1 Predicting Bit-sequences

Think of the online learning problem where on each round \( t \) we predict the next bit \( y_t \in \{\pm 1\} \). Also say \( \mathcal{F} \subset \{\pm 1\}^n \) and we want to minimize regret (in expectation):

\[
\text{Reg}_n = \frac{1}{n} \sum_{t=1}^{n} \mathbb{1}_{\hat{y}_t \neq y_t} - \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{t=1}^{n} \mathbb{1}_{f \neq y_t}
\]

When can we ensure \( \mathbb{E}[\text{Reg}_n] \to 0 \)? Let us denote the minimax rate as

\[
V_n = \min_{\text{algorithms}} \max_{\text{sequence}} \mathbb{E}[\text{Reg}_n]
\]

Claim 1.

\[
V_n = \frac{1}{2n} \mathbb{E}_f \left[ \sup_{y_n \in \{\pm 1\}} \mathbb{1}_{y_n} \sum_{t=1}^{n} f_t \right]
\]

Proof. The basic idea is to write down the minimax rate in a recursive form and get a characterization for it. To this end, say you had already played rounds 1 to \( n-1 \) optimally, then, on the last two rounds, what are the optimal moves for both the players. We write this value given \( y_1, \ldots, y_{n-1} \) were already produced as:

\[
V_n(y_1, \ldots, y_{n-1}) = \min_{q_n \in [0, 1]} \sup_{y_n \in \{\pm 1\}} \left\{ \mathbb{E}_{\hat{y}_n \sim q_n} \left[ \mathbb{1}_{\hat{y}_n \neq y_n} \right] - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \mathbb{1}_{f \neq y_t} \right\}
\]

That is on the last round, the learner picks distribution \( q_n \) that minimizes loss at the last step while the adversary picks \( y_n \) that maximizes the loss at last step while also minimizes loss of the target we are comparing our regret against. In fact if we define \( V_n(y_1, \ldots, y_n) = -\inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \mathbb{1}_{f \neq y_t} \) then we see that

\[
V_n(y_1, \ldots, y_{n-1}) = \min_{q_n \in [0, 1]} \sup_{y_n \in \{\pm 1\}} \left\{ \mathbb{E}_{\hat{y}_n \sim q_n} \left[ \mathbb{1}_{\hat{y}_n \neq y_n} \right] + V_n(y_1, \ldots, y_n) \right\}
\]

Thus we see that,

\[
V_n(y_1, \ldots, y_{n-1}) = \min_{q_n \in [0, 1]} \sup_{y_n \in \{\pm 1\}} \left\{ q_n \mathbb{1}_{y_n \neq y_n} + (1 - q_n) \mathbb{1}_{y_n = y_n} + V_n(y_1, \ldots, y_n) \right\}
\]

\[
= \min_{q_n \in [0, 1]} \max \left\{ (1 - q_n) + V_n(y_1, \ldots, y_{n-1}, +1), q_n + V_n(y_1, \ldots, y_{n-1}, -1) \right\}
\]
Solution is to pick $q_n$ such that the two terms are equal. Hence

$$V_n(y_1, \ldots, y_{n-1}) = \frac{1}{2} + \frac{V_n(y_1, \ldots, y_{n-1}, +1) + V_n(y_1, \ldots, y_{n-1}, +1)}{2}$$

$$= \frac{1}{2} + \mathbb{E}_{\epsilon_n} [V_n(y_1, \ldots, y_{n-1}, \epsilon_n)]$$

Now recursively we continue as

$$V_n(y_1, \ldots, y_{n-2}) = \min_{q_{n-1} \in [0,1]} \sup_{y_{n-1} \in \{\pm 1\}} \{q_{n-1} \mathbb{1}_{y_n = 1} + (1 - q_{n-1}) \mathbb{1}_{y_n = -1} + V_n(y_1, \ldots, y_{n-1})\}$$

$$= \frac{1}{2} + \mathbb{E}_{\epsilon_{n-1}} [V_n(y_1, \ldots, y_{n-2}, \epsilon_{n-1})]$$

Proceeding as follows we conclude that :

$$V_n(\cdot) = \mathbb{E}_{\epsilon_1} [V_n(\epsilon_1)] = \ldots = \mathbb{E}_{\epsilon_n} [V_n(\epsilon_1, \ldots, \epsilon_n)]$$

Hence we conclude that :

$$\text{Minimax}_n = \frac{V_n(\cdot)}{n} = \frac{1}{2} + \frac{1}{n} \mathbb{E}_{\epsilon_n} \left[ - \inf_{f \in \mathcal{F}} \sum_{t=1}^n \mathbb{1}_{f_t \neq \epsilon_t} \right] = \frac{1}{2} + \frac{1}{2n} \mathbb{E}_{\epsilon_n} \left[ \sup_{f \in \mathcal{F}} \sum_{t=1}^n f_t \epsilon_t \right] - \frac{1}{2}$$

\[\square\]

Prediction algorithm : the prediction algorithm corresponding to the above analysis is exactly the $q_t$ that minimizes the recursion at each step and hence is given by

$$q_t = \arg \min_{q \in [0,1]} \max_{y_t \in \{\pm 1\}} \{\mathbb{E}_{\hat{y}_t \sim q} [\mathbb{1}_{\hat{y}_t \neq y_t}] + V_n(y_1, \ldots, y_t)\}$$

$$= \frac{1}{2} (1 + V_n(y_1, \ldots, y_{t-1}, +1) - V_n(y_1, \ldots, y_{t-1}, -1))$$

$$= \frac{1}{2} (1 + \mathbb{E}_{\epsilon_{t+1:n}} [V_n(y_1, \ldots, y_{t-1}, +1, \epsilon_{t+1}, \ldots, \epsilon_n)] - \mathbb{E}_{\epsilon_{t+1:n}} [V_n(y_1, \ldots, y_{t-1}, -1, \epsilon_{t+1}, \ldots, \epsilon_n)])$$

In fact, we can also show that the following randomized algorithm works. Draw $\epsilon_{t+1}, \ldots, \epsilon_n$ and set :

$$q_t = \frac{1}{2} \left( 1 + \inf_{f \in \mathcal{F}} \left\{ \sum_{j=1}^{t-1} \mathbb{1}_{f_j \neq y_j} + \mathbb{1}_{f_t \neq 1} + \sum_{i=t+1}^n \mathbb{1}_{f_i \neq \epsilon_i} \right\} - \inf_{f \in \mathcal{F}} \left\{ \sum_{j=1}^{t-1} \mathbb{1}_{f_j \neq y_j} + \mathbb{1}_{f_t \neq -1} + \sum_{i=t+1}^n \mathbb{1}_{f_i \neq \epsilon_i} \right\} \right)$$

2 General Online Learning

For a general online learning problem, the minimax rate can be written recursively as:

$$\mathcal{V}_n^\text{eq}(x_1, y_1, \ldots, x_n, y_n) = - \inf_{f \in \mathcal{F}} \sum_{t=1}^n \ell(f(x_t), y_t)$$
and subsequently,
\[ V_{sq}^n((x_1, y_1), \ldots, (x_t, y_t)) = \sup_{x_{t+1} \in X, y_{t+1} \in Y} \inf_{\mathbb{Q}_{t+1} \in \Delta(Y)} \sup_Y \left\{ E_{\hat{y}_{t+1} \sim \mathbb{Q}_{t+1}} [\ell(\hat{y}_{t+1}, y_{t+1})] + V_{sq}^n((x_1, y_1), \ldots, (x_{t+1}, y_{t+1})) \right\} \]

Finally we get
\[ n V_{sq}^n = V_{sq}^n(\cdot) = \sup_{x_{t+1} \in X, q_{t+1} \in \Delta(Y), y_{t+1} \in Y} \inf_{a_t \in A_{t+1}} \sup_Y \left[ n \sum_{t=1}^{n} \ell(\hat{y}_{t}, y_{t}) - \inf_{f \in F} \sum_{t=1}^{n} \ell(f(x_t), y_{t}) \right] \]

3 Minimax Theorem

We shall use the celebrated minimax theorem as a key tool to bound the minimax rate for online learning problems. Below we state a generalization of Von Neuman’s minimax theorem.

**Theorem 2** (Browein’14). Let \( A \) and \( B \) be Banach spaces. Let \( A \subset A \) be nonempty, weakly compact, and convex, and let \( B \subset B \) be nonempty and convex. Let \( g : A \times B \mapsto \mathbb{R} \) be concave with respect to \( b \in B \) and convex and lower-semicontinuous with respect to \( a \in A \) and weakly continuous in \( a \) when restricted to \( A \). Then
\[ \sup_{b \in B} \inf_{a \in A} g(a, b) = \inf_{a \in A} \sup_{b \in B} g(a, b) \]

The above theorem states that under the right conditions, one can swap infimum and supremum. We shall use this in a sequential manner to swap the order of the learner and adversary and use this to get a handle on minimax rate for online learning. For instance using the above theorem, we can show that for any loss \( \ell \), lower semicontinuous in its first argument, as long as \( Y \) is well behaved (compact for instance),
\[ \inf_{q_t \in \Delta(Y)} \sup_Y E_{\hat{y}_t \sim q_t} [\ell(\hat{y}_t, y_t) + \Phi(y_t)] = \sup_{p_t \in \Delta(Y)} \inf_{y_t \in Y} E_{y_t \sim p_t} [\ell(\hat{y}_t, y_t) + \Phi(y_t)] \]

where \( \Phi \) is some arbitrary function that is lower semi-continuous. We shall use \( \Phi(y_t) = V_{sq}^n((x_1, y_1), \ldots, (x_t, y_t)) \)