CS 396: Online Markets

Lecture 4: Online Learning

Last Time:

- auction theory
- second-price auction
- first-price auction
- complete information analysis (Nash equilibrium)
- incomplete information analysis (Bayes-Nash equilbrium)

Today:

- online learning
- exponential weights

Exercise: Online Learning

Setup:

- $n = 10$ days
- you choose umbrella or not
- then nature chooses weather
- payoffs

	it rains	it is sunny
you take umbrella	1	0
you don't take umbrella	0	1

Question: What's your best strategy?

Online Learning

"make decisions over time, learn to do well"

Model:

- k actions
- n rounds
- action j's payoff in round i: $v_j^i \in [0, h]$
- in round i:
  (a) choose an action j^i
  (b) learn payoffs $v_1^i, \ldots, v_k^i$
  (c) obtain payoff $v_{j^i}^i$.
- payoff ALG $= \sum_{i=1}^n v_{j^i}^i$

Goal: profit close to best action in hindsight

Def: the **best in hindsight** payoff is

$$\text{OPT} = \max_j \sum_{i=1}^n v_j^i$$

Def: the **regret** of the algorithm is

$$\begin{aligned}\text{Regret}_n &= {}^1\!/\!_n[\text{OPT} - \text{ALG}] \\ &= {}^1\!/\!_n \left[\max_j \sum_{i=1}^n v_j^i - \sum_{i=1}^n v_{j^i}^i\right]\end{aligned}$$

Goal: vanishing regret, a.k.a. "no regret"

i.e., $\lim_{n\to\infty} \text{Regret}_n = 0$

Alg 0: follow the leader (FTL)

- let $V_j^i = \sum_{r=1}^i v_j^r$
- in round i choose: $> j^i = \text{argmax}_j V_j^{i-1}$

Example: $k = 2$ actions

	1	2	3	4	5	6	...
Action 1	1/2	0	1	0	1	0	...
Action 2	0	1	0	1	0	1	...

- OPT $\approx {}^n\!/\!_2$
- FTL ≈ 0
- worst-case regret is constant, i.e., $\Theta(1)$

Thm: all deterministic online learning algorithms have $\Theta(1)$ worst-case regret.

Proof Sketch: In each round i, nature gives payoff 0 to ALG's action, and payoff 1 to all other actions.

Conclusion: must randomized.

Exercise: Follow the Leader

Setup:

	1	2	3	4	5
Action 1	1/2	1	0	0	1
Action 2	0	1	1	1	1

Question: What action does follow the leader choose in rounds 3? And round 5?

Learning Algorithms

Idea: exponentially increase (resp. decrease) probability on good (resp. bad) actions.

Alg 1: exponential weights (EW)

- learning rate ϵ
- let $V_j^i = \sum_{r=1}^{i} v_j^r$
- in round i choose j with probability π_j^i proportional to $(1+\epsilon)^{V_j^{i-1}/h}$

  i.e., $\pi_j^i = \frac{(1+\epsilon)^{V_j^{i-1}/h}}{\sum_{j'} (1+\epsilon)^{V_{j'}^{i-1}/h}}$

Example:

- $\epsilon = 1$
- $v_j^i \in \{0, 1\}$
- exp. weights = "double score if payoff = 1"

	1	2	3	4
Action 1	1	1	0	0
Action 2	0	0	1	1
———	:-:	-	-	-
Weight 1	1	2	4	4
Weight 2	1	1	1	2

Intuition: learning rate ϵ

- small ϵ: takes a long time to make good decisions.
- large ϵ: long run decisions are not accurate.

Thm: for payoffs in $[0, h]$,

$$\mathbf{E}[\text{EW}] \geq (1-\epsilon)\,\text{OPT} - \frac{h}{\epsilon}\ln k.$$

Cor: in n steps and payoffs in $[0, h]$, tune learning rate ϵ so

$$\mathbf{E}[\text{Regret(EW)}] \leq 2h\sqrt{\frac{\ln k}{n}}$$

Proof:

- $\text{OPT} < hn$
- $\mathbf{E}[\text{EW}] \geq \text{OPT} - \epsilon hn - \frac{h}{\epsilon}\ln k$
- choose learning rate to equate: $\epsilon hn = \frac{h}{\epsilon}\ln k$
- $\Rightarrow \epsilon = \sqrt{\frac{\ln k}{n}}$
- Regret $= \frac{1}{n}[2hn\epsilon] = 2h\sqrt{\frac{\ln k}{n}}$

Note: to set learning rate

- larger $n \Rightarrow$ slower learning rate is optimal
- larger $k \Rightarrow$ faster learning rate is optimal
