CS 747: Assignment 1 - Sampling Algorithms

Sumrit Gupta - 170040044

September 25, 2020

1 Implementation Details

For all the algortihms, I have used numpy.argmax() to select the max term of the corresponding algorithm. In numpy.argmax function, tie breaking between multiple max elements is so that the first element is returned.

1.1 Epsilon-greedy

E-greedy is implemented by doing round robin twice for each arm and then exploiting the optimal arm w.p. 1-eps, and exploring (uniform sampling) w.p. eps.

1.2 UCB

In this case, I am doing round robin for each arm once initially and then choosing the arm by argmax of the $\text{ucb}^{t}{}_{a}$ term as mentioned in the slides.

1.3 KL-UCB

For KL-UCB, I am using the value of c = 0, as this was more efficient practically. To find the max value of q in [p,1], I used binary search algorithm with a precision of 1e-4. I am doing round robin for each arm once initially and then choosing the arm by argmax of the term given in slides.

1.4 Thompson Sampling

In this case, I have initialised the value of success and failure for the prior term of all arms as 0. Then I am finding the argmax from beta distribution as per the algorithm given in slides, and pulling the arm corresponding to argmax.

1.5 Thompson Sampling with Hint

Since, our algorithm has access to the true means of the arms, so we can use this fact to decide the prior for beta distribution in order to minimise the regret. I have initialised the prior as success = 1e4*(maxMean) and failure = 1e4 - success for each arm where maxMean is the maximum mean among all the arms. Due to this large value initialisation of the prior, the small changes in success and failures while running thompson sampling algorithm won't affect the mean of the optimal arm, it would remain close to the original value whereas mean of all other arms would reduce in the long run and regret incurred would be low.

2 T3: Experiments on epsilon-greedy

For the experiments in T3, I took the average REG of 50 random seeds (0 through 49) for the horizon of 102400 and the values of e1, e2, e3 such that e1 < e2 < e3 and regret of e2 is less compared to the other two are -

- For instance 1 - e1 = 0.002, e2 = 0.02, e3 = 0.6
- For instance 2 - e1 = 0.004, e2 = 0.1, e3 = 0.7
- For instance 3 - e1 = 0.005, e2 = 0.013, e3 = 0.4

3 Plots

From the plots of T1, we can infer that E-greedy performs the worst for all the instances and thompson sampling perfomrms the best, kl-ucb being closest to thompson sampling. From the plots of T2, it can be seen that Thompson Sampling with Hint performs much better than thompson sampling for all horizons and all instances.

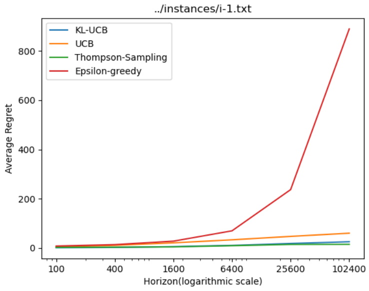

Figure 1: Plot for instance i-1

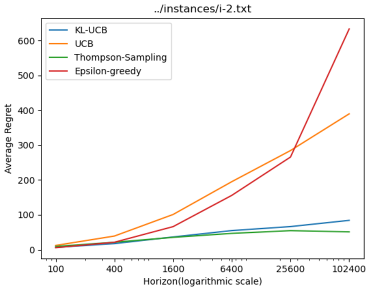

Figure 2: Plot for instance i-2

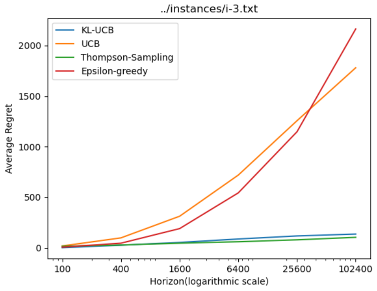

Figure 3: Plot for instance i-3

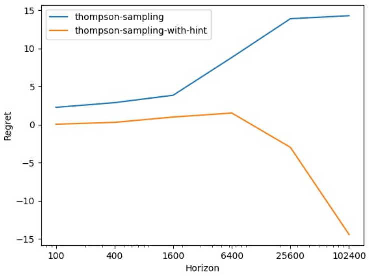

Figure 4: Plot for instance i-1

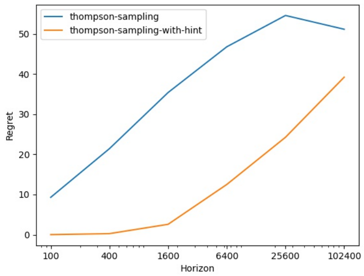

Figure 5: Plot for instance i-2

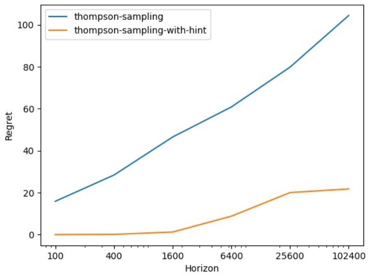

Figure 6: Plot for instance i-3