Adaptive Subgradient Methods for Online Learning and Stochastic Optimization

John Duchi[*]
University of California, Berkeley
jduchi@cs.berkeley.edu

Elad Hazan
IBM Almaden Research Center
ehazan@cs.princeton.edu

Yoram Singer
Google Research
singer@google.com

Abstract

We present a new family of subgradient methods that dynamically incorporate knowledge of the geometry of the data observed in earlier iterations to perform more informative gradient-based learning. The adaptation, in essence, allows us to find needles in haystacks in the form of very predictive yet rarely observed features. Our paradigm stems from recent advances in online learning which employ proximal functions to control the gradient steps of the algorithm. We describe and analyze an apparatus for adaptively modifying the proximal function, which significantly simplifies the task of setting a learning rate and results in regret guarantees that are provably as good as the best proximal function that can be chosen in hindsight. We corroborate our theoretical results with experiments on a text classification task, showing substantial improvements for classification with sparse datasets.

1 Introduction

In many applications of online and stochastic learning, the input instances are of very high dimension, yet within any particular instance only a few features are non-zero. It is often the case, however, that the infrequently occurring features are highly informative and discriminative. The informativeness of rare features has led practitioners to craft domain-specific feature weightings, such as TF-IDF (Salton and Buckley, 1988), which pre-emphasize infrequently occurring features. We use this old idea as a motivation for applying modern learning-theoretic techniques to the problem of online and stochastic learning, focusing specifically on (sub)gradient methods.

Standard stochastic subgradient methods largely follow a predetermined procedural scheme that is oblivious to the characteristics of the data being observed. In contrast, our algorithms dynamically incorporate knowledge of the geometry of the data from earlier iterations to perform more informative gradient-based learning. Informally, our procedures associate frequently occurring features with low learning rates and infrequent features high learning rates. This construction prompts the learner to "take notice" each time an infrequent feature is observed. Thus, the adaptation facilitates identification and adaptation of highly predictive but comparatively rare features.

1.1 The Adaptive Gradient Algorithm

For simplicity, consider the basic online convex optimization setting. The algorithm iteratively makes a prediction $x_t \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ is a closed convex set, and then receives a convex loss function f_t. Define the regret with respect to the (optimal) predictor $x^* \in \mathcal{X}$ as

$$R(T) \triangleq \sum_{t=1}^{T} \left[f_t(x_t) - f_t(x^*) \right] \ .$$

To achieve low regret, standard subgradient algorithms move the predictor x_t in the opposite direction of the subgradient $g_t \in \partial f_t(x_t)$ of the loss via the projected gradient update (e.g. Zinkevich, 2003)

$$x_{t+1} = \Pi_{\mathcal{X}} \left(x_t - \eta g_t \right) \ .$$

[*]Eligible for best student paper award

Our algorithm, called ADAGRAD, makes a second-order correction to the predictor using the previous loss functions. Denote the projection of a point y onto $\mathcal{X}$ by $\Pi_{\mathcal{X}}^{A}(y) = \operatorname{argmin}_{x \in \mathcal{X}} \|x - y\|_A$ (where $\|x\|_A = \sqrt{\langle x, Ax \rangle}$). In this notation, our adaptation of gradient descent employs the update

$$x_{t+1} = \Pi_{\mathcal{X}}^{G_t^{1/2}} \left(x_t - \eta G_t^{-1/2} g_t \right) , \tag{1}$$

where the matrix $G_t = \sum_{\tau=1}^{t} g_\tau g_\tau^\top$ is the outer product of all previous subgradients. The above algorithm may be computationally impractical in high dimensions since it requires computation of the matrix square root of G_t, the outer product matrix. We therefore also analyze a version in which we use $\operatorname{diag}(G_t)$, the diagonal of the outer product matrix, instead of G_t:

$$x_{t+1} = \Pi_{\mathcal{X}}^{\operatorname{diag}(G_t)^{1/2}} \left(x_t - \eta \operatorname{diag}(G_t)^{-1/2} g_t \right) . \tag{2}$$

This latter update rule can be computed in linear time. Moreover, as we discuss later, when the vectors g_t are sparse the update can often be performed in time proportional to the support of the gradient.

Let us compare the regret bounds attained by both variants of gradient descent. Let the diameter of $\mathcal{X}$ be bounded, so $\sup_{x,y \in \mathcal{X}} \|x - y\|_2 \leq D_2$. Then Zinkevich's analysis of online gradient descent—with the optimal choice in *hindsight* for the stepsize η—achieves regret

$$R(T) \leq \sqrt{2} D_2 \sqrt{\sum_{t=1}^{T} \|g_t\|_2^2} . \tag{3}$$

When $\mathcal{X}$ is bounded via $\sup_{x,y \in \mathcal{X}} \|x - y\|_\infty \leq D_\infty$, the following corollary is a consequence of our main Theorem 5.

Corollary 1 *Let the sequence $\{x_t\} \subset \mathbb{R}^d$ be generated by the update in Eq. (6) and let $\max_t \|x^* - x_t\|_\infty \leq D_\infty$. Then with stepsize $\eta = D_\infty / \sqrt{2}$, for any x^*,*

$$R(T) \leq \sqrt{2d} D_\infty \sqrt{\inf_{s \succeq 0, \langle 1, s \rangle \leq d} \sum_{t=1}^{T} \|g_t\|^2_{\operatorname{diag}(s)^{-1}}} = \sqrt{2} D_\infty \sum_{i=1}^{d} \|g_{1:T,i}\|_2 .$$

The important parts of the bound are the infimum under the root, which allows us to perform better than using the identity matrix, and the fact that the stepsize is easy to set a priori. For example, if $\mathcal{X} = \{x : \|x\|_\infty \leq 1\}$, then $D_2 = 2\sqrt{d}$ while $D_\infty = 2$. In the case of learning a dense predictor over a box, the bound in Corollary 1 is thus better than Eq. (3) as the identity matrix belongs to the set over which we take the infimum.

1.2 Improvement and Motivating Examples

In Section 6, we give empirical evidence in favor of adaptive algorithms. Here we give a few theoretical examples that show that for sparse data—input sequences where g_t has low cardinality—the adaptive methods are likely to perform better than non-adaptive methods. In all the cases we consider in this section we use the hinge loss, $f_t(x) = [1 - y_t \langle z_t, x \rangle]_+$, where y_t is the label of example t and $z_t \in \mathbb{R}^d$ is a data vector.

To begin, consider the following example of sparse random data. Assume that at each round t, feature i appears with probability $p_i = \min\{1, ci^{-\alpha}\}$ for some $\alpha \geq 2$ and a constant c. Suppose also that with probability 1, at least one feature appears, for instance by setting $p = 1$. Taking the expectation of the bound in Corollary 1, we have

$$\mathbb{E} \sum_{i=1}^{d} \|g_{1:T,i}\|_2 = \sum_{i=1}^{d} \mathbb{E} \sqrt{|\{t : |g_{t,i}| = 1\}|} \leq \sum_{i=1}^{d} \sqrt{\mathbb{E} |\{t : |g_{t,i}| = 1\}|} = \sum_{i=1}^{d} \sqrt{p_i T}$$

where to obtain the inequality above we used Jensen's inequality. Now, notice that for the rightmost sum, we have $c \sum_{i=1}^{d} i^{-\alpha/2} = O(\log d)$ since $\alpha \geq 2$. If the domain is a hypercube, $\mathcal{X} = \{x : \|x\|_\infty \leq 1\}$, then $D_\infty = 2$. Thus, the regret bound of ADAGRAD is $R(T) = O(\log d \sqrt{T})$. In contrast, the standard regret bound from Eq. (3) has $D_2 = 2\sqrt{d}$, and we know that $\|g_t\|_2^2 \geq 1$, yielding a regret bound $R(T) = O(\sqrt{dT})$.[1] Thus, ADAGRAD's regret guarantee is exponentially smaller than the non-adaptive regret bound as a function of dimension for this sparse data setting.

Next we give two concrete examples for which the adaptive methods learn a perfect predictor after d iterations, while standard online gradient descent (Zinkevich, 2003) suffers much higher loss. We assume the domain $\mathcal{X}$ is compact and thus for online gradient descent we set $\eta_t = 1/\sqrt{t}$, which gives $O(\sqrt{T})$ regret.

[1] For $\alpha \in (1, 2)$, ADAGRAD has regret $R(T) = O(d^{1-\alpha/2}\sqrt{T}) = o(\sqrt{dT})$.

Diagonal Adaptation In this first example, we consider the diagonal version of our proposed update in Eq. (2) with $\mathcal{X} = \{x : \|x\|_\infty \leq 1\}$. Evidently, this choice results in the update $x_{t+1} = x_t - \eta \operatorname{diag}(G_t)^{-1/2} g_t$ followed by projection onto $\mathcal{X}$. Let e_i denote the ith unit basis vector, and assume that for each t, $z_t = \pm e_i$ for some i. Also let $y_t = \operatorname{sign}(\langle \mathbb{1}, z_t \rangle)$ so that there exists a perfect classifier $x^* = 1 \in \mathcal{X}$. We initialize x_1 to be the zero vector. On rounds $t = 1, \ldots, d$, we set $z_t = \pm e_t$, selecting the sign at random. It is clear that both diagonal adaptive descent and online gradient descent suffer a unit loss on each of the first d examples. However, the updates to parameter x_i on iteration i differ and amount to

$$x_{t+1} = x_t + e_t \ \ (\textsc{AdaGrad}) \qquad x_{t+1} = x_t + \frac{1}{\sqrt{t}} e_t \ \ (\text{Gradient Descent}) \ .$$

After the first d rounds, the adaptive predictor has $x_{d+1} = x_{d+\tau} = \mathbb{1}$ for all $\tau \geq 1$ and suffers no further losses. The magnitude of the majority of the coordinates for gradient descent, though, is bounded by $\sum_{i=1}^{t} \frac{1}{\sqrt{d/2+id}} \leq \frac{2\sqrt{t}}{\sqrt{d}}$ after td iterations. Hence, for $\Omega(\sqrt{d})$ iterations, the loss suffered per coordinate is bounded from zero, for a total loss of $\Omega(d\sqrt{d})$ (compared with $O(d)$ for AdaGrad). With larger stepsizes $\eta/\sqrt{t}$, gradient descent may suffer lower loss; however, an adversary can play $z_t = e_1$ indefinitely, forcing online gradient descent to suffer $\Omega(d^2)$ loss while AdaGrad suffers constant regret per dimension.

Full Matrix Adaptation The above construction applies to the full matrix algorithm of Eq. (1) as well, but in more general scenarios, as per the following example. When using full matrix proximal functions we set $\mathcal{X} = \{x : \|x\|_2 \leq \sqrt{d}\}$. Let $V = [v_1 \ \ldots \ v_d] \in \mathbb{R}^{d \times d}$ be an orthonormal matrix. Instead of z_t cycling through the unit vectors, we have z_t cycle through the v_i so that $z_t = \pm v_{(t \mod d)+1}$. We let the label $y_t = \operatorname{sign}(\langle \mathbb{1}, V^\top z_t \rangle) = \operatorname{sign}(\sum_{i=1}^{d} \langle v_i, z_t \rangle)$. We provide an elaborated explanation in the full version of this paper (Duchi et al., 2010a). Intuitively, AdaGrad needs to observe each orthonormal vector v_i only once while stochastic gradient descent's loss is again $\Omega(d\sqrt{d})$.

1.3 Framework and Outline of Results

Before describing our results formally, let us establish notation. Vectors and scalars are lower case italic letters, such as $x \in \mathcal{X}$. We denote a sequence of vectors by subscripts, i.e. $x_t, x_{t+1}, \ldots$, and entries of each vector by an additional subscript, e.g. $x_{t,j}$. The subdifferential set of a function f evaluated at x is denoted $\partial f(x)$, and a particular vector in the subdifferential set is denoted by $f'(x) \in \partial f(x)$ or $g_t \in \partial f_t(x_t)$. We use $\langle x, y \rangle$ to denote the inner product between x and y. The Bregman divergence associated with a strongly convex and differentiable function ψ is

$$B_\psi(x, y) = \psi(x) - \psi(y) - \langle \nabla \psi(y), x - y \rangle \ .$$

For a matrix $A \in \mathbb{R}^{d \times d}$, $\operatorname{diag}(A) \in \mathbb{R}^d$ denotes its diagonal, while for a vector $s \in \mathbb{R}^d$, $\operatorname{diag}(s)$ denotes the diagonal matrix with s as its diagonal. We also make frequent use of the following two matrices. Let $g_{1:t} = [g_1 \ \cdots \ g_t]$ denote the matrix obtained by concatenating the subgradient sequence. We denote the ith row of this matrix, which amounts to the concatenation of the ith component of each subgradient we observe, by $g_{1:t,i}$. Lastly, we define the outer product matrix $G_t = \sum_{\tau=1}^{t} g_\tau g_\tau^\top$.

We describe and analyze several different online learning algorithms and their stochastic convex optimization counterparts. Formally, we consider online learning with a sequence of composite functions ϕ_t. Each function is of the form $\phi_t(x) = f_t(x) + \varphi(x)$ where f_t and φ are (closed) convex functions. In the learning settings we study, f_t is either an instantaneous loss or a stochastic estimate of the objective function. The function φ serves as a fixed regularization function and is typically used to control the complexity of x. At each round the algorithm makes a prediction $x_t \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ is a closed convex set, and then receives the function f_t. We define the regret with respect to the (optimal) predictor $x^* \in \mathcal{X}$ as

$$R_\phi(T) \triangleq \sum_{t=1}^{T} \left[\phi_t(x_t) - \phi_t(x^*)\right] = \sum_{t=1}^{T} \left[f_t(x_t) + \varphi(x_t) - f_t(x^*) - \varphi(x^*)\right] \ . \tag{4}$$

Our analysis applies to multiple methods for minimizing the regret defined in Eq. (4). The first is Nesterov's primal-dual subgradient method (Nesterov, 2009), and in particular Xiao's 2009 extension, regularized dual averaging (RDA) (Xiao, 2009), and the follow-the-regularized-leader (FTRL) family of algorithms (e.g. Kalai and Vempala, 2003; Hazan et al., 2006). In the primal-dual subgradient method the algorithm makes a prediction x_t on round t using the average gradient $\bar{g}_t = \frac{1}{t} \sum_{\tau=1}^{t} g_\tau$. The update encompasses a trade-off between a gradient-dependent linear term, the regularizer φ, and a strongly-convex term ψ_t for well-conditioned predictions. Here ψ_t is the *proximal* term. The update amounts to solving the problem

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ \eta \langle \bar{g}_t, x \rangle + \eta \varphi(x) + \frac{1}{t} \psi_t(x) \right\} \ , \tag{5}$$

where η is a step-size. The second method also has many names, such as proximal gradient, forward-backward splitting, and composite mirror descent (Tseng, 2008; Duchi and Singer, 2009; Duchi et al., 2010b). We use the term composite mirror descent. The composite mirror descent method employs a more immediate trade-off between the current gradient g_t, φ, and staying close to x_t using the proximal function ψ,

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ \eta \left\langle g_t, x \right\rangle + \eta \varphi(x) + B_{\psi_t}(x, x_t) \right\} . \tag{6}$$

Our work focuses on temporal adaptation of the proximal function in a data driven way, while previous work simply sets $\psi_t \equiv \psi$, $\psi_t(\cdot) = \sqrt{t}\psi(\cdot)$, or $\psi_t(\cdot) = t\psi(\cdot)$ for some fixed ψ.

We provide formal analyses equally applicable to the above two updates and show how to automatically choose the function ψ_t so as to achieve asymptotically small regret. We describe and analyze two algorithms. Both algorithms use squared Mahalanobis norms as their proximal functions, setting $\psi_t(x) = \frac{1}{2} \left\langle x, H_t x \right\rangle$ for a symmetric matrix $H_t \succeq 0$. The first uses diagonal matrices while the second constructs full dimensional matrices. Concretely, we set

$$H_t = \operatorname{diag}(G_t)^{1/2} \ \text{(Diagonal)} \quad \text{and} \quad H_t = G_t^{1/2} \ \text{(Full)} . \tag{7}$$

Plugging the appropriate matrix from the above equation into ψ_t in Eq. (5) or Eq. (6) gives rise to our AdaGrad family of algorithms. Informally, we obtain algorithms similar to second-order gradient descent by constructing approximations to the Hessian of the functions f_t. These approximations are conservative since we rely on the root of the gradient matrices.

We now outline our results, deferring formal statements of the theorems to later sections. Recall the definitions of $g_{1:t}$ as the matrix of concatenated subgradients and G_t as the outer product matrix in the prequel. When the proximal function $\psi_t(x) = \left\langle x, \operatorname{diag}(G_t)^{1/2} x \right\rangle$, the AdaGrad algorithm has bounds attainable in time at most linear in the dimension d of the problem of

$$R_\phi(T) = O\Big(\left\| x^* \right\|_\infty \sum_{i=1}^{d} \left\| g_{1:T,i} \right\|_2 \Big) \quad \text{and} \quad R_\phi(T) = O\Big(\max_{t \leq T} \left\| x_t - x^* \right\|_\infty \sum_{i=1}^{d} \left\| g_{1:T,i} \right\|_2 \Big).$$

We also show that

$$\sum_{i=1}^{d} \left\| g_{1:T,i} \right\|_2 = d^{1/2} \sqrt{\inf_s \left\{ \sum_{t=1}^{T} \left\langle g_t, \operatorname{diag}(s)^{-1} g_t \right\rangle \ : \ s \succeq 0, \left\langle \mathbb{1}, s \right\rangle \leq d \right\}} .$$

The AdaGrad algorithm with full matrix divergences entertains bounds of the form

$$R_\phi(T) = O\Big(\left\| x^* \right\|_2 \operatorname{tr}(G_T^{1/2}) \Big) \quad \text{and} \quad R_\phi(T) = O\Big(\max_{t \leq T} \left\| x_t - x^* \right\|_2 \operatorname{tr}(G_T^{1/2}) \Big).$$

Similar to the diagonal proximal function case, we further show that

$$\operatorname{tr}\left(G_T^{1/2} \right) = d^{1/2} \sqrt{\inf_S \left\{ \sum_{t=1}^{T} \left\langle g_t, S^{-1} g_t \right\rangle \ : \ S \succeq 0, \operatorname{tr}(S) \leq d \right\}} .$$

We formally state the above regret bounds in Theorems 5 and 8, respectively, and we give further discussion in their corollaries. Essentially, the theorems give oracle inequalities for online optimization. Though the specific sequence of gradients g_t received by the algorithm changes when there is adaptation, the inequalities say that our regret bounds are as good as the best quadratic proximal function in hindsight.

1.4 Related Work

The idea of adaptation in first order (gradient) methods is by no means new and can be traced back at least to the 1970s. There, we find Shor's work on space dilation methods (1972) as well as variable metric methods, such as the BFGS family of algorithms (e.g. Fletcher, 1970). This older work usually assumes that the function to be minimized is differentiable and, to our knowledge, did not consider stochastic, online, or composite optimization. More recently, Bordes et al. (2009) proposed carefully designed Quasi-Newton stochastic gradient descent, which is similar in spirit to our methods. However, their convergence results assume a smooth objective function whose Hessian is strictly positive definite and bounded away from 0. Our results are applicable in more general settings. In the online learning literature, there are results on adaptively choosing a learning rate η_t based on data seen so far (Auer et al., 2002; Bartlett et al., 2007). We, in contrast, actively adapt the proximal function ψ itself.

The framework that is most related to ours is probably confidence weighted learning, whose most recent success is the adaptive regularization of weights algorithm (AROW) of Crammer et al. (2009). Crammer et al.

give a mistake-bound analysis for online binary classification, which is similar in spirit to the second-order Perceptron (Cesa-Bianchi et al., 2005). AROW maintains a mean prediction vector $\mu_t \in \mathbb{R}^d$ and a covariance matrix $\Sigma_t \in \mathbb{R}^{d\times d}$ over μ_t as well. At every step of the algorithm, the learner receives a pair (z_t, y_t) where $z_t \in \mathbb{R}^d$ is the tth example and $y_t \in \{-1, +1\}$ is the label. Whenever the predictor μ_t has margin less than 1, AROW performs the update

$$\beta_t = \frac{1}{\langle z_t, \Sigma_t z_t\rangle + \lambda}, \quad \alpha_t = [1 - y_t \langle z_t, \mu_t\rangle]_+ , \quad \mu_{t+1} = \mu_t + \alpha_t \Sigma_t y_t z_t, \quad \Sigma_{t+1} = \Sigma_t - \beta_t \Sigma_t x_t x_t^\top \Sigma_t. \tag{8}$$

In the above, one can set Σ_t to be diagonal, which reduces run-time and storage requirements but still gives good performance (Crammer et al., 2009). In contrast to AROW, the ADAGRAD family uses the *root* of a covariance-like matrix, a consequence of our formal analysis. Crammer et al.'s algorithm and our algorithms have similar run times—linear in the dimension d—when using diagonal matrices. However, when using full matrices the runtime of their algorithm is $O(d^2)$, which is faster than ours.

Our approach differs from previous approaches since instead of focusing on a particular loss function or mistake bound, we view the problem of adapting the proximal function as an online (meta) learning problem. We then obtain bounds comparable to the bound obtained using the best proximal function chosen in hindsight. Our bounds are applicable to any convex Lipschitz loss and composite objective functions.

2 Adaptive Proximal Functions

In this section we give the template regret bounds for the family of subgradient algorithms we consider. Examining several well-known optimization bounds (e.g. Beck and Teboulle, 2003; Nesterov, 2009; Duchi et al., 2010b), we see that we can bound the regret as

$$R_\phi(T) \le \frac{1}{\eta} B_\psi(x^*, x_1) + \frac{\eta}{2} \sum_{t=1}^{T} \|f_t'(x_t)\|_*^2 . \tag{9}$$

Most of the regret depends on dual-norms of $f_t'(x_t)$, where the dual norm in turn depends on the choice of ψ. This naturally leads to the question of whether we can modify the proximal term ψ along the run of the algorithm in order to lower the contribution of the aforementioned norms. We achieve this goal by keeping second order information about the sequence f_t.

We begin by providing two corollaries based on previous work that give the regret of our base algorithms when the proximal function ψ_t is allowed to change. We assume that ψ_t is monotonically non-decreasing, that is, $\psi_{t+1}(x) \ge \psi_t(x)$. We also assume that ψ_t is 1-strongly convex with respect to a time-dependent seminorm $\|\cdot\|_{\psi_t}$. Formally,

$$\psi_t(y) \ge \psi_t(x) + \langle \nabla\psi_t(x), y - x\rangle + \frac{1}{2}\|x - y\|_{\psi_t}^2 .$$

Strong convexity is guaranteed if and only if $B_{\psi_t}(x, y) \ge \frac{1}{2}\|x - y\|_{\psi_t}^2$. We also denote the dual norm of $\|\cdot\|_{\psi_t}$ by $\|\cdot\|_{\psi_t^*}$. For completeness, we provide the proofs of following two corollaries in the long version of this paper (Duchi et al., 2010a), though they build straightforwardly on Duchi et al. (2010b) and Xiao (2009). For the primal-dual subgradient update of Eq. (5), the following regret bound holds.

Corollary 2 *Let the sequence $\{x_t\}$ be defined by the update in Eq. (5). Then for any x^*, we have*

$$R_\phi(T) \le \frac{1}{\eta}\psi_T(x^*) + \frac{\eta}{2}\sum_{t=1}^{T} \|f_t'(x_t)\|_{\psi_{t-1}^*}^2 . \tag{10}$$

For composite mirror descent algorithms (Eq. (6)), under the assumption w.l.o.g. that $\varphi(x_1) = 0$, we have

Corollary 3 *Let the sequence $\{x_t\}$ be defined by the update in Eq. (6). Then for any x^*,*

$$R_\phi(T) \le \frac{1}{\eta} B_{\psi_1}(x^*, x_1) + \frac{1}{\eta}\sum_{t=1}^{T-1}\left[B_{\psi_{t+1}}(x^*, x_{t+1}) - B_{\psi_t}(x^*, x_{t+1})\right] + \frac{\eta}{2}\sum_{t=1}^{T}\|f_t'(x_t)\|_{\psi_t^*}^2 . \tag{11}$$

The above corollaries allow us to prove regret bounds for a family of algorithms that iteratively modify the proximal functions ψ_t.

Algorithm 1 ADAGRAD with Diagonal Matrices

Input: $\eta > 0$, $\delta \geq 0$. Initialize $x_1 = 0$, $g_{1:0} = []$
for $t = 1$ to T **do**
Suffer loss $f_t(x_t)$, receive subgradient $g_t \in \partial f_t(x_t)$ of f_t at x_t
Update $g_{1:t} = [g_{1:t-1}\ g_t]$, $s_{t,i} = \|g_{1:t,i}\|_2$
Set $H_t = \delta I + \operatorname{diag}(s_t)$, $\psi_t(x) = \frac{1}{2}\langle x, H_t\, x\rangle$
Primal-Dual Subgradient Update (Eq. (5)):

$$x_{t+1} = \operatorname*{argmin}_{x\in\mathcal{X}} \left\{ \eta \left\langle \frac{1}{t}\sum_{\tau=1}^{t} g_\tau, x \right\rangle + \eta\varphi(x) + \frac{1}{t}\psi_t(x) \right\}.$$

Composite Mirror Descent Update (Eq. (6)):

$$x_{t+1} = \operatorname*{argmin}_{x\in\mathcal{X}} \left\{ \eta \langle g_t, x\rangle + \eta\varphi(x) + B_{\psi_t}(x, x_t) \right\}.$$

end for

3 Diagonal Matrix Proximal Functions

For now we restrict ourselves to using diagonal matrices to define matrix proximal functions and (semi)norms. This restriction serves a two-fold purpose. First, the analysis for the general case is somewhat complicated and thus the analysis of the diagonal case serves as a proxy for better understanding. Second, in problems with high dimension where we expect this type of modification to help, maintaining more complicated proximal functions is likely to be prohibitively expensive. A benefit of the adaptive algorithms is that there is no need to keep track of a learning rate as in previous algorithms, as it is implicitly given by the growth of the proximal function. To remind the reader, $g_{1:t,i}$ is the ith row of the matrix obtained by concatenating the subgradients from iteration 1 through t in the online algorithm.

To provide some intuition for Alg. 1, let us find the retrospectively optimal proximal function. If the proximal function chosen is $\psi(x) = \frac{1}{2}\langle x, \operatorname{diag}(s)x\rangle$ for some $s \succeq 0$, then the associated norm is $\|x\|^2 = \langle x, \operatorname{diag}(s)x\rangle$ and the dual norm is $\|x\|_*^2 = \langle x, \operatorname{diag}(s)^{-1}x\rangle$. Recalling Eq. (9), we consider the problem

$$\min_s \sum_{t=1}^{T} \left\langle g, \operatorname{diag}(s)^{-1} g \right\rangle \quad \text{s.t. } s \succeq 0,\ \langle \mathbb{1}, s\rangle \leq c\,.$$

This problem is solved by setting $s_i = \|g_{1:T,i}\|_2$ and scaling s so that $\langle s, \mathbb{1}\rangle = c$. To see this, we can write the Lagrangian of the minimization problem by introducing multipliers $\lambda \succeq 0$ and $\theta \geq 0$ to get

$$\mathcal{L}(s, \lambda, \theta) = \sum_{i=1}^{d} \frac{\|g_{1:T,i}\|_2^2}{s_i} - \langle \lambda, s\rangle + \theta(\langle 1, s\rangle - c).$$

Taking derivatives to find the infimum of $\mathcal{L}$, we see that $-\|g_{1:T,i}\|_2^2 / s_i^2 - \lambda_i + \theta = 0$, and the complementarity conditions (Boyd and Vandenberghe, 2004) on $\lambda_i s_i$ imply that $\lambda_i = 0$. Thus we have $s_i = \theta^{-\frac{1}{2}} \|g_{1:T,i}\|_2$, and normalizing using θ gives $s_i = c\,\|g_{1:T,i}\|_2 \,/\, \sum_{j=1}^{d} \|g_{1:T,j}\|_2$. As a final note, plugging s_i in gives

$$\inf_s \left\{ \sum_{t=1}^{T}\sum_{i=1}^{d} \frac{g_{t,i}^2}{s_i} \ :\ s \succeq 0, \langle 1, s\rangle \leq c \right\} = \frac{1}{c}\left(\sum_{i=1}^{d} \|g_{1:T,i}\|_2 \right)^2 . \tag{12}$$

It is natural to suspect that if we use a proximal function similar to $\psi(x) = \frac{1}{2}\langle x, \operatorname{diag}(s)x\rangle$, we should do well lowering the gradient terms in the regret in Eq. (10) and Eq. (11).

To prove a regret bound for our Alg. 1, we note that both types of updates have regret bounds including a term dependent solely on the gradients obtained along the algorithm's run. Thus, the following lemma, which says that the choice of ψ_t in Alg. 1 is optimal up to a multiplicative factor of 2, is applicable to both.

Lemma 4 *Let $g_t = f_t'(x_t)$ and $g_{1:t}$ and s_t be defined as in Alg. 1. Then*

$$\sum_{t=1}^{T} \left\langle g_t, \operatorname{diag}(s_t)^{-1} g_t \right\rangle \leq 2 \sum_{i=1}^{d} \|g_{1:T,i}\|_2 \ .$$

Proof: We prove the lemma by considering an arbitrary $\mathbb{R}$-valued sequence $\{a_i\}$ and its vector representation $a_{1:i} = [a_1 \ \cdots \ a_i]$. We are going to show that (where we set $0/0 = 0$)

$$\sum_{t=1}^{T} \frac{a_t^2}{\|a_{1:t}\|_2} \leq 2 \|a_{1:T}\|_2 \ . \tag{13}$$

We use induction on T. For $T = 1$, the inequality trivially holds. Assume Eq. (13) holds true for $T-1$, then

$$\sum_{t=1}^{T} \frac{a_t^2}{\|a_{1:t}\|_2} = \sum_{t=1}^{T-1} \frac{a_t^2}{\|a_{1:t}\|_2} + \frac{a_T^2}{\|a_{1:T}\|_2} \leq 2 \|a_{1:T-1}\|_2 + \frac{a_T^2}{\|a_{1:T}\|_2} \ ,$$

where the inequality follows from the inductive hypothesis. We now define $b_T = \sum_{t=1}^{T} a_t^2$ and use first-order inequality for concavity to obtain that so long as $b_T - a_T^2 \geq 0$, we have $\sqrt{b_T - a_T^2} \leq \sqrt{b_T} - a_T^2 \frac{1}{2\sqrt{b_T}}$ (we use an identical technique in the full-matrix case; see Lemma 10). Thus

$$2 \|a_{1:T-1}\|_2 + \frac{a_T^2}{\|a_{1:T}\|_2} = 2\sqrt{b_T - a_T^2} + \frac{a_T^2}{\sqrt{b_T}} \leq 2\sqrt{b_T} = 2 \|a_{1:T}\|_2 \ .$$

Having proved Eq. (13), we note that by construction $s_{t,i} = \|g_{1:t,i}\|_2$, thus,

$$\sum_{t=1}^{T} \left\langle g_t, \operatorname{diag}(s_t)^{-1} g_t \right\rangle = \sum_{t=1}^{T} \sum_{i=1}^{d} \frac{g_{t,i}^2}{\|g_{1:t,i}\|_2} \leq 2 \sum_{i=1}^{d} \|g_{1:T,i}\|_2 \ .$$

■

To get a regret bound, we consider the terms consisting of the dual-norm of the subgradients in Eq. (10) and Eq. (11). When $\psi_t(x) = \langle x, (\delta I + \operatorname{diag}(s_t))x \rangle$, the associated dual-norm is $\|g\|_{\psi_t^*}^2 = \left\langle g, (\delta I + \operatorname{diag}(s_t))^{-1} g \right\rangle$. From the definition of s_t in Alg. 1, we clearly have $\|f_t'(x_t)\|_{\psi_t^*}^2 \leq \left\langle g_t, \operatorname{diag}(s_t)^{-1} g_t \right\rangle$. We replace the inverse with a pseudo-inverse if needed, which is well defined since g_t is always in the column-space of $\operatorname{diag}(s_t)$. Thus, Lemma 4 gives

$$\sum_{t=1}^{T} \|f_t'(x_t)\|_{\psi_t^*}^2 \leq 2 \sum_{i=1}^{d} \|g_{1:T,i}\|_2 \, .$$

To obtain a bound for a primal-dual subgradient method, we set $\delta \geq \max_t \|g_t\|_\infty$, in which case $\|g_t\|_{\psi_{t-1}^*}^2 \leq \left\langle g_t, \operatorname{diag}(s_t)^{-1} g_t \right\rangle$, and follow the same lines of reasoning.

It remains to bound the various Bregman divergence terms in Corollary 3 and the term $\psi_T(x^*)$ in Corollary 2. We focus first on composite mirror-descent updates. Examining Eq. (11) and Alg. 1, we notice that

$$\begin{aligned} B_{\psi_{t+1}}(x^*, x_{t+1}) - B_{\psi_t}(x^*, x_{t+1}) &= \frac{1}{2} \left\langle x^* - x_{t+1}, \operatorname{diag}(s_{t+1} - s_t)(x^* - x_{t+1}) \right\rangle \\ &\leq \frac{1}{2} \max_i (x_i^* - x_{t+1,i})^2 \|s_{t+1} - s_t\|_1 \, . \end{aligned}$$

Since $\|s_{t+1} - s_t\|_1 = \langle s_{t+1} - s_t, 1 \rangle$ and $\langle s_T, 1 \rangle = \sum_{i=1}^{d} \|g_{1:T,i}\|_2$, we have

$$\begin{aligned} \sum_{t=1}^{T-1} B_{\psi_{t+1}}(x^*, x_{t+1}) - B_{\psi_t}(x^*, x_{t+1}) &\leq \frac{1}{2} \sum_{t=1}^{T-1} \|x^* - x_{t+1}\|_\infty^2 \left\langle s_{t+1} - s_t, \mathbb{1} \right\rangle \\ &\leq \frac{1}{2} \max_{t \leq T} \|x^* - x_t\|_\infty^2 \sum_{i=1}^{d} \|g_{1:T,i}\|_2 - \frac{1}{2} \|x^* - x_1\|_\infty^2 \left\langle s_1, \mathbb{1} \right\rangle \ . \end{aligned} \tag{14}$$

We also have

$$\psi_T(x^*) = \delta \|x^*\|_2^2 + \langle x^*, \operatorname{diag}(s_T) x^* \rangle \leq \delta \|x^*\|_2^2 + \|x^*\|_\infty^2 \sum_{i=1}^{d} \|g_{1:T,i}\|_2 \, .$$

Combining the above arguments with Corollaries 2 and 3, and combining Eq. (14) with the fact that $B_{\psi_1}(x^*, x_1) \leq \frac{1}{2} \|x^* - x_1\|_\infty^2 \langle 1, s_1 \rangle$, we have proved the following theorem.

Theorem 5 *Let the sequence $\{x_t\}$ be defined by Algorithm 1. If we generate x_t using the primal-dual subgradient update of Eq. (5) and $\delta \geq \max_t \|g_t\|_\infty$, then for any $x^* \in \mathcal{X}$ we have*

$$R_\phi(T) \leq \frac{\delta}{\eta} \|x^*\|_2^2 + \frac{1}{\eta} \|x^*\|_\infty^2 \sum_{i=1}^d \|g_{1:T,i}\|_2 + \eta \sum_{i=1}^d \|g_{1:T,i}\|_2 \,. \tag{15}$$

If we use Algorithm 1 with the composite mirror-descent update of Eq. (6), then for any $x^* \in \mathcal{X}$

$$R_\phi(T) \leq \frac{1}{2\eta} \max_{t \leq T} \|x^* - x_t\|_\infty^2 \sum_{i=1}^d \|g_{1:T,i}\|_2 + \eta \sum_{i=1}^d \|g_{1:T,i}\|_2 \,. \tag{16}$$

The above theorem is a bit unwieldy. We thus perform a few algebraic simplifications to get the next corollary. Let us assume that $\mathcal{X}$ is compact and set $D_\infty = \sup_{x \in \mathcal{X}} \|x - x^*\|_\infty$. Furthermore, define

$$\gamma_T = \sum_{i=1}^d \|g_{1:T,i}\|_2 = \sqrt{\inf_s \left\{ \sum_{t=1}^T \langle g_t, \operatorname{diag}(s)^{-1} g_t \rangle : \langle 1, s \rangle \leq \sum_{i=1}^d \|g_{1:T,i}\|_2 \,,\ s \succeq 0 \right\}} \,.$$

The following corollary is immediate.

Corollary 6 *Assume that D_∞ and γ_T are defined as above. If we generate the sequence $\{x_t\}$ be given by Algorithm 1 using the primal-dual subgradient update Eq. (5) with $\eta = \|x^*\|_\infty$, then for any $x^* \in \mathcal{X}$*

$$R_\phi(T) \leq 2 \|x^*\|_\infty \sum_{i=1}^d \|g_{1:T,i}\|_2 + \delta \frac{\|x^*\|_2^2}{\|x^*\|_\infty} \leq 2 \|x^*\|_\infty \gamma_T + \delta \|x^*\|_1 \,.$$

Using the composite mirror descent update of Eq. (6) to generate $\{x_t\}$ and setting $\eta = D_\infty / \sqrt{2}$, we have

$$R_\phi(T) \leq \sqrt{2} D_\infty \sum_{i=1}^d \|g_{1:T,i}\|_2 = \sqrt{2} D_\infty \gamma_T \,.$$

We can also prove Corollary 1.

Proof of Corollary 1: The proof simply uses Theorem 5, Corollary 6, and the fact that

$$\inf_s \left\{ \sum_{t=1}^T \sum_{i=1}^d \frac{g_{t,i}^2}{s_i} \ :\ s \succeq 0, \langle 1, s \rangle \leq d \right\} = \frac{1}{d} \left(\sum_{i=1}^d \|g_{1:T,i}\|_2 \right)^2$$

as in Eq. (12) in the beginning of this section. Plugging the γ_T term in from Corollary 6 and multiplying D_∞ by $\sqrt{d}$ completes the proof. ■

Intuitively, as discussed in the introduction, Alg. 1 should have good properties on sparse data. For example, suppose that our gradient terms are based on 0/1-valued features for a logistic regression task. Then the gradient terms in the bound $\sum_{i=1}^d \|g_{1:t,i}\|_2$ should all be much smaller than $\sqrt{T}$. If we assume that some features appear much more frequently than others, then the infimal representation of γ_T and the infimal equality in Corollary 1 show that we can have much lower learning rates on commonly appearing features and higher rates on uncommon features, and this will significantly lower the bound on the regret. Further, if we are constructing a relatively dense predictor x—as is often the case in sparse prediction problems—then $\|x^*\|_\infty$ is the best p-norm we can have in the regret.

4 Full Matrix Proximal Functions

In this section we derive and analyze new updates when we estimate a full matrix for the proximal function ψ_t instead of a diagonal one. In this generalized case, the algorithm uses the the square-root of the matrix of outer products of the gradients that observed to update the parameters. As in the diagonal case, we build on intuition garnered from an optimization problem. We seek a matrix S that solves the minimization problem

$$\min_S \ \sum_{t=1}^T \left\langle g_t, S^{-1} g_t \right\rangle \ \text{ s.t. } S \succeq 0, \ \operatorname{tr}(S) \leq c \,.$$

The solution is obtained by defining $G_t = \sum_{\tau=1}^t g_\tau g_\tau^\top$, and then setting S to be a normalized version of the root of G_T, that is, $S = c\, G_T^{1/2} / \operatorname{tr}(G_T^{1/2})$. The next proposition formalizes this statement, and also shows that when G_T is not full rank we can instead use its pseudo-inverse. The proof is in Duchi et al. (2010a).

Algorithm 2 ADAGRAD with Full Matrices

Input $\eta > 0$, $\delta \geq 0$. Initialize $x = 0$, $S_0 = 0$, $H_0 = 0$, $G_0 = 0$

for $t = 1$ to T **do**

Suffer loss $f_t(x_t)$, receive subgradient $g_t \in \partial f_t(x_t)$ of f_t at x_t.

Update $G_t = G_{t-1} + g_t g_t^\top$, $S_t = G_t^{\frac{1}{2}}$.

Let $H_t = \delta I + S_t$, $\psi_t(x) = \frac{1}{2}\langle x, H_t x\rangle$

Primal-Dual Subgradient Update (Eq. (5))

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ \eta \left\langle \frac{1}{t}\sum_{\tau=1}^{t} g_t, x \right\rangle + \eta\varphi(x) + \frac{1}{t}\psi_t(x) \right\}$$

Composite Mirror Descent Update (Eq. (6))

$$x_{t+1} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ \eta \langle g_t, x\rangle + \eta\varphi(x) + B_{\psi_t}(x, x_t) \right\}$$

end for

Proposition 7 *Consider the following minimization problem:*

$$\min_S \ \operatorname{tr}(S^{-1}A) \ \text{ subject to } \ S \succeq 0, \ \operatorname{tr}(S) \leq c \ \text{ where } \ A \succeq 0\,. \tag{17}$$

If A is of full rank, then the minimizer of Eq. (17) is $S = cA^{\frac{1}{2}}/\operatorname{tr}(A^{\frac{1}{2}})$. If A is not of full rank, then setting $S = cA^{\frac{1}{2}}/\operatorname{tr}(A^{\frac{1}{2}})$ gives

$$\operatorname{tr}(S^\dagger A) = \inf_S \left\{ \operatorname{tr}(S^{-1}A) : S \succeq 0, \ \operatorname{tr}(S) \leq c \right\}\,.$$

In either case, $\operatorname{tr}(S^\dagger A) = \operatorname{tr}(A^{\frac{1}{2}})^2/c$.

If we iteratively use proximal functions of the form $\psi_t(x) = \langle x, G_t^{1/2} x\rangle$, we hope as earlier to attain low regret and collect gradient information. We achieve our low regret goal by employing a similar doubling lemma to Lemma 4. The resulting algorithm is given in Alg. 2, and the next theorem provides a quantitative analysis of the motivation above.

Theorem 8 *Let G_t be the outer product matrix defined above. If we generate x_t using the primal-dual subgradient update of Eq. (5) and $\delta \geq \max_t \|g_t\|_2$, then for any $x^* \in \mathcal{X}$*

$$R_\phi(T) \leq \frac{\delta}{\eta}\|x^*\|_2^2 + \frac{1}{\eta}\|x^*\|_2^2 \operatorname{tr}(G_T^{1/2}) + \eta \operatorname{tr}(G_T^{1/2}). \tag{18}$$

If we use Algorithm 2 with the composite mirror-descent update of Eq. (6), then for any x^* and $\delta \geq 0$

$$R_\phi(T) \leq \frac{\delta}{\eta}\|x^*\|_2^2 + \frac{1}{2\eta}\max_{t \leq T}\|x^* - x_t\|_2^2 \operatorname{tr}(G_T^{1/2}) + \eta \operatorname{tr}(G_T^{1/2}). \tag{19}$$

Proof: To begin, we consider the difference between the divergence terms at time $t+1$ and time t from Eq. (11) in Corollary 3. Let $\lambda_{\max}(M)$ denote the largest eigenvalue of a matrix M. We have

$$\begin{aligned} B_{\psi_{t+1}}(x^*, x_{t+1}) - B_{\psi_t}(x^*, x_{t+1}) \ &= \ \frac{1}{2}\left\langle x^* - x_{t+1}, ({G_{t+1}}^{1/2} - {G_t}^{1/2})(x^* - x_{t+1}) \right\rangle \\ \leq \ \frac{1}{2}\|x^* - x_{t+1}\|_2^2 \lambda_{\max}(G_{t+1}^{1/2} - G_t^{1/2}) \ &\leq \ \frac{1}{2}\|x^* - x_{t+1}\|_2^2 \operatorname{tr}(G_{t+1}^{1/2} - G_t^{1/2})\,. \end{aligned}$$

For the last inequality we used the fact that the trace of a matrix is equal to the sum of its eigenvalues along with the property ${G_{t+1}}^{1/2} - {G_t}^{1/2} \succeq 0$ (Davis, 1963, Example 3) and therefore $\operatorname{tr}(G_{t+1}^{1/2} - G_t^{1/2}) \geq \lambda_{\max}(G_{t+1}^{1/2} - G_t^{1/2})$. Thus, we get

$$\begin{aligned} \sum_{t=1}^{T-1} B_{\psi_{t+1}}(x^*, x_{t+1}) - B_{\psi_t}(x^*, x_{t+1}) &\leq \frac{1}{2}\sum_{t=1}^{T-1}\|x^* - x_{t+1}\|_2^2 \left(\operatorname{tr}(G_{t+1}^{1/2}) - \operatorname{tr}(G_t^{1/2})\right) \\ &\leq \frac{1}{2}\max_{t \leq T}\|x^* - x_t\|_2^2 \operatorname{tr}({G_T}^{1/2}) - \frac{1}{2}\|x^* - x_1\|_2^2 \operatorname{tr}(G_1^{1/2})\,. \end{aligned} \tag{20}$$

For the last inequality we used the fact that G_1 is a rank 1 PSD matrix with non-negative trace. What remains is to bound the gradient terms common to both updates. The following lemma is directly applicable.

Lemma 9 *Let $S_t = {G_t}^{1/2}$ be as defined in Alg. 2. Then, using the pseudo-inverse when necessary,*

$$\sum_{t=1}^{T} \left\langle g_t, S_t^{-1} g_t \right\rangle \leq 2 \sum_{t=1}^{T} \left\langle g_t, S_T^{-1} g_t \right\rangle = 2 \operatorname{tr}({G_T}^{1/2}) \ .$$

Before we prove the lemma, we state two linear-algebraic lemmas that make its proof and that of the theorem much more straightforward. The lemmas are quite technical, so we prove them in the long version of this paper (Duchi et al., 2010a). The first auxiliary lemma is the matrix-analogue of the fact that for nonnegative x, y with $x \geq y$, $\sqrt{x-y} \leq \sqrt{x} - y/(2\sqrt{x})$, a consequence of the concavity of $\sqrt{\cdot}$.

Lemma 10 *Let $B \succeq 0$ and $B^{-1/2}$ denote the root of the inverse (or pseudo-inverse) of B. For any c such that $B - cgg^\top \succeq 0$, the following inequality holds:*

$$2 \operatorname{tr}((B - cgg^\top)^{1/2}) \leq 2 \operatorname{tr}(B^{1/2}) - c \operatorname{tr}(B^{-1/2} gg^\top) \ .$$

Lemma 11 *Let $\delta \geq \|g\|_2$ and $A \succeq 0$. Then $\left\langle g, (\delta I + A^{1/2})^{-1} g \right\rangle \leq \left\langle g, \left((A + gg^\top)^\dagger\right)^{1/2} g \right\rangle$.*

Proof of Lemma 9: We prove the lemma by induction. The base case is immediate, since $\langle g_1, G_1^{-1/2} g_1 \rangle = \frac{\langle g_1, g_1 \rangle}{\|g_1\|_2} = \|g_1\|_2 \leq 2 \|g_1\|_2$. Now, assume the lemma is true for $T-1$, so from the inductive assumption

$$\sum_{t=1}^{T} \left\langle g_t, S_t^{-1} g_t \right\rangle \leq 2 \sum_{t=1}^{T-1} \left\langle g_t, S_{T-1}^{-1} g_t \right\rangle + \left\langle g_T, S_T^{-1} g_T \right\rangle \ .$$

Since S_{T-1} does not depend on t, $\sum_{t=1}^{T-1} \left\langle g_t, S_{T-1}^{-1} g_t \right\rangle = \operatorname{tr}\left(S_{T-1}^{-1} \sum_{t=1}^{T-1} g_t g_t^\top\right) = \operatorname{tr}(G_{T-1}^{-1/2} G_{T-1})$, where the right-most equality follows from the definitions of S_t and G_t. Therefore, we get

$$\sum_{t=1}^{T} \left\langle g_t, S_t^{-1} g_t \right\rangle \leq 2 \operatorname{tr}(G_{T-1}^{-1/2} G_{T-1}) + \left\langle g_T, G_T^{-1/2} g_T \right\rangle = 2 \operatorname{tr}(G_{T-1}^{1/2}) + \left\langle g_T, G_T^{-1/2} g_T \right\rangle \ .$$

Lemma 10, which also justifies the use of pseudo-inverses, lets us exploit the concavity of the function $\operatorname{tr}(A^{1/2})$ to bound the above sum by $2 \operatorname{tr}(G_T^{1/2})$. ▲

We can now finalize our proof of the theorem. As in the diagonal case, we have that the squared dual norm (seminorm when $\delta = 0$) associated with ψ_t is

$$\|x\|_{\psi_t^*}^2 = \left\langle x, (\delta I + S_t)^{-1} x \right\rangle \ .$$

Thus it is clear that $\|g_t\|_{\psi_t^*}^2 \leq \left\langle g_t, S_t^{-1} g_t \right\rangle$. For the dual-averaging algorithms, we use Lemma 11 to see that $\|g_t\|_{\psi_{t-1}^*}^2 \leq \left\langle g_t, S_t^{-1} g_t \right\rangle$ so long as $\delta \geq \|g_t\|_2$. The doubling inequality from Lemma 9 implies that $\sum_{t=1}^{T} \|f_t'(x_t)\|_{\psi_t^*}^2 \leq 2 \operatorname{tr}(G_T^{1/2})$ for mirror-descent algorithms and that $\sum_{t=1}^{T} \|f_t'(x_t)\|_{\psi_{t-1}^*}^2 \leq 2 \operatorname{tr}(G_T^{1/2})$ for primal-dual subgradient algorithms.

Note that $B_{\psi_1}(x^*, x_1) \leq \frac{1}{2} \|x^* - x_1\|_2^2 \operatorname{tr}(G_1^{1/2})$ when $\delta = 0$. Combining the first of the last bounds in the previous paragraph with this and the bound on $\sum_{t=1}^{T-1} B_{\psi_{t+1}}(x^*, x^{t+1}) - B_{\psi_t}(x^*, x^{t+1})$ from Eq. (20), we see that Corollary 3 gives the bound for the mirror-descent family of algorithms. Combining the second of the bounds in the previous paragraph and Eq. (20) with Corollary 2 gives the desired bound on $R_\phi(T)$ for the primal-dual subgradient algorithms, which completes the proof of the theorem. ■

As before, we give a corollary that clarifies the bound implied by Theorem 8. The infimal equalities in the corollary use Proposition 7. The corollary suggests that if there is a rotation of the space in which the gradient vectors g_t have small inner products—a sparse basis for the subgradients g_t—then using full-matrix proximal functions can significantly lower the regret.

Corollary 12 *The sequence $\{x_t\}$ generated by Alg. 2 with the primal-dual update and $\eta = \|x^*\|_2$ satisfies*

$$R_\phi(T) \leq 2 \|x^*\|_2 \operatorname{tr}(G_T^{1/2}) + \delta \|x^*\|_2 = 2\sqrt{d} \|x^*\|_2 \sqrt{\inf_S \left\{ \sum_{t=1}^{T} \left\langle g_t, S^{-1} g_t \right\rangle \ : \ S \succeq 0, \operatorname{tr}(S) \leq d \right\}} + \delta \|x^*\|_2 \ .$$

Let $\mathcal{X}$ be compact so that $\sup_{x \in \mathcal{X}} \|x - x^*\|_2 \leq D_2$. Let $\eta = D_2/\sqrt{2}$ and $\{x_t\}$ be generated by Alg. 2 using the composite mirror descent update with $\delta = 0$. Then

$$R_\phi(T) \leq \sqrt{2} D_2 \operatorname{tr}(G_T^{1/2}) = \sqrt{2d} D_2 \sqrt{\inf_S \left\{ \sum_{t=1}^{T} \left\langle g_t, S^{-1} g_t \right\rangle \ : \ S \succeq 0, \operatorname{tr}(S) \leq d \right\}} \ .$$

5 Lowering the Regret for Strongly Convex Functions

It is now well established that strong convexity of the functions f_t can give significant improvements in the regret of online convex optimization algorithms (Hazan et al., 2006; Shalev-Shwartz and Singer, 2007). We can likewise derive lower regret bounds in the presence of strong convexity. We assume that our functions $f_t + \varphi$ are strongly convex with respect to a norm $\|\cdot\|$. For simplicity, we assume that each has the same strong convexity parameter λ,

$$f_t(y) + \varphi(y) \geq f_t(x) + \varphi(x) + \langle f_t'(x), y - x \rangle + \langle \varphi'(x), y - x \rangle + \frac{\lambda}{2} \|x - y\|^2 .$$

We focus on composite mirror descent algorithms, as the analysis of strongly convex variants of primal-dual subgradient algorithms does not seem to lend itself to dynamic learning rate adaptation. The tightest analysis of the primal-dual method for strongly-convex functions keeps the function ψ intact rather than growing it at a rate of $\sqrt{t}$, as in standard RDA (Xiao, 2009). Allowing ψ to grow makes attaining the stronger regret bound impossible. It may be possible to analyze RDA when the *regularization* function φ is time-dependent, but we leave this topic to future research. Without loss of generality let $\varphi(x_1) = 0$ and $x_1 = 0$. Rather than give the proof of the lower regret, we simply state the result, as it is not difficult to prove using techniques of Hazan et al. (2006), though we include the proof in the full version of this paper (Duchi et al., 2010a).

Theorem 13 *Assume that φ is λ-strongly convex with respect to $\|\cdot\|_2^2$ over the set $\mathcal{X}$. Assume further that $\|g\|_\infty \leq G_\infty$ for all $g \in \partial f_t(x)$ for $x \in \mathcal{X}$. Let $\{x_t\}$ be the sequence of vectors generated by Algorithm 1 with the diagonal proximal function $\psi_t(x) = \langle x, (\delta I + \operatorname{diag}(s_t))x \rangle$ and $s_{t,i} = \|g_{1:t,i}\|_2^2$. Setting $\eta \geq \frac{G_\infty^2}{\lambda}$, the regret is bounded by*

$$R_\phi(T) \leq \frac{2G_\infty^2 \delta}{\lambda} \|x_1 - x^*\|_2^2 + \frac{G_\infty^2}{\lambda} \sum_{i=1}^{d} \log \left(\frac{\|g_{1:T,i}\|_2^2}{\delta} + 1 \right) = O\left(\frac{dG_\infty^2}{\lambda} \log(TG_\infty) \right) .$$

6 Experiments

In this section, we present the results of experiments with natural datasets that suggest that adaptive methods significantly outperform related non-adaptive methods. We focus on the fully stochastic optimization setting, in which at each iteration the learning algorithm receives a single example. We measure performance using two metrics: the online loss or error and the test set performance of the predictor the learning algorithm outputs at the end of a single pass through the training data. We also give some results that show how imposing sparsity constraints (in the form of ℓ_1 and mixed-norm regularization) affects the learning algorithm's performance. One benefit of the ADAGRAD framework is its ability to straightforwardly generalize to domain constraints $\mathcal{X} \neq \mathbb{R}^d$ and arbitrary regularization functions φ, in contrast to previous adaptive online algorithms. See Duchi et al. (2010a) for a more complete experimental evaluation.

We experiment with RDA (Xiao, 2009), FOBOS (Duchi and Singer, 2009), adaptive RDA, adaptive FOBOS, the Passive-Aggressive (PA) algorithm (Crammer et al., 2006), and AROW (Crammer et al., 2009). To remind the reader, PA is an online learning procedure with the update

$$x_{t+1} = \operatorname*{argmin}_x \left[1 - y_t \langle z_t, x \rangle\right]_+ + \frac{\lambda}{2} \|x - x_t\|_2^2 ,$$

where λ is a regularization parameter. PA's update is similar to the update employed by AROW (see Eq. (8)), but the latter maintains second order information on x. Using the representer theorem, it is also possible to derive efficient updates for PA and AROW for the logistic loss, $\log(1 + \exp(-y_t \langle z_t, x_t \rangle))$. We thus compare the above six algorithms using both hinge and logistic loss.

The Reuters RCV1 dataset is a collection of approximately 800,000 text articles, each of which is assigned multiple labels. There are 4 high-level categories—Economics, Commerce, Medical, and Government (ECAT, CCAT, MCAT, GCAT)—and multiple more specific categories. We focus on training binary classifiers for each of the four major categories. The input features we use are 0/1 bigram features, which (post word stemming) yield a representation of approximately 2 million dimensions. The feature vectors are very sparse, however, and most examples have fewer than 5000 non-zero features.

We compare the twelve different algorithms mentioned in the prequel as well as variants of FOBOS and RDA with ℓ_1-regularization. We summarize the results of the ℓ_1-regularized runs as well as AROW and PA in Table 1. We found the results for both the hinge loss and the logistic loss to be qualitatively and quantitatively very similar. We thus report results only for training with the hinge loss in Table 1. Each row in the table represents the average of four different experiments in which we hold out 25% of the data for test and perform a single online learning pass on the remaining 75% of the data. For RDA and FOBOS, we cross-validate the

	RDA	FB	ADAGRAD-RDA	ADAGRAD-FB	PA	AROW
ECAT	.051 (.099)	.058 (.194)	**.044** (**.086**)	**.044** (.238)	.059	.049
CCAT	.064 (.123)	.111 (.226)	**.053** (**.105**)	**.053** (.276)	.107	.061
GCAT	.046 (.092)	.056 (.183)	**.040** (**.080**)	**.040** (.225)	.066	.044
MCAT	.037 (.074)	.056 (.146)	.035 (**.063**)	**.034** (.176)	.053	.039

Table 1: Test set error rates and proportion non-zero weights (in parenthesis) on Reuters RCV1.

stepsize parameter η by running multiple passes and then choosing the output of the learner that had the fewest mistakes during training. For PA and AROW we choose λ using the same approach. We use the same regularization multiplier for the ℓ_1 term to execute RDA and FOBOS. The regularization multiplier was selected so that RDA yielded a weight vector with approximately 10% non-zero components.

It is evident from the results presented in Table 1 that the adaptive algorithms (AROW and ADAGRAD) are far superior to non-adaptive algorithms in terms of error rate on test data. In addition, the ADAGRAD algorithms naturally incorporate sparsity since they were run with ℓ_1-regularization, though RDA obtained significantly higher sparsity levels while the solutions of PA and AROW are dense. Furthermore, although omitted from the table for brevity, in *every* test with the RCV1 corpus, the adaptive algorithms outperformed the non-adaptive algorithms. Moreover, both ADAGRAD-RDA and ADAGRAD-Fobos outperform AROW on all the classification tasks. Unregularized RDA and FOBOS attained similar results to the ℓ_1-regularized variants, though of course the solution of the former versions were not sparse.

7 Conclusions

We presented a paradigm that adapts subgradient methods to the geometry of the problem at hand. The adaptation allows us to derive strong regret guarantees, which for some natural data distributions achieve better performance guarantees than previous algorithms. Our online convergence results can be naturally converted into rate of convergence and generalization bounds (Cesa-Bianchi et al., 2004). The ADAGRAD family of algorithms incorporates regularization through φ and can thus easily generate sparse or otherwise structured solutions. Our algorithms are straightforward to implement and can be easily specialized to many useful constraint sets $\mathcal{X}$ and regularization terms φ. We conducted comprehensive experiments showing that adaptive methods clearly outperform their non-adaptive counterparts. These results are available in the long version of this paper (Duchi et al., 2010a). We believe that there are a few theoretical questions that are still unanswered in this line of work. The first is whether we can *efficiently* use full matrices in the proximal functions, as in Section 4, or whether a different algorithm is necessary. A second open issue is whether it is possible to use non-Euclidean proximal functions. For example, is it possible to adapt the KL divergence between distributions to characteristics of the problem at hand? We hope to investigate such extensions in the near future.

Acknowledgments

The first author was supported by the Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship program. Part of this work was carried out while the first author was working at Google Research.

References

J. Abernethy, P. Bartlett, A. Rakhlin, and A. Tewari. Optimal strategies and minimax lower bounds for online convex games. In *Proceedings of the Twenty First Annual Conference on Computational Learning Theory*, 2008.

P. Auer, N. Cesa-Bianchi, and C. Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002.

P. L. Bartlett, E. Hazan, and A. Rakhlin. Adaptive online gradient descent. In *Advances in Neural Information Processing Systems 20*, 2007.

A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31:167–175, 2003.

A. Bordes, L. Bottou, and P. Gallinari. Sgd-qn: Careful quasi-newton stochastic gradient descent. *Journal of Machine Learning Research*, 10:1737–1754, 2009.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

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

N. Cesa-Bianchi, A. Conconi, , and C. Gentile. A second-order perceptron algorithm. *SIAM Journal on Computing*, 34(3):640–668, 2005.

K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006.

K. Crammer, M. Dredze, and A. Kulesza. Adaptive regularization of weight vectors. In *Advances in Neural Information Processing Systems 23*, 2009.

C. Davis. Notions generalizing convexity for functions defined on spaces of matrices. In *Proceedings of the Symposia in Pure Mathematics*, volume 7, pages 187–201. American Mathematical Society, 1963.

J. Duchi and Y. Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2873–2908, 2009.

J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. Technical Report 2010-24, UC Berkeley Electrical Engineering and Computer Science, 2010a. URL `cs.berkeley.edu/~jduchi/projects/DuchiHaSi10.pdf`.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari. Composite objective mirror descent. In *Proceedings of the Twenty Third Annual Conference on Computational Learning Theory*, 2010b.

R. Fletcher. A new approach to variable metric algorithms. *Computer Journal*, 13:317–322, 1970.

E. Hazan, A. Kalai, S. Kale, and A. Agarwal. Logarithmic regret algorithms for online convex optimization. In *Proceedings of the Nineteenth Annual Conference on Computational Learning Theory*, 2006.

A. Kalai and S. Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2003.

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

G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5), 1988.

S. Shalev-Shwartz and Y. Singer. Logarithmic regret algorithms for strongly convex repeated games. Technical report, The Hebrew University, 2007. URL `http://www.cs.huji.ac.il/~shais`.

N. Z. Shor. Utilization of the operation of space dilation in the minimization of convex functions. *Cybernetics and Systems Analysis*, 6(1):7–15, 1972. Translated from *Kibernetika*.

P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Technical report, Department of Mathematics, University of Washington, 2008.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems 23*, 2009.

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on Machine Learning*, 2003.