{max} = \max_{1\le i\le m} L_i$, *where* L_i *is Lipschitz constant of* ∇f_i. *Suppose that* F *is* μ*-strongly convex and that the stepsize satisfies*

$$\rho = \frac{1}{\mu\gamma(1-2L_{max}\gamma)q} + \frac{2L_{max}\gamma}{(1-2L_{max}\gamma)q} < 1.$$

Then

$$\mathbb{E}[F(\overline{\mathbf{x}}^k) - F^\star] \le \rho^k (F(\overline{\mathbf{x}}^0) - F^\star).$$

- Allows the constant step-size.
- Obtains linear rate convergence.

Variance reduction techniques: Mini-batch variance reduction

Accelerated mini-batch prox-SVR (Acc. MB SPGD-VR)

1. Choose $q \in \mathbb{N}$, initialization $\overline{\mathbf{x}}^0 \in \mathbb{R}^p$, stepsize $\gamma > 0$, accelerated stepsize $\beta = (1 - \sqrt{\mu\gamma})/(1 + \sqrt{\mu\gamma})$.

2. For $k = 0, 1, \ldots$ perform:

2a. $\bar{\mathbf{x}} = \bar{\mathbf{x}}^k$, $\mathbf{x}^0 = \mathbf{y}^1 = \overline{\mathbf{x}}$; $\nabla f(\overline{\mathbf{x}}) = \frac{1}{m}\sum_{i=1}^m \nabla f_i(\overline{\mathbf{x}})$.

2b. For $l = 0, 1, \ldots, q-1$, perform:

$$\begin{cases} \text{pick } I_l \subset \{1, \ldots, m\}\text{: mini-batch of size } s, \\ \overline{G(\mathbf{y}^l, I_l)} = \nabla f_{I_l}(\mathbf{y}_l) - \nabla f_{I_l}(\overline{\mathbf{x}}) + \nabla f(\overline{\mathbf{x}}), \\ \mathbf{x}^{l+1} = \mathrm{prox}_{\gamma g}(\mathbf{y}^l - \gamma \overline{G(\mathbf{y}^l, I_l)}), \\ \mathbf{y}^{l+1} = \mathbf{x}^{l+1} + \beta(\mathbf{x}^{l+1} - \mathbf{x}^l). \end{cases}$$

3. Update $\overline{\mathbf{x}}^{k+1} = \mathbf{x}^q$.

• A mini-batch of size s is indexed by $I = \{i_1, \ldots, i_s\}$, where each $i_j \in \{1, \ldots, m\}$ is chosen uniformly at random, and

$$f_I = \frac{1}{s}\sum_{j=1}^{s} f_{i_j}.$$

• s components are chosen instead of one + an accelerated step.

Convergence of Acc. MB SPGD-VR

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := \frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x}) + g(\mathbf{x}) \right\}$$

Theorem (Mean convergence of Acc. MB SPGD-VR [6])

Set $L_{max} = \max_{1\le i\le m} L_i$, *where* L_i *is Lipschitz constant of* ∇f_i, *and suppose that:*

1. $0 < \gamma \le \gamma_{\max} = \min\left\{ \frac{(\alpha q)^2 (m-1)^2 \mu}{64(m-s)^2 L_{max}^2}, \frac{1}{2L_{max}} \right\}$ *for some* $0 < \alpha < 1/8$.
2. $q \ge \frac{1}{(1-\alpha)\sqrt{\mu\gamma}} \log \frac{1-\alpha}{\alpha}$.

Then,

$$\mathbb{E}[F(\overline{\mathbf{x}}^k) - F^\star] \le \rho^k (F(\overline{\mathbf{x}}^0) - F^\star),$$

where $\rho = 2\alpha(2+\alpha)/(1-\alpha) < 1$.

- Allows the constant step-size.
- Obtains linear rate convergence.

Taxonomy of algorithms

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := \frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x}) + g(\mathbf{x}) \right\}.$$

- $f(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x})$: μ-strongly convex with L-Lipschitz continuous gradient.

Gradient descent	Acc. MB SPGD-VR	SPGD-VR	SPGD
Linear	Linear	Linear	Sublinear

Table: Rate of convergence.

- $\kappa = L/\mu$ and $s_0 = 8\sqrt{\kappa}m(\sqrt{2}\alpha(m-1) + 8\sqrt{\kappa})^{-1}$ for $0 < \alpha \leq 1/8$.

SPGD-VR	Acc. MB SPGD-VR $s < \lceil s_0 \rceil$	AccProxGrad
$\mathcal{O}((m+\kappa)\log(1/\varepsilon))$	$\mathcal{O}((m+\kappa\frac{m-s}{m-1})\log(1/\varepsilon))$	$\mathcal{O}((m\kappa)\log(1/\varepsilon))$

Table: Complexity to obtain ε-solution.

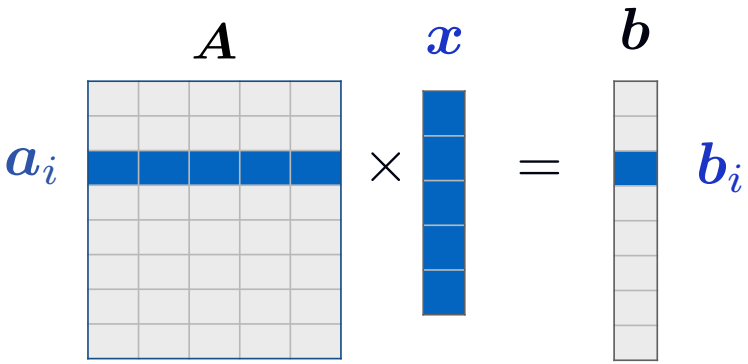
Remarks:

- $s = 1$: Acc. MB SPGD-VR has the same complexity as that of SPGD-VR.
- $s = m$: Acc. MB SPGD-VR has the same complexity of accelerated proximal gradient (AccProxGrad).

A good choice of mini-batch size may outperfom both AccProxGrad and SPGD-VR.

Another way of parsing data

Example (Least squares): $\min_{\mathbf{x}} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$



Using a subset of rows

We have mainly focused on using a subset of rows instead of the full data at each iteration.
This way, we compute an unbiased estimate $G(\mathbf{x}^k, i_k)$ of the gradient using

- a subset of data points: $(\mathbf{a}_{i_k}, b_{i_k})$,
- and the whole decision variable: $\mathbf{x}^k$:

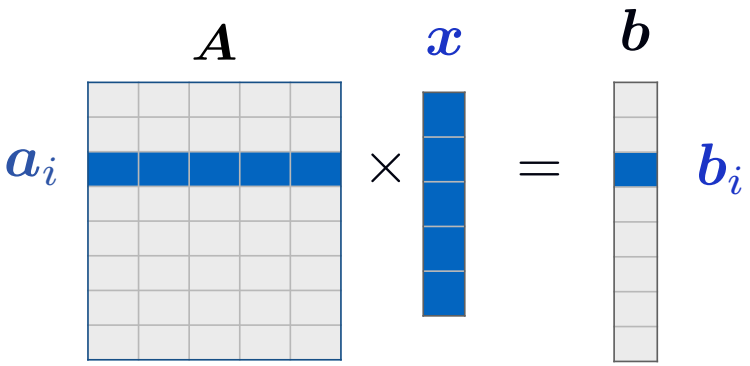
$$G(\mathbf{x}^k, i_k) = \mathbf{a}_{i_k}^T(\langle \mathbf{a}_{i_k}^T, \mathbf{x}\rangle - \mathbf{b}_{i_k}).$$

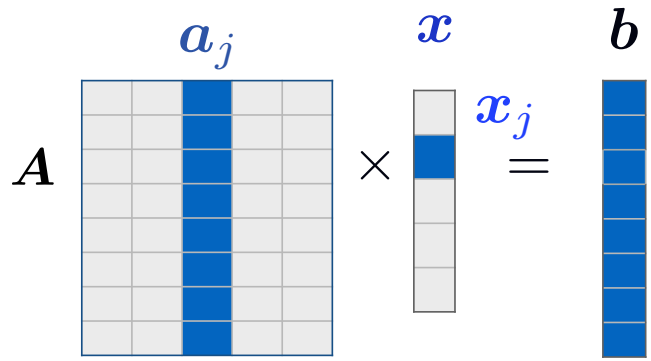
Estimate $G(\mathbf{x}^k, i_k)$ is dense, so we update the whole decision variable.

Next: Using a subset of columns.

Another way of parsing data

Example (Least squares): $\min_{\mathbf{x}} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$





Using a subset of columns

Denote the standard basis vectors by $\mathbf{e}_i$, and the corresponding directional derivatives by ∇_i. Let $\mathbf{a}_i$ represent the ith column of matrix $\mathbf{A}$. Consider the following unbiased estimate:

$$G(\mathbf{x}^k, i_k) = p\nabla_{i_k} f(\mathbf{x}^k)\mathbf{e}_{i_k} = p\langle \mathbf{a}_{i_k}, \mathbf{a}_{i_k}\mathbf{x}_{i_k}^k - \mathbf{b}\rangle \mathbf{e}_{i_k}.$$

This way, we compute an unbiased estimate $G(\mathbf{x}^k, i_k)$ of the gradient using

- a subset of columns $(\mathbf{a}_{i_k})$ and the whole measurement vector $\mathbf{b}$,
- and only the chosen coordinates of decision variable: $\mathbf{x}_{i_k}^k$.

Estimate $G(\mathbf{x}^k, i_k)$ is sparse, only coordinates chosen by i_k are nonzero.
Hence, we update these coordinates only.

Coordinate descent methods (CD)

A special case of stochastic gradient methods

Randomized CD methods can be viewed as a special case of SG methods, in which $G(\mathbf{x}^k, i_k) = p\nabla_{i_k} f(\mathbf{x}^k)\mathbf{e}_{i_k}$, where i_k is chosen uniformly random from $\{1, \ldots, p\}$, since,

$$\mathbb{E}[G(\mathbf{x}^k, i_k)] = p\mathbb{E}[\nabla_{i_k} f(\mathbf{x}^k)\mathbf{e}_{i_k}] = \sum_{i=1}^{p} \nabla_{i_k} f(\mathbf{x}^k)\mathbf{e}_{i_k} = \nabla f(\mathbf{x}^k).$$

Coordinate descent is more than a special instance!

A proper theoretical analysis for CD is required because of the following distinctions compared to the stochastic gradient methods:

- CD provides a descent lemma, so by properly choosing the step-size, we can guarantee $f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k)$.
- In some cases, variance of the gradient estimates can be characterized. As a simple example, variance shrinks to zero as we converge to $\mathbf{x}^\star$ in unconstrained smooth convex minimization.
- CD is more than unbiased estimates. Theoretical analysis shows that, properly constructed biased estimates may outperform.
- CD can take advantage of easily computable geometrical properties like the directional Lipschitz constants.

Coordinate descent (CD): Background

CD methods have been popular over many years since:

- Reduce to a sequence of easier optimization problems to be solved, e.g., one-dimensional optimization.
- Each iteration activates one coordinate (block), and only activated coordinates need to be updated $\Rightarrow$ reduces problem's dimension.
- Often easy to implement.

Basic coordinate descent framework

Problem (Unconstrained smooth minimization)

$$f^\star = \min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}).$$

Assume that f is a differentiable and the solution set is nonempty and bounded.

Basic coordinate descent algorithm

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$.
2. For $k = 0, 1, \ldots$ perform:
2a. Choose $i_k \in \{1, \ldots, p\}$.
2b. Choose stepsize γ_k.
2c. Update

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla_{i_k} f(\mathbf{x}^k) \boldsymbol{e}_{i_k}.$$

There are many variants within this framework, relying of different approaches for

- selection of coordinate i_k,
- and selection of step size γ_k.

Some variants of coordinate descent methods

Selection of coordinate i_k:

- Cyclic: Cycles through the coordinates: $i_{k+1} = i_k + 1 \mod p + 1$.
- Essentially cyclic: Touch each coordinate i at least once in each p iterations.
- Randomized: select i_k at random and independently at each iteration.

Selection of step size γ_k:

- Short step: γ_k prescribed by global knowledge about properties of f.
- Line search: choose γ_k to approximately minimize f along coordinate direction i_k.
- Exact: choose γ_k to exactly minimize f along i_k coordinate.

Cyclic CD does not always converge

Powell's example [8]

Consider the following non-convex, continuously differentiable function $f\colon \mathbb{R}^3 \to \mathbb{R}$:

$$f(x_1, x_2, x_3) = -(x_1 x_2 + x_2 x_3 + x_3 x_1) + \sum_{i=1}^{3} (|x_i| - 1)_+^2,$$

where $x_+^2 = \begin{cases} 0, & \text{if } x < 0, \\ x^2, & \text{if } x \geq 0. \end{cases}$

This function has minimizers at the corners $(1, 1, 1)$ and $(-1, -1, -1)$ of the unit cube.
This is a non-convex example, but this problem can be solved by gradient descent.

Cyclic CD does not always converge

- Consider cyclic CD with exact minimization.
- Choose $\mathbf{x}^0$ near one of the vertices of the unit cube other than the solutions.

Then, $\mathbf{x}^k$ cycles around the neighborhoods of six points that are close to the six non-optimal vertices [7].

Kaczmarz algorithm

Kaczmarz algorithm

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$.
2. For $k = 0, 1, \ldots$ perform:
2a. Choose $i_k \in \{1, \ldots, n\}$
2b. Update

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\langle \mathbf{a}_{i_k}, \mathbf{x}^k \rangle - b_{i_k}\right)\mathbf{a}_{i_k}.$$

Kaczmarz algorithm

Kaczmarz algorithm is a classical iterative method for solving linear systems of equations, $\mathbf{Ax} = \mathbf{b}$. Let us consider a consistent system (i.e., a system that admits a solution) such that:

- $\mathbf{A} \in \mathbb{R}^{n \times p}$,
- $\|\mathbf{a}_i\|_2 = 1$ for $i = 1, \ldots, p$, where $\mathbf{a}_i^T$ is the ith row of $\mathbf{A}$.

Note that, we can preprocess $\mathbf{A}$ to satisfy this property.

• Kaczmarz algorithm chooses a single equation from the system at each iteration (or a block of equations for the block Kaczmarz algorithm), and projects the current iterate to the solution space of this equation.

Kaczmarz algorithm and coordinate descent

Kaczmarz is CD applied to a dual formulation

Consider the following constrained convex problem, which seeks a least-norm solution to the system $\mathbf{A}\mathbf{x} = \mathbf{b}$:

$$\min_{\mathbf{x}\in\mathbb{R}^p} \ \frac{1}{2}\|\mathbf{x}\|_2^2, \text{ s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}.$$

Then, Lagrange dual problem is

$$\min_{\mathbf{y}\in\mathbb{R}^p} \ \frac{1}{2}\|\mathbf{A}^T\mathbf{y}\|_2^2 - \mathbf{b}^T\mathbf{y}.$$

The CD step on this dual formulation with step $\gamma_k = 1$ gives

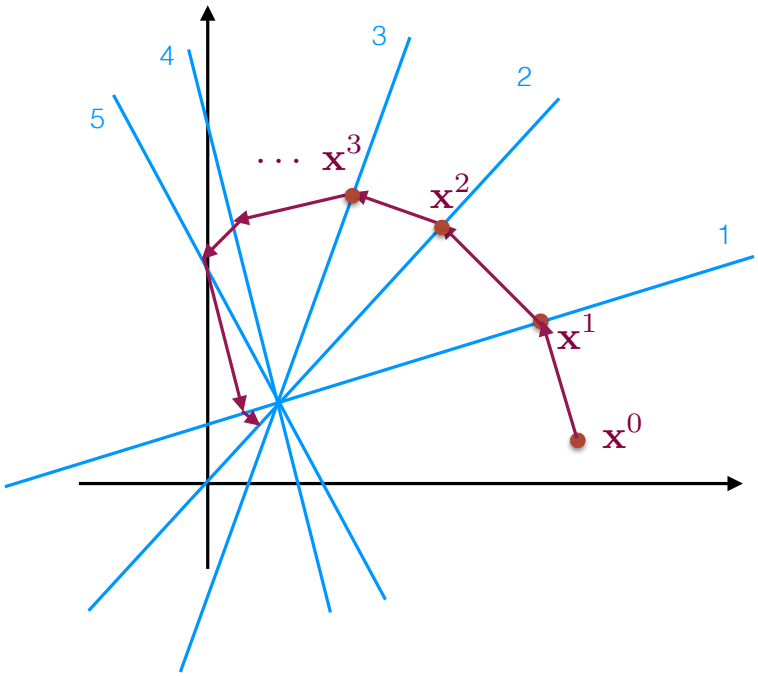
$$\mathbf{y}^{k+1} = \mathbf{y}^k - \left(\langle \mathbf{a}_{i_k}, \mathbf{A}^T\mathbf{y}^k\rangle - b_{i_k}\right)\boldsymbol{e}_{i_k}.$$

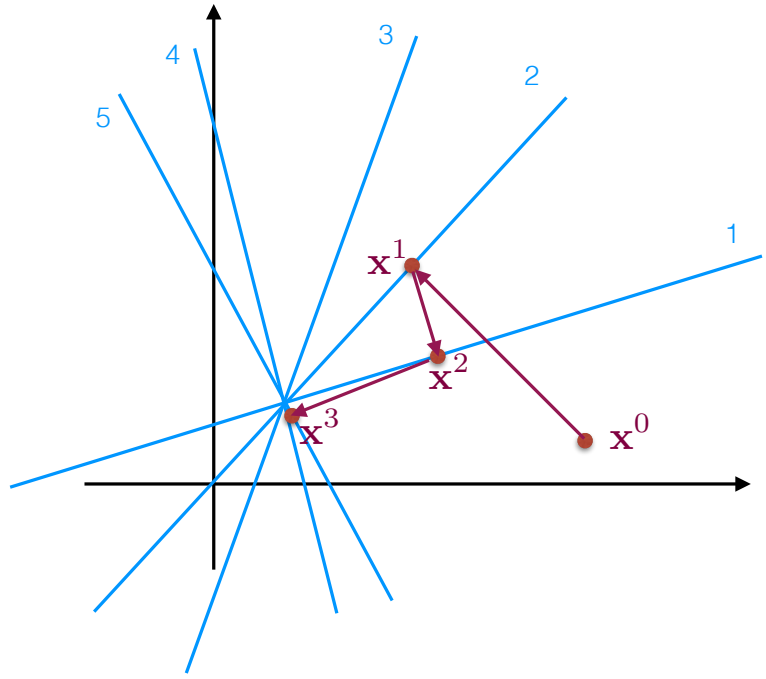
Multiplying both sides by $\mathbf{A}^T$, we get

$$\mathbf{A}^T\mathbf{y}^{k+1} = \mathbf{A}^T\mathbf{y}^k - \left(\langle \mathbf{a}_{i_k}, \mathbf{A}^T\mathbf{y}^k\rangle - b_{i_k}\right)\mathbf{a}_{i_k}.$$

Change of variable $\mathbf{x}^k = \mathbf{A}^T\mathbf{y}^k$ yields Kaczmarz algorithm.

Kaczmarz algorithm: Cyclic vs randomized





Kaczmarz algorithm: Cyclic vs randomized

- Convergence behavior depends heavily on the selection of i_k.
- Worst case characterization of cyclic variant does not capture the expected behavior well.
- Randomized variant performs better in the expectation.

Randomized CD algorithm

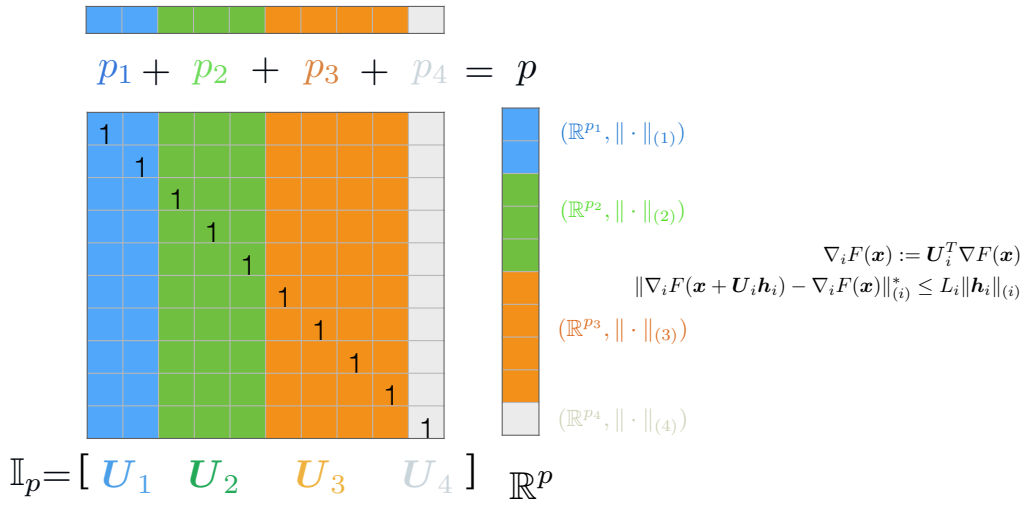
Randomized coordinate descent algorithm

1. Choose $\theta \in \mathbb{R}$ and $\mathbf{x}^0 \in \mathbb{R}^p$.

2. For $k = 0, 1, \ldots$ perform:

2a. Choose $i_k = \mathcal{A}_\theta$.

2b. Update $\mathbf{x}^{k+1} = \mathbf{x}^k - L_{i_k}^{-1} \boldsymbol{U}_{i_k} \left[\nabla_{i_k} f(\mathbf{x}^k)\right]^{\#}$.



- Sharp-operator : $[\mathbf{x}]^{\#} = \arg\max_{\boldsymbol{s} \in \mathbb{R}^p} \langle \mathbf{x}, \boldsymbol{s} \rangle - (1/2)\|\boldsymbol{s}\|^2 \implies$ for ℓ_2 norm, $[\mathbf{x}]^{\#} = \mathbf{x}$.
- $\mathcal{A}_\theta$ generates $i \in \{1, \ldots, s\}$ with probability $L_i^\theta / \sum_{j=1}^s L_j^\theta \implies$ for $\theta = 0$, uniform distribution.

Randomized CD algorithm

Theorem (Convergence of randomized CD [4, 7])

1. *Without strong convexity:*

$$\mathbb{E}[f(\mathbf{x}^k) - f^\star] \leq \begin{cases} \frac{\sum_{j=1}^s L_j^\theta}{k+4} R_{1-\theta}^2(\mathbf{x}^0), & \ell_2 - \textit{norm}, \\ \frac{s}{k+s}\left(R_1^2(\mathbf{x}_0)/2 + f(\mathbf{x}_0) - f^\star\right), & \theta = 0. \end{cases}$$

where $R_\theta(\mathbf{x}^0) = \max_{\{(\mathbf{x},\mathbf{x}^\star) | f(\mathbf{x}) \leq f(\mathbf{x}^0)\}} \|\mathbf{x} - \mathbf{x}^\star\|_{[\theta]}$ *and* $\|\mathbf{x}\|_{[\theta]}^2 = \sum_{i=1}^s L_i^\theta \|\mathbf{x}_i\|_{(i)}^2$.

2. *With strong convexity: Suppose that* f *is strongly convex with respect to the norm* $\|\cdot\|_{[1-\theta]}$ *with convexity parameter* $\mu_{1-\theta} > 0$. *Then*

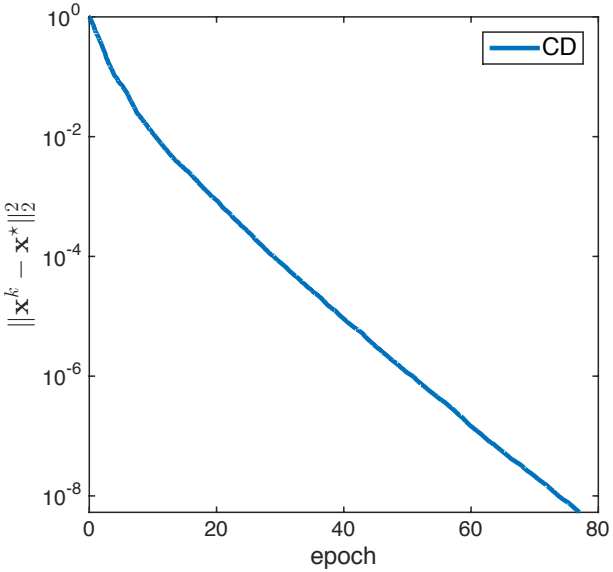
$$\mathbb{E}[f(\mathbf{x}^k) - f^\star] \leq \begin{cases} \left(1 - \mu_{1-\theta}/S_\theta\right)^k \left(f(\mathbf{x}^0) - f^\star\right), & \ell_2 - \textit{norm}, \\ \left(1 - 2\sigma/(s(1+\sigma))\right)^k \left(R_1^2(\mathbf{x}_0) + f(\mathbf{x}_0) - f^\star\right), & \theta = 0. \end{cases}$$

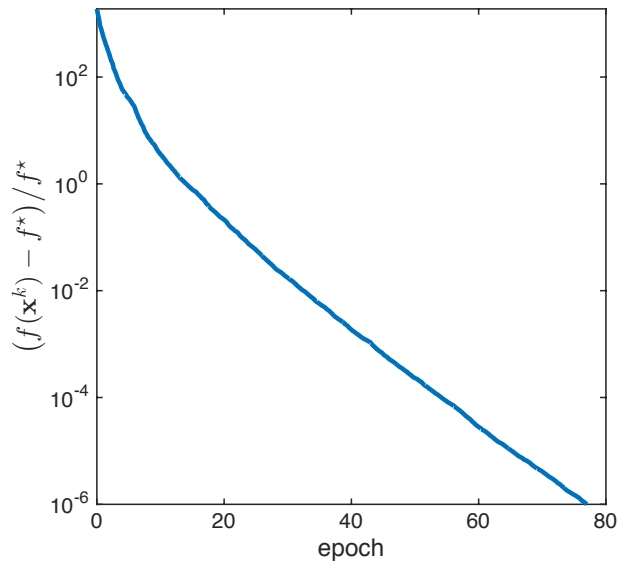
where $S_\theta = \sum_{i=1}^s L_i^\theta$.

- Recall that SPGM only gets the rate of $\mathcal{O}(1/\sqrt{k})$ for non strongly convex problems and $\mathcal{O}(1/k)$ for strongly convex problems.
- One needs the condition that the level set of f defined by $\mathbf{x}_0$ is bounded.

Example: Least squares problem

$$\min_{\mathbf{x}} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$$





Synthetic problem setup

- $\mathbf{A} := \mathrm{randn}(n, p)$ - standard Gaussian $\mathcal{N}(0, \mathbb{I})$, with $n = 1000$, $p = 500$.
- $\mathbf{x}^\natural \in \mathbb{R}^p$ with Gaussian i.i.d. entries, normalized to $\|\mathbf{x}^\natural\|_2 = 1$.
- $\mathbf{b} := \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is Gaussian white noise. SNR is 30dB.
- $\theta = 0$, so coordinates are chosen uniformly random.

Randomized accelerated CD

Randomized accelerated CD algorithm 1 (RACD1)

1. Choose $\mathbf{v}^0 = \mathbf{x}^0 \in \mathbb{R}^p$, $a_0 = 1/s$, $b_0 = 2$.

2. For $k = 0, 1, \ldots$ perform:

2a. Compute $\gamma_k \geq 1/s$ from equation $\gamma_k^2 - \frac{\gamma_k}{s} = \left(1 - \frac{\gamma_k \mu}{s}\right) \frac{a_k^2}{b_k^2}$

and set $\alpha_k = \frac{s - \gamma_k \mu}{\gamma_k (s^2 - \mu)}$ and $\beta_k = 1 - \frac{\gamma_k \mu}{s}$.

2b. Compute $\mathbf{y}^k = \alpha_k \mathbf{v}^k + (1 - \alpha_k)\mathbf{x}^k$.

2c. Choose $i_k \in \{1, \ldots, s\}$ uniformly at random.

2d. Update

$$\begin{cases} \mathbf{x}^{k+1} = \mathbf{y}^k - \frac{1}{L_{i_k}} \boldsymbol{U}_{i_k} \left[\nabla_{i_k} f(\mathbf{y}^k)\right]^{\#}, \\ \mathbf{v}^{k+1} = \beta_k \mathbf{v}^k + (1 - \beta_k)\mathbf{y}^k - \frac{\gamma_k}{L_{i_k}} \boldsymbol{U}_{i_k} \left[\nabla_{i_k} f(\mathbf{y}^k)\right]^{\#}. \end{cases}$$

2e. Update parameters $b_{k+1} = b_k / \sqrt{\beta_k}$ and $a_{k+1} = \gamma_k b_{k+1}$.

Recall

- s: number of blocks.
- L_i: Lipschitz constant of $\nabla_i f$; μ: strong convexity constant of f.
- Sharp-operator : $[\mathbf{x}]^{\#} = \arg\max_{\boldsymbol{s} \in \mathbb{R}^p} \ \langle \mathbf{x}, \boldsymbol{s} \rangle - (1/2)\|\boldsymbol{s}\|^2$.

Randomized accelerated CD

Theorem (Convergence of RACD1 [4])

$$\mathbb{E}[f(\mathbf{x}^k) - f^\star] \leq \left(\frac{s}{k+1}\right)^2 \left[2\|\mathbf{x}^0 - \mathbf{x}^\star\|_{[1]}^2 + \frac{1}{s^2}\left(f(\mathbf{x}^0) - f^\star\right)\right],$$

where

$$\|\mathbf{x}\|_{[1]} = \left(\sum_{i=1}^{s} L_i \|\mathbf{x}_i\|_{(i)}^2\right)^{1/2}$$

and L_i is Lipschitz constant of $\nabla_i f$.

The expected complexity of RACD1 for finding an ε-solution is of the order

$$\mathcal{O}\left(\frac{s}{\sqrt{\varepsilon}} \max_{1 \leq i \leq s} L_i\right)$$

which depends on the dimension.

Randomized accelerated CD: Dimensional independence [5]

Randomized accelerated CD algorithm 2 (RACD2)

1. Choose $\theta \in \mathbb{R}$, $\mathbf{v}^0 = \mathbf{x}^0 \in \mathbb{R}^p$, $a_0 = 1/s$, $b_0 = 1$, and $\sigma = \theta/2$.

2. For $k = 0, 1, \ldots$ perform:

2a. Choose $i_k = \mathcal{A}_\sigma$.

2b. Compute $\gamma_{k+1} > 0$ from equation $\gamma_{k+1}^2 S_\beta^2 = a_{k+1} b_{k+1}$ where $a_{k+1} = a_t + \gamma_{k+1}$ and $b_{k+1} = b_k + \mu_{1-\theta}\gamma_{k+1}$.

2c. Compute $\alpha_k = \frac{\gamma_{k+1}}{a_{k+1}}$, $\beta_k = \frac{\mu_{1-\alpha}\, a_{k+1}}{b_{k+1}}$, $\mathbf{y}^k = \alpha_k \mathbf{v}^k + (1-\alpha_k)\mathbf{x}^k$.

2d. Update

$$\begin{cases} \mathbf{x}^{k+1} = \mathbf{y}^k - \frac{1}{L_{i_k}} \boldsymbol{U}_{i_k} B_{i_k}^{-1} \nabla_{i_k} f(\mathbf{y}^k), \\ \mathbf{v}^{k+1} = \beta_k \mathbf{y}^k + (1-\beta_k)\mathbf{v}^k - \frac{\gamma_{k+1} \sum_{j=1}^s L_j^\sigma}{L_{i_k}^{1-\theta/2} b_{t+1}} \boldsymbol{U}_{i_k} B_{i_k}^{-1} \nabla_{i_k} f(\mathbf{y}^k). \end{cases}$$

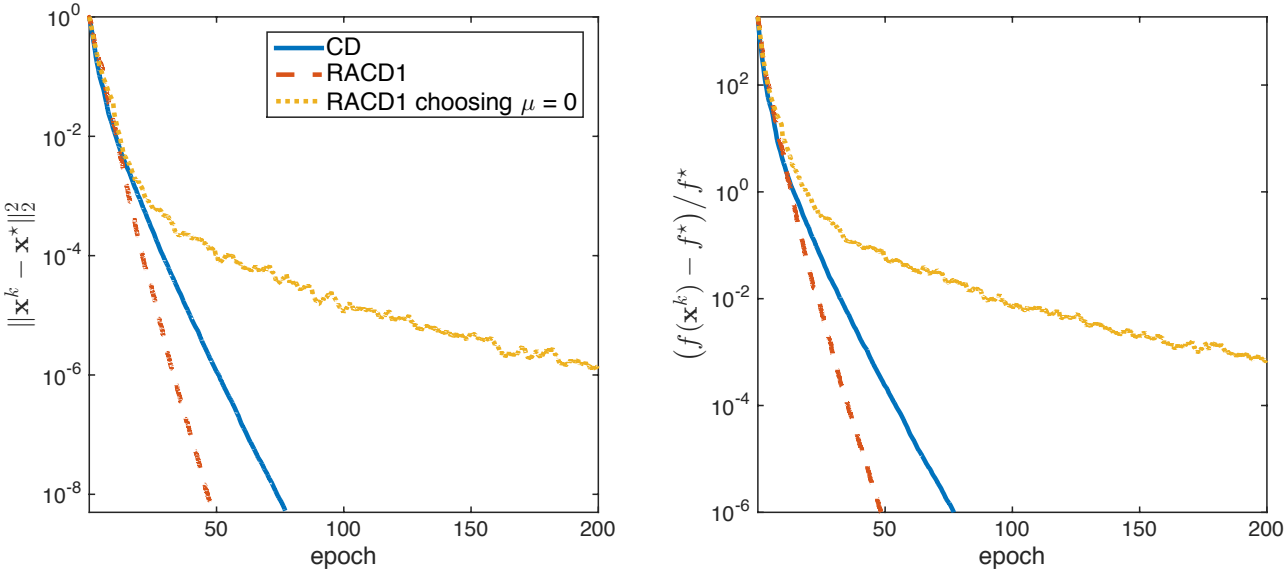
The expected complexity of RACD2 for finding an ε-solution is of the order

$$\mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}} \sum_{i=1}^s L_i^{1/2} \right).$$

This complexity is dimension independent.

Example: Least squares problem

$$\min_{\mathbf{x}} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$$

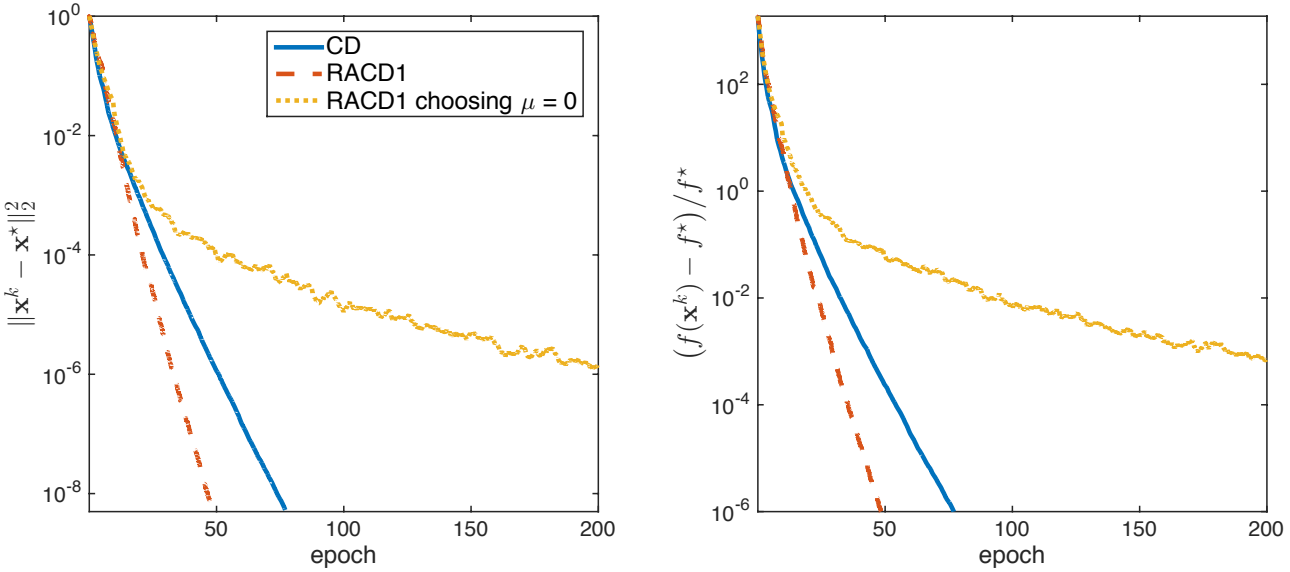


Synthetic problem setup

- $\mathbf{A} := \mathrm{randn}(n, p)$ - standard Gaussian $\mathcal{N}(0, \mathbb{I})$, with $n = 1000$, $p = 500$.
- $\mathbf{x}^\natural \in \mathbb{R}^p$ with Gaussian i.i.d. entries, normalized to $\|\mathbf{x}^\natural\|_2 = 1$.
- $\mathbf{b} := \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is Gaussian white noise. SNR is 30dB.
- $\theta = 0$, so coordinates are chosen uniformly random.

Example: Least squares problem

$$\min_{\mathbf{x}} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$$



Remarks

- CD adapts to the strong convexity without requiring μ as an input.
- RACD requires μ to be known. Otherwise, the rate becomes sublinear.
- Recall: This is also the case for gradient descent and its accelerated variants.

References

[1] O. Fercoq and P. Richatárik, Accelerated, parallel and proximal coordinate descent, *SIAM. J. Optim.,* vol. 25, pp. 1997–2023, 2016.

[2] O. Fercoq and P. Bianchi, A coordinate descent primal-dual algorithm with large step size and possibly non separable functions, *http://arxiv.org/abs/1508.04625,* 2015.

[3] Q. Lin, Z. Lu, L. Xiao, An accelerated proximal coordinate gradient method and its application to regularized empirical risk minimization. *SIAM J. Optim.,* vol. 25, pp. 2244–2273, 2015.

[4] Y. Nesterov, Efficiency of coordinate descent methods on huge-scale optimization, *SIAM. J. Optim.,* vol. 22, pp. 341–362, 2012.

[5] Y. Nesterov and S. Stich, Efficiency of accelerated coordinate descent method on structured optimization problems, *Preprint,* 2016.

[6] A. Nitanda, Stochastic proximal gradient descent with acceleration techniques. *NIPS*, pp. 1574–1582, 2014.

[7] S. J. Wright, Coordinates descent algorithms, *Math. Program.,* vol. 151, pp. 3–34, 2015.

[8] M.J.D. Powell, On search directions for minimization algorithms, *Math. Program.,* vol. 4, pp. 193–201, 1973.