1 Problem 9

The problem tries to formally capture errors that frequently occur in implementations of the multiplicative wt update (MWU) method.

Part (a) says that "roundoff errors" can be very dangerous.

Specifically, suppose we have just two experts (A and B), and suppose that A makes a mistake for the first $t = (1/\eta)\log(1/\varepsilon)$ rounds, while B is correct during those rounds, and is wrong thereafter. After the first t rounds, due to a roundoff error the probability q_A becomes zero, while $q_B = 1$. Subsequently, q_B will reduce by $e^{-\eta}$, but since we normalize, and since $q_A = 0$, q_B will be reset to 1, leading to always picking B.

Thus after $T \gg t$ rounds, the best expert (A) makes only t mistakes, while the algorithm makes $T - t$ mistakes. Since we can pick T/t larger than any desired constant, we have a regret of $T - o(T)$.

Part (b) pursues the idea that we should not store the probabilities (due to roundoff issues as above), but instead store the number of mistakes. This way, in each round, we can compute the probabilities 'on the fly', and make a decision.

How is $p_t^{(i)}$ related to the number of mistakes $m_t^{(i)}$? We saw this in class:

$$p_t^{(i)} = \frac{e^{-\eta m_t^{(i)}}}{\sum_j e^{-\eta m_t^{(j)}}}.$$

Part (c) pursues the 'on the fly' idea above. Suppose using the m_t values, we compute q_t which approximates p_t. (Note that this way, we can get rid of issues like the one in part (a).)

Suppose $\sum_i |p_t^{(i)} - q_t^{(i)}| < \varepsilon$. Then we wish to claim that the mistake probability when sampling according to q_t is *almost the same* as the mistake probability when sampling according to p_t (which is the standard MWU).

Let S be the set of experts who made a mistake in the t'th step. Then the two mistake probabilities are $\sum_{i\in S} q_t^{(i)}$ and $\sum_{i\in S} p_t^{(i)}$. The difference between these two is $\sum_{i\in S}(q_t^{(i)} - p_t^{(i)})$, which is clearly $\le \sum_i |p_t^{(i)} - q_t^{(i)}| < \varepsilon$.

The standard MWU has a mistake bound of $(1+\eta)\min_i m_T^{(i)} + O(\log N/\eta)$. Now we have an additional probability at most ε of making an error. This gives the extra $+\varepsilon T$ term.

Part (d) tries to improve on this bound, knowing that choices are binary, especially in the case when $\min_i m_T^{(i)}$ is small.

Set $\eta = 1/10$, as in the problem statement, and consider step t. The key is the following: as we have binary predictions, there will be a set S_0 of experts who predict 0 and a set S_1 who will predict 1. Precisely one set gets it right, while the others make a mistake. Let $p_{t,0}$ be the probability that the infinite precision MWU algorithm predicts 0 and $p_{t,1}$ the probability that it predicts 1. Now in part (c), if we can estimate these two numbers to an accuracy $\varepsilon = 1/T$, then we get the desired bound! (I.e., we don't need all the p_t's.)

Now, how can we estimate $p_{t,0}$? We use the idea of part (b) to store the number of mistakes. We know the expression for $p_{t,0} = \sum_{i \in S_0} p_t^{(i)}$, where p_t's are as in part (b). We can compute each $p_t^{(i)}$ to an additive $1/(NT)$ using only $O(\log(NT))$ word-size (exercise), and thus we can compute the sum over S_0 to an additive $1/T$ error using the same precision.