UCB vs Thomson Sampling for k-armed Bandits
[Assignment-1]
(CS 394R: Reinforcement Learning)

Shobhit Chaurasia
Department of Computer Science

September 1, 2016

---

1 Introduction

For non-associative k-armed Bandits problem, we studied multiple action-selection strategies in Chapter 2 of [2] including greedy, $\epsilon-$greedy, and UCB action-selection. Greedy methods do not take into account the variance in current action-value estimates while picking an action. UCB tries to incorporate variance into the picture, but not in a fully principled way that is truly Bayesian. Thompson sampling follows a Bayesian approach to action-selection by assuming a prior over the reward distributions for each arm, sampling the rewards from the posterior, and picking the arm with the highest estimated reward.

2 Problem Statement

The goal is to study Thompson sampling algorithm, and compare it with the UCB action-selection procedure. In addition to the comparison of the performance of the two algorithms for stationary and non-stationary tasks, the study also includes comparison of different priors for reward distribution in Thompson sampling, and evaluates the effect of non-stationarity on different priors.

3 Experiments

Thompson sampling algorithm proposed in [1] for the general stochastic bandits case is used. Here, the numerical rewards are assumed to be in the interval $[0, 1]$.

3.1 Setup

The experimental setup includes 100 $10-$armed bandits. Each bandit is played with for 2000 steps. The reward distribution for each arm follows a Logit-Normal distribution. Logit-Normal is essentially a Normal distribution shrunk to the $[0, 1]$ domain. If a random variable X is normally distributed, then $Y = f(X)$ has Logit-Normal distribution, where f is the logistic function,

$$f(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

Fig. 1 shows the reward distribution for the 10 arms. Note that the support is indeed $[0, 1]$ despite the curves in the figure showing a larger support. This is because the density was plotted from a collection of draws from the distribution using a kernel density estimate.

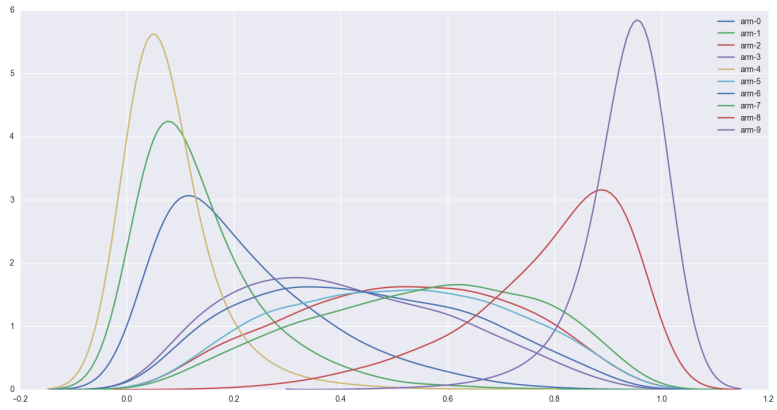


Figure 1: Reward distribution for the 10 arms in the stationary case

In all the experiments, the average reward at each time step was computed across all the hundred bandits to compare the performance of different instances of UCB/Thompson sampling. The prior for reward distributions in Thompson sampling is set to a Beta distribution with different parameters for different experiments. This is because the algorithm used is an adaptation of the Bernoulli Thompson sampling algorithm [1] (in which rewards can only be 0 or 1) for the general bandits case (with rewards in $[0, 1]$). The fact that Beta distribution is a conjugate prior for Bernoulli simplifies the posterior updates in the algorithm.

3.2 Experiment - 1

This experiment compares the UCB action selection strategy with Thompson sampling using a uniform prior (Beta(1, 1)) for reward distributions in a stationary task.

3.2.1 Hypothesis

I expected the two algorithms to eventually converge to similar average reward, but did not have a clear intuition of which one would converge faster. I expected the Thompson sampling to be more stable than UCB after it has converged to near-optimal action-selection strategy, because UCB encourages continual exploration.

3.2.2 Observations and Explanations

Fig. 2 shows the change in average reward with time step for the two algorithms. Thompson sampling quickly infers the optimal arm at each step by sampling from its posterior estimate which increasingly approximates the true reward distribution. Average reward in UCB continuously increases as its action-value estimates become better. However, the increase is much slower than Thompson sampling because the variance term in UCB encourages exploration even after a large number of steps. This is also why the reward curve for UCB is less stable than Thompson sampling.

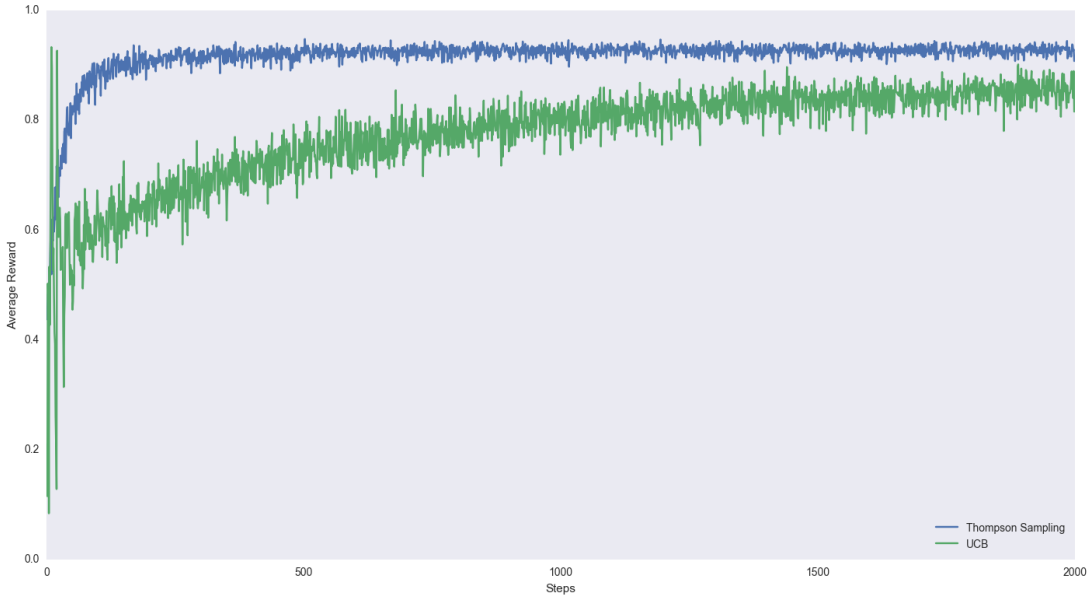


Figure 2: Average reward for UCB and Thompson sampling in a stationary task.

3.3 Experiment - 2

This experiment compares the effect of different priors in Thompson sampling. The different priors used are shown in Fig. 3.

3.3.1 Hypothesis

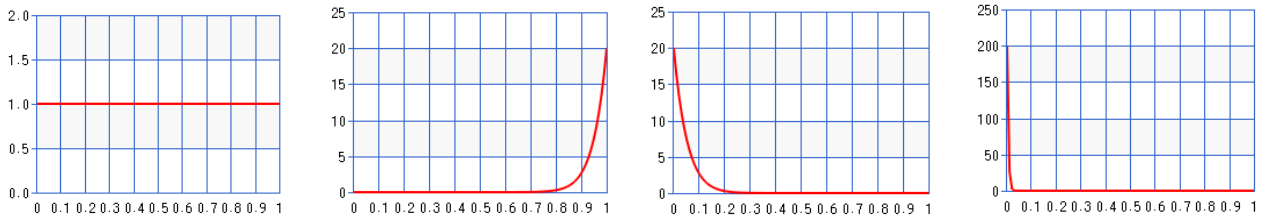
I expected the uniform prior (Beta(1,1)) to work at least as good as the ones which favor high-reward arms because of the general notion of non-informativeness that flat priors like Beta(1,1) encode. Further I expected the priors which favor low-reward arms to have slower convergence but in the long run, I expected them to achieve rewards similar to the other priors.

3.3.2 Observations and Explanations

While Beta(1,1) is the uniform prior, easily adaptable the evidence for the true distribution received through actual samples, the other priors have a high bias to start off with.

Fig. 2 shows the change in average reward with time step for different instances of Thompson sampling with different priors. Additionally, the curve for UCB is also shown alongside to facilitate comparison.

While the average reward for the case with uniform prior (Beta(1,1)) and that with Beta(20,1) prior eventually converges to the same value, the speed of convergence for the latter is slower. A simple argument for this can be made from the way the Thompson sampling algorithm is set up - with same prior distributions to start with, arms which result in low true rewards might end up being frequently chosen in the initial phase. Beta(1,1) prior being flat, observations from the reward distribution for that non-optimal arm can more easily distort it to get the posterior closer to the true distribution. However, since the shape of Beta(20,1) prior is almost anti to the true



(a) Beta(1,1) (b) Beta(20,1) (c) Beta(1,20) (d) Beta(1,200)

Figure 3: Density plots for different priors used in Thompson sampling

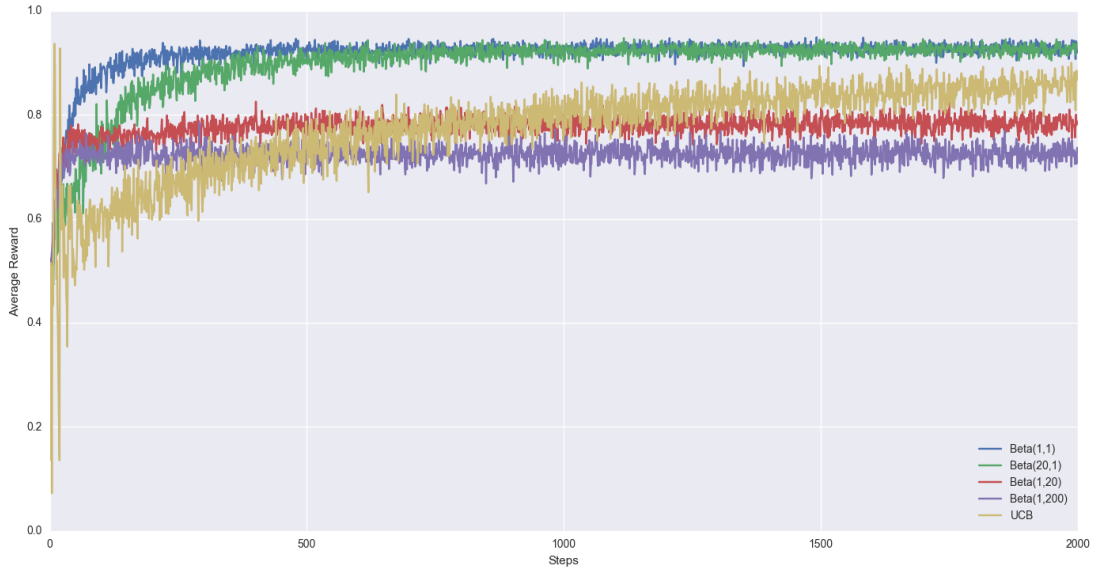


Figure 4: Average reward for Thompson sampling with different priors in a stationary task.

reward distributions for the non-optimal arms (favoring the arms with high reward), it requires far more reward observations to get a posterior which is close to the true reward distribution for the non-optimal arms. The posterior for the true optimal arm would quickly start resembling its true distribution; however, it is the delay in pushing out the posterior for non-optimal arms from the high-reward region that slows down the algorithm's convergence.

As evident from Fig. 5, the posteriors corresponding to arm 8 and arm 9 (which are also the arms with high true reward) quite closely resemble their true distribution, while the posterior for others are not close to their respective true distributions (compare to Fig. 1). This is because once the posteriors for the high-reward arms are separated from the others, those arms will be chosen most of the time, and their posteriors would keep on getting refined by the continual observations from their underlying true distribution, while others are neglected.

For the case with priors Beta(1,20) and Beta(1, 200), the average reward asymptotes to a lower value as compared to the others. These priors favor the arms with low rewards. The posterior for the low-reward arms would quickly start resembling their true distributions. The posterior for other arms will have a hard time getting closer to their true distributions. However, since we sample from the posteriors and pick the arm with the highest estimated reward, whichever

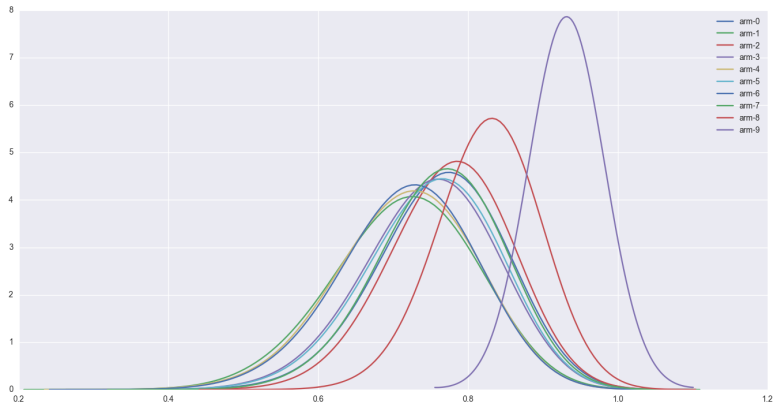


Figure 5: Posterior reward distributions starting with prior Beta(20,1) for one of the bandits.

Arm Index	Frequency
0	0
1	0
2	13
3	5
4	0
5	12
6	5
7	3
8	28
9	34

Table 1: Number of bandit instances (out of 100) in which different arms were continuously chosen as the optimal arm after 500 steps with Beta(1,200) prior.

arm among the ones with medium or high true rewards gets occasionally selected (by chance) in the initial few steps would start to get its posterior out of the region of the prior (which is the low reward zone). Once that happens, that arm will be chosen frequently as the optimal arm because the samples from its posterior yield the highest estimated reward. Hence, in this case, the selection of arms in the long-run depends upon chance. The argument is supported by Table. 1 where non-optimal arms are chosen in many bandit instances.

An important observation from this experiment is that poor choice of priors could lead the Thompson sampling algorithm to get stuck in a non-optimal action selection strategy.

3.4 Experiment - 3

This experiment compares the performance of UCB and Thompson sampling in a non-stationary task. The true reward distributions for all the arms were changed every 500 steps (by changing the parameters of the Logit-Normal distributions).

3.4.1 Hypothesis

Since Thompson sampling is based on posterior distribution of estimated rewards, I expected its adaptability to be slow since it would take the well-established posteriors many observations from the new reward distributions to re-adapt themselves. This would be especially true when the updated reward distributions are highly different from the inferred posteriors. On the other hand, since UCB almost continually encourages exploration, I expected it to adapt faster. However, I did not have an intuition about how fast this adaptation will be in UCB, and what factors could speed up or slow the adaptation down.

3.4.2 Observations and Explanations

Fig. 6 shows the change in average reward with time step for UCB and Thompson sampling, both in stationary and non-stationary setting. As expected, Thompson sampling takes a long time to adapt itself to the updated reward distributions. Surprisingly, UCB adapts itself almost instantaneously.

A grave issue with Thompson sampling in non-stationary setting is its heavy dependence on posteriors (which ironically is also its strength). In abrupt changes to reward distributions, such as those in this experiment, it could happen that Thompson sampling fails to recover completely if the updated true distribution for the optimal arm is far from the current posterior with high estimated reward. In such a case, Thompson sampling would continue to sample from the stale posterior oblivious of the new reward distributions. This is evident in Fig. 7 for instances with Beta(1,20) and Beta(1,200) prior.

4 Conclusion

This study compared UCB and Thompson sampling action-selection strategies in different settings. The main takeaway from this study is that while Thompson sampling is richer and more principled in its approach than UCB, it could perform worse with poor choice of priors, especially in a non-stationary setting.

References

[1] Shipra Agrawal and Navin Goyal. "Analysis of Thompson Sampling for the Multi-armed Bandit Problem." In:

[2] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. Vol. 1. 1. MIT press Cambridge, 1998.

Figure 6: Average reward for UCB and Thompson sampling in a stationary task.

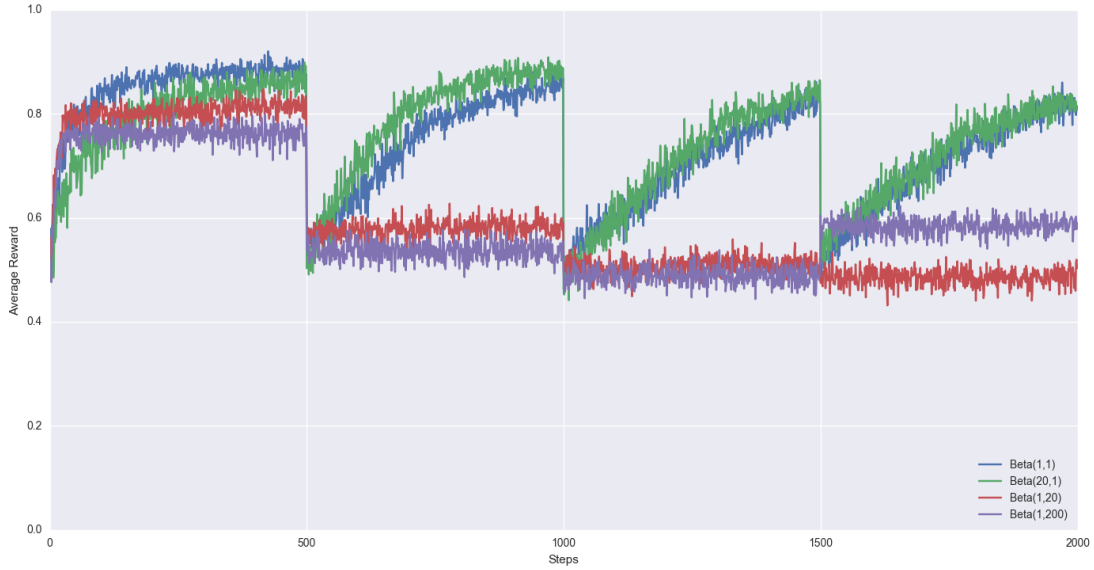


Figure 7: Average reward for Thompson sampling with different priors in a non-stationary task.