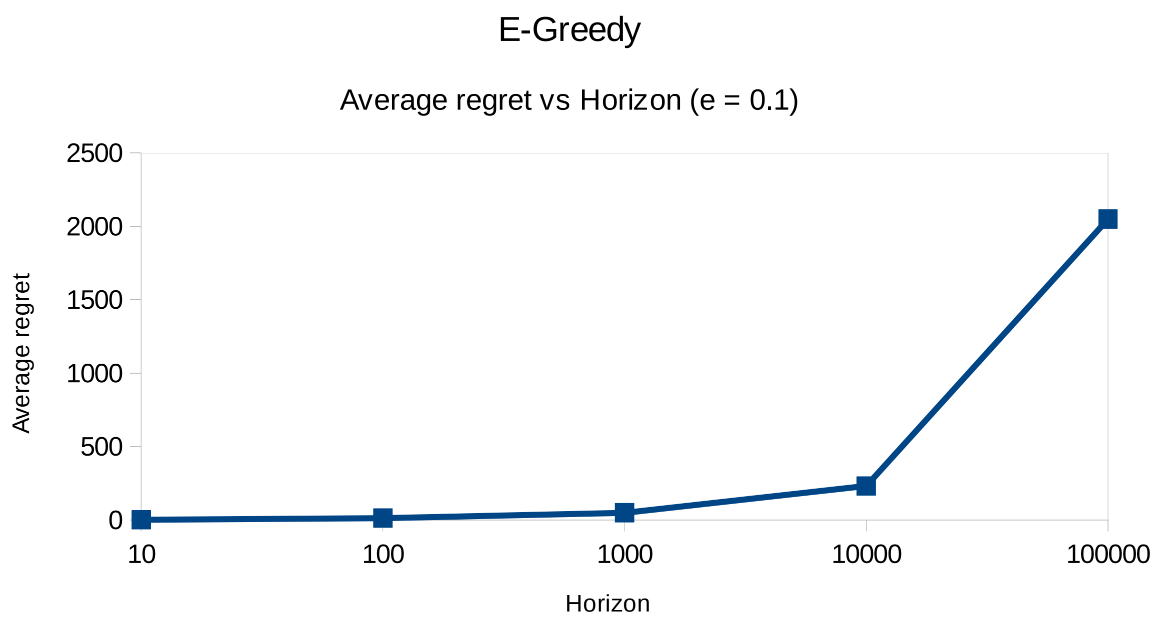
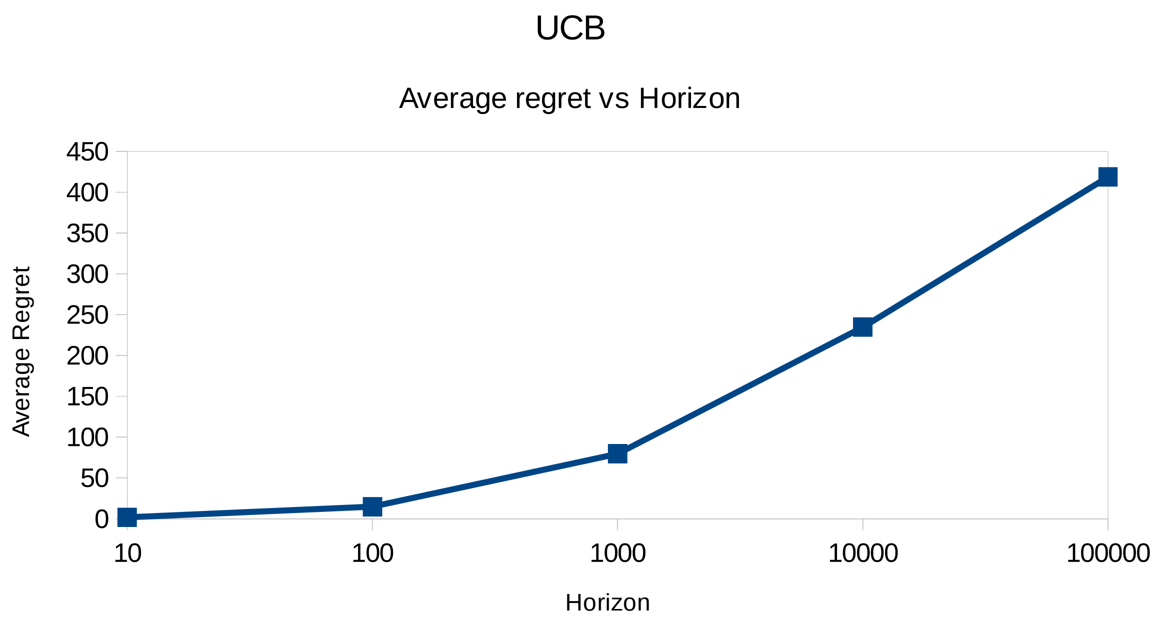
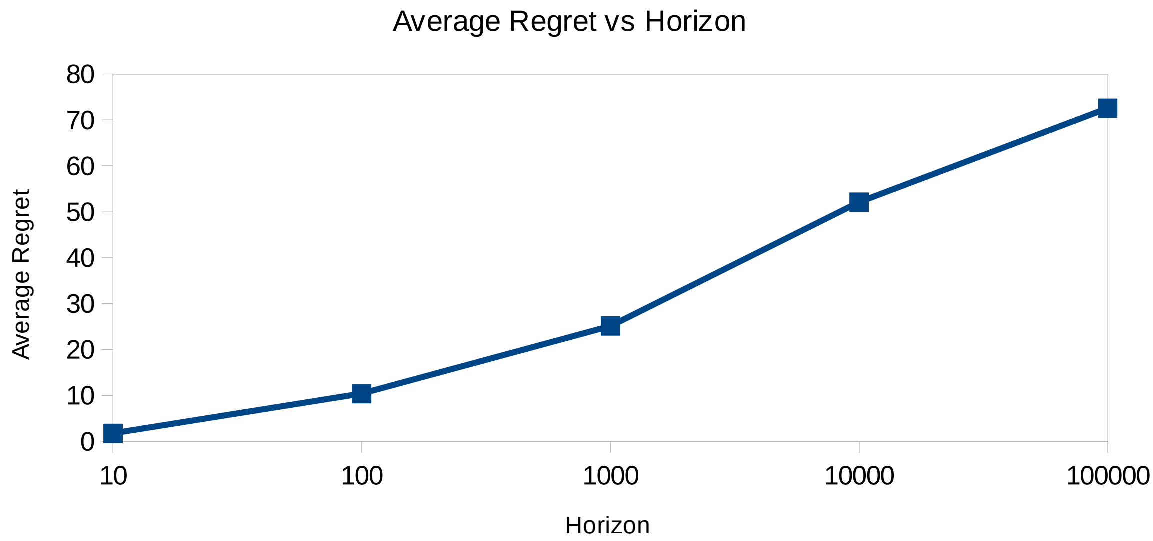
FILA Assignment 1
140050080
A.Srinath

instance – 5.txt

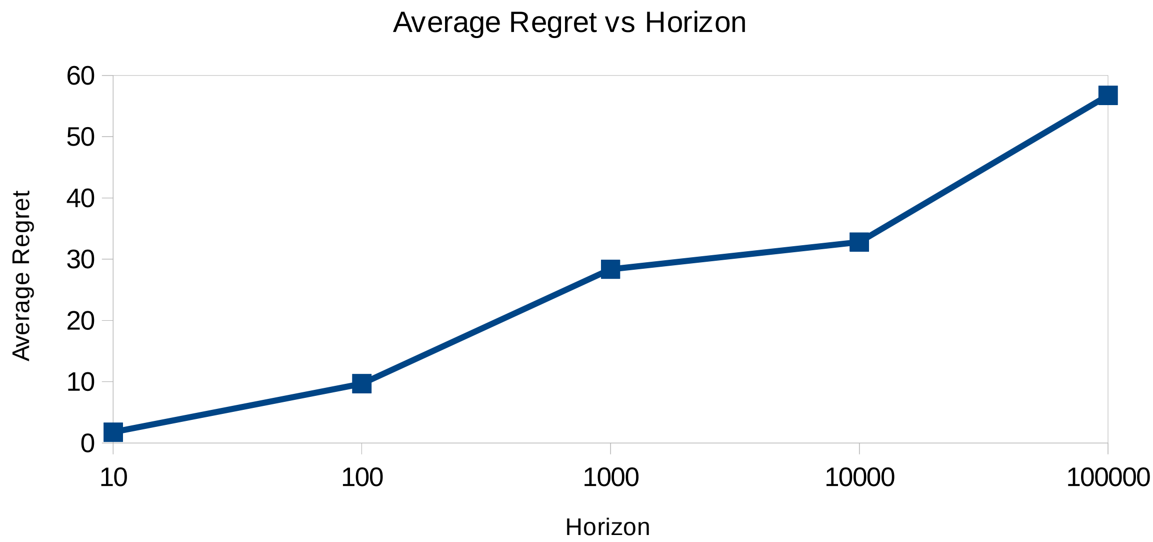


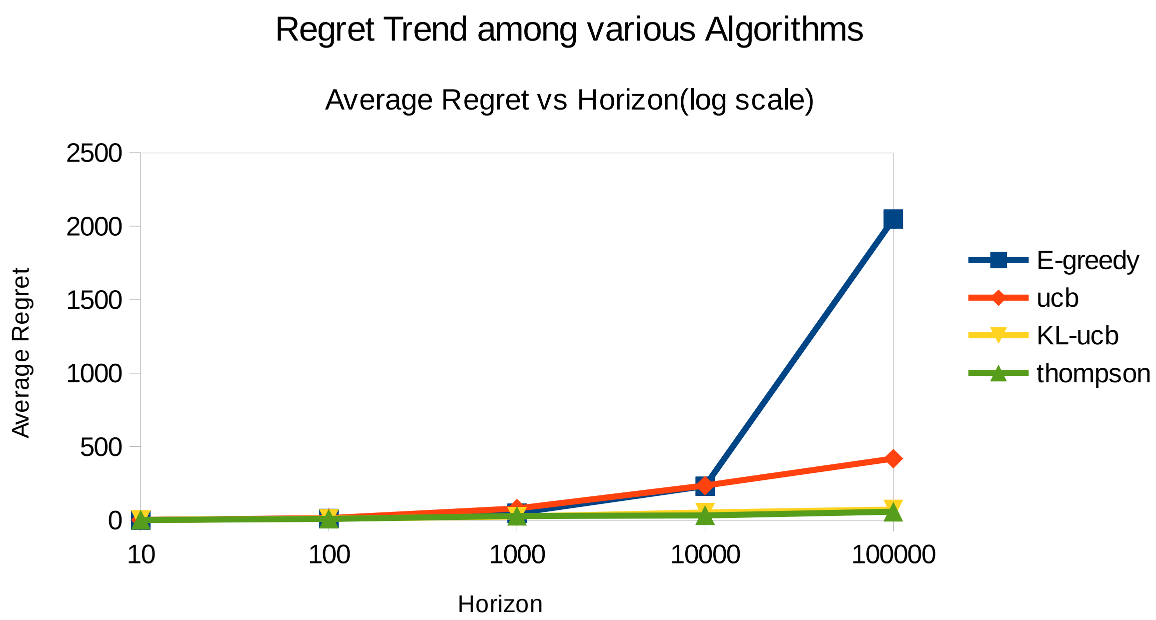


KL-UCB

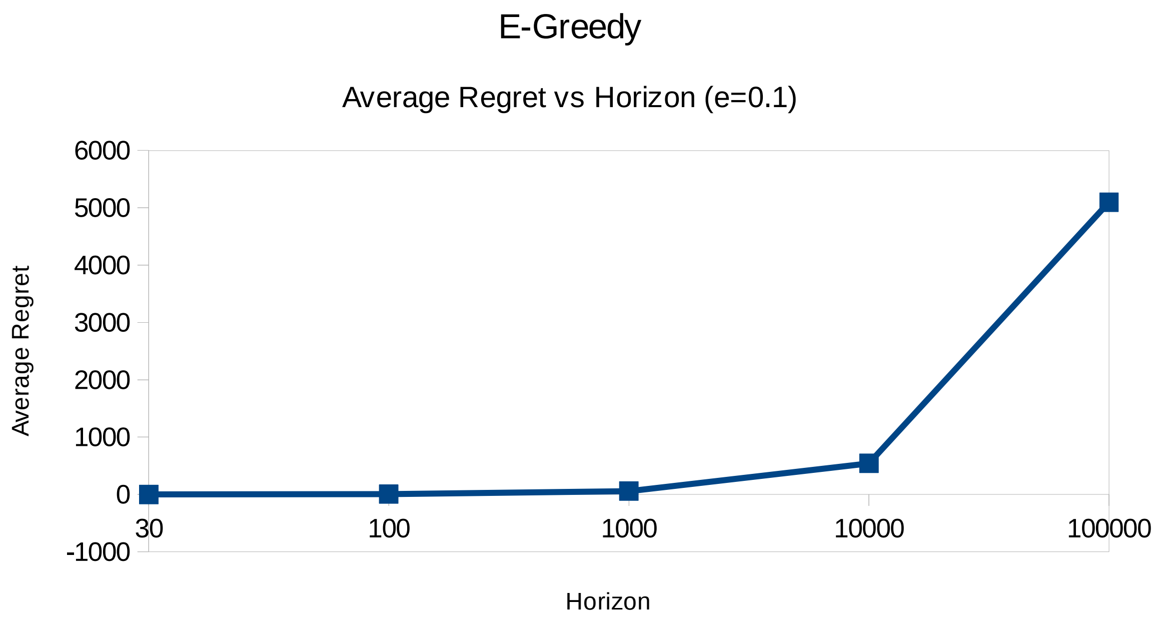


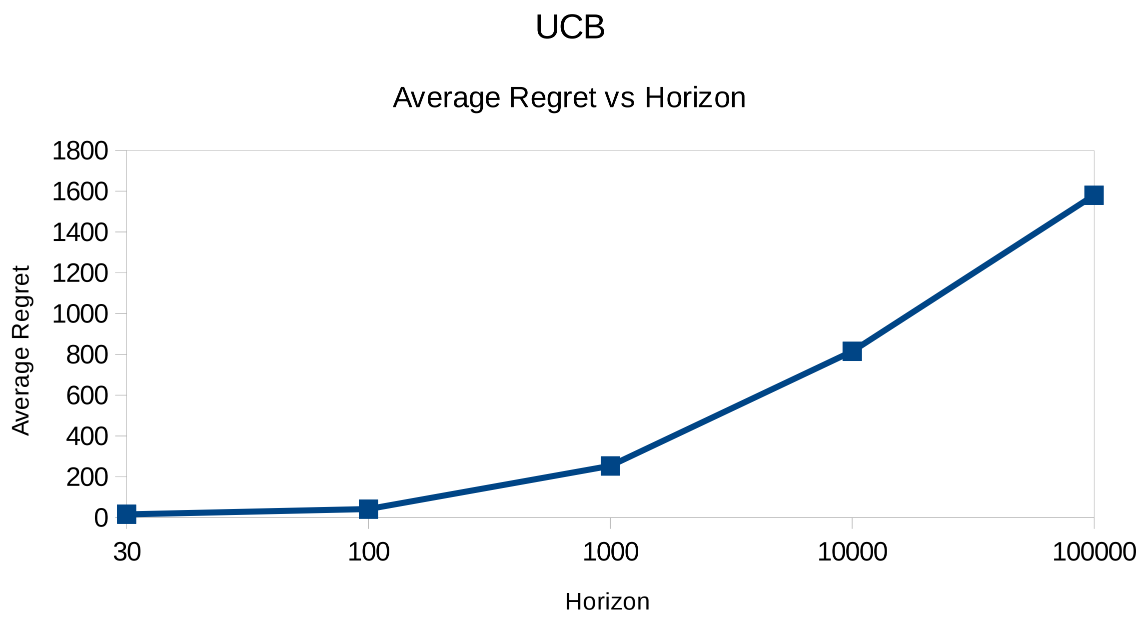
Thompson Sampling



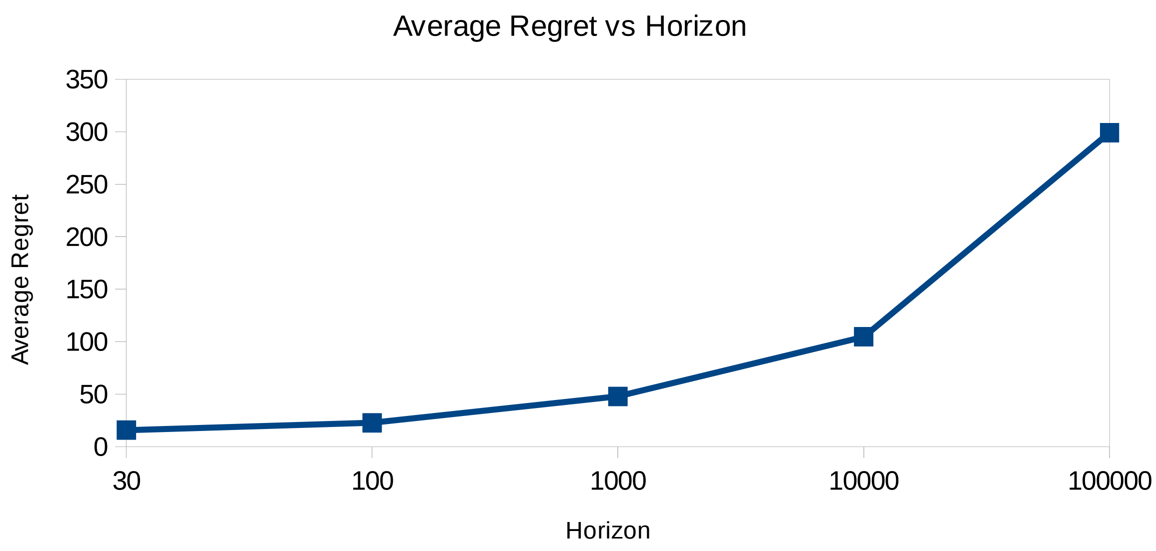

Regret Trend among various Algorithms
Average Regret vs Horizon(log scale)
Average Regret
2500
2000
1500
1000
500
0
10
100
1000
10000
100000
Horizon
E-greedy
ucb
KL-ucb
thompson

instance – 25.txt

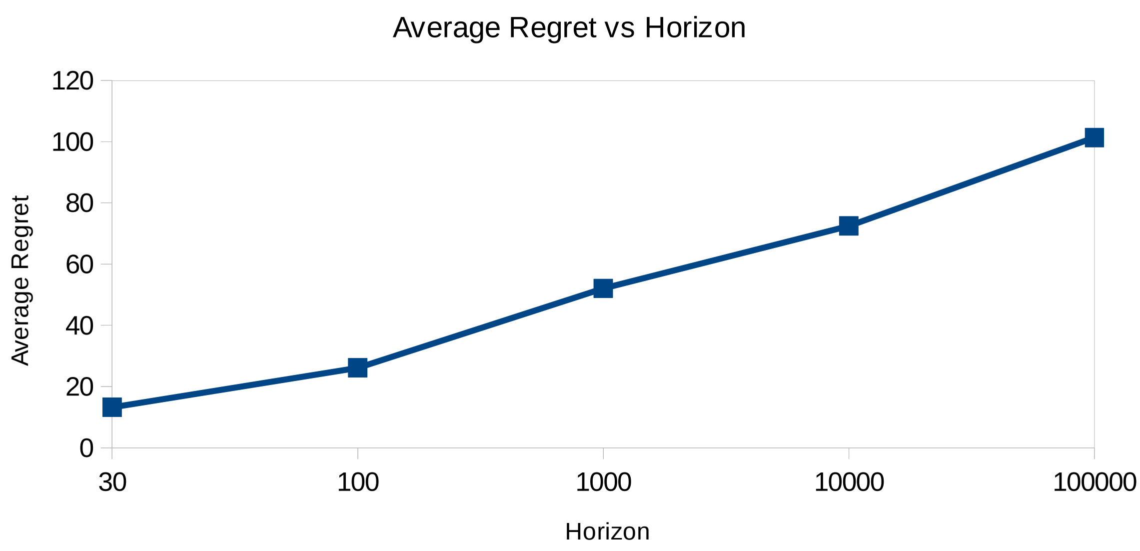




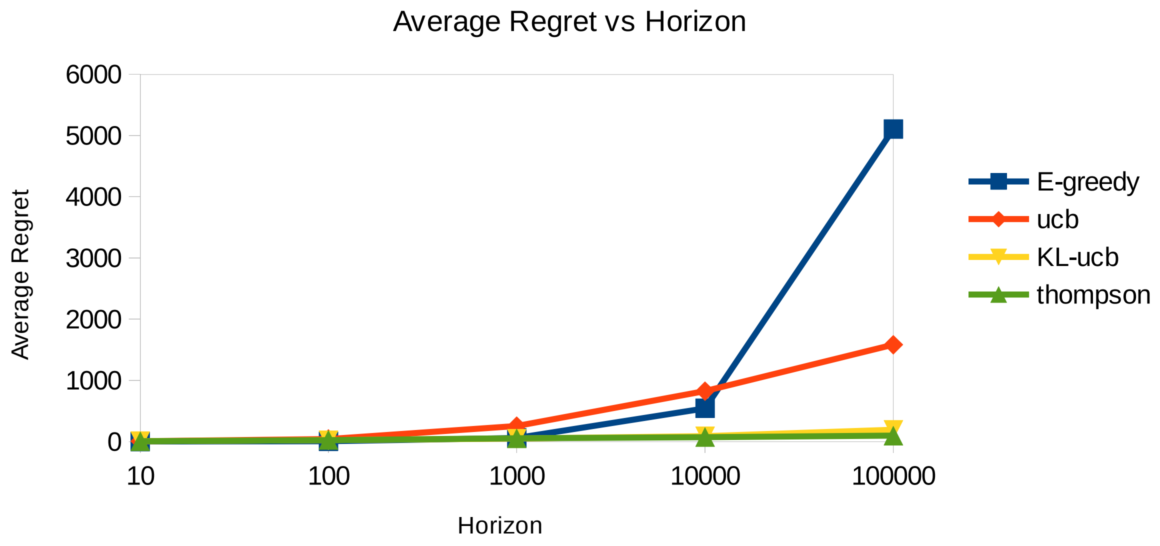
KL-UCB



Thompson Sampling



Regret Trends across Algorithms



Observations:

1) From the graphs it is clear that Thompson sampling works best for multi armed bandit problem among the 4 algorithms tested

2) In all the algorithms, as the horizon increases the probability of picking optimal arm increases

3) To rate the algorithms according to their performance it is
Thompson sampling > KL-UCB > UCB > E-greedy (more better is the one which has least average regret for sufficiently large horizon)

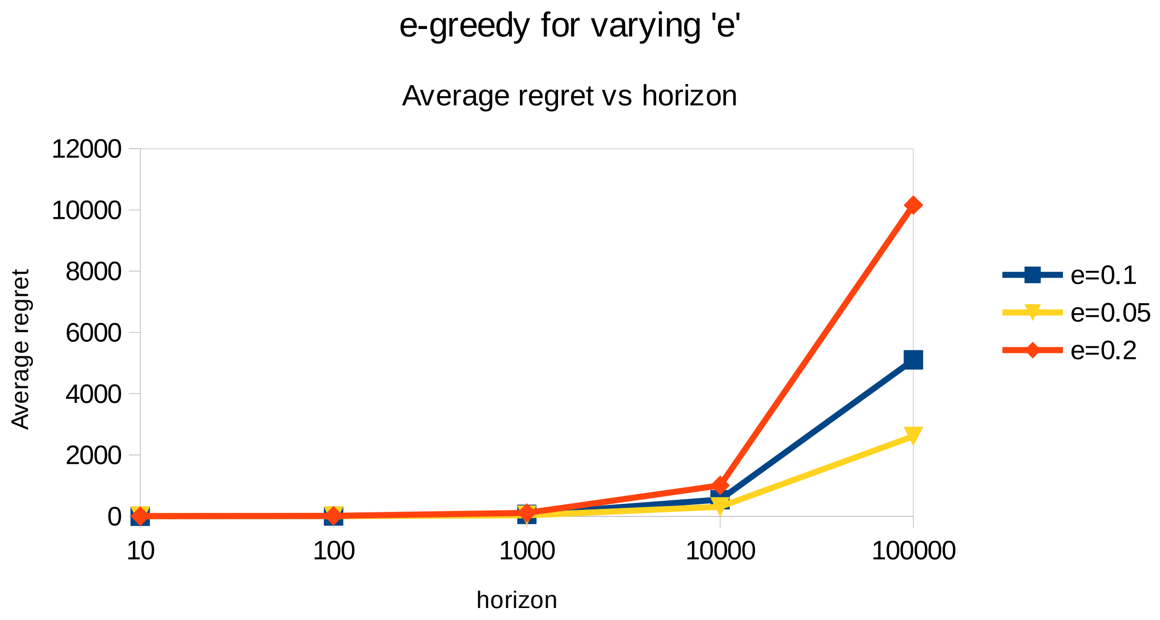
4)Comparing the best with the worst (on instance 25.txt, horizon = 100000) E-greedy gives avg regret of 5104 where as Thompson sampling gives avg regret of 97. clearly a lot more times better than e-greedy, i,e it chooses optimal arm with very high probability than e-greedy.

5)If we clearly observe then ,e-greedy performs better than UCB till certain point(say horizon <=10000) and after that a sharp increase (i,e a lot of deviation from UCB) can be seen. This is because even after a lot of trials, e-greedy performs random action with probability ‘e’ which might not pick the optimal arm. This can be improvised by using techniques such as ‘e(n)-greedy’ ,i,e to decrease value of ‘e’ as time increases and eventually make it to take only optimal action.

6) KL-UCB performs significantly better than UCB even for small time horizons.This is because the average regret of KL-UCB is tightly bounded than that of UCB, can be deduced from Pinsker’s inequality, d(μa, μa ∗) > 2(μa−μa ∗) ^2 .

7) regret of UCB, KL-UCB and Thompson-Sampling is O(log n). where n is horizon, althought the constants differ significantly from UCB to KL-UCB.

8) for instance-25 ,e-greedy ‘e’ is tuned a little bit and e=0.1 gives reasonable performance,sample graphs are shown below



it looks like if we decrease ‘e’ average regret decreases, but thats not in general. specific to my implementation, I pick the last arm in case of ties, so when we explore less,it initially picks last one(which has maximum mean in instance-25 <–> coincidence!) hence it mostly keeps picking it and average regret drops.

Implementation Details:

E-greedy:

with probability ‘e’ random arm is pulled
with probability ‘1-e’ largest mean(observed till now) arm is pulled

UCB:

if all arms are not pulled at least once, pull one which has not been pulled.
Else pull the arm that maximizes (Xavg+ sqrt((2*log n)/Nx))

KL-UCB:

if all arms are not pulled at least once, pull one which has not been pulled.
Else pull arm that maximizes

[max (q(0-1) such that Nx*d(Xavg , q) ≤ log(n) + c log(log(n))]

where d(x,y) is KL divergence. And c is taken to be 0 for practical purposes

Thompson-Sampling:

initialize all arms beta distribution to uniform

Repeat:

sample v1,v2,...vk from the distribution of each of the k arms.

Pull the arm with maximum ‘vi’.
Based on the reward obtained change the distribution of that arm which is pulled.