Theoretical Machine Learning
Homework 3

Instructor: Haipeng Luo

This homework is due on **11/08, 11:59pm**. See course website for more instructions on finishing and submitting your homework as well as late policy.

1. (**Hedge and Minimax Theorem**) Recall that in Lecture 4, the very first step to relax the sequential value $\mathcal{V}^{\text{seq}}(\mathcal{F}, n)$ is to apply the minimax theorem. A simplified version of the celebrated von Neumann's minimax theorem states that for any K_1 by K_2 matrix $M \in [0,1]^{K_1 \times K_2}$, the following holds

$$\min_{p \in \Delta(K_1)} \max_{q \in \Delta(K_2)} p^\top M q = \max_{q \in \Delta(K_2)} \min_{p \in \Delta(K_1)} p^\top M q \tag{1}$$

where $\Delta(K_1)$ and $\Delta(K_2)$ are probability simplex of $K_1 - 1$ and $K_2 - 1$ dimension respectively.

One way to interpret the theorem is to imagine a zero-sum game between a "min player" who chooses $p \in \Delta(K_1)$ and a "max player" who chooses $q \in \Delta(K_2)$. The loss for the min player is $p^\top M q$, which is also the reward for the max player (hence zero-sum). The left-hand side of Equation (1) is then the minimum loss for the min player if he/she plays first against an optimal max player who plays second. Similarly, the right-hand side of Equation (1) is the maximum reward for the max player if he/she plays first against an optimal min player who plays second. The theorem states that these two values are in fact the same, meaning that the order of play does not matter, as long as both players deploy the optimal strategy.

Follow the steps below to prove this theorem.

(a) (3pts) Prove

$$\min_{p \in \Delta(K_1)} \max_{q \in \Delta(K_2)} p^\top M q \geq \max_{q \in \Delta(K_2)} \min_{p \in \Delta(K_1)} p^\top M q.$$

Note that this is the easier and intuitive direction — playing second should always be at least as favorable as playing first.

(b) For the other (less intuitive) direction

$$\min_{p \in \Delta(K_1)} \max_{q \in \Delta(K_2)} p^\top M q \leq \max_{q \in \Delta(K_2)} \min_{p \in \Delta(K_1)} p^\top M q, \tag{2}$$

you will prove it by applying some algorithms (which might seem weird if this is the first time you see such arguments). Specifically, imagine that the game is repeatedly played between the two players for n rounds. On round t, the min player selects p_t according to the Hedge algorithm:

$$p_t(i) \propto \exp\left(-\eta \sum_{\tau=1}^{t-1} e_i^\top M q_\tau\right), \quad \forall i = 1, \ldots, K_1$$

where e_i is the i-th basis vector and $\eta = \sqrt{\frac{\ln K_1}{n}}$. The max player simply responds optimally:

$$q_t = \operatorname*{argmax}_{q \in \Delta(K_2)} p_t^\top M q.$$

i. (5pts) Prove the following:

$$\frac{1}{n}\sum_{t=1}^{n} p_t^\top M q_t \le \max_{q\in\Delta(K_2)} \min_{p\in\Delta(K_1)} p^\top M q + 2\sqrt{\frac{\ln K_1}{n}} \tag{3}$$

using Lemma 1 from Lecture 6.

ii. (4pts) Prove

$$\frac{1}{n}\sum_{t=1}^{n} p_t^\top M q_t \ge \min_{p\in\Delta(K_1)} \max_{q\in\Delta(K_2)} p^\top M q \tag{4}$$

using the definition of q_t.

iii. (2pts) Combine Equation (3) and Equation (4) to conclude Equation (2), hence proving the minimax theorem.

(c) (6pts) Prove Equation (2) again in a similar way except that q_t, the decision of the max player, is now changed from the optimal response to another Hedge algorithm:

$$q_t(i) \propto \exp\left(\eta' \sum_{\tau=1}^{t-1} p_\tau^\top M e_i\right), \quad \forall i = 1, \dots, K_2.$$

where $\eta' = \sqrt{\frac{\ln K_2}{n}}$. (Hint: apply Lemma 1 of Lecture 6 again with $\ell_t(i) = 1 - p_t^\top M e_i$.)

2. (**Fat-shattering dimension and Perceptron**) Recall the fat-shattering dimension $\mathrm{fat}(\mathcal{F}, \alpha)$ and sequential fat-shattering dimension $\mathrm{sfat}(\mathcal{F}, \alpha)$ defined in Lecture 4 and 6 respectively. Let $\mathcal{X} = B_2^d$ and $\mathcal{F} = \left\{ f_\theta(x) = \langle \theta, x \rangle \mid \theta \in B_2^d \right\}$. In this exercise you need to prove

$$\min\left\{d, \left\lfloor \frac{4}{\alpha^2} \right\rfloor\right\} \le \mathrm{fat}(\mathcal{F}, \alpha) \le \mathrm{sfat}(\mathcal{F}, \alpha) \le \frac{16}{\alpha^2}.$$

(a) (3pts) First prove $\mathrm{fat}(\mathcal{F}, \alpha) \le \mathrm{sfat}(\mathcal{F}, \alpha)$, which is in fact true for any $\mathcal{F}$.

(b) (6pts) Prove $\mathrm{fat}(\mathcal{F}, \alpha) \ge \min\left\{d, \left\lfloor \frac{4}{\alpha^2} \right\rfloor\right\}$ by explicitly constructing a set of $\min\left\{d, \left\lfloor \frac{4}{\alpha^2} \right\rfloor\right\}$ samples that is α-shattered by $\mathcal{F}$. (Hint: consider using basis vectors as samples.)

(c) (5pts) Prove $\mathrm{sfat}(\mathcal{F}, \alpha) \le \frac{16}{\alpha^2}$ by using the Perceptron guarantee (once again this is another example of using an algorithm to prove an (in)equality, just like Problem 1(b)). Specifically, suppose $\boldsymbol{x}$ is a $\mathcal{X}$-valued tree of depth n that is α-shattered by $\mathcal{F}$, with witness $\boldsymbol{y}$, a $[-1, +1]$-valued tree. Consider running Perceptron with example $x_t' = \frac{1}{\sqrt{2}}(\boldsymbol{x}_t(\boldsymbol{\epsilon}), \boldsymbol{y}_t(\boldsymbol{\epsilon})) \in B_2^{d+1}$ and label ϵ_t for time t, where ϵ_t is exactly the opposite of what Perceptron predicts. Note that this is a valid environment even though ϵ_t depends on what the algorithm predicts, since Perceptron is a deterministic algorithm (and thus the environment can simulate its behavior). Finally, use the guarantee of Perceptron (Theorem 1 of Lecture 7) to conclude $n \le \frac{16}{\alpha^2}$, which implies $\mathrm{sfat}(\mathcal{F}, \alpha) \le \frac{16}{\alpha^2}$.

3. (**Winnow and Hedge**) When the γ-margin assumption holds with $p = q = 2$, we have seen that Perceptron makes at most $\frac{1}{\gamma^2}$ mistakes. In this exercise, you will prove that when the γ-margin assumption holds with $p = 1$ and $q = \infty$, a different algorithm called *Winnow*, makes at most $\frac{16 \ln(2d)}{\gamma^2}$ mistakes.

(a) (6pts) We will start from a slightly strong assumption which replaces B_1^d in the γ-margin assumption with $\Delta(d) \subset B_1^d$, the $d-1$ dimensional simplex. In other words, the assumption is: $\|x_t\|_\infty \le 1$ for all t, and there exists $\theta^\star \in \Delta(d)$ such that $y_t \langle \theta^\star, x_t \rangle \ge \gamma$ for all t. Now consider the Winnow algorithm outlined below.

Initialize $p_1 = (\frac{1}{d}, \cdots, \frac{1}{d}) \in \Delta(d)$.
For $t = 1, \ldots, n$:

- receive x_t and predict $s_t = \text{sign}(\langle p_t, x_t \rangle)$;
- receive y_t and compute p_{t+1} such that

$$p_{t+1}(i) \propto p_t(i) \exp(\eta \mathbf{1}\{y_t \neq s_t\} y_t x_t(i)), \quad \forall i = 1, \ldots, d.$$

Figure 1: (Simplified) Winnow Algorithm

Show that this is in fact an instance of Hedge (Hint: recall how this is done similarly in Problem 1(c)). Then use the Hedge guarantee (Lemma 1 of Lecture 6) to prove that the number of mistakes made by this algorithm is at most $\frac{16 \ln d}{\gamma^2}$ if $\eta = \frac{\gamma}{8}$.

(b) (4pts) Now we come back to the original γ-margin assumption, that is: $\|x_t\|_\infty \le 1$ for all t, and there exists $\theta^\star \in B_1^d$ such that $y_t \langle \theta^\star, x_t \rangle \ge \gamma$ for all t. Consider the following algorithm:

Initialize $p_1 = (\frac{1}{2d}, \cdots, \frac{1}{2d}) \in \Delta(2d)$.
For $t = 1, \ldots, n$:

- receive x_t and predict $s_t = \text{sign}\left(\sum_{i=1}^d p_t(i) x_t(i) - \sum_{i=d+1}^{2d} p_t(i) x_t(i)\right)$;
- receive y_t and compute p_{t+1} such that

$$p_{t+1}(i) \propto p_t(i) \exp\left(\eta \mathbf{1}\{y_t \neq s_t\} \text{sign}(d + 1/2 - i) y_t x_t(i)\right), \quad \forall i = 1, \ldots, 2d.$$

Figure 2: Winnow Algorithm

Prove that this algorithm makes at most $\frac{16 \ln(2d)}{\gamma^2}$ mistakes if $\eta = \frac{\gamma}{8}$. Hint: reduce this problem (algorithm and margin assumption) to the last problem and apply the last result directly.

4. (**Deriving FTRL**) In this exercise, you will follow the general recipe discussed in Lecture 7 to derive a version of the Follow-the-Regularized-Leader algorithm. Consider the following setup (which uses notation from the lectures): $\mathcal{F} \subset \mathbb{R}^d$ is a convex set and $\mathcal{Z} \subset \mathbb{R}^d$ is some arbitrary set. The loss function is linear: $\ell(f, z) = \langle f, z \rangle$. A regularizer $\psi : \mathcal{F} \to \mathbb{R}$ is a differentiable 1-strongly convex function with respect to some norm $\|\cdot\|$. We assume for all $z \in \mathcal{Z}$, $\|z\|_\star \le 1$, where $\|\cdot\|_\star$ is the dual norm of $\|\cdot\|$. This means that loss function is 1-Lipschitz.

We will need the concept of *convex conjugate* (but this exercise assumes no prior knowledge on it from you). The convex conjugate $\psi^\star : \mathbb{R}^d \to \mathbb{R}$ of ψ is defined as

$$\psi^\star(g) = \sup_{f \in \mathcal{F}} \left(\langle f, g \rangle - \psi(f) \right).$$

You will need to use the following three properties of convex conjugate. First, by definition it is clear that for any $f \in \mathcal{F}$ and $g \in \mathbb{R}^d$:

$$\langle f, g \rangle \le \psi(f) + \psi^\star(g). \tag{5}$$

Second, ψ being 1-strongly convex implies $\psi^\star$ being *1-smooth*, that is, for any $g, g' \in \mathbb{R}^d$:

$$\psi^\star(g) \le \psi^\star(g') + \langle \nabla\psi^\star(g'), g - g' \rangle + \tfrac{1}{2} \|g - g'\|_\star^2 . \tag{6}$$

Finally, it can also be verified that for any $g \in \mathbb{R}^d$,

$$\nabla\psi^\star(g) = \operatorname*{argmax}_{f \in \mathcal{F}} \left(\langle f, g \rangle - \psi(f) \right), \tag{7}$$

that is, the gradient of $\psi^\star$ is the vector that achieves the maximum in its definition.

(a) (6pts) Recall the generalized sequential Rademacher complexity:

$$\mathcal{R}_n(z_{1:t}) = \sup_{\boldsymbol{z}} \mathbb{E}_{\epsilon_{t+1:n}} \sup_{f \in \mathcal{F}} \left(2 \sum_{s=t+1}^{n} \epsilon_s \ell(f, \boldsymbol{z}_{s-t}(\epsilon_{t+1:s-1})) - \sum_{s=1}^{t} \ell(f, z_s) \right).$$

Prove that for any $\eta > 0$,

$$\mathcal{R}_n(z_{1:t}) \le \frac{R}{\eta} + \frac{1}{\eta}\psi^\star\left(-\eta \sum_{s=1}^{t} z_s \right) + 2\eta(n-t)$$

where $R = \sup_{f \in \mathcal{F}} \psi(f)$. (Hint: first use Equation (5), then apply Equation (6) repeatedly.)

(b) (8pts) Define relaxation

$$\mathrm{Rel}_n(z_{1:t}) = \min_{\eta > 0} \left(\frac{R}{\eta} + \frac{1}{\eta}\psi^\star\left(-\eta \sum_{s=1}^{t} z_s \right) + 2\eta(n-t) \right).$$

Show that this relaxation is admissible by proving the following (using Equations (5), (6) and (7))

$$\mathrm{Rel}_n(z_{1:n}) \ge - \inf_{f \in \mathcal{F}} \left\langle f, \sum_{t=1}^{n} z_t \right\rangle$$
$$\text{and } \forall t < n, \quad \mathrm{Rel}_n(z_{1:t}) \ge \inf_{\widehat{y} \in \mathcal{F}} \sup_{z \in \mathcal{Z}} \left(\langle \widehat{y}, z \rangle + \mathrm{Rel}_n(z_{1:t}, z) \right).$$

Hint: in the process of proving the last inequality, you will also derive the algorithm:

$$\widehat{y}_{t+1} = \operatorname*{argmin}_{f \in \mathcal{F}} \left\langle f, \sum_{s=1}^{t} z_s \right\rangle + \frac{1}{\eta_t}\psi(f)$$

where $\eta_t = \operatorname{argmin}_{\eta > 0} \left(\frac{R}{\eta} + \frac{1}{\eta}\psi^\star\left(-\eta \sum_{s=1}^{t} z_s \right) + 2\eta(n-t) \right)$. This is a version of FTRL with adaptive learning rate.

(c) (2pts) Finally write down the regret bound of the algorithm derived from the last step.