1 Programming: Adaptive Learning Rates

Recall in programming HW #1, part 2(c), you implemented the OGD algorithm with a constant learning rate $\eta$ and used it to train a linear support-vector machine on a small spam-classification task. Now you will solve the same problem, but using adaptive per-coordinate learning rates. In particular, the update will be computed separately for each coordinate $i \in \{1, 2, \ldots, d\}$ based on the rule

$$w_{t+1,i} = w_{t,i} - \eta_{t,i} g_{t,i},$$

(1)

where the learning rates have the form

$$\eta_{t,i} = \frac{\alpha}{\sqrt{1 + \sum_{s=1}^{t} g_{s,i}^2}}.$$

Here $\alpha$ is a parameter you will choose, and $g_{s,i} \in \mathbb{R}$ is the $i$th coordinate of the $g_s \in \partial f_s(w_s)$, a subgradient of the $s$th loss function at $w_s$. In addition to your code, you will produce a plot showing the average per-round loss as a function of $t$ for $t = 1, \ldots, 4601$, with three lines corresponding to $\alpha \in \{0.2\alpha_0, \alpha_0, 5.0\alpha_0\}$ with $\alpha_0 = 7.2$. We have chosen these values so that $\alpha = \alpha_0$ should produce the lowest average per-round loss on the final round; since both a somewhat lower and higher value of $\alpha$ produce worse loss, this is a good indication we have done a good job picking $\alpha$. For a real application, you would want to try a larger range of $\alpha$s, and plot the final cumulative loss as a function of $\alpha$ — you should see a nice, $U$-shaped curve. We did this in order to choose the value $\alpha_0$, see Figure [ ].

For comparison, again solve the problem with fixed learning-rate OGD, where the update is just

$$w_{t+1} = w_t - \eta g_t.$$

Plot three lines for constant-learning rate OGD for $\eta \in \{0.2\eta_0, \eta_0, 5.0\eta_0\}$ with $\eta_0 = 0.22$.

Recall that the loss function for a linear SVM is the hinge loss, defined as

$$f_t(w) = \max\{0, 1 - y_t w^T x_t\},$$
where $x_t, w_t \in \mathbb{R}^d$ and $y_t \in \{-1, +1\}$. Note that while we can view OGD as FTRL on linearized loss functions $\hat{f}_t(w) = g_t \cdot w$ for $g_t \in \partial f_t(w_t)$ (which drops constant terms), when computing the average per-round loss, it is critical you use the original true loss functions $f_t$, not the linearized functions $\hat{f}_t$. (You should think about why this is the case, but you do not need to write up your answer.)

Comment: In order for regret bounds of the form $BG\sqrt{T}$ to hold, where the $L_2$ norm of the post-hoc comparator $u$ is less than $B$, technically we should use the update that first applies the per-coordinate gradient update of (1), and then projects that point into the feasible set $W$ (usually an $L_\infty$ ball when using per-coordinate rates). However, in practice this is often unnecessary, and requires tuning an extra parameter (the radius of the feasible set), and so we will not implement this here.

2 Theory: Adaptive Regret Bounds for Strongly Convex Functions

Recall we proved the following theorem, using the Strong FTRL Lemma and some results from convexity theory:

**Theorem 1.** Consider the FTRL algorithm that plays according to

$$w_{t+1} = \arg\min_w f_{1:t}(w) + r_0: t(w),$$

where the proximal regularizers $r_t(w) \geq 0$ for $t \in \{0, 1, \ldots, T\}$, and $r_t(w_t) = 0$, and the functions $f_t : \mathbb{R}^d \to \mathbb{R}$ are convex. Let $h_0 = r_0$, and $h_t = r_t + f_t$ for $t \geq 1$. Then, further suppose the $r_t$ are chosen such that $h_{0:t}$ is 1-strongly-convex w.r.t. some norm $\| \cdot \|_{(t)}$ for
\( w \in \text{dom } r_{0:T}. \) Then, choosing any \( g_t \in \partial f_t(w_t) \) on each round, for any \( u \in \mathbb{R}^d, \)

\[
\text{Regret}(u) \leq r_{0:T}(u) + \sum_{t=1}^{T} \| g_t \|_{(t),*}^2. \tag{3}
\]

We will use this theorem to prove a regret bound for the Follow-The-Leader algorithm on strongly-convex functions, which plays

\[
w_{t+1} = \arg\min_w f_{1:t}(w). \tag{4}
\]

Suppose each \( f_t \) is 1-strongly convex w.r.t a fixed norm \( \| \cdot \|, \) and let \( G_T = \max_{t \in \{1, \ldots, T\}} \| g_t \|_*. \) (Typically in order to provide such a guarantee on the \( g_t \) in advance, we would have to constrain \( w_t \in W \) for some bounded feasible set, but we won’t worry about that for this problem.) You will prove the regret bound

\[
\text{Regret}(u) \leq G_T^2 (1 + \log T),
\]

which holds simultaneously for all \( T: \)

a) Define regularizers such that the update of (4) is equal to that of (2) (this is trivial).

b) Prove that \( \| w \|_{(t)} = \sqrt{t} \| w \| \) can be used in Theorem 1, and further that \( \| g \|_{(t),*} = \frac{1}{\sqrt{t}} \| g \|_* \). Prove the first fact from the definition of strong convexity, and the second from the definition of the dual norm (see the lecture 5 notes for both definitions). You don’t need to prove that \( \| w \|_{(t)} \) is actually a norm (though you might want to check this for yourself).

c) Plug the definition of \( r_t \) and \( \| \cdot \|_{(t),*} \) into (3), and simplify using the definition of \( G_T, \) and the fact that \( \sum_{t=1}^{T} \frac{1}{t} \leq 1 + \log T. \)

Observe that this log \( T \) regret bound is significantly better than the \( \sqrt{T} \) bounds achievable for general convex functions. The key is that the strongly-convex functions are essentially self-regularizing.