Problem Set 3

Stats 311/EE 377

Due: Thursday, February 18 in class

Question 3.1 (Minimax redundancy and different loss functions)**:** In this question, we consider expected losses under the Bernoulli distribution. Assume that $X_i \stackrel{\text{i.i.d.}}{\sim} \mathsf{Bernoulli}(p)$, meaning that $X_i = 1$ with probability p and $X_i = 0$ with probability $1 - p$. We consider four different loss functions, and their associated expected regret, for measuring the accuracy of our predictions of such X_i. For each of the four choices below, we prove expected regret bounds on

$$\mathsf{Red}_n(\widehat{\theta}, P, \ell) := \sum_{i=1}^n \mathbb{E}_P[\ell(\widehat{\theta}(X_1^{i-1}), X_i)] - \inf_\theta \sum_{i=1}^n \mathbb{E}_P[\ell(\theta, X_i)], \tag{1}$$

where $\widehat{\theta}$ is a predictor based on $X_1, \ldots, X_{i-1}$ at time i. Define $S_i = \sum_{j=1}^i X_j$ to be the partial sum up to time i. For each of parts (a)–(c), at time i use the predictor

$$\widehat{\theta}_i = \widehat{\theta}(X_1^{i-1}) = \frac{S_{i-1} + \frac{1}{2}}{i}.$$

(a) Loss function: $\ell(\theta, x) = \frac{1}{2}(x - \theta)^2$. Show that $\mathsf{Red}_n(\widehat{\theta}, P, \ell) \le C \cdot \log n$ where C is a constant.

(b) Loss function: $\ell(\theta, x) = x \log \frac{1}{\theta} + (1 - x) \log \frac{1}{1-\theta}$, the usual log loss for predicting probabilities. Show that $\mathsf{Red}_n(\widehat{\theta}, P, \ell) \le C \cdot \log n$ whenever the true probability $p \in (0, 1)$, where C is a constant. *Hint: Note that there exists a prior π for which $\widehat{\theta}$ is a Bayes strategy. What is this prior?*

(c) Loss function: $\ell(\theta, x) = |x - \theta|$. Show that $\mathsf{Red}_n(\widehat{\theta}, P, \ell) \ge c \cdot n$, where $c > 0$ is a constant, whenever the true probability $p \notin \{0, \frac{1}{2}, 1\}$.

(d) **Extra credit:** Show that there is a numerical constant $c > 0$ such that for any procedure $\widehat{\theta}$, the worst-case redundancy $\sup_{p \in [0,1]} \mathsf{Red}_n(\widehat{\theta}, \mathsf{Bernoulli}(p), \ell) \ge c\sqrt{n}$ for the absolute loss ℓ in part (c). Give a strategy attaining this redundancy.

Question 3.2 (Strong versions of redundancy)**:** Assume that for a given $\theta \in \Theta$ we draw $X_1^n \sim P_\theta$. We define the Bayes redundancy for a family of distributions $\mathcal{P} = \{P_\theta\}_{\theta \in \Theta}$ as

$$C_n^\pi := \inf_Q \int D_{\rm kl}\left(P_\theta \| Q\right) d\pi(\theta) = I_\pi(T; X_1^n),$$

where π is a probability measure on Θ, T is distributed according to π, and conditional on $T = \theta$, we draw $X_1^n \sim P_\theta$, and I_π denotes the mutual information when T is drawn according to π. Define

the maximin redundancy $C_n^* := \sup_\pi C_n^\pi$ as the worst-case Bayes redundancy. We show that for "most" points θ under the prior π, if $\overline{Q} = \int P_\theta d\pi(\theta)$ is the mixture of all the P_θ under the prior π, then no distribution Q can have subtantially better redundancy that $\overline{Q}$.

Consider any distribution Q on the set $\mathcal{X}$ and let $\epsilon \in [0, 1]$, and define the set of points θ where Q is ϵ-better than the worst case redundancy as

$$B_\epsilon := \{\theta \in \Theta : D_{\rm kl}\left(P_\theta \| Q\right) \le (1 - \epsilon) C_n^*\}.$$

(a) Show that for any prior π, we have

$$\pi(B_\epsilon) \le \frac{\log 2 + C_n^* - I_\pi(T; X_1^n)}{\epsilon C_n^*}.$$

As an aside, note this implies that if π_i is a sequence of priors tending to $\sup_\pi I_\pi(T; X_1^n)$ and the redundancy $C_n^* \to \infty$, then so long as $C_n^* - I_{\pi_i}(T; X_1^n) \ll \epsilon C_n^*$, we have $\pi_i(B_\epsilon) \approx 0$.

(b) Assume that π attains the supremum in the definition of C_n^*. Show that

$$\pi(B_\epsilon) \le O(1) \cdot \exp(-\epsilon C_n^*).$$

Hint: Introduce the random variable Z to be 1 if the random variable $T \in B_\epsilon$ and 0 otherwise, then use that $Z \to T \to X_1^n$ forms a Markov chain, and expand the mutual information. For part (b), the inequality $\frac{1-x}{x} \log \frac{1}{1-x} \le 1$ for all $x \in [0, 1]$ may be useful.

Question 3.3: We consider the doubling trick, a frequently used technique in online learning to allow good performance of online learning procedures even without knowledge of the number of steps n they will be run. For this question, we define the regret in the usual way as

$$\mathsf{Reg}_n := \sup_{w^\star \in \mathcal{W}} \sum_{t=1}^n [\ell_t(w_t) - \ell_t(w^\star)].$$

(a) Suppose that we have a procedure (algorithm) $\mathsf{A}(\eta)$ parameterized by the real value $\eta \ge 0$ (usually, this is simply a stepsize) that achieves the regret bound

$$\mathsf{Reg}_n \le \frac{r^2}{2\eta} + \frac{\eta}{2} L^2 n$$

where r and L are known constants. Consider the following procedure, which proceeds in epochs $k = 1, 2, \ldots$, each of which lasts for $n_k = 2^k$ steps. At the start of epoch k, restart the algorithm $\mathsf{A}(\eta)$ with parameter choice $\eta_k = \frac{r}{L\sqrt{2^k}}$, and run the algorithm with this choice of parameter for 2^k steps. Show that

$$\mathsf{Reg}_n \le C \cdot Lr\sqrt{n},$$

where C is some numerical constant (in our solution, we have $C \le 2/(\sqrt{2} - 1)$).

(b) Now we consider a slightly more restrictive setting, but we obtain better guarantees. Consider the mixture of experts problem, in d experts suffer losses in $[0, 1]$ at each timestep; we let

$g_t \in [0,1]^d$ denote the loss vector. We play a mixture of experts $w_t \in \mathcal{W} = \Delta_d$, suffering (expected) loss $\ell_t(w_t) = \langle g_t, w_t \rangle$. In the course notes, we show that in this setting

$$\mathsf{Reg}_n = \sum_{t=1}^n [\ell_t(w_t) - \ell_t(w^\star)] \le \frac{\log d}{\eta} + \frac{\eta}{2} \sum_{t=1}^n \sum_{j=1}^d w_{t,j} g_{t,j}^2$$

when using the exponential weights algorithm with stepsize η, which in turn implies

$$\sum_{t=1}^n [\ell_t(w_t) - \ell_t(w^\star)] \le \left(1 - \frac{\eta}{2}\right)^{-1} \left[\frac{\log d}{\eta} + \frac{\eta}{2} \sum_{t=1}^n \ell_t(w^\star)\right]$$

for any $w^\star \in \mathcal{W}$. Consider the following procedure, which proceeds in epochs $k = 1, 2, \ldots$, within each of which we perform exponential weights with stepsize $\eta_k = \min\{1, \sqrt{\log d}/2^k\}$. Let E_k denote those times t belonging to epoch k, which correspond to times when we run exponential weights with parameter η_k. Define $L^{(k)} = \min_j \sum_{t \in E_k} g_{t,j}$ to be the loss incurred by the best expert in epoch k as the procedure runs, and continue epoch k until the best expert's loss in epoch k satisfies $L^{(k)} \ge 4^k$. Then begin a new epoch. Show that with this procedure,

$$\sum_{t=1}^n [\ell_t(w_t) - \ell_t(w^\star)] \le C_1 \log\log d \cdot \log d + C_2 \sqrt{\log d \cdot \sum_{t=1}^n \ell_t(w^\star)}$$

for numerical constants C_1 and C_2 (we obtain $C_1 \le 3$ and $C_2 \le 8\sqrt{2}$).

Question 3.4 (An empirical comparison of Bandit algorithms)**:** In this question, you will investigate three algorithms for solving the Bandit problem: the Upper Confidence Bound algorithm (UCB), Thompson sampling (also known as Posterior Sampling), and exponential gradient. You will attempt to maximize the reward achieved by the algorithms (note that in the notes, we sometimes maximize and sometimes minimize; make sure you have your signs correct!).

In particular, set the rewards for the arms in the following way:

i. Let $\theta_1 = \frac{1}{2}$ and $\theta_2 = \frac{1}{2} - \epsilon, \ldots, \theta_K = \frac{1}{2} - \epsilon$.

ii. When arm j is sampled, return $Y = 1$ with probability θ_j and $Y = 0$ with probability $1 - \theta_j$.

Now, repeat the following experiment with the values

(a) $K = 10$, $\epsilon = .1$, and $n = 10^6$ steps

(b) $K = 10$, $\epsilon = .02$, and $n = 10^6$ steps.

Perform Thompson sampling (Example 12.5 in the notes) assuming that the prior on θ is to have each coordinate independent with $\mathsf{Beta}(1,1)$ distribution. Perform UCB with the confidence parameter $\delta_t = 1/\sqrt{t}$ (Algorithm 12.1 in the notes) and the appropriate choice of the sub-Gaussian parameter σ^2 (*Hint:* use Hoeffding's lemma for σ^2). Perform exponentiated gradient (Algorithm 12.3) using the optimal stepsize choice η, when assuming that $\sigma^2 = \frac{1}{2}$ in the bound.

Plot your results for each of experiments (a) and (b). Which algorithm do you prefer?