Analysis of an Idealized Stochastic Polyak Method and its Application to Black-Box Model Distillation

Anonymous Authors[1]

Abstract

We provide a general convergence theorem of an idealized stochastic Polyak step size called SPS*. Besides convexity, we only assume a local expected gradient bound, that includes locally smooth and locally Lipschitz losses as special cases. We refer to SPS* as idealized because it requires access to the loss for every training batch evaluated at a solution. It is also ideal, in that it achieves the optimal lower bound for globally Lipschitz function, and is the first Polyak step size to have a $\mathcal{O}(1/\sqrt{t})$ anytime convergence in the smooth setting. We show how to combine SPS* with momentum to achieve the same favorable rates for the last iterate. We conclude with several experiments to validate our theory, and a more practical setting showing how we can distill a teacher GPT-2 model into a smaller student model without any hyperparameter tuning.

1. Introduction

Consider the problem

$$x_* \in \underset{x\in\mathbb{R}^d}{\operatorname{argmin}}\, f(x), \quad f(x) := \mathbb{E}_{\xi\sim\mathcal{P}}\left[f_\xi(x)\right], \tag{1}$$

where $\mathcal{P}$ is the distribution over data, and we assume there exists a minimizer $x_* \in \mathbb{R}^d$. We refer to $f_\xi(x)$ as the *loss* function over the data $\xi \sim \mathcal{P}$ under parameters $x \in \mathbb{R}^d$.

One of the main costs in developing new machine learning models is training them, that is, finding an approximate solution to (1). The training of GPT-4 is estimated to have cost over \$40M (Cottier et al., 2024). The elevated cost of training bigger models, and the success of Adam (Kingma & Ba, 2015), has sparked an intense research effort into developing new stochastic optimization methods. Yet the performance difference among many newly developed methods is minimal when the step size is tuned (Schmidt et al., 2021). Finding a good step size often involves multiple re-runs on a subset of the data, which adds considerably to this cost.

Here we advance the theory of an adaptive stochastic Polyak step size. The Polyak step size uses both the current loss and gradient norm to compute a step size at each iteration.

We show that if we had access to $f_\xi(x_*)$, the value of the loss at the solution for each batch ξ of data, a variant of the stochastic Polyak step we call SPS* achieves the best known rates across several subclasses of convex functions. Specifically, we show that SPS* achieves either the optimal rate when known, or the best known rate, for convex functions, including Lipschitz, smooth, and strongly convex. Furthermore we only require that these assumptions hold in a ball around the solution. This mirrors the same result in the deterministic setting for the Polyak step size (Hazan & Kakade, 2019).

We also prove convergence in the finite-sum, convex and continuous setting, without any additional assumption, for which we are unaware of any other stochastic method that provably converges.

We then show how to combine this Polyak step size with momentum, in such a way that the last-iterate converges at the optimal (competitive) rate in the Lipschitz (smooth) setting. For this we use *iterate averaging*, which is one of the many equivalent ways of writing momentum (Sebbouh et al., 2021).

These fast and adaptive convergence results speak to the strength of the SPS* method. However, they also show that having access to $f_\xi(x_*)$ for every ξ is a strong assumption, which we can not expect to hold in general. But we do consider two settings where $f_\xi(x_*)$ is known or can be approximated. The first setting is that of interpolation, where typically $f_\xi(x_*) = 0$ or is relatively easy to compute (Loizou et al., 2021). The second setting is one we call *blackbox model distillation*. In this setting, we can query a teacher (a larger pretrained model) with any input, but we do not have access to the teachers architecture or weights. Our objective is to train the student (a smaller model) on one

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

of the tasks that the teacher is accomplished. The teacher's loss on each input serves as an approximation of $f_\xi(x_*)$ for the student. This enables us to use SPS* with momentum to set the step size for the student, and train it efficiently without having to tune any hyper-parameters.

1.1. Stochastic Polyak Step Size

Here we analyse the following variant of the SPS (Stochastic Polyak step size) method

$$x_{t+1} = x_t - \gamma_t^{\texttt{SPS*}} g_t, \quad \gamma_t^{\texttt{SPS*}} := \frac{(f_t(x_t) - f_t(x_*))_+}{\|g_t\|^2} \tag{2}$$

where $\xi_t \sim \mathcal{P}$ is sampled i.i.d at each iteration, and g_t denotes either a gradient (smooth setting) or a subgradient (non-smooth setting) of $f_t := f_{\xi_t}$ evaluated at x_t. Throughout, we use the notation $(z)_+ := \max\{z, 0\}$ for $z \in \mathbb{R}$. We refer to (2) as a the SPS* method. We will prove several anytime convergence rates for SPS*. By *anytime*, we mean a proof that the method converges to any predefined tolerance without prior knowledge of that tolerance.

See Table 1 for a comparison between our rates of convergence, that of other variants of SPS, and the best known anytime rates for SGD in each setting. For the SGD rates within each setting, we included rates that rely on the global problem constants. For instance, to achieve the $GD/\sqrt{t}$ rate in the G-Lipschitz setting, we need to set the step size as $\gamma = \frac{D}{G}\frac{1}{\sqrt{t}}$, and we need to project the iterates of SGD back onto the ball of radius $D := \|x_0 - x_*\|$. In contrast, SPS* achieves this rate without without access to G or D, but with access to $f_\xi(x_*)$ instead.

The main downside to (2) is that it requires access to $f_t(x_*)$. This is why we refer to SPS* as an idealized variant, both because of its ideal convergence rates, and this idealized setting of assuming access to $f_t(x_*)$. In this sense, the comparisons in Table 1 to alternative Polyak type methods are not entirely fair, because they do not require such access to $f_t(x_*)$. Our message here is not that SPS* is a better method than $\texttt{SPS}_{\max}$, NGN or DecSPS, but rather that $f_t(x_*)$ is the object that we should try to approximate, or learn on the fly.

Despite our claim that SPS* is an idealized method, we do consider two settings where access to, or approximating, $f_t(x_*)$ is reasonable. One setting where $f_t(x_*)$ is often known is the interpolation setting, where we assume that there exists a minimizer $x_* \in \mathbb{R}^d$ such that the loss over every data is simultaneously minimized, in other words

$$f_\xi(x_*) = \inf_{x\in\mathbb{R}^d} f_\xi(x), \quad \forall \xi \in \text{support}(\mathcal{P}). \tag{3}$$

Thus under interpolation, our model has a perfect fit (as measured by $f_\xi(x)$) for every data point. Typically the loss is a non-negative function and its infimum is zero (Loizou et al., 2021), that is $\inf_{x\in\mathbb{R}^d} f_\xi(x) = 0$. When this is the case, we have access to every $f_\xi(x_*)$, which happens to be zero. Alternatively when $\inf f_\xi(x)$ is close to zero, then using zero as approximation is reasonable. Finally, even when $\inf f_\xi(x)$ is far from zero, it can sometimes be efficiently approximated (Loizou et al., 2021).

The ease of approximating $\inf f_\xi(x)$ is what motivated $\texttt{SPS}_{\max}$ (Loizou et al., 2021) which uses the step size

$$\gamma_t^{\texttt{SPS}_{\max}} := \min\left\{\frac{f_t(x_t) - \inf_x f_t(x)}{\|g_t\|^2}, \gamma_b\right\}, \tag{4}$$

where $\gamma_b > 0$ is an additional hyperparameter to safe-guard against excessively large step sizes. Loizou et al. (2021) present a comprehensive analysis of $\texttt{SPS}_{\max}$ in the non-smooth, smooth and strongly convex setting. But in all these cases, $\texttt{SPS}_{\max}$ is only guaranteed to converge when interpolation holds. Outside of interpolation, $\texttt{SPS}_{\max}$ converges to a neighborhood of the solution. Here we show that it is not necessary to assume that interpolation holds to establish convergence of a SPS type method. Having access to $f_\xi(x_*)$ is sufficient.

To be clear, assuming access to $f_\xi(x_*)$ is not the same as assuming that interpolation holds. Interpolation (3) imposes constraints on the data and the model, usually requiring the model to be overparameterized (Ma et al., 2018; Liu et al., 2022; Gower et al., 2021). In contrast having access to $f_\xi(x_*)$ imposes no constraints on the model and data. Furthermore, there are settings outside of interpolation where $f_\xi(x_*)$ can be known or reasonably approximated, such as model distillation which we consider in Section 4.1.

As a secondary objective of our work, we also present IAM (Iterate Averaging Adaptive method), a variant of SPS* with momentum. We prove that in the smooth and Lipschitz setting the *last* iterate of IAM converges as fast as the *average* iterate of SPS*. As the last iterate is usually more relevant in practice, this is the first time that a version of SPS with momentum has some theoretical advantage.

Next we describe the related work to ours, and use the context to detail our specific contributions. See Table 2 for a high-level resume of our results.

1.2. Related Work and Contributions

Polyak step size. The Polyak step size was first introduced by Polyak (1987) in the deterministic setting, where he also proved convergence for non-smooth and convex functions. Hazan & Kakade (2019) revisited the Polyak step size and showed that for the class of gradient descent methods (where we can only choose the step size), it has the optimal convergence rate in the Lipschitz, smooth, and strongly convex setting. Furthermore, it is optimal without having access to any of the Lipschitz (G), smoothness (L),

Table 1. A summary of anytime convergence rates for variants of stochastic Polyak step size. Notation: $D = \|x_0 - x_*\|$, $\sigma_*^2 = f(x_*) - \mathbb{E}[\inf f_\xi]$, $\sigma_{pos} = \mathbb{E}[\inf f_\xi]$, $G^2 = \max_x \mathbb{E}_\xi\left[\|\nabla f_\xi(x)\|^2\right]$. We compare to the stochastic Polyak methods DecSPS[(3)], SPSmax (Loizou et al., 2021) and NGN (Orvieto & Xiao, 2024). The proof of convergence for SPS* in the Lipschitz convex and strongly convex setting was first given in (Garrigos & Gower, 2023) and (Pedregosa & Schaipp, 2023), respectively.

Algorithm	**Convex finite sum**	**G-Lipschitz problems**	**L-Smooth problems**	**L-Smooth μ-Convex**	G-Lipschitz **μ-Convex**
DecSPS[(3)]	✗	✗	✗	$\frac{LD^2+\sigma_*^2}{\sqrt{t}}$	✗
SPSmax	✗	✗	$\frac{LD^2}{t} + \sigma_*^2 L$	$\left(1-\frac{\mu}{L}\right)^t D^2 + \frac{\sigma_*^2 L}{\mu}$	✗
NGN	✗	✗	$\frac{L^2D^2}{\sqrt{t}} + \frac{L(\sigma_*^2 + L\sigma_{pos})\log(t)}{\sqrt{t}}$	✓[(4)]	✗
SGD*[(2)]	✗	$\frac{GD}{\sqrt{t}}$	$\frac{LD^2}{\sqrt{t}} + \frac{\sigma_*^2 \log(t)}{L\sqrt{t}}$	$\frac{\sigma_*^2}{\mu^2}\frac{1}{t} + \frac{L^2D^2}{\mu^2 t^2}$	$\frac{B^2}{\mu^2}\frac{1}{t}$
SPS*	$\frac{GD}{\sqrt{t}}$[(1)] Remark 2.4	$\frac{GD}{\sqrt{t}}$ Corollary 2.2	$\frac{LD^2}{t} + \frac{\sigma_*^2 D}{\sqrt{t}}$ Corollary 2.3	$\frac{\sigma_*^2}{\mu^2}\frac{1}{t}$ Theorem G.1	$\frac{B^2}{\mu^2}\frac{1}{t}$ Theorem G.1
IAM (new)	$\frac{GD}{\sqrt{t}}$[(1)] Remark 2.4	$\frac{GD}{\sqrt{t}}$ Theorem 3.2	$\frac{LD^2\log(t+1)}{t} + \frac{\sqrt{L\sigma_*^2}D}{\sqrt{t}}$ Theorem 3.3	✗	✗

[(1)] The convex finite sum result assumes $\mathbb{E}_\xi[f_\xi] = \frac{1}{n}\sum_{i=1}^n f_i$ and f_i is continuous for $i = 1, \ldots, n$.

[(2)] SGD* denotes SGD where we can use all the global constants D, G, L, σ_*^2 and μ to set the step size. For the left to right, these results can be found in Thm. 9.12 (Garrigos & Gower, 2023), Thm. 4.1 (Gower et al., 2021), Thm. 3.1 (Gower et al., 2019), Section 3.2 (Lacoste-Julien et al., 2012).

[(3)] Under the additional assumption that the iterates of DecSPS are bounded, we have from (Orvieto et al., 2022) that DecSPS converges at a $\mathcal{O}\left(1/\sqrt{t}\right)$ rate in the G-Lipschitz and L-smooth setting.

[(4)] The paper claims an $\mathcal{O}\left(\log(t)/t\right)$ anytime rate is possible, but does not give the explicit proof or constants.

or strong convexity (μ) parameters. Recently, the proof of convergence in the smooth setting has been generalized to a broader class of relatively smooth functions (Takezawa et al., 2024) and locally smooth functions (Richtárik et al., 2024). In the smooth and strongly convex setting, Barré et al. (2020) show how to accelerate gradient descent with the Polyak step size, and without having access to the strong convexity parameter, but estimating it instead.

The stochastic Polyak step size. The current research into the stochastic Polyak step size was kick-started by the ALI-G method (Berrada et al., 2020) and $\texttt{SPS}_{\max}$ (Loizou et al., 2021). Both ALI-G and $\texttt{SPS}_{\max}$ offered a practical stochastic variant of the Polyak step size with strong empirical results to support their use. In terms of convergence theory, for smooth and convex functions, $\texttt{SPS}_{\max}$ was shown to converge to a neighborhood of the solution (Loizou et al., 2021). To enforce that $\texttt{SPS}_{\max}$ does converge in the smooth setting, Orvieto et al. (2022) proposed the DecSPS method that combines $\texttt{SPS}_{\max}$ with a decreasing step size sequence, and show that if the stochastic loss functions are strongly convex and smooth, then suboptimality converges at a rate of $\mathcal{O}(1/\sqrt{T})$, where T is the number of iterations. This rate is slower than SGD in the same setting, which is $\mathcal{O}\left(1/T\right)$.

As for SPS*, Garrigos et al. (2023) showed that it converges with the optimal rate in the Lipschitz non-smooth setting. Convergence in the smooth setting was shown in (Garrigos et al., 2023; Gower et al., 2021), but under interpolation.

A proximal version of SPS was introduced in (Schaipp et al., 2023) in order to handle regularization terms. More recently, a new variant of SPS called NGN was introduced in (Orvieto & Xiao, 2024) for specifically non-negative functions. NGN uses a combination of Gauss-Newton and truncation to introduce a dampened version of the Polyak step sizes. Though NGN also converges to a neighborhood of the solution for smooth functions, Orvieto & Xiao (2024) prove a $\mathcal{O}(1/\sqrt{T})$ and $\mathcal{O}\left(1/T\right)$ complexity for convex and strongly convex functions, respectively. Orvieto & Xiao (2024) also give a $\mathcal{O}(\log(T)/\sqrt{T})$ anytime result in the smooth and convex setting.

Contributions. We present a unifying anytime convergence in the smooth and non-smooth setting in Theorem 2.1 for SPS*. Besides convexity, Theorem 2.1 only makes local assumptions and thus applies to a broader class of functions as compared to prior results. We then specialize this result into the locally Lipschitz and locally smooth setting in Corollary 2.2 and Corollary 2.3, respectively. Our proof also leverages a new trick, where we explicitly invert a convex monotone function (Lemma C.1). We show how this trick is used in a sketch of the proof of Theorem 2.1 in Section D.1. Finally, our convergence result in the smooth setting in Corollary 2.3 is the first $\mathcal{O}(1/\sqrt{T})$ anytime result. Furthermore, this convergence result is adaptive to interpolation: As we get closer interpolation, σ_*^2 approaches zero, and the convergence rate in Corollary 2.3 automatically switches from $\mathcal{O}(1/\sqrt{T})$ to $\mathcal{O}(1/T)$.

Momentum. Polyak (1964) introduced the momentum method through the heavy-ball viewpoint. In the deterministic setting, Polyak (1964) showed that it converges at an accelerated rate for strongly convex quadratic functions. Only rather recently, a global convergence was established for smooth and non-smooth functions without strong convexity (Ghadimi et al., 2015).

In the stochastic setting, there is little to no theoretical advantage for using momentum for `SGD`, unless we consider the specialized setting of minimizing a quadratic (Lee et al., 2024; Bollapragada et al., 2024). The main theoretical improvement from using momentum in the stochastic setting for general convex functions is that the last iterate x_t of momentum converges at the same favourable rate as the average iterate of the `SGD` iterates (Sebbouh et al., 2021; Defazio & Gower, 2021). The analysis in (Sebbouh et al., 2021) relies on an equivalent reformulation of momentum known as the *iterate averaging* viewpoint, which we also use in this work. Recent online-to-batch conversion techniques can also achieve the same rate of convergence of the last iterate of `SGD` without momentum, albeit with slightly worse constants (Cutkosky, 2019b). These online-to-batch techniques rely on monotonic step sizes, and thus are not applicable to Polyak-type step sizes.

Stochastic Polyak with momentum. In the stochastic setting, some very recent works have considered different ways of blending `SPS` with momentum (Schaipp et al., 2024; Wang et al., 2023). The first analysis of a variant of `SPS` with momentum was developed in Wang et al. (2023). Their `ALR-SMAG` method is the result of choosing a learning rate that minimizes a particular upper bound on $\|x_{t+1} - x_*\|$ for the iterates of momentum or heavy-ball. The current analysis for `ALR-SMAG` shows that it has a slower convergence as compared to `SPS` unless $\beta_t = 0$, which corresponds to using no momentum. The same issue holds for the recently introduced `MoMo` method (Schaipp et al., 2024), which empirically reduces the tuning effort for the learning rate across many tasks, but theoretically has best bounds with no momentum, that is, when the method is equal to `SPS`. Another recent approach that combines `SPS` with momentum is proposed by Oikonomou & Loizou (2024), introducing `MomSPSmax` and its variants, `MomDecSPS` and `MomAdaSPS`. These step sizes guarantee convergence in the stochastic setting without relying on the interpolation condition. Instead, they assume in addition that the iterates remain bounded. Specifically, `MomSPSmax` achieves an $\mathcal{O}(1/t)$ convergence rate to a neighborhood of the solution, while `MomDecSPS` and `MomAdaSPS` converge to the exact solution with a rate of $\mathcal{O}(1/\sqrt{t})$.

Contributions. We prove that the last iterate of our momentum variant of `SPS` (Algorithm 1) converges anytime in (i) the convex and *locally* Lipschitz case (see Theorem 3.2) and (ii) the *locally* smooth case (see Theorem 3.3). Furthermore, in the non-smooth setting, the convergence rate in Theorem 3.2 is *at least as fast* as the corresponding rate for `SPS*` in Corollary 2.2.

Adaptive methods. Historically, line search procedures, such as Armijo line search (Armijo, 1966), used to be commonly employed to estimate the smoothness around the current point when the exact smoothness constant L was not known. More recent works have shown that it is also possible to estimate the value of L using the previously observed gradients (Malitsky & Mishchenko, 2020; Latafat et al., 2024). Furthermore, in the last decade, more line-search (Nesterov, 2014) and bisection (Carmon & Hinder, 2022) methods have been proposed that adapt simultaneously to smooth and non-smooth objectives. Unfortunately, most of the approaches either don't have strong guarantees in the stochastic case or require large batch sizes.

In online learning, when the Lipschitz constant of the objective is known, coin-betting approaches (Orabona & Pál, 2016) can be used to adaptively estimate distances to a solution. When the Lipschitz constant is not known, one can either use restarts (Mhammedi & Koolen, 2020), which require a lot of extra work, or use a technique called hints (Cutkosky, 2019a), but the latter introduces even more hyperparameters.

`AdaGrad` (Streeter & McMahan, 2012; Duchi et al., 2011) and its variants offer an alternative by estimating the gradient magnitudes instead of estimating smoothness. These methods can be combined with momentum and achieve strong complexity results, but they either require bounded domain (Levy et al., 2018; Kavis et al., 2019) or are only studied in the deterministic setting (Li & Lan, 2023). Furthermore, most variants use step sizes that can only decrease over time, meaning they will not adapt if the problem curvature becomes flatter. This has been partially addressed by a series of new methods that have an increasing estimate of distances to the solution set (Defazio & Mishchenko, 2023; Ivgi et al., 2023; Khaled et al., 2023), but their stochastic guarantees are provided only for large batch sizes (Ivgi et al.,

2023) or the interpolation setting. We compare to the most relevant of these works in Table 2.

Contributions. Our theoretical results show that the SPS* method is adaptive to the following settings and parameters: smoothness (L), initial distance (D), Lipschitz (G), interpolation (σ_*^2) and strong convexity (μ). The precise definition of these parameters and constants are given later.

2. Stochastic Polyak Step Size

Before giving our convergence proofs, we first will motivate SPS* as the step size that minimizes an upper bound on the distance to a minimizer. Suppose we are at iteration t, have drawn a batch of data ξ_t, and let $g_t := g_{\xi_t}(x_t)$ be the stochastic (sub)gradient evaluated at x_t. For short-hand we will also use $f_t := f_{\xi_t}$. Consider an iterate of SGD,

$$x_{t+1} = x_t - \gamma_t g_t,$$

where $\gamma_t > 0$ is the step size. The subgradient $g_t \in \partial f_t(x_t)$, by definition satisfies

$$f_t(x) \geq f_t(x_t) + \langle g_t, x - x_t \rangle, \quad \forall x \in \mathbb{R}^d. \tag{5}$$

Now consider the task of choosing γ_t that brings x_{t+1} as close as possible to the solution x_*. In general, this is impossible since we do not know x_*. However, we can minimize the upper bound

$$\begin{aligned} &\|x_{t+1} - x_*\|^2 - \|x_t - x_*\|^2 \\ &\quad = -2\gamma_t \langle g_t, x_t - x_* \rangle + \gamma_t^2 \|g_t\|^2 \\ &\quad \leq -2\gamma_t (f_t(x_t) - f_t(x_*)) + \gamma_t^2 \|g_t\|^2, \end{aligned} \tag{6}$$

where we use (5) in the inequality. Minimizing the right-hand side under the constraint $\gamma_t \geq 0$ gives the step size

$$\gamma_t^{\mathrm{SPS*}} = \frac{(f_t(x_t) - f_t(x_*))_+}{\|g_t\|^2}, \tag{7}$$

which together with SGD gives the SPS* method (2).

Note that in (7) we divide by the squared norm of the stochastic gradient, which could be equal to zero. This is only possible if $(f_t(x_t) - f_t(x_*))_+ = 0$, so we define $\gamma_t^{\mathrm{SPS*}} := 0$ if $g_t = 0$. That is, if the stochastic gradient is zero, no step is taken.

2.1. Convergence Theory for Convex Problems

We now give our unifying convergence theorem for SPS*, that aside from convexity, only assumes in (9) that the expected norm of the stochastic gradients is bounded within

$$\mathbb{B}_D(x_*) := \{x \in \mathbb{R}^d : \|x - x_*\| \leq \|x_0 - x_*\|\}.$$

Later we show how this local bound can be specialized into a smooth and non-smooth setting. In Appendix G we give an analogous result for the strongly convex setting.

Theorem 2.1. [Convergence of SPS*] Consider problem (1) and let the iterates $(x_t)_{t \geq 0}$ be given by (2), and let $D := \|x_0 - x_*\|$. If $f_\xi : \mathbb{R}^d \to \mathbb{R}$ is convex with probability one, then the iterates are almost surely monotone:

$$\|x_{t+1} - x_*\|^2 \leq \|x_t - x_*\|^2 \text{ with probability } 1. \tag{8}$$

If there exist $A, B > 0$ such that for all $x \in \mathbb{B}_D(x_*)$

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq A(f(x) - f(x_*)) + B, \tag{9}$$

then for $\bar{x}_T := \frac{1}{T}\sum_{t=0}^{T-1} x_t$ we have that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{D^2 A}{T} + \sqrt{\frac{D^2 B}{T}}, \quad \forall T \in \mathbb{N}. \tag{10}$$

Because our proof makes use of a new technical lemma that may find uses elsewhere, we give a sketch of the proof in Appendix D.2. The full proof is also in Appendix D.2.

Next we specialize Theorem 2.1 to a non-smooth and smooth setting, as we show in the next two corollaries.

Corollary 2.2 (Non-smooth setting)**.** Consider the setting of Theorem 2.1 where $A = 0$ and $B = G \geq 0$. In other words, the following *expected locally Lipschitz* assumption holds:

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq G^2, \quad \forall x \in \mathbb{B}_D(x_*). \tag{11}$$

It follows that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{GD}{\sqrt{T}}, \quad \forall T \in \mathbb{N}. \tag{12}$$

Corollary 2.3 (Smooth setting)**.** Consider the setting of Theorem 2.1 where $A = 2L$ and $B = \sigma_*^2 := \inf f - \mathbb{E}_\xi[\inf f_\xi]$. That is, we assume local *expected smoothness*:

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq 2L\big(f(x) - \inf f + \sigma_*^2\big), \ \forall x \in \mathbb{B}_D(x_*). \tag{13}$$

It then follows that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{4L\|x_0 - x_*\|^2}{T} + \frac{\sqrt{2}\|x_0 - x_*\|\sigma_*^2}{\sqrt{T}}. \tag{14}$$

For the non-smooth setting, it is typically assumed that the loss functions are *globally* Lipschitz, uniformly with respect to ξ, which in turn gives a global bound on the stochastic subgradients. Here instead we require very little: the convexity of our losses entails that f_ξ is G_ξ-Lipschitz on $\mathbb{B}_D(x^*)$ (see Corollary 8.41 in (Bauschke & Combettes, 2017)), so we only need to assume that the expectation $\mathbb{E}_\xi[G_\xi]$ is finite. An advantage of this local Lipschitz assumption is that it always holds true for finite sums (just take the maximum over G_ξ). Another advantage of our local assumption is that it is compatible with strong convexity. Indeed there is no function which is both globally Lipschitz and strongly convex, see e.g. Lemma 9.13 in (Garrigos & Gower, 2023).

Despite this additional generality, we achieve a $\mathcal{O}(1/\sqrt{T})$ convergence rate which is the optimal lower bound for the class of convex Lipschitz functions (Drori & Teboulle, 2016). Currently this rate can only be achieved by combining adaptive methods such as AdaGrad (Duchi et al., 2011) together with knowing and using $\|x_0 - x_*\|$ to set the learning rate or a projection radius (Orabona, 2019). In contrast, our oracle requires knowing $f_\xi(x_*)$. We note that a weaker version of Corollary 2.2 was first established in (Garrigos et al., 2023, Thm. 2.3), where the losses f_ξ are assumed to be globally Lipschitz.

As for the smooth setting, it is typically assumed in the literature that the loss functions f_ξ are *globally* smooth, uniformly with respect to ξ (Gower et al., 2020; 2021; 2019). This assumption is a sufficient condition for our inequality (14) to be true, see Garrigos & Gower (2023, Lem. 4.19). Our result instead requires much less: all we need is that the losses f_ξ are *locally* smooth, and that their local smoothness constants are uniformly bounded with respect to ξ. We defer to Proposition B.6 in the appendix for a formal proof that such local smoothness implies (14). In particular, one can see that our assumption is always verified if we are dealing with a finite sum of class C^2 losses.

Our smooth result in Corollary 2.3 is, as far as we know, the first $\mathcal{O}(1/\sqrt{T})$ anytime convergence rate for a stochastic variant of the Polyak step size, assuming only smoothness and convexity. Note that SGD has a $\mathcal{O}\left(\log(T)/T\right)$ anytime rate in this setting, see Appendix F.

Another interesting aspect of the convergence rate in (12) is that it is adaptive to interpolation. When there is no interpolation ($\sigma_*^2 > 0$), the convergence is dominated by the $\mathcal{O}(1/\sqrt{T})$ factor. On the other hand, as σ_*^2 gets closer to zero, the convergence rate in (12) approaches $\mathcal{O}(1/T)$, which is the expected accelerated rate of SGD under interpolation (Vaswani et al., 2019). We are unaware of prior work that establishes an anytime rate of convergence that is adaptive to interpolation. Though we show in Theorem E.1 in the appendix that the *complexity* of SGD can adapt to interpolation. We further contrast our rate to the best known anytime rate for SGD and $\texttt{SPS}_{\max}$ in Appendix F.

Remark 2.4 (Finite sum)**.** We emphazise that for finite-sum minimization $f = \frac{1}{n}\sum_{i=1}^n f_i$, our assumptions are drastically simplified. Assumption (11) in Corollary 2.2 is automatically true ; and assumption (14) in Corollary 2.3 is true when the f_i are locally smooth, for instance if they are of class C^2.

We have shown that SPS* has the optimal rate of convergence in the non-smooth setting, and a fast adaptive anytime rate in the smooth setting. This motivates us to think of SPS* as an idealized variant of the stochastic Polyak step size. In Appendix H we show how several practical variants of the stochastic Polyak step size, that do not need access to $f_\xi(x_*)$, can be viewed as approximations of SPS*.

3. Momentum and the Iterate Moving Average Method

The SPS* method is missing one important and practical ingredient, which is momentum. Furthermore, our previous convergence only holds for the average iterate, whereas the last iterate is often preferred since it is used in practice.

Momentum is often presented as replacing the gradient with an exponential moving average of gradients as follows: for $\gamma_t > 0$ and $\beta_t \in [0, 1)$, let

$$m_t = \beta_t m_{t-1} + g_t, \quad x_{t+1} = x_t - \gamma_t m_t. \tag{15}$$

To derive our momentum variant of SPS, we will make use of the equivalent reformulation given by

$$z_t = z_{t-1} - \eta_t g_t, \tag{16}$$

$$x_{t+1} = \frac{\lambda_{t+1}}{1+\lambda_{t+1}} x_t + \frac{1}{1+\lambda_{t+1}} z_t, \tag{17}$$

where $\eta_t > 0$ and $\lambda_t \in [0, 1]$ are hyperparameters. Though not obvious, the x_t iterates in (17) are equivalent to the x_t iterates of Momentum (15) by choosing a particular mapping between (β_t, γ_t) and (λ_t, η_t), see Defazio & Gower (2021, Thm. 1) and Lemma I.1 for convenience.

Inspired by both Wang et al. (2023) and Schaipp et al. (2024), we now choose the learning rate η_t in (16) that minimizes an upper bound on $D_t := \|z_t - x_*\|^2$. We have

$$D_t = D_{t-1} - 2\eta_t \langle g_t, z_{t-1} - x_* \rangle + \eta_t^2 \|g_t\|^2.$$

Using convexity we have that

$$\begin{aligned}\langle g_t, z_{t-1} - x_* \rangle &= \langle g_t, x_t - x_* \rangle + \langle g_t, z_{t-1} - x_t \rangle \\ &\geq f_t(x_t) - f_t(x_*) + \langle g_t, z_{t-1} - x_t \rangle.\end{aligned}$$

With this bound we have that

$$\begin{aligned}D_t &\leq D_{t-1} + \eta_t^2 \|g_t\|^2 \\ &\quad - 2\eta_t \left[f_t(x_t) - f_t(x_*) + \langle g_t, z_{t-1} - x_t \rangle \right].\end{aligned} \tag{18}$$

We will use this upper bound to choose an adaptive learning rate. Minimizing the right-hand side over $\eta_t \geq 0$ gives

$$\eta_t = \frac{\left[f_t(x_t) - f_t(x_*) + \langle g_t, z_{t-1} - x_t \rangle \right]_+}{\|g_t\|^2}. \tag{19}$$

We refer to (17) with the learning rate (19) as the *Iterate Averaging Adaptive Method* (IAM) method, for which we give the complete pseudo-code in Algorithm 1.

Algorithm 1 IAM: Iterate Averaging Adaptive Method.

Input: $z_{-1} = x_0 \in \mathbb{R}^d$, $\lambda_t > 0$, for $t = 0, \ldots, T$
for $t = 0$ to $T-1$ **do**

$$\eta_t = \frac{\left[f_t(x_t) - f_t(x_*) + \langle g_t, z_{t-1} - x_t \rangle \right]_+}{\|g_t\|^2}$$

$$z_t = z_{t-1} - \eta_t \nabla f_t(x_t)$$

$$x_{t+1} = \frac{\lambda_{t+1}}{1+\lambda_{t+1}} x_t + \frac{1}{1+\lambda_{t+1}} z_t$$

Return: x_T

3.1. Convergence Theorems

Again $D_t = \|z_t - x_*\|^2$. Our proofs all start from plugging in the step size (19) into (18) giving

$$D_t \le D_{t-1} - \frac{\left(f_t(x_t) - f_t(x_*) + \langle g_t, z_{t-1} - x_t \rangle \right)_+^2}{\|g_t\|^2}.$$

Lemma 3.1. Let f_ξ be convex for every ξ. The distances of iterates z_t of Algorithm 1 to a solution $x_* \in \mathbb{R}^d$ decreases monotonically, that is, with probability one

$$\|z_t - x_*\|^2 \le \|z_{t-1} - x_*\|^2 \le \cdots \le \|z_0 - x_*\|^2.$$

This type of monotonicity for stochastic methods is very rare, with the only other example that we are aware of being SPS* (cf. Theorem 2.1). To complete the convergence proofs, we will telescope the recurrence on D_t and bound the gradient norm on the denominator.

3.2. Non-smooth Setting

For our first proof we consider the setting where f_ξ could be non-smooth, thus g_ξ denotes a subgradient of f_ξ.

Theorem 3.2 (Non-smooth setting)**.** Consider the iterates of IAM in Algorithm 1 with the learning rate (19) and $\lambda_t = t$. Let f_ξ be convex for all ξ. Let $D := \|x_0 - x_*\|$,

$$G^2 := \max_{x \in \mathbb{B}_D(x_*)} \mathbb{E}_\xi \|g_\xi(x)\|^2,$$
$$B_f(x,y) := f(x) - f(y) - \langle \nabla f(y), x - y \rangle.$$

The suboptimality of the *last iterate* x_T is bounded by

$$\mathbb{E}\left[f(x_T) - f(x_*)\right] + \frac{1}{T+1} \sum_{t=1}^{T} t\, \mathbb{E}\left[B_f(x_{t-1}, x_t)\right] \le \frac{GD}{\sqrt{T+1}}. \tag{20}$$

This rate of convergence (20) is the same as SPS* (Corollary 2.2), with two notable differences: First this rate for IAM holds for the last iterate, as opposed to the average of the iterates, and second, this rate for IAM (20) can be faster than that of SPS* due to the additional Bregman divergences.

Theorem 3.2 restricts the parameter choice of $\lambda_t = t$, which when translated back (See Appendix I for details) to the momentum method (15) restricts the parameters (γ_t, β_t) to $\beta_t = \frac{t}{t+1} \frac{\eta_{t-1}}{\eta_t}$ and $\gamma_t = \frac{\eta_t}{t+2}$ for all t. To allow for other parameter settings, we provide Thm. J.1 in the Appendix, which allows for any deceasing $(\lambda_t)_t$, but does not establish a last-iterate convergence.

3.3. Smooth Setting

Here we consider the setting where we assume that the loss functions f_ξ satisfy a local *expected smoothness* condition.

Theorem 3.3 (Smooth setting)**.** Let f_ξ be convex for all ξ. Assume local *expected smoothness* (13) holds. Let x_t be the iterates of Algorithm 1 (IAM) with $\lambda_t = t$. It holds

$$\mathbb{E}\left[f(x_{T-1}) - f(x_*)\right] \le \frac{2L\|x_0 - x_*\|^2(\log(T)+1)}{T} + \frac{\sqrt{2L\sigma_*^2}\|x_0 - x_*\|}{\sqrt{T}}. \tag{21}$$

Analogous to the SPS* result in (14), the above shows that IAM is adaptive to interpolation, since equation (21) gives a $\tilde{\mathcal{O}}(1/T)$ convergence in the case of interpolation ($\sigma_*^2 = 0$).

In contrast to the convergence of SPS* in Corollary 2.3 the rate of convergence of IAM in (21) has an additional $\log(T+1)$ on the non-dominant $\mathcal{O}(\frac{1}{T+1})$ term.

4. Experiments

Here we present several numerical results. First, we test the extent of our convergence theory for SPS* and IAM. According to Remark 2.4, both SPS* and IAM will converge for differentiable convex finite-sum problems, even when the loss is non-smooth and non-Lipschitz. We test this on Poisson regression in Appendix L.1, where we show that IAM converges to a loss value comparable to L-BFGS, and to SGD with the best step size chosen from a grid. In Appendix L.2 we investigate how IAM behaves when $f_\xi(x_*)$ is wrongly specified (or guessed inaccurately). Finally, in Section 4.1 we use IAM and an Adam variant of IAM for model distillation.

4.1. Black-box Model Distillation

Here we consider a variant of knowledge distillation where the goal is to train a small model (called *student*) while having access to a pretrained, large model (called *teacher*).

The main idea we propose here is that, when training the

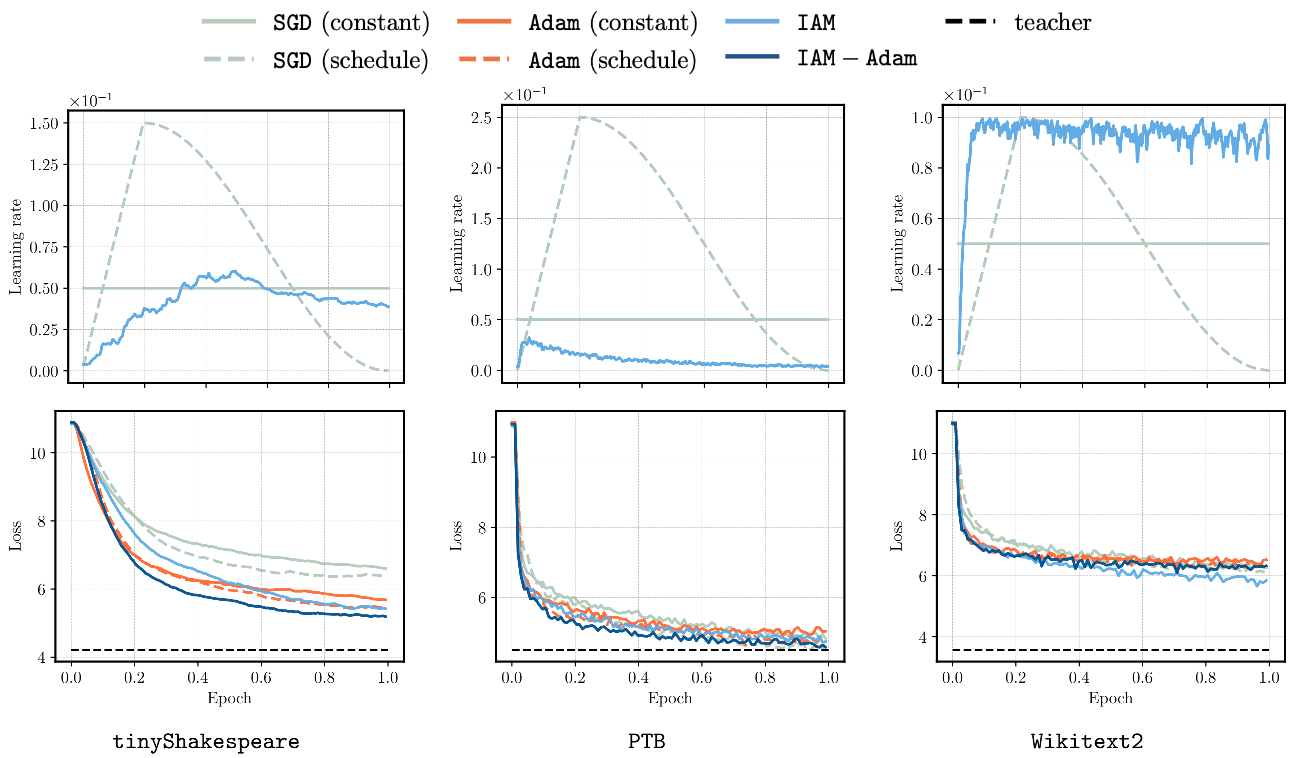


Figure 1. Distilling a teacher GPT2 on three datasets. Adaptive learning rate of IAM and learning rates of SGD **(top)** and cross-entropy training loss **(bottom)**. Black line marks the average teacher loss.

student, the loss of teacher model for a given batch ξ can be used as an approximation of $f_\xi(x_*)$.

For a given batch $\xi \sim \mathcal{P}$ from the training set of the student, denote by $f^s_\xi(x)$ the loss function[1] of the student with weights x for batch ξ. Denote by $f^\mathcal{T}_\xi$ the loss of the pretrained teacher model for the same batch. Since the teacher is a significantly larger and more expressive model, we can assume that even after training the student, its loss will not fall below $f^\mathcal{T}_\xi$. Thus, we use $f^s_\xi(x_*) \approx f^\mathcal{T}_\xi$ for the IAM method (Algorithm 1) to train the student.

Many variations of knowledge distillation have been proposed (Hinton et al., 2015; Beyer et al., 2022; Hsieh et al., 2023). The variant we present here is slightly different to previous works in that it requires only access to the batch loss of the teacher model (and not to the logits). We discuss this relationship in more detail in Appendix L.3.

We use three different datasets, tinyShakespeare, PTB and Wikitext2. As teacher model we use a pretrained GPT2 model with 774M parameters (Radford et al., 2019; Wolf et al., 2020). The student models are much smaller GPT2 architectures. All details are deferred to Appendix L.3. Our results are in Figure 1. We compare IAM and IAM-Adam (IAM with an Adam preconditioner, see Appendix K) to SGD and Adam with (i) constant learning rate, and (ii) *warmup+cosine-decay* schedule; tuning procedures are detailed in Appendix L.3.

We find that both versions of IAM achieve the best resulting loss on all three problems. Consequently, when we are able to load a suitable pretrained teacher model, we find that IAM is able to efficiently train a student model without any hyperparameter tuning.

5. Limitations

The limitation of our methods is that they require the batch loss at an optimal point. Because of this, outside of applications that interpolate, or our model distillation setup, it could be hard to find an applications for SPS* and IAM.

[1]This is usually the cross-entropy loss for the language modeling tasks we consider.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Armijo, L. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16:1–3, 1966.

Barré, M., Taylor, A., and d'Aspremont, A. Complexity guarantees for Polyak steps with momentum. In *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 452–478. PMLR, 09–12 Jul 2020.

Bauschke, H. H. and Combettes, P. L. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, 2nd edition edition, 2017.

Berrada, L., Zisserman, A., and Kumar, M. P. Training neural networks for and by interpolation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 799–809, 13–18 Jul 2020.

Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., and Kolesnikov, A. Knowledge distillation: A good teacher is patient and consistent. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10915–10924, 2022.

Bollapragada, R., Chen, T., and Ward, R. On the fast convergence of minibatch heavy ball momentum. *IMA Journal of Numerical Analysis*, 2024.

Carmon, Y. and Hinder, O. Making SGD parameter-free. In *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 2360–2389. PMLR, 02–05 Jul 2022.

Combettes, P. L. Perspective functions: Properties, constructions, and examples. *Set-Valued and Variational Analysis*, 26(2):247–264, April 2017.

Cottier, B., Rahman, R., Fattorini, L., Maslej, N., and Owen, D. The rising costs of training frontier AI models, 2024, arXiv:2405.21015.

Cutkosky, A. Artificial constraints and hints for unbounded online learning. In *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pp. 874–894. PMLR, 25–28 Jun 2019a.

Cutkosky, A. Anytime online-to-batch, optimism and acceleration. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1446–1454. PMLR, 09–15 Jun 2019b.

Defazio, A. and Gower, R. M. The power of factorial powers: New parameter settings for (stochastic) optimization. In *Proceedings of The 13th Asian Conference on Machine Learning*, volume 157 of *Proceedings of Machine Learning Research*, pp. 49–64. PMLR, 17–19 Nov 2021.

Defazio, A. and Mishchenko, K. Learning-rate-free learning by D-adaptation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7449–7479. PMLR, 2023.

Drori, Y. and Teboulle, M. An optimal variant of kelley's cutting-plane method. *Mathematical Programming*, 160: 321–351, 2016.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. Least angle regression. *The Annals of Statistics*, 32(2):407–499, 2004.

Fanaee-T, H. and Gama, J. Bike sharing dataset. UCI Machine Learning Repository, 2014.

Garrigos, G. and Gower, R. M. Handbook of convergence theorems for (stochastic) gradient methods. *arXiv preprint arXiv:2301.11235*, 2023.

Garrigos, G., Gower, R. M., and Schaipp, F. Function value learning: Adaptive learning rates based on the Polyak stepsize and function splitting in ERM. *arXiv preprint arXiv:2307.14528*, 2023.

Ghadimi, E., Feyzmahdavian, H. R., and Johansson, M. Global convergence of the heavy-ball method for convex optimization. In *2015 European Control Conference (ECC)*, pp. 310–315, 2015.

Gower, R., Sebbouh, O., and Loizou, N. SGD for structured nonconvex functions: Learning rates, minibatching and interpolation. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 1315–1323. PMLR, 13–15 Apr 2021.

Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. SGD: General Analysis and Improved Rates. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 5200–5209. PMLR, June 2019.

Gower, R. M., Richtárik, P., and Bach, F. Stochastic quasi-gradient methods: Variance reduction via Jacobian sketching. *Mathematical Programming*, 2020.

Gower, R. M., Blondel, M., Gazagnadou, N., and Pedregosa, F. Cutting some slack for SGD with adaptive Polyak stepsizes, 2022, arXiv:2202.12328.

Hazan, E. and Kakade, S. Revisiting the Polyak step size, 2019, arXiv:1905.00313.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hsieh, C.-Y., Li, C.-L., Yeh, C.-k., Nakhost, H., Fujii, Y., Ratner, A., Krishna, R., Lee, C.-Y., and Pfister, T. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8003–8017. Association for Computational Linguistics, July 2023.

Ivgi, M., Hinder, O., and Carmon, Y. DoG is SGD's best friend: A parameter-free dynamic step size schedule. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14465–14499. PMLR, 2023.

Karpathy, A. char-rnn. `https://github.com/karpathy/char-rnn`, 2015.

Kavis, A., Levy, K. Y., Bach, F., and Cevher, V. UniXGrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. *Advances in neural information processing systems*, 32, 2019.

Khaled, A., Mishchenko, K., and Jin, C. DoWG unleashed: An efficient universal parameter-free gradient descent method. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Lacoste-Julien, S., Schmidt, M., and Bach, F. A simpler approach to obtaining an $O(1/t)$ convergence rate for the projected stochastic subgradient method, 2012, arXiv:1212.2002.

Latafat, P., Themelis, A., Stella, L., and Patrinos, P. Adaptive proximal algorithms for convex optimization under local Lipschitz continuity of the gradient. *Mathematical Programming*, 10 2024.

Lee, K., Cheng, A. N., Paquette, C., and Paquette, E. Trajectory of mini-batch momentum: batch size saturation and convergence in high dimensions. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 2024.

Levy, K. Y., Yurtsever, A., and Cevher, V. Online adaptive methods, universality and acceleration. *Advances in Neural Information Processing Systems*, 31, 2018.

Li, T. and Lan, G. A simple uniformly optimal method without line search for convex optimization. *arXiv preprint arXiv:2310.10082*, 2023.

Liu, C., Zhu, L., and Belkin, M. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *Applied and Computational Harmonic Analysis*, 59:85–116, 2022.

Liu, D. C. and Nocedal, J. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, 1989.

Loizou, N., Vaswani, S., Laradji, I. H., and Lacoste-Julien, S. Stochastic Polyak step-size for SGD: An adaptive learning rate for fast convergence. In *AISTATS*, volume 130 of *Proceedings of Machine Learning Research*, pp. 1306–1314. PMLR, 2021.

Loshchilov, I. and Hutter, F. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Ma, S., Bassily, R., and Belkin, M. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3325–3334. PMLR, 10–15 Jul 2018.

Malitsky, Y. and Mishchenko, K. Adaptive gradient descent without descent. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6702–6712. PMLR, 13–18 Jul 2020.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models, 2016, arXiv:1609.07843.

Mhammedi, Z. and Koolen, W. M. Lipschitz and comparator-norm adaptivity in online learning. In *Conference on Learning Theory*, pp. 2858–2887. PMLR, 2020.

Mishchenko, K. and Defazio, A. Prodigy: An expeditiously adaptive parameter-free learner. In *Forty-first International Conference on Machine Learning*, 2024.

Nesterov, Y. Universal gradient methods for convex optimization problems. *Mathematical Programming*, 152 (1-2):381–404, 2014.

Oikonomou, D. and Loizou, N. Stochastic Polyak step-sizes and momentum: Convergence guarantees and practical performance. *arXiv preprint arXiv:2406.04142*, 2024.

Orabona, F. A modern introduction to online learning, 2019, arXiv:1912.13213.

Orabona, F. and Pál, D. Coin betting and parameter-free online learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 577–585, 2016.

Orvieto, A. and Xiao, L. An adaptive stochastic gradient method with non-negative Gauss-Newton stepsizes, 2024, arXiv:2407.04358.

Orvieto, A., Lacoste-Julien, S., and Loizou, N. Dynamics of SGD with stochastic Polyak stepsizes: Truly adaptive variants and convergence to exact solution. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26943–26954. Curran Associates, Inc., 2022.

Pedregosa, F. and Schaipp, F. Stochastic Polyak step-size, faster rates under strong convexity. http://fa.bianp.net/blog/2023/sps2/, 2023.

Peypouquet, J. *Convex Optimization in Normed Spaces*. SpringerBriefs in Optimization. Springer International Publishing, Cham, 2015.

Polyak, B. T. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Polyak, B. T. *Introduction to Optimization*. Optimization Software, New York, 1987.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. *OpenAI*, 2019.

Richtárik, P., Giancola, S. M., Lubczyk, D., and Yadav, R. Local curvature descent: Squeezing more curvature out of standard and Polyak gradient descent, 2024, arXiv:2405.16574.

Rodomanov, A., Kavis, A., Wu, Y., Antonakopoulos, K., and Cevher, V. Universal gradient methods for stochastic convex optimization. *arXiv preprint arXiv:2402.03210*, 2024.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. In *3rd International Conference on Learning Representations*, 2015.

Schaipp, F., Gower, R. M., and Ulbrich, M. A stochastic proximal Polyak step size. *Transactions on Machine Learning Research*, 2023.

Schaipp, F., Ohana, R., Eickenberg, M., Defazio, A., and Gower, R. M. MoMo: Momentum models for adaptive learning rates. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 43542–43570. PMLR, 21–27 Jul 2024.

Schmidt, R. M., Schneider, F., and Hennig, P. Descending through a crowded valley - benchmarking deep learning optimizers. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9367–9376. PMLR, 18–24 Jul 2021.

Sebbouh, O., Gower, R. M., and Defazio, A. Almost sure convergence rates for stochastic gradient descent and stochastic heavy ball. In *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 3935–3971. PMLR, 15–19 Aug 2021.

Streeter, M. and McMahan, H. B. No-regret algorithms for unconstrained online convex optimization. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, pp. 2402–2410, 2012.

Takezawa, Y., Bao, H., Sato, R., Niwa, K., and Yamada, M. Polyak meets parameter-free clipped gradient descent, 2024, arXiv:2405.15010.

Vaswani, S., Bach, F., and Schmidt, M. Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1195–1204. PMLR, 16–18 Apr 2019.

Wang, B. Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

Wang, X., Johansson, M., and Zhang, T. Generalized Polyak step size for first order optimization with momentum. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35836–35863. PMLR, 23–29 Jul 2023.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, October 2020.

Contents

1 Introduction **1**

1.1 Stochastic Polyak Step Size . . . 2

1.2 Related Work and Contributions . . . 2

2 Stochastic Polyak Step Size **5**

2.1 Convergence Theory for Convex Problems . . . 5

3 Momentum and the Iterate Moving Average Method **6**

3.1 Convergence Theorems . . . 7

3.2 Non-smooth Setting . . . 7

3.3 Smooth Setting . . . 7

4 Experiments **7**

4.1 Black-box Model Distillation . . . 7

5 Limitations **8**

A Comparison of Adaptive Methods **14**

B Convex Analysis and Subgradients **14**

C Auxiliary Lemmas **16**

D Missing Proofs **18**

D.1 Sketch proof of Theorem 2.1 . . . 18

D.2 Proof of Theorem 2.1 . . . 19

D.3 Proof of Corollary 2.2 . . . 20

D.4 Proof of Corollary 2.3 . . . 21

D.5 Preliminary Lemmas for `IAM` . . . 21

D.6 Proof of Theorem 3.2 . . . 23

D.7 Proof of Theorem 3.3 . . . 24

E Complexity of `SGD` with Adaptivity to Interpolation **25**

F Detailed Comparison of `SPS*` Convergence in Smooth Case **26**

G Convergence in (Locally) Strongly Convex Case **26**

H Approximating `SPS*` and Safe-guards **29**

Table 2. A summary of related work and conceptual differences to our approach and the work in AcceleGrad (Levy et al., 2018), UniXGrad (Kavis et al., 2019), AC-FGM (Li & Lan, 2023), Prodigy (Mishchenko & Defazio, 2024), and USFGM (Rodomanov et al., 2024).

Algorithm	Last iterate	Smooth problems	Non-smooth problems	Unbounded domain	Stoch. gradients	Can increase step size
AcceleGrad	✗	✗[(1)]	✓	✗	✓	✗
UniXGrad	✗	✓	✓	✗	✓	✗
AC-FGM	✗	✓	✓	✓	✗	✓
Prodigy	✗	✓	✓	✓	✗	✓
USFGM	✓	✓	✓	✗	✓	✗
`SPS*` (our result)	✗	✓	✓	✓	✓	✓
`IAM` (ours)	✓	✓	✓	✓	✓	✓

[(1)] AcceleGrad's smooth analysis is for deterministic problems.

I Momentum and Iterate Averaging **29**

J Additional Proof for `IAM` with Decreasing λ_t **30**

K An `Adam` Variant of `IAM` **32**

L Experiments **32**

L.1 Non-Lipschitz Non-smooth Convex Problem 32

L.2 Misspecification of $f_\xi(x_*)$ 33

L.3 Supplementary Material on Distillation Experiment 34

A. Comparison of Adaptive Methods

In Appendix A we make a qualitative comparison between our methods `SPS*` and `IAM` and other adaptive methods.

B. Convex Analysis and Subgradients

Here we introduce and define some of the more technical bits of convex analysis we need throughout the paper. In particular we make precise the technical assumptions that we are making on the functions f_ξ, which correspond to the assumptions made in the Section 9 of Garrigos & Gower (2023).

Throughout our paper, we consider for every sampled data ξ a loss function $f_\xi : \mathbb{R}^d \to \mathbb{R}$ taking finite values. We also always assume that f_ξ is convex, which implies that it is continuous on $\mathbb{R}^d$ (see Proposition 3.5 in (Peypouquet, 2015)). Nevertheless, we do not always assume that our loss functions f_ξ are differentiable. For example, $f_\xi(x)$ could be defined with an absolute value, such as $f_\xi(x) = |w_\xi^\top x - y_\xi|$ where w_ξ is a sample feature vector and y_ξ a target value. In general, instead of using gradients we will making use of *subgradients*, which play a similar role.

Definition B.1. Let $f : \mathbb{R}^d \to \mathbb{R}$, and $x \in \mathbb{R}^d$. We say that $g \in \mathbb{R}^d$ is a **subgradient** of f at $x \in \mathbb{R}^d$ if

$$\text{for every } y \in \mathbb{R}^d, \quad f(y) - f(x) - \langle g, y - x \rangle \geq 0.$$

Since our loss functions f_ξ are convex and continuous, we are guaranteed that at every $x \in \mathbb{R}^d$, there exists some subgradient that we will note $g_\xi(x)$ (the existence of such subgradient is stated in [Prop. 3.25](Peypouquet, 2015) and [Cor. 8.40](Bauschke & Combettes, 2017)). In our proofs we will often need to take the expectation of these subgradients $g_\xi(x)$ with respect to ξ. To be able to do this, we must formally assume throughout that the function $\xi \mapsto g_\xi(x)$ is measurable for every $x \in \mathbb{R}^d$. This will for instance allow us to say that the expectation of $g_\xi(x)$ is a subgradient of f at x (see Lemma 9.5 in (Garrigos & Gower, 2023)).

We know give some technical details about locally smooth functions, which is the assumption made in Corollary 2.2.

Definition B.2. We say that $f : \mathbb{R}^d \to \mathbb{R}$ is locally smooth if it is differentiable and if ∇f is locally Lipschitz continuous.

Note that this definition is equivalent to require ∇f to be Lipschitz continuous over any bounded subset of $\mathbb{R}^d$. A simple example of locally smooth functions are C^2 functions: their hessians are locally bounded by continuity, so the mean value inequality entails that their gradients are locally Lipschitz.

Lemma B.3 (Local descent lemma)**.** If f is locally smooth, then for every bounded set $B \subset \mathbb{R}^d$ there exists $L_B \geq 0$ such that

$$\text{for all } x, y \in B, \quad f(y) - f(x) - \langle \nabla f(x), y - x \rangle \leq \frac{L_B}{2} \|y - x\|^2. \tag{22}$$

Proof. This is just a local version of the classic proof of the descent lemma, see e.g. Lemma 1.30 from (Peypouquet, 2015). Without loss of generality, we can assume that B is convex and compact (simply replace B with its closed convex hull). By compactness, we know that ∇f is Lipschitz continous on B, for some constant $L_B \geq 0$. We can then start the proof and fix $x, y \in B$. Define the auxiliary function $g(t) = f((1-t)x + ty) - t\langle \nabla f(x), y - x\rangle$ for $t \in [0, 1]$. It is differentiable and verifies

$$g(1) - g(0) = \int_0^1 g'(t)\, dt$$

which is equivalent, by definition of g, to

$$f(y) - f(x) - \langle \nabla f(x), y - x \rangle = \int_0^1 \langle \nabla f((1-t)x + ty) - \nabla f(x), y - x \rangle \, dt.$$

Now we use the Cauchy-Schwarz inequality, together with the Lipschitzness of ∇f (note that $z := (1-t)x + ty)$ belongs to B which is convex!), to obtain

$$\begin{aligned}
& f(y) - f(x) - \langle \nabla f(x), y - x \rangle \\
\leq \;& \int_0^1 \|\nabla f((1-t)x + ty) - \nabla f(x)\| \|y - x\| \, dt \\
\leq \;& \int_0^1 L_B \|(1-t)x + ty) - x\| \|y - x\| \, dt \\
= \;& \int_0^1 L_B t \|y - x\|^2 \, dt \\
= \;& \frac{L_B}{2} \|y - x\|^2.
\end{aligned}$$

□

Locally smooth functions verify locally the following useful bound:

Proposition B.4. If $f : \mathbb{R}^d \to \mathbb{R}$ is locally smooth and bounded from below, then for every bounded set $B \subset \mathbb{R}^d$ there exists $L_B \geq 0$ such that

$$\text{for all } x \in B, \quad \frac{1}{2L_B} \|\nabla f(x)\|^2 \leq f(x) - \inf f.$$

Proof. This proof is just an adaptation of a classical result (see e.g. Lemma 2.28 from (Garrigos & Gower, 2023)) by making use of additional local arguments. Here again, without loss of generality, we can assume that B is compact. Let L_B be the local smoothness constant provided by the local descent lemma B.3. Let $T : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ be the map defined by

$$T(x, \gamma) = x - \gamma \nabla f(x).$$

Because ∇f is supposed continuous, we know that T is continuous. Now we define

$$K := \{x - \gamma \nabla f(x) \mid x \in B, \gamma \in [0, \tfrac{1}{L_B}]\} \subset \mathbb{R}^d.$$

From our definitions it is clear that $K = T(B \times [0, \frac{1}{L_B}])$. In other words, it is the image of a compact set by a continuous function, which means that K is compact. It is also clear that K contains B (simply take $\gamma = 0$). Now we can use again the local descent lemma B.3 to obtain that f verifies (22) with a constant L_K. Without loss of generality, we can assume that $L_K \geq L_B$ (simply replace L_K with $\max\{L_K, L_B\}$). Now we can end the proof. Let $x \in B$ be fixed, and define $y := x - \frac{1}{L_K}\nabla f(x)$. By construction, $x \in B \subset K$ and $y = T(x, \frac{1}{L_K}) \in K$. So we can use the descent lemma inequality on K to obtain

$$f(x - \tfrac{1}{L_K}\nabla f(x)) - f(x) - \langle \nabla f(x), -\tfrac{1}{L_K}\nabla f(x)\rangle \leq \frac{L_k}{2}\|\tfrac{1}{L_K}\nabla f(x)\|^2.$$

Rewriting and reorganizing terms, we obtain further

$$f(x - \tfrac{1}{L_K}\nabla f(x)) - f(x) \leq -\frac{1}{2L_K}\|\nabla f(x)\|^2.$$

We obtain the desired result by observing that $f(x - \frac{1}{L_K}\nabla f(x)) \geq \inf f$. □

Definition B.5. We say that the family (f_ξ) is uniformly locally smooth if, for every bounded set $B \subset \mathbb{R}^d$, there exists a constant $L_B \geq 0$ independent of ξ such that each f_ξ is L_B-smooth on B.

It is easy to see that any *finite* family of locally smooth functions is uniformly locally smooth: simply take the maximum of the local smoothness constants. In particular, any finite sum of C^2 functions is uniformly locally smooth.

Proposition B.6. Suppose that the family of functions (f_ξ) is uniformly locally smooth and bounded from below. Then, for every bounded set $B \subset \mathbb{R}^d$, there exists $L_B \geq 0$ such that

$$\text{for all } x \in B, \quad \mathbb{E}\left[\|\nabla f_\xi(x)\|^2\right] \leq 2L_B(f(x) - \mathbb{E}\left[\inf f_\xi\right]).$$

Proof. By definition of uniformly locally smooth functions, there exists $L_B \geq 0$ such that each function f_ξ is L_B-smooth on B, which means that we can use Proposition B.4 to write

$$\text{for all } x \in B, \quad \frac{1}{2L_B}\|\nabla f_\xi(x)\|^2 \leq f_\xi(x) - \inf f_\xi.$$

The conclusion follows after taking expectation with respect to ξ. □

C. Auxiliary Lemmas

Lemma C.1. Let $A, B \geq 0$ which are not simultaneously zero. Let $\psi(t) = \frac{t^2}{At+B}$ be defined for $t \geq 0$. Then ψ is convex and increasing over $[0, +\infty)$, and its inverse is $\psi^{-1}(s) = \frac{1}{2}(sA + \sqrt{s^2A^2 + 4sB})$.

Proof. The function ψ is twice differentiable over $[0, +\infty)$, and we can compute

$$\psi'(t) = \frac{At^2 + 2Bt}{(At+B)^2} \quad \text{and} \quad \psi''(t) = \frac{(2At+2B)(At+B)^2 - 2(At^2+2Bt)(At+B)A}{(At+B)^4} = \frac{2B^2}{(At+B)^3}.$$

It is immediate to see that ψ' and ψ'' are positive, from which we deduce that ψ is convex and increasing.

Next, consider two cases. If $At + B = 0$, this implies $t = 0$ and $\psi(0) = 0$, thus $\psi^{-1}(0) = 0$. If, however, $At + B \neq 0$, for $s \geq 0$ it holds

$$\psi(t) = s \iff \frac{t^2}{At + B} = s \iff t^2 - Ast - Bs = 0.$$

The last equation has a unique nonnegative solution which is $t = \frac{1}{2}(sA + \sqrt{s^2A^2 + 4sB})$, from which we deduce the expression for ψ^{-1}. □

We will use the following lemma which is often used to study methods AdaGrad type methods.

Lemma C.2. Let $c_0, \ldots, c_k \geq 0$ be some non-negative numbers with $c_0 > 0$, and denote $S_t = \sum_{i=0}^t c_i$, then

$$\sqrt{S_t} \leq \sum_{k=0}^{t} \frac{c_k}{\sqrt{S_k}}. \tag{23}$$

Proof. The proof of the lemma can be found in various sources, for instance in the Appendix A of Levy et al. (2018), but since it is very short, we will provide it here for completeness as well. Observe that for any $\alpha \in [0, 1]$, it holds $\alpha \geq 1 - \sqrt{1-\alpha}$. Substituting $\alpha = c_k/S_k \in [0, 1]$, we get

$$\frac{c_k}{S_k} \geq 1 - \sqrt{1 - \frac{c_k}{S_k}} \implies \frac{c_k}{\sqrt{S_k}} \geq \sqrt{S_k} - \sqrt{S_k - c_k} = \sqrt{S_k} - \sqrt{S_{k-1}}.$$

Summing the last inequality from $k = 1$ to $k = t$ and using $\sqrt{S_0} = \frac{c_0}{\sqrt{S_0}}$, we get the claim. □

We also rely on the following result.

Lemma C.3 (Extended Titu's Lemma)**.** For any random variable X and positive-valued random variable Y, it holds

$$\mathbb{E}\left[\frac{(X)_+^2}{Y}\right] \geq \frac{(\mathbb{E}[X])_+^2}{\mathbb{E}[Y]}. \tag{24}$$

In addition, for any numbers $a_0, \ldots, a_k$ and positive numbers $b_0, \ldots, b_k$, we have

$$\sum_{t=0}^{k} \frac{(a_t)_+^2}{b_t} \geq \frac{\left(\sum_{t=0}^k a_t\right)_+^2}{\sum_{t=0}^k b_t}. \tag{25}$$

Proof. The proof follows from applying Jensen's inequality to the function $\varphi(x, y) = (x)_+^2/y$. To prove that φ is convex takes some work, and it is given in Lemma A.4 in Garrigos & Gower (2023). We also provide a different proof that $\varphi(x, y)$ is convex in the following Lemma C.4 by viewing $\varphi(x, y)$ as a perspective function. The discrete result (25) follows from applying (24) with uniform distribution over $\{a_0, \ldots, a_k\}$ and $\{b_0, \ldots, b_k\}$. □

Lemma C.4. Consider the function $\varphi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, $(x, y) \mapsto \varphi(x, y)$, where

$$\varphi(x, y) := \begin{cases} \frac{(x)_+^2}{y} & \text{if } y > 0, \\ 0 & \text{if } (y = 0) \wedge (x \leq 0), \\ +\infty & \text{else.} \end{cases} \tag{26}$$

Then, φ is closed, proper and convex on $\mathbb{R} \times \mathbb{R}$.

Proof. Define the convex function $h(x) := (x)_+^2$. From Combettes (2017, Def. 2.1), it follows that $\varphi(x, y)$ defined as in (26) is the perspective function of h, that is, for $y > 0$ we have $\varphi(x, y) = yh(x/y)$; for $y = 0$, we compute $\lim_{\alpha \to \infty} \frac{(\alpha x)_+^2}{\alpha} = 0$

if $x \le 0$ and $+\infty$ otherwise. The perspective functions of closed, proper, convex functions is convex itself (Combettes, 2017, Prop. 2.3). □

Here we show that the expected smoothness bound (13) is a consequence of assuming that f_ξ is almost surely L–smooth.

Lemma C.5. Let f_ξ be L–smooth for every ξ, that is let

$$f_\xi(y) \;\le\; f_\xi(x) + \langle \nabla f_\xi(x), y - x\rangle + \frac{L}{2}\|y-x\|^2. \tag{27}$$

As a consequence we have that

$$\mathbb{E}\left[\|\nabla f_\xi(x)\|^2\right] \le 2L\big(f(x) - \inf f + \sigma_*^2\big), \tag{28}$$

where

$$\sigma_*^2 := f(x_*) - \mathbb{E}\left[\inf f_\xi\right] \ge 0.$$

The proof can be found in Garrigos & Gower (2023, Lem. 4.19).

D. Missing Proofs

D.1. Sketch proof of Theorem 2.1

Here we give a sketch of the proof of Theorem 2.1 so that we can better highlight the main ideas behind the proof, and the main novelty.

Theorem 2.1. [Convergence of SPS*] Consider problem (1) and let the iterates $(x_t)_{t\ge 0}$ be given by (2), and let $D := \|x_0 - x_*\|$. If $f_\xi : \mathbb{R}^d \to \mathbb{R}$ is convex with probability one, then the iterates are almost surely monotone:

$$\|x_{t+1} - x_*\|^2 \;\le\; \|x_t - x_*\|^2 \text{ with probability } 1. \tag{8}$$

If there exist $A, B > 0$ such that for all $x \in \mathbb{B}_D(x_*)$

$$\mathbb{E}_\xi\left[\|g_\xi(x)\|^2\right] \le A(f(x) - f(x_*)) + B, \tag{9}$$

then for $\bar{x}_T := \frac{1}{T}\sum_{t=0}^{T-1} x_t$ we have that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \le \frac{D^2 A}{T} + \sqrt{\frac{D^2 B}{T}}, \quad \forall T \in \mathbb{N}. \tag{10}$$

Proof Sketch. Plugging the SPS* step size (7) into (6) and re-arranging gives

$$\frac{(f_t(x_t) - f_t(x_*))_+^2}{\|g_t\|^2} \quad\le\quad \|x_t - x_*\|^2 - \|x_{t+1} - x_*\|^2.$$

Taking expectation conditioned on x_t, and using that the map $(z_1, z_2) \mapsto (z_1)_+^2/z_z$ is jointly convex on $\mathbb{R} \times \mathbb{R}_{\ge 0}$ (cf. Lemma C.4) together with Jensen's inequality, we get

$$\frac{(f(x_t) - f(x_*))_+^2}{\mathbb{E}_t\left[\|g_t\|^2\right]} \quad\le\quad \|x_t - x_*\|^2 - \mathbb{E}_t\left[\|x_{t+1} - x_*\|^2\right].$$

We can then use our main assumption (9) to bound the denominator of the left hand side giving

$$\frac{(f(x_t) - f(x_*))^2}{A(f(x_t) - f(x_*)) + B} \quad\le\quad \|x_t - x_*\|^2 - \mathbb{E}_t\left[\|x_{t+1} - x_*\|^2\right].$$

Taking expectation again, and averaging both sides over $t = 0, \ldots, T-1$ and telescoping we have that

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{(f(x_t)-f(x_*))^2}{A(f(x_t)-f(x_*))+B}\right] \leq \frac{\|x_0-x_*\|^2}{T} - \frac{\mathbb{E}\left[\|x_T-x_*\|\right]^2}{T} \leq \frac{\|x_0-x_*\|^2}{T}.$$

The final step of the proof, and the main technical novelty, follows by defining the function $\psi(r) = \frac{r^2}{Ar+B}$ for $r \geq 0$, and noting that the left hand side of the above is equal to $\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\psi(f(x_t)-f(x_*))\right]$. We then apply Lemma C.1 in the appendix that shows that ψ is a convex monotone function. Being convex, we can bring the average over t and the expectation inside ψ giving

$$\psi(\mathbb{E}\left[f(\bar{x}_t)-f(x_*)\right]) \leq \frac{\|x_0-x_*\|^2}{T}.$$

Finally, Lemma C.1 also proves that ψ has an inverse given by

$$\psi^{-1}(s) = \tfrac{1}{2}(sA + \sqrt{s^2A^2+4sB}).$$

Applying this inverse to both sides and using that ψ^{-1} is monotone, gives the result. □ *End proof sketch.*

Next we give the complete and detailed proof of Theorem 2.1.

D.2. Proof of Theorem 2.1

Theorem 2.1. [Convergence of SPS*] Consider problem (1) and let the iterates $(x_t)_{t\geq 0}$ be given by (2), and let $D := \|x_0 - x_*\|$. If $f_\xi : \mathbb{R}^d \to \mathbb{R}$ is convex with probability one, then the iterates are almost surely monotone:

$$\|x_{t+1}-x_*\|^2 \;\leq\; \|x_t-x_*\|^2 \text{ with probability } 1. \tag{8}$$

If there exist $A, B > 0$ such that for all $x \in \mathbb{B}_D(x_*)$

$$\mathbb{E}_\xi\left[\|g_\xi(x)\|^2\right] \leq A(f(x)-f(x_*))+B, \tag{9}$$

then for $\bar{x}_T := \frac{1}{T}\sum_{t=0}^{T-1} x_t$ we have that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{D^2A}{T} + \sqrt{\frac{D^2B}{T}}, \quad \forall T \in \mathbb{N}. \tag{10}$$

Proof. For short-hand we use $f_t := f_{\xi_t}$ to be the stochastic function sampled at iteration t. Expanding the squares, using the definition of the algorithm and using the convexity of f_ξ, we have that

$$\begin{aligned}\|x_{t+1}-x_*\|^2 - \|x_t-x_*\|^2 &= 2\gamma_t^{\mathsf{SPS}*}\langle g_t, x_*-x_t\rangle + (\gamma_t^{\mathsf{SPS}*})^2\|g_t\|^2 \\ &\leq -2\gamma_t^{\mathsf{SPS}*}(f_t(x_t)-f_t(x_*)) + (\gamma_t^{\mathsf{SPS}*})^2\|g_t\|^2.\end{aligned}$$

If $g_t = 0$, then by definition we have that $\gamma_t^{\mathsf{SPS}*} = 0$, thus the right-hand side of the above is zero, and (8) holds. Suppose instead that $g_t \neq 0$. Substituting in $\gamma_t^{\mathsf{SPS}*}$ gives

$$\begin{aligned}\|x_{t+1}-x_*\|^2 - \|x_t-x_*\|^2 &\leq -2\frac{(f_t(x_t)-f_t(x_*))_+}{\|g_t\|^2}(f_t(x_t)-f(x_*)) + \frac{(f_t(x_t)-f_t(x_*))_+^2}{\|g_t\|^2} \\ &= -\frac{(f_t(x_t)-f_t(x_*))_+^2}{\|g_t\|^2},\end{aligned}$$

where in the last equality we use the identity $z(z)_+ = (z)_+^2$. Note that in both cases we obtained a nonpositive right-hand side, from which we deduce that (8) holds, that is, $(x_t)_{t\geq 0}$ is Fejér monotone.

Now, let $a_t := f_t(x_t) - f_t(x_*)$ and $b_t := \|g_t\|^2$, and define the function

$$\phi(a,b) = \begin{cases} \frac{(a)_+^2}{b} & \text{if } a \in \mathbb{R}, b > 0, \\ 0 & \text{if } a \leq 0, b = 0, \end{cases}$$

so that the previous inequality can be rewritten as

$$\phi(a_t, b_t) \leq \|x_t - x_*\|^2 - \|x_{t+1} - x_*\|^2. \tag{29}$$

Note that $\phi(a_t, b_t)$ is well-defined even in the case that $g_t = 0$. Indeed, the convexity of f_t implies in this case that x_t minimizes f_t, which means that $a_t \leq 0$ while $b_t = 0$. Our main trick is to use Jensen's inequality with regard to the function ϕ which is convex (see Lemma C.4 or the Appendix in Garrigos & Gower (2023) for a proof):

$$\phi(\mathbb{E}\left[a_t\right], \mathbb{E}\left[b_t\right]) \leq \mathbb{E}\left[\phi(a_t, b_t)\right] \leq \mathbb{E}\left[\|x_t - x_*\|^2\right] - \mathbb{E}\left[\|x_{t+1} - x_*\|^2\right]. \tag{30}$$

We can compute $\mathbb{E}\left[a_t\right] = \mathbb{E}\left[f_{\xi_t}(x_t) - f_{\xi_t}(x_*)\right] = \mathbb{E}\left[f(x_t) - \inf f\right]$ and $\mathbb{E}\left[b_t\right] = \mathbb{E}\left[\|g_t\|^2\right]$.

For the rest of the proof, we are going to use the fact that there exist two constants $A, B \geq 0$, which are not simultaneously zero, and such that (9) holds, that is

$$\mathbb{E}\left[\|g_\xi(x)\|^2\right] \leq A(f(x) - \inf f) + B, \text{ for every } x \in \mathbb{B}(x_*, D). \tag{31}$$

We are now going to inject this inequality (31) into (30). If $\mathbb{E}\left[\|g_t\|^2\right] \neq 0$, using the fact that $f(x_t) - \inf f \geq 0$ we obtain

$$\frac{\mathbb{E}\left[f(x_t) - \inf f\right]^2}{A\mathbb{E}\left[f(x_t) - \inf f\right] + B} \leq \phi(\mathbb{E}\left[a_t\right], \mathbb{E}\left[b_t\right]) \leq \mathbb{E}\left[\|x_t - x_*\|^2\right] - \mathbb{E}\left[\|x_{t+1} - x_*\|^2\right]. \tag{32}$$

Recall that we defined $\psi(r) = \frac{r^2}{Ar+B}$ for any $r \geq 0$. Let $r_t := \mathbb{E}\left[f(x_t) - \inf f\right]$. With this notation, the inequality (32) can be rewritten as

$$\psi(r_t) \leq \mathbb{E}\left[\|x_t - x_*\|^2\right] - \mathbb{E}\left[\|x_{t+1} - x_*\|^2\right]. \tag{33}$$

We observe that (33) remains true when $\mathbb{E}\left[\|g_t\|^2\right] = 0$. Indeed in this case, from the variance bound we have that

$$0 = \mathbb{E}\left[\|g_t\|^2\right] \geq \|\mathbb{E}\left[g_t\right]\|^2.$$

Furthermore it follows that $\mathbb{E}_t\left[g_t\right]$ is a subgradient of the full loss $f(x_t)$ (see Lemma 9.5 in (Garrigos & Gower, 2023)). Consequently x_t minimizes f, meaning in this case that we would have $r_t = 0$, and so $\psi(r_t) = 0 = \phi(0, 0)$.

For the last part of this proof, we sum over $t = 0, \ldots, T-1$ and divide by T to obtain, after telescoping terms:

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left[\psi(r_t)\right] \leq \frac{1}{T}\mathbb{E}\left[\|x_0 - x_*\|^2\right] - \frac{1}{T}\mathbb{E}\left[\|x_T - x_*\|^2\right] \leq \frac{D^2}{T}.$$

We now lower-bound the left-hand side term by using Jensen's inequality twice

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left[\psi(r_t)\right] \geq \psi\left(\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} r_t\right]\right) = \psi\left(\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}(f(x_t) - \inf f)\right]\right) \geq \psi\left(\mathbb{E}\left[f(\bar{x}_T) - \inf f\right]\right),$$

where in the first inequality we use the convexity of ψ, and in the second we use the convexity of f together with the fact that ψ is increasing, and we note the average of the iterates $\bar{x}_T := \frac{1}{T}\sum_{t=0}^{T-1} x_t$. The reader can look at Lemma C.1 for a proof that ψ is convex and monotone. Combining the two previous inequalities, we obtain

$$\psi\left(\mathbb{E}\left[f(\bar{x}_T) - \inf f\right]\right) \leq \frac{D^2}{T}.$$

Since ψ is increasing on $[0, +\infty)$, it has an inverse which is also increasing. Applying the inverse of ψ on both sides gives

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \psi^{-1}\left(\frac{D^2}{T}\right).$$

From Lemma C.1 we know that $\psi^{-1}(s) = \frac{1}{2}(sA + \sqrt{s^2A^2 + 4sB})$, and using the sublinearity of the square root we further have

$$\psi^{-1}(s) \;\leq\; \tfrac{1}{2}(sA + \sqrt{s^2A^2} + \sqrt{4sB}) = sA + \sqrt{sB}. \tag{34}$$

From this we finally obtain (10). $\square$

D.3. Proof of Corollary 2.2

Corollary 2.2 (Non-smooth setting)**.** Consider the setting of Theorem 2.1 where $A = 0$ and $B = G \geq 0$. In other words, the following *expected locally Lipschitz* assumption holds:

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq G^2, \quad \forall x \in \mathbb{B}_D(x_*). \tag{11}$$

It follows that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{GD}{\sqrt{T}}, \quad \forall T \in \mathbb{N}. \tag{12}$$

Proof. Observing that by assuming (11) we have that (9) holds with $A = 0$ and $B = G^2$. Thus the result follows by plugging in these constant into (10). $\square$

D.4. Proof of Corollary 2.3

Corollary 2.3 (Smooth setting)**.** Consider the setting of Theorem 2.1 where $A = 2L$ and $B = \sigma_*^2 := \inf f - \mathbb{E}_\xi\left[\inf f_\xi\right]$. That is, we assume local *expected smoothness*:

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq 2L\big(f(x) - \inf f + \sigma_*^2\big), \ \forall x \in \mathbb{B}_D(x_*). \tag{13}$$

It then follows that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{4L\|x_0 - x_*\|^2}{T} + \frac{\sqrt{2}\|x_0 - x_*\|\sigma_*^2}{\sqrt{T}}. \tag{14}$$

Proof. From the assumption in equation (13), we have that (9) holds with $A = 4L$ and $B = 2\sigma_*^2$. Thus the result follows by plugging in these constant into (10). Furthermore, note that (13) is a consequence of smoothness, see the variance transfer Lemma in [Section 4.3.3](Garrigos & Gower, 2023). $\square$

D.5. Preliminary Lemmas for `IAM`

Our proofs all start from the following Lemma.

Lemma D.1. Consider the iterates of Algorithm 1 with $\lambda_t > 0$. Assume that $g_t \neq 0$ for all $t \geq 0$. Let $g(x)$ denote the subgradient of $f(x)$. Denote by $\mathcal{F}_t$ the filtration generated by $\xi_0, \ldots, \xi_{t-1}$. If f_ξ is convex for every ξ, then:

(i) (Almost sure boundedness). With probability one, we have $\|z_t - x_*\| \leq \|x_0 - x_*\|$ and $\|x_t - x_*\| \leq \|x_0 - x_*\|$ for all $t \geq 0$.

(ii) (Single recurrence) It holds for any $t \geq 0$

$$\mathbb{E}\left[\|z_t - x_*\|^2 \mid \mathcal{F}_t\right] \ \leq \|z_{t-1} - x_*\|^2 - \frac{\big(f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\big)_+^2}{\mathbb{E}\left[\|g_t\|^2 \mid \mathcal{F}_t\right]}. \tag{35}$$

(iii) (Summed recurrence) It holds for any $k \geq 0$

$$\mathbb{E}\left[\|z_k - x_*\|^2\right] \ \leq \|z_0 - x_*\|^2 - \frac{\big(\sum_{t=0}^k \mathbb{E}\left[f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\right]\big)_+^2}{\sum_{t=0}^k \mathbb{E}\left[\|g_t\|^2\right]}. \tag{36}$$

Proof. Substituting (19) back into the bound (18) gives

$$\|z_t - x_*\|^2 \ \leq \|z_{t-1} - x_*\|^2 - \frac{\big(f_{\xi_t}(x_t) - f_{\xi_t}(x_*) + \langle g_t, z_{t-1} - x_t\rangle\big)_+^2}{\|g_t\|^2}.$$

This shows that $\|z_t - x_*\| \leq \|z_0 - x_*\| = \|x_0 - x_*\|$ almost surely for all $t \geq 0$. Since x_{t+1} is a convex combination of x_t and z_t (see line 5 in Algorithm 1) this also shows by a straightforward induction that $\|x_t - x_*\| \leq \|x_0 - x_*\|$ almost surely for all $t \geq 0$.

To prove (ii), we apply conditional expectation on the above inequality and using Lemma C.3, (24) we obtain

$$\mathbb{E}\left[\|z_t - x_*\|^2 \mid \mathcal{F}_t\right] \;\leq \|z_{t-1} - x_*\|^2 - \frac{\left(f(x_t) - f(x_*) + \langle \mathbb{E}\left[g_t \mid \mathcal{F}_t\right], z_{t-1} - x_t\rangle\right)_+^2}{\mathbb{E}\left[\|g_t\|^2 \mid \mathcal{F}_t\right]}.$$

Using that the expectation with respect to this filtration is independent of x_t, we have that the stochastic subgradient $\mathbb{E}\left[g_t \mid \mathcal{F}_t\right]$ is a subgradient of $f(x_t)$, see Lemma 9.5 in Garrigos & Gower (2023) for details[2]. Thus we can write $g(x_t) = \mathbb{E}\left[g_t \mid \mathcal{F}_t\right]$.

Now, define $a_t := f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle$ and $b_t = \mathbb{E}\left[\|g_t\|^2 \mid \mathcal{F}_t\right]$. Using (35) subsequently for $t = 0, \ldots, k$ and using the tower property, we obtain

$$\mathbb{E}\left[\|z_k - x_*\|^2\right] \;\leq \|z_0 - x_*\|^2 - \mathbb{E}\left[\sum_{t=0}^{k} \frac{(a_t)_+^2}{b_t}\right].$$

Now using Lemma C.3, (25) yields

$$\sum_{t=0}^{k} \frac{(a_t)_+^2}{b_t} \geq \frac{\left(\sum_{t=0}^{k} a_t\right)_+^2}{\sum_{t=0}^{k} b_t},$$

which implies, using (24), that

$$\mathbb{E}\left[\sum_{t=0}^{k} \frac{(a_t)_+^2}{b_t}\right] \geq \mathbb{E}\left[\frac{\left(\sum_{t=0}^{k} a_t\right)_+^2}{\sum_{t=0}^{k} b_t}\right] \geq \frac{\left(\sum_{t=0}^{k} \mathbb{E}\left[a_t\right]\right)_+^2}{\sum_{t=0}^{k} \mathbb{E}\left[b_t\right]}.$$

Altogether, we obtain (iii), that is

$$\mathbb{E}\left[\|z_k - x_*\|^2\right] \;\leq \|z_0 - x_*\|^2 - \frac{\left(\sum_{t=0}^{k} \mathbb{E}\left[f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\right]\right)_+^2}{\sum_{t=0}^{k} \mathbb{E}\left[\|g_t\|^2\right]}.$$

□

For our forthcoming proofs we will also make use of a *Bregman viewpoint* of the `IAM` step size.

Lemma D.2 (Bregman View)**.** For any $x_t, x_{t-1}, x_* \in \mathbb{R}^d$ and $\lambda_t \geq 0$ it holds

$$\begin{aligned} &f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle \\ &\quad = (1+\lambda_t)(f_{\xi_t}(x_t) - f_{\xi_t}(x_*)) - \lambda_t(f_{\xi_t}(x_{t-1}) - f_{\xi_t}(x_*)) + \lambda_t B_{f_{\xi_t}}(x_{t-1}, x_t), \end{aligned} \tag{37}$$

where $B_{f_\xi}(x,y)$ is the Bregman divergence

$$B_{f_\xi}(x,y) := f_\xi(x) - f_\xi(y) - \langle g_\xi(y), x - y\rangle.$$

Proof. By re-arranging (17) at time $t-1$ we have that

$$z_{t-1} - x_t \;=\; -\lambda_t(x_{t-1} - x_t). \tag{38}$$

Consequently

$$f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle = f(x_t) - f(x_*) - \lambda_t \langle g(x_t), x_{t-1} - x_t\rangle.$$

The proof follows by adding and subtracting $\lambda_t f_{\xi_t}(x_{t-1})$ as follows

$$\begin{aligned} &f_{\xi_t}(x_t) - f_{\xi_t}(x_*) - \lambda_t \langle g_t, x_{t-1} - x_t\rangle \\ &= (1+\lambda_t) f_{\xi_t}(x_t) - f_{\xi_t}(x_*) - \lambda_t f_{\xi_t}(x_{t-1}) + \lambda_t\left(f_{\xi_t}(x_{t-1}) - f_{\xi_t}(x_t) - \langle g_t, x_{t-1} - x_t\rangle\right) \\ &= (1+\lambda_t)(f_{\xi_t}(x_t) - f_{\xi_t}(x_*)) - \lambda_t(f_{\xi_t}(x_{t-1}) - f_{\xi_t}(x_*)) + \lambda_t B_{f_{\xi_t}}(x_{t-1}, x_t). \end{aligned}$$

□

[2]Very formally, here we need to assume the subgradients $g_\xi(x)$ are measurable in ξ so that this expectation is well defined.

Lemma D.3. Consider the iterates of Algorithm 1 with $\lambda_t = t$ and assume that f_ξ is convex for every ξ, with subgradients g_ξ. Let $g(x)$ be subgradients of $f(x)$. It holds

$$\sum_{t=0}^{k} f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle = (k+1)[f(x_k) - f(x_*)] + \sum_{t=1}^{k} \lambda_t B_f(x_{t-1}, x_t),$$

where B_f is defined as in Lemma D.2. In particular, it holds $B_f(x_{t-1}, x_t) \geq 0$.

Proof. Note that for this proof, we need an additional, and artificial iterate $x_{-1} = x_0$. Summing over $t = 0, \ldots, k$ in (37) we have that

$$\begin{aligned}
&\sum_{t=0}^{k} \left(f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\right) \\
&\quad \overset{(37)}{=} \sum_{t=0}^{k} (1+\lambda_t)(f(x_t) - f(x_*)) - \lambda_t(f(x_{t-1}) - f(x_*)) + \sum_{t=0}^{k} \lambda_t B_f(x_{t-1}, x_t) \\
&\quad = \sum_{t=0}^{k} \lambda_{t+1}(f(x_t) - f(x_*)) - \lambda_t(f(x_{t-1}) - f(x_*)) + \sum_{t=0}^{k} \lambda_t B_f(x_{t-1}, x_t) \\
&\quad = (k+1)[f(x_k) - f(x_*)] + \sum_{t=1}^{k} \lambda_t B_f(x_{t-1}, x_t),
\end{aligned}$$

where the second step used $1 + \lambda_t = 1 + t = \lambda_{t+1}$, and the last step we used telescoping and the fact that $\lambda_0 = 0$. $\square$

D.6. Proof of Theorem 3.2

Theorem 3.2 (Non-smooth setting)**.** Consider the iterates of `IAM` in Algorithm 1 with the learning rate (19) and $\lambda_t = t$. Let f_ξ be convex for all ξ. Let $D := \|x_0 - x_*\|$,

$$\begin{aligned}
G^2 &:= \max_{x \in \mathbb{B}_D(x_*)} \mathbb{E}_\xi \|g_\xi(x)\|^2, \\
B_f(x, y) &:= f(x) - f(y) - \langle \nabla f(y), x - y\rangle .
\end{aligned}$$

The suboptimality of the *last iterate* x_T is bounded by

$$\begin{aligned}
&\mathbb{E}\left[f(x_T) - f(x_*)\right] + \frac{1}{T+1}\sum_{t=1}^{T} t\,\mathbb{E}\left[B_f(x_{t-1}, x_t)\right] \\
&\qquad \leq \frac{GD}{\sqrt{T+1}}. \qquad (20)
\end{aligned}$$

Proof. We start by applying Lemma D.1, which states that $x_t \in D$ and $z_t \in D$ almost surely for all $t \geq 0$. Further, Lemma D.1, (ii) implies that

$$\mathbb{E}\left[\|z_t - x_*\|^2 \mid \mathcal{F}_t\right] \leq \|z_{t-1} - x_*\|^2 - \frac{\left(f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\right)_+^2}{\mathbb{E}\left[\|g_t\|^2 \mid \mathcal{F}_t\right]}. \qquad (39)$$

For the denominator of (39), we can therefore estimate

$$\mathbb{E}\left[\|g_t\|^2 \mid \mathcal{F}_t\right] \leq G^2.$$

Applying expectation, and summing from $t = 0, \ldots, k$ (recall that $z_{-1} = x_0$), we get

$$\sum_{t=0}^{k} \mathbb{E}\left[\left(f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\right)_+^2\right] \leq G^2\left[\|x_0 - x_*\|^2 - \mathbb{E}\left[\|z_k - x_*\|^2\right]\right]. \qquad (40)$$

Now, applying (25) with $b_t = 1$ we get for any $a_0, \dots, a_k$ that $\sum_{t=0}^k (a_t)_+^2 \geq \frac{1}{k+1}\big(\sum_{t=0}^k a_t\big)_+^2$. Therefore, we conclude

$$\begin{aligned}\sum_{t=0}^k \big(f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\big)_+^2 &\geq \frac{1}{k+1}\Big(\sum_{t=0}^k f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle\Big)_+^2 \\ &\geq \frac{1}{k+1}\Big((k+1)[f(x_k) - f(x_*)] + \sum_{t=1}^k \lambda_t B_f(x_{t-1}, x_t)\Big)_+^2 \\ &= \Big(\sqrt{k+1}[f(x_k) - f(x_*)] + \sum_{t=1}^k \frac{\lambda_t}{\sqrt{k+1}} B_f(x_{t-1}, x_t)\Big)^2,\end{aligned}$$

where we used Lemma D.3 in the second step, and non-negativity of all terms in the third step. Define $\bar{B}_k := \sum_{t=1}^k \lambda_t B_f(x_{t-1}, x_t) \geq 0$. Plugging this into (40), we get

$$\mathbb{E}\left[\Big(\sqrt{k+1}[f(x_k) - f(x_*)] + \frac{1}{\sqrt{k+1}}\bar{B}_k\Big)^2\right] \leq G^2\left[\|x_0 - x_*\|^2 - \mathbb{E}\left[\|z_k - x_*\|^2\right]\right].$$

Now, using Jensen's inequality $\mathbb{E}[X]^2 \leq \mathbb{E}\left[X^2\right]$, taking the square-root, and dividing by $\sqrt{k+1}$, we finally obtain

$$\mathbb{E}[f(x_k) - f(x_*)] + \frac{1}{k+1}\mathbb{E}\left[\bar{B}_k\right] \leq \frac{G\|x_0 - x_*\|}{\sqrt{k+1}}.$$

□

D.7. Proof of Theorem 3.3

Theorem 3.3 (Smooth setting)**.** Let f_ξ be convex for all ξ. Assume local *expected smoothness* (13) holds. Let x_t be the iterates of Algorithm 1 (`IAM`) with $\lambda_t = t$. It holds

$$\begin{aligned}\mathbb{E}[f(x_{T-1}) - f(x_*)] \leq &\frac{2L\|x_0 - x_*\|^2(\log(T) + 1)}{T} \\ &+ \frac{\sqrt{2L\sigma_*^2}\|x_0 - x_*\|}{\sqrt{T}}.\end{aligned} \tag{21}$$

Proof. We start the proof by applying Lemma D.1, (iii), which yields

$$\mathbb{E}\left[\|z_k - x_*\|^2\right] \leq \|z_0 - x_*\|^2 - \frac{\big(\sum_{t=0}^k \mathbb{E}[f(x_t) - f(x_*) + \langle \nabla f(x_t), z_{t-1} - x_t\rangle]\big)_+^2}{\sum_{t=0}^k \mathbb{E}[\|g_t\|^2]}.$$

For the nominator of the last term, use Lemma D.3 and the fact that $(\cdot)_+^2$ is monotonic to obtain

$$\begin{aligned}\big(\sum_{t=0}^k \mathbb{E}[f(x_t) - f(x_*) + \langle \nabla f(x_t), z_{t-1} - x_t\rangle]\big)_+^2 &\geq \big(\mathbb{E}[(k+1)(f(x_k) - f(x_*)]\big)_+^2 \\ &= (k+1)^2\mathbb{E}[f(x_k) - f(x_*)]^2.\end{aligned}$$

For the denominator, observe that $\mathbb{E}\left[\|g_t\|^2\right] \leq 2L\mathbb{E}\left[(f(x_t) - f(x_*) + \sigma_*^2)\right]$. Thus, using $z_0 = x_0$, we get

$$\mathbb{E}\left[\|z_k - x_*\|^2\right] \leq \|x_0 - x_*\|^2 - \frac{(k+1)^2\mathbb{E}[(f(x_k) - f(x_*))]^2}{2L\sum_{t=0}^k \mathbb{E}[(f(x_t) - f(x_*) + \sigma_*^2)]}.$$

Let $c_t = \mathbb{E}[f(x_t) - f(x_*)] + \sigma_*^2$ and $S_k = \sum_{t=0}^k c_t$, then we can rewrite the above as

$$\frac{\mathbb{E}[(f(x_k) - f(x_*))]^2}{S_k} \leq \frac{2L}{(k+1)^2}(\|x_0 - x_*\|^2 - \mathbb{E}\left[\|z_k - x_*\|^2\right]) \leq \frac{2L\|x_0 - x_*\|^2}{(k+1)^2}.$$

Taking the square-root yields

$$\frac{\mathbb{E}\left[f(x_k)-f(x_*)\right]}{\sqrt{S_k}} \leq \frac{\sqrt{2L}\|x_0-x_*\|}{k+1}. \tag{41}$$

Finally, notice that $\mathbb{E}\left[f(x_k)-f(x_*)\right]=c_k-\sigma_*^2$, so we arrive at the inequality

$$\frac{c_t}{\sqrt{S_t}} \leq \frac{\sqrt{2L}\|x_0-x_*\|}{t+1}+\frac{\sigma_*^2}{\sqrt{S_t}}.$$

Summing this from $t=0$ to k and then applying $\sum_{t=0}^{k}\frac{1}{t+1}\leq \log(k+1)+1$ and $S_t \geq (t+1)\sigma_*^2$ gives

$$\begin{aligned}\sqrt{S_k} &\overset{(23)}{\leq} \sum_{t=0}^{k}\frac{c_t}{\sqrt{S_t}} \leq \sum_{t=0}^{k}\frac{\sqrt{2L}\|x_0-x_*\|}{t+1}+\sum_{t=0}^{k}\frac{\sigma_*^2}{\sqrt{S_t}} \\ &\leq \sqrt{2L}\|x_0-x_*\|(\log(k+1)+1)+\sum_{t=0}^{k}\frac{\sqrt{\sigma_*^2}}{\sqrt{t+1}}.\end{aligned}$$

Furthermore, it holds $\sum_{t=0}^{k}\frac{1}{\sqrt{t+1}}\leq 2\sqrt{k+1}$, so we finally get

$$\sqrt{S_k} \leq \sqrt{2L}\|x_0-x_*\|(\log(k+1)+1)+\sqrt{\sigma_*^2}\sqrt{k+1}.$$

Using the above inequalities in (41) gives

$$\mathbb{E}\left[f(x_k)-f(x_*)\right] \leq \frac{\sqrt{2L}\|x_0-x_*\|\sqrt{S_k}}{k+1} \leq \frac{2L\|x_0-x_*\|^2(\log(k+1)+1)}{k+1}+\frac{\sqrt{2L\sigma_*^2}\|x_0-x_*\|}{\sqrt{k+1}}.$$

□

E. Complexity of SGD with Adaptivity to Interpolation

Theorem E.1 (Complexity of SGD)**.** Let $f=\frac{1}{n}\sum_{i=1}^{n}f_i$ where each $f_i:\mathbb{R}^d\to\mathbb{R}$ is convex and L-smooth, and assume that f admits a minimizer, noted x_*. Let $x_0\in\mathbb{R}^d$, and note $D:=\|x_0-x_*\|$ and $\sigma_*^2=\mathbb{E}\left[\|\nabla f_i(x_*)\|^2\right]$. Let $T\geq 1$, let $\gamma=\frac{\gamma_0}{\sqrt{\sigma_*^2 T+1}}$ where $\gamma_0\leq\frac{1}{4L}$, and let $(x_t)_{t=0}^{T}$ be the sequence generated by the SGD algorithm with constant stepsize γ. Then

$$\mathbb{E}\left[f(\bar{x}_T)-\inf f\right] \leq \frac{D^2}{\gamma_0 T}+\frac{\sigma_*^2}{\sqrt{T}}\left(\frac{D^2}{\gamma_0}+2\gamma_0\right),$$

where $\bar{x}_T=\frac{1}{T}\sum_{t=0}^{T-1}x_t$.

The above theorem shows that SGD enjoys a $\mathcal{O}\left(1/T+\sigma_*^2/\sqrt{T}\right)$ complexity, which is similar to the result of SPS* in Corollary 2.3. But there are some important differences. First, SPS* has an anytime convergence rate valid for every T, while SGD has a complexity rate: it is a "finite horizon" rate where the horizon must be known before setting the stepsize. The other difference is that for SGD to achieve this complexity, we need access to both the smoothness constant L and the interpolation constant σ_*. Whereas SPS* adapts to both smoothness and non-smoothness. Though to achieve this SPS* requires access to $f_\xi(x_*)$,

Proof. We are using the complexity rate of SGD with convex functions from (Garrigos & Gower, 2023). To be able to use this result, we need the stepsize γ to verify $\gamma<1/2L$. Here we have $\gamma\leq\gamma_0\leq 1/4L$ so we are good to go. From (Garrigos & Gower, 2023, Thm. 5.5) we get

$$(1-2\gamma L)\mathbb{E}\left[f(\bar{x}_T)-\inf f\right] \leq \frac{D^2}{2\gamma T}+\gamma\sigma_*^2.$$

Now it is just a matter of cleaning the constants. First, use again the fact that $\gamma \leq 1/4L$ to see that $(1-2\gamma L) \geq 1/2$. Second, write

$$\frac{1}{\gamma} = \frac{\sqrt{1+\sigma_*^2 T}}{\gamma_0} \leq \frac{1+\sigma_*^2\sqrt{T}}{\gamma_0} \quad \text{and} \quad \gamma\sigma_*^2 = \frac{\gamma_0\sigma_*^2}{\sqrt{1+\sigma_*^2 T}} \leq \frac{\gamma_0\sigma_*^2}{\sqrt{T}}.$$

It remains to combine all the above inequalities to conclude that

$$\mathbb{E}\left[f(\bar{x}_T) - \inf f\right] \leq \frac{D^2}{\gamma T} + 2\gamma\sigma_*^2 \leq \frac{D^2(1+\sigma_*^2\sqrt{T})}{\gamma_0 T} + 2\frac{\gamma_0\sigma_*^2}{\sqrt{T}} = \frac{D^2}{\gamma_0 T} + \frac{D^2\sigma_*^2}{\gamma_0\sqrt{T}} + 2\frac{\gamma_0\sigma_*^2}{\sqrt{T}}.$$

□

F. Detailed Comparison of SPS* Convergence in Smooth Case

In the smooth setting, our result in Corollary 2.3 relies on the expected smoothness bound, which is a generalization over assuming that the f_ξ is almost surely L–smooth, see Gower et al. (2020; 2019), and Lemma C.5. In the smooth setting, proofs of convergence for SGD hold by assuming that (13) holds *globally* (Gower et al., 2021; 2019).

Our smooth result in Corollary 2.3 is, as far as we know, the first $\mathcal{O}\left(1/\sqrt{T}\right)$ anytime convergence rate for a stochastic variant of the Polyak stepsize. To contrast our result, both [Thm. 3.4](Loizou et al., 2021) and [Thm. 8.3](Garrigos & Gower, 2023) establish a $\mathcal{O}\left(1/T\right)$ convergence up to a distance to the solution proportional to σ_*^2. For example, in [Theorem 8.3](Garrigos & Gower, 2023) the authors show that

$$\mathbb{E}\left[f(\bar{x}_T) - f(x_*)\right] \leq \frac{2L}{T+1}\|x_0 - x_*\|^2 + \sigma_*^2$$

where $\bar{x}_T = \frac{1}{T}\sum_{t=0}^{T-1} x_t$. Because of this constant factor of σ_*^2 cannot be controlled, the above result cannot be converted into a complexity result. This is in contrast to recent work on the NGN variant (Orvieto & Xiao, 2024), which does establish a $\mathcal{O}(1/\sqrt{T})$ complexity.

Another interesting aspect of the convergence rate in (12) is that it is adaptive to interpolation. To see this, consider the case that $\sigma_*^2 > 0$ (no interpolation). In this case, have a $\mathcal{O}(1/\sqrt{T})$ anytime rate which is compared to the $\mathcal{O}(\log(T)/\sqrt{T})$ rate of SGD. On the other hand, as σ_*^2 gets closer to zero, the convergence rate in (12) approaches $\mathcal{O}(1/T)$, which is the expected accelerated rate of SGD under interpolation (Vaswani et al., 2019). We are unaware of prior work that establishes an anytime rate of convergence that is adaptive to interpolation.

We can also compare Theorem 3.3 to the best *anytime* convergence of SGD. That is, consider the iterates given by (15) when $\beta = 0$. For SGD with a learning rate of $\eta_t = \eta/\sqrt{t+1}$ where $\eta \leq \frac{1}{2L}$, the average iterate converges according to Garrigos & Gower (2023, Thm. 5.5):

$$\mathbb{E}\left[f(\bar{x}_T) - f(x_*)\right] \leq \frac{\|x_0 - x_*\|^2}{2\eta\sqrt{T+1}} + \frac{\eta\log(T+1)}{\sqrt{T+1}}\sigma_*^2. \tag{42}$$

where $\bar{x}_T := \sum_{t=0}^{T-1} p_{T,t}x_t$, with $p_{T,t} := \frac{\eta_T(1-2\eta_T L)}{\sum_{i=0}^{t-1}\eta_i(1-2\eta_i L)}$. In comparison to (42) the analysis in (21) of the IAM method has two advantages: First, there is no additional $\log(T+1)$ term multiplying the dominating $\mathcal{O}(1/\sqrt{T+1})$ term, and it is adaptive to L. That is, IAM does not need access to L to achieve this same anytime convergence. Furthermore (42) is not adaptive to interpolation, in that when $\sigma_*^2 = 0$, the resulting rate of convergence is $\mathcal{O}(1/\sqrt{T+1})$, as opposed to the $\mathcal{O}(1/(T+1))$ rate that can be achieved under interpolation (Ma et al., 2018; Gower et al., 2021).

G. Convergence in (Locally) Strongly Convex Case

If we assume our loss functions is (locally) strongly convex, then we can improve the rate of convergence of SPS* from $\mathcal{O}\left(1/\sqrt{t}\right)$ to $\mathcal{O}\left(1/t\right)$.

Theorem G.1. [Convergence of SPS*] Consider (1) and let the iterates $(x_t)_{t\geq 0}$ be given by (2), and let $D := \|x_0 - x_*\|$.

Assume that f_ξ is convex for any ξ. Let $f(x)$ be convex and satisfy the μ–quadratic growth bound

$$\frac{\mu}{2}\|x - x_*\|^2 \leq f(x) - \inf f, \quad \text{for every } x \in \mathbb{B}(x_*, D) \tag{43}$$

and the expected smoothness bound

$$\mathbb{E}_\xi \left[\|g_\xi(x)\|^2\right] \leq A(f(x) - \inf f) + B, \quad \text{for every } x \in \mathbb{B}(x_*, D). \tag{44}$$

Let $T_0 := \frac{4A}{\mu} \log\left(\frac{D^2\mu^2}{16B}\right)$. It follows that

$$\mathbb{E}\|x_t - x_*\|^2 \leq \frac{16B}{\mu^2}\frac{1}{t+1-T_0}, \quad \forall t \geq \frac{2A}{\mu}\left(2\log\left(\frac{D^2\mu^2}{16B}\right)+1\right). \tag{45}$$

In the non-smooth setting where $A = 0$ and $B = G^2$ we get

$$\mathbb{E}\left[\|x_t - x_*\|^2\right] \leq \frac{16G^2}{\mu^2}\frac{1}{t+1}, \quad \text{for } t \geq 0. \tag{46}$$

This matches the rate given by Pedregosa & Schaipp (2023) for the finite sum setting upto a factor of 4.

In the smooth setting where $A = 4L$ and $B = \sigma_*^2$ we get

$$\mathbb{E}\left[\|x_t - x_*\|^2\right] \leq \frac{64\sigma_*^2}{\mu^2}\frac{1}{t+1-T_0}, \quad \text{for } t \geq \frac{8L}{\mu}\left(2\log\left(\frac{D^2\mu^2}{16\sigma_*^2}\right)+1\right). \tag{47}$$

Proof. Let $\delta_t := \mathbb{E}\left[\|x_t - x_*\|^2\right]$. We start the proof from (33), which we repeat here for convenience:

$$\psi(r_t) \leq \delta_t - \delta_{t+1}, \tag{48}$$

where $r_t = \mathbb{E}[f(x_t) - f(x_*)]$ and $\psi(r) := \frac{r^2}{Ar+B}$ for $r \geq 0$. Due to the monotonicity of the iterates (8) we have that $\delta_t - \delta_{t+1} \geq 0$. Applying Lemma C.1 together with (34) gives

$$\begin{aligned} \mathbb{E}[f(x_t) - f(x_*)] &\leq \psi^{-1}(\delta_t - \delta_{t+1}) \\ &\leq A(\delta_t - \delta_{t+1}) + \sqrt{B(\delta_t - \delta_{t+1})}. \end{aligned}$$

Using the quadratic growth bound $\frac{\mu}{2}\|x_t - x_*\|^2 \leq f(x_t) - f(x_*)$ gives

$$\frac{\mu}{2}\delta_t \leq A\left(\delta_t - \delta_{t+1}\right) + \sqrt{B\left(\delta_t - \delta_{t+1}\right)}. \tag{49}$$

Our proofs will consider two cases by comparing the two terms on the right hand side of (49). To this end, note that

$$A\left(\delta_t - \delta_{t+1}\right) \leq \sqrt{B\left(\delta_t - \delta_{t+1}\right)} \quad \iff \quad \delta_t - \delta_{t+1} \leq \frac{B}{A^2}. \tag{50}$$

The remainder of the proof is divided into two parts. First we show that for $t_0 := \lceil \frac{4A}{\mu}\log\left(\frac{D^2\mu^2}{16B}\right)\rceil$, we have that $\delta_t \leq \frac{16B}{\mu^2}$. For the second part we prove by induction that for $t \geq t_0$ $\delta_{t+1} \leq \frac{16B}{\mu^2}\frac{1}{t+1}$, where the first part will serve as the base case of the induction.

Base case: First we prove that for all $t \geq \frac{4A}{\mu}\log\left(\frac{D^2\mu^2}{16B}\right)$ we have that $\delta_t \leq \frac{16B}{\mu^2}$. We divide this proof also into two cases based on the comparison (50). If $\delta_t - \delta_{t+1} \leq \frac{B}{A^2}$ for any $t < \frac{4A}{\mu}\log\left(\frac{D^2\mu^2}{16B}\right)$ then by (49) and (50) we have that

$$\begin{aligned} &\frac{\mu}{2}\delta_t \leq A\left(\delta_t - \delta_{t+1}\right) + \sqrt{B\left(\delta_t - \delta_{t+1}\right)} \leq 2\sqrt{B\left(\delta_t - \delta_{t+1}\right)} \leq 2\sqrt{B\delta_t} \quad \implies \\ &\delta_t \leq \frac{16B}{\mu^2}, \end{aligned} \tag{51}$$

which would prove our result.

Alternatively, suppose that $\delta_t - \delta_{t+1} \geq \frac{B}{A^2}$ for every $t \leq \frac{4A}{\mu} \log\left(\frac{D^2\mu^2}{16B}\right)$. By (49) and (50) we have that

$$\frac{\mu}{2}\delta_t \leq A\left(\delta_t - \delta_{t+1}\right) + \sqrt{B\left(\delta_t - \delta_{t+1}\right)} \leq 2A\left(\delta_t - \delta_{t+1}\right). \tag{52}$$

Re-arranging the above gives

$$\delta_{t+1} \leq \left(1 - \frac{\mu}{4A}\right)\delta_t. \tag{53}$$

Unrolling this for every $t \leq \frac{4A}{\mu} \log\left(\frac{D^2\mu^2}{16B}\right)$ gives

$$\delta_t \leq \left(1 - \frac{\mu}{4A}\right)^t \delta_0. \tag{54}$$

It now follows by taking logarithm and using standard techniques (for example Lemma A.2 in Garrigos & Gower (2023)) that

$$t \geq \frac{4A}{\mu}\log\left(\frac{D^2\mu^2}{16B}\right) \quad \Longrightarrow \quad \delta_t \leq \left(1 - \frac{\mu}{4A}\right)^t \delta_0 \leq \frac{16B}{\mu^2}.$$

Induction step: Now, for ease of notation, let us re-name our iterates so that δ_0 is the first iterate for which $\delta_0 \leq \frac{16B}{\mu^2}$.

If $\delta_t - \delta_{t+1} \leq \frac{B}{A^2}$ then by (49) and (50) we have that

$$\begin{aligned}
&\frac{\mu}{2}\delta_t \leq A\left(\delta_t - \delta_{t+1}\right) + \sqrt{B\left(\delta_t - \delta_{t+1}\right)} \leq 2\sqrt{B\left(\delta_t - \delta_{t+1}\right)} \quad \Leftrightarrow \\
&\frac{\mu^2}{4}\delta_t^2 \leq 4B\left(\delta_t - \delta_{t+1}\right) \quad \Leftrightarrow \\
&\delta_{t+1} \leq (1 - \frac{\mu^2}{16B}\delta_t)\delta_t.
\end{aligned} \tag{55}$$

Let $a_t = \frac{\mu^2}{16B}\delta_t$. Multiplying both sides of (55) by $\frac{\mu^2}{16B}$ and using the induction hypothesis

$$a_t = \frac{\mu^2}{16B}\delta_t \leq \frac{\mu^2}{16B}\frac{16B}{\mu^2}\frac{1}{t+1} = \frac{1}{t+1}$$

gives

$$a_{t+1} \leq (1 - a_t)a_t \leq \max_{x \in [0, \frac{1}{t+1}]} (1-x)x = \left(1 - \frac{1}{t+1}\right)\frac{1}{t+1} \leq \frac{1}{t+2}.$$

Alternatively if $\delta_t - \delta_{t+1} \geq \frac{B}{A^2}$ then by (49) and (50) we have that

$$\frac{\mu}{2}\delta_t \leq A\left(\delta_t - \delta_{t+1}\right) + \sqrt{B\left(\delta_t - \delta_{t+1}\right)} \leq 2A\left(\delta_t - \delta_{t+1}\right). \tag{56}$$

Re-arranging the above gives

$$\delta_{t+1} \leq \left(1 - \frac{\mu}{4A}\right)\delta_t.$$

Using the induction hypothesis and $t \geq \frac{2A}{\mu}$ we have that

$$\delta_{t+1} \leq \left(1 - \frac{\mu}{4A}\right)\delta_t \leq \left(1 - \frac{\mu}{4A}\right)\frac{16B}{\mu^2}\frac{1}{t+1} \leq \frac{16B}{\mu^2}\frac{1}{t+2}$$

where the last inequality follows from

$$\left(1 - \frac{\mu}{4A}\right)\frac{1}{t+1} \leq \frac{1}{t+2} \quad \Leftrightarrow \quad t \geq \frac{2A}{\mu} - 2 \quad \Leftarrow \quad t \geq \frac{2A}{\mu}.$$

□

H. Approximating SPS* and Safe-guards

In practice, outside of the interpolation regime, it is unlikely that we would have access to $f_\xi(x_*)$. To derive a practical method that is more generally applicable, we would need to estimate $f_\xi(x_*)$. Let us call this estimate ℓ_ξ^*, and consider the step size

$$\gamma_t^{\texttt{SPS}} := \frac{(f_\xi(x_t) - \ell_\xi^*)_+}{\|g_t\|^2}. \tag{57}$$

The estimates ℓ_ξ^* would have to be *underestimates*, otherwise the resulting method would stop early. Indeed, as $x_t \to x_*$ we have that $f_\xi(x_t) \to f_\xi(x_*)$, and the step size (57) would be zero before reaching convergence.

There are two natural underestimates for $f_\xi(x_*)$. The first is to use $\inf f_\xi$. This is the approach used in SPSmax (Loizou et al., 2021). The advantage of using $\inf f_\xi$ is that it often can be computed, indeed if no weight decay is being used (no L2 regularization), then often $\inf f_\xi = 0$. Which brings us to the second approach, which is to simply use 0 as an underestimate , which holds for the ubiquitous case of having a positive loss (Berrada et al., 2020; Orvieto & Xiao, 2024).

An issue with using an underestimate is that the step size (57) can become too large, potentially even being unbounded if $\|g_t\| \to 0$, which could lead to divergence.

To safeguard against taking exceeding large step sizes, we can use clipping (Loizou et al., 2021), dampening (Orvieto & Xiao, 2024), or a combination of both (Berrada et al., 2020). By clipping, we mean to take the minimum between the stepsize in (57) and a hyperparameter $\gamma_b > 0$ as is done in Loizou et al. (2021) in the $\texttt{SPS}_{\max}$ method[3]

$$\gamma_t^{\texttt{SPS}_{\max}} := \min\left\{\frac{(f_\xi(x_t) - \ell_\xi^*)_+}{\|g_t\|^2}, \gamma_b\right\}. \tag{58}$$

We refer to dampening by adding an additional constant ϵ to the denominator, as is done in Orvieto & Xiao (2024), Gower et al. (2022) and Berrada et al. (2020):

$$\gamma_t^{\texttt{SPS}dam} := \frac{(f_\xi(x_t) - \ell_\xi^*)_+}{\|g_t\|^2 + \epsilon}. \tag{59}$$

In particular in Orvieto & Xiao (2024), this dampening parameter depends in the iteration and is proportional to $f_\xi(x_t)$.

Thus we can view several practical variants of SPS as approximations of SPS*, where $f_\xi(x_*)$ is replaced by an underestimate, and a further safeguard is included to avoid large step sizes. These safeguards can also be motivated through a variational viewpoint based on solving relaxations of the interpolation condition (Gower et al., 2022).

I. Momentum and Iterate Averaging

Here we detail the relationship between momentum and iterate averaging, which hinges on the following lemma.

Lemma I.1. (Garrigos & Gower (2023, Lemma 7.3) and Defazio & Gower (2021, Theorem 1)) The iterates $(x_t)_{t\geq 0}$ generated by (15) and the *iterate-moving-average* (IAM) are equivalent to if $z_{-1} = x_0$, $m_{-1} = 0$ and the (γ_t, β_t) parameters of momentum and the IAM parameters (η_t, λ_t) satisfy

$$\beta_t = \frac{\lambda_t}{1+\lambda_t}\frac{\eta_{t-1}}{\eta_t}, \quad \text{and} \quad \gamma_t = \frac{\eta_t}{1+\lambda_{t+1}}, \quad \forall t \geq 0. \tag{60}$$

As an example of using the above lemma, a constant learning rate $\eta_t \equiv \eta$ and $\lambda_t = t$ in the IAM method (16–17) corresponds to a decreasing learning rate $\gamma_t = \frac{\eta}{1+t}$ and an increasing momentum $\beta_t = \frac{t}{1+t}$ in the momentum method (15).

Proof. The proof is by induction. Our induction hypothesis is that x_t iterates in (17) and (15) are equivalent upto step t and that the z_t iterates in (16) and m_t in (15) satisfy

$$z_t = x_t - (1+\lambda_{t+1})\gamma_t m_t. \tag{61}$$

[3] Though $\texttt{SPS}_{\max}$ has an additional constant c in $\frac{(f_\xi(x_t) - \ell_\xi^*)_+}{c\|g_t\|^2}$.

For the base case $t = 0$ we have from (16) that

$$\begin{aligned} z_0 &= z_{-1} - \eta_0 g_0 \\ &= x_0 - (1+\lambda_1)\gamma_0 g_0, \end{aligned}$$

where in the second equality we used $z_{-1} = x_0$ and (60). Since $m_{-1} = 0$, we have from (15) that $m_0 = g_0$, which proves (61) for the base case. As for the x_t iterates in (17) and (15) being equivalent for $t = 0$ from (17) and (61) we have that

$$\begin{aligned} x_1 &= \frac{\lambda_1}{1+\lambda_1} x_0 + \frac{1}{1+\lambda_1} z_0 \\ &= \frac{\lambda_1}{1+\lambda_1} x_0 - \frac{1}{1+\lambda_1}(x_0 - (1+\lambda_1)\gamma_0 m_0) \\ &= x_0 - \gamma_0 m_0, \end{aligned}$$

which is equivalent to the first step of (15).

Suppose now that x_t iterates in (17) and (15) are equivalent and (61) holds upto time t. From (16) at step $t+1$ we have that

$$\begin{aligned} z_{t+1} &= z_t - \eta_{t+1} g_{t+1} \\ &= x_t - (1+\lambda_{t+1})\gamma_t m_t - \eta_{t+1} g_{t+1}. && \text{Using (61)} \\ &= x_t - (1+\lambda_{t+1})\gamma_t m_t - (1+\lambda_{t+2})\gamma_{t+1} g_{t+1} && \text{Using (60)} \\ &= x_t - \gamma_t m_t + (1+\lambda_{t+2})\gamma_{t+1}\left(\frac{\lambda_{t+1}}{1+\lambda_{t+2}}\frac{\gamma_t}{\gamma_{t+1}} m_t + g_{t+1}\right) \\ &= x_{t+1} - (1+\lambda_{t+2})\gamma_{t+1}\left(\beta_{t+1} m_t + g_{t+1}\right) && \text{Using (17) and (60)} \\ &= x_{t+1} - (1+\lambda_{t+2})\gamma_{t+1} m_{t+1} && \text{Using (15)}, \end{aligned}$$

which shows that (61) holds at time $t+1$. Finally $t+1$ step. From (17) and (16) we have that

$$\begin{aligned} x_{t+1} &= \frac{\lambda_{t+1}}{1+\lambda_{t+1}} x_t + \frac{1}{1+\lambda_{t+1}} z_t \\ &= \frac{\lambda_{t+1}}{1+\lambda_{t+1}} x_t + \frac{1}{1+\lambda_{t+1}}(x_t - (1+\lambda_{t+1})\gamma_t m_t) \\ &= x_t - \gamma_t m_t, \end{aligned}$$

which is equivalent to (15), and thus concludes the proof. □

J. Additional Proof for `IAM` with Decreasing λ_t

Theorem J.1. Consider the setting of Theorem 3.2, except that $\lambda_0 = 0$ and $(\lambda_t)_{t=1}^k$ is any decreasing sequence of nonnegative reals starting. It follows that

$$\mathbb{E}[f(\overline{x}_k) - f(x_*)] + \frac{1}{k+1}\sum_{t=0}^{k} \lambda_t \mathbb{E}[B_f(x_{t-1}, x_t)] \leq \frac{G\|x_0 - x_*\|}{\sqrt{k+1}} + \frac{\lambda_1}{k+1}\mathbb{E}[f(x_0) - f(x_*)].$$

The advantage of this result is that it holds for any constant $\lambda_t = \lambda$. Translating this to the momentum method (15), this allows for other parameter setting of (γ_t, β_t). In particular the setting $\lambda_t = \lambda = 0$ which corresponds to no momentum. In this setting we retrieve the exact same rate as the `SPS*` method in Corollary 2.2. However, as mentioned earlier the price we need to pay for this, is that this result holds for the Cesaro average and not of the last iterate.

Proof. Starting from Lemma D.1:

$$\|z_t - x_*\|^2 \leq \|z_{t-1} - x_*\|^2 - \frac{\big(f(x_t) - f(x_*) + \langle g_t, z_{t-1} - x_t\rangle\big)_+^2}{\|g_t\|^2}. \tag{62}$$

Taking expectation, and using our extended Titu's Lemma C.3 and Bregman viewpoint Lemma D.2 to get

$$\begin{aligned}
\mathbb{E}\|z_t - x_*\|^2 &\leq \mathbb{E}\|z_{t-1} - x_*\|^2 - \frac{\mathbb{E}[f(x_t) - f(x_*) + \langle g(x_t), z_{t-1} - x_t\rangle]_+^2}{\mathbb{E}\|g_t\|^2} \\
&\leq \mathbb{E}\|z_{t-1} - x_*\|^2 - \frac{\mathbb{E}[(1+\lambda_t)[f(x_t) - f(x_*)] - \lambda_t[f(x_{t-1}) - f(x_*)] + \lambda_t B_f(x_{t-1}, x_t)]_+^2}{\mathbb{E}\|g_t\|^2} \\
&\leq \mathbb{E}\|z_{t-1} - x_*\|^2 - \frac{\mathbb{E}[(1+\lambda_t)[f(x_t) - f(x_*)] - \lambda_t[f(x_{t-1}) - f(x_*)] + \lambda_t B_f(x_{t-1}, x_t)]_+^2}{G^2}.
\end{aligned}$$

Multiplying through by G^2 gives

$$\mathbb{E}[(1+\lambda_t)[f(x_t) - f(x_*)] - \lambda_t[f(x_{t-1}) - f(x_*)] + \lambda_t B_f(x_{t-1}, x_t)]_+^2 \leq G^2\mathbb{E}\|z_{t-1} - x_*\|^2 - G^2\mathbb{E}\|z_t - x_*\|^2.$$

Now let $\Delta_t = (1+\lambda_t)[f(x_t) - f(x_*)] - \lambda_t[f(x_{t-1}) - f(x_*)] + \lambda_t B_f(x_{t-1}, x_t)$. Averaging both sides of the above over $t = 0, \ldots, k$, telescoping terms, and using Jensen's inequality with respect to the convex function $x \mapsto (x_+)^2$ gives

$$\begin{aligned}
\frac{G^2\|x_0 - x_*\|^2}{k+1} &\geq \frac{G^2}{k+1}\left(\mathbb{E}\|x_0 - x_*\|^2 - \mathbb{E}\|z_{k+1} - x_*\|^2\right) \\
&\geq \frac{1}{k+1}\sum_{t=0}^{k}\mathbb{E}[\Delta_t]_+^2 \\
&\geq \left(\frac{1}{k+1}\sum_{t=0}^{k}\mathbb{E}[\Delta_t]\right)_+^2.
\end{aligned}$$

Taking the square root gives

$$\left(\frac{1}{k+1}\sum_{t=0}^{k}\mathbb{E}[\Delta_t]\right)_+ \leq \frac{G\|x_0 - x_*\|}{\sqrt{k+1}}. \tag{63}$$

Now since (λ_t) is decreasing and using Jensen's inequality with respect to $x \mapsto f(x)$ we have that

$$\begin{aligned}
\sum_{t=0}^{k}\mathbb{E}[\Delta_t] &= \sum_{t=0}^{k}(1+\lambda_t)\mathbb{E}[f(x_t) - f(x_*)] - \lambda_t\mathbb{E}[f(x_{t-1}) - f(x_*)] + \lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] \\
&= \sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + \sum_{t=0}^{k}\mathbb{E}[f(x_t) - f(x_*)] + \sum_{t=0}^{k}\lambda_t\mathbb{E}[f(x_t) - f(x_*)] - \sum_{t=0}^{k}\lambda_t\mathbb{E}[f(x_{t-1}) - f(x_*)] \\
&= \sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + \sum_{t=0}^{k}\mathbb{E}[f(x_t) - f(x_*)] + \sum_{t=1}^{k}\lambda_t\mathbb{E}[f(x_t) - f(x_*)] - \sum_{t=1}^{k}\lambda_t\mathbb{E}[f(x_{t-1}) - f(x_*)] \\
&= \sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + \sum_{t=0}^{k}\mathbb{E}[f(x_t) - f(x_*)] + \sum_{t=1}^{k-1}(\lambda_t - \lambda_{t+1})\mathbb{E}[f(x_t) - f(x_*)] \\
&\quad + \lambda_k\mathbb{E}[f(x_k) - f(x_*)] - \lambda_1\mathbb{E}[f(x_0) - f(x_*)] \\
&\geq \sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + \sum_{t=0}^{k}\mathbb{E}[f(x_t) - f(x_*)] - \lambda_1\mathbb{E}[f(x_0) - f(x_*)] \\
&\geq \sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + (k+1)\mathbb{E}[f(\overline{x}_k) - f(x_*)] - \lambda_1\mathbb{E}[f(x_0) - f(x_*)],
\end{aligned}$$

where we used that $\lambda_0 = 0$ and $\lambda_t - \lambda_{t+1} \geq 0$ since (λ_t) is a decreasing sequence. Dividing through by $(k+1)$ gives

$$\frac{1}{k+1}\sum_{t=0}^{k}\mathbb{E}[\Delta_t] \geq \frac{1}{k+1}\sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1}, x_t)] + \mathbb{E}[f(\overline{x}_k)\ - f(x_*)] - \frac{\lambda_1}{k+1}\mathbb{E}[f(x_0) - f(x_*)].$$

Using the above and (63) gives

$$\left(\frac{1}{k+1}\sum_{t=0}^{k}\lambda_t\mathbb{E}[B_f(x_{t-1},x_t)]+\mathbb{E}[f(\overline{x}_k)-f(x_*)]-\frac{\lambda_1}{k+1}\mathbb{E}[f(x_0)-f(x_*)]\right)\leq\left(\frac{1}{k+1}\sum_{t=0}^{k}\mathbb{E}[\Delta_t]\right)_+ \leq\frac{G\|x_0-x_*\|}{\sqrt{k+1}}.$$

Re-arranging gives the result. □

K. An Adam Variant of IAM

Following an analogous reasoning used in Section 3, we can derive variants of IAM that use preconditioning. This is particularily important for models such as Transformers, where using an Adam preconditioner is required to achieve a reasonable performance.

To arrive at a preconditioned version of IAM, let $\boldsymbol{D}_t\in\mathbb{R}^{d\times d}$ be our positive definite symmetric preconditioner, and let $\|z\|^2_{\boldsymbol{D}_t}:=\langle\boldsymbol{D}_t z,z\rangle$ be the norm induced by this preconditioner. Now consider the iterative averaging method with this preconditioner:

$$z_t = z_{t-1}-\eta_t\boldsymbol{D}_t^{-1}g_t, \tag{64}$$

$$x_{t+1} = \frac{\lambda_{t+1}}{1+\lambda_{t+1}}x_t+\frac{1}{1+\lambda_{t+1}}z_t. \tag{65}$$

Now we upper bound the distance between z_t and a solution x_* under the preconditioned norm via

$$\begin{aligned}\|z_t-x_*\|^2_{\boldsymbol{D}_t} &= \|z_{t-1}-x_*\|^2_{\boldsymbol{D}_t}-2\eta_t\left\langle\boldsymbol{D}_t^{-1}g_t,z_{t-1}-x_*\right\rangle_{\boldsymbol{D}_t}+\eta_t^2\|g_t\|^2_{\boldsymbol{D}_t^{-1}}\\ &= \|z_{t-1}-x_*\|^2_{\boldsymbol{D}_t}-2\eta_t\left\langle g_t,z_{t-1}-x_*\right\rangle+\eta_t^2\|g_t\|^2_{\boldsymbol{D}_t^{-1}}\\ &\leq \|z_{t-1}-x_*\|^2_{\boldsymbol{D}_t}-2\eta_t\big(f_{\xi_t}(x_t)-f_{\xi_t}(x_*)+\langle g_t,z_{t-1}-x_t\rangle\big)+\eta_t^2\|g_t\|^2_{\boldsymbol{D}_t^{-1}},\end{aligned}$$

where in the inequality we used that f_{ξ_t} is convex. Minimizing the right-hand side with respect to η_t now gives the step size given in line 3 in Algorithm 2. To arrive at our IAM-Adam method, we simply set $\boldsymbol{D}_t$ to be the preconditioner used by Adam,

Algorithm 2 IAM-Adam

1: **Input:** $z_{-1}=x_0\in\mathbb{R}^d$, $\lambda_t>0$
2: **for** $t=0$ to $T-1$ **do**
3: $\quad\eta_t = \dfrac{\left[f_{\xi_t}(x_t)-\ell^*_{\xi_t}+\langle g_t,z_{t-1}-x_t\rangle\right]_+}{\|g_t\|^2_{\boldsymbol{D}_t^{-1}}}$,
4: $\quad z_t = z_{t-1}-\eta_t\boldsymbol{D}_t^{-1}g_t$
5: $\quad x_{t+1} = \dfrac{\lambda_{t+1}}{1+\lambda_{t+1}}x_t+\dfrac{1}{1+\lambda_{t+1}}z_t$
6: **Return:** x_T

that is $\boldsymbol{D}_t=\mathtt{diag}(\sqrt{v_t}+\epsilon)$ where

$$v_{t+1}=\beta_2 v_t+(1-\beta_2)g_t\odot g_t.$$

L. Experiments

L.1. Non-Lipschitz Non-smooth Convex Problem

To model discrete events with a Poisson regression, we need to solve

$$\min_{w\in\mathbb{R}^d}\frac{1}{n}\sum_{i=1}\left(\ell(w^\top x_i)-y_i\log\left(\ell(w^\top x_i)\right)\right), \tag{66}$$

where $\ell : \mathbb{R} \mapsto \mathbb{R}$ is called the link function. One of the most commonly used link functions is the exponential function $\ell(z) = \exp z$. With this link function (66) becomes

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1} \left(\exp(w^\top x_i) - y_i w^\top x_i \right). \tag{67}$$

We fit two different data sets. The first data set is on diabetes patients sourced from (Efron et al., 2004), which is a medical dataset containing information on 442 patients (n), each described by 10 physiological and lifestyle features (d). The second data set is a bike sharing records (Fanaee-T & Gama, 2014) in Washington, D.C., over a two-year period (2011-2012). It includes a total of 17,379 data points, and 12 features such as weather conditions, seasonal information, and temporal data. The target variable is the count of total bike rentals on an hourly basis.

As a baseline, we ran `L-BFGS` (Liu & Nocedal, 1989) in full batch mode, and `SGD` with constant learning rate tuned across

$$\gamma \in 0.001 \cdot \{0.01, 0.1, 0.5, 1.0, 2.0, 5.0, 20, 50\}.$$

Each method was given the same budget in terms of epochs. To highlight how important the choice of the learning rate is, in Figure 3 we plot the resulting loss (y-axis) of the last iterate of each method for different learning rates (x-axis). We find that the `IAM` method converges to a loss that is comparable to `LBFGS` and `SGD` with the best possible learning rate. Furthermore, `IAM` is the only method guaranteed to converge on this non-smooth and non-Lipschitz objective.

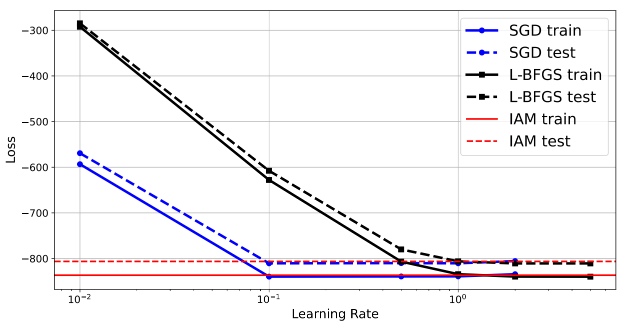


Figure 1. Bike Sharing Data, 7 epochs

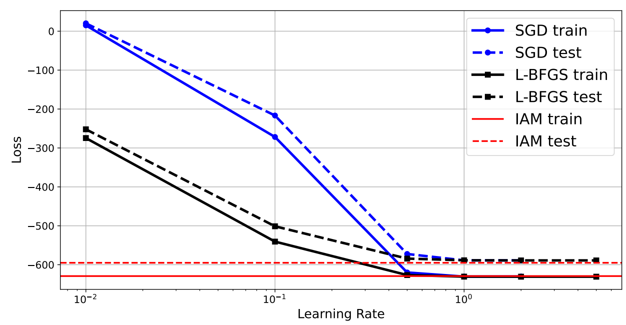


Figure 2. Diabetes Data, 15 epochs

Figure 3. Sensitivity to learning rate for each method. Larger learning rates diverged.

L.2. Misspecification of $f_\xi(x_*)$

In numerous machine learning applications a lower bound of $f_\xi(x_*)$ is known a priori, because loss functions are typically non-negative. We study the following three versions of `IAM`:

- *theoretical* version where we specify correctly $f_{\xi_t}(x_*)$ in every iteration t, computed from the oracle values $f_i^*,\ i \in [n]$,
- *averaged* version, where we specify $f_{\xi_t}(x_*)$ with $f(x_*)$ in every iteration,
- *lower-bound* version, where we specify $f_{\xi_t}(x_*)$ with zero.

Description of experimental setup. Consider the following problem setup, which is adopted from (Orvieto et al., 2022): solve

$$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^n f_i(x), \quad f_i(x) := (x - x_*^i)^T H_i (x - x_*^i) + f_i^*,$$

where $H_i \in \mathbb{R}^{d \times d}$ are symmetric positive definite matrices and $x_*^i \in \mathbb{R}^d$. This is clearly an instance of (1), where $\mathcal{D}$ is the uniform distribution over $[n]$ and $f_\xi(x) = f_i(x)$, and it holds $f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x)$.

We consider two cases, (i) the interpolated case with $x_*^i = \bar{x}$ for all $i \in [n]$, and (ii) $x_*^i = \bar{x} + 0.05\varepsilon_i$, where $\varepsilon_i \in \mathbb{R}^d$ is standard normal. Following (Orvieto et al., 2022), we generate $H_i = A_i^T A_i/(3d)$ where the entries of $A_i \in \mathbb{R}^{3d\times d}$ are standard normal. We generate f_i^* from a uniform distribution with mean 0.5 and standard deviation ν, followed by truncation at zero to make sure all f_i^* are non-negative.

Note that in case (i) $\bar{x}$ is the minimizer of f and of each f_i. Further, $f_i(x_*) = \inf_{x\in\mathbb{R}^d} f_i(x) = f_i^*$, and $f(x_*) = \frac{1}{n}\sum_{i=1}^n f_i^* = \inf_{x\in\mathbb{R}^d} f(x)$. In the other case (ii), we compute the solution x_* by solving a linear system, and then compute $f_i(x_*)$. We always compute $f_\xi(x_*)$ by averaging $f_i(x_*)$ over the corresponding mini-batch.

We vary the standard deviation $\nu \in \{0.01, 0.1\}$ and the batch size $b \in \{4, 16\}$.

We run all versions of `IAM` with $\lambda_t = 9$ for all $t \geq 0$, as suggested by our convergence Theorems 3.3 and 3.2. As a baseline, we compare to `SGD-M` with constant learning rate and momentum $\beta = 0.9$. We set the learning rate to the theoretical value $\frac{1}{4L_{\max}}$ (cf. Sebbouh et al. (2021)), where $L_{\max} := \max_{i=1,\dots,n} L_i$ and $L_i := 2\lambda_{\max}(H_i)$ denotes the smoothness constant of f_i (here $\lambda_{\max}$ denotes the largest eigenvalue).[4] We further compare to `MoMo` that has access to $f_\xi(x_*)$, cf. (Schaipp et al., 2024, Eq. 17).

Discussion. In the interpolated case, see Figure 4, the theoretical version of `IAM` matches the rate of `SGD-M` *without any tuning*. However, if $f_\xi(x_*)$ is mis-specified, the convergence stales. This effect is more pronounced if the noise is large, or the batch size is small. In the non-interpolated case, see Figure 5, we observe that the theoretical version of `IAM` obtains a smaller final loss than `SGD-M`. This matches our theoretical result in the smooth setting, where we showed that we get convergence even if $\sigma_*^2 > 0$.

Compared to `MoMo`, we observe roughly the same convergence behaviour, with `IAM` typically having a slightly bigger slope. As a side note, we observe that `MoMo` also converges without interpolation, even though this case is not covered by the theory of Schaipp et al. (2024).

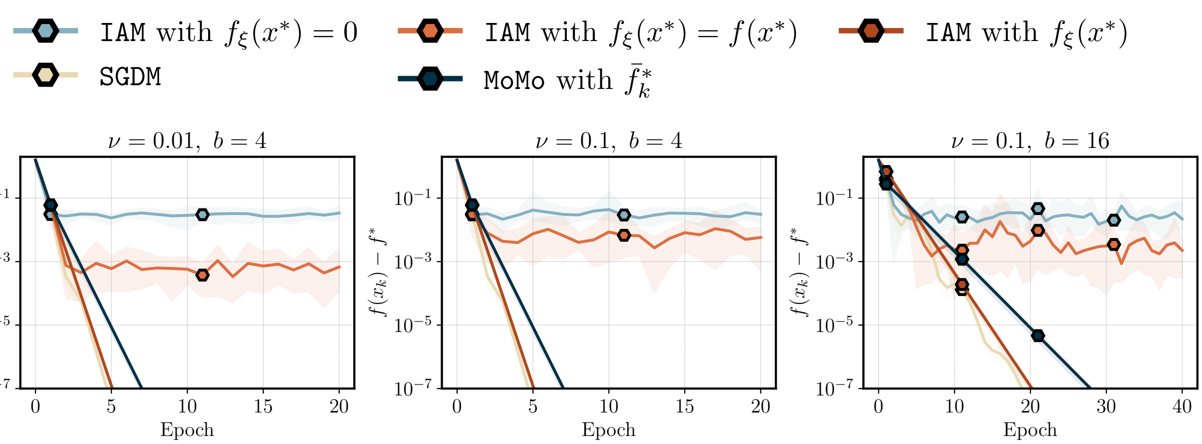


Figure 4. **Interpolation true:** `IAM` with the correct $f_{\xi_t}(x_*)$ converges as fast as `SGD-M` with the theoretical step size $\frac{1}{4L_{\max}}$. When ν is small **(left)**, the initial progress of `IAM` with the average $f(x_*)$ is equally good, before it stales. For ν large, the convergence stales earlier **(midlle)**. Increasing the batch size **(right)** slightly increases the gap between `IAM` with $f_{\xi_t}(x_*) = 0$ and $f_{\xi_t}(x_*) = f(x_*)$.

L.3. Supplementary Material on Distillation Experiment

Here we provide the complete details of our distillation experiments in Section 4.1, together with some additional plots.

Datasets and models. The datasets we consider are below. We used the `GPT2Tokenizer` from the Transformers library.

- `tinyShakespeare` (Karpathy, 2015): 40 000 lines from Shakespeare plays. The dataset has 303 688 tokens.
  Source: `https://huggingface.co/datasets/karpathy/tiny_shakespeare`

[4] Note that in `Pytorch` this requires setting `dampening`=0.9.

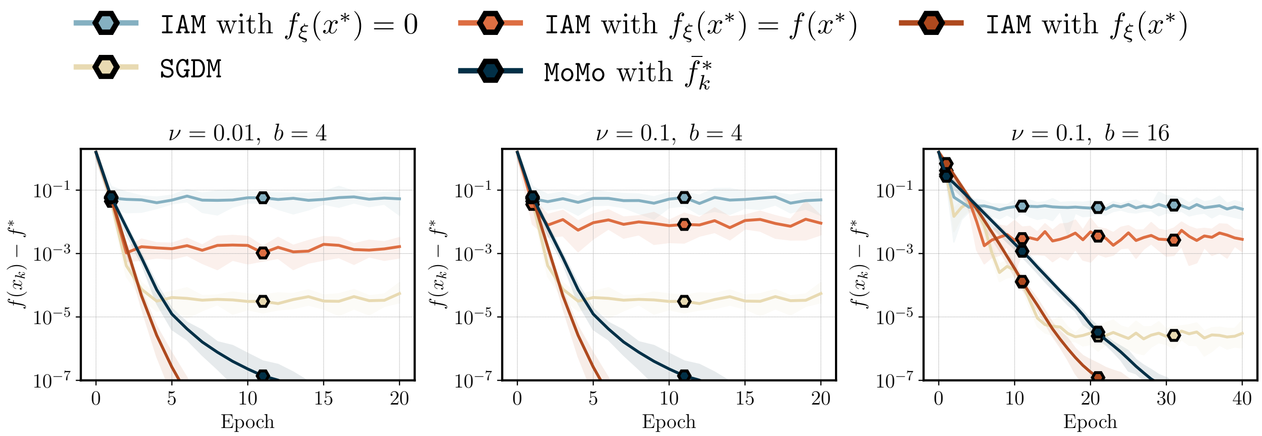


Figure 5. **Interpolation false:** see caption of Figure 4.

- `PTB` (Penn Treebank) (Marcus et al., 1993): The dataset contains 1 094 404 tokens.

  Source: `https://huggingface.co/datasets/ptb-text-only/ptb_text_only`

- `Wikitext2` (Merity et al., 2016): This is a subset of a 100 million token large collection of featured articles from Wikipedia. The dataset contains 2 389 828 tokens.

  Source: `https://huggingface.co/datasets/Salesforce/wikitext`

We use the module `GPT2LMHeadModel` from the HuggingFace `transformers` library (Wolf et al., 2020) to define our `GPT2` models [link]. The teacher model we use for `tinyShakespeare` and `Wikitext2` is the `gpt2-large` configuration within this module, which has 774 million parameters, and was pretrained by a team from OpenAI (Radford et al., 2019). We use the same tokenizer from the teacher model for the student model. For the `PTB` dataset, we use a different teacher model, as we found that `gpt2-large` had a poor fit, with the loss being above 5.0 on this dataset. So instead, we used the `GPT-J-6B` model (Wang, 2021), which has 6 billion parameters [link].

For the student model, we specify the configuration in Table 3.

Table 3. Parameters of the Student GPT2 model

Dataset	Embedding size	Number of layers	Number of attention heads
`tinyShakespeare`	768	2	4
`PTB`	768	2	4
`Wikitext2`	1200	12	12

Hyperparameter tuning. For our methods `IAM` (Algorithm 1) and `IAM-Adam` (Algorithm 2) we set $\lambda_t = 9$, which corresponds to using momentum $\beta = 0.9$ if the learning rates were constant, see Lemma I.1. Note that there is no hyperparameter tuning at all for `IAM` and `IAM-Adam`.

For the baseline methods `SGD` and `Adam`, we do the following tuning: we run bot `SGD` and `Adam` with constant learning rate and with a *warmup+cosine decay* schedule (Loshchilov & Hutter, 2017). This schedule does a linear warmup over the first 20% of iterations to a peak learning-rate γ, then performs a cosine decay to 0 over the remaining steps. For `Adam`, we set the learning rate to its default value of 10^{-3} for the constant schedule, and we set $\gamma = 1.5 \times 10^{-3}$ for the *warmup+cosine decay* schedule.

For `SGD` the learning rate needs to be tuned to get a reasonable performance: for the constant schedule, we chose the best-performing learning rate from the set

$$\gamma_{\text{constant}} \in \{0.0001, 0.001, 0.01, 0.05, 0.1, 0.2\}.$$

When using a scheduler with SGD, we take the best-performing value γ_{constant} and then independently tune the peak learning rate within the set

$$\gamma_{\text{constant}} \cdot \{1.2, 1.5, 2, 3, 5\}.$$

For SGD we use a momentum parameter of 0.9. In the Pytorch implementation of SGD, we also set the dampening parameter to 0.9 to ensure comparability of the tuned learning rate to the one of IAM.

Relationship to existing distillation techniques. In this paragraph, we aim to give a short overview over various distillation techniques which often vary in terms of their general setup and loss function. However, as model distillation is not the main focus of this paper, we point to the references below for additional background. In their seminal work, Hinton et al. (2015) propose to minimize the KL divergence between the teacher and student output probabilities. Follow-up works use a loss function that combines KL divergence and the standard loss for the student task (e.g., cross-entropy loss for classification, squared loss for regression) (Romero et al., 2015). On the other hand, Hsieh et al. (2023) propose to use the teacher output as surrogate labels in case of unavailable labeled training data for the students. We also refer to Beyer et al. (2022) for an overview of training techniques that improve the distillation performance.

The distillation setup that we propose in this paper is slightly different: we use only the final batch loss of the teacher model. The reason for this is that the IAM methods we investigate rely on an accurate guess of the optimal batch loss $f_\xi(x_*)$. In the distillation setting, we can leverage the pretrained teacher model in order to approximate the optimal batch loss values. The notion of distillation we use here might of independent interest, as it only needs access to the final batch loss value, but not the output probabilities of the model (the *logits*) nor its weights.

Additional plots. In Figure 6 we give the full plot of our distillation experiments, including the evolution of the learning rates for IAM and IAM-Adam.

In Figure 7 we give the distillation of several different small GPT2 models for the tinyShakespeare data set.

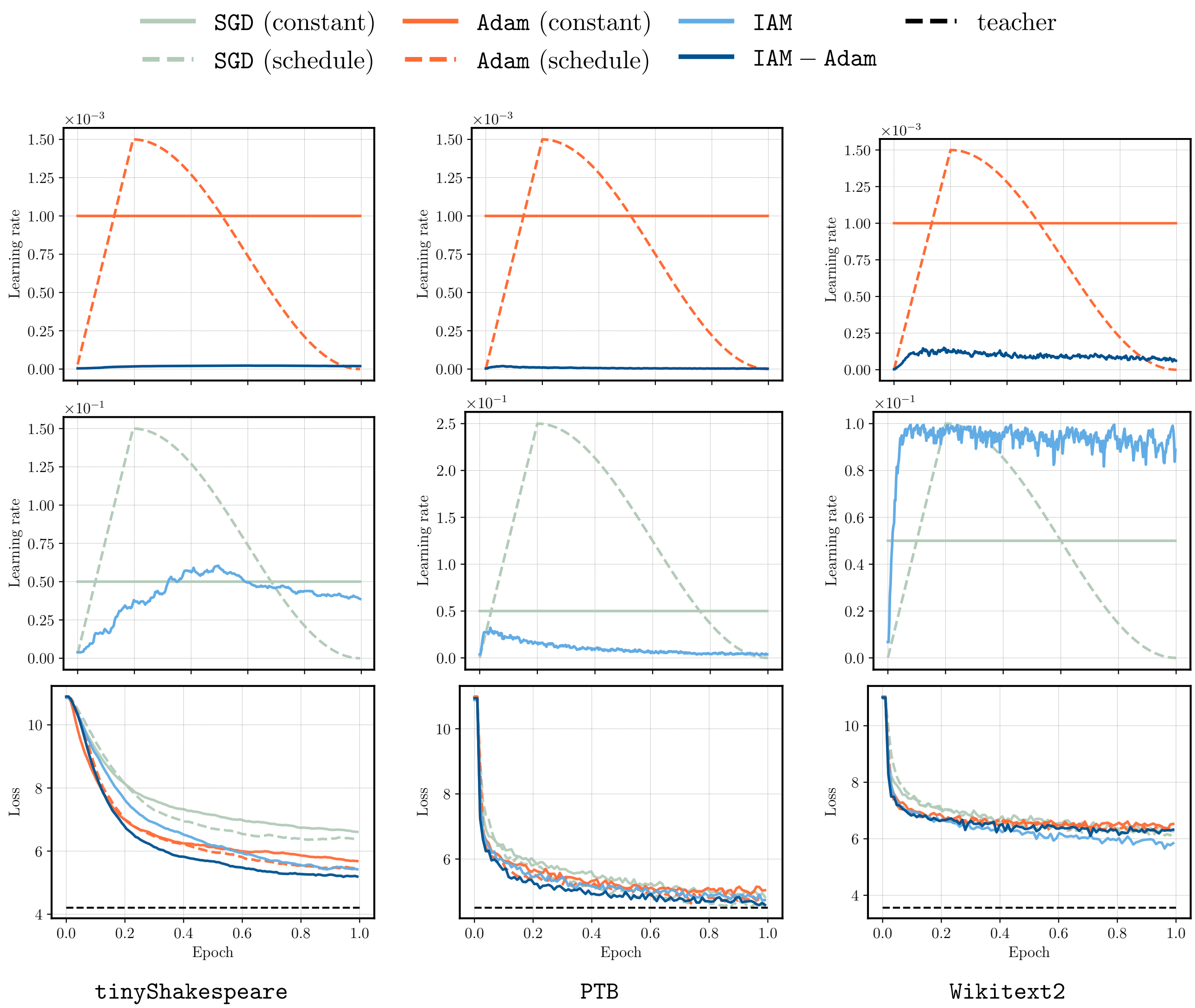
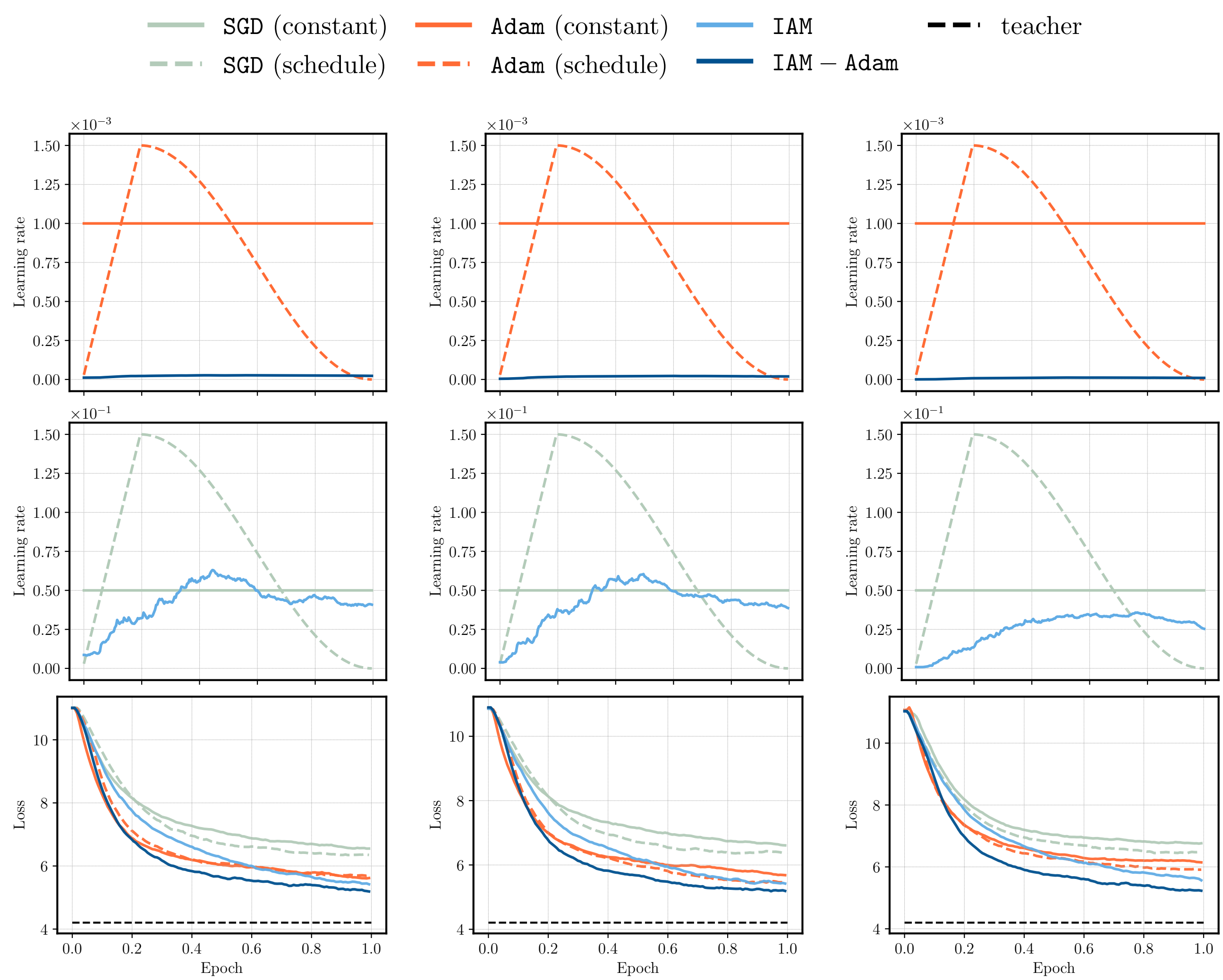


Figure 6. Full display of Figure 1. Adaptive learning rate of `IAM-Adam` compared to `Adam` **(top)**, of `IAM` compared to `SGD` **(middle)**, and the cross-entropy training loss **(bottom)**. Black line marks the average teacher loss.



n_embd=768, n_layer=2, n_head=4 n_embd=768, n_layer=4, n_head=8 n_embd=1200, n_layer=12, n_head=12

Figure 7. Distilling `gpt2-medium` into successively larger student models for the `tinyShakespeare` dataset.