You can try Problems 1 to 3 at home after the lecture; if you are stuck with one, ask during the tutorial for guidance, but attempt it on your own afterwards.

Problem 4 is quite technical: good practice, but don't have time during the tutorial or afterwards, feel free to skip it and only read the solution.

Problems 5 and 6 (Advanced) are good practice, a bit longer and less guided. Discuss them during the tutorial: attempt to fill in the gaps on your own or in groups if you don't have time to finish during the tutorial.

Warm-up

Problem 1. Try various parameters for Theorem 60: plot, for $\beta \in (0,1)$, the bound, when

- $C^* = 0, n = 1000$
- $C^* = 10, n = 1000$
- $C^* = 100, n = 1000$
- $C^* = 1000, n = 1000$

Do the same for Theorem 61.

Problem 2. Assume you know both n and (an upper bound on) C^* in advance. How would you set β in the MWU? In the Randomised MWU?

Problem 3. Prove Fact 56.3: namely, consider $n = 2$ experts, one predicting always 0 and the other always 1, and consider all 2^T possible sequences $(u_1, \dots, u_T)$. Show that for any deterministic algorithm A, there exists a sequence on which the algorithm makes T mistakes, and use this to conclude.

Problem solving

Problem 4. ($\star$) Suppose $C^* = C^*(T)$ is known in advance. We will show how to modify the MWU algorithm to achieve

$$C(T) \leq 2C^* + O(\sqrt{C^*(T)\log n} + \log n)$$

a) Argue that, if $C^* \leq \log n$, we are done.

b) Suppose $C^* > \log n$. Show how to achieve the desired bound by setting $\beta = 1 - \varepsilon$, for some suitable $\varepsilon = \varepsilon(C^*, n)$.

c) Conclude.

Problem 5. We again have n experts, each making a binary prediction at each time step. However, we would like to make sure we do well even if the best expert does badly overall, as long as for each "chunk" $I_{t_1,t_2} = \{t_1, t_1+1, \dots, t_2-1, t_2\}$, we do well compared to the best expert *for this chunk*.

To try and get this, consider the variant of the MWU, where we only penalise an expert by multiplying its weight by $1/2$ *if its current weight is at least* $1/3$ *of the average weight of all experts*.

We want to show that for every $1 \leq t_1 \leq t_2 \leq T$, the maximum number of mistakes $C(t_1, t_2)$ that the algorithm makes over I_{t_1,t_2} is at most $O(C^*(t_1,t_2) + \log n)$, where $C^*(t_1, t_2)$ is the number of mistakes made by the best expert *in that chunk*. (Considering $\beta \in (0,1)$ to be a constant, e.g., $\beta = 1/2$.)

a) Write down the algorithm.

b) Consider any chunk $I = I_{t_1,t_2}$, and let $t \in I$ be a time step where a mistake is made. Let W_t be the total weight at the beginning of step t, and W_G, W_B, W_L be the total weight of (1) experts who made a mistake, (2) experts who did not, and (3) experts who made a mistake but have weight less than $\frac{1}{3} \cdot \frac{W}{n}$. Bound the weight W_{t+1} at the end of step t as a function of W_G, W_B, W_L, β.

c) Bound the weight W_{t+1} at the end of step t as a function of W_t, β: show that

$$W_{t+1} \leq \frac{5+\beta}{6} W_t$$

d) Give a lower bound on the weight w_{i,t_1} of *any* expert i at time t_1 (start of the chunk). Namely, show that

$$w_{i,t_1} \geq \frac{\beta W_{t_1}}{3n}, \qquad 1 \leq i \leq n$$

e) Letting W_{t_1} the total weight at the beginning of the chunk, and W_{t_2} at the end, show that

$$W_{t_2} \geq \beta^{C^*(t_1,t_2)} \cdot \frac{\beta W_{t_1}}{3n}$$

f) Conclude.

Advanced

Problem 6. In the setting of the MWU, we have n experts, each making a binary prediction at each time step. Now, assume that we know that, for every $1 \leq k \leq n$, the k-th expert makes at most k mistakes.

a) What bound can you show on $C(T)$ when running the MWU algorithm with parameter β?

b) What bound can you show on $\mathbb{E}[C(T)]$ when running the Randomised MWU algorithm with parameter β?