

Machine Learning Notes 11.25: Introduction to Online Learning

Notes Group

December 22, 2025

1 Introduction: Online Learning with Expert Advice

1.1 Problem Formulation

We consider the classical *Online Learning with Expert Advice* framework:

There are N experts. For each round $t = 1, 2, \dots, T$:

1. Each expert $i \in [N]$ outputs a prediction

$$\hat{y}_{t,i} \in \{-1, +1\}.$$

2. After observing all experts' predictions, the adversary reveals the true label

$$y_t \in \{-1, +1\}.$$

(Importantly, y_t may even depend on the experts' outputs $\hat{y}_{t,i}$.)

3. The learner then makes a prediction x_t (possibly randomized and based on the experts' advice).

1.2 Goal

The learner makes predictions x_t based on expert advice in each round, aiming to minimize cumulative loss.

Under the 0–1 loss, the learner incurs a loss of

$$\mathbf{1}[x_t \neq y_t]$$

in round t .

The total loss after T rounds is

$$L_T = \sum_{t=1}^T \mathbf{1}[x_t \neq y_t].$$

The objective is to keep the learner's total loss “close” to that of the best expert in hindsight:

$$L_T \approx \min_{i \in [N]} \sum_{t=1}^T \mathbf{1}[\hat{y}_{t,i} \neq y_t].$$

Equivalently, we aim for small regret, defined as

$$\text{Regret}_T = L_T - \min_{i \in [N]} \sum_{t=1}^T \mathbf{1}[\hat{y}_{t,i} \neq y_t].$$

2 Deterministic Weighted Majority Algorithm

A foundational approach to this problem is the **Weighted Majority Algorithm (WMA)**. The learner will maintain a weight for each expert.

2.1 Algorithm Description

Algorithm 1 Weighted Majority Algorithm

- 1: **Input:** Number of experts N , parameter $\beta \in (0, 1)$.
- 2: **Initialize:** Weight $w_{1,i} = 1$ for all $i \in \{1, \dots, N\}$.
- 3: **for** $t = 1, \dots, T$ **do**
- 4: Receive predictions $\tilde{y}_{t,i}$ from all experts i .
- 5: Make prediction based on the weighted majority vote:

$$x_t = \text{sgn} \left(\sum_{i=1}^N w_{t,i} \tilde{y}_{t,i} \right)$$

- 6: Receive true label y_t .
- 7: **if** $x_t \neq y_t$ **then**
- 8: Update weights for experts who made a mistake:

$$w_{t+1,i} = \begin{cases} \beta w_{t,i} & \text{if } \tilde{y}_{t,i} \neq y_t \\ w_{t,i} & \text{if } \tilde{y}_{t,i} = y_t \end{cases}$$

- 9: **else**
 - 10: $w_{t+1,i} = w_{t,i}$
 - 11: **end if**
 - 12: **end for**
-

2.2 Loss Bound Analysis

Theorem 1. Let $m_T^* = \min_i m_{T,i}$ be the loss made by the best expert. The loss of the learner based on the Weight Majority Algorithm L_T satisfies:

$$L_T \leq \frac{\log(1/\beta)}{\log(2/(1+\beta))} m_T^* + \frac{\log N}{\log(2/(1+\beta))}$$

Proof. Define the $W_t = \sum_{i=1}^N w_{t,i}$ as the sum of weights at the beginning of round t . We can find that $W_1 = N$.

Consider the relationship between W_{t+1} and W_t . The weights are updated only when the learner makes a mistake ($x_t \neq y_t$). Since the learner predicts based on the weighted majority, if the learner is wrong, the experts contributing to the wrong prediction must account for at least half of the total weight W_t . This means that after the update, at least half of the total weight is multiplied by β .

$$W_{t+1} \leq \frac{1}{2} W_t + \frac{1}{2} \beta W_t = \frac{1+\beta}{2} W_t$$

Note that the learner makes L_T mistakes over T rounds, we have:

$$W_{T+1} \leq N \left(\frac{1+\beta}{2} \right)^{L_T}$$

Meanwhile, consider the weight of the best expert i^* . This weight is multiplied by β at most m_T^* times. Since weights are non-negative:

$$W_{T+1} = \sum_{i=1}^N w_{T+1,i} \geq w_{T+1,i^*} \geq \beta^{m_T^*}$$

Combining the upper and lower bounds:

$$\beta^{m_T^*} \leq N \left(\frac{1+\beta}{2} \right)^{L_T}$$

Taking the logarithm of both sides:

$$m_T^* \log \beta \leq \log N + L_T \log \left(\frac{1+\beta}{2} \right)$$

Rearranging to solve for L_T (noting that $\log(\frac{1+\beta}{2}) < 0$):

$$L_T \log\left(\frac{2}{1+\beta}\right) \leq m_T^* \log\left(\frac{1}{\beta}\right) + \log N$$

Thus

$$L_T \leq \frac{\log(1/\beta)}{\log(2/(1+\beta))} m_T^* + \frac{\log N}{\log(2/(1+\beta))}$$

□

3 Randomized Weighted Majority

Using the same idea of weighted majority as above, we introduce randomness to make things harder for the adversary. Specifically, we listen to expert i with probability $w_i / \sum w$.

3.1 Algorithm Description

Algorithm 2 Weighted Majority Algorithm

- 1: **Input:** Number of experts N , parameter $\beta \in (0, 1)$.
- 2: **Initialize:** Weight $w_{1,i} = 1$ for all $i \in \{1, \dots, N\}$.
- 3: **for** $t = 1, \dots, T$ **do**
- 4: Receive predictions $\tilde{y}_{t,i}$ from all experts i .
- 5: Sample r from $\{1, \dots, N\}$. For each i in $\{1, \dots, N\}$, i have probability $w_i / \sum w$ being selected.
- 6: Make prediction according to sampled expert $x_t := \tilde{y}_{t,r}$.
- 7: Receive true label y_t .
- 8: Update weights for experts who made a mistake:

$$w_{t+1,i} = \begin{cases} \beta w_{t,i} & \text{if } \tilde{y}_{t,i} \neq y_t \\ w_{t,i} & \text{if } \tilde{y}_{t,i} = y_t \end{cases}$$

9: **end for**

3.2 Expected Regret Bound

In the randomized case, we consider the expected loss defined as follows:

$$L_T := \sum_{t=1}^T \sum_{i=1}^n \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}} \cdot \mathbf{1}[\tilde{y}_{t,i} \neq y_t].$$

We also similarly define $m_{T,i}, m_T^*$ as in the deterministic case. Then, for fixed $\beta \in (0.5, 1)$, we have:

$$L_T \leq (2 - \beta) m_T^* + \frac{\log N}{1 - \beta}.$$

Taking $\beta = 1 - \sqrt{\frac{\log N}{T}}$ and we get:

$$L_T \leq m_T^* + 2\sqrt{T \log N}.$$

The proof of this bound is left as homework, so we omit the proof here.

3.3 The Doubling Trick

When the time horizon T is known in advance, the Randomized Weighted Majority algorithm achieves a regret bound of $O(\sqrt{T \log N})$. When the total number of rounds is unknown, the Doubling Trick transforms the problem so that RWM still attains the same regret bound.

Specifically, we start by setting the initial horizon to $T = 1$. Whenever the actual number of rounds reaches T , we let $T \leftarrow 2T$, recompute the learning rate β , and reset all expert weights w to the uniform initialization.

Let $K = \lfloor \log_2 T \rfloor$. The total regret is

$$R = \sum_{k=0}^K O(\sqrt{T_k \log N}) = O\left(\sqrt{\log N} \sum_{k=0}^K \sqrt{2^k}\right).$$

Since $\sum_{k=0}^K \sqrt{2^k}$ forms a geometric series, this is $C\sqrt{T}$.
Thus, the total regret satisfies

$$R = O(\sqrt{T \log N}).$$

4 Connection to Zero-Sum Games

A zero-sum game is a strategic interaction between two players where one player's gain is exactly balanced by the other player's loss. Consider a two-player zero-sum game with:

Player 1's strategy set X (probability distributions over actions)

Player 2's strategy set Y (probability distributions over actions)

A payoff matrix M where $M(x, y)$ represents the payoff to Player 1 when Player 1 chooses strategy x and Player 2 chooses strategy y

4.1 Minimax Theorem

Theorem 2 (Minimax Theorem). *For any finite two-player zero-sum game, the following equality holds:*

$$\max_{x \in X} \min_{y \in Y} M(x, y) = \min_{y \in Y} \max_{x \in X} M(x, y)$$

This common value is called the value of the game.

4.2 Solving Zero-Sum Games via Online Learning

We first discuss why a learning-theoretic perspective can be used to justify the Minimax Theorem. If strategies are deterministic, the second mover always has an advantage. However, when strategies are randomized (mixed strategies), the advantage disappears and the order of play becomes irrelevant.

Assume players use mixed strategies and the game is characterized by a payoff matrix M .

1. Repeated Zero-Sum Matrix Game

Instead of considering a single-round game, we examine a repeated zero-sum matrix game played over many rounds.

2. Row Player as an Online Learner

Assume the row player moves first in each round. We interpret the row player as an online learner using the *online learning with expert advice* framework:

- Each row of the payoff matrix is viewed as an “expert.”
- In each round, the column player chooses a column.
- Each expert (row) incurs a loss depending on the chosen column.

Thus, every expert has an associated loss sequence determined by the opponent's moves.

3. Regret Guarantees and the Minimax Theorem

After many rounds of play, the online learner's cumulative performance is close to that of the best expert in hindsight. This no-regret property implies that the average payoff achieved by the row player approaches the value of the game, which leads directly to the equality in the Minimax Theorem. (The precise justification of why this implication holds will be covered in the next lecture.)