James (Wervyn) Wert – gt6264a

1. **Weighted-Majority.** For this problem you may use either the deterministic or randomized weighted-majority algorithm.

   a) Suppose we have some initial belief about which expert is likely to be the best one. In that case, a natural modification to the Weighted-Majority algorithm is that instead of initializing all the weights to 1, we instead initialize $w_i = p_i$, where p_i is our initial belief that expert i is going to be best ($\sum_{i=1}^{n} p_i = 1$). Show how this results in a bound where the $\ln n$ term is replace d with $\ln(1/p_i)$. For example, if you pick the randomized algorithm, you should get a bound of:

   $$M \leq \frac{1}{\varepsilon}\left[m_i \ln\left(\frac{1}{1-\varepsilon}\right) + \ln\left(\frac{1}{p_i}\right)\right]$$

   where m_i is the number of mistakes of expert i. So this bound is better if our prior beliefs turn out to be reasonable.

   b) What if we have infinitely many experts? Use your answer to part (a) to show how you can replace $\ln n$ with $O(\log i)$ in comparing our performance to that of the ith expert.

For the first part, if we consider a deterministic weighted-majority algorithm and assign some initial weight p_i to the ith expert, then if after time T expert i has made m_i mistakes, his weight will be $p_i(1/2)^{m_i}$. As with the normal deterministic weighted-majority algorithm, every time we make a mistake it's because more than half of the probability distribution among the experts was in error, and so the total weight after reduction will be at most 75% of its prior value. The fact that there was some initial, uneven distribution is immaterial. So we know that for all i, $p_i(1/2)^{m_i} \leq (3/4)^M$. Taking the log of both sides, we solve for M on the left side:

$$p_i\left(\frac{1}{2}\right)^{m_i} \leq \left(\frac{3}{4}\right)^M$$

$$\left(\frac{4}{3}\right)^M \leq \left(\frac{1}{p_i}\right)2^{m_i}$$

$$M \log\left(\frac{4}{3}\right) \leq m_i \log 2 + \log\left(\frac{1}{p_i}\right)$$

$$M \leq 2.4\left[m_i + \log_2\left(\frac{1}{p_i}\right)\right]$$

James (Wervyn) Wert – gt6264a

2. **Tracking a moving target.** Here is a variation on the deterministic Weighted-Majority algorithm, designed to make it more adaptive.
   (a) Each expert begins with weight 1 (as before).
   (b) We predict the result of a weighted-majority vote of the experts (as before).
   (c) If an expert makes a mistake, we penalize it by dividing its weight by 2, but *only* if its weight was at least ¼ of the average weight of experts.

   Prove that in any contiguous block of trials (e.g., the 51st example through the 77th example), the number of mistakes made by the algorithm is at most $O(m + \log n)$, where m is the number of mistakes made by the best expert *in that block*, and n is the total number of experts.

The basic intuition here is that putting a lower bound on the weight of each expert will prevent any one expert from making so many mistakes over an arbitrarily long time so as to become insignificant, so that if they become the best expert for a period of time in the future, our algorithm will recognize it.

First, we must consider how the weight of the algorithm changes with each mistake it makes. The first consideration is the effect of the group of "unreliable" experts whose weight does not get halved: call this group U. Because $\forall i \in U, w_i \leq \frac{\sum_{j=1}^{n} w_j}{4n}$, then $\frac{\sum_{i\in U} w_i}{|U|} \leq \frac{\sum_{j=1}^{n} w_j}{4n}$. And, since $|U| < n$, $\sum_{i\in U} w_i < \frac{\sum_{j=1}^{n} w_j}{4}$. In other words, the total weight of U is strictly less than one quarter of the total weight among all experts. Now when the algorithm makes a mistake, it is because more than half of the weight of the algorithm predicted incorrectly. Assuming all of the experts in U make a mistake at the same time as us, this still means that at least ¼ of the misplaced weight belongs to experts not in U, and will be cut in half. Therefore, the total weight of the algorithm M mistakes after some time T will be at most $W_T(7/8)^M$ (where W_T is the weight of the algorithm at time T).

Now turning our attention to the best expert in the group during some block starting from time T, its own weight will be at most cut in half every time it makes a mistake (possibly less because of the lower threshold), making its weight at least $w_T(1/2)^m$. Further, we can approximate a lower bound for $w_T \approx \frac{W_T}{4n}$, and write $\frac{(1/2)^m}{4n} \leq (7/8)^M$ or $(1/2)^m \leq 4n(7/8)^M$. Solving for M, we find that $(8/7)^M \leq 4n \cdot 2^m$, $M \log(8/7) \leq m \log(2) + \log(4n)$, and so $M \leq O(m + \log n)$.

James (Wervyn) Wert – gt6264a

3. **2-Player Zero Sum Games.** In this problem, you will prove that the Nash equilibria of a 2-player zero sum game have several interesting properties. First, they all exhibit the same value. Second, they are *interchangeable*, meaning that given two Nash equilibrium points (σ_1, σ_2) and(τ_1, τ_2), the strategy pairs (τ_1, σ_2) and (σ_1, τ_2) are also Nash equilibria.

   Specifically, let G be a two person zero sum game, let A_i be the set of possible pure strategies for player i, $i = 1, 2$ and let $\pi : A_1 \times A_2 \rightarrow \mathbb{R}$ be the function describing the payoff value for player I, or the loss value for player II. The goal of player I is to maximize π, while the goal of player II is to minimize π. Let (τ_1, τ_2) and (σ_1, σ_2) be two (mixed) Nash equilibria for G. Prove:
   (a) $\pi(\tau_1, \tau_2) = \pi(\tau_1, \sigma_2) = \pi(\sigma_1, \tau_2) = \pi(\sigma_1, \sigma_2)$
   (b) Both (σ_1, τ_2) and (τ_1, σ_2) are Nash equilibria of G.

   Conclude that the set of Nash equilibrium points of a 2-player zero sum game is the Cartesian product of the equilibrium strategies of each player.

The key to understanding this behavior is the fact that the game is a zero sum one. In a general two-player game, the row player and the column player each have different utility functions, whereas here they share the same function. This means that we can define each player's payoff in terms of the other player's loss, and vice versa.

Consider two given Nash equilibria at $s = \pi(\sigma_1, \sigma_2)$ and $t = \pi(\tau_1, \tau_2)$. First, assume that $s < t$. In this case, we ask what the value of $u = \pi(\tau_1, \sigma_2)$. If $u < t$, then player II would have an incentive to change her strategy from τ_2 to σ_2, meaning (τ_1, τ_2) could not be a Nash equilibrium. So $u \geq t$. But if $s < u$, player I would want to change his strategy from σ_1 to τ_1, again contradicting the given. So $s \geq u$. Therefore our assumption was wrong, and $s \geq t$. We can similarly argue that $t \geq s$ using the value $v = \pi(\sigma_1, \tau_2)$, and so we have proven that $s = t$ and $\pi(\tau_1, \tau_2) = \pi(\sigma_1, \sigma_2)$.
Furthermore, since we have shown that $s \geq u \geq t$ and by implication $t \geq v \geq s$, then $s = t$ implies that $s = u = t$ and $s = v = t$, and so all four strategy combinations must be equivalent.

Finally, this implies that (σ_1, τ_2) and (τ_1, σ_2) are also Nash equilibria. In the case of (σ_1, τ_2), if we imagine some joint strategy (σ_1', τ_2) such that $\pi(\sigma_1', \tau_2) > \pi(\sigma_1, \tau_2)$, then $\pi(\sigma_1', \tau_2) > \pi(\sigma_1, \sigma_2)$, and (σ_1, σ_2) cannot be a Nash equilibrium either. Likewise, (σ_1, τ_2') such that $\pi(\sigma_1, \tau_2') < \pi(\sigma_1, \tau_2)$ implies that $\pi(\sigma_1, \tau_2') < \pi(\tau_1, \tau_2)$. Therefore, since any joint strategy that would cause (σ_1, τ_2) to not be a Nash equilibrium would also invalidate the given equilibria, no such strategy can exist, and (σ_1, τ_2) is a Nash equilibrium. We can make an identical argument for (τ_1, σ_2).

James (Wervyn) Wert – gt6264a

4. **External regret vs. Swap regret.** In Rock-Paper-Scissors, Rock beats Scissors (winner has loss 0, loser has loss 1), Scissors beats Paper, and Paper beats Rock; if both player play the same action, they tie (each gets loss ½).

   Consider playing T games of Rock-Paper-Scissors against an opponent who first plays Rock $T/3$ times, then plays Scissors $T/3$ times, then plays Paper $T/3$ times.

   a) Thinking of Rock, Paper, and Scissors as three "experts", describe in words what Randomized Weighted Majority would do against such an opponent. To be concrete, consider the version of RWM that, when expert i incurs loss l, updates using $w_i \leftarrow w_i(1-\varepsilon)^l$. Assume a learning rate $\varepsilon \gg \frac{1}{T}$, or if you like, you can think of $\lim_{e\to 1}$. Approximately (ignoring terms that are $o(T)$) what is the total loss of RWM and how does that compare to the loss of the best expert?
   b) What approximately is the *swap regret* of RWM?
   c) Since external regret is defined as the difference between the loss of the algorithm and the loss of the best expert, any two sequences of actions with the same total loss will result in the same external regret. Is this true for the swap regret? In the context of this Rock-Paper-Scissors example, is there a behavior with approximately the same total loss as RWM but with much less swap regret?

Thinking about this at a high level first, it's fairly obvious that while our RWM starts out favoring each strategy evenly, the strategy for Paper will quickly eclipse the other two options and become the de facto choice for the first third of the match. At the end of that period, Paper will have a weight of 1 while Rock will have a weight of $(1-\varepsilon)^{T/6}$ and Scissors will have a weight of $(1-\varepsilon)^{T/3}$. For this round, the approximate loss will be 0.
Next, during the Scissors round, our strategy will be heavily in favor of the worst strategy, and will generally lose rounds until halfway through, when the weight of the Paper strategy becomes $(1-\varepsilon)^{T/6}$, equivalent to the Rock strategy. At that point, Rock will shift to being the dominant strategy, and overall we will have won about half as many games as we lost. At the end of the round, Paper has weight $(1-\varepsilon)^{T/3}$, Rock still has weight $(1-\varepsilon)^{T/6}$, and Scissors will have weight $(1-\varepsilon)^{T/2}$.
Finally, the last phase will have Rock be the dominant strategy until the very end, and so we will lose almost all of the games. The probability distributions here will mirror the distributions in the first round for optimality, and so in total the approximation of losing none of the games in the first round, half the games in the second, and all of the games in the third, will even out. At the end, we should have a total loss of approximately $T/2$, which is the same loss we would have incurred if we played a pure strategy throughout. Thus the external regret of the algorithm tends to zero.

In terms of swap regret however, we could have achieve a much better result if we were to have played Scissors every time we played Rock during the second half of the match. The effect here would mean that while, for the second half of the second round we would incur an average loss of 1/2, during the third round we would take no losses. We would trade $T/12$ loss to gain back $T/3$, making the total swap regret $T/4$.

James (Wervyn) Wert – gt6264a

One simple algorithm that would achieve good swap regret would be to simply play Rock, Paper, or Scissors with a uniform random distribution. In this case, we would end up winning $T/3$ of the games, drawing $T/3$ of the games, and losing the final $T/3$, for a net loss of $T/2$. However, swapping any of our actions would not produce a better outcome, since each action wins, loses, or draws with equal probability over the whole match. Both the external regret and the swap regret of this algorithm are zero.