Randomized Algorithms for Big Data Optimization

Peter Richtárik

University of Edinburgh

Graduate School in Systems, Optimization, Control and Networks
Belgium 2015

Contents I

Contents II

Contents III

Part 1

Randomized Gradient Methods for Strongly Convex Problems

The Problem

In order to quickly illustrate the topics and notions that we will study in more depth later, we first consider the following problem:

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x = (x^{(1)}, \ldots, x^{(n)}) \in \mathbb{R}^n \end{aligned} \tag{1}$$

We will assume that f is:

- **"smooth"** (will be made precise later)
- **strongly convex** (will be made precise later)

NSync: Randomized Gradient Descent with Arbitrary Sampling

Algorithm (NSync, R. and Takáč [11])

Input: initial point $x_0 \in \mathbb{R}^n$

subset probabilities $\{p_S\}$ for each $S \subseteq [n] \stackrel{\text{def}}{=} \{1, 2, \ldots, n\}$

stepsize parameters $v_1, \ldots, v_n > 0$

for $k = 0, 1, 2, \ldots$ **do**

a) **Select a random set of coordinates** $S_k \subseteq [n]$ following the law

$$\mathbf{P}(S_k = S) = p_S, \qquad S \subseteq [n]$$

b) **Update (possibly in parallel) selected coordinates:**

$$x_{k+1} = x_k - \sum_{i \in S_k} \frac{1}{v_i} (e_i^T \nabla f(x_k)) e_i$$

end for

Remark: This **NSync algorithm** was introduced in 2013. The first algorithm unifying deterministic gradient methods and randomized coordinate descent methods.

Two More Ways of Writing the Update Step

1. **Coordinate-by-coordinate:**

$$x_{k+1}^{(i)} = \begin{cases} x_k^{(i)}, & i \notin S_k, \\ x_k^{(i)} - \frac{1}{v_i}(\nabla f(x_k))^{(i)}, & i \in S_k. \end{cases}$$

2. **Via projection to a subset of blocks:** If for $h \in \mathbb{R}^n$ and $S \subseteq [n]$ we write

$$h_{[S]} \overset{\text{def}}{=} \sum_{i \in S} h^{(i)} e_i,$$

then

$$x_{k+1} = x_k + h_{[S_k]} \qquad \text{for} \qquad h = -(\mathrm{Diag}(v))^{-1} \nabla f(x_k). \tag{2}$$

We shall interchangeably write:

$$\nabla_i f(x) = e_i^T \nabla f(x) = (\nabla f(x))^{(i)}$$

Samplings

Definition 1 (Sampling)

By the name **sampling** we refer to a set valued random mapping with values being subsets of $[n] = \{1, 2, \ldots, n\}$. For sampling $\hat{S}$ we define the **probability vector** $p = (p_1, \ldots, p_n)^T$ by

$$p_i = \mathbf{P}(i \in \hat{S}) \tag{3}$$

We say that $\hat{S}$ is **proper**, if $p_i > 0$ for all i.

- A sampling $\hat{S}$ is uniquely characterized by the **probability mass function**

$$p_S \stackrel{\text{def}}{=} \mathbf{P}(\hat{S} = S), \quad S \subseteq [n]; \tag{4}$$

  that is, by assigning probabilities to all subsets of $[n]$.
- Later on it will be useful to also consider the **probability matrix** $P = (p_{ij})$ given by

$$p_{ij} \stackrel{\text{def}}{=} \mathbf{P}(i \in \hat{S}, j \in \hat{S}) = \sum_{S:\{i,j\} \subset S} p_S. \tag{5}$$

Samplings: A Basic Identity

Lemma 2 ([5])

For any sampling $\hat{S}$ we have

$$\sum_{i=1}^{n} p_i = \mathbf{E}[|\hat{S}|]. \tag{6}$$

Proof.

$$\sum_{i=1}^{n} p_i \overset{(3)+(4)}{=} \sum_{i=1}^{n} \sum_{S\subseteq[n]:i\in S} p_S = \sum_{S\subseteq[n]} \sum_{i:i\in S} p_S = \sum_{S\subseteq[n]} p_S|S| = \mathbf{E}[|\hat{S}|].$$

$\square$

Sampling Zoo - Part I

Why consider different samplings?

1. **Basic Considerations.** It is important that each block i has a positive probability of being chosen, otherwise NSync will not be able to update some blocks and hence will not converge to optimum. For technical/sanity reasons, we define:
   - **Proper sampling.** $p_i = \mathbf{P}(i \in \hat{S}) > 0$ for all $i \in [n]$
   - **Nil sampling:** $\mathbf{P}(\hat{S} = \emptyset) = 1$
   - **Vacuous sampling:** $\mathbf{P}(\hat{S} = \emptyset) > 0$
2. **Parallelism.** Choice of sampling affects the level of parallelism:
   - $\mathbf{E}[|\hat{S}|]$ is the average number of updates performed in parallel in one iteration; and is hence closely related to the number of iterations.
   - **serial sampling:** picks one block:

     $$\mathbf{P}(|\hat{S}| = 1) = 1$$

     We call this sampling serial although nothing prevents us from computing the actual update to the block, and/or to apply he update in parallel.

Sampling Zoo - Part II

- **fully parallel sampling:** always picks all blocks:

$$\mathbf{P}(\hat{S} = \{1, 2, \ldots, n\}) = 1$$

3. **Processor reliability.** Sampling may be induced/informed by the computing environment:
   - **Reliable/dedicated processors.** If one has reliable processors, it is sensible to choose sampling $\hat{S}$ such that $\mathbf{P}(|\hat{S}| = \tau) = 1$ for some τ related to the number of processors.
   - **Unreliable processors.** If processors given a computing task are busy or unreliable, they return answer later or not at all - it is then sensible to ignore such updates and move on. This then means that $|\hat{S}|$ varies from iteration to iteration.
4. **Distributed computing.** In a distributed computing environment it is sensible:
   - to allow each compute node as much autonomy as possible so as to **minimize communication cost**,
   - to make sure **all nodes are busy** at all times

Sampling Zoo - Part III

This suggests a strategy where the set of blocks is partitioned, with each node owning a partition, and independently picking a "chunky" subset of blocks at each iteration it will update, ideally from local information.

5. **Uniformity.** It may or may not make sense to update some blocks more often than others:
   - **uniform samplings:**

     $$\mathbf{P}(i \in \hat{S}) = \mathbf{P}(j \in \hat{S}) \quad \text{for all} \quad i, j \in [n]$$

     - **doubly uniform (DU):** These are samplings characterized by:

       $$|S'| = |S''| \;\Rightarrow\; \mathbf{P}(\hat{S} = S') = \mathbf{P}(\hat{S} = S'') \quad \text{for all} \quad S', S'' \subseteq [n]$$

     - **τ-nice:** DU sampling with the additional property that

       $$\mathbf{P}(|\hat{S}| = \tau) = 1$$

     - **distributed τ-nice:** will define later
     - **independent sampling:** union of independent uniform serial samplings
   - **nonuniform samplings**

Sampling Zoo - Part IV

6. **Complexity of generating a sampling.** Some samplings are computationally more efficient to generate than others: the potential benefits of a sampling may be completely ruined by the difficulty to generate sets according to the sampling's distribution.
   - a τ-nice sampling can be well approximated by an independent sampling, which is easy to generate...
   - in general, many samplings will be hard to generate

Assumption: Strong convexity

Assumption 1 (Strong convexity)

Let $\gamma > 0$ and $s = (s_1, \dots, s_n) \in \mathbb{R}^n$. We assume that function f is differentiable and γ-strongly convex (with $\gamma > 0$) with respect to the weighted Euclidean norm

$$\|h\|_s \quad \overset{def}{=} \quad \left(\sum_{i=1}^{n} s_i (h^{(i)})^2 \right)^{1/2} .$$

That is, we assume that for all $x, h \in \mathbb{R}^n$,

$$f(x+h) \geq f(x) + \langle \nabla f(x), h \rangle + \frac{\gamma}{2} \|h\|_s^2. \tag{7}$$

Assumption: Expected Separable Overapproximation

Assumption 2 (ESO)

Assume $\hat{S}$ is proper and that for some vector of positive weights $v = (v_1, \ldots, v_n)$ and all $x, h \in \mathbb{R}^n$,

$$\mathbf{E}[f(x + h_{[\hat{S}]})] \leq f(x) + \langle \nabla f(x), h \rangle_p + \tfrac{1}{2}\|h\|^2_{p \bullet v}. \qquad (8)$$

Note that **the ESO parameters v, p depend on both f and $\hat{S}$.** *For simplicity, we will often instead of (8) use the compact notation*

$$(f, \hat{S}) \sim ESO(v).$$

Notation used above:

$$h_{[S]} \overset{\text{def}}{=} \sum_{i \in S} h^{(i)} e_i \in \mathbb{R}^n \qquad \text{(projection of } h \in \mathbb{R}^n \text{ onto coordinates } i \in S)$$

$$\langle g, h \rangle_p \overset{\text{def}}{=} \sum_{i=1}^{n} p_i g^{(i)} h^{(i)} \in \mathbb{R} \qquad \text{(weighted inner product)}$$

$$p \bullet v \overset{\text{def}}{=} (p^{(1)} v^{(1)}, \ldots, p^{(n)} v^{(n)}) \in \mathbb{R}^n \qquad \text{(Hadamard product)}$$

Complexity of NSync

Theorem 3 ([11])

Let x_* be a minimizer of f. Let Assumptions 1 and 2 be satisfied for a proper sampling $\hat{S}$ (that is, $(f, \hat{S}) \sim ESO(v)$). Choose

- *starting point $x_0 \in \mathbb{R}^n$,*
- *error tolerance $0 < \epsilon < f(x_0) - f(x_*)$ and*
- *confidence level $0 < \rho < 1$.*

If $\{x_k\}$ are the random iterates generated by **NSync** *where the random sets S_k are iid following the distribution of $\hat{S}$, then*

$$\mathbf{K} \geq \frac{\mathbf{\Lambda}}{\gamma} \log\left(\frac{f(x_0) - f(x_*)}{\epsilon\rho}\right) \;\Rightarrow\; \mathbf{P}(f(x_{\mathbf{K}}) - f(x_*) \leq \epsilon) \geq 1 - \rho, \qquad (9)$$

where

$$\mathbf{\Lambda} \overset{def}{=} \max_{i=1,\dots,n} \frac{v_i}{p_i s_i} \geq \frac{\sum_{i=1}^n \frac{v_i}{s_i}}{\mathbf{E}[|\hat{S}|]}. \qquad (10)$$

What does this mean?

- **Linear convergence.** NSync converges linearly (i.e., logarithmic dependence on ϵ)
- **High confidence is not a problem.** ρ appears inside the logarithm, so it easy to achieve high confidence (by running the method longer; there is no need to restart)
- **Focus on the leading term.** The leading term is Λ; and we have a closed-form expression for it in terms of
  - parameters $v_1, \ldots, v_n$ (which depend on f and $\hat{S}$)
  - parameters $p_1, \ldots, p_n$ (which depend on $\hat{S}$)
- **Parallelization speedup.** The lower bound suggests that *if it was the case that* the parameters v_i did not grow with increasing $\tau \overset{\text{def}}{=} \mathbf{E}[|\hat{S}|]$, then we could potentially be getting linear speedup in τ (average number of updates per iteration).
  - So we shall **study the dependence of v_i on τ** (this will depend on f and $\hat{S}$)
  - As we shall see, speedup is often guaranteed for **sparse or well-conditioned problems.**

Question: How to **design** sampling $\hat{S}$ so that Λ is minimized?

Proof of Theorem 3 - Part I

- If we let $\mu \stackrel{\text{def}}{=} \gamma/\Lambda$, then

$$\begin{aligned} f(x+h) &\stackrel{(7)}{\geq} f(x) + \langle \nabla f(x), h\rangle + \tfrac{\gamma}{2}\|h\|_s^2 \\ &\geq f(x) + \langle \nabla f(x), h\rangle + \tfrac{\mu}{2}\|h\|_{v\bullet p^{-1}}^2. \end{aligned} \tag{11}$$

  Indeed, μ is defined to be the largest number for which $\gamma\|h\|_s^2 \geq \mu\|h\|_{v\bullet p^{-1}}^2$ holds for all h. Hence, f is μ-strongly convex with respect to the norm $\|\cdot\|_{v\bullet p^{-1}}$.

- Let x_* be a minimizer of f, i.e., an optimal solution of (22). Minimizing both sides of (11) in h, we get

$$\begin{aligned} f(x_*) - f(x) &\stackrel{(11)}{\geq} \min_{h\in\mathbb{R}^n} \langle \nabla f(x), h\rangle + \tfrac{\mu}{2}\|h\|_{v\bullet p^{-1}}^2 \\ &= -\tfrac{1}{2\mu}\|\nabla f(x)\|_{p\bullet v^{-1}}^2. \end{aligned} \tag{12}$$

Proof of Theorem 3 - Part II

- Let $h_k \stackrel{\text{def}}{=} -v^{-1} \bullet \nabla f(x_k)$. Then in view of (2), we have $x_{k+1} = x_k + (h_k)_{[\hat{S}]}$, and utilizing Assumption 2, we get

$$\begin{aligned}
\mathbf{E}[f(x_{k+1}) \mid x_k] &= \mathbf{E}\left[f(x_k + (h_k)_{[\hat{S}]}) \mid x_k\right] \\
&\overset{(8)}{\leq} f(x_k) + \langle \nabla f(x_k), h_k \rangle_p + \tfrac{1}{2}\|h_k\|^2_{p \bullet v} \\
&= f(x_k) - \tfrac{1}{2}\|\nabla f(x_k)\|^2_{p \bullet v^{-1}} \\
&\overset{(12)}{\leq} f(x_k) - \mu(f(x_k) - f(x_*)).
\end{aligned}$$

- Taking expectations in the last inequality (i.e., via the tower property), we get $\mathbf{E}[f(x_{k+1}) - f(x_*)] \leq (1-\mu)\mathbf{E}[f(x_k) - f(x_*)]$. Unrolling the recurrence, we get

$$\mathbf{E}[f(x_k) - f(x_*)] \leq (1-\mu)^k (f(x_0) - f(x_*)). \tag{13}$$

Proof of Theorem 3 - Part III

- Using Markov inequality, (13) and the definition of K, we finally get

$$\begin{aligned}\mathbf{P}(f(x_K) - f(x_*) \geq \epsilon) &\leq \mathbf{E}[f(x_K) - f(x_*)]/\epsilon \\ &\overset{(13)}{\leq} (1-\mu)^K (f(x_0) - f(x_*))/\epsilon \overset{(9)}{\leq} \rho.\end{aligned}$$

- Finally, let us now establish the lower bound on Λ. Letting $\Delta \overset{\text{def}}{=} \{p' \in \mathbb{R}^n : p' \geq 0, \sum_i p'_i = \mathbf{E}[|\hat{S}|]\}$, we have

$$\Lambda \overset{(10)}{=} \max_i \frac{v_i}{p_i s_i} \overset{(6)}{\geq} \min_{p' \in \Delta} \max_i \frac{v_i}{p'_i s_i} = \frac{1}{\mathbf{E}[|\hat{S}|]} \sum_{i=1}^n \frac{v_i}{s_i},$$

where the last equality follows since optimal p'_i is proportional to v_i/s_i.

Exercises

Exercise 1

Prove that a doubly uniform sampling is uniform.

Exercise 2

Let $f(x) = \frac{1}{2}\|Ax - b\|_2^2$ *and let* $\hat{S}$ *be a serial sampling. Show that then* $(f, \hat{S}) \sim ESO(v)$ *with* $v_i = \|A_{:i}\|_2^2$ *for* $i \in [n]$.

Exercise 3

Assume that f *is a convex function for which there exist constants* $L_1, \ldots, L_n > 0$ *such that for all* $x \in \mathbb{R}^n$, $t \in \mathbb{R}$ *and* $i \in [n]$, *the following inequality holds:*

$$|e_i^T \nabla f(x + te_i) - e_i^T \nabla f(x)| \leq L_i |t|.$$

Show that then for any serial sampling $\hat{S}$, *we have* $(f, \hat{S}) \sim ESO(v)$ *with* $v = (L_1, \ldots, L_n)$.

Exercise 4

Argue in detail why (12) *follows.*

Exercise 5

Argue in detail why $(1 - \mu)^K (f(x_0) - f(x_*))/\epsilon \leq \rho$.

Part 2

Blocks

The idea

We now assume the decision vector x has N **coordinates**

$$x \in \mathbb{R}^N$$

which we partition into n **"blocks"**.

Idea: We let the algorithm operate on "block level" instead $\Rightarrow$ **block coordinate descent.** That is, at iteration k,

- a random subset S_k of blocks $[n] = \{1, 2, \ldots, n\}$ is chosen
- and updated.

What do we gain by introducing blocks?

- **Flexibility:** We **can** partition the coordinates any way we like for any reason we might have.
  - Sometimes block structure is implied by the problem at hand. In L1 optimization, one often chooses $N_i = 1$ for all i. In group LASSO problems, groups correspond to blocks.
- **Generality:** By allowing for general block structure, we simultaneously analyze several classes of algorithms:
  - **coordinate descent** (if we choose $N_i = 1$ for all i)
  - **block coordinate descent** (if we choose $N_i > 1$ and $n > 1$)
  - **gradient descent** (if we choose $n = 1$)
  - **fast** ($O(1/k^2)$) versions of the above...
- **Efficiency:** It is sometimes more efficient to have blocks because:
  - this leads to a **more "chunky" workload for each processor** if we think that each processor handles one block
  - one can design **block-norms** based on data, which leads to better approximation and hence faster convergence
  - one can try to **optimize the partitioning of coordinates to blocks** (say, by trying to optimize complexity bounds, which depend on block structure)

Block Decomposition of $\mathbb{R}^N$

- **Partition.** Let $H_1, \dots, H_n$ be a partition of the set of coordinates/variables $\{1, 2, \dots, N\}$ into n nonempty subsets. Let $N_i = |H_i|$.
- **Projection/lifting matrices.** Let $U_i \in \mathbb{R}^{N \times N_i}$ be the column submatrix of the $N \times N$ identity matrix corresponding to coordinates in H_i.
- **Projection of $\mathbb{R}^N$ to $\mathbb{R}^{N_i}$** For $x \in \mathbb{R}^N$, define

$$x^{(i)} \stackrel{\text{def}}{=} U_i^T x \in \mathbb{R}^{N_i}, \quad i = 1, 2, \dots, n.$$

  Notice that $x^{(i)}$ is the block of coordinates of x belonging to H_i.
- **Lifting $\mathbb{R}^{N_i}$ to $\mathbb{R}^N$.** Given $x^{(i)} \in \mathbb{R}^{N_i}$, notice that the vector $s = U_i x^{(i)} \in \mathbb{R}^N$ has all blocks equal to 0 except for block i, which is equal to $x^{(i)}$. That is,

$$s^{(j)} = \begin{cases} x^{(j)} & j = i \\ 0 & \text{otherwise.} \end{cases}$$

Examples - Part I

Example 4

1. Single block.

$$n = 1; \qquad H_1 = \{1, 2, \ldots, N\}; \qquad U_1 = I$$

2. Blocks of size 1. This is the setting already introduced in NSync:

$$N = n; \qquad H_i = \{i\}; \qquad U_i = e_i$$

3. Two blocks of different sizes. Let $N = 5$ (5 coordinates), $n = 2$ (2 blocks) and let the partitioning be given by

$$H_1 = \{1, 3\}, \quad H_2 = \{2, 4, 5\}.$$

Then

$$U_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \qquad U_2 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Examples - Part II

For $x \in \mathbb{R}^N = \mathbb{R}^5$ we have

$$x^{(1)} = U_1^T x = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_3 \end{pmatrix} \in \mathbb{R}^{N_1} = \mathbb{R}^2$$

$$x^{(2)} = U_2^T x = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_4 \\ x_5 \end{pmatrix} \in \mathbb{R}^{N_2} = \mathbb{R}^3$$

On the other hand, for any $x \in \mathbb{R}^5$:

$$U_1 x^{(1)} = U_1(U_1^T x) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_1 \\ 0 \\ x_3 \\ 0 \\ 0 \end{pmatrix} \in \mathbb{R}^5$$

Examples - Part III

and

$$U_2 x^{(2)} = U_2(U_2^T x) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_2 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 0 \\ x_2 \\ 0 \\ x_4 \\ x_5 \end{pmatrix} \in \mathbb{R}^5$$

So, we have the **unique decomposition:**

$$x = U_1 x^{(1)} + U_2 x^{(2)}$$

The next simple result will formalize this.

Block Decomposition: Formal Statement

Proposition 1 (Block Decomposition)

Any vector $x \in \mathbb{R}^N$ *can be written uniquely as*

$$x = \sum_{i=1}^{n} U_i x^{(i)}, \tag{14}$$

where $x^{(i)} \in \mathbb{R}^{N_i}$. *Moreover,*

$$x^{(i)} = U_i^T x. \tag{15}$$

Proof.

Fix any $x \in \mathbb{R}^N$. Noting that $\sum_i U_i U_i^T$ is the $N \times N$ identity matrix, we have $x = \sum_i U_i U_i^T x$, where $U_i^T x \in \mathbb{R}^{N_i}$. Let us now show uniqueness. Assume that $x = \sum_i U_i x_1^{(i)} = \sum_i U_i x_2^{(i)}$, where $x_1^{(i)}, x_2^{(i)} \in \mathbb{R}^{N_i}$. Since

$$U_j^T U_i = \begin{cases} N_j \times N_j \quad \text{identity matrix,} & \text{if } i = j, \\ N_j \times N_i \quad \text{zero matrix,} & \text{otherwise,} \end{cases} \tag{16}$$

we get $0 = U_j^T(x - x) = U_j^T \sum_i U_i(x_1^{(i)} - x_2^{(i)}) = x_1^{(j)} - x_2^{(j)}$, for all j. $\square$

Projection onto (a subspace spanned by) a set of blocks

For $h \in \mathbb{R}^N$ and $\emptyset \neq S \subseteq [n] \overset{\text{def}}{=} \{1, 2, \ldots, n\}$, we write

$$h_{[S]} = \sum_{i \in S} U_i h^{(i)}. \tag{17}$$

In words, $h_{[S]}$ is a vector in $\mathbb{R}^N$ obtained from $h \in \mathbb{R}^N$ by zeroing out the blocks that do not belong to S. Hence:

$$(h_{[S]})^{(i)} = \begin{cases} h^{(i)}, & i \in S, \\ 0, & i \notin S. \end{cases}$$

Remark: This generalizes the decomposition on the slide defining ESO.

Norms in $\mathbb{R}^{N_i}$ and $\mathbb{R}^N$ - Part I

Let $\langle \cdot, \cdot \rangle$ denote the **standard inner product** between two vectors of equal size (i.e., $\langle x, y \rangle = x^\top y$).

With each block $i \in [n]$ we associate a positive definite matrix $B_i \in \mathbb{R}^{N_i \times N_i}$ and a scalar $v_i > 0$, and equip $\mathbb{R}^{N_i}$ and $\mathbb{R}^N$ with the **norms**

$$\|x^{(i)}\|_{(i)} \stackrel{\text{def}}{=} \langle B_i x^{(i)}, x^{(i)} \rangle^{1/2}, \quad \|x\|_v \stackrel{\text{def}}{=} \left(\sum_{i=1}^n v_i \|x^{(i)}\|_{(i)}^2 \right)^{1/2}. \qquad (18)$$

The corresponding **conjugate norms,** defined by

$$\|s\|^* = \max\{\langle s, x \rangle : \|x\| \leq 1\}$$

are given by

$$\|x^{(i)}\|_{(i)}^* \stackrel{\text{def}}{=} \langle B_i^{-1} x^{(i)}, x^{(i)} \rangle^{1/2}, \quad \|x\|_v^* = \left(\sum_{i=1}^n \frac{1}{v_i} \left(\|x^{(i)}\|_{(i)}^* \right)^2 \right)^{1/2}. \qquad (19)$$

Norms in $\mathbb{R}^{N_i}$ and $\mathbb{R}^N$ - Part II

For $w \in \mathbb{R}^n_{++}$ and $x, y \in \mathbb{R}^N$ we further define the **weighted inner product**

$$\langle x, y \rangle_w \stackrel{\text{def}}{=} \sum_{i=1}^{n} w_i \langle x^{(i)}, y^{(i)} \rangle. \tag{20}$$

For $x \in \mathbb{R}^N$, by Bx we mean the vector

$$Bx = \sum_{i=1}^{n} U_i B_i x^{(i)}.$$

That is, Bx is the vector in $\mathbb{R}^N$ whose ith block is equal to $B_i x^{(i)}$.

Lemma 5

For vectors $x, y \in \mathbb{R}^N$ we have

$$\|x + y\|_w^2 = \|x\|_w^2 + 2\langle Bx, y \rangle_w + \|y\|_w^2. \tag{21}$$

Norms: Examples

Example 6

Consider the following extreme special cases:

1. **Single block.** Let $n = 1$, $v = 1$ and B be a positive definite matrix. Then
$$\|x\|_{(1)} = \|x\|_v = \langle Bx, x\rangle^{1/2}, \quad x \in \mathbb{R}^N.$$
For instance, if $f(x) = \frac{1}{2}\|Ax - b\|^2$ we may choose:
   - $B = A^T A$ (assuming $A^T A$ is positive definite)
   - $B = \mathrm{Diag}(A^T A)$ (assuming no column in A is zero, $A^T A$ is positive definite)
2. **Blocks of size one.** Let $N_i = 1$ for all i and set $B_i = 1$. Then
$$\|t\|_{(i)} = \|t\|^*_{(i)} = |t|, \qquad t \in \mathbb{R}$$
and
$$\|x\|_v = \left(\sum_{i=1}^n v_i (x^{(i)})^2\right)^{1/2}, \qquad x \in \mathbb{R}^N.$$

Exercises

Exercise 7

Show that $\|\cdot\|_{(i)}^*$ *(resp.* $\|\cdot\|_v^*$*), as defined in* (19)*, is indeed the conjugate norm of* $\|\cdot\|_{(i)}$ *(resp.* $\|\cdot\|_v$*).*

Exercise 8

Prove Lemma 5.

Exercise 9

Generalize NSync to the block setting and provide a complexity analysis.

Part 3

Accelerated Randomized Gradient Methods for Weakly Convex Problems

The Problem

We will now consider the following problem:

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in \mathbb{R}^N \end{aligned} \tag{22}$$

We assume that f is:

- **"smooth"** (ESO Assumption 2)
- **(weakly) convex** (that is, Assumption 1 holds with $\gamma = 0$)

Remark: Notice that we now work in $\mathbb{R}^N$ as opposed to $\mathbb{R}^n$, as before. In this part we will partition the N variables into n **blocks**, and the algorithm we will describe and analyze—**ALPHA**—shall operate on blocks instead of individual coordinates.

Further simplifying notation

- By abuse of notation, we denote by u^2 the elementwise square of the vector u, by u^{-1} the elementwise inverse of vector u and by u^{-2} the elementwise square of u^{-1}.
- For vectors $v \in \mathbb{R}^n$ and $x \in \mathbb{R}^N$ we will write

$$v \cdot x \stackrel{\text{def}}{=} \sum_{i=1}^{n} v_i (U_i x^{(i)}). \tag{23}$$

  That is, $v \cdot x$ is the vector in $\mathbb{R}^N$ obtained from x by multiplying its block i by v_i for each $i \in [n]$.

Example 7

If all blocks are of size one ($N_i = 1$ for all i), then

$$v \cdot x = \mathrm{Diag}(v)x,$$

where $\mathrm{Diag}(v)$ is the diagonal matrix with diagonal vector v.

The ALPHA Algorithm

We now present an **accelerated** variant of **NSync**, called **ALPHA** [14] (for an earlier version, developed for *uniform* samplings, see [12]).

Algorithm (ALPHA)

Parameters: proper sampling $\hat{S}$ with probability vector $p = (p_1, \ldots, p_n)$, vector $v \in \mathbb{R}^n_{++}$, sequence $\{\theta_k\}_{k \geq 0}$
Initialization: choose $x_0 \in \mathbb{R}^N$, set $z_0 = x_0$
for $k \geq 0$ **do**
  $y_k = (1 - \theta_k) x_k + \theta_k z_k$
  Generate a random set of blocks $S_k \sim \hat{S}$
  $z_{k+1} \leftarrow z_k$
  for $i \in S_k$ **do**
    $z_{k+1}^{(i)} = z_k^{(i)} - \frac{p_i}{v_i \theta_k} B_i^{-1} \nabla_i f(y_k)$
  end for
  $x_{k+1} = y_k + \theta_k p^{-1} \cdot (z_{k+1} - z_k)$
end for

Efficient Implementation

Remark: The update step for y_k is expensive as it involves the addition of two potentially dense vectors in $\mathbb{R}^N$: x_k and z_k. However, this can be completely avoided by writing the method in an equivalent form (via a change of variables). See [12, 14] for details.

Iteration Complexity of ALPHA: Accelerated Case

Theorem 8

Let $\hat{S}$ be an arbitrary proper sampling and $v \in \mathbb{R}^n_{++}$ be such that $(f, \hat{S}) \sim ESO(v)$. Choose $\theta_0 \in (0,1]$ and define the sequence $\{\theta_k\}_{k\geq 0}$ by

$$\theta_{k+1} = \frac{\sqrt{\theta_k^4 + 4\theta_k^2} - \theta_k^2}{2}. \tag{24}$$

Then for any $y \in \mathbb{R}^N$ such that $C \geq 0$, the iterates $\{x_k\}_{k\geq 1}$ of **ALPHA** *satisfy:*

$$\mathbf{E}[f(x_k)] - f(y) \leq \frac{4C}{((k-1)\theta_0 + 2)^2}, \tag{25}$$

where

$$C = (1-\theta_0)\,(f(x_0) - f(y)) + \frac{\theta_0^2}{2}\|x_0 - y\|^2_{v \bullet p^{-2}}.$$

In particular, if we choose $\theta_0 = 1$, then for all $k \geq 1$,

$$\mathbf{E}[f(x_k)] - f(y) \leq \frac{2\|x_0 - y\|^2_{v\bullet p^{-2}}}{(k+1)^2} = \frac{2\sum_{i=1}^n \frac{v_i}{p_i^2}\|x_0^{(i)} - y^{(i)}\|_i^2}{(k+1)^2} . \tag{26}$$

Iteration Complexity of ALPHA: Non-Accelerated Case

Theorem 9

Let $\hat{S}$ be an arbitrary proper sampling and $v \in \mathbb{R}^n_{++}$ be such that $(f, \hat{S}) \sim ESO(v)$. Choose $\theta_k = \theta_0 \in (0, 1]$ for all $k \geq 0$. Then for any $y \in \mathbb{R}^N$, the iterates $\{x_k\}_{k\geq 1}$ of **ALPHA** *satisfy:*

$$\max\left\{\mathbf{E}[f(\hat{x}_k)], \min_{l=1,\ldots,k} \mathbf{E}[f(x_l)]\right\} - f(y) \leq \frac{C}{(k-1)\theta_0 + 1}, \forall k \geq 1 \quad (27)$$

where

$$\hat{x}_k = \frac{x_k + \theta_0 \sum_{l=1}^{k-1} x_l}{1 + (k-1)\theta_0}$$

and

$$C = (1 - \theta_0)\left(f(x_0) - f(y)\right) + \frac{\theta_0^2}{2}\|x_0 - y\|^2_{v \bullet p^{-2}}.$$

Analysis of ALPHA I

Let us extract the relations between the three sequences. Define

$$\tilde{z}_{k+1} \stackrel{\text{def}}{=} \arg\min_{z \in \mathbb{R}^N} \{\langle \nabla f(y_k), z \rangle + \frac{\theta_k}{2} \|z - z_k\|^2_{p^{-1} \bullet v}\}. \tag{28}$$

Then

$$z^{(i)}_{k+1} = \begin{cases} \tilde{z}^{(i)}_{k+1} & i \in S_k \\ z^{(i)}_k & i \notin S_k \end{cases}, \tag{29}$$

and hence $z_{k+1} - z_k = (\tilde{z}_{k+1} - z_k)_{[S_k]}$ and

$$x_{k+1} = y_k + \theta_k p^{-1} \cdot (\tilde{z}_{k+1} - z_k)_{[S_k]}. \tag{30}$$

Note also that from the definition of y_k in **ALPHA**, we have:

$$\theta_k (y_k - z_k) = (1 - \theta_k)(x_k - y_k). \tag{31}$$

Analysis of ALPHA: First Lemma

Lemma 10 ([14])

For any sampling $\hat{S}$ and any $x, a \in \mathbb{R}^N$ and $w \in \mathbb{R}^n_{++}$, the following identity holds:

$$\|x\|_w^2 - \mathbf{E}\left[\|x + a_{[\hat{S}]}\|_w^2\right] = \|x\|_{w\bullet p}^2 - \|x + a\|_{w\bullet p}^2.$$

Proof.

It is sufficient to notice that

$$\begin{aligned}
\mathbf{E}\left[\|x + a_{[\hat{S}]}\|_w^2\right] &\overset{(18)}{=} \mathbf{E}\left[\sum_{i\notin\hat{S}} w_i\|x^{(i)}\|_{(i)}^2 + \sum_{i\in\hat{S}} w_i\|x^{(i)} + a^{(i)}\|_{(i)}^2\right] \\
&= \sum_{i=1}^n \left[(1-p_i)w_i\|x^{(i)}\|_{(i)}^2 + p_i w_i\|x^{(i)} + a^{(i)}\|_{(i)}^2\right].
\end{aligned}$$

Analysis of ALPHA: Second Lemma

Lemma 11 ([14])

Let $\hat{S}$ be an arbitrary proper sampling and $v \in \mathbb{R}^n_{++}$ be such that

$$(f, \hat{S}) \sim ESO(v).$$

Let $\{\theta_k\}_{k\geq 0}$ be an arbitrary sequence of positive numbers in $(0,1]$ and fix $y \in \mathbb{R}^N$. Then for the sequence of iterates produced by **ALPHA** *and all $k \geq 0$, the following recursion holds:*

$$\mathbf{E}_k\left[f(x_{k+1}) + \frac{\theta_k^2}{2}\|z_{k+1} - y\|^2_{v \bullet p^{-2}}\right] \leq \left[f(x_k) + \frac{\theta_k^2}{2}\|z_k - y\|^2_{v \bullet p^{-2}}\right] - \theta_k(f(x_k) - f(y)) \ . \tag{32}$$

Proof of Theorem 8

If $\theta_0 \in (0, 1]$, the sequence $\{\theta_k\}_{k\geq 0}$ has the following properties (see [1]):

$$0 < \theta_{k+1} \leq \theta_k \leq \frac{2}{k + 2/\theta_0} \leq 1, \tag{33}$$

$$\frac{1-\theta_{k+1}}{\theta_{k+1}^2} = \frac{1}{\theta_k^2}. \tag{34}$$

After dividing both sides of (32) by θ_k^2, using (34) and taking expectations, we obtain:

$$\frac{1-\theta_{k+1}}{\theta_{k+1}^2}\phi_{k+1} + r_{k+1} \leq \frac{1-\theta_k}{\theta_k^2}\phi_k + r_k \leq \frac{1-\theta_0}{\theta_0^2}\phi_0 + r_0, \tag{35}$$

where $\phi_k \stackrel{\text{def}}{=} \mathbf{E}[f(x_k)] - f(y)$ and $r_k \stackrel{\text{def}}{=} \frac{1}{2}\mathbf{E}[\|z_k - y\|_{v\bullet p^{-2}}^2]$. Finally,

$$\begin{aligned}
\phi_k \quad &\stackrel{(34)}{=} \quad \tfrac{(1-\theta_k)\theta_{k-1}^2}{\theta_k^2}\phi_k \quad \leq \quad \tfrac{(1-\theta_k)\theta_{k-1}^2}{\theta_k^2}\phi_k + \theta_{k-1}^2 r_k \quad \stackrel{(35)}{\leq} \quad \tfrac{(1-\theta_0)\theta_{k-1}^2}{\theta_0^2}\phi_0 + \theta_{k-1}^2 r_0 \\
&= \quad \tfrac{\theta_{k-1}^2}{\theta_0^2}\left((1-\theta_0)\phi_0 + \theta_0^2 r_0\right) \quad = \quad \tfrac{\theta_{k-1}^2}{\theta_0^2} C \quad \stackrel{(33)}{\leq} \quad \tfrac{4C}{((k-1)\theta_0+2)^2}.
\end{aligned}$$

Note that in the last inequality we used the assumption that $C \geq 0$.

Proof of Theorem 9

Using the fact that $\theta_k = \theta_0$, for all k and taking expectation on both sides of (32), we obtain the recursion

$$\phi_{k+1} + \theta_0^2 r_{k+1} \leq (1-\theta_0)\phi_k + \theta_0^2 r_k, \quad k \geq 0.$$

Combining these inequalities, we get

$$(1+\theta_0(k-1)) \min_{l=1,\dots,k} \phi_l \leq \phi_k + \theta_0 \sum_{l=1}^{k-1} \phi_l \leq (1-\theta_0)\phi_0 + \theta_0^2 r_0. \quad (36)$$

Let $\alpha_k = 1 + (k-1)\theta_0$. By convexity,

$$f(\hat{x}_k) = f\left(\frac{x_k + \sum_{l=1}^{k-1} \theta_0 x_l}{\alpha_k}\right) \leq \frac{f(x_k) + \sum_{l=1}^{k-1} \theta_0 f(x_l)}{\alpha_k}.$$

Finally, subtracting $f(y)$ from both sides and taking expectations, we obtain

$$\mathbf{E}[f(\hat{x}_k)] - f(y) \leq \frac{\phi_k + \sum_{l=1}^{k-1} \theta_0 \phi_l}{\alpha_k} \overset{(36)}{\leq} \frac{(1-\theta_0)\phi_0 + \theta_0^2 r_0}{\alpha_k}.$$

Proof of Lemma 11 I

Based on how z_k is updated in ALPHA, we can write

$$a \stackrel{\text{def}}{=} \tilde{z}_{k+1} - z_k = -\theta_k^{-1}(v^{-1} \bullet p) \cdot B^{-1}\nabla f(y_k), \tag{37}$$

or equivalently, $-\nabla f(y_k) = \theta_k(v \bullet p^{-1}) \bullet Ba$. Using this notation, the update of vector x in ALPHA can be written as

$$x_{k+1} = y_k + \theta_k p^{-1} \cdot a_{[S_k]} = y_k + (\theta_k p^{-1} \bullet a)_{[S_k]}. \tag{38}$$

Letting $b = \tilde{z}_{k+1} - y$ and $t = \theta_k^2(v \bullet p^{-1})$, we apply the ESO assumption and rearrange the result:

$$\begin{aligned}
\mathbf{E}_k[f(x_{k+1})] &\stackrel{(8)+(38)}{\leq} f(y_k) + \left\langle \nabla f(y_k), \theta_k p^{-1} \cdot a \right\rangle_p + \frac{1}{2}\|\theta_k p^{-1} \cdot a\|^2_{v \bullet p} \\
&\stackrel{(20)+(18)+(37)}{=} f(y_k) - \frac{1}{2}\|a\|_t^2 \\
&\stackrel{(21)}{=} f(y_k) - \frac{1}{2}\|b\|_t^2 + \frac{1}{2}\|b - a\|_t^2 + \langle Ba, b - a\rangle_t . \quad (39)
\end{aligned}$$

Proof of Lemma 11 II

Note that $\|b\|_t^2 = \theta_k^2 \|\tilde{z}_{k+1} - y\|^2_{v \bullet p^{-1}}$, $\|b - a\|_t^2 = \theta_k^2 \|z_k - y\|^2_{v \bullet p^{-1}}$ and

$$
\begin{aligned}
\langle Ba, b-a\rangle_t &= \langle -Ba, a-b\rangle_t = \langle \theta_k^{-1}(v^{-1} \bullet p) \cdot \nabla f(y_k), y - z_k\rangle_t \\
&= \theta_k \langle \nabla f(y_k), y - z_k\rangle \\
&\overset{(31)}{=} \theta_k \langle \nabla f(y_k), y - y_k\rangle + (1-\theta_k)\langle \nabla f(y_k), x_k - y_k\rangle \\
&\leq \theta_k (f(y) - f(y_k)) + (1-\theta_k)(f(x_k) - f(y_k)).
\end{aligned}
$$

Substituting these expressions to (39), we obtain the recursion:

$$
\mathbf{E}_k[f(x_{k+1})] \leq \theta_k f(y) + (1-\theta_k) f(x_k) + \frac{\theta_k^2}{2}\|z_k - y\|^2_{v\bullet p^{-1}} - \frac{\theta_k^2}{2}\|\tilde{z}_{k+1} - y\|^2_{v\bullet p^{-1}}. \tag{40}
$$

It now only remains to apply Lemma 10 to the last two terms in (40), with $x \leftarrow z_k - y$, $w \leftarrow v \bullet p^{-2}$ and $\hat{S} \leftarrow S_k$, and rearrange the resulting inequality.

Exercises

Exercise 10
Prove (33).

Exercise 11
Prove (34).

Exercise 12 (*)
Prove a version of Theorem 9 where the left hand side is $\mathbf{E}[f(x_k)] - f(y)$.

Part 4

Samplings

Samplings: Definition

Recall:

Definition 12 (Sampling)

Sampling is a *random set-valued mapping* $\hat{S}$ with values in $2^{[n]}$, the collection of subsets of $[n] = \{1, 2, \ldots, n\}$.

Sum Over a Random Index Set

Theorem 13 (Sum over a random index set)

Let $\emptyset \neq J, J_1, J_2 \subset [n]$ *and* $\hat{S}$ *be any sampling. If* θ_i, $i \in [n]$, *and* θ_{ij}, *for* $(i,j) \in [n] \times [n]$ *are real constants, then*[1]

$$\mathbf{E}\left[\sum_{i \in J \cap \hat{S}} \theta_i\right] = \sum_{i \in J} p_i \theta_i,$$

$$\mathbf{E}\left[\sum_{i \in J \cap \hat{S}} \theta_i \mid |J \cap \hat{S}| = k\right] = \sum_{i \in J} \mathbf{P}(i \in \hat{S} \mid |J \cap \hat{S}| = k)\theta_i, \tag{41}$$

$$\mathbf{E}\left[\sum_{i \in J_1 \cap \hat{S}} \sum_{j \in J_2 \cap \hat{S}} \theta_{ij}\right] = \sum_{i \in J_1} \sum_{j \in J_2} p_{ij}\theta_{ij}. \tag{42}$$

[1]Sum over an empty index set will, for convenience, be defined to be zero.

Proof of Theorem 13

We prove the first statement, proof of the remaining statements is essentially identical:

$$
\begin{aligned}
\mathbf{E}\left[\sum_{i\in J\cap\hat{S}} \theta_i\right] &\overset{(4)}{=} \sum_{S\subset[n]} \left(\sum_{i\in J\cap S} \theta_i\right) \mathbf{P}(\hat{S}=S) \\
&= \sum_{i\in J} \sum_{S:i\in S} \theta_i \mathbf{P}(\hat{S}=S) \\
&= \sum_{i\in J} \theta_i \sum_{S:i\in S} \mathbf{P}(\hat{S}=S) \\
&= \sum_{i\in J} p_i \theta_i .
\end{aligned}
$$

Consequences of Theorem 13

Corollary 14 ([5])

Let $\emptyset \neq J \subset [n]$ *and* $\hat{S}$ *be an arbitrary sampling. Further, let* $a, h \in \mathbb{R}^N$, $w \in \mathbb{R}^n_+$ *and let* g *be a block separable function, i.e.,* $g(x) = \sum_i g_i(x^{(i)})$. *Then*

$$\mathbf{E}\left[|J \cap \hat{S}|\right] = \sum_{i \in J} p_i, \tag{43}$$

$$\mathbf{E}\left[|J \cap \hat{S}|^2\right] = \sum_{i \in J} \sum_{j \in J} p_{ij}, \tag{44}$$

$$\mathbf{E}\left[\langle a, h_{[\hat{S}]} \rangle_w\right] = \langle a, h \rangle_{p \bullet w}, \tag{45}$$

$$\mathbf{E}\left[\|h_{[\hat{S}]}\|_w^2\right] = \|h\|^2_{p \bullet w}, \tag{46}$$

$$\mathbf{E}\left[g(x + h_{[\hat{S}]})\right] = \sum_{i=1}^n \left[p_i g_i(x^{(i)} + h^{(i)}) + (1 - p_i) g_i(x^{(i)})\right]. \tag{47}$$

Moreover, the matrix $P \stackrel{def}{=} (p_{ij})$ *is positive semidefinite.*

Proof of Corollary 14

All 5 identities follow by applying Lemma 13 and observing that:

- $|J \cap \hat{S}| = \sum_{i \in J \cap \hat{S}} 1$
- $|J \cap \hat{S}|^2 = (\sum_{i \in J \cap \hat{S}} 1)^2 = \sum_{i \in J \cap \hat{S}} \sum_{j \in J \cap \hat{S}} 1$
- $\langle a, h_{[\hat{S}]} \rangle_w = \sum_{i \in \hat{S}} w_i \langle a^{(i)}, h^{(i)} \rangle$
- $\|h_{[\hat{S}]}\|_w^2 = \sum_{i \in \hat{S}} w_i \|h^{(i)}\|_{(i)}^2$ and
- $$\begin{aligned} g(x + h_{[\hat{S}]}) &= \sum_{i \in \hat{S}} g_i(x^{(i)} + h^{(i)}) + \sum_{i \notin \hat{S}} g_i(x^{(i)}) \\ &= \sum_{i \in \hat{S}} g_i(x^{(i)} + h^{(i)}) + \sum_{i=1}^{n} g_i(x^{(i)}) - \sum_{i \in \hat{S}} g_i(x^{(i)}), \end{aligned}$$

Finally, for any $\theta = (\theta_1, \dots, \theta_n)^T \in \mathbb{R}^n$,

$$\theta^T P \theta = \sum_{i=1}^n \sum_{j=1}^n p_{ij} \theta_i \theta_j \overset{(42)}{=} \mathbf{E}\left[\left(\sum_{i \in \hat{S}} \theta_i\right)^2\right] \geq 0.$$

Remark: The above results hold for arbitrary samplings. Let us specialize them, in order of decreasing generality, to uniform, doubly uniform and nice samplings.

Identities: uniform samplings

If $\hat{S}$ is **uniform,** then from (43) using $J = [n]$ we get

$$p_i = \frac{\mathbf{E}[|\hat{S}|]}{n}, \qquad i \in [n]. \tag{48}$$

Plugging (48) into (43), (45), (46) and (47) yields

$$\mathbf{E}\left[|J \cap \hat{S}|\right] = \frac{|J|}{n}\mathbf{E}[|\hat{S}|], \tag{49}$$

$$\mathbf{E}\left[\langle a, h_{[\hat{S}]}\rangle_w\right] = \frac{\mathbf{E}\left[|\hat{S}|\right]}{n}\langle a, h\rangle_w, \tag{50}$$

$$\mathbf{E}\left[\|h_{[\hat{S}]}\|_w^2\right] = \frac{\mathbf{E}\left[|\hat{S}|\right]}{n}\|h\|_w^2, \tag{51}$$

$$\mathbf{E}\left[g(x + h_{[\hat{S}]})\right] = \frac{\mathbf{E}[|\hat{S}|]}{n}g(x+h) + \left(1 - \frac{\mathbf{E}[|\hat{S}|]}{n}\right)g(x). \tag{52}$$

Identities: doubly uniform samplings

Consider the case $n > 1$; the case $n = 1$ is trivial. For **doubly uniform** $\hat{S}$, p_{ij} is constant for $i \neq j$:

$$p_{ij} = \frac{\mathbf{E}[|\hat{S}|^2 - |\hat{S}|]}{n(n-1)}. \tag{53}$$

Indeed, this follows from

$$p_{ij} = \sum_{k=1}^{n} \mathbf{P}(\{i,j\} \subseteq \hat{S} \mid |\hat{S}| = k)\mathbf{P}(|\hat{S}| = k) = \sum_{k=1}^{n} \frac{k(k-1)}{n(n-1)} \mathbf{P}(|\hat{S}| = k).$$

Substituting (53) and (48) into (44) then gives

$$\mathbf{E}[|J \cap \hat{S}|^2] = (|J|^2 - |J|) \frac{\mathbf{E}[|\hat{S}|^2 - |\hat{S}|]}{n \max\{1, n-1\}} + |J| \frac{|\hat{S}|}{n}. \tag{54}$$

Identities: τ-nice sampling

Finally, if $\hat{S}$ is **τ-nice** (and $\tau \neq 0$), then $\mathbf{E}[|\hat{S}|] = \tau$ and $\mathbf{E}[|\hat{S}|^2] = \tau^2$, which used in (54) gives

$$\mathbf{E}[|J \cap \hat{S}|^2] = \frac{|J|\tau}{n}\left(1 + \frac{(|J|-1)(\tau-1)}{\max\{1, n-1\}}\right). \tag{55}$$

Moreover, assume that $\mathbf{P}(|J \cap \hat{S}| = k) \neq 0$ (this happens precisely when $0 \leq k \leq |J|$ and $k \leq \tau \leq n - |J| + k$). Then for all $i \in J$,

$$\mathbf{P}(i \in \hat{S} \mid |J \cap \hat{S}| = k) = \frac{\binom{|J|-1}{k-1}\binom{n-|J|}{\tau-k}}{\binom{|J|}{k}\binom{n-|J|}{\tau-k}} = \frac{k}{|J|}.$$

Substituting this into (41) yields

$$\mathbf{E}\left[\sum_{i \in J \cap \hat{S}} \theta_i \mid |J \cap \hat{S}| = k\right] = \frac{k}{|J|}\sum_{i \in J} \theta_i. \tag{56}$$

Elementary Samplings, Intersection and Restriction

Definition 15 (Elementary samplings)

Elementary sampling associated with $J \subseteq [n]$ is sampling $\hat{E}_J$ for which

$$\mathbf{P}(\hat{E}_J = J) = 1.$$

Definition 16 (Intersection of samplings)

For two samplings $\hat{S}_1$ and $\hat{S}_2$ we define the intersection $\hat{S} \stackrel{\text{def}}{=} \hat{S}_1 \cap \hat{S}_2$ as the sampling for which:

$$\mathbf{P}(\hat{S} = S) = \mathbf{P}(\hat{S}_1 \cap \hat{S}_2 = S), \quad S \subseteq [n].$$

Definition 17 (Restriction of a sampling to a subset)

Let $\hat{S}$ be a sampling and $J \subseteq [n]$. By restriction of $\hat{S}$ to J we mean the sampling

$$\hat{E}_J \cap \hat{S}.$$

Probability matrices associated with samplings - Part I

Definition 18 (Probability matrix; [5])

With arbitrary sampling $\hat{S}$ we associate an n-by-n matrix $P = P(\hat{S})$ with entries

$$[P(\hat{S})]_{ij} = \mathbf{P}(i \in \hat{S}, j \in \hat{S}).$$

Lemma 19 (Intersection of independent samplings; [15])

Let $\hat{S}_1, \hat{S}_2$ be independent samplings. Then

$$P(\hat{S}_1 \cap \hat{S}_2) = P(\hat{S}_1) \bullet P(\hat{S}_2).$$

That is, the probability matrix of an intersection of independent samplings is the Hadamard product of their probability matrices.

Proof.

$[P(\hat{S}_1 \cap \hat{S}_2)]_{ij} = \mathbf{P}(\{i,j\} \in \hat{S}_1 \cap \hat{S}_2) = \mathbf{P}(\{i,j\} \in \hat{S}_1)\mathbf{P}(\{i,j\} \in \hat{S}_2) = [P(\hat{S}_1)]_{ij}[P(\hat{S}_2)]_{ij}$. $\square$

Probability matrices associated with samplings - Part II

Example 20 (Probability Matrix of an Elementary Sampling)

Note that the probability matrix of the elementary sampling $\hat{E}_J$ is the matrix

$$P(\hat{E}_J) \stackrel{\text{def}}{=} e_J e_J^T, \tag{57}$$

where e_J we denote the binary vector in $\mathbb{R}^n$ with ones in places corresponding to set J. That is,

$$[P(\hat{E}_J)]_{ij} = \begin{cases} 1 & i,j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

Hence, for arbitrary sampling $\hat{S}$, the probability matrix of $J \cap \hat{S}$ is the submatrix of $P(\hat{S})$ corresponding to the rows and columns indexed by J:

$$[P(J \cap \hat{S})]_{ij} = [P(\hat{E}_J) \bullet P(\hat{S})]_{ij} = \begin{cases} [P(\hat{S})]_{ij}, & i,j \in J, \\ 0, & \text{otherwise.} \end{cases} \tag{58}$$

Probability matrices associated with samplings - Part III

Lemma 21 (Decomposition of a Probability Matrix; [15])

Let $\hat{S}$ be any sampling. Then

$$P(\hat{S}) = \sum_{S \subseteq [n]} P(\hat{S} = S) P(\hat{E}_S). \tag{59}$$

That is, the probability matrix of arbitrary sampling is a convex combination of elementary probability matrices.

Proof.

Fix any $i, j \in [n]$. Since $(P(\hat{E}_S))_{ij} = 1$ iff $\{i, j\} \subseteq S$, from definition we have

$$\begin{aligned} (P(\hat{S}))_{ij} &= \textstyle\sum_{S:\{i,j\}\subseteq S} \mathbf{P}(\hat{S} = S) \\ &= \textstyle\sum_{S:\{i,j\}\subseteq S} \mathbf{P}(\hat{S} = S)(P(\hat{E}_S))_{ij} \\ &= \left(\textstyle\sum_{S:\{i,j\}\subseteq S} \mathbf{P}(\hat{S} = S) P(\hat{E}_S) \right)_{ij}. \end{aligned}$$

$\square$

Sampling Identity for a Quadratic

Lemma 22 ([15])

Let G be any real $n \times n$ matrix and $\hat{S}$ an arbitrary sampling. Then for any $h \in \mathbb{R}^n$ we have

$$\mathbf{E}\left[h_{[\hat{S}]}^T G h_{[\hat{S}]}\right] = h^T \left(P(\hat{S}) \bullet G\right) h, \tag{60}$$

where $\bullet$ denotes the Hadamard (elementwise) product of matrices.

Proof.

$$\begin{aligned}
\mathbf{E}\left[h_{[\hat{S}]}^T G h_{[\hat{S}]}\right] &\overset{(17)}{=} \mathbf{E}\left[\sum_{i \in \hat{S}} \sum_{j \in \hat{S}} G_{ij} h^{(i)} h^{(j)}\right] \\
&\overset{(42)}{=} \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} G_{ij} h^{(i)} h^{(j)} = h^T \left(P(\hat{S}) \bullet G\right) h.
\end{aligned}$$

$\square$

Distributed sampling

The following sampling is useful in the design of a **distributed coordinate descent method.**

Definition 23 (Distributed τ-nice sampling; [10, 13])

Let $\mathcal{P}_1, \dots, \mathcal{P}_c$ be a partition of $\{1, 2, \dots, n\}$ such that $|\mathcal{P}_l| = s$ for all l. That is, $sc = n$. Now let $\hat{S}_1, \dots, \hat{S}_c$ be independent τ-nice samplings from $\mathcal{P}_1, \dots, \mathcal{P}_c$, respectively. Then the sampling

$$\hat{S} \overset{\text{def}}{=} \cup_{l=1}^{c} \hat{S}_l, \tag{61}$$

is called **distributed τ-nice sampling.**

Idea: Blocks in $\mathcal{P}_l$, and all associated data, will be handled/stored by computer/node l only. Node l picks blocks in $\hat{S}_l$, computes the updates fro local information, and applies the updates to locally stored $x^{(i)}$ for $i \in \mathcal{P}_l$.

Probability Matrix of Distributed τ-nice Sampling

Consider the distributed τ-nice sampling and define:

- $E = P(\hat{E}_{[n]})$: the $n \times n$ matrix of all ones
- I be the $n \times n$ identity matrix
- $B = \sum_{l=1}^{c} P(\hat{E}_{\mathcal{P}_l})$: the 0-1 matrix with $B_{ij} = 1$ iff i, j belong to the same partition

Lemma 24 ([10, 15])

Consider the distributed τ-nice sampling $\hat{S}$. Its probability matrix can be written as

$$P(\hat{S}) = \frac{\tau}{s}\left[\alpha_1 I + \alpha_2 E + \alpha_3 (E - B)\right], \tag{62}$$

where

$$\alpha_1 = 1 - \frac{\tau - 1}{s s_1}, \qquad \alpha_2 = \frac{\tau - 1}{s_1}, \qquad \alpha_3 = \frac{\tau}{s} - \frac{\tau - 1}{s_1},$$

and $s_1 = \max\{1, s - 1\}$.

Proof of Lemma 24

Let $P = P(\hat{S})$. It is easy to see that

- $P_{ij} = \frac{\tau}{s} \stackrel{\text{def}}{=} \beta_3$ if $i = j$,
- $P_{ij} = \frac{\tau(\tau-1)}{ss_1} \stackrel{\text{def}}{=} \beta_2$ if $i \neq j$ and i, j belong to the same partition,
- $P_{ij} = \frac{\tau^2}{s^2} \stackrel{\text{def}}{=} \beta_3$ if $i \neq j$ belong to different partitions.

So, we can write

$$\begin{aligned} P &= \beta_1 I + \beta_2(B - I) + \beta_3(E - B) \\ &= (\beta_1 - \beta_2)I + \beta_2 E + (\beta_3 - \beta_2)(E - B). \end{aligned}$$

Exercises

Exercise 13

Find an expression for the probability matrix of

- *the τ-nice sampling,*
- *arbitrary doubly uniform sampling.*

Exercise 14

Let $\hat{S}$ be any sampling. Show that

- $\lambda_{max}(P) \leq \mathbf{E}[|\hat{S}|]$ *and that the bound is tight,*
- $P \succeq pp^T$.

Part 5

Functions

Introduction

- In this part we describe **three models** for f.
- These models can be thought of as function classes described by a list of properties.
- However, a single function may belong to more function classes.

In big data setting, some information is computationally difficult to extract from data.

Consider $f(x) = \frac{1}{2}\|Ax - b\|^2$.

- It is difficult to compute the largest eigenvalue of $A^T A$ if A is large (this is the Lipschitz constant of ∇f with respect to the standard Euclidean norm)
- It is easier to compute the squared norm of each column (these correspond to coordinate-wise Lipschitz constants).

Important point: The models differ in the amount of information they reveal about f.

Model: Quadratic Bound

Model 1 ([10, 13, 15])

We assume that

1. **Structure and Smoothness:** *$f : \mathbb{R}^N \to \mathbb{R}$ is differentiable and for all $x, h \in \mathbb{R}^N$ satisfies*

$$f(x+h) \leq f(x) + (\nabla f(x))^T h + \tfrac{1}{2} h^T A^T A h, \qquad (63)$$

where $A \in \mathbb{R}^{m \times N}$.

2. **Sparsity:** *Row j of A depends on blocks $i \in C_j$ only. Formally,*

$$C_j \stackrel{def}{=} \{i \ : \ A_{ji} \neq 0\},$$

where $A_{ji} \stackrel{def}{=} e_j^T A U_i \in \mathbb{R}^{1 \times N_i}$. Let $\omega_j \stackrel{def}{=} |C_j|$.

3. **Convexity:** *f is convex.*

Remark: Information about f is contained in the matrix A.

Examples

Example 25

In machine learning (ML), functions f of the following form are common:

$$f(x) = \sum_{j=1}^{m} f_j(x) = \sum_{j=1}^{m} \ell(x; a_j, y^j),$$

where N is the number of features, m number of examples, $a_j \in \mathbb{R}^N$ corresponds to jth example and y^j is a label associated with jth example.

Here are some convex loss functions ℓ often used in ML for which the total loss f satisfies (63):

Loss function ℓ	$f_j(x)$	(63) satisfied for A given by
square loss (SL)	$\frac{1}{2}(y^j - a_j^T x)^2$	$A_{j:} = a_j^T$
logistic loss (LL)	$\log(1 + \exp(-y^j a_j^T x))$	$A_{j:} = \frac{1}{2} a_j^T$
square hinge loss (HL)	$\frac{1}{2} \max\{0, 1 - y^j a_j^T x\}^2$	$A_{j:} = a_j^T$

Interpretation of ω_j (point 2 in Model 1) : # features in example j

Block gradients

Definition 26 (Block Gradients)

The ith **block gradient** of $f : \mathbb{R}^N \to \mathbb{R}$ at x is defined to be the ith block of the gradient of f at x:

$$\nabla_i f(x) \overset{\text{def}}{=} (\nabla f(x))^{(i)} = U_i^T \nabla f(x) \in \mathbb{R}^{N_i}. \tag{64}$$

In other words, $\nabla_i f(x)$ is the vector of partial derivatives with respect to coordinates belonging to block i.

Model: Classical

Model 2 ([2, 5, 9])

We assume that

1. **Structure:** *Function* $f : \mathbb{R}^N \to \mathbb{R}$ *is of the form*

$$f(x) = \sum_{j=1}^{m} f_j(x).$$

2. **Sparsity:** f_j *depends on* x *via blocks* $i \in C_j$ *only.*
3. **Convexity:** *Functions* $\{f_j\}$ *are convex.*
4. **Smoothness:** *Function* f *has block-Lipschitz gradient with constants* $L_1, \dots, L_n > 0$. *That is, for all* $i = 1, 2, \dots, n$,

$$\|\nabla_i f(x + U_i t) - \nabla_i f(x)\|_{(i)}^* \leq L_i \|t\|_{(i)}, \quad x \in \mathbb{R}^N, \ t \in \mathbb{R}^{N_i}. \quad (65)$$

Remark: Information about f is contained in the constants $L_1, \dots, L_n$.

Examples

Example 27 (Least squares)

Consider the quadratic function $f(x) = \frac{1}{2}\|Ax - b\|^2$.

(i) Consider the block setup with $N_i = 1$ (all blocks are of size 1) and $B_i = 1$ for all $i \in [n]$ (standard Eucl. norms for each block: $\|t\|_{(i)} = |t|$). Then $U_i = e_i$ and

$$\begin{aligned} \|\nabla_i f(x + U_i t) - \nabla_i f(x)\|_{(i)}^* &= |e_i^T A^T (A(x + te_i) - b) - e_i^T A^T (Ax - b)| \\ &= |e_i^T A^T A e_i||t| = \|A_{:i}\|^2 |t|, \end{aligned}$$

whence $L_i = \|A_{:i}\|^2$.

(ii) Choose nontrivial block sizes ($N_i > 1$) and define data-driven block norms with $B_i = A_i^T A_i$, where $A_i = AU_i$, assuming that $B_i \succ 0$. Then

$$\begin{aligned} \|\nabla_i f(x + U_i t) - \nabla_i f(x)\|_{(i)}^* &= \|U_i^T A^T (A(x + U_i t) - b) - U_i^T A^T (Ax - b)\|_{(i)}^* \\ &= \|U_i^T A^T A U_i t\|_{(i)}^* \\ &\overset{(19)}{=} \langle (A_i A_i^T)^{-1} U_i^T A^T A U_i t, U_i^T A^T A U_i t \rangle^{1/2} \\ &= \langle B_i t, t \rangle^{1/2} \overset{(18)}{=} \|t\|_{(i)}, \end{aligned}$$

whence $L_i = 1$.

Model: Newest

Model 3 ([12])

We assume that

1. **Structure:** $f : \mathbb{R}^N \to \mathbb{R}$ *is of the form*

$$f(x) = \sum_{j=1}^{m} f_j(x). \tag{66}$$

2. **Sparsity:** f_j *depends on* x *via blocks* $i \in C_j$ *only. Let* $\omega_j = |C_j|$.
   (Note that $i \notin C_j \Rightarrow L_{ji} = 0$*)*
3. **Convexity:** *Functions* $\{f_j\}$ *are convex.*
4. **Smoothness:** *Functions* $\{f_j\}$ *have block-Lipschitz gradient with constants* $L_{ji} \geq 0$*. That is, for all* $j = 1, 2, \ldots, m$ *and* $i = 1, 2, \ldots, n$,

$$\|\nabla_i f_j(x + U_i t) - \nabla_i f_j(x)\|_{(i)}^* \leq L_{ji} \|t\|_{(i)}, \quad x \in \mathbb{R}^N, \; t \in \mathbb{R}^{N_i}. \tag{67}$$

Remark: Information about f is contained in the constants $\{L_{ji}\}$

Computation of L_{ji}

We now give a formula for the constants L_{ji} in the case when f_j arises as a composition of a scalar function ϕ_j whose derivative has a known Lipschitz constant (this is often easy to compute), and a linear functional.

Proposition 2 ([12])

Let $f_j(x) = \phi_j(e_j^T Ax)$, where $\phi_j : \mathbb{R} \to \mathbb{R}$ is a function with L_{ϕ_j}-Lipschitz derivative:

$$|\phi_j(s) - \phi_j(s')| \leq L_{\phi_j}|s - s'|, \qquad s, s' \in \mathbb{R}. \tag{68}$$

Then f_j has a block Lipshitz gradient (i.e., satisfies (67)) with constants

$$L_{ji} = L_{\phi_j}\left(\|A_{ji}^T\|_{(i)}^*\right)^2, \qquad i = 1, 2, \ldots, n, \tag{69}$$

where

$$A_{ji} = e_j^T AU_i \tag{70}$$

(i.e., A_{ji} is the ith block of j-th row of A).

Proof of Proposition 2

For any $x \in \mathbb{R}^N$, $t \in \mathbb{R}^{N_i}$ and i we have

$$
\begin{aligned}
& \|\nabla_i f_j(x + U_i t) - \nabla_i f_j(x)\|_{(i)}^* \\
\overset{(64)}{=} \quad & \|U_i^T (e_j^T A)^T \phi_j'(e_j^T A(x + U_i t)) - U_i^T (e_j^T A)^T \phi_j'(e_j^T A x)\|_{(i)}^* \\
= \quad & \|A_{ji}^T \phi_j'(e_j^T A(x + U_i t)) - A_{ji}^T \phi_j'(e_j^T A x)\|_{(i)}^* \\
\leq \quad & \|A_{ji}^T\|_{(i)}^* |\phi_j'(e_j^T A(x + U_i t)) - \phi_j'(e_j^T A x)| \\
\overset{(68)}{\leq} \quad & \|A_{ji}^T\|_{(i)}^* L_{\phi_j} |A_{ji} t| \quad \leq \quad \|A_{ji}^T\|_{(i)}^* L_{\phi_j} \|A_{ji}^T\|_{(i)}^* \|t\|_{(i)},
\end{aligned}
$$

where the last step follows by applying the Cauchy-Schwartz inequality.

Examples

Example 28 (Least squares)

Consider the quadratic function

$$f(x) = \tfrac{1}{2}\|Ax - b\|^2 = \tfrac{1}{2}\textstyle\sum_{j=1}^{m}(e_j^T Ax - b_j)^2.$$

Then $f_j(x) = \phi_j(e_j^T Ax)$, where $\phi_j(s) = \frac{1}{2}(s - b_j)^2$ and $L_{\phi_j} = 1$.

(i) Consider the block setup with $N_i = 1$ (all blocks are of size 1) and $B_i = 1$ for all $i \in [n]$ (standard Euclidean norms for each block). Then by Proposition 2,

$$L_{ji} \overset{(69)}{=} L_{\phi_j}(\|A_{ji}^T\|_{(i)}^*)^2 = A_{ji}^2.$$

(ii) Choose nontrivial block sizes ($N_i > 1$) and define data-driven block norms with $B_i = A_i^T A_i$, where $A_i = AU_i$, assuming that the matrices $A_i^T A_i$ are positive definite. Then by Proposition 2,

$$L_{ji} \overset{(69)}{=} L_{\phi_j}(\|A_{ji}^T\|_{(i)}^*)^2 \overset{(19)}{=} \langle (A_i^T A_i)^{-1} A_{ji}^T, A_{ji}^T \rangle \overset{(70)}{=} e_j^T A_i (A_i^T A_i)^{-1} A_i^T e_j.$$

Part 6

Expected Separable Overapproximation

Introduction

In this part we shall look at the three models of f (Part 3) and various types of samplings $\hat{S}$ (Part 4) and compute parameters $v = (v_1, \ldots, v_n)$ such

$$(f, \hat{S}) \sim ESO(v).$$

These parameters are important since:

- They are **stepsize parameters** needed in the algorithm (in NSync, but also in other randomized block coordinate descent methods).
- Their size as a function of $\tau = \mathbf{E}[|\hat{S}|]$ describes achievable **parallelization speedup.**
- By computing v we get one step closer to ultimate goal of designing sampling $\hat{S}$ optimizing the complexity bound.

ESO($f \sim$ Model 1, $\hat{S} \sim$ arbitrary)

Theorem 29 ([15])

Let f satisfy assumptions in Model 1, assume **all blocks are of size 1** *($N_i = 1$) and $\hat{S}$ be* **any sampling.** *Then for all $x, h \in \mathbb{R}^N$,*

$$\mathbf{E}\left[f(x + h_{[\hat{S}]})\right] \leq f(x) + \langle \nabla f(x), h \rangle_p + \frac{1}{2}\|h\|^2_{p \bullet v}, \tag{71}$$

where v is any vector such that

$$P \bullet A^T A \preceq \mathrm{Diag}(p \bullet v), \tag{72}$$

where $P = P(\hat{S})$ is the probability matrix associated with $\hat{S}$.

Remark: The Hadamard product of two PSD matrices is PSD (P is PSD by Corollary 14).

Proof of Theorem 29

We have

$$
\begin{aligned}
\mathbf{E}\left[f(x+h_{[\hat{S}]})\right] &\overset{(63)}{\leq} \mathbf{E}\left[f(x)+\langle \nabla f(x), h_{[\hat{S}]}\rangle + \tfrac{1}{2}\langle A^T A h_{[\hat{S}]}, h_{[\hat{S}]}\rangle\right] \\
&\overset{(45)}{=} f(x)+\langle \nabla f(x), h\rangle_p + \tfrac{1}{2}\mathbf{E}\left[h_{[\hat{S}]}^T A^T A h_{[\hat{S}]}\right] \\
&\overset{(*)}{=} f(x)+\langle \nabla f(x), h\rangle_p + \tfrac{1}{2} h^T \left(P \bullet A^T A\right) h \\
&\leq f(x)+\langle \nabla f(x), h\rangle_p + \tfrac{1}{2}\underbrace{h^T \mathrm{Diag}(p\bullet v) h}_{=\|h\|^2_{p\bullet v}},
\end{aligned}
$$

where (*) comes from Lemma 22.

Ways of satisfying (72)

Let us fix a sampling $\hat{S}$ (and hence P) and data A. We can find v for which $P \bullet A^T A \preceq \mathrm{Diag}(p \bullet v)$ in several ways:

1. $v_i = \lambda_1 \|A_{:i}\|^2$ and

$$\lambda_1 = \max_{\theta \in \mathbb{R}^n} \{\theta^T (P \bullet A^T A)\theta \;:\; \theta^T \mathrm{Diag}(P \bullet A^T A)\theta \leq 1\}.$$

2. $v_i = \frac{\lambda_{max}(P \bullet A^T A)}{p_i}$.
3. $v_i = \lambda_{max}(A^T A)\frac{(\max_j p_j)}{p_i}$ (using Lemma 30 with $X = P$)
4. $v_i = \frac{\lambda_{max}(P)}{p_i} \max_i \|A_{:i}\|^2$ (using Lemma 30 with $X = A^T A$)

Lemma 30

For any two PSD matrices X, Y with nonnegative elements,

$$\lambda_{max}(X \bullet Y) \leq \lambda_{max}(X) \max_j Y_{jj}.$$

Eigenvalues of Probability Matrices

Definition 31 (Eigenvalues)

For arbitrary sampling $\hat{S}$ we define

$$\lambda(\hat{S}) \stackrel{\text{def}}{=} \max_{\theta\in\mathbb{R}^n}\{\theta^T P(\hat{S})\theta \;:\; \theta^T \mathrm{Diag}(P(\hat{S}))\theta \leq 1\}. \tag{73}$$

and

$$\lambda'(\hat{S}) \stackrel{\text{def}}{=} \max_{\theta\in\mathbb{R}^n}\{\theta^T P(\hat{S})\theta \;:\; \theta^T\theta \leq 1\}. \tag{74}$$

Example 32 (Elementary Sampling)

Fix $S \subseteq [n]$ and consider the elementary sampling $\hat{E}_S$. Note that

$$\lambda(\hat{E}_S) = \lambda_{max}(P(\hat{E}_S)) = \lambda_{max}(e_S e_S^T) = \|e_S\|^2 = |S|. \tag{75}$$

Since $J \cap \hat{E}_S = \hat{E}_{J\cap S}$, we get

$$\lambda(J \cap \hat{E}_S) = \lambda(\hat{E}_{J\cap S}) \stackrel{(75)}{=} |J \cap S|. \tag{76}$$

Insightful and Easily Computable Bound

Issues with Theorem 29:

- It does *not* provide insightful nor **easily computable** expressions for v_i (which are needed to run the algorithm).
- It does *not* answer the following **inverse problem:** given data matrix A and/or its sparsity pattern $\{C_j\}$, **design a "good" sampling.**

The following two results go a good way to overcoming these issues.

Theorem 33 (Useful ESO; [15])

Let the assumptions of Theorem 29 be satisfied. Then (72) *holds (i.e.,* $(f, \hat{S}) \sim ESO(v)$*) with* v *given by:*

$$v_i = \sum_{j=1}^{m} \lambda(C_j \cap \hat{S}) A_{ji}^2, \quad i = 1, 2, \dots, n. \tag{77}$$

Proof of Theorem 33

Note that it follows from (42) that for any vector $\theta \in \mathbb{R}^n$ and any j the following identity holds:

$$\mathbf{E}\left[\left(\sum_{i \in C_j \cap \hat{S}} \theta_i\right)^2\right] = \sum_{i=1}^{n} [P(C_j \cap \hat{S})]_{ij}\theta_i\theta_j = \theta^T P(C_j \cap \hat{S})\theta. \tag{78}$$

Fix $h \in \mathbb{R}^n$. Let $z_j = (z_j^{(1)}, \dots, z_j^{(n)})^T \in \mathbb{R}^n$ be defined as follows: $z_j^{(i)} = h^{(i)} A_{ji}$. We then have

$$\begin{aligned}
\mathbf{E}\left[h_{[\hat{S}]}^T A^T A h_{[\hat{S}]}\right] &= \sum_{j=1}^{m} \mathbf{E}\left[h_{[\hat{S}]}^T A_{j:}^T A_{j:} h_{[\hat{S}]}\right] = \sum_{j=1}^{m} \mathbf{E}\left[\left(\sum_{i \in C_j \cap \hat{S}} h^{(i)} A_{ji}\right)^2\right] \\
&\overset{(78)}{=} \sum_{j=1}^{m} z_j^T P(C_j \cap \hat{S}) z_j \overset{(73)}{\leq} \sum_{j=1}^{m} \lambda(C_j \cap \hat{S}) \left(z_j^T \operatorname{Diag}(P(C_j \cap \hat{S})) z_j\right) \\
&\overset{(58)}{=} \sum_{j=1}^{m} \lambda(C_j \cap \hat{S}) \sum_{i \in C_j} p_i (h^{(i)} A_{ji})^2 = \sum_{j=1}^{m} \lambda(C_j \cap \hat{S}) \sum_{i=1}^{n} p_i (h^{(i)} A_{ji})^2 \\
&= \sum_{i=1}^{n} p_i (h^{(i)})^2 \sum_{j=1}^{m} \lambda(C_j \cap \hat{S}) A_{ji}^2 = \sum_{i=1}^{n} p_i (h^{(i)})^2 v_i.
\end{aligned}$$

Useful bounds on $\lambda(\hat{S})$

Theorem 34 ([15])

Let $\hat{S}$ be an arbitrary sampling.

1. **Lower bound.** *If $\hat{S}$ is not nill, then* $\frac{\mathbf{E}[|\hat{S}|^2]}{\mathbf{E}[|\hat{S}|]} \leq \lambda(\hat{S})$.
2. **Upper bound.** *If $|\hat{S}| \leq \tau$ with probability 1, then $\lambda(\hat{S}) \leq \tau$.*
3. **Identity.** *If $|\hat{S}| = \tau$ with probability 1, then $\lambda(\hat{S}) = \tau$.*

Let us apply the 2nd part of the above theorem to the sampling $J \cap \hat{S}$:

Corollary 35

Let $\hat{S}$ be an arbitrary sampling, $J \subseteq [n]$ and c a constant such that $|J \cap \hat{S}| \leq c$ with probability 1. Then

$$\lambda(J \cap \hat{S}) \leq c.$$

In particular, if $|\hat{S}| \leq \tau$ with probability 1, then $|J \cap \hat{S}| \leq \min\{|J|, \tau\}$ with probability 1, and hence $\lambda(J \cap \hat{S}) \leq \min\{|J|, \tau\}$.

Remark: The above corollary is useful as we can apply it in connection with Theorem 33 with $J = C_j$ for $j = 1, 2, \ldots, m$.

Computing $\lambda(J \cap \hat{S})$: Product Sampling

Example 36 (Product Sampling)

Assume that the sets $\{C_j\}$ in Model 1 form a partition of $[n]$. The consider the sampling $\hat{S}$ defined as follows:

$$\mathbf{P}(\hat{S} = S) = \begin{cases} \left(\prod_{j=1}^{m} |C_j|\right)^{-1}, & S \in C_1 \times C_2 \times \cdots \times C_m, \\ 0, & \text{otherwise.} \end{cases}$$

Note that $|C_j \cap \hat{S}| = 1$ with probability 1, and hence by Corollary 35,

$$\lambda(C_j \cap \hat{S}) \leq 1.$$

On the other hand, by the first part of Theorem 34, $\lambda(C_j \cap \hat{S}) \geq 1$, and hence this sampling achieves the smallest possible value of the "λ parameters" in (77) (which is "good" as other things equal, ESO with small $\{v_i\}$ is better). Let us remark that $\mathbf{E}[|\hat{S}|] = m$.

Computing $\lambda(J \cap \hat{S})$: τ-Nice Sampling

Exercise 15 (τ-Nice Sampling)

Show by direct computation that if $\hat{S}$ is a τ-nice sampling, then the lower bound in part 1 of Theorem 34 is attained for $C_j \cap \hat{S}$ for all j:

$$\lambda(C_j \cap \hat{S}) \quad = \quad \frac{\mathbf{E}[|C_j \cap \hat{S}|^2]}{\mathbf{E}[|C_j \cap \hat{S}|]} \quad \overset{(55)+(49)}{=} \quad 1 + \frac{(\omega_j - 1)(\tau - 1)}{\max\{n-1, 1\}}, \qquad (79)$$

where $\omega_j = |C_j|$.

Computing $\lambda(J \cap \hat{S})$: Distributed τ-Nice Sampling - Part I

Exercise 16 (Distributed τ-Nice Sampling; [15])

Show that if $\hat{S}$ is the distributed τ-nice sampling, then

$$\lambda(C_j \cap \hat{S}) \leq \underbrace{1 + \frac{(\tau - 1)(\omega_j - 1)}{s_1}}_{\lambda_{1,j}} + \underbrace{\left(\frac{\tau}{s} - \frac{\tau - 1}{s_1}\right) \frac{\omega'_j - 1}{\omega'_j} \omega_j}_{\lambda_{2,j}}, \tag{80}$$

where $s_1 = \max\{1, s - 1\}$, $\omega_j = |C_j|$, *and* ω'_j *is the number of partitions "active" at row* j *of* A:

$$\omega'_j \overset{def}{=} |\{l \ : \ A_{ji} \neq 0 \text{ for some } i \in \mathcal{P}_l\}|.$$

Exercise 17

Show that if the number of partitions is 1 ($c = 1$), bound (80) for the distributed τ-nice sampling specializes to the bound (79) for the τ-nice sampling.

Computing $\lambda(J \cap \hat{S})$: Distributed τ-Nice Sampling - Part II

Lemma 37 ([15])

Consider the distributed τ-nice sampling. Suppose $\tau \geq 2$. For any $1 \leq \eta \leq s$ the following holds:

$$\left(\frac{\tau}{s} - \frac{\tau - 1}{s - 1}\right)\eta \leq \frac{1}{\tau - 1}\left(1 + \frac{(\tau - 1)(\eta - 1)}{s - 1}\right).$$

Note that Lemma 37 implies that

$$\lambda_{1,j} + \lambda_{2,j} \leq \left(1 + \frac{1}{\tau - 1}\right)\lambda_{1,j}. \tag{81}$$

Distributed NSync: Cost of Distribution

Assume f is 1-strongly convex, and consider running NSync with the distributed τ-nice sampling. Then $p_i = \frac{\mathbf{E}[\hat{S}]}{n} = \frac{\tau c}{sc} = \frac{\tau}{s}$ and hence the leading term in the complexity bound is

$$\Lambda = \max_i \frac{v_i}{p_i} \overset{(77)}{=} \max_i \frac{s \sum_{j=1}^m \lambda(C_j \cap \hat{S})}{\tau} \overset{(81)}{\leq} \max_i \frac{s \sum_{j=1}^m (\lambda_{1,j} + \lambda_{2,j}) A_{ji}^2}{\tau} \overset{\text{def}}{=} \Lambda'.$$

- Notice that the effect of partitioning on complexity comes only through $\lambda_{2,j}$.
- Define a new quantity that **does not depend on partitioning:**

$$\Lambda'' = \max_i \frac{s \sum_{j=1}^m \lambda_{1,j} A_{ji}^2}{\tau}$$

  and notice that (81) implies that

$$\Lambda'' \leq \Lambda' \leq (1 + \tfrac{1}{\tau - 1})\Lambda''$$

This means that:

Theorem 38 (Cost of Distribution: compare with [10, 13])

If $\tau \geq 2$, the worst-case partitioning is at most $(1 + \frac{1}{\tau})$ times worse than the optimal partitioning, in terms of the number of iterations of NSync.

Proof of Theorem 34 - Part I

Point 1. For simplicity of notation, put $P = P(\hat{S})$. If we choose $\theta \in \mathbb{R}^n$ with $\theta_i = (\mathrm{Tr}(P))^{-1/2}$ for all i, we get $\theta^T D^P \theta = \sum_i P_{ii}\theta_i^2 = 1$ and hence

$$\lambda(\hat{S}) \overset{(73)}{\geq} \theta^T P \theta \overset{(78)}{=} \mathbf{E}\left[\left(\sum_{i \in \hat{S}} \theta_i\right)^2\right] = \frac{\mathbf{E}\left[\left(\sum_{i \in \hat{S}} 1\right)^2\right]}{\mathrm{Tr}(P)} \overset{(43)}{=} \frac{\mathbf{E}[|\hat{S}|^2]}{\mathbf{E}[|\hat{S}|]}.$$

Point 2. Let us represent $\hat{S}$ as a convex combination of elementary samplings: $\hat{S} = \sum_{S \subseteq [n]} q_S \hat{E}_S$, where $q_S = \mathbf{P}(\hat{S} = S)$. Note that then we also have

$$P(\hat{S}) = \sum_{S \subseteq [n]} q_S P(\hat{E}_S) \overset{(73)}{=} \sum_{S \subseteq [n]} q_S e_S e_S^T. \tag{82}$$

Proof of Theorem 34 - Part II

Since $|\hat{S}| \leq \tau$ with probability 1, we have $|S| \leq \tau$ whenever $q_S > 0$. For any $\theta \in \mathbb{R}^n$ we can now estimate:

$$
\begin{aligned}
\theta^T P(\hat{S})\theta \overset{(82)}{=} \sum_{S: q_S > 0} q_S (e_S^T \theta)^2 \quad & \leq \quad \sum_{S: q_S > 0} q_S \|e_S\|^2 \sum_{i \in S} \theta_i^2 \\
& \overset{(75)}{=} \quad \sum_{S: q_S > 0} q_S |S| \sum_{i \in S} \theta_i^2 \\
& \leq \quad \tau \sum_{S: q_S > 0} q_S \theta^T \operatorname{Diag}(e_S e_S^T) \theta \\
& = \quad \tau \theta^T \left(\sum_{S: q_S > 0} q_S \operatorname{Diag}(e_S e_S^T) \right) \theta \\
& \overset{(82)}{=} \quad \tau \left(\theta^T \operatorname{Diag}(P(\hat{S}))\theta \right).
\end{aligned}
$$

We thus see that $\lambda(\hat{S}) \leq \tau$.

Proof of Theorem 34 - Part III

Point 3. The result follows by combining the upper and lower bounds. Alternatively, we can see this by inspecting the derivation in part 2. Indeed, if $|\hat{S}| = \tau$ with probability 1, then $|S| = \tau$ whenever $q_S > 0$, and hence the second inequality in point 2 above is an equality. By choosing $\theta_i = \alpha$ for any constant α, the first inequality turns into an equality (this is because we then have equality in the Cauchy-Schwartz inequality $e_S^T \theta \leq \|e_S\|^2 \sum_{i \in S} \theta_i^2$ for *all* S).

ESO($f \sim$ Model 3, $\hat{S} \sim \tau$-nice)

Theorem 39 ([12])

Let f satisfy assumptions in Model 3 and $\hat{S}$ be the τ-nice sampling. Then for all $x, h \in \mathbb{R}^N$,

$$\mathbf{E}\left[f(x + h_{[\hat{S}]})\right] \leq f(x) + \frac{\tau}{n}\left(\langle \nabla f(x), h\rangle + \frac{1}{2}\|h\|_v^2\right), \qquad (83)$$

where

$$v_i \stackrel{def}{=} \sum_{j=1}^{m} \beta_j L_{ji} = \sum_{j : i \in C_j} \beta_j L_{ji}, \qquad i = 1, 2, \dots, n, \qquad (84)$$

$$\beta_j \stackrel{def}{=} 1 + \frac{(\omega_j - 1)(\tau - 1)}{\max\{1, n-1\}}, \qquad j = 1, 2, \dots, m.$$

That is, $(f, \hat{S}) \sim ESO(v)$.

Proof of Theorem 39 - Part I

- We first claim that for all j,

$$\mathbf{E}\left[f_j(x+h_{[\hat{S}]})\right] \le f_j(x) + \frac{\tau}{n}\left(\langle \nabla f_j(x), h\rangle + \frac{\beta_j}{2}\|h\|^2_{L_{j:}}\right), \qquad (85)$$

  where $L_{j:} = (L_{j1}, \dots, L_{jn}) \in \mathbb{R}^n$. That is, $(f_j, \hat{S}) \sim ESO(\beta_j L_{j:})$. Equation (83) then follows by adding up the inequalities (85) for all j. In the rest we prove the claim.
- A well known consequence of (67) is that for all $x \in \mathbb{R}^N$, $t \in \mathbb{R}^{N_i}$,

$$f_j(x + U_i t) \le f_j(x) + \langle \nabla_i f_j(x), t\rangle + \frac{L_{ji}}{2}\|t\|^2_{(i)}. \qquad (86)$$

Proof of Theorem 39 - Part II

- We fix x and define

$$\hat{f}_j(h) \stackrel{\text{def}}{=} f_j(x+h) - f_j(x) - \langle \nabla f_j(x), h \rangle. \tag{87}$$

Since

$$\begin{aligned} \mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]})\right] &\stackrel{(87)}{=} \mathbf{E}\left[f_j(x+h_{[\hat{S}]}) - f_j(x) - \langle \nabla f_j(x), h_{[\hat{S}]} \rangle\right] \\ &\stackrel{(50)}{=} \mathbf{E}\left[f_j(x+h_{[\hat{S}]})\right] - f_j(x) - \tfrac{\tau}{n}\langle \nabla f_j(x), h \rangle, \end{aligned}$$

it now only remains to show that

$$\mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]})\right] \leq \tfrac{\tau \beta_j}{2n} \|h\|^2_{L_{j:}}. \tag{88}$$

- We now adopt the convention that expectation conditional on an event which happens with probability 0 is equal to 0. Let $\eta_j \stackrel{\text{def}}{=} |C_j \cap \hat{S}|$, and using this convention, we can write

$$\mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]})\right] = \sum_{k=0}^{n} \mathbf{P}(\eta_j = k)\mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]}) \mid \eta_j = k\right]. \tag{89}$$

Proof of Theorem 39 - Part III

- For any $k \geq 1$ for which $\mathbf{P}(\eta_j = k) > 0$, we now use use convexity of $\hat{f}_j$ to write

$$\begin{aligned}
\mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]}) \mid \eta_j = k\right] &= \mathbf{E}\left[\hat{f}_j\left(\frac{1}{k}\sum_{i\in C_j\cap\hat{S}} kU_i h^{(i)}\right) \mid \eta_j = k\right] \\
&\leq \mathbf{E}\left[\frac{1}{k}\sum_{i\in C_j\cap\hat{S}} \hat{f}_j\left(kU_i h^{(i)}\right) \mid \eta_j = k\right] \\
&\overset{(56)}{=} \frac{1}{\omega_j}\sum_{i\in C_j} \hat{f}_j\left(kU_i h^{(i)}\right) \\
&\overset{(86)+(87)}{\leq} \frac{1}{\omega_j}\sum_{i\in C_j} \frac{L_{ji}}{2}\|kh^{(i)}\|_{(i)}^2 = \frac{k^2}{2\omega_j}\|h\|_{L_{j:}}^2. \quad (90)
\end{aligned}$$

Proof of Theorem 39 - Part IV

- Finally,

$$\begin{aligned}
\mathbf{E}\left[\hat{f}_j(h_{[\hat{S}]})\right] &\overset{(89)+(90)}{\leq} \sum_k \mathbf{P}(\eta_j = k)\tfrac{k^2}{2\omega_j}\|h\|^2_{L_{j:}} \\
&= \tfrac{1}{2\omega_j}\|h\|^2_{L_{j:}}\mathbf{E}[|C_j \cap \hat{S}|^2] \\
&\overset{(55)}{=} \tfrac{\tau\beta_j}{2n}\|h\|^2_{L_{j:}},
\end{aligned}$$

and hence (88) is proved.

DSO($f \sim$ Model 3)

Corollary 40

Let f satisfy assumptions in Model 3 and $\hat{S}$ be a τ-nice sampling. Then for all $x, h \in \mathbb{R}^N$ we have

$$f(x+h) \leq f(x) + \langle \nabla f(x), h \rangle + \frac{\bar{\omega}\bar{L}}{2}\|h\|_w^2, \tag{91}$$

where

$$\bar{\omega} \overset{def}{=} \sum_j \omega_j \frac{\sum_i L_{ji}}{\sum_{k,i} L_{ki}}, \quad \bar{L} \overset{def}{=} \frac{\sum_{ji} L_{ji}}{n}, \quad w_i \overset{def}{=} \frac{n}{\sum_{j,i} \omega_j L_{ji}} \sum_j \omega_j L_{ji}. \tag{92}$$

Note that $\bar{\omega}$ is a data-weighted average of the values $\{\omega_j\}$ and that $\sum w_i = n$.

Proof.

This follows from Theorem 39 used with $\tau = n$ (notice that $\bar{\omega}\bar{L}w = v$). □

ESO and Lipschitz Continuity I

We will now study the collection of functions $\hat{\phi}_x : \mathbb{R}^N \to \mathbb{R}$ for $x \in \mathbb{R}^N$ defined by

$$\hat{\phi}_x(h) \stackrel{\text{def}}{=} \mathbf{E}\left[\phi(x + h_{[\hat{S}]})\right]. \tag{93}$$

Let us first establish some basic connections between ϕ and $\hat{\phi}_x$.

Lemma 41 ([9])

Let $\hat{S}$ be any sampling and $\phi : \mathbb{R}^N \to \mathbb{R}$ any function and $x \in \mathbb{R}^N$. Then

(i) *if ϕ is convex, so is $\hat{\phi}_x$,*

(ii) $\hat{\phi}_x(0) = \phi(x)$,

(iii) *If $\hat{S}$ is proper and uniform, and $\phi : \mathbb{R}^N \to \mathbb{R}$ is continuously differentiable, then*

$$\nabla \hat{\phi}_x(0) = \frac{\mathbf{E}[|\hat{S}|]}{n} \nabla \phi(x).$$

Proof of Lemma 41

Fix $x \in \mathbb{R}^N$. Notice that

$$\hat{\phi}_x(h) = \mathbf{E}[\phi(x + h_{[\hat{S}]})] = \sum_{S \subseteq [n]} \mathbf{P}(\hat{S} = S)\phi(x + U_S h),$$

where

$$U_S \overset{\text{def}}{=} \sum_{i \in S} U_i U_i^T.$$

As $\hat{\phi}_x$ is a convex combination of convex functions, it is convex, establishing (i). Property (ii) is trivial. Finally,

$$\nabla \hat{\phi}_x(0) = \mathbf{E}\left[\nabla\ \phi(x + h_{[\hat{S}]})\Big|_{h=0}\right] = \mathbf{E}\left[U_{\hat{S}} \nabla \phi(x)\right] = \mathbf{E}\left[U_{\hat{S}}\right] \nabla \phi(x) = \frac{\mathbf{E}[|\hat{S}|]}{n} \nabla \phi(x).$$

The last equality follows from the observation that $U_{\hat{S}}$ is an $N \times N$ binary diagonal matrix with ones in positions (v, v) for coordinates $v \in \{1, 2, \ldots, N\}$ belonging to blocks $i \in \hat{S}$ only, coupled with the fact that for uniform samplings, $p_i = \mathbf{E}[|\hat{S}|]/n$.

ESO and Lipschitz Continuity II

We now establish a connection between ESO and a uniform bound in x on the Lipschitz constants of the gradient "at the origin" of the functions $\{\hat{\phi}_x,\ x \in \mathbb{R}^N\}$.

Theorem 42

Let $\hat{S}$ be proper and uniform, and $\phi : \mathbb{R}^N \to \mathbb{R}$ be continuously differentiable. Then the following statements are equivalent:

(i) $(\phi, \hat{S}) \sim ESO(v)$,

(ii) $\hat{\phi}_x(h) \leq \hat{\phi}_x(0) + \langle \nabla \hat{\phi}_x(0), h \rangle + \frac{1}{2} \frac{\mathbf{E}[|\hat{S}|]}{n} \|h\|_v^2, \qquad x, h \in \mathbb{R}^N.$

Proof.

We only need to substitute (93) and Lemma 41(ii-iii) into inequality (ii) and compare the result with the definition of ESO (8). □

References I

[1] Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. Technical Report, 2008.

[2] Yurii Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341-362, 2012

[3] Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(2):1–38, 2014

[4] Peter Richtárik and Martin Takáč. Efficient serial and parallel coordinate descent methods for huge-scale truss topology design. *Operations Research Proceedings 2011*, pp. 27-32, 2012

[5] Peter Richtárik and Martin Takáč. Parallel coordinate descent methods for big data optimization. Mathematical Programming, 2015

[6] Martin Takáč, Avleen Bijral, Peter Richtárik and Nathan Srebro. Mini-batch primal and dual methods for SVMs. *ICML*, 2013

References II

[7] Rachael Tappenden, Peter Richtárik and Jacek Gondzio. Inexact coordinate descent: complexity and preconditioning. arXiv:1304.5530, 2013

[8] Rachael Tappenden, Peter Richtárik, Burak Büke. Separable approximations and decomposition methods for the augmented Lagrangian. to appear in *Optimization Methods and Software* 30(3):643–668, 2015

[9] Olivier Fercoq and Peter Richtárik. Smooth minimization of nonsmooth functions with parallel coordinate descent methods. arXiv:1309.5885, 2013

[10] Peter Richtárik and Martin Takáč. Distributed coordinate descent method for learning with big data. arXiv:1310.2059, 10/2013

[11] Peter Richtárik and Martin Takáč. On optimal probabilities in stochastic coordinate descent methods. *Optimization Letters*, 2015

[12] Olivier Fercoq and Peter Richtárik. Accelerated, parallel and proximal coordinate descent. *SIAM Journal on Optimization*, 2015

References III

[13] Olivier Fercoq, Zheng Qu, Peter Richtárik, Martin Takáč. Fast distributed coordinate descent for minimizing non-strongly convex losses. *IEEE International Workshop on Machine Learning for Signal Processing*, 2014

[14] Zheng Qu and Peter Richtárik. Coordinate descent with arbitrary sampling I: algorithms and complexity, arXiv:1412.8060, 2014

[15] Zheng Qu and Peter Richtárik. Coordinate descent with arbitrary sampling II: expected separable overapproximation, arXiv:1412.8063, 2014