Weakly Convex Optimization

Danail Krzhalovski,
Giacomo Rotondi,
Francesca Di Matteo,
Sandra Andovska,

University of Padua – Department of Mathematics,
Optimization – aa 2019-2020,
Professor: Francesco Rinaldi,

July 11, 2020

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
MATEMATICA

Contents

1 Introduction

We consider algorithms for solving weakly convex optimization problems, a wide class of (possibly nondifferentiable) nonconvex optimization problems. These problems take the form

$$\min_{x \in \mathbb{R}^n} f(x)$$

where f is a **weakly convex** function.

We analyze algorithms to deal with such class of functions both in a deterministic and stochastic framework. In both contexts we have *proximal methods*, i.e. methods that use the *proximal operator* to smooth a possibly non smooth problem. The concept of proximal operator and its use for optimization is explained in the following sections.

The analyzed algorithms for the **deterministic framework** are three *incremental algorithms*, whose convergence rate is stated in the weakly convex setting, in relation to different assumptions and settings of the parameters. Incremental algorithms are then compared to their stochastic counterparts.
In the **probabilistic framework** we illustrate the *stochastic proximal algorithms*, which are methods that again use the *proximal operator* to minimize stochastic approximations of an objective function, and *adaptive gradient algorithms*, with a focus on *exponential moving average-type* ones, which are related to recursively improve learning rates for gradient methods.

Before illustrating the algorithms and the differences between the frameworks, **weak convexity** is first addressed. Properties and notation common to the three classes of algorithm are denoted in the following section, while notation specific to a single class problems is addressed within the specific section.
Detailed experiments are then conducted following a similar approach as in the Li et al. paper.

2 Weak convexity

Some preliminary notation is needed for characterizing the problem at hand.

Definition 1: *For a function $\varphi : \mathbf{R^d} \to \mathbf{R} \cup \{+\infty\}$, let*

$$\text{dom}\varphi := \{x \in \mathbf{R^d} : \ \varphi(x) < \ +\infty\}, \text{ and}$$
$$\text{epi}\varphi := \{(x, z) \in \mathbf{R^d} \text{ x } \mathbf{R} : \ | \ \varphi(x) \leq \ z\}.$$

Then, a function φ is said to be closed if epiφ *is a closed set. A function φ is called proper if its effective domain is non-empty and it does not attain $+\infty$.*

Definition 2: *A function φ is Lipschitz continuous if there exists $L > 0$ such that $|\varphi(x) - \varphi(y)| \leq L\|x - y\| \quad \forall x, y \in \text{dom}\varphi$. Furthermore, a function ϕ is $\mathcal{M}_\phi$-smooth if its gradient ∇_ϕ is $\mathcal{M}_\phi$-Lipschitz continuous.*

Definition 3: *A function $\sigma : \mathbb{R}^n \to \mathbb{R}$ is said to be **weakly convex** with parameter $\tau \geq 0$ if the assignment $\mathbf{x} \to \sigma(\mathbf{x}) + \frac{\tau}{2}\|\mathbf{x}\|^2$ is convex.*

Weak convexity of σ with parameter τ is equivalent to:

$$\sigma(\mathbf{w}) - \sigma(\mathbf{x}) \geq \langle \mathbf{d}, \mathbf{w} - \mathbf{x} \rangle - \frac{\tau}{2}\|\mathbf{w} - \mathbf{x}\|^2, \forall \mathbf{d} \in \partial\sigma(\mathbf{x})$$

for any $\mathbf{w}, \mathbf{x} \in \mathbb{R}^{\mathrm{n}}$

3 Proximal operators and algorithms

The *proximal operator* $\mathbf{prox}_f : R^n \to R^n$ pf f is defined as

$$\mathbf{prox}_f(v) = \arg\min_x \{f(x) + \frac{1}{2}\|x - v\|_2^2\}$$

where $\|\cdot\|_2$ is the usual Euclidean norm. The function minimized is **strongly convex** and not everywhere infinite, so it has a *unique minimizer* for every $v \in R^n$.

The proximal operator can be applied to the scaled function λf, where $\lambda \geq 0$, which can be expressed as

$$\mathbf{prox}_{\lambda f}(v) = \arg\min_x \{f(x) + \frac{1}{2\lambda}\|x - v\|_2^2\}$$

Interpretation: The figure illustrates how the proximal operator works: the thick black line indicates the boundary of the domain of the convex function f. Applying $\mathbf{prox}_f$ to the blue points moves them to the corresponding red points. The three points in the domain of the function stay in the domain and move towards the minimum of the function, while the other two move to the boundary of the domain and towards the minimum of the function. The parameter λ controls the extent to which the proximal operator maps points towards the minimum of f.

Proximal algorithms: A proximal algorithm solves convex optimization problems using the proximal operators of the objective terms. For example, the proximal minimization algorithm, minimizes a convex function f by repeatedly applying $\mathbf{prox}_f$ to some initial point x_0:

$$x_{k+1} := \mathbf{prox}_{\lambda f}(x_k)$$

Definition 4: The **subdifferential** of a function σ at $\mathbf{x}$ is defined as

$$\partial\sigma(x) := \{\widetilde{\nabla}\sigma(\mathbf{x}) \in \mathbb{R}^{\mathrm{n}} : \liminf_{\mathrm{y}\to\mathrm{x}} \frac{\sigma(\mathrm{y}) - \sigma(\mathrm{x}) - \langle\widetilde{\nabla}\sigma(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle}{\|\mathbf{y} - \mathbf{x}\|} \geq 0\},$$

where $\widetilde{\nabla}\sigma(\mathbf{x}) \in \partial\sigma(\mathbf{x})$ is called a subgradient of σ at x.
Subgradient generalizes the derivative to convex functions which are not necessarily differentiable. The subdifferential of σ at x is the set of hyperplanes going through the point ($\mathbf{x}$, $\sigma(\mathbf{x})$) that are everywhere either touching or below the graph of σ. The slope of such hyperplanes is a subgradient $\widetilde{\nabla}\sigma(\mathbf{x})$.

The proximal operator $\mathbf{prox}_{\lambda f}$ and the subdifferential operator ∂f are related as follows:

$$\mathbf{prox}_{\lambda f} = (I + \lambda\partial f)^{-1}$$

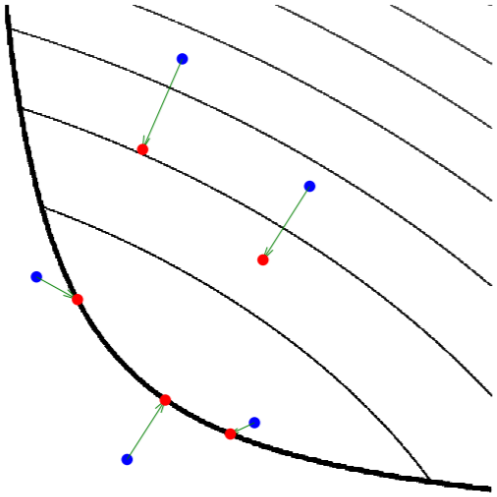


Figure 1.1: Evaluating a proximal operator at various points.

.

Moreau envelope

Searching for stationary points in the weakly convex setting, we need a continuous measure of the progress of algorithm, which cannot be found in gradient or subgradient because f need not be differentiable. Weakly convex problems naturally admit a continuous measure of stationarity through **implicit smoothing**. The surrogate optimality measure is found in the implicit smoothing provided by **Moreau envelope** and its corresponding **proximal** mapping:

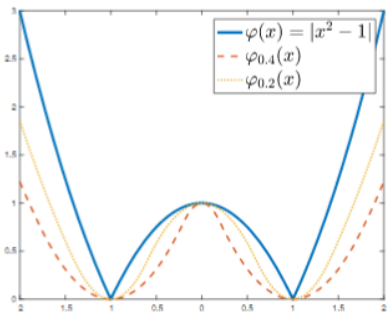
$$\varphi_\lambda(\mathbf{x}) := \min_{\mathbf{y}} \{\varphi(\mathbf{y}) + \frac{1}{2\lambda}\|\mathbf{y}-\mathbf{x}\|^2\}$$

$$prox_{\lambda\varphi}(\mathbf{x}) := \arg\min_{\mathbf{y}} \sigma(\mathbf{y}) + \frac{1}{2\lambda}\|\mathbf{y}-\mathbf{x}\|^2$$

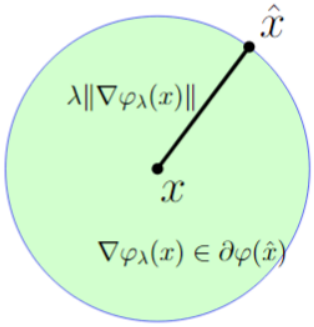
Moreau envelope approximates the objective function from below, i.e. $\varphi_\lambda(\mathbf{x}) \leq \varphi(\mathbf{x})$ $\forall \mathbf{x} \in \mathbb{R}^n$; $\lambda > 0$ is the penalty parameter controlling for the approximation error.
The Moreau envelope $\varphi_\lambda(\mathbf{x})$ is basically a smoothed form of $\varphi(\mathbf{x})$: it has domain R^n, even when φ does not, and it is continuously differentiable, even when φ is not.
Moreover if φ is τ-weakly convex, then φ_λ is smooth for any $\lambda < \tau^{-1}$, with gradient $\nabla\varphi_\lambda(\mathbf{x}) = \lambda^{-1}(\mathbf{x} - prox_{\lambda,x}(\mathbf{x}))$
The following results allows to assess how close x is to a critical point of f using its proximal map, denoted as where $\mathbf{x}^* = prox_{\lambda\varphi}(\mathbf{x})$.

$$\begin{cases} \lambda^{-1}\|\mathbf{x}-\mathbf{x}^*\| = \|\nabla\varphi_\lambda(\mathbf{x})\| \\ dist(0,\partial\varphi(\mathbf{x}^*)) \leq \|\nabla\varphi_\lambda(\mathbf{x})\| \end{cases}$$

If $\|\nabla\varphi_\lambda(\mathbf{x})\|$ is small, then $\mathbf{x}$ is close to $\bar{\mathbf{x}}$, which is nearly stationary since also dist$(0,\partial\varphi(\mathbf{x}^*))$ is small. The problems of minimizing f and $\varphi_{\lambda f}$ are thus equivalent, and the latter is always a smooth optimization problem. The convergence results of the analyzed algorithms are indeed stated in terms of $\|\nabla\varphi_\lambda(\mathbf{x})\|$.



(a) *Moreau envelope of* $\varphi(x) = |x^2 - 1|$.



(b) *Approximate stationarity.*

4 Incremental Algorithms

Incremental algorithms are suited for solving the finite sum optimization problem:

$$\min_{x\in\mathbb{R}^n} f(x) = \sum_{i=1}^{m} f_i(x)$$

We analyze the setting where each component function f_i is **weakly convex**, lower semi-continuous and lower bounded; the global minimum set χ is nonempty and closed.

The finite sum of weakly convex f_i is weakly convex by definition and its minimum value is denoted as f^*.

At each iteration, incremental methods update the value of $\mathbf{x}$ using only one component function f_i, cyclically selected from f_1 to f_m. Hence, at iteration $(k+1)$, $\mathbf{x_{k,0}} = \mathbf{x_k}$ which is updated as $\mathbf{x_{k,i}}$ using f_i for all $i = \{1, ..., m\}$, reaching $\mathbf{x_{k,m}} = \mathbf{x_{k+1}}$.

Each iteration is indeed made of m updates, using one after the other all the m component functions. The three algorithms that we illustrate share this basic *incremental* structure.

Incremental (sub-)gradient descent:

$$\mathbf{x}_{k.i} = \mathbf{x}_{k.i-1} - \mu_k \widetilde{\nabla} f_i(\mathbf{x}_{k.i-1})$$

where $\widetilde{\nabla} f_i(\mathbf{x}_{k.i-1})$ is any subgradient belonging to the subdifferential ∂f_i.

When the objective function is differentiable, sub-gradient descent for unconstrained problems use the same search direction as the method of gradient descent.

Incremental proximal point algorithm:

$$\mathbf{x}_{k.i} = \arg\min_{\mathbf{x}\in R^n} f_i(\mathbf{x}) + \frac{1}{2\mu_k}\|\mathbf{x} - \mathbf{x}_{k.i-1}\|^2$$

At each update the point $\mathbf{x}_{k,i}$ is set as the $\mathbf{prox}_{\mu_k f_i}(\mathbf{x}_{k,i-1})$. Broadly speaking at k-th, $\mathbf{x}_{k,i}$ is set as the one minimizing the current function f_i without moving too far away from the point minimizing f_{i-1}.

Incremental prox-linear algorithm:
It is used for a wide class of nondifferentiable weakly convex functions with the composite form

$$\sigma(\mathbf{x}) = h(c(\mathbf{x}))$$

where $h : R^d \to R$ is a Lipschitz convex function and $c : R^n \to R^d$ is a smooth mapping with Lipschitz continuous Jacobian.
Given this composition it is possible to denote

$$f_i(\mathbf{x}; \mathbf{x}_{k,i-1}) = h_i(c_i(\mathbf{x}_{k,i-1}) + \nabla c_i(\mathbf{x}_{k,i-1})^T(\mathbf{x} - \mathbf{x}_{k,i-1})$$

as convex relaxation of f_i.
The approximation of c equal to $c_i(\mathbf{x}_{k,i-1}) + \nabla c_i(\mathbf{x}_{k,i-1})^T(\mathbf{x} - \mathbf{x}_{k,i-1}$ has an upper bounded error that goes to 0 as $\mathbf{x}_{k,i-1}$ approaches x.
The update of the algorithm is then:

$$\mathbf{x}_{k.i} = \arg\min_{\mathbf{x}\in R^n} f_i(\mathbf{x}; \mathbf{x}_{k,i-1}) + \frac{1}{2\mu_k}\|\mathbf{x} - \mathbf{x}_{k.i-1}\|^2$$

The algorithm has again a proximal structure like **incremental proximal point** but it is applied on a composite approximated f.

For any $i \in \{1, .., m\}$ and $k \geq 0$, the Moreau envelope of f at $\mathbf{x}_{k,i}$ with penalty parameter $\lambda = 1/\hat{\tau}$ is denoted as:

$$f_{1/\hat{\tau}}(\mathbf{x}_{k,i}) := \min_{\mathbf{y}\in R^n} f(\mathbf{y}) + \frac{\hat{\tau}}{2}\|\mathbf{y} - \mathbf{x}_{k,i}\|^2 \tag{1}$$

$$\bar{\mathbf{x}}_{k,i} = prox_{1/\hat{\tau},f}(\mathbf{x}_{k,i}) := \arg\min_{\mathbf{y}\in R^n} f(\mathbf{y}) + \frac{\hat{\tau}}{2}\|\mathbf{y} - \mathbf{x}_{k,i}\|^2$$

One counterpart scheme to incremental methods are the stochastic algorithms which at each iteration take one component function independently and uniformly from $\{f_1, ..., f_m\}$ to update. Such sampling scheme plays a key role in the analysis of stochastic algorithms. For example, the random uniform sampling scheme results in an unbiased estimation of the full (sub)-gradient information in each iteration, which makes the stochastic algorithms in expectation the same as the one using full components.

5 Convergence rate

Before presenting the convergence result, we first the proximal updates of the incremental proximal methods are translated into similar forms as in the subgradient updates.
Lemma 1: For all $k \geq 0, 1 \leq 1 \leq m$:

1. for incremental proximal point method, there exists a subgradient $\tilde{\nabla} f_i(\mathbf{x}_{k,i}) \in \partial f_i(\mathbf{x}_{k,i})$ such that
$$\mathbf{x}_{k,i} = \mathbf{x}_{k,i} + \mu_k \tilde{\nabla} f_i(\mathbf{x}_{k,i})$$
2. for incremental prox-linear point method, there exists a subgradient $\tilde{\nabla} f_i(\mathbf{x}_{k,i}; \mathbf{x}_{k,i-1}) \in \partial f_i(\mathbf{x}_{k,i}; \mathbf{x}_{k,i-1})$ such that
$$\mathbf{x}_{k,i} = \mathbf{x}_{k,i} + \mu_k \tilde{\nabla} f_i(\mathbf{x}_{k,i}; \mathbf{x}_{k,i-1})$$

It is easy to note a similarity to the updates in incremental (sub)-gradient descent. The only difference is that the subgradients in incremental proximal methods are evaluated at $x_{k,i}$, while in incremental (sub)-gradient descent it is evaluated at $x_{k,i-1}$. This similarity is exploited for building a unique proof, with slight modifications, for the convergence rates of the three algorithms. The proofs are left to the referenced paper.

5.1 Sublinear convergence rate

Given the problem
$$\min_{x \in \mathbb{R}^n} f(x) \tag{2}$$
where:

1. Each component function f_i in **weakly convex** with parameter τ_i. We denote $\tau = max_{i \in \{1,..,m\}} \tau_i$;
2. There exists a constant $\tau \geq 0$ such that each component function f_i satisfies
$$f_i(\mathbf{x}; \mathbf{y}) - f_i(\mathbf{x}) \leq \frac{\tau}{2} \|\mathbf{x} - \mathbf{y}\|^2$$
.

***Assumption 1*: Bounded subgradients** For any $i \in \{1, .., m\}$ there exists a constant $L > 0$, such that $\|\widetilde{\nabla} f_i(\mathbf{x})\| \leq L$, for all $\widetilde{\nabla} f_i(\mathbf{x}) \in \partial f_i(\mathbf{x})$ and bounded $\mathbf{x}$.
This assumption concerns the **Lipschitz continuity** of each f_i.

Assumption 2 For any $i \in \{1, .., m\}$ there exists a constant $L > 0$, such that $\|\widetilde{\nabla} f_i(\mathbf{x}; \mathbf{y})\| \leq L$, for all $\widetilde{\nabla} f_i(\mathbf{x}, \mathbf{y}) \in \partial f_i(\mathbf{x}; \mathbf{y})$ and bounded $\mathbf{x}, \mathbf{y}$.

Convergence rate with fixed stepsize

Suppose **Assumption 1** is valid for incremental (sub)-gradient descent and incremental proximal point algorithm and **Assumption 2** is valid for incremental prox-linear method.

Setting the **stepsize** $\mu_k = \frac{1}{m\sqrt{N+1}}$ for all $K \geq 0$, where integer N is the total iteration number. The stepsize μ_k corresponds to the penalization parameter of the proximal operator for the proximal algorithms.

Then for any $\hat{\tau} \geq 2\tau$, if the sequence $\{\mathbf{x}_k\}$ is generated by one of the three incremental methods for solving (2) we have:

$$\min_{0\leq k\leq N} \|\nabla f_{1/\hat{\tau}}(\mathbf{x}_k)\|^2 = \mathcal{O}(\frac{1}{\sqrt{N+1}})$$

The optimality of x_k is indeed stated in terms of squared norm of the gradient of the Moreau envelope $\nabla f_{1/\hat{\tau}}(\cdot)$ with suitable parameter $1/\hat{\tau}$.

5.2 Linear convergence rate

The paper illustrates that when the weakly convex function satisfies an additional regularity condition called **sharpness**, all the three incremental algorithms with a **geometrical diminishing** stepsize and an appropriate initialization converge linearly to the optimal solution set.

Definition 5: A mapping $\sigma : R^n \to R$ is said to be α-sharp where $\alpha \geq 0$ if

$$\sigma(\mathbf{x}) - \sigma^* \geq \alpha dist(\mathbf{x}, \chi)$$

for all $\mathbf{x} \in \mathbb{R}^n$, where χ denotes the set of global minimizers of σ, σ^* represents the minimal value of σ, and $dist(\mathbf{x}, \chi)$ is the distance of x to χ, i.e., $dist(\mathbf{x}, \chi) = \min_{x'\in\chi} \|\mathbf{x} - \mathbf{x}'\|$

The stepsize depends only on the problem parameters (sharpness parameters α, weak convexity parameter τ and local Lipschitz constant L) and is equal to:

$$\mu_k = \rho^k \mu_0$$

where

$$0 \leq \mu_0 \leq \frac{\alpha^2}{5m\tau L^2}, 1 \geq \rho \geq \bar{\rho} := \sqrt{1 - 2m\tau\mu_0 + \frac{5m^2\tau^2L^2}{\alpha^2}\mu_0^2}$$

Now, supposing each f_i is weakly convex and sharp and the previous assumptions hold, then the three incremental methods, ***appropriately initialized*** at a point $\mathbf{x}_0$ satisfying $dist(\mathbf{x}_0, \chi) \leq \alpha$ and using the step size (3), converge locally **linearly** to the optimal solution set of problem χ, i.e:

$$dist(\mathbf{x}_k, \chi) \leq \rho^k \cdot \frac{\alpha}{2\tau}, \forall k \geq 0$$

This result only characterizes sufficient (but possibly not necessary) conditions for the algorithms to converge and provides an upper bound for the sequence of distances to the optimal set χ.

It is worth noticing that the radius of initialization region equals to $\frac{\alpha}{2\tau}$, therefore the more sharp (larger α) or the closer to a convex function (smaller τ) f is, the more we are enabled to enlarge the initialization region. In the final sections the incremental algorithms are applied to two different practical problems.

6 Stochastic Optimization

The goal of stochastic optimization in data science is to investigate relationships between variables and learn decision rules from a sample of data, that should be well representative of the entire population of interest. Its general approach consists of successively sampling and minimizing simple stochastic models of the objective function, which, in data science, is usually given by the expected population risk. It is also common practice to include a regularization term r, which makes our stochastic optimization problem a minimization of the regularized population risk:

$$\min_{x \in \mathbb{R}^{\mathrm{d}}} \varphi(x) := f(x) + r(x)$$

where $f(x) = \mathbb{E}_{\xi \sim P}[f(x, \xi)]$. Here, ξ encodes the population data, assumed to follow some unknown but fixed probability distribution P, meaning that ξ is the random variable associated with the random sampling of the population. Indeed $f(x, \xi)$ evaluates the loss of the decision rule parametrized by x on an observed data sample ξ, while $r : R^d \to R \cup \{\infty\}$ constraints the parameters x to particular structural features, such as sparsity or low rank.

The *proximal stochastic (sub)gradient* method is one of the simplest and most widely used optimization algorithm, of particular success in the field of large scale problems in machine learning. In the simplest setting, where each possible $f(x, \xi)$ is smooth and strongly convex and $r = 0$, the algorithm searches for the minimizer of $\varphi(x)$ by constructing a sequence of approximations $\{\mathbf{x_t}\}$ and moving, at each iteration, from x_t towards the opposite direction of a sampled gradient:

$$\begin{cases} \text{sample } \xi_t \sim P \\ \text{set } x_{t+1} = prox_{\alpha_t r}(x_t - \alpha_t \nabla_x f(x_t, \xi_t)). \end{cases} \tag{3}$$

where $\alpha_t > 0$ is an appropriate stepsize and $prox_{\alpha r}(\cdot)$ is the proximal map:

$$prox_{\alpha r}(x) := \arg\min_y r(y) + \frac{1}{2\alpha}\|y - x\|^2.$$

This is equivalent to applying a gradient descent method to the proximal map of f.
Nonsmooth convex problems may be similarly optimized by replacing sample gradients with sample subgradients $v_t \in \partial_x f(x_t, \xi_t)$, where $\partial_x f(x, \xi)$ is the subdifferential. The stochastic gradient methods' performance is evaluated in terms of number of $i.i.d$ samples (realizations $\xi_1, ..., \xi_N \sim P$) required to reach the desired level of accuracy of the decision rule, called *sample complexity*. Despite its wide use, the full knowledge about this method is limited to the simplest setting described above. It is known that in case of stochastic gradient applied to:

- convex problems: the method requires $\mathcal{O}(\epsilon^{-2})$ samples to reach functional accuracy ϵ in expectation.
- smooth problems (nonconvex): the method requires $\mathcal{O}(\epsilon^{-4})$ samples to reach a point with the gradient norm at most ϵ in expectation.

However, the sample complexity of this method is still unknown as we move beyond convexity and smoothness. The analyzed paper aims [2] at providing complexity bounds for stochastic algorithms applied to a reasonably wide class of optimization problems, nonconvex and nonsmooth, in particular we will focus on weak convexity.
As explained in the previous sections, the primary goal of optimization is the search for stationary points. Due to nonconvexity of the objective function, the traditional stationary measure $dist(0, \partial\varphi(x))$ will not necessarily tend to zero along the iterations and a solution need to be found in order to evaluate the progress of the algorithm. Again, we will monitor the Moreau envelope, which can provide a continuous measure (approximated) of stationarity, instead of the highly discontinuous function $x \to dist(0, \partial\varphi(x))$. The properties of the Moreau envelope and of its proximal map have already been described in the previous sections.
The main contrib ution of the paper is showing that when $f(\cdot, \xi) + r(\cdot)$ are σ-weakly convex and mild Lipschitz conditions hold, the proximal stochastic subgradient method will converges to a solution x, such that $\mathbb{E}(\|\nabla_{\frac{1}{2\sigma}}(x)\|) \leq \epsilon$, after at most $\mathcal{O}(\epsilon^{-4})$ iterations.
We will see that the stochastic subgradient method is based on sampling subgradient estimates from f (section 6.1) or equivalently sampling linear models of the function (section 6.2), in which the setting is extended to weak convexity.

6.1 Proximal stochastic subgradient method

We consider the optimization problem:

$$min_{x \in R^d} \varphi(x) = f(x) + r(x)$$

where $r : R^d \to R \cup \{+\infty\}$ is a closed convex function and $f : R^d \to R$ is a ρ-weakly convex function.
The oracle concept used assumes that the only access to f is through a stochastic subgradient.
We fix a probability space $(\Omega, \mathbb{F}, P)$, equip R^d with a Borel σ-algebra and make the following *assumptions*:

- (A1) It is possible to generate i.i.d. realizations $\xi_1, \xi_2 ... \sim P$;
- (A2) There is an open set U containing $domr$ and a measurable mapping $G : U \times \Omega \to R^d$ satisfying $\mathbb{E}_\xi[G(x, \xi)] \in \partial f(x)$ for all $x \in U$;
- (A3) There is a real $L \geq 0$ such that the inequality $\mathbb{E}_\xi[\|G(x, \xi)\|^2] \leq L^2$ holds for all $x \in domr$

The proximal stochastic subgradient algorithm is then formalized as follows:

Algorithm 1 Proximal stochastic subgradient method

Input: Initial point$x_0 \in domr$, a sequence $\{a_t\}_{t\geq 0} \subset R_+$ and iteration count T.

Step $t = 0, ..., T$:

$$\begin{cases} sample\xi_t \sim P, \\ setx_{t+1} = prox_{\alpha_t r}(x_t - \alpha_t G(x_t, \xi_t)) \end{cases}$$

Sample $x_{t^*} \in \{x_t\}_{t=0}^T$ according to $\mathbf{P}(t^* = t) = \alpha_t / \sum_{i=0}^T \alpha_i$.

The method successively uses the proximal mapping with parameter α_t of subgradient updates.

Convergence rate: It can be proved that, fixing a real $\hat{\rho} \in (\rho, 2\rho]$ and a stepsize sequence $\alpha_t \in (0, 1/\hat{\rho}]$, the iterates x_t generated with the proximal stochastic subgradient algorithm satisfies:

$$\mathbb{E}[\varphi_{1/\hat{\rho}}(x_{t+1})] \leq \mathbb{E}[\varphi_{1/\hat{\rho}}(x_t)] - \frac{\alpha_t(\hat{\rho} - \rho)}{\hat{\rho}} \mathbb{E}[\|\nabla \varphi_{1/\hat{\rho}}(x_t)\|^2] + \alpha_t^2 \hat{\rho} L^2$$

and the point $\mathbf{x}_{t^*}$ returned by the algorithm satisfies:

$$\mathbb{E}[\|\nabla \varphi_{1/\hat{\rho}}(x_{t^*})\|^2] \leq \frac{\hat{\rho}}{\hat{\rho} - \rho} \cdot \frac{(\varphi_{1/\hat{\rho}}(x_0) - min\varphi) + 2\hat{\rho} L^2 \sum_{t=0}^T \alpha_t^2}{\sum_{t=0}^T \alpha_t}.$$

In particular if we use $\alpha_t = \frac{1}{2} min\left\{\frac{1}{\rho}, \sqrt{\frac{\Delta}{\rho L^2 (T+1)}}\right\}$ for some real $\Delta \geq \varphi_{1/\hat{\rho}}(x_0) - min\varphi$, then the output x_{t^*} satisfies:

$$\mathbb{E}[\|\nabla \varphi_{1/2\rho}(x_{t^*})\|^2] \leq 8 \cdot max\left\{\frac{\rho\Delta}{T+1}, L\sqrt{\frac{\rho\Delta}{T+1}}\right\}.$$

As we can see the convergence rate is defined in terms of expectation due to the stochastic setting, and in terms of Moreau envelope's gradient (the implicit smoothing of our loss function). The convergence rate above stated improves in case of convex and/or smooth functions f, we leave all the proofs to the papers.

6.2 Stochastic model-based optimization

Stochastic modeling comes into play if one wants to model the behaviour of the random estimator of the objective function f. Given an arbitrary f, suppose that for every point x we have available a family of "models" $\{f_x(\cdot, \xi)\}_{\xi \sim P}$, indexed by a random sample $\xi \sim P$.

Our oracle concept assumes that the f is only accessed by sampling a model $f_x(\cdot, \xi)$

centered around any base point x. The assignment $(x, y, \xi) \to f_x(\cdot, \xi)$ is a ***stochastic one-sided model*** if, $\forall x, y$, it satisfies:

- $\mathbb{E}_\xi[f_x(x, \xi)] = f(x)$
- $\mathbb{E}_\xi[f_x(y, \xi) - f(y)] \leq \frac{\tau}{2}\|y - x\|^2$

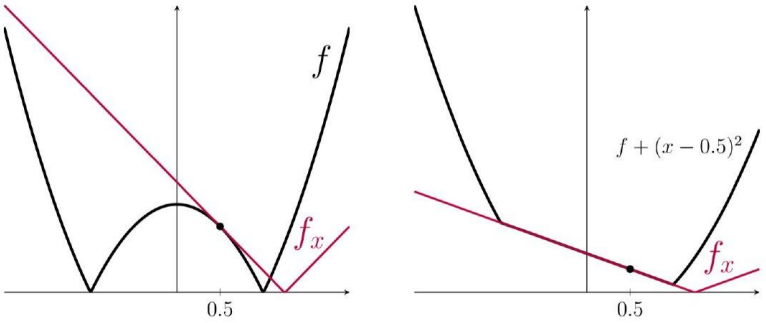


Illustration of a one-sided model: $f(x) = |x^2 - 1|$, $f_{0.5}(y) = |1.25 - y|$.

Thus, the sampled $f_x(\cdot, \xi)$ has to be an unbiased estimator of $f(x)$ and each model $f_x(\cdot, \xi)$ should lower bound f, in expectation, up to a quadratic error.
Our methods will then simply iterate:

$$\begin{cases} sample\, \xi \sim P \\ set\, x_{t+1} = \arg\min_y \{f_{x_t}(y, \xi_t) + r(y) + \frac{1}{2\alpha_t}\|y - x_t\|^2\} \end{cases}$$

One key aspect of the general algorithm is that it can be interpreted as an *approximate descent method* on the Moreau envelope.
Under mild Lipschitz conditions and provided that each function $f_x(\cdot, \xi) + r(\cdot)$ is τ-weakly convex, the algorithm converges to a point x with $\mathbb{E}\|\nabla\varphi_{\frac{1}{2\tau}}(x)\| \leq \epsilon$ after at most $\mathcal{O}(\epsilon^{-4})$

Let us consider now the setting of ***stochastic composite optimization***,

$$f(x, \xi) = h(c(x, \xi), \xi)$$

where the functions $h(\cdot, \xi)$ are convex and the maps $c(\cdot, \xi)$ are smooth. In the simplest setting when P is a discrete distribution on $\{1, ..., m\}$, the problem$\min_{x \in R^d} \varphi(x) = f(x) + r(x)$ reduces to minimizing a regularized empirical average of composite functions:

$$f(x) = \frac{1}{m}\sum_{i=1}^{m} h_i(c_i(x)).$$

We consider three possible stochastic one-sided models for approximating the functions f, h and c:

1. $f_x(y,\xi) = f(x) + \langle \nabla c(x,\xi)^T w(x,\xi), y - x \rangle$
2. $f_x(y,\xi) = h(c(x,\xi) + \nabla c(x,\xi)(y-x), \xi)$
3. $f_x(y,\xi) = h(c(y,\xi))$

where $w(x,\xi) \in \partial h(c(x,\xi),\xi)$ is a subgradient selection. Each iteration of the algorithm with model 1 reduces to the *stochastic proximal subgradient algorithm* (SPS), with model 2 to *stochastic prox-linear algorithm* (SPL) and with model 3 to *stochastic proximal-point algorithm* (SPP). Hence, the 3 algorithms share the same structure and update rule but differ in the way the stochastic model is specified, using different level of approximations of the composite function. We will see with in the application section that SPL and SPP provide a significantly better approximation quality than SPS. This is due to the fact that they are two-sided stochastic models which require solving an auxiliary subproblem.

Stochastic model-based algorithm

We now illustrate that the sample complexity $\mathbb{O}(\epsilon^{-4})$ found for the proximal stochastic subgradient method holds for a wider class of stochastic algorithms, including also stochastic proximal point and prox-linear algorithms. The general structure of stochastic model-based algorithms is the following. Denote the problem as

$$min_{x \in R^d} \ \varphi(x) := f(x) + r(x) \tag{4}$$

where $r : R^d \to R \cup \{\infty\}$ is a closed function and $f : R^d \to R$ is locally Lipschitz.
In this setting, we assume that the only access to f is possible through a stochastic one-sided model.
We fix a probability space (Ω, $\mathcal{F}$,P), equip R^d with a Borel σ-algebra and assume that there exist $\tau, \eta, L \in R$ such that the following assumptions hold:

- **(A1)** (*Sampling*) It is possible to generate i.i.d. $\xi_1, \xi_2 ... \sim P$;
- **(A2)** (*One-sided accuracy*) There is an open convex set U containing $domr$ and a measurable function $(x,y,\xi) \to f_x(y,\xi)$ defined on $U \times U \times \Omega$, which is an unbiased estimator of f and that lower bounds f up to a quadratic error, i.e. $\mathbb{E}_\xi[f_x(y,\xi) - f(y)] \leq \frac{\tau}{2}\|y-x\|^2 \ \forall x, y \in U$;
- **(A3)** (*Weak convexity*) The function $f_x(\cdot,\xi) + r(\cdot)$ is η-weakly convex $\forall x \in U$ and $\xi \in \Omega$;
- **(A4)** (*Lipschitz property*) There exists a measurable function $L : \Omega \to R_+$ satisfying $\sqrt{\mathbb{E}_\xi[L(\xi)^2]} \leq L$ and such that:

  $$f_x(x,\xi) - f_x(y,\xi) \leq L(\xi)\|x-y\|$$

  $\forall x, y \in U, \xi \sim P$.

For deriving the convergence rate of the algorithm we have to keep in mind that the objective function φ is itself $(\tau+\eta)$-weakly convex based on our assumptions.
The algorithms structure is then:

Input: $x_0 \in R^d$; real $\bar{\rho} \geq \tau + \eta$; a sequence of stepsizes $\{\beta_t\}_{t\geq 0} \subseteq (\bar{\rho}, \infty)$ and iteration count T.
Step $t = 0, ..., T$

$$\begin{cases} sample \xi_t \sim P \\ set x_{t+1} = \arg\min_x \{r(x) + f_x(x, \xi_t) + \frac{\beta_t}{2}\|x - x_t\|^2\} \end{cases}$$

Sample $t^* \in \{0, ..., T\}$ according to the probability distribution:

$$\mathbb{P}(t^* = t) \propto \frac{\bar{\rho} - \eta - \tau}{\beta_t * -\eta}$$

Return x_{t^*}.

Convergence rate
The convergence rate of the stochastic model-based algorithm depends on the fine tuning of the parameters. If we set:

- real $\bar{\rho} = 2(\rho + \eta)$
- costant $\beta_t = \hat{\rho} + \sqrt{\frac{2\rho L^2 (T+1}{\Delta}}$
- real $\Delta \geq \varphi_{1/\hat{\rho}} - min\varphi$

then the point x_t^* satisfies:

$$\mathbb{E}\|\nabla \varphi_{1/\bar{\rho}}\|^2 \leq \frac{4\bar{\rho}\Delta}{T+1} + 8L\sqrt{\frac{2\bar{\rho}\Delta}{T+1}}.$$

We now show that the convergence rate of general stochastic model-based algorithm holds for all the three algorithms above mentioned, which differ from each other in the assumptions made and in the stochastic model used to estimate f.

1. Stochastic proximal point algorithm
Assumptions:

1. It is possible to generate i.i.d. realizations $\xi_1, \xi_2 ... \sim P$;
2. There is an open convex set U containing $domr$ and a measurable function $(x, y, \xi) \rightarrow f_x(y, \xi)$ defined on $U \times U \times \Omega$ satisfying $\mathbb{E}_\xi[f_y(x, \xi)] = f(y)$ $\forall x, y \in U$;
3. Each function $r(\cdot) + f_x(\cdot, \xi)$ is ρ-weakly convex $\forall x \in U$ and $\xi \in \Omega$;

4. There exists a measurable function $L : \Omega \to R_+$ satisfying $\sqrt{\mathbb{E}_\xi[L(\xi)^2]} \leq L$ and such that:
$$f_x(x,\xi) - f_x(y,\xi) \leq L(\xi)\|x-y\|$$
$\forall x, y \in U, \xi \in \Omega$.

The one-sided model adopted by stochastic proximal point is $f_x(y,\xi)$.

2. Stochastic proximal subgradient algorithm
Assumptions:

1. It is possible to generate i.i.d. $\xi_1, \xi_2 ... \sim P$;
2. The function f is ρ_1-weakly convex and r is ρ_2-weakly convex for some $\rho_1, \rho_1 \geq 0$;
3. There is an open convex set U containing $dom(r)$ and a measurable mapping $G : U \times \omega \to \mathbb{R}^d$ satisfying $\mathbb{E}_\xi[G(x,\xi)] \in \partial f(x) \forall x \in U$;
4. There is a real $L \geq 0$ such that $\mathbb{E}_\xi\|G(x,\xi)\|^2 \leq L \forall x in U$.

The one-sided model adopted by stochastic proximal subgradient is the linear model:
$$f_x(x,\xi) = f(x) + \langle G(x,\xi), y - x\rangle.$$

3. Stochastic prox-linear algorithm
Consider the usual optimization problem with $f(x) = \mathbb{E}_\xi[h(c(x,\xi),\xi)]$. Assumptions: there exists an open convex set U contaning $domr$ such that the followings are true:

1. It is possible to generate i.i.d. $\xi_1, \xi_2 ... \sim P$
2. Th e assignments $h : R^m \times \Omega \to R$ and $c : U \times \Omega \to R^m$ are measurables;
3. The function r is ρ-weakly convex, and there exist square integrable functions‘$\ll$, γ, $M : \Omega \to R$ such that $\forall \xi \in \Omega$, the function $z \mapsto h(z,\xi)$ is convex and ‘$\mathbb{L}(\xi)$-Lipschitz, the map $x \mapsto c(x,\xi)$ is C^1-smooth with $\gamma(\xi)$-Lipschitz Jacobian, and the inequality $\|\nabla c(x,\xi) \leq M(\xi)\|$ holds for all $x \in U$ and for all $\xi in \Omega$

The one-sided model adopted by stochastic prox-linear algorithm is the convex model:
$$f_x(x,\xi) = h(c(x,\xi) + \nabla c(x,\xi)(y-x),\xi).$$

In the final sections the stochastic proximal algorithms are practically implemented and their performance is compared to incremental algorithms, which can be considered their deterministic counterparts.

7 Exponential Moving Average-type methods

Recent works have proposed a variety of accelerated versions of both *gradient descent* and *stochastic gradient descent*. We can say that these variations fall into three categories:

1. Momentum methods which carefully design the descent direction Δ_t
2. Adaptive learning rate methods which determine good learning rates α_t
3. Adaptive gradient methods that leverage both advantages mentioned in 1. and 2.

Adaptive gradient methods update the descent direction and the learning rate simultaneously using knowledge from the past, and hence enjoy dual advantages of momentum and adaptive learning rate methods. Algorithms of this family include RMSProp, Adam, AMSGrad, etc. More precisely, these adopt *exponential moving averages*, with decaying factors, of the past gradients to update the descent direction and learning rate. This is because algorithms such as AdaGrad, which accumulate all of the squared historical gradients, have shown to have significant improvement in empirical performance of gradient-based methods only when the gradients are sparse. Additionally, accumulating the squared gradients from beginning of training may result in premature or excessive decrease in effective learning.

7.1 Notation

Let $\mathbb{R}_+$ denote the set of non-negative real numbers and $\mathbb{R}^d$ denote the coordinate space of d dimensions. $\mathcal{J}_{++}^d$ denotes the set of all symmetric positive definite d x d; while the minimum and maximum eigenvalues of the matrix Q are denoted by $\lambda_{min}(Q)$ and $\lambda_{max}(Q)$, respectively. For any vectors $a, b \in R^d$, all operations are done element-wise. For any positive integers d and T, we set $[d] := \{1, ..., d\}$ and $(T] := \{0, ..., T\}$ the total number of iterations in an algorithm. Furthermore, for any vector $x_t \in R^d$, $(x_t)_i$ denotes its i^{th} coordinate where $i \in [d]$. We use $\|\cdot\|$, $\|\cdot\|_1$ and $\|\cdot\|_\infty$ to denote the ℓ_2-norm, ℓ_1-norm, and the infinity norm, respectively and nore that $\|\cdot\|_\infty \leq \|\cdot\| \leq \sqrt{d}\|\cdot\|_\infty$. We let $diag(x)$ denote the diagonal matrix with diagonal entries $x_1, ..., x_d$ and $\overrightarrow{\mathbf{1}}$ denote all-ones vector.

For any matrix $Q \in \mathcal{J}_{++}^d$ and any set $\mathcal{X}$,

$$\Pi_{\mathcal{X},Q}(x) = argmin_{y \in \mathcal{X}} \|Q^{1/2}(x - y)\|^2 \tag{5}$$

denoting the scaled Euclidean projection of a vector x onto $\mathcal{X}$.

Given two sequences $\{a_n\}$ and $\{b_n\}$, we write $a_n = O(b_n)$ if there exists a constant $0 < c < +\infty$ such that $a_n \leq cb_n$. The expectation conditioned on all the realizations $\xi_0, \xi_1, ..., \xi_{t-1} \sim P$ is denoted by $\mathbf{E_t}[\cdot]$. Some of the following definitions have already been defined but now are stated in a setting where points are linearly transformed by $Q : R^d \to R^d \in \mathcal{J}_{++}^d$.

Definition 6: *For a function $\varphi : \mathbf{R^d} \to \mathbf{R} \cup \{+\infty\}$ and a point $x \in \text{dom}\varphi$, we let*

$$\partial\varphi(x) = \{w \in \mathbf{R^d} \mid \varphi(y) \geq \varphi(x) + \langle w, \langle y - x\rangle + o(\|Q^{\frac{1}{2}}(y - x)\|),\ as\ y \to x\}$$

denote the Frechet subdifferential of φ at x. We set $\partial\varphi(x) = \emptyset$ if $x \notin \text{dom}\varphi$.

Definition 7: *A function φ is said to be (ρ, Q)-weakly convex if, for some $\rho \in \mathbf{R}$ and some $Q \in \mathcal{J}^d_{++}$, the function $x \mapsto \varphi(x) + \frac{\rho}{2}\|Q^{\frac{1}{2}}x\|^2$ is convex. In the case when $Q = I$, $\varphi(x)$ is called ρ-weakly convex, which we have discussed earlier in this paper.*

Lemma 2: Let $\varphi : \mathbf{R^d} \to \mathbf{R} \cup \{+\infty\}$ be a closed function. Then, the following are equivalent:

(i) φ is (ρ, Q)-weakly convex for some $\rho \in \mathbf{R}$ and some $Q \in \mathcal{J}^d_{++}$.

(ii) For all $x, y \in \mathbf{R^d}$ with $\overline{\omega} \in \partial\varphi(x)$,

$$\varphi(y) \geq \varphi(x) + \langle\overline{\omega}, y - x\rangle - \frac{\rho}{2}\|Q^{\frac{1}{2}}(y - x)\|^2. \tag{6}$$

(iii) The subdifferential map is hypomonotone. That is, for all $x, y \in \mathbf{R^d}$ with $\omega \in \partial\varphi(x)$ and $\overline{\omega} \in \partial\varphi(y)$,

$$\langle\omega - \overline{\omega}, x - y\rangle \geq -\rho\|Q^{\frac{1}{2}}(y - x)\|^2.$$

Definition 8: (scaled Moreau envelope and proximal mapping) *For a proper, lower semi-continuous function $\varphi : \mathbf{R^d} \to \mathbf{R} \cup \{+\infty\}$, a constant $\lambda > 0$ and $Q \in \mathcal{J}^d_{++}$, the scaled Moreau envelope and the scaled proximal mapping are defined as*

$$\varphi_{\lambda,Q}(x) := min_y\{\varphi(y) + \frac{1}{2\lambda}\|Q^{\frac{1}{2}}(x - y)\|^2\}, \tag{7}$$

$$prox_{\lambda\varphi,Q}(x) := argmin_y\{\varphi(y) + \frac{1}{2\lambda}\|Q^{\frac{1}{2}}(x - y)\|^2\}. \tag{8}$$

In the case when $Q = I$, (7) and (8) reduce to the (unscaled) Moreau envelope and proximal mapping discussed in previous sections. The set of minimizers of the scaled Moreau envelope coincides with the set of minimizers of the unscaled one.

Lemma 3: (properties of proximal mapping) *Let $\varphi : \mathbf{R^d} \to \mathbf{R} \cup \{+\infty\}$ be a (ρ, Q)-weakly convex function. Then, for all $\lambda \in (0, \rho^{-1})$, and $x \in \mathbf{R^d}$ the following hold:*

(i) $prox_{\lambda\varphi,Q}(x)$ exists and is unique.

(ii) $prox_{\lambda\varphi,Q}(x)$ is non-expansive. That is, for all $y \in \mathbf{R^d}$,

$$\|Q^{\frac{1}{2}}(prox_{\lambda\varphi,Q}(x) - prox_{\lambda\varphi,Q}(y))\|^2 \leq \|Q^{\frac{1}{2}}(x - y)\|^2.$$

subsectionAdaptive gradient methods Adaptive gradient methods have became favoured by Deep Learning practitioners. Unlike the Stochastic Gradient Descent which uses the same learning rate at each iteration in all dimensions, these kind of methods adaptively scale the learning rate.

To make things easier to interpret, the following algorithm (Algorithm 1) outlines the generic adaptive method setup:

Algorithm 2 Generic Adaptive Method Setup

Input: Initial point $x_0 \in \mathcal{F}$, number of iterations T, stepsize $\{\alpha_t\}_{t=0}^{T} > 0$, sequence of functions $\{\phi_t, \psi_t\}_{t=0}^{T}$
For $t = 0$ to T do
$g_t = \nabla f_t(x_t)$
$m_t = \phi_t(g_1, ..., g_t)$ and $\hat{v}_t = \psi_t(g_1, ..., g_t)$
$x_{t+1} = \Pi_{\mathcal{F}, \hat{V}^{\frac{1}{2}}}[x_t - \alpha_t \hat{V}_t^{-\frac{1}{2}} m_t]$

where $\mathcal{F} \in \mathbf{R}_+^{\mathbf{d}}$ is the feasible set of points, and m_t and $\hat{v}_t$ are averaging functions that relate to historical gradients. The algorithm is abstract because ϕ_t and ψ_t have not been specified. Here, $\phi_t : \mathcal{F} \to \mathbf{R}^{\mathbf{d}}$ and $\psi_t : \mathcal{F} \to \mathcal{J}_{++}^{d}$. Feasibility is maintained by projecting onto the set $\mathcal{F}$ via the update rule $x_{t+1} = \Pi_{\mathcal{F}, \hat{V}^{\frac{1}{2}}}[\hat{x}_{t+1}]$, where $\Pi_{\mathcal{F}, \hat{V}^{\frac{1}{2}}}$ denotes the projection of $x \in \mathbf{R}_+^{\mathbf{d}}$ onto the set $\mathcal{F}$. We are restricted to diagonal variants of adaptive methods, where $\hat{V}_t = diag(\hat{v}_t)$ and α_t is the stepsize. We refer to $\alpha_t \hat{V}_t^{-\frac{1}{2}}$ as the learning rate.

By setting

$$\phi_t(g_1, ..., g_t) = g_t \text{ and } \psi_t(g_1, ..., g_t) = \overrightarrow{1},$$

we see that the algorithm yields the standard SGD. As stated in previous analysis, a practical issue of SGD is that to ensure its convergence α_t has to decay to zero - leading to slower convergence. This is exactly why adaptive methods have became so popular. Adaptive methods need to update both the direction of the search and the learning rate by appropriately choosing averaging functions to ensure better convergence.

Accordant to what we mentioned previously, Adam and RMSProp both employ exponential moving average to discard history from extreme past so that more rapid convergence is ensured. In particular, Adam uses the following formulas in this setting:

$$m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t,$$
$$\hat{v}_t = \beta_2 \hat{v}_{t-1} + (1 - \beta_2) g_t,$$

where $\{\beta_{1,t}\}_{t=0}^{T}$, $\beta_2 \in [0, 1)$ and $m_{-1} = \hat{v}_{-1} = 0$. In the case when $\beta_{1,t} = 0$, Adam reduces to RMSProp.

Findings have shown that Adam possesses convergence concerns even in the convex setting. That is why AMSGrad has been proposed, as a correction of Adam, which uses the following update rules:

$$m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t),$$
$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t,$$
$$\hat{v}_t = max(\hat{v}_{t-1}, v_t),$$

where $\{\beta_{1,t}\}_{t=0}^{T}$, $\beta_2 \in [0, 1)$ and $m_{-1} = v_{-1} = \hat{v}_{-1} = 0$.

7.2 First-order EMA-type method (FEMA)

As stated, the convergence proof of Adam is problematic, hence the appearance of AMSGrad. But, it has been established that there is a problem in the convergence proof of AMSGrad. Explicitly, the problem lies in handling the hyper-parameters, i.e. treating them as equal while they most certainly are not. This is also the neglected issue in the convergence proof of Adam.

FEMA is a new model-based adaptive method proposed by P. Nazari et al, inspired by AMSGrad. This method scales down the gradient by the square roots of exponential moving average of past squared gradients. Opposed to AMSGrad, FEMA takes a larger step toward the optimal point and yet incorporates the intuition of slowly decaying the effect of past gradients on the learning rate. This algorithm, as previously analyzed, aims to solve composite problems of the form

$$min_{x \in \mathbf{R}^{\mathbf{d}}} \varphi(x) = f(x) + r(x), \tag{9}$$

where the following assumptions are satisfied:

Assumption A. *f is (ρ, Q)-weakly convex for some $\rho \in \mathbf{R}$ and $Q = diag(q)$ where $q \in \mathbf{R}^{\mathbf{d}}_{++}$; r is closed convex; and φ is bounded below over its domain, i.e. $\varphi^* = min_{x \in \mathbf{R}^{\mathbf{d}}} \varphi(x)$ is finite.*

Assumption B. *Access to f is through a stochastic subgradient first-order oracle as in the case of proximal stochastic subgradient method that satisfies the proceeding assumption (Assumption C).*

Assumption C. *Let $(\Omega, \mathcal{F}, P)$ denote a probability space and equip $\mathbf{R}^{\mathbf{d}}$ with the Borel σ-algebra. Then, it is possible to generate i.i.d. realizations $\xi_0, \xi_1, ... \sim P$. Furthermore, there is an open set U containing* dom$r$ *and a measurable mapping $G : U$ x $\Omega \to \mathbf{R}^{\mathbf{d}}$ satisfying*

$$\mathbf{E}_{\xi}[G(x, \xi)] \in \partial f(x), \text{ for all } x \in U.$$

The algorithm steps are shown in the following figure (Algorithm 3):

Algorithm 3 First-order EMA-type method (FEMA)

Input: Initial point$x_0 \in \text{dom}r$, number of iterations T, stepsize $\{\alpha_t\}_{t=0}^T > 0$, decay parameters $\{\beta_{1,t}\},\ \beta_2,\ \beta_3 \in [0,1)$ and a vector $q \in \mathbf{R}_+^{\mathbf{d}}$ satisfying Assumption A;

Initialize $m_{-1} = v_{-1} = 0$ and $\hat{v}_{-1} = q$;

For $t = 0$ to T do:
$g_t = G(x_t, \xi_t)$;
$m_t = \beta_{1,t} m_{t-1} + (1-\beta_{1,t}) g_t$;
$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$;
$\hat{v}_t = \beta_3 \hat{v}_{t-1} + (1-\beta_3) max(\hat{v}_{t-1}, v_t)$;
$x_{t+1} = prox_{\alpha_t r, \hat{V}^{\frac{1}{2}}}[x_t - \alpha_t \hat{V}^{-\frac{1}{2}} m_t]$;

Choose x_{t^*} from $\{x_t\}_{t=0}^T$ with probability $\mathbf{P}(t^* = t) = \alpha_t / \sum_{t=0}^T \alpha_t$.

Convergence guarantees have been established for solving the constrained (ρ, Q)-weakly convex for the case of a composite problem of outlined in *(9)*. For the results to hold, additional lemmas had to be provided [1]:

Lemma 4: *For $\overline{\rho} > \rho$, let*

$$\overline{x}_t = prox_{\varphi/\overline{\rho}, Q}(x_t) \quad \text{and} \quad \overline{\gamma}_t = \mathbf{E}_t[G(\overline{x}_t, \xi_t)] \in \partial f(\overline{x}_t)).$$

Then, under assumption A, it follows that

$$\overline{x}_t = prox_{\alpha_t r, Q}(\alpha_t \overline{\rho} x_t - \alpha_t Q^{-1} \overline{\gamma}_t + (1 - \alpha_t \overline{\rho}) \overline{x}_t).$$

Lemma 5: [4] *Let $0 \le \beta_{1,t} \le \beta_1,\ \ \{\beta_i\}_{i=1}^3 \in [0,1)$ and $\tau = \beta_1 / \sqrt{\beta_2} < 1$. Then, for m_t and $\hat{V}_t$ generated by Algorithm 2, it holds that*

$$\|\hat{V}_t^{-\frac{1}{4}} m_t\|^2 \le \frac{\sum_{k=0}^t \tau^{t-k} \|g_k\|_1}{(1-\beta_1)\sqrt{(1-\beta_2)(1-\beta_3)}}.$$

Theorem 1 (proximal FEMA): *With assumptions A-C satisfied, $\|G(x,\cdot)\|_\infty \le G_\infty$ and $\|x - y\|_\infty \le D_\infty$ for all $x, y \in dom(r)$. Moreover, for all $t \in [0, T)$, let*

$$0 \le \beta_{1,t} \le \beta_1,\ \ \{\beta_i\}_{i=1}^3 \in [0,1),\ \ \tau = \frac{\beta_1}{\sqrt{\beta_2}} < 1,\ \ \overline{\rho} \in (\rho, 2\rho],\ \ \text{and} \ \ 0 < \alpha_t \le \frac{1}{\overline{\rho}}.$$

Then, for x_{t^*} generated by Algorithm 2, it holds that

$$\mathbf{E}[\|\varphi_{1/\overline{\rho}, \hat{V}_{t^*}^{\frac{1}{2}}}(x_{t^*})\|^2] \le \frac{\hat{\rho}\Delta_\varphi + \overline{\rho}^2 (\sum_{t=0}^T \alpha_t^2 C_{1,t} + C_{2,T})}{(\overline{\rho} - \rho) \sum_{t=0}^T \alpha_t}$$

where

$$\Delta_\varphi = \varphi_{1/\overline{\rho}, \hat{V}_{-1}^{\frac{1}{2}}}(x_0) - \varphi^*,$$

$$C_{1,t} = \frac{2\sum_{k=0}^t \tau^{t-k} \mathbf{E}[\|g_k\|_1]}{(1-\beta_1)\sqrt{(1-\beta_2)(1-\beta_3)}} + \frac{d G_\infty^2}{\sqrt{(1-\beta_2)(1-\beta_3)}} + \frac{d G_\infty^2}{\lambda_{min}(\hat{V}_t^{\frac{1}{2}})}, \ \text{and}$$

$$C_{2,T} = \frac{D_\infty^2}{2} (\sum_{t=0}^T \beta_{1,t}^2 \mathbf{E}[\|\hat{v}_t^{\frac{1}{2}}\|_1] + \mathbf{E}[\|\hat{v}_T^{\frac{1}{2}}\|_1])$$

This shows that the convergence of FEMA depends on stepsize α_t and $\overline{\rho} \in \mathbf{R}$ in weakly convex settings. Although the assumption $\|G(x,\xi)\|_\infty \leq G_\infty$ is strong, it is crucial for analyzing adaptive sub-gradient methods in non-convex setting. We choose α_t and $\overline{\rho}$ the following way:

Corollary 1. *Under the conditions for proximal FEMA* (Theorem 1)*, for all $t \in T$, let the parameters be set to*

$$\beta_{1,t} = \beta_1 \pi^{t-1}, \;\; \pi \in (0,1), \;\; \overline{\rho} = 2\rho, \;\; \alpha_t = \frac{\alpha}{\sqrt{T+1}} \;\; \text{and} \;\; 0 < \alpha \leq \frac{1}{2\rho}.$$

Then, for x_{t*} we conclude that

$$\mathbf{E}[\|\nabla \varphi_{1/(2p),\hat{V}_{t^*}^{\frac{1}{2}}}(x_{t^*})\|^2] \leq O(\tfrac{d}{\sqrt{T}}).$$

The corollary above states that FEMA has an overall complexity of $O(\frac{d}{\sqrt{T}})$, but unlike previously discussed stochastic-based minimization of weakly convex functions, it uses adaptive learning rates to accelerate convergence and improve sparsity issues. Moreover, it does not require batching which in non-sparse setting might provide large gradients but only quite rarely, and while these large gradients are quite informative, their influence dies out rather quickly due to the exponential averaging, thus leading to poor convergence.

8 Concrete Examples

8.1 Robust Matrix Sensing (RMS)

Low-rank matrices have a very broad application in many areas such as Computer Vision and Machine Learning. A very important computational task is to recover a Positive Semidefinite low-rank matrix $X^* \in \mathbb{R}^{n \times n}$ with $rank(X^*) = r \leq n$ from a small number of linear measurements arbitrarily corrupted with outliers

$$y = \mathcal{A}(X^*) + s^*$$

where $\mathcal{A}$ is a linear measurement operator consisting of a set of sensing matrices $\mathcal{A}_1, ..., \mathcal{A}_m$ and s^* is a sparse outliers vector. An effective approach to recover the low-rank matrix X^* is by using a factored representation of the matrix variable

$$X = UU^T, U \in \mathbb{R}^{n \times r}$$

and employing a l_1-loss function to robustify the solution against outliers.

$$\min_{U \in \mathbb{R}^{n \times r}} \frac{1}{m} ||y - \mathcal{A}(UU^T)||_1 = \frac{1}{m} \sum_{i=1}^{m} |y_i - \langle \mathcal{A}_i, UU^T \rangle|.$$

It can be shown that if $\mathcal{A}$ has i.i.d. Gaussian ensembles, the fraction of outliers in s^* is less than 0.5 and $m \geq \mathcal{O}(nr)$, then the set of global minizers is:

$$\mathcal{U} = \{U^* R : R \in \mathbb{R}^{r \times r}, RR^T = I\}$$

and the objective function is sharp with parameter $\alpha = c \cdot \sigma_r(X^*)$ where $c > 0$ is a constant depending on the fraction of outliers in s^*.

8.2 Robust Phase Retrieval (RPR)

Robust Phase Retrieval aims to recover a signal $x^* \in \mathbb{R}$ from its magnitude-wise measurements which are arbitrarily corrupted with outliers, like in the RMS case:

$$b = |Ax^*|^2 + s^*$$

where the operator $|\cdot|$ means coordinate-wise taking modulus and then squaring. Here the matrix $A \in \mathbb{R}^{m \times n}$ is a measurement matrix. The problem of recovering both the sign and the magnitude information of x^* can be formulated as:

$$\min_{x \in \mathbb{R}^n} \frac{1}{m} \left|\left||Ax|^2 - b\right|\right|_1 = \frac{1}{m} \sum_{i=1}^{m} ||\langle a_i, x\rangle|^2 - b_i|$$

Given some strong assumptions, it can be proven that $\mathcal{D} = \{\pm x^*\}$ is exactly the set of optimizers to the objective function which is sharp with parameter $\alpha = c \cdot ||x^*||$, where c is a constant that, like in the RMS case depends on the fraction of outliers in s^*.

9 Experiments

We conduct a series of experiments on Robust Matrix Sensing and Robust Phase Retrieval to try and replcate the results in the Li et. al. paper. For simplicity, the following abbreviations will be used:

- GD: Full (sub)-Gradient Descent
- SGD: Stochastic (sub)-gradient descent
- SPL: Stochastic prox-linear algorithm
- IGD: Incremental (sub)-gradient descent
- IPP: Incremental proximal point algorithm
- IPL: Incremental prox-linear algorithm

9.1 Comparison between Stochastic and Incremental counterparts

In this subsection, we will do a comparison between the incremental and stochastic methods described and show the superior power of the incremental methods with respect to their stochastic counterparts. We will omit the results obtained for the proximal points methods since for the RMS and RPR problems, the update step for the IPL method has a closed form solution, which speeds up the algorithm substantially and in general both IPP and IPL perform the same on these tasks.

We set a constant number of iterations - 500, with the initial step-size varying from $1/m$ to $1/210$ and with the geometrical step-size decay covering the range from 0.65 to 0.95 with a constant step of 0.05. Like in [1], the appropriate method and the chosen hyper-parameters are considered successful if the distance is no more than 10^{-8} and

failed on the contrary. We used two kinds of plots, namely distance vs. iterations and a simple grid image plot to show whether the choice leads to convergence (labeled as white) or not (labeled as black).

Looking at Figure 1. we can draw several conclusions. Firstly, the incremental methods in the general case cover a broader area that can be chosen from in order for the algorithm to converge and produce a satisfying result. This not only makes them a more robust choice for the weakly convex problems, but also a better one due to the flexibility of the geometrical step-size decay ρ. The possibility of choosing a smaller ρ allows, as discussed in the previous section and the results obtained about the correlation between convergence and this parameter, enables an almost certain faster convergence rate.

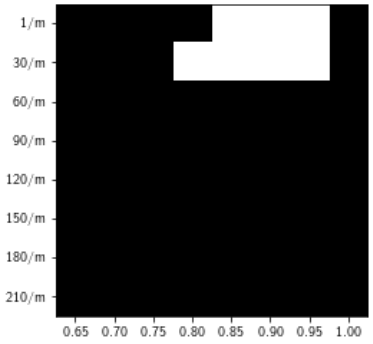
Moreover, (e) and (f) show a sample convergence plots of the algorithms and an additional sampling technique for the component function called "random shuffling" that chooses an independent and uniform permutation of the set $\{1, \cdots, m\}$ and does the inner update by following that sequence. From the experiments, we concluded that this yields results comparable to the incremental methods but usually the lowest rho value is bigger than the one obtained for its incremental counterpart.

9.2 IGD versus GD

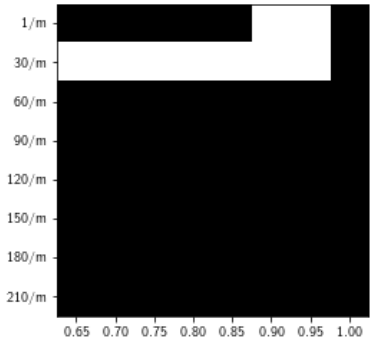
Moving on, we will now compare IGD with GD. We will also use the geometrically diminishing step-size rule for GD. Looking at Figure 3. It can clearly be seen that a the IGD has a much wider choice of ρ. Specifically, for RPR application, the smallest ρ that can be chosen is somewhere in the $[0.90, 0.95]$ range, whereas for the IGD can set values as low as 0.65. This figure implies that IGD outperforms GD by a large margin, since it can choose a much smaller ρ which indicates faster convergence.

9.3 Visualizing the effect of ρ

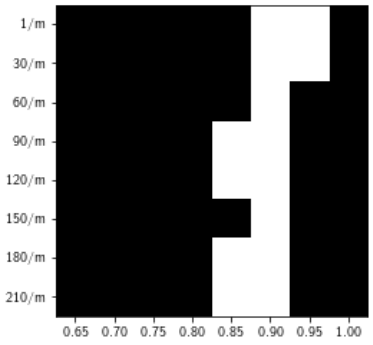
Having a solid mathematical understanding of these algorithms is not always sufficient to truly understand the effectiveness of correctly choosing the geometrically diminishing step-size. We can observe this by looking at Figure 2. These plots show us that when a sufficiently small ρ makes the algorithm converge, it makes it converge very fast. Figure 2. shows the performance of incremental algorithms using different hyper-parameter combinations and problems and we observe the same result, when rho is perfectly tuned, we obtain convergence with a little over hundred iterations.



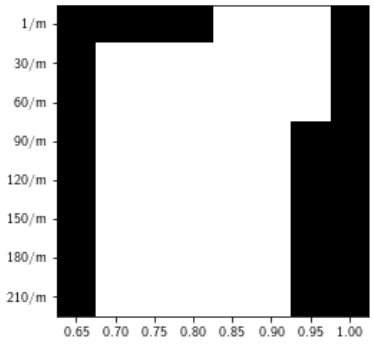
(a) SGD Convergence on RPR



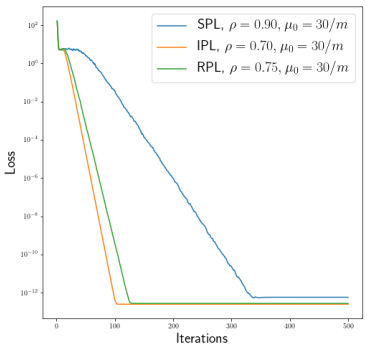
(b) IGD Convergence on RPR



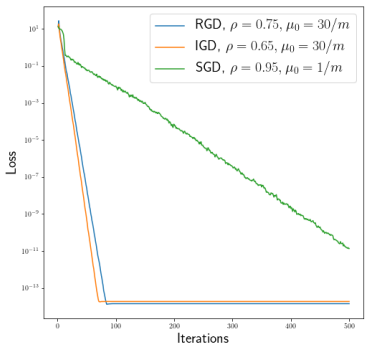
(c) SPL Convergence on RMS



(d) IPL Convergence on RMS

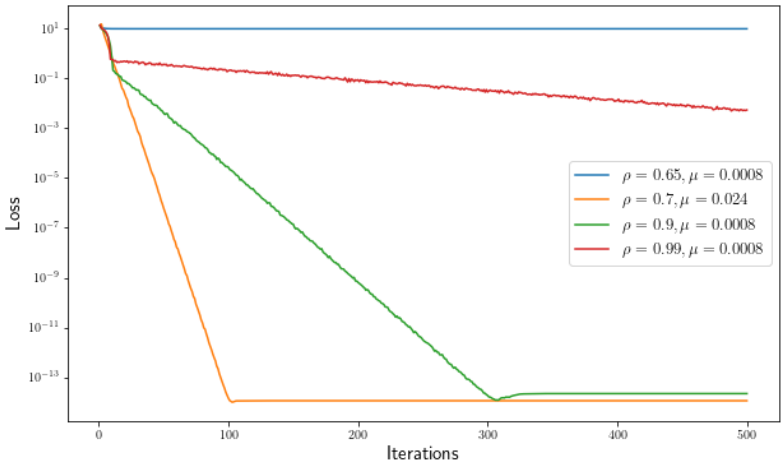


(e) RMS: Sampling Methods

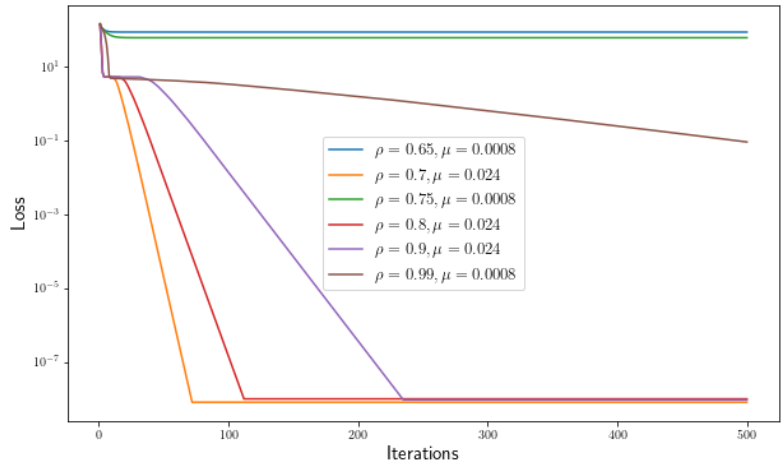


(f) RPR: Sampling Methods

Figure 2: Comparisons of robustness of incremental methods and their stochastic counterparts for both RMS and RPR applications

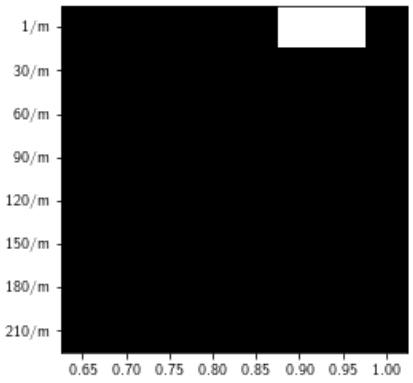


(a) RPR: IGD convergence plot

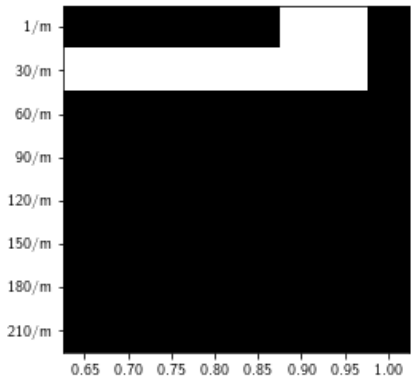


(b) RMS: IPL convergence plot

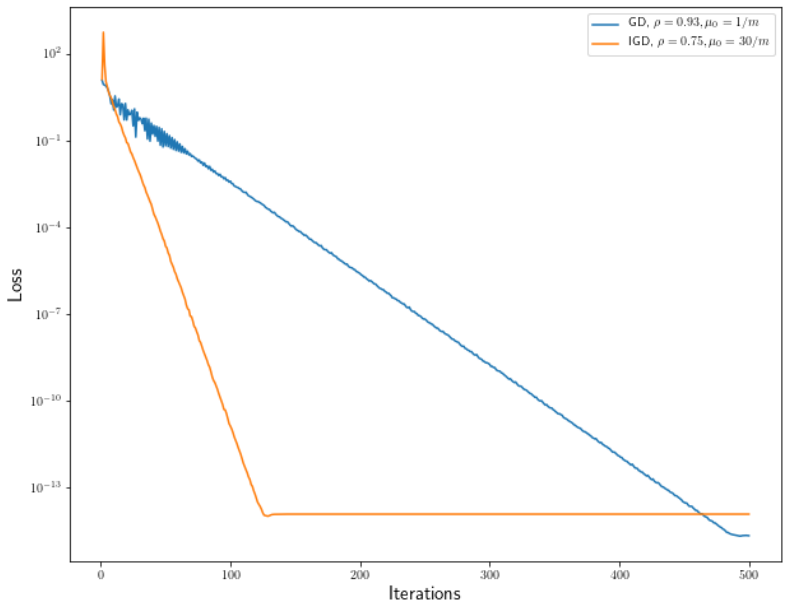
Figure 3: Convergence plots for several choices of algorithm parameter pairs ρ, μ_0 in Figure 1. where the top figure corresponds to Figure 1 (b) and the bottom figure corresponds to Figure 1 (d).



(a) RPR: GD convergence plot



(b) RPR: IGD convergence plot



(c) RMS: IPL convergence plot

Figure 4: GD and IGD comparison