

École polytechnique de Louvain

Stochastic gradient based methods for large-scale optimization: review, unification and new approaches

Author: **Guillaume VAN DESSEL**
Supervisors: **François GLINEUR, Sebastian STICH**
Reader: **Yurii NESTEROV**
Academic year 2018–2019
Master [120] in Data Sciences Engineering

Abstract

With the advent of massive data collection and machines everyday more powerful the interest put in large scale optimization problems is increasing exponentially fast. Typical machine learning tasks require to solve an optimization problem of a composite loss function for which the number of terms scales with the number n of observations. Since n can reach orders of magnitude up to 10^9 or even 10^{12} , traditional techniques based on full gradient information at each update step fail to remain time-efficient. Therefore arises the necessity of getting rid of those plain methods to build strategies that keep a light 'per update iteration' cost. This is the aim of stochastic gradient methods, the topic of this master thesis.

Here below the reader can find the main contributions of this master thesis.

- **Framework unification** in Chapter 4:

A formal definition of the different stochastic optimization eras is proposed.

The term "era" first appeared in *Katyusha: The First Direct Acceleration of Stochastic Gradient Methods* (Allen-Zhu, 2018) [Zhu18] to depict classes of optimization methods sharing a similar stochastic gradient structure. The most commonly used methods will be classified according to the definitions proposed. Furthermore an extensive analysis of each era's stochastic gradient structure is undertaken. New lemmas about sampling methods yielding unbiased stochastic gradients are devised.

- **Online behavior of the stochastic gradient variance** in Chapter 3,4,5:

The variance of the stochastic gradient actually refers to the expectation of the squared distance between the true gradient at a given iterate and its expectation. The impact of such quantity in the convergence of stochastic gradient algorithms has been shown for a while. Nevertheless not many papers provide experiments highlighting clearly its evolution during optimization processes: it is done in this master thesis for both first and second eras methods.

- **New approaches** in Chapter 5:

The main contribution of this master thesis is double.

1. Based on the work *Stochastic Optimization with Importance Sampling* (Zhao & Zhang, 2015)[ZZ15] a new importance sampling scheme is motivated, namely K -greedy sampling. This latter is built upon a quadratic upper-bound information on the objective function; e.g. smoothed-hinge loss, quadratic loss and logistic loss classification. Several tests are conducted to show its potential benefits.
2. Again using quadratic upper-bound information a new optimization method, namely $\tilde{K}$ Prox-SVRG, is shown. Inherently the concept of $\tilde{K}$ proximal mapping is defined. A detailed proof of convergence of $\tilde{K}$ Prox-SVRG, directly inspired from *A Proximal Stochastic Gradient Method with Progressive Variance Reduction* (Xiao & Zhang, 2014)(Prox-SVRG) [XZ14], is given. Experiments showed a real speed-up with respect to Prox-SVRG using the presented method, especially in the case of imbalanced features for machine learning classification tasks.

Keywords: convex stochastic optimization, variance reduction, importance sampling, quadratic upper-bound information, proximal gradient descent, $\tilde{K}$ proximal mapping

Contents

1	Framework	6
1.1	Empirical Risk Minimization (ERM)	6
2	Optimization Concepts	9
2.1	Definitions and reminders	9
2.1.1	Classical convexity and smoothness	9
2.1.2	Generalized convexity and smoothness	11
2.1.3	Proximal operator	11
2.1.4	Polyak- Lojasiewicz (PL) condition	14
2.2	Optimality conditions	15
2.3	Gradient mapping	16
2.4	Quality criteria	17
3	Motivation for Stochastic Gradient Methods	18
3.1	Proximal Gradient Descent Method (Prox-GD)	19
3.2	Proximal Stochastic Gradient Method (Prox-SGD)	20
3.3	Comparison	22
4	Evolution of Stochastic Gradient Methods	26
4.1	Pioneer Era	27
4.1.1	General Scheme	27
4.1.2	Analysis of the Stochastic Gradient	28
4.1.3	Methods	34
4.2	Variance Reduction Era	43
4.2.1	General Scheme	46
4.2.2	Analysis of the Stochastic Gradient	46
4.2.3	Methods	50
5	New approaches and paradigms for Stochastic Gradient Methods	57
5.1	Greedy-Sampling	58
5.2	$\tilde{K}$ -ProxVR	80
6	Index of state of the art Stochastic Gradient Methods	93

Acknowledgment

Firstly, I would like to thank my two supervisors, François and Sebastian, for the advice they gave me, their patience and the freedom they let me have during the whole master thesis.

This latter allowed me to deepen a research domain as I have never done before.

This experience provided me a sample of what being a researcher is all about.

A bunch of constraints but, before and after all, the possibility to let our own mind speaks for itself. There is room for creativity, challenging problem solving and ideas sharing with pairs... Actually, the master thesis gave growth to the will of pursuing the academic adventure with a Ph.D. for which the time constraint will be relaxed. I look forward for this journey.

Secondly, I would like to thank my family more generally for their support during my whole studies. They raised me and without them it is 100% certain that I wouldn't have been able to produce all the work I have done in order to get this final engineering degree.

Finally, I give a big thank to all my friends. They are for me an essential mental support. Therefore, they are also part of this accomplishment.

Introduction

As indicated in the abstract, more and more large datasets become available worldwide. For many reasons, the opportunity to apply machine learning techniques on those datasets is appealing. However one must be provided the right tools to tackle such a project.

One of these tools is numerical optimization and not of any kind: large-scale optimization. The so-called large label comes either from n (number of training samples) or/and d (data dimensionality). Two of the main families of optimization methods in the landscape of large-scale problems are, respectively, (i) the (stochastic) gradient descent based methods and (ii) the (randomized) primal coordinate descent/ dual coordinate ascent based methods. The keywords stochastic/randomized are put into brackets as for certain problems, purely deterministic methods are still competitive with respect to their stochastic/randomized equivalents.

This master thesis will focus on the family (i). The reader should note that in this work, by "stochastic optimization methods", it will be implicitly meant *stochastic gradient descent based methods*. However this former term depicts a broader concept.¹

Just as many research teams currently working in this domain, the goal is to come up with the optimization method that is the most time efficient in this "*big-data*" context. Even though there exist much more advanced techniques, two baseline methods in the world of stochastic gradient optimization are SGD [RM51] (initiated by the pioneer work of Robbins & Monro in 1951) and SVRG (Johnson & Zhang, 2013) [JZ13]. Implementations of those rather simple methods are available to the public which make of them "*references*" in the literature. Both these methods suffer from inherent lacks and the work produced by research teams often resume in tackling one or a few of these.

One major contribution of this master thesis is to propose a new kind of methods: $\tilde{K}$ proximal methods. These methods are constructed upon some quadratic upper-bound information that is often available for typical machine learning problems. Two instances of this class are namely $\tilde{K}$ Prox-SVRG and $\tilde{K}$ Prox-SARAH. The first one enjoys a rate of convergence similar to (Prox)-SVRG whereas the other remains heuristic for now. However tests show consistently better results against Prox-SVRG. Furthermore, an heuristic improved version of the proposed methods have shown promising results and will be motivated.

Another major contribution is the analysis of a new approach of importance sampling, namely K -greedy sampling. Based on the same quadratic upper-bound information, this latter is shown to be interesting and for a class of problems (mainly quadratic) it yields better theoretical convergence results than the "*reference*" paper IS-Prox-SGD (Zhao & Zhang, 2014) [ZZ15] in the specific domain of importance sampling for stochastic gradient descent.

¹See: Stochastic-Optimization.

A last contribution is the unification under *eras* (classes) of several well-known stochastic gradient optimization methods. Besides is presented an extensive analysis of the properties of each *eras*. Some lemmas defining conditions under which a stochastic gradient is an unbiased estimator of the true gradient are proven.

Here below the reader can find the logical continuation of the different chapters.

In Chapter 1 will be described the precise framework to which this master thesis belongs based on the principle of *empirical risk minimization* (ERM) [Bot98].

Chapter 2 will be the occasion to review optimization concepts useful for the next chapters.

In Chapter 3, stochastic gradient methods are motivated.

As a research type master thesis, an incumbent task resumes in a deep understanding of the topic at hand. Therefore in Chapter 4 comes the review and a huge analysis of two main classes of stochastic gradient methods.

In Chapter 5, in the light of the existing work carried out in the past, are presented the major contributions of this master thesis.

Finally, in Chapter 6, a short list of state of the art articles is displayed.

Chapter 1

Framework

Purpose

This first chapter aims to introduce the context in which the objective function structure (sum of individual loss functions) arose. Let's cite the following reference article: *Online Learning and Stochastic Approximations* (Bottou, 1998) [Bot98].

1.1 Empirical Risk Minimization (ERM)

Many optimization problems in the real world are formulated with an objective function that represents an expectation. For instance, in operational research one would like to minimize the expected risk of a decision, in machine learning one tries to train a regression model in order to minimize the expected prediction error, ... etc. In order to invoke a proper definition of such optimization problems one has to make a starting hypothesis: observable events related to the objective functions are assumed to be realizations of an underlying random vector Υ .

Note: For what follows we will stick to the case of a real random vector Υ .

That is we assume the existence of probability space $(\Omega, \mathcal{F}, \mathbf{P})$ and a measurable space $(E, \mathcal{E}) = (\mathbb{R}^s, \mathcal{B}(\mathbb{R}^s))$ where $\mathcal{B}(\mathbb{R}^s)$ is the Borel σ -algebra of $\mathbb{R}^s$ and the dimension s depends on the context of application. Then Υ is a function from Ω ('world of possible outcomes') to values in E that are real vectors of size s .

Hence such problems admit a cost function to be minimized of the form:

$$f_{\mathcal{T}} : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}, w \rightarrow f_{\mathcal{T}}(w) = \mathbb{E}_{\Upsilon}[\mathcal{L}(w, \Upsilon)]$$

Where $\mathcal{L}$ is the stochastic loss function parametrized by a decision vector $w \in D$. Given w , for each observation v from Υ , we can compute a cost $\mathcal{L}(w, v)$. In the integral form :

$$f_{\mathcal{T}}(w) = \int_{\Omega} \mathcal{L}(w, \Upsilon(\theta)) d\mathbf{P}(\theta)$$

Unfortunately, one rarely has access to the probability measure $\mathbf{P}$ that would define exactly the expectation function to minimize. This is the reason why most of the *expected risk minimization* problems are approximated by what we call *empirical risk minimization* problems that rely on a set $V = \{v_i \in \mathbb{R}^s | i \in [n]\}$ of collected realizations of Υ . In the absence of total knowledge about the distribution of Υ , the most common practice is to use the traditional Monte-Carlo estimator for $f_{\mathcal{T}}(w)$ as follows :

$$f_{\mathcal{T}}(w) \simeq f(w) := \frac{1}{n} \sum_{i=1}^n \mathcal{L}(w, v_i) = \frac{1}{n} \sum_{i=1}^n f_i(w)$$

defining sample losses by $f_i : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$, $f_i(w) = \mathcal{L}(w, v_i)$ for all $i \in [n]$.

One may observe that this expression is rather general; it plays a major role in *machine learning* when one hopes to get nice generalized performances on unknown data, that is getting a new *empirical cost*

$$\tilde{f}(w_T) = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} \mathcal{L}(w_T, \tilde{v}_i)$$

based on a new set of realizations of Υ , $\tilde{V} = \{\tilde{v}_i \in \mathbb{R}^s | i \in [\tilde{n}]\}$, as low as possible based on the w_T 'learned'. Let's distinguish w_T and w_* in order to remain clear for the rest of this document. Given V and $\mathcal{L}$ as mentioned above, the ERM problem is defined as follows :

$$\min_{w \in D} f(w) = \frac{1}{n} \sum_{i=1}^n \mathcal{L}(w, v_i) \quad (1.1)$$

Any w_* , when one exists at least, is immediately defined as

$$w_* := \arg \min_{w \in D} f(w)$$

Now let's imagine we are given a numerical algorithm '*Alg-ERM*($\cdot, \cdot$)' computing, once having been fed by V and $\mathcal{L}$, an admissible solution w_T that is either an ϵ -accurate¹ solution for (1.1) or simply the iterate got after T major steps of this algorithm. w_T is the output of such learning algorithm and in any case $f(w_*) \leq f(w_T)$. Nevertheless it may happen that $\tilde{f}(w_*) > \tilde{f}(w_T)$.

Often in the literature one decides to split f_i into two parts : a data related part ϕ_i and what we call a *regularization term* $R = \lambda r$ which will drive the optimization of f towards admissible solutions w that are not too complex ; e.g. embedding many zero-components for descriptive purposes. The *regularization term* implements the principle of *Occam's razor*². Indeed in classical *machine learning* most of the entries of v_i are scores with respect to explanatory *features* and the entries of w are weights associated those scores in a (non)-linear model. If a component of w equals 0, the model becomes '*simpler*' in the sense that it does not require the corresponding feature for predictions/classifications.

Example: In some cases, one might know some informations (yet, this is not a total knowledge about $\mathbf{P}$) about the conditional distribution of a part of the random vector Υ given the rest of this vector. For instance, for supervised machine learning tasks such as *classification*, the observations have a common structure $v_i = (x_i, y_i) \in \mathbb{R}^d \times \{\pm 1\}$ and it happens that a model of $\mathbb{P}[Y = y_i | X = x_i, w]$ is available. Under the i.i.d assumption for the observations, the problem of maximizing the (log-)likelihood of the observations can be reformulated as an empirical risk minimization problem where $\phi_i(w) = -\log(\mathbb{P}[Y = y_i | X = x_i, w])$ and $r = \alpha \|w\|_1 + \frac{1-\alpha}{2} \|w\|_G^2$.

For the well known logistic regression, $\phi_i(w) = \log(1 + \exp(-y_i x_i^T w))$, $\alpha \in [0, 1]$ and G is often taken as either I_d or, in a more statistically sustained approach $G = \text{diag}\{0, 1, \dots, 1\} \in \mathbb{R}^{d \times d}$ in order to not penalize an eventual intercept (generally, present at the first coordinate of w). The term $\alpha \|w\|_1$ tends to sparsify w_T . In highly intensive computational tasks³, this is generally more than welcome. However it's not the main topic of this master thesis.

¹The performance criterion is yet to be chosen. Please refer to Chapter 2 for a discussion.

²Philosophical state of mind : among multiple possible interpretations leading to equivalent satisfactory explanations, choose the simplest.

³For example, predictions/classifications once w_T is computed.

Classical Formulation

At the end of the day, let's summarize what is a general ERM problem:

$$\min_{w \in D} \left\{ f(w) := \frac{1}{n} \sum_{i=1}^n f_i(w) = \frac{1}{n} \sum_{i=1}^n [\phi_i(w) + \lambda r(w)] = \phi(w) + R(w) \right\} \quad (1.2)$$

Note: Generally, the problems presented in this master thesis are not constrained ($D = \mathbb{R}^d$). Nevertheless one can easily handle constrained optimization on a closed convex set D and even more through the use of a well suited proximal operator during update steps of w . This latter is presented in Chapter 2. Assumptions about the ϕ_i 's and r will be formulated after the section about definitions. However f will be assumed bounded from below and a w_* exists.

Chapter 2

Optimization Concepts

Purpose

This second chapter presents **existing** optimization concepts that will be helpful for the rest of this master thesis. Apart from punctual citations, the two main references for this chapter are: *Introductory Lectures on Convex Programming Volume: A Basic course, volume I* (Nesterov, 1998) [Nes98] and the already mentioned [ZZ15].

If not stated otherwise, any of the statement made below is proven by either the first and/or the second source cited above. Some proofs are shown in what concerns the proximal operator as this might represent non familiar content for the reader.

2.1 Definitions and reminders

The content below uses the notations $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$ referring to the standard euclidean inner-product and deduced norm on $\mathbb{R}^d$. However the vast majority of the results written below would hold if ever the inner-product was defined differently.

2.1.1 Classical convexity and smoothness

Let's begin by recalling the classical concepts of convexity and smoothness. Let's start with the concept of L -smoothness with respect to a norm $\|\cdot\|$.

Definition 1 (L -smoothness w.r.t a norm $\|\cdot\|$) : For $L \geq 0$, a function $\psi : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be L -smooth w.r.t $\|\cdot\|$ if and only if $\psi \in \mathcal{C}^1(D)$ and we have

$$\psi(u) \leq \psi(v) + \langle \nabla \psi(v), u - v \rangle + \frac{L}{2} \|u - v\|^2 \quad \forall u, v \in \mathbb{R}^d$$

When ψ is convex it's equivalent to require L -Lipschitz gradients :

$$\|\nabla \psi(v) - \nabla \psi(u)\| \leq L \|v - u\| \quad \forall u, v \in \mathbb{R}^d$$

Furthermore if $\psi \in \mathcal{C}^2(D)$, L -smoothness is equivalent to

$$\nabla^2 \psi(u) \preceq L I_d \quad \forall u \in D$$

Note: If a function ψ as depicted is L_l -smooth w.r.t $\|\cdot\|$ then it's also L_b -smooth w.r.t $\|\cdot\|$ for any L_b such that $0 \leq L_l \leq L_b$. It follows immediately from the definition of L -smoothness w.r.t. $\|\cdot\|$ and the fact that $(L_b - L_l)I_d \succ \mathbf{0}$.

Now it's time to recall the definition of (strong-)convexity with respect to a norm $\|\cdot\|$ for both differentiable and non-differentiable functions.

Definition 2 (σ -strong convexity w.r.t a norm $\|\cdot\|$) : For $\sigma \geq 0$, a function $\psi : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be σ -strongly convex w.r.t $\|\cdot\|$ on D convex if and only if we have $\forall s \in [0, 1]$,

$$\psi(su + (1-s)v) \leq s\psi(u) + (1-s)\psi(v) - \frac{\sigma s(1-s)}{2} \|u - v\|^2 \quad \forall u, v \in D$$

Furthermore, if additionally $\psi \in \mathcal{C}^1(D)$ then the following statement must hold

$$\psi(u) \geq \psi(v) + \langle \nabla \psi(v), u - v \rangle + \frac{\sigma}{2} \|u - v\|^2 \quad \forall u, v \in D$$

Finally, if additionally $\psi \in \mathcal{C}^2(D)$, the σ -strong convexity is equivalent to

$$\sigma I_d \preceq \nabla^2 \psi(u) \quad \forall u \in D$$

A generalization exists for convex but non-differentiable functions ψ :

$$\psi(u) \geq \psi(v) + \langle g(v), u - v \rangle + \frac{\sigma}{2} \|u - v\|^2$$

where $g(v)$ is any vector contained in the subdifferential set

$$\partial \psi(v) := \{g \in \mathbb{R}^d \mid \forall u \in D, \psi(u) \geq \psi(v) + \langle g, u - v \rangle\}$$

When ψ is differentiable, this set resumes in only one vector : the gradient of ψ at v , i.e. $\partial \psi(v) := \{\nabla \psi(v)\}$ and we fall back to the previous inequality.

Note 1: Let's assume that ψ is σ strongly convex with respect to a norm $\|\cdot\|$ and bounded by below. Then if $\sigma > 0$, $\mathbf{0} \in \partial \psi(w)$ is a sufficient condition for $w \in D$ to be an unique global minimizer of ψ on D . Indeed, $\forall u \in D, \psi(u) - \frac{\sigma}{2} \|u - w\|^2 \geq \psi(w) \Rightarrow \psi(u) > \psi(w), \forall u \in D \setminus \{w\}$. The unicity is lost if $\sigma = 0$.

Note 2: If a function ψ as depicted is σ_b -strongly convex w.r.t $\|\cdot\|$ then it's also σ_l -strongly convex w.r.t $\|\cdot\|$ for any σ_l such that $0 \leq \sigma_l \leq \sigma_b$. When the biggest $\sigma_b \geq 0$ such that ψ is σ_b -strongly convex w.r.t $\|\cdot\|$ is $\sigma_b = 0$ then the function is simply said convex.

Note 3: Strong convexity plays a major role, through what we call the *condition number* $\kappa := \frac{L}{\sigma}$, in the convergence rate of many numerical algorithms for optimization. Good news is that it's easy to tune the classical strong-convexity of a function ψ . Indeed for any function ψ σ_1 -strongly convex w.r.t $\|\cdot\|$, $\psi + \frac{\sigma_2}{2} \|\cdot\|^2$ is $(\sigma_1 + \sigma_2)$ -strongly convex w.r.t $\|\cdot\|$. Bad news is that the shape of the original function ψ has changed and maybe the minimizers of ψ are (absolutely) not anymore minimizers of $\psi + \frac{\sigma_2}{2} \|\cdot\|^2$.

Now let's introduce the most important concept in order to understand what will follow :

In the text of this master thesis, (strong)-convexity and smoothness are defined with respect to $\|\cdot\|$ as usual.

If the concepts of generalized convexity/smoothness with respect to a function (presented below) are used, it will be clearly highlighted.

2.1.2 Generalized convexity and smoothness

Let's write down a generalization of convexity and smoothness displayed in [ZZ15]. First the definition of the *Bregman divergence* of a differentiable convex function ψ is necessary.

Definition 3 (*Bregman's divergence*) : Be ψ a convex differentiable function. *Bregman's divergence* of ψ is the function $\mathcal{B}_\psi : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ defined $\forall u, v \in \mathbb{R}^d$ by

$$\mathcal{B}_\psi(u, v) = \psi(u) - \psi(v) - \langle \nabla \psi(v), u - v \rangle \geq 0$$

Now that the *Bregman divergence* is introduced, the following definitions can be read:

Definition 4 (μ -strong convexity w.r.t a convex differentiable function ψ) : For $\mu \geq 0$, and a convex differentiable function $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$. $\phi : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be μ -strongly convex w.r.t ψ if and only if we have $\forall s \in [0, 1]$,

$$\phi(su + (1-s)v) \leq s\phi(u) + (1-s)\phi(v) - \mu s(1-s)\mathcal{B}_\psi(u, v) \quad \forall u, v \in D$$

Furthermore, if additionally $\phi \in \mathcal{C}^1(D)$ then the following statement must hold

$$\phi(u) \geq \phi(v) + \langle \nabla \phi(v), u - v \rangle + \mu \mathcal{B}_\psi(u, v) \quad \forall u, v \in D$$

Again for a non differentiable function ϕ , let's mention this generalization $\forall g(v) \in \partial \phi(v)$:

$$\phi(u) \geq \phi(v) + \langle g(v), u - v \rangle + \mu \mathcal{B}_\psi(u, v)$$

Definition 5 (L -smoothness w.r.t a convex differentiable function ψ) : For $L \geq 0$ and a convex differentiable function $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$. $\phi : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be L -smooth w.r.t ψ if and only if $\phi \in \mathcal{C}^1(D)$ and we have

$$\phi(u) \leq \phi(v) + \langle \nabla \phi(v), u - v \rangle + L \mathcal{B}_\psi(u, v) \quad \forall u, v \in D$$

Note : The *Bregman divergence* won't be useful until Chapter 5. However let's already state the following result. Taking $\psi(\cdot) = \frac{1}{2} \|\cdot\|^2$ and thus $\nabla \psi(\cdot) = (\cdot)$ on $\mathbb{R}^d$ leads to

$$\mathcal{B}_\psi(u, v) = \frac{1}{2} \|u\|^2 - \frac{1}{2} \|v\|^2 - \langle v, u - v \rangle = \frac{1}{2} \|u - v\|^2$$

Therefore *Bregman divergence* based (strong)-convexity and smoothness generalize the classical definitions w.r.t a norm $\|\cdot\|$. There's equivalence for the special case where $\psi(\cdot) = \frac{1}{2} \|\cdot\|^2$.

2.1.3 Proximal operator

Now the concept of proximal operator is introduced. Apart from [ZZ15], the main reference for the following content is *First-order optimization methods* (Amir Beck, 2017) [Bec17a].

Definition 6 (proximal mapping of a function R) : The proximal mapping of a proper closed convex function $R : \mathbb{R}^d \rightarrow \bar{\mathbb{R}}$ at $u \in \mathbb{R}^d$ is defined as the unique

$$\text{prox}_R(u) := \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2} \|u - y\|^2 \right\}$$

Note: Let's remark that this concept generalizes the classical projection on a closed convex set. Indeed, let's consider $C \subseteq \mathbb{R}^d$ as a non-empty closed convex set and

$$\mathbb{1}_{\{C\}} : \mathbb{R}^d \rightarrow \{0, +\infty\}, y \rightarrow \mathbb{1}_{\{C\}}(y) = \begin{cases} 0 & y \in C \\ +\infty & y \notin C \end{cases}$$

Then computing $\text{prox}_{\mathbb{1}_{\{C\}}}(w) = \arg \min_{y \in C} \|w - y\|^2 = \arg \min_{y \in C} \|w - y\| = \text{proj}_C(w)$ shows that one possible use of the proximal operator is to implement a projection.

An interesting link can be made between a proximal mapping and the standard projected gradient method for a L -smooth function ϕ . The core idea behind the gradient descent update step is to optimize the *isotropic* quadratic upper-bound of ϕ at the current iterate w .

$$w_+ := \arg \min_{y \in \mathbb{R}^d} \left\{ \phi(w) + \langle \nabla \phi(w), y - w \rangle + \frac{L}{2} \|y - w\|^2 \right\} = w - \frac{1}{L} \nabla \phi(w)$$

Eventually, in the constrained framework, one takes the projection of such *isotropic* quadratic upper-bound minimizer

$$w_+ := \text{proj}_C(w - \frac{1}{L} \nabla \phi(w)) = \text{prox}_{\mathbb{1}_{\{C\}}}(w - \frac{1}{L} \nabla \phi(w))$$

Let's show that it's actually equivalent to directly minimize another function, the indicator-regularized isotropic quadratic upper-bound of ϕ . For any regularizer R with $w_+ := \text{prox}_R(w - \frac{1}{L} \nabla \phi(w))$:

$$\begin{aligned} w_+ &:= \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2} \|(w - \frac{1}{L} \nabla \phi(w)) - y\|^2 \right\} \\ &= \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2} (\|y\|^2 + \|w\|^2) - \langle w - \frac{1}{L} \nabla \phi(w), y \rangle + \frac{1}{2} \|\frac{1}{L} \nabla \phi(w)\|^2 - \langle w, \frac{1}{L} \nabla \phi(w) \rangle \right\} \\ &= \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2} (\|y\|^2 - 2\langle y, w \rangle + \|w\|^2) + \frac{1}{L} \langle \nabla \phi(w), y - w \rangle \right\} \\ &= \arg \min_{y \in \mathbb{R}^d} \left\{ L R(y) + \phi(w) + \langle \nabla \phi(w), y - w \rangle + \frac{L}{2} \|y - w\|^2 \right\} \end{aligned}$$

Immediately one can observe that the particular choice $R(\cdot) = 0$ for every $y \in \mathbb{R}^d$ leads to a proximal operator degenerating into an identity operator.

$$\text{prox}_0(w) = w$$

Let's now state a few important results about proximal operators that will be helpful to derive first-order optimality conditions at the next subsection and understand the true motivation behind proximal methods.

Property 1 (image of a proximal mapping) : Let $R : \mathbb{R}^d \rightarrow \bar{\mathbb{R}}$ be a proper closed convex function then the following equivalence holds $\forall u \in \mathbb{R}^d$

$$v = \text{prox}_R(u) \Leftrightarrow u - v \in \partial R(v)$$

Proof: Given the assumptions on R , the *effective domain* of $R(\cdot) + \frac{1}{2} \|\cdot - u\|^2$ is $D = \mathbb{R}^d$. This latter function is 1-strongly convex and sufficient conditions of optimality for such class of functions require that the unique minimizer y_* is such that

$$0 \in \partial(R(y_*) + \frac{1}{2} \|y_* - u\|^2)$$

Hence $\mathbf{0} \in \partial R(y_*) + \{\nabla_y(\frac{1}{2}\|y - u\|^2)|_{y=y_*}\}$ where '+' denotes *Minkowski's addition*. This resumes in imposing that $u - y_* \in \partial R(y_*)$. By definition of proximal operator, $v = y_*$.

Conversely we can backtrack ; if $u - v \in \partial R(v)$ then $\mathbf{0} \in \partial(R(v) + \frac{1}{2}\|v - u\|^2)$. As the interior of D is $\mathbb{R}^d$, the optimality condition was also necessary and thus

$$v = \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2}\|y - u\|^2 \right\} = \text{prox}_R(u)$$

□

Property 2 (firm/simple non-expansivity of proximal operators) : Let $R : \mathbb{R}^d \rightarrow \bar{\mathbb{R}}$ be a proper closed convex function then the following inequalities hold $\forall u, v \in \mathbb{R}^d$

$$\|\text{prox}_R(u) - \text{prox}_R(v)\|^2 \leq \langle \text{prox}_R(u) - \text{prox}_R(v), u - v \rangle \quad (\text{firm})$$

$$\|\text{prox}_R(u) - \text{prox}_R(v)\| \leq \|u - v\| \quad (\text{simple})$$

Proof¹: Let $u' := \text{prox}_R(u)$ and $v' := \text{prox}_R(v)$. Recalling **Property 1**, $u - u' \in \partial R(u')$ and $v - v' \in \partial R(v')$. Then, by definition of subdifferential set :

$$\begin{aligned} R(v') &\geq R(u') + \langle u - u', v' - u' \rangle \\ R(u') &\geq R(v') + \langle v - v', u' - v' \rangle \end{aligned}$$

Summing both inequalities leads to :

$$0 \geq \langle v - v' - u + u', u' - v' \rangle$$

$$\Leftrightarrow$$

$$\langle u' - v', u' - v' \rangle = \|\text{prox}_R(u) - \text{prox}_R(v)\|^2 \leq \langle u - v, u' - v' \rangle = \langle \text{prox}_R(u) - \text{prox}_R(v), u - v \rangle$$

Finally, using *Cauchy-Schwarz* inequality and the last inequality it immediately comes that

$$\|\text{prox}_R(u) - \text{prox}_R(v)\| \leq \|u - v\|$$

□

Note: This means that the proximal operator applied on two points u, v will never make them more distant than what they were initially. Furthermore let's imagine that for the R chosen there exists v_* such that $v_* = \text{prox}_R(v_*)$. Since R is convex, v_* would be a minimizer of R as $v_* - v_* = \mathbf{0} \in \partial R(v_*)$. When R is a proper closed convex function this is always the case. Applying $\text{prox}_R(\cdot)$ on any other point u of the domain leads to $\|\text{prox}_R(u) - \text{prox}_R(v_*)\| = \|\text{prox}_R(u) - v_*\| = \|u' - v_*\| \leq \|u - v_*\|$. We can see that the general purpose of the proximal operator is to drive, smoothly if R is smooth, its argument towards a region of lower R -penalty.

¹Source: **Theorem 6.42** of [Bec17a].

The (PL) condition stated below won't play a major role in this master thesis.

The content introduced here is taken from *Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Lojasiewicz Condition* (Karimi et al., 2018) [KNS18].

It stands in order to show the reader a practical condition under which the convergence towards a point where the norm of a subgradient is arbitrary small implies the convergence towards a point whose cost is arbitrary close from f^* (see **section 2.2** below).

2.1.4 Polyak-Lojasiewicz (PL) condition

Definition 7 (Polyak-Lojasiewicz (PL) condition) : Let $f : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ a differentiable function. f is said to fulfill the Polyak-Lojasiewicz (PL) condition with constant μ if and only if f admits a minimizer on D for which we define the optimal cost f^* and there exists $\mu > 0$ such that

$$\mu(f(w) - f^*) \leq \frac{1}{2} \|\nabla f(w)\|^2 \quad \forall w \in D$$

Note 1: Immediately one can see that any stationary point of a function f verifying the PL condition is a global minimizer of f on its domain ! However PL-condition **does not** imply convexity that enjoys this same feature. For instance one could check that the function $f : \mathbb{R} \rightarrow \mathbb{R}, w \rightarrow f(w) = w^2 + 3 \sin(w)$ fulfills PL-condition with constant $\mu = \frac{1}{4}$. Indeed, the inequality $\frac{1}{2}(2w + 3 \cos(w))^2 - \frac{1}{4}(w^2 + 3 \sin(w) + 1) \geq 0$ is verified for any $w \in \mathbb{R}$ and $f^* \geq -1$. However f is obviously not convex hence not strongly convex as well. PL actually implies what one calls invexity² but this concept won't be deepened in this present document.

Note 2: In the literature³ sometimes PL-condition is replaced by τ -gradient domination condition. This condition is resumed by requiring that f is differentiable with an existing minimizer on its domain and that $\forall w \in D, (f(w) - f^*) \leq \tau \|\nabla f(w)\|^2$. Actually a function f that fulfills PL-condition with parameter μ is equivalently $\tau = \frac{1}{2\mu}$ -gradient dominated.

Let's show that a well known class of functions, e.g. the μ -strongly convex functions ($\mu > 0$) is contained in the class of μ -PL functions or equivalently $\frac{1}{2\mu}$ -gradient dominated functions.

Property 3 (strong convexity $\Rightarrow$ PL-condition) : Let $f : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ be a differentiable μ -strongly convex function ($\mu > 0$) then f fulfills the PL-condition with parameter μ .

Proof : Since f is μ -strongly convex ($\mu > 0$) **and** differentiable f admits a minimizer w_* on its domain. Following strong-convexity definition for differentiable functions,

$$f(w_*) \geq f(w) + \langle \nabla f(w), w_* - w \rangle + \frac{\mu}{2} \|w_* - w\|^2 \quad \forall w \in \text{dom}(f)$$

Minimizing the *isotropic* quadratic lower-bound of f (RHS) with respect to w and noticing that it's also a μ -strongly convex function leads to the necessary and sufficient condition that $-\nabla f(w) - \mu(w_* - w) = \mathbf{0} \Leftrightarrow w = \frac{\nabla f(w)}{\mu} + w_*$. Hence taking back the expression of the RHS and injecting the expression of a w minimizing the *isotropic* quadratic lower-bound gives $f(w_*) \geq f(w) + \langle \nabla f(w), w_* - (w_* + \frac{\nabla f(w)}{\mu}) \rangle + \frac{\mu}{2} \|w_* - ((w_* + \frac{\nabla f(w)}{\mu}))\|^2 = f(w) - \frac{\|\nabla f(w)\|^2}{2\mu}$.

□

²Source: [KNS18].

³Source: [Ngu+17].

2.2 Optimality conditions

Let's consider problem (1.2). After **Definition 2**, in **Note 1**, a comment recalls first-order optimality conditions applied to a convex (non-necessarily differentiable everywhere on its domain D) objective function. More generally for functions f that are either convex or non-convex but differentiable we define *stationary* points u as elements of $S^* := \{u \in \mathbb{D} \subseteq \mathbb{R}^d | \mathbf{0} \in \partial f(u)\}$.

At any such point, one faces either a local minimum/maximum or a saddle point of f .

While solving (1.2) one would like to approach at least a local minimum.

Recall that by assumption f is bounded from below so at least one w_* exists.

Multiple cases arise :

1. $D = \mathbb{R}^d$: as there's no border to the feasible domain any minimizer w_* of (1.2) is in S^* .

If f is convex then any element of S^* is a minimizer of (1.2). If f is non-convex and if one has access to the whole set S^* one may simply evaluate f at every point of S^* to detect the global minimizers of (1.2). Otherwise, if f is non-convex and if one has found a single element of S^* , this latter may eventually be a local minimum of f but then are required "*stronger results such as approximate second-order optimality or strictly local minimum require additional assumptions and more sophisticated optimization methods such as cubic regularized Newton-type schemes*"⁴.

2. $D \subset \mathbb{R}^d$: if f is convex, any minimizer w_* fulfills the condition $\forall w \in D, \forall \nu \in \partial f(w_*)$

$$\langle \nu, w - w_* \rangle \geq 0$$

Indeed, since $\forall \nu \in \partial f(w_*)$ and $\forall w \in D$, $f(w) \geq f(w_*) + \langle \nu, w - w_* \rangle$ one would then have $f(w) \geq f(w_*)$ for any $w \in D$ ensuring that w_* is well a minimizer of f on D .

If f is differentiable, any point $u \in S^*$ is such that the only $\nu \in \partial f(u)$ is $\mathbf{0}$ for which the condition above is fulfilled. The only difference with $D = \mathbb{R}^d$ is that, in this same framework, there might be points not in S^* for which the condition holds as well !

This time if f is non-convex, even with the access to the whole set S^* one is not sure to find a global minimum. A proper inspection of the behaviour of f on the border of D is necessary... Again *sophisticated tools* exist in order to spot whether a single element of S^* is a local minimum or not.

To resume, for both convex and non-convex settings there's an interest in looking after *stationary* points for solving (1.2). The point is that the convex framework provides many more guarantees. In machine learning both types of settings appear and will be investigated in the experiments. However most of the theoretical analysis will cover the convex setting.

When no counter indication is given f will be decomposed in a smooth (strongly)-convex part $\phi(\cdot) = \frac{1}{n} \sum_{i=1}^n \phi_i(\cdot)$ and R will be simply convex but not necessarily smooth nor differentiable.

Thus, when searching after a point $u \in S^*$, we will generically write :

$$\mathbf{0} \in \partial f(u) = \nabla \phi(u) + \partial R(u) \Leftrightarrow -\nabla \phi(u) \in \partial R(u) \quad (2.1)$$

Trivially if R is a constant function, this condition is equivalent to look after points for which

$$\nabla \phi(u) = \mathbf{0}$$

⁴This comment is in the **Optimality Conditions** section of [Pha+19]

Without loss of generality we will further assume that $D = \mathbb{R}^d$. If one wishes to optimize on a smaller non-empty closed convex set D , R will embed $\mathbb{1}_D$, the indicator function of D .

Before digging into the core content of this master thesis about stochastic-gradient based methods first will be presented the traditional plain-gradient method used in order to approach "as close as possible" a minimizer w_* for problem (1.2). Again we will look after such minimizers in the set S^* , assuming it's not empty! As a reminder, any point of S^* is a minimizer when the function f is convex ! As it's hopeless generally to require an exact point from S^* , one speaks about ϵ -accurate admissible solution w_T . Two criteria are widely used to define ϵ -accuracy. However in this master thesis it is the first one that is investigated. Hereafter the reader will find both criteria but first one needs an explanation about the concept of *gradient mapping* [Nes98] and its implication in the assessment of algorithm's convergence.

2.3 Gradient mapping

Definition 8 (Gradient mapping of a function f) : Be $f : D \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ a function defined as the sum of two sub-functions $\phi : D \rightarrow \mathbb{R}$ differentiable and $R : D \rightarrow \mathbb{R}$ convex, i.e. $\forall w \in D, f(w) = \phi(w) + R(w)$. For any $\eta > 0$ and $w \in D$, the *gradient mapping* of f at w is

$$G_\eta(w) := \frac{1}{\eta} \left[w - \text{prox}_{\eta R}(w - \eta \nabla \phi(w)) \right] \quad (2.2)$$

where D is defined in such a way that the proximal operator for R is well defined.

Note : Trivially, $G_\eta(u) = \mathbf{0}$ if and only if $u = \text{prox}_{\eta R}(u - \eta \nabla \phi(u))$. Let's consider such an u . Recalling Property 1, $u - \eta \nabla \phi(u) - u = -\eta \nabla \phi(u) \in \eta \partial R(u)$ or equivalently,

$$-\nabla \phi(u) \in \partial R(u)$$

In other words, u would be contained in S^* looking at condition (2.1). One can easily establish a link with traditional *zero-gradient* requirement for the case where R is a constant. Indeed in that case the minimization of f is unconstrained and $\text{prox}_{\eta R}(u - \eta \nabla \phi(u)) = u - \eta \nabla \phi(u)$,

$$G_\eta(u) = \nabla \phi(u) = \mathbf{0}$$

Thus, there's an equivalence between : $\forall u \in S^* \Leftrightarrow$ for any $\eta > 0, G^* := G_\eta(u) = \mathbf{0}$.

Algorithmic scheme

Now let's propose an intuitive iterative scheme that will produce points $w \in S^*$. Actually w is a fixed point for an update step of the following type. Be $\eta > 0$,

$$w \leftarrow w - \eta G_\eta(w) \quad (2.3)$$

Under some hypothesis that will be covered soon, this iterative scheme converges within an arbitrary precision imposing that $\|G_\eta(w)\| \rightarrow 0$.

2.4 Quality criteria

Be $\epsilon > 0$ the accuracy desired. Let's denote by f^* the value taken by f at any minimizer w_* . Furthermore G^* represents the gradient mapping at any $w_* \in S^*$ which's equal to $\mathbf{0}$ as previously shown. We will focus on the two following forms of ϵ -accurate solution :

1. $\mathbb{E}[f(w_T) - f^* | w_0] \leq \epsilon$: expected slack between the value of f at the output iterate w_T and the optimal value.
2. $\mathbb{E}[\|G_{\eta_T}(w_T) - G^*\|^2 | w_0] \leq \epsilon$: expected norm of difference between the *gradient mapping* of f at the output iterate w_T and a minimizer w_* of S^* .

where w_0 is a starting point, an initial guess for the solvers fed into the algorithms.

Note that (2) may be rewritten as $\mathbb{E}[\|G_{\eta_T}(w_T)\|^2 | w_0] \leq \epsilon$ since $G^* = \mathbf{0}$ assuming $S^* \neq \emptyset$.

Note: When using plain methods as in Proximal Gradient Descent (Prox-GD) there's no other form of randomness in the optimization process and the expectation operator is useless. However the criteria are stated as above in order to pave the way through stochastic-gradient based methods for which convergence results are expressed in terms of overall expectation !

The first criterion is intrinsically the most important since the ultimate goal is to solve (1.2). Disposing of an iterate for which $f(w_T)$ is the lowest possible is fine. Unfortunately unless knowing the value f^* and looking after points in S^* for which this optimal cost is met we cannot stop a numerical algorithm based on ϵ -accuracy for criterion 1.

Nevertheless under τ -gradient domination it's easy to get that for functions f with a constant part R , leading to the equality $G_\eta(w) = \nabla\phi(w) = \nabla f(w)$ for any $\eta > 0$ in order to obtain ϵ -accuracy with respect to criterion 1 it's sufficient to require an $\frac{\epsilon}{\tau}$ accuracy with respect to criterion 2. Let's consider the case where R is convex but ϕ is L -smooth. In [KNS18] it's shown that a generalized version of the PL-condition based on

$$\mathcal{D}_R(w, \frac{1}{\eta}) = -\frac{2}{\eta} \min_{y \in \mathbb{R}^d} \left\{ \langle \nabla\phi(w), y - w \rangle + \frac{1}{2\eta} \|y - w\|^2 + R(y) - R(w) \right\}$$

that is requiring

$$\mu(f(w) - f^*) \leq \frac{1}{2} \mathcal{D}_R(w, \frac{1}{\eta}) \quad (2.4)$$

for every $w \in D$ is sufficient to obtain ϵ -convergence with respect to criterion 1 while applying algorithmic scheme (2.3) with $0 < \eta \leq \frac{1}{L}$. The reader may convince himself that the expression of $\mathcal{D}_R(w, \frac{1}{\eta})$ is closely related to $\|G_\eta(w)\|^2$. One may verify that whenever R is constant, $\mathcal{D}_R(w, \frac{1}{\eta})$ is identical to $\|G_\eta(w)\|^2 = \|\nabla f(w)\|^2 = \|\nabla\phi(w)\|^2$. It actually follows from paper's result that if one finds $\tilde{w}_T \in D$ such that $\mathcal{D}_R(\tilde{w}_T, \frac{1}{\eta}) \leq \frac{2\mu}{1-\eta\mu}\epsilon$ then the w_T produced at the next iteration verifies $f(w_T) - f^* \leq \epsilon$.

Criterion 2 is always *tractable* under the assumption of a first-order oracle at disposal and a proximal operator leading to a unique vector in $\mathbb{R}^d$, i.e. we can always compute the *gradient mapping* time by time to assess the convergence of the numerical algorithm used.

We may want to add a non-differentiable part to ϕ (hence making it non-smooth) then we require $\mathbf{0} \in \partial\phi(u) + \partial R(u)$. Therefore all the $\nabla\phi(w)$ represent one subgradient of ϕ at w . Again by default, ϕ is at least smooth.

Chapter 3

Motivation for Stochastic Gradient Methods

Purpose

In this chapter is motivated the use of stochastic optimization methods in order to tackle the problem (1.2). A fair comparison between the basic (proximal) stochastic gradient descent method and its full-gradient counterpart, namely (proximal) gradient descent method, is operated on a problem for which n is big enough to reveal the power of stochastic gradient methods.

The performances of the algorithms studied will be compared, depending on the context, with respect to two measures:

1. Given an ϵ -accuracy for criterion 2 and/or 1, the first measure corresponds to the number of sample gradient computations in order to get a final valid output w_T .
2. Fixing the number of sample gradient computations, the second measure is the goodness of the output w_T produced by the analyzed algorithms.

Note 1: A single sample gradient computation represents the computation at any iterate $u \in \mathbb{R}^d$ given any $i \in [n]$ of the vector $\nabla\phi_i(u)$. For instance, $\nabla\phi(u)$ requires exactly n sample gradient computations since ϕ is a compound function made of a sum of n ϕ_i , $i \in [n]$.

Note 2: w_T will always depict the output iterate of any algorithm covered in the present work, whether it represents the last iterate produced or an average of past iterates or any other combination of previous iterates...

Why sample gradient computations?

As n grows, the computational time taken by numerical optimization algorithms to compute a full-gradient of ϕ becomes the critical time consuming step at each iterate update.

Such operation scales with n and the more sophisticated the loss function is (e.g. neural networks) the more time a single sample gradient computation requires. For many practical applications (e.g. logistic loss regression, quadratic loss regression, ...) the complexity of a full-gradient computation is represented by $\mathcal{O}(nd)$. Therefore even though they still need the least number of update iterations of w in order to produce w_T starting from w_0 , classical full-gradient based algorithms present, by definition, the major bottleneck of an update iteration that costs not less than n single sample gradient computations. The key point used in stochastic gradient methods is to proceed to much more update iterations but at a *light per-iteration cost*, much less than $\mathcal{O}(n)$ sample gradients computations

3.1 Proximal Gradient Descent Method (Prox-GD)

The motivation behind the proximal gradient descent method simply generalizes the idea behind classical gradient descent. Let's derive it and highlight the fact that whenever f can be interpreted as being only a differentiable part $f = \phi$ then Prox-GD is equivalent to GD.

One assumes to dispose of an upper-bound of f at each iterate w and chooses the next iterate w_+ as being the minimizer of such upper-bound. In this deterministic framework, under certain conditions we can tune the upper-bound in such a way that $f(w_+)$ is ensured to be lower than $f(w)$ which ultimately leads to convergence as f is supposed to admit a minimizer.

Typically, one makes the assumption to have access to $\eta > 0$ such that $\forall w_+, w \in \mathbb{R}^d$ to

$$f(w_+) = \phi(w_+) + R(w_+) \leq \phi(w) + \langle \nabla \phi(w), w_+ - w \rangle + \frac{1}{2\eta} \|w_+ - w\|^2 + R(w_+)$$

Such common assumption is trivially verified under L -smoothness of ϕ just by choosing $\eta = \frac{1}{L}$. It might happen that one does not have access to L , even though it exists. Fortunately it can be estimated and updated iteratively. If $R = 0$, minimizing the RHS leads to choose $w_+ = w - \eta \nabla \phi(w)$ as for GD. More generally, looking back to Definition 6, we actually set-up

$$w_+ = \text{prox}_{\eta R}(w - \eta \nabla \phi(w))$$

Note: In the case $R = 0$, the gradient method may converge even if ϕ is not L -smooth. One then picks up a convenient η for a line-search in the direction of the gradient of ϕ (e.g. fulfilling Wolfe's conditions¹) but this is outside the main topic of this master thesis.

Algorithm 1 Prox-GD

Initialization : $\eta > 0$, w_0 starting iterate, T number of update steps,
for $t = 0 \dots T - 1$ **do**
 $w_{t+1} \leftarrow \text{prox}_{\eta R}(w_t - \eta \nabla \phi(w_t))$
end for
Output : w_T

Main convergence results

Please refer to the bibliography for the proofs. In any case ϕ is a proper, closed differentiable function and R is a proper, closed convex function both defined $\mathbb{R}^d$.

further assumptions on ϕ and/or R	result	comment	reference
ϕ L -smooth	$\min_{t \in \{0, \dots, T-1\}} \ G_\eta(w_t)\ \leq \sqrt{\frac{2(f(w_0) - f^*)}{\eta T}}$	$0 < \eta \leq \frac{2}{L}$	[Bec17b]
ϕ L -smooth and μ -generalized PL	$f(w_T) - f^* \leq (1 - \mu\eta)^T (f(w_0) - f^*)$	$0 < \eta \leq \frac{1}{L}$	[KNS18]
ϕ L -smooth and convex	$f(w_T) - f^* \leq \frac{\ w_0 - w^*\ ^2}{2\eta T}$	$\eta = \frac{1}{L}$	[Sta]
ϕ L -smooth and μ -strongly convex	$f(w_T) - f^* \leq \frac{1}{2\eta} (1 - \eta\mu)^T \ w_0 - w^*\ ^2$	$\eta = \frac{1}{L}$	[Bec17b]

Note: For every setting under L -smoothness $f(w) - f(w_+) \geq \eta(1 - \frac{\eta L}{2}) \|G_\eta(w)\|^2$ [Sta].

Results are presented here after T update iterations or Tn sample gradients computations.

¹Source : *Convergence Conditions for Ascent Methods. II: Some Corrections* (Wolfe, 1971)

3.2 Proximal Stochastic Gradient Method (Prox-SGD)

Now let's introduce Prox-SGD in its most primitive version. As argued multiple times in the present document computing a full-gradient at each update iteration is prohibitive when n becomes large. Instead the very first versions of Prox-SGD² aims to use an estimation of that full gradient that only costs a single sample gradient computation. Therefore as discussed earlier, for most of the practical applications, one iterate update would costs $\mathcal{O}(d)$ basic operations.

Imagine we dispose of an iterate $w \in \mathbb{R}^d$. The core idea behind Prox-SGD is to draw at random a sample $i \sim \mathbf{U}([n])$ and follow the direction the linked sample gradient $\nabla\phi_i'(w)$.

Obviously, if one defines p_i by $p_i := \mathbb{P}[i' = i]$ (here $p_i = \frac{1}{n}$ given the uniform sampling),

$$\mathbb{E}_{i'}[\nabla\phi_{i'}(w)] = \sum_{i=1}^n p_i \nabla\phi_i(w) = \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(w) = \nabla\phi(w)$$

Hence, by taking $\nabla\phi_{i'}$ instead of $\nabla\phi$, in average one follows the direction of the true gradient. The update step of Prox-SGD is then pretty simple. It's essentially the same as for Prox-GD. Yet there is one fundamental difference : one uses what we will denote by the term *a stochastic gradient*, g , instead of the full-gradient $\nabla\phi$.

More precisely, the update step works as follows given w and step-size η ,

$$\begin{aligned} i' &\sim \mathbf{U}([n]) \\ g &= \nabla\phi_{i'}(w) \\ w_+ &= \text{prox}_{\eta R}(w - \eta g) \end{aligned}$$

w_+ is computed as being the minimizer of a stochastic upper-bound that holds '*in average*'. Therefore in general we can't ensure any decrease of f with probability 1 as for Prox-GD.

Actually for the iterate update step of Prox-SGD, w_+ minimizes a single realization of a random function

$$\bar{f}_{i',w} : \mathbb{R}^d \rightarrow \mathbb{R}, u \rightarrow \bar{f}_{i',w}(u) = \phi(w) + \langle \nabla\phi_{i'}(w), u - w \rangle + \frac{L}{2} \|u - w\|^2 + R(w_+)$$

nevertheless it doesn't mean that $f(w_+)$ will be smaller than $f(w)$.

Note 1: Despite the fact that for any iterate $w \in \mathbb{R}^d$, g is an unbiased estimate of the full-gradient, the convergence results of Prox-SGD are weaker than a version of the convergence results for Prox-GD holding only in expectation. The main reason explaining this downfall is the fact that the expected decrease at given steps depend on the whole history of randomness. To provide the reader with some intuition at this point here is an argument. With stochastic gradient based methods some update iterations might be terrible. These bad update steps will condition the optimization process at least for a few more updates...

Note 2: In the original Prox-SGD algorithm the distribution of i' is always uniform on $[n]$. Nevertheless, and this is one aspect covered in this master thesis, other sampling schemes can be investigated. Importance sampling arises in Chapter 4 and Chapter 5.

²Actually the proximal setting (Prox) was proposed years after (Robbins and Monro, 1951) but doesn't change the core idea of SGD which's to estimate the gradient of ϕ . R doesn't play any role in the scalability of the studied algorithms.

Algorithm 2 Prox-SGD

Initialization : $\{\eta_t\}_{t=0}^{T-1}$ step-size schedule $\forall t \in \{0, \dots, T-1\}$, $\eta_t > 0$, w_0 starting iterate, T number of update steps
for $t = 0 \dots T-1$ **do**
 $i_t \sim \mathbf{U}([n])$
 $g_t \leftarrow \nabla \phi_{i_t}(w_t)$
 $w_{t+1} \leftarrow \text{prox}_{\eta_t R}(w_t - \eta_t g_t)$
end for
Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note: Obviously in practice g_t is not especially stored in memory. The scheme of Prox-SGD is intentionally left as explicit in order to proceed to clear comparisons in Chapter 4 !

Every convergence rate for (Prox)-SGD relies on a decreasing step-size scheme $\{\eta_t\}_{t=0}^{T-1}$ and a further assumption of bounded stochastic gradient *variance* is required.

Definition 9 (direct bounded stochastic gradient norm variance) : Dealing with problem (1.2), ϕ is said to have *bounded-variance* if and only if there exists $\sigma^2 \geq 0$ such that

$$\mathbb{E}_{i'}[||\nabla \phi_{i'}(w) - \nabla \phi(w)||^2] \leq \sigma^2 \quad \forall w \in \mathbb{R}^d \quad (3.1)$$

Note : More sophisticated versions of this assumption will be introduced when analyzing the convergence bounds of stochastic gradient based methods for which the stochastic gradient used is not 'simply' $g = \nabla \phi_{i'}(w)$. However fulfilling the above assumption will often be sufficient. Noticing the unbiasedness of g as estimator of the true gradient, $\mathbb{E}_{i'}[||\nabla \phi_{i'}(w) - \nabla \phi(w)||^2] = \mathbb{E}_{i'}[||\nabla \phi_{i'}(w)||^2] - ||\nabla \phi(w)||^2$. A stronger condition called *bounded second order moment* than the above *bounded-variance* is to require that $\mathbb{E}_{i'}[||\nabla \phi_{i'}(w)||^2] \leq C^2$ for any $w \in \mathbb{R}^d$ and $C^2 \geq 0$. Actually one needs that those assumptions hold at every iterate w encountered during the optimization process, this is usually true.

As mentioned in the following table when using a constant step-size η one can only ensure expected convergence until a lower-bound accuracy. Imagine $R = 0$, this may seem rather intuitive since when approaching a minimizer w_* of f nothing guarantees us that sample gradients $\nabla \phi_i(\cdot)$ will all converge to *zero-norm* vectors despite the fact that it holds that $\nabla \phi(\cdot)$ will converge to $\mathbf{0}$. Progressively decreasing η_t allows to 'stabilize' around a certain iterate.

Main convergence results

The proofs of the following convergence bounds are available in the sources. The assumptions about ϕ and R do not differ from the Prox-GD case but we further require ϕ to have *bounded-variance* (σ^2) or *bounded second order moment* (C^2). We only state global convergence results.

further assumptions on ϕ and/or R	result	comment	reference
ϕ L -smooth , $R = 0$	complicated results exist	(see Note)	(see Note)
ϕ L -smooth and μ -PL , $R = 0$	$\mathbb{E}[f(w_T) - f^* w_0] \leq \frac{LC^2}{2\mu^2 T}$	$\eta_t = \frac{2t+1}{2\mu(t+1)^2}$	[KNS18]
ϕ L -smooth and μ -PL , $R = 0$	$\mathbb{E}[f(w_T) - f^* w_0] \leq (1 - \eta\mu)^T (f(w_0) - f^*) + \frac{LC^2\eta}{4\mu}$	$0 < \eta < \frac{1}{2\mu}$	[KNS18]
ϕ L -smooth and μ -strongly convex	$\mathbb{E}[f(\bar{w}_T) - f^* w_0] \leq \frac{L w_0 - w^* ^2}{T} + \frac{\sum_{t=0}^{T-1} \eta_t \sigma^2}{T}$	$\eta_t = \frac{1}{L+\mu t}$	[ZZ15]

Note 1: In their paper *When Does Stochastic Gradient Algorithm Work Well?* (Nguyen et al., 2018) [Pha+18] present rather poor and complicated results for the ϵ -convergence of the value $\frac{1}{T} \sum_{t=0}^{T-1} \|\nabla \phi(w_t)\|^2$ in the SGD setting for $f = \phi$ L -smooth and non-convex.

Note 2: One can observe the linear convergence of SGD ($R = 0$) under PL condition to a $A = \frac{LC^2\eta}{4\mu}$ -accurate solution with respect to criterion 1 as stated in the Quality criteria section. This observation led people trying adaptive SGD schemes for which time by time the step-size η is halved in order to avoid getting stuck at A -accuracy.

Note 3: $\bar{w}_T$ is one iterate drawn uniformly at random among the T newly visited iterates $\{w_t\}_{t=1}^T$ during the optimization process. Please note that the real bound derived in [ZZ15] is tighter as it embeds $\frac{\sum_{t=0}^{T-1} \eta_t \mathbb{E}_{i_t} [\|\nabla \phi_{i_t}(w_t) - \nabla \phi(w_t)\|^2]}{T}$ which's less or equal than $\frac{\sum_{t=0}^{T-1} \eta_t \sigma^2}{T}$. The reader may already feel the importance of the stochastic gradient *variance* in the convergence of such stochastic optimization algorithms. A broad class of modern stochastic methods is dedicated to reduce the variance associated with the estimation of the true gradient as much as possible.

3.3 Comparison

As announced in the **purpose** of the current chapter, let's conduct an experiment to highlight the superiority of Prox-SGD over Prox-GD when n is large enough. This experiment is personal. However the idea behind the structure of such the used objective function f is borrowed from the course *Non-Linear Programming* LINMA2460³.

a priori outcome

Using Prox-SGD we assume implicitly to dispose of a first-order oracle **per** sample function f_i . Hence we also dispose of a first-order oracle for the whole composite function f . Therefore one would be very much disappointed if this complement of information didn't bring any improvement in the computational time when tackling problem (1.2) with numerical algorithms.

The comparison will be operated in terms of sample gradients computations in order to reach a given ϵ -accuracy for criterion 1.

Creation of a problem

We would like ϕ to be smooth and strongly convex and R to be convex. After the random generation of a dataset made of n points $\{\tilde{x}_i\}_{i \in [n]}$ (normal distribution with moderate variance), we can build our objective function.

The proposed objective function is the following depending on w_* , the optimum chosen :

$$f : \mathbb{R}^d \rightarrow \mathbb{R}, w \rightarrow f(w) = \phi(w) + R(w) = \phi(w) + \lambda \|w - w_*\|_1$$

where $\phi(w) = \frac{1}{n} \sum_{i=1}^n \phi_i(w)$, $r(w) = \|w - w_*\|_1$, $\forall i \in [n]$, $h_i(w) = \log(1 + e^{-\langle w, \tilde{x}_i \rangle})$ and $h(w) = \frac{1}{n} \sum_{i=1}^n h_i(w)$. The ϕ_i 's are built based on the h_i 's . For $\lambda_{in} > 0$, $\forall i \in [n]$:

$$\phi_i : \mathbb{R}^d \rightarrow \mathbb{R}, w \rightarrow \phi_i(w) = h_i(w) - h(w_*) - \langle \nabla h(w_*), w - w_* \rangle + \frac{\lambda_{in}}{2} \|w - w_*\|^2$$

³Source: **UCLouvain web site**

Now let's demonstrate that f presents the desired properties.

Let's first notice that each ϕ_i is λ_{in} -strongly convex and twice continuously differentiable. The average of those sample cost functions gives ϕ which's then also λ_{in} -strongly convex and twice differentiable. As proportional to a distance, R is convex as required.

One can remark that for any $w \in \mathbb{R}^d$, $\phi(w) \geq 0$. Indeed using the equivalent definition of convex functions under differentiability we have that $\forall w \in \mathbb{R}^d$, $h(w) - h(w_*) - \langle \nabla h(w_*), w - w_* \rangle \geq 0$.

Obviously the term $\frac{\lambda_{\text{in}}}{2} \|w - w_*\|^2$ is also positive. Hence the chosen w_* is a global minimizer of ϕ . If we choose R in such a way that w_* is also a minimizer of R then w_* is a minimizer of the whole objective function f . Finally let's underline the fact that ϕ is L -smooth as there exists an upper-bound on its hessian matrix, $K = \lambda_{\text{in}} I_d + \frac{1}{4n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T$.

Indeed let's observe that $\max_{s \in \mathbb{R}} \frac{d^2(\log(1+e^{-s}))}{ds^2} = \frac{1}{4}$, $\forall w \in \mathbb{R}^d$,

$$\nabla^2 \phi(w) = \lambda_{\text{in}} I_d + \frac{1}{n} \sum_{i=1}^n \tilde{x}_i \left(\frac{d^2(\log(1+e^{-s}))}{ds^2} \right)_{|s=\langle w, \tilde{x}_i \rangle} \tilde{x}_i^T \preceq \lambda_{\text{in}} I_d + \frac{1}{4n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T = K$$

One may be convinced that $\bar{L} = \lambda_{\text{in}} + \frac{1}{4n} \sum_{i=1}^n \|\tilde{x}_i\|^2 \geq L = \|K\|_2$ is enough to ensure that

$$\nabla^2 \phi(w) \preceq L I_d \preceq \bar{L} I_d \quad \forall w \in \mathbb{R}^d$$

Scale of the problem

Let's formally recapitulate the problem at hand given a random dataset of $\{v_i\}_{i=1}^n$.

$$\min_{w \in \mathbb{R}^d} f(w) := R(w) + \phi(w) = \lambda r(w) + \frac{1}{n} \sum_{i=1}^n \phi_i(w)$$

where ϕ is λ_{in} -strongly convex differentiable on $\mathbb{R}^d$ and R is convex on $\mathbb{R}^d$, $\lambda \geq 0$

elements	expression/value
problem size (n, d)	$n = 50000$, $d = 15$
global minimizer w_*	$w_{*,j} \sim \mathbf{U}([-2, 2]) \quad \forall j \in [d]$
inner convexity enhancer λ_{in}	$\frac{1}{n}$
sample differentiable cost function ϕ_i	$\log(1 + e^{-\langle w, \tilde{x}_i \rangle}) - h(w_*) - \langle \nabla h(w_*), w - w_* \rangle + \frac{\lambda_{\text{in}}}{2} \ w - w_*\ ^2$
regularization constant λ	multiple values (see experiments)
regularizer $R(\cdot)$	$\lambda \ w - w_*\ _1$
step-size	(Prox-GD) $\eta = \frac{1}{L}$ (best), (Prox-SGD) $\eta_t = \frac{1}{\bar{L}(1 + \frac{t}{5\sqrt{n}})}$ (tuned)
distance w_0 to w_*	10

Note 1: w_* is fixed once and for all for every run for both methods ! The parameter λ_{in} ensures that ϕ is strongly convex. L is computed as proposed above and $\bar{L} = \lambda_{\text{in}} + \frac{1}{4} \sum_{i=1}^n \|\tilde{x}_i\|^2$. This step-size scheme is safer than the one, $\eta_t = \frac{1}{L + \mu t}$, proposed in [ZZ15].

This latter is very good for the first iterations but doesn't decrease fast enough for small μ . This comment underlines one practical drawback of Prox-SGD, one has to 'play' with parameters in order to get the best result possible whereas for deterministic methods the well known $\frac{1}{L}$ step gives the best convergence bound.

Note 2: As stated before the $\tilde{x}_i$'s were generated randomly. There is no *classification* underneath this optimization problem.

Results

The times collected are measured within the methods. A counter is set-up to 0 when their principal loop starts. The counter is stopped/restarted whenever a computation of objective value is required. Each test has been conducted 10 times and the results were averaged in order to lower their intrinsic variance. Classical bootstrap confidence intervals are shown for the performances. Note that the performance measure for ProxSGD is taken with $w_T \leftarrow \frac{1}{T} \sum_{i=0}^{T-1} w_t$. The ϵ -accuracy targeted is $5 \cdot 10^{-4}$. ProxGD takes respectively 2316 and 3593 epochs to achieve the required accuracy for the first and second experiments.

Smooth Case

For the first experiment $\lambda = 0$, meaning there's actually no regularization and the algorithms fall back to their primitive version, resp. GD and SGD. Thus $f = \phi$ is smooth.

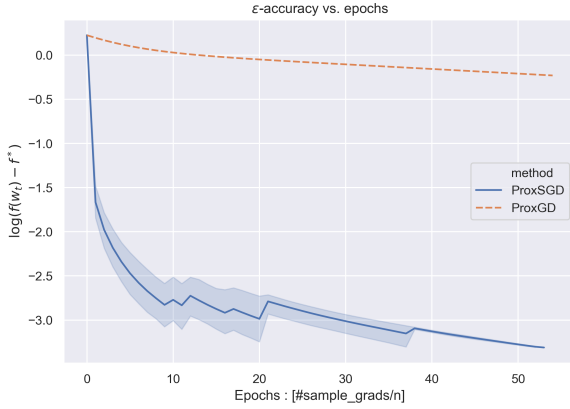


Figure 3.1: accuracy vs. epochs

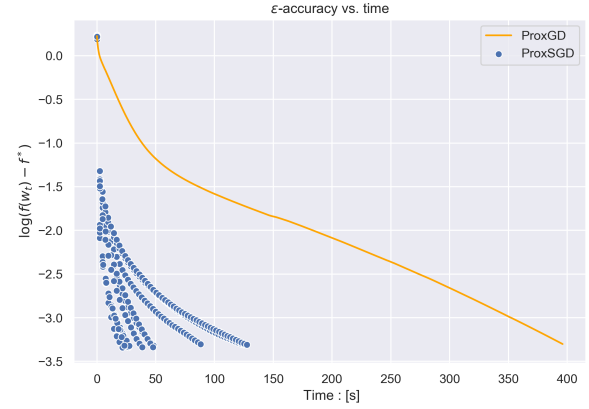


Figure 3.2: accuracy vs. time

Non-Smooth Case

For the second experiment $\lambda = \frac{1}{n}$ (classical choice in machine learning), meaning that the proximal operator does play a role and the overall problem is not smooth anymore. For the problem being it actually penalizes iterates too far away from w_* with respect to the L_1 norm.

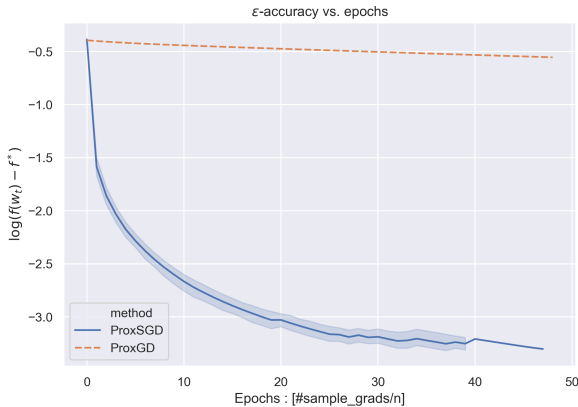


Figure 3.3: accuracy vs. epochs

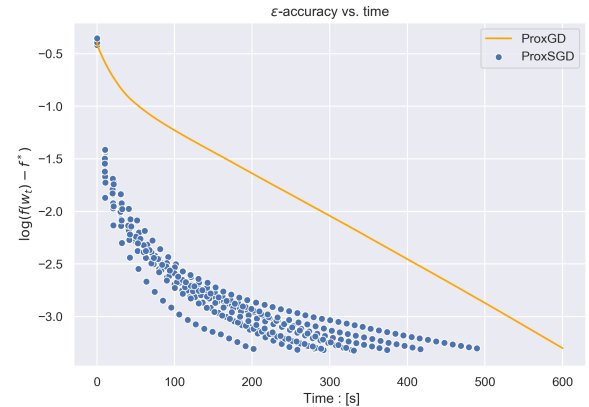


Figure 3.4: accuracy vs. time

Comments

Let's first analyze the results obtained for the smooth case. Figure 3.1 clearly shows that SGD outperforms by far GD in terms of sample gradient computations to achieve $\epsilon = 5 \cdot 10^{-4}$ accuracy with respect to the first criterion. One can observe in Figure 3.2 that every of the 10 runs of SGD was faster in time than plain GD. However one can already come up with the fact that GD achieves an epoch much more time-efficiently than SGD. Looking at the data (available) one can compute that in average SGD performs 0.4113 epochs/[s] while the raw performance of GD is ~ 5.8725 epochs/[s]. Among other reasons (passes in a *while loop*, memory access,...) one is the fact that the iterate w is updated n times during one epoch for SGD while w is updated only once per epoch for GD. However both methods present an epoch time complexity of $\mathcal{O}(nd)$ basic operations.⁴

Then let's look at the results of the non-smooth experiment (3.3 and 3.4). Now a non trivial proximal operator is used. Indeed at any update step one has to solve :

$$w_{t+1} = \text{prox}_{\eta_t \lambda \|\cdot - w_*\|_1}(w_t - \eta_t g_t) = \arg \min_{y \in \mathbb{R}^d} \frac{1}{2} \|y - (w_t - \eta_t g_t)\|^2 + \eta_t \lambda \|y - w_*\|_1$$

Hopefully there exists a closed form for such update step, yet no so usual as here the '*lasso*' (classical regularization technique in machine learning) is *shifted* by w_* not equal to $\mathbf{0}$ in general.

Example 6.8 in [Bec17a] gives the closed form for $\text{prox}_{\lambda \|\cdot\|_1}(u)$ (known as *soft thresholding*)

$$\text{prox}_{\lambda \|\cdot\|_1}(u) = \text{sgn}(u) \odot [|u| - \lambda \mathbf{e}]_+$$

Using **Theorem 6.11** of the same book, one can come up with

$$\text{prox}_{\lambda \|\cdot - w_*\|_1}(u) = w_* + \text{prox}_{\lambda \|\cdot\|_1}(u - w_*) = w_* + \text{sgn}(u - w_*) \odot [|u - w_*| - \lambda \mathbf{e}]_+$$

Again the hegemony of Prox-SGD in terms of sample gradient computations is left undefeated. Despite this large advantage we see that for this much more difficult problem Prox-SGD is barely more time-efficient than (Figure 3.4) Prox-GD. Here the iterate update cost plays a non negligible effect. Luckily for Prox-SGD the accuracy seek for this experiment wasn't too low so it did converge in decent times. Without any doubt, looking at the behavior of the time curves for Prox-SGD and Prox-GD at Figure 3.4 if ϵ was much lower Prox-GD would have beaten Prox-SGD in terms of time-efficiency. Actually looking at the convergence bounds for both Prox-GD and Prox-SGD already gives hints. Prox-GD shows a *linear* convergence towards a minimizer of f when ϕ is smooth and strongly convex (obvious while looking at graph 3.4) while the stochastic gradient norm variance prevents Prox-SGD to converge linearly.

Instead Prox-SGD converges with a *sublinear*-rate that usually beats *linear* convergence for a value of $\rho = (1 - \eta\mu)$ (rate) close to 1 and a low to medium accuracy ϵ .

This experiment was intended to motivate the use of stochastic optimization methods in order to solve (1.2) and the results are convincing. That being said, accelerated versions of Prox-GD exist and are, when n is not huge, more efficient than basic Prox-SGD (even at medium accuracy). Those methods make use of momentum directly inspired from *Nesterov's* momentum (1983). However stochastic equivalents (briefly reviewed in Chapter 6) exist meaning that the interest for stochastic optimization is far from vanishing !

⁴Assuming evaluating an exponential for $\nabla \phi_i(\cdot)$ does not yield extra complexity time ...

Chapter 4

Evolution of Stochastic Gradient Methods

Purpose

Now that stochastic gradient methods are motivated (Chapter 3) it is time to review, compare and classify the standard techniques in the domain. This is the occasion to pave the way through Chapter 5 by introducing some baseline methods but also by highlighting possible limitations & bottlenecks a researcher could face while trying to create new methods.

In what concerns the present chapter the contributions of this work are multiple:

1. for each developed era, proposition of a formal structure of the stochastic gradient
2. MSE analysis of the stochastic gradients, presentation of two general sampling schemes, multiple lemmas giving conditions under which these stochastic gradients are unbiased estimates of the true gradient

When text will be written in **color**, the paragraph content that is below will refer to one of the two contributions above.

Eras

As cleverly highlighted in the introduction of the 6-*th* version of *Katyusha* [Zhu18] the stochastic gradient optimization research field has known 3 major eras that will be explained.

In this document these eras will be referred as respectively the "*Pioneer*" era (1), the "*Variance Reduction*" era (2) and finally the "*Momentum*" era (3). Each era is linked with a new paradigm on which it is based on. Although the first era has started a couple of decades ago (*A Stochastic Approximation Method* (Robbin & Monro, 1951)) [RM51] with SGD there's still research undergoing for each of these paradigms. Nevertheless, to the best of my knowledge, much more interest is spent on the two latest. This chapter's structure will follow this idea of eras. Whenever a method is introduced it will be explained in what exactly it tackles previous problems encountered by other existing techniques and it will also be criticized in order to leave a gap to be filled-in by the upcoming methods. Time by time clear references to Chapter 5 will be made to underline personal new investigations ! Finally some of the latest papers of era (2) and others fallen strictly in era (3) will be briefly presented in Chapter 6 as they are part of the current research in the domain but hard to unify under a common framework.

4.1 Pioneer Era

The first era is the one motivated in Chapter 3. The paradigm associated with that era is to provide methods that converge in expectation at lower cost per iteration than plain gradient based methods. As a reminder one does typically not want that the number of sample gradient computations scales linearly with the number of observations (n) available at hand.

It is well recognized that first era methods have specific benefits, limitations and utility. Those methods are really suited for quick online decision updates in statistical machine learning. Indeed w is sometimes called the decision vector. When one only has access to a reduced-size batch of observations coming from the generating random variable Υ (looking back to notations of Chapter 1) w is quickly updated using SGD-like rules that do not require a total information of the data defining the cost function. First era methods are particularly useful for low ϵ -precision requirements for either criterion 2 or 1. Finally, their implementation is rather easy !

First era methods subscribe to the paradigm of stochastic approximation initiated by [RM51]. From that arose this idea of online learning. Nowadays this paradigm is widely borrowed by a specific branch of machine learning : *reinforcement learning*. However this goes beyond the scope of this master thesis about optimization.

Despite all those pros we will observe that the two main limitations of first era methods are their *sublinear* rate of convergence using diminishing step-sizes hence making it practically impossible to get very accurate solutions in decent times and performances mainly driven by the choice of the step-size schedule.

4.1.1 General Scheme

Here below is given a personal definition of how first era methods work. One full update step is presented, shedding light on my general expression of their stochastic gradient (4.2).

1st ERA: at any $t \in \mathbb{N}$, conditional on $w_t \in \mathbb{R}^d$, first-era update iteration proceeds as

$$S_t \sim \mathcal{P}_t \tag{4.1}$$

$$g_t = \Lambda(S_t) \left[\sum_{i \in S_t} m_{t,i,l(i)} \nabla \phi_i(w_t) \right] \tag{4.2}$$

$$w_{t+1} = \text{prox}_{\eta_t R}(w_t - \eta_t g_t) \tag{4.3}$$

Note 1: Step (4.1) is one mandatory step common to all the eras that will be called the *lottery* at a given iteration. By default the lottery is organized as a random multiset drawing following a probability distribution $\mathcal{P}_t$. Nevertheless the only requirement of step (4.1) is to provide a set S_t of indices in $[n]$ for which the sample gradients will be computed. The size of S_t , $|S_t|$, may vary with iterations and can be stochastic as well.

Note 2: In (4.2) one constructs the so called *stochastic gradient* g_t . The *weighting coefficients* $\{m_{t,i,l(i)}\}_{i=1}^n$ are positive quantities that will play a key role in the lemmas. If S_t is built upon a selection of elements, one by one, then $l(i)$ represents the step at which $i \in [n]$ had been added to S_t . Otherwise, if every indice of S_t is drawn at the same time $l(i) = 1$ for all $i \in S_t$ and this subscript is dropped. The purpose of the multiplicative term $\Lambda(S_t)$ is to give a scaling degree of freedom depending exclusively on S_t as a whole. Thus there is a clear distinction between $\Lambda(S_t)$ and the *weighting coefficients*.

4.1.2 Analysis of the Stochastic Gradient

In this subsection, the properties of the stochastic gradient introduced above are studied in many details. A few sampling concepts must be deepened in the current section in order to not rewrite them in the part dedicated to the analysis of the stochastic gradient for second eras.

Note: To keep consistent with the notations used so far T represents the total number of update steps as in (4.3) that one method achieves. Let me remind that $\mathbb{E}[\cdot]$ is the global expectation taking into account all the random events occurring within a method, i.e. all the "lottery" steps defining $S_0, \dots, S_{T-1}$. Note that the knowledge of the sequence of lottery steps until and including $t-1 \leq T$ defines uniquely the sequence $\{w_l\}_{l=0}^t$ conditional to the knowledge of w_0 . Let's introduce then for every $t \in \{0, \dots, T-1\}$, $\mathcal{F}_t := \sigma(w_0, S_0, \dots, S_{t-1})$ that is the σ -algebra generated by the random events until and including step t . We express the "one-step" expectation as follows $\mathbb{E}_{S_t}[\cdot]$ or equivalently $\mathbb{E}[\cdot|\mathcal{F}_t]$. For the rest of this document $2^{\mathcal{A}}$ will denote the *power set* of $\mathcal{A}$, that is the set of all the subsets of $\mathcal{A}$.

Expectation

Let's analyze the bias of g_t as an estimator of $\nabla\phi(w_t)$ thanks to the computation of the "one-step" expectation $\mathbb{E}[g_t|\mathcal{F}_t]$. It will be helpful to derive conditions for unbiasedness.

In order compute to $\mathbb{E}[g_t|\mathcal{F}_t]$, one first needs to know informations about the sampling scheme operated in the lottery step! Therefore, the analysis of the bias will be cut in two parts, each part is associated to one sampling scheme. The first sampling scheme is "natural" and the second one is inspired from *Nonconvex Variance Reduced Optimization with Arbitrary Sampling* (Samuel Horváth & Peter Richtárik, 2018) [HR18] but this master thesis generalizes their talk.

Let's fix $t \in \mathbb{N}$, let the following content be conditional on $\mathcal{F}_t$.

1. **Natural:** S_t is drawn according to a direct probability distribution $\mathcal{P}_t$. One must then know the mapping $\mathcal{M}_t : 2^{[n]} \rightarrow \mathbb{R}^+$, $S_t \subseteq [n] \rightarrow \mathcal{M}_t(S_t) := \mathbb{P}[S_t] \geq 0$ that assigns the "one-step" probability (following $\mathcal{P}_t$'s distribution) of realization of each subset $S_t \subseteq [n]$.

One can write

$$\mathbb{E}[g_t|\mathcal{F}_t] = \sum_{S_t \in 2^{[n]}} \mathcal{M}_t(S_t) \Lambda(S_t) \left[\sum_{i \in S_t} m_{t,i} \nabla\phi_i(w_t) \right] \quad (4.4)$$

Note : This general setting is rather rare. Without any further assumption assuming a *proper* mapping $\mathcal{M}_t$ there are $N = 2^n$ non-zero probabilities to consider. Hence one has to generate a single point but in a finite state space of 2^n elements. Such univariate sampling procedure usually uses either $\mathcal{O}(\log(N))$ operations for an efficient binary search sampling method or up to $\mathcal{O}(N \log(N))$ operations only once at pre-processing time for *alias-algorithm*.¹ It quickly becomes prohibitive for large n . Nevertheless when one decides that the only non-zero probabilities on $2^{[n]}$ are for subsets of fixed size $\hat{b} \in \{1, \dots, n\}$ then the sample space size reduces to $N_{\hat{b}} := \binom{n}{\hat{b}}$; tractable if $\hat{b}$ is not too large.

A common practice that falls in the definition of this "natural" sampling procedure is to perform Uniform Sampling (US) on the set of subsets of a fixed batch size $\hat{b}$. Let's call it $2^{[n],\hat{b}}$ for future writing purposes. In that context, $\mathcal{M}_t(S_t) = \frac{1}{N_{\hat{b}}}$ if $|S_t| = \hat{b}$ and $\mathcal{M}_t(S_t) = 0$ otherwise. Examples are given after **Lemma 1**.

¹Source : *Alias Method* (A.J. Walker, 1971), see : https://en.wikipedia.org/wiki/Alias_method

Under the current sampling scheme it seems in general really difficult to ensure that

$$\mathbb{E}[g_t|\mathcal{F}_t] = \nabla\phi(w_t) \quad (\text{unbiasedness property})$$

However imposing that the weighting coefficients $\{m_{t,i}\}_{i=1}^n$ are all neutral and set to 1 and under the hypothesis that given $\hat{b} \in \{1, \dots, n\}$, either $\mathcal{M}_t(S_t) > 0 \forall S_t \in 2^{[n],\hat{b}}$ or $\mathcal{M}_t(S_t) = 0 \forall S_t \in 2^{[n],\hat{b}}$, it is possible to state sufficient conditions to get the unbiasedness.

Lemma 1 (ERA 1 , first sampling scheme : sufficient condition for unbiased updates) : Consider the stochastic gradient g_t as built in (4.2) with the first sampling scheme defined by $\mathcal{M}_t$. Let's suppose that $\forall i \in [n]$, $m_{t,i} = 1$ and that $\forall \hat{b} \in \{1, \dots, n\}$, one either verifies that $\mathcal{M}_t(S_t) > 0 \forall S_t \in 2^{[n],\hat{b}}$ or $\mathcal{M}_t(S_t) = 0 \forall S_t \in 2^{[n],\hat{b}}$. Let's denote by K the set of batch sizes $\hat{b}$ such that every subset of size $\hat{b} \in K$ has a strictly positive probability to be drawn.

The choice

$$\Lambda(S_t) = \begin{cases} \frac{1}{|K|n \binom{n-1}{|S_t|-1} \mathcal{M}_t(S_t)} & \text{if } \forall S_t \in 2^{[n],|S_t|}, \mathcal{M}_t(S_t) > 0 \\ 0 & \text{otherwise} \end{cases}$$

ensures that $\mathbb{E}[g_t|\mathcal{F}_t] = \nabla\phi(w_t)$

Proof: Under our hypothesis, let's rewrite the equation (4.4) in a more convenient way

$$\mathbb{E}[g_t|\mathcal{F}_t] = \sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n],\hat{b}}} \mathcal{M}_t(S_t) \Lambda(S_t) \left[\sum_{i \in S_t} \nabla\phi_i(w_t) \right]$$

Applying the above choice for $\Lambda(\cdot)$ leads to

$$\begin{aligned} \mathbb{E}[g_t|\mathcal{F}_t] &= \sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n],\hat{b}}} \frac{1}{|K|n \binom{n-1}{|S_t|-1}} \left[\sum_{i \in S_t} \nabla\phi_i(w_t) \right] \\ &= \sum_{\hat{b} \in K} \frac{1}{|K|n \binom{n-1}{\hat{b}-1}} \sum_{S_t \in 2^{[n],\hat{b}}} \sum_{i \in S_t} \nabla\phi_i(w_t) \end{aligned}$$

Here the important thing to notice is that the term $\sum_{S_t \in 2^{[n],\hat{b}}} \sum_{i \in S_t} \nabla\phi_i(w_t)$ is actually equivalent to $\binom{n-1}{|S_t|-1} \sum_{i=1}^n \nabla\phi_i(w_t)$. Among the $\binom{n}{|S_t|}$ subsets of size $|S_t|$, a single indice i will be present in $\binom{n-1}{|S_t|-1}$ of them ! Then we are left with

$$\mathbb{E}[g_t|\mathcal{F}_t] = \sum_{\hat{b} \in K} \frac{1}{|K|n} \sum_{i=1}^n \nabla\phi_i(w_t) = \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(w_t) = \nabla\phi(w_t)$$

□

Let's now mention some existing, easy-to-read, examples of stochastic gradients g_t that can be put in the defined first era. These latter can be generated by this first sampling scheme. The goal of the next paragraph is to show that the theoretical structure that is built step by step already covers some familiar cases in stochastic gradient optimization.

Indeed let's consider that the assumptions of **Lemma 1** hold. In the case where $K = \{1\}$, we have $\Lambda(S_t) = \frac{1}{n\mathcal{M}_t(S_t)}$ for any singleton $S_t = \{i\}$ ($i \in [n]$) and $\Lambda(S_t) = 0$ for any other subset of $[n]$. Actually in that case we have the equivalence $\mathcal{M}_t(\{i\}) = \mathbb{P}[S_t = \{i\}] = p_{t,i}$ and the stochastic gradient g_t simply boils down to

$$g_t = \frac{\nabla \phi_i(w_t)}{np_{t,i}} \quad (4.5)$$

and eventually when using Uniform Sampling (US), $p_{t,i} = \frac{1}{n}$ at any iteration t

$$g_t = \nabla \phi_i(w_t) \quad (4.6)$$

The two last g_t are respectively the single sample stochastic gradient with (4.5) / without (4.6) importance sampling, the most widely spread stochastic gradients.

For the generalized case when $\hat{b}$ is non-necessarily equal to 1, using (US) but still considering that $|K| = 1$, e.g. $K = \{\hat{b}\}$, we have $\mathcal{M}_t(S_t) = \frac{1}{\binom{n}{\hat{b}}}$ for any subset $S_t \in 2^{[n],\hat{b}}$ and 0 for any other subset. Then by noticing that $\frac{\binom{n}{\hat{b}}}{\binom{n-1}{\hat{b}-1}} = \frac{n}{\hat{b}}$ we fall back on the expression of the typical mini-batch stochastic gradient

$$g_t = \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t) \quad (4.7)$$

2. Independent sequential lotteries on $\{0\} \cup [n]$: now let's detail the second, personal, sampling scheme investigated. As a reminder it generalizes the work of [HR18] It will be explained in why in the notes after **Lemma 2**.

Here, S_t is constructed upon multiple univariate lotteries. Let's extend the possibility for indice 0 associated with a neutral sample gradient $\mathbf{0}$ to be drawn during an univariate lottery on the set $\{0\} \cup [n]$. The *weighting coefficient* of the index 0 as defined in **Note 2** after the presentation of the first era structure will be always 0.

One fixes in advance D univariate probability distributions on $\{0\} \cup [n]$, $\{\mathcal{P}_{t,l}\}_{l=1}^D$. Let's start with an empty set of indices $\tilde{S}_t = \emptyset$. Then the sampling procedure uses sequentially $\mathcal{P}_{t,l}$ in order to add an l -th element $\tilde{i}_{t,l}$ to $\tilde{S}_t$, beginning with $l = 1$ and ending with $l = D$. That process ensures that the indices chosen are independent. One supposes that for any indice $k \in \{0\} \cup [n]$, one has access to its probability $p_{t,k,l} \geq 0$ to be drawn at univariate lottery l , under any form of randomness.² $\forall l \in \{1, \dots, D\}, \sum_{k=0}^n p_{t,k,l} = 1$.

One can write

$$\mathbb{E}[g_t | \mathcal{F}_t] = \sum_{\tilde{\mathbf{i}}_t \in \{0, \dots, n\}^D} \left(\prod_{l=1}^D p_{t, \tilde{i}_{t,l}, l} \right) \Lambda(\{\tilde{\mathbf{i}}_t\}) \sum_{i \in \{\tilde{\mathbf{i}}_t\}} m_{t,i,l(i)} \nabla \phi_i(w_t) \quad (4.8)$$

This time the term $\Lambda(\cdot)$ is set to a constant value Λ in order to state easy sufficient conditions for which g_t is an unbiased estimate of $\nabla \phi(w_t)$ conditional to $\mathcal{F}_t$ while using the second sampling scheme. In that case the above expression is separable

$$\mathbb{E}[g_t | \mathcal{F}_t] = \Lambda \sum_{l=1}^D \sum_{\tilde{i}=0}^n p_{t,\tilde{i},l} m_{t,\tilde{i},l} \nabla \phi_i(w_t) = \sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(w_t) \quad (4.9)$$

²Here, by independence of the events, $p_{t,k,l}$ is only defined by $\mathcal{P}_{t,l}$.

Again, let's state sufficient conditions to get the unbiasedness.

Lemma 2 (ERA 1, second sampling scheme : sufficient condition for unbiased updates) : Consider the stochastic gradient g_t as built in (4.2) with the second sampling scheme defined by the probabilities $p_{t,k,l}$ for any $k \in \{0, \dots, n\}$, $l \in \{1, \dots, D\}$. Let's suppose that $\Lambda(S_t) = \Lambda > 0$ for any $S_t \in 2^{[n]}$. For every $i \in [n]$, let $\zeta_i := \{l \in \{1, \dots, D\} | p_{t,i,l} > 0\}$ and $\Lambda = \frac{1}{D}$. If for every set ζ_i , $|\zeta_i| > 0$ then the choice

$$m_{t,k,l} = \begin{cases} \frac{D}{n|\zeta_k|p_{t,k,l}} & p_{t,k,l} > 0 \\ 0 & \text{otherwise} \end{cases}$$

ensures that $\mathbb{E}[g_t | \mathcal{F}_t] = \nabla \phi(w_t)$

Proof: Let's rewrite (4.9) using the hypothesis. The desired result follows trivially.

$$\begin{aligned} \mathbb{E}[g_t | \mathcal{F}_t] &= \sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(w_t) \\ &= \frac{1}{D} \sum_{i=1}^n \left(\sum_{l \in \zeta_i} p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(w_t) \\ &= \frac{1}{D} \sum_{i=1}^n \left(\sum_{l \in \zeta_i} \frac{D}{n|\zeta_i|} \right) \nabla \phi_i(w_t) \\ &= \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(w_t) = \nabla \phi(w_t) \end{aligned}$$

□

Note: In *Arbitrary Sampling* [HR18], Richtárik develops this idea of independent lotteries in the particular case where $D = n$. Each lottery l is a coin flip and either l is added to S_t with probability $p_{t,l,l}$ or 0 with probability $1 - p_{t,l,l}$. This sampling will be detailed in two pages.

Trace of MSE

The study of an estimator, here g_t , often consists in the study of the trace of its MSE (so called "Mean Squared Error"). Generally one would like to build an estimator with the lowest MSE (trace) possible. The ultimate goal of this subsection is to give an upper-bound on that MSE trace for first-era methods. We will observe that it cannot decrease to 0. As it will be reminded when convenient, here under I only transpose existing results to my defined first-era.

Let's first recall the classical definition of the MSE.

Definition 10 (MSE) : Let χ be a (possibly random) variable living in a d -dimensional vectorial space and a (possibly random) target quantity τ contained in the same space. The MSE of χ as an estimator of τ is defined as

$$\text{MSE}(\chi) := (\mathbb{E}[\chi - \tau])(\mathbb{E}[\chi - \tau])^T + \mathbb{E}[(\chi - \mathbb{E}[\chi])(\chi - \mathbb{E}[\chi])^T] \quad (4.10)$$

bias term
variance term

In the case where $\mathbb{E}[\chi] = \mathbb{E}[\tau]$ then

$$\text{MSE}(\chi) = \mathbb{E}[(\chi - \mathbb{E}[\tau])(\chi - \mathbb{E}[\tau])^T] = \mathbb{E}[(\chi - \mathbb{E}[\chi])(\chi - \mathbb{E}[\chi])^T] \quad (4.11)$$

Note: The MSE as defined above is a matrix of $\mathbb{R}^{d \times d}$. The ultimate goal, sometimes not achievable, for an estimator as χ is to present a $\text{MSE}(\chi)$ such that for any other estimator χ' the following conic inequality holds

$$\text{MSE}(\chi') - \text{MSE}(\chi) \succeq_{\mathbb{S}_+^d} \mathbf{0}$$

A relaxation of such requirement is the following constraint for any other χ'

$$\text{tr}(\text{MSE}(\chi')) - \text{tr}(\text{MSE}(\chi)) \geq 0 \quad (4.12)$$

Let's consider the class of unbiased estimators. The intuition behind such requirement (4.12) is to get an estimator with the lowest mean per coordinate variance. Taking the trace of the MSE in (4.11) leads to

$$\text{tr}(\text{MSE}(\chi)) = \sum_{j=1}^d \mathbb{E}[(\chi_{(j)} - \mathbb{E}[\tau_{(j)}])^2] = \mathbb{E}[\|\chi - \mathbb{E}[\tau]\|_2^2]$$

The reader might recognize this last expression when χ actually depicts an one-step unbiased stochastic gradient g_t ($\mathbb{E}[g_t|\mathcal{F}_t] = \nabla\phi(w_t)$) for which $\tau = \mathbb{E}[\tau|\mathcal{F}_t] = \mathbb{E}[\nabla\phi(w_t)|\mathcal{F}_t] = \nabla\phi(w_t)$.

In that context,

$$\text{tr}(\text{MSE}(g_t)) = V_t = \mathbb{E}_{S_t}[\|g_t - \nabla\phi(w_t)\|_2^2] = \mathbb{E}[\|g_t - \nabla\phi(w_t)\|_2^2 | \mathcal{F}_t]$$

In Chapter 5 a proof of convergence for first-era methods will be developed. This latter will show the theoretical impact of V_t on the convergence rate.

In a more generic way the trace of the MSE is given by :

$$\begin{aligned} \text{tr}(\text{MSE}(\chi)) &= \text{tr}((\mathbb{E}[\chi - \tau])(\mathbb{E}[\chi - \tau])^T) + \text{tr}(\mathbb{E}[(\chi - \mathbb{E}[\chi])(\chi - \mathbb{E}[\chi])^T]) \\ &= \sum_{j=1}^d (\mathbb{E}[\chi_{(j)} - \tau_{(j)}])^2 + \mathbb{E}[(\chi_{(j)} - \mathbb{E}[\chi_{(j)}])^2] \\ &= \|\mathbb{E}[\chi - \tau]\|_2^2 + \mathbb{E}[\|\chi - \mathbb{E}[\chi]\|_2^2] \end{aligned}$$

As announced, let's review "Arbitrary Sampling results" presented in [HR18] and in *Coordinate descent with arbitrary sampling II: expected separable over-approximation* (Qu & Richtárik, 2016) [QR14]. The results below assume that when S_t is picked-up, an indice $i \in [n]$ cannot be found more than once in S_t : that is $S_t \subseteq 2^{[n]}$ (S_t is a subset but not a multiset of $[n]$).

Lemma 3 is displayed as it provides an upper-bound on $\text{tr}(\text{MSE}(\chi))$ when χ is an unbiased estimator of $\bar{\chi}$. Hence, we could easily apply Lemma 3 to the case of $\chi = g_t$, yielding the desired upper-bound on $\text{tr}(\text{MSE}(g_t))$ desired in this subsection.

Lemma 3 (variance bound in [HR18]) : Let $\chi_1, \dots, \chi_n$ be vectors in $\mathbb{R}^d$ and $\bar{\chi} = \frac{1}{n} \sum_{i=1}^n \chi_i$ their average. Let $S \subseteq 2^{[n]}$ be chosen under a *proper* sampling : i.e. sampling scheme that ensures that for any $i \in [n]$, $p_i := \mathbb{P}[i \in S] > 0$. Assume that there exists $v \in \mathbb{R}^d$ such that

$$\text{Diag}(p \circ v) - (\mathbf{P} - p p^T) \succeq_{\mathbb{R}_+^d} \mathbf{0} \text{ where } \forall i, j \in [n], (\mathbf{P})_{ij} = p_{ij} = \begin{cases} \mathbb{P}[i \in S] & \text{if } i = j \\ \mathbb{P}[\{i, j\} \in S] & \text{if } i \neq j \end{cases}$$

and $p = (p_1, \dots, p_n)^T$. Then one can prove that the following bound on the trace on the covariance matrix for χ holds :

$$\mathbb{E}[\|\sum_{i \in S} \frac{\chi_i}{n p_i} - \bar{\chi}\|_2^2] \leq \frac{1}{n^2} \sum_{i=1}^n \frac{v_i}{p_i} \|\chi_i\|_2^2$$

Note : The proof of this lemma can be found in the supplementary part of reference [HR18].

An upper-bound on $\text{tr}(\text{MSE}(g_t))$ will only be given for the the **Independent sequential lotteries** case. The **natural** sampling is too complex to analyze excepted in the uniform case.

As required by the above lemma, let's only consider cases for which S_t is a subset of $2^{[n]}$, i.e. no repetition of indices. Then the subscript l in the *weighting coefficients* $m_{t,i,l}$ is dropped since we must assume that among the D lotteries, for any $i \in [n]$, only one of the $p_{t,i,l}$ will be strictly positive.

Then without loss of generality let's rewrite first era stochastic gradients (considering then cases for which $\Lambda(S_t)$ is a constant Λ) as

$$\begin{aligned} g_t &= \Lambda \left(\sum_{i \in S_t} m_{t,i} \nabla \phi_i(w_t) \right) \\ &= \sum_{i=1}^n \frac{\tilde{m}_{t,i} \nabla \phi_i(w_t)}{np_{t,i}} 1_{i \in S_t} \end{aligned}$$

where $1_{i \in S_t}$ is a "truth" random variable (previously the source of randomness was the set S_t as whole) taking value 1 if and only if $i \in [n]$ is in S_t and 0 otherwise : $p_{t,i} = \mathbb{E}[1_{i \in S_t} | \mathcal{F}_t] = \mathbb{P}[i \in S_t]$. $\tilde{m}_{t,i} = \frac{\Lambda m_{t,i}}{np_{t,i}}$ is a generic compound coefficient for case i . The one-step expectation of g_t is

$$\begin{aligned} \mathbb{E}[g_t | \mathcal{F}_t] &= \sum_{i=1}^n \frac{\tilde{m}_{t,i} \nabla \phi_i(w_t)}{np_{t,i}} \mathbb{E}[1_{i \in S_t} | \mathcal{F}_t] \\ &= \frac{1}{n} \sum_{i=1}^n \tilde{m}_{t,i} \nabla \phi_i(w_t) \end{aligned}$$

In order to match the above notations let's note $\chi_{t,i} = \tilde{m}_{t,i} \nabla \phi_i(w_t)$ and $\mathbb{E}[g_t | \mathcal{F}_t] = \bar{\chi}_t$.

$$g_t = \sum_{i \in S_t} \frac{\chi_{t,i}}{np_{t,i}} \quad \mathbb{E}[g_t | \mathcal{F}_t] = \bar{\chi}_t$$

Now we have all the tools necessary to give an upper-bound on the trace of the MSE of first era stochastic gradients as estimators of the true gradient $\nabla \phi(w_t)$ at an iterate $w_t \in \mathbb{R}^d$! Assuming³ there exists a vector v_t for the used sampling scheme fulfilling the conditions of **Lemma 3**, one states:

$$\begin{aligned} \text{tr}(\text{MSE}(g_t)) &= \|\mathbb{E}[g_t | \mathcal{F}_t] - \nabla \phi(w_t)\|_2^2 + \mathbb{E}[\|g_t - \mathbb{E}[g_t | \mathcal{F}_t]\|_2^2 | \mathcal{F}_t] \\ &\leq \frac{1}{n^2} \left\| \sum_{i=1}^n (\tilde{m}_{t,i} - 1) \nabla \phi_i(w_t) \right\|_2^2 + \frac{1}{n^2} \sum_{i=1}^n \frac{v_{t,i}}{p_{t,i}} \|\tilde{m}_{t,i} \nabla \phi_i(w_t)\|_2^2 \end{aligned}$$

Operational v 's

Thanks to the works of [HR18] and [QR14], one can find results for practical v 's (proofs again in the *supplementary part* of [HR18]). Richtárik also proposes an approximation of his independent sequential lottery ($D = n$) which's presented below for the curious reader.

1. **Independent sequential lotteries with $D = n$.** Since we want that any indice $i \in [n]$ can be taken at most once in S_t , the $D = n$ lotteries must be such that $p_{t,i,l} = 0$ for every $l \neq i$ and $p_{t,i,l} = p_{t,i} = \mathbb{P}[i \in S_t]$ for $l = i$. Actually it boils down to what authors call *Independent Sampling* (S^*) in [HR18] : $D = n$ flip coins are required. At any $l \in \{1, \dots, D\}$, we have a $p_{t,i}$ chance to draw i and a $1 - p_{t,i}$ chance to draw the *neutral case* 0 (meaning we do not add any further sample gradient to g_t at lottery l). $v_t = (1 - p_{t,1}, \dots, 1 - p_{t,n})^T$ is a valid choice. In that case $\mathbb{P}[\{i, j\} \in S_t] = p_{t,i}p_{t,j}$ when $i \neq j$.

³It is the case for all the practical implementations I review.

2. **Approximated independent sequential lotteries with $D = a$.** One drawback to $D = n$ independent lotteries is that such sampling procedures takes a while when n is big. It actually can be well approximated with a much lower number of individual lotteries. Let's denote by k the number of cases for which $p_{t,i} < 1$ and let's introduce $a = \lceil \max\{kp_{t,i} | p_{t,i} < 1\} \rceil$. The *Approximated independent sampling* in [HR18] works as follows. One chooses uniformly at random a subset $\hat{S}_t$ of indices from $[n]$ of size a .

Then for each element $\hat{i}$ of $\hat{S}_t$ a coin is tossed ; $\hat{i}$ has a $\frac{kp_{t,i}}{a}$ to be added to S_t .

That way $\mathbb{P}[i \in S_t] = \frac{\binom{k}{a-1}}{\binom{k}{a}} \frac{kp_{t,i}}{a} = p_{t,i}$ whereas $\mathbb{P}[\{i, j\} \in S_t] = p_{t,i}p_{t,j} \frac{k(a-1)}{a(k-1)}$.

Hence this approximated sampling require a independent coin tosses. A practical example with $\hat{b} = \sqrt{n}$, $k = n$ and $\max_{i \in [n]} \{p_{t,i} = 10 \frac{\hat{b}}{n}\}$ gives $a = 10\sqrt{n}$. Imagine $n = 10^6$, the quantity $\frac{k(a-1)}{a(k-1)}$ becomes $\frac{10^{-2}n^2 - n}{10^{-2}n^2 - 10^{-2}n} = 0.9999 \dots \simeq 1$ but $D = a = \frac{n}{100}$!

This time $v_t = (1 - p_{t,1}(1 - \frac{k-a}{a(k-1)}), \dots, 1 - p_{t,n}(1 - \frac{k-a}{a(k-1)}))^T$ is a valid choice.

3. **Direct uniform mini-batch sampling.** This case was the classical uniform mini-batch sampling (first sampling scheme) without repetition where for every set S_t of size $\hat{b}$, $\mathcal{M}_t(S_t) = \frac{1}{\binom{\hat{b}}{b}}$. Then for every $i \in [n]$, $p_{t,i} = \frac{\hat{b}}{n}$ and a valid choice for v_t is $(\frac{n-\hat{b}}{n-1}, \dots, \frac{n-\hat{b}}{n-1})^T$. Actually *direct uniform mini-batch sampling* can be seen as special case of *approximated independent sequential lotteries* where $a = \hat{b}$ and $\frac{kp_{t,i}}{a} = 1$.

Apart from the interesting Lemma 1 and 2, the main take away message of this theoretical subsection is that for first-era methods, $\text{tr}(\text{MSE}(g_t))$ ($= V_t$ for unbiased methods) do not decrease to 0. Indeed, in general at $w_* \in S^*$, $\|\nabla \phi_i(w_*)\| \neq 0 \forall i \in [n]$. This will be the starting point for second-era methods.

4.1.3 Methods

Now that the theoretical background for the first era is well established let's review with ease some of the most widely spread methods that fall in my first-era framework.

(MB)-SGD

At the end of Chapter 3 and at the very beginning of the present chapter, (Prox)-SGD has been extensively explained. Let's investigate its direct successor.

Motivation:

An intuitive extension to this basic method resides in computing more than one single sample gradient in order to build the used stochastic gradient. Of course the computational cost per iteration for SGD scales linearly with the effective batch size $|S|$. Nevertheless many empirical tests show the empirical benefits of using batch updates in SGD.

One immediate comment is that the stochastic gradient g constructed is a more accurate estimator of the true gradient : actually its MSE trace is divided by $|S|$. SGD proposes an unbiased stochastic gradient so no improvement with respect to the trace of the bias term but the variance term is typically divided by $|S|$ and eventually more depending on the sampling.

More importantly with the advent of parallel computing Mini-Batch (MB) SGD becomes obvious and necessary in order to remain competitive. The time cost per iteration will actually not increase that much since the work of computing $|S|$ sample gradients can be split up among multiple U computing units. The optimal effective computational time is then of $\frac{|S|}{U}$.

Algorithm

Algorithm 3 MB-SGD

Initialization : $\{\eta_t\}_{t \in \mathbb{N}}$ step-size schedule $\forall t \in \mathbb{N}$, $\eta_t > 0$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter

while stopping criterion not verified **do**

$S_t \sim \mathbf{U}(2^{[n], \hat{b}})$ (without replacement) or $S_t \sim \mathbf{U}([n]^{\hat{b}})$ (with replacement)

$g_t \leftarrow \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t)$ (for both settings, with/without replacement)

$w_{t+1} \leftarrow \text{prox}_{\eta_t R}(w_t - \eta_t g_t)$

$t \leftarrow t + 1$

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note : With SGD in its "without replacement" setting we have $p_{t,i} = \frac{\hat{b}}{n}$ for every $i \in [n]$ and then the coefficient $\frac{1}{np_{t,i}} = \frac{1}{\hat{b}}$. The updates are unbiased with certainty as $\tilde{m}_{t,i} = 1$ for any $t \in \mathbb{N}$ and $i \in [n]$. Even formulated in its "with replacement" setting SGD is still unbiased but g_t obviously cannot match the expression $g_t = \sum_{i \in S_t} \frac{\tilde{m}_{t,i}}{np_{t,i}} \nabla \phi_i(w_t)$. It fits however with the general second sampling scheme where an indice can be added more than once in S_t . Here for every of the $D = \hat{b}$ lotteries each indice $i \in [n]$ has a $p_{t,i,l} = \frac{1}{n}$ probability to be drawn. Hence $|\zeta_i| = \hat{b}$, $\lambda = \frac{1}{\hat{b}}$ and finally $m_{t,i,l} = 1$.

Experiments

The goal pursued by the following personal experiments is to show how the size of the mini batch $\hat{b}$ can affect SGD's performances and to see how, empirically, is the upper-bound on $\text{tr}(\text{MSE}(g_t))$ related to the observed behaviors. **For all the set-up problems both in Chapter 4 and 5, please refer to Appendix 2.)** to discover how the problems were built, especially how was w_* determined.

Problem (1)

Keeping the same formalism used so far, the first tackled problem to analyze mini-batch SGD against single sample update SGD is a classical logistic regression with ℓ_1 -norm regularization (also called *lasso*). Let's summarize the integrated elements :

elements	expression/value
problem size (n, d)	$n = 50000$, $d = 15$
inner convexity enhancer λ_{in}	0
sample differentiable cost function ϕ_i	$\log(1 + e^{-\langle w, \tilde{x}_i \rangle}) + \frac{\lambda_{\text{in}}}{2} \ w\ ^2$
regularization constant λ	$\frac{1}{n}$
regularizer $R(\cdot)$	$\lambda \ w\ _1$
step-size	$\eta_t = \frac{1}{\bar{L}(1 + \frac{B}{\sqrt{n}})}$ for multiple values of $B \geq 0$ (basis)

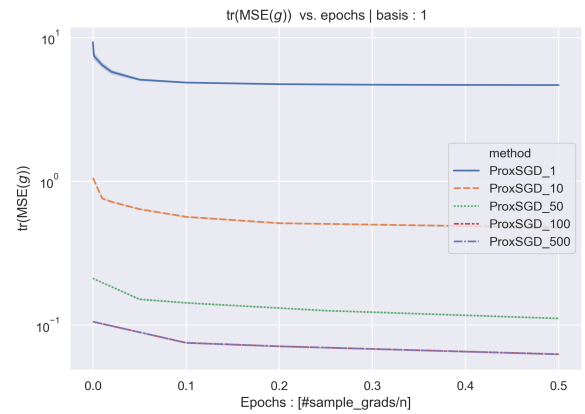
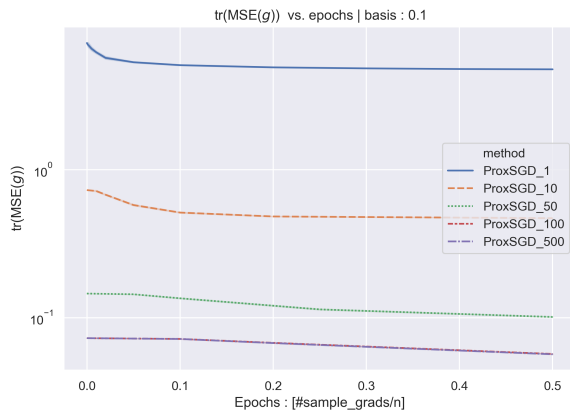
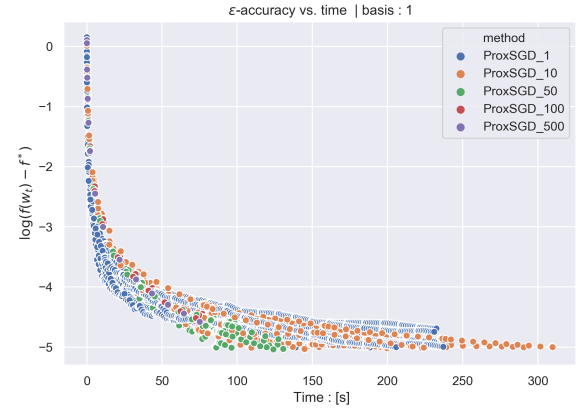
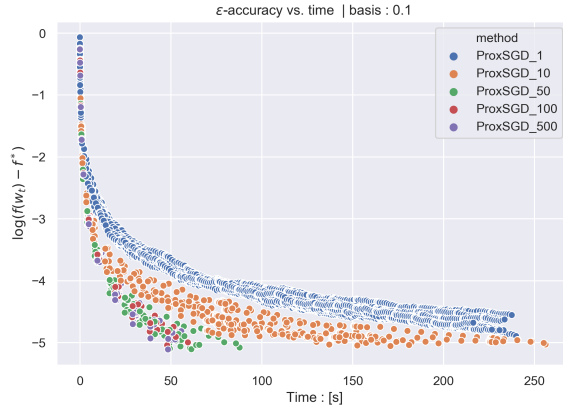
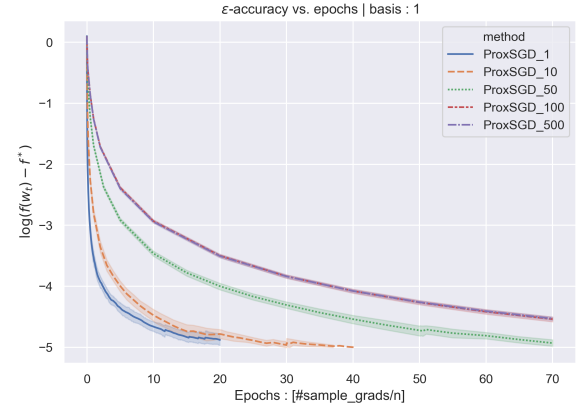
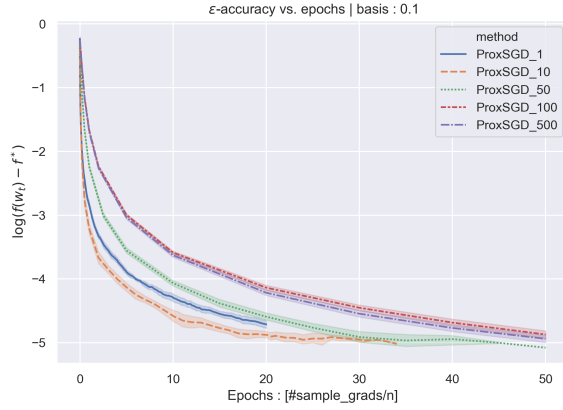
Note: This problem is certainly convex but non-smooth and *a priori* non-strongly convex. Nonetheless, empirically, at each of the $\{w_t\}_{t=0}^T$ that will be encountered during the optimization process, ϕ will be probably strongly convex with a tiny μ in practice. The values of B (basis) that have been tested are $\{0.1, 0.5, 1, 5\}$ but we will only display results for $B = 0.1$ and $B = 1$ since there isn't much more to say for the other cases. Lipschitz constants for the ϕ_i have been devised as in Chapter 3, $L_i = \frac{\|\tilde{x}_i\|_2^2}{4}$ for every $i \in [n]$.

Results (1)

Each test has been conducted 10 times and the results were averaged in order to lower their intrinsic variance. Classical bootstrap confidence intervals are shown for the performances.

Note that the performance measure for ProxSGD is taken with $w_T \leftarrow \frac{1}{T} \sum_{i=0}^{T-1} w_t$.

The ϵ -accuracy targeted is 10^{-5} . The versions of Mini-Batch ProxSGD had a maximum of $20n$ update steps in order to achieve this accuracy and couldn't spend more than 50 epochs.



Comments (1)

One can observe that for this first example the utilization of mini-batches can significantly speed up the optimization process while still requiring reasonable accuracies ($\epsilon \simeq \frac{1}{n} \in [10^{-5}, 10^{-4}]$). The step-size schedule plays a non-negligible role in the convergence of the methods ! As we will see with the other experiments for first-era methods, the dependency on $\{\eta_t\}_{t \in \mathbb{N}}$ will remain one of the major bottlenecks of such class of stochastic optimization methods.

The most efficient scheme for the current experiment comes for the mini-batch SGD approach for $\hat{b} = 10$ or 50 or even 100. At $\hat{b} = 100$ one observes a well known phenomenon of *batch effectiveness squeeze*. The upper-bound on MSE's trace (that resumes here in the variance term since the updates are unbiased) doesn't decrease for $\hat{b} = 500$ even though we compute 5 times more sample gradients! However the effectiveness in terms of accuracy by epochs seems slightly better (hopefully) when taking such a big $\hat{b}$; indicating *that it's not only MSE's trace that plays a role* in the convergence of mini-batch SGD and more generally of first era methods.

The worked-out example is kept relatively small in order to be, at this stage, relatively tractable on personal computers. However one might see that already some optimizations, that are furthermore shut-down before convergence, can take up to a 20-25 minutes for a single run at a single batch size.... When dealing with datasets of about $n = 10^9$ data points, parallel computing is mandatory and one will try to pick up the smallest $\hat{b}$ (reducing communication costs between workers ... etc: see *Sparsified SGD with Memory* (Stich, Cordonnier, Jaggi., 2018)) leading to a same level of variance.

We can see that the variance is indeed reduced, at least by $\hat{b}$ for low values of $\hat{b}$; it seems to play a role in the number of update-steps needed for a given method to converge. The most efficient scheme in terms of the number of sample gradients computed considers $\hat{b} \in [10, 50]$.

We can conclude by saying that for their best step-size schedule, mini-batch versions of SGD outperform the standard SGD for such convex problems. The lower variance of the update-steps linked to the true gradient estimation gives the opportunity to use bigger step-sizes that are still not too aggressive for mini-batch versions of SGD.

Problem (2)

The second problem is about a *smoothed* version of the hinge loss SVM. Again the ℓ_1 -norm regularization has been used. Let's summarize the structure of the problem :

elements	expression/value
problem size (n, d)	$n = 50000$, $d = 15$
inner convexity enhancer λ_{in}	0
sample differentiable cost function ϕ_i	$([1 - \langle w, \tilde{x}_i \rangle]_+)^2 + \frac{\lambda_{\text{in}}}{2} \ w\ ^2$
regularization constant λ	$\frac{1}{n}$
regularizer $R(\cdot)$	$\lambda \ w\ _1$
step-size	$\eta_t = \frac{1}{\hat{L}(1 + \frac{Bt}{\sqrt{n}})}$ for multiple values of $B \geq 0$ (basis)

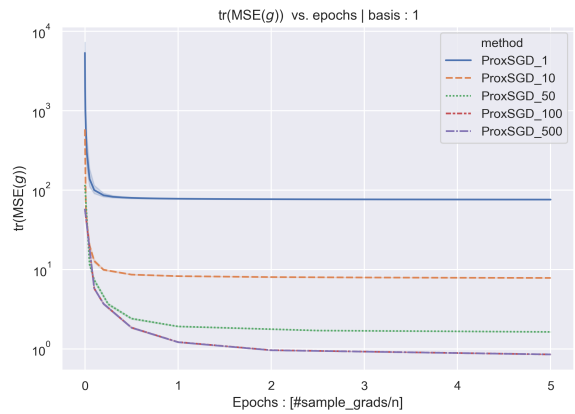
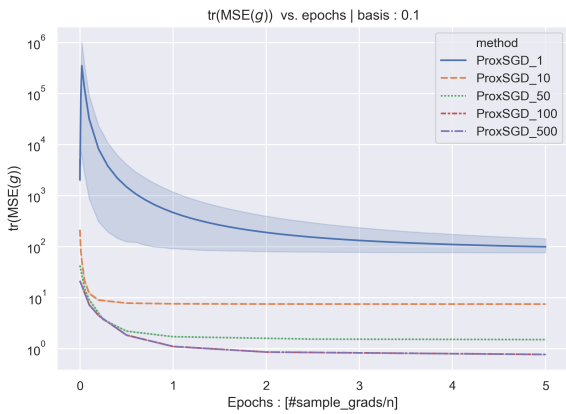
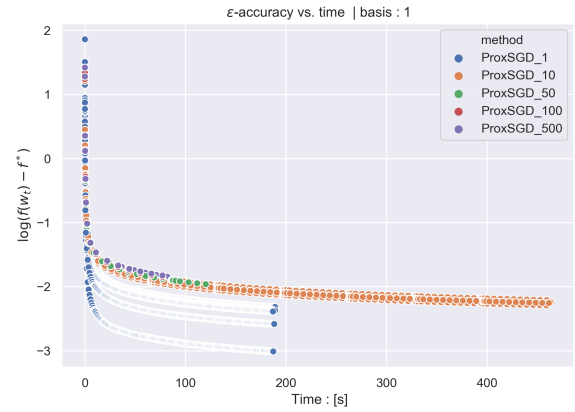
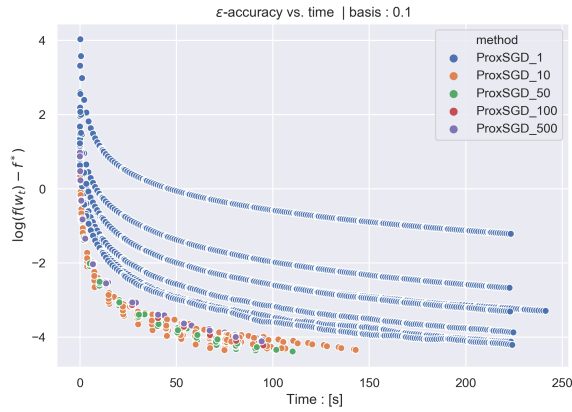
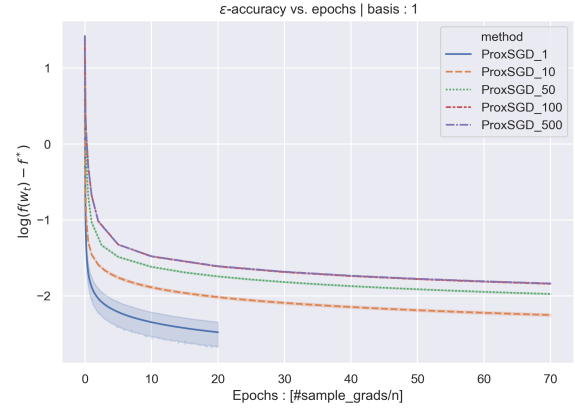
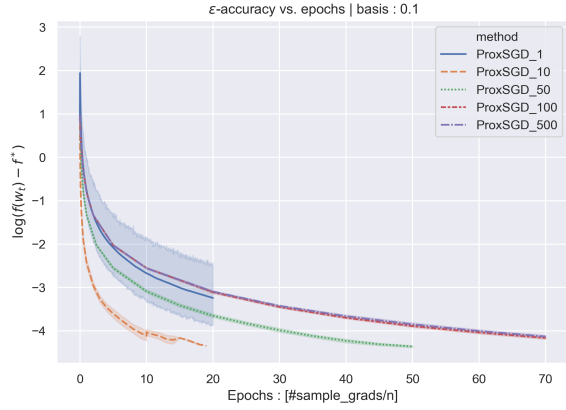
Note: This problem is certainly convex but non-smooth and *a priori* non-strongly convex. What differs the most with the last problem is that even the smoothness of ϕ is not guaranteed here and Lipschitz constants for the ϕ_i are not analytically available. Hence an overall $\hat{L}$ had to be estimated.

Results (2)

Each test has been conducted 7 times and the results were averaged in order to lower their intrinsic variance. Classical bootstrap confidence intervals are shown for the performances.

Note that the performance measure for ProxSGD is taken with $w_T \leftarrow \frac{1}{T} \sum_{i=0}^{T-1} w_t$.

The ϵ -accuracy targeted is $5 \cdot 10^{-5}$. The versions of Mini-Batch ProxSGD had a maximum of $20n$ update steps in order to achieve this accuracy and couldn't spend more than 50 epochs.



Comments (2)

Again comparing the best performances for classical SGD versus its mini-batch version shows the supremacy of this latter. However and more than previously we see the critical role played by the step-size scheduler. One can observe a large discrepancy in the results for classical SGD whereas those for the mini-batch version are far more concentrated around their mean.

The comments stated in *Comments (1)* seem to hold in this case even though ϕ is not L -smooth this time. What changes is the dramatic impact on the efficiency of the mini-batch versions of SGD for a non-suited step size schedule (here for $B = 1$).

(IS)-SGD

The paradigm of Importance Sampling (IS) came rather late in the big-data stochastic optimization field. To the best of my knowledge an extensive analysis of stochastic optimization methods with (IS) hadn't been conducted before 2014 with the forerunner paper "*Stochastic Optimization with Importance Sampling*" (Zhao & Zhang, 2014) [ZZ15]. The authors called their findings (IS)-SGD (or, when it does apply, (IS)-ProxSGD). Note that some second era methods were already born in 2014; e.g. the initial SVRG back in 2013. This comment underlines the fact that research for first/second/third era methods overlap in time !

Motivation:

While using the structure of first era stochastic gradient the paper cited above intended to tackle the problem of minimizing the trace of the variance term of the MSE. They provided a powerful analysis of convergence for first-era methods embedding a sampling for elements in S_t that is not uniform. This degree of freedom yields better theoretical bounds since non uniform sampling can degenerate into uniform sampling when there's no appeal to sample differently.

Despite its powerful theoretical analysis the paper only considers mini-batches of size 1 ; i.e. $|S_t| = 1$ with probability 1. Without loss of generality with respect to the formalism of first era methods, they did study the impact of the probabilities $\{p_{t,i}\}_{i=1}^n$ ($= \{p_{t,i}\}_{i=1}^n$ since $D = 1$ here as the sampling scheme might be seen as an independent sequential lottery where there's only a single lottery) present in the expression

$$i_t = i \text{ with probability } p_{t,i}$$

$$g_t = \frac{\nabla \phi_{i_t}(w_t)}{np_{t,i_t}}$$

$$\text{on the quantity } V_t := \mathbb{E}_{i_t}[\|g_t - \nabla \phi(w_t)\|_2^2] = \left(\frac{1}{n^2} \sum_{i=1}^n \frac{\|\nabla \phi_i(w_t)\|_2^2}{p_{t,i}} \right) - \|\nabla \phi(w_t)\|_2^2$$

Under L -smoothness assumption and without regularization ($f = \phi$) one can immediately motivate the use of $p_{t,i}$'s in order to speed-up classical SGD. Indeed, one would have

$$\begin{aligned} \mathbb{E}_{i_t}[f(w_{t+1})] &= \mathbb{E}_{i_t}[\phi(w_{t+1})] \\ &= \mathbb{E}_{i_t}[\phi(w_t - \eta_t \frac{\nabla \phi_{i_t}(w_t)}{np_{t,i_t}})] \\ &\leq \mathbb{E}_{i_t}[\phi(w_t) - \eta_t \langle \nabla \phi(w_t), \frac{\nabla \phi_{i_t}(w_t)}{np_{t,i_t}} \rangle + \eta_t^2 \frac{L}{2} \|\frac{\nabla \phi_{i_t}(w_t)}{np_{t,i_t}}\|^2] \\ &\leq \phi(w_t) - \eta_t \|\nabla \phi(w_t)\|^2 + \eta_t^2 \frac{L}{2} \left(\frac{1}{n^2} \sum_{i=1}^n \frac{\|\nabla \phi_i(w_t)\|^2}{p_{t,i}} \right) \end{aligned}$$

Hence, in the above setting, one can see that minimizing the variance term V_t with respect to the probabilities $p_{t,i}$ leads to minimizing the one-step expected value of the *isotropic* quadratic upper-bound of f !

In Chapter 5 the assumption $R = 0 \Leftrightarrow f = \phi$ will be relaxed and I will propose a more general proof of convergence than in [ZZ15].

It can be shown easily that the proper distribution $\mathcal{P}_t$ that minimizes the term $\frac{1}{n^2} \sum_{i=1}^n \frac{\|\nabla \phi_i(w_t)\|^2}{p_{t,i}}$ (hence V_t as well, since there's no one-step randomness for the term $-\|\nabla \phi(w_t)\|^2$) is the unique discrete distribution for which for every $i \in [n]$,

$$p_{t,i}^* = \frac{\|\nabla \phi_i(w_t)\|}{\sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|} \quad (4.13)$$

This result will follow from the demonstration given in Chapter 5 as well.

Immediately one can see that if the norms of the sample gradients are relatively uniform, IS presented as such will not lead to a major speed-up compared to SGD used with US.

One huge practical issue with such sampling probabilities is that it requires to compute all the sample gradients : **this is exactly what we would like to prevent !**

Zhao and Zhang came up with the idea of a relaxation to the problem of minimizing V_t with respect to the $p_{t,i}$. If one disposes of upper-bounds G_i on the values $\|\nabla \phi_i(w_t)\|$ ($i \in [n]$) then the proposal distribution would be constructed upon $\hat{p}_{t,i}^*$'s that minimize $\frac{1}{n^2} \sum_{i=1}^n \frac{G_i^2}{\hat{p}_{t,i}}$.

Trivially it comes then that

$$\hat{p}_{t,i}^* = \frac{G_i}{\sum_{i'=1}^n G_{i'}}$$

For instance, let's assume that the ϕ_i 's are L_i -Lipschitz on $\mathbb{R}^d$.

Then equivalently when ϕ is differentiable for any $w_t \in \mathbb{R}^d$, $\|\nabla \phi_i(w_t)\| \leq L_i$.

When the functions ϕ_i 's are L_i smooth then Zhao and Zhang suggest to use $G_i = L_i$ as well as long as the norm of the iterate w_t is bounded ; poor assumption.

Despite the attractive potential of importance sampling, one has to keep in mind that sampling non-uniformly might be time consuming ! At some point, especially when dealing with sampling without replacement, this extra sampling cost becomes non-negligible!

In the above special setting the one-step expected speed-up using the proposed sampling ($p_{t,i}^*$ for $i \in [n]$) depends on the quantity (ratio of the minimized term in V_t) :

$$R := \frac{(\frac{1}{n} \sum_{i=1}^n \|\nabla \phi_i(w_t)\|)^2}{\frac{1}{n} \sum_{i=1}^n \|\nabla \phi_i(w_t)\|^2}$$

If $R \simeq 1$, it will signify that IS is not worth it while a value close to 0 will be very attractive !

Algorithm

Algorithm 4 SGD with IS

Initialization : $\{\eta_t\}_{t \in \mathbb{N}}$ step-size schedule $\forall t \in \mathbb{N}, \eta_t > 0$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, univariate probability distribution oracle $\{\mathcal{P}_t\}_{t \in \mathbb{N}}$ on $[n]$,
 $t = 0$ update steps counter
while stopping criterion not verified **do**
 $S_t \sim (\mathcal{P}_t)^{\hat{b}}$ (with replacement)
 $g_t \leftarrow \frac{1}{\hat{b}} \sum_{i \in S_t} \frac{\nabla \phi_i(w_t)}{np_{t,i}}$ (with replacement)
 $w_{t+1} \leftarrow \text{prox}_{\eta_t R}(w_t - \eta_t g_t)$
 $t \leftarrow t + 1$
end while
Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note: Here a mini-batch version is proposed yet not studied in the original paper [ZZ15]. The principle is to apply a sequential independent lottery but under the same univariate distribution at each lottery : i.e. $p_{t,i} = p_{t,i,l}$ for every $l \in \{1, \dots, \hat{b}\}$. Hence $p_{t,i} \neq \mathbb{P}[i \in S_t]$ in this case! When dealing with sampling with replacement, $p_{t,i}$ will depict the expected number of times i will be part of S_t . That being said, drawing with replacement is at some point, especially when using non-uniform probabilities, less efficient in terms of the number of non identical indices taken in S_t than the case of a sampling without replacement. However it presents the advantage, as discussed at the beginning of this chapter, to lead easily to unbiased updates. Furthermore it's also done faster as no re-normalization is required every time a case is selected. When the probability oracle yields p_t^* as defined above SGD with IS refers to the true IS-SGD.

The non-tractable $p_{t,i}^*$'s are *adaptive* probabilities with the update iterations. The two practical examples given above embedding the L_i constants (Lipschitz and/or smoothness) are not anymore adaptive strategies. The main drawback for such approaches is that it might occur that those constants may badly reflect the current sample gradients norms hence leading to an even worse sampling than the uniform strategy. In Chapter 5 I will propose, as a door left open, a tractable adaptive sampling that will converge to uniform sampling.

Experiments

Let's run another personal experiment in order to assess the gravity of this problem.

Experiment (1)

The experiment conducted is exactly the same as the first experiment attempted to compare MB-SGD and SGD. The probability oracle fed to SGD with IS is the one suggested by Zhao and Zhang for the $\{L_i\}_{i=1}^n$, Lipschitz gradient constants : $\forall t \in \mathbb{N}, p_{t,i} = \frac{L_i}{\sum_{i'=1}^n L_{i'}}$.

However a small amount $\tilde{\epsilon}$ has been added to each L_i in order to achieve anyway $\min_{i \in [n]} p_{t,i} \geq \epsilon_p = \frac{10^{-3}}{n}$. In other words the minimal importance probability must be bigger than one thousandth of the uniform probability $\frac{1}{n}$.

Comments (1)

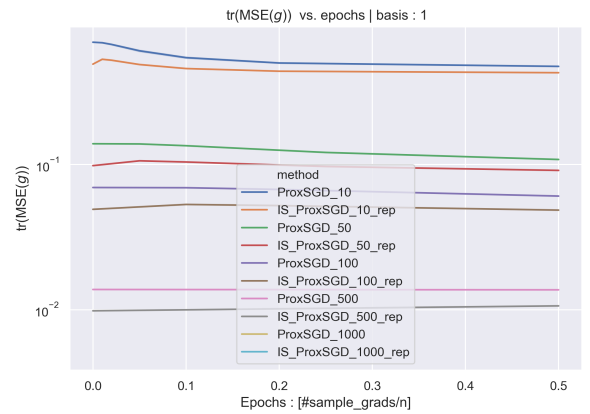
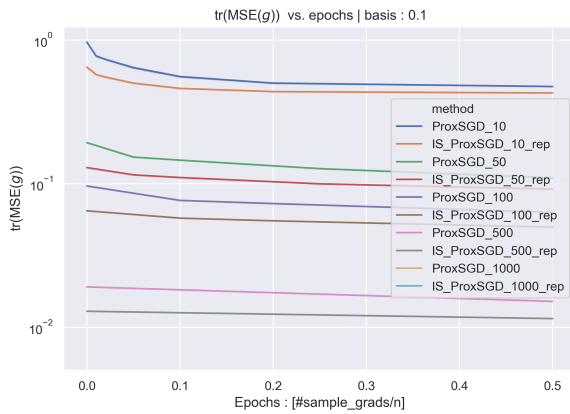
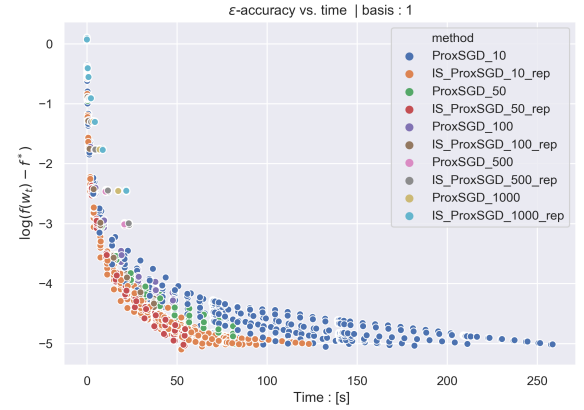
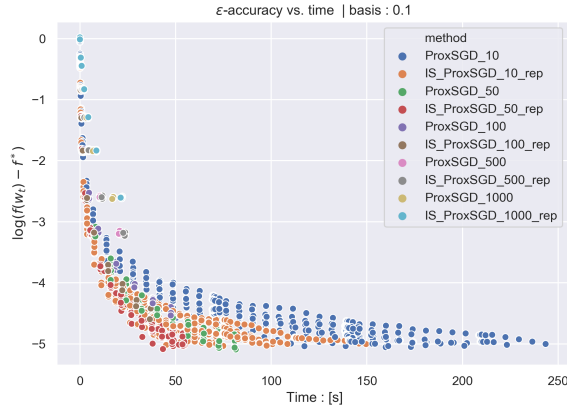
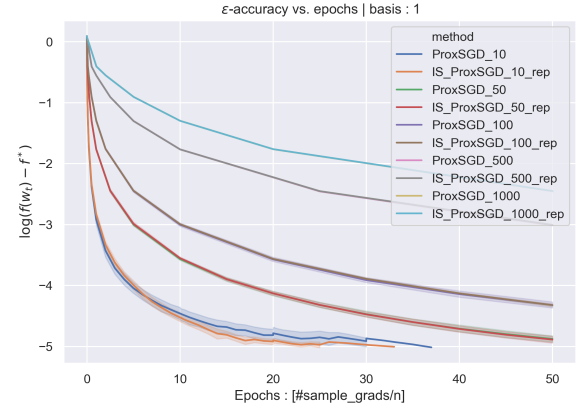
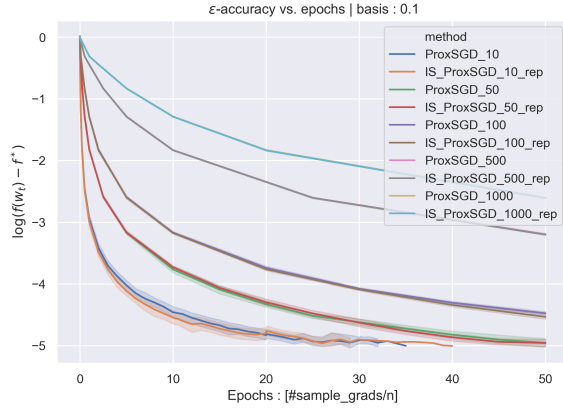
Referring to Results (1) right below one observes that indeed the non-adaptive character of our probability oracle doesn't spark appealing results ; SGD with IS barely beats (MB)-SGD. Zhao and Zhang have found some experimental settings in which the sampling with respect L_i 's yields an interesting speed-up but in most of their results the beneficial effect of NUS quickly vanishes disappointingly.

Results (1)

Each test has been conducted 10 times and the results were averaged in order to lower their intrinsic variance. Classical bootstrap confidence intervals are shown for the performances.

Note that the performance measure for (IS)-ProxSGD is taken with $w_T \leftarrow \frac{1}{T} \sum_{i=0}^{T-1} w_t$.

The ϵ -accuracy targeted is 10^{-5} . The versions of Mini-Batch IS-ProxSGD had a maximum of $20n$ update steps in order to achieve this accuracy and couldn't spend more than 50 epochs.



4.2 Variance Reduction Era

We have seen so far that the intrinsic variance of the update steps for first-era methods seem to play a role in the convergence of the investigated methods. The efficiency in terms of sample gradient computations was sometimes quite similar for mini-batch versions of SGD and the initial single sample version of SGD. However MB-SGD was clearly more time efficient : hence it took less update iterations to converge !

One straightforward comment to recall from 4.1.2 resides in the fact that first-era methods are unable to achieve

$$\lim_{t \rightarrow \infty} \text{tr}(\text{MSE}(g_t)) = 0$$

Indeed in what concerns the variance term, in general, one does not have any guarantee that

$$\lim_{t \rightarrow \infty} \mathbb{E}[||g_t||^2 | \mathcal{F}_t] = || \lim_{t \rightarrow \infty} \mathbb{E}[g_t | \mathcal{F}_t] ||^2$$

Of course $\lim_{t \rightarrow \infty} |S_t| < n$ otherwise the randomness source of the update-steps vanishes trivially since we would actually apply plain proximal gradient update steps.

As an example let's consider SGD and let's imagine that $\lim_{t \rightarrow \infty} w_t = w_* \in S^{*4}$

For $S_t \neq [n]$,

$$\begin{aligned} \lim_{t \rightarrow \infty} \mathbb{E}[||g_t||^2 | \mathcal{F}_t] &= \lim_{t \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n ||\nabla \phi_i(w_t)||^2 \\ &= \frac{1}{n} \sum_{i=1}^n ||\nabla \phi_i(w_*)||^2 \\ &\neq ||\nabla \phi(w_*)||^2 \\ &= ||\frac{1}{n} \sum_{i=1}^n \nabla \phi_i(w_*)||^2 \\ &= || \lim_{t \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(w_t) ||^2 \\ &= || \lim_{t \rightarrow \infty} \mathbb{E}[g_t | \mathcal{F}_t] ||^2 \end{aligned}$$

This *disability* to decrease its variance during update-steps prevents first-era methods to enjoy linear rates of convergence with a step-size schedule that is constant. As highlighted in the next chapter (5) the only possibility for first-era methods to converge is then to lower η_t progressively in order to ensure that

$$\lim_{t \rightarrow \infty} \eta_t V_t = 0$$

Unfortunately this brings first-era methods to converge sub-linearly. The core idea behind second-era methods is to use previously computed sample gradients in order to estimate more accurately the true current gradient and, eventually when $t \rightarrow \infty$, having that $V_t \rightarrow 0$.

Note : It doesn't mean that either $|S_t|$ becomes larger and larger until $|S_t| \rightarrow n$ nor that we do not use stochastic-updates anymore. For any $t \in \mathbb{N}$, V_t will remain strictly positive but it will become arbitrarily small as the iterates w_t converge towards a stationary point $w_* \in S^*$.

The reader should know that, as the name suggests it, the second-era aims to further reduce $\text{tr}(\text{MSE}(g_t))$ (or more simply when the updates are unbiased : $\text{tr}(\text{MSE}(g_t)) = V_t$).

⁴The set of stationary points for problem (1.2).

Before digging into the theoretical analysis of the stochastic gradient g_t two examples will be given in order to clarify everything. A little *detour* into the variance reduction framework for stochastic simulations is necessary to fully understand which concept is hidden behind second-era methods.

Control Variate Technique

This subsection is inspired from the introduction of *SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives* (Defazio et al., 2014) (SAGA) [DBL14] and a course at EPFL (stochastic simulations : MATH414) ⁵.

Let's assume one disposes of a random variable $\chi \in \mathbb{R}^d$ for which the expectation $\mathbb{E}[\chi]$ is unknown and desired. When generating $\hat{b}$ *i.i.d* samples of χ , namely $\{\chi_i\}_{i=1}^{\hat{b}}$, is possible the crude *Monte-Carlo* technique in order to estimate $\bar{\chi} := \mathbb{E}[\chi]$ is to average the generated samples

$$\hat{\chi}_{CMC} = \frac{1}{\hat{b}} \sum_{i=1}^{\hat{b}} \chi_i$$

This estimator ($\hat{\chi}$) is an unbiased estimator of $\bar{\chi}$ for any value of $\hat{b}$. However the biggest $\hat{b}$ is the more accurately $\bar{\chi}$ will be approached ! Now let's imagine that we also have access to an auxiliary random variable Z taking values in $\mathbb{R}^c$ for which the expectation is this time known $\bar{Z} = \mathbb{E}[Z]$. If we can also generate $\hat{b}$ *i.i.d* samples from Z then for any matrix $\gamma \in \mathbb{R}^{d \times c}$, independent from $\{\chi_i\}_{i=1}^{\hat{b}}$ and $\{Z_i\}_{i=1}^{\hat{b}}$,

$$\hat{\chi}_{CV} = \frac{1}{\hat{b}} \sum_{i=1}^{\hat{b}} \chi_i + \gamma(Z_i - \bar{Z})$$

is called a *control variate* estimator of $\bar{\chi}$. This estimator is unbiased as well.

$$\mathbb{E}[\hat{\chi}_{CV}] = \frac{1}{\hat{b}} \sum_{i=1}^{\hat{b}} \mathbb{E}[\chi_i + \gamma(Z_i - \bar{Z})] = \frac{1}{\hat{b}} \left[\hat{b}(\mathbb{E}[\chi] + \mathbb{E}[Z - \bar{Z}]) \right] = \mathbb{E}[\chi]$$

There's an interest in using a *control variate* Z as long as Z is correlated with χ . In that case we can tune γ in order to obtain a *control variate* estimator of $\bar{\chi}$ such that

$$\mathcal{V}(\hat{\chi}_{CMC}) - \mathcal{V}(\hat{\chi}_{CV}) = \mathbb{E}[(\hat{\chi}_{CMC} - \bar{\chi})(\hat{\chi}_{CMC} - \bar{\chi})^T] - \mathbb{E}[(\hat{\chi}_{CV} - \bar{\chi})(\hat{\chi}_{CV} - \bar{\chi})^T] \succeq \mathbf{0}$$

where $\mathcal{V}[\cdot]$ depicts the (co)variance matrix of a random variable.

Indeed one may write

$$\begin{aligned} \min_{\gamma \in \mathbb{R}^{d \times s}} \mathcal{V}[\hat{\chi}_{CV}(\gamma)] &= \frac{1}{\hat{b}} \min_{\gamma \in \mathbb{R}^{d \times s}} \mathcal{V}[\chi + \gamma(Z - \bar{Z})] \\ &= \frac{1}{\hat{b}} \min_{\gamma \in \mathbb{R}^{d \times s}} \mathcal{V}[\chi + \gamma Z] \\ &= \frac{1}{\hat{b}} \min_{\gamma \in \mathbb{R}^{d \times s}} \mathcal{V}[\chi] + \gamma \mathcal{V}[Z] \gamma^T + 2\gamma \mathcal{C}[\chi, Z] \\ &\leq \frac{1}{\hat{b}} \mathcal{V}[\chi] = \mathcal{V}[\hat{\chi}_{CMC}] \end{aligned}$$

⁵Source: <http://edu.epfl.ch/coursebook/en/stochastic-simulations-MATH-414>

Choosing $\gamma = \gamma^* = -(\mathcal{V}[Z]^{-1}\mathcal{C}[Z, \chi])^T$ ensures the minimum variance for a control variate estimator of $\bar{\chi}$ given Z , its known mean and the fixed sample size $\hat{b}$.

This optimal choice would lead to :

$$\begin{aligned} \min_{\gamma \in \mathbb{R}^{d \times c}} \mathcal{V}[\hat{\chi}_{CV}(\gamma)] &= \frac{1}{\hat{b}} \left(\mathcal{V}[\hat{\chi}_{CV}(\gamma^*)] \right) \\ &= \frac{1}{\hat{b}} \left(\mathcal{V}[\chi] - \mathcal{C}[\chi, Z] \mathcal{V}[Z]^{-1} \mathcal{C}[Z, \chi] \right) \\ &= (I_d - R_{\chi, Z}^2) \frac{\mathcal{V}[\chi]}{\hat{b}} = (I_d - R_{\chi, Z}^2) \mathcal{V}[\hat{\chi}_{CMC}] \preceq \mathcal{V}[\hat{\chi}_{CMC}] \end{aligned}$$

where $\mathbf{0} \preceq R_{\chi, Z}^2 = \mathcal{C}[\chi, Z] \mathcal{V}[Z]^{-1} \mathcal{C}[Z, \chi] \mathcal{V}[\chi]^{-1} \preceq I_d$ is the multidimensional equivalent of the correlation coefficient. $(I_d - R_{\chi, Z}^2)$ depicts the variance reduction term. The closer this latter matrix is defined to $\mathbf{0}$ the more effective is the variance reduction.

There is still research undergoing for state of the art techniques commonly referred as *JacSketch* [GRB01] methods. These latter try to build a Jacobian matrix constituted by sample gradients using implicitly a matrix control variate as above with γ .

The classical SVRG like methods (listed below in the *Methods* subsection) do not use a matrix $\gamma \in \mathbb{R}^{d \times c}$ but rather a single constant $\gamma \in \mathbb{R}$. Furthermore this constant is not chosen in order to minimize $\mathcal{V}[\hat{\chi}_{CV}]$ but fixed, usually taking the value 1 (hence it's equivalent to fix $\gamma = I_d$). This choice might seem quite simple and, indeed, it is. For a general estimation of $\mathbb{E}[\chi]$ it does not even guarantee that $\mathcal{V}[\hat{\chi}_{CV}]$ will be smaller compared to $\mathcal{V}[\hat{\chi}_{CMC}]$!

Why would it work then?

Let's already discover the initial SVRG⁶ stochastic gradient structure. Given what we will call a *snapshot*, that is a point $\tilde{w} \in \mathbb{R}^d$ where first-order information for at least one sample gradient has been recorded, and an update iteration $t \in \mathbb{N}$,

$$g_t^{\text{SVRG}} = \nabla \phi_{i'}(w_t) - \nabla \phi_{i'}(\tilde{w}) + \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\tilde{w}) \quad \text{where } i' \sim \mathbf{U}([n])$$

One can verify that $\mathbb{E}_{i'}[g_t^{\text{SVRG}}] = \nabla \phi(w_t)$. Actually SVRG's stochastic gradient is the control variate estimate of $\chi = \nabla \phi(w_t)$ where $\hat{b} = 1$, $\chi_1 = \nabla \phi_{i'}(w_t)$, $Z_1 = \nabla \phi_{i'}(\tilde{w})$ and $\bar{Z} = \mathbb{E}_{i'}[Z] = \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\tilde{w})$ and finally, $\gamma = I_d$ or $\gamma = 1$.

The point is that when the used snapshot, $\tilde{w}$, is updated in such a way that the distance $d(w_t, \tilde{w})$ tends to 0 when t increases, we have that for every $i \in [n]$, $d(\nabla \phi_i(w_t), \nabla \phi_i(\tilde{w})) \rightarrow 0$ as well under the smoothness assumption of the sample gradients. Hence the stochastic gradient g_t^{SVRG} would ultimately be arbitrarily close to $\frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\tilde{w})$ (itself arbitrarily close from $\nabla \phi(w_t)$ then) which's a purely deterministic quantity yielding no variance at all!

As I will investigate in Chapter 5 the early update steps of SVRG are the ones for which the variance of the stochastic gradient is eventually bigger than in SGD paving the way through a mixed method passing from one regime (SGD-like) to the other (SVRG-like). Indeed a few update steps after the very first snapshot $\tilde{w}$ taken, $\gamma = 1$ might not be giving variance reduction with respect to the crude Monte-Carlo estimator represented by SGD's stochastic gradient.

⁶Source: *Accelerating Stochastic Gradient Descent using Predictive Variance Reduction* (Johnson & Zhang, 2013) [JZ13]. Further comments will be added in the *Methods* subsection.

The core idea behind SVRG can be theoretically further strengthened while using several snapshots instead of one. At any update iteration $t \in \mathbb{N}$ one may pair each case $i \in [n]$ to a snapshot $\theta_{t,i} \in \mathbb{R}^d$. As long as all the $\theta_{t,i}$ converge to w_* the stochastic gradient $g_t = \nabla \phi_{i'}(w_t) - \nabla \phi_{i'}(\theta_{t,i'}) + \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\theta_{t,i})$ is also unbiased for $i' \sim \mathbf{U}([n])$.

Such a stochastic gradient was first proposed in *A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives (SAGA)* (Defazio, Bach, Lacoste-Julien, 2014) [DBL14]. We will discover how it is implemented in practice and what are the main differences between SVRG-like methods and SAGA-like methods.

4.2.1 General Scheme

Here below is given a personal definition of how second era methods work. One full update step is presented, shedding light on my general expression of their stochastic gradient (4.15).

2nd ERA : at any $t \in \mathbb{N}$, conditional on $w_t \in \mathbb{R}^d$ and *snapshots* $\{\theta_{t,i}\}_{i=1}^n$, γ_t

$$S_t \sim \mathcal{P}_t \quad (4.14)$$

$$g_t = \underbrace{\Lambda(S_t) \sum_{i \in S_t} m_{t,i,l(i)} \nabla \phi_i(w_t)}_{\text{first-era part}} + \underbrace{\gamma_t \left(\frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) - \Lambda(S_t) \sum_{i \in S_t} m_{t,i,l(i)} \nabla \phi_i(\theta_{t,i}) \right)}_{\text{control variate part}} \quad (4.15)$$

$$w_{t+1} = \text{prox}_{\eta_t R}(w_t - \eta_t g_t) \quad (4.16)$$

Note 1: Let's just repeat that the goal of (4.14) is only to build the set of indices S_t that will define which sample gradients must be computed at update iteration $t \in \mathbb{N}$. As a reminder the subscript $l(i)$ indicates the lottery l at which the indice i has been added to S_t ; if this subscript is dropped either there is a single lottery in the implementation of (4.14) or the *weighting coefficients* $m_{t,i,l}$ are the same for every lottery $l \in \{1, \dots, D\}$.

Note 2: The purpose of γ_t has been stated in the introducing passage of this section. If γ_t is set to 0 we get rid of the latent control variate part of the stochastic gradient and we fall back on first-era methods.

4.2.2 Analysis of the Stochastic Gradient

Since the different ways in order to build/draw S_t have been extensively studied in 4.1.2, these won't be exposed again. Let's keep the very same notations to derive lemmas about the bias and the variance of second-era methods.

Expectation

Again, let's briefly analyze the bias of g_t to derive conditions for unbiasedness.

1. **Natural:** S_t is drawn according to a direct probability distribution $\mathcal{P}_t$. Again we assume that one must then know the mapping $\mathcal{M}_t : 2^{[n]} \rightarrow \mathbb{R}^+$, $S_t \subseteq [n] \rightarrow \mathcal{M}_t(S_t) := \mathbb{P}[S_t] \geq 0$ that assigns the "one-step" probability (following $\mathcal{P}_t$'s distribution) of realization of each subset S_t of indices from $[n]$. One can immediately write

$$\mathbb{E}[g_t | \mathcal{F}_t] = \frac{\gamma_t}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) + \sum_{S_t \in 2^{[n]}} \mathcal{M}_t(S_t) \Lambda(S_t) \left[\sum_{i \in S_t} m_{t,i} \left(\nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i}) \right) \right] \quad (4.17)$$

Lemma 4 (ERA 2, first sampling scheme : sufficient condition for unbiased updates): Consider the stochastic gradient g_t as built in (4.15) with the first sampling scheme defined by $\mathcal{M}_t$. Let's suppose that $\forall i \in [n]$, $m_{t,i} = 1$ and that $\forall \hat{b} \in \{1, \dots, n\}$, one either verifies that $\mathcal{M}_t(S_t) > 0 \forall S_t \in 2^{[n], \hat{b}}$ or $\mathcal{M}_t(S_t) = 0 \forall S_t \in 2^{[n], \hat{b}}$. Let's denote by K the set of batch sizes $\hat{b}$ such that every subset of size $\hat{b} \in K$ has a strictly positive probability to be drawn.

The choice

$$\Lambda(S_t) = \begin{cases} \frac{1}{|K|n \binom{n-1}{|S_t|-1} \mathcal{M}_t(S_t)} & \text{if } \forall S_t \in 2^{[n], |S_t|}, \mathcal{M}_t(S_t) > 0 \\ 0 & \text{otherwise} \end{cases}$$

ensures that $\mathbb{E}[g_t | \mathcal{F}_t] = \nabla \phi(w_t)$

Proof : Under our hypothesis, let's rewrite the equation (4.17) in a more convenient way

$$\mathbb{E}[g_t | \mathcal{F}_t] = \frac{\gamma_t}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) + \sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n], \hat{b}}} \mathcal{M}_t(S_t) \Lambda(S_t) \left[\sum_{i \in S_t} m_{t,i} \left(\nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i}) \right) \right]$$

Using the results of **Lemma 1**, we can already state that, under our choice of $\Lambda(\cdot)$,

$$\mathbb{E}[g_t | \mathcal{F}_t] = \sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n], \hat{b}}} \mathcal{M}_t(S_t) \Lambda(S_t) \left(\sum_{i \in S_t} m_{t,i} \nabla \phi_i(w_t) \right) = \nabla \phi(w_t)$$

Then it remains to show that $\sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n], \hat{b}}} \mathcal{M}_t(S_t) \Lambda(S_t) \left(\sum_{i \in S_t} \nabla \phi_i(\theta_{t,i}) \right) = \frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'})$. Applying the above choice for $\Lambda(\cdot)$ leads to

$$\sum_{\hat{b} \in K} \sum_{S_t \in 2^{[n], \hat{b}}} \frac{1}{|K|n \binom{n-1}{|S_t|-1}} \left(\sum_{i \in S_t} \nabla \phi_i(\theta_{t,i}) \right) = \sum_{\hat{b} \in K} \frac{1}{|K|n \binom{n-1}{\hat{b}-1}} \sum_{S_t \in 2^{[n], \hat{b}}} \sum_{i \in S_t} \nabla \phi_i(\theta_{t,i})$$

Here the important thing to notice is that the term $\sum_{S_t \in 2^{[n], \hat{b}}} \sum_{i \in S_t} \nabla \phi_i(\theta_{t,i})$ is actually equivalent to $\binom{n-1}{|S_t|-1} \sum_{i=1}^n \nabla \phi_i(\theta_{t,i})$. Among the $\binom{n}{|S_t|}$ subsets of size $|S_t|$, a single indice i will be present in $\binom{n-1}{|S_t|-1}$ of them ! Then we are left with

$$\sum_{\hat{b} \in K} \frac{1}{|K|n} \sum_{i=1}^n \nabla \phi_i(\theta_{t,i}) = \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\theta_{t,i})$$

□

The extension of the easy-to-read examples discussed in 4.1.2 is articulated as follows. Under the assumptions of the above lemma. If $K = \{1\}$, we have $\Lambda(S_t) = \frac{1}{n \mathcal{M}_t(S_t)}$ for any singleton $S_t = \{i\}$ ($i \in [n]$) and $\Lambda(S_t) = 0$ for any other subset of $[n]$. Actually in that case we have the equivalence $\mathcal{M}_t(\{i\}) = \mathbb{P}[S_t = \{i\}] = p_{t,i}$ and the stochastic gradient g_t simply boils down to

$$g_t = \frac{\nabla \phi_i(w_t)}{n p_{t,i}} - \gamma_t \frac{\nabla \phi_i(\theta_{t,i})}{n p_{t,i}} + \frac{\gamma_t}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) \quad (4.18)$$

and eventually while using Uniform Sampling (US), $p_{t,i} = \frac{1}{n}$ at any iteration t

$$g_t = \nabla \phi_i(w_t) - \nabla \phi_i(\theta_{t,i}) + \frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) \quad (4.19)$$

The reader may recognize that this last g_t corresponds to SAGA for $\gamma_t = 1$.

More generally, when $K = \{\hat{b}\}$ and still under uniform sampling on $2^{[n],\hat{b}}$, the stochastic gradient corresponds to the natural mini-batch extension of SAGA (again $\gamma_t = 1$) :

$$g_t = \frac{1}{\hat{b}} \left(\sum_{i \in S_t} \nabla \phi_i(w_t) \right) - \nabla \phi_i(\theta_{t,i}) + \frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'})$$

2. **Independent sequential lotteries on $\{0\} \cup [n]$:** in what concerns this sampling strategy, it was previously assumed that Λ was a constant hence not depending on S_t . With $\mathbb{P}[i \text{ is added to } S_t \text{ at } l\text{'s independent lottery}] := p_{t,i,l}$

$$\mathbb{E}[g_t | \mathcal{F}_t] = \frac{\gamma_t}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) + \sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \left(\nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i}) \right) \quad (4.20)$$

Lemma 2 still holds in this context ensuring an unbiased update.

Lemma 5 (ERA 2, second sampling scheme : sufficient condition for unbiased updates): Consider the stochastic gradient g_t as built in (4.15) with the second sampling scheme defined by the probabilities $p_{t,k,l}$ for any $k \in \{0, \dots, n\}$, $l \in \{1, \dots, D\}$. Let's suppose that $\Lambda(S_t) = \Lambda > 0$ for any $S_t \in 2^{[n]}$. For every $i \in [n]$, let $\zeta_i := \{l \in \{1, \dots, D\} | p_{t,i,l} > 0\}$ and $\Lambda = \frac{1}{D}$. If for every set ζ_i , $|\zeta_i| > 0$ then the choice

$$m_{t,k,l} = \begin{cases} \frac{D}{n |\zeta_k| p_{t,k,l}} & p_{t,k,l} > 0 \\ 0 & \text{otherwise} \end{cases}$$

ensures that $\mathbb{E}[g_t | \mathcal{F}_t] = \nabla \phi(w_t)$

Proof: Since the assumptions of the present lemma are the same as in Lemma 2 we already have that

$$\sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(w_t) = \nabla \phi(w_t)$$

It remains to show that $\sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(\theta_{t,i}) = \frac{1}{n} \sum_{i=1}^n \nabla \phi_{i'}(\theta_{t,i})$

Using the same proof structure as in **Lemma 2** :

$$\begin{aligned} \sum_{i=1}^n \left(\Lambda \sum_{l=1}^D p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(\theta_{t,i}) &= \frac{1}{D} \sum_{i=1}^n \left(\sum_{l \in \zeta_i} p_{t,i,l} m_{t,i,l} \right) \nabla \phi_i(\theta_{t,i}) \\ &= \frac{1}{D} \sum_{i=1}^n \left(\sum_{l \in \zeta_i} \frac{D}{n |\zeta_i|} \right) \nabla \phi_i(\theta_{t,i}) \\ &= \frac{1}{n} \sum_{i=1}^n \nabla \phi_i(\theta_{t,i}) \end{aligned}$$

□

Trace of MSE

The analysis performed for first-era methods is still valid for second-era methods. The reader should recall that it began with the assumption that each indice is at most taken once in S_t and that only a single $m_{t,i,l} = m_{t,i}$ is non-zero and that $\Lambda(\cdot)$ has to be constant. Let's summarize a few facts adapting results from first-era methods ; i.e. the asymptotical vanishing MSE !

Let's rewrite the generic stochastic gradient for second-era under the above hypothesis :

$$\begin{aligned}
g_t &= \Lambda \sum_{i \in S_t} m_{t,i} \nabla \phi_i(w_t) + \gamma_t \left(\frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) - \Lambda \sum_{i \in S_t} m_{t,i} \nabla \phi_i(\theta_{t,i}) \right) \\
&= \sum_{i=1}^n \frac{\tilde{m}_{t,i} \left(\nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i}) \right)}{np_{t,i}} 1_{i \in S_t} + \frac{\gamma_t}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) \\
&= \sum_{i=1}^n \frac{\tilde{m}_{t,i} \beta_{t,i}^{\gamma_t}}{np_{t,i}} 1_{i \in S_t} + \gamma_t \bar{\alpha}_t \\
g_t &= \sum_{i=1}^n \frac{\tilde{m}_{t,i} \beta_{t,i}^{\gamma_t} + \gamma_t \bar{\alpha}_t}{np_{t,i}} 1_{i \in S_t} \tag{4.21}
\end{aligned}$$

where $\bar{\alpha}_t := \frac{1}{n} \sum_{i'=1}^n \nabla \phi_{i'}(\theta_{t,i'}) = \frac{1}{n} \sum_{i'=1}^n \alpha_{t,i'}$ and $\beta_{t,i}^{\gamma_t} = \nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i})$.

Recall that $\mathbb{E}[1_{i \in S_t} | \mathcal{F}_t] = p_{t,i} = \mathbb{P}[i \in S_t]$. In order to match the notations of **Lemma 3**, this time

$$\chi_{t,i} = \tilde{m}_{t,i} \beta_{t,i}^{\gamma_t} + \gamma_t \bar{\alpha}_t \quad \forall i \in [n]$$

Immediately it follows from the definitions given that

$$\mathbb{E}[g_t | \mathcal{F}_t] = \sum_{i=1}^n \frac{\chi_{t,i}}{np_{t,i}} \mathbb{E}[1_{i \in S_t} | \mathcal{F}_t] = \frac{1}{n} \sum_{i=1}^n \chi_{t,i} = \bar{\chi}_t = \left[\frac{1}{n} \sum_{i=1}^n \tilde{m}_{t,i} \nabla \phi_i(w_t) \right] + \gamma_t \left[\frac{1}{n} \sum_{i=1}^n (1 - \tilde{m}_{t,i}) \nabla \phi_i(\theta_{t,i}) \right]$$

Then under the assumptions of **Lemma 3**, we can give the following generic upper-bound on the trace of the MSE for second-era methods :

$$\begin{aligned}
\text{tr}(\text{MSE}(g_t)) &= \|\mathbb{E}[g_t | \mathcal{F}_t] - \nabla \phi(w_t)\|_2^2 + \mathbb{E}[\|g_t - \mathbb{E}[g_t | \mathcal{F}_t]\|_2^2 | \mathcal{F}_t] \\
&\leq \frac{1}{n^2} \left\| \sum_{i=1}^n (\tilde{m}_{t,i} - 1) \left(\nabla \phi_i(w_t) - \gamma_t \nabla \phi_i(\theta_{t,i}) \right) \right\|_2^2 + \frac{1}{n^2} \sum_{i=1}^n \frac{v_{t,i}}{p_{t,i}} \|\chi_{t,i}\|_2^2 \\
&= \frac{1}{n^2} \left\| \sum_{i=1}^n (\tilde{m}_{t,i} - 1) \beta_{t,i}^{\gamma_t} \right\|_2^2 + \frac{1}{n^2} \sum_{i=1}^n \frac{v_{t,i}}{p_{t,i}} \|\tilde{m}_{t,i} \beta_{t,i}^{\gamma_t} + \gamma_t \bar{\alpha}_t\|_2^2
\end{aligned}$$

Valid vectors v_t are defined as for first-era methods and depend on the sampling technique.

Let's suppose that for $t \rightarrow \infty$, γ_t tends to 1 and all the snapshots $\theta_{t,i}$ converge to $\lim_{t \rightarrow \infty} w_t = w_* \in S^*$, then $\forall i \in [n]$, $\lim_{t \rightarrow \infty} \beta_{t,i}^{\gamma_t} = \nabla \phi_i(w_*) - 1 \cdot \nabla \phi_i(w_*) = \mathbf{0}$. Furthermore, we have that $\lim_{t \rightarrow \infty} \gamma_t \bar{\alpha}_t = \nabla \phi(w_*)$. If $R = 0$ for problem (1.2) then $\nabla \phi(w_*) = 0$ and the upper-bound on the trace of the MSE for the above second-era stochastic gradient vanishes asymptotically ! Note that it can be biased ($\tilde{m}_{t,i} \neq 1$ for all $i \in [n]$ usually leads to a biased g_t) but still achieve an asymptotical zero-MSE. This was not possible for a first-era g_t .

If $R \neq 0$, we don't have $\nabla \phi(w_*) = \mathbf{0}$ in general. Nevertheless we dealt so far with an upper-bound for $\text{tr}(\text{MSE}(g_t))$. One may see with ease that under the assumption of the last paragraph ($\gamma_t \rightarrow 1$, $\theta_{t,i} \rightarrow w_*$ for all $i \in [n]$) g_t tends to $\bar{\alpha}_t$. Hence it loses all its stochasticity!

The main take-away messages from the analysis of the defined second-era stochastic gradient are :

1. Respectively Lemma 4 and 5 provide similar conditions for unbiasedness to resp. Lemma 1 and 2.
2. When second-era methods converge, the quantity $\text{tr}(\text{MSE}(g_t)) \rightarrow 0$.

4.2.3 Methods

The goal of this subsection is again to review useful existing techniques that fall in the second-era framework built above.

SAG

SAG as introduced in [SLB13] really represented the transition between first and second era methods. It's a special case of second era method for which the sampling is operated uniformly, all the *weighting coefficients* $\tilde{m}_{t,i}$ are equal to $\frac{1}{n}$ and $\gamma_t = 1$ for (4.21).

Motivation:

Unlike to the traditional SGD, SAG is motivated by the idea of using the latest information one has about a given sample gradient. For instance whenever a sample gradient $\nabla\phi_i(w_t)$ is computed since the linked indice, i , has been selected in S_t , the snapshot θ for i is set to w_t . Given an update iteration let's imagine we dispose of $\{\theta_{t,i}\}_{i=1}^n$ (hence the term $\bar{\alpha}_t = \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\theta_{t,i})$ is available) then a $\hat{b}$ -size mini-batch S_t of new indices is drawn and we set g_t as

$$g_t = \bar{\alpha}_t - \frac{1}{n} \sum_{i \in S_t} \nabla\phi_i(\theta_{t,i}) + \frac{1}{n} \sum_{i \in S_t} \nabla\phi_i(w_t)$$

$$\forall i \in S_t, \theta_{t+1,i} = w_t \quad \forall i \in [n] \setminus S_t, \theta_{t+1,i} = \theta_{t,i}$$

The updates of the snapshots is equivalent to set $\bar{\alpha}_{t+1} = g_t$. Such a stochastic gradient is biased ! However the algorithm proposed below has been shown to converge in expectation in the original paper although the proof is known to be rather complicated.

The reason why I judge SAG as being the transition from first to second-eras is that at the end of the day $g_t = \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\theta_{t+1,i})$. Its stochastic gradient expressed as so might make the reader thinking about first-order stochastic gradients: just a direct average of sample gradients. The control variate part is not explicit. The only difference is the subscript $t + 1$.

Algorithm

Algorithm 5 SAG

Initialization : $\eta_t = \eta > 0 \ \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla\phi_i(w_0)$: *warm start* or **0**)

while stopping criterion not verified **do**

$S_t \sim \mathbf{U}(2^{[n],\hat{b}})$ (without replacement) or $S_t \sim \mathbf{U}([n]^{\hat{b}})$ (with replacement)

$g_t \leftarrow \bar{\alpha}_t + \frac{1}{n} \sum_{i \in S_t} \nabla\phi_i(w_t) - \alpha_{t,i}$ (for both settings, with/without replacement)

$w_{t+1} \leftarrow w_t - \eta_t g_t$

$\alpha_{t+1,i} = \nabla\phi_i(w_t)$ if $i \in S_t$, $\alpha_{t+1,i} = \alpha_{t,i}$ if $i \notin S_t$ and then $t \leftarrow t + 1$

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

SAGA

SAGA [DBL14] comes on stage right after SAG since both methods are very close from each other. However chronologically speaking SVRG actually came first (2013). SAGA is the direct extension of SAG that admits a well-studied proximal version, unbiased updates, theoretical support for non-convex ϕ . The only difference with SAG resides in the coefficients $\tilde{m}_{t,i}$ in (4.21) that are equal to $\frac{1}{\hat{b}}$ for SAGA under uniform sampling.

Motivation :

Defazio et al. came back in 2014 to propose a new version of SAG that was unbiased ($\mathbb{E}[g_t|\mathcal{F}_t] = \nabla\phi(w_t)$). Its theoretical convergence analysis has been made way more easy than the convergence analysis of SAG thanks to this unbiasedness and relative simplicity of g_t .

They also provided theoretical support for the non-convex case, even for the proximal setting. Alongside their introducing passage in the paper the authors highlighted clearly that SAGA is in line with the class of control variate variance reducing methods !

Nothing changes with respect to SAG excepted that now for a same $\hat{b}$ -size mini batch S_t of new indices, g_t is built as

$$g_t = \bar{\alpha}_t - \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla\phi_i(\theta_{t,i}) + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla\phi_i(w_t)$$

The snapshots are updated as in SAG. This means that $\bar{\alpha}_{t+1} \neq g_t$.

Algorithm

Algorithm 6 SAGA

Initialization : $\eta_t = \eta > 0 \ \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla\phi_i(w_0)$: *warm start* or **0**)
while stopping criterion not verified **do**
 $S_t \sim \mathbf{U}(2^{[n],\hat{b}})$ (without replacement) or $S_t \sim \mathbf{U}([n]^{\hat{b}})$ (with replacement)
 $g_t \leftarrow \bar{\alpha}_t + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla\phi_i(w_t) - \alpha_{t,i}$ (for both settings, with/without replacement)
 $w_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$
 $\alpha_{t+1,i} = \nabla\phi_i(w_t)$ if $i \in S_t$, $\alpha_{t+1,i} = \alpha_{t,i}$ if $i \notin S_t$ and then $t \leftarrow t + 1$
end while
Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

SVRG

SVRG [JZ13] addresses one of the practical bottlenecks of SAG (still present in SAGA though) for *big-data* (n very large) in what concerns the memory consumption. The reader might have noticed that both SAG and SAGA require to store n snapshots or immediately n sample gradients computed at those snapshots in order to build g_t ! With SVRG the memory consumption does not scale with n as we will see but at the price of twice more sample gradient computations !

Let's note that the proximal version of SVRG (alongside with an importance sampling add-on) and its analysis came a bit later in *A Proximal Stochastic Gradient Method with Progressive Variance Reduction* (Xiao & Zhang, 2014) [XZ14]. In order to keep things clear and to develop the methods step by step I will introduce IS versions of SAGA/SVRG with HVRG.

Motivation:

The extra memory required by SAG(A) is in the general case of $\mathcal{O}(nd)$ (n sample gradients of size d) which might be prohibitive sometimes ; especially for non-parallel settings where this memory must take place on one single machine ! The idea of SVRG is to keep all the snapshots equivalent : i.e. at a given update step $t \in \mathbb{N}$, $\theta_{t,i} = \tilde{w}$ for every $i \in [n]$. The snapshots are not updated at every iteration but only once in m iterations. Therefore given S_t , one can build g_t only by recomputing $\frac{1}{b} \sum_{i \in S_t} \nabla \phi_i(\tilde{w})$ and storing $\bar{\alpha}_t$ in memory ($\mathcal{O}(d)$). Of course the new sample gradients $\{\nabla \phi_i(w_t)\}_{i \in S_t}$ are needed as well ! In total the update-step requires $\hat{b}$ extra sample gradient computations with respect to SAG(A) while keeping an extra memory cost with respect to SGD-like methods of $\mathcal{O}(d)$ instead of $\mathcal{O}(nd)$.

The implementation of SVRG uses a new structure that will actually define the standard algorithmic structure of second era-methods ; embedding SAG(A) as well.

1. An outer-loop at which a "fresh" estimation $\bar{\alpha} = \frac{1}{n} \sum_{i=1}^n \alpha_i$ of the gradient is produced.
2. An inner loop of m update-iterations where the α_i 's are updated (by definition $\bar{\alpha}$ as well).

Then we can clearly distinguish SVRG and SAGA :

- SAGA is then a degenerated case for that structure where there's no "fresh" estimation of $\bar{\alpha}$: the last inner-loop $\bar{\alpha}$ is taken as estimate.
- SVRG-like methods typically computes a whole gradient at outer-loop ; $\bar{\alpha}$ is then the true gradient at some outer-loop iterate $\tilde{w}$ (often the last inner-loop w_t encountered) and the α_i 's are not updated during the inner-loop.

Algorithm

Algorithm 7 SVRG

Initialization : $\eta_t = \eta > 0 \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla \phi_i(w_0)$: *warm start* or **0**), m number of inner-loop iterations

while stopping criterion not verified **do**

$\tilde{w} \leftarrow w_t$ (or average over the last inner-loop)

$\alpha_{t,i} \leftarrow \nabla \phi_i(\tilde{w}) \forall i \in [n]$

for $(t+1) \bmod (m+1) \neq 0$ **do**

$S_t \sim \mathbf{U}(2^{[n],\hat{b}})$ (without replacement) or $S_t \sim \mathbf{U}([n]^{\hat{b}})$ (with replacement)

$g_t \leftarrow \bar{\alpha}_t + \frac{1}{b} \sum_{i \in S_t} \nabla \phi_i(w_t) - \alpha_{t,i}$ (for both settings, with/without replacement)

$w_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$

$\alpha_{t+1,i} = \alpha_{t,i}$ and then $t \leftarrow t+1$

end for

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note -: In several cases SAGA is preferred to SVRG as the storage of the $\alpha_{t,i} = \nabla \phi_i(\theta_{t,i})$ requires only $\mathcal{O}(n)$ extra memory for linear model loss functions. E.g. if the image of each ϕ_i , namely $\phi_i(w)$ at $w \in \mathbb{R}^d$, can be written as $\tilde{\phi}(\langle w, \tilde{x}_i \rangle)$ for a general loss function $\tilde{\phi}$ common to all the cases $i \in [n]$ then $\nabla \phi_i(w) = \left(\frac{d\tilde{\phi}(u)}{du} \right)_{|u=\langle w, \tilde{x}_i \rangle} \tilde{x}_i$. Thus to store $\nabla \phi_i(\theta_{t,i})$ one would only need to store the scalar $\left(\frac{d\tilde{\phi}(u)}{du} \right)_{|u=\langle \theta_{t,i}, \tilde{x}_i \rangle}$ since $\tilde{x}_i$ is part of the accessible dataset.

Let's notice that when $m \rightarrow \infty$, SVRG has no chance to converge in an inner-loop.

What makes the variance decreasing is the *fresh* updates at each outer-loop beginning.

Conversely SAGA might converge in the inner-loop since the $\alpha_{t,i}$'s are kept up-to-date and then $\bar{\alpha}_t$ might converge to $\nabla \phi(w_*)$ as $t \rightarrow \infty$ for $m = \infty$.

AdaSVRG/AdaSAGA & HVRG

Just as for SGD importance sampling approaches were studied for SVRG and even for SAG with *Non-Uniform Stochastic Average Gradient Method for Training Conditional Random Fields* (Schmidt et al., 2015) (SAG-NUS) [Sa15]. Presented in the paper *Adaptive Variance Reducing for Stochastic Gradient Descent* (Shen, Qian et al., 2016) [She+16] (AdaSVRG, AdaSAGA and HVRG) borrow the idea of IS-SGD of analytically minimizing the trace of the variance term of the MSE under unbiased updates with mini-batches of size $\hat{\beta} = 1$.

The sampling scheme used in the paper is just a single lottery on $[n]$ under probability distribution $\mathcal{P}_{t,1} = \mathcal{P}_t$. The derivation of the optimal probabilities in that setting follows exactly the same steps as in IS-SGD. The original update iteration scheme of AdaSAGA (take $\tilde{w} = \theta_{t,i}$ for AdaSVRG) is simply resumed by

$i_t = i$ with probability $p_{t,i}$

$$g_t = \frac{\nabla\phi_{i_t}(w_t) - \nabla\phi_{i_t}(\theta_{t,i_t})}{np_{t,i_t}} + \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\theta_{t,i}) = \frac{\beta_{t,i_t}^1}{np_{t,i_t}} + \bar{\alpha}_t = \frac{\nabla\phi_{i_t}(w_t) - \alpha_{t,i_t}}{np_{t,i_t}} + \bar{\alpha}_t$$

$$V_t = \mathbb{E}_{i_t} [\|g_t - \mathbb{E}[g_t|\mathcal{F}_t]\|_2^2 | \mathcal{F}_t] = \left(\frac{1}{n^2} \sum_{i=1}^n \frac{\|\nabla\phi_i(w_t) - \alpha_{t,i}\|_2^2}{p_{t,i}} \right) - \|\nabla\phi(w_t) - \bar{\alpha}_t\|_2^2$$

Immediately in phase with the results obtained for IS-SGD, one derives the minimizing probabilities for V_t

$$p_{t,i}^* = \frac{\|\nabla\phi_i(w_t) - \alpha_{t,i}\|}{\sum_{i'=1}^n \|\nabla\phi_{i'}(w_t) - \alpha_{t,i'}\|} = \frac{\|\beta_{t,i}^1\|}{\sum_{i'=1}^n \|\beta_{t,i'}^1\|} \quad (4.22)$$

This distribution, yet not-practical since it also demands a full pass over the data at each update iteration, is rather interesting in the sense that now what one considers the most is the slack between the current sample gradients ($\nabla\phi_i(w_t)$) and their last known value ($\alpha_{t,i}$) !

This idea goes with the intuition : if $\alpha_{t,i}$ represents well $\nabla\phi_i(w_t)$ then the case i won't yield much variance in the estimated gradient g_t since $\alpha_{t,i}$ is part of it through $\bar{\alpha}_t$.

Algorithm

Algorithm 8 SAGA with IS

Initialization : $\eta_t = \eta > 0 \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla\phi_i(w_0)$: *warm start* or $\mathbf{0}$), $\{\mathcal{P}_t\}_{t \in \mathbb{N}}$ univariate probability oracles on $[n]$

while stopping criterion not verified **do**

$S_t \sim (\mathcal{P}_t)^{\hat{b}}$ (with replacement)

$g_t \leftarrow \bar{\alpha}_t + \frac{1}{\hat{b}} \sum_{i \in S_t} \frac{\nabla\phi_i(w_t) - \alpha_{t,i}}{np_{t,i}}$

$w_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$

$\alpha_{t+1,i} = \nabla\phi_i(w_t)$ if $i \in S_t$, $\alpha_{t+1,i} = \alpha_{t,i}$ if $i \notin S_t$ and then $t \leftarrow t + 1$

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note: Again a mini-batch version is proposed but not studied in the original paper of Shen, Qian et al. The sampling scheme is identical as in SGD with IS. When the probability oracle yields p_t^* as defined just above (4.22), SAGA with IS refers to the true AdaSAGA.

HVRG comes at the end of the paper to interpolate between AdaSVRG and AdaSAGA : it is AdaSAGA together with a full update of the α_i 's after m iterations (as in (Ada)SVRG) and adaptive probabilities embedding shrinkage that do not require $\mathcal{O}(n)$ sample gradient computations per update iterations ! The authors ensure that Vitter's algorithm (Vitter, 1987) allows to maintain a probability distribution after shrinkage in $\mathcal{O}(\log n)$ operations when $\hat{b} = 1$. To the best of my knowledge no practical implementation of Vitter's algorithm for such a requirement has been released so I decided not to further investigate that method.

Algorithm

Algorithm 9 HVRG

Initialization : $\eta_t = \eta > 0 \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla \phi_i(w_0)$: *warm start* or $\mathbf{0}$), m number of inner-loop iterations, $\rho > 1$ shrinkage parameter

while stopping criterion not verified **do**

$\tilde{w} \leftarrow w_t$ (or average over the last inner-loop)

$\alpha_{t,i} \leftarrow \nabla \phi_i(\tilde{w}) \forall i \in [n]$

for $(t+1) \bmod (m+1) \neq 0$ **do**

if $t+1 \bmod m = 2$ **then**

$p_{t,i}$ for every $i \in [n]$ defining $\mathcal{P}_t$ is computed as in (4.22)

end if

$S_t \sim (\mathcal{P}_t)^{\hat{b}}$ (with replacement)

$g_t \leftarrow \bar{\alpha}_t + \frac{1}{\hat{b}} \sum_{i \in S_t} \frac{\nabla \phi_i(w_t) - \alpha_{t,i}}{np_{t,i}}$

$w_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$

$\tilde{p}_{t,i} = \frac{p_{t,i}}{\rho}$ if $i \in S_t$ else $\tilde{p}_{t,i} = p_{t,i}$ and then $p_{t,i} = \frac{\tilde{p}_{t,i}}{\sum_{i'=1}^n \tilde{p}_{t,i'}}$

$\alpha_{t+1,i} = \nabla \phi_i(w_t)$ if $i \in S_t$, $\alpha_{t+1,i} = \alpha_{t,i}$ if $i \notin S_t$ and then $t \leftarrow t+1$

end for

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note: Just as SAGA, HVRG is able to converge at constant step-size $\eta_t = \eta$ if the inner-loop is infinitely long ($m \rightarrow \infty$). One concrete drawback for very large scale n is that a full-pass over the dataset is made twice after each outer-loop start... The first one, corresponding to the SVRG part is executed in the outer-loop in order to get new α_i 's. The second one comes slightly after when the task to achieve is to compute the probabilities $p_{t,i}$ for every $i \in [n]$ as equation (4.22) suggests it. However, for medium-scale datasets, the authors showed the efficiency of their method in terms of sample gradients computations but they didn't comment on whether they count the probabilities computations as a full-pass or not.

SARAH

Let's present a widely spread method, marking the transition between second and third era methods, achieving the best known theoretical bounds as the authors of its last version *An Efficient Algorithmic Framework for Stochastic Composite Nonconvex Optimization* (ProxSARAH) (Pham, Nguyen, et al., 2019) [Pha+19] say. Note that the original version (non-proximal) was first launched in *A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient* (SARAH) (Nguyen et al., 2017) [Ngu+17].

Note: Currently no extension of SARAH playing with non-uniform sampling exists !

Motivation :

(Prox)SARAH combines the best of both worlds ; SVRG for its memory consumption, SAGA for its adaptive nature at every update iteration. Instead of updating the knowledge one has about the sample gradients at all the snapshots $\{\theta_{t,i}\}_{i=1}^n$ and keeping that information in memory in order to construct the new deterministic gradient estimate $\bar{\alpha}_{t+1}$ SARAH intends to only store $\bar{\alpha}_{t+1} = g_t$. Hence SARAH only requires $\mathcal{O}(d)$ extra memory while adapting for free $\bar{\alpha}$ during inner-loop iterations !

Let's motivate (mini-batch) SARAH's stochastic gradient. Let's consider the t -update iteration ($t \geq 1$) ; given S_t ($\hat{b} = |S_t|$) and $\bar{\alpha}_t = g_{t-1}$ a deterministic gradient estimate

$$g_t = \bar{\alpha}_t + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t) - \nabla \phi_i(w_{t-1}) \quad (4.23)$$

SARAH papers talk about recursive variance reduction. Indeed, one may also write down :

$$g_t = g_{t-1} + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t) - \nabla \phi_i(w_{t-1}) \quad (4.24)$$

Compared to the other second-era methods SARAH assumes at some point that g_{t-1} was a good approximation of $\nabla \phi(w_{t-1})$ and thus that $\bar{\alpha}_t = g_{t-1}$ should be a fair representation of $\frac{1}{n} \sum_{i=1}^n \nabla \phi_i(w_{t-1})$. In other words, at an update iteration $t \geq 1$, SARAH assumes that all the previous snapshots $\{\theta_{t-1,i}\}_{i=1}^n$ were all relatively close to w_{t-1} itself close from w_t . This happens whenever the iterates $\{w_t\}_{t \in \mathbb{N}}$ accumulates around a converging w_* .

One immediate critical comment would be to argue that the early steps of SARAH rely on an assumption *a priori* wrong. However under sufficiently small step-sizes the stochastic gradient of SARAH is robust enough to ensure a convergence until a zone for which the iterates do not vary much. Then SARAH reveals all its power with respect to SAGA as it implicitly updates all the snapshots at each inner-loop iteration without being charged with any further computational cost.

A fast paced convergence towards accurate ϵ -solutions is empirically observed once the optimization process arrives at an iterate w_t close enough from a $w_* \in \mathcal{S}^*$ after enough iterations such that g_t had the time to include local information about most of the sample gradients.

As the authors mention clearly g_t is **not unbiased** for the one step expectation :

$$\mathbb{E}[g_t | \mathcal{F}_t] = \nabla \phi(w_t) - \nabla \phi(w_{t-1}) + g_{t-1} \neq \nabla \phi(w_t)$$

It is still, hopefully, asymptotically unbiased for the one-step expectation if one experiences well $\lim_{t \rightarrow \infty} \|\nabla \phi(w_t) - g_t\| = 0$ and unbiased under a total expectation across all the possible random events since for $t \geq 1$,

$$\mathbb{E}[g_t] = \nabla \phi(w_t) - \nabla \phi(w_{t-1}) + \mathbb{E}[g_{t-1}]$$

Reasoning backwards by induction until $\mathbb{E}[g_t] = \nabla \phi(w_t) - \nabla \phi(w_0) - \mathbb{E}[g_0]$ and knowing that $g_0 = \nabla \phi(w_0)$ allows to conclude that

$$\mathbb{E}[g_t] = \nabla \phi(w_t)$$

The algorithm presented below is one special instance of the more complex algorithm one would find, adapted to our notations, in the latest version of ProxSARAH. The quick motivation given above considered tacitly that $q_t = q = 1$ for every $t \in \mathbb{N}$.

Algorithm

Algorithm 10 SARAH

Initialization : $\eta_t = \eta > 0 \ \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, m number of inner-loop iterations and $w_{-1} = w_0$

while stopping criterion not verified **do**

$\tilde{w} \leftarrow w_t$ (or average over the last inner-loop)

$\bar{\alpha}_t = \nabla \phi(w_t)$

for $(t + 1) \bmod (m + 1) \neq 0$ **do**

$S_t \sim \mathbf{U}(2^{[n], \hat{b}})$ (without replacement) or $S_t \sim \mathbf{U}([n]^{\hat{b}})$ (with replacement)

$g_t \leftarrow \bar{\alpha}_t + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t) - \nabla \phi_i(w_{t-1})$

$\hat{w}_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$

$w_{t+1} = q_t \hat{w}_{t+1} + (1 - q_t) w_t$ and then $t \leftarrow t + 1$

end for

end while

Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note : As wanted by the authors and like SAGA, SARAH is theoretically able to converge within an inner-loop of infinite size under an adequate step-size choice.

Chapter 5

New approaches and paradigms for Stochastic Gradient Methods

Oath Taking

This chapter will summarize personal researches / experiments achieved after the reading of a large amount of papers and scientific content. To the best of my knowledge those procedures/ideas have not been directly stated nor tried but this research field is so active that at some point one needs to write down things and hope that it doesn't totally overlap with existing work. Chapters 4 and 6 may convince the reader that the many things have already been attempted in the field of stochastic gradient methods for machine-learning large-scale optimization. However there exists *little* gaps for which slight improvements can be made. This is exactly what this chapter aims to accomplish.

- ***K*-Greedy Sampling :**

A rather surprising remark is the fact that none of the methods presented in Chapter 3/4 nor in Chapter 6 aims to immediately minimize the one-step expected value of f . I open a discussion about the optimization of *surrogates* under the randomness of the mini-batches of indices S_t elected to produce g_t at any update iteration $t \in \mathbb{N}$. In the case of "*single sample*" unbiased updates there exist closed forms for the optimal probability distribution to adopt in order to minimize the expected value of f after one-step for quadratic surrogates.

Content: An extensive analysis about that topic is developed, some interesting lemmas are derived and a heuristic adaptive strategy to overcome the practical bottleneck of importance sampling is proposed. Unfortunately, preliminary tests did not show great results. More tests need to be run to draw final conclusions.

- **$\tilde{K}$ Prox-VR Framework:**

Initially motivated by a general update scheme presented in the *K-Greedy Sampling* section, leading to convergence in expectation, I introduce a new technique taking into account a refined information about a quadratic upper-bound on the objective function when this latter is smooth. The varied literature about stochastic (quasi)-second-order methods has not attempted yet to tackle immediately an update step with a non-smooth regularizer in a "*proximal fashion like*" way.

Content: Convincing tests are conducted in order to get a better insight about the efficiency of my two new methods : $\tilde{K}$ Prox-SVRG, $\tilde{K}$ KProx-SARAH. Furthermore, a proof of convergence is given and discussed.

- **Adaptive Heuristic Method:**

Another idea of mine is based on the results obtained in the section "K-Greedy Sampling" and aims to increase progressively the step-size used by SVRG using again an available quadratic upper-bound, K . Some related works exist but their procedures are different.

Content: This last section will be merged into $\tilde{K}$ Prox-VR's section. The heuristic method is defined, motivated and validated on two experiments using $(\tilde{K})$ Prox-SVRG/SARAH.

5.1 Greedy-Sampling

Introduction

As announced above, none of the papers I have read seem to directly tackle a key aspect in my opinion : choosing an univariate probability distribution for samples $[n]$ in order to *minimize the expected decrease of the objective function given the current iterate*.

That is, given $w_t \in \mathbb{R}^d$ and the fact that g_t is built upon a single-sample $i_t \sim \mathcal{P}_t$, one would like to find the vector of probabilities $p_t^G \in \Delta^n$ defining $\mathcal{P}_t$ that satisfies :

$$p_t^G := \arg \min_{p_t \in \Delta^n} \mathbb{E}_{i_t} [f(w_{t+1})] \quad (5.1)$$

As displayed in Chapter 4 in the case where $r(w) = 0, \forall w \in \mathbb{R}^d$ and more generally in [ZZ15], the sampling induced by

$$p_{t,i}^* = \frac{\|\nabla \phi_i(w_t)\|}{\sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|} \quad \forall i \in [n]$$

minimizes $\text{tr}(\text{MSE}(g_t))$ for unbiased updates for first-era methods. Still using first-era methods this latter sampling also achieves the biggest one-step expected decrease in the absence of further information about the global smoothness of the function.

Inspiring myself of Lemma 1 in [ZZ15] and using a personal derivation of the value of $\mathbb{E}_{i_t} [f(w_{t+1})]$ I will show a closed form for p_t^G for quadratic loss functions. Furthermore I will state conditions under which the convergence bound of Theorem 1 in [ZZ15] is improved using p_t^G instead of p_t^* .

Strategy

-As a starter, the **Master Lemma** of this section will be shown. It will be used multiple times in other proofs. **Master Lemma's** own proof is personal but the result is well known...

-Then, the **GDE** (General Decrease Expansion) for quadratic functions will be introduced. It consists in the expression of the value of a quadratic function $\bar{\phi}$ after an update step of the type $w_{t+1} \leftarrow w_t - \eta_t g_t$ where g_t is a realization of a stochastic gradient.

-The **GDE** paves the way through **GEDE** (General Expected Decrease Expansion) theorems for both first and second era methods. As the name suggests it, these theorems give the expected decrease, according to the used stochastic gradient, for $\bar{\phi}$ quadratic functions.

-Finally, the expressions appearing in **GEDE** theorems will display a term to be minimized w.r.t the sampling probabilities $p_{t,i}$; the **Master Lemma** will then help to derive p_t^G .

Master Lemma

Lemma 6 (classical risk minimization optimal solution) Be $|c_j|$ for all $j \in [n]$ a positive coefficient and be the cost function $g : \Delta^n \subset \mathbb{R}^d \rightarrow \mathbb{R}, p \rightarrow g(p) = \sum_{j=1}^n \frac{c_j^2}{p_j}$. The optimization problem

$$\min_{p \in \mathbb{R}^d} g(p) \text{ s.t } p \in \text{int}(\Delta^n) := \{p \in \mathbb{R}^d | p_j \geq 0, \forall j \in [n] \text{ and } \sum_{j=1}^n p_j = 1\}$$

admits as optimal solution $p_i^* = \frac{|c_i|}{\sum_{j=1}^n |c_j|}$ for all $i \in [n]$.

Proof: First one has to notice that this problem is convex. Indeed, g is a compound function of n convex univariate functions that are defined on $[0, 1]$ (otherwise, if one of the p_i is situated outside this interval the whole vector p won't be feasible). The image of those univariate functions is of the generic form $g_i(x) = \frac{c_i^2}{x}$, thus twice continuously differentiable on the interior of $[0, 1]$. $\forall x \in]0, 1[, g''(x) = \frac{2c_i^2}{x^3} \succ 0$. The positivity of the curvature of all the g_i shows their convexity and hence the convexity of g . The domain of g is also convex¹. Since we have to minimize a convex function over a convex feasibility domain, the problem is a constrained 'smooth' convex problem. As seen in Chapter 2 for such problems if there exists a point $p^* \in \text{dom}(g)$ for which $\langle \nabla g(p^*), p - p^* \rangle \geq 0$ for any $p \in \text{dom}(g)$ then p^* is a global minimum for g over $\text{dom}(g)$.

Let's verify that the proposed p^* fulfills the above requirement. $\forall p \in \Delta^n$,

$$\nabla g(p) = [-\frac{c_1^2}{p_1^2}, \dots, -\frac{c_n^2}{p_n^2}]^T$$

Therefore,

$$\nabla g(p^*) = -[\left(\sum_{j=1}^n |c_j|\right)^2, \dots, \left(\sum_{j=1}^n |c_j|\right)^2]^T = \left(\sum_{j=1}^n |c_j|\right)^2 \mathbf{e}_n$$

Thus for any two vectors $u, v \in \Delta^n$,

$$\langle \nabla g(p^*), u - v \rangle = \sum_{i=1}^n \left(\sum_{j=1}^n |c_j|\right)^2 u_i - v_i = \left(\sum_{j=1}^n |c_j|\right)^2 \sum_{i=1}^n u_i - v_i = \left(\sum_{j=1}^n |c_j|\right)^2 \left(\left(\sum_{i=1}^n u_i\right) - \left(\sum_{i=1}^n v_i\right)\right) = 0$$

replacing u by any $p \in \Delta^n$ and $v = p^*$ we get the desired result.

□

¹It's called the n -simplex.

As announced, after the **Master Lemma** derivation, the reader can find the **GDE**.

General Decrease Expansion (GDE) : quadratic case

Let's consider a function $\bar{\phi} : \mathbb{R}^d \rightarrow \mathbb{R}$ defined for every $v \in \mathbb{R}^d$ by

$$\bar{\phi}(v) = \kappa + \langle k, v \rangle + \frac{1}{2} \langle Kv, v \rangle \quad (5.2)$$

where $\kappa \in \mathbb{R}$, $k \in \mathbb{R}^d$ and finally $K = K^T \succeq \mathbf{0} \in \mathbb{R}^{d \times d}$. Let's consider an iterate $w_t \in \mathbb{R}^d$, any direction $g_t \in \mathbb{R}^d$ and any step-size $\eta_t \in \mathbb{R}^+$. The expansion of $\bar{\phi}(w_t - \eta_t g_t)$ follows easily :

$$\bar{\phi}(w_t - \eta_t g_t) = \kappa + \langle k, w_t - \eta_t g_t \rangle + \frac{1}{2} \langle K(w_t - \eta_t g_t), w_t - \eta_t g_t \rangle \quad (5.3)$$

$$= \underbrace{\kappa + \langle k, w_t \rangle + \frac{1}{2} \langle Kw_t, w_t \rangle}_{\bar{\phi}(w_t)} - \underbrace{\eta_t [\langle k + Kw_t, g_t \rangle - \frac{\eta_t}{2} \langle Kg_t, g_t \rangle]}_{\text{decrease term of } \bar{\phi}} \quad (5.4)$$

Now I mark down **Theorem 1** that describe the optimal sampling probability vector p_t^K , at an iterate w_t , for a first era stochastic gradient of the type ($\mathbb{P}[i_t = i] = p_{t,i}^K$)

$$g_t = \frac{\nabla \phi_{i_t}(w_t)}{np_{i_t}}$$

Theorem 1 (GEDE : quadratic case — first-era) : Let's consider a function $\bar{\phi}$ as defined in (5.2) and let ϕ be the differentiable compound function defined as usual. Given an iterate $w_t \in \mathbb{R}^d$ and a *proper* probability distribution $\mathcal{P}_t$ over $[n]$, let g_t be equal to $\frac{\nabla \phi_{i_t}(w_t)}{np_{t,i_t}}$ then

$$\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] = \bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, \nabla \phi(w_t) \rangle - \frac{\eta_t}{2n^2} \sum_{i=1}^n \frac{\|\nabla \phi_i(w_t)\|_K^2}{p_{t,i}} \right] \quad (5.5)$$

Furthermore for any $\eta_t \geq 0$, the probability vector p_t^K minimizing $\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)]$ is characterized by

$$p_{t,i}^K = \frac{\|\nabla \phi_i(w_t)\|_K}{\sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|_K} \quad \forall i \in [n] \quad (5.6)$$

Proof: Let's denote by $g_{t,i}$ the deterministic i -th realization of g_t ; i.e. the term $\frac{\nabla \phi_i(w_t)}{np_{t,i}}$. Then

$$\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] = \sum_{i=1}^n p_{t,i} \bar{\phi}(w_t - \eta_t g_{t,i})$$

Using the expansion in (5.4) we can further develop :

$$\begin{aligned} \mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] &= \sum_{i=1}^n p_{t,i} \left(\bar{\phi}(w_t) - \eta_t [\langle k + Kw_t, g_{t,i} \rangle - \frac{\eta_t}{2} \langle Kg_{t,i}, g_{t,i} \rangle] \right) \\ &= \bar{\phi}(w_t) - \eta_t [\langle k + Kw_t, \nabla \phi(w_t) \rangle - \frac{\eta_t}{2} \sum_{i=1}^n p_{t,i} \langle K \frac{\nabla \phi_i(w_t)}{np_{t,i}}, \frac{\nabla \phi_i(w_t)}{np_{t,i}} \rangle] \\ &= \bar{\phi}(w_t) - \eta_t [\langle k + Kw_t, \nabla \phi(w_t) \rangle - \frac{\eta_t}{2n^2} \sum_{i=1}^n \frac{\|\nabla \phi_i(w_t)\|_K^2}{p_{t,i}}] \end{aligned}$$

For the second part of the proof it's sufficient to notice that minimizing $\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)]$ boils down to be a practical case covered by the **Master Lemma** for $|c_i| = \|\nabla \phi_i(w_t)\|_K$.

It remains to conclude that $p_{t,i}^K = \frac{|c_i|}{\sum_{i'=1}^n |c_{i'}|} = \frac{\|\nabla \phi_i(w_t)\|_K}{\sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|_K} \quad \forall i \in [n]$.

□

Note 1: K -norm is naturally defined as $\|\cdot\|_K = \sqrt{\langle K\cdot, \cdot \rangle}$.

Note 2: Whenever $|S_t| = 1$ (as $S_t = \{i_t\}$ here) with probability 1, first-era (unbiased) stochastic gradients can always be written as $\lambda \frac{\nabla \phi_i(w_t)}{np_{t,i}}$ for the case $i \in [n]$. However $\lambda \in \mathbb{R}$ is useless while putting a step-size η_t before g_t ; in the expression $w_t - \eta_t g_t$. Without loss of generality then we will assume $\lambda = 1$ hence the assumption in **Theorem 1**.

Discussion: Let's consider the non-regularized setting for the problem (1.2) ($r(w) = 0$). Whenever one deals with an objective function $f = \phi$ that is quadratic, one direct implication of **Theorem 1** is that p_t^K is the looked for p_t^G *greedy sampling* that minimizes the one-step expected outcome $\mathbb{E}_{i_t}[f(w_{t+1})] = \mathbb{E}_{i_t}[f(w_t - \eta_t g_t)]$.

Indeed when ϕ is quadratic we can choose $\bar{\phi}$ to be equal to ϕ ; thanks to **Theorem 1** it follows then in that case that

$$\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] = \mathbb{E}_{i_t}[f(w_t - \eta_t g_t)] = \mathbb{E}_{i_t}[f(w_{t+1})]$$

would be minimized with respect to the sample probabilities by taking $p_t^G = p_t^K$.

For first-era methods with $|S_t| = 1$, based on that observation, one can be motivated by choosing p_t^K to construct $\mathcal{P}_t$; sampling distribution at a given (t, w_t) . As it will be shown right after, this intuition is correct only whenever one disposes of a quadratic upper-bound $\bar{\phi}$ that is accurate enough. In other words ϕ should behave, locally, like the quadratic function defined by K , k and κ in order to observe a potential interest using K -greedy sampling.

One can notice that p_t^K degenerates in p_t^* if the information about the smoothness, stored in K , turned out to be a multiple of the identity matrix. E.g. if one decides to take $\bar{\phi}$ as being the (isotropic) quadratic upper-bound of L -smooth functions then $K = L I$ and

$$p_{t,i}^K = \frac{\|\nabla \phi_i(w_t)\|_{LI}}{\sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|_{LI}} = \frac{L \|\nabla \phi_i(w_t)\|}{L \sum_{i'=1}^n \|\nabla \phi_{i'}(w_t)\|} = p_t^*$$

Experiments : K -greedy sampling for first-era methods

The ultimate goal of those experiments is to expose results that confirm the results I derived for **Theorem 1**. It's also the occasion to get a better insight about whenever K -greedy sampling might be an interesting option ; i.e. leading to a significant improvement with respect to the widespread p_t^* (gradients' norms based sampling).

Since exact expectations are computed and confidence intervals are estimated based on a classical *bootstrap* approach, the dataset used will be of moderate size $n = 684, d = 10$.

Fixing ourselves at one iterate $w_t \in \mathbb{R}^d$ the tests compare the expected value for $f(w_{t+1}) = f(w_t - \eta_t g_t)$ depending on the sampling used (US,IS, K -greedy) but also on the value of the step-size ! Let's say we consider C step-sizes. For each one of them the objective value (n sample loss functions) according to each possible choice (n in total) of $i_t \sim \mathcal{P}_t$ must be computed.

The number of operations, not accounting for the dimension of the data d is of order $\mathcal{O}(Cn^2)$ which scales really fast with n ! The benchmark dataset is a cleaned version of the real-world *Breast Cancer Wisconsin (Original) Data Set*². To create a bad instance for K -greedy sampling a same-sized homemade dataset is built and used. Just as for Chapter 4, the data generation is discussed in the **Appendix 2**.)

Summary

Two loss functions will be tested ; i.e. the binary classification quadratic loss (1) and the quadratically smoothed hinge-loss for SVM (2).

elements	expression/value
data size (n, d)	$n = 684, d = 10$
sample differentiable cost functions ϕ_i	(1) : $(1 - \langle \tilde{x}_i, w \rangle)^2$ (2) : $([1 - \langle \tilde{x}_i, w \rangle]_+)^2$
regularization constant λ	0
regularizer $R(\cdot)$	0
step-size	parameter taking multiple values

For both cases it stands : $\forall v \in \mathbb{R}^d, \phi(v) = \frac{1}{n} \sum_{i=1}^n \phi_i(v) \leq 1 + \langle \frac{2}{n} \sum_{i=1}^n \tilde{x}_i, v \rangle + \frac{1}{2} \langle (\frac{2}{n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T) v, v \rangle$

Hence **Theorem 1** applies and we can identify for $\bar{\phi}$; $\kappa = 1, k = \frac{2}{n} \sum_{i=1}^n \tilde{x}_i$ and

$$K = \frac{2}{n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T$$

Experiments with *Breast Cancer Wisconsin Data Set* (wdbc) were also undertaken, at the very same iterates w_t , for second-era methods in order to produce a fair comparison afterwards. This latter will be shown in the section about experiments for second-era methods with K -greedy sampling.

²Source : <https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+%28original%29>

Quadratic case (1) : good (left) vs. bad (right) instance

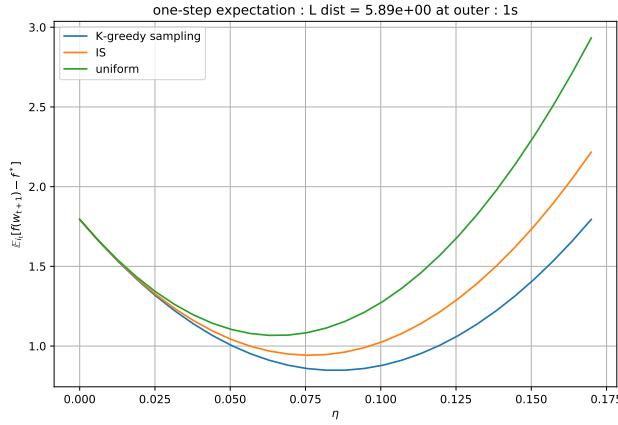


Figure 5.1: $\mathbb{E}_t[f(w_{t+1}) - f^*]$: *wdbc*

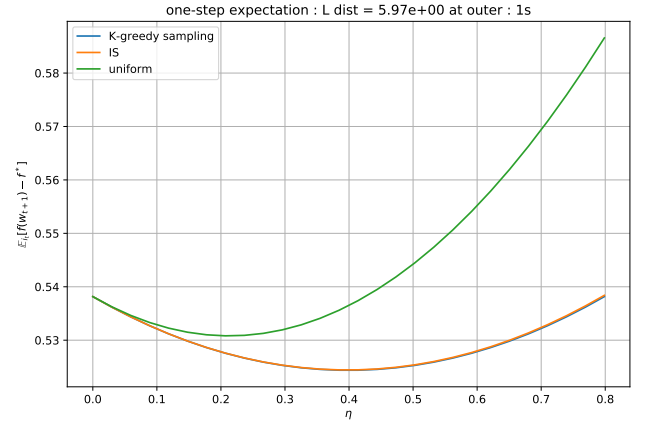


Figure 5.2: $\mathbb{E}_t[f(w_{t+1}) - f^*]$: *homemade*

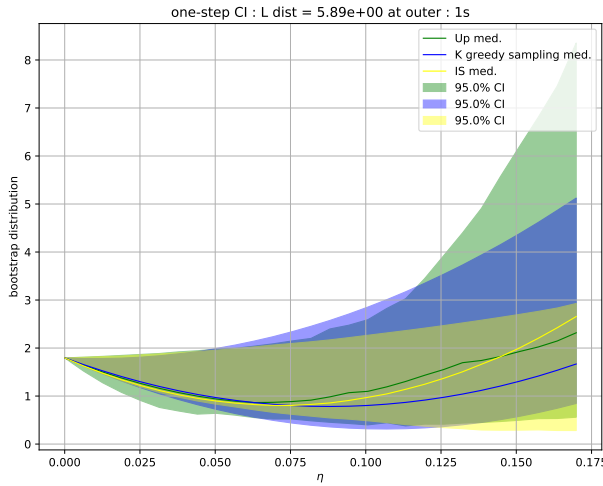


Figure 5.3: 95% bootstrap CI: *wdbc*

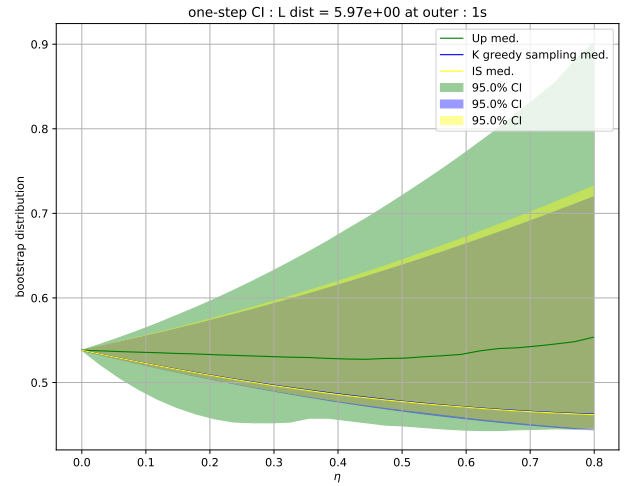


Figure 5.4: 95% bootstrap CI: *homemade*

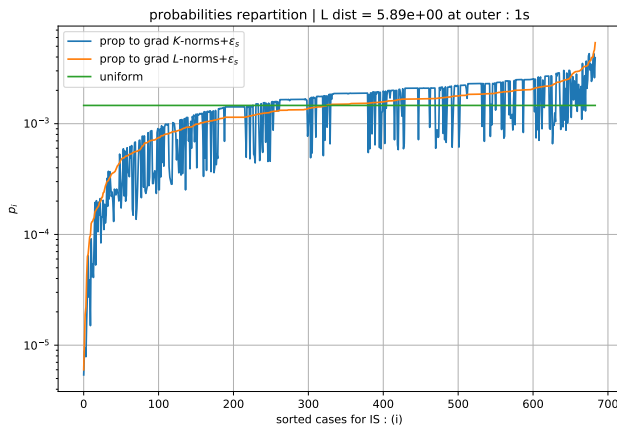


Figure 5.5: $\mathcal{P}_t$: *wdbc*

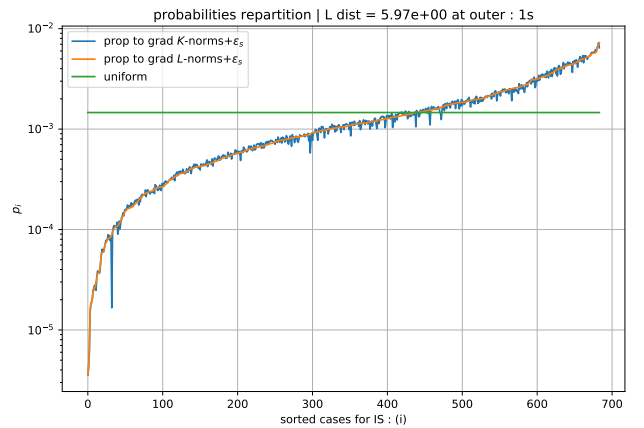


Figure 5.6: $\mathcal{P}_t$: *homemade*

Smoothed hinge case (1) : good (left) vs. bad (right) instance

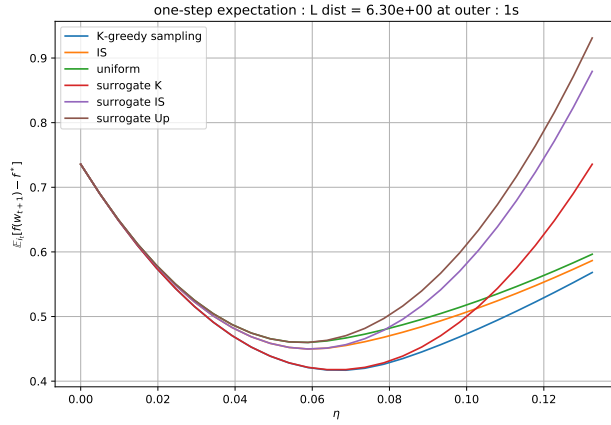


Figure 5.7: $\mathbb{E}_t[f(w_{t+1}) - f^*]$: *wdbc*

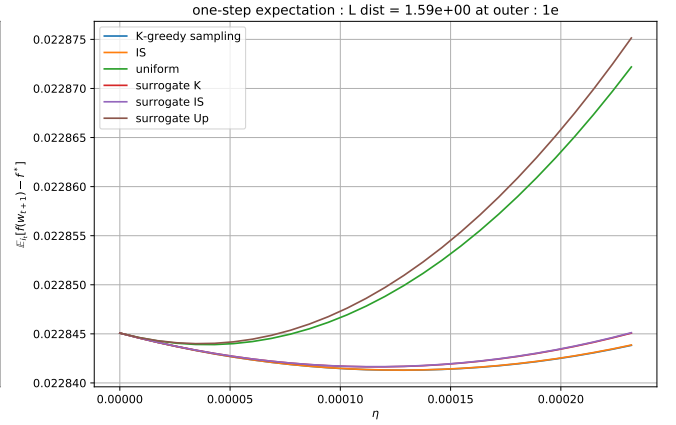


Figure 5.8: $\mathbb{E}_t[f(w_{t+1}) - f^*]$: *homemade*

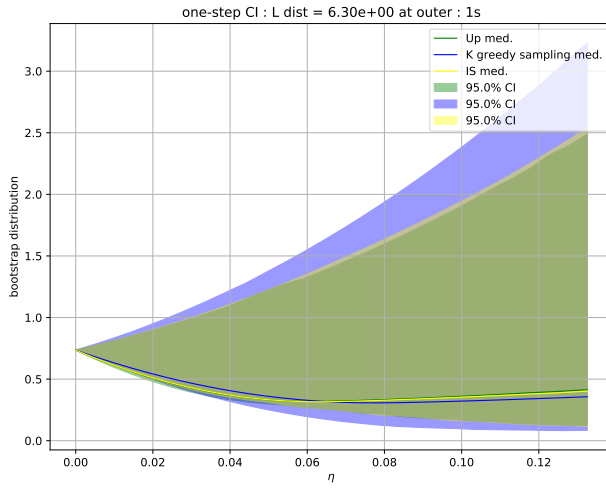


Figure 5.9: 95% bootstrap CI: *wdbc*

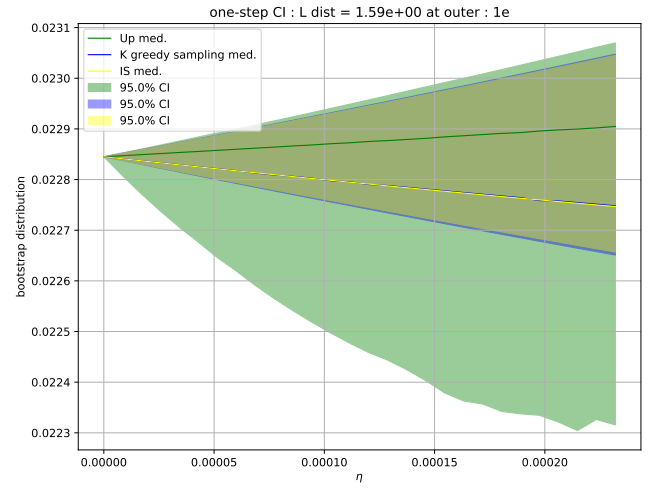


Figure 5.10: 95% bootstrap CI: *homemade*

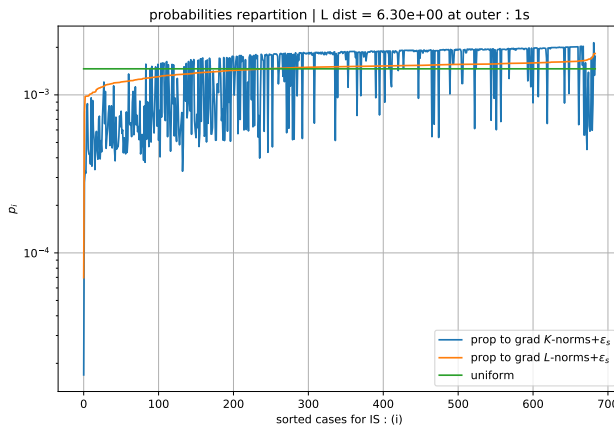


Figure 5.11: $\mathcal{P}_t$: *wdbc*

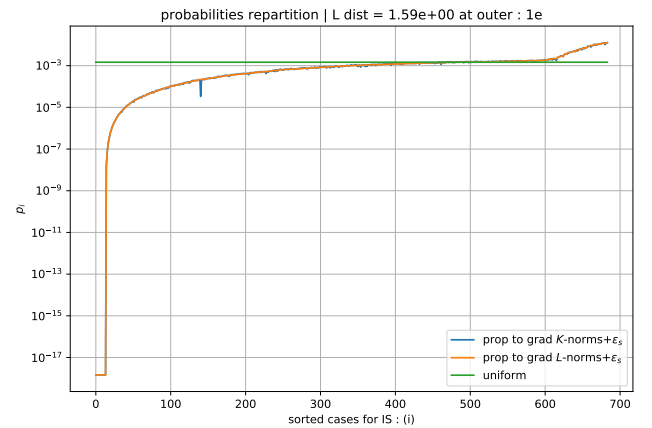


Figure 5.12: $\mathcal{P}_t$: *homemade*

Quadratic & Smoothed hinge case (1) : good instances

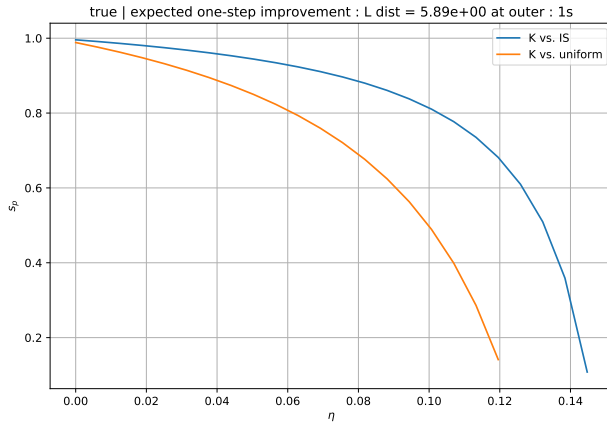


Figure 5.13: expected decrease ratio (quadratic) : *wdbc*

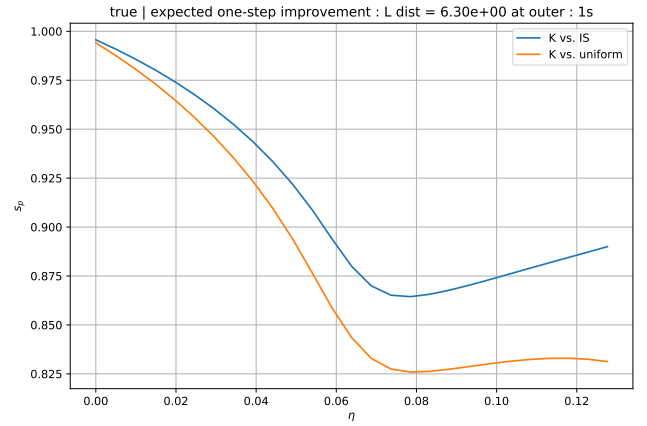


Figure 5.14: expected decrease ratio (smoothed-hinge) : *wdbc*

Comments

For both cases (2) the theoretical interest for importance sampling at a one step-horizon is well-sustained for K -greedy sampling and IS ; looking at those graphs uniform sampling is always the worst choice one can make !

As the intuition would have suggested it the most attractive results concerning K -greedy sampling arise whenever p^K is the most different from p^* itself quite different from (US) $(\frac{1}{n})$. A measurable quantity assessing the closeness between two discrete distributions can be described by the concept of discrepancy following later on. For now, the reader simply has to keep in mind that the quantity $n\|p^K - p^*\|_1$ is bigger for the left instances than for the right ones.

$f(w_{t+1})$'s *bootstrap*-CI is typically wider when using K -greedy sampling instead of IS. However, after multiple tests, the advantage goes for K -greedy in what concerns the median value for $f(w_{t+1})$, especially for larger step-sizes for which *IS* and *US* yield worse expected values for $f(w_{t+1})$! One could emphasize the ability of importance sampling to enhance the range of step-sizes for which an expected decrease is yet to come.

For the experimental case (2) we can make the distinction between $\bar{\phi}$ (quadratic surrogate) and ϕ (smoothed hinge loss). A non-negligible amount of probabilities on the graph to the right are almost 0 (theoretically 0 but I do not allow probabilities to fall under ϵ_s of the order of the ϵ -machine) since the gradients can vanish easily in case (2) ; e.g. each time $\langle \tilde{x}_i, w \rangle \geq 1$.

Even though the instances to the right do not show it, sometimes I've experienced step-sizes for which $\mathbb{E}_{i_t}[\phi(w_{t+1})]$ is lower for IS than for K -greedy but the same statement did not hold for $\mathbb{E}_{i_t}[\bar{\phi}(w_{t+1})]$. Actually nothing prevents such a phenomenon to happen, the more the actual ϕ behaves differently from $\bar{\phi}$ the more such events are plausible.

On the two last graphs one can spot the one-step horizon speed-up gain in terms of expected decrease ratio $\frac{\mathbb{E}_{i_t}[f(w_{t+1}) - f(w_t)]_{IS \text{ or } US}}{\mathbb{E}_{i_t}[f(w_{t+1}) - f(w_t)]_{K-greedy}}$ depending on the step-size used. Again when η is too big for IS or US we can empirically win up to 15% for the smoothed hinge loss and much more for the purely quadratic case.

Let's adapt **Theorem 1** to SAGA-like methods, again for size-1 mini-batch updates ($|S_t| = 1 \forall t \in \mathbb{N}$). The theoretical optimal sampling as defined by (5.1), p_t^G , will slightly differ from the one obtained for first-era methods. It will look like to the one the authors of HVRG derived.³

Theorem 2 (GEDE : quadratic case — second-era, SAGA-like) : Let's consider a function $\bar{\phi}$ as defined in (5.2) and let ϕ be the differentiable compound function defined as usual. Given an iterate $w_t \in \mathbb{R}^d$ and a *proper* probability distribution $\mathcal{P}_t$ over $[n]$, $\{\theta_{t,i}\}_{i=1}^n$ a list of snapshots in $\mathbb{R}^d$, let g_t be equal to $\frac{\nabla\phi_{i_t}(w_t) - \nabla\phi_{i_t}(\theta_{t,i_t})}{np_{t,i_t}} + \frac{1}{n} \sum_{i'=1}^n \nabla\phi_{i'}(\theta_{t,i'})$,^a

$$\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] = \bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, \nabla\phi(w_t) \rangle - \frac{\eta_t}{2} \left[-\|\bar{\alpha}_t\|_K^2 + 2\langle K\nabla\phi(w_t), \bar{\alpha}_t \rangle + \frac{1}{n^2} \sum_{i=1}^n \frac{\|\beta_{t,i}\|_K^2}{p_{t,i}} \right] \right] \quad (5.7)$$

Furthermore for any $\eta_t \geq 0$, the probability vector p_t^K minimizing $\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)]$ is characterized by

$$p_{t,i}^K = \frac{\|\beta_{t,i}\|_K}{\sum_{i'=1}^n \|\beta_{t,i'}\|_K} \quad \forall i \in [n] \quad (5.8)$$

^a Let's recall the notations : $\bar{\alpha}_t = \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\theta_{t,i})$ and $\beta_{t,i}^1 = \beta_{t,i} = \frac{\nabla\phi_i(w_t) - \nabla\phi_i(\theta_{t,i})}{np_{t,i}}$.

Proof: Again since $\mathbb{E}_{i_t}[g_t] = \nabla\phi(w_t)$ and given the fact that ϕ is quadratic the proof follows exactly the same steps as in **Theorem 1** until and including :

$$\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] = \sum_{i=1}^n p_{t,i} \left(\bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, g_{t,i} \rangle - \frac{\eta_t}{2} \langle Kg_{t,i}, g_{t,i} \rangle \right] \right)$$

Here $g_{t,i} = \frac{\nabla\phi_i(w_t) - \nabla\phi_i(\theta_{t,i_t})}{np_{t,i}} + \frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\theta_{t,i})$ hence :

$$\begin{aligned} \mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)] &= \bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, \nabla\phi(w_t) \rangle - \frac{\eta_t}{2} \left[\sum_{i=1}^n p_{t,i} \left\langle K \left(\frac{\beta_{t,i}}{np_{t,i}} + \bar{\alpha}_t \right), \frac{\beta_{t,i}}{np_{t,i}} + \bar{\alpha}_t \right\rangle \right] \right] \\ &= \bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, \nabla\phi(w_t) \rangle - \frac{\eta_t}{2} \left[\sum_{i=1}^n p_{t,i} \left(\|\bar{\alpha}_t\|_K^2 + 2\langle K \frac{\beta_{t,i}}{np_{t,i}}, \bar{\alpha}_t \rangle + \langle K \frac{\beta_{t,i}}{np_{t,i}}, \frac{\beta_{t,i}}{np_{t,i}} \rangle \right) \right] \right] \end{aligned}$$

Using the fact that $\mathbb{E}_{i_t} \left[\frac{\beta_{t,i_t}}{p_{t,i_t}} \right] = \sum_{i=1}^n p_{t,i} \frac{\nabla\phi_i(w_t) - \nabla\phi_i(\theta_{t,i})}{np_{t,i}} = \nabla\phi(w_t) - \bar{\alpha}_t$

$$\bar{\phi}(w_t) - \eta_t \left[\langle k + Kw_t, \nabla\phi(w_t) \rangle - \frac{\eta_t}{2} \left[-\|\bar{\alpha}_t\|_K^2 + 2\langle K\nabla\phi(w_t), \bar{\alpha}_t \rangle + \frac{1}{n^2} \sum_{i=1}^n \frac{\|\beta_{t,i}\|_K^2}{p_{t,i}} \right] \right]$$

Just as for **Theorem 1** the only part depending on $\mathcal{P}_t$ is the term $\sum_{i=1}^n \frac{\|\beta_{t,i}\|_K^2}{p_{t,i}}$ for which the minimization falls in the practical cases handled by the **Master Lemma** ; e.g. for $|c_i| = \|\beta_{t,i}\|_K$.

We can conclude that $p_{t,i}^K \propto \|\beta_{t,i}\|_K$ for second-era SAGA-like methods.

□

Note: The structure of the stochastic gradient does not change the quadratic nature of $\mathbb{E}_{i_t}[\bar{\phi}(w_t - \eta_t g_t)]$ with respect to η_t . **Theorem 2** generalizes **Theorem 1** in the sense that if one drops the control variate part inherent to second-era methods, that is $\nabla\phi_i(\theta_{t,i}) = \mathbf{0}, \forall i \in [n]$ and $\bar{\alpha}_t = \mathbf{0}$, the expression (5.7) is simply (5.5).

³Cfr. end of Chapter 4.

Discussion : Looking at expression (5.7) for $f = \phi$ being quadratic (that is, we can choose $\bar{\phi} = \phi$) let's underline the fact that for second-era methods, whenever $\nabla\phi_i(\theta_{t,i}) \rightarrow \nabla\phi_i(w_t)$ for every $i \in [n]$ occurs as $t \rightarrow \infty$, the potential speed-up effect of any importance sampling will progressively disappear.

This latter comment supports the idea that second era methods embedding importance sampling should probably adopt an adaptive probability scheme that would converge to uniform sampling.

Indeed the potential importance sampling gain may become arbitrary small while the sampling cost will stay identical , unless one decides to switch back to uniform sampling at some point.

Finally let's point out that p_t^K defined in **Theorem 2** corresponds, according to the same arguments used in the discussion after **Theorem 1**, to p_t^G (5.1) when using SAGA-like stochastic gradients to optimize ϕ that is quadratic.

Experiments : K -greedy sampling for second-era methods, SAGA-like

In order to compare the effects of K -greedy sampling for first/second era methods I did apply classical SVRG with a constant step-size of $\eta = \eta_t = \frac{1}{5L}$, a mini-batch size of $b = \lfloor \sqrt{n} \rfloor$ and an inner-loop size of $m = \lfloor \sqrt{n} \rfloor$ on the cases (1) and (2) for the *Breast Cancer Wisconsin Data Set* and time by time I did extract the current iterate w_t in order to keep track of the results one would have got if ever K -greedy sampling was applied with the next update-step. Alongside I decided to collect the values for $\text{tr}(\text{MSE}(g_t))$ to motivate clearly how the variance of SVRG⁴ increases within the inner-loop. This has been stated many times already, now come the results.

Remark: By IS for second-era I depict the probabilities $p_{t,i}^* \propto \|\nabla\phi_i(w_t)\|$ just as for the first era methods. As one will observe IS is absolutely not a good fit when using second-era methods. The adaptive sampling explained in HVRG [She+16]⁵, $p_{t,i} \propto \|\beta_{t,i}\|$, is in practice very close to my K -greedy sampling so in order to present clear/ discriminant graphs I've only kept US,IS and K -greedy of **Theorem 2** when analyzing the one-step expectation for second-era methods.

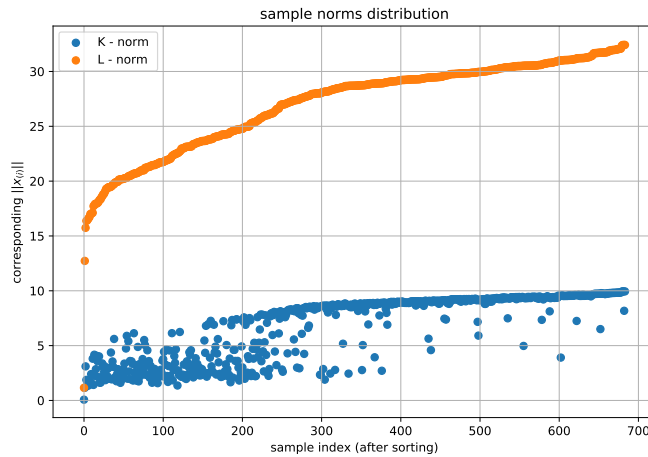


Figure 5.15: Distribution of the norms of the samples $\tilde{x}_i$: K -norm and L -norm (standard euclidean scaled with $\sqrt{L}$)

⁴SAGA did not yield much more satisfying results for my experiments so I decided to stick to SVRG.

⁵Again cfr. end of Chapter 4. This sampling is the true one minimizing V_t for SAGA-like methods.

Quadratic case (1) : first-era (l) vs. second-era (r) — start of inner-loop

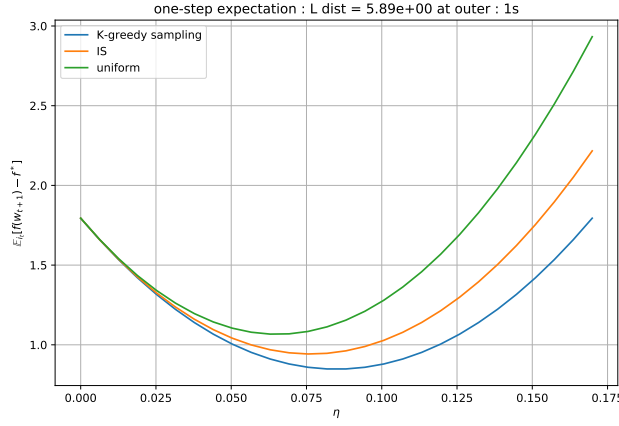


Figure 5.16: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*] : wdbc$

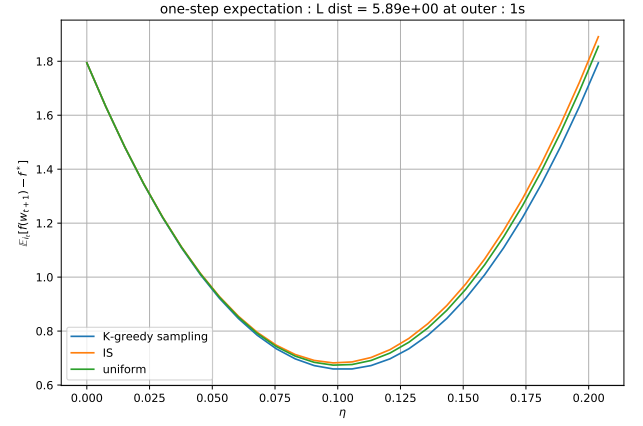


Figure 5.17: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*] : wdbc$

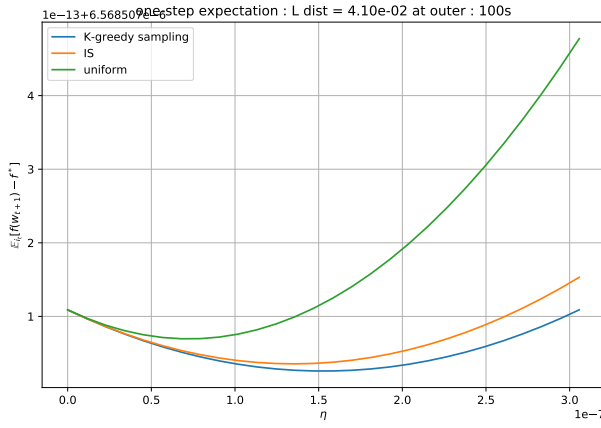


Figure 5.18: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*] : wdbc$

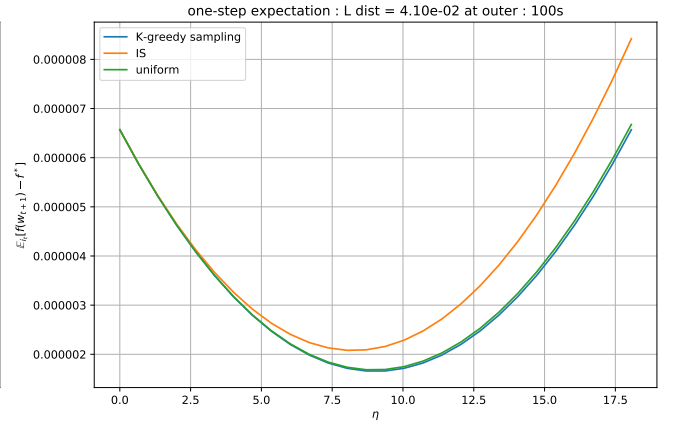


Figure 5.19: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*] : wdbc$

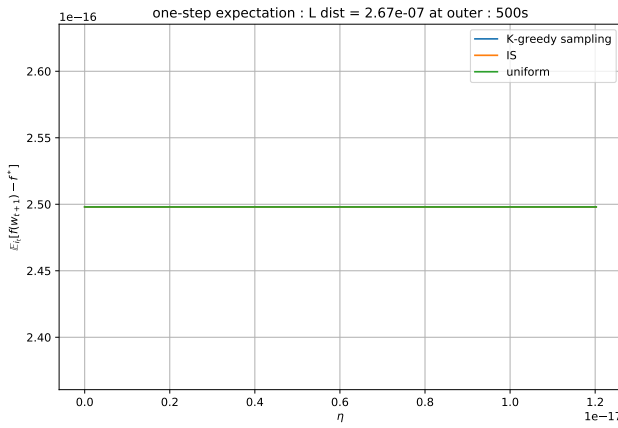


Figure 5.20: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*] : wdbc$

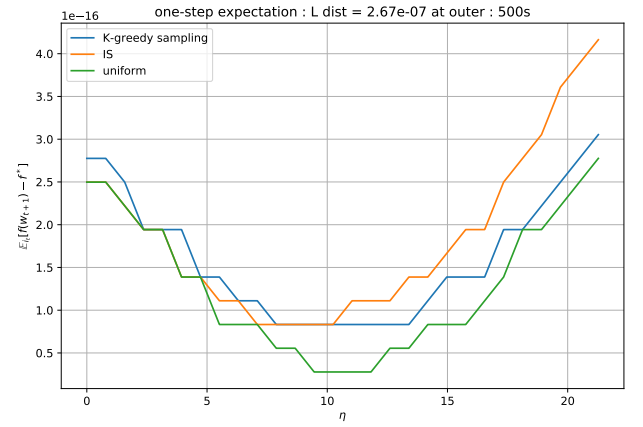


Figure 5.21: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*]$ (numerical instability) : $wdbc$

Quadratic case (1) : second-era — start (l) vs. end (r) of inner-loop

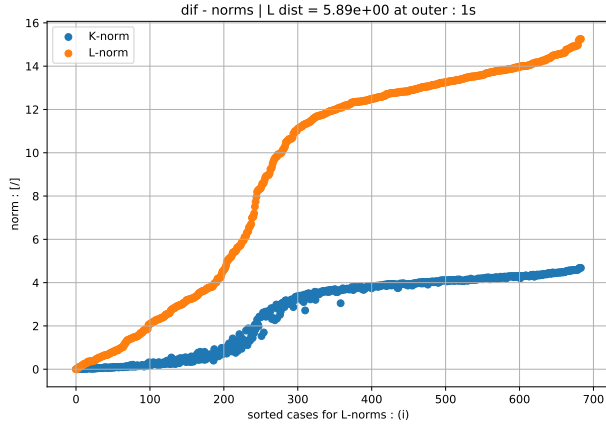


Figure 5.22: $\|\beta_i\|_K$ vs. $\|\beta_i\|_{LI}$: *wdbc*

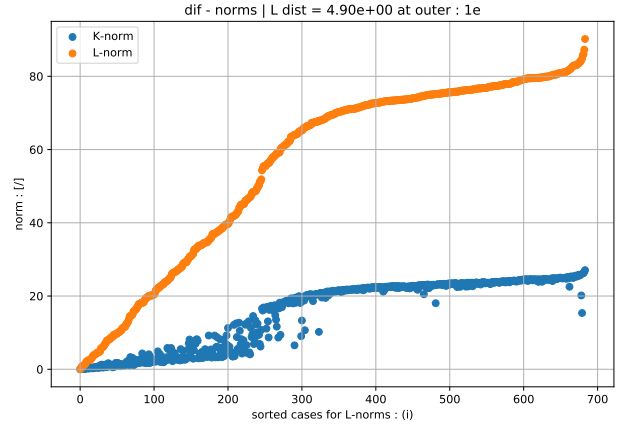


Figure 5.23: $\|\beta_i\|_K$ vs. $\|\beta_i\|_{LI}$: *wdbc*

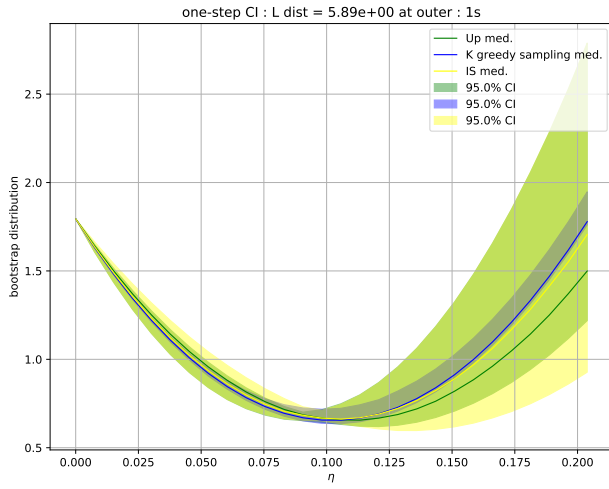


Figure 5.24: 95% bootstrap CI: *wdbc*

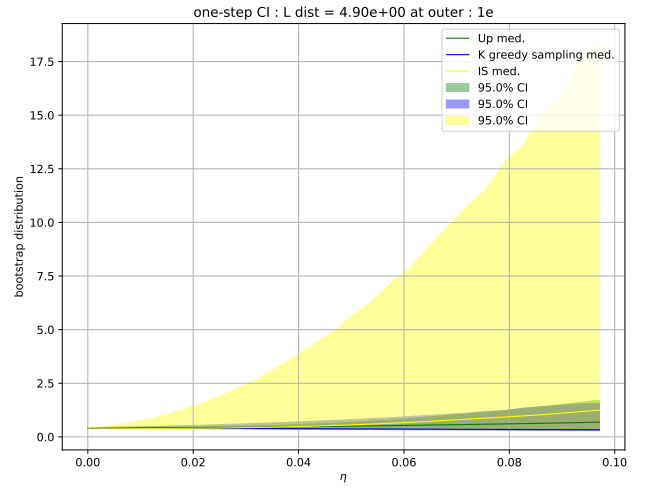


Figure 5.25: 95% bootstrap CI: *wdbc*

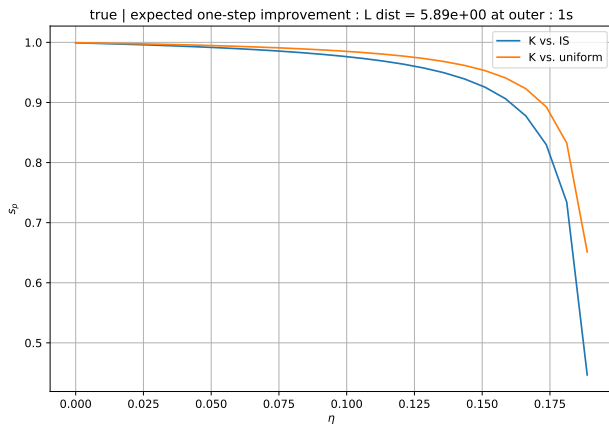


Figure 5.26: expected decrease ratio : *wdbc*

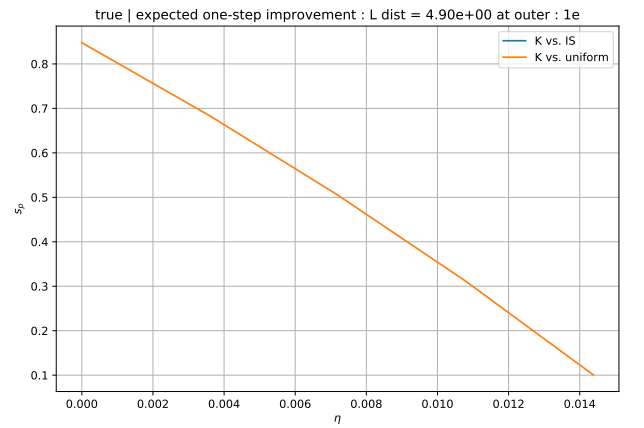


Figure 5.27: expected decrease ratio : *wdbc*

Numerical Results

Quadratic case (1)

inner-loop	state	tr(MSE(g_t))			tr(MSE(g_t))		
		1st-era : US, IS, K -greedy			2nd-era : US, IS, K -greedy		
1	start	26.396	17.270	25.542	0.3520	0.6136	0.1604
	end	10.798	4.7334	6.1122	12.979	428.95	10.583
10	start	4.0918	1.7932	2.1607	1.91e-05	0.0001	1.42e-05
	end	3.9075	1.6998	2.0461	0.0095	0.0685	0.0069
100	start	2.8835	1.3718	1.6567	2.61e-10	1.52e-09	2.06e-10
	end	2.8835	1.3718	1.6567	1.31e-07	5.15e-07	7.68e-08
500	start	2.8847	1.3715	1.6563	8.52e-21	7.91e-20	6.08e-21
	end	2.8847	1.3715	1.6563	4.67e-18	2.36e-17	2.57e-18

Comments

On the first page one can observe that K -greedy sampling, as supposed to in the quadratic case ($\bar{\phi} = \phi$), is leading the game and achieves the best value for $E_{i_t}[f(w_{t+1})]$. It doesn't matter whether we are getting closer or not from the optimum⁶.

However in what concerns first-era methods, a brief glance to the x-axis (η) shows that as the optimization process advances the range of step-sizes ensuring an expected decrease drops drastically. On the second-era side it's all the contrary ! Some existing methods try to adapt the step-size of SGD/SVRG in an online fashion (*Barzilai-Borwein Step Size for Stochastic Gradient Descent* (Tan et al., 2016)) [Tan+16] based on a (Quasi)-Newton approach. The authors also found out that incremental schemes for SVRG step-sizes yields better results. Based on the **GEDE** developments above I will try a new heuristic method increasing progressively the step-size for SVRG/SAGA.⁷ Note that another way to improve SVRG, used by many modern methods (*SVRG++ with Non-uniform Sampling* (Kern & György, 2016)) [KG16] is to extend the inner-loop size ; e.g. double the parameter m at each new outer-loop iteration.

Looking at the graphs on the second page (second-era methods comparison within an inner-loop) one can observe the poor efficiency speed-up for K -greedy sampling at the beginning of an inner-loop. At the end of that same inner-loop we can underline the total inadequacy of IS. Its graph doesn't even appear on the "*expected decrease ratio*" figure since the step-size ensuring an expected decrease for IS is too small. Furthermore the 95% confidence interval built explodes for IS. Nonetheless the gain made against US at the end of an inner-loop is relatively good (straight line graph) !

Looking at the table above the intuition is sustained by the numbers. $\text{tr}(\text{MSE}(g_t))$ decreases as the update iterations grow until a *plateau* is reached for first-era. For second-era methods that same quantity decreases to 0 while looking at the values obtained at each beginning of inner-loop. As forecasted $\text{tr}(\text{MSE}(g_t))$ increases within the inner-loop.

A final interesting comment can be stated about the norms (LI -norm or K -norms) of the $\beta_{t,i}$. Looking at the scale on the left we can see that those norms have swollen in the inner-loop indicating well a bigger slack between each $\nabla\phi_i(w_t)$ and its corresponding $\nabla\phi_i(\theta_{t,i})$.

As indicated in the introduction of this chapter the variance in second-era methods can be larger than the one for first-era at the end of the first inner-loops ; e.g. 1 end.

⁶The reader can take look at the L -distances, representing $\sqrt{L}\|w_t - w_*\|_2$, in the titles.

⁷Please refer to the end of section 5.2.

Smoothed Hinge case (2) : first-era (l) vs. second-era (r) — start of inner-loop

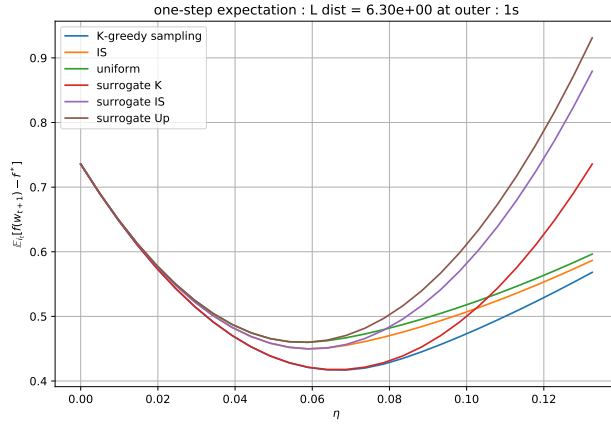


Figure 5.28: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*]$: *wdbc*

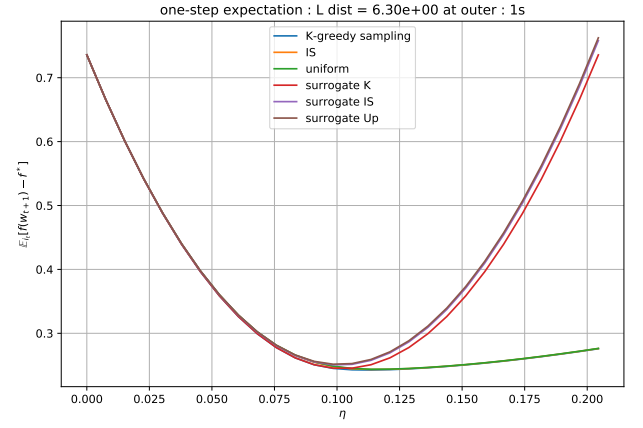


Figure 5.29: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*]$: *wdbc*

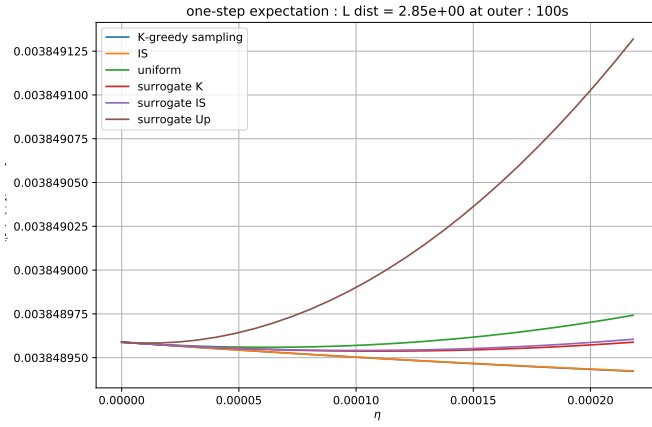


Figure 5.30: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*]$: *wdbc*

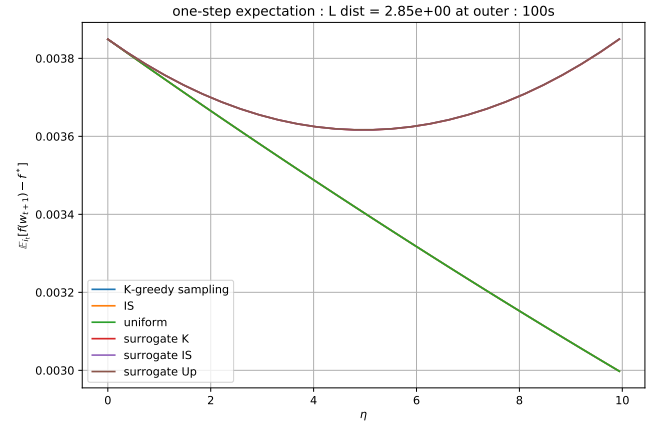


Figure 5.31: $\mathbb{E}_{i_t}[f(w_{t+1}) - f^*]$: *wdbc*

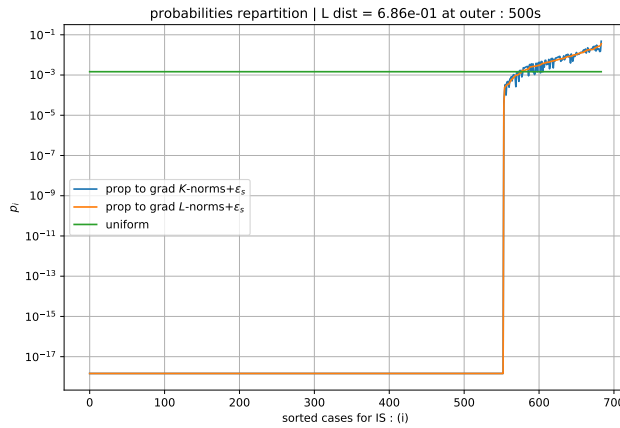


Figure 5.32: sampling probabilities : *wdbc*

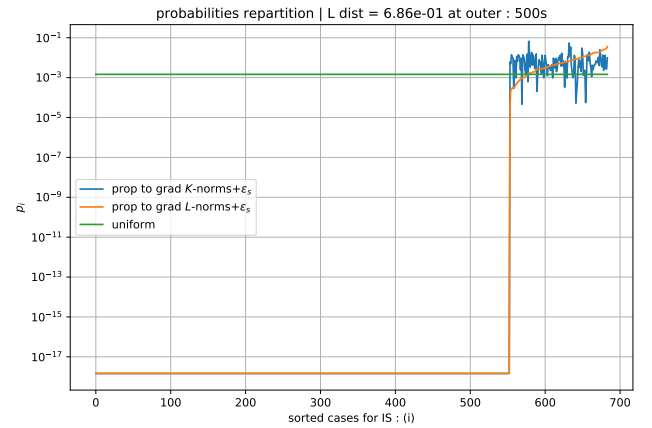


Figure 5.33: sampling probabilities : *wdbc*

Smoothed Hinge case (2) : second-era — start (l) vs. end (r) of inner-loop

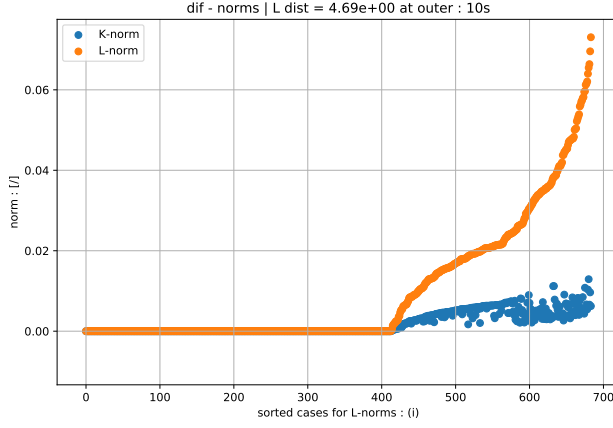


Figure 5.34: $\|\beta_i\|_K$ vs. $\|\beta_i\|_{LI}$: *wdbc*

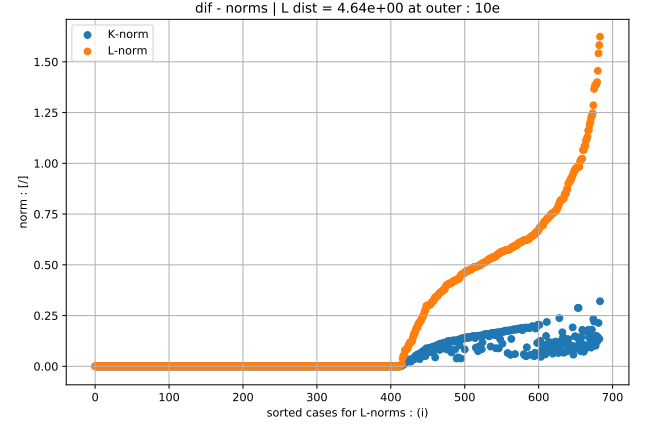


Figure 5.35: $\|\beta_i\|_K$ vs. $\|\beta_i\|_{LI}$: *wdbc*

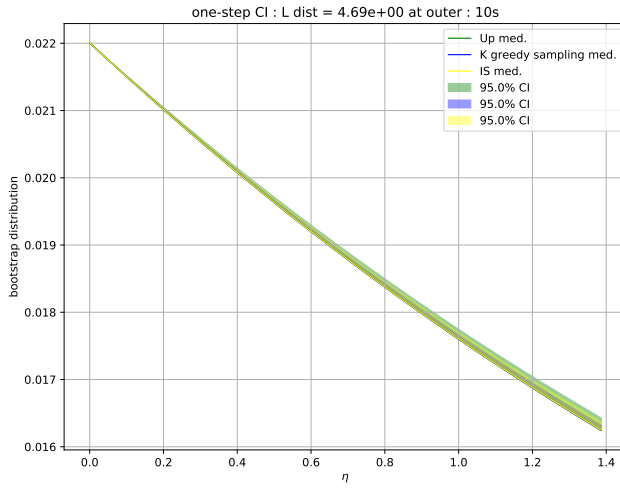


Figure 5.36: 95% bootstrap CI: *wdbc*

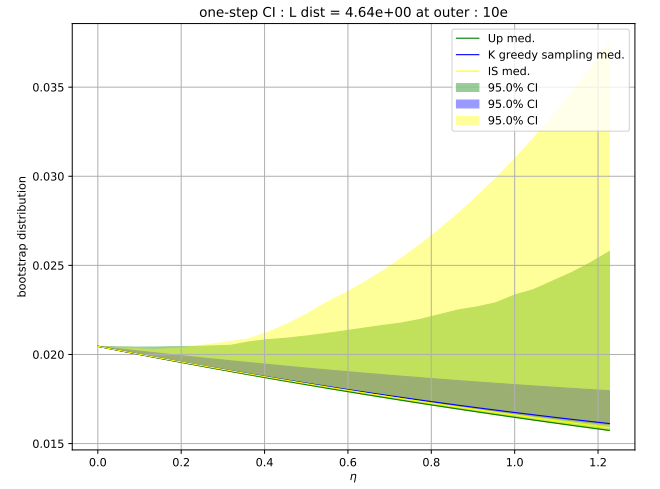


Figure 5.37: 95% bootstrap CI: *wdbc*

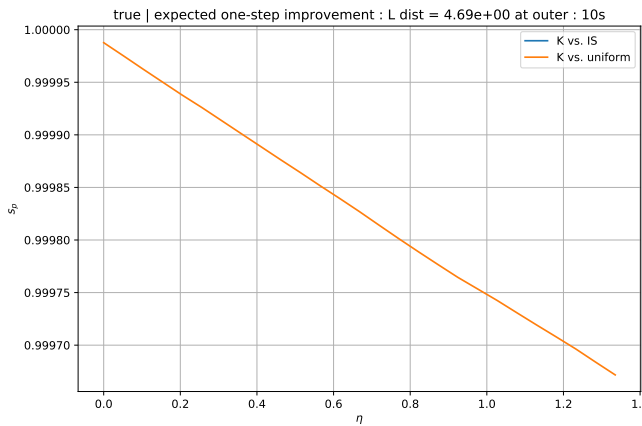


Figure 5.38: expected decrease ratio : *wdbc*

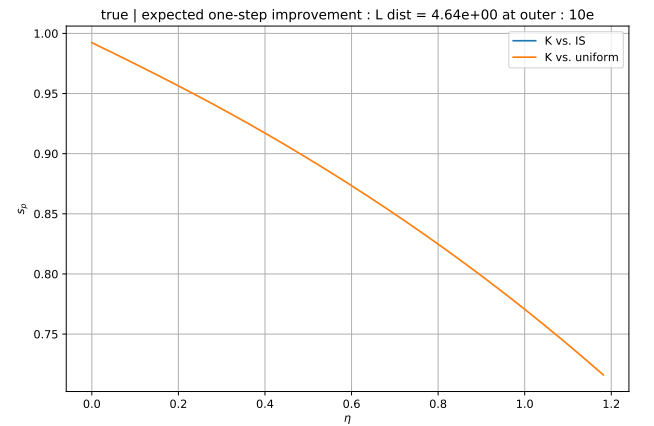


Figure 5.39: expected decrease ratio : *wdbc*

Numerical Results

Smoothed Hinge case (2) / : numerical issue

inner-loop	state	tr(MSE(g_t))			tr(MSE(g_t))		
		1st-era : US, IS, K -greedy			2nd-era : US, IS, K -greedy		
1	start	12.344	12.099	16.855	0.1555	0.1347	0.0719
	end	4.0986	1.2610	1.5677	7.9153	1.2652	5.7434
10	start	1.9336	0.3505	0.4192	3.01e-06	/	1.04e-06
	end	1.9001	0.3280	0.3908	0.0015	/	0.0005
100	start	1.5319	0.1645	0.1832	3.68e-09	4.40e-09	4.12e-10
	end	1.5312	0.1643	0.1830	2.29e-06	3.03e-06	2.51e-07
500	start	1.4568	0.1423	0.1575	7.43e-11	6.48e-11	5.78e-12
	end	1.4568	0.1423	0.1574	4.64e-08	4.07e-08	3.60e-09

Comments

Most of the comments made about the quadratic case (1) experiment hold here as well. Without any surprise K -greedy sampling yields the best one-step expectations for the quadratic surrogates. As the smoothed hinge loss is equivalent to the quadratic loss whenever $\langle \tilde{x}_i, w \rangle \leq 1$ even for the true one-step expectation of ϕ itself K -greedy wins it ; not by much though !

We can see that later during the optimization (after 100 inner-loops for SVRG, hence for a w_t closer to w_*) there's a huge gap between $\mathbb{E}_{i_t}[\bar{\phi}(w_{t+1})]$ and $\mathbb{E}_{i_t}[\phi(w_{t+1})]$ for both eras. K -greedy sampling is out of its "operating" zone for which it was theoretically sustained !

Expected decrease ratios shown after 10 outer-loops do not go further than 75% at the end of the inner-loop whereas at the beginning of the inner-loop one cannot expect to win more than 1% in this configuration using K -greedy sampling. Once again the comparison with IS is not established for the very same reasons as in the quadratic case (1).

One can spot on the graphs about sorted probabilities that for second-era methods K -greedy sampling stabilizes around the neutral probabilities of US ($\frac{1}{n}$). It appears quite clear when looking at K -norms for the $\beta_{t,i}$'s (blue dots).

Conclusion

Those experiments (and many others that are not displayed since they do not really carry extra informations) confirm well the theoretical results derived so far and the intuitive comments that I've done alongside the writing of this document.

As one could have guessed it traditional IS totally fails to capture the randomness source for second-era methods. It seems rather obvious since the stochastic gradient is of the form $g_t = \frac{\nabla \phi_{i_t}(w_t) - \alpha_{t,i_t}}{np_{t,i_t}} + \bar{\alpha}_t$ where $\bar{\alpha}_t$ is a constant for the lottery $i_t \sim \mathcal{P}_t$. HVRG importance probabilities are, in my opinion, the most adequate for second-era methods since they minimize $\text{tr}(\text{MSE}(g_t))$ while being very close from K -greedy sampling ; the difference is not worth mentioning but the computations of K -norms is more expensive !

At this stage K -greedy sampling appears to be a promising research direction for real optimization (multi-steps) for functions ϕ close from quadratics and in the absence of regularization.

Now that K -greedy sampling has been extensively motivated for a one-step expectation minimization procedure let's demonstrate convergence results after several update-steps.

To generate a sequence $\{w_t\}_{t=1}^T$ of iterates we consider the application the General Unbiased Update Scheme (it will be referred as **GUUS**) :

General Unbiased Update Scheme when dealing with problem (1.2)

Require: $\{\eta_t\}_{t \in \mathbb{N}}$ step-size schedule, $\{\mathcal{P}_t\}_{t \in \mathbb{N}}$ univariate distributions over $[n]$, $\{\mathcal{O}_t\}_{t \in \mathbb{N}}$ first-order stochastic optimization oracle yielding, $\forall t \in \mathbb{N}$, an unbiased estimate g_t of $\nabla \phi(w_t)$, $w_0 \in \mathbb{R}^d$ starting iterate, $t = 0$
while stopping criterion not satisfied **do**
 $g_t \leftarrow \mathcal{O}_t$ — Call of Oracle
 $w_{t+1} \leftarrow \arg \min_{w \in \mathbb{R}^d} \left[\phi(w_t) + \langle g_t, w - w_t \rangle + \frac{1}{\eta_t} \mathcal{B}_\psi(w, w_t) + \lambda r(w) \right]$ — Surrogate Optimization
 $t \leftarrow t + 1$
end while

Note 1: Let's remind that $\mathcal{B}_\psi(\cdot, \cdot)$ represents Bregman's divergence introduced in Chapter 2. ψ is a differentiable convex function, by default $\psi(\cdot) = \frac{1}{2} \|\cdot\|_2^2$.

Note 2: When $\psi(\cdot)$ takes its default value, first and second era methods can be translated in the **GUUS** framework as the update step resumes in a proximal update step. Indeed in Chapter 2 it's proved that $\mathcal{B}_{\frac{1}{2}\|\cdot\|^2}(u, v) = \frac{1}{2} \|u - v\|^2$, $\forall u, v \in \mathbb{R}^d$. Regarding **Definition 6** of Chapter 2 and the note that succeeds it, the "*Surrogate Optimization*" step above is equivalent $\text{prox}_{\eta_t \lambda r}(w_t - \eta_t g_t)$.

What differentiates both eras under the scope of the **GUUS** is the way the oracle $\mathcal{O}_t$ outputs a stochastic gradient. We've seen in Chapter 4 that second-era methods use an oracle that rely on past computed information (sample gradients at snapshots) whereas first-era methods work purely online, depending only on the current iterate w_t .

Master Convergence Theorem

In this subsection is presented a first, rather general, proof of convergence for size-1 mini-batch updates for unbiased stochastic optimization methods. This latter proof includes the proximal setting since it deals with the **GUUS**. It will be useful in three different ways:

1. Showing that using first/second era methods under any *proper* sampling distribution, hence including K -greedy sampling, leads to convergence in expectation under an adequate choice of step-sizes.
2. Showing that the knowledge of such a metric defining matrix K allows to enhance the convergence bound found by [ZZ15]. However it requires more assumptions than in [ZZ15]; one cannot simply say that that convergence bound is better since it does not apply in all the contexts handled in [ZZ15]. It is more about a refinement.
3. Opening the door to the $\tilde{K}$ Prox-VR framework that constitutes the second part of this first topic of Chapter 5.

Let's warn the reader that the generalized versions of smoothness and convexity, introduced in Chapter 2, are used in the upcoming proof in order to prepare the groundwork for K Prox-VR.

Substantially inspired from the proof of **Theorem 1** from [ZZ15], I derive in **Theorem 3** a slightly more general convergence bound that holds for both eras and more types of step-size schedules.

Then I present a lemma requiring more assumptions but underlining a potential speed-up using K -greedy sampling. Latter's lemma demonstration will follow the same strategy using a specific *Bregman's* divergence and other types of lower/upper bounds for ϕ than the one provided by classical strong-convexity and smoothness assumptions respectively. Finally I present a simple lemma treating about the specific case of $f = \phi$ that is quadratic.

It should be reminded that applying K -greedy sampling or IS at each update-iteration is still not tractable ! The analysis remains purely theoretical.

Theorem 3 (GUUS methods convergence under smoothness and strong convexity assumptions) Be ψ a differentiable σ -strongly convex function with respect to a norm $\|\cdot\|$ and ϕ is L -smooth and μ -strongly convex w.r.t ψ . Also be r convex and such that $r(0) = 0$, $r(w) \geq 0$, $\forall w \in \mathbb{R}^d$. Then if the sequence $\{w_t\}_{t=0}^T$ ($T \geq 1$) has been generated according to the GUUS with $\{\eta_t\}_{t=0}^{T-1}$ such that $\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} \leq \mu$ and $\eta_t \in]0, \frac{1}{L}] \forall t \in \{0, \dots, T-1\}$, it holds that

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{1}{T} \left[\frac{1}{\eta_0} \mathcal{B}_\psi(w_*, w_0) + \mathbb{E} \sum_{t=0}^{T-1} \frac{\eta_t V_t}{\sigma} \right]^a \quad (5.9)$$

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \left[\frac{1}{\eta_0} \mathcal{B}_\psi(w_*, w_0) + \mathbb{E} \sum_{t=0}^{T-1} \frac{\eta_t (-\|\nabla \phi(w_t)\|^2 + \mathbb{E}_{i_t}[\|g_t\|^2])}{\sigma} \right] \quad (5.10)$$

with $\bar{w}_T := \frac{1}{T} \sum_{t=1}^T w_t$.

^aLet's recall the notation $V_t = \mathbb{E}_{S_t}[\|g_t - \mathbb{E}_{S_t}[g_t]\|^2]$ equivalent to $\mathbb{E}_{i_t}[\|g_t - \nabla \phi(w_t)\|^2]$ in the present context.

Proof : In any case w_t, w_* and w_{t+1} are in $\mathbb{R}^d$. Using the available QLB defined by $\mathcal{B}_\psi$ for ϕ and the convexity of r :

$$f(w_*) \geq \phi(w_t) + \langle \nabla \phi(w_t), w_* - w_t \rangle + \mu \mathcal{B}_\psi(w_*, w_t) + \lambda r(w_{t+1}) + \lambda \langle \partial r(w_{t+1}), w_* - w_{t+1} \rangle$$

where here $\partial r(w_{t+1})$ represents one particular subgradient contained in the subdifferential set at iterate w_{t+1} for the regularizer. A QUB for ϕ , also defined by $\mathcal{B}_\psi$, implies that :

$$\phi(w_t) \geq \phi(w_{t+1}) - \langle \nabla \phi(w_t), w_{t+1} - w_t \rangle - L \mathcal{B}_\psi(w_{t+1}, w_t)$$

Combining the two last inequalities leads to :

$$f(w_{t+1}) - f^* \leq \langle \nabla \phi(w_t) + \lambda \partial r(w_{t+1}), w_{t+1} - w_* \rangle + L \mathcal{B}_\psi(w_{t+1}, w_t) - \mu \mathcal{B}_\psi(w_*, w_t)$$

The usual optimality conditions on the update step

$$w_{t+1} = \arg \min_{w \in \mathbb{R}^d} [\phi(w_t) + \langle g_t, w - w_t \rangle + \frac{1}{\eta_t} \mathcal{B}_\psi(w, w_t) + \lambda r(w)]$$

imply consequently that

$$\nabla \psi(w_t) - \nabla \psi(w_{t+1}) - \eta_t g_t \in \eta_t \lambda \partial r(w_{t+1})$$

We can rewrite the last inequality substituting $\lambda \partial r(w_{t+1})$ by $[\nabla \psi(w_t) - \nabla \psi(w_{t+1})]/\eta_t - g_t$.

Let's first remark that

$$\begin{aligned} \langle \nabla \psi(w_t) - \nabla \psi(w_{t+1}), w_* - w_{t+1} \rangle &= \psi(w_*) - \psi(w_{t+1}) - \langle \nabla \psi(w_{t+1}), w_* - w_{t+1} \rangle + \\ &\quad \psi(w_{t+1}) - \psi(w_t) - \langle \nabla \psi(w_t), w_{t+1} - w_t \rangle - \\ &\quad (\psi(w_*) - \psi(w_t) - \langle \nabla \psi(w_t), w_* - w_t \rangle) \\ &= \mathcal{B}_\psi(w_*, w_{t+1}) + \mathcal{B}_\psi(w_{t+1}, w_t) - \mathcal{B}_\psi(w_*, w_t) \end{aligned}$$

We have,

$$f(w_{t+1}) - f^* \leq \langle \nabla \phi(w_t) - g_t, w_{t+1} - w_* \rangle + \frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1}) - \mathcal{B}_\psi(w_{t+1}, w_t)] \\ + L\mathcal{B}_\psi(w_{t+1}, w_t) - \mu\mathcal{B}_\psi(w_*, w_t)$$

As this inequality holds for any valid choice of $w_*, w_t, w_{t+1} \in \mathbb{R}^d$ it must hold in the overall expectation $\mathbb{E}[\cdot]$. Therefore one can write :

$$\mathbb{E}[f(w_{t+1}) - f^*] \leq \mathbb{E}[\langle \nabla \phi(w_t) - g_t, w_{t+1} - w_* \rangle + \frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1}) - \mathcal{B}_\psi(w_{t+1}, w_t)] \\ + L\mathcal{B}_\psi(w_{t+1}, w_t) - \mu\mathcal{B}_\psi(w_*, w_t)]$$

One can show (cfr. **Appendix** of [ZZ15]) that $\mathbb{E}[\langle \nabla \phi(w_t) - g_t, w_{t+1} - w_* \rangle] \leq \frac{\eta_t}{\sigma} \mathbb{E}[V_t]$ for any kind of g_t as long as $\mathbb{E}[g_t | \mathcal{F}_t] = \nabla \phi(w_t)$; including both eras unbiased methods.

$$\mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{\eta_t}{\sigma} \mathbb{E}[V_t] + \mathbb{E}[\frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1}) - \mathcal{B}_\psi(w_{t+1}, w_t)] + L\mathcal{B}_\psi(w_{t+1}, w_t) - \mu\mathcal{B}_\psi(w_*, w_t)] \quad (5.11)$$

Since $\forall t \in \{0, \dots, T-1\}$, $-\frac{1}{\eta_t} + L \leq 0$ and for any arguments $u, v \in \mathbb{R}^d$, $\mathcal{B}_\psi(u, v) \geq 0$:

$$\mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{\eta_t}{\sigma} \mathbb{E}[V_t] + \mathbb{E}[\frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1})] - \mu\mathcal{B}_\psi(w_*, w_t)]$$

Using our assumption that $\forall t \in \{0, \dots, T-1\}$, $\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} \leq \mu$, let's sum up the last inequality :

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{1}{T} \sum_{t=0}^{T-1} \frac{\eta_t}{\sigma} \mathbb{E}[V_t] + \mathbb{E}[\frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1})] - \mu\mathcal{B}_\psi(w_*, w_t)] \\ = \frac{1}{T} \frac{1}{\eta_0} \mathcal{B}_\psi(w_*, w_0) + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[(\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} - \mu) \mathcal{B}_\psi(w_*, w_{t+1})] + \\ \frac{1}{T} \sum_{t=0}^{T-1} \frac{\eta_t}{\sigma} \mathbb{E}[V_t] - \frac{1}{T} (\mu \mathcal{B}_\psi(w_*, w_0) + \frac{1}{\eta_{T-1}} \mathcal{B}_\psi(w_*, w_T))$$

Jensen's inequality for convex functions allows us to conclude

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{1}{T} \left[\frac{1}{\eta_0} \mathcal{B}_\psi(w_*, w_0) + \mathbb{E} \sum_{t=0}^{T-1} \frac{\eta_t}{\sigma} V_t \right]$$

□

Notes : A working practical choice is to set $\eta_t = \frac{1}{\alpha + \mu t}$ for every $t \in \mathbb{N}$ with $\alpha \geq 0$.

As argued multiple times in the first chapters of this master thesis, one can observe an immediate impact of $\text{tr}(\text{MSE}(g_t)) (= V_t$ for unbiased updates) on the upper-bound derived for $\mathbb{E}[f(\bar{w}_T) - f^*]$. Furthermore one retrieves well the quantities $\eta_t V_t$ introduced in Chapter 4. In order to ensure convergence in expectation for the above theorem $\frac{1}{T} \sum_{t=0}^{T-1} \frac{\eta_t V_t}{\sigma}$ must be asymptotically arbitrary small for $T \rightarrow \infty$.

Since $V_t \not\rightarrow 0$ as $t \rightarrow \infty$ for first-era methods (experiments showed a stabilization of V_t),

$$\sum_{t=0}^{T-1} \eta_t \in o(T)$$

For second-era methods there exist specific theorems proving a linear convergence since V_t decrease itself fast enough ; η_t might be kept constant as stated in the *Variance Reduction Era* section of Chapter 4.

Lemma 7 (GUUS methods convergence under quadratically sandwiched ϕ) Be ψ defined for every $u \in \mathbb{R}^d$ by $\psi(u) = \frac{1}{2}\|u\|^2$. Hence ψ is $\sigma = 1$ -strongly convex. Be ϕ convex and differentiable. Let us assume that there's no regularization ($r(w) = 0, \forall w \in \mathbb{R}^d$) and that there exists two matrices $\mathbf{0} \preceq K^- = (K^-)^T \preceq K^+ = (K^+)^T$ such that $\forall u, v \in \mathbb{R}^d$

$$\phi(u) + \langle \nabla \phi(u), u - v \rangle + \frac{1}{2} \langle K^-(u - v), u - v \rangle \leq \phi(v) \leq \phi(u) + \langle \nabla \phi(u), u - v \rangle + \frac{1}{2} \langle K^+(u - v), u - v \rangle$$

Then if the sequence $\{w_t\}_{t=0}^T$ ($T \geq 1$) has been generated according to the GUUS with $\{\eta_t\}_{t=0}^{T-1}$ such that $K^- - \max_{t \in \{0, \dots, T-1\}} \left\{ \frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} \right\} I \succeq \mathbf{0}$ (I), it holds that

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \left[\frac{1}{\eta_0} \frac{\|w_0 - w_*\|^2}{2} + \mathbb{E} \sum_{t=0}^{T-1} \eta_t (-\|\nabla \phi(w_t)\|^2 + \mathbb{E}_{i_t} [\|g_t\|_{\frac{\eta_t K^+ + I}{2}}^2]) \right] \quad (5.12)$$

Proof : Until equation (5.11) the proof of the above lemma is exactly identical to the proof of **Theorem 3**. One exception to that, instead of the terms $L\mathcal{B}_\psi(w_{t+1}, w_t)$ and $\mu\mathcal{B}_\psi(w_*, w_t)$

we should write $\frac{\|w_{t+1} - w_t\|_{K^+}^2}{2}$ and $\frac{\|w_t - w_*\|_{K'}^2}{2}$ respectively using the QLB/QUB at hand.

Hence let's rewrite (5.11) :

$$\mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{\eta_t}{\sigma} \mathbb{E}[V_t] + \mathbb{E} \left[\frac{1}{\eta_t} [\mathcal{B}_\psi(w_*, w_t) - \mathcal{B}_\psi(w_*, w_{t+1}) - \mathcal{B}_\psi(w_{t+1}, w_t)] + \frac{\|w_{t+1} - w_t\|_{K^+}^2}{2} - \frac{\|w_t - w_*\|_{K'}^2}{2} \right]$$

Using the fact that the *Bregman* divergence here simply corresponds to the squared euclidean distance divided by 2 (neutral case) as $\psi(\cdot) = \frac{1}{2}\|\cdot\|^2$:

$$\mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{\eta_t}{\sigma} \mathbb{E}[V_t] + \mathbb{E} \left[\frac{1}{\eta_t} \left[\frac{\|w_t - w_*\|^2}{2} - \frac{\|w_{t+1} - w_*\|^2}{2} - \frac{\|w_{t+1} - w_t\|^2}{2} \right] + \frac{\|w_{t+1} - w_t\|_{K^+}^2}{2} - \frac{\|w_t - w_*\|_{K'}^2}{2} \right]$$

As $\mathbb{E}[V_t]$ is simply equal to $\mathbb{E}[-\|\nabla \phi(w_t)\|^2 + \mathbb{E}_{i_t}[\|g_t\|^2]]$ and given the fact that a single GUUS step is equivalent to set $w_{t+1} = w_t - \eta_t g_t$ when there's no regularization and under a neutral *Bregman* divergence : $\frac{\eta_t}{\sigma} \mathbb{E}[V_t] - \frac{1}{\eta_t} \mathbb{E} \left[\frac{\|w_{t+1} - w_t\|^2}{2} \right] = \mathbb{E} \left[-\frac{\eta_t}{\sigma} \|\nabla \phi(w_t)\|^2 + \eta_t \left(\frac{1}{\sigma} - \frac{1}{2} \right) \mathbb{E}_{i_t}[\|g_t\|^2] \right]$

$$\begin{aligned} \mathbb{E}[f(w_{t+1}) - f^*] &\leq \mathbb{E} \left[\frac{\sigma \eta_t^2 \|g_t\|_{K^+}^2 + \eta_t \frac{2-\sigma}{2} \|g_t\|^2 - \eta_t \|\nabla \phi(w_t)\|^2}{\sigma} + \right. \\ &\quad \left. \frac{1}{\eta_t} \mathbb{E} \left[\frac{\|w_t - w_*\|^2 - \|w_t - w_*\|_{\eta_t K^-}^2 - \|w_{t+1} - w_*\|^2}{2} \right] \right] \end{aligned}$$

Again summing up this inequality from $t = 0$ to $T - 1$ and using *Jensen's* inequality yields :

$$\begin{aligned} \mathbb{E}[f(\bar{w}_T) - f^*] &\leq \frac{1}{T} \left[\mathbb{E} \sum_{t=0}^{T-1} \frac{\eta_t (-\|\nabla \phi(w_t)\|^2 + \mathbb{E}_{i_t} [\|g_t\|_{\frac{\sigma \eta_t K^+ + I(2-\sigma)}{2}}^2])}{\sigma} + \frac{1}{\eta_0} \frac{\|w_0 - w_*\|^2}{2} \right. \\ &\quad \left. - \frac{1}{\eta_T} \mathbb{E} \left[\frac{\|w_T - w_*\|^2}{2} \right] + \sum_{t=0}^{T-1} \mathbb{E} \left[\frac{\|w_{t+1} - w_*\|_{\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} - K^-}^2}{2} \right] \right] \end{aligned}$$

Finally using our last assumption (I) and the fact that $\sigma = 1$, one can conclude. $\square$

Notes : It's important to notice that the above lemma applies to more contexts than just the quadratic case ! Indeed let's imagine the sum of a logistic loss function $\frac{1}{n} \sum_{i=1}^n \log(1 + e^{-\langle \tilde{x}_i, \cdot \rangle})$ or even the smoothed hinge loss $\frac{1}{n} \sum_{i=1}^n ([1 - \langle \tilde{x}_i, \cdot \rangle]_+)^2$, with an inner convexity enhancer term $\frac{\lambda_{\text{in}} \|\cdot\|^2}{2}$. In both cases there exist such K^- and K^+ ! K^- would be at worse $\lambda_{\text{in}} I$ while K^+ can take the value, as previously shown, $\lambda_{\text{in}} I + \frac{c}{n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T$ ($c = 2$ for the quadratic case and the smoothed hinge loss while $c = \frac{1}{4}$ for the logistic loss).

It's easy to observe that whenever $\eta_t \leq \|K^+\|_2$ for all $t \in \{0, \dots, T-1\}$ the bound in (5.12) is stronger than the one formulated in (5.10), in the same context of neutral *Bregman* divergence and without regularization.

At any update iteration $t \in \mathbb{N}$, one can furthermore tune the importance probabilities $p_{t,i}$ and use $K_t = \frac{\eta_t K^+ I}{2}$ -greedy sampling in order to achieve the best value for $\mathbb{E}_{i_t}[\|g_t\|_{\frac{\eta_t K^+ I}{2}}^2]$.

One might be surprised to find out that K -greedy sampling for the last lemma does not simply correspond to $K_t = K^+$. Once again the class of functions subject to the last lemma is larger than pure quadratics as $\bar{\phi}$. If one focuses on $\bar{\phi}$ the following straight forward lemma applies thanks to the **GDE** derived above.

Lemma 8 (GUUS methods convergence under quadratic $\phi = \bar{\phi}$) Be ψ defined for every $u \in \mathbb{R}^d$ by $\psi(u) = \frac{1}{2}\|u\|^2$. Hence ψ is $\sigma = 1$ -strongly convex. Be ϕ a quadratic convex and differentiable function defined for every $w \in \mathbb{R}^d$ as $\phi(w) = \kappa + \langle k, w \rangle + \frac{1}{2}\langle Kw, w \rangle$ and let us assume that there's no regularization ($r(w) = 0, \forall w \in \mathbb{R}^d$). Then if the sequence $\{\eta_t\}_{t=0}^T$ ($T \geq 1$) has been generated according to the GUUS with $\{\eta_t\}_{t=0}^{T-1}$, it holds that

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \left[f(w_0) - f^* + \mathbb{E} \sum_{t=0}^{T-1} \eta_t (-\|\nabla \phi(w_t)\|^2 + \frac{\eta_t}{2} \mathbb{E}_{i_t}[\|g_t\|_K^2]) \right] \quad (5.13)$$

Proof: First let's highlight that $\phi = \bar{\phi} = f$ in the present case. Using the GEDE, taking the overall expectation at any $w_t \in \mathbb{R}^d$ and considering the update step $i_t \sim \mathcal{P}_t$, $w_{t+1} = w_t - \eta_t g_{t,i_t}$,

$$\mathbb{E}[f(w_t - \eta_t g_t)] = \mathbb{E}[f(w_{t+1})] = \mathbb{E}[f(w_t)] - \mathbb{E} \left[\eta_t [\langle k + Kw_t, g_t \rangle - \frac{\eta_t}{2} \langle Kg_t, g_t \rangle] \right]$$

Since $k + Kw_t$ is then equal to $\nabla \phi(w_t)$ and injecting the unbiasedness of g_t one can write

$$\begin{aligned} \mathbb{E}[f(w_{t+1})] &= \mathbb{E}[f(w_t)] - \mathbb{E} \left[\eta_t [\|\nabla \phi(w_t)\|^2 - \frac{\eta_t}{2} \mathbb{E}_{i_t}[\|g_t\|_K^2]] \right] \\ \mathbb{E}[f(w_{t+1})] - \mathbb{E}[f(w_t)] &= \mathbb{E} \left[\eta_t \left[\frac{\eta_t}{2} \mathbb{E}_{i_t}[\|g_t\|_K^2] - \|\nabla \phi(w_t)\|^2 \right] \right] \end{aligned}$$

Summing up those equations and using *Jensen's* inequality one last time :

$$\mathbb{E}[f(\bar{w}_T) - f^*] \leq \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[f(w_{t+1}) - f^*] \leq \frac{1}{T} \left[f(w_0) - f^* + \mathbb{E} \sum_{t=0}^{T-1} \eta_t [-\|\nabla \phi(w_t)\|^2 + \frac{\eta_t}{2} \mathbb{E}_{i_t}[\|g_t\|_K^2]] \right]$$

□

Note: Once more the quantity $\mathbb{E}_{i_t}[\|g_t\|_K^2] = \sum_{i=1}^n p_{t,i} \|g_{t,i}\|_K^2$ arises in a worst-case bound and is minimized using K -greedy sampling thanks to the master lemma.

Practical Issues and Possible Solutions

Just as for the sampling defined by p_t^* the main bottleneck of an adaptive p_t^K , even for quadratic functions, is the computation of the probabilities at each iteration requiring generally the computational time of a full-gradient... Nevertheless it might not be unreasonable to update the probabilities time by time and eventually opting for an adaptive strategy within the inner-loop.

Thus the idea would be to get either p^K or p^* once every m update-iterations and then gently decrease the difference between each $p_{t,i}$ and $\frac{1}{n}$ (US) as the inner-loop iterations advance.

This adaptive scheme progressively lowers the discrepancy with respect to (US) defined as

$$\mathcal{D}(p_t) = \frac{\sum_{i=1}^n |p_{t,i} - \frac{1}{n}|}{\frac{1}{n}} = n \|p_t - \frac{\mathbf{e}_n}{n}\|_1 \quad (5.14)$$

The smaller the value of $\mathcal{D}_t$ the less importance sampling might yield either extreme decrease or extreme increase of the objective function compared to the neutral uniform sampling.

Here below I provide one example of an algorithm that may overcome the cited practical issues, at least partially. One remaining bottleneck, just as for HVRG [She+16], is to require a second full-pass on the data each time the inner-loop iteration counter takes the value 1.

Aside from that fact, let A be a set of inner-loop iterations indices (e.g. $A := \{\frac{m}{4}, \frac{m}{2}, \frac{3m}{4}\}$). Each time the inner-loop iteration counter (here represented by $(t+1) \bmod m$) is contained in A , the univariate distribution $\mathcal{P}_t$ computed at the beginning of the inner-loop is updated.

For instance, if one enters the "*probability update condition*" for the x -th time in the current inner-loop then the probability distribution used for the sampling of the next iterate updates (until entering again the "*probability update condition*") may be defined by the vector

$$p_{t,\text{new}}(x) = \frac{|A| - x}{|A|} p_t + \frac{x}{|A|} \frac{\mathbf{e}}{n}$$

The discrepancy of $p_{t,\text{new}}(x)$, $D(p_{t,\text{new}}(x))$ decreases with x .

Algorithm 11 K -greedy sampling Prox-SVRG

Initialization : $\eta_t = \eta > 0 \forall t \in \mathbb{N}$, w_0 starting iterate, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla \phi_i(w_0)$: *warm start* or $\mathbf{0}$), m number of inner-loop iterations
while stopping criterion not verified **do**
 $\tilde{w} \leftarrow w_t$ (or average over the last inner-loop)
 $\alpha_{t,i} \leftarrow \nabla \phi_i(\tilde{w}) \forall i \in [n]$
 for $(t+1) \bmod (m+1) \neq 0$ **do**
 if $t+1 \bmod m = 2$ **then**
 $p_{t,i}$ for every $i \in [n]$ defining $\mathcal{P}_t$ is computed as in (5.8)
 end if
 $i_t \sim \mathcal{P}_t$
 $g_t \leftarrow \bar{\alpha}_t + \frac{\nabla \phi_{i_t}(w_t) - \alpha_{t,i_t}}{np_{t,i_t}}$
 $w_{t+1} \leftarrow \text{prox}_{\eta_t R}(\eta w_t - \eta_t g_t)$
 // probability update condition
 if $(t+1) \bmod m \in A$ **then**
 the sampling probabilities are updated
 end if
 $t \leftarrow t+1$
 end for
end while
Output : w_T (last iterate) or $w_T \leftarrow \frac{1}{T} \sum_{t=0}^{T-1} w_t$ (averaged iterate)

Note: Unfortunately, none of my experiments has been worth it so far. This door is left open.

5.2 $\tilde{K}$ -ProxVR

Initial motivation and related work

The **GUUS** presented in the last section does not yield obvious, practical & tractable cases apart from taking $\psi(\cdot)$ "neutral" (leading to the euclidean *Bregman* divergence). Yet, when one deals with a ϕ function having properties such as in **Lemma 7**, it is possible to refine the **GUUS** with

$$\mathcal{B}_\psi(w, w_t) = \frac{1}{2} \|w - w_t\|_{K^+}^2 = \frac{1}{2} \langle K^+(w - w_t), w - w_t \rangle \quad (5.15)$$

The (semi-)positive definition of K^+ allows one to use *Cholesky*'s decomposition in order to build a lower triangular matrix C^+ (unique if $K^+ \succ \mathbf{0}$) that is (semi-)positive definite.

$$K^+ = C^+(C^+)^T$$

Taking $\psi(\cdot) = \frac{1}{2} \|(C^+)^T \cdot\|^2 = \frac{1}{2} \langle (C^+)^T \cdot, (C^+)^T \cdot \rangle = \frac{1}{2} \langle K^+ \cdot, \cdot \rangle$ leads to (5.15).

Indeed ψ is a convex differentiable function and $\forall u \in \mathbb{R}^d$, $\nabla \psi(u) = K^+ u$. As a result,

$$\mathcal{B}_\psi(u, v) = \psi(u) - \psi(v) - \langle \nabla \psi(v), u - v \rangle = \frac{1}{2} [\langle K^+ u, u \rangle - \langle K^+ v, v \rangle - 2 \langle K^+ v, u - v \rangle] = \frac{1}{2} \|u - v\|_{K^+}^2$$

Given a stochastic gradient realization g_t and a step-size η_t , under the latter *Bregman* divergence, a single **GUUS** step trivially becomes a quadratic surrogate minimization.

$$w_{t+1} \leftarrow \arg \min_{w \in \mathbb{R}^d} \left[\phi(w_t) + \langle g_t, w - w_t \rangle + \frac{1}{2\eta_t} \|w - w_t\|_{K^+}^2 + \lambda r(w) \right] \quad (5.16)$$

In the absence of regularization one can retrieve the stochastic equivalent of full *Newton*'s method when K^+ is the Hessian of ϕ : $w_{t+1} \leftarrow w_t - \eta_t (K^+)^{-1} g_t$. Even though we might expect a better convergence rate than a sublinear one, the **Master Convergence Theorem** already ensures that such update iterations imply that the value $\mathbb{E}[f(\bar{w}_T) - f^*]$ converge to 0 with an appropriate step-size schedule.

Sadly for this master thesis there exists plenty of papers about stochastic *Quasi-Newton* methods; e.g. *Accelerating SVRG via second-order information* (Kolte et al., 2015) [KEÖ15], *A Stochastic Quasi-Newton Method for Large-Scale Optimization* (Byrd et al., 2015) [Byr+15], *Stochastic L-BFGS: Improved Convergence Rates and Practical Acceleration Strategies* (R. Zhao, Haskell and Tan, 2018) [ZHT17] to mention just a few of them. This last paper is the most advanced I've seen concerning this topic; it combines L-BFGS updates alongside with non-uniform subsampling for the creation of both the inverse Hessian approximation and a second-order stochastic gradient.

Nevertheless none of the cited papers seems to handle the setting (5.16) at its root.⁸ Indeed, in the presence of a regularizer, their stochastic gradient g_t is then naturally taken as a subgradient in the subdifferential set $\nabla \phi_{i_t}(w_t) + \partial r(w_t)$ for $i_t \sim \mathcal{P}_t$.

Hence there's room for a new approach :

Tuning a matrix $\tilde{K} \succeq K^+$ suitable for *light-cost* pseudo proximal updates.

One criterion will be given in the upcoming pages in order to fix $\tilde{K}$ properly.

⁸However projected versions of a stochastic (Quasi-)Newton have been studied.

Note : The goal is to use $\tilde{K}$ instead of K^+ such that the update step (5.16) becomes efficiently operated. One should notice that (5.16) cannot be set in a proximal closed form for a matrix $\tilde{K}$ that is not $\propto I$. Therefore I will call this kind of update steps ” $\tilde{K}$ proximal updates” or eventually ”*pseudo* proximal updates” to make a clear differentiation.

Approach and its rationale

Let's first state the working assumption for what will follow.

Assumption

Be $t \in \mathbb{N}$. Let's consider the step-size $\eta_t > 0$ and the iterate $w_t \in \mathbb{R}^d$. Let us assume that we dispose of a matrix $\mathbf{0} \preceq K^+ = (K^+)^T$ ($L = \|K^+\|_2$) such that $\forall w \in \mathbb{R}^d$,

$$f(w) = \phi(w) + \lambda r(w) \leq \phi(w_t) + \langle \nabla \phi(w_t), w - w_t \rangle + \frac{1}{2\eta_t} \|w - w_t\|_{K^+}^2 + \lambda r(w)$$

Trivially it holds that for any matrix $K^+ \preceq \tilde{K} = (\tilde{K})^T$, one can write $\forall w \in \mathbb{R}^d$

$$f(w) = \phi(w) + \lambda r(w) \leq \phi(w_t) + \langle \nabla \phi(w_t), w - w_t \rangle + \frac{1}{2\eta_t} \|w - w_t\|_{\tilde{K}}^2 + \lambda r(w)$$

Elastic-Net regularization

The practical case I have considered to develop the $\tilde{K}$ -ProxVR approach is an optimization problem under a regularizer of the form

$$r(w) = \alpha \|w\|_1 + \frac{1-\alpha}{2} \|w\|_2^2 \quad \forall w \in \mathbb{R}^d$$

In order to achieve (5.16) the optimality condition devised in Chapter 2, (2.1), states that in w_{t+1} it must hold :

$$-\nabla_w \left(\phi(w_t) + \langle g_t, w - w_t \rangle + \frac{1}{2\eta_t} \|w - w_t\|_{\tilde{K}}^2 \right) \Big|_{w_{t+1}} \in \lambda \partial_w \left(\alpha \|w\|_1 + \frac{1-\alpha}{2} \|w\|_2^2 \right) \Big|_{w_{t+1}}$$

Before going through this requirement let's just introduce the matrix $K_{\text{eff}} = \frac{\tilde{K}}{L}$.

This substitution is done for the sake of clarity when degenerated cases will be provided.

In the same way let $\frac{1}{\eta_{t,\text{eff}}}$ be equal to $\frac{L}{\eta_t}$. A neutral K_{eff} is then simply the identity matrix and $\frac{1}{\eta_{t,\text{eff}}} \geq L$ can be naturally chosen. Let's deal back with the optimality condition :

$$-\nabla_w \left(\phi(w_t) + \langle g_t, w - w_t \rangle + \frac{1}{2\eta_{t,\text{eff}}} \|w - w_t\|_{K_{\text{eff}}}^2 \right) \Big|_{w_{t+1}} \in \lambda \partial_w \left(\alpha \|w\|_1 + \frac{1-\alpha}{2} \|w\|_2^2 \right) \Big|_{w_{t+1}}$$

Hence,

$$\mathbf{0} \in g_t + \frac{1}{\eta_{t,\text{eff}}} K_{\text{eff}} (w_{t+1} - w_t) + (1-\alpha) \lambda w_{t+1} + \alpha \lambda \text{sgn}(w_{t+1})$$

The fact that K_{eff} would be diagonal allows the problem to be separable !

As we will see right after the $\tilde{K}$ proximal update step needs the computation of $K_{\text{eff}}^{-1} g_t$.

Since $\tilde{K}$ (and therefore K_{eff}) is diagonal this operation will only cost $\mathcal{O}(d)$ by update instead of $\mathcal{O}(d^2)$ for a dense matrix! That being said even though d might be small, there exists no closed form for (5.16) when $\tilde{K}$ is dense. Searching for K_{eff} offers a double benefit!

In each dimension $j \in [d]$, one must have

$$\left(\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}}w_{t,j} - g_{t,j}\right) - \left(\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}} + (1-\alpha)\lambda\right)w_{t+1,j} \in \alpha\lambda \text{sgn}(w_{t+1,j})$$

It finally follows⁹

$$w_{t+1,j} = \text{sgn}\left(\frac{\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}}w_{t,j} - g_j}{\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}} + (1-\alpha)\lambda}\right) \left[\left|\frac{\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}}w_j - g_j}{\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}} + (1-\alpha)\lambda}\right| - \frac{\alpha\lambda}{\frac{K_{\text{eff},jj}}{\eta_{t,\text{eff}}} + (1-\alpha)\lambda}\right]_+ \quad (5.17)$$

we recognize then the standard proximal elastic net operator for the *neutral case* where $K_{\text{eff},jj} = 1$ by noticing that the last equality would then correspond to

$$w_{t+1,j} = \text{sgn}\left(\frac{w_{t,j} - \eta_{t,\text{eff}}g_j}{1 + (1-\alpha)\eta_{t,\text{eff}}\lambda}\right) \left[\left|\frac{w_j - \eta_{t,\text{eff}}g_j}{1 + (1-\alpha)\eta_{t,\text{eff}}\lambda}\right| - \frac{\alpha\eta_{t,\text{eff}}\lambda}{1 + (1-\alpha)\eta_{t,\text{eff}}\lambda}\right]_+$$

Pre-Processing - SDP solving

Now comes the real novelty. Given this *per coordinate* paralleling one can conceive that there is somehow a *per-coordinate* step-size $\eta_{t,j}$. This is exactly true when one considers the (non-smooth) ℓ_1 regularization; e.g. $\alpha = 1$. Indeed the *per coordinate* update resumes then in:

$$w_{t+1,j} = \text{sgn}\left(w_{t,j} - K_{\text{eff},jj}^{-1}\eta_{t,\text{eff}}g_j\right) \left[\left|w_{t,j} - K_{\text{eff},jj}^{-1}\eta_{t,\text{eff}}g_j\right| - K_{\text{eff}}^{-1}\eta_{t,\text{eff}}\lambda\right]_+$$

Here it becomes clear that $\eta_{t,j} = K_{\text{eff},jj}^{-1}\eta_{t,\text{eff}}$. Heuristically, one would expect better empirical results when choosing K_{eff} such that the mean *per coordinate* step-size is maximized !

This is exactly what $\tilde{K}\text{Prox-VR}$ aims to achieve by solving a SDP in a pre-processing fashion !

$$\begin{aligned} \min_{K_{\text{eff}} \in \mathbb{D}_d^+} \quad & \frac{1}{d} \sum_{j=1}^d K_{\text{eff},jj} \\ \text{s.t.} \quad & K_{\text{eff}} \succeq \frac{K^+}{L} \\ & K_{\text{eff}} \preceq \frac{L_B}{L} I \end{aligned}$$

where $L_B \geq L$ depicts an upper-bound on L . The closest L_B is from L , the less leverage the above problem has in order to produce an eventual gain ; that is having $\frac{1}{d} \sum_{j=1}^n \eta_{t,j} \geq \eta_{t,\text{eff}}$. In practice I will use $L_B = \bar{L} = \frac{1}{n} \sum_{i=1}^n L_i$ (assuming that the ϕ_i are L_i smooth). $\mathbb{D}_d^+$ represents the set all positive definite diagonal matrices of order d .

The problem above is not in a standard form in what concerns SDP's. Nevertheless CVXOPT uses *DSQP*'s algorithm (*Dual-Scaling Semidefinite Programming*) and can solve problems mixing a linear objective and matrix constraints. This is precisely what's needed in this case since K_{eff} is diagonal. In order to build K_{eff} efficiently one must invoke :

$$K_{\text{eff}} := \text{diag}\left(\arg \min_{h \in \mathbb{R}_+^d} \left\{ e_{d,1}^T x \mid \sum_{j=1}^d h_j \Delta_{jj} \preceq \frac{L_B}{L} I, \sum_{j=1}^d h_j \Delta_{jj} \succeq \frac{K^+}{L} \right\}\right) \quad (5.18)$$

Where $e_{d,1}$ is the column vector of size d embedding only 1's and $\forall j \in \{1, \dots, d\}$, Δ_{jj} is the square matrix of order d filled-in only with 0's excepted at entry (j, j) for which there's a 1.

⁹Direct adaptation of `stats.stackexchange/elastic-net-regularization`.

For what follows, by default, the choice for such K^+ is the problem dependent matrix K . As a reminder, a valid K was $\frac{c}{n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T$ with $c = 2$ for smoothed-hinge and quadratic losses and $c = \frac{1}{4}$ for the logistic loss.

Extreme cases

As argued before the degenerated (so called "neutral") case arises whenever K_{eff} becomes simply the identity matrix. Another question that could pass through reader's mind would be in which situation one could expect a large difference for the performances between traditional proximal updates and $\tilde{K}$ proximal updates. In other words this question resumes in :

"In which situations $\tilde{K}$ and LI might be really different ?"

This could be left as an open problem ; it is indeed not so trivial to give a general answer. However there exists a straightforward setting in which there's a huge potential gain using $\tilde{K}$ Prox-VR and this will be confirmed by practical experiments on homemade datasets.

Let's imagine that K^+ for a reason or another is luckily quite similar to a diagonal matrix! In an extreme case let $K^+ = \mathbf{diag}(h_g)$ for $h_g \succ \mathbf{0}$ a vector in $\mathbb{R}^d$ that generates K^+ .

Trivially L boils down to $\max_{j \in [d]} h_{g,j}$. Choosing $K_{\text{eff}} = \frac{K^+}{L}$ is obviously the optimal solution to the SDP problem that occupies us and would lead in this theoretical case to $\mathbf{diag}(\frac{h_g}{L}) \preceq I$.

Potentially one entry $j' \in [d]$ of $\frac{h_g}{L}$ is much smaller than 1 if $0 < h_{g,j} \ll L$.

The resulting *per-coordinate* step-size, $\eta_{t,j'} = K_{\text{eff},j'j'}^{-1} \eta_{t,\text{eff}} \stackrel{\text{here}}{=} \frac{L}{h_{g,j'}} \eta_{t,\text{eff}}$, rises substantially!

One shouldn't be too enthusiastic though ! Let's consider a practical example for such $K^+ = K$. For a sufficiently large number n of observations/data points, K in its typical structure is not likely at all to be equivalent to a sparse diagonal matrix. Indeed K is a sum of n dense rank-1 matrices :

$$K^+ = K \propto \frac{1}{n} \sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T = \frac{1}{n} \sum_{i=1}^n \begin{bmatrix} \tilde{x}_{i,1}^2 & \cdots & \tilde{x}_{i,1} \tilde{x}_{i,d} \\ \vdots & \ddots & \vdots \\ \tilde{x}_{i,d} \tilde{x}_{i,1} & \cdots & \tilde{x}_{i,d}^2 \end{bmatrix}$$

Even though K^+ won't be exactly equal to a diagonal matrix, it might happen that K^+ is so *diagonally dominant* that it looks like it. A concrete example of such a phenomenon is encountered when a feature embedded in the $\tilde{x}_i$'s presents values that are much bigger than the average span of values for the other features. E.g. if the scale of the first component is in *tens* whereas the scale of the other features are in *tenths* then for each $i \in [n]$ the sample matrix $\tilde{x}_i \tilde{x}_i^T$ will have an entry $\tilde{x}_{i,1}^2$ in *hundreds* or even *thousands*, a first row and a first column of entries in the *unitary scale* and the rest of matrix's entries will be in *hundredths*. For moderate d (assumption of this section since building K^+ takes $\mathcal{O}(nd^2)$ operations) K^+ would be *diagonally dominant* by far. In this setting $\|\mathbf{diag}([\tilde{x}_i^2, \mathbf{0}_{1,d-1}]) - \tilde{x}_i \tilde{x}_i^T\|_F^2$ might be relatively small...

In practice such extreme cases arise when users are not cautious enough to rescale their features or, eventually, by adding an intercept (a 1 is placed in position 0 for every $\tilde{x}_i$) term whenever the other features are valued in *hundredths* or even smaller.

Theoretical analysis

I managed to adapt an existing proof for Prox-SVRG (**Theorem 1** in [XZ14]) to show the linear convergence of what will be called $\tilde{K}$ Prox-SVRG under strong-convexity and smoothness with respect to $\psi(\cdot) = \frac{1}{2} \|K_{\text{eff}}^{\frac{1}{2}} \cdot\|^2$. As an extension I have also shown the linear convergence for a full gradient version of $\tilde{K}$ Prox-SVRG, namely $\tilde{K}$ Prox-GD, using smoothness with respect to this same ψ and, this time, a slightly modified proximal PL condition as introduced at the end of Chapter 2. This latter is not stochastic anymore, it is simply an improved version of the classical Prox-GD under a slightly better knowledge of ϕ . This last proof is in **Appendix 3**.)

Algorithms

Let's first describe $\tilde{K}$ Prox-SVRG and $\tilde{K}$ Prox-GD by their respective algorithm.

Algorithm 12 $\tilde{K}$ Prox-SVRG

Initialization : $\eta_{\text{eff}} > 0 \forall t \in \mathbb{N}$, w_0 starting iterate, fixed deterministic batch-size $\hat{b}$, $t = 0$ update steps counter, initial $\bar{\alpha}_0$ (for every $i \in [n]$, $\alpha_{0,i} = \nabla \phi_i(w_0)$: *warm start* or $\mathbf{0}$), m number of inner-loop iterations, $s = 0$ outer-loop counter, K_{eff} as devised in (5.18)
while stopping criterion not verified **do**
 $\tilde{w}_s \leftarrow$ last iterate (version 1) / average iterate over the last inner-loop (version 2)
 $\bar{\alpha} \leftarrow \nabla \phi(\tilde{w}_s)$
 for $(t+1) \bmod (m+1) \neq 0$ **do**
 $S_t \sim \mathbf{U}([n]^{\hat{b}})$
 $g_t \leftarrow \bar{\alpha} + \frac{1}{\hat{b}} \sum_{i \in S_t} \nabla \phi_i(w_t) - \nabla \phi_i(\tilde{w}_s)$ // SVRG stochastic gradient
 $w_{t+1} \leftarrow \arg \min_{w \in \mathbb{R}^d} \{ \langle g_t, w - w_t \rangle + \frac{1}{2\eta_{\text{eff}}} \|w - w_t\|_{K_{\text{eff}}}^2 + R(w) - R(w_t) \}$
 $t \leftarrow t + 1$
 end for
 $s \leftarrow s + 1$
end while
Output : w_T last outer loop iterate

Algorithm 13 $\tilde{K}$ Prox-GD

Initialization : $\eta = \frac{1}{\hat{L}}$, $t = 0$ update steps counter, K_{eff} as devised in (5.18)
while stopping criterion not verified **do**
 $g_t \leftarrow \nabla \phi(w_t)$ // full gradient
 $w_{t+1} \leftarrow \arg \min_{w \in \mathbb{R}^d} \{ \langle g_t, w - w_t \rangle + \frac{\hat{L}}{2} \|w - w_t\|_{K_{\text{eff}}}^2 + R(w) - R(w_t) \}$
 $t \leftarrow t + 1$
end while
Output : w_T last iterate

Note : $\tilde{K}$ Prox-GD is a degenerated case of $\tilde{K}$ Prox-SVRG for which $m = 0$ (no inner-loop).

In the following pages the proof of convergence for $\tilde{K}$ Prox-SVRG is built step by step. It will be shown for $\hat{b} = 1$ (size 1 mini-batch) and under a general sampling procedure; $i_t \sim \mathcal{P}$ where $\mathcal{P}$ is a non-adaptive univariate probability distribution for the samples (not especially uniform as above).

Before going through the core of the proof, one needs to develop some theoretical aspects that will be helpful to keep it clear.

I have found out that it is possible to define $\tilde{K}$ proximal updates in a form that is closely related to standard proximal updates. Thus, let me define the $\tilde{K}\text{prox}_R(\cdot)$ operator.

Definition 11 ($\tilde{K}$ proximal mapping of a function R) : Be $L > 0$ a real number and $\tilde{K} \succ \mathbf{0}$ a symmetric matrix of $\mathbb{R}^{d \times d}$. Let K_{eff} be equal to $\frac{\tilde{K}}{L}$. The $\tilde{K}$ proximal mapping of a proper closed convex function $R : \mathbb{R}^d \rightarrow \bar{\mathbb{R}}$ at $u \in \mathbb{R}^d$ is defined as the unique

$$\tilde{K}\text{prox}_R(u) := \arg \min_{y \in \mathbb{R}^d} \left\{ R(y) + \frac{1}{2} \|y - u\|_{K_{\text{eff}}}^2 \right\} \quad (5.19)$$

Note: Similarly to Chapter 2, one can show the following equality :

$$\tilde{K}\text{prox}_{\eta_{t,\text{eff}}R}(w_t - \eta_{t,\text{eff}}K_{\text{eff}}^{-1}g_t) = \arg \min_{w \in \mathbb{R}^d} \left\{ \langle g_t, w - w_t \rangle + \frac{1}{2\eta_{t,\text{eff}}} \|w - w_t\|_{K_{\text{eff}}}^2 + R(w) - R(w_t) \right\} \quad (5.20)$$

Taking $\eta_{t,\text{eff}} = \eta$ and $K_{\text{eff}} = I$ ($\tilde{K} = LI$) yields Chapter's 2 $\text{prox}_{\eta_t R}(w_t - \eta_t g_t)$ as a special case.

The property below is a substitute to **Property 2** (Chapter 2) about the non-expansivity of the proximal operator. Let's give a revised version for the $\tilde{K}$ proximal mapping.

Property 4 (bounded expansivity of $\tilde{K}$ proximal operators) : Let $R : \mathbb{R}^d \rightarrow \bar{\mathbb{R}}$ be a proper closed convex function and $K_{\text{eff}} \succ \mathbf{0}$ defined as usual then the following inequality holds $\forall u, v \in \mathbb{R}^d$

$$\|\tilde{K}\text{prox}_R(u) - \tilde{K}\text{prox}_R(v)\| \leq \frac{\|K_{\text{eff}}(u - v)\|}{\|K_{\text{eff}}\|_{-2}}$$

Proof : Let $u' := \tilde{K}\text{prox}_R(u)$ and $v' := \tilde{K}\text{prox}_R(v)$. Looking at (5.20), the $\tilde{K}$ proximal step requires that we have $K_{\text{eff}}(u - u') \in \partial R(u')$ and $K_{\text{eff}}(v - v') \in \partial R(v')$. Then, by definition of subdifferential set :

$$\begin{aligned} R(v') &\geq R(u') + \langle K_{\text{eff}}(u - u'), v' - u' \rangle \\ R(u') &\geq R(v') + \langle K_{\text{eff}}(v - v'), u' - v' \rangle \end{aligned}$$

Summing both inequalities leads to :

$$0 \geq \langle K_{\text{eff}}(v - v' - u + u'), u' - v' \rangle$$

$$\Leftrightarrow$$

$$\langle K_{\text{eff}}(u' - v'), u' - v' \rangle \leq \langle K_{\text{eff}}(u' - v'), u - v \rangle$$

Finally, using *Cauchy-Schwarz* inequality and the last inequality it comes that

$$\|K_{\text{eff}}\|_{-2} \|u' - v'\|^2 \leq \|K_{\text{eff}}(u' - v')\| \|u' - v'\| \leq \|u' - v'\|_{K_{\text{eff}}}^2 \leq \|K_{\text{eff}}(u - v)\| \|u' - v'\|$$

where $\|K_{\text{eff}}\|_{-2}$ represents the lowest singular value of K_{eff} .¹⁰ We can now conclude by stating

$$\|\tilde{K}\text{prox}_R(u) - \tilde{K}\text{prox}_R(v)\| \leq \frac{\|K_{\text{eff}}(u - v)\|}{\|K_{\text{eff}}\|_{-2}}$$

□

Now I mark down **Corollary 3** of the reference paper [XZ14] cited above.

It aims to give a bound to the variance term V_t .

¹⁰In `matrixLowerBound`, one can find $\|A^{-1}\|^{-1}\|x\| \leq \|Ax\|$: for 2-norms, it boils down to $\|A^{-1}\|^{-1} = \|A\|_{-2} = \sigma_{\min}(A)$ as argued.

Lemma 9 (bound on $V_t = \mathbb{E}_{i_t}[\|g_t - \nabla\phi(w_t)\|^2]$) : Be $\mathcal{P}$ a proper probability distribution over $[n]$. Be $\tilde{w}_s \in \mathbb{R}^d$ a snapshot point and $w_t \in \mathbb{R}^d$ the current iterate. Let $i_t \sim \mathcal{P}$.

Then for a stochastic gradient $g_t = \frac{\nabla\phi_{i_t}(w_t) - \nabla\phi_{i_t}(\tilde{w}_s)}{np_{i_t}} + \nabla\phi(\tilde{w}_s)$, the following inequality holds

$$\mathbb{E}_{i_t}[\|g_t - \nabla\phi(w_t)\|^2] \leq 4L_{\mathcal{P}}[f(w_t) - f(w_*) + f(\tilde{w}_s) - f(w_*)] \quad (5.21)$$

where $L_{\mathcal{P}} := \max_{i \in [n]} \frac{L_i}{np_i}$ and w_* is a minimizer of $f = \phi + R$ in $\mathcal{S}^*$.

Reminder : The L_i are the gradient Lipschitz constants of the sample loss functions ϕ_i .

Let's adapt **Lemma 3** coming from that same reference. As stated by the authors it aims to give a lower bound on f depending on the gradient mapping. The difference with Prox-SVRG is that my bound will embed K_{eff} norms ($\|\cdot\|_{K_{\text{eff}}}$) instead of the euclidean norm $\|\cdot\|$. I would like to insist on the fact that the steps pursued in this proof are totally inspired from Prox-SVRG, my real contribution here is then the generalization with K_{eff} .

Lemma 10 (lower bound on f embedding a gradient mapping when using $\tilde{K}$ proximal updates): Be K_{eff} defined as in (5.18). Be $f = \phi + R$ where R is convex and ϕ is both $\hat{L}$ smooth and $\hat{\mu}$ strongly-convex with respect to $\psi(\cdot) = \frac{1}{2}\|\cdot\|_{K_{\text{eff}}}^2$. Let's fix $0 < \eta_{\text{eff}} \leq \frac{1}{\hat{L}}$ and let's define the following quantities for any $w \in \mathbb{R}^d$ and any $g \in \mathbb{R}^d$:

$$w_+ = \arg \min_{y \in \mathbb{R}^d} \left\{ \langle g, y - w \rangle + \frac{1}{2\eta_{\text{eff}}} \|y - w\|_{K_{\text{eff}}}^2 + R(y) - R(w) \right\}$$

$$G = \frac{1}{\eta_{\text{eff}}}(w - w_+)$$

$$\Delta = g - \nabla\phi(w)$$

Then $\forall u \in \mathbb{R}^d$, the following inequality holds :

$$f(u) \geq f(w_+) + \langle K_{\text{eff}} G, u - w \rangle + \frac{\eta_{\text{eff}}}{2} \|G\|_{K_{\text{eff}}}^2 + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2 + \langle \Delta, w_+ - u \rangle \quad (5.22)$$

Proof : To start this proof let's denote by $\xi \in \partial R(w_+)$ a vector such that

$$g + \frac{K_{\text{eff}}}{\eta_{\text{eff}}}(w_+ - w) + \xi = \mathbf{0}$$

in order for w_+ to fulfill optimality requirements. Based on G defined above, we can write

$$\xi = K_{\text{eff}} G - g$$

Noticing that ξ is a subgradient of R at w_+ and using the $\hat{\mu}$ strong-convexity of ϕ w.r.t to ψ , $\forall u \in \mathbb{R}^d$,

$$\begin{aligned} f(u) &= \phi(u) + R(u) \\ &\geq \phi(w) + \langle \nabla\phi(w), u - w \rangle + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2 + R(w_+) + \langle \xi, u - w_+ \rangle \end{aligned}$$

We can bound $\phi(w)$ as well. This time let's use the $\hat{L}$ smoothness of ϕ w.r.t ψ :

$$\phi(w) \geq \phi(w_+) - \langle \nabla\phi(w), w_+ - w \rangle - \frac{\hat{L}}{2} \|w_+ - w\|_{K_{\text{eff}}}^2$$

Let's combine the two last inequalities :

$$f(u) \geq \phi(w_+) - \langle \nabla\phi(w), w_+ - w \rangle - \frac{\hat{L}}{2} \|w_+ - w\|_{K_{\text{eff}}}^2 + \langle \nabla\phi(w), u - w \rangle + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2 + R(w_+) + \langle \xi, u - w_+ \rangle$$

$$f(u) \geq f(w_+) + \langle \nabla \phi(w), u - w_+ \rangle + \langle K_{\text{eff}} G - g, u - w_+ \rangle - \frac{\hat{L}}{2} \eta_{\text{eff}}^2 \|G\|_{K_{\text{eff}}}^2 + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2$$

Since,

$$\begin{aligned} \langle \nabla \phi(w), u - w_+ \rangle + \langle K_{\text{eff}} G - g, u - w_+ \rangle &= \langle K_{\text{eff}} G, u - w_+ \rangle + \langle \nabla \phi(w) - g, u - w_+ \rangle \\ &= \langle K_{\text{eff}} G, (u - w) + (w - w_+) \rangle + \langle \Delta, w_+ - u \rangle \\ &= \eta_{\text{eff}} \|G\|_{K_{\text{eff}}}^2 + \langle K_{\text{eff}} G, u - w \rangle + \langle \Delta, w_+ - u \rangle \end{aligned}$$

Injecting this last equality in the main inequality and using the fact that $\hat{L}\eta_{\text{eff}} \geq 1$ yields :

$$\begin{aligned} f(u) &\geq f(w_+) + \langle K_{\text{eff}} G, u - w \rangle + \frac{\eta_{\text{eff}}}{2} (2 - \hat{L}\eta_{\text{eff}}) \|G\|_{K_{\text{eff}}}^2 + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2 + \langle \Delta, w_+ - u \rangle \\ f(u) &\geq f(w_+) + \langle K_{\text{eff}} G, u - w \rangle + \frac{\eta_{\text{eff}}}{2} \|G\|_{K_{\text{eff}}}^2 + \frac{\hat{\mu}}{2} \|u - w\|_{K_{\text{eff}}}^2 + \langle \Delta, w_+ - u \rangle \end{aligned}$$

□

Now we dispose of the necessary material in order to give proof of convergence of $\tilde{K}$ Prox-SVRG.

Theorem 4 (linear convergence in expectation for $\tilde{K}$ Prox-SVRG under strong-convexity and smoothness assumptions) Let's assume that K_{eff} , ϕ , R fulfill previous lemma's assumptions. Be $0 < \eta_{\text{eff}} < \frac{\|K_{\text{eff}}\|_{-2}}{4L_{\mathcal{P}}} \leq \frac{1}{\hat{L}}$ and $w_* \in \mathcal{S}^*$ a minimizer of f on $\mathbb{R}^d$. Choosing $m \in \mathbb{N}$ (number of inner-loop iterations) sufficiently large such that

$$\rho = \left(\frac{\|K_{\text{eff}}\|_{-2}}{\hat{\mu} m \eta_{\text{eff}} (\|K_{\text{eff}}\|_{-2} - 4L_{\mathcal{P}} \eta_{\text{eff}})} + \frac{4L_{\mathcal{P}} \eta_{\text{eff}} (m+1)}{m (\|K_{\text{eff}}\|_{-2} - 4L_{\mathcal{P}} \eta_{\text{eff}})} \right) < 1$$

ensures that after $s \in \mathbb{N}$ outer-loop iterations of $\tilde{K}$ Prox-SVRG using a constant step-size η_{eff} ,

$$\mathbb{E}[f(\tilde{w}_s) - f(w_*)] \leq \rho^s \mathbb{E}[f(\tilde{w}_0) - f(w_*)]$$

Proof: Let's fix an iterate $w_t \in \mathbb{R}^d$ for a given $t \in \mathbb{N}$. Applying last lemma using $u = w_*$, $w_+ = w_{t+1}$, $w = w_t$ immediately gives $(\Delta_t := g_t - \nabla \phi(w_t), G_t := \frac{w_t - w_{t+1}}{\eta_{\text{eff}}})$:

$$-\langle K_{\text{eff}} G_t, w_t - w_* \rangle + \frac{\eta_{\text{eff}}}{2} \|G_t\|_{K_{\text{eff}}}^2 \leq f(w_*) - f(w_{t+1}) - \frac{\hat{\mu}}{2} \|w_t - w_*\|_{K_{\text{eff}}}^2 - \langle \Delta_t, w_{t+1} - w_* \rangle$$

Let's analyze K_{eff} 's distance of w_{t+1} produced by $\tilde{K}$ Prox-SVRG to the minimizer w_* .

$$\begin{aligned} \|w_{t+1} - w_*\|_{K_{\text{eff}}}^2 &= \|w_t - \eta_{\text{eff}} G_t - w_*\|_{K_{\text{eff}}}^2 \\ &= \|w_t - w_*\|_{K_{\text{eff}}}^2 - 2\eta_{\text{eff}} \langle K_{\text{eff}} G_t, w_t - w_* \rangle + \eta_{\text{eff}}^2 \|G_t\|_{K_{\text{eff}}}^2 \end{aligned}$$

Thus,

$$\begin{aligned} \|w_{t+1} - w_*\|_{K_{\text{eff}}}^2 &\leq \|w_t - w_*\|_{K_{\text{eff}}}^2 + 2\eta_{\text{eff}} \left[f(w_*) - f(w_{t+1}) - \frac{\hat{\mu}}{2} \|w_t - w_*\|_{K_{\text{eff}}}^2 - \langle \Delta_t, w_{t+1} - w_* \rangle \right] \\ &\leq \|w_t - w_*\|_{K_{\text{eff}}}^2 - 2\eta_{\text{eff}} \left[f(w_{t+1}) - f(w_*) \right] - 2\eta_{\text{eff}} \langle \Delta_t, w_{t+1} - w_* \rangle \end{aligned}$$

Now the goal is to upper-bound the term $-2\eta_{\text{eff}} \langle \Delta_t, w_{t+1} - w_* \rangle$. To that purpose we will make use of the **Property 4** (bounded expansivity of $\tilde{K}$ proximal operators).

Let's first depict by $\hat{w}_{t+1}$ the output of $\tilde{K}\text{prox}_{\eta_{\text{eff}}R}(w_t - \eta_{\text{eff}}K_{\text{eff}}^{-1}\nabla\phi(w_t))$ ($\tilde{K}$ proximal update using a plain gradient). **The following lines are important and discussed below.**

$$\begin{aligned}
-2\eta_{\text{eff}}\langle\Delta_t, w_{t+1} - w_*\rangle &= -2\eta_{\text{eff}}\langle\Delta_t, w_{t+1} - \hat{w}_{t+1}\rangle - 2\eta_{\text{eff}}\langle\Delta_t, \hat{w}_{t+1} - w_*\rangle \\
&\leq 2\eta_{\text{eff}}\left(\|\Delta_t\| \|\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1}\| - \langle\Delta_t, \hat{w}_{t+1} - w_*\rangle\right) \\
&\leq 2\eta_{\text{eff}}\frac{\|K_{\text{eff}}((w_t - \eta_{\text{eff}}K_{\text{eff}}^{-1}g_t) - (w_t - \eta_{\text{eff}}K_{\text{eff}}^{-1}\nabla\phi(w_t)))\|}{\|K_{\text{eff}}\|_{-2}}\|\Delta_t\| - 2\eta_{\text{eff}}\langle\Delta_t, \hat{w}_{t+1} - w_*\rangle \\
&\leq 2\eta_{\text{eff}}^2\frac{\|\Delta_t\|^2}{\|K_{\text{eff}}\|_{-2}} - 2\eta_{\text{eff}}\langle\Delta_t, \hat{w}_{t+1} - w_*\rangle
\end{aligned}$$

Let ζ be equal to $\|K_{\text{eff}}\|_{-2}$ and let's take the one-step expectation with respect to $i_t \sim \mathcal{P}$ on both sides of the main inequality :

$$\mathbb{E}_{i_t}\left[\|w_{t+1} - w_*\|_{K_{\text{eff}}}^2\right] \leq \mathbb{E}_{i_t}\left[\|w_t - w_*\|_{K_{\text{eff}}}^2 - 2\eta_{\text{eff}}[f(w_{t+1}) - f(w_*)] + 2\eta_{\text{eff}}^2\frac{\|\Delta_t\|^2}{\zeta} - 2\eta_{\text{eff}}\langle\Delta_t, \hat{w}_{t+1} - w_*\rangle\right]$$

Both $\hat{w}_{t+1}$ and w_* are independent of i_t , $\mathbb{E}_{i_t}[\langle\Delta_t, \hat{w}_{t+1} - w_*\rangle] = \langle\mathbb{E}_{i_t}[g_t - \nabla\phi(w_t)], \hat{w}_{t+1} - w_*\rangle = 0$ as g_t (SVRG stochastic gradient) is a one-step unbiased estimate of $\nabla\phi(w_t)$.

The reader might notice the appearance of the term $2\frac{\eta_{\text{eff}}^2}{\zeta}\mathbb{E}_{i_t}[\|\Delta_t\|^2]$. Thanks to **Corollary** 3 of [XZ14], this quantity is bounded by $\frac{8\eta_{\text{eff}}^2L\mathcal{P}}{\zeta}[f(w_t) - f(w_*) + f(\tilde{w}_s) - f(w_*)]$.

Note that s is the outer-loop iteration number, meaning that g_t was computed based on $\tilde{w}_s$.

The main inequality rewritten leads to :

$$\mathbb{E}_{i_t}\left[\|w_{t+1} - w_*\|_{K_{\text{eff}}}^2\right] \leq \|w_t - w_*\|_{K_{\text{eff}}}^2 - 2\eta_{\text{eff}}\mathbb{E}_{i_t}[f(w_{t+1}) - f(w_*)] + \frac{8\eta_{\text{eff}}^2L\mathcal{P}}{\zeta}[f(w_t) - f(w_*) + f(\tilde{w}_s) - f(w_*)]$$

Now let's define the next snapshot at the end of the current s -th inner-loop, $\tilde{w}_{s+1} = \frac{1}{m}\sum_{t'=1}^m w_{sm+t'}$. In other words, the next snapshot is the average of the m last produced iterates within the s -th inner loop and $\tilde{w}_s = w_{sm}$. Since we want to express the decrease rate over one outer-loop iteration (= after a whole inner-loop) let's use the following notations : $\tilde{w}_s = w'_0$ and $\tilde{w}_{s+1} = \frac{1}{m}\sum_{t'=1}^m w'_{t'}$. Taking the expectation over all the randomness $\{i'_0, \dots, i'_{m-1}\}$ during the considered inner-loop and summing the inequalities for $t' = 0, \dots, m-1$, we can write :

$$\begin{aligned}
&\mathbb{E}[\|w'_m - w_*\|_{K_{\text{eff}}}^2] + 2\eta_{\text{eff}}\mathbb{E}[f(w'_m) - f(w_*)] + 2\eta_{\text{eff}}\left(1 - \frac{4L\mathcal{P}\eta_{\text{eff}}}{\zeta}\right)\sum_{t'=0}^{m-2}\mathbb{E}[f(w'_{t'+1}) - f(w_*)] \\
&\leq \\
&\|w'_0 - w_*\|_{K_{\text{eff}}}^2 + m\frac{8L\mathcal{P}\eta_{\text{eff}}^2}{\zeta}\mathbb{E}[f(\tilde{w}_s) - f(w_*)] + \frac{8L\mathcal{P}\eta_{\text{eff}}^2}{\zeta}\mathbb{E}[f(w'_0) - f(w_*)]
\end{aligned}$$

As $w'_0 = \tilde{w}_s$ and given that $2\eta_{\text{eff}} \leq 2\eta_{\text{eff}}(1 - \frac{4L\mathcal{P}\eta_{\text{eff}}}{\zeta})$ we can write :

$$2\eta_{\text{eff}}\left(1 - \frac{4L\mathcal{P}\eta_{\text{eff}}}{\zeta}\right)\sum_{t'=0}^{m-1}\mathbb{E}[f(w'_{t'+1}) - f(w_*)] \leq \|\tilde{w}_s - w_*\|_{K_{\text{eff}}}^2 + \frac{8L\mathcal{P}\eta_{\text{eff}}^2}{\zeta}(m+1)\mathbb{E}[f(\tilde{w}_s) - f(w_*)]$$

As ϕ is convex, *Jensen's* inequality implies $m\mathbb{E}[f(\tilde{w}_{s+1}) - f(w_*)] \leq \sum_{t'=0}^{m-1}\mathbb{E}[f(w'_{t'+1}) - f(w_*)]$. Furthermore $\forall v \in \partial f(w_*)$ and $\forall \tilde{w}_s \in \mathbb{R}^d$, $\langle v, \tilde{w}_s - w_* \rangle \geq 0$, we can write

$$f(\tilde{w}_s) \geq f(w_*) + \langle v, \tilde{w}_s - w_* \rangle + \frac{\hat{\mu}}{2}\|\tilde{w}_s - w_*\|_{K_{\text{eff}}}^2 \Rightarrow \frac{2}{\hat{\mu}}[f(\tilde{w}_s) - f(w_*)] \geq \|\tilde{w}_s - w_*\|_{K_{\text{eff}}}^2$$

Thus we can state :

$$\mathbb{E}[f(\tilde{w}_{s+1}) - f(w_*)] \leq (2m\eta_{\text{eff}}(1 - \frac{4L\mathcal{P}\eta_{\text{eff}}}{\zeta}))^{-1}(\frac{2}{\hat{\mu}} + \frac{8L\mathcal{P}\eta_{\text{eff}}^2(m+1)}{\zeta})\mathbb{E}[f(\tilde{w}_s) - f(w_*)]$$

We can conclude by rearranging the terms and apply a recursion :

$$\mathbb{E}[f(\tilde{w}_s) - f(w_*)] \leq \rho^s \mathbb{E}[f(\tilde{w}_0) - f(w_*)]$$

$$\text{where } \rho = \left(\frac{\zeta}{\hat{\mu}m\eta_{\text{eff}}(\zeta - 4L\mathcal{P}\eta_{\text{eff}})} + \frac{4L\mathcal{P}\eta_{\text{eff}}(m+1)}{m(\zeta - 4L\mathcal{P}\eta_{\text{eff}})} \right)$$

□

Note 1 : We fall back to Prox-SVRG bound if $K_{\text{eff}} = I$ as $\|K_{\text{eff}}\|_{-2} = 1$ then.

Note 2 : The reader might convince himself that the above theorem yields a slower theoretical convergence rate when $\|K_{\text{eff}}\|_{-2} < 1^{11}$, which is on the other hand surprising because when $\|K_{\text{eff}}\|_{-2}$ is small, one of the *per coordinate* step-size must be big ! Actually the reason why the above rate $\rho(K_{\text{eff}})$ is worse resides in the fact that the worst case of the *expansivity* of the $\tilde{K}$ proximal operator is above the worst case of *expansivity* for the classical proximal operator. That is, there always exist $u, v \in \mathbb{R}^d$ such that for the given R, K_{eff} , $\frac{\|K_{\text{eff}}(u-v)\|}{\|K_{\text{eff}}\|_{-2}} \geq \|u-v\|$. These quantities appear in the proof of convergence right below the comment in [green](#).

Nevertheless, K_{eff} (diagonal matrix) is built on purpose to present entries that are, in average, less than 1 ! If we consider a uniform distribution on $\mathbb{R}^d$, we can presume that the quantity $\|\tilde{\mathbf{K}}\text{prox}_{\mathbf{R}}(\mathbf{u}) - \tilde{\mathbf{K}}\text{prox}_{\mathbf{R}}(\mathbf{v})\|$ will be actually smaller than $\|\text{prox}_R(u) - \text{prox}_R(v)\|$.

Let's try to motivate why. By definition, $\tilde{K}\text{prox}_R(u) := \arg \min_{y \in \mathbb{R}^d} \{R(y) + \frac{1}{2}\|y - u\|_{K_{\text{eff}}}^2\}$. Let's suppose that R admits a global minimum and let's imagine that the weight of R in the expression of both $\tilde{K}$ /classical proximal mappings is dominating their respective quadratic term. Since $\frac{\text{tr}(K_{\text{eff}})}{d} \leq \frac{\text{tr}(I)}{d}$, for a uniform distribution of vectors $\nu \in \mathbb{R}^d$ defined on any closed ball centered in $\mathbf{0}$, the expectation of $\|\nu\|_{K_{\text{eff}}}^2$ will be smaller than $\|\nu\|_I^2 = \|\nu\|^2$.

It implies that in an overall expectation under uniform distribution, the term (for $\tilde{K}$ proximal mapping) $\frac{1}{2}\|y - u\|_{K_{\text{eff}}}^2$ will be smaller than $\frac{1}{2}\|y - u\|^2$ (for classical proximal mapping). Hence, we can expect that for any input argument $u \in \mathbb{R}^d$, $\tilde{K}\text{prox}_R(\cdot)$ operator will guide u towards $w_*^R := \arg \min_{w \in \mathbb{R}^d} R(w)$ **more** than $\text{prox}_R(\cdot)$ would ! In an extreme case, K_{eff} is ill-conditioned in almost every direction $j \in [d]$ and for any vectors $u, v \in \mathbb{R}^d$, $\tilde{K}\text{prox}_R(u) \simeq \tilde{K}\text{prox}_R(v) \simeq w_*^R \Rightarrow \|\tilde{K}\text{prox}_R(u) - \tilde{K}\text{prox}_R(v)\| \simeq 0$. This would never happen using the classical proximal operator. Tests will confirm the efficiency of $\tilde{K}\text{Prox-SVRG}$.

Experiments : $\tilde{K}\text{Prox-VR}$ and heuristic improvement

This subsection is aimed to show the empirical efficiency of $\tilde{K}\text{Prox-SVRG}$. I have also tried $\tilde{K}\text{Prox-SARAH}$ (understand $\tilde{K}$ proximal update steps using SARAH's stochastic gradient) but I did not have the time to prove its convergence too.

The objective functions for the tests are, as usual, the logistic loss, the smoothed-hinge loss and the quadratic loss. The scale of the tests below was $n = 5000$ and $d = 20$. Both homemade datasets and a real world dataset (**Abalone**¹²) served as benchmarks.

As discussed in the **Extreme Cases** paragraph, in what concerns Figure 5.42, a single feature of the generated samples $\{\tilde{x}_i\}_{i=1}^n$ has been set, in average, 10 times bigger than the other ones.

¹¹In practice, $\|K_{\text{eff}}\|_{-2}$ is always smaller than 1 ; if it's not the case there are no advantage in using K_{eff} instead of I in terms of mean *per coordinate* step-size.

¹²Source: UCI-ML repository.

logistic with l_1 reg. — (l) *Abalone Data Set*
quadratic with l_1 reg. — (r) *Homemade*

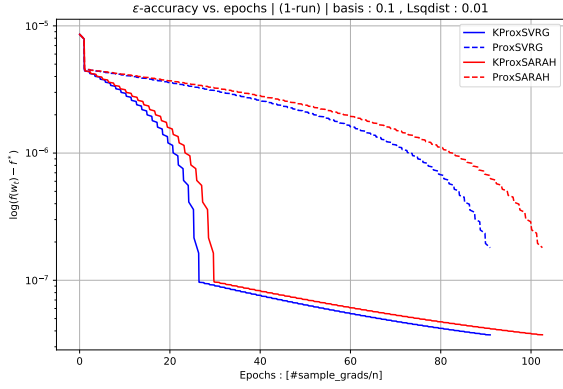


Figure 5.40: primal objective function
slack : $|S_t| = 10$, $m = \lfloor \sqrt{n} \rfloor$, $\eta = \frac{1}{10L}$

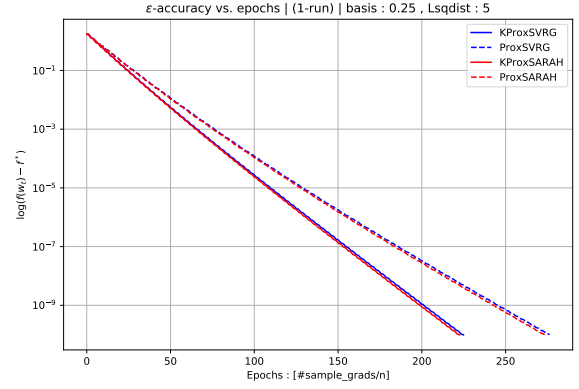


Figure 5.41: primal objective function
slack : $|S_t| = \lfloor \sqrt{n} \rfloor$, $m = \lfloor \sqrt{n} \rfloor$, $\eta = \frac{1}{4L}$

smoothed-hinge with l_1 reg. — (c) *Homemade*

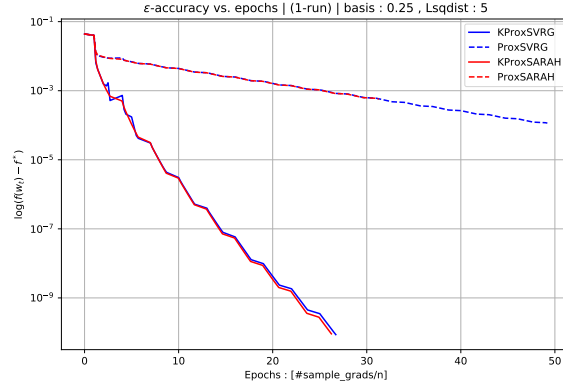


Figure 5.42: primal objective function slack : $|S_t| = \lfloor \sqrt{n} \rfloor$, $m = \lfloor \sqrt{n} \rfloor$, $\eta = \frac{1}{4L}$

logistic with l_1 reg. — (c) *Homemade* ($\lambda_{\text{in}} = \frac{1}{n}$) — multiple runs

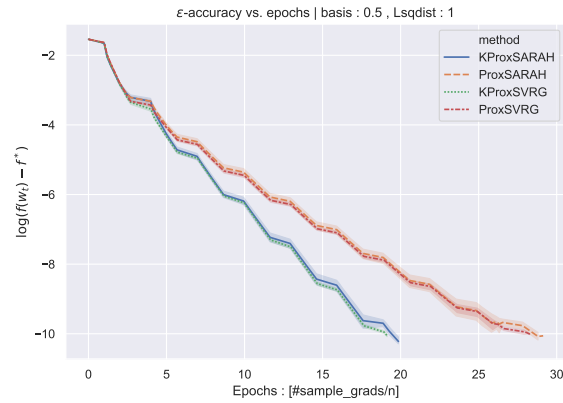


Figure 5.43: primal objective function slack : $|S_t| = \lfloor \sqrt{n} \rfloor$, $m = \lfloor \sqrt{n} \rfloor$, $\eta = \frac{1}{2L}$

Numerical Results

The quantity $\frac{\text{tr}(K_{\text{eff}})}{d}$ is of first importance and has been monitored. It is related to the mean multiplicative constant by which the *per coordinate* step-sizes have been multiplied. The smaller the best and the closer from 1 the least difference it will be observed between $\tilde{K}$ Prox-SVRG and classical Prox-SVRG.

	Figure 5.40	Figure 5.41	Figure 5.42	Figure 5.43
$\frac{\text{tr}(K_{\text{eff}})}{d}$	0.95	0.869	0.095	0.752

Comments

We can spot on Figure 5.42 the high impact, as forecasted in **Extreme Cases**, of $\tilde{K}$ Prox-SVRG in the case of a feature imbalance. Classical Prox-SVRG fails to capture the advantageous structure of the objective function. Conversely, $\tilde{K}$ Prox-SVRG/SARAH adapt well the *per-coordinate* step-sizes and take full-profit of the feature imbalance.

Figure 5.40 seems to show a terrific speed-up for $\tilde{K}$ Prox methods! Actually this is not case, the only reason explaining the observed graph is that $\tilde{K}$ Prox reaches first a favorable zone for which the optimization process "*accelerates*". Prox-SVRG takes more iterations but finally reaches that zone as well. It is the other way around in the next subsection.

The above experiments were added lately to this master thesis. It will be further deepened whether or not $\tilde{K}$ Prox-VR is really worth it. Nevertheless those first experiments were quite explicit and certainly promising.

Heuristic improvement

In the section 5.1. K -greedy sampling, it had been highlighted that for second eras methods, the range of step-sizes under which one can expect a one-step decrease becomes wider and wider as the optimization process progresses.

Let's consider having fixed a baseline step-size $\eta_{\text{lower}} > 0$ for $(\tilde{K})$ Prox-SVRG. My heuristic intends to propose at the beginning of each new inner-loop a step-size candidate, $\hat{\eta}$. Depending on the value of $\hat{\eta}$, larger step-sizes than η_{lower} might be used during that inner-loop.

The heuristic is described as follows. At any start of a new outer-loop iteration, values $\{\nabla\phi_i(\tilde{w}_s)\}_{i=1}^n$ are implicitly computed in order to build $\bar{\alpha}_t$ (see $\tilde{K}$ Prox-SVRG algorithm). Let us assume that these values are all stored¹³. As a reminder, this is not a big deal ($\mathcal{O}(n)$ extra memory) when the sample loss functions share a common structure $\tilde{\phi}$ of the form $\phi_i(\cdot) = \tilde{\phi}(\langle \tilde{x}_i, \cdot \rangle)$. The logistic, smoothed-hinge and quadratic losses fulfill this requirement. Indeed, one would store each $\nabla\phi_i(\tilde{w}_s)$ as the product of the scalar $\tilde{\phi}'(\tilde{w}_s)$ and the vector $\tilde{x}_i$ that should already be in memory!

At the end of an inner-loop starts a new outer-loop iteration. While keeping in memory the values $\{\nabla\phi_i(\tilde{w}_s)\}_{i=1}^n$, the new ones, namely $\{\nabla\phi_i(\tilde{w}_{s+1})\}_{i=1}^n$ are calculated. These latter define the current gradient of ϕ at the starting iterate for the upcoming inner-loop.

Based on these two sets, $\{\nabla\phi_i(\tilde{w}_{s+1})\}_{i=1}^n$ and $\{\nabla\phi_i(\tilde{w}_s)\}_{i=1}^n$, let's define *slacks* $\{\hat{\beta}_{s,i}\}_{i=1}^n$.

$$\hat{\beta}_{s,i} = \nabla\phi_i(\tilde{w}_{s+1}) - \nabla\phi_i(\tilde{w}_s) \quad \forall i \in [n]$$

¹³They won't be updated in the inner-loop though. $(\tilde{K})$ Prox-SAGA appeared "*too greedy*" in my experiments...

Assuming ϕ and its quadratic upper-bound $\bar{\phi}$ defined by $K^+ = K$ are not too far from each other, we have that the term $k + K\tilde{w} \simeq \nabla\phi(\tilde{w})$ at a given iterate $\tilde{w} \in \mathbb{R}^d$ in the RHS of (5.7). Then, at this point, the *slacks* $\{\hat{\beta}_{s,i}\}_{i=1}^n$ and the new value for $\bar{\alpha}_t$ ($\frac{1}{n} \sum_{i=1}^n \nabla\phi_i(\tilde{w}_{s+1})$) are enough to compute the terms of the RHS of (5.7). Let's denote by $\hat{\eta}$ the step-size minimizing that RHS. One can compute:

$$\hat{\eta} := \frac{\langle k + K\tilde{w}_{s+1}, \nabla\phi(\tilde{w}_{s+1}) \rangle}{-||\bar{\alpha}_t||_K^2 + 2\langle K\nabla\phi(\tilde{w}_{s+1}), \bar{\alpha}_t \rangle + \frac{1}{n^2} \sum_{i=1}^n \frac{||\hat{\beta}_{s,i}||_K^2}{p_i}} \simeq \frac{||\nabla\phi(\tilde{w}_{s+1})||^2}{-||\bar{\alpha}_t||_K^2 + 2\langle K\nabla\phi(\tilde{w}_{s+1}), \bar{\alpha}_t \rangle + \frac{1}{n^2} \sum_{i=1}^n \frac{||\hat{\beta}_{s,i}||_K^2}{p_i}}$$

$\hat{\eta}$ is an approximated optimal step-size for the next update step using a second era stochastic gradient and under the assumption that we use the old, rather loose, values $\{\nabla\phi_i(\tilde{w}_s)\}_{i=1}^n$ as last information at hand about the sample gradients. Obviously, only the new values $\{\nabla\phi_i(\tilde{w}_{s+1})\}_{i=1}^n$ will be used to create the stochastic gradients in the new inner-loop.

There is the trick. $\hat{\eta}$ will certainly be a very safe step-size for the first update steps of the new inner-loop since a fresh update of the sample gradients has been made available. However, as the inner-loop iteration counter progresses, the new values $\{\nabla\phi_i(\tilde{w}_{s+1})\}_{i=1}^n$ will become themselves poor estimates of the true sample gradients. $\hat{\eta}$ might become, at some point, too greedy itself. This heuristic's idea is to use η_{lower} if $\hat{\eta} \leq \eta_{\text{lower}}$ and a decreasing pattern of step-sizes from $\hat{\eta}$ to η_{lower} if $\hat{\eta} > \eta_{\text{lower}}$. E.g. $\eta_t = \eta_{\text{basis}} + (\eta_{\text{lower}} - \hat{\eta})(\frac{t \bmod m}{m})$.

logistic with l_1 reg. — (l) *Abalone Data Set*

logistic with l_1 reg. — (r) *Homemade*

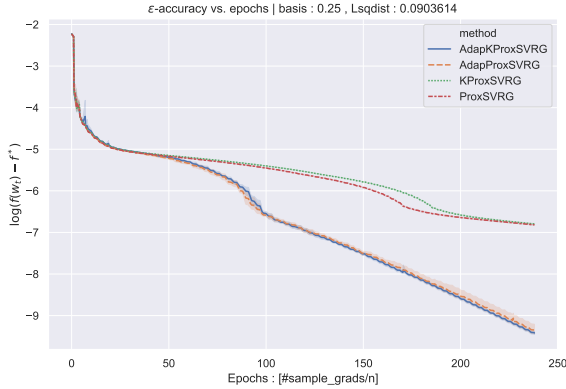


Figure 5.44: primal objective function slack : $|S_t| = 10$, $m = \lfloor \sqrt{n} \rfloor$, $\eta_{\text{lower}} = \frac{1}{10L}$

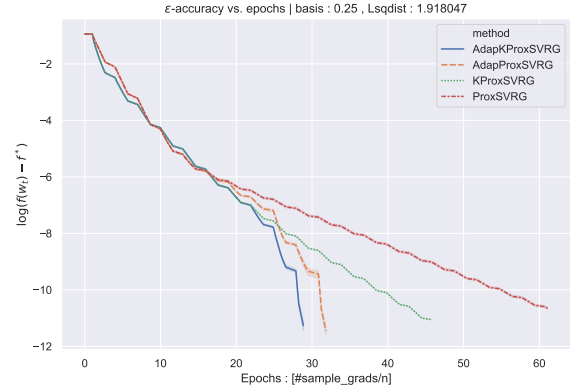


Figure 5.45: primal objective function slack : $|S_t| = \lfloor \sqrt{n} \rfloor$, $m = \lfloor \sqrt{n} \rfloor$, $\eta_{\text{lower}} = \frac{1}{4L}$

Numerical Results

Again the quantity $\frac{\text{tr}(K_{\text{eff}})}{d}$ is of first importance for $\tilde{K}$ Prox-VR methods.

	Figure 5.44	Figure 5.45
$\frac{\text{tr}(K_{\text{eff}})}{d}$	0.95	0.78

Comments

One can observe the efficiency of *Adap* $\tilde{K}$ Prox-SVRG/*Adap*Prox-SVRG in these two experiments. Both figures show pretty well the epoch number from which $\hat{\eta}$ becomes larger than η_{lower} ; respectively 90 for Figure 5.44 and 20 for Figure 5.45.

Chapter 6

Index of state of the art Stochastic Gradient Methods

Purpose

The purpose of this last tiny chapter is to inform the reader about existing state of the art papers in the domain of stochastic gradient methods. It is not an easy task to compose a state of the art in such a super-active domain as the research evolves continuously. Besides the research directions can be quite different. Hence I only intend to cite a few papers I have reviewed (or briefly read) per identified search direction.

Momentum Era

Based on Nesterov's momentum (excepted for Prox-SARAH), those methods present their own minor specificities. They all achieve, as they say, "*the best known oracle complexities*".

Katyusha [Zhu18]	MiG [ZSC18]
ASVRG [Sha+18]	Prox-SARAH [Pha+19]

Stochastic Quasi-Newton

This paper has already been cited in Chapter 5. As a reminder, it integrates a non uniform sampling strategy to build SVRG like stochastic gradients used both for the iterate update step and the creation of more and more accurate versions of the hessian in a L-BFGS fashion.

Stochastic L-BFGS [ZHT17]

Stochastic Quasi-Gradient

As mentioned in Chapter 4, JacSketch like methods aim to build/use the best representation possible of all the sample gradients, stored in a matrix called the *Jacobian sketch*.

JacSketch [GRB01]

Anchor based

S3GD is an outstanding paper. It proposes to create a network of a limited number of "*anchors*", a manifold, that represents an embedding for the original data points $\{\tilde{x}_i\}_{i=1}^n$. Then SVRG like update steps are applied using the anchors gradients.

S3GD [MLF16]

Conclusion

Let's summarize in this conclusion what are the main take away messages at the end of this extensive work about stochastic gradient methods. *It is also the occasion to underline what are the possible future research directions after this master thesis.*

Review & Unification

The unification of stochastic gradient methods under eras (classes), well defined, has been a mindful idea. Three main eras were distinguished, namely the *Pioneer Era*, the *Variance Reduction Era* and, finally, the *Momentum Era*. However, Chapter 6 showed that there exist methods falling outside those three eras; e.g. S3GD [MLF16] or QSN [Byr+15]. This master thesis shed light on properties shared by methods belonging to the same era. Only the two first eras cited above have been properly defined and studied since the last era is too wide; e.g. a common structure is hard to identify. Two rather general sampling schemes were exposed in order to build the stochastic gradient inherent to methods from the first/second eras.

Then, two lemmas per era were devised in order to state sufficient conditions under which a stochastic gradient is an unbiased estimate of the true gradient.

⇒ Those lemmas together with the sampling schemes proposed allows the reader to think about his/her own first/second era methods. A door is left open to build methods for which the weighting coefficients in front the sample gradients are tuned in the most efficient way.

Both theoretically and experimentally, this master thesis has shown the impact of the quantity $\text{tr}(\text{MSE}(g))$ (where g is an estimate of the true gradient ϕ) on the convergence of stochastic gradient based methods. One key aspect that has been highlighted many times is the fact that first era methods are unable to get $\text{tr}(\text{MSE}(g))$ converging to 0. Hence a decreasing step-size schedule must be employed yielding sublinear rates of convergence. Conversely, for second era methods the term $\text{tr}(\text{MSE}(g))$ typically vanishes. The use of a constant step size is then allowed and a linear rate of convergence is observed.

New Approaches

Traditional importance sampling for stochastic optimization that had been presented in [ZZ15] is not optimal in every context. A new definition of optimality for importance sampling in stochastic gradient methods is introduced and validated. When the objective function can be approximated by a quadratic function, for both eras, closed forms generalizing the traditional importance sampling were derived. Two generic comments have been stated in what concerns the use of non-uniform sampling. Firstly, it leads to better convergence bounds and are well supported by theory. The one-step expectation tests revealed that the use of well suited non-uniform sampling techniques expanded the range of step-sizes for which a decrease is expected. Secondly, the studied non-uniform samplings cannot be applied efficiently nowadays.

The tractable approximated versions are disappointing.

⇒ The problem of approximating powerful non-uniform samplings remains open.

Finally, it has been shown that the knowledge of a matrix $K^+ = K$ defining a quadratic upper-bound on the smooth part of f can be used to set up two new efficient methods: $\tilde{K}$ Prox-SVRG and $\tilde{K}$ Prox-SARAH. Thanks to a SDP problem solved in a pre-processing fashion, $\tilde{K}$ Prox-SVRG and especially its heuristic improvement seem very promising. The linear convergence rate of $\tilde{K}$ Prox-SVRG has been demonstrated.

$\Rightarrow \tilde{K}$ Prox-VR techniques must be tested on other types of problems to assess their robustness. Due to the lack of time, this has not been possible in the present work.

$\Rightarrow$ Theoretical support must be found for the working heuristic of $(\tilde{K})$ Prox-SVRG.

A few final words...

As stated in Chapter 5, this research domain (stochastic gradient based methods) is super-active and it has not been easy to innovate. Many times, Sebastian and I (first semester) or François and I (second semester), had an idea together that actually had been already deepened. The work of other people in the same domain can either be seen as a burden or as an opportunity to start again on a new basis. With positivity, courage and dedication, the only possible outcomes are great things.

Appendix

The first commitment to this appendix is the rationale about the relinquishment in applying a dependent sequential lottery scheme in order to produce stochastic gradients of the first era.

1.) Depending sequential lotteries on $\{0\} \cup [n]$

The goal is to draw sequentially D indices among $\{0\} \cup [n]$. Contrary to the **independent** case treated in 4.1.2, in this last setting one fixes *in advance* only the first probability distribution $\mathcal{P}_{t,1}$. For any $l \geq 2$, $\mathcal{P}_{t,l}$ will depend on the historic of the generated samples $\tilde{i}_{t,1} \sim \mathcal{P}_{t,1}, \dots, \tilde{i}_{t,l-1} \sim \mathcal{P}_{t,l-1}$. Conceptually this sampling technique is feasible. Despite its simplicity it is rather hard, in the general case, to provide *weighting coefficients* in such a way that g_t is unbiased ! Let's first express the expectation of g_t

$$\mathbb{E}[g_t | \mathcal{F}_t] = \sum_{\tilde{\mathbf{i}}_t \in \{0, \dots, n\}^D} p_{t, \tilde{i}_{t,1}, 1} \left(\prod_{l=2}^D p_{t, \tilde{i}_{t,l}, l}(\{\tilde{i}_{t,1}, \dots, \tilde{i}_{t,l-1}\}) \right) \Lambda(\{\tilde{\mathbf{i}}_t\}) \sum_{i \in \{\tilde{\mathbf{i}}_t\}} m_{t,i,l(i)} \nabla \phi_i(w_t) \quad (6.1)$$

Unfortunately here whatever the constant value that $\Lambda(\tilde{\mathbf{i}}_t)$ might take, the expectation cannot be split into D chunks as for the independent case. Depending on the structural dependency of an univariate lottery l with respect to the previous ones $1, \dots, l-1$ and depending on the value of l intrinsically, the computation complexity of the true $p_{t, \tilde{i}_{t,l}, l}$ is **exponentially hard** ! This result might seem rather disappointing and perhaps surprising. Let's bring up on the table an easy-to-read example that will clarify this fact.

Example : Let's assume one would like to sample, **without replacement**, from an initial *proper* univariate distribution $\mathcal{P}_{t,1}$ on $[n]$ (here we discard the possibility to draw indice 0, we use the notation i instead of $\tilde{i}$) This operation is usually defined as follows.

$$i_{t,1} \sim \mathcal{P}_{t,1}$$

Then, for all $l \in \{2, \dots, D\}$ ($D \leq n$ of course), $i_{t,l} = i$ with probability $P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1})$ defined by

$$P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1}) = \begin{cases} \frac{P_{t,i,l-1}(\{i_{t,v}\}_{v=0}^{l-2})}{\sum_{k \in [n], k \neq \tilde{i}_{t,l-1}} P_{t,k,l-1}(\{i_{t,v}\}_{v=0}^{l-2})} & \text{if } i \text{ not chosen yet at any step in } \{2, \dots, l-1\} \\ 0 & \text{if } i \text{ already chosen once at any step in } \{2, \dots, l-1\} \end{cases}$$

By convention, as there are no other source of randomness at the first univariate sample generation $P_{t,i,1}(\{i_{t,0}\}) := P_{t,i,1} := p_{t,i,1}$ This scheme is obviously recursive.

$$\text{After, when looking at } l \geq 2, P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1}) \neq p_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1})$$

This formula looks pretty straightforward. One shrinks the effective probability of taking an indice i if i has been chosen once in the previous univariate sample generations. Nevertheless the reader may already realize that the right hand side of the last equation, about the conditions, leads to an exponential growth of computations to get $p_{t,i,l}$ as l increases ! Let's show that.

When $D = 2$, the case is tractable yet not so useful when one intend to use a batch stochastic gradient... The reader should keep in mind that all the $p_{t,i,1}$ are known. There exist only two possible outcomes at $l = 2$.

- i has been chosen at $l = 1$ ($i_{t,1} = i$) with probability of occurrence $p_{t,i,1}$
- i has **not** been chosen at $l = 1$ ($i_{t,1} \neq i$) with probability of occurrence $1 - p_{t,i,1}$

The overall probability that $i_{t,2} = i$ is given in a closed form as

$$p_{t,i,2}(\{i_{t,1}\}) = (1 - p_{t,i,1}) \cdot P_{t,i,2}(\{i_{t,1}\}) = \frac{(1 - p_{t,i,1}) p_{t,i,1}}{\sum_{k \in [n], k \neq i_{t,1}} p_{t,k,1}} \quad (6.2)$$

Indeed it combines the probability of **not** having been selected until l (here $(1 - p_{t,i,1})$) and the effective probability at lottery l to draw i , $P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1})$ (here $P_{t,i,2}(\{i_{t,1} \neq i\}) = \frac{p_{t,i,1}}{\sum_{k \in [n], k \neq i_{t,1}} p_{t,k,1}}$ thanks to the fact that $i_{t,1} \neq i$).

Let's now try to tackle the case $D = 3$ in order to get the general intuition.

Again there are only two possible outcomes...

- i has already been chosen at any step of $\{1, 2\}$ ($i_{t,v} = i$ for one $v \in \{1, 2\}$) with probability of occurrence $p_{t,i,1} + p_{t,i,2}$
- i has **not** been chosen yet at any step of $\{1, 2\}$ ($i_{t,v} \neq i$ for every $v \in \{1, 2\}$) with probability of occurrence $1 - p_{t,i,1} - p_{t,i,2}$

The overall probability that $i_{t,3} = i$ is given in a closed form by

$$p_{t,i,3}(\{i_{t,1}, i_{t,2}\}) = (1 - p_{t,i,1} - p_{t,i,2}(\{i_{t,1}\})) \cdot P_{t,i,3}(\{i_{t,1}, i_{t,2}\}) \quad (6.3)$$

The computational complexity of $P_{t,i,3}(\{i_{t,1}, i_{t,2}\})$ is already one order above the one for $P_{t,i,2}(\{i_{t,1}\})$. Indeed, as we know that neither $i_{t,1}$ nor $i_{t,2}$ was equal to i by assumption, $P_{t,i,3}(\{i_{t,1}, i_{t,2}\})$ should be equal to $\frac{P_{t,i,2}(\{i_{t,1}\})}{\sum_{k \in [n], k \neq i_{t,2}} P_{t,k,2}(\{i_{t,1}\})}$. All the $P_{t,i,2}(\{i_{t,1}\})$ have to be computed and stored in memory in order to get a single $p_{t,i,3}$...

By induction one can show that

$$p_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1}) = (1 - \sum_{q=1}^{l-1} p_{t,i,q}(\{i_{t,v}\}_{v=0}^{q-1})) \cdot P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1})$$

As the reader has probably guessed, this phenomenon is of exponential nature as l increases making the computation of the $p_{t,i,l}$ not practical... The problem is that those $p_{t,i,l}$ are necessary to build an unbiased stochastic gradient in this sampling scheme !

Dependent sequential lotteries are a field of interest for stochastic optimization methods of every era since it can be shown that sampling without replacement is more efficient to reduce the variance of the norm of the stochastic gradient. Unfortunately apart from the case of (US) without replacement (for which independently of the flow of samples $\{i_{t,v}\}_{v=0}^{l-1}$, $P_{t,i,l}(\{i_{t,v}\}_{v=0}^{l-1}) = \frac{1}{n-l+1}$ yielding tractable computations) it's mostly not practical to draw $\hat{b}$ -size batches and trying to get an unbiased g_t . As *heuristic* though one can apply the generation algorithm proposed above while choosing $m_{t,i,l}$ exactly the same way as for the independent sequential lotteries while imposing that the values of $p_{t,i,l}$ are the ones of (US) without replacement.

One other idea, whenever the sampling is not step-dependent (understand : dependency towards t), is to proceed to Monte-Carlo simulations in order to estimate those probabilities $p_{t,i,l}$. This would be done in a pre-processing fashion.

For the reasons cited above *dependent sequential lotteries* were not investigated in this work. One just have to keep in mind that this sampling scheme might be interesting in practice but the true expectation of g_t is rarely accessible then.

The second commitment to this appendix is about homemade datasets generation and the creation of homemade classification problems for the logistic/smoothed-hinge/quadratic losses.

2.) Datasets and problems generation

Data generation

Let's first mention that all my Jupyter Notebooks will be provided on Dropbox/codes for full transparency. Nevertheless, here below the reader can find a small recipe of mine to create "*homemade*" datasets.

1. First, n and d are fixed in $\mathbb{N}$.
2. A reference norm is chosen for the samples to generate; usually between 0.5 and 10.
3. Samples $\{x_i\}_{i=1}^n$ are drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma)$, centered in $\mu \sim \mathcal{U}([-5, 5]^d)$ and presenting a covariance matrix $\Sigma = 2I + \nu\nu^T$ where $\nu \sim \mathcal{U}([0, 1]^d)$.
4. If a perturbation is wanted (as in one of the experiments of Chapter 5, section 2), for every x_i , a coordinate $j_i \in [n]$ is chosen at random and then x_i, j_i is increased by the desired amplitude. The perturbation coordinate might be deterministic.
5. The choice is made to rescale every vector to the same reference norm (default case) or, for experiments that required the Lipschitz constants of the sample gradients to present a large volatility, each $x_i \leftarrow R \cdot \frac{x_i}{\|x_i\|} \cdot (\frac{i+1}{n+1})^2$ for a scaling parameter R (10 by default).
6. It remains to instantiate the $\{y_i\}_{i=1}^n$. To that purpose, a generator vector $w_{\text{basis}} \sim \mathcal{U}([-5, 5]^d)$ is uniformly drawn and then $y_i = \text{sgn}(\langle x_i, w_{\text{basis}} \rangle) \forall i \in [n]$. In order for the problem to gain in complexity, a proportion P_f of "*sign flips*" is decided beforehand ($\simeq 15\%$). That means that $\lceil nP_f \rceil$ indices are drawn randomly and their corresponding y switches from +1 to -1 and the other way around.

Determination of a valid w_*

In order to get a very accurate reference solution for (1.1) I first run FISTA [BT09] until I get an approximate solution $\hat{w}$ for which the norm of the subgradient is low enough ($\leq 10^{-4}$ usually). Then, I intend to add a sample to the created dataset that will "correct" this non-negligible subgradient norm. At the end of the day, subgradient's norm at $\hat{w}$ will be of order 10^{-17} which's accurate enough! Let $\tilde{x}_{n+1}$ represent a sample fulfilling the requirements.

- If λ scales with n , e.g. $\lambda = \frac{c}{n}$, $\tilde{x}_{n+1}$ is chosen such that

$$\mathbf{0} \in \frac{n}{n+1} \nabla \phi_{-(n+1)}(\hat{w}) + \frac{\nabla \phi_{n+1}(\hat{w})}{n+1} + \frac{nc}{n+1} \partial R(\hat{w})$$

- If λ is not a function of n , e.g. $\lambda = c$, $\tilde{x}_{n+1}$ is chosen such that

$$\mathbf{0} \in \frac{n}{n+1} \nabla \phi_{-(n+1)}(\hat{w}) + \frac{\nabla \phi_{n+1}(\hat{w})}{n+1} + c \partial R(\hat{w})$$

The solution of these problems are implemented in the Jupyter Notebook files.

For $\tilde{x}_{n+1}$ to be accepted, its norm **must be** smaller or equal to the mean norm of the samples (in order not to assign too much importance on this last added sample).

The last commitment to this appendix is a small proof of convergence of $\tilde{K}$ Prox-GD

3.) Convergence of $\tilde{K}$ Prox-GD

The proof given below is directly inspired from **Theorem 5** of [KNS18].

ϕ , $\hat{L}$ -smooth w.r.t $\psi(\cdot) = \frac{1}{2} \|\cdot\|_{K_{\text{eff}}}^2$, fulfills K_{eff} proximal PL's condition iff.

$$\exists \hat{\mu} > 0 \text{ s.t. } \forall u \in \mathbb{R}^d, \mathcal{D}^{K_{\text{eff}}}(u, \hat{L}) \geq \hat{\mu}(f(u) - f^*)$$

where

$$\mathcal{D}^{K_{\text{eff}}}(u, c) := -2c \arg \min_{w \in \mathbb{R}^d} \left\{ \langle \nabla \phi(u), w - u \rangle + \frac{c}{2} \|w - u\|_{K_{\text{eff}}}^2 + R(w) - R(u) \right\}$$

Theorem 5 (linear convergence for $\tilde{K}$ Prox-GD under K_{eff} proximal PL condition and smoothness assumptions) Be $\hat{L} > 0$ and $\hat{\mu} > 0$ constants. If we assume that ϕ that is $\hat{L}$ -smooth w.r.t $\frac{1}{2} \|\cdot\|_{K_{\text{eff}}}^2$ fulfills K_{eff} proximal PL condition with parameter $\hat{\mu}$ then after T iterations of $\tilde{K}$ Prox-GD,

$$f(w_T) - f^* \leq (1 - \hat{\mu})^T [f(w_0) - f^*]$$

Proof : Considering any iterate $w_{t+1} \in \mathbb{R}^d$ produced by $\tilde{K}$ Prox-GD from a $w_t \in \mathbb{R}^d$. $\hat{L}$ -smoothness ensures that :

$$\begin{aligned} f(w_{t+1}) &= \phi(w_{t+1}) + R(w_{t+1}) \\ &\leq \phi(w_t) + \langle \nabla \phi(w_t), w_{t+1} - w_t \rangle + \frac{\hat{L}}{2} \|w_{t+1} - w_t\|^2 + R(w_{t+1}) - R(w_t) + R(w_t) \\ &\leq f(w_t) + \langle \nabla \phi(w_t), w_{t+1} - w_t \rangle + \frac{\hat{L}}{2} \|w_{t+1} - w_t\|^2 + R(w_{t+1}) - R(w_t) \\ &\leq f(w_t) - \frac{1}{2\hat{L}} \mathcal{D}^{K_{\text{eff}}}(w_t, \hat{L}) \\ &\leq f(w_t) - \frac{\hat{\mu}}{\hat{L}} (f(w_t) - f^*) \end{aligned}$$

□

Bibliography

- [Bec17a] Amir Beck. “First-Order Methods in Optimization”. In: SIAM, 2017. Chap. 6.
- [Bec17b] Amir Beck. “First-Order Methods in Optimization”. In: SIAM, 2017. Chap. 10.
- [Bot98] L. Bottou. *Online Algorithms and Stochastic Approximations: Online Learning and Neural Networks*. Cambridge University Press, 1998.
- [BT09] A. Beck and M. Teboulle. “A fast iterative shrinkage-thresholding algorithm for linear inverse problems”. In: (2009).
- [Byr+15] R. H. Byrd et al. “A Stochastic Quasi-Newton Method for Large-Scale Optimization”. In: (2015). URL: <https://arxiv.org/pdf/1401.7020.pdf>.
- [DBL14] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. “SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives”. In: (2014). URL: <https://arxiv.org/pdf/1407.0202.pdf>.
- [GRB01] Robert M. Gower, Peter Richtárik, and Francis Bach. “Stochastic Quasi-Gradient Methods: Variance Reduction via Jacobian Sketching”. In: (201). URL: <https://arxiv.org/pdf/1805.02632v1.pdf>.
- [HR18] Samuel Horváth and Peter Richtárik. “Nonconvex Variance Reduced Optimization with Arbitrary Sampling”. In: (2018). URL: <https://arxiv.org/abs/1809.04146>.
- [JZ13] Rie Johnson and Tong Zhang. “Accelerating Stochastic Gradient Descent using Predictive Variance Reduction”. In: (2013). URL: <https://papers.nips.cc/paper/4937-accelerating-stochastic-gradient-descent-using-predictive-variance-reduction.pdf>.
- [KEÖ15] Ritesh Kolte, Murat Erdogdu, and Ayfer Özgür. “Accelerating SVRG via second-order information”. In: (2015). URL: http://opt-ml.org/papers/OPT2015_paper_41.pdf.
- [KG16] Tamás Kern and András György. “SVRG++ with Non-uniform Sampling”. In: (2016). URL: http://opt-ml.org/oldopt/papers/OPT2016_paper_31.pdf.
- [KNS18] Hamed Karimi, Julie Nutini, and Mark Schmidt. “Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Lojasiewicz Condition”. In: (2018). URL: <https://arxiv.org/pdf/1608.04636.pdf>.
- [MLF16] Yadong Mu, Wei Liu, and Wei Fan. “Stochastic Gradient Made Stable: A Manifold Propagation Approach for Large-Scale Optimization”. In: (2016). URL: <https://arxiv.org/pdf/1506.08350.pdf>.
- [Nes98] Yurii Nesterov. *Introductory Lectures on Convex Programming Volume: A Basic course, volume I*. 1998. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.693.855&rep=rep1&type=pdf>.
- [Ngu+17] Lam M. Nguyen et al. “Stochastic Recursive Gradient Algorithm for Nonconvex Optimization”. In: (2017). URL: <https://arxiv.org/pdf/1703.00102.pdf>.
- [Pha+18] Nhan H. Pham et al. “When Does Stochastic Gradient Algorithm Work Well?” In: (2018). URL: <https://arxiv.org/pdf/1703.00102.pdf>.

- [Pha+19] Nhan H. Pham et al. “ProxSARAH: An Efficient Algorithmic Framework for Stochastic Composite Nonconvex Optimization”. In: (2019). URL: <https://arxiv.org/pdf/1902.05679.pdf>.
- [QR14] Zheng Qu and Peter Richtárik. “Coordinate descent with arbitrary sampling II: expected separable over-approximation”. In: (2014). URL: https://www.researchgate.net/publication/270222481_Coordinate_Descent_with_Arbitrary_Sampling_II_Expected_Separable_Overapproximation.
- [RM51] Herbert Robbins and Sutton Monro. “A Stochastic Approximation Method”. In: (1951). URL: <https://projecteuclid.org/euclid.aoms/1177729586#abstract>.
- [Sa15] Mark Schmidt and al. “Non-Uniform Stochastic Average Gradient Method for Training Conditional Random Fields”. In: (2015). URL: <https://arxiv.org/pdf/1504.04406.pdf>.
- [Sha+18] Fanhua Shang et al. “ASVRG: Accelerated Proximal SVRG”. In: (2018). URL: <https://arxiv.org/pdf/1810.03105.pdf>.
- [She+16] Zebang Shen et al. “Adaptive Variance Reducing for Stochastic Gradient Descent”. In: (2016). URL: <https://www.ijcai.org/Proceedings/16/Papers/284.pdf>.
- [SLB13] Mark Schmidt, Nicolas LeRoux, and Francis Bach. “Minimizing Finite Sums with the Stochastic Average Gradient”. In: (2013). URL: <https://arxiv.org/pdf/1309.2388.pdf>.
- [Sta] StanfordMath301. *Proximal Gradient Descent Method*. URL: https://statweb.stanford.edu/~candes/math301/Lectures/proximal_methods.pdf.
- [Tan+16] Conghui Tan et al. “Barzilai-Borwein Step Size for Stochastic Gradient Descent”. In: (2016). URL: http://www.optimization-online.org/DB_FILE/2016/05/5458.pdf.
- [XZ14] Lin Xiao and Tong Zhang. “A Proximal Stochastic Gradient Method with Progressive Variance Reduction”. In: (2014). URL: <https://arxiv.org/pdf/1403.4699.pdf>.
- [ZHT17] Renbo Zhao, William B. Haskell, and Vincent Y. F. Tan. “Stochastic L-BFGS: Improved Convergence Rates and Practical Acceleration Strategies”. In: (2017). URL: <https://arxiv.org/pdf/1704.00116.pdf>.
- [Zhu18] Zeyuan Zhu. “Katyusha: The First Direct Acceleration of Stochastic Gradient Methods”. In: (2018). URL: <https://arxiv.org/pdf/1603.05953.pdf>.
- [ZSC18] Kaiwen Zhou, Fanhua Shang, and James Cheng. “A Simple Stochastic Variance Reduced Algorithm with Fast Convergence Rates”. In: (2018). URL: <https://arxiv.org/pdf/1806.11027.pdf>.
- [ZZ15] Peilin Zhao and Tong Zhang. “Stochastic Optimization with Importance Sampling”. In: (2015). URL: <https://arxiv.org/pdf/1401.2753.pdf>.

Note : For each paper, only its latest online version is quoted.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | www.uclouvain.be/epl