Online Learning with Maximal No-Regret ℓ_1 Regularization

Daniel Golovin
Google, Inc.
dgg@google.com

H. Brendan McMahan
Google, Inc.
mcmahan@google.com

D. Sculley
Google, Inc.
dsculley@google.com

Abstract

This paper asks: what is the maximum amount of ℓ_1 regularization can we apply while preserving cumulative regret in online learning? We show that scaling the ℓ_1 penalty by $\sqrt{T}$, rather than a fixed ℓ_1 or a linearly increasing ℓ_1 penalty, provides the maximum possible ℓ_1 regularization and corresponding model sparsity while preserving cumulative regret bounds, and provide supporting empirical results.

1 Introduction

In this paper, we consider the general online convex optimization problem, which naturally encompasses many important learning tasks [13, 6]. We consider a sequence of rounds, where on each round t our algorithm makes a prediction via model $x_t \in \mathbb{R}^n$, nature (which may be adversarial) reveals a convex loss function f_t, and then we pay the cost $f_t(x_t)$ and update our model. The primary goal here is to minimize *regret*, defined with respect to a comparator model x^*. The standard definition is: $\text{Regret}(x^*) := \sum_{t=1}^{T}(f_t(x_t) - f_t(x^*))$.

An important secondary goal is to produce *sparse models*, that is, models that have few non-zero entries [5, 12]. Inducing sparsity can help match prior beliefs about model structure, is a useful a method of feature selection, and leads to computational efficiency in massive-scale settings. Perhaps the most popular method to induce model sparsity is to penalize the ℓ_1 norm of the coefficients [14]. Instead of running our online algorithm on $f_t(x)$, we can consider $f_t(x) + \alpha_t\|x\|_1$, where α_t is the strength of the ℓ_1 penalty associated with the t^{th} training example.

In this paper, we focus on the Follow-The-Regularized-Leader (FTRL) family of algorithms. Let $\psi_{a:b} := \sum_{i=a}^{b} \psi_i$ for a sequence of variables or functions $\{\psi_i\}_{i\geq 0}$, and recall that such algorithms select points

$$x_{t+1} = \arg\min_{x\in\mathcal{X}} g_{1:t} \cdot x + r_{0:t}(x). \tag{1}$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is a convex feasible set, $g_t = \nabla f_t(x_t)$ with f_t the loss function on round t, r_t is the incremental regularization penalty added on round t, and $x_t \in \mathbb{R}^n$ are the model parameters chosen by the algorithm on round t. Both Mirror Descent [2, 4] and Regularized Dual Averaging (RDA) [15] can effectively incorporate an ℓ_1 penalty (or other non-smooth convex penalties), but Follow-The-Regularized-Leader (FTRL) algorithms like Dual Averaging generally produce better sparsity vs. accuracy tradeoffs [15, 8]. Both RDA and FTRL-Proximal [8] (a close relative of Mirror Descent) can be expressed as the update of Eq. (1) with suitable choices of $r_{0:t}$.

While these algorithms can effectively incorporate ℓ_1 penalties, the analysis of Xiao [15] and Duchi and Singer [2] assumes a fixed ℓ_1 penalty is associated with every training example, implying the cumulative strength of the ℓ_1 regularization increases linearly in the number of examples (that is $\alpha_t = \lambda$ for all t). We will show this approach must necessarily lead to over-regularization in a truly online setting. For FTRL algorithms including RDA, it is also fairly straightforward to analyze the case of a constant cumulative ℓ_1 penalty that stays fixed as T increases (meaning $\alpha_1 = \lambda$ and $\alpha_t = 0$ for $t > 1$). However, we will show this approach leads to under-regularization in that even features that emit random noise eventually tend to get non-zero coefficients in the model.

The natural question is then how to set the incremental ℓ_1 regularization parameters α_t so as to both achieve low regret but also continue to produce sparse models even as T grows arbitrarily. We will show that choosing the α_t so that on round t the cumulative ℓ_1 penalty is $\Theta\left(\sqrt{t}\right)$ is essentially the optimal choice.

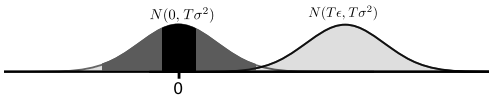

Figure 1: Distribution of gradient sums for an uninformative ($\mu = 0$) and an informative ($\mu = T\epsilon$) feature after T rounds in the thought experiment, and the regions where different ℓ_1 regularization schemes suppress the corresponding coefficient. Constant cumulative ℓ_1 regularization is in black, $\sqrt{T}$ is in dark gray, and linear regularization (which covers both distributions, suppressing everything) is in light gray.

2 Intuitions

To build intuitions for the formal results presented later, first consider the simple case of online linear optimization in the stochastic setting. A key observation is that FTRL and RDA algorithms have the property that the coefficient value $x_{t,i}$ is a function of the sum of loss gradients experienced by that feature [8, 15]. For example, the update rule for RDA is:

$$x_{t+1} = \operatorname{argmin}\left(g_{1:t}x + \frac{1}{2}\sigma_{1:t}||x||_2^2 + \alpha_{1:t}||x||_1\right) = \begin{cases} 0 & \text{if } |g_{1:t}| \leq \alpha_{1:t} \\ -(g_{1:t} - \operatorname{sgn}(g_{1:t})\alpha_{1:t})/\sigma_{1:t} & \text{otherwise} \end{cases} \quad (2)$$

In particular, a feature with gradient sum of zero should have a zero coefficient value.

We will show that, under some very mild assumptions, with high probability the absolute gradient sum for a completely random feature (i.e., pure noise) will be $\Theta(\sqrt{T})$, which suggests that we should discount features with gradient sums of $\mathcal{O}(\sqrt{T})$ in order to suppress noise, but should not be any more aggressive than that lest we also suppress genuinely informative features.

A Simple Thought Experiment We begin with a simple thought experiment, illustrated in Fig. 1. Suppose the model only has one coefficient, (i.e., $n = 1$), and on each round t the loss gradient g_t is drawn i.i.d. from $N(\mu, \sigma^2)$. We consider two scenarios corresponding to the (single) feature being useless noise (i.e., $\mu = 0$) or informative (i.e., $\mu = \epsilon > 0$). After T rounds, the gradient sum $g_{1:t}$ is distributed as shown in Fig. 1 for the two cases. The black, dark gray, and light gray regions depict the probability mass in which the coefficient will be set to zero for constant, $\Theta\left(\sqrt{T}\right)$, and $\Theta\left(T\right)$ cumulative ℓ_1 regularization, respectively. Note the constant cumulative ℓ_1 regularization ($\alpha_{1:T} = \mathcal{O}(1)$, black) is not strong enough to suppress the feature in the useless noise case. Conversely, linear cumulative regularization ($\alpha_{1:T} = cT$, light gray) is too strong if $c > \epsilon$, suppressing even the informative feature. On the other hand, cumulative square root regularization ($\alpha_{1:T} = c\sqrt{T}$, dark gray) will eventually suppress the useless noise feature if c is sufficiently large (e.g., $c = 6\sigma$), but will not suppress the informative one once T is sufficiently large (e.g., once $\epsilon T \geq (c + 6\sigma)\sqrt{T}$). Hence, for a wide range of leading constants c, the situation illustrated in Fig. 1 will occur after sufficiently many rounds.

Extending the Thought Experiment We next consider what happens when there are multiple dimensions, and the distribution on gradients is more general. We assume there is an unknown distribution $\mathcal{G}$ on gradient vectors in $[-1, 1]^n$ and the loss is simply $f_t(x) = g_t \cdot x$ where each g_t is drawn i.i.d. from $\mathcal{G}$. Further suppose that there is a fixed coordinate, say n, such that $g_{t,n}$ has zero mean and positive variance σ^2 for all t. Note $g_{t,n}$ need *not* be independent of $g_{t,i}$ for $i \neq n$. Since $\mathbf{E}\left[g_{t,n}\right] = 0$ by assumption, playing $x_{t,n} = 0$ on round t (i.e., before seeing $g_{t,n}$) minimizes our expected loss, regardless of the values of the other coefficients.

Next, consider the gradient sum for coordinate n, that is, $g_{1:T,n}$. Note it is distributed as a sum of T independent random variables which have zero mean and lie in $[-1, 1]$. The Azuma–Hoeffding inequality then implies that for all $\beta \geq 0$,

$$\mathbf{Pr}\left[|g_{1:T,n}| > \beta\sqrt{T}\right] \leq 2\exp\left(-\beta/2\right).$$

Hence, for any constant $\delta > 0$ there is a constant $c(\delta)$ such that using ℓ_1 regularization $c(\delta)\sqrt{T}$ is sufficient to suppress such spurious features (i.e., ensure $x_{T,n} = 0$) with probability $1 - \delta$.

Finally, we cannot expect to suppress such spurious features with far less regularization. The (classical) Central Limit Theorem indicates that as $T \to \infty$, the distribution $g_{1:T}/\sqrt{T}$ will converge almost surely to $N(0, \sigma^2)$. Hence the standard deviation of $g_{1:T}$ is $\Theta(\sqrt{T})$, and therefore if we use a regularization scaling function $\lambda(t) = o(\sqrt{t})$ then the probability that we play $x_{t,n} \neq 0$ will approach one in the limit $t \to \infty$.

3 Regret Analysis for FTRL Algorithms

Our formal analysis builds on the general results for the analysis of adaptive online algorithms of McMahan [9]. We focus on RDA and FTRL-Proximal, which can both be expressed as instantiations of the general update Eq. (1). RDA (as in Eq. (2)) fits this form with $r_t(x) = \sigma_t\|x\|_2^2 + \alpha_t\|x\|_1$, and the FTRL-Proximal algorithm with ℓ_1 regularization (with regularization parameters σ_t and α_t) uses $r_t(x) = \sigma_t\|x - x_t\|_2^2 + \alpha_t\|x\|_1$. The parameters $\sigma_t \geq 0$ determine a learning rate for the round t update of the form $\eta_t = \frac{1}{\sigma_{1:t}}$.

Using [9], we can prove the following upper bounds on the regret of these algorithms.

Theorem 1. *Suppose $\|x\|_2 \leq R_2$ for all $x \in \mathcal{X}$ and $\|g_t\|_2 \leq G_2$ for all g_t. Then, the FTRL-Proximal algorithm with ℓ_1 regularization has*

$$\text{Regret}(x^*) \leq 2\sqrt{2}R_2G_2\sqrt{T} + \alpha_{1:t}\|x^*\|_1. \tag{3}$$

when we use learning rates $\eta_t = \frac{\sqrt{2}R_2}{G_2\sqrt{t}}$ (equivalently, when we set σ_t such that $\frac{1}{\sigma_{1:t}} = \frac{\sqrt{2}R_2}{G_2\sqrt{t}}$). Under the same conditions, RDA with learning rates $\eta_t = \frac{1}{\sigma_{0:t}} = \frac{R_2}{\sqrt{2}G_2\sqrt{t+1}}$ achieves

$$\text{Regret}(x^*) \leq \sqrt{2}R_2G_2\sqrt{T} + \alpha_{1:t}\|x^*\|_1.$$

Proof. The inclusion of the ℓ_1 penalties in the regularizers $r_{0:t}$ does not change their strong convexity with respect to the ℓ_2-norm. These results are then a straightforward consequence of McMahan [9][Thms. 1 and 2]. □

Suppose for all t, $g_t^2 \leq G_2^2$ for a constant $G_2 > 0$, and we choose the α_t so that $\alpha_{1:T} = \sqrt{T}$. Then, the bound of Eq. (3) becomes $2\sqrt{2}R_2G_2\sqrt{T} + \sqrt{T}\|x^*\|_1$. Since $\|x^*\|_1$ as a constant, adding ℓ_1 regularization at this rate only costs us constant factors in the regret bound. On the other hand, if $\alpha_{1:T}$ is $\Omega(T^{\frac{1}{2}+\kappa})$ for $\kappa > 0$, then of course we no longer have an $\mathcal{O}(\sqrt{T})$ regret bound. This is not simply a consequence of this particular regret analysis, as the following lower bound shows. (The proof of this appears in the full length version of this paper, but is omitted here for space reasons.)

Theorem 2. *Consider any algorithm that against a series of convex functions f_t achieves a regret bound that is $\mathcal{O}(\sqrt{T})$ for any comparator x^* with $\|x^*\| \leq 1$ for any problem where the f_t satisfy $\|\nabla f_t(x_t)\|_2 \leq 1$. Then, for any $\kappa \in (0, \frac{1}{2}]$ there exists a loss sequence $\{f_t\}_{t\geq 1}$ such that when we run the algorithm on loss functions $h_t(x) = f_t(x) + \alpha_t\|x\|_1$, if $\alpha_{1:T} = \Omega(T^{\frac{1}{2}+\kappa})$ for all T, then regret against $f_1, \ldots, f_T$ must be $\Omega(T^{\frac{1}{2}+\kappa})$.*

These results demonstrate that choosing $\alpha_{1:t} = \mathcal{O}(\sqrt{t})$ is the maximal amount of regularization that can be added in the online setting without incurring asymptotically worse regret guarantees. However, we can do better with a data-dependent argument, as we show in the next section.

3.1 Per-coordinate learning rates and regularization

In the worst case we have $\|g_t\|_2 = G_2$ for all t, and the above bounds are essentially tight. However, in sparse settings, the above bound can be quite loose. Tighter data-dependent bounds could be obtained using per-coordinate learning rates in style of AdaGrad [3, 10].

The same arguments that justify using a per-coordinate learning rate also suggest using per-coordinate ℓ_1 regularization: we need to distinguish common but uninformative features from rare but highly informative ones. The problem with a global penalty $\alpha_{1:t}\|x\|_1$ is that it is likely that the absolute gradient sum $|g_{1:t,i}|$ will be larger for an uninformative feature that has, say, $g_{t,i}$ uniformly at random from $\{-1, 1\}$ on every round than for an informative feature that only rarely occurs with non-zero value. We would like to allow the latter a non-zero coefficient in our model, while still suppressing the former.

To address this, we can scale the ℓ_1 penalty on a per-coordinate basis: instead of a penalty $\alpha_{1:t}\|x\|_1$, we use a penalty

$$\Psi(x) = \sum_{i=1}^{n} \alpha_{1:t,i}|x_i|.$$

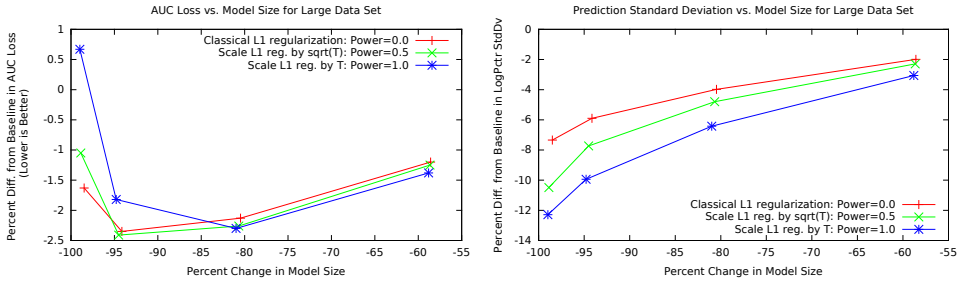

Figure 2: Results for the large pCTR private data set experiments.

Letting $c_{t,i} = \sum_{s=1}^{t} \mathbb{I}(g_{t,i} \neq 0)$, we choose $\alpha_{1:t,i} \approx \sqrt{c_{t,i}}$, that is, the penalty for coordinate i scales not with the global count t, but with the number of times we have seen a non-zero gradient for that coordinate (e.g., the number of times the feature has occurred). Following a simple extension of the analysis from Theorem 1, suppose $|g_{t,i}| \leq G_\infty$ for all i and t. Then, we have

$$\text{Regret}(x^*) \leq \sum_{i=1}^{n} \left(2\sqrt{2} R_2 \sqrt{c_{T,i}} G_\infty^2 + \sqrt{c_{T,i}} |x_i^*| \right), \tag{4}$$

and so we now balance the cost of the regularization and the intrinsic regret of the problem on a per-coordinate basis (both scale as $\sqrt{c_{T,i}}$). We show in the experiments section that this approach is highly effective.

4 Experimental Results

We ran two sets of experiments on large-scale sparse data, one public and one private. The publicly available malicious URL dataset [7] contains about 2.4×10^6 examples with 3.2×10^6 features. The private data set is a proprietary data set around ad click through prediction (pCTR) with more than a hundred billion examples and billions of features, a natural setting for sparse learning [11].

We ran our algorithms in an online fashion using progressive validation [1] to assess predictive performance and prediction variance. We assess model accuracy using the AUC loss metric, $1 - \text{AUC}$, where AUC is the familiar Area Under the ROC Curve metric showing the probability that a randomly drawn positive example is scored more highly by our model than a randomly drawn negative example. We also report prediction variance as a way to assess the suppression of uninformative "noise" coefficients; all things being equal from an accuracy perspective, lower variance predictions are preferable as this reduces the risk of outlier predictions due to noise. In all experiments, the baseline for comparison is the same model definition with zero ℓ_1 regularization applied.

The pCTR results shown in Figure 2 show a classic regularization curve, with $\ell_1$regularization improving model quality in the mid ranges. As the theorems in this paper suggest, we see that the 0.5 power improves AucLoss and reduces prediction variance. The 1.0 power further reduces variance but harms AucLoss. The public URL results in Figure 3 agree. In the regime of model sizes from about $3e4$ to $3e5$ using a power $p = 0.5$ leads to optimal results in terms of accuracy with a model size that is an order-of-magnitude smaller than the best models achieved with $p = 0$, and also significantly smaller than the best models achieved with $p = 1$. These models also exhibit a significant decrease prediction variance compared to $p = 0$, *even when we control for model size.*

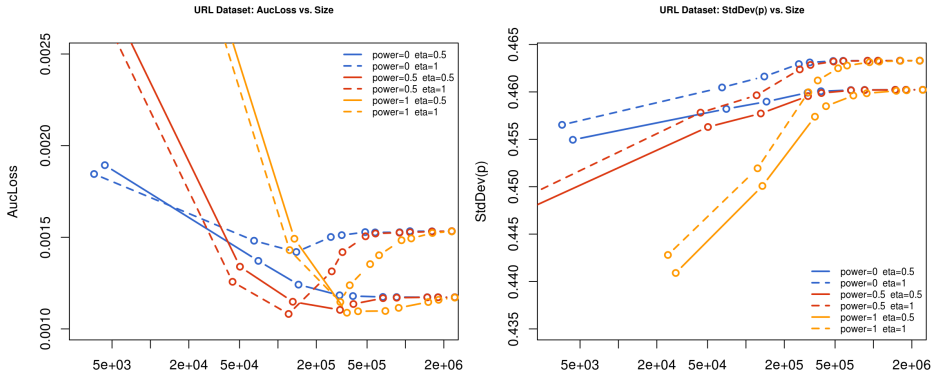

Figure 3: Experiments on the URL dataset, varying both ℓ_1 regularization strength and the learning rate.

References

[1] Avrim Blum, Adam Kalai, and John Langford. Beating the hold-out: Bounds for k-fold and progressive cross-validation. In *COLT*, 1999.

[2] John Duchi and Yoram Singer. Efficient learning using forward-backward splitting. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 495–503. 2009.

[3] John Duchi, Elad Hazan, and Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. In *COLT*, 2010.

[4] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Ambuj Tewari. Composite objective mirror descent. In *COLT*, 2010.

[5] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

[6] Elad Hazan. Introduction to Online Convex Optimization. Lecture Notes, 2015. URL `http://ocobook.cs.princeton.edu/OCObook.pdf`.

[7] Justin Ma, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. Identifying suspicious urls: An application of large-scale online learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, 2009.

[8] H. Brendan McMahan. Follow-the-regularized-leader and mirror descent: Equivalence theorems and L1 regularization. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[9] H. Brendan McMahan. Analysis techniques for adaptive online learning. *CoRR*, abs/1403.3465, 2014. URL `http://arxiv.org/abs/1403.3465`.

[10] H. Brendan McMahan and Matthew Streeter. Adaptive Bound Optimization for Online Convex Optimization. In *COLT*, 2010.

[11] H. Brendan McMahan, Gary Holt, D. Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, Sharat Chikkerur, Dan Liu, Martin Wattenberg, Arnar Mar Hrafnkelsson, Tom Boulos, and Jeremy Kubica. Ad click prediction: a view from the trenches. In *KDD*, 2013.

[12] Irina Rish and Genady Grabarnik. *Sparse Modeling: Theory, Algorithms, and Applications*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 2014. ISBN 1439828695, 9781439828694.

[13] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 2012.

[14] Diego Vidaurre, Concha Bielza, and Pedro Larrañaga. A Survey of $L1$ Regression. *International Statistical Review, 81, 3,*, 81(3):361–387, 2013.

[15] Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *NIPS*, 2009.