Online convex optimization: ridge regression, adaptivity

Lecturer: Sasha Rakhlin
Scribe: Alekh Agarwal, Lester Mackey

1 Lecture Outline

In the previous lectures we saw a general scheme for deriving regret bounds for online learning algorithms that try to minimize a regularized loss at every time step. We also argued that the linear loss is the optimal loss for the adversary if it is constrained to play convex loss functions. The goal of this lecture is to obtain logarithmic regret bound when the adversary plays sub-optimally in that it plays curved loss functions. Further, we will see how a time varying learning rate can be used to smoothly interpolate between the regret regimes of log $T$ and $\sqrt{T}$ corresponding to curved and linear losses respectively.

2 Curvature in Online learning

2.1 A rewriting of the regret bound

Last time we saw the game, where at every step we solve the problem:

$$x_{t+1} = \text{argmin}_{x \in \mathbb{R}^n} \sum_{s=1}^{t} \ell_s(x) + R(x)$$

(1)

Then we showed last time that $\forall u \in \mathbb{R}^n$:

$$\sum_{t=1}^{T} (\ell_t(x_t) - \ell_t(u)) = \eta^{-1} D_{\Phi_0}(u, x_1) - \eta^{-1} D_{\Phi_t}(u, x_{T+1}) + \eta^{-1} \sum_{t=1}^{T} D_{\Phi_t}(x_t, x_{t+1})$$

(2)

where $\Phi_0 = R, \Phi_t = \Phi_{t-1} + \eta \ell_t(\cdot)$.

Also, we showed in an earlier lecture on properties of Bregman divergences that:

$$D_{\Phi_t}(x_t, x_{t+1}) = D_{\Phi_t}(\nabla \Phi_t(x_{t+1}), \nabla \Phi_t(x_t))$$

$$= D_{\Phi_t}(0, \nabla \Phi_t(x_t))$$

(3)

where the second line follows from the fact that $x_{t+1}$ is the minimizer of (1). Also, we have that:

$$\nabla \Phi_t(x_t) = \nabla \Phi_{t-1}(x_t) + \eta \nabla \ell_t(x_t)$$

$$= \eta \nabla \ell_t(x_t)$$

(4)

where the second line is again using the fact that $x_t$ minimizes (1) at time $t$. Combining (3) and (4), we get:

$$D_{\Phi_t}(x_t, x_{t+1}) = D_{\Phi_t}(0, \eta \nabla \ell_t(x_t))$$

(5)
This might look like just some algebraic manipulation at the first sight. But note that the divergence we have rewritten is exactly the divergence that gets added up over time. In particular, we will now exploit this simple form to obtain interesting conclusions for the specific case of online ridge regression, which is an instance of a strongly convex loss.

### 2.2 Online Ridge Regression

Suppose the game is defined as follows:

For \( t = 1 \ldots T \)
- We pick \( w_t \in \mathbb{R}^n \)
- Adversary picks \((x_t, y_t)\), \((x_t \in \mathbb{R}^n, y_t \in \mathbb{R}, \|x_t\|_2 \leq B)\)
- We suffer \( \frac{1}{2}(w_t^\top x_t - y_t)^2 = \ell_t(w_t) \).

This game is exactly online ridge regression within a bounded \( \ell_2 \) ball if the regularizer is \( \frac{1}{2} \| \cdot \|^2 \). We will in fact assume that the \( y \)'s are bounded too. Further, we assume that there exists \( p^* \) such that at every time, \( \ell_t(w_t) \leq p^* \).

It isn’t clear how the boundedness of \( x \) and \( y \) implies this directly as the weights can still be large, but the assumption is needed for our analysis and will be clarified in the later lectures. Then the regret of this game is:

\[
R_T = \frac{1}{2} \sum_{t=1}^{T} (w_t^\top x_t - y_t)^2 - \frac{1}{2} \sum_{t=1}^{T} (x_t^\top w^* - y_t)^2
\]  

where \( w^* \) is the minimizer of the cumulative loss. Our goal is to demonstrate that \( R_T \leq O \left( \frac{\log T}{T} \right) \) which is significantly better than the \( O \left( \frac{1}{\sqrt{T}} \right) \) rate that we obtained without any assumptions on the losses.

Now for the ridge regularizer \( \frac{1}{2} \| \cdot \|^2 \), we have:

\[
w_{t+1} = \arg\min_w \frac{\eta}{2} \sum_{s=1}^{t} (w^\top x_s - y_s)^2 + \frac{1}{2} \| w \|^2
\]  

Then we have \( \Phi_0(u) = \frac{1}{2} \| u \|^2 \) and

\[
\Phi_t(u) = \frac{1}{2} \| u \|^2 + \frac{\eta}{2} \sum_{s=1}^{t} (w^\top x_s - y_s)^2
\]

\[
= \frac{1}{2} u^\top I u + \frac{\eta}{2} u^\top \left( \sum_{s=1}^{t} x_s x_s^\top \right) u + \frac{\eta}{2} \sum_{s=1}^{t} y_s^2 - \eta \sum_{s=1}^{t} u^\top (y_s x_s) + c_t
\]

where \( c_t = \sum_{s=1}^{t} y_s^2 \).

We will now state a lemma that allows us to manipulate the above quantity.

**Lemma 2.1.** If \( \Phi(u) = \frac{1}{2} u^\top M u + u^\top v + c \) for \( M \) positive definite (elliptic potential), then:

(a) \( \nabla \Phi(u) = M u + v \)

(b) \( \Phi^*(u) = \frac{1}{2} u^\top M^{-1} u - u^\top M^{-1} v + \frac{1}{2} v^\top M^{-1} v \)
Lemma 2.2. Let $B$ be an arbitrary $n \times n$ full rank matrix and $x$ be any vector. Let $A = B + xx^\top$. Then $x^\top A^{-1} x = x^\top (B + xx^\top)^{-1} x = 1 - \frac{\det(B)}{\det(A)}$

where $\det(A)$ is the determinant of the matrix $A$. In our setup, we intend to apply this lemma with $B = A_{t-1}$, $x = x_t$ and $A = A_t$ to the quadratic form of (9) which gives us:

$$R_T \leq \frac{1}{2} \|u\|^2 + 2p^* \sum_{t=1}^T \left( 1 - \frac{\det(A_{t-1})}{\det(A_t)} \right)$$

$$\leq \frac{1}{2} \|u\|^2 - 2p^* \sum_{t=1}^T \log \frac{\det(A_{t-1})}{\det(A_t)} \quad \text{(using } 1 - x \leq - \log x \text{ for } x > 0)$$

$$= \frac{1}{2} \|u\|^2 + 2p^* \log \frac{\det(A_T)}{A_0}$$

$$= \frac{1}{2} \|u\|^2 + 2p^* \log \det(A_T)$$

(10)

where the first equality follows from telescoping of terms in the summation, and the second one follows from the fact that $\det(A_0) = \det(I) = 1$.

Next we note that

$$\log(\det(A_T)) = \log \left( \det \left( I + \sum_{t=1}^T x_t x_t^\top \right) \right) = \log \left( \prod_{i=1}^n (1 + \lambda_i) \right) = \sum_{i=1}^n \log(1 + \lambda_i)$$
where \( \lambda_1, \ldots, \lambda_n \) are the eigenvalues of the matrix \( \sum_{t=1}^{T} x_t x_t^\top \). The eigenvalues of \( \sum_{t=1}^{T} x_t x_t^\top \) are known to be equal to the eigenvalues of the Gram matrix \( G \), where \( G_{ij} = x_i^\top x_j \). This implies

\[
\sum_{i=1}^{n} \lambda_i = \sum_{t=1}^{T} x_t^\top x_t \leq TB^2.
\]

Our expression for \( \log(\det(A_T)) \) is therefore maximized when \( \lambda_1 = \lambda_2 = \ldots = \lambda_n = \frac{TB^2}{n} \). We conclude that

\[
R_T \leq \frac{1}{2} \|u\|^2 + 2p^*n \log \left( 1 + \frac{TB^2}{n} \right)
\]  

(11)

### 2.3 Mirror Descent with Euclidean Norm Squared

In this section, we propose an algorithm that can exploit the suboptimal play of an adversary who plays only curved (strictly convex) loss functions. Consider again the mirror descent algorithm with \( R(x) = \frac{1}{2} \|x\|^2 \):

For \( t = 1 \ldots T \)
- Play \( x_t \)
- Observe \( \ell_t(\cdot) \)
- Update \( x_{t+1} = x_t - \eta_{t+1} \nabla \ell_t(x_t) \)

After observing \( \ell_t(\cdot) \), we can calculate

\[
H_t \geq 0, \quad \text{s.t.} \quad \nabla^2 \ell_t \geq H_t I \quad (12)
\]
\[
G_t \geq 0, \quad \text{s.t.} \quad G_t = \|\nabla \ell_t(x_t)\| \quad (13)
\]

where \( \nabla^2 \) is the Hessian, and \( A \succeq B \) means \( A - B \) is positive semidefinite.

The following lemma bounds our regret in terms of \( H_t \) and \( G_t \).

**Lemma 2.3.** Define \( H_{1:t} = \sum_{s=1}^{t} H_s \). If we set \( \eta_{t+1} = \frac{1}{H_{t+1}} \) in \( x_{t+1} = x_t - \eta_{t+1} \nabla \ell_t(x_t) \), then \( R_T \leq \frac{1}{2} \sum_{t=1}^{T} G_t^2 \eta_{t+1} \).

**Proof.** Let \( x^* = \arg\min_x \sum_{t=1}^{T} \ell_t(x) \) and \( \nabla_t = \nabla \ell_t(x_t) \). By (12),

\[
\ell_t(x_t) - \ell_t(x^*) \leq \nabla_t^\top (x_t - x^*) - H_t \frac{1}{2} \|x_t - x^*\|^2.
\]

Further,

\[
\frac{1}{2} \|x_{t+1} - x^*\|^2 \leq \frac{1}{2} \|x_t - x^*\|^2 - \eta_{t+1} \nabla_t^\top (x_t - x^*) + \eta_{t+1}^2 \frac{1}{2} \|\nabla_t\|^2,
\]

which implies

\[
\nabla_t^\top (x_t - x^*) \leq \frac{1}{2} \|x_t - x^*\|^2 - \frac{1}{2} \|x_{t+1} - x^*\|^2 + \frac{\eta_{t+1}}{2} \|\nabla_t\|^2.
\]

Combining, we get

\[
\ell_t(x_t) - \ell_t(x^*) \leq \frac{1}{2} \|x_t - x^*\|^2 - \frac{1}{2} \|x_{t+1} - x^*\|^2 + \frac{\eta_{t+1} G_t^2}{2} - H_t \frac{1}{2} \|x_t - x^*\|^2,
\]

which summed over \( t = 1, \ldots, T \) gives,

\[
\sum_{t=1}^{T} (\ell_t(x_t) - \ell_t(x^*)) \leq \sum_{t=1}^{T} \left( \frac{1}{2} \|x_t - x^*\|^2 \left( \frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} - H_t \right) \right) + \frac{1}{2} \sum_{t=1}^{T} G_t^2 \eta_{t+1}.
\]
Since \( \forall t, \frac{1}{2} \| x_t - x^* \|^2 \left( \frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} - H_t \right) \leq 0 \), we have established
\[
R_T \leq \frac{1}{2} \sum_{t=1}^{T} \frac{G_t^2}{H_{1:t}}.
\]

Note that our procedure does not require that we know the curvature of an adversary’s functions in advance. With the sole requirement that our adversary play only curved functions, our algorithm provides a logarithmic regret bound. To see this, note that when \( H_1 = \ldots = H_T = \sigma \),
\[
R_T \leq \frac{1}{2} \sum_{t=1}^{T} \frac{G_t^2}{\sigma} = O(\log(T)).
\]

However, if the adversary begins to play linear functions at some time step, we do not achieve the optimal \( O(\sqrt{T}) \) regret guarantee for arbitrary convex functions: our regret bound instead grows like \( O(T) \). Is it possible to guarantee \( O(\sqrt{T}) \) growth when arbitrary convex functions are played while still exploiting the suboptimality of curved functions? We explore this issue in the next section.

### 2.4 Adaptive Mirror Descent

Consider the following modified version of our mirror descent algorithm:

For \( t = 1 \ldots T \)
- Play \( x_t \)
- Observe \( \ell_t(\cdot) \)
- Pretend you observed \( \tilde{\ell}_t(\cdot) = \ell_t(\cdot) + \lambda_t \frac{1}{2} \| \cdot \|^2 \)
- Update \( x_{t+1} = x_t - \eta_{t+1} \nabla \tilde{\ell}_t(x_t) \).

If we choose \( H_t \) and \( G_t \) according to (12) and (13) and set \( \eta_{t+1} = \frac{1}{H_{1:t} + \lambda_{1:t}} \), we have
\[
\sum_{t=1}^{T} (\ell_t(x_t) + \lambda_t \frac{1}{2} \| x_t \|^2) \leq \sum_{t=1}^{T} (\ell_t(x^*) + \lambda_t \frac{1}{2} \| x^* \|^2) + \frac{1}{2} \sum_{t=1}^{T} \left( \frac{G_t + \lambda_tD^2}{H_{1:t} + \lambda_{1:t}} \right).
\]

where \( D \) is the diameter of the set from which vectors \( x_t \) are selected.

The purpose of the transformation \( \tilde{\ell}_t \) is to inject curvature into a possibly linear function \( \ell_t \). \(^1\) We want to select \( \lambda_t \) values which increase the curvature of linear functions without overly penalizing those functions which are already curved. That is, we want to choose \( \lambda_t \) values which balance the two terms in the following upper bound:
\[
R_T = \sum_{t=1}^{T} \ell_t(x_t) - \sum_{t=1}^{T} \ell_t(x^*) \leq \frac{1}{2} D^2 \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^{T} \left( \frac{G_t + \lambda_tD^2}{H_{1:t} + \lambda_{1:t}} \right).
\]

As it turns out, the right choice is given by
\[
\lambda_t = \frac{1}{2} \left( \sqrt{(H_{1:t} + \lambda_{1:t})^2 + \frac{8G_t^2}{3D^2}} - (H_{1:t} + \lambda_{1:t}) \right).
\]

\(^1\)Alternatively, we may view the transformation as L-2 regularization, which favors shrinking the player’s actions \( x_t \) toward the origin.
This small modification to our original mirror descent algorithm allows us to adapt to the curvature of an adversary’s functions and to achieve regret guarantees between $O(\log(T))$ and $O(\sqrt{T})$ depending on that curvature.