CS 747

Assignment - 1

Roll no. 160050062

Description of submitted files :-

The folder contains 1 sub-folder :

> submission :- Contains bandit.sh, wrapper.sh, allbandits.py, outputData.txt, report.pdf, references.txt
>
> **bandit.sh :-** takes arguments as specified in the problem statement, pass those arguments to allbandits.py file, runs the code and outputs a single line with 6 comma separated entries.
>
> **wrapper.sh :-** runs the bandit.sh file for all 3 instances, 7 horizon values, 7 algorithms (counting epsilon greedy as 3) and 50 seed values (from 0 to 49) and outputs the values on terminal
>
> **allbandits.py :-** It contains all the algorithms. It gets called by bandit.sh.
>
> **outputData.txt :-** Contains 7350 lines output of wrapper.sh script

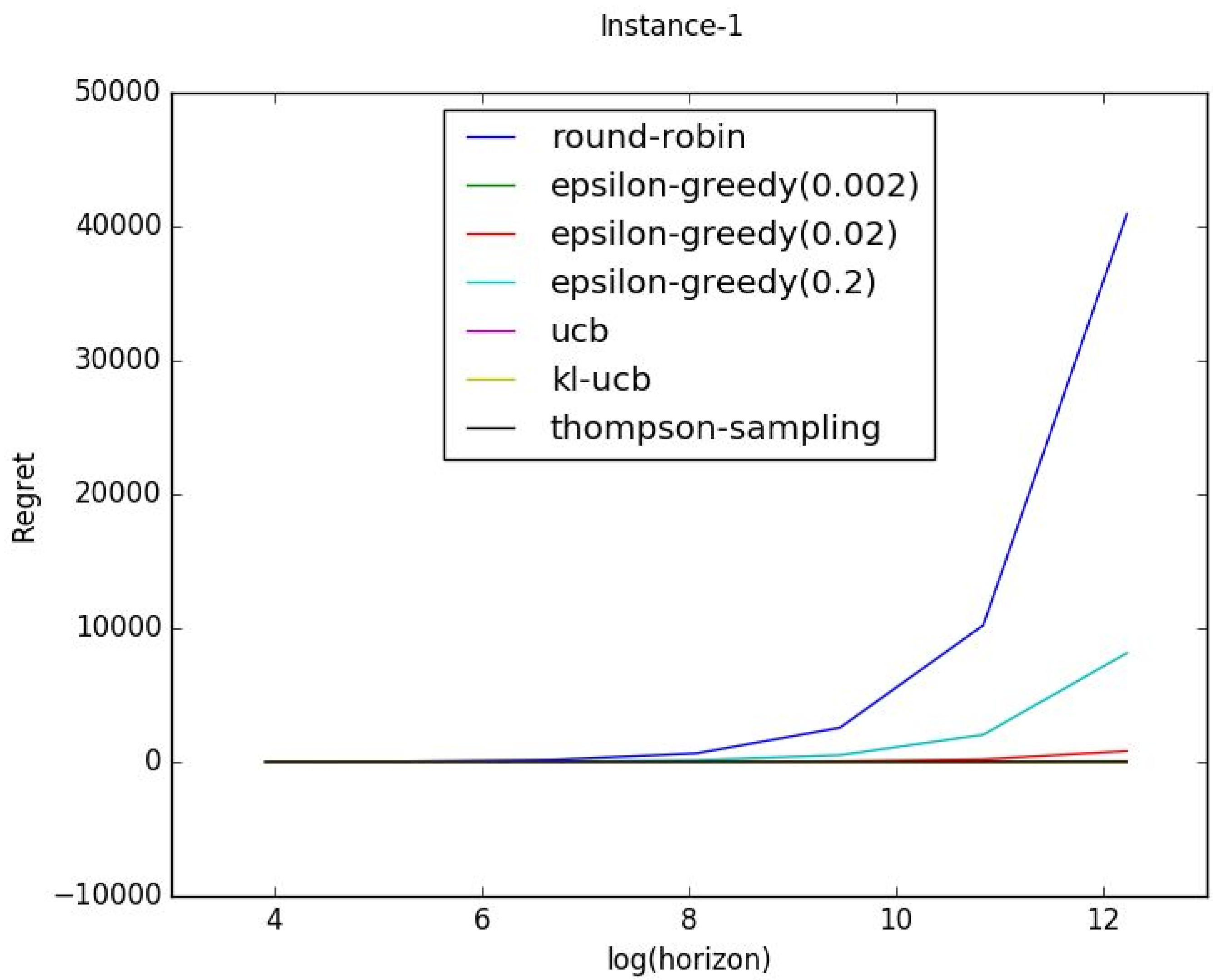
Some assumptions made :-

All the 5 algorithms are coded in allbandits.py file. Here I am writing some assumptions that I have made :

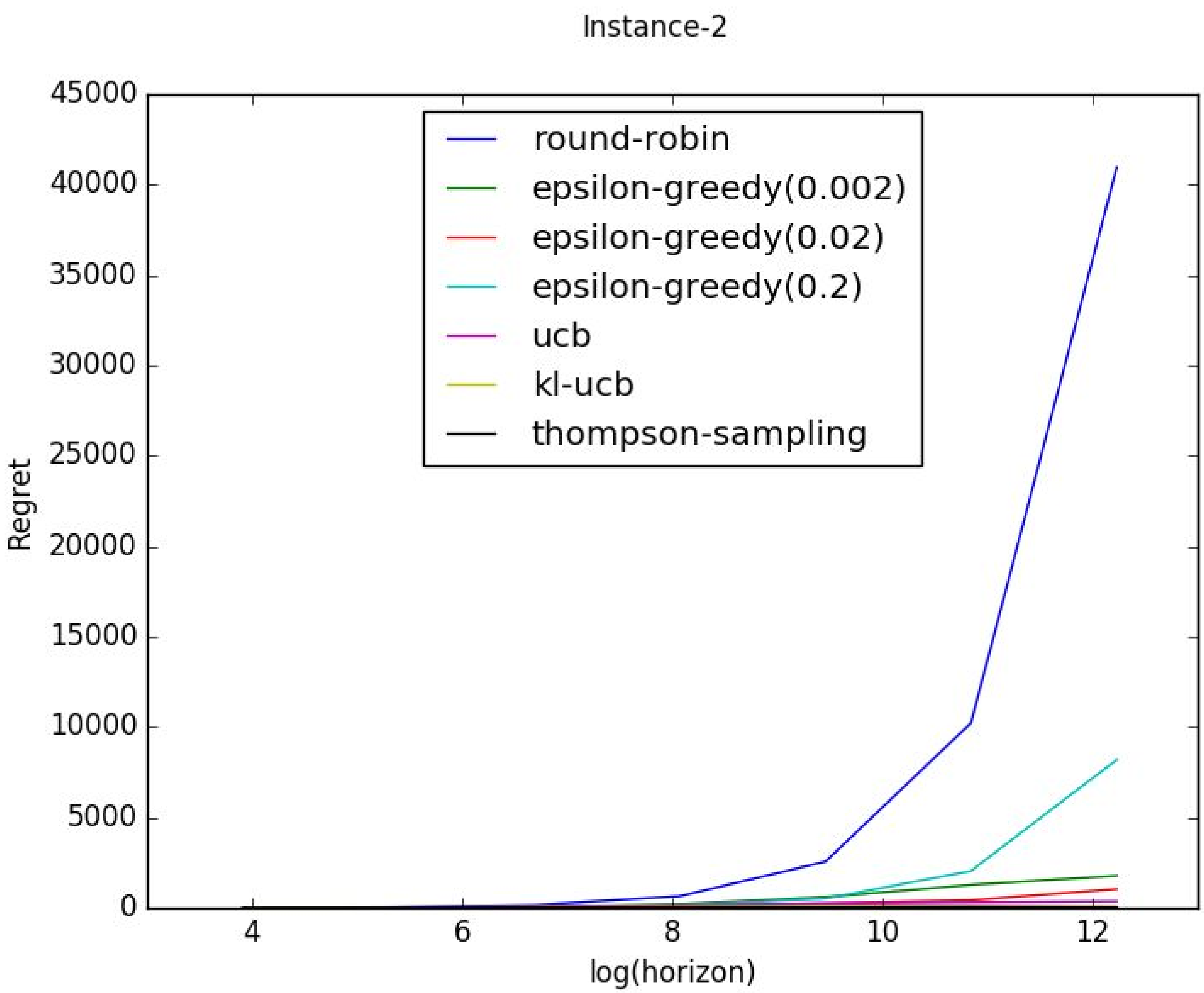
1. There will always be 10 command line arguments to bandit.sh script.
   ./bandit.sh --instance ../instances/i-1.txt --algorithm round-robin --randomSeed 1 --epsilon 0.5 --horizon 320
2. For round-robin algorithm, say, no of arms are 10 and horizon is 15. Then, first of all I will pull all arms once. After that, for the remaining 5 pulls, I will pull first 5 arms. Here, another thing that could be done is choosing these 5 arms randomly. But I have opted the simplistic way.
3. For epsilon-greedy algorithm, with probability (1 - ε), we choose the arm with highest empirical mean. In case there are more than one arm with highest empirical mean, I am putting the indexes of these arms in a list and randomly selecting an element of this list. Here I could have deterministically chosen one, but I felt this way is little better.
4. For UCB as well, we take the arm with highest ucb value to pull. Here also, I have done the same thing as above for choosing arm with highest ucb.
5. For kl-UCB, we have to solve an inequality (get the maximum value of q such that kl-divergence of p and q is less than certain quantity, where p is the empirical mean of an arm). To solve the inequality, I have used **Bisection method with tolerance of 1e-7.** After getting the q values for all arms, I picked the arm with maximum q value by the same method as mentioned in point 3
6. For thompson sampling, we have to pick arm with highest value, where values are picked from beta distribution (parameterized on no of successes and failures of that arm). Again to choose Highest beta value, I have used above mentioned method(point 3).

The plots for all 3 instances are as follows :-

Instance - 1

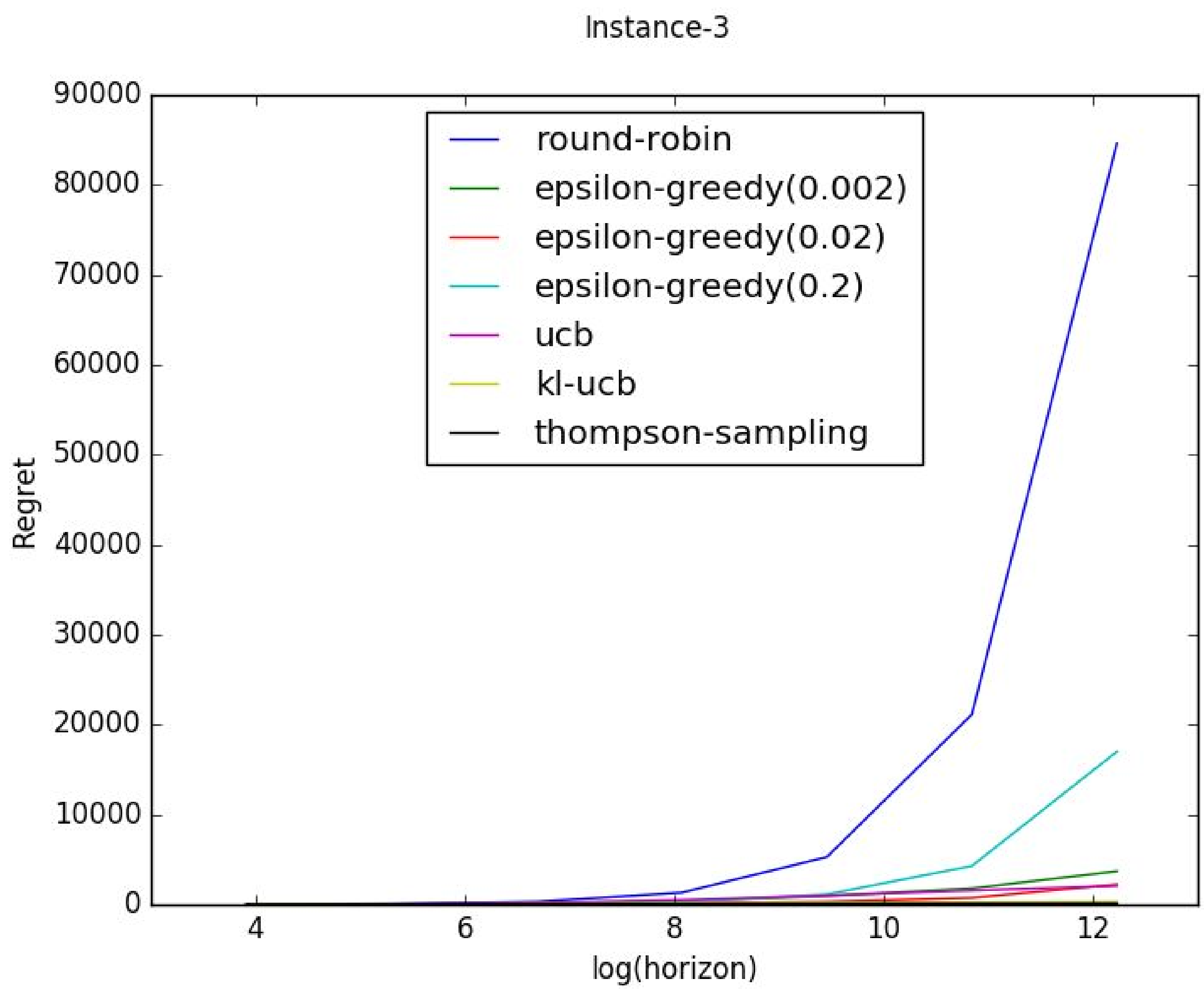

Instance-1
round-robin
epsilon-greedy(0.002)
epsilon-greedy(0.02)
epsilon-greedy(0.2)
ucb
kl-ucb
thompson-sampling
Regret
log(horizon)
50000
40000
30000
20000
10000
0
−10000
4
6
8
10
12

Instance - 2



Instance-2
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
Regret
4
6
8
10
12
log(horizon)
round-robin
epsilon-greedy(0.002)
epsilon-greedy(0.02)
epsilon-greedy(0.2)
ucb
kl-ucb
thompson-sampling

Instance - 3



Instance-3
round-robin
epsilon-greedy(0.002)
epsilon-greedy(0.02)
epsilon-greedy(0.2)
ucb
kl-ucb
thompson-sampling
Regret
log(horizon)
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
4
6
8
10
12

Some key observations :-

1. From all plots, we can infer that **round-robin is worst performing algorithm** in terms of regret.
2. In case of **epsilon-greedy algorithm,** the parameter epsilon controls the exploration-exploitation trade-off. This is depicted in the plots as well. In plot-3, we can see the order of regrets are as:

   epsilon-0.2 > *epsilon-0.002* > *epsilon-0.02*

   0.02 is the best value for epsilon. With epsilon 0.002, we are **not exploring enough**. With epsilon 0.2, we are **not exploiting enough**.
3. **Thompson sampling** is the best performing algorithm. Although **kl-UCB** is also not behind. But other algorithms aren't so close to any of them.
4. Rate of increase of regret is linear with respect to log(Horizon) for thompson sampling, kl-ucb and ucb.
5. Rate of increase of regret is exponential with respect to log(Horizon) for round-robin and epsilon greedy algorithms
6. Time complexity of KL-ucb is the worst and round-robin's is best, followed by epsilon-greedy, then thompson sampling.

Thus, summarising everything, we can conclude that thomson sampling is best in terms of regret. KL-ucb is also good but takes time. Epsilon-greedy depends on epsilon value for the regret and it can make a lot of difference as well.

Some key observations :-

1.