CS281B/Stat241B. Statistical Learning Theory. Lecture 15.

Peter Bartlett

Online Prediction

- Repeated game:

$$\text{Decision method plays } a_t \in \mathcal{A}$$
$$\text{World reveals } \ell_t \in \mathcal{L}$$

- Cumulative loss: $\hat{L}_n = \sum_{t=1}^{n} \ell_t(a_t)$.
- Aim to minimize regret, that is, perform well compared to the best (in retrospect) from some class:

$$\text{regret} = \underbrace{\sum_{t=1}^{n} \ell_t(a_t)}_{\hat{L}_n} - \underbrace{\min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a)}_{L_n^*}.$$

- Data can be adversarially chosen.

Online Prediction

Minimax regret is the value of the game:

$$\min_{a_1 \in \mathcal{A}} \max_{\ell_1 \in \mathcal{L}} \cdots \min_{a_n \in \mathcal{A}} \max_{\ell_n \in \mathcal{L}} \left(\hat{L}_n - L_n^* \right).$$

$$\hat{L}_n = \sum_{t=1}^{n} \ell_t(a_t), \qquad L_n^* = \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a).$$

Online Convex Optimization

1. Problem formulation
2. Empirical minimization fails.
3. Gradient algorithm.
4. Regularized minimization
5. Regret bounds

Online Convex Optimization: Problem Formulation

- $\mathcal{A} =$ convex subset of $\mathbb{R}^d$.
- $\mathcal{L} =$ set of convex real functions on $\mathcal{A}$.

Example:

- $\ell_t(a) = (x_t \cdot a - y_t)^2$.
- $\ell_t(a) = |x_t \cdot a - y_t|$.
- $\ell_t(a) = -\log\left(\exp(a'T(y_t) - A(a))\right)$, for $A(a)$ the log normalization of this exponential family, with sufficient statistic $T(y)$.

Online Convex Optimization: Examples

Example: Experts.

$$\mathcal{A} = \Delta^{K-1} = \left\{ w \in \mathbb{R}^K : w_i \geq 0, \sum_i w_i = 1 \right\},$$

$$\mathcal{L} = \left\{ a \mapsto x^T a : x \in [0,1]^K \right\}$$

NB: Regret is

$$\sum_{t=1}^n a_t^T x_t - \min_{a \in \mathcal{A}} \sum_{t=1}^n a^T x_t = \sum_{t=1}^n a_t^T x_t - \min_k \sum_{t=1}^n x_{t,k}.$$

Online Convex Optimization: Examples

Example: Online shortest path.

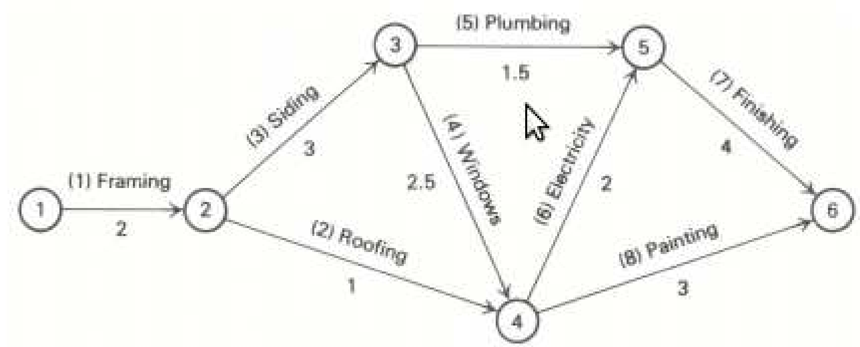
Fix a directed graph $G = (V, E)$, a source $s \in V$ and a sink $t \in V$.

$\mathcal{A} = \Delta^{K-1}$, where K is number of paths from s to t.

Adversary chooses cost function $f_t : E \to [0, 1]$,

$\ell_t(a) = \mathbf{E}_{p \sim a} \sum_{e \in p} f_t(e)$.

- Navigation; cost is time.
- Scheduling: identify critical path; cost is negative time.



(Applied Math Programming. Bradley, Hax, and Magnanti. Addison-Wesley. 1977.)

Online Convex Optimization: Examples

Example: Online shortest path.

Can represent $\mathcal{A}$ as a convex subset of $\mathbb{R}^E$: $a \in [0,1]^E$ s.t.

$$\sum_{(i,j)\in E} a_{i,j} - \sum_{(k,i)\in E} a_{k,i} = \begin{cases} 1 & \text{if } i = s, \\ -1 & \text{if } i = t, \\ 0 & \text{otherwise.} \end{cases}$$

Then $a^T f_t = \mathbf{E}_{p\sim a} \sum_{e\in p} f_t(e) = \ell_t(a)$.

Again, best distribution on paths is best path.

Online Convex Optimization: Example

Choosing a_t to minimize past losses, $a_t = \arg\min_{a \in \mathcal{A}} \sum_{s=1}^{t-1} \ell_s(a)$, can fail. ('fictitious play,' 'follow the leader')

- Suppose $\mathcal{A} = [-1, 1]$, $\mathcal{L} = \{a \mapsto v \cdot a : |v| \le 1\}$. Consider:

$$
\begin{aligned}
a_1 &= 0, & \ell_1(a) &= \frac{1}{2}a, \\
a_2 &= -1, & \ell_2(a) &= -a, \\
a_3 &= 1, & \ell_3(a) &= a, \\
a_4 &= -1, & \ell_4(a) &= -a, \\
a_5 &= 1, & \ell_5(a) &= a, \\
&\vdots & &\vdots
\end{aligned}
$$

- $a^* = 0$ shows $L_n^* \le 0$, but $\hat{L}_n = n - 1$.

Online Convex Optimization: Example

- Choosing a_t to minimize past losses can fail.
- The strategy must avoid overfitting, just as in probabilistic settings.
- Similar approaches (regularization; Bayesian inference) are applicable in the online setting.
- First approach: gradient steps.
  Stay close to previous decisions, but move in a direction of improvement.

Online Convex Optimization: Gradient Method

$$a_1 \in \mathcal{A},$$
$$a_{t+1} = \Pi_{\mathcal{A}}\left(a_t - \eta \nabla \ell_t(a_t)\right),$$

where $\Pi_{\mathcal{A}}$ is the Euclidean projection on $\mathcal{A}$,

$$\Pi_{\mathcal{A}}(x) = \arg\min_{a \in \mathcal{A}} \|x - a\|.$$

Theorem: For $G = \max_t \|\nabla \ell_t(a_t)\|$ and $D = \text{diam}(\mathcal{A})$, the gradient strategy with $\eta = D/(G\sqrt{n})$ has regret satisfying

$$\hat{L}_n - L_n^* \leq GD\sqrt{n}.$$

Online Convex Optimization: Gradient Method

Example: (2-ball, 2-ball)
$\mathcal{A} = \{a \in \mathbb{R}^d : \|a\| \leq 1\}$, $\mathcal{L} = \{a \mapsto v \cdot a : \|v\| \leq 1\}$. $D = 2$, $G \leq 1$.
Regret is no more than $2\sqrt{n}$.

(And $O(\sqrt{n})$ is optimal.)

Example: (1-ball, ∞-ball)
$\mathcal{A} = \Delta^{K-1}$, $\mathcal{L} = \{a \mapsto v \cdot a : \|v\|_\infty \leq 1\}$.
$D = 2$, $G \leq \sqrt{K}$.
Regret is no more than $2\sqrt{Kn}$.

Since competing with the whole simplex is equivalent to competing with the vertices (experts) for linear losses, this is worse than exponential weights ($\sqrt{K}$ versus $\log K$).

Gradient Method: Proof

Define
$$\tilde{a}_{t+1} = a_t - \eta \nabla \ell_t(a_t),$$
$$a_{t+1} = \Pi_{\mathcal{A}}(\tilde{a}_{t+1}).$$

Fix $a \in \mathcal{A}$ and consider the measure of progress $\|a_t - a\|$.

$$\begin{aligned}\|a_{t+1} - a\|^2 &\leq \|\tilde{a}_{t+1} - a\|^2 \\ &= \|a_t - a\|^2 + \eta^2 \|\nabla \ell_t(a_t)\|^2 - 2\eta \nabla_t(a_t) \cdot (a_t - a).\end{aligned}$$

By convexity,

$$\begin{aligned}\sum_{t=1}^{n} (\ell_t(a_t) - \ell_t(a)) &\leq \sum_{t=1}^{n} \nabla \ell_t(a_t) \cdot (a_t - a) \\ &\leq \frac{\|a_1 - a\|^2 - \|a_{n+1} - a\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{n} \|\nabla \ell_t(a_t)\|^2\end{aligned}$$

Online Convex Optimization

1. Problem formulation
2. Empirical minimization fails.
3. Gradient algorithm.
4. Regularized minimization
   - Bregman divergence
   - Regularized minimization $\Leftrightarrow$ minimizing latest loss and divergence from previous decision
   - Constrained minimization equivalent to unconstrained plus Bregman projection
   - Linearization
   - Mirror descent
5. Regret bounds

Online Convex Optimization: A Regularization Viewpoint

- Suppose ℓ_t is linear: $\ell_t(a) = g_t \cdot a$.
- Suppose $\mathcal{A} = \mathbb{R}^d$.
- Then minimizing the regularized criterion

$$a_{t+1} = \arg\min_{a \in \mathcal{A}} \left(\eta \sum_{s=1}^{t} \ell_s(a) + \frac{1}{2} \|a\|^2 \right)$$

corresponds to the gradient step

$$a_{t+1} = a_t - \eta \nabla \ell_t(a_t).$$

Online Convex Optimization: Regularization

Regularized minimization

Consider the family of strategies of the form:

$$a_{t+1} = \arg\min_{a \in \mathcal{A}} \left(\eta \sum_{s=1}^{t} \ell_s(a) + R(a) \right).$$

The regularizer $R : \mathbb{R}^d \to \mathbb{R}$ is strictly convex and differentiable.

- R keeps the sequence of a_ts stable: it diminishes ℓ_t's influence.
- We can view the choice of a_{t+1} as trading off two competing forces: making $\ell_t(a_{t+1})$ small, and keeping a_{t+1} close to a_t.
- This is a perspective that motivated many algorithms in the literature. We'll investigate why regularized minimization can be viewed this way.

Properties of Regularization Methods

In the unconstrained case ($\mathcal{A} = \mathbb{R}^d$), regularized minimization is equivalent to minimizing the latest loss and the distance to the previous decision. The appropriate notion of distance is the Bregman divergence $D_{\Phi_{t-1}}$:

Define

$$\Phi_0 = R,$$
$$\Phi_t = \Phi_{t-1} + \eta \ell_t,$$

so that

$$a_{t+1} = \arg\min_{a \in \mathcal{A}} \left(\eta \sum_{s=1}^{t} \ell_s(a) + R(a) \right)$$
$$= \arg\min_{a \in \mathcal{A}} \Phi_t(a).$$

Bregman Divergence

Definition 1. *For a strictly convex, differentiable* $\Phi : \mathbb{R}^d \to \mathbb{R}$*, the Bregman divergence wrt* Φ *is defined, for* $a, b \in \mathbb{R}^d$*, as*

$$D_\Phi(a, b) = \Phi(a) - (\Phi(b) + \nabla\Phi(b) \cdot (a - b)) \, .$$

$D_\Phi(a, b)$ is the difference between $\Phi(a)$ and the value at a of the linear approximation of Φ about b. (PICTURE)

Bregman Divergence

Example: For $a \in \mathbb{R}^d$, the squared euclidean norm, $\Phi(a) = \frac{1}{2}\|a\|^2$, has

$$D_\Phi(a,b) = \frac{1}{2}\|a\|^2 - \left(\frac{1}{2}\|b\|^2 + b \cdot (a-b)\right)$$
$$= \frac{1}{2}\|a-b\|^2,$$

the squared euclidean norm.