Journal of Machine Learning Research 10 (2009) 2899-2934 Submitted 2/09; Revised 7/09; Published 12/09

Efficient Online and Batch Learning Using Forward Backward Splitting

John Duchi JDUCHI@CS.BERKELEY.EDU
Computer Science Division
University of California, Berkeley
Berkeley, CA 94720 USA

Yoram Singer SINGER@GOOGLE.COM
Google
1600 Amphitheatre Pkwy
Mountain View, CA 94043 USA

Editor: Yoav Freund

Abstract

We describe, analyze, and experiment with a framework for empirical loss minimization with regularization. Our algorithmic framework alternates between two phases. On each iteration we first perform an *unconstrained* gradient descent step. We then cast and solve an instantaneous optimization problem that trades off minimization of a regularization term while keeping close proximity to the result of the first phase. This view yields a simple yet effective algorithm that can be used for batch penalized risk minimization and online learning. Furthermore, the two phase approach enables sparse solutions when used in conjunction with regularization functions that promote sparsity, such as ℓ_1. We derive concrete and very simple algorithms for minimization of loss functions with ℓ_1, ℓ_2, ℓ_2^2, and ℓ_∞ regularization. We also show how to construct efficient algorithms for mixed-norm ℓ_1/ℓ_q regularization. We further extend the algorithms and give efficient implementations for very high-dimensional data with sparsity. We demonstrate the potential of the proposed framework in a series of experiments with synthetic and natural data sets.

Keywords: subgradient methods, group sparsity, online learning, convex optimization

1. Introduction

Before we begin, we establish notation for the content of this paper. We denote scalars by lower case letters and vectors by lower case bold letters, for example, $\boldsymbol{w}$. The inner product of two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is denoted $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$. We use $\|\boldsymbol{x}\|_p$ to denote the p-norm of the vector $\boldsymbol{x}$ and $\|\boldsymbol{x}\|$ as a shorthand for $\|\boldsymbol{x}\|_2$.

The focus of this paper is an algorithmic framework for regularized convex programming to minimize the following sum of two functions:

$$f(\boldsymbol{w}) + r(\boldsymbol{w}), \tag{1}$$

where both f and r are convex bounded below functions (so without loss of generality we assume they are into $\mathbb{R}_+$). Often, the function f is an empirical loss and takes the form $\sum_{i \in S} \ell_i(\boldsymbol{w})$ for a sequence of loss functions $\ell_i : \mathbb{R}^n \to \mathbb{R}_+$, and $r(\boldsymbol{w})$ is a regularization term that penalizes for excessively complex vectors, for instance $r(\boldsymbol{w}) = \lambda \|\boldsymbol{w}\|_p$. This task is prevalent in machine learning, in which a learning problem for decision and prediction problems is cast as a convex optimization problem. To that end, we investigate a general and intuitive algorithm, known as *forward-backward splitting*, to minimize Eq. (1), focusing especially on derivations for and use of non-differentiable regularization functions.

Many methods have been proposed to minimize general convex functions such as that in Eq. (1). One of the most general is the subgradient method (see, e.g., Bertsekas, 1999), which is elegant and very simple. Let $\partial f(\boldsymbol{w})$ denote the subgradient set of f at $\boldsymbol{w}$, namely,

$$\partial f(\boldsymbol{w}) = \{\boldsymbol{g} \mid \forall \boldsymbol{v} : f(\boldsymbol{v}) \geq f(\boldsymbol{w}) + \langle \boldsymbol{g}, \boldsymbol{v} - \boldsymbol{w} \rangle\}.$$

Sub-gradient procedures then minimize the function $f(\boldsymbol{w})$ by iteratively updating the parameter vector $\boldsymbol{w}$ according to the update rule

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f \quad ,$$

where η_t is a constant or diminishing a step size and $\boldsymbol{g}_t^f \in \partial f(\boldsymbol{w}_t)$ is an arbitrary vector from the subgradient set of f evaluated at $\boldsymbol{w}_t$. A more general method than the above is the projected gradient method, which iterates

$$\boldsymbol{w}_{t+1} \; = \; \Pi_\Omega\left(\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f\right) \; = \; \operatorname*{argmin}_{\boldsymbol{w}\in\Omega} \left\{ \left\| \boldsymbol{w} - (\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f) \right\|_2^2 \right\}$$

where $\Pi_\Omega(\boldsymbol{w})$ is the Euclidean projection of $\boldsymbol{w}$ onto the set Ω. Standard results (Bertsekas, 1999) show that the (projected) subgradient method converges at a rate of $O(1/\varepsilon^2)$, or equivalently that the error $f(\boldsymbol{w}) - f(\boldsymbol{w}^\star) = O(1/\sqrt{T})$, given some simple assumptions on the boundedness of the subdifferential set and Ω (we have omitted constants dependent on $\|\partial f\|$ or $\dim(\Omega)$).

If we use the subgradient method to minimize Eq. (1), the iterates are simply $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f - \eta_t \boldsymbol{g}_t^r$, where $\boldsymbol{g}_t^r \in \partial r(\boldsymbol{w}_t)$. A common problem in subgradient methods is that if r or f is non-differentiable, the iterates of the subgradient method are very rarely at the points of non-differentiability. In the case of regularization functions such as $r(\boldsymbol{w}) = \|\boldsymbol{w}\|_1$, however, these points (zeros in the case of the ℓ_1-norm) are often the true minima of the function. Furthermore, with ℓ_1 and similar penalties, zeros are desirable solutions as they tend to convey information about the structure of the problem being solved, and in the case of statistical inference, can often yield the correct sparsity structure of the parameters (Zhao and Yu, 2006; Meinshausen and Bühlmann, 2006).

There has been a significant amount of work related to minimizing Eq. (1), especially when the function r is a sparsity-promoting regularizer, and much of it stems from the machine learning, statistics, and optimization communities. We can hardly do justice to the body of prior work, and we provide a few references here to the research we believe is most directly related. The approach we pursue below is known as *forward-backward splitting* in the optimization literature, which is closely related to the proximal method.

The forward-backward splitting method was first proposed by Lions and Mercier (1979) and has been analyzed by several researches in the context of maximal monotone operators in the optimization literature. Chen and Rockafellar (1997) and Tseng (2000) give conditions and modifications of forward-backward splitting to attain linear convergence rates. Combettes and Wajs (2005) give proofs of convergence for forward-backward splitting in Hilbert spaces under asymptotically negligible perturbations, though without establishing strong rates of convergence. Prior work on convergence of the method often requires an assumption of strong monotonicity of the maximal monotone operators (equivalent to strong convexity of at least one of the functions in Eq. (1)), and as far as we know, all analyses assume that f is differentiable with Lipschitz-continuous gradient. The analyses have also been carried out in a non-stochastic and non-online setting.

More recently, Wright et al. (2009) suggested the use of the method for sparse signal reconstruction, where $f(\boldsymbol{w}) = \|\boldsymbol{y} - A\boldsymbol{w}\|^2$, though they note that the method can apply to suitably smooth convex functions f. Nesterov (2007) gives analysis of convergence rates using gradient mapping techniques when f has Lipschitz continuous gradient, which was inspired by Wright et al. In the special case that $r(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_1$, similar methods to the algorithms we investigate have been proposed and termed iterative thresholding (Daubechies et al., 2004) or truncated gradient (Langford et al., 2008) in signal processing and machine learning, but the authors were apparently unaware of the connection to splitting methods.

Similar projected-gradient methods, when the regularization function r is no longer part of the objective function but rather cast as a constraint so that $r(\boldsymbol{w}) \le \lambda$, are also well known (Bertsekas, 1999). In signal processing, the problem is often termed as an inverse problem with sparsity constraints, see for example, Daubechies et al. (2008) and the references therein. Duchi et al. (2008) give a general and efficient projected gradient method for ℓ_1-constrained problems. We make use of one of Duchi et al.'s results in obtaining an efficient algorithm for the case when $r(\boldsymbol{w}) = \|\boldsymbol{w}\|_\infty$ (a setting useful for mixed-norm regularization). There is also a body of literature on regret analysis for online learning and online convex programming with convex constraints, which we build upon here (Zinkevich, 2003; Hazan et al., 2006; Shalev-Shwartz and Singer, 2007). Learning sparse models generally is of great interest in the statistics literature, specifically in

the context of consistency and recovery of sparsity patterns through ℓ_1 or mixed-norm regularization across multiple tasks (Meinshausen and Bühlmann, 2006; Obozinski et al., 2008; Zhao et al., 2006).

In this paper, we describe a general gradient-based framework for online and batch convex programming. To make our presentation a little simpler, we call our approach FOBOS, for FOrward-Backward Splitting.[1] Our proofs are made possible through the use of "forward-looking" subgradients, and FOBOS is a distillation of some the approaches mentioned above for convex programming. Our alternative view lends itself to unified analysis and more general settings, efficient implementation, and provides a flexible tool for the derivation of algorithms for old and new convex programming settings.

The paper is organized as follows. In the next section, we begin by introducing and formally defining the method, giving some simple preliminary analysis. We follow the introduction by giving in Sec. 3 rates of convergence for batch (offline) optimization. We then extend the results to stochastic gradient descent and provide regret bounds for online convex programming in Sec. 4. To demonstrate the simplicity and usefulness of the framework, we derive in Sec. 5 algorithms for several different choices of the regularizing function r, though most of these results are known. We then extend these methods to be efficient in very high dimensional learning settings where the input data is sparse in Sec. 6. Finally, we conclude in Sec. 7 with experiments examining various aspects of the proposed framework, in particular the runtime and sparsity selection performance of the derived algorithms.

2. Forward-Looking Subgradients and Forward-Backward Splitting

Our approach to Forward-Backward Splitting is motivated by the desire to have the iterates $\boldsymbol{w}_t$ attain points of non-differentiability of the function r. The method alleviates the problems of non-differentiability in cases such as ℓ_1-regularization by taking analytical minimization steps interleaved with subgradient steps. Put informally, FOBOS can be viewed as analogous to the *projected* subgradient method while replacing or augmenting the projection step with an instantaneous minimization problem for which it is possible to derive a closed form solution. FOBOS is succinct as each iteration consists of the following two steps:

$$\boldsymbol{w}_{t+\frac{1}{2}} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f, \tag{2}$$

$$\boldsymbol{w}_{t+1} = \operatorname*{argmin}_{\boldsymbol{w}} \left\{ \frac{1}{2} \left\| \boldsymbol{w} - \boldsymbol{w}_{t+\frac{1}{2}} \right\|^2 + \eta_{t+\frac{1}{2}} r(\boldsymbol{w}) \right\} . \tag{3}$$

In the above, $\boldsymbol{g}_t^f$ is a vector in $\partial f(\boldsymbol{w}_t)$ and η_t is the step size at time step t of the algorithm. The actual value of η_t depends on the specific setting and analysis. The first step thus simply amounts to an unconstrained subgradient step with respect to the function f. In the second step we find a new vector that interpolates between two goals: (i) stay close to the interim vector $\boldsymbol{w}_{t+\frac{1}{2}}$, and (ii) attain a low complexity value as expressed by r. Note that the regularization function is scaled by an interim step size, denoted $\eta_{t+\frac{1}{2}}$. The analyses we describe in the sequel determine the specific value of $\eta_{t+\frac{1}{2}}$, which is either η_t or η_{t+1}.

A key property of the solution of Eq. (3) is the necessary condition for optimality and gives the reason behind the name FOBOS. Namely, the zero vector must belong to subgradient set of the objective at the optimum $\boldsymbol{w}_{t+1}$, that is,

$$\mathbf{0} \in \partial \left\{ \frac{1}{2} \left\| \boldsymbol{w} - \boldsymbol{w}_{t+\frac{1}{2}} \right\|^2 + \eta_{t+\frac{1}{2}} r(\boldsymbol{w}) \right\} \Bigg|_{\boldsymbol{w}=\boldsymbol{w}_{t+1}} .$$

Since $\boldsymbol{w}_{t+\frac{1}{2}} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f$, the above property amounts to

$$\mathbf{0} \in \boldsymbol{w}_{t+1} - \boldsymbol{w}_t + \eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \partial r(\boldsymbol{w}_{t+1}). \tag{4}$$

1. An earlier draft of this paper referred to our algorithm as FOLOS, for FOrward LOoking Subgradients. In order not to confuse readers of the early draft, we attempt to stay close to the earlier name and use the acronym FOBOS rather than Fobas.

The property $\mathbf{0} \in \boldsymbol{w}_{t+1} - \boldsymbol{w}_t + \eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \partial r(\boldsymbol{w}_{t+1})$ implies that so long as we choose $\boldsymbol{w}_{t+1}$ to be the minimizer of Eq. (3), we are guaranteed to obtain a vector $\boldsymbol{g}_{t+1}^r \in \partial r(\boldsymbol{w}_{t+1})$ such that

$$\mathbf{0} = \boldsymbol{w}_{t+1} - \boldsymbol{w}_t + \eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r \quad .$$

The above equation can be understood as an update scheme where the new weight vector $\boldsymbol{w}_{t+1}$ is a linear combination of the previous weight vector $\boldsymbol{w}_t$, a vector from the subgradient set of f evaluated at $\boldsymbol{w}_t$, and a vector from the subgradient of r evaluated at the yet to be determined $\boldsymbol{w}_{t+1}$, hence the name Forward-Looking Subgradient. To recap, we can write $\boldsymbol{w}_{t+1}$ as

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \, \boldsymbol{g}_t^f - \eta_{t+\frac{1}{2}} \, \boldsymbol{g}_{t+1}^r, \tag{5}$$

where $\boldsymbol{g}_t^f \in \partial f(\boldsymbol{w}_t)$ and $\boldsymbol{g}_{t+1}^r \in \partial r(\boldsymbol{w}_{t+1})$. Solving Eq. (3) with r above has two main benefits. First, from an algorithmic standpoint, it enables sparse solutions at virtually no additional computational cost. Second, the forward-looking gradient allows us to build on existing analyses and show that the resulting framework enjoys the formal convergence properties of many existing gradient-based and online convex programming algorithms.

3. Convergence Analysis of FOBOS

Upon first look FOBOS looks substantially different from sub-gradient and online convex programming methods. However, the fact that FOBOS actually employs a forward-looking subgradient of the regularization function lets us build nicely on existing analyses. In this section we modify known results while using the forward-looking property of FOBOS to provide convergence rate analysis for FOBOS. To do so we will set $\eta_{t+\frac{1}{2}}$ properly. As we show in the sequel, it is sufficient to set $\eta_{t+\frac{1}{2}}$ to η_t or η_{t+1}, depending on whether we are doing online or batch optimization, in order to obtain convergence and low regret bounds. We start with an analysis of FOBOS in a batch setting. In this setting we use the subgradient of f, set $\eta_{t+\frac{1}{2}} = \eta_{t+1}$ and update $\boldsymbol{w}_t$ to $\boldsymbol{w}_{t+1}$ as prescribed by Eq. (2) and Eq. (3).

Throughout the section we denote by $\boldsymbol{w}^\star$ the minimizer of $f(\boldsymbol{w}) + r(\boldsymbol{w})$. In what follows, define $\|\partial f(\boldsymbol{w})\| \triangleq \sup_{\boldsymbol{g} \in \partial f(\boldsymbol{w})} \|\boldsymbol{g}\|$. We begin by deriving convergence results under the fairly general assumption (see, e.g., Langford et al. 2008 or Shalev-Shwartz and Tewari 2009) that the subgradients are bounded as follows:

$$\|\partial f(\boldsymbol{w})\|^2 \leq A f(\boldsymbol{w}) + G^2, \quad \|\partial r(\boldsymbol{w})\|^2 \leq A r(\boldsymbol{w}) + G^2 \quad . \tag{6}$$

For example, any Lipschitz loss (such as the logistic loss or hinge loss used in support vector machines) satisfies the above with $A = 0$ and G equal to the Lipschitz constant. Least squares estimation satisfies Eq. (6) with $G = 0$ and $A = 4$. The next lemma, while technical, provides a key tool for deriving all of the convergence results in this paper.

Lemma 1 (Bounding Step Differences) *Assume that the norms of the subgradients of the functions f and r are bounded as in Eq. (6):*

$$\|\partial f(\boldsymbol{w})\|^2 \leq A f(\boldsymbol{w}) + G^2, \quad \|\partial r(\boldsymbol{w})\|^2 \leq A r(\boldsymbol{w}) + G^2 \quad .$$

Let $\eta_{t+1} \leq \eta_{t+\frac{1}{2}} \leq \eta_t$ and suppose that $\eta_t \leq 2\eta_{t+\frac{1}{2}}$. If we use the FOBOS *update of Eqs. (2) and (3), then for a constant $c \leq 4$ and any vector $\boldsymbol{w}^\star$,*

$$\begin{aligned} & 2\eta_t(1 - cA\eta_t) f(\boldsymbol{w}_t) + 2\eta_{t+\frac{1}{2}}(1 - cA\eta_{t+\frac{1}{2}}) r(\boldsymbol{w}_{t+1}) \\ & \quad \leq \quad 2\eta_t f(\boldsymbol{w}^\star) + 2\eta_{t+\frac{1}{2}} r(\boldsymbol{w}^\star) + \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2 + 7\eta_t \eta_{t+\frac{1}{2}} G^2 \quad . \end{aligned}$$

Proof We begin with a few simple properties of the forward-looking subgradient steps before proceeding with the core of the proof. Note first that for some $\boldsymbol{g}_t^f \in \partial f(\boldsymbol{w}_t)$ and $\boldsymbol{g}_{t+1}^r \in \partial r(\boldsymbol{w}_{t+1})$, we have as in Eq. (5)

$$\boldsymbol{w}_{t+1} - \boldsymbol{w}_t = -\eta_t \boldsymbol{g}_t^f - \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r \ . \tag{7}$$

The definition of a subgradient implies that for any $\boldsymbol{g}_{t+1}^r \in \partial r(\boldsymbol{w}_{t+1})$ (and similarly for any $\boldsymbol{g}_t^f \in \partial f(\boldsymbol{w}_t)$ with $f(\boldsymbol{w}_t)$ and $f(\boldsymbol{w}^\star)$)

$$r(\boldsymbol{w}^\star) \geq r(\boldsymbol{w}_{t+1}) + \left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}^\star - \boldsymbol{w}_{t+1} \right\rangle \quad \Rightarrow \quad -\left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}_{t+1} - \boldsymbol{w}^\star \right\rangle \leq r(\boldsymbol{w}^\star) - r(\boldsymbol{w}_{t+1}). \tag{8}$$

From the Cauchy-Shwartz Inequality and Eq. (7), we obtain

$$\begin{aligned}
\left\langle \boldsymbol{g}_{t+1}^r, (\boldsymbol{w}_{t+1} - \boldsymbol{w}_t) \right\rangle &= \left\langle \boldsymbol{g}_{t+1}^r, (-\eta_t \boldsymbol{g}_t^f - \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r) \right\rangle \\
&\leq \|\boldsymbol{g}_{t+1}^r\| \|\eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r + \eta_t \boldsymbol{g}_t^f\| \ \leq \ \eta_{t+\frac{1}{2}} \|\boldsymbol{g}_{t+1}^r\|^2 + \eta_t \|\boldsymbol{g}_{t+1}^r\| \|\boldsymbol{g}_t^f\| \\
&\leq \eta_{t+\frac{1}{2}} \left(A r(\boldsymbol{w}_{t+1}) + G^2 \right) + \eta_t \left(A \max\{f(\boldsymbol{w}_t), r(\boldsymbol{w}_{t+1})\} + G^2 \right) \ .
\end{aligned} \tag{9}$$

We now proceed to bound the difference between $\boldsymbol{w}^\star$ and $\boldsymbol{w}_{t+1}$, and using a telescoping sum we will eventually bound $f(\boldsymbol{w}_t) + r(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star) - r(\boldsymbol{w}^\star)$. First, we expand norm squared of the difference between $\boldsymbol{w}_t$ and $\boldsymbol{w}_{t+1}$,

$$\begin{aligned}
\|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2 &= \|\boldsymbol{w}_t - (\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r) - \boldsymbol{w}^\star\|^2 \\
&= \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - 2\left[\eta_t \left\langle \boldsymbol{g}_t^f, \boldsymbol{w}_t - \boldsymbol{w}^\star \right\rangle + \eta_{t+\frac{1}{2}} \left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}_t - \boldsymbol{w}^\star \right\rangle \right] + \|\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r\|^2 \\
&= \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - 2\eta_t \left\langle \boldsymbol{g}_t^f, \boldsymbol{w}_t - \boldsymbol{w}^\star \right\rangle + \|\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r\|^2 \\
&\quad -2\eta_{t+\frac{1}{2}} \left[\left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}_{t+1} - \boldsymbol{w}^\star \right\rangle - \left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}_{t+1} - \boldsymbol{w}_t \right\rangle \right] \ .
\end{aligned} \tag{10}$$

We can bound the third term above by noting that

$$\begin{aligned}
&\|\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r\|^2 \\
&\quad = \eta_t^2 \|\boldsymbol{g}_t^f\|^2 + 2\eta_t \eta_{t+\frac{1}{2}} \left\langle \boldsymbol{g}_t^f, \boldsymbol{g}_{t+1}^r \right\rangle + \eta_{t+\frac{1}{2}}^2 \|\boldsymbol{g}_{t+1}^r\|^2 \\
&\quad \leq \eta_t^2 A f(\boldsymbol{w}_t) + 2\eta_t \eta_{t+\frac{1}{2}} A \max\{f(\boldsymbol{w}_t), r(\boldsymbol{w}_{t+1})\} + \eta_{t+\frac{1}{2}}^2 A r(\boldsymbol{w}_{t+1}) + 4\eta_t^2 G^2 \ .
\end{aligned}$$

We now use Eq. (9) to bound the last term of Eq. (10) and the above bound on $\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r$ to get that

$$\begin{aligned}
&\|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2 \\
&\quad \leq \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - 2\eta_t \left\langle \boldsymbol{g}_t^f, \boldsymbol{w}_t - \boldsymbol{w}^\star \right\rangle - 2\eta_{t+\frac{1}{2}} \left\langle \boldsymbol{g}_{t+1}^r, \boldsymbol{w}_{t+1} - \boldsymbol{w}^\star \right\rangle + \|\eta_t \boldsymbol{g}_t^f + \eta_{t+\frac{1}{2}} \boldsymbol{g}_{t+1}^r\|^2 \\
&\qquad + 2\eta_{t+\frac{1}{2}} \left(\eta_{t+\frac{1}{2}} A r(\boldsymbol{w}_{t+1}) + 2\eta_t A \max\{f(\boldsymbol{w}_t), r(\boldsymbol{w}_{t+1})\} + 2\eta_t G^2 \right) \\
&\quad \leq \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 + 2\eta_t \left(f(\boldsymbol{w}^\star) - f(\boldsymbol{w}_t) \right) + 2\eta_{t+\frac{1}{2}} \left(r(\boldsymbol{w}^\star) - r(\boldsymbol{w}_t) \right) + 7\eta_t^2 G^2 \\
&\qquad + \eta_t^2 A f(\boldsymbol{w}_t) + 3\eta_t \eta_{t+\frac{1}{2}} A \max\{f(\boldsymbol{w}_t), r(\boldsymbol{w}_t)\} + 2\eta_{t+\frac{1}{2}}^2 A r(\boldsymbol{w}_{t+1}) \qquad (11) \\
&\quad \leq \|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 + 7\eta_t^2 G^2 \\
&\qquad + 2\eta_t \left(f(\boldsymbol{w}^\star) - (1 - c\eta_t A) f(\boldsymbol{w}_t) \right) + 2\eta_{t+\frac{1}{2}} \left(r(\boldsymbol{w}^\star) - (1 - c\eta_{t+\frac{1}{2}} A) r(\boldsymbol{w}_{t+1}) \right) \ . \qquad (12)
\end{aligned}$$

To obtain Eq. (11) we used the standard convexity bounds established earlier in Eq. (8). The final bound given by Eq. (12) is due to the fact that $3A\eta_t \eta_{t+\frac{1}{2}} \leq 6A\eta_t^2$ and that for any $a, b \geq 0$, $\max\{a, b\} \leq a + b$. Rearranging

the terms $f(\cdot)$ and $r(\cdot)$ yields the desired inequality. ■

The lemma allows a proof of the following theorem, which constitutes the basis for deriving concrete convergence results for FOBOS. It is also demonstrates the ease of proving convergence results based on the lemma and the forward looking property.

Theorem 2 *Assume the following hold: (i) the norm of any subgradient from ∂f and the norm of any subgradient from ∂r are bounded as in Eq. (6), (ii) the norm of $\boldsymbol{w}^\star$ is less than or equal to D, (iii) $r(\mathbf{0}) = 0$, and (iv) $\frac{1}{2}\eta_t \leq \eta_{t+1} \leq \eta_t$. Then for a constant $c \leq 4$ with $\boldsymbol{w}_1 = \mathbf{0}$ and $\eta_{t+\frac{1}{2}} = \eta_{t+1}$,*

$$\sum_{t=1}^{T} \left[\eta_t \left((1 - cA\eta_t) f(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star)\right) + \eta_t \left((1 - cA\eta_t) r(\boldsymbol{w}_t) - r(\boldsymbol{w}^\star)\right)\right] \leq D^2 + 7G^2 \sum_{t=1}^{T} \eta_t^2 \ .$$

Proof Rearranging the $f(\boldsymbol{w}^\star)$ and $r(\boldsymbol{w}^\star)$ terms from the bound in Lemma 1, we sum the loss terms over t from 1 through T and get a canceling telescoping sum:

$$\begin{aligned}
&\sum_{t=1}^{T} \left[\eta_t \left((1 - cA\eta_t) f(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star)\right) + \eta_{t+1} \left((1 - cA\eta_{t+1}) r(\boldsymbol{w}_{t+1}) - r(\boldsymbol{w}^\star)\right)\right] \\
&\quad \leq \quad \|\boldsymbol{w}_1 - \boldsymbol{w}^\star\|^2 - \|\boldsymbol{w}_{T+1} - \boldsymbol{w}^\star\|^2 + 7G^2 \sum_{t=1}^{T} \eta_t^2 \ \leq \ \|\boldsymbol{w}_1 - \boldsymbol{w}^\star\|^2 + 7G^2 \sum_{t=1}^{T} \eta_t^2 \ . \qquad (13)
\end{aligned}$$

We now bound the time-shifted $r(\boldsymbol{w}_{t+1})$ terms by noting that

$$\begin{aligned}
&\sum_{t=1}^{T} \eta_{t+1} \left((1 - cA\eta_{t+1}) r(\boldsymbol{w}_{t+1}) - r(\boldsymbol{w}^\star)\right) \\
&\quad = \quad \sum_{t=1}^{T} \eta_t \left((1 - cA\eta_t) r(\boldsymbol{w}_t) - r(\boldsymbol{w}^\star)\right) + \eta_{T+1} \left((1 - cA\eta_{T+1}) r(\boldsymbol{w}_{t+1}) - r(\boldsymbol{w}^\star)\right) + \eta_1 r(\boldsymbol{w}^\star) \\
&\quad \geq \quad \sum_{t=1}^{T} \eta_t \left((1 - cA\eta_t) r(\boldsymbol{w}_{t+1}) - r(\boldsymbol{w}^\star)\right) + r(\boldsymbol{w}^\star)(\eta_1 - \eta_{T+1}) \\
&\quad \geq \quad \sum_{t=1}^{T} \eta_t \left((1 - cA\eta_t) r(\boldsymbol{w}_t) - r(\boldsymbol{w}^\star)\right) \ . \qquad (14)
\end{aligned}$$

Finally, we use the fact that $\|\boldsymbol{w}_1 - \boldsymbol{w}^\star\| = \|\boldsymbol{w}^\star\| \leq D$, along with with Eq. (13)) and Eq. (14) to get the desired bound. ■

In the remainder of this section, we present a few corollaries that are consequences of the theorem. The first corollary underscores that the rate of convergence in general is approximately $1/\varepsilon^2$, or, equivalently, $1/\sqrt{T}$.

Corollary 3 (Fixed step rate) *Assume that the conditions of Thm. 2 hold and that we run* FOBOS *for a predefined T iterations with $\eta_t = \frac{D}{\sqrt{7T}G}$ and that $(1 - cA\frac{D}{\sqrt{7T}G}) > 0$. Then*

$$\min_{t \in \{1,\ldots,T\}} f(\boldsymbol{w}_t) + r(\boldsymbol{w}_t) \leq \frac{1}{T} \sum_{t=1}^{T} f(\boldsymbol{w}_t) + r(\boldsymbol{w}_t) \leq \frac{3DG}{\sqrt{T}\left(1 - \frac{cAD}{G\sqrt{7T}}\right)} + \frac{f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star)}{1 - \frac{cAD}{G\sqrt{7T}}} .$$

Proof Since we have $\eta_t = \eta$ for all t, the bound on the convergence rate from Thm. 2 becomes

$$\begin{aligned}&\eta(1-cA\eta)T \min_{t\in\{1,\ldots,T\}} [f(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f(\boldsymbol{w}^\star)-r(\boldsymbol{w}^\star)]\\ &\leq \quad \eta(1-cA\eta)\sum_{t=1}^{T}[f(\boldsymbol{w}_t)-f(\boldsymbol{w}^\star)]+[r(\boldsymbol{w}_t)-r(\boldsymbol{w}^\star)] \;\leq\; D^2+7G^2T\eta^2 \;\;.\end{aligned}$$

Plugging in the specified value for η gives the bound. ■

Another direct consequence of Thm. 2 is convergence of the minimum over t when running FOBOS with $\eta_t \propto 1/\sqrt{t}$ or with non-summable step sizes decreasing to zero.

Corollary 4 (Convergence of decreasing step sizes) *Assume that the conditions of Thm. 2 hold and the step sizes η_t satisfy $\eta_t \to 0$ and $\sum_{t=1}^{\infty}\eta_t = \infty$. Then*

$$\liminf_{t\to\infty} f(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-(f(\boldsymbol{w}^\star)+r(\boldsymbol{w}^\star)) = 0.$$

Finally, when f and r are Lipschitz with Lipschitz constant G, we immediately have

Corollary 5 *In addition to the conditions of Thm. 2, assume that the norm of any subgradient from ∂f and the norm of any subgradient from ∂r are bounded by G. Then*

$$\min_{t\in\{1,\ldots,T\}} (f(\boldsymbol{w}_t)+r(\boldsymbol{w}_t))-(f(\boldsymbol{w}^\star)+r(\boldsymbol{w}^\star)) \leq \frac{D^2+7G^2\sum_{t=1}^{T}\eta_t^2}{2\sum_{t=1}^{T}\eta_t}. \tag{15}$$

Bounds of the above form, where we obtain convergence for one of the points in the sequence $\boldsymbol{w}_1,\ldots,\boldsymbol{w}_T$ rather than the last point $\boldsymbol{w}_T$, are standard in subgradient optimization. The main reason that this weaker result occurs is due to the fact that we cannot guarantee a strict descent direction when using arbitrary subgradients (see, for example, Theorem 3.2.2 from Nesterov 2004). Another consequence of using non-differentiable functions means that analyses such as those carried out by Tseng (2000) and Chen and Rockafellar (1997) are difficult to apply, as the stronger rates rely on the existence and Lipschitz continuity of $\nabla f(\boldsymbol{w})$. However, it is possible to show linear convergence rates under suitable smoothness and strong convexity assumptions. When $\nabla f(\boldsymbol{w})$ is Lipschitz continuous, a more detailed analysis yields convergence rates of $1/\varepsilon$ (namely, $1/T$ in terms of number of iterations needed to be ε close to the optimum). A more complicated algorithm related to Nesterov's "estimate functions" (Nesterov, 2004) leads to $O(1/\sqrt{\varepsilon})$ convergence (Nesterov, 2007). For completeness, we give a simple proof of $1/T$ convergence in Appendix C. Finally, the above proof can be modified slightly to give convergence of the stochastic gradient method. In particular, we can replace $\boldsymbol{g}_t^f$ in the iterates of FOBOS with a stochastic estimate of the gradient $\tilde{\boldsymbol{g}}_t^f$, where $E[\tilde{\boldsymbol{g}}_t^f] \in \partial f(\boldsymbol{w}_t)$. We explore this approach in slightly more depth after performing a regret analysis for FOBOS below in Sec. 4 and describe stochastic convergence rates in Corollary 10.

We would like to make further comments on our proof of convergence for FOBOS and the assumptions underlying the proof. It is often not necessary to have a Lipschitz loss to guarantee boundedness of the subgradients of f and r, so in practice an assumption of bounded subgradients (as in Corollary 5 and in the sequel for online analysis) is not restrictive. If we know that an optimal $\boldsymbol{w}^\star$ lies in some compact set Ω and that $\Omega \subseteq \mathrm{dom}(f+r)$, then Theorem 24.7 of Rockafellar (1970) guarantees that ∂f and ∂r are bounded. The lingering question is thus whether we can guarantee that such a set Ω exists and that our iterates $\boldsymbol{w}_t$ remain in Ω. The following simple setting shows that ∂f and ∂r are indeed often bounded.

If $r(\boldsymbol{w})$ is a norm (possibly scaled) and f is lower bounded by 0, then we know that $r(\boldsymbol{w}^\star) \leq f(\boldsymbol{w}^\star)+r(\boldsymbol{w}^\star) \leq f(\boldsymbol{w}_1)+r(\boldsymbol{w}_1)$. Using standard bounds on norms, we get that for some $\gamma > 0$

$$\|\boldsymbol{w}^\star\|_\infty \leq \gamma r(\boldsymbol{w}^\star) \leq \gamma(f(\boldsymbol{w}_1)+r(\boldsymbol{w}_1)) = \gamma f(\boldsymbol{w}_1) \;\;,$$

where for the last inequality we used the assumption that $r(w_1) = 0$. Thus, we obtain that $w^\star$ lies in a hypercube. We can easily project onto this box by truncating elements of w_t lying outside it at any iteration without affecting the bounds in Eq. (12) or Eq. (15). This additional Euclidean projection Π_Ω to an arbitrary convex set Ω with $w^\star \in \Omega$ satisfies $\|\Pi_\Omega(w_{t+1}) - w^\star\| \leq \|w_{t+1} - w^\star\|$. Furthermore, so long as Ω is an ℓ_p-norm ball, we know that

$$r(\Pi_\Omega(w_{t+1})) \leq r(w_{t+1}) \ . \tag{16}$$

Thus, looking at Eq. (11), we notice that $r(w^\star) - r(w_{t+1}) \leq r(w^\star) - r(\Pi_\Omega(w_{t+1}))$ and the series of inequalities through Eq. (12) still hold (so long as $c\eta_{t+\frac{1}{2}}A \leq 1$, which it will if Ω is compact so that we can take $A = 0$). In general, so long as Eq. (16) holds and $w^\star \in \Omega$, we can project w_{t+1} into Ω without affecting convergence guarantees. This property proves to be helpful in the regret analysis below.

4. Regret Analysis of FOBOS

In this section we provide regret analysis for FOBOS in online settings. In an online learning problem, we are given a sequence of functions $f_t : \mathbb{R}^n \to \mathbb{R}$. The learning goal is for the sequence of predictions w_t to attain low regret when compared to a single optimal predictor $w^\star$. Formally, let $f_t(w)$ denote the loss suffered on the t^{th} input loss function when using a predictor w. The regret of an online algorithm which uses $w_1, \ldots, w_t, \ldots$ as its predictors w.r.t. a fixed predictor $w^\star$ while using a regularization function r is

$$R_{f+r}(T) = \sum_{t=1}^{T} \left[f_t(w_t) + r(w_t) - (f_t(w^\star) + r(w^\star))\right].$$

Ideally, we would like to achieve 0 regret to a stationary $w^\star$ for arbitrary length sequences.

To achieve an online bound for a sequence of convex functions f_t, we modify arguments of Zinkevich (2003). Using the bound from Lemma 1, we can readily state and prove a theorem on the online regret of FOBOS. It is possible to avoid the boundedness assumptions in the proof of the theorem (getting a bound similar to that of Theorem 2 but for regret), however, we do not find it significantly more interesting. Aside from its reliance on Lemma 1, this proof is quite similar to Zinkevich's, so we defer it to Appendix A.

Theorem 6 *Assume that $\|w_t - w^\star\| \leq D$ for all iterations and the norm of the subgradient sets ∂f_t and ∂r are bounded above by G. Let $c > 0$ an arbitrary scalar. Then, the regret bound of* FOBOS *with $\eta_t = c/\sqrt{t}$ satisfies*

$$R_{f+r}(T) \leq 2GD + \left(\frac{D^2}{2c} + 7G^2c\right)\sqrt{T} \ .$$

The following Corollary is immediate from Theorem 6.

Corollary 7 *Assume the conditions of Theorem 6 hold. Then, setting $\eta_t = \frac{D}{4G\sqrt{t}}$, the regret of* FOBOS *is*

$$R_{f+r}(T) \leq 2GD + 4GD\sqrt{T} \ .$$

We can also obtain a better regret bound for FOBOS when the sequence of loss functions $f_t(\cdot)$ or the function $r(\cdot)$ is strongly convex. As demonstrated by Hazan et al. (2006), with the projected gradient method and strongly convex functions, it is possible to achieve regret on the order of $O(\log T)$ by using the curvature of the sequence of functions f_t rather than simply using convexity and linearity as in Theorems 2 and 6. We can extend these results to FOBOS for the case in which $f_t(w) + r(w)$ is strongly convex, at least over the domain $\|w - w^\star\| \leq D$. For completeness, we recap a few definitions and provide the logarithmic regret bound for FOBOS. A function f is called H-strongly convex if

$$f(w) \geq f(w_t) + \langle \nabla f(w_t), w - w_t \rangle + \frac{H}{2}\|w - w_t\|^2.$$

Thus, if r and the sequence of functions f_t are strongly convex with constants $H_f \geq 0$ and $H_r \geq 0$, we have $H = H_f + H_r$ and H-strong convexity gives

$$f_t(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star) + r(\boldsymbol{w}_t) - r(\boldsymbol{w}^\star) \leq \left\langle \boldsymbol{g}_t^f + \boldsymbol{g}_t^r, \boldsymbol{w}_t - \boldsymbol{w}^\star \right\rangle - \frac{H}{2}\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 \ . \tag{17}$$

We do not need to assume that both f_t and r are strongly convex. We only need assume that at least one of them attains a positive strong convexity constant. For example, if $r(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2$, then $H \geq \lambda$ so long as the functions f_t are convex. With Eq. (17) in mind, we can readily build on Hazan et al. (2006) and prove a stronger regret bound for the online learning case. The proof is similar to that of Hazan et al., so we defer it also to Appendix A.

Theorem 8 *Assume as in Theorem 6 that $\|\boldsymbol{w}_t - \boldsymbol{w}^\star\| \leq D$ and that ∂f_t and ∂r are bounded above by G. Assume further that $f_t + r$ is H-strongly convex for all t. Then, when using step sizes $\eta_t = 1/Ht$, the regret of* FOBOS *is*

$$R_{f+r}(T) = O\left(\frac{G^2}{H}\log T\right).$$

We now provide an easy lemma showing that for Lipschitz losses with ℓ_2^2 regularization, the boundedness assumptions above hold. This, for example, includes the interesting case of support vector machines. The proof is not difficult but relies tacitly on a later result, so we leave it to Appendix A.

Lemma 9 *Let the functions f_t be G-Lipschitz so that $\|\partial f_t(\boldsymbol{w})\| \leq G$. Let $r(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2$. Then $\|\boldsymbol{w}^\star\| \leq G/\lambda$ and the iterates $\boldsymbol{w}_t$ generated by* FOBOS *satisfy $\|\boldsymbol{w}_t\| \leq G/\lambda$.*

Using the regret analysis for online learning, we are able to return to learning in a batch setting and give stochastic convergence rates for FOBOS. We build on results of Shalev-Shwartz et al. (2007) and assume as in Sec. 3 that we are minimizing $f(\boldsymbol{w}) + r(\boldsymbol{w})$. Indeed, suppose that on each step of FOBOS, we choose instead of some $\boldsymbol{g}_t^f \in \partial f(\boldsymbol{w}_t)$ a stochastic estimate of the gradient $\tilde{\boldsymbol{g}}_t^f$ where $E[\tilde{\boldsymbol{g}}_t^f] \in \partial f(\boldsymbol{w}_t)$. We assume that we still use the true r (which is generally easy, as it is simply the regularization function). It is straightforward to use theorems 6 and 8 above as in the derivation of theorems 2 and 3 from Shalev-Shwartz et al. (2007) to derive the following corollary on the expected convergence rate of FOBOS as well as a guarantee of convergence with high probability.

Corollary 10 *Assume that the conditions on ∂f, ∂r, and $\boldsymbol{w}^\star$ hold as in the previous theorems and let* FOBOS *be run for T iterations. Let s be an integer chosen uniformly at random from $\{1, \ldots, T\}$. If $\eta_t = \frac{D}{4G\sqrt{t}}$, then*

$$E_s[f(\boldsymbol{w}_s) + r(\boldsymbol{w}_s)] \leq f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star) + \frac{2GD + 4GD\sqrt{T}}{T}.$$

With probability at least $1 - \delta$,

$$f(\boldsymbol{w}_s) + r(\boldsymbol{w}_s) \leq f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star) + \frac{2GD + 4GD\sqrt{T}}{\delta T}.$$

If $f + r$ is H-strongly convex and we choose $\eta_t \propto 1/t$, we have

$$E_s[f(\boldsymbol{w}_s) + r(\boldsymbol{w}_s)] = f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star) + O\left(\frac{G^2 \log T}{HT}\right)$$

and with probability at least $1 - \delta$,

$$f(\boldsymbol{w}_s) + r(\boldsymbol{w}_s) = f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star) + O\left(\frac{G^2 \log T}{H\delta T}\right).$$

5. Derived Algorithms

In this section we derive a few variants of FOBOS by considering different regularization functions. The emphasis of the section is on non-differentiable regularization functions, such as the ℓ_1 norm of $\boldsymbol{w}$, which lead to sparse solutions. We also derive simple extensions to mixed-norm regularization (Zhao et al., 2006) that build on the first part of this section.

First, we make a few changes to notation. To simplify our derivations, we denote by $\boldsymbol{v}$ the vector $\boldsymbol{w}_{t+\frac{1}{2}} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f$ and let $\tilde{\lambda}$ denote $\eta_{t+\frac{1}{2}} \cdot \lambda$. Using this newly introduced notation the problem given in Eq. (3) can be rewritten as

$$\operatorname*{minimize}_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{v}\|^2 + \tilde{\lambda} r(\boldsymbol{w}). \tag{18}$$

Let $[z]_+$ denote $\max\{0,z\}$. For completeness, we provide full derivations for all the regularization functions we consider, but for brevity we do not state formally well established technical lemmas. We note that many of the following results were given tacitly by Wright et al. (2009).

5.1 FOBOS with ℓ_1 Regularization

The update obtained by choosing $r(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_1$ is simple and intuitive. First note that the objective is decomposable as we can rewrite Eq. (18) as

$$\operatorname*{minimize}_{\boldsymbol{w}} \sum_{j=1}^{n} \left(\frac{1}{2}(w_j - v_j)^2 + \tilde{\lambda}|w_j|\right).$$

Let us focus on a single coordinate of $\boldsymbol{w}$ and for brevity omit the index j. Let $w^\star$ denote the minimizer of $\frac{1}{2}(w-v)^2 + \tilde{\lambda}|w|$. Clearly, $w^\star \cdot v \geq 0$. If it were not, then we would have $w^\star \cdot v < 0$, however $\frac{1}{2}v^2 < \frac{1}{2}v^2 - w^\star \cdot v + \frac{1}{2}(w^\star)^2 < \frac{1}{2}(v-w^\star)^2 + \tilde{\lambda}|w^\star|$, contradicting the supposed optimality of $w^\star$. The symmetry of the objective in v also shows us that we can assume that $v \geq 0$; we therefore need to minimize $\frac{1}{2}(w-v)^2 + \tilde{\lambda}w$ subject to the constraint that $w \geq 0$. Introducing a Lagrange multiplier $\beta \geq 0$ for the constraint, we have the Lagrangian $\frac{1}{2}(w-v)^2 + \tilde{\lambda}w - \beta w$. By taking the derivative of the Langrangian with respect to w and setting the result to zero, we get that the optimal solution is $w^\star = v - \tilde{\lambda} + \beta$. If $w^\star > 0$, then from the complimentary slackness condition that the optimal pair of $w^\star$ and β must have $w^\star\beta = 0$ (Boyd and Vandenberghe, 2004) we must have $\beta = 0$, and therefore $w^\star = v - \tilde{\lambda}$. If $v < \tilde{\lambda}$, then $v - \tilde{\lambda} < 0$, so we must have $\beta > 0$ and again by complimentary slackness, $w^\star = 0$. The case when $v \leq 0$ is analogous and amounts to simply flipping signs. Summarizing and expanding notation, the components of the optimal solution $\boldsymbol{w}^\star = \boldsymbol{w}_{t+1}$ are computed from $\boldsymbol{w}_{t+\frac{1}{2}}$ as

$$w_{t+1,j} = \operatorname{sign}\left(w_{t+\frac{1}{2},j}\right)\left[|w_{t+\frac{1}{2},j}| - \tilde{\lambda}\right]_+ = \operatorname{sign}\left(w_{t,j} - \eta_t g_{t,j}^f\right)\left[\left|w_{t,j} - \eta_t g_{t,j}^f\right| - \eta_{t+\frac{1}{2}} \cdot \lambda\right]_+ . \tag{19}$$

Note that this update can lead to sparse solutions. Whenever the absolute value of a component of $\boldsymbol{w}_{t+\frac{1}{2}}$ is smaller than $\tilde{\lambda}$, the corresponding component in $\boldsymbol{w}_{t+1}$ is set to zero. Thus, Eq. (19) gives a simple online and offline method for minimizing a convex f with ℓ_1 regularization.

Such soft-thresholding operations are common in the statistics literature and have been used for some time (Donoho, 1995; Daubechies et al., 2004). Langford et al. (2008) recently proposed and analyzed the same update, terming it the "truncated gradient." The analysis presented here is different from the analysis in the aforementioned paper as it stems from a more general framework. Indeed, as illustrated in this section, the derivation and method is also applicable to a wide variety of regularization functions. Nevertheless, both analyses merit consideration as they shed light from different angles on the problem of learning sparse models using gradients, stochastic gradients, or online methods. This update can also be implemented very efficiently when the support of $\boldsymbol{g}_t^f$ is small (Langford et al., 2008), but we defer details to Sec. 6, where we give a unified view that facilitates efficient implementation for all the norm regularization functions we discuss.

5.2 FOBOS with ℓ_2^2 Regularization

When $r(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2$, we obtain a very simple optimization problem,

$$\underset{\boldsymbol{w}}{\text{minimize}}\ \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{v}\|^2 + \frac{1}{2}\tilde{\lambda}\|\boldsymbol{w}\|^2,$$

where for conciseness of the solution we replace $\tilde{\lambda}$ with $\frac{1}{2}\tilde{\lambda}$. Differentiating the above objective and setting the result equal to zero, we have $\boldsymbol{w}^\star - \boldsymbol{v} + \tilde{\lambda}\boldsymbol{w}^\star = 0$, which, using the original notation, yields the update

$$\boldsymbol{w}_{t+1} = \frac{\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f}{1+\tilde{\lambda}}. \tag{20}$$

Informally, the update simply shrinks $\boldsymbol{w}_{t+1}$ back toward the origin after each gradient-descent step. In Sec. 7 we briefly compare the resulting FOBOS update to modern stochastic gradient techniques and show that the FOBOS update exhibits similar empirical behavior.

5.3 FOBOS with ℓ_2 Regularization

A lesser used regularization function is the ℓ_2 norm of the weight vector. By setting $r(\boldsymbol{w}) = \tilde{\lambda}\|\boldsymbol{w}\|$ we obtain the following problem,

$$\underset{\boldsymbol{w}}{\text{minimize}}\ \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{v}\|^2 + \tilde{\lambda}\|\boldsymbol{w}\|. \tag{21}$$

The solution of Eq. (21) must be in the direction of $\boldsymbol{v}$ and takes the form $\boldsymbol{w}^\star = s\boldsymbol{v}$ where $s \geq 0$. To show that this is indeed the form of the solution, let us assume for the sake of contradiction that $\boldsymbol{w}^\star = s\boldsymbol{v} + \boldsymbol{u}$ where $\boldsymbol{u}$ is in the null space of $\boldsymbol{v}$ (if $\boldsymbol{u}$ has any components parallel to $\boldsymbol{v}$, we can add those to $s\boldsymbol{v}$ and obtain an orthogonal $\boldsymbol{u}'$) and s may be negative. Since $\boldsymbol{u}$ is orthogonal to $\boldsymbol{v}$, the objective function can be expressed in terms of s, $\boldsymbol{v}$, and $\boldsymbol{u}$ as

$$\frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \frac{1}{2}\|\boldsymbol{u}\|^2 + \tilde{\lambda}(\|\boldsymbol{v}\| + \|\boldsymbol{u}\|) \geq \frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \tilde{\lambda}\|\boldsymbol{v}\|.$$

Thus, $\boldsymbol{u}$ must be equal to the zero vector, $\boldsymbol{u} = \boldsymbol{0}$, and we can write the optimization problem as

$$\underset{s}{\text{minimize}}\ \frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \tilde{\lambda}s\|\boldsymbol{v}\| \ .$$

Next note that a negative value for s cannot constitute the optimal solution. Indeed, if $s < 0$, then

$$\frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \tilde{\lambda}s\|\boldsymbol{v}\| \ < \ \frac{1}{2}\|\boldsymbol{v}\|^2 \ .$$

This implies that by setting $s = 0$ we can obtain a lower objective function, and this precludes a negative value for s as an optimal solution. We therefore end again with a constrained scalar optimization problem, $\text{minimize}_{s\geq 0}\ \frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \tilde{\lambda}s\|\boldsymbol{v}\|$. The Lagrangian of this problem is

$$\frac{1}{2}(1-s)^2\|\boldsymbol{v}\|^2 + \tilde{\lambda}s\|\boldsymbol{v}\| - \beta s \ ,$$

where $\beta \geq 0$. By taking the derivative of the Langrangian with respect to s and setting the result to zero, we get that $(s-1)\|\boldsymbol{v}\|^2 + \tilde{\lambda}\|\boldsymbol{v}\| - \beta = 0$ which gives the following closed form solution: $s = 1 - \tilde{\lambda}/\|\boldsymbol{v}\| + \beta/\|\boldsymbol{v}\|^2$. Whenever $s > 0$ then the complimentary slackness conditions imply that $\beta = 0$ and s can be further simplified and written as $s = 1 - \tilde{\lambda}/\|\boldsymbol{v}\|$. The last expression is positive iff $\|\boldsymbol{v}\| > \tilde{\lambda}$. If $\|\boldsymbol{v}\| < \tilde{\lambda}$, then β must be positive and complimentary slackness implies that $s = 0$.

Summarizing, the second step of the FOBOS update with ℓ_2 regularization amounts to

$$\boldsymbol{w}_{t+1} = \left[1 - \frac{\tilde{\lambda}}{\|\boldsymbol{w}_{t+\frac{1}{2}}\|}\right]_+ \boldsymbol{w}_{t+\frac{1}{2}} \ = \ \left[1 - \frac{\tilde{\lambda}}{\|\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f\|}\right]_+ (\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f) \ .$$

Thus, the ℓ_2 regularization results in a zero weight vector under the condition that $\|w_t - \eta_t g_t^f\| \leq \tilde{\lambda}$. This condition is rather more stringent for sparsity than the condition for ℓ_1 (where a weight is sparse based only on its value, while here, sparsity happens only if the entire weight vector has ℓ_2-norm less than $\tilde{\lambda}$), so it is unlikely to hold in high dimensions. However, it does constitute a very important building block when using a mixed ℓ_1/ℓ_2-norm as the regularization function, as we show in the sequel.

5.4 FOBOS with ℓ_∞ Regularization

We now turn to a less explored regularization function, the ℓ_∞ norm of w. This form of regularization is not capable of providing strong guarantees against over-fitting on its own as many of the weights of w may not be penalized. However, there are settings in which it is desirable to consider blocks of variables as a group, such as ℓ_1/ℓ_∞ regularization. We continue to defer the discussion on mixing different norms and focus merely on the ℓ_∞ norm as it serves as a building block. That is, we are interested in obtaining an efficient solution to the following problem,

$$\operatorname*{minimize}_{w} \frac{1}{2}\|w - v\|^2 + \tilde{\lambda}\|w\|_\infty \quad . \tag{22}$$

It is possible to derive an efficient algorithm for finding the minimizer of Eq. (22) using properties of the subgradient set of $\|w\|_\infty$. However, a solution to the dual form of Eq. (22) is well established. Recalling that the conjugate of the quadratic function is a quadratic function and the conjugate of the ℓ_∞ norm is the ℓ_1 barrier function, we immediately obtain that the dual of the problem given by Eq. (22) is

$$\operatorname*{maximize}_{\alpha} -\frac{1}{2}\|\alpha - v\|_2^2 \;\; \text{s.t.} \;\; \|\alpha\|_1 \leq \tilde{\lambda} \;\; . \tag{23}$$

Moreover, the vector of dual variables α satisfies the relation $\alpha = v - w$. Thus, by solving the dual form in Eq. (23) we can readily obtain a solution for Eq. (22). The problem defined by Eq. (23) is equivalent to performing Euclidean projection onto the ℓ_1 ball and has been studied by numerous authors. The solution that we overview here is based on recent work of Duchi et al. (2008). The maximizer of Eq. (23), denoted $\alpha^\star$, is of the form

$$\alpha_j^\star = \operatorname{sign}(v_j)\left[|v_j| - \theta\right]_+ \quad , \tag{24}$$

where θ is a non-negative scalar. Duchi et al. (2008) describe a linear time algorithm for finding θ. We thus skip the analysis of the algorithm and focus on its core properties that affect the solution of the original problem Eq. (22). To find θ we need to locate a pivot element in v, denoted by the ρ^{th} order statistic $v_{(\rho)}$ (where $v_{(1)}$ is the largest magnitude entry of v), with the following property, $v_{(\rho)}$ is the smallest magnitude element in v such that

$$\sum_{j:|v_j|>|v_{(\rho)}|} \left(|v_j| - |v_{(\rho)}|\right) < \tilde{\lambda} \;\; .$$

If all the elements in v (assuming that we have added an extra 0 element to handle the smallest entry of v) satisfy the above requirement then the optimal choice for θ is 0. Otherwise,

$$\theta = \frac{1}{\rho}\left(\sum_{j:|v_j|>|v_{(\rho)}|} |v_j| - \tilde{\lambda}\right) \quad .$$

Thus, the optimal choice of θ is zero when $\sum_{j=1}^n |v_j| \leq \tilde{\lambda}$ (this is, not coincidentally, simply the subgradient condition for optimality of the zero vector in Eq. (22)).

Using the linear relationship $\alpha = v - w \Rightarrow w = v - \alpha$ along with the solution of the dual problem as given by Eq. (24), we obtain the following solution for Eq. (22),

$$w_{t+1,j} = \operatorname{sign}\left(w_{t+\frac{1}{2},j}\right)\min\left\{\left|w_{t+\frac{1}{2},j}\right| \, , \, \theta\right\} \; . \tag{25}$$

As stated above, $\theta = 0$ iff $\|\boldsymbol{w}_{t+\frac{1}{2}}\|_1 \leq \tilde{\lambda}$ and otherwise $\theta > 0$ and can be found in $O(n)$ steps. In words, the components of the vector $\boldsymbol{w}_{t+1}$ are the result of capping all of the components of $\boldsymbol{w}_t$ at θ where θ is zero when the 1-norm of $\boldsymbol{w}_{t+\frac{1}{2}}$ is smaller than $\tilde{\lambda}$. Interestingly, this property shares a duality resemblance with the ℓ_2-regularized update, which results in a zero weight vector when the 2-norm (which is self-dual) of $\boldsymbol{v}$ is less than $\tilde{\lambda}$. We can exploit these properties in the context of mixed-norm regularization to achieve sparse solutions for complex prediction problems, which we describe in the sequel and for which we present preliminary results in Sec. 7. Before doing so, we present one more norm-related regularization.

5.5 FOBOS with Berhu Regularization:

We now consider a regularization function which is a hybrid of the ℓ_1 and ℓ_2 norms. Similar to ℓ_1 regularization, Berhu (for inverse Huber) regularization results in sparse solutions, but its hybridization with ℓ_2^2 regularization prevents the weights from being excessively large. Berhu regularization (Owen, 2006) is defined as

$$r(\boldsymbol{w}) = \lambda \sum_{j=1}^{n} b(w_j) = \lambda \sum_{j=1}^{n} \left[|w_j| \, [\![|w_j| \leq \gamma]\!] + \frac{w_j^2 + \gamma^2}{2\gamma} [\![|w_j| > \gamma]\!] \right].$$

In the above, $[\![\cdot]\!]$ is 1 if its argument is true and is 0 otherwise. The positive scalar γ controls the value for which the Berhu regularization switches from ℓ_1 mode to ℓ_2 mode. Formally, when $w_j \in [-\gamma, \gamma]$, $r(\boldsymbol{w})$ behaves as ℓ_1, and for w_j outside this region, the Berhu penalty behaves as ℓ_2^2. The Berhu penalty is convex and differentiable except at 0, where it has the same subdifferential set as $\lambda \|\boldsymbol{w}\|_1$.

To find a closed form update from $\boldsymbol{w}_{t+\frac{1}{2}}$ to $\boldsymbol{w}_{t+1}$, we minimize in each variable $\frac{1}{2}(w - v) + \tilde{\lambda} b(w)$; the derivation is fairly standard but technical and is provided in Appendix B. The end result is aesthetic and captures the ℓ_1 and ℓ_2 regions of the Berhu penalty,

$$w_{t+1,j} = \begin{cases} 0 & \left| w_{t+\frac{1}{2},j} \right| \leq \tilde{\lambda} \\ \operatorname{sign}(w_{t+\frac{1}{2},j}) \left[|w_{t+\frac{1}{2},j}| - \tilde{\lambda} \right] & \tilde{\lambda} < |w_{t+\frac{1}{2},j}| \leq \tilde{\lambda} + \gamma \\ \frac{w_{t+\frac{1}{2},j}}{1 + \tilde{\lambda}/\gamma} & \gamma + \tilde{\lambda} < \left| w_{t+\frac{1}{2},j} \right| \end{cases}. \tag{26}$$

Indeed, as Eq. (26) indicates, the update takes one of two forms, depending on the magnitude of the coordinate of $w_{t+\frac{1}{2},j}$. If $|w_{t+\frac{1}{2},j}|$ is greater than $\gamma + \tilde{\lambda}$, the update is identical to the update for ℓ_2^2-regularization of Eq. (20), while if the value is no larger than $\gamma + \tilde{\lambda}$, the resulting update is equivalent to the update for ℓ_1-regularization of Eq. (19).

5.6 Extension to Mixed Norms

We saw above that when using either the ℓ_2 or the ℓ_∞ norm as the regularization function, we obtain an all zeros vector if $||\boldsymbol{w}_{t+\frac{1}{2}}||_2 \leq \tilde{\lambda}$ or $||\boldsymbol{w}_{t+\frac{1}{2}}||_1 \leq \tilde{\lambda}$, respectively. The zero vector does not carry any generalization properties, which surfaces a concern regarding the usability of the these norms as a form of regularization. This seemingly problematic phenomenon can, however, be useful in the setting we discuss now. In many applications, the set of weights can be grouped into subsets where each subset of weights should be dealt with uniformly. For example, in multiclass categorization problems each class r may be associated with a different weight vector $\boldsymbol{w}^r$. The prediction for an instance $\boldsymbol{x}$ is a vector $\langle \boldsymbol{w}^1, \boldsymbol{x} \rangle, \ldots, \langle \boldsymbol{w}^k, \boldsymbol{x} \rangle$ where k is the number of different classes. The predicted class is the index of the inner-product attaining the largest of the k values, $\operatorname{argmax}_j \langle \boldsymbol{w}^j, \boldsymbol{x} \rangle$. Since all the weight vectors operate over the same instance space, in order to achieve a sparse solution, it may be beneficial to tie the weights corresponding to the same input feature. That is, we would like to employ a regularization function that tends to zero the row of weights $w_j^1, \ldots, w_j^k$ *simultaneously*. In these circumstances, the nullification of the entire weight vector by ℓ_2 and ℓ_∞ regularization becomes a powerful tool.

Formally, let W represent a $n \times k$ matrix where the j^{th} column of the matrix is the weight vector $\boldsymbol{w}^j$ associated with class j. Thus, the i^{th} row corresponds to the weight of the i^{th} feature with respect to all classes. The mixed ℓ_r/ℓ_s-norm (Obozinski et al., 2007) of W, denoted $\|W\|_{\ell_r/\ell_s}$, is obtained by computing the ℓ_s-norm of each row of W and then applying the ℓ_r-norm to the resulting n dimensional vector, for instance, $\|W\|_{\ell_1/\ell_\infty} = \sum_{j=1}^{n} \max_j |W_{i,j}|$. Thus, in a mixed-norm regularized optimization problem (such as multiclass learning), we seek the minimizer of the objective function

$$f(W) + \lambda \|W\|_{\ell_r/\ell_s}.$$

Given the specific variants of various norms described above, the FOBOS update for the ℓ_1/ℓ_∞ and the ℓ_1/ℓ_2 mixed-norms is readily available. Let $\bar{\boldsymbol{w}}^r$ denote the r^{th} row of W. Analogously to the standard norm-based regularization, we let $V = W_{t+\frac{1}{2}}$ be a shorthand for the result of the gradient step. For the ℓ_1, ℓ_p mixed-norm, where $p = 2$ or $p = \infty$, we need to solve the problem

$$\operatorname*{minimize}_{W} \frac{1}{2}\|W - V\|_{\text{Fr}}^2 + \tilde{\lambda}\|W\|_{\ell_1/\ell_2} \equiv \operatorname*{minimize}_{\bar{\boldsymbol{w}}^1,\ldots,\bar{\boldsymbol{w}}^k} \sum_{i=1}^{n} \left(\frac{1}{2}\left\|\bar{\boldsymbol{w}}^i - \bar{\boldsymbol{v}}^i\right\|_2^2 + \tilde{\lambda}\left\|\bar{\boldsymbol{w}}^i\right\|_p \right), \tag{27}$$

where $\bar{\boldsymbol{v}}^i$ is the i^{th} row of V. It is immediate to see that the problem given in Eq. (27) is decomposable into n separate problems of dimension k, each of which can be solved by the procedures described in the prequel. The end result of solving these types of mixed-norm problems is a sparse matrix with numerous zero rows. We demonstrate the merits of FOBOS with mixed-norms in Sec. 7.

6. Efficient Implementation in High Dimensions

In many settings, especially online learning, the weight vector $\boldsymbol{w}_t$ and the gradients $\boldsymbol{g}_t^f$ reside in a very high-dimensional space, but only a relatively small number of the components of $\boldsymbol{g}_t^f$ are non-zero. Such settings are prevalent, for instance, in text-based applications: in text categorization, the full dimension corresponds to the dictionary or set of tokens that is being employed while each gradient is typically computed from a single or a few documents, each of which contains words and bigrams constituting only a small subset of the full dictionary. The need to cope with gradient sparsity becomes further pronounced in mixed-norm problems, as a single component of the gradient may correspond to an entire row of W. Updating the entire matrix because a few entries of $\boldsymbol{g}_t^f$ are non-zero is clearly undesirable. Thus, we would like to extend our methods to cope efficiently with gradient sparsity. For concreteness, we focus in this section on the efficient implementation of ℓ_1, ℓ_2, and ℓ_∞ regularization, recognizing that the extension to mixed-norms (as in the previous section) is straightforward. The upshot of following proposition is that when $\boldsymbol{g}_t^f$ is sparse, we can use lazy evaluation of the weight vectors and defer to later rounds the update of components of $\boldsymbol{w}_t$ whose corresponding gradient entries are zero. We detail this after the proposition, which is technical so the interested reader may skip the proof to see the simple algorithms for lazy evaluation.

Proposition 11 *Let $\boldsymbol{w}_T$ be the end result of solving a succession of T self-similar optimization problems for $t = 1, \ldots, T$,*

$$\mathcal{P}.1: \quad \boldsymbol{w}_t = \operatorname*{argmin}_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_{t-1}\|^2 + \lambda_t \|\boldsymbol{w}\|_q \ . \tag{28}$$

Let $\boldsymbol{w}^\star$ be the optimal solution of the following optimization problem:

$$\mathcal{P}.2: \quad \boldsymbol{w}^\star = \operatorname*{argmin}_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_0\|^2 + \left(\sum_{t=1}^{T} \lambda_t\right) \|\boldsymbol{w}\|_q \ .$$

Then for $q \in \{1, 2, \infty\}$ the vectors $\boldsymbol{w}_T$ and $\boldsymbol{w}^\star$ are identical.

Proof It suffices to show that the proposition is correct for $T = 2$ and then use an inductive argument, because the proposition trivially holds for $T = 1$. We provide here a direct proof for each norm separately by examining the updates we derived in Sec. 5 and showing that $\boldsymbol{w}_2 = \boldsymbol{w}^\star$.

Note that the objective functions are separable for $q = 1$. Therefore, for ℓ_1-regularization it suffices to prove the proposition for any component of the vector $\boldsymbol{w}$. We omit the index of the component and denote by $w_0, w_1, w_2, w_3, \ldots$ one coordinate of $\boldsymbol{w}$ along the iterations of $\mathcal{P}.1$ and by $w^\star$ the result for the same component when solving $\mathcal{P}.2$. We need to show that $w^\star = w_2$. Expanding the ℓ_1-update of Eq. (19) over two iterations we get the following,

$$w_2 = \text{sign}(w_1)\,[|w_1| - \lambda_2]_+ = \text{sign}(w_1)\,\left[\left|\text{sign}(w_0)\,[|w_0| - \lambda_1]_+\right| - \lambda_2\right]_+ = \text{sign}(w_0)\,[|w_0| - \lambda_1 - \lambda_2]_+ \quad ,$$

where we used the positivity of $|\cdot|$. Examining $\mathcal{P}.2$ and using Eq. (19) again we get

$$w^\star = \text{sign}(w_0)\,[|w_0| - \lambda_1 - \lambda_2]_+ \quad .$$

Therefore, $w^\star = w_2$ as claimed.

Next we prove the proposition for ℓ_2, returning to using the entire vector for the proof. Using the explicit ℓ_2-update from Eq. (20), we can expand the norm of the vector $\boldsymbol{w}_1$ due to the program $\mathcal{P}.1$ as follows,

$$\|\boldsymbol{w}_1\| \;=\; \left[1 - \frac{\lambda_1}{\|\boldsymbol{w}_0\|}\right]_+ \|\boldsymbol{w}_0\| \;=\; [\|\boldsymbol{w}_0\| - \lambda_1]_+ \quad .$$

Similarly, we get that $\|\boldsymbol{w}_2\| = [\|\boldsymbol{w}_1\| - \lambda_2]_+$. Combining the norm equalities we see that the norm of $\boldsymbol{w}_2$ due to the succession of the two updates is

$$\|\boldsymbol{w}_2\| \;=\; \left[[\|\boldsymbol{w}_0\| - \lambda_1]_+ - \lambda_2\right]_+ \;=\; [\|\boldsymbol{w}_0\| - \lambda_1 - \lambda_2]_+ \quad .$$

Computing directly the norm of $\boldsymbol{w}^\star$ due to the update given by Eq. (20) yields

$$\|\boldsymbol{w}^\star\| \;=\; \left[1 - \frac{\lambda_1 + \lambda_2}{\|\boldsymbol{w}_0\|}\right]_+ \|\boldsymbol{w}_0\| \;=\; [\|\boldsymbol{w}_0\| - \lambda_1 - \lambda_2]_+ \quad .$$

Thus, $\boldsymbol{w}^\star$ and $\boldsymbol{w}_2$ have the same norm. Since the update itself retains the direction of the original vector $\boldsymbol{w}_0$, we get that $\boldsymbol{w}^\star = \boldsymbol{w}_2$ as needed.

We now turn to the most complicated update and proof of the three norms, the ℓ_∞ norm. We start by recapping the programs $\mathcal{P}.1$ and $\mathcal{P}.2$ for $T = 2$ and $q = \infty$,

$$\mathcal{P}.1: \quad \boldsymbol{w}_1 \;=\; \underset{\boldsymbol{w}}{\text{argmin}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_0\|^2 + \lambda_1 \|\boldsymbol{w}\|_\infty \right\} \tag{29}$$

$$\boldsymbol{w}_2 \;=\; \underset{\boldsymbol{w}}{\text{argmin}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_1\|^2 + \lambda_2 \|\boldsymbol{w}\|_\infty \right\} \quad , \tag{30}$$

$$\mathcal{P}.2: \quad \boldsymbol{w}^\star \;=\; \underset{\boldsymbol{w}}{\text{argmin}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{v}\|_2^2 + (\lambda_1 + \lambda_2) \|\boldsymbol{w}\|_\infty \right\} \quad . \tag{31}$$

We prove the equivalence of the two programs in two stages. First, we examine the case $\|\boldsymbol{w}_0\|_1 > \lambda_1 + \lambda_2$, and then consider the complement case $\|\boldsymbol{w}_0\|_1 \leq \lambda_1 + \lambda_2$. For concreteness and simplicity, we assume that $\boldsymbol{w}_0 \succeq \boldsymbol{0}$, since, clearly, the objective is symmetric in $\boldsymbol{w}_0$ and $-\boldsymbol{w}_0$. We thus can assume that all entries of $\boldsymbol{w}_0$ are non-negative. In the proof we use the following operators: $[\boldsymbol{v}]_+$ now denotes the positive component of each entry of $\boldsymbol{v}$, $\min\{\boldsymbol{v}, \theta\}$ denotes the component-wise minimum between the elements of $\boldsymbol{v}$ and θ, and likewise $\max\{\boldsymbol{v}, \theta\}$ is the component-wise maximum. Starting with the case $\|\boldsymbol{w}_0\|_1 > \lambda_1 + \lambda_2$, we examine Eq. (29). From Lagrange duality we know that that $\boldsymbol{w}_1 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1$, where $\boldsymbol{\alpha}_1$ is the solution of

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \frac{1}{2}\|\boldsymbol{\alpha} - \boldsymbol{w}_0\|_2^2 \;\text{ s.t. }\; \|\boldsymbol{\alpha}\|_1 \leq \lambda_1 \quad .$$

As described by Duchi et al. (2008) and reviewed above in Sec. 5, $\boldsymbol{\alpha}_1 = [\boldsymbol{w}_0 - \theta_1]_+$ for some $\theta_1 \in \mathbb{R}_+$. The form of $\boldsymbol{\alpha}_1$ readily translates to the following form for $\boldsymbol{w}_1$: $\boldsymbol{w}_1 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1 = \min(\boldsymbol{w}_0, \theta_1)$. Applying similar reasoning to the second step of $\mathcal{P}.1$ yields $\boldsymbol{w}_2 = \boldsymbol{w}_1 - \boldsymbol{\alpha}_2 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2$, where $\boldsymbol{\alpha}_2$ is the minimizer of

$$\frac{1}{2}\|\boldsymbol{\alpha} - \boldsymbol{w}_1\|_2^2 = \frac{1}{2}\|\boldsymbol{\alpha} - (\boldsymbol{w}_0 - \boldsymbol{\alpha}_1)\|_2^2 \quad \text{s.t. } \|\boldsymbol{\alpha}\|_1 \leq \lambda_2 \quad .$$

Again, we have $\boldsymbol{\alpha}_2 = [\boldsymbol{w}_1 - \theta_2]_+ = [\boldsymbol{w}_0 - \boldsymbol{\alpha}_1 - \theta_2]_+$ for some $\theta_2 \in \mathbb{R}_+$. The successive steps then imply that

$$\boldsymbol{w}_2 = \min\{\boldsymbol{w}_1, \theta_2\} = \min\left\{\min\{\boldsymbol{w}_0, \theta_1\}, \theta_2\right\} \ .$$

We next show that regardless of the ℓ_1-norm of $\boldsymbol{w}_0$, $\theta_2 \leq \theta_1$. Intuitively, if $\theta_2 > \theta_1$, the second minimization step of $\mathcal{P}.1$ would perform no shrinkage of $\boldsymbol{w}_1$ to get $\boldsymbol{w}_2$. Formally, assume for the sake of contradiction that $\theta_2 > \theta_1$. Under this assumption, we would have that $\boldsymbol{w}_2 = \min\{\min\{\boldsymbol{w}_0, \theta_1\}, \theta_2\} = \min\{\boldsymbol{w}_0, \theta_1\} = \boldsymbol{w}_1$. In turn, we obtain that $\mathbf{0}$ belongs to the subgradient set of Eq. (30) when evaluated at $\boldsymbol{w} = \boldsymbol{w}_1$, thus,

$$\mathbf{0} \in \boldsymbol{w}_1 - \boldsymbol{w}_1 + \lambda_2 \partial \left\|\boldsymbol{w}_1\right\|_\infty = \lambda_2 \partial \left\|\boldsymbol{w}_1\right\|_\infty \ .$$

Clearly, the set $\partial \left\|\boldsymbol{w}_1\right\|_\infty$ can contain $\mathbf{0}$ only when $\boldsymbol{w}_1 = \mathbf{0}$. Since we assumed that $\lambda_1 < \left\|\boldsymbol{w}_0\right\|_1$, and hence that $\boldsymbol{\alpha}_1 \preceq \boldsymbol{w}_0$ and $\boldsymbol{\alpha}_1 \neq \boldsymbol{w}_0$, we have that $\boldsymbol{w}_1 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1 \neq \mathbf{0}$. This contradiction implies that $\theta_2 \leq \theta_1$.

We now examine the solution vectors to the dual problems of $\mathcal{P}.1$, $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$. We know that $\left\|\boldsymbol{\alpha}_1\right\|_1 = \lambda_1$ so that $\left\|\boldsymbol{w}_0 - \boldsymbol{\alpha}_1\right\|_1 > \lambda_2$ and hence $\boldsymbol{\alpha}_2$ is at the boundary $\left\|\boldsymbol{\alpha}_2\right\|_1 = \lambda_2$ (see again Duchi et al. 2008). Furthermore, the sum of the these vectors is

$$\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2 = [\boldsymbol{w}_0 - \theta_1]_+ + \left[\boldsymbol{w}_0 - [\boldsymbol{w}_0 - \theta_1]_+ - \theta_2\right]_+ \ . \tag{32}$$

Let v denote a component of $\boldsymbol{w}_0$ greater than θ_1. For any such component the right hand side of Eq. (32) amounts to

$$[v - (v - \theta_1) - \theta_2]_+ + [v - \theta_1]_+ = [\theta_1 - \theta_2]_+ + v - \theta_1 = v - \theta_1 = [v - \theta_1]_+ \ ,$$

where we used the fact that $\theta_2 \leq \theta_1$ to eliminate the term $[\theta_1 - \theta_2]_+$. Next, let u denote a component of $\boldsymbol{w}_0$ smaller than θ_1. In this case, the right hand side of Eq. (32) amounts to $[u - 0 - \theta_2]_+ + 0 = [u - \theta_2]_+$. Recapping, the end result is that the vector sum $\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2$ equals $[\boldsymbol{w}_0 - \theta_2]_+$. Moreover, $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ are in $\mathbb{R}^n_+$ as we assumed that $\boldsymbol{w}_0 \succeq \mathbf{0}$, and thus

$$\left\|\,[\boldsymbol{w}_0 - \theta_2]_+\,\right\|_1 = \left\|\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2\right\|_1 = \lambda_1 + \lambda_2 \ . \tag{33}$$

We now show that $\mathcal{P}.2$ has the same dual solution as the sequential updates above. The dual of $\mathcal{P}.2$ is

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \ \frac{1}{2}\left\|\boldsymbol{\alpha} - \boldsymbol{w}_0\right\|_2^2 \ \text{ s.t. } \ \left\|\boldsymbol{\alpha}\right\|_1 \leq \lambda_1 + \lambda_2.$$

Denoting by $\boldsymbol{\alpha}_0$ the solution of the above dual problem, we have $\boldsymbol{w}^\star = \boldsymbol{w}_0 - \boldsymbol{\alpha}_0$ and $\boldsymbol{\alpha}_0 = [\boldsymbol{w}_0 - \theta]_+$ for some $\theta \in \mathbb{R}_+$. Examining the norm of $\boldsymbol{\alpha}_0$ we obtain that

$$\left\|\boldsymbol{\alpha}_0\right\|_1 = \left\|[\boldsymbol{w}_0 - \theta]_+\right\|_1 = \lambda_1 + \lambda_2 \tag{34}$$

because we assumed that $\left\|\boldsymbol{w}_0\right\|_1 > \lambda_1 + \lambda_2$. We can view the terms $\left\|[\boldsymbol{w}_0 - \theta_2]_+\right\|_1$ from Eq. (33) and $\left\|[\boldsymbol{w}_0 - \theta]_+\right\|_1$ from Eq. (34) as functions of θ_2 and θ, respectively. The functions are strictly decreasing functions of θ and θ_2 over the interval $[0, \left\|\boldsymbol{w}_0\right\|_\infty]$. Therefore, they are invertible for $0 < \lambda_1 + \lambda_2 < \left\|\boldsymbol{w}_0\right\|_1$. Since $\left\|[\boldsymbol{w}_0 - \theta]_+\right\|_1 = \left\|[\boldsymbol{w}_0 - \theta_2]_+\right\|_1$, we must have $\theta_2 = \theta$. Recall that the solution of Eq. (31) is $\boldsymbol{w}^\star = \min\{\boldsymbol{w}_0, \theta\}$, and the solution of the sequential update induced by Eq. (29) and Eq. (30) is $\min\{\min\{\boldsymbol{w}_0, \theta_1\}, \theta_2\} = \min\{\boldsymbol{w}_0, \theta_2\}$. The programs $\mathcal{P}.1$ and $\mathcal{P}.2$ therefore result in the same vector $\min\{\boldsymbol{w}_0, \theta_2\} = \min\{\boldsymbol{w}_0, \theta\}$ and their induced updates are equivalent.

We now examine the case when $\left\|\boldsymbol{w}_0\right\|_1 \leq \lambda_1 + \lambda_2$. If the 1-norm of $\boldsymbol{w}_0$ is also smaller than λ_1, $\left\|\boldsymbol{w}_0\right\|_1 \leq \lambda_1$, then the dual solution for the first step of $\mathcal{P}.1$ is $\boldsymbol{\alpha}_1 = \boldsymbol{w}_0$, which makes $\boldsymbol{w}_1 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1 = \mathbf{0}$ and hence $\boldsymbol{w}_2 = \mathbf{0}$. The dual solution for the combined problem is clearly $\boldsymbol{\alpha}_0 = \boldsymbol{w}_0$; again, $\boldsymbol{w}^\star = \boldsymbol{w}_0 - \boldsymbol{\alpha}_0 = \mathbf{0}$. We are thus left with the case $\lambda_1 < \left\|\boldsymbol{w}_0\right\|_1 \leq \lambda_1 + \lambda_2$. We straightforwardly get that the solution to Eq. (31) is $\boldsymbol{w}^\star = \mathbf{0}$. We now prove that the iterated solution obtained by $\mathcal{P}.1$ results in the zero vector as well. First, consider the dual solution $\boldsymbol{\alpha}_1$, which is the minimizer of $\left\|\boldsymbol{\alpha} - \boldsymbol{w}_0\right\|^2$ subject to $\left\|\boldsymbol{\alpha}\right\|_1 \leq \lambda_1$. Since $\boldsymbol{\alpha}_1 = [\boldsymbol{w}_0 - \theta_1]_+$ for

some $\theta_1 \geq 0$, we know that each component of $\boldsymbol{\alpha}_1$ is between zero and its corresponding component in $\boldsymbol{w}_0$, therefore, $\|\boldsymbol{w}_0 - \boldsymbol{\alpha}_1\|_1 = \|\boldsymbol{w}_0\|_1 - \|\boldsymbol{\alpha}\|_1 = \|\boldsymbol{w}_0\|_1 - \lambda_1 \leq \lambda_2$. The dual of the second step of $\mathcal{P}.1$ distills to the minimization $\frac{1}{2}\|\boldsymbol{\alpha} - (\boldsymbol{w}_0 - \boldsymbol{\alpha}_1)\|^2$ subject to $\|\boldsymbol{\alpha}\|_1 \leq \lambda_2$. Since we showed that $\|\boldsymbol{w}_0 - \boldsymbol{\alpha}\|_1 \leq \lambda_2$, we get $\boldsymbol{\alpha}_2 = \boldsymbol{w}_0 - \boldsymbol{\alpha}_1$. This means that $\theta_2 = 0$. Recall that the solution of $\mathcal{P}.1$ is $\min\{\boldsymbol{w}_0, \theta_2\}$, which amounts to the zero vector when $\theta_2 = 0$. We have thus showed that both optimization problems result in the zero vector. This proves the equivalence of $\mathcal{P}.1$ and $\mathcal{P}.2$ for $q = \infty$. ■

The algorithmic consequence of Proposition 11 is that it is possible to perform a lazy update on each iteration by omitting the terms of $\boldsymbol{w}_t$ (or whole rows of the matrix W_t when using mixed-norms) that are outside the support of $\boldsymbol{g}_t^f$, the gradient of the loss at iteration t. However, we do need to maintain the step-sizes used on each iteration and have them readily available on future rounds when we need to update coordinates of $\boldsymbol{w}$ or W that were not updated in previous rounds. Let Λ_t denote the sum of the step sizes times regularization multipliers $\lambda\eta_t$ used from round 1 through t. Then a simple algebraic manipulation yields that instead of solving

$$\boldsymbol{w}_{t+1} = \operatorname*{argmin}_{\boldsymbol{w}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_t\|_2^2 + \lambda\eta_t \|\boldsymbol{w}\|_q \right\}$$

repeatedly when $\boldsymbol{w}_t$ is not being updated, we can simply cache the last time t_0 that $\boldsymbol{w}$ (or a coordinate in $\boldsymbol{w}$ or a row from W) was updated and, when it is needed, solve

$$\boldsymbol{w}_{t+1} = \operatorname*{argmin}_{\boldsymbol{w}} \left\{ \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_t\|_2^2 + (\Lambda_t - \Lambda_{t_0})\|\boldsymbol{w}\|_q \right\} \quad .$$

The advantage of the lazy evaluation is pronounced when using mixed-norm regularization as it lets us avoid updating entire rows so long as the row index corresponds to a zero entry of the gradient $\boldsymbol{g}_t^f$. We would like to note that the lazy evaluation due to Proposition 11 includes as a special case the efficient implementation for ℓ_1-regularized updates first outlined by Langford et al. (2008). In sum, at the expense of keeping a time stamp t for each entry of $\boldsymbol{w}$ or row of W and maintaining a list of the cumulative sums $\Lambda_1, \Lambda_2, \ldots$, we can get $O(s)$ updates of $\boldsymbol{w}$ when the gradient $\boldsymbol{g}_t^f$ is s-sparse, that is, it has only s non-zero components.

7. Experiments

In this section we describe the results of experiments we performed whose goal are to demonstrate the merits and underscore a few weaknesses of FOBOS. To that end, we also evaluate specific instantiations of FOBOS with respect to several state-of-the-art optimizers and projected subgradient methods on different learning problems. In the experiments that focus on efficiency and speed of convergence, we evaluate the methods in terms of their number of operations, which is approximately the number of floating point operations each method performs. We believe that this metric offers a fair comparison of the different algorithms as it lifts the need to cope with specific code optimization such as cache locality or different costs per iteration of each of the methods.

7.1 Sensitivity Experiments

We begin our experiments by performing a sensitivity analysis of FOBOS. We perform some of the analysis in later sections during our comparisons to other methods, but we discuss the bulk of it here. We focus on two tasks in our sensitivity experiments: minimizing the hinge loss (used in Support Vector Machines) with ℓ_2^2 regularization and minimizing the ℓ_1-regularized logistic loss. These set the loss function f as

$$f(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} [1 - y_i \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle]_+ + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 \quad \text{and} \quad f(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + e^{-y_i \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle}\right) + \lambda\|\boldsymbol{w}\|_1$$

respectively. Note that both loss functions have subgradient sets bounded by $\frac{1}{n}\sum_{i=1}^{n}\|y_i \boldsymbol{x}_i\|_2$. Therefore, if all the instances are of bounded norm, so are the subgradients of the empirical loss functions.

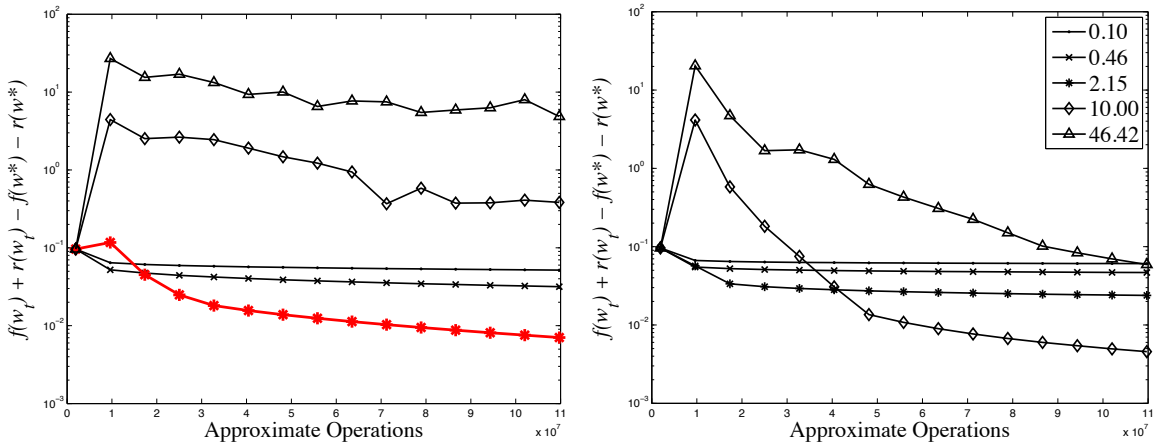

Figure 1: Sensitivity of deterministic FOBOS to initial step size on logistic regression. Key is initial step size. Left: $\eta_t \propto 1/\sqrt{t}$. Right: $\eta_t \propto 1/t$.

We perform analysis using dimensions $d \in \{50, 100, 200, 400, 800\}$ and data set sizes $n \in \{200, 400\}$. We investigate the effect of correlation of the features x_{ij} with one another by generating uncorrelated, moderately correlated, and highly correlated features. Specifically, to generate feature vectors $\boldsymbol{x}_i \in \mathbb{R}^d$, we sample n random vectors $\boldsymbol{z}_i \sim N(0, I)$, $\boldsymbol{z}_i \in \mathbb{R}^d$. We then construct a random covariance matrix Σ whose correlations $|\rho_{ij}| = |\Sigma_{ij}|/\sqrt{\Sigma_{ii}\Sigma_{jj}}$ have mean .2 for the moderately correlated experiments and .3 for the highly correlated experiments (on the highly correlated data, more than one-tenth of the features were more than 80% correlated). To get $\boldsymbol{x}$, we then set $\boldsymbol{x}_i = L\boldsymbol{z}_i$, where L is the Cholesky decomposition of Σ (the identity in the uncorrelated case), and normalize $\boldsymbol{x}_i$ to have $\|\boldsymbol{x}_i\|_\infty = 1$. We compared different stochastic gradient estimators that were based on varying sample sizes: a single example, 10% of the training set, 20% of the training set, and deterministic gradient using the entire data set. We also tested ten different initial step sizes. However, we give in the graphs results for only a subset of the initial steps that reveals the overall dependency on the step size. Further, we also checked three schemes for decaying the step size: $\eta_t \propto 1/t$, $\eta_t \propto 1/\sqrt{t}$, and $\eta_t \propto 1$ (constant step size). We discuss the results attained by constant step sizes only qualitatively, though, to keep the clarity of the figures. When η_t was a constant we divided the initial step size by $\sqrt{T}$, the total number of iterations being taken. We performed each experiment 10 times and averaged the results.

We distill the large number of experiments into a few figures here, deferring some of the analysis to the sequel. Thus in this section we focus on the case when $n = 400$ and $d = 800$, since the experiments with other training set sizes and dimensions gave qualitatively similar results. Most of the results in this section focus on the consequences of the initial step size η_1, though we also discuss different schedules of the learning rate and the sample size for computing the subgradients. In each experiment, we set $\lambda = .25/\sqrt{n}$, which in the logistic case gave roughly 50% sparsity. Before our discussion, we note that we can bound the ℓ_2-norm of $\boldsymbol{w}^\star$ for both the logistic and the hinge loss. In the case of the logistic, we have $\lambda\|\boldsymbol{w}^\star\|_2 \le \lambda\|\boldsymbol{w}^\star\|_1 \le f(\boldsymbol{0}) = \log 2$. Similarly, for the hinge loss we have $\frac{\lambda}{2}\|\boldsymbol{w}^\star\|_2^2 \le f(\boldsymbol{0}) = 1$. In both cases we can bound G by the norm of the $\|\boldsymbol{x}_i\|$, which is in our settings approximately 9. Thus, looking at the bounds from Sec. 3 and Sec. 4, when $\eta_t \propto 1/\sqrt{T}$, the initial step size amounts to $\eta_1 \approx D/\sqrt{7}G \approx 2.3$ and when $\eta_t \propto 1/\sqrt{t}$, the initial step should be $\eta_1 \approx D/4G \approx 1.5$ for logistic regression. For the hinge loss, when $\eta_t \propto 1/\sqrt{t}$, and the initial step ends being $\eta_1 \approx D/4G \approx .35$. We see in the sequel that these approximate step sizes yield results competitive with the best initial step sizes, which can be found only in hindsight.

We begin by considering the effect of initial step size for ℓ_1-regularized logistic regression. We plot results for the moderately correlated data sets, as we investigate the effect of correlation later on. The results are given in Figures 1 and 2, where we plot the objective value at each time step minus the optimal objective

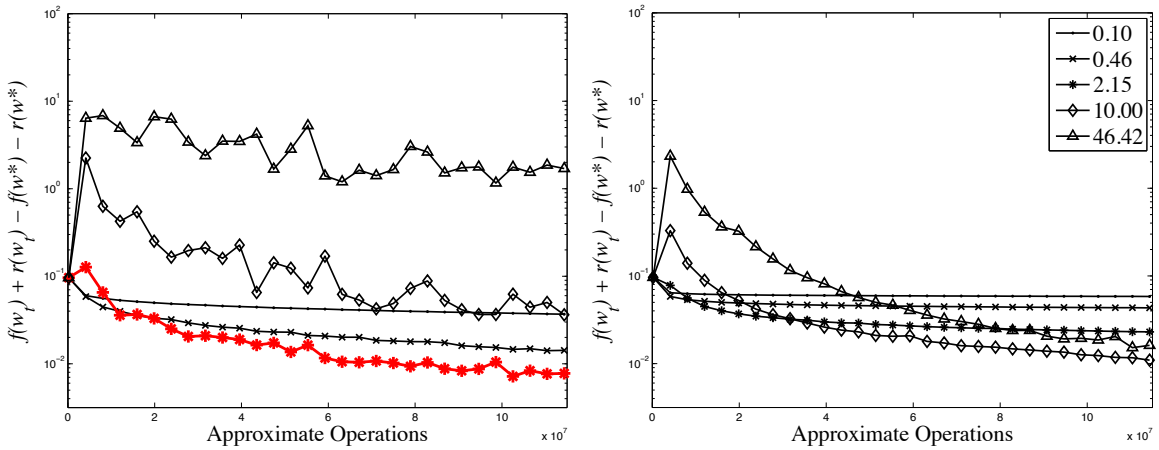

Figure 2: Sensitivity of stochastic FOBOS to initial step size on logistic regression. Key is initial step size. Left: $\eta_t \propto 1/\sqrt{t}$. Right: $\eta_t \propto 1/t$.

value attained, $f(\boldsymbol{w}_t) + r(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star) - r(\boldsymbol{w}^\star)$. We plot the function values of the initial step size that was chosen automatically, by estimating D and G as described above, in bold red in the figures. Interestingly, the left side of Fig. 1 suggests that the best performing initial steps when $\eta_t \propto 1/\sqrt{t}$ are near 1.5: $\eta_1 = 2.15$ gives good regret (we also saw that $\eta_1 = 4.6$ performed well, but do not include it in the plot). In Fig. 2, we see similar behavior for stochastic FOBOS when using 10% of the samples to estimate the gradient. Though we do not plot these results, the stochastic case with constant step sizes also performs well when the step sizes are $\eta_t \approx 2.2/\sqrt{T}$. The deterministic variant of FOBOS could not not take as many steps in the allotted time as the stochastic versions, so its performance was not competitive. All methods are somewhat sensitive to initial step size. Nonetheless, for these types of learning problems it seems plausible to estimate an initial step size based on the arguments given in the previous paragraph, especially when using the step rate of $\eta_t \propto 1/\sqrt{t}$ that was suggested by the analysis in Sec. 4.

We performed similar experiments with the hinge loss with ℓ_2^2 regularization. As the objective is strongly convex,[2] our analysis from Sec. 4 and Theorem 8 suggests a step rate of $\eta_t = 1/\lambda t$. From the right plot in Fig. 3, we see that the best performing step sizes where the two largest, which is consistent with our analysis since $1/\lambda = 80$. For the the $1/\sqrt{t}$ steps, which we present on the left of Fig. 3 respectively, such initial step sizes are too large. However, we can see again that the approximation suggested by our arguments above gives good performance. In sum, it seems that while FOBOS and stochastic FOBOS are fairly sensitive to initial step sizes and rate schedule, our theoretical results from the previous sections give relatively good initial step size heuristics. We will see similar behavior in the sequel.

7.2 Comparison to Subgradient Optimizers

We now move on to the description of experiments using FOBOS to solve ℓ_2^2-regularized learning problems, focusing on comparison to the state-of-the-art subgradient optimizer Pegasos (Shalev-Shwartz et al., 2007). Pegasos was originally implemented and evaluated on Support Vector Machine (SVM) problems by using the hinge-loss as the empirical loss function along with an ℓ_2^2 regularization term. Nonetheless, Pegasos can be rather simply extended to the binary logistic loss function. We thus experimented with both the hinge and logistic loss functions. To generate data for our experiments, we chose a vector $\boldsymbol{w}$ with entries distributed normally with a zero mean and unit variance, while randomly zeroing 50% of the entries in the vector. The

2. We assume there is no bias term in the objective, since any optimization method must deal with this so we find it outside the scope of the paper.

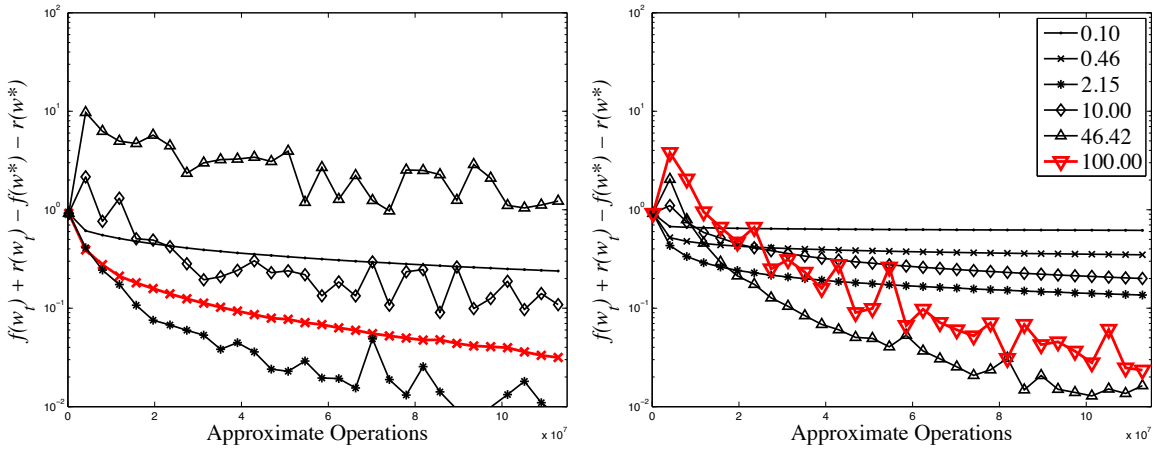

Figure 3: Sensitivity of stochastic methods to initial step size on hinge loss minimization with ℓ_2^2-regularization. Key is initial step size. Left: $\eta_t \propto 1/\sqrt{t}$. Right: $\eta_t \propto 1/t$.

examples $\boldsymbol{x}_i \in \mathbb{R}^d$ were also chosen at random with entries i.i.d. normally distributed. We also performed experiments using correlated data. The results attained on correlated data were similar, so we do not report them in our comparison to Pegasos. To generate target values, we set $y_i = \text{sign}(\langle \boldsymbol{x}_i, \boldsymbol{w} \rangle)$, and negated the sign of 10% of the examples to add label noise. In all experiments, we used $n = 1000$ training examples of dimension $d = 400$.

The graphs of Fig. 4 show (on a log-scale) the objective function, namely, the regularized empirical loss of the algorithms, minus the optimal value of the objective function. These results were averaged over 20 independent runs of the algorithms. In all experiments with a regularization of the form $\frac{1}{2}\lambda\|\boldsymbol{w}\|_2^2$, we used step sizes of the form $\eta_t = 1/(\lambda t)$ to achieve the logarithmic regret bound of Sec. 4. The left graph of Fig. 4 conveys that FOBOS performs comparably to Pegasos on hinge (SVM) loss. Both algorithms quickly approach the optimal value. In this experiment we let both Pegasos and FOBOS employ a projection after each gradient step onto a ℓ_2 norm ball in which $\boldsymbol{w}^\star$ must lie (see Shalev-Shwartz et al. 2007 and the discussion following the proof of Theorem 2). However, in the experiment corresponding to the right plot of Fig. 4, we eliminated the additional projection step and ran the algorithms with the logistic loss. In this case, FOBOS slightly outperforms Pegasos. We hypothesize that the slightly faster rate of FOBOS is due to the explicit shrinkage that FOBOS performs in the ℓ_2^2 update (see Eq. (20) and Lemma 9).

7.3 Comparison to Other Methods for Smooth Problems

As mentioned in our discussion of related work, many methods have been proposed in the optimization and machine learning literature for minimizing Eq. (1) when f is smooth, particularly when f has a Lipschitz-continuous gradient. For the case of ℓ_1-regularized logistic regression, Koh et al. (2007) propose an efficient interior point method. Tseng and Yun (2007) give analysis of a (block) coordinate descent method that uses approximations to the Hessian matrix and an Armijo-type backtracking line search to solve non-smooth regularized problems with smooth objectives; their method was noted to be effective for ℓ_1/ℓ_2-regularized logistic regression, for example, by Meier et al. (2008). We also compare FOBOS and its stochastic variants to the SPARSA method of Wright et al. (2009), which shares the same update as FOBOS but uses a simple line search strategy to choose the its steps. Note that none of these methods apply when f is non-smooth. Lastly, we compare FOBOS to projected-gradient methods. For ℓ_1-regularized problems, Duchi et al. (2008) show how to compute projections to an ℓ_1-ball in linear time, and Schmidt et al. (2009) extend the method to show that projection of a matrix $W \in \mathbb{R}^{d \times k}$ to an ℓ_1/ℓ_2-constraint can be computed in $O(dk)$ time. To compare

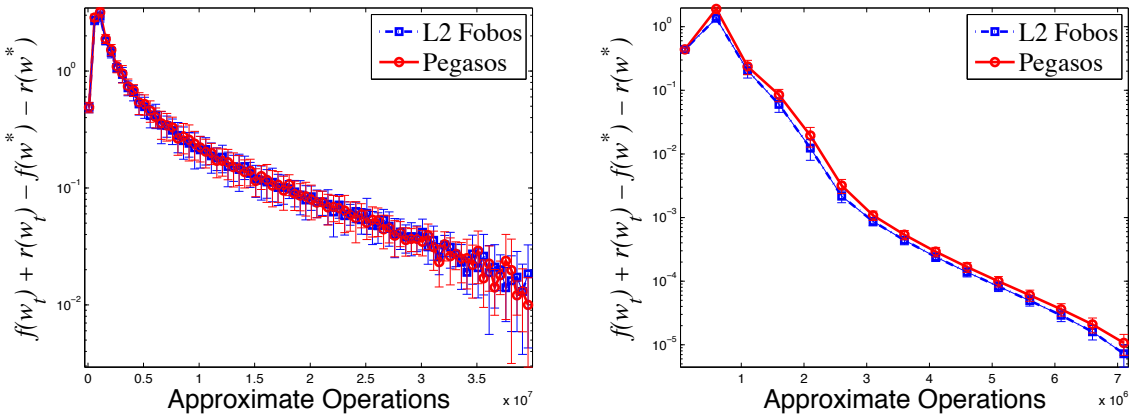

Figure 4: Comparison of FOBOS with Pegasos on the SVM (left) and logistic regression (right). The right hand side plot shows the performance of the algorithms without a projection step.

our methods to projected gradient methods, we first solve the regularized version of the problem. We then constrain the norm of $\boldsymbol{w}$ or the mixed-norm of W to lie within a ball of radius $\lambda\|\boldsymbol{w}^\star\|_1$ or $\lambda\|W^\star\|_{\ell_1/\ell_2}$.

We compared FOBOS to each of the above methods on many different synthetic data sets, and we report a few representative results. In our experiments, SPARSA seemed to outperform FOBOS and the projected gradient methods when using full (deterministic) gradient information. The additional function evaluations incurred by the line search in SPARSA seem to be insignificant to runtime, which is a plausible explanation for SPARSA's superior performance. We therefore do not report results for the deterministic versions of FOBOS and projected gradient methods to avoid clutter in the figures.

In the first set of experiments, we compare FOBOS's performance on ℓ_1-regularized logistic regression to each of the above methods. That is, we set $f(\boldsymbol{w})$ to be the average logistic loss and $r(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_1$ and use a data set with $n = 1000$ examples and $d = 400$ dimensions. We compare the performance of stochastic FOBOS to the other algorithms in terms of two aspects. The first is the value of λ, which we set to five logarithmically spaced values that gave solution vectors $\boldsymbol{w}^\star$ ranging from 100% non-zero entries to only 5% non-zero entries. The second aspect is on the correlation of the features. We generated random data sets with uncorrelated features, features that were on average 20% correlated with one another, and features that were on average 30% correlated with one another. In the latter case about 350 pairs of features had above 80% correlation (see the description of feature generation at the beginning of the section). We normalized each example $\boldsymbol{x}$ to have features in the range $[-1,1]$. We assigned labels for each example by randomly choosing a 50% sparse vector $\boldsymbol{w}$, setting $y_i = \text{sign}(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)$, and negating 5% of the y values.

The results comparing FOBOS to the other algorithms for different settings of the regularizer λ are in Fig. 5. The y-axis is $f(\boldsymbol{w}_t) + r(\boldsymbol{w}_t) - f(\boldsymbol{w}^\star) - r(\boldsymbol{w}^\star)$, the distance of the current value from the optimal value, and the x-axis is the approximate number of operations (FLOPs) for each method. We used the approximation we derived based on Corollary 7 in our earlier discussion of sensitivity to set the initial step size and used $\eta_t \propto 1/\sqrt{t}$. Tseng and Yun's method requires setting of constants for the backtracking-based line search. We thus use the settings in Meier et al. (2008). In attempt to make the comparisons as fair as possible, we used some of Tseng and Yun's code yet reimplemented the method to take advantage of the specific structure of logistic regression. Similarly, we used the line-search parameters in Wright et al.'s publicly available Matlab code for SPARSA, though we slightly modified their code to handle arbitrary loss functions. From the figure, we see that as λ grows, yielding sparser solutions for $\boldsymbol{w}^\star$, the performance of coordinate descent and especially the interior point method start to degrade relative to the stochastic methods and SPARSA.

In our experiments we found that the stochastic methods were quite resilient to overly-large initial step-sizes, as they quickly took a large number of steps. SPARSA employs an easy to implement and efficient line search, and in general yielded good performance. The coordinate descent method, with its somewhat

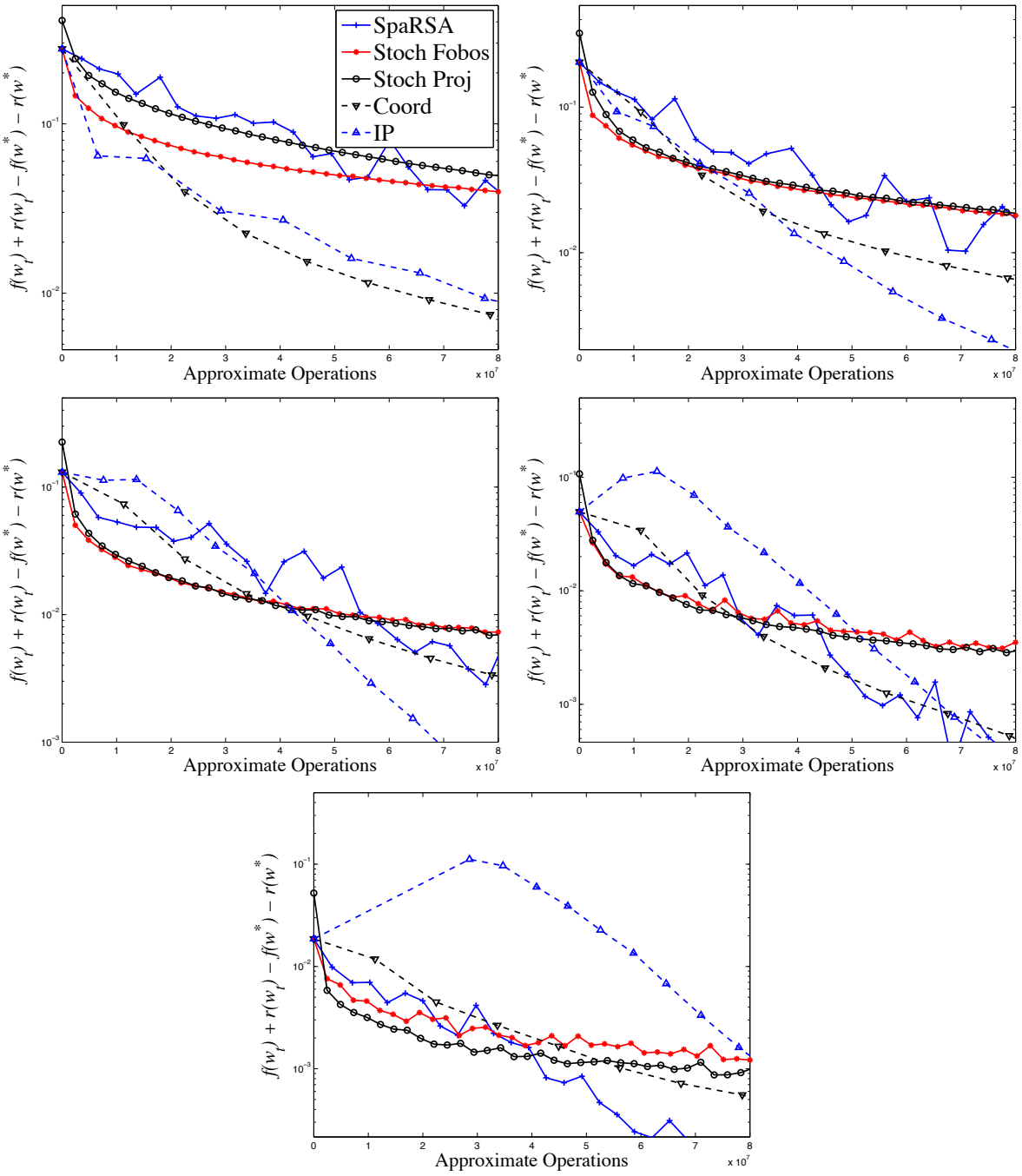

Figure 5: Performance of ℓ_1-regularized logistic regression methods with different settings of λ on correlated synthetic data. Left to right, top to bottom: $\boldsymbol{w}^\star$ has 0% sparsity, $\boldsymbol{w}^\star$ has 25% sparsity, $\boldsymbol{w}^\star$ has 40% sparsity, $\boldsymbol{w}^\star$ has 70% sparsity, and $\boldsymbol{w}^\star$ has 95% sparsity.

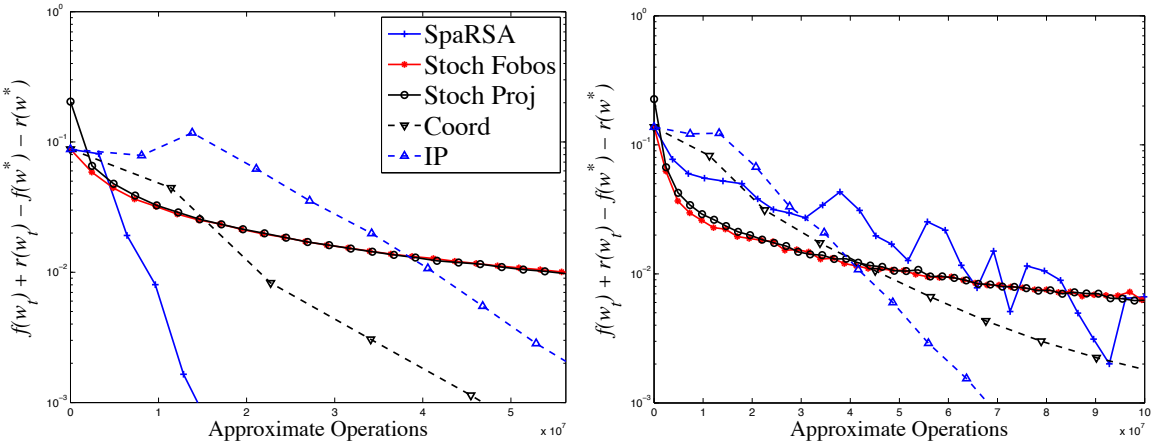

Figure 6: Performance of ℓ_1-regularized logistic regression methods with different correlations on synthetic data. Left: uncorrelated data. Right: highly correlated data.

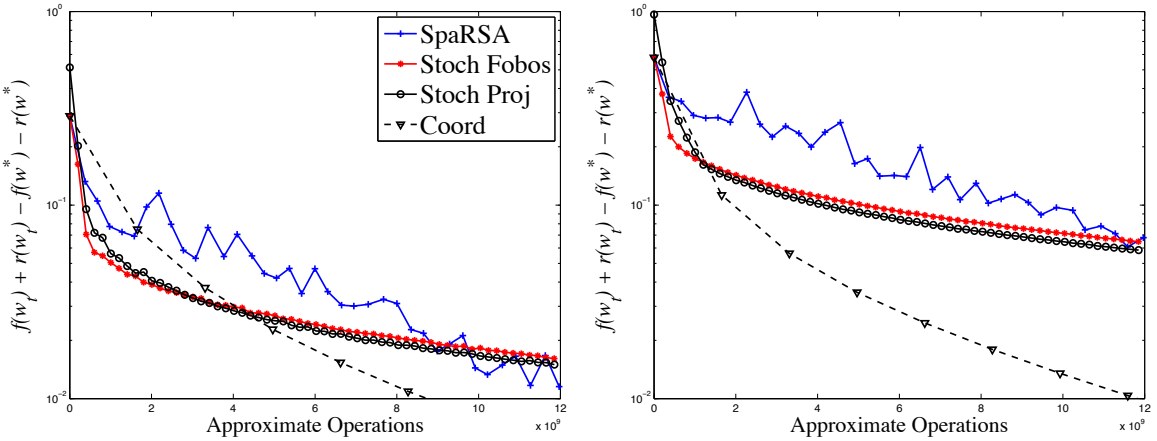

Figure 7: Performance of ℓ_1/ℓ_2-regularized multiclass logistic regression methods with different settings of λ on correlated synthetic data. Left: $\boldsymbol{w}^\star$ has 60% sparsity. Right: $\boldsymbol{w}^\star$ has 30% sparsity.

complicated backtracking line search, was difficult to implement correctly. Therefore, our experiments and experience suggest that SPARSA is likely to be preferred for smooth problems. Nonetheless, stochastic FOBOS quickly obtains a solution within about 10^{-2} of the optimal value. Since the regularized empirical loss serves as a proxy for attaining good generalization performance, we found that in numerous cases this accuracy sufficed to achieve competitive *test* loss.

In Fig. 6 we compare FOBOS to the other methods on data with uncorrelated, moderately correlated, and very correlated features. These plots all have λ set so that $\boldsymbol{w}^\star$ has approximately 40% sparsity. From the plots, we see that stochastic FOBOS and projected gradient actually perform very well on the more correlated data, very quickly getting to within 10^{-2} of the optimal value, though after this they essentially jam. As in the earlier experiments, SPARSA seems to perform quite well for these moderately sized experiments, though the interior point method's performance improves as the features become more correlated.

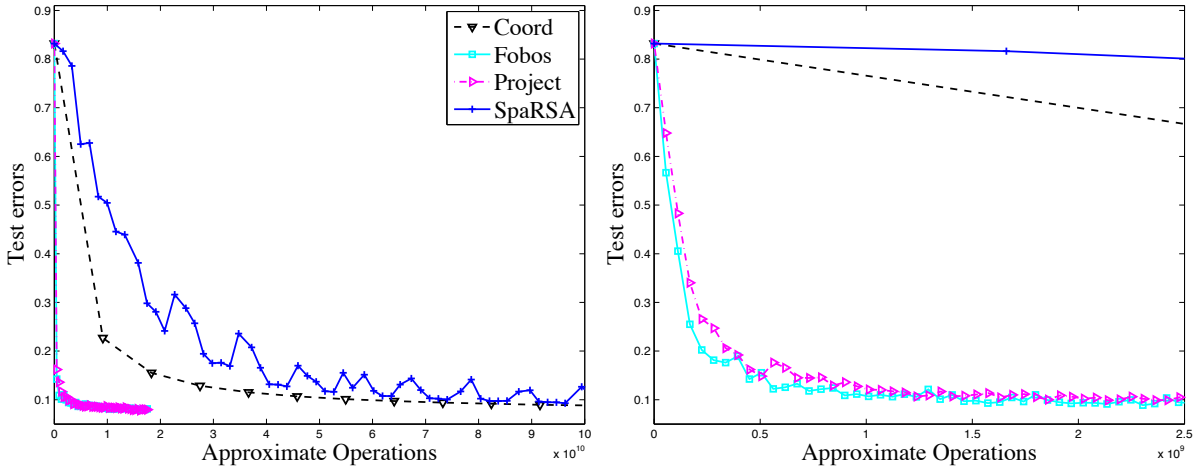

Figure 8: Comparison test error rate of FOBOS, SPARSA, projected gradient, and coordinate descent on MNIST digit recognition data set. Right: magnified view of left plot.

The last set of experiments with synthetic data sets was on mixed-norm-regularized multiclass logistic regression. The objective that we used in this case is

$$\frac{1}{n}\sum_{i=1}^{n}\log\left(1+\sum_{j\neq y_i}e^{\langle x_i, w^j - w^{y_i}\rangle}\right)+\lambda\|W\|_{\ell_1/\ell_q} \quad . \tag{35}$$

In the above equation q represents the norm over rows of the matrix W, and in our experiments it is either 1, 2, or ∞ (in this section, $q=2$). The goal is to classify correctly examples whose labels are in $\{1,\ldots,k\}$ while jointly regularizing entries of the vectors w^j. We used $n=5000$ datapoints of dimension $d=1000$ with $k=10$ classes, meaning we minimize a loss over a matrix $W\in\mathbb{R}^{d\times k}$ with 10000 parameters. To generate data, we sample examples x_i from a normal distribution with moderate correlation, randomly choose a matrix W, and set $y_i=\operatorname{argmax}_j\langle x_i, w^j\rangle$ with 5% label noise. We show results in Fig. 7. In the three figures, vary λ to give solutions $W^\star$ with roughly 60% zero rows, 30% zero rows, and completely non-sparse $W^\star$. From the figures, it is apparent that the stochastic methods, both FOBOS and projected gradient, exhibit very good initial performance but eventually lose to the coordinate descent method in terms of optimization speed. As before, if one is willing to use full gradient information, SPARSA seems a better choice than the deterministic counterpart of FOBOS and projected gradient algorithms. We thus again do not present results for deterministic FOBOS without any line search.

7.4 Experiments with Real Data Sets

Though in the prequel we focus on optimization speed, the main goal in batch learning problems is attaining good test-set error rates rather than rapid minimization of $f(w)+r(w)$. In order to better understand the merits of different optimization methods, we compared the performance of different optimizers on achieving a good test-set error rate on different data sets. Nonetheless, for these tests the contours for the training objective value were qualitatively very similar to the test-set error rate curves. We used the StatLog LandSat Satellite data set (Spiegelhalter and Taylor, 1994), the MNIST handwritten digit database, and a sentiment classification data set (Blitzer et al., 2007).

The MNIST database consists of 60,000 training examples and a 10,000 example test set with 10 classes. We show average results over ten experiments using random 15,000 example subsamples of the training set. For MNIST, each digit is a 28×28 gray scale image z which is represented as a $28^2=784$ dimensional

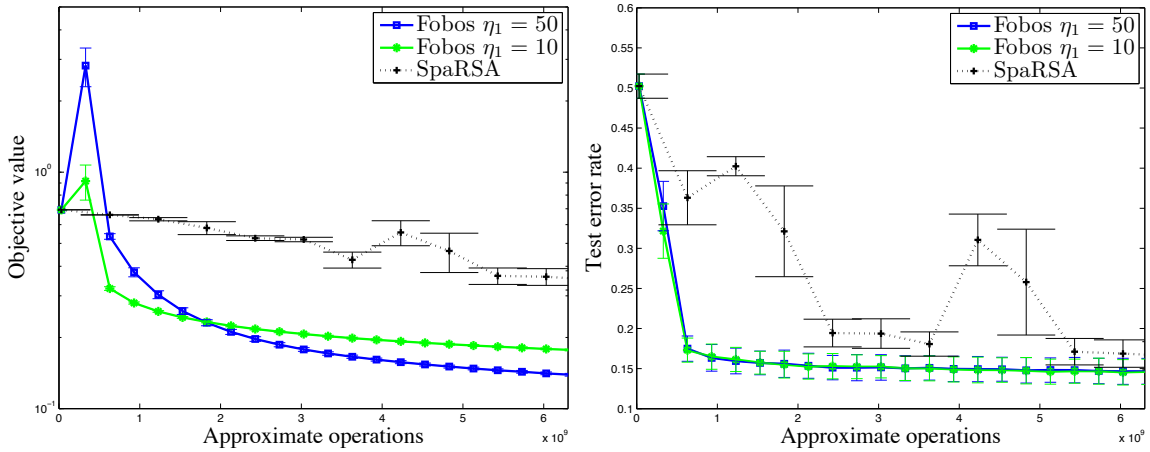

Figure 9: Comparison of FOBOS and SPARSA on sentiment classification task.

vector. Direct linear classifiers do not perform well on this data set. Thus, rather than learning weights for the original features, we learn the weights for a kernel machine with Gaussian kernels, where the value of the j^{th} feature for the i^{th} example is

$$x_{ij} = K(\boldsymbol{z}_i, \boldsymbol{z}_j) \triangleq e^{-\frac{1}{2}\|\boldsymbol{z}_i - \boldsymbol{z}_j\|^2} \quad .$$

We used ℓ_1/ℓ_2 regularization and compared FOBOS, SPARSA, coordinate descent, and projected gradient methods on this test (as well as stochastic gradient versions of FOBOS and projected gradient). The results for deterministic FOBOS and projected gradient were similar to SPARSA, so we do not present them. We also experimented with stochastic group sizes of 100 and 200 examples for FOBOS, but the results were similar, so we plot only the results from the 100 example runs. As before, we used the ℓ_1/ℓ_2 norm of the solution vectors for FOBOS, SPARSA, and coordinate descent as the constrained value for the projected gradient method. For each of the gradient methods, we estimated the diameter D and maximum gradient G as in the synthetic experiments, which led us to use a step size of $\eta_t = 30/\sqrt{t}$. The test set error rate as a function of number of operations for each of the methods are shown in Fig. 8. From Fig. 8, it is clear that the stochastic gradient methods (FOBOS and projected gradient) were significantly faster than any of the other methods. Before the coordinate descent method has even visited every coordinate once, stochastic FOBOS (and similarly stochastic projected gradient) have attained the minimal test-set error. The inferior performance of coordinate descent and deterministic gradient methods can be largely attributed to the need to exhaustively scan the data set. Even if we use only a subset of 15,000 examples, it takes a nontrivial amount of time to simply handle each example. Moreover, the objective values attained during training are qualitatively very similar to the test loss, so that stochastic FOBOS *much* more quickly reduces the training objective than the deterministic methods.

We also performed experiments on a data set that was very qualitatively different from the MNIST and LandSAT data sets. For this experiment, we used the multi-domain sentiment data set of Blitzer et al. (2007), which consists of product reviews taken from Amazon.com for many product types. The prediction task is to decide whether an article is a positive or negative review. The features are bigrams that take values in $\{0,1\}$, totalling about 630,000 binary features. In any particular example, at most a few thousand features are non-zero. We used 10,000 examples in each experiment and performed 10 repetitions while holding out 1000 of the examples as a test set and using 9000 of the examples for training. We used ℓ_1-regularized logistic regression and set $\lambda = 3 \cdot 10^{-5}$, which gave the best generalization performance and resulted in roughly 5% non-zeros in the final vector $\boldsymbol{w}$. We compare stochastic FOBOS to SPARSA in Fig. 9, since the projected gradient method is much slower than FOBOS (detailed in the sequel). For FOBOS we use 900 examples to compute each stochastic gradient. We use two different initial step sizes, one estimated using the approximation described earlier and a second where we scale it by $1/5$. The left plot in Fig. 9 shows the

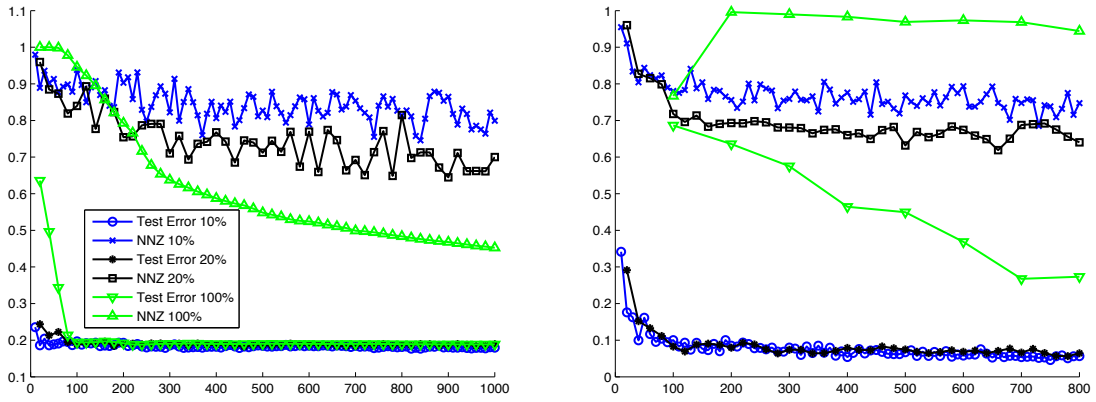

Figure 10: Left: FOBOS sparsity and test error for LandSat data set with ℓ_1-regularization. Right: FOBOS sparsity and test error for MNIST data set with ℓ_1/ℓ_2-regularization. Key is identical for both plots.

training objective value as a function of the number of operations for each of the three methods as well as error bars equal to the standard deviation of the objective. The right plot shows the error rates on the test sets. The behavior in the experiment is similar to that in Fig. 8, where the stochastic methods very quickly attain a small test error. Effectively, before SPARSA finishes two steps, the stochastic methods have arrived at approximate solutions that attain the minimal test set error rates.

We now change our focus from training time to the attained sparsity levels for multiclass classification with ℓ_1, ℓ_1/ℓ_2, and ℓ_1/ℓ_∞ regularization on MNIST and the StatLog LandSat data set. For the LandSat data set we attempt to classify 3×3 neighborhoods of pixels in a satellite image as a particular type of ground, and we expanded the input 36 features into 1296 features by taking the product of all features.

In the left plot of Fig. 10, we show the test set error and sparsity level of W as a function of training time (100 times the number of single-example gradient calculations) for the ℓ_1-regularized multiclass logistic loss with 720 training examples. The green lines show results for using all 720 examples to calculate the gradient, black using 20% of the examples, and blue using 10% of the examples to perform stochastic gradient. Each used the same learning rate $\eta_t \propto 1/\sqrt{t}$, and the reported results are averaged over 5 independent runs with different training data sets. The righthand figure shows a similar plot for training FOBOS on the MNIST data set with ℓ_1/ℓ_2-regularization. The objective value in training has a similar contour to the test loss. As expected, FOBOS with stochastic gradient descent gets to its minimum test classification error, and as the training set size increases this behavior is consistent. However, the deterministic version increases the level of sparsity throughout its run, while the stochastic-gradient version has highly variable sparsity levels and does not give solutions as sparse as the deterministic counterpart. We saw similar behavior when using stochastic versus deterministic projected gradient methods. The slowness of the deterministic gradient means that we do not see the sparsification immediately in larger tests; nonetheless, for longer training times similar sparsifying behavior emerges.

As yet, we do not have a compelling justification for the difference in sparsity levels between stochastic and deterministic gradient FOBOS. We give here some intuitive arguments, leaving a more formal analysis to future work. We develop one possible explanation by exploring the effect of taking a stochastic gradient step starting from the true solution vector $\boldsymbol{w}^\star$. Consider ℓ_1-regularized FOBOS with regularization multiplier λ. Let $\boldsymbol{g}^f$ be the gradient of $f(\boldsymbol{w}^\star)$. For j such that $w_j^\star > 0$, we have $g_j^f = -\lambda$, for j with $w_j^\star < 0$, $g_j^f = \lambda$, and for zero entries in $\boldsymbol{w}^\star$, we have $g_j^f \in [-\lambda, \lambda]$. The FOBOS step then amounts to

$w_j^+ = \text{sign}(w_j^\star)\left[|w_j^\star - \eta_t g_j^f| - \eta_t\lambda\right]_+$, which by inspection simply yields $\boldsymbol{w}^\star$. Now suppose that instead of $\boldsymbol{g}^f$, we use a stochastic estimate $\tilde{\boldsymbol{g}}^f$ of $\boldsymbol{g}^f$. Then the probability that the update w_j^+ of $w_j^\star$ is zero is

$$\mathbb{P}\left(\left|w_j^\star - \eta_t \tilde{g}_j^f\right| \leq \eta_t\lambda\right) = \mathbb{P}\left(\tilde{g}_j^f \in \left[\frac{w_j^\star}{\eta_t} - \lambda, \frac{w_j^\star}{\eta_t} + \lambda\right]\right) \quad .$$

When $w_j^\star = 0$, the probability is simply $\mathbb{P}(\tilde{g}_j^f \in [-\lambda, \lambda])$, which does not change as a function of η_t. However, when $w_j^\star > 0$, we have $E[\tilde{g}_j^f] = -\lambda$, while $w_j^\star/\eta_t \to \infty$ as η_t shrinks (and analogously for $w_j^\star < 0$). In essence, the probability of a non-zero parameter staying non-zero is high, however, the probability of a zero parameter staying zero is constant. Intuitively, then, we expect that stochastic gradient descent will result in more non-zero coefficients than the deterministic variant of FOBOS.

7.5 Experiments with Sparse Gradient Vectors

In this section, we consider the implications of Proposition 11 for learning with sparse data. We show that it is much more efficient to use the updates described in Proposition 11 than to maintain ℓ_1-constraints on $\boldsymbol{w}$ as in Duchi et al. (2008). Intuitively, the former requires rather simple bookkeeping for maintaining the sum Λ_t discussed in Sec. 6, and it is significantly easier to implement and more efficient than Duchi et al.'s red-black tree-based implementation. Indeed, whereas a red-black tree requires at least a thousand of lines of code for its balancing, joining, and splitting operations, the efficient FOBOS updates require fewer than 100 lines of code.

We simulated updates to a weight vector $\boldsymbol{w}$ with a sparse gradient $\boldsymbol{g}_t^f$ for different dimensions d of $\boldsymbol{w} \in \mathbb{R}^d$ and different sparsity levels s for $\boldsymbol{g}_t^f$ with $\text{card}(\boldsymbol{g}_t^f) = s$. To do so, we generate a completely dense $\boldsymbol{w}$ whose ℓ_1-norm is at most a pre-specified value b and add a random vector $\boldsymbol{g}$ to $\boldsymbol{w}$ with s non-zeros. We then either project $\boldsymbol{w} + \boldsymbol{g}$ back to the constraint $\|\boldsymbol{w}\|_1 \leq b$ using the algorithm of Duchi et al. (2008) or perform a FOBOS update to $\boldsymbol{w} + \boldsymbol{g}$ using the algorithm in Sec. 6. We chose λ in the FOBOS update to give approximately the same sparsity as the constraint on b. In Table 1 we report timing results averaged over 100 independent experiments for different dimensions d of $\boldsymbol{w}$ and different cardinalities s of $\boldsymbol{g}$. Though theoretically the sparse FOBOS updates should have no dependence on the dimension d, we found that cache locality can play a factor when performing updates to larger dimensional vectors. Nonetheless, it is clear from the table that the efficient FOBOS step is on the order of ten to twenty times faster than its projected counterpart. Furthermore, the sparse FOBOS updates apply equally as well to mixed norm regularization, and while there are efficient algorithms for both projection to both ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ balls (Schmidt et al., 2009; Quattoni et al., 2009), they are more complicated than the FOBOS steps. Lastly, though it may be possible to extend the efficient data structures of Duchi et al. (2008) to the ℓ_1/ℓ_2 case, there is no known algorithm for efficient projections with sparse updates to an ℓ_1/ℓ_∞ constraint.

Dimension d	$s = 5000$		$s = 10000$		$s = 20000$	
	Project	FOBOS	Project	FOBOS	Project	FOBOS
$5 \cdot 10^4$	0.72	0.07	2.12	0.12	4.53	0.23
$2 \cdot 10^5$	0.80	0.10	2.06	0.16	5.09	0.34
$8 \cdot 10^5$	0.86	0.15	2.22	0.17	5.34	0.39
$3.2 \cdot 10^6$	1.07	0.13	2.75	0.16	6.31	0.52
$6.4 \cdot 10^6$	1.20	0.10	2.83	0.29	6.62	0.48

Table 1: Comparison of the average time (in hundredths of a second) required to compute projection of $\boldsymbol{w} + \boldsymbol{g}$ onto an ℓ_1-constraint to the analogous update required by the FOBOS step.

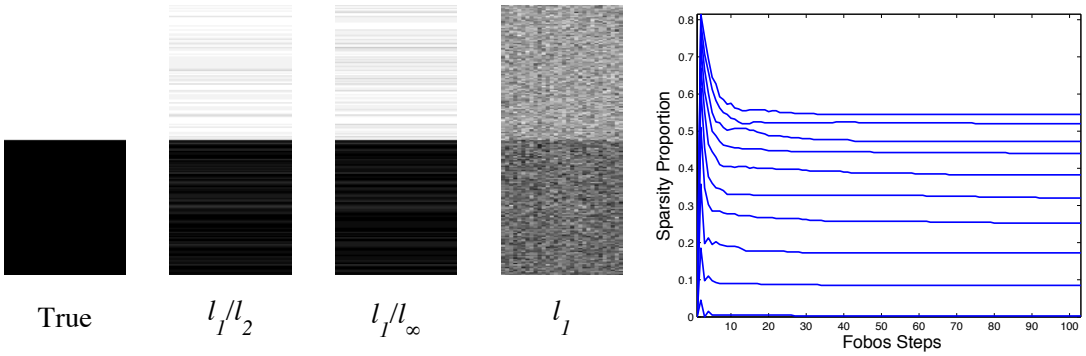

Figure 11: The sparsity patterns attained by FOBOS using mixed-norm and ℓ_1 regularization for a multiclass logistic regression problem.

7.6 Effects of Regularization

Our final experiments focus mostly on sparsity recovery of the different regularizers and their effects on test-set performance. While these are somewhat orthogonal to the previous experiments, we believe there is a relative paucity of investigation of the effects of mixed-norm regularization on classifier performance.

As a verification experiment of FOBOS with a mixed-norm regularizer, we solved a multiclass logistic regression problem whose objective is given in Eq. (35). To solve this task, we randomly generated a matrix W of dimension 200×30. The instances had $d = 200$ dimensions, the number of classes was $k = 30$, and we zeroed out the first 100 rows of W. We next generated $n = 1000$ samples $x_i \in \mathbb{R}^d$ with zero mean and unit variance. We set $y_i = \operatorname{argmax}_j \left\langle x_i, w^j \right\rangle$ and added 10% label noise. We then used FOBOS to find an approximate minimizer of the objective defined by Eq. (35).

To compare the effects of different regularizers, we minimized Eq. (35) using ℓ_1/ℓ_1, ℓ_1/ℓ_2, and ℓ_1/ℓ_∞ regularization. We repeated the experiment 20 times with different randomly selected W that had the same sparsity pattern. On the left side of Fig. 11 we illustrate the sparsity pattern (far left) of the weight vector that generated the data and color-coded maps of the sparsity patterns learned using FOBOS with the different regularization schemes. The colors indicate the fraction of times a weight of W was set to be zero. A white color indicates that the weight was found (or selected) to be zero in all of the experiments while a black color means that it was never zero. The regularization value λ was set so that the learned matrix W would have approximately 50% zero weights. From Fig. 11, we see that both ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ were capable of zeroing entire rows of parameters and often learned a sparsity pattern that was close to the sparsity pattern of the matrix that was used to generate the examples. The standard ℓ_1 regularizer (far right) performed very poorly in terms of structure recovery. In fact, the ℓ_1-regularizer did not yield a single row of zeros in any of the experiments, underscoring one of the merits of using mixed-norm regularization in structured problems. Quantitatively, 96.3% and 94.5% of the zero rows of W were correctly identified when using FOBOS with ℓ_1/ℓ_2 and ℓ_1/ℓ_∞ regularization, respectively. In contrast, not one of the zero rows of W was identified correctly as an all zero row using pure ℓ_1 regularization.

The right plot in Fig. 11 shows the sparsity levels (fraction of non-zero weights) achieved by FOBOS as a function of the number of iterations of the algorithm. Each line represents a different synthetic experiment as λ is modified to give more or less sparsity to the solution vector $w^\star$. The results demonstrate that FOBOS quickly selects the sparsity pattern of $w^\star$, and the level of sparsity persists throughout its execution. We found this sparsity pattern common to all problems we tested, including mixed-norm problems. This is not particularly surprising, as Hale et al. (2007) recently gave an analysis showing that after a finite number of iterations, FOBOS-like algorithms attain the sparsity of the true solution $w^\star$.

% Non-zero	ℓ_1 Test	ℓ_1/ℓ_2 Test	ℓ_1/ℓ_∞ Test
5	.43	**.29**	.40
10	.30	**.25**	.30
20	.26	**.22**	.26
40	.22	**.19**	.22

Table 2: LandSat classification error versus sparsity

% Non-zero	ℓ_1 Test	ℓ_1/ℓ_2 Test	ℓ_1/ℓ_∞ Test
5	.37	**.36**	.47
10	**.26**	**.26**	.31
20	**.15**	**.15**	.24
40	**.08**	**.08**	.16

Table 3: MNIST classification error versus sparsity

For comparison of the different regularization approaches, we report in Table 2 and Table 3 the test set error as a function of row sparsity of the learned matrix W. For the LandSat data, we see that using the block ℓ_1/ℓ_2 regularizer yields better performance for a given level of structural sparsity. However, on the MNIST data the ℓ_1 regularization and the ℓ_1/ℓ_2 achieve comparable performance for each level of structural sparsity. Moreover, for a given level of structural sparsity, the ℓ_1-regularized solution matrix W attains significantly higher overall sparsity, roughly 90% of the entries of each non-zero row are zero. The different performance on the different data sets might indicate that structural sparsity is effective only when the set of parameters indeed exhibit natural grouping.

For our final experiment, we show the power of FOBOS with mixed-norm regularization in the context of image compression. For this experiment, we represent each image as a set of patches where each patch is in turn represented as a 79 dimensional vector as described by Grangier and Bengio (2008). The goal is to jointly describe the set of patches by a single high-dimensional yet sparse set of dictionary features. Each of the dictionary terms is also in $\mathbb{R}^{79}$. Let $\boldsymbol{x}_j$ denote the j^{th} patch of an image with k patches to be compressed and $\boldsymbol{c}_i$ be the i^{th} dictionary vector from a dictionary of n vectors. The regularized objective is thus

$$\frac{1}{2}\sum_{j=1}^{k}\left\|\boldsymbol{x}_j-\sum_{i=1}^{n}w_{ij}\boldsymbol{c}_i\right\|_2^2+\lambda\sum_{i=1}^{n}\|\bar{\boldsymbol{w}}^i\|_q \ .$$

In our experiments, the number of dictionary vectors n was 1000 and the number of patches k was around 120 on average. We report results averaged over 100 different images. We experiment with the three settings for q we have used in prior experiments, namely $q \in \{1,2,\infty\}$. In Fig. 12 we report the average reconstruction error as a function of the fraction of dictionary vectors actually used. As one would expect, the mixed-norm regularizers (ℓ_1/ℓ_2 and ℓ_1/ℓ_∞) achieve lower reconstruction error as a function of dictionary sparsity than strict ℓ_1-regularization. The ℓ_1/ℓ_2-regularization also gives a slight, but significant, reconstruction improvement over ℓ_1/ℓ_∞-regularization. We hypothesize that this is related to the relative efficiency of ℓ_1/ℓ_∞ as a function of the geometry of the input space, as was theoretically discussed in Negahban and Wainwright (2008). Further investigation is required to shed more light into this type of phenomenon, and we leave it for future research.

8. Conclusions and Future Work

In this paper we analyzed a framework for online and batch convex optimization with a diverse class of regularization functions. We provided theoretical justification for a type of convex programming method we call

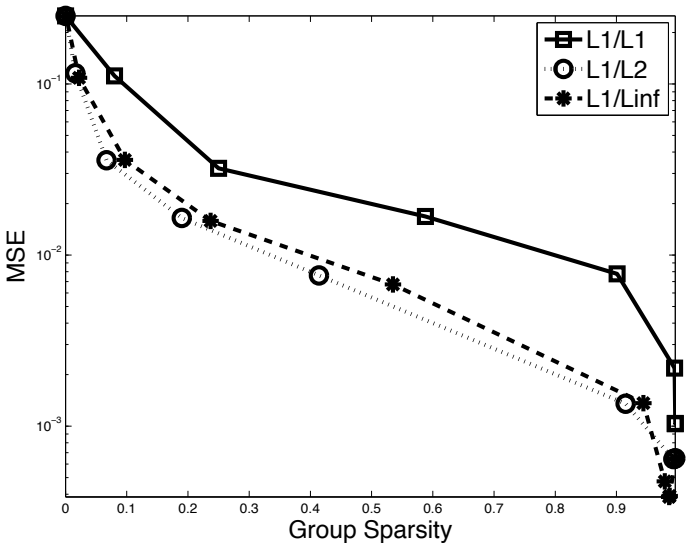

Figure 12: Image reconstruction error as a function of group sparsity.

FOBOS, which is known also as forward-backward splitting, iterative shrinkage and thresholding for the special case of ℓ_1-regularized problems, or SPARSA. Specifically, we described both offline convergence rates for arbitrary convex functions with regularization as well as regret bounds for online convex programming of regularized losses. Our derivation includes as a corollary the case of ℓ_1 regularization, which was concretely studied by Langford et al. (2008). Our approach provides a simple mechanism for solving online convex programs with many regularization functions, giving sparsity in parameters and different types of block or group regularization straightforwardly. Furthermore, the FOBOS framework is general and able to minimize any convex subdifferentialable function f so long as the forward looking step of Eq. (3) can be computed.

We have also provided a good deal of empirical evaluation of the method in comparison to other modern optimization methods for similar problems. Our practical experience suggests that for small to medium problems, SPARSA is effective and simple to implement (as opposed to more complicated coordinate descent methods), while for large scale problems, performing stochastic FOBOS is probably preferable. We have also shown that FOBOS is efficient for online learning with sparse data.

A few directions for further research suggest themselves, but we list here only two. The first is the question of whether we can modify the algorithm to work with arbitrary Bregman divergences of a function h instead of squared Euclidean distance, that is, we would like to form a generalized FOBOS update which is based on instantaneous optimization problems with Bregman divergences for convex differentiable h, where $B_h(\boldsymbol{u},\boldsymbol{v}) = h(\boldsymbol{u}) - h(\boldsymbol{v}) - \langle \nabla h(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle$. We assume the generalized update would, loosely speaking, be analogous to nonlinear projected subgradient methods and the mirror descent (see, e.g., Beck and Teboulle 2003). This might allow us to give bounds for our algorithms in terms of other dual norms, such as ℓ_1/ℓ_∞ norms on the gradients or diameter of the space, rather than simply ℓ_2. We believe the attainment and rate of sparsity when using stochastic gradient information, as suggested by the discussion of Fig. 10, merits deeper investigation that will be fruitful and interesting.

Acknowledgments

We would like to thank Sam Roweis and Samy Bengio for helpful discussions and Shai Shalev-Shwartz and Tong Zhang for useful feedback on earlier drafts. We also would like to thank the anonymous reviewers and

editor Yoav Freund for their constructive comments and guidance. A large portion of John Duchi's work was performed at Google.

Appendix A. Online Regret Proofs

Proof of Theorem 6 Looking at Lemma 1, we immediately see that if $\|\partial f\|$ and $\|\partial r\|$ are bounded by G,

$$f_t(\boldsymbol{w}_t) - f_t(\boldsymbol{w}^\star) + r(\boldsymbol{w}_{t+1}) - r(\boldsymbol{w}^\star) \leq \frac{1}{2\eta_t}\left(\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2\right) + \frac{7}{2}G^2\eta_t. \tag{36}$$

Now we use Eq. (36) to obtain that

$$\begin{aligned} R_{f+r}(T) &= \sum_{t=1}^{T}\left(f_t(\boldsymbol{w}_t) - f_t(\boldsymbol{w}^\star) + r(\boldsymbol{w}_t) - r(\boldsymbol{w}^\star)\right) + r(\boldsymbol{w}_{T+1}) - r(\boldsymbol{w}^\star) - r(\boldsymbol{w}_1) + r(\boldsymbol{w}^\star) \\ &\leq GD + \sum_{t=1}^{T}\frac{1}{2\eta_t}\left(\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2\right) + \frac{7G^2}{2}\sum_{t=1}^{T}\eta_t \end{aligned}$$

since $r(\boldsymbol{w}) \leq r(\mathbf{0}) + G\|\boldsymbol{w}\| \leq GD$. We can rewrite the above bound and see

$$\begin{aligned} R_{f+r}(T) &\leq GD + \frac{1}{2\eta_1}\|\boldsymbol{w}_1 - \boldsymbol{w}^\star\|^2 + \frac{1}{2}\sum_{t=2}^{T}\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + \frac{7G^2}{2}\sum_{t=1}^{T}\eta_t \\ &\leq GD + \frac{D^2}{2\eta_1} + \frac{D^2}{2}\sum_{t=2}^{T}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + \frac{7G^2}{2}\sum_{t=1}^{T}\eta_t \ , \end{aligned}$$

where we used again the bound on the distance of each $\boldsymbol{w}_t$ to $\boldsymbol{w}^\star$ for the last inequality. Lastly, we use the fact that the sum $\frac{1}{\eta_1} + \sum_{t=2}^{T}(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}})$ telescopes and get that

$$R_{f+r}(T) \ \leq \ GD + \frac{D^2}{2\eta_T} + \frac{7G^2}{2}\sum_{t=1}^{T}\eta_t \ .$$

Setting $\eta_t = c/\sqrt{t}$ and recognizing that $\sum_{t=1}^{T}\eta_t \leq 2c\sqrt{T}$ concludes the proof. ■

Proof of Theorem 8 The proof builds straightforwardly on Theorem 1 from Hazan et al. (2006) and our proof of Theorem 6. We sum Eq. (17) from $t = 1$ to T and get

$$\begin{aligned} R_{f+r}(T) &\leq \sum_{t=1}^{T}\left(\left\langle \boldsymbol{g}_t^f + \boldsymbol{g}_t^r, \boldsymbol{w}_t - \boldsymbol{w}^\star\right\rangle - \frac{H}{2}\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2\right) \\ &\leq 2GD + \frac{1}{2}\sum_{t=1}^{T-1}\left(\frac{1}{\eta_t}\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2 - \frac{1}{\eta_t}\|\boldsymbol{w}_{t+1} - \boldsymbol{w}^\star\|^2 - H\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2\right) + \frac{7G^2}{2}\sum_{t=1}^{T-1}\eta_t \\ &\leq 2GD + \frac{1}{2}\sum_{t=2}^{T-1}\|\boldsymbol{w}_t - \boldsymbol{w}^\star\|^2\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - H\right) + \frac{1}{\eta_1}\|\boldsymbol{w}_1 - \boldsymbol{w}^\star\|^2 + \frac{7G^2}{2}\sum_{t=1}^{T-1}\eta_t. \end{aligned}$$

The second inequality follows from Eq. (36) and the third inequality from a rearrangement of the sum and removal of the negative term $(1/\eta_{T-1})\|\boldsymbol{w}_T - \boldsymbol{w}^\star\|^2$. Taking $\eta_t = \frac{1}{Ht}$, we see that $1/\eta_t - 1/\eta_{t-1} - H = Ht - H(t-1) - H = 0$, so we can bound the regret by

$$R_{f+r}(T) \leq 2GD + HD^2 + \frac{7G^2}{2}\sum_{t=1}^{T-1}\frac{1}{Ht} \leq 2GD + HD^2 + \frac{7G^2}{2H}(1 + \log T) = O\left(\frac{G^2}{H}\log T\right) \ .$$

■

Proof of Lemma 9 The triangle inequality implies that

$$\left\|\sum_{t=1}^{T} \partial f_t(\boldsymbol{w}_t)\right\| \leq \sum_{t=1}^{T} \|\partial f_t(\boldsymbol{w}_t)\| \leq TG \ .$$

Let $\boldsymbol{g}_t^\star \in \partial f_t(\boldsymbol{w}^\star)$ be such that

$$\mathbf{0} = \sum_{t=1}^{T} \boldsymbol{g}_t^\star + T\lambda \boldsymbol{w}^\star \in \sum_{t=1}^{T} \partial f_t(\boldsymbol{w}^\star) + T\partial r(\boldsymbol{w}^\star)$$

so $\|\boldsymbol{w}^\star\| = \|\sum_{t=1}^{T} \boldsymbol{g}_t^\star\|/(T\lambda) \leq G/\lambda$.

For the second part, assume that $\boldsymbol{w}_0 = \mathbf{0}$, and for our induction that $\boldsymbol{w}_t$ satisfies $\|\boldsymbol{w}_t\| \leq G/\lambda$. Then computing the FOBOS update from Eq. (20),

$$\|\boldsymbol{w}_{t+1}\| = \frac{\|\boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f\|}{1+\lambda\eta_t} \leq \frac{\|\boldsymbol{w}_t\| + \eta_t\|\boldsymbol{g}_t^f\|}{1+\lambda\eta_t} \leq \frac{G/\lambda + \eta_t G}{1+\lambda\eta_t} = \frac{G(1+\lambda\eta_t)}{\lambda(1+\lambda\eta_t)} \ .$$

■

Appendix B. Update for Berhu Regularization

Recalling the Berhu regularizer and combining it with Eq. (18) for one variable, we see that we want to minimize

$$\frac{1}{2}(w-v)^2 + \tilde{\lambda} b(w) = \frac{1}{2}(w-v)^2 + \tilde{\lambda}\left[|w|\,[\![|w| \leq \gamma]\!] + \frac{w^2+\gamma^2}{2\gamma}\,[\![|w| > \gamma]\!]\right].$$

First, if $|v| \leq \tilde{\lambda}$, then exactly reasoning to that for minimization of the ℓ_1-regularized minimization step implies that the optimal solution is $w = 0$.

When $|v| > \tilde{\lambda}$, there are two remaining cases to check. Let us assume without loss of generality that $v > \tilde{\lambda}$. It is immediate to verify that $w \geq 0$ at the optimum. Now, suppose that $v - \tilde{\lambda} \leq \gamma$. Taking $w = v - \tilde{\lambda} \leq \gamma$ (so that $w > 0$) gives us that $\partial b(w) = \{\tilde{\lambda}\}$. Thus, the subgradient set of our objective contains a single element, $w - v + \tilde{\lambda}\partial|w| = v - \tilde{\lambda} - v + \tilde{\lambda}1 = 0$. Therefore, when $v - \tilde{\lambda} \leq \gamma$ the optimal value of w is $v - \tilde{\lambda}$. The last case we need to examine is when $v - \tilde{\lambda} > \gamma$, which we as we show shortly puts the solution $w^\star$ in the ℓ_2^2 realm of $b(w)$. By choosing $w = \frac{v}{1+\frac{\tilde{\lambda}}{\gamma}}$ we get that,

$$w = \frac{v}{1+\frac{\tilde{\lambda}}{\gamma}} = \frac{v\gamma}{\gamma+\tilde{\lambda}} > \frac{(\gamma+\tilde{\lambda})\gamma}{\gamma+\tilde{\lambda}} = \gamma \ .$$

Therefore, $w > \gamma$ and thus w is in the ℓ_2^2 region of the Berhu penalty $b(w)$. Furthermore, for this choice of w the derivative of the penalty is

$$w - v + \tilde{\lambda}\frac{w}{\gamma} = \frac{v\gamma}{\gamma+\tilde{\lambda}} - v + \tilde{\lambda}\frac{v\gamma}{\gamma(\gamma+\tilde{\lambda})} = \frac{v\gamma}{\gamma+\tilde{\lambda}} - \frac{v(\gamma+\tilde{\lambda})}{\gamma+\tilde{\lambda}} + \tilde{\lambda}\frac{v}{\gamma+\tilde{\lambda}} = 0.$$

Combining the above results, inserting the conditions on the sign, and expanding $v = w_{t+\frac{1}{2},j}$ gives Eq. (26).

Appendix C. Fast Convergence Rate for Smooth Objectives

In this appendix, we describe an analysis of FOBOS which yields an $O(1/T)$ rate of convergence when f has Lipschitz-continuous gradient. Our analysis is by no means new. It is a distilled and simplified adaptation of the analysis of Nesterov (2007) to our setting.

Throughout the appendix we assume that $\nabla f(\boldsymbol{w})$ is Lipschitz continuous with a constant L, that is, $\|\nabla f(\boldsymbol{w}) - \nabla f(\boldsymbol{v})\| \leq L\|\boldsymbol{w} - \boldsymbol{v}\|$. The fundamental theorem of calculus then readily implies that (Nesterov, 2004, Lemma 1.2.3)

$$|f(\boldsymbol{w}) - f(\boldsymbol{v}) - \langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle| \leq \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2 \ . \tag{37}$$

To see that Eq. (37) holds, add and subtract $\langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle$ to note that

$$\begin{aligned} f(\boldsymbol{w}) - f(\boldsymbol{v}) &= \int_0^1 \langle \nabla f(\boldsymbol{v} + t(\boldsymbol{w} - \boldsymbol{v})), \boldsymbol{w} - \boldsymbol{v}\rangle \, dt \\ &= \langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle + \int_0^1 \langle \nabla f(\boldsymbol{v} + t(\boldsymbol{w} - \boldsymbol{v})) - \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle \, dt \end{aligned}$$

which, by using Cauchy-Shwartz inequality, yields

$$\begin{aligned} |f(\boldsymbol{w}) - f(\boldsymbol{v}) - \langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle| \ &\leq \ \int_0^1 |\langle \nabla f(\boldsymbol{v} + t(\boldsymbol{w} - \boldsymbol{v})) - \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle| \, dt \\ \leq \ \int_0^1 \|\nabla f(\boldsymbol{v} + t(\boldsymbol{w} - \boldsymbol{v})) - \nabla f(\boldsymbol{v})\| \|\boldsymbol{w} - \boldsymbol{v}\| dt \ &\leq \ \int_0^1 tL\|\boldsymbol{w} - \boldsymbol{v}\|^2 dt \ = \ \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2 \ . \end{aligned}$$

For the remainder of this section, we assume that $f(\boldsymbol{w}) + r(\boldsymbol{w})$ is coercive, so that as $\|\boldsymbol{w}\| \to \infty$, $f(\boldsymbol{w}) + r(\boldsymbol{w}) \to \infty$. We thus have that the level sets of $f(\boldsymbol{w}) + r(\boldsymbol{w})$ are bounded: $\|\boldsymbol{w} - \boldsymbol{w}^\star\| \leq D$ for all $\boldsymbol{w}$ such that $f(\boldsymbol{w}) + r(\boldsymbol{w}) \leq f(\boldsymbol{0}) + r(\boldsymbol{0})$. Consider the "composite gradient mapping" (Nesterov, 2007)

$$m(\boldsymbol{v}, \boldsymbol{w}) = f(\boldsymbol{v}) + \langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle + \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2 + r(\boldsymbol{w}) \ . \tag{38}$$

Before proceeding with the proof of fast convergence rate, we would like to underscore the equivalence of the FOBOS update and the composite gradient mapping. Formally, minimizing $m(\boldsymbol{v}, \boldsymbol{w})$ with respect to $\boldsymbol{w}$ is completely equivalent to taking a FOBOS step with $\eta = 1/L$ and $\boldsymbol{v} = \boldsymbol{w}_t$. To obtain the FOBOS update from Eq. (38) we simply need to divide $m(\boldsymbol{v}, \boldsymbol{w})$ by $L = 1/\eta$, omit terms that solely depend on $\boldsymbol{v} = \boldsymbol{w}_t$, and use the fact that $\boldsymbol{w}_{t+\frac{1}{2}} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t^f = \boldsymbol{v} - \nabla f(\boldsymbol{v})/L$.

For notational convenience, let $\phi(\boldsymbol{w}) = f(\boldsymbol{w}) + r(\boldsymbol{w})$. Denote by $\boldsymbol{w}^+$ the vector minimizing $m(\boldsymbol{v}, \boldsymbol{w})$. Then from Eq. (37) we get that

$$\phi(\boldsymbol{w}^+) = f(\boldsymbol{w}^+) + r(\boldsymbol{w}^+) \leq f(\boldsymbol{v}) + \left\langle \nabla f(\boldsymbol{v}), \boldsymbol{w}^+ - \boldsymbol{v}\right\rangle + \frac{L}{2}\|\boldsymbol{w}^+ - \boldsymbol{v}\|^2 + r(\boldsymbol{w}^+) = \inf_{\boldsymbol{w}} m(\boldsymbol{v}, \boldsymbol{w}). \tag{39}$$

Further, because $f(\boldsymbol{v}) + \langle \nabla f(\boldsymbol{v}), \boldsymbol{w} - \boldsymbol{v}\rangle \leq f(\boldsymbol{w})$ for all $\boldsymbol{w}$, we have

$$\inf_{\boldsymbol{w}} m(\boldsymbol{v}, \boldsymbol{w}) \leq \inf_{\boldsymbol{w}} \left[f(\boldsymbol{w}) + \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2 + r(\boldsymbol{w}) \right] = \inf_{\boldsymbol{w}} \left[\phi(\boldsymbol{w}) + \frac{L}{2}\|\boldsymbol{w} - \boldsymbol{v}\|^2 \right] \ . \tag{40}$$

Now we consider the change in function value from $\boldsymbol{w}_t$ to $\boldsymbol{w}_{t+1}$ for the FOBOS update with $\eta = 1/L$. To do this, we take an arbitrary optimal point $\boldsymbol{w}^\star$ and restrict $\boldsymbol{w}_{t+1}$ to lie on the line between $\boldsymbol{w}_t$ and $\boldsymbol{w}^\star$, which constrains the set of infimum values above and allows us to carefully control them. With this construction,

along with Eqs. (39) and (40) we get that

$$\begin{aligned}
\phi(\boldsymbol{w}_{t+1}) &\leq \inf_{\boldsymbol{w}} \left[\phi(\boldsymbol{w}) + \frac{L}{2} \|\boldsymbol{w} - \boldsymbol{w}_t\|^2 \right] \\
&\leq \inf_{\alpha \in [0,1]} \left[\phi(\alpha \boldsymbol{w}^\star + (1-\alpha)\boldsymbol{w}_t) + \frac{L}{2} \|\alpha \boldsymbol{w}^\star + (1-\alpha)\boldsymbol{w}_t - \boldsymbol{w}_t\|^2 \right] \\
&\leq \inf_{\alpha \in [0,1]} \left[\alpha \phi(\boldsymbol{w}^\star) + (1-\alpha)\phi(\boldsymbol{w}_t) + \frac{\alpha^2 L}{2} \|\boldsymbol{w}^\star - \boldsymbol{w}_t\|^2 \right] . \qquad (41)
\end{aligned}$$

The bound in Eq. (41) follows due to the convexity of ϕ. One immediate consequence of Eq. (41) is that $\phi(\boldsymbol{w}_{t+1}) \leq \phi(\boldsymbol{w}_t)$, since at $\alpha = 0$ we obtain the same objective for ϕ. Thus, all iterates of the method satisfy $\|\boldsymbol{w}_t - \boldsymbol{w}^\star\| \leq D$. We therefore can distill the bound to be

$$\phi(\boldsymbol{w}_{t+1}) \leq \inf_{\alpha \in [0,1]} \left[\phi(\boldsymbol{w}_t) + \alpha \left(\phi(\boldsymbol{w}^\star) - \phi(\boldsymbol{w}_t) \right) + \alpha^2 \frac{LD^2}{2} \right] .$$

The argument of the infimum of the equation above is a quadratic equation in α. We need to analyze two possible cases for the optimal solution. In the first case when $\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star) > LD^2$, the optimal value of α is 1 and $\phi(\boldsymbol{w}_{t+1}) \leq \phi(\boldsymbol{w}^\star) + LD^2/2$. Therefore, we will never encounter again this case in future iterations. The second occurs when $\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star) \leq LD^2$, so we have $\alpha = (\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star))/LD^2 \in [0,1]$, which yields

$$\phi(\boldsymbol{w}_{t+1}) \leq \phi(\boldsymbol{w}_t) - \frac{\left(\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star)\right)^2}{2LD^2} . \qquad (42)$$

To obtain the form of the convergence rate let us define the inverse residual value $\rho_t = 1/(\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star))$. By analysing the rate at which ρ_t tends to infinity we obtain our desired convergence rate. From the definition of ρ_t and the bound of Eq. (42) we get that

$$\begin{aligned}
\rho_{t+1} - \rho_t &= \frac{1}{\phi(\boldsymbol{w}_{t+1}) - \phi(\boldsymbol{w}^\star)} - \frac{1}{\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star)} = \frac{\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star) - \phi(\boldsymbol{w}_{t+1}) + \phi(\boldsymbol{w}^\star)}{(\phi(\boldsymbol{w}_{t+1}) - \phi(\boldsymbol{w}^\star))(\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star))} \\
&= \rho_t \rho_{t+1} (\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}_{t+1})) \geq \rho_t \rho_{t+1} \frac{\left(\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star)\right)^2}{2LD^2} = \frac{\rho_t \rho_{t+1}}{2\rho_t^2 LD^2} \geq \frac{1}{LD^2} ,
\end{aligned}$$

where the last inequality is due to the fact that $\rho_{t+1} \geq \rho_t$ and therefore $\rho_t \rho_{t+1}/\rho_t^2 \geq 1$. Summing the differences $\rho_{t+1} - \rho_t$ from $t = 0$ through $T-1$, we get $\rho_T \geq T/2LD^2$. Thus, for $t \geq 1$ we have

$$\phi(\boldsymbol{w}_t) - \phi(\boldsymbol{w}^\star) = 1/\rho_t \leq \frac{2LD^2}{t} .$$

To recap, by setting $\eta = 1/L$ while relaying on the fact that f has Lipschitz continuous gradient with constant L, we obtain a $1/T$ rate of convergence

$$f(\boldsymbol{w}_T) + r(\boldsymbol{w}_T) \leq f(\boldsymbol{w}^\star) + r(\boldsymbol{w}^\star) + \frac{2LD^2}{T} .$$

References

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

D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

J. Blitzer, M. Dredze, and F. Pereira. Biographies, Bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. In *Association for Computational Linguistics*, 2007.

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

G. Chen and R. T. Rockafellar. Convergence rates in forward-backward splitting. *SIAM Journal on Optimization*, 7(2), 1997.

P. Combettes and V. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200, 2005.

I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communication on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.

I. Daubechies, M. Fornasier, and I. Loris. Accelerated projected gradient method for linear inverse problems with sparsity constraints. *Fourier Analysis and Applications*, 14(5):764–792, 2008.

D. L. Donoho. De-noising via soft thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, 2008.

D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1371–1384, 2008.

E. Hale, W. Yin, and Y. Zhang. A fixed-point continuation method for ℓ_1-regularized minimization with applications to compressed sensing. Technical Report TR07-07, Rice University Department of Computational and Applied Mathematics, July 2007.

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.

K. Koh, S.J. Kim, and S. Boyd. An interior-point method for large-scale ℓ_1-regularized logistic regression. *Journal of Machine Learning Research*, 8:1519–1555, 2007.

J. Langford, L. Li, and T. Zhang. Sparse online learning via truncated gradient. In *Advances in Neural Information Processing Systems 22*, 2008.

P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16:964–979, 1979.

L. Meier, S. van de Geer, and P. Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society Series B*, 70(1):53–71, 2008.

N. Meinshausen and P. Bühlmann. High dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 34:1436–1462, 2006.

S. Negahban and M. Wainwright. Phase transitions for high-dimensional joint support recovery. In *Advances in Neural Information Processing Systems 22*, 2008.

Y. Nesterov. *Introductory Lectures on Convex Optimization*. Kluwer Academic Publishers, 2004.

Y. Nesterov. Gradient methods for minimizing composite objective function. Technical Report 76, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (UCL), 2007.

G. Obozinski, B. Taskar, and M. Jordan. Joint covariate selection for grouped classification. Technical Report 743, Dept. of Statistics, University of California Berkeley, 2007.

G. Obozinski, M. Wainwright, and M. Jordan. High-dimensional union support recovery in multivariate regression. In *Advances in Neural Information Processing Systems 22*, 2008.

Art Owen. A robust hybrid of lasso and ridge regression. Technical report, Stanford University, 2006.

A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for L1,infinity regularization. In *Proceedings of the 26th International Conference on Machine Learning*, 2009.

R.T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

M. Schmidt, E. van den Berg, M. Friedlander, and K. Murphy. Optimizing costly functions with simple constraints: a limited-memory projected quasi-Newton method. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, 2009.

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

S. Shalev-Shwartz and A. Tewari. Stochastic methods for ℓ_1-regularized loss minimization. In *Proceedings of the 26th International Conference on Machine Learning*, 2009.

S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

D. Spiegelhalter and C. Taylor. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, 1994.

P. Tseng. A modified forward backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization*, 38:431–446, 2000.

P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming Series B*, 117:387–423, 2007.

S. Wright, R. Nowak, and M. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.

P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7: 2541–2567, 2006.

P. Zhao, G. Rocha, and B. Yu. Grouped and hierarchical model selection through composite absolute penalties. Technical Report 703, Statistics Department, University of California Berkeley, 2006.

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