UPenn NETS 412: Algorithmic Game Theory
Homework 2

Instructor: Bo Waggoner
Due: by beginning of class, February 8, 2018
Turn in *electronically* via Gradescope.

Problem 1 (6 points)

Consider a game in which two firms selling the same product make simultaneous quantity choices. Co-operation is ruled out. If Firm 1 chooses to produce q_1 units and Firm 2 chooses q_2 units, the market will assign the price for each unit to be $1 - q_1 - q_2$. Both firms have a constant unit cost of production of $c < 1$. Suppose Firm 1 chooses a quantity q_1 and Firm 2 chooses a quantity q_2 with $q_1 + q_2 \leq 1$. Then firm 1's profit will be

$$q_1(1 - q_1 - q_2) - cq_1,$$

and Firm 2's profit will be

$$q_2(1 - q_1 - q_2) - cq_2.$$

Notice that the players (the firms) want to *maximize* utility. So in this context, an ordinal potential function ϕ is one whose change in sign is always the *negative* of the change in sign of player's utility.

Part a (3 points) Show that the function:

$$\phi(q_1, q_2, c) = -cq_1q_2$$

is *not* an ordinal potential function for the game.

Part b (3 points) Consider the function:

$$\phi(q_1, q_2, c) = -q_1q_2[1 - q_1 - q_2 - c].$$

Show that it is an ordinal potential function for the game.

Problem 2 (10 points)

Now we consider the expert-advice problem from a probabilistic perspective. Consider the expert setting, with the following difference: Each expert errs each time step with some probability p. That is, each round t, if the true stock movement is $o^t = U$, then each expert correctly predicts U independently with probability $1 - p$, and incorrectly predicts D with probability p. Similarly, if the true stock movement is $o^t = D$, then each expert independently predicts D with probability $1 - p$ and D with probability p. There are k experts. Suppose we play for T rounds.

Part a (4 points) What is the expected number of mistakes that any given expert will make? What is the probability that *no* expert will make a mistake in a given round? What is the probability that *every* expert will make a mistake on a given round? What about the probability that *at least* one expert will make a mistake in a given round?

Hint: What distribution can we use to represent the number of mistakes?

Part b (2 points) Now suppose that every period, we ask the experts to vote, and make the prediction chosen by the larger half of the experts (assume k is odd, so you need at least $(k+1)/2$ to be correct). Let Q be the probability that we will make the correct prediction. Give an expression for Q (i.e., describe how to calculate Q). Give an expression for the expected number of rounds that we will make the correct prediction (in terms of Q).

Part c (2 points) Fix a particular expert. What is the probability that, after t rounds, she has not made any mistakes?

Part d (2 points) Suppose there are two experts. One makes mistakes with probability p as described above, but the other is always perfectly correct. You plan to observe them for t rounds, then guess which is the perfect one. What is the chance you guess correctly?

Hint: Use your answer to the previous part.

Part e (1 (bonus) points) Suppose there are k experts, one of which is perfect while the others all err indpendently with probability p. You want to guess the perfect one, and you want your guess to be wrong with probability at most 0.01. After how many rounds of observation can you make your guess? Give the number of rounds and prove the probability of error is at most 0.01. (Solve for $k = 2$ for half a point.)

Hint: You may use the "union bound", which states that the probability that any of n events occurring is upper-bounded by the sum of their probabilities.

Problem 3 (10 points)

Part a (3 points) Consider a Follow-the-Leader expert advice algorithm in which there are N experts indexed $\{1, 2, \ldots, N\}$. At each time step, you look for the expert who has made the fewest mistakes thus far and predict the same action as her (if there is more than one such expert, break ties by picking the one with the lowest index).

What regret is guaranteed by this algorithm? Justify your answer. You can describe it in words or use asymptotic notation.

Part b (3 points) Now, consider an Unweighted Majority expert advice algorithm in which there are N experts, and at each time step, you predict the same action as the majority of the experts (we can assume N is odd, so there are no ties).

What regret is guaranteed by this algorithm? Justify your answer. You can describe it in words or use asymptotic notation.

Part c (4 points) Both of these algorithms are *deterministic*, in that there is no element of randomness in the selection of the action. Explain why no deterministic algorithm can do significantly better than the algorithms described in (a) and (b).

Problem 4 (5 points)

Similar to a zero-sum game, a constant-sum game is one in which at every action profile, the utilities of the players sum to some constant k. Observe that a zero-sum game is a special case of a constant-sum game where $k = 0$.

Argue that the Von Neumann MinMax theorem holds for constant-sum games, even if $k \neq 0$.

Hint: How would you transform a constant-sum game into a zero-sum game? Why is this transformation valid?

Problem 5 (10 points)

For this problem you will be implementing the polynomial weights algorithm. Along with the homework we have provided the file "experts.csv". This file shows the realization of predictions of 10 experts over 1000 time steps. In the file each row is a single time step and each column is an expert. A "0" represents a correct prediction from the column's expert and a "1" is incorrect. The columns are separated by spaces. You can use any language you want, include your code with your submission.

Part a (3 points) For this part, we want to determine the best expert. At each time step, determine the best expert so far, and the average number of mistakes of that expert up to that time step. (e.g. an expert with 10 mistakes in the first 100 steps has an average of 0.1). Print these values to a file called "opt.csv". Each line should take the form "expert index,average mistakes", and your file should have 1000 lines. Submit this file along with your write up, and in your write up note the best expert and their optimal regret for the last time step.

Part b (5 points) For this part, you will implement the polynomial weights algorithm. Use the randomized algorithm to choose an expert at each time step for your prediction. The loss of your algorithm is the number of times you choose an expert that predicts incorrectly; keep track of the regret of your algorithm at each time step. Make another file called "pwa.csv". Each line should take the form "expert index,average mistakes" where the first number is the index of the expert chosen by your algorithm and the second number is the average loss of the algorithm up to that time step. (e.g. if the algorithm has made 20 mistakes in the first 100 time steps, the average loss is 0.2.) Submit this file along with your write up, and in your write up note the final average regret of your algorithm and the final weights of each expert at the end.

Part c (2 points) Finally, plot the **average regret** of your algorithm over time. Time should be on the horizontal axis, and average regret should be on the vertical axis. In your write up, interpret the results and compare to the theoretical guarantee proved in class. Submit this plot as a file called "regret.pdf" along with your write up, or include it as a figure inline under this subproblem.

Part d (1 (bonus) points) Run some more experiments, produce some plots, and report your findings. You can try other values for epsilon in the algorithm, or consider the average regret from running the algorithm many times and how that compares to just one round, or so on.