IE613: Online Machine Learning **Jan-Apr 2017**

Assignment 2: March 17

Instructions: *You are free to code in Python/Matlab/C/R. Discussion among the class participants is highly encouraged. But please make sure that you understand the algorithms and write your own code. Submit the code by* 24*th March.*

Question 1 (Full information setting) *Consider the problem of prediction with expert advice with* $d = 10$*. Assume that the losses assigned to each expert are generated according to independent Bernoulii distributions. The adversary/environment generates loss for experts* 1 *to* 8 *according to* $Ber(.5)$ *in each round. For the* 9*th expert, loss is generated according to* $Ber(.5 - \Delta)$ *in each round. The losses for the* 10*th expert are generated according to different Bernoulii random variable in each round– for the first* $T/2$ *rounds, they are generated according to* $Ber(0.5+\Delta)$ *and the remaining* $T/2$ *rounds they are generated according to Bernoulii random variable* $Ber(0.5-2\Delta)$*.* $\Delta = 0.1$ *and* $T = 10^6$*. Generate (pseudo) regret values for different learning rates (*η*) for each of the following algorithms. The averages should be taken over atleast* 50 *sample paths (more is better). Display* 95% *confidence intervals for each plot. Vary c in the interval* [0.1 2.1] *in steps of size* 0.2 *to get different learning rates.*

- *Weighted Majoirty algorithm. Set* $\eta = c\sqrt{2\log(d)/T}$
- *Follow the Regularized Leader (FTRL) with linear loss functions and negative entropy as the regularizer. In round* $t = 1, 2, \cdots, T$*, the linear function is* $f_t(w) = w'v_t$*, where* $v_t \in \{0, 1\}^d$ *is loss vector generated in round* t*. What is the optimal* η*? Let the optimal value be* η^**. Set* $\eta = c\eta^*$*.*

Question 2 (Bandit setting) *Consider the problem of multi-armed bandit with* $K = 10$ *arms. Assume that the losses are generated as in Question* 1*. For each of the following algorithms generate (pseudo) regret for different learning rates (*η*) for each of the following algorithms. The averages should be taken over atleast* 50 *sample paths (more is better). Display* 95% *confidence intervals for each plot. Vary c in the interval* [0.1 2.1] *in steps of size* 0.2 *to get different learning rates.*

- *EXP3. Set* $\eta = c\sqrt{2\log(K)/KT}$*.*
- *EXP3.P. Set* $\eta = c\sqrt{2\log(K)/KT}$*,* $\beta = \eta$*,* $\gamma = K\eta$*.*
- *EXP-IX. Set* $\eta = c\sqrt{2\log(K)/KT}$*,* $\gamma = \eta/2$*.*

Question 3 *Comment on the performance of EXP, EXP3.P and EXP-IX. Which one performs better? Can you think of a your own method which performs better than any of the above methods. Give pseudo-code of your algorithm and compare its performance with others.*

Submission format: Your should submit a report along with your code. Please zip all your files and upload via moodle. The zipped folder should named as YourRegistrationNo.zip. The report should contain two figures: one figure should have two plots corresponding to each algorithm in Q.1 and the other should have 3 plots one corresponding to each algorithm in Q.2. For each figure, write a brief summary of your observations. Specify the value of η^* you set in Q.1. For Q.3, write pseudo-code of your algorithm and show its regret plot along with that of algorithms in Q.2.

Please calculate (pseudo) regret for each algorithm in Q.2 for a given set of parameters as follows:

Let μ_t^i denote the mean of arm i in round t. Suppose adversary generates sequence of loss vectors $\{l_t\}_{t=1}^T$ and an algorithm generates sequence of pulls $\{I_t\}_{t=1}^T$, the (pseudo) regret for this sample path is

$$\sum_{t=1}^{T} \mathbb{E}[l_t(I_t)] - \min_i \sum_{t=1}^{T} \mathbb{E}[l_t(i)] \tag{2.1}$$

$$= \sum_{t=1}^{T} \mu_t^{I_t} - \min_i \sum_{t=1}^{T} \mu_t^i \tag{2.2}$$

Note that in this calculation we only considered the mean values of losses, not the actual losses suffered. It is Okay if this value turns out to be negative. There is no expectation over random choices of I_ts here. Now generate 50 such sample paths and take their average.