In this section, we use a simple example to show how Thompson Sampling (TS) works.

Example. Beta-Bernoulli Bandit

The scenario is again the Multi-arm Bandit machine.

The machine has K arms. An action $k \in \{1, 2, 3, \cdots, K\}$ means pulling the k^{th} arm. Arm k produces a reward of one with probability θ_k and a reward of zero with probability $1-\theta_k$. The player knows that Bernoulli is a good model, but $\theta_1, \theta_2 \cdots, \theta_K$ are unknown but fixed (unchanged)

In round i, when action $x_i \in [K]$ is applied, a reward $r_i \in \{0, 1\}$ is generated with $P(r_i = 1 \mid x_i, \theta) = \theta_{x_i}$.

After observing r_i, the player updates its estimation of θ.

Recall that, the conjugate prior of Bernoulli distribution is Beta-distribution. So, we use K Beta distributions to model $P(\theta_i)$.

$\theta_i \sim \text{Beta}(\alpha_i, \beta_i)$ which has pdf.

$$p(\theta_i) = \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)} \theta_i^{\alpha_i - 1} (1-\theta_i)^{\beta_i - 1}$$

If action $x_t \in [K]$ is taken, then

$$\alpha_i, \beta_i \longrightarrow \alpha_i, \beta_i \quad \text{if } x_t \neq i$$

$$\longrightarrow \alpha_i + r_t, \beta_i + (1 - r_t) \quad \text{if } x_t = i,$$

Ok, then we present Greedy Algorithm and TS to show how they are different.

Greedy Algorithm:

for round $t = 1, 2, \cdots$ do

Get the MAP estimation of θ by

$\hat{\theta}_i = \alpha_i / (\alpha_i + \beta_i)$

Then the action is chosen by taking

$x_t = \underset{i}{\operatorname{argmax}}\ \hat{\theta}_i$

Apply x_t, then update (α_i, β_i) accordingly.

TS Algorithm.

for round $t = 1, 2, \cdots$ do

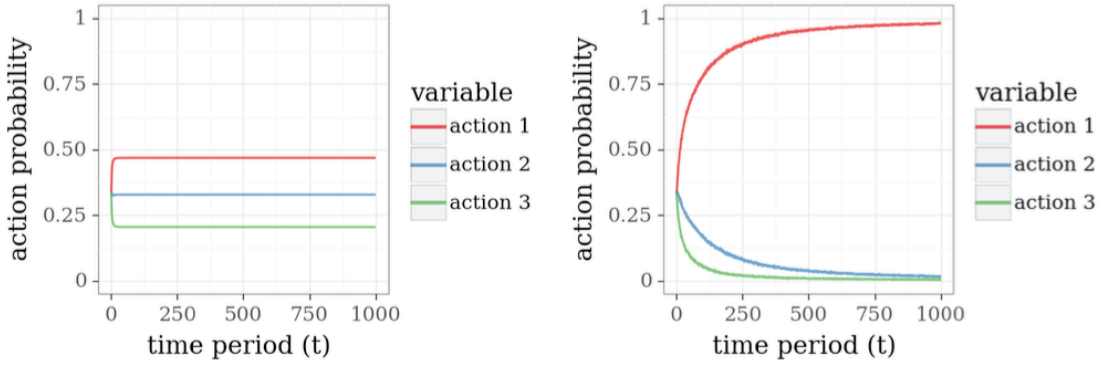
Sample each Beta distribution

Y_i is a random sample from $\text{Beta}(\alpha_i, \beta_i)$

Then the action is chosen by taking

$x_t = \underset{i}{\operatorname{argmax}}\ Y_i$

Apply x_t, then update (α_i, β_i) accordingly.



(a) greedy algorithm (b) Thompson sampling

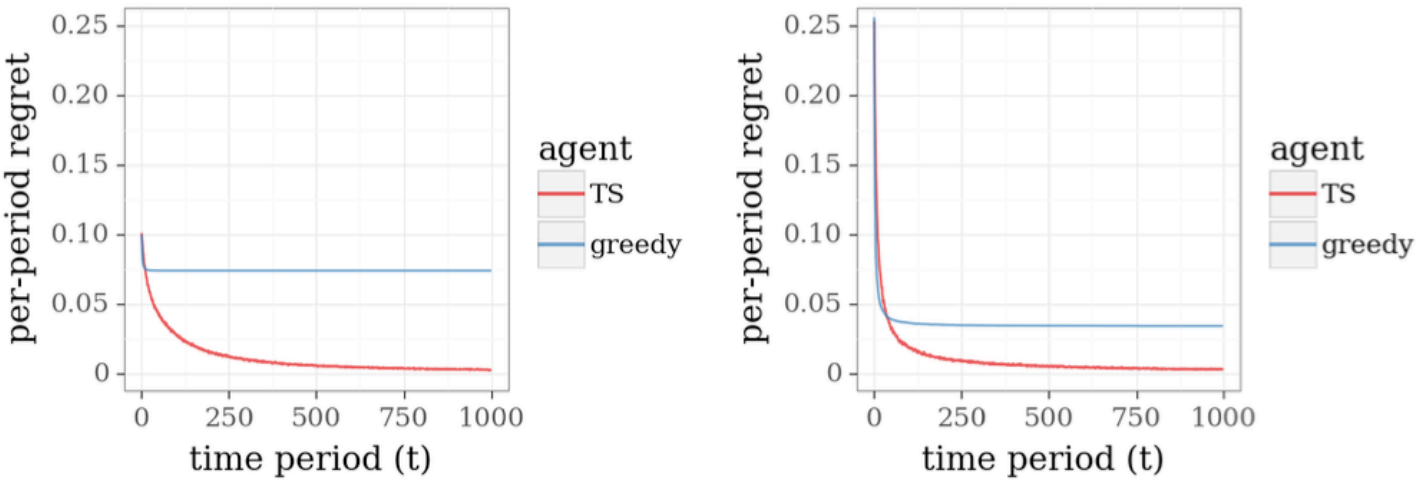
Figure 3.1: Probability that the greedy algorithm and Thompson sampling selects an action.

From the simulation result we can see that Greedy Algorithm may not converge to Action 1, but TS algorithm can.

Regret: To evaluate the performance of algorithm, we use

$$\text{regret} \triangleq \sum_t \left(\max_i r(x_i)\right) - r_t$$

or $\left(\max_i r(x_i)\right) - r_t$ for simultaneous regret.



(a) $\theta = (0.9, 0.8, 0.7)$ **(b)** average over random θ

Figure 3.2: Regret from applying greedy and Thompson sampling algorithms to the three-armed Bernoulli bandit.