Accelerating Online Convex Optimization via Adaptive Prediction

Mehryar Mohri
Courant Institute and Google Research
New York, NY 10012
mohri@cims.nyu.edu

Scott Yang
Courant Institute
New York, NY 10012
yangs@cims.nyu.edu

Abstract

We present a powerful general framework for designing data-dependent online convex optimization algorithms, building upon and unifying recent techniques in adaptive regularization, optimistic gradient predictions, and problem-dependent randomization. We first present a series of new regret guarantees that hold at any time and under very minimal assumptions, and then show how different relaxations recover existing algorithms, both basic as well as more recent sophisticated ones. Finally, we show how combining adaptivity, optimism, and problem-dependent randomization can guide the design of algorithms that benefit from more favorable guarantees than recent state-of-the-art methods.

1 Introduction

Online convex optimization algorithms represent key tools in modern machine learning. These are flexible algorithms used for solving a variety of optimization problems in classification, regression, ranking and probabilistic inference. These algorithms typically process one sample at a time with an update per iteration that is often computationally cheap and easy to implement. As a result, they can be substantially more efficient both in time and space than standard batch learning algorithms, which often have optimization costs that are prohibitive for very large data sets.

In the standard scenario of online convex optimization (Zinkevich, 2003), at each round $t = 1, 2, \ldots$, the learner selects a point x_t out of a compact convex set $\mathcal{K} \subset \mathbb{R}^n$ and incurs loss $f_t(x_t)$, where f_t is a convex function defined over $\mathcal{K}$. The learner's objective is to find an algorithm $\mathcal{A}$ that minimizes the regret with respect to a fixed point x^*:

$$\mathrm{Reg}_T(\mathcal{A}, x^*) = \sum_{t=1}^{T} f_t(x_t) - f_t(x^*)$$

that is the difference between the learner's cumulative loss and the loss in hindsight incurred by x^*, or with respect to the loss of the best x^* in $\mathcal{K}$, $\mathrm{Reg}_T(\mathcal{A}) = \max_{x^* \in \mathcal{K}} \mathrm{Reg}_T(\mathcal{A}, x^*)$. We will assume only that the learner has access to the gradient or an element of the sub-gradient of the loss functions f_t, but that the loss functions f_t can be arbitrarily singular and flat, e.g. not necessarily strongly convex or strongly smooth. This is the most general setup of convex optimization in the full information setting. It can be applied to standard convex optimization and online learning tasks as well as to many optimization problems in machine learning such as those of SVMs, logistic regression, and ridge regression. Favorable bounds in online convex optimization can also be translated into strong learning guarantees in the standard scenario of batch supervised learning using online-to-batch conversion guarantees (Littlestone, 1989; Cesa-Bianchi et al., 2004; Mohri et al., 2012).

In the scenario of online convex optimization just presented, minimax optimal rates can be achieved by standard algorithms such as online gradient descent (Zinkevich, 2003). However, general minimax optimal rates may be too conservative. Recently, *adaptive regularization* methods have been introduced for standard descent methods to achieve tighter data-dependent regret bounds (see (Bartlett et al., 2007), (Duchi et al., 2010), (McMahan and Streeter, 2010), (McMahan, 2014), (Orabona et al., 2013)). Specifically, in the "AdaGrad" framework of Duchi et al. (2010), there exists a sequence of convex functions ψ_t such that the update $x_{t+1} = \mathrm{argmin}_{x \in \mathcal{K}} \eta g_t^\top x + B_{\psi_t}(x, x_t)$ yields re-

Appearing in Proceedings of the 19^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2016, Cadiz, Spain. JMLR: W&CP volume 41.

gret:

$$\mathrm{Reg}_T(\mathcal{A}, x) \le \sqrt{2} \max_t \|x - x_t\|_\infty \sum_{i=1}^n \sqrt{\sum_{t=1}^T |g_{t,i}|^2},$$

where $g_t \in \partial f_t(x_t)$ is an element of the subgradient of f_t at x_t, $g_{1:T,i} = \sum_{t=1}^T g_{t,i}$, and B_{ψ_t} is the Bregman divergence defined using the convex function ψ_t. This upper bound on the regret has shown to be within a factor $\sqrt{2}$ of the a posteriori optimal regret with respect to a family of quadratic regularizers:

$$\max_t \|x - x_t\|_\infty \sqrt{n \inf_{s \succcurlyeq 0, 1^\top s \le n} \sum_{t=1}^T \|g_t\|^2_{\mathrm{diag}(s)^{-1}}}.$$

The comparison family demonstrated here is the set of all non-negative diagonal matrices with bounded trace, which is both computationally efficient and the most commonly used benchmark in practice. Note, however, that this upper bound on the regret can still be very large, even if the functions f_t admit some favorable properties (e.g. $f_t \equiv f$, linear). This is because the dependence is directly on the norm of g_ts.

An alternative line of research has been investigated by a series of recent publications that have analyzed online learning in "slowly-varying" scenarios (Hazan and Kale, 2009; Chiang et al., 2012; Rakhlin and Sridharan, 2013a,b; Chiang et al., 2013). In the framework of (Rakhlin and Sridharan, 2013b), if $\mathcal{R}$ is a self-concordant function, $\|\cdot\|_{\nabla^2 \mathcal{R}(x_t)}$ is the semi-norm induced by its Hessian at the point x_t,[1] and $\tilde{g}_{t+1} = \tilde{g}_{t+1}(g_1, \ldots, g_t, x_1, \ldots, x_t)$ is a "prediction" of a time $t+1$ subgradient g_{t+1} based on information up to time t, then one can obtain regret bounds of the following form:

$$\mathrm{Reg}_T(\mathcal{A}, x) \le \sqrt{4 \max_{x \in \mathcal{K}} \mathcal{R}(x) \left[\sum_{t=1}^T \|g_t - \tilde{g}_t\|^2_{\nabla^2 \mathcal{R}(x_t), *} + 1 \right]}.$$

Here, $\|\cdot\|_{\nabla^2 \mathcal{R}(x_t), *}$ denotes the dual norm of $\|\cdot\|_{\nabla^2 \mathcal{R}(x_t)}$: for any x, $\|x\|_{\nabla^2 \mathcal{R}(x_t), *} = \sup_{\|y\|_{\nabla^2 \mathcal{R}(x_t)} \le 1} x^\top y$. This guarantee can be very favorable in the *optimistic* case where $\tilde{g}_t \approx g_t$ for all t. Nevertheless, it admits the drawback that the chosen regularization admits no guarantee of "near-optimality" with respect to a family of regularizers as with the adaptive algorithm.

The discussion above naturally compels one to ask whether it is possible to combine the two data-dependent methods into a technique that captures the desirable attributes of both while discarding the deficiencies. For instance, one would aspire to attain regret guarantees of the form:

$$\max_t \|x - x_t\|_\infty \sqrt{n \inf_{s \succcurlyeq 0, 1^\top s \le n} \sum_{t=1}^T \|g_t - \tilde{g}_t\|^2_{\mathrm{diag}(s)^{-1}}}. \quad (1)$$

This paper presents a powerful general framework for designing online convex optimization algorithms combining adaptive regularization and optimistic gradient prediction which helps address several of the issues just pointed out. Our framework builds upon and unifies recent techniques in adaptive regularization and optimistic gradient predictions, and as an example, attains the regret bound (1). In Section 2, we describe a series of *adaptive and optimistic* algorithms for which we prove strong regret guarantees, including a new *Adaptive and Optimistic Follow-the-Regularized-Leader* (AO-FTRL) algorithm (Section 2.1) and a more general version of this algorithm with composite terms (Section 2.3). These new regret guarantees hold at any time and under very minimal assumptions. We also show how different relaxations recover both basic existing algorithms as well as more recent sophisticated ones. In a specific application, we will also show how a certain choice of regularization functions will produce an optimistic regret bound that is also nearly a posteriori optimal, combining the two different desirable properties mentioned above. Lastly, in Section 3, we further analyze adaptivity and optimism in the stochastic optimization framework, constructing algorithms for random coordinate descent and stochastic empirical risk minimization that benefit from more favorable guarantees than recent state-of-the-art methods.

2 Adaptive and Optimistic Follow-the-Regularized-Leader algorithms

2.1 AO-FTRL algorithm

In view of the discussion in the previous section, we present an adaptive and optimistic version of the Follow-the-Regularized-Leader (FTRL) family of algorithms. In each round of standard FTRL, a point is chosen that is the minimizer of the average linearized loss incurred plus a regularization term. In our new version of FTRL, we will find a minimizer of not only the average loss incurred, but also a prediction of the next round's loss. In addition, we will define a dynamic time-varying sequence of regularization functions that can be used to optimize against this new loss term. Algorithm 1 shows the pseudocode of

[1] The norm induced by a symmetric positive definite (SPD) matrix A is defined for any x by $\|x\|_A = \sqrt{x^\top A x}$.

Algorithm 1 AO-FTRL

Input: regularization function $r_0 \geq 0$.
Initialize: $\tilde{g}_1 = 0$, $x_1 = \operatorname{argmin}_{x \in \mathcal{K}} r_0(x)$.
for $t = 1, \ldots, T$: **do**
  Compute $g_t \in \partial f_t(x_t)$.
  Construct regularizer $r_t \geq 0$.
  Predict $\tilde{g}_{t+1} = \tilde{g}_{t+1}(g_1, \ldots, g_t, x_1, \ldots, x_t)$.
  Update $x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{K}} g_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x)$.
end for

our Adaptive and Optimistic Follow-the-Regularized-Leader (AO-FTRL) algorithm.

Note here that we have subsumed the step-size that is common to many incremental gradient methods into the regularization function. This is because we will later tune our regularization in a problem-dependent way to yield state-of-the-art guarantees. Combining the regularization with the step-size avoids the need to handle an additional parameter.

In the rest of the paper, we say that a regularization function $r_t : \mathbb{R}^n \to \mathbb{R}$ corresponding to an algorithm is *proximal* if $\operatorname{argmin}_{x \in \mathcal{K}} r_t(x) = x_t$. The following result provides a regret guarantee for the Algorithm 1 when using such regularizers.

Theorem 1 (AO-FTRL-Prox)**.** *Let $\{r_t\}$ be a sequence of proximal non-negative functions, and let $\tilde{g}_t$ be the learner's estimate of g_t given the history of functions $f_1, \ldots, f_{t-1}$ and points $x_1, \ldots, x_{t-1}$. Assume further that the function $h_{0:t} \colon x \mapsto g_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x)$ is 1-strongly convex with respect to some norm $\|\cdot\|_{(t)}$ (i.e. $r_{0:t}$ is 1-strongly convex with respect to $\|\cdot\|_{(t)}$). Then, the following regret bound holds for AO-FTRL (Algorithm 1):*

$$\begin{aligned}&\operatorname{Reg}_T(\text{AO-FTRL}, x)\\ &= \sum_{t=1}^T f_t(x_t) - f_t(x) \leq r_{0:T}(x) + \sum_{t=1}^T \|g_t - \tilde{g}_t\|_{(t),*}^2.\end{aligned}$$

Proof. Recall that $x_{t+1} = \operatorname{argmin}_x (g_{1:t} + \tilde{g}_{t+1})^\top x + r_{0:t}(x) = \operatorname{argmin}_x h_{0:t}(x)$, and let $y_t = \operatorname{argmin}_x x^\top g_{1:t} + r_{0:t}(x)$. Then, by convexity, the following inequality holds:

$$\begin{aligned}\sum_{t=1}^T f_t(x_t) - f_t(x) &\leq \sum_{t=1}^T g_t^\top (x_t - x)\\ &= \sum_{t=1}^T (g_t - \tilde{g}_t)^\top (x_t - y_t)\\ &\quad + \tilde{g}_t^\top (x_t - y_t) + g_t^\top (y_t - x).\end{aligned}$$

Now, we first prove by induction on T that for all $x \in \mathcal{K}$ the following inequality holds:

$$\sum_{t=1}^T \tilde{g}_t^\top (x_t - y_t) + g_t^\top y_t \leq \sum_{t=1}^T g_t^\top x + r_{0:T}(x).$$

For $T = 1$, since $\tilde{g}_1 = 0$ and $r_1 \geq 0$, the inequality follows by the definition of y_1. Now, suppose the inequality holds at iteration T. Then, we can write

$$\begin{aligned}&\sum_{t=1}^{T+1} \tilde{g}_t^\top (x_t - y_t) + g_t^\top y_t\\ &= \left[\sum_{t=1}^T \tilde{g}_t^\top (x_t - y_t) + g_t^\top y_t\right]\\ &\quad + \tilde{g}_{T+1}^\top (x_{T+1} - y_{T+1}) + g_{T+1}^\top y_{T+1}\\ &\leq \left[\sum_{t=1}^T g_t^\top x_{T+1} + r_{0:T}(x_{T+1})\right]\\ &\quad + \tilde{g}_{T+1}^\top (x_{T+1} - y_{T+1}) + g_{T+1}^\top y_{T+1}\\ &\quad \text{(by the induction hypothesis for } x = x_{T+1})\\ &\leq \left[(g_{1:T} + \tilde{g}_{T+1})^\top x_{T+1} + r_{0:T+1}(x_{T+1})\right]\\ &\quad + \tilde{g}_{T+1}^\top (-y_{T+1}) + g_{T+1}^\top y_{T+1}\\ &\quad \text{(since } r_t \geq 0, \forall t)\\ &\leq \left[(g_{1:T} + \tilde{g}_{T+1})^\top y_{T+1} + r_{0:T+1}(y_{T+1})\right]\\ &\quad + \tilde{g}_{T+1}^\top (-y_{T+1}) + g_{T+1}^\top y_{T+1}\\ &\quad \text{(by definition of } x_{T+1})\\ &\leq g_{1:T+1}^\top y + r_{0:T+1}(y), \text{ for any } y.\\ &\quad \text{(by definition of } y_{T+1})\end{aligned}$$

Thus, we have that $\sum_{t=1}^T f_t(x_t) - f_t(x) \leq r_{0:T}(x) + \sum_{t=1}^T (g_t - \tilde{g}_t)^\top (x_t - y_t)$ and it suffices to bound $\sum_{t=1}^T (g_t - \tilde{g}_t)^\top (x_t - y_t)$. Notice that, by duality, one can immediately write $(g_t - \tilde{g}_t)^\top (x_t - y_t) \leq \|g_t - \tilde{g}_t\|_{(t),*} \|x_t - y_t\|_{(t)}$. To bound $\|x_t - y_t\|_{(t)}$ in terms of the gradient, recall first that since r_t is proximal and $x_t = \operatorname{argmin}_x h_{0:t-1}$,

$$\begin{aligned}x_t &= \operatorname*{argmin}_x h_{0:t-1}(x) + r_t(x),\\ y_t &= \operatorname*{argmin}_x h_{0:t-1}(x) + r_t(x) + (g_t - \tilde{g}_t)^\top x.\end{aligned}$$

The fact that $r_{0:t}(x)$ is 1-strongly convex with respect to the norm $\|\cdot\|_{(t)}$ implies that $h_{0:t-1} + r_t$ is as well. In particular, it is 1-strongly convex at the points x_t and y_t. But this then implies that the conjugate function is 1-strongly smooth on the image of the gradient, including at $\nabla(h_{0:t-1} + r_t)(x_t) = 0$ and $\nabla(h_{0:t-1} + r_t)(y_t) = -(g_t - \tilde{g}_t)$ (see Lemma 1 in the appendix or (Rockafellar, 1970) for a general reference), which means that $\|\nabla((h_{0:t-1} + r_t)^*)(-(g_t - \tilde{g}_t)) - \nabla((h_{0:t-1} + r_t)^*)(0)\|_{(t)} \leq \|g_t - \tilde{g}_t\|_{(t),*}$.

Since $\nabla((h_{0:t-1}+r_t)^*)(-(g_t-\tilde{g}_t)) = y_t$ and $\nabla((h_{0:t-1}+r_t)^*)(0) = x_t$, we have that $\|x_t - y_t\|_{(t)} \leq \|g_t - \tilde{g}_t\|_{(t),*}$. □

The regret bound just presented can be vastly superior to the adaptive methods of (Duchi et al., 2010), (McMahan and Streeter, 2010), and others. For instance, one common choice of gradient prediction is $\tilde{g}_{t+1} = g_t$, so that for slowly varying gradients (e.g. functions that change "predictably"), $g_t - \tilde{g}_t \approx 0$, but $\|g_t\|_{(t)} = \|g\|_{(t)}$. Moreover, for reasonable gradient predictions, $\|\tilde{g}_{t+1}\|_{(t)} \approx \|g_t\|_{(t)}$ generally, so that in the worst case, Algorithm 1's regret will be at most a factor of two more than standard methods. At the same time, the use of non self-concordant regularization allows one to more explicitly control the induced norm in the regret bound and consequently learn faster per update than in the methods of (Rakhlin and Sridharan, 2013a). Section 2.2.1 presents an upgraded version of online gradient descent as an example, where our choice of regularization allows our algorithm to *accelerate* as the gradient predictions become more accurate.

Note that the assumption of strong convexity of $h_{0:t}$ is not a significant constraint, as any quadratic or entropic regularizer from the standard mirror descent algorithms will satisfy this property.

Moreover, if the loss functions $\{f_t\}_{t=1}^{\infty}$ themselves are 1-strongly convex, then one can set $r_{0:t} \equiv 0$ and still get a favorable induced norm $\|\cdot\|_{(t),*}^2 = \frac{1}{t}\|\cdot\|_2^2$. If the gradients and gradient predictions are uniformly bounded, this recovers the worst-case $\log(T)$ regret bounds. At the same time, Algorithm 1 would also still retain the potentially highly favorable data-dependent and optimistic regret bound.

Liang and Steinhardt (2014) (Steinhardt and Liang, 2014) also studied adaptivity and optimism in online learning in the context of mirror descent-type algorithms. If, in the proof above, we assume their condition:

$$r^*_{0:t+1}(-\eta g_{1:t}) \leq r^*_{0:t}(-\eta(g_{1:t} - \tilde{g}_t)) - \eta x_t^\top (g_t - \tilde{g}_t),$$

then we obtain the following regret bound: $\sum_{t=1}^T f_t(x_t) - \sum_{t=1}^T f_t(x) \leq \frac{r_1^*(0)+r_{0:T+1}(x)}{\eta}$. Our algorithm, however, is generally easier to use since it holds for any sequence of regularization functions and does not require checking for that condition.

In some cases, it may be preferable to use non-proximal adaptive regularization. Since non-adaptive non-proximal FTRL corresponds to dual averaging, this scenario arises, for instance, when one wishes to use regularizers such as the negative entropy to derive algorithms from the Exponentiated Gradient (EG) family (see (Shalev-Shwartz, 2012) for background). We thus present the following theorem for this family of algorithms: Adaptive Optimistic Follow-the-Regularized-Leader - General version (AO-FTRL-Gen).

Theorem 2 (AO-FTRL-Gen). *Let $\{r_t\}$ be a sequence of non-negative functions, and let $\tilde{g}_t$ be the learner's estimate of g_t given the history of functions $f_1, \ldots, f_{t-1}$ and points $x_1, \ldots, x_{t-1}$. Assume further that the function $h_{0:t}\colon x \mapsto g_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x)$ is 1-strongly convex with respect to some norm $\|\cdot\|_{(t)}$ (i.e. $r_{0:t}$ is 1-strongly convex with respect to $\|\cdot\|_{(t)}$). Then, the following regret bound holds for AO-FTRL (Algorithm 1):*

$$\sum_{t=1}^T f_t(x_t) - f_t(x) \leq r_{0:T-1}(x) + \sum_{t=1}^T \|g_t - \tilde{g}_t\|_{(t-1),*}^2$$

Due to spatial constraints, the proof of this theorem, as well as that of all further results in the remainder of Section 2, are presented in Appendix 6.

As in the case of proximal regularization, Algorithm 1 applied to general regularizers still admits the same benefits over the standard adaptive algorithms. In particular, the above algorithm is an easy upgrade over any dual averaging algorithm.

Corollary 1. *With the following suitable choices of the parameters in Theorem 3, the following regret bounds can be recovered:*

1. *Adaptive FTRL-Prox of (McMahan, 2014) (up to a constant factor of 2): $\tilde{g} \equiv 0$.*
2. *Primal-Dual AdaGrad of (Duchi et al., 2010): $r_{0:t} = \psi_t$, $\tilde{g} \equiv 0$.*
3. *Optimistic FTRL of (Rakhlin and Sridharan, 2013a): $r_0 = \eta\mathcal{R}$ where $\eta > 0$ and $\mathcal{R}$ a self-concordant function, $r_t = \psi_t = 0, \forall t \geq 1$.*

2.2 Applications

2.2.1 Adaptive and Optimistic Gradient Descent

Corollary 2 (AO-GD). *Let $\mathcal{K} \subset \times_{i=1}^n [-R_i, R_i]$ be an n-dimensional rectangle, and denote $\Delta_{s,i} = \sqrt{\sum_{a=1}^s (g_{a,i} - \widetilde{g}_{a,i})^2}$. Set*

$$r_{0:t} = \sum_{i=1}^n \sum_{s=1}^t \frac{\Delta_{s,i} - \Delta_{s-1,i}}{2R_i}(x_i - x_{s,i})^2.$$

Then, if we use the martingale-type gradient prediction

$\tilde{g}_{t+1} = g_t$, *the following regret bound holds:*

$$Reg_T(\text{AO-GD}, x) \leq 4 \sum_{i=1}^{n} R_i \sqrt{\sum_{t=1}^{T} (g_{t,i} - g_{t-1,i})^2}.$$

Moreover, this regret bound is nearly a posteriori optimal over a family of quadratic regularizers :

$$R_i \sum_{i=1}^{n} \sqrt{\sum_{t=1}^{T} (g_{t,i} - g_{t-1,i})^2}$$

$$= \max_i R_i \sqrt{n \inf_{s \succcurlyeq 0, \langle s, 1 \rangle \leq n} \sum_{t=1}^{T} \|g_t - g_{t-1}\|^2_{diag(s)^{-1}}}.$$

Notice that the regularization function is minimized when the gradient predictions become more accurate. Thus, if we interpret our regularization as an implicit learning rate, our algorithm uses a larger learning rate and *accelerates* as our gradient predictions become more accurate. This is in stark contrast to other adaptive regularization methods, such as AdaGrad, where learning rates are inversely proportional to simply the norm of the gradient.

Moreover, since the regularization function decomposes over the coordinates, this acceleration can occur on a per-coordinate basis. If our gradient predictions are more accurate in some coordinates than others, then our algorithm will be able to adapt accordingly. Under the simple martingale prediction scheme, this means that our algorithm will be able to adapt well when only certain coordinates of the gradient are slowly-varying, even if the entire gradient is not.

As an illustrative example, suppose we are trying to predict the gradient and did so to varying degrees of accuracy per coordinate. Specifically, let $A_k = |\{i \in [1, n] | \frac{1}{2^{k+1}} \leq \sum_{s=1}^{T} (g_{s,i} - \tilde{g}_{s,i})^2 < \frac{1}{2^k}\}|$, and assume that A_k is roughly constant over k such that $\forall k, j$, $A_k \leq C A_j$ for some mild universal constant C. Then the bound in Corollary 2 can be upper bounded by $\sum_{k=0}^{\infty} A_k \sqrt{\frac{1}{2^k}} = A_1 C \frac{\sqrt{2}}{\sqrt{2}-1}$, while standard learning algorithms with predictable sequences will incur a regret of $\sqrt{n} \sqrt{\sum_{k=0}^{\infty} A_k \frac{1}{2^k}} = A_1 \sqrt{n} \sqrt{2}$, so that we improve by a $\sqrt{d}$ factor.

Moreover, since it is unlikely that the gradient predictions will ever be uniformly accurate across coordinates, some degree of the above improvement should take place for any model and problem.

Furthermore, notice that if we are in the sparse case in which AdaGrad performs well (see Duchi et al. (2010), Section 1.3), the martingale-type predictor $\tilde{g}_t = g_{t-1}$ will automatically tune the regularization of Algorithm 7 into an AdaGrad-type one.

Algorithm 2 CAO-FTRL

Input: regularization function $r_0 \geq 0$, composite functions $\{\psi_t\}_{t=1}^{\infty}$ where $\psi_t \geq 0$.
Initialize: $\tilde{g}_1 = 0$, $x_1 = \text{argmin}_{x \in \mathcal{K}} r_0(x)$.
for $t = 1, \ldots, T$: **do**
  Compute $g_t \in \partial f_t(x_t)$.
  Construct regularizer $r_t \geq 0$.
  Predict the next gradient $\tilde{g}_{t+1} = \tilde{g}_{t+1}(g_1, \ldots, g_t, x_1, \ldots, x_t)$.
  Update $x_{t+1} = \text{argmin}_{x \in \mathcal{K}} g_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x) + \psi_{1:t+1}(x)$.
end for

In terms of computation, the AO-GD update can be executed in time linear in the dimension (the same as for standard gradient descent). Moreover, since the gradient prediction is simply the last gradient received, the algorithm also does not require much more storage than the standard gradient descent algorithm. However, as we mentioned in the general case, the regret bound here can be significantly more favorable than the standard $\mathcal{O}\left(\sqrt{T \sup_{t \in [1,T} \|g_t\|_2^2 \sum_{i=1}^{n} R_i^2]}\right)$ bound of online gradient descent, or even its adaptive variants.

2.3 CAO-FTRL algorithm (Composite Adaptive Optimistic Follow-the-Regularized-Leader)

In some cases, we may wish to impose some regularization on our original optimization problem to ensure properties such as generalization (e.g. l_2-norm in SVM) or sparsity (e.g. l^1-norm in Lasso). This "composite term" can be treated directly by modifying the regularization in our FTRL update. However, if we wish for the regularization penalty to appear in the regret expression but do not wish to linearize it (which could mitigate effects such as sparsity), then some extra care needs to be taken.

We modify Algorithm 1 to obtain Algorithm 2, and we provide accompanying regret bounds for both proximal and general regularization functions. The latter is presented in Appendix 6.

In each theorem, we give a pair of regret bounds, depending on whether the learner considers the composite term as an additional part of the loss. All proofs are provided in Appendix 6.

Theorem 3 (CAO-FTRL-Prox). *Let $\{r_t\}$ be a sequence of proximal non-negative functions, such that $\text{argmin}_{x \in \mathcal{K}} r_t(x) = x_t$, and let $\tilde{g}_t$ be the learner's estimate of g_t given the history of functions $f_1, \ldots, f_{t-1}$ and points $x_1, \ldots, x_{t-1}$. Let $\{\psi_t\}_{t=1}^{\infty}$ be a sequence of non-negative convex functions, such that $\psi_1(x_1) = 0$.*

Assume further that the function $h_{0:t} : x \mapsto g_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x) + \psi_{1:t+1}(x)$ *is 1-strongly convex with respect to some norm* $\|\cdot\|_{(t)}$. *Then the following regret bounds hold for CAO-FTRL (Algorithm 2):*

$$\sum_{t=1}^{T} f_t(x_t) - f_t(x) \leq \psi_{1:T-1}(x) + r_{0:T-1}(x) + \sum_{t=1}^{T} \|g_t - \tilde{g}_t\|_{(t-1),*}^2$$

$$\sum_{t=1}^{T} [f_t(x_t) + \psi_t(x_t)] - [f_t(x) + \psi_t(x)] \leq r_{0:T}(x) + \sum_{t=1}^{T} \|g_t - \tilde{g}_t\|_{(t),*}^2 .$$

Notice that if we don't consider the composite term as part of our loss, then our regret bound resembles the form of AO-FTRL-Gen. This is in spite of the fact that we are using proximal adaptive regularization. On the other hand, if the composite term is part of our loss, then our regret bound resembles the one using AO-FTRL-Prox.

3 Adaptive Optimistic and Stochastic Follow-the-Regularized-Leader algorithms

3.1 CAOS-FTRL algorithm (Composite Adaptive Optimistic Stochastic Follow-the-Regularized-Leader)

We now generalize the scenario to that of stochastic online convex optimization, where, instead of exact subgradient elements g_t, we receive only estimates. Specifically, we assume access to a sequence of vectors of the form $\hat{g}_t$, where $\mathbb{E}[\hat{g}_t | g_1, \ldots, g_{t-1}, x_1, \ldots, x_t] = g_t$. This extension is in fact well-documented in the literature (see (Shalev-Shwartz, 2012) for a reference), and the extension of our adaptive and optimistic variant follows accordingly. For completeness, we provide a proof of the following theorem as well as its non-proximal analogue in Appendix 8.

Theorem 4 (CAOS-FTRL-Prox). *Let* $\{r_t\}$ *be a sequence of proximal non-negative functions, such that* $\operatorname{argmin}_{x \in \mathcal{K}} r_t(x) = x_t$, *and let* $\tilde{g}_t$ *be the learner's estimate of* $\hat{g}_t$ *given the history of noisy gradients* $\hat{g}_1, \ldots, \hat{g}_{t-1}$ *and points* $x_1, \ldots, x_{t-1}$. *Let* $\{\psi_t\}_{t=1}^{\infty}$ *be a sequence of non-negative convex functions, such that* $\psi_1(x_1) = 0$. *Assume further that the function* $h_{0:t}(x) = \hat{g}_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x) + \psi_{1:t+1}(x)$ *is 1-strongly convex with respect to some norm* $\|\cdot\|_{(t)}$. *Then, the update* $x_{t+1} = \operatorname{argmin}_x h_{0:t}(x)$ *of Algorithm 3 yields the following regret bounds:*

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(x_t) - f_t(x)\right] \leq \mathbb{E}\left[\psi_{1:T-1}(x) + r_{0:T-1}(x) + \sum_{t=1}^{T} \|\hat{g}_t - \tilde{g}_t\|_{(t-1),*}^2\right]$$

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(x_t) + \psi_t(x_t) - f_t(x) - \alpha_t \psi_t(x)\right] \leq \mathbb{E}\left[r_{0:T}(x) + \sum_{t=1}^{T} \|\hat{g}_t - \tilde{g}_t\|_{(t),*}^2\right] .$$

Algorithm 3 CAOS-FTRL

Input: regularization function $r_0 \geq 0$, composite functions $\{\psi_t\}_{t=1}^{\infty}$ where $\psi_t \geq 0$.
Initialize: $\tilde{g}_1 = 0$, $x_1 = \operatorname{argmin}_{x \in \mathcal{K}} r_0(x)$.
for $t = 1, \ldots, T$: **do**
  Query $\hat{g}_t$ where $\mathbb{E}[\hat{g}_t | x_1, \ldots, x_t, \hat{g}_1, \ldots, \hat{g}_{t-1}] = g_t \in \partial f_t(x_t)$.
  Construct regularizer $r_t \geq 0$.
  Predict next gradient $\tilde{g}_{t+1} = \tilde{g}_{t+1}(\hat{g}_1, \ldots, \hat{g}_t, x_1, \ldots, x_t)$.
  Update $x_{t+1} = \operatorname{argmin}_{x \in \mathcal{K}} \hat{g}_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x) + \psi_{1:t+1}(x)$.
end for

The algorithm above enjoys the same advantages over its non-adaptive or non-optimistic predecessors. Moreover, the choice of the adaptive regularizers $\{r_t\}_{t=1}^{\infty}$ and gradient predictions $\{\tilde{g}\}_{t=1}^{\infty}$ now also depend on the randomness of the gradients received. While masked in the above regret bounds, this interplay will come up explicitly in the following two examples, where we, as the learner, impose randomness into the problem.

3.2 Applications

3.2.1 Randomized Coordinate Descent with Adaptive Probabilities

Randomized coordinate descent is a method that is often used for very large-scale problems where it is impossible to compute and/or store entire gradients at each step. It is also effective for directly enforcing sparsity in a solution since the support of the final point x_t cannot be larger than the number of updates introduced.

The standard randomized coordinate descent update is to choose a coordinate uniformly at random (see e.g. (Shalev-Shwartz and Tewari, 2011)). Nesterov (2012) (Nesterov, 2012) analyzed random coordinate descent

in the context of loss functions with higher regularity and showed that one can attain better bounds by using non-uniform probabilities.

In the randomized coordinate descent framework, at each round t we specify a distribution p_t over the n coordinates and pick a coordinate $i_t \in \{1, \ldots, n\}$ randomly according to this distribution. From here, we then construct an unbiased estimate of an element of the subgradient: $\hat{g}_t = \frac{(g_t^\top e_{i_t}) e_{i_t}}{p_{t,i_t}}$. This technique is common in the online learning literature, particularly in the context of the multi-armed bandit problem (see e.g. (Cesa-Bianchi and Lugosi, 2006) for more information).

The following theorem can be derived by applying Theorem 4 to the gradient estimates just constructed. We provide a proof in Appendix 9.

Theorem 5 (CAO-RCD). *Assume $\mathcal{K} \subset \times_{i=1}^n [-R_i, R_i]$. Let i_t be a random variable sampled according to the distribution p_t, and let*

$$\hat{g}_t = \frac{(g_t^\top e_{i_t}) e_{i_t}}{p_{t,i_t}}, \quad \hat{\tilde{g}}_t = \frac{(\tilde{g}_t^\top e_{i_t}) e_{i_t}}{p_{t,i_t}},$$

be the estimated gradient and estimated gradient prediction. Denote $\Delta_{s,i} = \sqrt{\sum_{a=1}^s (\hat{g}_{a,i} - \hat{\tilde{g}}_{a,i})^2}$, and let

$$r_{0:t} = \sum_{i=1}^n \sum_{s=1}^t \frac{\Delta_{s,i} - \Delta_{s-1,i}}{2R_i} (x_i - x_{s,i})^2$$

be the adaptive regularization. Then, the regret of the algorithm can be bounded by:

$$\mathbb{E}\left[\sum_{t=1}^T f_t(x_t) + \alpha_t \psi(x_t) - f_t(x) - \alpha_t \psi(x)\right] \leq 4 \sum_{i=1}^n R_i \sqrt{\sum_{t=1}^T \mathbb{E}\left[\frac{(g_{t,i} - \tilde{g}_{t,i})^2}{p_{t,i}}\right]}$$

In general, we do not have access to an element of the subgradient g_t before we sample according to p_t. However, if we assume that we have some per-coordinate upper bound on an element of the subgradient uniform in time, i.e. $|g_{t,j}| \leq L_j \ \forall t \in \{1, \ldots, T\}, j \in \{1, \ldots, n\}$, then we can use the fact that $|g_{t,j} - \tilde{g}_{t,j}| \leq \max\{L_j - \tilde{g}_{t,j}, \tilde{g}_{t,j}\}$ to motivate setting $\tilde{g}_{t,j} := \frac{L_j}{2}$ and $p_{t,j} = \frac{(R_j L_j)^{2/3}}{\sum_{k=1}^n (R_k L_k)^{2/3}}$ (by computing the optimal distribution). This yields the following regret bound.

Corollary 3 (CAO-RCD-Lipschitz). *Assume that at any time t the following per-coordinate Lipschitz bounds hold on the loss function: $|g_{t,i}| \leq L_i, \quad \forall i \in \{1, \ldots, n\}$. Set $p_{t,i} = \frac{(R_i L_i)^{2/3}}{\sum_{j=1}^n (R_j L_j)^{2/3}}$ as the probability distribution at time t, and set $\tilde{g}_{t,i} = \frac{L_i}{2}$. Then, the regret of the algorithm can be bounded as follows:*

$$\mathbb{E}\left[\sum_{t=1}^T f_t(x_t) + \alpha_t \psi(x_t) - f_t(x) - \alpha_t \psi(x)\right] \leq 2\sqrt{T} \left(\sum_{i=1}^n (R_i L_i)^{2/3}\right)^{3/2}.$$

An application of Hölder's inequality will reveal that this bound is strictly smaller than the $2RL\sqrt{nT}$ bound one would obtain from randomized coordinate descent using the uniform distribution. As a motivating example, if the problem were poorly conditioned and there was only a single dominant coordinate i^*, then the bound of Corollary 3 would be $2\sqrt{T} R_{i^*} L_{i^*}$ while the standard bound would have an extra $\sqrt{n}$ factor.

Moreover, the algorithm above still entertains the intermediate data-dependent bound of Theorem 5.

Notice the similarity between the sampling distribution generated here with the one suggested by (Nesterov, 2012). However, Nesterov assumed higher regularity in his algorithm (i.e. $f_t \in C^{1,1}$) and generated his probabilities from there. In our setting, we only need $f_t \in C^{0,1}$. It should be noted that (Afkanpour et al., 2013) also proposed an importance-sampling based approach to random coordinate descent for the specific setting of multiple kernel learning. In their setting, they propose updating the sampling distribution at each point in time instead of using uniform-in-time Lipschitz constants, which comes with a natural computational tradeoff. Moreover, the introduction of adaptive per-coordinate learning rates in our algorithm allows for tighter regret bounds in terms of the Lipschitz constants.

We can also derive the analogous mini-batch update:

Corollary 4 (CAO-RCD-Lipschitz-Mini-Batch). *Assume $\mathcal{K} \subset \times_{i=1}^n [-R_i, R_i]$. Let $\cup_{j=1}^k \{\Pi_j\} = \{1, \ldots, n\}$ be a partition of the coordinates, and let $e_{\Pi_j} = \sum_{i \in \Pi_j} e_i$. Assume we had the following Lipschitz condition on the partition: $\|g_t^\top e_{\Pi_j}\| \leq L_j \ \forall j \in \{1, \ldots, k\}$.*

Define $S_i = \sum_{j \in \Pi_i} R_j$. Set $p_{t,i} = \frac{(S_i L_i)^{2/3}}{\sum_{j=1}^k (S_j L_j)^{2/3}}$ as the probability distribution at time t, and set $\tilde{g}_{t,i} = \frac{L_i}{2}$.

Then the regret of the resulting algorithm is bounded by:

$$\mathbb{E}\left[\sum_{t=1}^T f_t(x_t) + \alpha_t \psi(x_t) - f_t(x) - \alpha_t \psi(x)\right] \leq 2\sqrt{T} \left(\sum_{i=1}^k (S_i L_i)^{2/3}\right)^{3/2}$$

While the expression is similar to the non-mini-batch version, the L_i and R_i terms now have different meaning. Specifically, L_i is a bound on the 2-norm of the components of the gradient in each batch, and R_i is the 1-norm of the corresponding sides of the hypercube.

3.2.2 Stochastic Regularized Empirical Risk Minimization

Many learning algorithms can be viewed as instances of regularized empirical risk minimization (e.g. SVM, Logistic Regression, Lasso), where the goal is to minimize an objective function of the following form:

$$H(x) = \sum_{j=1}^{m} f_j(x) + \alpha\psi(x).$$

If we denote the first term by $F(x) = \sum_{j=1}^{m} f_j(x)$, then we can view this objective in our CAOS-FTRL framework, where $f_t \equiv F$ and $\psi_t \equiv \alpha\psi$. In the same spirit as for non-uniform random coordinate descent, we can estimate the gradient of H at x_t by sampling according to some distribution p_t and use importance weighting to generate an unbiased estimate: If $g_t \in \partial F(x_t)$ and $g_t^j \in \partial f_j(x_t)$, then

$$g_t = \sum_{j=1}^{m} g_t^j \approx \frac{g_t^{j_t}}{p_{t,j_t}}.$$

This motivates the design of an algorithm similar to the one derived for randomized coordinate descent. Here we elect to use as gradient prediction the last gradient of the current function being sampled f_j. However, we may run into the problem of never having seen a function before. A logical modification would be to separate optimization into epochs and do a full batch update over all functions f_j at the start of each epoch. This is similar to the technique used in the Stochastic Variance Reduced Gradient (SVRG) algorithm of Johnson and Zhang (2013). However, we do not assume extra function regularity as they do in their paper, so the bounds are not comparable. The algorithm is presented in Algorithm 4 and comes with the following guarantee:

Corollary 5. *Assume $\mathcal{K} \subset \times_{i=1}^{n}[-R_i, R_i]$. Denote $\Delta_{s,i} = \sqrt{\sum_{a=1}^{s}(\hat{g}_{a,i} - \tilde{g}_{a,i})^2}$, and let $r_{0:t} = \sum_{i=1}^{n}\sum_{s=1}^{t}\frac{\Delta_{s,i}-\Delta_{s-1,i}}{2R_i}(x_i - x_{s,i})^2$ be the adaptive regularization.*

Then the regret of Algorithm 4 is bounded by:

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(x_t) + \alpha\psi(x_t) - f_t(x) - \alpha\psi(x)\right]$$
$$\leq \sum_{i=1}^{n} 4R_i \sqrt{\sum_{s=1}^{k}\sum_{t=(s-1)(T/k)+1}^{(s-1)(T/k)+T/k}\sum_{j=1}^{m}\frac{\left|g_{t,i}^j - \bar{g}_{s,i}^j\right|^2}{p_{t,j}}}$$

Algorithm 4 CAOS-Reg-ERM-Epoch

Input: scaling constant $\alpha > 0$, composite term ψ, $r_0 = 0$.
Initialize: initial point $x_1 \in \mathcal{K}$, distribution p_1.
Sample j_1 according to p_1, and set $t = 1$.
for $s = 1, \ldots, k$: **do**
  Compute $\bar{g}_s^j = \nabla f_j(x_1)\ \forall j \in \{1, \ldots, m\}$.
  for $a = 1, \ldots, T/k$: **do**
    If $T \mod k = 0$, compute $g^j = \nabla f_j(x_t)\ \forall j$.
    Set $\hat{g}_t = \frac{g_t^{j_t}}{p_{t,j_t}}$, and construct $r_t \geq 0$.
    Sample $j_{t+1} \sim p_{t+1}$ and set $\tilde{g}_{t+1} = \frac{\bar{g}_s^{j_t}}{p_{t,j_t}}$.
    Update $x_{t+1} = \operatorname{argmin}_{x\in\mathcal{K}} \hat{g}_{1:t}^\top x + \tilde{g}_{t+1}^\top x + r_{0:t}(x) + (t+1)\alpha\psi(x)$ and $t = t + 1$.
  end for
end for

Moreover, if $\|\nabla f_j\|_\infty \leq L_j\ \forall j$, then setting $p_{t,j} = \frac{L_i}{\sum_{j=1}^{m} L_j}$ yields a worst-case bound of: $8\sum_{i=1}^{n} R_i\sqrt{T\left(\sum_{j=1}^{m} L_j\right)^2}$.

We also include a mini-batch version of this algorithm in Appendix 10, which can be useful due to the variance reduction of the gradient prediction.

4 Conclusion

We presented a general framework for developing efficient adaptive and optimistic algorithms for online convex optimization. Building upon recent advances in adaptive regularization and predictable online learning, we improved upon each method. We demonstrated the power of this approach by deriving algorithms with better guarantees than those commonly used in practice. In addition, we also extended adaptive and optimistic online learning to the randomized setting. Here, we highlighted an additional source of problem-dependent adaptivity (that of prescribing the sampling distribution), and we showed how one can perform better than traditional naive uniform sampling.

5 Acknowledgements

We thank the reviewers for their comments, many of which were very insightful. This work was partly funded by the NSF awards IIS-1117591 and CCF-1535987 and was also supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE 1342536.

References

Afkanpour, A., A. György, C. Szepesvári, and M. Bowling (2013). A randomized mirror descent algorithm for large scale multiple kernel learning. In *ICML*, JMLR Proceedings, pp. 374–382.

Bartlett, P. L., E. Hazan, and A. Rakhlin (2007). Adaptive online gradient descent. In *NIPS*, pp. 65–72.

Cesa-Bianchi, N., A. Conconi, and C. Gentile (2004). On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory 50*(9), 2050–2057.

Cesa-Bianchi, N. and G. Lugosi (2006). *Prediction, Learning, and Games*. New York, NY, USA: Cambridge University Press.

Chiang, C.-K., C.-J. Lee, and C.-J. Lu (2013). Beating bandits in gradually evolving worlds. In *COLT*, pp. 210–227.

Chiang, C.-K., T. Yang, C.-J. Lee, M. Mahdavi, C.-J. Lu, R. Jin, and S. Zhu (2012). Online optimization with gradual variations. In *COLT*, pp. 6.1–6.20.

Duchi, J. C., E. Hazan, and Y. Singer (2010). Adaptive subgradient methods for online learning and stochastic optimization. In *COLT*, pp. 257–269.

Hazan, E. and S. Kale (2009). Better algorithms for benign bandits. In *SODA*, pp. 38–47.

Johnson, R. and T. Zhang (2013). Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, pp. 315–323.

Littlestone, N. (1989). From on-line to batch learning. In *COLT*, pp. 269–284.

McMahan, H. B. (2014). Analysis techniques for adaptive online learning. *CoRR*.

McMahan, H. B. and M. J. Streeter (2010). Adaptive bound optimization for online convex optimization. In *COLT*, pp. 244–256.

Mohri, M., A. Rostamizadeh, and A. Talwalkar (2012). *Foundations of Machine Learning*. The MIT Press.

Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 341–362.

Orabona, F., K. Crammer, and N. Cesa-Bianchi (2013). A generalized online mirror descent with applications to classification and regression. *CoRR*.

Rakhlin, A. and K. Sridharan (2013a). Online learning with predictable sequences. In *COLT*, pp. 993–1019.

Rakhlin, A. and K. Sridharan (2013b). Optimization, learning, and games with predictable sequences. In *NIPS*, pp. 3066–3074.

Rockafellar, R. T. (1970). *Convex analysis*. Princeton University Press.

Shalev-Shwartz, S. (2012). Online learning and online convex optimization. *Found. Trends Mach. Learn.*, 107–194.

Shalev-Shwartz, S. and A. Tewari (2011). Stochastic methods for l1-regularized loss minimization. *Journal of Machine Learning Research*, 1865–1892.

Steinhardt, J. and P. Liang (2014). Adaptivity and optimism: An improved exponentiated gradient algorithm. In *ICML*, pp. 1593–1601.

Zhao, P. and T. Zhang (2014). Stochastic optimization with importance sampling. *CoRR*.

Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, pp. 928–936.