Homework #4

Due at 17:00 on April 23, 2007.

You may work on this assignment in groups, but you must acknowledge your collaborators, and each person must write up the solutions in their own words. In writing your solutions, you may use any theorem which was proven in class or in the lecture notes. Turn in your solutions by 17:00 on Monday, April 23. You may bring them to class or email them to Professor Kleinberg. (If you prepare an electronic copy of your solutions, email is preferred.)

1. **(Online learning with i.i.d. inputs)**
In class we have concentrated on worst-case analysis of online learning algorithms. In this exercise you will analyze the performance of online algorithms for the best-expert and multi-armed bandit problems when the cost functions are assumed to be i.i.d. samples from a distribution, but the distribution is unknown to the algorithm designer. All of the algorithms considered in this exercise work with strategy set $\mathscr{S} = [n]$. Accordingly, we identify a cost function $c : \mathscr{S} \to [0,1]$ with a vector $c \in [0,1]^n$ whose i-th component is the cost $c(i)$. If μ is a probability measure on $[0,1]^n$ then $\mu^{\otimes T}$ denotes the probability measure on $\left([0,1]^n\right)^T$ such that a random sample from $\mu^{\otimes T}$ is a sequence $(c_1, \ldots, c_T)$ of independent random samples from μ. If ALG is a randomized (or deterministic) online algorithm which chooses a sequence of strategies $x_1, x_2, \ldots, x_T \in \mathscr{S}$ in response to a sequence of cost functions $c_1, c_2, \ldots, c_T$, then the *regret* $R(\mathsf{ALG}, \mu^{\otimes T})$ is defined to be

$$R(\mathsf{ALG}, \mu^{\otimes T}) = \max_{x \in \mathscr{S}} \mathbf{E}\left[\sum_{t=1}^{T} \left(c_t(x_t) - c_t(x)\right)\right],$$

where the expectation is over $(c_1, \ldots, c_T)$ sampled from $\mu^{\otimes T}$, and over the algorithm's random choices (if any). Note that the maximization over $x \in \mathscr{S}$ occurs *outside* the expectation operator in this definition; this ordering of maximization and expectation operators influences the answers to some of the following questions.

   (a) Recall the "follow the leader" algorithm (FTL) which selects, at each time t, a strategy

   $$x_t \in \arg\min_{x \in \mathscr{S}} \left(\sum_{s=1}^{t-1} c_s(x)\right).$$

   Prove that for every input distribution μ, the regret of FTL does not tend to infinity with T, i.e.

   $$\forall \mu \ \sup_{T \in \mathbb{N}} R(\mathsf{FTL}, \mu^{\otimes T}) < \infty.$$

   (b) Suppose we adapt the FTL to the bandit setting by adopting the following algorithm BFTL. It begins by sampling each strategy once: $x_t = t$ for $1 \le t \le n$.

Then, for $t > n$, it always chooses the strategy x whose average cost (averaged over all the times when the algorithm picked x) is minimum. In other words, for $t > n$ it selects

$$x_t \in \arg\min_{x \in \mathscr{S}} \left(\frac{\sum_{s=1}^{t-1} c_s(x) \mathbf{1}_{x_s = x}}{\sum_{s=1}^{t-1} \mathbf{1}_{x_s = x}} \right).$$

Here $\mathbf{1}_{x_s=x}$ denotes the number 1 if $x_s = x$, 0 otherwise. Prove that the regret of BFTL can grow linearly with T, i.e.

$$\exists \mu \ \liminf_{T \to \infty} \frac{R(\mathsf{BFTL}, \mu^{\otimes T})}{T} > 0.$$

(c) We will now describe a modification of BFTL, called $\mathsf{UCB1}$, whose regret grows only *logarithmically* with T. Once again, the algorithm begins by sampling each strategy once: $x_t = t$ for $1 \le t \le n$. For $t > n$ and $x \in \mathscr{S}$ let

$$n(x,t) = \sum_{s=1}^{t-1} \mathbf{1}_{x_s = x}, \qquad \hat{c}(x,t) = \frac{\sum_{s=1}^{t-1} c_s(x) \mathbf{1}_{x_s=x}}{n(x,t)}, \qquad w(x,t) = \sqrt{\frac{8 \ln(t)}{n(x,t)}}$$

$$I(x,t) = \text{the interval } [\hat{c}(x,t) - w(x,t), \hat{c}(x,t) + w(x,t)].$$

For $t > n$ the $\mathsf{UCB1}$ algorithm chooses

$$x_t \in \arg\min_{x \in \mathscr{S}} \left(\hat{c}(x,t) - w(x,t) \right).$$

i. Let μ be any probability measure on $[0,1]^n$. Let $\bar{c}_\mu(x) = \mathbf{E}_{c \leftarrow \mu}[c(x)]$ for every $x \in \mathscr{S}$. Prove that if $c_1, c_2, \ldots, c_T$ are sampled according to $\mu^{\otimes T}$ and $x_1, x_2, \ldots, x_T$ are chosen by an arbitrary online algorithm, then for all $t > n$,

$$\Pr\left(\bar{c}_\mu(x) \notin I(x,t) \right) = O\left(\frac{1}{t^3} \right).$$

ii. If x is any element of $\mathscr{S}$ and $\Delta(x) = \max_{y \in \mathscr{S}} \left(\bar{c}_\mu(x) - \bar{c}_\mu(y) \right)$, prove that

$$\Pr\left(x_t = x \right) \le O\left(\frac{1}{t^3} \right) + \Pr\left(w(x,t) \ge \frac{1}{2} \Delta(x) \ \text{ and } \ x_t = x \right).$$

iii. Prove that for every input distribution μ the regret of $\mathsf{UCB1}$ grows only logarithmically with T, i.e.

$$\forall \mu \ \sup_{T \in \mathbb{N}} \left(\frac{R(\mathsf{UCB1}, \mu^{\otimes T})}{\log T} \right) < \infty.$$

iv. **(Extra credit:)** Prove that the logarithmic dependence on T is unavoidable. In other words, for every online multi-armed bandit algorithm ALG, there exists a probability measure μ on $[0,1]^n$ such that

$$\liminf_{T \to \infty} \left(\frac{R(\mathsf{UCB1}, \mu^{\otimes T})}{\log T} \right) > 0.$$

2. **(The secretary problem with interview costs)**
An employer is interviewing n applicants for a job. The interviews take place in random order (i.e. the order in which the applicants are interviewed is equally likely to be each of the $n!$ possible permutations of the applicants) and after each interview the employer must decide whether to hire or dismiss the most recently interviewed applicant. Only one applicant may be hired, and an applicant who has been dismissed cannot later be hired. Assume that for any two applicants a and b, there is a well-defined relative ranking (either $a \prec b$ or $b \prec a$) and the binary relation "$\prec$" constitutes a total ordering of the set of applicants. The decision to hire or dismiss the applicant interviewed at time t can depend on the relative ranking of all applicants interviewed at or before time t, but not after time t. The objective is to maximize the probability of hiring the best applicant, i.e. the applicant a such that $b \prec a$ for all other applicants b.[1]

   (a) Formulate the problem as a Markov decision process (MDP), compute the state transitions and costs for this MDP, and determine the optimal policy. (One suggestion is to define the set of states to be $([n] \times \{0, 1\}) \cup \{*\}$. The special state $*$ denotes that an applicant has already been hired, whereas state (t, q) means that the employer is currently conducting the tth interview, and q is a flag which indicates whether the applicant being interviewed is the best one the employer has seen so far.) Prove that if one uses the optimal policy, the probability of hiring the best applicant tends to $1/e$ as $n \to \infty$.

   (b) Now assume that there is a cost $c < 1$ for conducting each interview, and there is a reward of n for hiring the best candidate. The employer has three possible actions after every interview: hire the applicant, dismiss the applicant and continue the interview process, or terminate the interview process (collecting zero reward but incurring zero additional cost). Find the best policy in this case, i.e. express the optimal action at time t in terms of c and n.

3. **(A chess problem)**
Let n be an even integer greater than 2. Suppose that a chess king starts at a random square of a $n \times n$ chessboard (i.e., grid) and takes a random walk according to the following process. If the king is sitting on square s_t at time t, let A_t denote the set of squares adjacent[2] to s_t which have not been visited at any time prior to t. If A_t is empty, then the king moves from s_t to an adjacent square chosen uniformly at random. If A_t is nonempty, then the king moves to a random element of A_t chosen by sampling from a distribution on A_t which assigns to each white[3] square a probability proportional to $1 + \frac{1}{10n}$ and assigns to each black square a probability proportional to $1 - \frac{1}{10n}$. The process terminates at the first time when all squares of the chessboard have been visited. Prove that with probability at least $1/4$, the last square visited by the king is black.

[1] This is a well-known problem that goes by the names "secretary problem", "best-choice problem", "marriage problem", or "dowry problem."

[2] Two squares (x, y) and (z, w) are *adjacent* if $|x - z| \leq 1$ and $|y - w| \leq 1$.

[3] A square (x, y) is *white* if $x + y$ is even, and otherwise it is black.