Experts & Multiplicative Weight Updates (MWU)

Warm up: The 2-action Setting

There are n experts
On each day t the following happens:

1) Each expert i makes a prediction: Up / Down
2) The algorithm chooses Up or Down
3) The adversary reveals the actual outcome.
   - This outcome can depend on the alg's choice.

Define L_A to be the total "loss" of the algorithm, i.e., the # of mistakes.

Simplifying Assumption: Perfect Expert
↳ never makes a mistake

The Halving Algorithm

- Consider all experts $\mathcal{E}'$ with no mistakes so far
- Take the advice of majority in $\mathcal{E}'$.

Every time we make a mistake the size of $\mathcal{E}'$ reduces by a factor of 2. Thus, $L_A \leq \lceil \lg_2 n \rceil$.

No Perfect Expert

Let L_* be the total loss of the best expert.

Goal: Keep L_A close to L_*.

Lemma: There is an algorithm that guarantees

$$L_A \leq \lg n \cdot (L_* + 1).$$

The Generalized Halving Algorithm

Divide time into "epochs"

In each epoch, run the Halving algorithm:

* keep track of $\mathcal{E}'$, reset $\mathcal{E}' \leftarrow \mathcal{E}$ when $\mathcal{E}'$ gets empty starting a new epoch.

Analysis:

When we start a new epoch, all experts have made some mistake in the prev epoch. Thus:

$$L_* \geq \#\text{epochs} - 1.$$

On the other hand, in each epoch we make at most $\lceil \lg_2 n \rceil$ mistakes as each mistake reduces $\mathcal{E}'$ by a factor of 2. Thus:

$$L_A \leq \#\text{epochs} \cdot \lg n$$
$$\leq (L_* + 1) \lg n. \quad \square$$

Remark: There is an algorithm that guarantees

$$L_A \leq (2+\varepsilon) L_* + O\left(\frac{\lg n}{\varepsilon}\right).$$

Lemma: There is an adversary that guarantees $L_A \geq 2 L_*$.

Pf: Let's say there are two experts "Up" & "Down".
The "Up" expert always predicts Up.
The "Down" expert always predicts Down.

Adversary:
W/e the alg chooses, adv chooses the opposite outcome.

After T days, $L_A = T$. But $L_* \leq T/2$. $\square$

The General Setting

Each expert has its own independent suggestion.

(Think of this as n actions to choose from.)

On each day t:

1. The algorithm chooses a distribution $p^t = (p_1^t, \dots, p_n^t)$ over the experts.
   * we follow expert i w.p. p_i^t.
2. The adversary, aware of p^t, reveals the loss vector $\ell^t = (\ell_1^t, \dots, \ell_n^t)$ where $\ell_i^t \in [-1, +1]$ is the loss for expert i's suggestion.
3. The expected loss of the algorithm is

$$\ell_A^t = \langle p^t . \ell^t \rangle = \sum_{i=1}^{n} p_i^t . \ell_i^t$$

* $L_A = \sum_{t=1}^{T} \ell_A^t$
* $L_* = \min_i \sum_{t=1}^{T} \ell_i^t$. $\longrightarrow$ expert i's loss on day t

---

Suggestion 1: On day t, choose the advice of the best expert so far.

Adv: If the algorithm chooses expert i on day t, then let $\ell_i^t = 1$ & $\ell_j^t = -1$ for $j \neq i$.

$L_A = T$

$$L_* \leq \frac{T}{n} + \left(T - \frac{T}{n}\right) \times -1 = -T + \frac{2T}{n} = \left(\frac{2}{n} - 1\right) T$$

There is an expert that is the leader $\leq \frac{T}{n}$ times.

Thm (MWU): For every $0 < \varepsilon \leq \frac{1}{2}$, there is an algorithm that guarantees

$$L_A \leq L_* + \varepsilon T + \frac{\lg n}{\varepsilon}.$$

If there is no gain, i.e., $\ell_i^t \in [0, 1]$ then

$$L_A \leq (1+\varepsilon) L_* + \frac{\lg n}{\varepsilon}.$$

The Multiplicative Weight Update Algorithm.

MWU(ε):

* $w_i^1 \leftarrow 1$
* For $t = 1 \dots T$:
  - Follow expert i with prob $p_i^t = \dfrac{w_i^t}{\sum_{j=1}^{n} w_j^t}$
  - After ℓ^t is revealed, we set $w_i^{t+1} \leftarrow w_i^t . \exp(-\varepsilon \ell_i^t)$

$\approx w_i^t (1 - \varepsilon \ell_i^t)$

Call $L_A - L_*$ the "regret" of the algorithm.

We have low regret if it is $o(T)$.

Setting $\varepsilon = \sqrt{\frac{\lg n}{T}}$ we get $\sqrt{T \lg n} = o(T)$ regret.