The Power of Factorial Powers: New Parameter settings for (Stochastic) Optimization

Aaron Defazio ADEFAZIO@FB.COM
Facebook AI Research
770 Broadway, New York

Robert M. Gower GOWERROBERT@GMAIL.COM
Telecom Paris, IPP[1]

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

The convergence rates for convex and non-convex optimization methods depend on the choice of a host of constants, including step-sizes, Lyapunov function constants and momentum constants. In this work we propose the use of factorial powers as a flexible tool for defining constants that appear in convergence proofs. We list a number of remarkable properties that these sequences enjoy, and show how they can be applied to convergence proofs to simplify or improve the convergence rates of the momentum method, accelerated gradient and the stochastic variance reduced method (SVRG).

Keywords: List of keywords

1. Introduction

Consider the stochastic optimization problem

$$x_* \in \arg\min_{x \in C} f(x) = \mathbb{E}_\xi \left[f\left(x, \xi\right) \right], \tag{1}$$

where each $f\left(x, \xi\right)$ is convex but potentially non-smooth in x and $C \subset \mathbb{R}^d$ is a bounded convex set. To solve (1) we use an iterative method that at the kth iteration samples a stochastic (sub-)gradient $\nabla f(x_k, \xi)$ and uses this gradient to compute a new, and hopefully improved, x_{k+1} iterate. The simplest of such methods is *Stochastic Gradient Descent* (SGD) with projection:

$$x_{k+1} = \Pi_C \left(x_k - \eta_k \nabla f(x_k, \xi) \right), \tag{2}$$

where Π_C is the projection onto C and η_k is a sequence of step-sizes. Both the variance from the sampling procedure, as well as the non-smoothness of f prevent the sequence of x_k iterates from converging. The two most commonly used tools to deal with this variance are iterate averaging techniques (Polyak, 1964) and decreasing step-sizes (Robbins and Monro, 1951). By carefully choosing a sequence of averaging parameters and decreasing step-sizes we can guarantee that the variance of SGD will be kept under control and the method will converge. In this work we focus on an alternative to averaging: momentum. Momentum can be used as a replacement for averaging for non-smooth problems, both in the stochastic and non-stochastic setting. Projected SGD with

1. Research conducted while at Facebook AI Research New York

Method	Alg #	Smooth	Str. Conv	Polytopic Rate	Std. Rate	Reference
SGDM	Eq (18)	No	No	$(n+2)^{\overline{-1/2}}$	$(n+1)^{-1/2}$	Tao et al. (2020)
SGDM	Eq (18)	No	Yes	$(n+2)^{\overline{-1}}$	$(n+1)^{-1}$	Tao et al. (2020)
SVRGM	Alg 1	Yes	No	$1/n$	$1/n$	Allen Zhu & Yuan [2016]
SVRGM	Alg 1	Yes	Yes	$(3/5)^{n/\kappa}$	$(3/4)^{n/\kappa}$	Allen Zhu & Yuan [2016]
Nesterov	Eq (27)	Yes	No	$1/n^{\overline{2}}$	$1/n^2$	Nesterov (2013)

Table 1: List of convergence results together with previously known results. We say that the function is smooth if (7) holds with constant L, otherwise we assume that the function is G–Lipschitz (6). Finally when assuming the function if μ–strongly convex we use $\kappa := L/\mu$. The SVRGM is in fact a new method which is closely related to the SVRG++ Allen Zhu and Yuan (2016) method.

momentum can be written as

$$\begin{aligned} m_{k+1} &= \beta m_k + (1-\beta)\nabla f(x_k, \xi_k), \\ x_{k+1} &= \Pi_C\left(x_k - \alpha_k m_{k+1}\right), \end{aligned} \tag{3}$$

where α_k and β are step-size and momentum parameters respectively. Using averaging and momentum to handle variance introduces a new problem: choosing and tuning the additional sequence of parameters. In this work we introduce the use of factorial powers for the averaging, momentum, and step-size parameters. As we will show, the use of factorial powers simplifies and strengthens the convergence rate proofs.

Contributions

1. We introduce factorial powers as a tool for providing tighter or more elegant proofs for the convergence rates of methods using averaging, including dual averaging and Nesterov's accelerated gradient method, see row 5 in Table 1.
2. We leverage factorial powers to prove tighter any-time convergence rates for SGD with momentum in the non-smooth convex and strongly-convex cases, see rows 1 and 2 in Table 1.
3. We describe a novel SVRG variant with inner-loop factorial power momentum, which improves upon the SVRG++ (Allen Zhu and Yuan, 2016) method in both the convex and strongly convex case, see rows 3 and 4 in Table 1.
4. We identify and unify a number of existing results in the literature that make use of factorial power averaging, momentum or step-sizes.

2. Factorial Powers

The (rising) factorial powers (Graham et al., 1994) are typically defined using a positive integer r and a non-negative integer k as

$$k^{\overline{r}} = k(k+1)\cdots(k+r-1) = \prod_{i=1}^{r}(k+i-1). \tag{4}$$

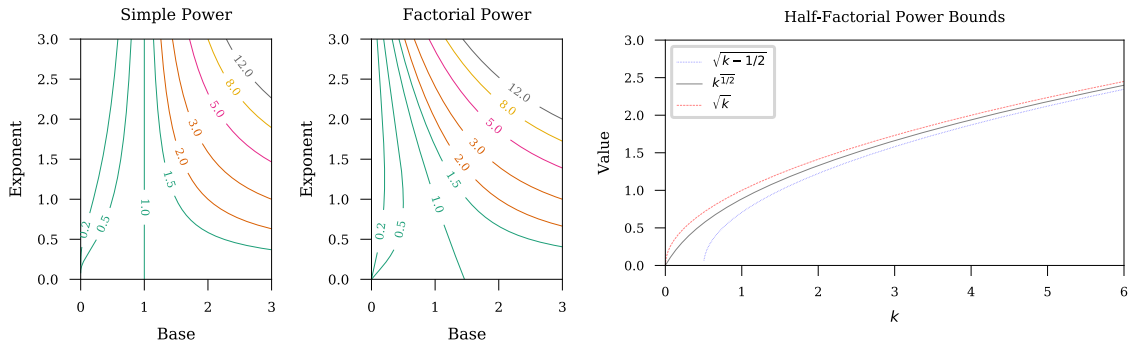

Figure 1: (left) Contour plots of the simple powers and the factorial powers. (right) The half-factorial power and associated upper and lower bounds.

Their behavior is similar to the simple powers k^r as $k^{\overline{r}} = O(k^r)$, and as we will show, they can typically replace the use of simple powers in proofs. They are closely related to the simplicial polytopic numbers $P_r(k)$ such as the triangular numbers $k(k+1)$, and tetrahedral numbers $\frac{1}{6}k(k+1)(k+2)$, by the relation $P_r(k) = \frac{1}{r!}\, k^{\overline{r}}$. See the left of Figure 1 for contour plots comparing factorial and simple powers.

The advantage of $k^{\overline{r}}$ over k^r is that in many cases that arise in proofs, additive, rather than multiplicative operations, are applied to the constants. As we show in Section 3, summation and difference operations applied to $k^{\overline{r}}$ result in other factorial powers, that is, factorial powers are *closed* under summation and differencing. In contrast, when summing or subtracting simple powers of the form k^r, the resulting quantities are polynomials rather than simple powers. It is this closure under summation and differencing that allows us to derive improved convergence rates when choosing step-sizes and momentum parameters based on factorial powers.

Our theory will use a generalization of the factorial powers to non-integers $r \in \mathbb{R}$ and integers $k \geq 1$ such that $k + r > 0$ using the *Gamma function* $\Gamma(k) := \int_0^\infty x^{k-1}e^{-x}dx$ so that

$$\boxed{k^{\overline{r}} := \frac{\Gamma(k+r)}{\Gamma(k)}} \tag{5}$$

We also use the convention that $0^{\overline{r}} = 0$ except for $0^{\overline{0}} = 1$. This is a proper extension because, when k is integer we have that $\Gamma(k) = (k-1)!$ and consequently (5) is equal to (4). This generalized sequence is particularly useful for the values $r = 1/2$ and $r = -1/2$, as they may replace the use of $\sqrt{k}$ and $1/\sqrt{k}$ respectively in proofs.

The factorial powers can be computed efficiently using the log-gamma function to prevent overflow. Using the factorial powers as step-sizes or momentum constants adds no computational overhead as they may be computed recursively using simple algebraic operations as we show below. The base values for the recursion may be precomputed as constants to avoid the overhead of gamma function evaluations entirely.

2.1. Notation and Assumptions

We assume throughout that $f(x,\xi)$ is convex in x. Let $\nabla f(x,\xi_k)$ denote the subgradient of $f(x,\xi_k)$ given to the optimization algorithm at step k. Let $C \subset \mathbb{R}^d$ be a convex set and let $R > 0$ be the radius of the smallest Euclidean-norm ball around the origin that contains the set C. We define the projection onto C as $\Pi_C(x) := \arg\min_{z\in C} \|z - x\|$. In addition to assuming that $f(x,\xi)$ is convex, we will use one of the following two sets of assumptions depending on the setting.

Non-smooth functions. The function $f(\cdot,\xi)$ is Lipschitz with constant $G > 0$ for all ξ, that is

$$|f(x,\xi) - f(y,\xi)| \;\leq\; G\,\|x - y\|, \quad \forall x, y \in \mathbb{R}^d. \tag{6}$$

Smooth functions The gradient $\nabla f(\cdot,\xi)$ is Lipschitz with constant $L > 0$ for all ξ, that is

$$\|\nabla f(x,\xi) - \nabla f(y,\xi)\| \;\leq\; L\,\|x - y\|, \quad \forall x, y \in \mathbb{R}^d. \tag{7}$$

We assume that $\sigma^2 < \infty$ where $\sigma^2 = \mathbb{E}_\xi \left\|\nabla f\left(x_*,\xi\right)\right\|^2$.

Strongly convex functions We say that $f(x)$ is μ–strongly convex if $f(x) - \frac{\mu}{2}\|x\|^2$ is convex.

We use the shorthand notation $\mathbb{E}_\xi \|\cdot\|^2 = \mathbb{E}_\xi \left[\|\cdot\|^2\right]$ and will write $\mathbb{E}$ instead of $\mathbb{E}_\xi$ when the conditional context is clear. We defer all proofs to the supplementary material.

3. Properties of Factorial Powers

The factorial powers obey a number of properties, see Table 2. These properties allow for a type of "finite" or "umbral" calculus that uses sums instead of integrals (Graham et al., 1994). A few of these properties, such as the summation and differencing, are given in Chapter 2.6 for integers values in (Graham et al., 1994). We carefully extend these properties to the non-integer setting. All the proofs of these properties can be found in Section A in the supplementary material.

These properties are key for deriving simple and tight convergence proofs. For instance, often when using telescoping in a proof of convergence, we often need a *summation* property. For the factorial powers we have the simple formula (10). This shows that the factorial powers are *closed* under summation because on both sides of (10) we have factorial powers. This formula is a discrete analogue of the definite integral $\int_a^b x^r dx = \frac{1}{r+1}b^{r+1} - \frac{1}{r+1}a^{r+1}$. In contrast, when summing power sequences, we rely on Faulhaber's formula:

$$\sum_{i=1}^{k} i^r = \frac{k^{r+1}}{r+1} + \frac{1}{2}k^r + \sum_{i=2}^{r} \frac{B_i}{i!}\frac{r!}{(r-k+1)!}k^{r-i+1}, \tag{14}$$

which involves the Bernoulli numbers $B_j := \sum_{i=0}^{j}\sum_{\nu=0}^{i}(-1)^v\binom{i}{\nu}\frac{(\nu+1)^j}{i+1}$. This is certainly not as simple as (10). Furthermore, to extend (14) to non-integer r complicates matters further (McGown and Parks, 2007). In contrast the summation property (10) holds for non-integer values.

Another common property used in telescoping arguments is the *difference* property (11). Once again we have that factorial powers are closed under differencing. In contrast, the simple powers instead require the use of inequalities such as

$$\begin{aligned} rx^{r-1} \leq\; & (x+1)^r - x^r \;\leq r\,(x+1)^{r-1}, \\ r\,(x+1)^{r-1} \leq\; & (x+1)^r - x^r \;\leq rx^{r-1}, \end{aligned}$$

Recursion	$(k+1)^{\overline{r}} = \frac{k+r}{k} k^{\overline{r}}$	(8)
	$(k+1)^{\overline{r}} = (k+r)\,(k+1)^{\overline{r-1}}$	(9)
Summation	$\sum_{i=a}^{b} i^{\overline{r}} = \frac{1}{r+1} b^{\overline{r+1}} - \frac{1}{r+1} a^{\overline{r+1}}$	(10)
Differences	$(k+1)^{\overline{r}} - k^{\overline{r}} = r\,(k+1)^{\overline{r-1}}$	(11)
Ratios	$\frac{k^{\overline{r+q}}}{k^{\overline{r}}} = (k+r)^{\overline{q}}$	(12)
Inversion	$k^{\overline{-r}} = \frac{1}{(k-r)^{\overline{r}}}$	(13)

Table 2: Fundamental Properties of the factorial powers. Properties (8), (9), (11) and (12) hold for $k + r > 0$ and $k \geq 1$. The Summation property (10) holds for $a + r > 0$ and $a \geq 1$. The Inversion property (13) holds for $k > r$ and $k \geq 1$. We are not aware of an existing source for these properties for the rising factorial powers, however similar relations for the falling factorial powers are established in Graham et al. (1994).

where the first row of bounds hold for $r < 0$ or $r > 1$ and the second row holds for $r \in (0, 1)$. Using the above bounds adds slack into the convergence proof and ultimately leads to suboptimal convergence rates.

3.1. Half-Powers

The factorial *half*–powers $k^{\overline{1/2}}$ and $k^{\overline{-1/2}}$ are particularly interesting since they can be used to set the learning rate of the momentum method in lieu of the standard $O(1/\sqrt{k})$ learning rate, as we will show in Theorem 2. The factorial half–powers are similar to the standard half-powers, in that, their growth is sandwiched by the standard half-powers as illustrated in Figure 1, in fact:

$$\sqrt{(k-1/2)} \leq k^{\overline{1/2}} \leq \sqrt{k}, \tag{15}$$

$$\frac{1}{\sqrt{k-1/2}} < k^{\overline{-1/2}} < \frac{1}{\sqrt{k-1}}. \tag{16}$$

We also believe this is the first time factorial half–powers have been used in the stochastic optimization literature.

4. From Averaging to Momentum

Here we show that averaging techniques and momentum techniques have a deep connection. We use this connection to motivate the use of factorial power momentum. Our starting point for this is SGD

with averaging which can be written using the online updating form

$$\begin{aligned} x_{k+1} &= \Pi_C \left(x_k - \eta_k \nabla f(x_k, \xi_k) \right), \\ \bar{x}_{k+1} &= (1 - c_{k+1})\, \bar{x}_k + c_{k+1} x_{k+1}. \end{aligned} \tag{17}$$

At first glance (17) is unrelated to SGD with momentum (3). But surprisingly, SGD with momentum can be re-written in the strikingly similar iterate averaging form given by

$$\begin{aligned} z_{k+1} &= \Pi_C \left(z_k - \eta_k \nabla f(x_k, \xi_k) \right), \\ x_{k+1} &= (1 - c_{k+1})\, x_k + c_{k+1} z_{k+1}. \end{aligned} \tag{18}$$

This equivalence only holds without the projection operation in Equation 3. We are not aware of any analysis of Equation 3's convergence with the projection operation included, and we believe that incorporating projection as we do in Equation 18 is better given it's much more amenable to analysis. The following theorem rephrases this equivalence, established by Sebbouh et al. (2020), in terms of the constants α and β:

Theorem 1 *If $C = \mathbb{R}^d$ then the x_k iterates of (3) and (18) are the same so long as $z_0 = x_0$, $c_1 \in (0,\, 1)$ and*

$$\eta_k = \frac{\alpha_k}{c_{k+1}}(1 - \beta), \quad c_{k+1} = \beta \frac{\alpha_k}{\alpha_{k-1}} \frac{c_k}{1 - c_k}. \tag{19}$$

Proof The proof is by induction.

Base case $k = 0$. From (3) we have that

$$x_1 = x_0 - \alpha_0 m_1 \overset{(3)}{=} x_0 - \alpha_0 (1 - \beta) \nabla f(x_0, \xi_0), \tag{20}$$

where we used that $m_0 = 0$. Similarly for (18) we have that

$$\begin{aligned} x_1 &= (1 - c_1)\, x_0 + c_1 z_1 \\ &= (1 - c_1)\, x_0 + c_1 (x_0 - \eta_0 \nabla f(x_0, \xi_0)) \\ &= x_0 - c_1 \eta_0 \nabla f(x_0, \xi_0), \end{aligned} \tag{21}$$

where we used that $z_0 = x_0$. Now (21) and (20) are equivalent since $c_1 \eta_0 \overset{(19)}{=} \alpha_0 (1 - \beta)$.

Induction step. Suppose that the x_k iterates in (17) and (18) are equivalent for k and let us consider the $k + 1$ step. Let

$$z_{k+1} = x_k - \frac{\alpha_k}{c_{k+1}} m_{k+1}. \tag{22}$$

Consequently

$$\begin{aligned} z_{k+1} &= x_k - \frac{\alpha_k}{c_{k+1}} m_{k+1} \\ &\overset{(18)+(3)}{=} (x_{k-1} - \alpha_{k-1} m_k) \\ &\quad - \frac{\alpha_k}{c_{k+1}} (\beta m_k + (1-\beta)\nabla f(x_k, \xi_k)) \\ &= x_{k-1} - \left(\alpha_{k-1} + \beta \frac{\alpha_k}{c_{k+1}}\right) m_k \\ &\quad - \frac{\alpha_k}{c_{k+1}} (1-\beta)\nabla f(x_k, \xi_k) \\ &\overset{(19)}{=} x_{k-1} - \frac{\alpha_{k-1}}{c_k} m_k - \eta_k \nabla f(x_k, \xi_k) \\ &\overset{(22)}{=} z_k - \eta_k \nabla f(x_k, \xi_k), \end{aligned}$$

where in the last but one step we used that $c_{k+1} = \beta \frac{\alpha_k}{\alpha_{k-1}} \frac{c_k}{1-c_k}$ which when re-arranged gives

$$\alpha_{k-1} + \beta \frac{\alpha_k}{c_{k+1}} = \frac{\alpha_{k-1}}{c_k}.$$

Finally

$$\begin{aligned} x_{k+1} &= x_k - \alpha_k m_{k+1} \\ &\overset{(22)}{=} x_k - c_{k+1}(x_k - z_{k+1}) \\ &= (1 - c_{k+1}) x_k + c_{k+1} z_{k+1}. \end{aligned}$$

Which concludes the induction step and the proof. ■

Due to this equivalence, we refer to (18) as the projected SGDM method. The x_k update (18) is similar to the moving average in (17), but now the averaging occurs directly on the x_k sequence that the gradient is evaluated on. As we will show, convergence rates of the SGDM method can be shown for the x_k sequence, with no additional averaging necessary. This method is also known as *primal-averaging*, and under this name it was explored by Sebbouh et al. (2020) in the context of smooth optimization and by Tao et al. (2020) and Taylor and Bach (2019) without drawing an explicit link to stochastic momentum methods.

Factorial powers play a key role in the choice of the *momentum parameters* c_{k+1}, and the resulting convergence rate of (17). Standard (equal-weighted) averaging given by

$$\begin{aligned} \bar{x}_k &:= \frac{1}{k+1} \sum_{i=0}^{k} x_i \quad \text{or equivalently} \\ \bar{x}_k &:= \left(1 - \frac{1}{k+1}\right) \bar{x}_{k-1} + \frac{1}{k+1} x_k. \end{aligned} \tag{23}$$

results in a sequence that "forgets the past" at a rate of $1/k$. Indeed, if we choose an arbitrary initial point x_0 (or at least without any special insight), to converge to the solution we must "forget" x_0. To forget x_0 faster, we can use a weighted average that puts more weight on recent iterates. We propose

the use of the factorial powers to define a family of such weights that allows us to tune how fast we forget the past. In particular, we propose the use of momentum constants as described in the following proposition.

Proposition 1 *Let $x_k \in \mathbb{R}^n$ for $k = 1, \ldots$ be a sequence of iterates, and let $r > -1$ be a real number. For $k \geq 0$, the factorial power average*

$$\boxed{\bar{x}_k = \frac{r+1}{(k+1)^{\overline{r+1}}} \sum_{i=0}^{k} (i+1)^{\overline{r}} x_i} \tag{24}$$

is equal to the moving average

$$\bar{x}_{k+1} = (1 - c_k)\, \bar{x}_k + c_k x_{k+1}, \tag{25}$$

where $c_k := \dfrac{r+1}{k+r+1}$.

Shamir and Zhang (2013) introduced the *polynomial-decay averaging* (25) for averaged SGD under the restriction that integer $r > 0$. Proposition 1 extends the result to non-integer values with a range of $r > -1$. Next we use factorial power averaging to get state-of-the-art convergence results for `SGDM`.

4.1. Applying factorial powers

The any-time convergence of `SGDM` is a good case study for the application of the half-factorial powers.

Theorem 2 *Let $f(x, \xi)$ be G-Lipschitz and convex in x. The projected `SGDM` method (18) with $\eta_k = \eta(k+1)^{\overline{-1/2}}$ for $\eta > 0$ and $c_{k+1} = 1/(k+1)$ converges according to*

$$\mathbb{E}\left[f(x_n) - f(x_*)\right] \;\leq\; \frac{1}{2}\left(\eta^{-1}R^2 + 2\eta G^2\right)(n+2)^{\overline{-1/2}}.$$

Furthermore, optimizing over η gives $\eta = \sqrt{1/2}\frac{R}{G}$ and the resulting convergence

$$\mathbb{E}\left[f(x_n) - f(x_*)\right] \leq \sqrt{2}RG\,(n+2)^{\overline{-1/2}}.$$

This result is strictly tighter than the $\sqrt{2}RG/\sqrt{n+1}$ convergence rate that arises from the use of square-root sequences (see Theorem 10 in the appendix) as used by Tao et al. (2020). The use of half-factorial powers also yields more direct proofs, as inequalities are replaced with equalities in many places. For instance, when $\eta_k = \eta/\sqrt{k+1}$, a bound of the following form arises in the proof:

$$\sqrt{k+1} - \sqrt{k} \leq \frac{1}{2\sqrt{k}}.$$

If factorial power step sizes $\eta_k = \eta(k+1)^{\overline{-1/2}}$ are used instead, then this bounding operation is replaced with an equality that we call the inverse difference property:

$$\frac{1}{(k+1)^{\overline{-1/2}}} - \frac{1}{k^{\overline{-1/2}}} = \frac{1}{2}\frac{1}{k^{\overline{1/2}}}.$$

The standard proof also requires summing the step-sizes, requiring another bounding operation

$$\sum_{i=0}^{k} \frac{1}{\sqrt{i+1}} \leq 2\sqrt{k+1}.$$

Again when the factorial power step-sizes are used instead, this inequality is replaced by the equality $\sum_{i=0}^{k}(i+1)^{\overline{-1/2}} = 2\,(k+1)^{\overline{1/2}}$.

We can also use factorial power momentum with $r = 3$ to show that SGDM converges at a rate of $\mathcal{O}(1/n)$ for strongly-convex non-smooth problems in the following theorem.

Theorem 3 *Let $f(x, \xi)$ be G-Lipschitz and $\mu-$strongly convex in x for every ξ. The projected SGDM method (18) with $\eta_k = \frac{1}{\mu(k+1)}$ and $c_{k+1} = \frac{4}{k+4}$ (i.e. $r = 3$) satisfies*

$$\mathbb{E}\left[f(x_n) - f(x_*)\right] \leq \frac{2G^2}{\mu}\,(n+2)^{\overline{-1}} = \frac{2G^2}{\mu(n+1)}.$$

This $\mathcal{O}(1/n)$ rate of convergence is the fastest possible in this setting (Agarwal et al., 2009). This rate of convergence has better constants than that established by using a different momentum scheme in Tao et al. (2020). Higher order averaging is also necessary to obtain this rate for the averaged SGD method, as established by Lacoste-Julien et al. (2012) and Shamir and Zhang (2013), however in that case only $r = 1$ averaging is necessary to obtain the same rate. The $r = 3$ front weighted average corresponds to a much heavier front weighting sequence:

$$x_k = \frac{4}{(k+1)(k+2)(k+3)(k+4)} \sum_{i=0}^{k} (i+1)(i+2)(i+3) z_i \tag{26}$$

5. From Momentum to Acceleration

A higher order r for the factorial powers is useful when the goal is to achieve convergence rates of the order $\mathcal{O}(1/n^{r+1})$. Methods using equal weighted $r = 0$ momentum cannot achieve convergence rates faster than $\mathcal{O}(1/n)$, since that is the rate that they "forget" the initial conditions. To see this, note that in a sum $1/(n+1)\sum_{i=0}^{n} z_i$, the z_0 value decays at a rate of $\mathcal{O}(1/n)$. When using the order r factorial power for averaging (24), the initial conditions are forgotten at a rate of $\mathcal{O}(1/n^{r+1})$. The need for $r = 1$ averaging arises in a natural way when developing accelerated optimization methods for non-strongly convex optimization, where the best known rates are of the order $\mathcal{O}(1/n^2)$ obtained by Nesterov's method. As with the SGDM method, Nesterov's method can also be written in an equivalent iterate averaging form (Auslender and Teboulle, 2006):

$$\begin{aligned} y_k &= (1 - c_{k+1})\, x_k + c_{k+1} z_k, \\ z_{k+1} &= z_k - \rho_k \nabla f(y_k), \\ x_{k+1} &= (1 - c_{k+1})\, x_k + c_{k+1} z_{k+1}, \end{aligned} \tag{27}$$

where ρ_k are the step-sizes, and initially $z_0 = x_0$. In this formulation of Nesterov's method we can see that the x_k sequence uses iterate averaging of the form (18). To achieve accelerated rates with this method, the standard approach is to use $\rho_k = 1/(L c_{k+1})$ and to choose momentum constants c_k that satisfy the inequality

$$c_k^{-2} - c_k^{-1} \leq c_{k-1}^{-2}.$$

Algorithm 1 Our proposed SVRGM method

$z^0_{m_0-1} = x^0_{m_0-1} = x_0$
for $s = 1, 2, \ldots, S$ **do**
  $\tilde{x}^{s-1} = x^{s-1}_{m_{s-1}-1},\ x^s_0 = x^{s-1}_{m_{s-1}-1},\ z^s_0 = z^{s-1}_{m_{s-1}}$
  $\nabla f(\tilde{x}^{s-1}) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\tilde{x}^{s-1})$
  for $t = 0, 1, \ldots, m_s - 1$ **do**
    Sample j uniformly at random
    $g^s_t = \nabla f_j(x^s_t) - \left[\nabla f_j\left(\tilde{x}^{s-1}\right) - \nabla f(\tilde{x}^{s-1})\right]$
    $z^s_{t+1} = z^s_t - \eta g^s_t$
    $x^s_{t+1} = (1 - c_{t+1})\, x^s_t + c_{t+1} z^s_{t+1}$
  end
end

This inequality is satisfied with equality when using the following recursive formula:

$$c^{-1}_{k+1} = \tfrac{1}{2}\left(1 + \sqrt{1 + 4c^{-1}_{k-1}}\right),$$

but the opaque nature and lack of closed form for this sequence is unsatisfying. Remarkably, the sequence $c_{k+1} = 2/(k+2)$ also satisfies this inequality, as pointed out by Tseng (2008), which is a simple application of $r = 1$ factorial power momentum. We show in the supplementary material how using factorial powers together with the iterate averaging form of momentum gives an elegant proof of convergence for this method, which uses the same proof technique and Lyapunov function as the proof of convergence of the regular momentum method `SGDM` . By leveraging the properties of factorial powers, the proof follows straightforwardly with no "magic" steps.

Theorem 4 *Let x_k be given by (27). Let $f(x, \xi)$ be L–smooth and convex. If we set $c_k = 2/(k+2)$ and $\rho_k = (k+1)/(2L)$ then*

$$f(x_n) - f(x_*) \leq \frac{2L}{n^2}\left\|x_0 - x_*\right\|^2. \tag{28}$$

This matches the rate given by Beck and Teboulle (2009) asymptotically, and is faster than the rate given by Nesterov's estimate sequence approach Nesterov (2013) by a constant factor.

6. Variance Reduction with Momentum

Since factorial power momentum has clear advantages in situations where averaging of the iterates is otherwise used, we further explore a problem where averaging is necessary and significantly complicates matters: the stochastic variance-reduced gradient method (SVRG). The SVRG method (Johnson and Zhang, 2013) is a double loop method, where the iterations in the inner loop resemble SGD steps, but with an additional additive variance reducing correction. In each outer loop, the average of the iterates from the inner loop are used to form a new "snapshot" point. We propose the SVRGM method (Algorithm 1). This method modifies the improved SVRG++ formulation of Allen Zhu and Yuan (2016) to further include the use of iterate averaging style momentum in the inner loop. See Algorithm 1.

Our formulation has a number of advantages over existing schemes. In terms of simplicity, it includes no resetting operations[1], so the x and z sequences start each outer loop at the values from the end of the previous one. Additionally, the *snapshot* $\tilde{x}$ is up-to-date, in the sense that it matches the final output point x from the previous step, rather than being set to an average of points as in SVRG/SVRG++.

The non-strongly convex case is an application of non-integer factorial power momentum. Using a large step-size $\eta = 1/6L$ we show in Theorem 5 that Algorithm 1 converges at a favourable rate if we choose the momentum parameters c_k corresponding to a $(k+1)^{\overline{1/2}}$ factorial power averaging of the iterates. The strongly convex case in Theorem 6 uses fixed momentum (i.e. an exponential moving average), since no rising factorial sequence can give linear convergence rates. In both cases we are able to give improved constants over the SVRG++ method.

Theorem 5 *(non-strongly convex case) Let* $f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$ *where each* f_i *is* L*-smooth and convex. By setting* $c_t = \frac{1/2+1}{t+1/2+1}$, $\eta = \frac{1}{6L}$, *and* $m_s = 2m_{s-1}$ *in Algorithm 1 we have that*

$$\mathbb{E}\left[f(x^S_{m_s-1}) - f^*\right] \leq \frac{f(x_0) - f^*}{2^S} + \frac{9L\left\|x_0 - x_*\right\|^2}{2^S m_0}.$$

The non-strongly convex convergence rate is linear in the number of epochs, however each epoch is twice as long as the previous one, resulting in an overall $1/t$ rate.

Theorem 6 *(strongly convex case) Let* $f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$ *where each* f_i *is* L*-smooth and* μ*-strongly convex. Let* $\kappa = L/\mu$. *By setting* $m_s = 6\kappa$, $c_k = \frac{5}{3}\frac{1}{4\kappa+1}$, *and* $\eta_k = 1/(10L)$ *in Algorithm 1 we have that*

$$\mathbb{E}\left[f(\tilde{x}^S) - f^*\right] \leq \left(\frac{3}{5}\right)^S \left[f(x_0) - f(x_*) + \frac{3}{4}\mu\,\delta_0\right],$$

where $\delta_0 := \left\|x_0 - x_*\right\|^2$.

7. Further Applications

Factorial powers have applications across many areas of optimization theory. We detail two further instances of popular first order methods where factorial powers are particularly useful.

7.1. Dual Averaging

Classical (non-stochastic) dual averaging uses updates of the form (Nesterov, 2009):

$$\begin{aligned} s_{k+1} &= s_k + \nabla f(x_k), \\ x_{k+1} &= \arg\min_x \left\{\langle s_{k+1}, x\rangle + \hat{\beta}_{k+1}\frac{\gamma}{2}\left\|x - x_0\right\|^2\right\}, \end{aligned} \tag{29}$$

where the sequence $\hat{\beta}_k$ is defined recursively with $\hat{\beta}_0 = \hat{\beta}_1 = 1$, and $\hat{\beta}_{k+1} = \hat{\beta}_{k+1} + 1/\hat{\beta}_{k+1}$. This sequence grows approximately following the square root, as $\sqrt{2k-1} \leq \hat{\beta}_{k+1} \leq \frac{1}{1+\sqrt{3}} + \sqrt{2k-1}$ for $k \geq 1$, and obeys a kind of summation property $\sum_{i=0}^{k}\frac{1}{\hat{\beta}_i} = \hat{\beta}_{k+1}$. Nesterov's sequence has the

1. This is also a feature of the variant known as *free-SVRG* (Sebbouh et al., 2019)

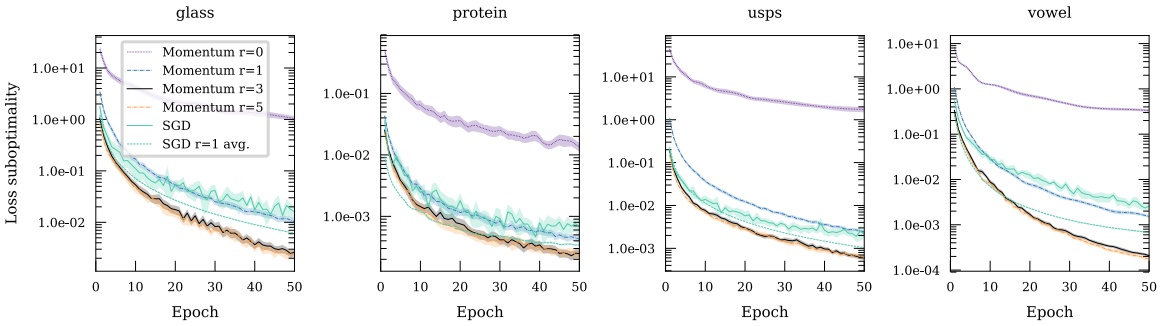

Figure 2: Training loss sub-optimality on 4 LIBSVM test problems, comparing SGD, SGD with $r = 1$ post-hoc averaging to SGD with factorial power momentum.

disadvantage of not having a simple closed form, but it otherwise provides tighter bounds than using $\beta_k = \sqrt{k+1}$. In particular, the precise bound on the duality gap (as we show in Theorem 15 in the supplementary material) is given by

$$\max_{x,\|x\|\leq R} \left\{ \frac{1}{n+1} \sum_{i=0}^{n} \langle \nabla f(x_i), x_i - x \rangle \right\} \leq \left(\frac{\sqrt{2}}{\left(1+\sqrt{3}\right)} \frac{1}{(n+1)} + \frac{2}{\sqrt{n+1}} \right) RG.$$

The factorial powers obey a similar summation relation, and they have the advantage of an explicit closed form, which we exploit to give a strictly tighter convergence rate.

Theorem 7 *After n steps of the dual averaging method (29) with $\hat{\beta}_k = 1/\left(k+1\right)^{\overline{-1/2}}$ and $\gamma = G/R$ we have that*

$$\max_{x,\|x\|\leq R} \left\{ \frac{1}{n+1} \sum_{i=0}^{n} \langle \nabla f(x_i), x_i - x \rangle \right\} \leq 2RG(n+2)^{\overline{-1/2}} < \frac{2RG}{\sqrt{n+1}}.$$

Proof Nesterov (2009) establishes the following bound:

$$\delta_k \leq \gamma \hat{\beta}_{k+1} R^2 + \frac{1}{2} G^2 \frac{1}{\gamma} \sum_{i=0}^{k} \frac{1}{\hat{\beta}_i}.$$

We use $\hat{\beta}_i = 1/\left(i+1\right)^{\overline{-1/2}}$ the sum is:

$$\sum_{i=0}^{k} \frac{1}{\hat{\beta}_i} = \frac{1}{1-1/2} \left(k+1\right)^{\overline{1/2}} - \frac{1}{1-1/2} \left(1\right)^{\overline{1/2}}.$$

Recall also that:

$$\hat{\beta}_{k+1} = \frac{1}{(k+2)^{\overline{-1/2}}} = (k+3/2)^{\overline{1/2}}.$$

So:

$$\delta_k \leq \gamma R^2 (k+3/2)^{\overline{1/2}} + G^2 \left((k+1)^{\overline{1/2}} - 2(1)^{\overline{1/2}} \right).$$

Using step-size $\gamma = G/R$:

$$\begin{aligned}
\delta_k &\leq RG (k+3/2)^{\overline{1/2}} + RG \left((k+1)^{\overline{1/2}} - 2(1)^{\overline{1/2}} \right) \\
&= RG \left((k+3/2)^{\overline{1/2}} + (k+1)^{\overline{1/2}} - 2(1)^{\overline{1/2}} \right) \\
&\leq 2RG (k+1)^{\overline{1/2}}.
\end{aligned}$$

Now to normalize by $1/(k+1)$ we use:

$$\frac{(k+1)^{\overline{r+q}}}{(k+1)^{\overline{r}}} = (k+1+r)^{\overline{q}},$$

with $r = 1$ and $q = -1/2$, so that:

$$\frac{(k+1)^{\overline{1/2}}}{k+1} = (k+2)^{\overline{-1/2}}.$$

We further use $(k+2)^{\overline{-1/2}} < (k+1)^{-1/2}$, giving:

$$\frac{1}{k+1}\delta_k < \frac{2RG}{\sqrt{k+1}}.$$

■

7.2. Conditional Gradient Method

Factorial power step-size schemes have also arisen for the conditional gradient method

$$\begin{aligned}
p_{k+1} &= \arg\min_{p \in C} \langle p, \nabla f(x_k) \rangle, \\
x_{k+1} &= (1 - c_{k+1}) x_k + c_{k+1} p_{k+1}.
\end{aligned}$$

For this method the most natural step-sizes satisfy the following recurrence ("open loop" step-sizes) $c_{k+1} = c_k - \frac{1}{2}c_k^2$, which Dunn and Harshbarger (1978) note may be replaced with $c_{k+1} = 1/(k+1)$. Another approach that more closely approximates the open-loop steps is the factorial power weighting $c_{k+1} = 2/(k+2)$ as used in Jaggi (2013) and Bach (2015).

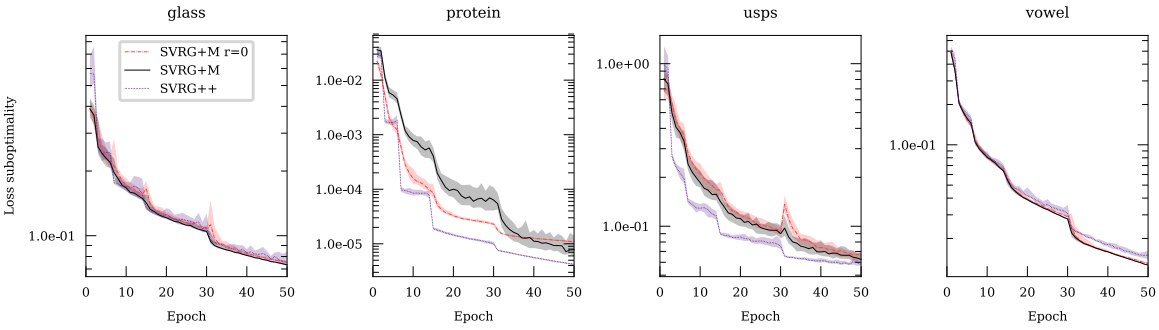

Figure 3: SVRGM training loss convergence

8. Experiments

For our experiments we compared the performance of factorial power momentum on a strongly-convex but non-smooth machine learning problem: regularized multi-class support vector machines. We consider two problems from the LIBSVM (Chang and Lin, 2011) repository: *PROTEIN* and *USPS*, and two from the UCI (Dua and Graff, 2017) repository: *GLASS* and *VOWEL*. We used batch-size 1 and the step-sizes recommended by the theory for both SGD with $r = 1$ averaging, as well as SGD with factorial power momentum as we developed in Theorem 3. We induced strong convexity by using weight decay of strength 0.001. The median as well as interquartile range bars from 40 runs are shown. Since our theory suggests $r = 3$, we tested $r = 0, 1, 3, 5$ to verify that $r = 3$ is the best choice. The results are shown in Figure 2. We see that when using factorial power momentum, using $r = 0, 1$ is worse than $r = 3$, and using $r = 5$ is no better that $r = 3$, so the results agree with our theory. The momentum method also performs a little better than SGD with post-hoc averaging, however it does appear to be substantially more variable between runs, as the interquartile range shows. We provide further experiments covering the SVRGM method in the supplementary material.

8.1. SVRGM Experiments

We compared the SVRGM method against SVRG both with the $r = 1/2$ momentum suggested by the theory as well as equal weighted momentum. We used the same test setup as for our `SGDM` experiments, except without the addition of weight decay in order to test the non-strongly convex convergence. Since the selection of step-size is less clear in the non-strongly convex case, here we used a step-size sweep on a power-of-2 grid, and we reported the results of the best step-size for each method. As shown in Figure 3, SVRGM is faster on two of the test problems and slower on two. The flat momentum variant is a little slower than $r = 1/2$ momentum, however not significantly so.

9. Conclusion

Factorial powers are a flexible and broadly applicable tool for establishing tight convergence rates as well as simplifying proofs. As we have shown, they have broad applicability both for stochastic optimization and beyond.

References

Alekh Agarwal, Martin J Wainwright, Peter L. Bartlett, and Pradeep K. Ravikumar. Information-theoretic lower bounds on the oracle complexity of convex optimization. In *Advances in Neural Information Processing Systems 22*, pages 1–9, 2009.

Zeyuan Allen Zhu and Yang Yuan. Improved SVRG for non-strongly-convex or sum-of-non-convex objectives. In *Proceedings of the 33nd International Conference on Machine Learning, ICML*, volume 48, pages 1080–1089, 2016.

Alfred Auslender and Marc Teboulle. Interior gradient and proximal methods for convex and conic optimization. *SIAM Journal on Optimization*, 2006.

Francis Bach. Duality between subgradient and conditional gradient methods. *SIAM Journal on Optimization*, 25(1):115–129, 2015.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. IMAGING SCIENCES*, 2009.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27, 2011.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

J. C. Dunn and S. Harshbarger. Conditional gradient algorithms with open loop step size rules. *Journal of Mathematical Analysis and Applications*, 1978.

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics*. Addison-Wesley, 2nd edition edition, 1994.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*. PMLR, 2013.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Proceedings of the 26th International Conference on Neural Information Processing Systems*, 2013.

Simon Lacoste-Julien, Mark Schmidt, and Francis Bach. A simpler approach to obtaining an o(1/t) convergence rate for the projected stochastic subgradient method. arXiv, 2012.

Kevin J. McGown and Harold R. Parks. The generalization of faulhaber's formula to sums of non-integral powers. *Journal of Mathematical Analysis and Applications*, 330(1):571 – 575, 2007.

Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Math. Program.*, 2009.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*. Springer, 2013.

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

H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

Othmane Sebbouh, Nidham Gazagnadou, Samy Jelassi, Francis Bach, and Robert M. Gower. Towards closing the gap between the theory and practice of svrg. *Neurips*, 2019.

Othmane Sebbouh, Robert M. Gower, and Aaron Defazio. On the convergence of the stochastic heavy ball method, 2020.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. *Proceedings of the 30th International Conference on Machine Learning*, 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, 2020.

Adrien Taylor and Francis Bach. Stochastic first-order methods: non-asymptotic and computer-aided analyses via potential functions. In *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2934–2992, Jun 2019.

Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. Technical report, MIT, 2008.