On the Last Iterate Convergence of Momentum Methods

Xiaoyu Li   XIAOYULI@BU.EDU
Division of System Engineering, Boston University, Boston, MA 02215, USA

Mingrui Liu   MINGRUIL@GMU.EDU
Department of Computer Science, George Mason University, Fairfax, VA 22030, USA

Francesco Orabona   FRANCESCO@ORABONA.COM
Electrical & Computer Engineering, Boston University, Boston, MA 02215, USA

Abstract

SGD with Momentum (SGDM) is a widely used family of algorithms for large-scale optimization of machine learning problems. Yet, when optimizing generic convex functions, no advantage is known for any SGDM algorithm over plain SGD. Moreover, even the most recent results require changes to the SGDM algorithms, like averaging of the iterates and a projection onto a bounded domain, which are rarely used in practice. In this paper, we focus on the convergence rate of the last iterate of SGDM. For the first time, we prove that for any constant momentum factor, there exists a Lipschitz and convex function for which the last iterate of SGDM suffers from a suboptimal convergence rate of $\Omega(\frac{\ln T}{\sqrt{T}})$ after T iterations. Based on this fact, we study a class of (both adaptive and non-adaptive) Follow-The-Regularized-Leader-based SGDM algorithms with *increasing momentum* and *shrinking updates*. For these algorithms, we show that the last iterate has optimal convergence $O(\frac{1}{\sqrt{T}})$ for unconstrained convex stochastic optimization problems without projections onto bounded domains nor knowledge of T. Further, we show a variety of results for FTRL-based SGDM when used with adaptive stepsizes. Empirical results are shown as well.

Keywords: Convex Optimization, Momentum methods , Stochastic Optimization

1. Introduction

Momentum methods have become one of the most used first-order optimization algorithms in machine learning applications. When momentum is used together with Stochastic Gradient Descent (SGD), there are two main variants considered in the literature: the stochastic version of the Heavy Ball momentum (SHB) (Polyak, 1964) and Nesterov's momentum (also called Nesterov Accelerate Gradient method) (Nesterov, 1983). Besides these two, there are other variations as well. For example, an exponential moving average of the (stochastic) gradients can be used to replace the gradients in the updates (Kingma and Ba, 2015; Reddi et al., 2016; Alacaoglu et al., 2020; Liu et al., 2020).

Despite this zoo of variants, due to the presence of noise, it is well-known that Stochastic Gradient Descent with Momentum (SGDM) *does not* guarantee an accelerated rate of convergence of noise nor any real advantage over plain SGD on generic convex problems. For example, recent works have proved that a variant of SGD with momentum improves only the non-dominant terms in the convergence rate on some specific stochastic problems (Dieuleveut et al., 2017; Jain et al., 2018). Moreover, often an idealized version of SGDM is used in the theoretical analysis rather than the actual SGDM people use in practice. For example, projections onto bounded domains at each step, averaging of the iterates (e.g., Alacaoglu et al., 2020), and knowledge of the total number of

iterations (Ghadimi and Lan, 2012) are often assumed. The mismatch between theory and practice is concerning because, for example, it is known that in some cases the lack of projections can destroy the convergence of some algorithms (Orabona and Pál, 2018). Overall, recent analyses seem unable to pinpoint any advantage of using a momentum term in SGD in the stochastic optimization of general convex functions.

In the following, we denote by SGDM the following updates

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta_t \boldsymbol{m}_t, \quad \boldsymbol{m}_t = \beta_t \boldsymbol{m}_{t-1} + (1 - \beta_t)\boldsymbol{g}_t, \tag{1}$$

where $0 \le \beta_t \le 1$.

In this paper, to show a discriminant difference between SGD and SGDM, we focus on the convergence of the last iterate. Hence, we study the convergence of the *last iterate* of SGDM for unconstrained optimization of convex functions. Unfortunately, our first result is a negative one: We show that the last iterate of SGDM can have a suboptimal convergence rate for *any constant momentum setting*.

Hence, motivated by the above result, we analyze yet another variant of SGDM. We start from the very recent observation (Defazio, 2020) that SGDM can be seen as a primal averaging procedure (Nesterov and Shikhman, 2015; Tao et al., 2018; Cutkosky, 2019) applied to the iterates of Online Mirror Descent (OMD) (Nemirovsky and Yudin, 1983; Warmuth and Jagota, 1997). Based on this fact, we analyze SGDM algorithms based on the Follow-the-Regularized-Leader (FTRL) framework[1] (Shalev-Shwartz, 2007; Abernethy et al., 2008) and the primal averaging. The use of FTRL instead of OMD removes the necessity of projections onto bounded domains, while the primal averaging acts as a momentum term and guarantees the optimal convergence of the last iterate. The resulting algorithm has an *increasing momentum* and *shrinking updates* that precisely allow to avoid our lower bound.

More in detail, we prove that the expected suboptimality gap of the last iterate of FTRL-based SGDM converges at the optimal rate of $O(1/\sqrt{T})$ on convex functions, without assuming bounded domains nor the knowledge of the total number of iterations. This also disproves a more general conjecture than the one in (Jain et al., 2019, 2021), removing the bounded assumption. Moreover, we show that our construction is general enough to allow for an entire family of FTRL-based SGDM methods, both adaptive and non-adaptive. For example, we show that "adaptive" learning rates give rise to convergence rates that are adaptive to gradients, noise, and to the interpolation regime.

The rest of the paper is organized as follows: We discuss the related work in Section 2 and the setting and assumptions in Section 3. We then present our main results: the lower bound (Section 4) and the new FTRL-based SGDM (Section 5). Finally, in Section 6 we present an empirical evaluation of our algorithms and in Section 7 we outline a future work direction.

2. Related Work

Stochastic Momentum Methods SGDM has become a popular tool in deep learning and its importance has been discussed by recent studies (Sutskever et al., 2013). Polyak (1964) first proposed the use of momentum in gradient descent, calling it the Heavy-Ball method. In the stochastic setting, there are multiple work analyzing the use of momentum in SGD. In particular, Yang et al. (2016)

---

1. FTRL is known in the offline optimization literature as Dual Averaging (DA) (Nesterov, 2009), but in reality, DA is a special case of FTRL when the functions are linearized.

prove a convergence rate of $O(1/\sqrt{T})$ for the averaged iterate in the convex setting, and for an iterate taken uniformly at random in the nonconvex setting. Liu et al. (2020) provide a convergence analysis for SGDM and Multistage SGDM for smooth functions in the strongly convex and nonconvex settings. Also, adaptive variants of momentum methods (Kingma and Ba, 2015; Reddi et al., 2018; Luo et al., 2018) are very popular in the deep learning literature, even if their guarantees are only for the online convex optimization setting assuming a decreasing momentum factor and projections onto bounded domains. Alacaoglu et al. (2020) recently removed the assumption of a vanishing momentum factor, but they still require projections over a bounded domain. In the non-convex and smooth case, Cutkosky and Orabona (2019) introduce a variant of SGDM with a variance-reduction effect and a faster convergence rate than SGD on non-convex functions, but it requires two stochastic gradients per step.

Lower Bound Harvey et al. (2019) prove the tight convergence bound $O(\ln T/\sqrt{T})$ of the last iterate of SGD for convex and Lipschitz functions. Kidambi et al. (2018) provide a lower bound for the Heavy Ball method for least square regression problems. To the best of our knowledge, there is no lower bound for the last iterate of SGDM in the general non-smooth non-strongly-convex setting.

Last Iterate Convergence of SGDM Nesterov and Shikhman (2015) introduces a quasi-monotone subgradient method, which uses double averaging (both in Primal and Dual) based on Dual Averaging, to achieve the optimal convergence of the last iterate for the convex and Lipschitz functions. However, they just considered the batch case. This approach was then rediscovered and extended by Cutkosky (2019). Our FTRL-based SGDM is a generalization of the approach in Nesterov and Shikhman (2015) with generic regularizers and stochastic gradients. Tao et al. (2018) extends Nesterov and Shikhman (2015)'s method to Mirror Descent, calling it stochastic primal averaging. They recover the same bound for convex functions, again with a bounded domain assumption. Defazio (2020) points out that the sequence generated by the stochastic primal averaging (Tao et al., 2018) can be identical to that of stochastic gradient descent with momentum for specific choices of the hyper-parameters. Accordingly, they give a Lyapunov analysis in the nonconvex and smooth case. Based on this work, Jelassi and Defazio (2020) introduce "Modernized dual averaging method", which is actually equal to the one by Nesterov and Shikhman (2015). They also give a similar Lyapunov analysis as in Defazio (2020) with specific choices of hyper-parameters in the non-convex and smooth optimization setting, where they assume a bounded domain and get a convergence bound $O(\ln T/\sqrt{T})$. Recently, Tao et al. (2021) propose the very same algorithm as in Tao et al. (2018) and analyze it as a modified Polyak's Heavy-ball method (already pointed out by Defazio (2020)). They give an analysis in the convex cases and extend it to an adaptive version, obtaining in both cases an optimal convergence of the last iterate. However, they still assume the use of projections onto bounded domains.

Last iterate convergence rate $O(\frac{1}{\sqrt{T}})$ Ghadimi and Lan (2012) present the last iterate of AC-SA (Nemirovski et al., 2009; Lan, 2012) for convex functions in the unconstrained setting, that in the Euclidean case reduces to SGD with an increasing Nesterov momentum, showing that it can achieve a convergence rate $O(\frac{1}{\sqrt{T}})$ if the number of iterations T is known in advance. Sebbouh et al. (2021) analyze Stochastic Heavy Ball-Iterave Moving Average method (SHB-IMA), which is equal to the Stochastic Heavy Ball method (SHB) with a specific choice of hyper-parameters. They prove a convergence rate for the last iterate of of $O(\frac{1}{\sqrt{T}})$ if T is given in advance, and is $O(\frac{\ln T}{\sqrt{T}})$ if T is unknown. Jain et al. (2019, 2021) conjecture that under assumption **(H3')** (see next Section) "for any-time algorithm (i.e., without apriori knowledge of T) expected error rate of $\frac{DG\ln T}{\sqrt{T}}$ is information-theoretically optimal", where D is the diameter of the bounded domain. This

Table 1: Last iterate convergence of momentum methods in convex setting

Algorithm	Assumption	Bounded Domain	Requires T	Rate	Reference
Adaptive-HB	**(H3')**	Yes	No	$O(\frac{1}{\sqrt{T}})$	Tao et al. (2021)
SHB-IMA	**(H1)** + **(H2)**	No	Yes	$O(\frac{1}{\sqrt{T}})$	Sebbouh et al. (2021)
			No	$O(\frac{\ln T}{\sqrt{T}})$	
AC-SA	**(H2)** + **(H1)** or **(H2)** + Lipschitz	No	Yes	$O(\frac{1}{\sqrt{T}})$	Ghadimi and Lan (2012)
FTRL-SGDM	**(H3)**	No	No	$O(\frac{1}{\sqrt{T}})$	This paper, Corollary 1
	(H1)+(H2)+(H3')	No	No	$O(\frac{\ln T}{T} + \frac{\sigma}{\sqrt{T}})$	This paper, Corollary 4

was already disproved by the results in Tao et al. (2021), but here we disprove it even in the more challenging unconstrained setting.

We summarize the results on the last iterate convergence for convex optimization and their assumptions in Table 1. The assumptions are defined in the next section.

3. Problem Set-up

Notation We denote vectors by bold letters, e.g. $\boldsymbol{x} \in \mathbb{R}^d$. All standard operations on the vectors, e.g., $\boldsymbol{xy}, \boldsymbol{x}/\boldsymbol{y}, \sqrt{\boldsymbol{x}}$ and $\boldsymbol{x} < \boldsymbol{y}$, are to be considered element-wise. We denote by $\mathbb{E}[\cdot]$ the expectation with respect to the underlying probability space and by $\mathbb{E}_t[\cdot]$ the conditional expectation with respect to the past. Any norm without particular notation in this work is the ℓ_2 norm.

Setting We consider the unconstrained optimization problem $\min_{\boldsymbol{x} \in \mathbb{R}^d} f(\boldsymbol{x})$, where $f(\boldsymbol{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ is a convex function and we denote its infimum by $f^\star$. We also assume to have access to a first-order black-box optimization oracle that returns a stochastic subgradient in any point $\boldsymbol{x} \in \mathbb{R}^d$. In particular, we assume that we receive a vector $\boldsymbol{g}(\boldsymbol{x}, \xi)$ such that $\mathbb{E}_\xi[\boldsymbol{g}(\boldsymbol{x}, \xi)] = \nabla f(\boldsymbol{x})$ for any $\boldsymbol{x} \in \mathbb{R}^d$. To make the notation concise, we let $\boldsymbol{g}_t \triangleq \boldsymbol{g}(\boldsymbol{x}_t, \xi_t)$ and $\mathbb{E}_t[\boldsymbol{g}_t] = \nabla f(\boldsymbol{x}_t), \forall t$.

We will make different assumptions on the objective function f. Sometimes, we will assume that

- **(H1)** f is L-smooth, that is, f is continuously differentiable and its gradient is L-Lipschitz, i.e., $\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\| \le L\|\boldsymbol{x} - \boldsymbol{y}\|$.

We also use one or more of the following assumptions on the stochastic gradients $\boldsymbol{g}_t$.

- **(H2)** bounded variance: $\mathbb{E}_t\|\boldsymbol{g}_t - \nabla f(\boldsymbol{x}_t)\|^2 \le \sigma^2$.
- **(H3)** bounded in expectation: $\mathbb{E}\|\boldsymbol{g}_t\|^2 \le G^2$.
- **(H3')** ℓ_2 bounded: $\|\boldsymbol{g}_t\| \le G$.
- **(H3")** ℓ_∞ bounded: $\|\boldsymbol{g}_t\|_\infty \le G_\infty$.

4. Lower bound for SGDM

First of all, as we discussed in the related work, most of the analyses of SGDM assume a vanishing momentum or a constant one. However, is constant momentum the best setting for stochastic optimization of convex functions, especially for the convergence of the last iterate? For this question,

it is worth remembering that the use of a constant momentum term is mainly motivated by the empirical evidence in the deep learning literature. However, deep learning objective functions are non-convex and the convex setting might be different. Also, the deep learning literature offers no theoretical explanations.

In this section, we show the surprising result that for SGD with any constant momentum, there exists a function for which the lower bound of the last iterate is $\Omega\left(\ln T/\sqrt{T}\right)$. Our proof extends the one in Harvey et al. (2019) to SGD with momentum.

We consider SGDM with constant momentum factor β in (1), where $\boldsymbol{g}_t \in \partial f(\boldsymbol{x}_t)$ and a polynomial stepsize $\eta_t = c \cdot t^{-\alpha}, 0 \le \alpha \le \frac{1}{2}$.

Let $\mathcal{X}$ denote the Euclidean ball with radius $\frac{2c}{1-\beta}$ in $\mathbb{R}^T$. For any fixed β and α and $L > 0$, we introduce the following function. Define $f \colon \mathcal{X} \to \mathbb{R}$ and $\boldsymbol{h}_i \in \mathbb{R}^T$ for $i \in [T+1]$ by

$$f(\boldsymbol{x}) = \max_{i \in [T+1]} \boldsymbol{h}_i^T \boldsymbol{x}, \quad h_{i,j} = \begin{cases} a_j, & 1 \le j < i \\ -b_j, & i = j < T \\ 0, & i < j \le T \end{cases} \tag{2}$$

where $b_j = \frac{Lj^\alpha}{2T^\alpha}$ and $a_j = \frac{L(1-\beta)}{8(T-j+1)}$. We have that $\partial f(\boldsymbol{x}_t)$ is the convex hull of $\boldsymbol{h}_i : i \in \mathcal{I}(\boldsymbol{x})$ where $\mathcal{I}(\boldsymbol{x}) = \{i : \boldsymbol{h}_i^T \boldsymbol{x} = f(\boldsymbol{x})\}$. Note that f is L-Lipschitz over $\mathbb{R}^T$ since

$$\|\boldsymbol{h}_i\|^2 \le \sum_{i=1}^T a_i^2 + b_T^2 \le \frac{L^2(1-\beta)^2}{64} \sum_{i=1}^T \frac{1}{i^2} + \frac{L^2}{4} \le L^2 .$$

Claim 1 *For f defined in (2), it satisfies that $\inf_{\boldsymbol{x} \in \mathbb{R}^T} f(\boldsymbol{x}) = 0$.*

Proof First, since $f(0) = 0$, we have that $\inf_{\boldsymbol{x} \in \mathbb{R}^T} f(\boldsymbol{x}) \le 0$.
We continue to prove this claim by contradiction. Assume that there exists $\boldsymbol{x}^\star = [x_1^\star, x_2^\star, \ldots, x_T^\star]$ such that
$$f(\boldsymbol{x}^\star) < 0 .$$

By the definition of f, it satisfies that

$$\boldsymbol{h}_i^T \boldsymbol{x}^\star < 0, \quad \forall i \in [T+1] . \tag{3}$$

In particular, $\boldsymbol{h}_1^T \boldsymbol{x}^\star = -b_1 x_1^\star < 0$. Since that b_1 is positive, we know that $x_1^\star > 0$. Also, $\boldsymbol{h}_2^T \boldsymbol{x}^\star = a_1 x_1^\star - b_2 x_2^\star < 0$. Due to the positiveness of $a_1, x_1^\star$, and b_2, $x_2^\star$ has to be positive. Similarly, we have that for any $x_j^\star, j \in [T], x_j^\star > 0$.
Then, we have

$$\boldsymbol{h}_{T+1}^T \boldsymbol{x}^\star = \sum_{j=1}^T a_j x_j^\star > 0 .$$

However, this is contradict with (3).
Thus, we conclude that $\inf_{\boldsymbol{x} \in \mathbb{R}^T} f(\boldsymbol{x}) = 0$. ∎

Theorem 1 (Lower bound of SGDM) *Fix a polynomial stepsize sequence $\eta_t = c \cdot t^{-\alpha}$, where $0 \le \alpha \le \frac{1}{2}$, a momentum factors $\beta \in [0, 1)$, a Lipschitz constant $L > 0$ and a number of iterations T. Then, there exists a sequence $\boldsymbol{z}_t$ generated by SGDM with stepsizes η_t and momentum factor β on the function f in (2), where the T-th iterate satisfies*

$$f(\boldsymbol{z}_T) - f^\star \ge \frac{L^2(1 - \beta)^2 c \ln T}{4T^\alpha} \ .$$

We stress that $\ln T$ cannot be cancelled by any setting of β or c. Indeed, the above lower bound can be instantiated by any β and any T. Hence, for a given β, there exists T large enough such that $\ln T$ is constant-times bigger than $\frac{1}{(1-\beta)^2}$.

When $\beta = 1$, the algorithm is basically staying at the initial point. We can choose an arbitrary positive number $C > 0$ and let $z_1 = C$, then

$$f(\boldsymbol{z}_T) - f^\star \ge C, \quad C > 0 \ .$$

We will use the following lemma in the proof.

Proof Define a sequence $\boldsymbol{z}_t$ for $t \in [T + 1]$ as follows: $\boldsymbol{z} = 0$, where s is a positive number decided later, and

$$\boldsymbol{z}_{t+1} = \boldsymbol{z}_t - (1 - \beta)\eta_t \sum_{i=1}^{t} \beta^{t-i}\boldsymbol{h}_i \ . \tag{4}$$

We will show that $\boldsymbol{z}_t$ are exactly the updates of SGDM and $f(\boldsymbol{z}_{T+1}) \ge \Omega\left(\frac{\ln T}{T^\alpha}\right)$. We will use the following two lemmas.

Lemma 2 *Let $b_j = \frac{Lj^\alpha}{2T^\alpha}$, $a_j = \frac{L(1-\beta)}{8(T-j+1)}$, and $\eta_j = c \cdot j^{-\alpha}$. $\boldsymbol{z}_t$ is defined as in (4). Then, for $1 \le t < j$, $z_{t,j} = 0$, and for $t > j$, $z_{t,j} \ge \frac{L(1-\beta)c}{4T^\alpha}$.*

Proof We first prove by induction that when $1 \le t \le j$, $z_{t,j} = 0$. First, $\boldsymbol{z}_1 = 0$ Also, suppose it holds for t. Then, in the case of $t + 1$, for any $j \ge t + 1$,

$$z_{t+1,j} = z_{t,j} - (1 - \beta)\eta_t \sum_{i=1}^{t} \beta^{t-i}h_{i,j} = 0 - 0 = 0,$$

which implies $t \le j$, $z_{t,j} = 0$ holds. Next, we claim that $\boldsymbol{z}_t$ satisfies

$$z_{t,j} \ge z_{j,j} + (1 - \beta)b_j\eta_j - a_j \sum_{k=j+1}^{t-1} \eta_k, \quad 1 \le j < t \le T \ . \tag{5}$$

We prove (5) by induction. For any t, $z_{t,t-1}$ satisfies (5) since

$$z_{t+1,t} = z_{t,t} - (1 - \beta)\eta_t \sum_{i=1}^{t} \beta^{t-i}h_{i,t} = -(1 - \beta)\eta_t h_{t,t} = (1 - \beta)\eta_t b_t \ .$$

Then, suppose (5) holds for any $j < t$. We show that it holds for any $j < t + 1$. We already proved for $j = t$. For $j < t$,

$$z_{t+1,j} = z_{t,j} - (1-\beta)\eta_t \sum_{i=1}^{t} \beta^{t-i} h_{i,j} = z_{t,j} - (1-\beta)\eta_t \sum_{i=j}^{t} \beta^{t-i} h_{i,j}$$

$$= z_{t,j} + (1-\beta)\eta_t \beta^{t-j} b_j - (1-\beta)\eta_t \sum_{i=j+1}^{t} \beta^{t-i} h_{i,j}$$

$$\geq z_{t,j} - (1-\beta)\eta_t \sum_{i=j+1}^{t} \beta^{t-i} a_j$$

$$\geq (1-\beta)b_j\eta_j - a_j \sum_{k=j+1}^{t-1} \eta_k - (1-\beta)a_j\eta_t \sum_{i=j+1}^{t} \beta^{t-i}$$

$$\geq (1-\beta)b_j\eta_j - a_j \sum_{k=j+1}^{t} \eta_k, \tag{6}$$

where in the second inequality we used the induction hypothesis.

Using that $b_j = \frac{Lj^\alpha}{2T^\alpha}$, $a_j = \frac{L(1-\beta)}{8(T-j+1)}$ and $\eta_j = \frac{c}{j^\alpha}$, we have

$$(6) = \frac{L(1-\beta)c}{2T^\alpha} - \frac{L(1-\beta)c}{8(T-j+1)} \sum_{k=j+1}^{t} \frac{1}{k^\alpha}. \tag{7}$$

By Lemma 8 in the Appendix, we have that for $0 < \alpha \leq \frac{1}{2}$,

$$(7) \geq \frac{L(1-\beta)c}{2T^\alpha} - \frac{L(1-\beta)c}{4T^\alpha} \geq \frac{L(1-\beta)c}{4T^\alpha},$$

and for $\alpha = 0$,

$$(7) \geq \frac{L(1-\beta)c}{2} - \frac{L(1-\beta)(t-j-1)c}{8(T-j+1)} \geq \frac{L(1-\beta)c}{4}.$$

Thus, we have $z_{t,j} \geq \frac{L(1-\beta)c}{4T^\alpha} \geq \frac{L(1-\beta)c}{4T^\alpha}$. ∎

Lemma 3 $f(z_t) = h_t^T z_t$ *for any* $t \in [T+1]$. *The subgradient oracle for f at z_t returns h_t.*

Proof We claim that $h_i^T z_t = h_t^T z_t$ for all $i > t \geq 1$ and $h_t^T z_t > h_i^T z_t$ for all $1 \leq i < t$.

When $i > t \geq 2$, z_t is supported on the first $t-1$ coordinates, while h_t and h_i agree on the first $t-1$ coordinates.

In the case of $1 \leq i < t$, by the definition of z_t and h_t, we have

$$z_t^T(h_t - h_i) = \sum_{j=1}^{t-1} z_{t,j}(h_{t,j} - h_{i,j}) = \sum_{j=i}^{t-1} z_{t,j}(h_{t,j} - h_{i,j}) = z_{t,i}(a_i + b_i) + \sum_{j=i+1}^{t-1} z_{t,j} a_j > 0,$$

where in the last inequality we used the fact that a_i, b_i and $z_{t,i}$ are at least non-negative.

Thus, we have proved $f(z_t) = h_t^T z_t$ by the definition. Moreover, $\mathcal{I}(z_t) = \{i : h_i^T z_t = f(z_t)\} = \{t, \ldots, T+1\}$. So the subgradient evaluated at z_t is h_t. ∎

Algorithm 1 FTRL-based SGDM

1: **Input:** A sequence $\alpha_1, ..., \alpha_T$, with $\alpha_1 > 0$. Non-increasing sequence $\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_{T-1}$. $\boldsymbol{m}_0 = 0$.
    $\boldsymbol{x}_1 \in \mathbb{R}^d$.
2: **for** $t = 1, \ldots, T$ **do**
3:     Get $\boldsymbol{g}_t$ at $\boldsymbol{x}_t$ such that $\mathbb{E}_t[\boldsymbol{g}_t] = \nabla f(\boldsymbol{x}_t)$
4:     $\beta_t = \frac{\sum_{i=1}^{t-1} \alpha_i}{\sum_{i=1}^{t} \alpha_i}$ (Define $\sum_{i=1}^{0} \alpha_i = 0$)
5:     $\boldsymbol{m}_t = \beta_t \boldsymbol{m}_{t-1} + (1 - \beta_t)\boldsymbol{g}_t$
6:     $\boldsymbol{\eta}_t = \frac{\alpha_{t+1} \sum_{i=1}^{t} \alpha_i}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{\gamma}_t$
7:     $\boldsymbol{x}_{t+1} = \frac{\sum_{i=1}^{t} \alpha_i}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{x}_t + \frac{\alpha_{t+1}}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{x}_1 - \boldsymbol{\eta}_t \boldsymbol{m}_t$
8: **end for**

Now, we first get a lower bound and an upper bound of $\boldsymbol{z}_t$ using Lemma 2. Then, by Lemma 3, we have shown that $\boldsymbol{z}_t$ are exactly the updates of SGDM.

Thus, for $\beta \in [0, 1)$, we have

$$f(\boldsymbol{z}_{T+1}) = \boldsymbol{h}_{T+1}^T \boldsymbol{z}_{T+1} = \sum_{j=1}^{T} h_{T+1,j} z_{T+1,j}$$

$$\geq \frac{L^2 (1-\beta)^2 c}{4T^\alpha} \sum_{j=1}^{T} \frac{1}{T-j+1} \geq \frac{L^2 (1-\beta)^2 c \ln T}{4T^\alpha} \ .$$

        ■

5. FTRL-based SGDM

The lower bound for the last iterate in the previous section motivates us to study a different variant of SGDM. In particular, we aim to find a way to remove the $\ln T$ term from the convergence rate.

Defazio (2020) points out that the stochastic primal averaging method (Tao et al., 2018) (which is also an instance of Algorithm 1 in Cutkosky (2019) with OMD):

$$\boldsymbol{z}_{t+1} = \boldsymbol{z}_t - \gamma_t \boldsymbol{g}_t, \quad \boldsymbol{x}_{t+1} = s_t \boldsymbol{x}_t + (1 - s_t)\boldsymbol{z}_t$$

could be one-to-one mapped to the momentum method

$$\boldsymbol{m}_{t+1} = \beta_t \boldsymbol{m}_t + \boldsymbol{g}_t, \quad \boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \alpha_t \boldsymbol{m}_t$$

by setting $\gamma_{t+1} = \frac{\gamma_t - \alpha_t}{\beta_{t+1}}$. While this is true, the convergence rate depends on the convergence rate of OMD with time-varying stepsizes, that in turn requires to assume that $\|\boldsymbol{x}_t - \boldsymbol{x}^\star\|^2 \leq D^2$. This is possible only by using a projection onto a bounded domain in each step.

Thus, to go beyound bounded domains, we propose to study a new variant of SGDM which has the following form (details in Algorithm 1),

$$\boldsymbol{m}_{t+1} = \beta_t \boldsymbol{m}_t + (1 - \beta_t)\boldsymbol{g}_t, \quad \boldsymbol{x}_{t+1} = s_t \boldsymbol{x}_t - \alpha_t \boldsymbol{m}_t \ .$$

Note the presence of a shrinking factor $s_t \leq 1$ in the iterates in each step. This variant comes naturally when using the primal averaging scheme with FTRL rather than OMD. Hence, we just denote it by FTRL-based SGDM. Now, this momentum variant inherits all the good properties of FTRL. In particular, we no longer need the bounded domain assumption. Moreover, we will show that it guarantees the optimal convergence $O(\frac{1}{\sqrt{T}})$ (Agarwal et al., 2012) of the last iterate for convex and Lipschitz functions.

5.1. Convergence Rates for FTRL-based SGDM

We first present a very general theorem for FTRL-based SGDM.

Theorem 4 *Under the assumption in Section 3, Algorithm 1 guarantees*

$$\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{1}{\sum_{t=1}^{T}\alpha_t}\mathbb{E}\left[\left\|\frac{\boldsymbol{x}_1 - \boldsymbol{x}^\star}{\sqrt{\boldsymbol{\gamma}_{T-1}}}\right\|^2 + \sum_{t=1}^{T}\langle\boldsymbol{\gamma}_{t-1}, \alpha_t^2\boldsymbol{g}_t^2\rangle\right] .$$

The above theorem is very general and it gives rise to a number of different variations of the FTRL-based SGDM. In particular, we can instantiate it with the following choices.

First, we consider the most used polynomial stepsize $\frac{c}{\sqrt{t}}$ for convex and Lipschitz function, and the constant stepsize $\frac{c}{\sqrt{T}}$ if T is given in advance.

Corollary 1 *Assume (H3) and set $\alpha_t = 1$ for all t. Algorithm 1 with either $\boldsymbol{\gamma}_{t-1} = \frac{c}{G\sqrt{t}} \cdot \boldsymbol{1}$ or $\boldsymbol{\gamma}_{t-1} = \frac{c}{G\sqrt{T}} \cdot \boldsymbol{1}$ guarantees*

$$\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2 G}{c\sqrt{T}} + \frac{2cG}{\sqrt{T}} .$$

The above corollary tells that both of these two stepsizes give the optimal bound $O(\frac{1}{\sqrt{T}})$ for the last iterate. Next, we will show that if we use an adaptive[2] stepsize, Algorithm 1 gives a data-dependent convergence rate for the last iterate. We first consider a global version of the AdaGrad stepsize as in Streeter and McMahan (2010); Li and Orabona (2019); Ward et al. (2019).

Corollary 2 *Assume (H3') and take $\boldsymbol{\gamma}_t = \frac{\alpha \cdot \boldsymbol{1}}{\sqrt{\epsilon + \sum_{i=1}^{t}\alpha_i^2\|\boldsymbol{g}_i\|^2}}, \epsilon > 0, 1 \leq t \leq T$ and $\alpha_t = 1$. Then, Algorithm 1 guarantees*

$$\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{1}{T}\left[\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\mathbb{E}\sqrt{\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2 + \epsilon} + \frac{\alpha G^2}{\sqrt{\epsilon}}\right] .$$

We also state a result for the coordinate-wise AdaGrad stepsizes (McMahan and Streeter, 2010; Duchi et al., 2010).

Corollary 3 *Assume (H3") and set $\boldsymbol{\gamma}_t = \frac{\alpha}{\sqrt{\epsilon + \sum_{i=1}^{t}\alpha_i^2\boldsymbol{g}_i^2}}, \epsilon > 0, 1 \leq t \leq T$ and $\alpha_t = 1$. Then, Algorithm 1 guarantees*

$$\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{1}{T}\left[\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\sum_{j=1}^{d}\mathbb{E}\sqrt{\sum_{t=1}^{T}\boldsymbol{g}_{t,j}^2 + \epsilon} + \frac{\alpha d G_\infty^2}{\sqrt{\epsilon}}\right] .$$

---

2. Even if widely used in the literature, it is a misnomer to call these stepsize "adaptive": an algorithm can be adaptive to some unknown quantities (if proved so), not the stepsizes.

The above two corollaries show that the convergence bound are adaptive to the stochastic gradients. In words, in the worst case (i.e., $\sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2 = O(T)$ and $\sum_{t=1}^{T-1} \boldsymbol{g}_{t,j}^2 = O(T)$), the convergence rate is $O(\frac{1}{\sqrt{T}})$. However, when the stochastic gradients are small or sparse, the rate could be much faster than $O(\frac{1}{\sqrt{T}})$. Moreover, the above results give very simple ways to obtain optimal convergence for the last iterate of first-order stochastic methods, that was still unclear if it could be obtained as discussed in Jain et al. (2019, 2021).

Also, we now show that if in addition f is smooth, the last iterate of FTRL-based momentum with the global adaptive stepsize of Corollary 2 gives adaptive rates of convergence that interpolate between $O(\frac{1}{\sqrt{T}})$ and $O(\frac{\ln T}{T})$.

Corollary 4 *Assume (H1). Then, under the same assumption and parameter setting of Corollary 2, Algorithm 1 guarantees*

$$
\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{C}{T} \left(\sqrt{\epsilon + 4L^2 C^2 \ln^2 T + 4LC\sqrt{\epsilon}\ln T + \frac{2\alpha G^2}{\sqrt{\epsilon}}} + \frac{\alpha G^2}{\sqrt{\epsilon}} \right) + \frac{\sqrt{2}C\sigma}{\sqrt{T}} .
$$

where $C \triangleq \left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha \right)$.

Observe that when $\sigma = 0$, namely when there is no noise on the gradients, the rate of $O(\frac{\ln T}{T})$ is obtained. As far as we know, the above theorems are the first convergence guarantees for the last iterate of momentum algorithms with adaptive learning rates in unconstrained convex optimization.

5.2. Convergence Rate in Interpolation Regime

Now we assume that $F(\boldsymbol{x}) = \mathbb{E}_\xi[f(\boldsymbol{x}, \xi)]$ and that the stochastic gradient is calculated drawing one function in each time step and calculating its gradient: $\boldsymbol{g}_t = \nabla f(\boldsymbol{x}_t, \xi_t)$. In this scenario, it makes sense to consider the *interpolation* condition (Needell et al., 2015; Ma et al., 2018)

$$
\boldsymbol{x}^\star \in \operatorname*{argmin}_{\boldsymbol{x}} F(\boldsymbol{x}) \Rightarrow \boldsymbol{x}^\star \in \operatorname*{argmin}_{\boldsymbol{x}} f(\boldsymbol{x}, \xi), \ \forall \xi . \tag{8}
$$

This condition says that the problem is "easy", in the sense that all the functions in the expectation share the same minimizer. This case morally corresponds to the case in which there is no noise on the stochastic gradients. However, this condition seems weaker because it says that only in the optimum the gradient is exact and noisy everywhere else. We will also assume that each function $f(\boldsymbol{x}, \xi)$ is L-smooth in the first argument.

Theorem 5 *Assume (H1) (H3'). Then, under the interpolation assumption in (8), Algorithm 1 with $\boldsymbol{\gamma}_t = \frac{\alpha \cdot \mathbf{1}}{\sqrt{\epsilon + \sum_{i=1}^{t} \alpha_i^2 \|\boldsymbol{g}_i\|^2}}, \epsilon > 0$ guarantees*

$$
\mathbb{E}\left[F(\boldsymbol{x}_T)\right] - F(\boldsymbol{x}^\star) \leq \frac{C}{T} \left(\sqrt{\epsilon + 4L^2 C^2 \ln^2 T + 4LC\sqrt{\epsilon}\ln T + \frac{2\alpha G^2}{\sqrt{\epsilon}}} + \frac{\alpha G^2}{\sqrt{\epsilon}} \right) .
$$

where $C \triangleq \left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha \right)$.

To the best of our knowledge, this is the first convergence rate for the last iterate of momentum methods in the interpolation setting.

Algorithm 2 Anytime Online-to-Batch (Cutkosky, 2019)

1: **Input:** Online learning algorithm $\mathcal{A}$ with convex domain D, $\alpha_1, ..., \alpha_T$, with $\alpha_1 > 0$.
2: Get Initial point $\boldsymbol{w}_1$ from $\mathcal{A}$
3: **for** $t = 1, \ldots, T$ **do**
4:     $\boldsymbol{x}_t = \frac{\sum_{i=1}^{t} \alpha_i \boldsymbol{w}_i}{\sum_{i=1}^{t} \alpha_i}$
5:     Play $\boldsymbol{x}_t$, receive subgradient $\boldsymbol{g}_t$
6:     Send $\ell_t(\boldsymbol{x}) = \langle \alpha_t \boldsymbol{g}_t, \boldsymbol{x} \rangle$ to $\mathcal{A}$ as the tth loss
7:     Get $\boldsymbol{w}_{t+1}$ from $\mathcal{A}$
8: **end for**

Algorithm 3 Anytime Online-to-Batch with FTRL

1: **Input:** $\alpha_1, ..., \alpha_T$, with $\alpha_t > 0$. $0 < \gamma_{t+1} \leq \gamma_t$.
2: Initialize $\boldsymbol{w}_1$
3: **for** $t = 1, \ldots, T$ **do**
4:     $\boldsymbol{x}_t = \frac{\sum_{i=1}^{t} \alpha_i \boldsymbol{w}_i}{\sum_{i=1}^{t} \alpha_i}$
5:     Play $\boldsymbol{x}_t$, receive subgradient $\boldsymbol{g}_t$
6:     $\boldsymbol{w}_{t+1} = \boldsymbol{w}_1 - \boldsymbol{\gamma}_t \sum_{i=1}^{t} \alpha_i \boldsymbol{g}_i$
7: **end for**

5.3. Proofs

Before presenting the proofs of our convergence rates, we revisit the Online-to-Batch algorithm (Algorithm 2) by Cutkosky (2019), which introduce a modification to any online learning algorithm to obtain a guarantee on the last iterate in the stochastic convex setting.

Lemma 6 *(Cutkosky, 2019, Theorem 1) Assume **(H2)**. Then, for all $\boldsymbol{x}^\star \in D$, Algorithm 2 guarantees*

$$\mathbb{E}[f(\boldsymbol{x}_T)] - f^\star \leq \mathbb{E}\left[\frac{R_T(\boldsymbol{x}^\star)}{\sum_{t=1}^{T} \alpha_t}\right] .$$

Set $\psi_t(\boldsymbol{x}) = \|\frac{\boldsymbol{x}_1 - \boldsymbol{x}}{\sqrt{\gamma_{t-1}}}\|^2, 1 \leq t \leq T$ as the regularizers of FTRL, where $\boldsymbol{\gamma}_{t+1} \leq \boldsymbol{\gamma}_t$ and $\boldsymbol{\gamma}_0 > 0$. Then, we write FTRL with loss $\ell_t(\boldsymbol{w}) = \langle \alpha_t \boldsymbol{g}_t, \boldsymbol{w} \rangle$ as

$$\boldsymbol{w}_t \in \underset{\boldsymbol{w} \in \mathbb{R}^d}{\operatorname{argmin}} \ \psi_t(\boldsymbol{w}) + \sum_{i=1}^{t-1} \langle \alpha_i \boldsymbol{g}_i, \boldsymbol{w} \rangle = \boldsymbol{w}_1 - \boldsymbol{\gamma}_{t-1} \sum_{i=1}^{t-1} \alpha_i \boldsymbol{g}_i .$$

We then plug FTRL into Algorithm 2 and it gives Algorithm 3. Hence, using the well-known regret upper bound of FTRL (Lemma 9 in the Appendix B), we get the following Lemma.

Lemma 7 *Under the same setting with Lemma 6, Algorithm 3 guarantees*

$$\mathbb{E}[f(\boldsymbol{x}_T)] - f^\star \leq \frac{1}{\sum_{t=1}^{T} \alpha_t} \mathbb{E}\left[\left\|\frac{\boldsymbol{u} - \boldsymbol{x}_1}{\sqrt{\boldsymbol{\gamma}_{T-1}}}\right\|^2 + \sum_{t=1}^{T} \langle \boldsymbol{\gamma}_{t-1}, \alpha_t^2 \boldsymbol{g}_t^2 \rangle\right] .$$

Now we prove the connection between the FTRL-based SGDM and Algorithm 3.

Proof [Proof of Theorem 4] We prove that the updates of $\boldsymbol{x}_t$ in Algorithm 1 can be one-to-one mapped to the updates of $\boldsymbol{x}_t$ Algorithm 3 when $\boldsymbol{w}_1 = \boldsymbol{x}_1$.

The update of $\boldsymbol{x}_t$ in Algorithm 3 can be written as following:

$$\boldsymbol{x}_{t+1} = \frac{\sum_{i=1}^{t} \alpha_i}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{x}_t + \frac{\alpha_{t+1}}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{w}_{t+1} = \frac{\sum_{i=1}^{t} \alpha_i}{\sum_{i=1}^{t+1} \alpha_i} \boldsymbol{x}_t + \frac{\alpha_{t+1}}{\sum_{i=1}^{t+1} \alpha_i} \left(\boldsymbol{w}_1 - \boldsymbol{\gamma}_t \sum_{i=1}^{t} \alpha_i \boldsymbol{g}_i \right) .$$

It is enough to prove that for any t, $\boldsymbol{\eta}_t \boldsymbol{m}_t = \frac{\alpha_{t+1}}{\sum_{i=1}^{t+1} \alpha_i} \left(\boldsymbol{\gamma}_t \sum_{i=1}^{t} \alpha_i \boldsymbol{g}_i \right)$. We claim it is true and prove it by induction.

When $t = 1$, it holds that $\boldsymbol{\eta}_1 \boldsymbol{m}_1 = \frac{\alpha_2 \alpha_1}{\alpha_1 + \alpha_2} \boldsymbol{\gamma}_1 \boldsymbol{g}_1$. Suppose it holds for $t = k - 1, k \geq 2$. Then in the case of $t = k$, we have

$$\boldsymbol{\eta}_k \boldsymbol{m}_k$$

$$= \left(\frac{\sum_{i=1}^{k-1} \alpha_i}{\sum_{i=1}^{k} \alpha_i} \boldsymbol{m}_{k-1} + \frac{\alpha_k}{\sum_{i=1}^{k} \alpha_i} \boldsymbol{g}_k \right) \cdot \frac{\alpha_{k+1} \sum_{i=1}^{k} \alpha_i}{\sum_{i=1}^{k+1} \alpha_i} \boldsymbol{\gamma}_k$$

$$= \left(\frac{\sum_{i=1}^{k-1} \alpha_i}{\sum_{i=1}^{k} \alpha_i} \left(\frac{1}{\eta_{k-1}} \frac{\alpha_k}{\sum_{i=1}^{k} \alpha_i} \boldsymbol{\gamma}_{k-1} \sum_{i=1}^{k-1} \alpha_i \boldsymbol{g}_i \right) + \frac{\alpha_k}{\sum_{i=1}^{k} \alpha_i} \boldsymbol{g}_k \right) \cdot \frac{\alpha_{k+1} \sum_{i=1}^{k} \alpha_i}{\sum_{i=1}^{k+1} \alpha_i} \boldsymbol{\gamma}_k$$

$$= \frac{\alpha_{k+1} \sum_{i=1}^{k} \alpha_i}{\sum_{i=1}^{k+1} \alpha_i} \boldsymbol{\gamma}_k \cdot \left(\frac{\sum_{i=1}^{k-1} \alpha_i}{\sum_{i=1}^{k} \alpha_i} \left(\frac{\sum_{i=1}^{k-1} \alpha_i \boldsymbol{g}_i}{\sum_{i=1}^{k-1} \alpha_i} \right) + \frac{\alpha_k}{\sum_{i=1}^{k} \alpha_i} \boldsymbol{g}_k \right) = \frac{\alpha_{k+1}}{\sum_{i=1}^{k+1} \alpha_i} \boldsymbol{\gamma}_k \sum_{i=1}^{k} \alpha_i \boldsymbol{g}_i .$$

where in the first equation we used the definitions of η_k and $\boldsymbol{m}_k$ and in the second equality we used the induction step. So we proved the above claim. Thus, we can directly use Lemma 7. ∎

The proof of Corollary 1 is immediate and we omit it, while the proofs of Corollaries 2-4 are standard and they are presented in the Appendix C. Instead, here we show the proof of Theorem 5.
Proof [Proof of Theorem 5] By Theorem 4, we have

$$\mathbb{E}\left[F(\boldsymbol{x}_T)\right] - F(\boldsymbol{x}^\star) \leq \frac{2}{T} \left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha \right) \sqrt{\mathbb{E} \sum_{t=1}^{T} \|\nabla f(\boldsymbol{x}_t, \xi_t)\|^2 + \epsilon} + \frac{\alpha G^2}{\sqrt{\epsilon}} . \quad (9)$$

Under the interpolation condition and L-smoothness of the functions f, it satisfies that

$$\mathbb{E} \sum_{t=1}^{T} \|\nabla f(\boldsymbol{x}_t, \xi_t)\|^2 \leq 2L \mathbb{E} \left[\sum_{t=1}^{T} \left(f(\boldsymbol{x}_t, \xi_t) - f(\boldsymbol{x}^\star, \xi_t) \right) \right] \leq 2L \sum_{t=1}^{T} \mathbb{E}\left[F(\boldsymbol{x}_t)\right] - F(\boldsymbol{x}^\star) .$$

Use (9) on each t to get

$$\sum_{t=1}^{T} \mathbb{E}\left[F(\boldsymbol{x}_t)\right] - F(\boldsymbol{x}^\star) \leq \sum_{t=1}^{T} \frac{1}{t} \left[\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha \right) \sqrt{\mathbb{E} \sum_{i=1}^{t} \|\nabla f(\boldsymbol{x}_i, \xi_i)\|^2 + \epsilon} + \frac{\alpha G^2}{\sqrt{\epsilon}} \right]$$

$$\leq \left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha \right) \cdot \left(\sqrt{\mathbb{E} \sum_{t=1}^{T} \|\nabla f(\boldsymbol{x}_t, \xi_t)\|^2 + \epsilon} + \frac{\alpha G^2}{\sqrt{\epsilon}} \right) \ln T .$$
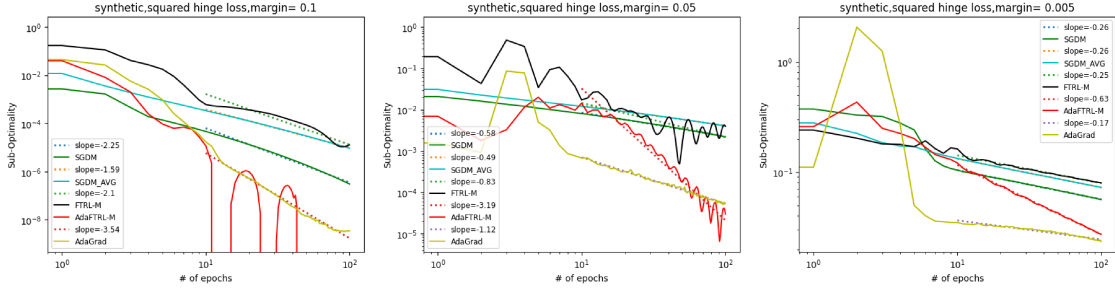
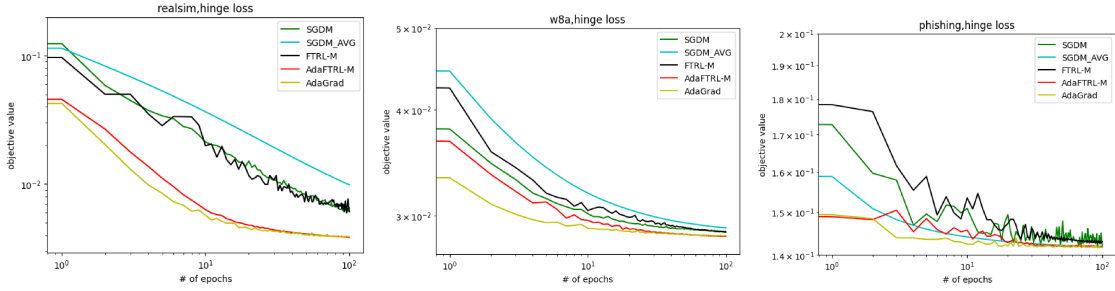Figure 1: Squared hinge loss for classification, objective value vs number of epoch.

Figure 2: Hinge loss for classification, objective value vs number of epoch.

Then, we solve for $\mathbb{E}\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t, \xi_t)\|^2$ and get

$$
\mathbb{E}\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t, \xi_t)\|^2
$$
$$
\leq 4L^2\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)^2 \ln^2 T + 4L\sqrt{\epsilon}\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\ln T + \frac{2\alpha G^2}{\sqrt{\epsilon}}\ .
$$

Using this expression in (9), we have the stated bound. ∎

6. Empirical Results

We have presented a family of FTRL-based SGDM algorithms, that exhibit optimal convergence of the last iteration. These algorithms are motivated by a new lower bound that shows that constant momentum SGDM is provably suboptimal to minimize convex Lipschitz functions. However, the theory guarantees only an improvement of $\ln T$, so it is unlikely to make a difference in practical applications. Yet, we also perform some experiments to show that FTRL-based momentum methods have also interesting empirical properties.

We compare FTRL-M (Algorithm 1, $\boldsymbol{\gamma}_t = \frac{c \cdot \mathbf{1}}{\sqrt{t}}$), AdaFTRL-M (Algorithm 1, $\boldsymbol{\gamma}_t = \frac{\alpha}{\sqrt{\epsilon + \sum_{i=1}^{t}\alpha_i^2 \boldsymbol{g}_i^2}}$) with classic SGDM ($\beta = 0.9$), SGDM-AVG (averaged iterates of SGDM, $\beta = 0.9$), and Ada-Grad (McMahan and Streeter, 2010; Duchi et al., 2010). The initial stepsizes for all the algorithms were tuned with a fine grid-searching procedure.

13

Synthetic Data For the first experiment, we generate synthetic data and test the algorithms following the protocol in Vaswani et al. (2019). We generate a synthetic binary classification dataset with $n = 8000$ and the dimension $d = 100$. We make the data linearly separable with a margin, in which case the interpolation condition is satisfied. We train linear classifiers with the squared hinge loss: $f(\boldsymbol{w}) = \sum_{i=1}^{n} \left(\max \left(0, 1 - \boldsymbol{y}_i \boldsymbol{w}^T \boldsymbol{x}_i \right) \right)^2$. Note that the loss function is smooth and $f(\boldsymbol{w}^\star) = 0$. In this case, the optimal convergence rate is at least as fast as $1/T$.

We plot the suboptimality gap versus the number of epochs with different margin values in Figure 1, in loglog plots. Also, we add a line to fit the curves, where the slopes represent the power of t. From Figure 1, we observe that two adaptive algorithms AdaGrad and AdaFTRL-M bring faster convergence and AdaFTRL-M has the biggest slope in all the cases. Also, the performance of FTRL-M is on par with SGDM and SGDM-AVG.

Real Data We also test the algorithms on real datasets. We use classification datasets from the LIBSVM website (Chang and Lin, 2011); *real-sim*, *w8a*, and *phishing*. The details of the datasets are in Appendix D.

We train linear classifiers with the hinge loss and no regularization: $f(\boldsymbol{w}) = \sum_{i=1}^{n} (\max(0, 1 - \boldsymbol{y}_i \boldsymbol{w}^T \boldsymbol{x}_i))$. The stochastic gradients are obtained evaluating the subgradient on one example at the time. We repeat the experiments for 5 times for each algorithm and report the average of 5 repetitions. We show the objective value versus the number of epochs in Figure 2.

The results show that the algorithms with non-adaptive stepsizes tend to perform worse than the ones with adaptive stepsizes. Moreover, the performance of AdaFTRL-M is close to the last iterate of AdaGrad and sometimes outperforms all the other algorithms, especially in the last iterations.

7. Conclusion

We have presented an analysis of the convergence of the last iterate of SGDM in the convex setting. We prove for the first time through a lower bound the suboptimal convergence rate for the last iterate of SGDM with constant momentum after T iterations. Moreover, we study a class of FTRL-based SGDM algorithms with increasing momentum and shrinking updates, of which the last iterate has optimal convergence rate without projections onto bounded domain nor knowledge of T. Furthermore, we present empirical results showing that FTRL-based SGDM with adaptive stepsize matches or outperforms the other similar algorithms in the last iterations.

In the future, we plan on studying the convergence in high probability of FTRL-based SGDM, similarly to the analysis in Li and Orabona (2020).

References

J. D. Abernethy, E. Hazan, and A. Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In Rocco A. Servedio and Tong Zhang, editors, *Proc. of Conference on Learning Theory (COLT)*, pages 263–274. Omnipress, 2008.

A. Agarwal, P. L. Bartlett, P. Ravikumar, and M. J. Wainwright. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *IEEE Transactions on Information Theory*, 58(5):3235–3249, 2012.

A. Alacaoglu, Y. Malitsky, P. Mertikopoulos, and V. Cevher. A new regret analysis for Adam-type algorithms. In *International Conference on Machine Learning*, pages 202–210. PMLR, 2020.

C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

A. Cutkosky. Anytime online-to-batch, optimism and acceleration. In K. Chaudhuri and R. Salakhutdinov, editors, *Proc. of the 36th International Conference on Machine Learning*, volume 97 of *Proc. of Machine Learning Research*, pages 1446–1454, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

A. Cutkosky and F. Orabona. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems*, pages 15236–15245, 2019.

A. Defazio. Understanding the role of momentum in non-convex optimization: Practical insights from a lyapunov analysis. *arXiv preprint arXiv:2010.00406*, 2020.

A. Dieuleveut, N. Flammarion, and F. Bach. Harder, better, faster, stronger convergence rates for least-squares regression. *J. Mach. Learn. Res.*, 18(1):3520–3570, January 2017.

J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *COLT*, 2010.

P. Gaillard, G. Stoltz, and T. Van Erven. A second-order bound with excess losses. In *Conference on Learning Theory*, pages 176–196. PMLR, 2014.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22 (4):1469–1492, 2012.

N. J. Harvey, C. Liaw, Y. Plan, and S. Randhawa. Tight analyses for non-smooth stochastic gradient descent. In *Conference on Learning Theory*, pages 1579–1613, 2019.

P. Jain, S. M. Kakade, R. Kidambi, P. Netrapalli, and A. Sidford. Accelerating stochastic gradient descent for least squares regression. In S. Bubeck, V. Perchet, and P. Rigollet, editors, *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 545–604. PMLR, 06–09 Jul 2018.

P. Jain, D. Nagaraj, and P. Netrapalli. Making the last iterate of SGD information theoretically optimal. In A. Beygelzimer and D. Hsu, editors, *Proc. of the Conference on Learning Theory (COLT)*, volume 99 of *Proceedings of Machine Learning Research*, pages 1752–1755, Phoenix, USA, 25–28 Jun 2019. PMLR.

P. Jain, D. M. Nagaraj, and P. Netrapalli. Making the last iterate of sgd information theoretically optimal. *SIAM Journal on Optimization*, 31(2):1108–1130, 2021. doi: 10.1137/19M128908X.

S. Jelassi and A. Defazio. Dual averaging is surprisingly effective for deep learning optimization. *arXiv preprint arXiv:2010.10502*, 2020.

R. Kidambi, P. Netrapalli, P. Jain, and S. Kakade. On the insufficiency of existing momentum schemes for stochastic optimization. In *2018 Information Theory and Applications Workshop (ITA)*, pages 1–9. IEEE, 2018.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

G. Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133 (1):365–397, 2012.

X. Li and F. Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *Proc. of the 22nd International Conference on Artificial Intelligence and Statistics, AISTATS*, 2019.

X. Li and F. Orabona. A high probability analysis of adaptive SGD with momentum. In *ICML 2020 Workshop on Beyond First Order Methods in ML Systems*, 2020.

Y. Liu, Y. Gao, and W. Yin. An improved analysis of stochastic gradient descent with momentum. *Advances in Neural Information Processing Systems*, 33, 2020.

L. Luo, Y. Xiong, Y. Liu, and X. Sun. Adaptive gradient methods with dynamic bound of learning rate. In *International Conference on Learning Representations*, 2018.

S. Ma, R. Bassily, and M. Belkin. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *International Conference on Machine Learning*, pages 3325–3334. PMLR, 2018.

H. B. McMahan and M. J. Streeter. Adaptive bound optimization for online convex optimization. In *COLT*, 2010.

D. Needell, N. Srebro, and R. Ward. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Mathematical Programming*, 155:549–573, 2015.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

A. S. Nemirovsky and D. Yudin. *Problem complexity and method efficiency in optimization*. Wiley, New York, NY, USA, 1983.

Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. In *Doklady AN SSSR (translated as Soviet. Math. Docl.)*, volume 269, pages 543–547, 1983.

Y. Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120(1):221–259, 2009.

Y. Nesterov and V. Shikhman. Quasi-monotone subgradient methods for nonsmooth convex minimization. *Journal of Optimization Theory and Applications*, 165(3):917–940, 2015.

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

F. Orabona and D. Pál. Scale-free online learning. *Theoretical Computer Science*, 716:50–69, 2018. Special Issue on ALT 2015.

B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

S. J. Reddi, A. Hefny, S. Sra, B. Poczos, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323, 2016.

S. J. Reddi, S. Kale, and S. Kumar. On the convergence of Adam and beyond. In *International Conference on Learning Representations*, 2018.

O. Sebbouh, R. M Gower, and A. Defazio. Almost sure convergence rates for stochastic gradient descent and stochastic heavy ball. In *Conference on Learning Theory*, pages 3935–3971. PMLR, 2021.

S. Shalev-Shwartz. *Online Learning: Theory, Algorithms, and Applications*. PhD thesis, The Hebrew University, 2007.

M. Streeter and H. B. McMahan. Less regret via online conditioning, 2010. arXiv:1002.4862.

I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.

W. Tao, Z. Pan, G. Wu, and Q. Tao. Primal averaging: A new gradient evaluation step to attain the optimal individual convergence. *IEEE transactions on cybernetics*, 50(2):835–845, 2018.

W. Tao, S. Long, G. Wu, and Q. Tao. The role of momentum parameters in the optimal convergence of adaptive Polyak's heavy-ball methods. In *International Conference on Learning Representations*, 2021.

S. Vaswani, F. Bach, and M. Schmidt. Fast and faster convergence of SGD for over-parameterized models and an accelerated Perceptron. In K. Chaudhuri and M. Sugiyama, editors, *Proc. of the 22nd International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proc. of Machine Learning Research*, pages 1195–1204. PMLR, 16–18 Apr 2019.

R. Ward, X. Wu, and L. Bottou. AdaGrad stepsizes: Sharp convergence over nonconvex landscapes. In *International Conference on Machine Learning*, pages 6677–6686. PMLR, 2019.

M. K. Warmuth and A. K. Jagota. Continuous and discrete-time nonlinear gradient descent: Relative loss bounds and convergence. In *Electronic proceedings of the 5th International Symposium on Artificial Intelligence and Mathematics*, volume 326, 1997.

T. Yang, Q. Lin, and Z. Li. Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization. *arXiv preprint arXiv:1604.03257*, 2016.

Appendix A. Lemma for the Proof of Theorem 1

Lemma 8 *For any $1 \leq j \leq t \leq T$ and $0 < \alpha \leq \frac{1}{2}$, we have $\frac{1}{T-j+1} \sum_{k=j+1}^{t} \frac{1}{j^\alpha} \leq \frac{2}{T^\alpha}$.*

Proof First, we observe that

$$\sum_{k=j+1}^{t} \frac{1}{k^\alpha} \leq \int_{j}^{t} \frac{1}{x^\alpha} dx = \frac{t^{1-\alpha} - j^{1-\alpha}}{1-\alpha} = \frac{1}{1-\alpha} \frac{t^{2-2\alpha} - j^{2-2\alpha}}{t^{1-\alpha} + j^{1-\alpha}}$$

$$\leq \frac{1}{1-\alpha} \frac{(2-2\alpha)t^{1-2\alpha}(t-j)}{t^{1-\alpha} + j^{1-\alpha}} \leq \frac{2(t-j)}{t^\alpha},$$

where in the second inequality we used the convexity of $f(x) = x^{2-2\alpha}, 0 < \alpha \leq \frac{1}{2}$.

Then, we claim $\frac{1}{T-j+1} \frac{t-j}{t^\alpha} \leq \frac{1}{T^\alpha}$.

Let $g(x) = \frac{x-j}{x^\alpha}$. The derivative $g'(x) = \frac{1-\alpha+\frac{j}{\alpha x}}{x^\alpha}$ is positive for all $x > 0$ and $j \geq 0$. So it satisfies that $\frac{t-j}{t^\alpha} \leq \frac{T-j}{T^\alpha}$, which implies the claim. ∎

Appendix B. Lemma for the Proof of Theorem 4

Algorithm 4 Follow-the-Regularized-Leader on Linearized Losses

---
1: **Input:** Regularizers $\psi_1, \ldots, \psi_T : \mathbb{R}^d \to (-\infty, \infty]$.
2: **for** $t = 1, \ldots, T$ **do**
3:     $\boldsymbol{w}_t \in \operatorname{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} \psi_t(\boldsymbol{w}) + \sum_{i=1}^{t-1} \langle \boldsymbol{g}_i, \boldsymbol{w} \rangle$
4:     Receive $\ell_t : \mathbb{R}^d \to (-\infty, \infty]$ and pay $\ell_t(\boldsymbol{w}_t)$
5:     Set $\boldsymbol{g}_t \in \partial \ell_t(\boldsymbol{w}_t)$
6: **end for**

The following lemma is a well-known result for FTRL (see, e.g., Orabona, 2019).

Lemma 9 *Let ℓ_t a sequence of convex loss functions. Set the sequence of regularizers as $\psi_t(\boldsymbol{x}) = \left\| \frac{\boldsymbol{x}_1 - \boldsymbol{u}}{\sqrt{\boldsymbol{\gamma}_{t-1}}} \right\|^2$, where $\boldsymbol{\gamma}_{t+1} \leq \boldsymbol{\gamma}_t$, $t = 1, \ldots, T$. Then, FTRL (Algorithm 4) guarantees*

$$\sum_{t=1}^{T} \ell_t(\boldsymbol{x}_t) - \ell_t(\boldsymbol{u}) \leq \left\| \frac{\boldsymbol{u} - \boldsymbol{x}_1}{\sqrt{\boldsymbol{\gamma}_{T-1}}} \right\|^2 + \frac{1}{2} \sum_{t=1}^{T} \langle \boldsymbol{\gamma}_{t-1}, \boldsymbol{g}_t^2 \rangle.$$

Appendix C. Proofs of Corollaries 2-4

First, we state some technical lemmas.

Lemma 10 *(Li and Orabona, 2019, Lemma 4) Let $f : \mathbb{R}^d \to \mathbb{R}$ be M -smooth and bounded from below, then for all $\boldsymbol{x} \in \mathbb{R}^d$*

$$\|\nabla f(\boldsymbol{x})\|^2 \leq 2M(f(\boldsymbol{x}) - \inf_{\boldsymbol{y} \in \mathbb{R}} f(\boldsymbol{y})) .$$

Lemma 11 *(Gaillard et al., 2014, Lemma 14) Let $a_0 > 0$ and $a_1, \ldots, a_m \in [0, A]$ be real numbers and let $f : (0, +\infty) \to [0, +\infty)$ nonincreasing function. Then*

$$\sum_{i=1}^{m} a_i f(a_0 + \cdots + a_{i-1}) \leq \int_{a_0}^{\sum_{i=0}^{m} a_i} f(u)du + Af(a_0) .$$

Proof Denote by $s_t = \sum_{i=0}^{t} a_i$.

$$
\begin{aligned}
\sum_{i=1}^{m} a_i f(s_{i-1}) &= \sum_{i=1}^{m} a_i f(s_i) + \sum_{i=1}^{m} a_i (f(s_{i-1}) - f(s_i)) \\
&\leq \sum_{i=1}^{m} a_i f(s_i) + A \sum_{i=1}^{m} (f(s_{i-1}) - f(s_i)) \\
&\leq \sum_{i=1}^{m} \int_{s_{i-1}}^{s_i} f(x)dx + A \sum_{i=1}^{m} (f(s_{i-1}) - f(s_i)) \\
&\leq \int_{a_0}^{\sum_{i=0}^{m} a_i} f(u)du + Af(a_0) ,
\end{aligned}
$$

where the first inequality holds because $f(x_{i-1}) \geq f(s_i)$ and $a_i \leq A$, while the second inequality uses the fact that f is nonincreasing together with $s_i - s_{i-1} = a_i$. ∎

We can now present the proofs of the Corollaries 2-4.

Proof [Proof of Corollary 2 and Corollary 3] By Lemma 11, for adaptive stepsize $\gamma_t = \frac{\alpha \cdot \mathbf{1}}{\sqrt{\epsilon + \sum_{i=1}^{t} \alpha_i^2 \|\boldsymbol{g}_i\|^2}}$, we have

$$\sum_{t=1}^{T} \gamma_{t-1} \|\boldsymbol{g}_t\|^2 = \sum_{t=1}^{T} \frac{\alpha \|\boldsymbol{g}_t\|^2}{\sqrt{\epsilon + \sum_{i=1}^{t-1} \|\boldsymbol{g}_i\|^2}} \leq 2\alpha \sqrt{\sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2} + \frac{\alpha G^2}{\sqrt{\epsilon}} .$$

Similarly for $\gamma_t = \frac{\alpha}{\sqrt{\epsilon + \sum_{i=1}^{t} \alpha_i^2 \boldsymbol{g}_i^2}}$, we have

$$\sum_{t=1}^{T} \langle \gamma_{t-1}, \boldsymbol{g}_t^2 \rangle = \sum_{j=1}^{d} \sum_{t=1}^{T} \frac{\alpha \boldsymbol{g}_{t,j}^2}{\sqrt{\epsilon + \sum_{i=1}^{t-1} \boldsymbol{g}_{i,j}^2}} \leq 2\alpha \sum_{j=1}^{d} \sqrt{\sum_{t=1}^{T} \boldsymbol{g}_{t,j}^2} + \frac{\alpha d G_\infty^2}{\sqrt{\epsilon}} .$$

∎

Proof [Proof of Corollary 4] By Corollary 2, we have

$$\mathbb{E}\left[f(\boldsymbol{x}_T)\right] - f^\star \leq \frac{1}{T}\left[\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\sqrt{\epsilon + \mathbb{E}\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2} + \frac{\alpha G^2}{\sqrt{\epsilon}}\right] . \qquad (10)$$

From the unbiasedness of the gradients, we have

$$\mathbb{E}\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2 \leq \mathbb{E}\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|^2 + T\sigma^2,$$

and

$$\mathbb{E}\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|^2 \leq 2L\sum_{t=1}^{T}\mathbb{E}\left[f(\boldsymbol{x}_t)\right] - f^\star$$

$$\leq 2L\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\sum_{t=1}^{T}\frac{\sqrt{\epsilon + \mathbb{E}\sum_{i=1}^{t}\|\boldsymbol{g}_i\|^2}}{t}$$

$$\leq 2L\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\cdot\left(\sqrt{\mathbb{E}\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2 + \epsilon} + \frac{\alpha G^2}{\sqrt{\epsilon}}\right)\ln T,$$

where in the second inequality we used Lemma 10 and Holder's and Jensen's inequalities in the third inequality.

Solve for $\mathbb{E}\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2$ to have

$$\mathbb{E}\sum_{t=1}^{T}\|\boldsymbol{g}_t\|^2$$

$$\leq 4L^2\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)^2\ln^2 T + 4L\sqrt{\epsilon}\left(\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^\star\|^2}{\alpha} + 2\alpha\right)\ln T + \frac{2\alpha G^2}{\sqrt{\epsilon}} + 2T\sigma^2 .$$

Putting it back to (10), we get the stated bound. ∎

Appendix D. Details of Experiments

Table 2: Real datasets

Name	# of Samples	# of Features
real-sim	72,309	20,958
w8a	49,749	300
phishing	11,055	68