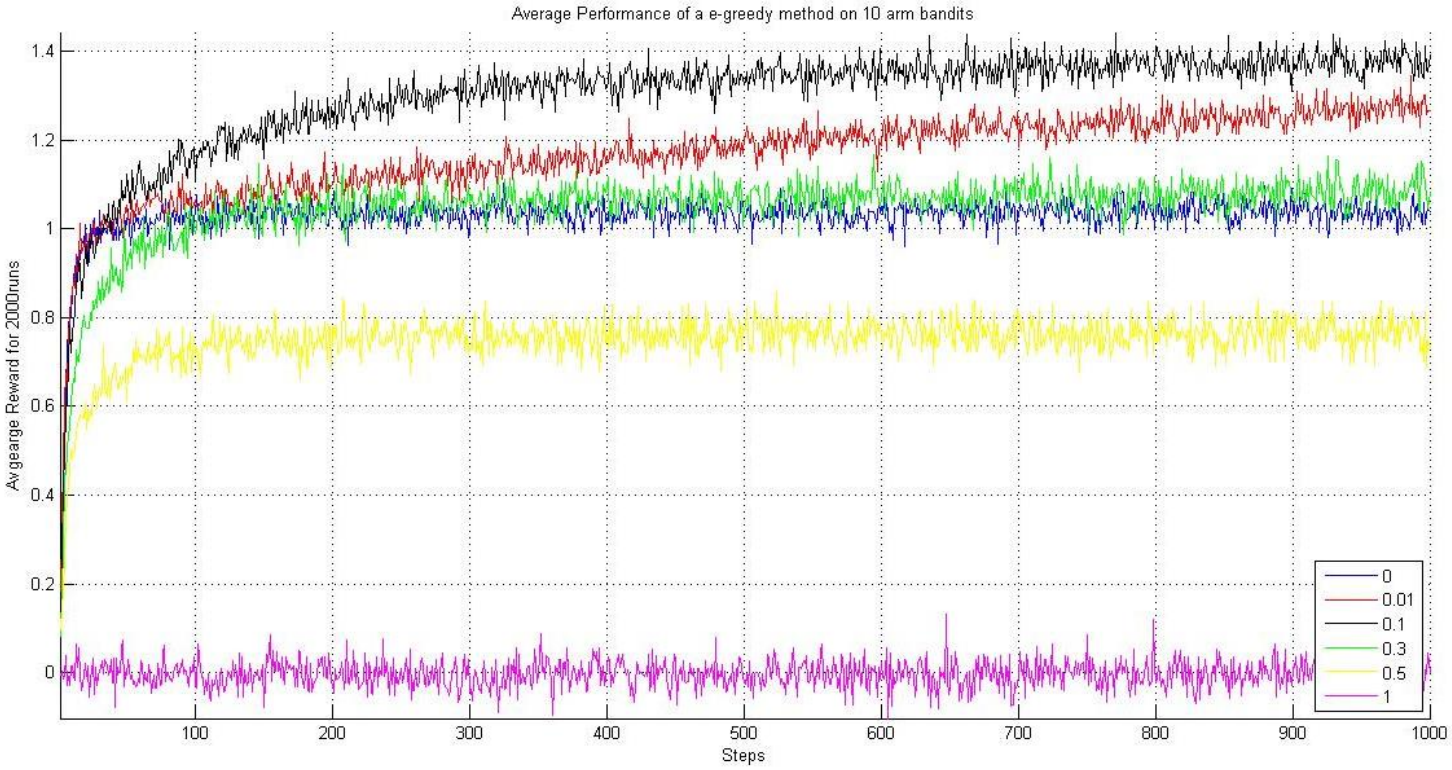
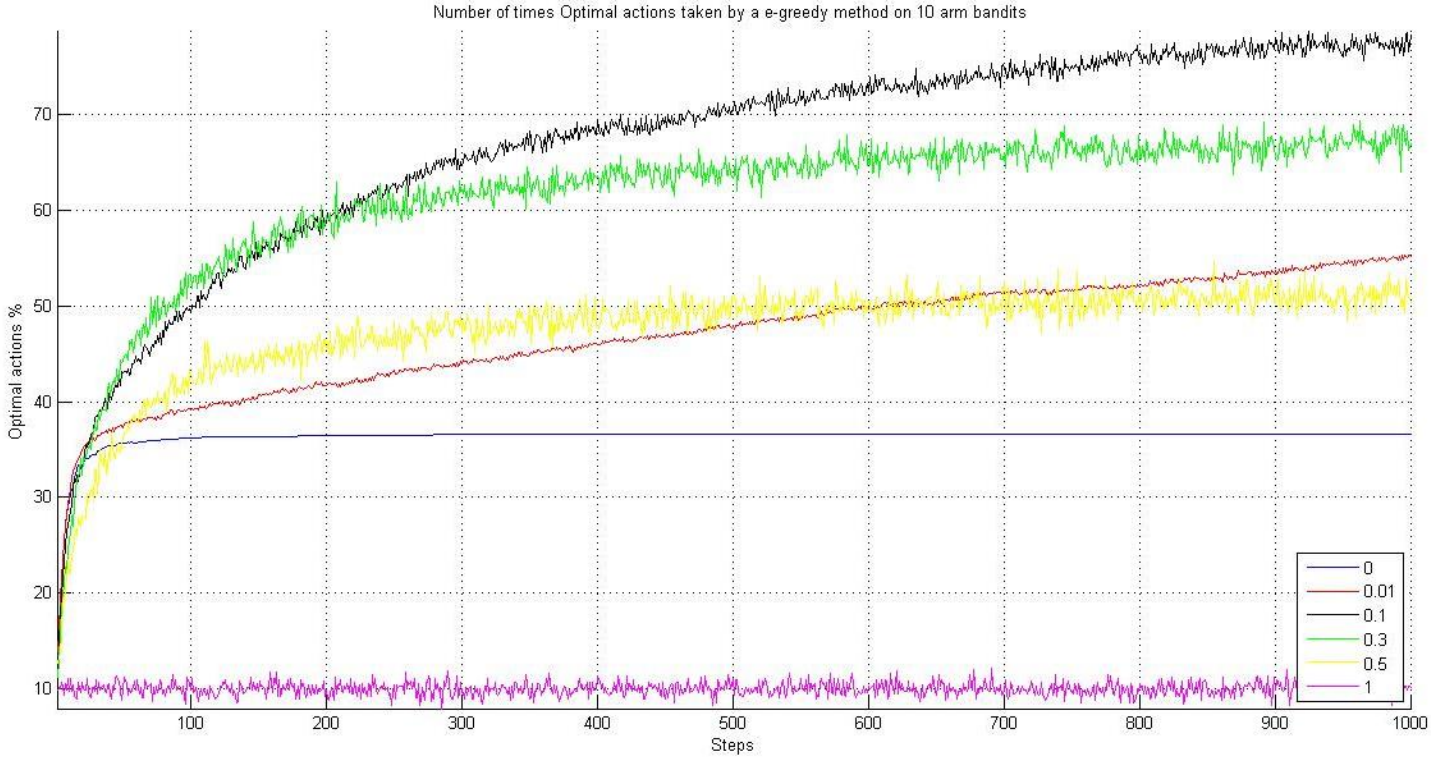
CS6700: Reinforcement Learning PA1

Question-1:

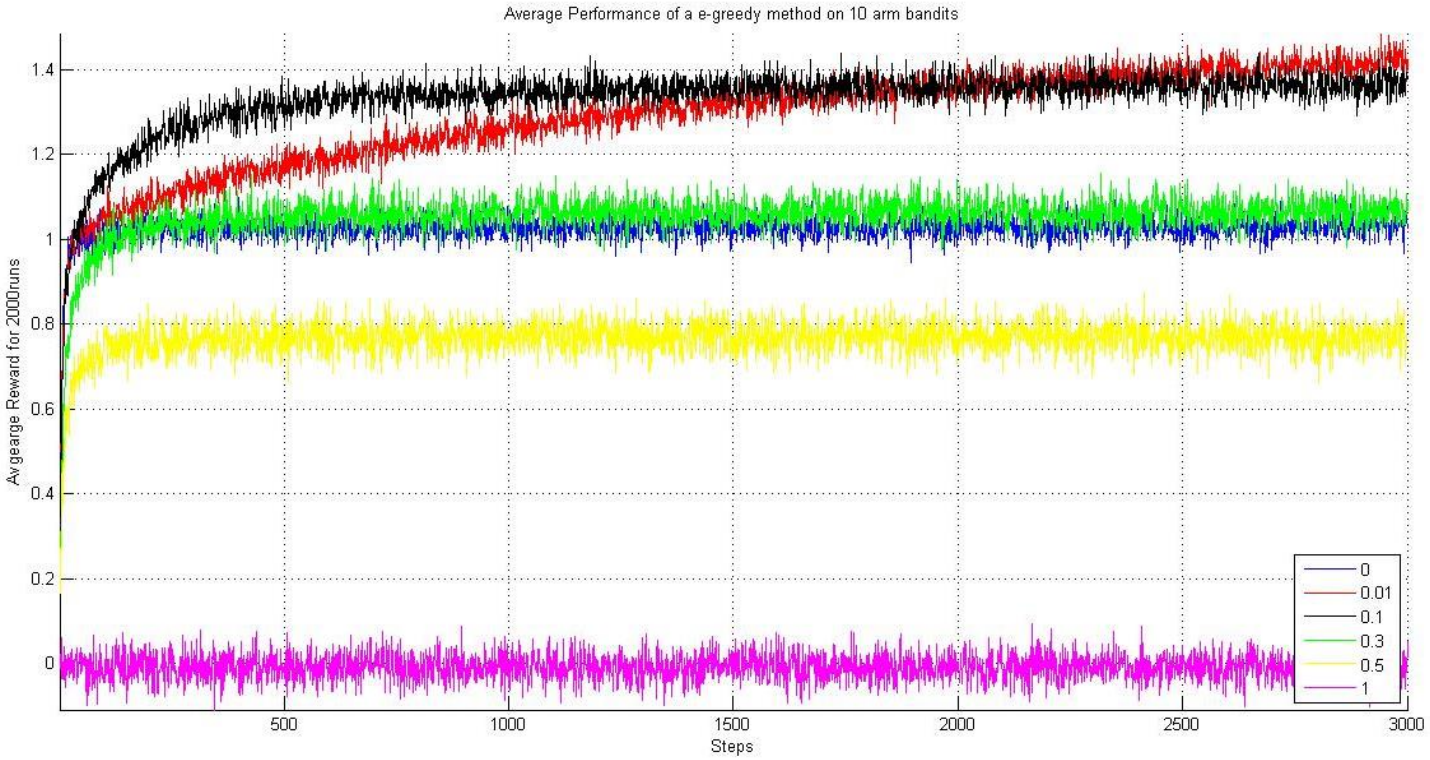
- To reproduce the results achieved in this section, run the Q1.m
- Experiments were conducted for different epsilons $\{0,0.01,0.1,0.3,0.5,1\}$ for 1000 steps and 2000 runs.
- Choice of epsilons: $\{0\ and\ 1\}$ was chosen based on the fact that they correspond to extremes i.e. $\{0\}$ correspond to greedy and $\{1\}$ correspond to complete exploratory. Remaining epsilons are chosen to find out how the epsilons effect the average reward.
- Experiments were also conducted for 3000 runs to check whether average reward for epsilon $\{0.01\}$ will exceed than those achieved from epsilon $\{0.01\}$
- Observations are as follows.



- Clearly, epsilon value $\{0.1\}$ gives the best performance out of all other values.
- Epsilon value $\{1\}$ performance is as expected poor.
- We can see that too much exploration epsilon value $\{0.5\}$ gave poor performance compared to greedy setting.



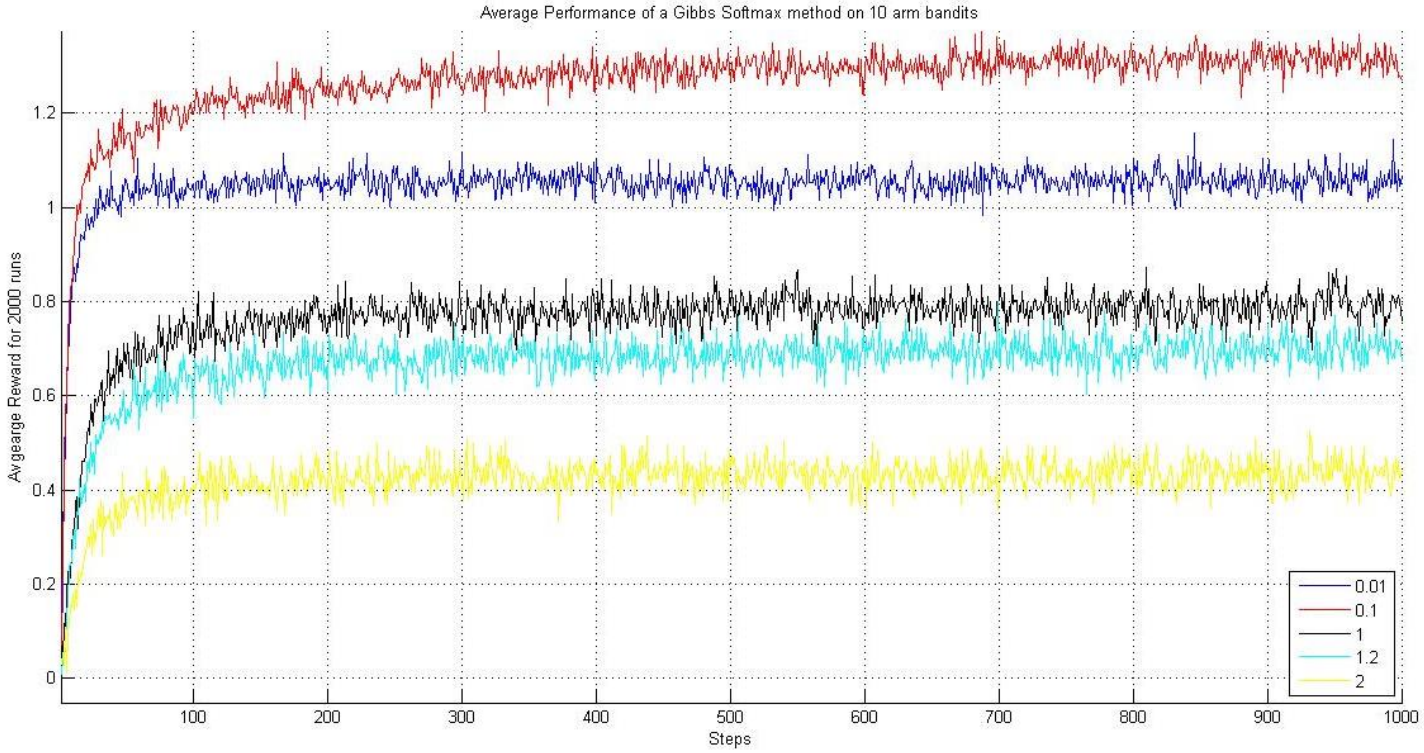
- Optimal actions selection performance resulted as given in the text book.



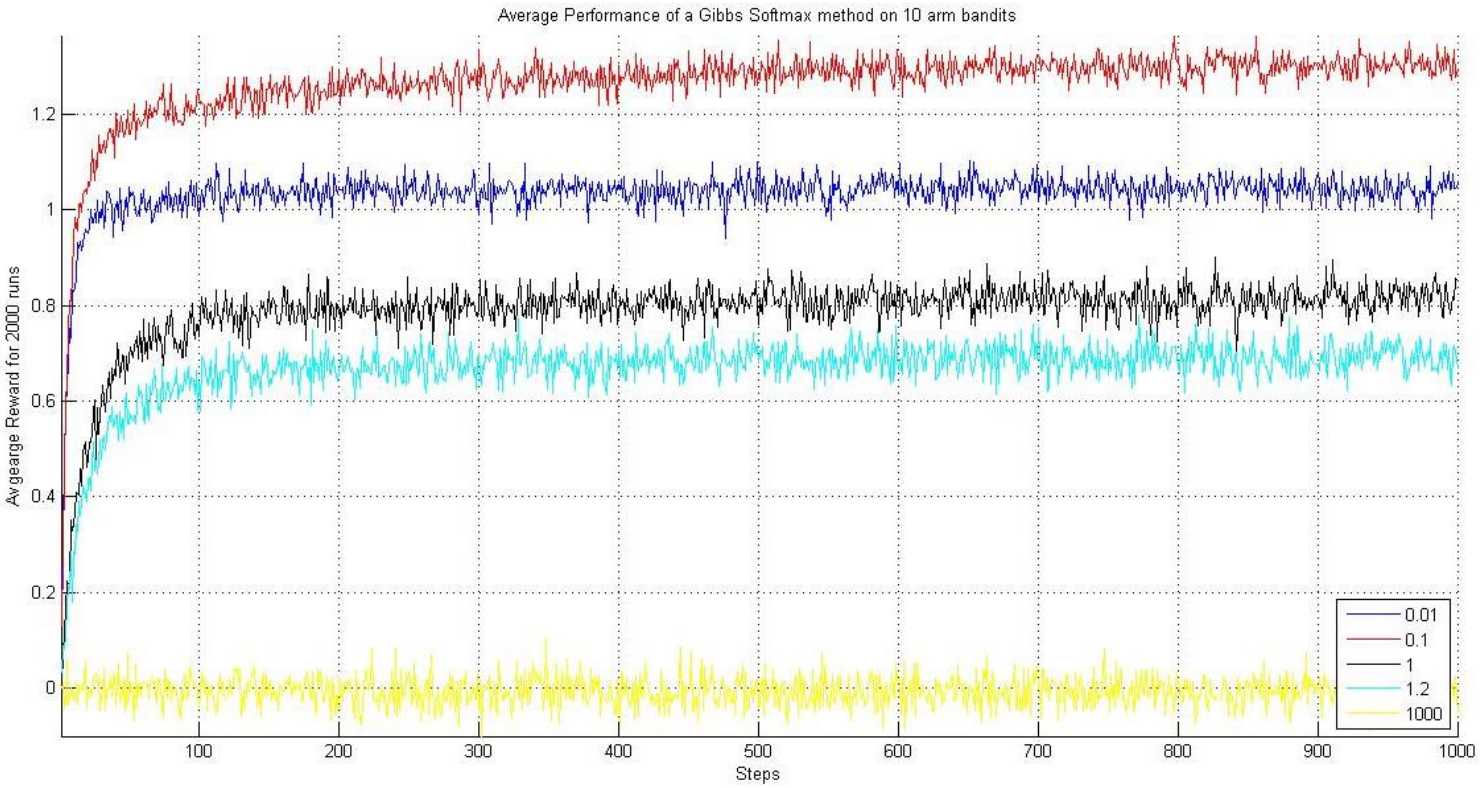
- As they mentioned, epsilon with $\{0.01\}$ is out performing epsilon with $\{0.1\}$ in the longer runs.

Question-2:

- To reproduce the results achieved in this section, run the Q2.m
- Experiments for Softmax action selection were conducted using Gibbs probability distribution. Different temperatures $\{0.01,0.1,1,1000\}$ were used in the experiment which lasts for 1000 steps and 2000 runs.
- Choice of temperatures: $\{0.01\}$ was chosen based on the fact that as temperature tends to {0} it becomes greedy approach. Similarly, high temperature {1000} is selected to verify the claim in the book "all actions become equiprobable leading to exploratory moves at high temperature". Other temperatures were selected to produce results similar to e-greedy.
- Results and observations are as follows.



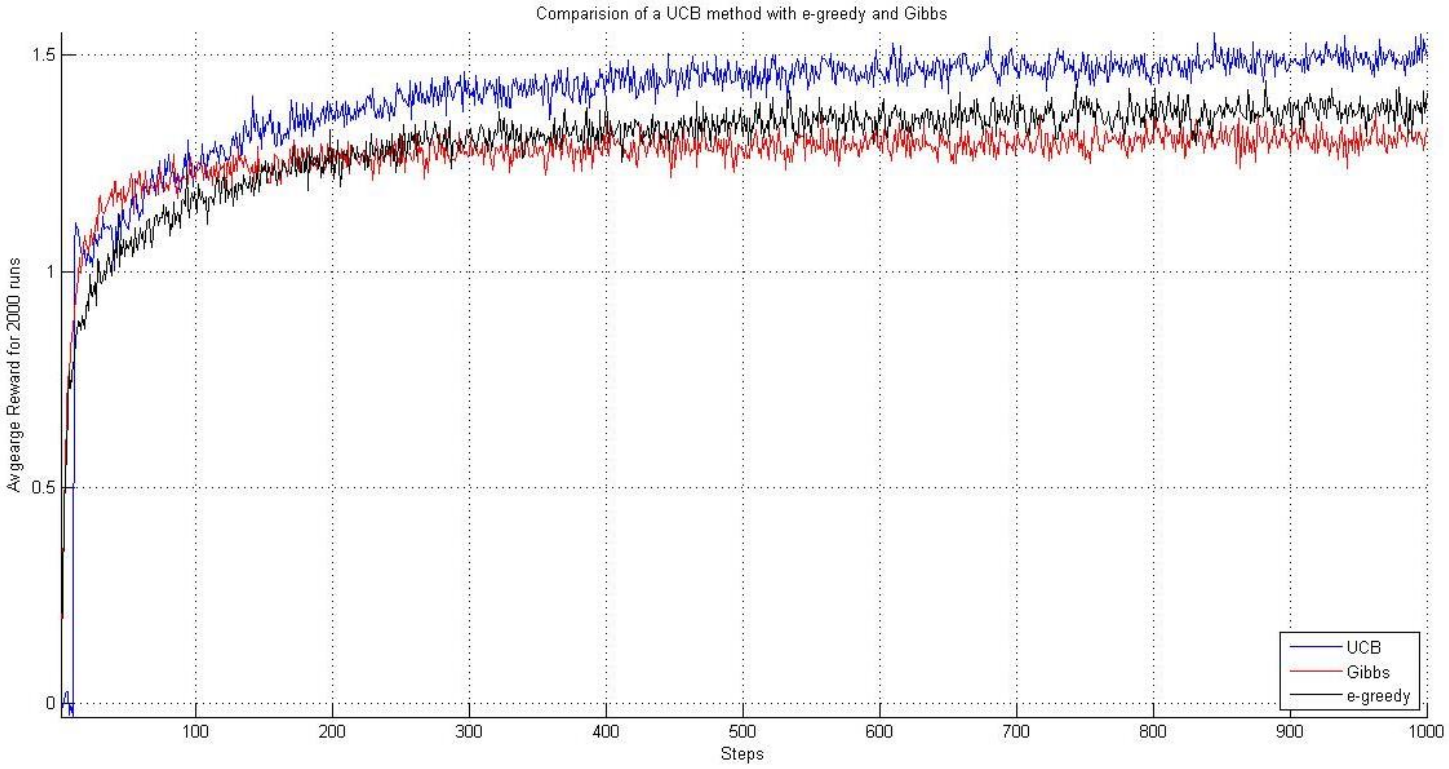
- We can see that for $\{0.1\}$ average reward received is maximum. And it gradually decreases as temperature increases.



- In this graph high value of temperature is selected to evaluate the claim. Clearly, it is behaving like an exploratory agent as observed in e-greedy method for epsilon $\{1\}$
- From both the graphs we can realize that Gibbs method doesn't fare well with 10 arm test bed compared to e-greedy whose maximum reward per step reached till 1.5 as compared to Gibbs whose value reached only till 1.25.
- May be more tuning of temperature is required till its performance is equivalent to or greater than e-greedy.

Question-3:

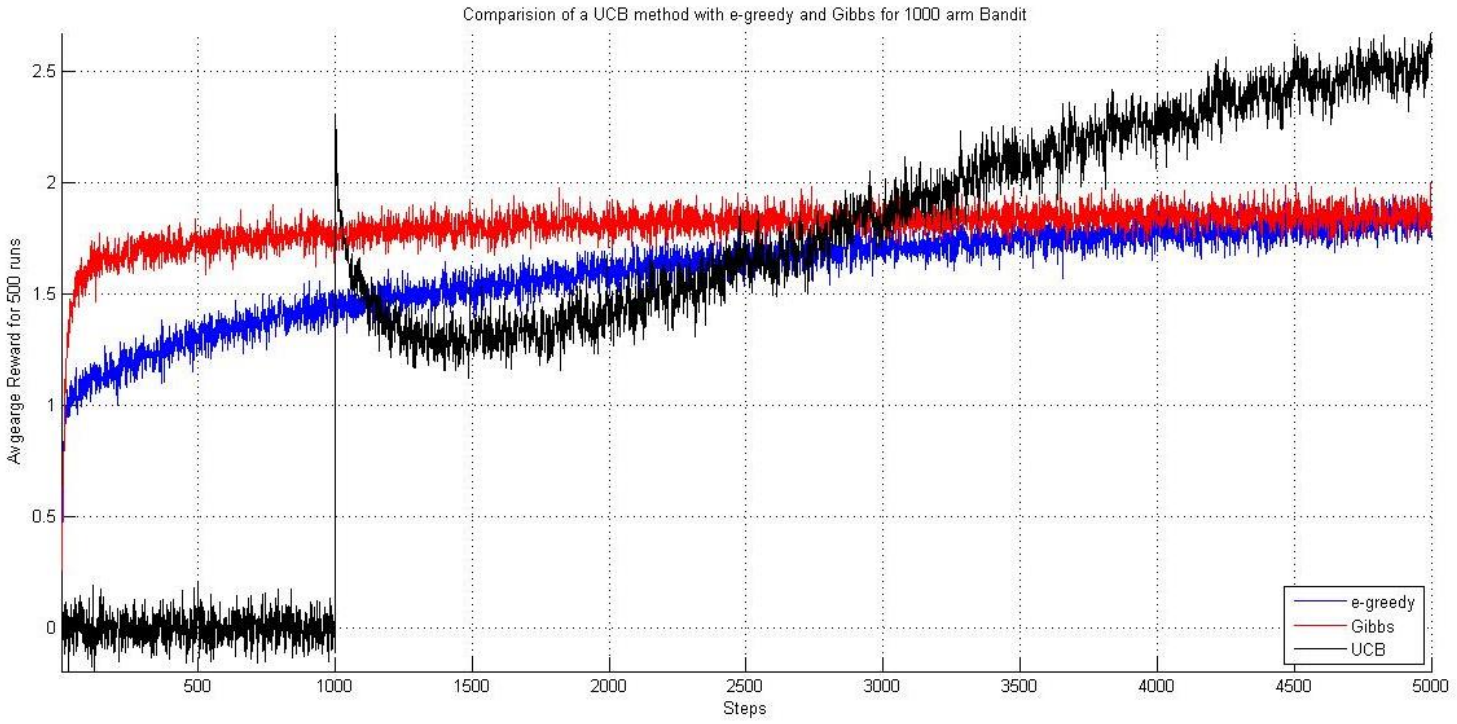
- To reproduce the results achieved in this section, run the Q3.m
- Implemented the UCB1 action selection for $c = 2$. E-greedy is implemented for epsilon = 0.1 and Gibbs is also implemented at temperature 0.1.
- Choice of values: epsilon is selected to be {0.1} since it receives maximum reward at this value. Similar is the case for Gibbs temperature.
- Observations are as follows.



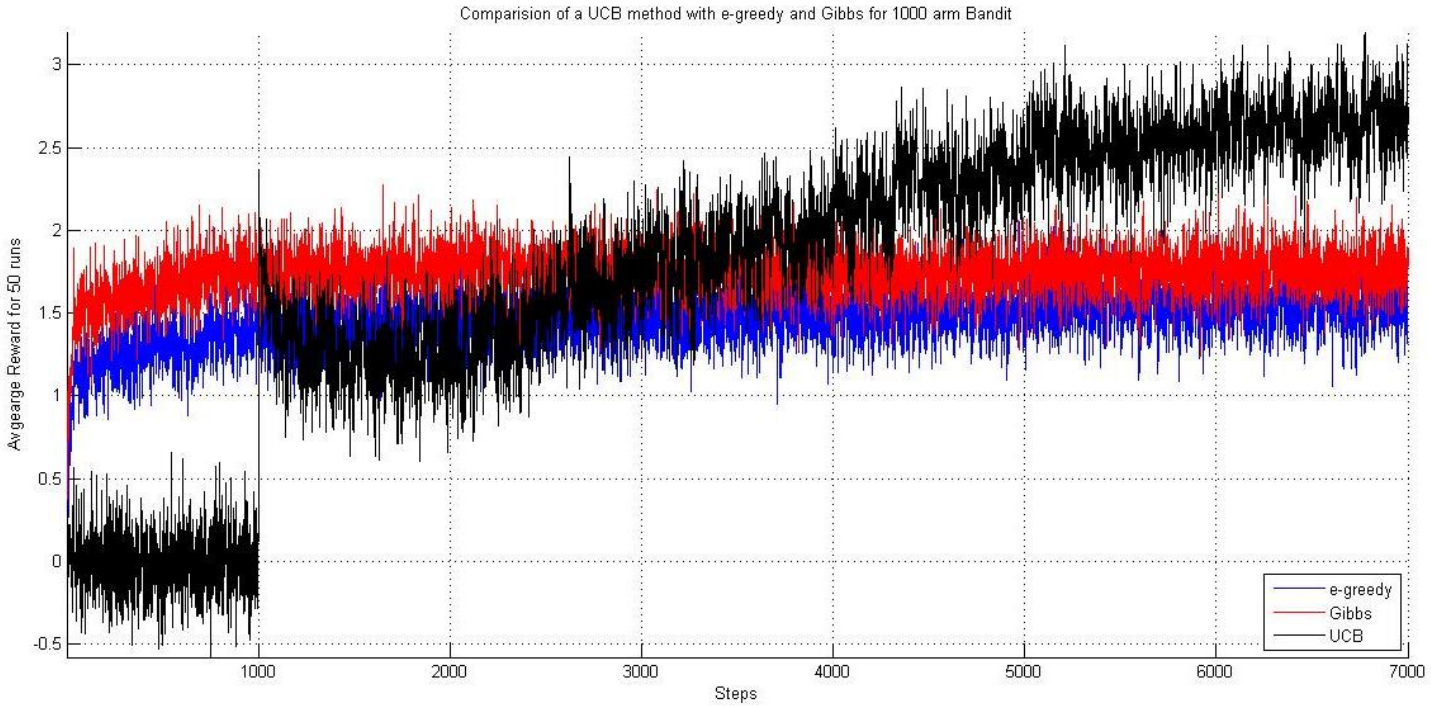
- Clearly, at their best performance UCB, which is a deterministic method, out performs both e-greedy and Gibbs Softmax selection. This can be attributed due to the fact that, in UCB1 we select action based on its confidence rather than selecting randomly. In a way we formulated the UCB solution by modelling and understanding the underlying reward distribution and selecting action based on that.
- Another interesting thing to observe is Gibbs Softmax out performs both e-greedy and UCB at the initial stages.
- This is because, while UCB is still exploring all the arms, Softmax and e-greedy gained some momentum with *already* explored states. Though they may not be optimal, the initial reward achieved will be greater compared to UCB's average reward.
- Similarly we can explain why Softmax avg reward is greater than e-greedy. While, e-greedy method selects actions randomly, Softmax action selection selects actions based on their "weights" which is nothing but the Gibbs distribution in this case. Since initial action selections have higher weight than others Softmax action selection outperformed e- greedy (which is exploring randomly) .

Question-4:

- To reproduce the results achieved in this section, run the Q4.m. In this set of experiments Matlab parallel computing toolbox is used. Or more specifically, function *parfor* is used. In this problem, experiment is run for 500 runs each with 5000 steps.
- 500 runs are considered primarily because it is time consuming and for getting an idea of its convergence and average reward etc., 500 runs seems reasonable. Except we need to live with the fact that there will more aberrations. And 5000 steps is chosen empirically, that is, after a set of experiments UCB started to converge, hence this value is chosen.
- UCB1 action selection is implemented for $c = 2$. E-greedy is implemented for epsilon = 0.01 and Gibbs is implemented at temperature 0.1.
- Choice of values: Since, we have 1000 arms if epsilon value is more it will take more time to converge and we can't even be sure if the chosen epsilon is optimal. This is due to fact that, more exploration will also lead to poor average reward. Therefore, choosing small epsilon is optimal. In other words, exploiting is better than exploring for it to converge fast. We also realized that for epsilon $\{0.01\}$ 10 armed bandit gave maximum reward in the long run. Therefore for 1000 arm bandit, experiments are conducted with epsilon $\{0.01\}$. Same argument applies to temperature parameter in Gibbs Softmax action Selection. Though in this case, we observed maximal average reward for temperature $\{0.1\}$. Therefore, setting the same value experiment is conducted.



- Average reward per step is increased from 1.5 to 2.75 (from following graph)
- For reasons explained in the previous question, Gibbs dominated both e-greedy and UCB.
- From the following graph, is a result of the experiment when the number of steps per play is increased to 7000 steps and number of runs is decreased to 50. From this we can get an idea of average reward convergence.


Comparision of a UCB method with e-greedy and Gibbs for 1000 arm Bandit
Avgearge Reward for 50 runs
Steps
e-greedy
Gibbs
UCB
3
2.5
2
1.5
1
0.5
0
-0.5
1000
2000
3000
4000
5000
6000
7000