Thompson Sampling Algorithm Intuition

The Multi-Armed Bandit Problem

The Multi-Armed Bandit Problem

D1 D2 D3 D4 D5

The Multi-Armed Bandit Problem

- We have d arms. For example, arms are ads that we display to users each time they connect to a web page.
- Each time a user connects to this web page, that makes a round.
- At each round n, we choose one ad to display to the user.
- At each round n, ad i gives reward $r_i(n) \in \{0, 1\}$: $r_i(n) = 1$ if the user clicked on the ad i, 0 if the user didn't.
- Our goal is to maximize the total reward we get over many rounds.

Bayesian Inference

- Ad i gets rewards $\boldsymbol{y}$ from Bernoulli distribution $p(\boldsymbol{y}|\theta_i) \sim \mathcal{B}(\theta_i)$.
- θ_i is unknown but we set its uncertainty by assuming it has a uniform distribution $p(\theta_i) \sim \mathcal{U}([0,1])$, which is the prior distribution.
- Bayes Rule: we approach θ_i by the posterior distribution

$$\underbrace{p(\theta_i|\mathbf{y})}_{\text{posterior distribution}} = \frac{p(\boldsymbol{y}|\theta_i)p(\theta_i)}{\int p(\boldsymbol{y}|\theta_i)p(\theta_i)d\theta_i} \propto \underbrace{p(\boldsymbol{y}|\theta_i)}_{\text{likelihood function}} \times \underbrace{p(\theta_i)}_{\text{prior distribution}}$$

- We get $p(\theta_i|\mathbf{y}) \sim \beta(\text{number of successes} + 1, \text{number of failures} + 1)$
- At each round n we take a random draw $\theta_i(n)$ from this posterior distribution $p(\theta_i|\boldsymbol{y})$, for each ad i.
- At each round n we select the ad i that has the highest $\theta_i(n)$.

Thompson Sampling Algorithm

Step 1. At each round n, we consider two numbers for each ad i:

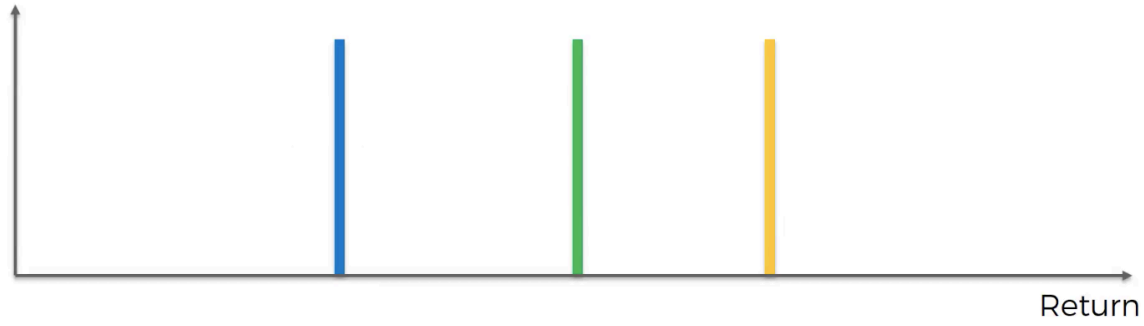
- $N_i^1(n)$ - the number of times the ad i got reward 1 up to round n,
- $N_i^0(n)$ - the number of times the ad i got reward 0 up to round n.

Step 2. For each ad i, we take a random draw from the distribution below:

$$\theta_i(n) = \beta(N_i^1(n) + 1, N_i^0(n) + 1)$$

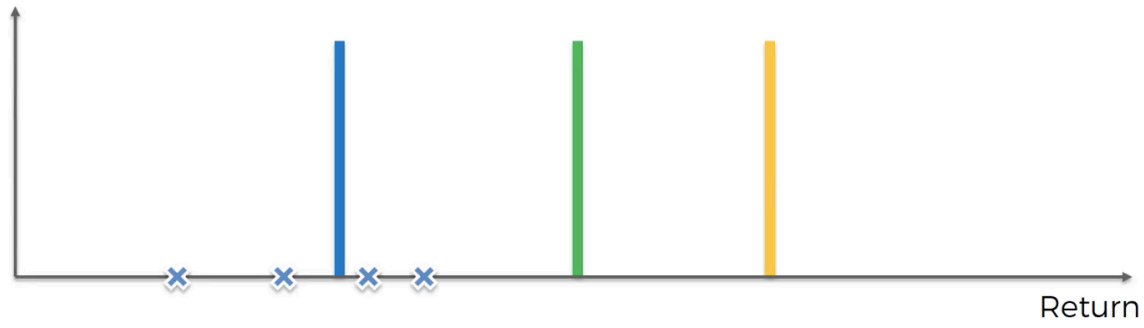
Step 3. We select the ad that has the highest $\theta_i(n)$.

Thompson Sampling Algorithm



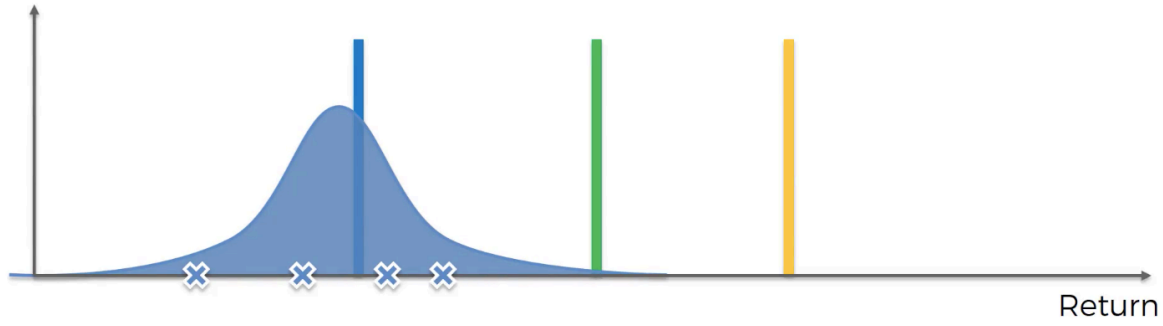
The horizontal axis is expected result from the 3 bandit which each of them represent its expected return. Each bandit has it distribution behind it and the vertical lines are central distribution or actual expected return from that machine. Remember that algorithm itself doesn't know these actual lines so it is hidden. We always bet on the machine that has highest expected return.

Thompson Sampling Algorithm



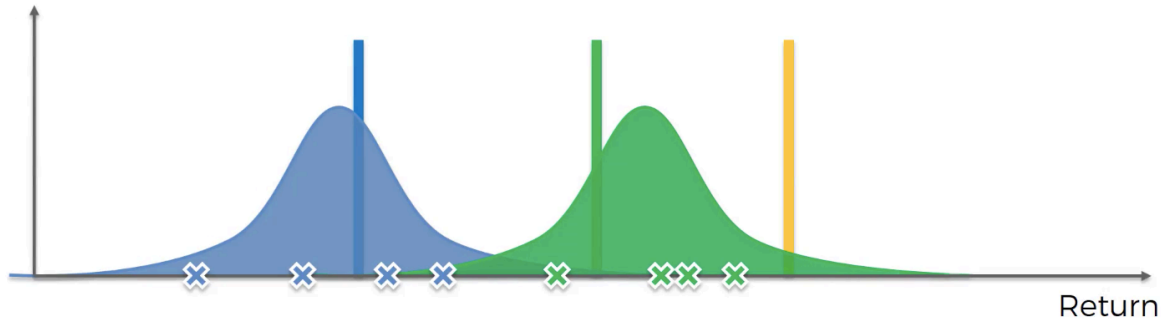
Since we don't have prior knowledge of the current situation then we try to have couple of trail rounds just to get some data to analyse.

Thompson Sampling Algorithm

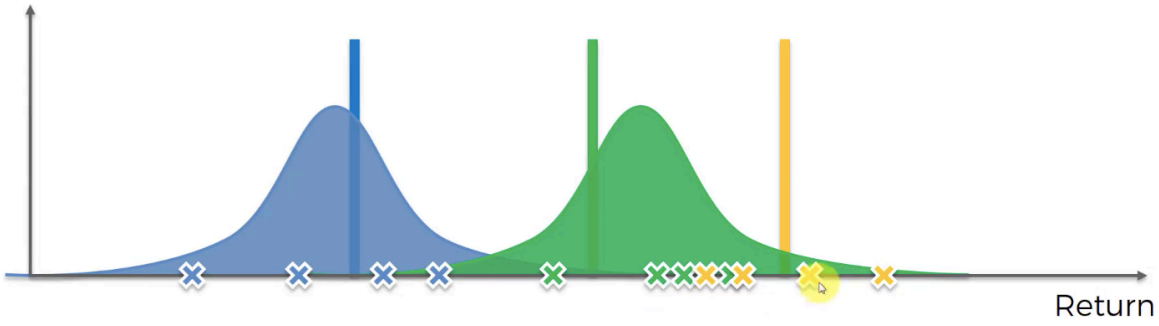


After the trial round, the Thompson algorithms construct a distribution.

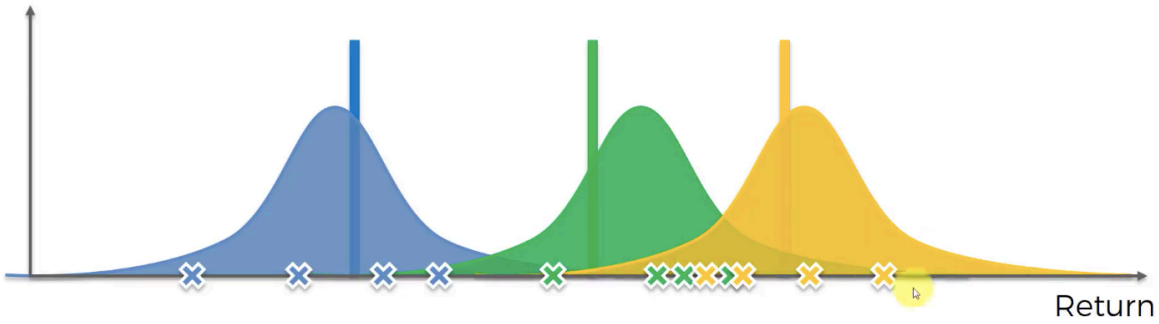
Thompson Sampling Algorithm



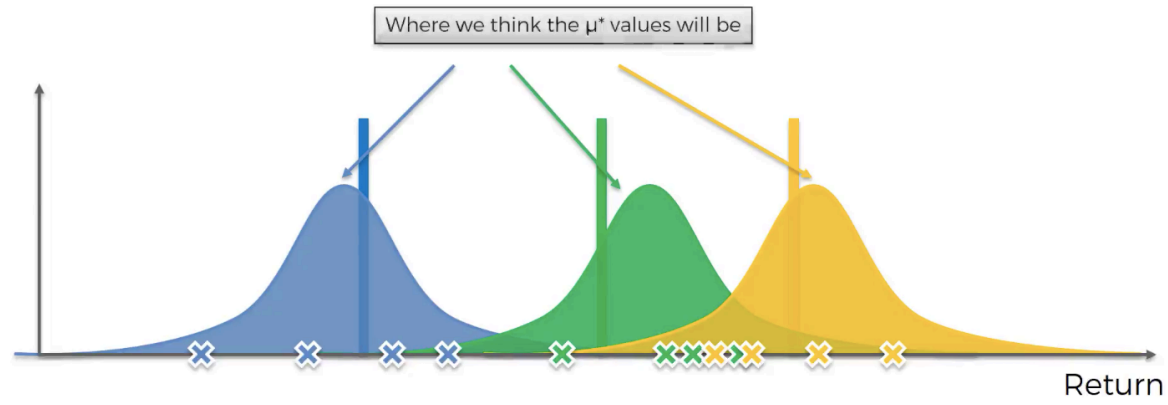
Thompson Sampling Algorithm



Thompson Sampling Algorithm

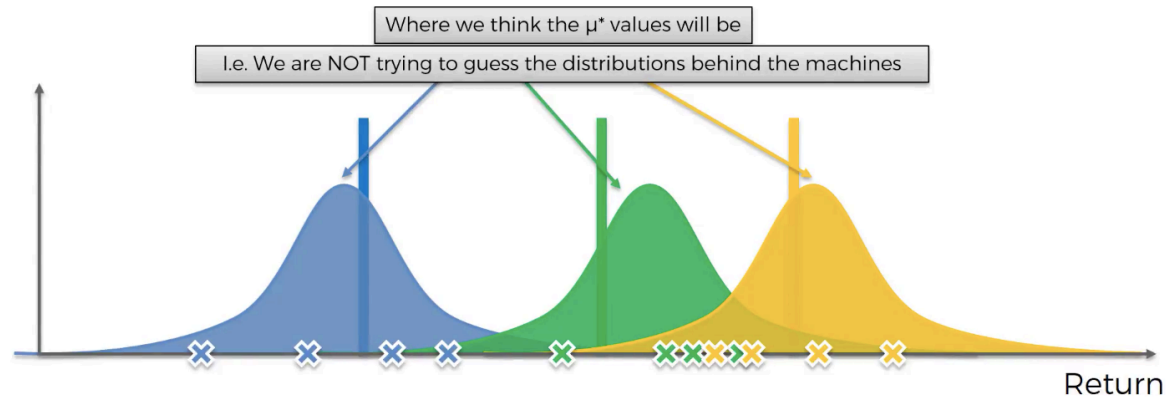


Thompson Sampling Algorithm



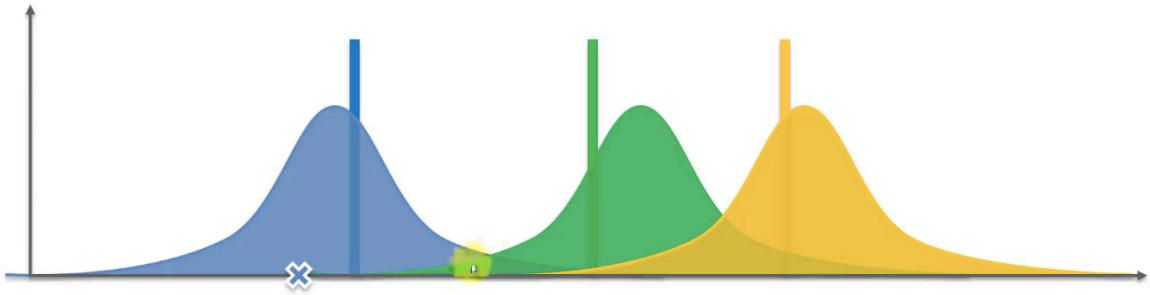
It is important to realize that we are not trying to guess the distribution but we are constructing the distributions of where we think the ***actual expected value*** might lie

Thompson Sampling Algorithm



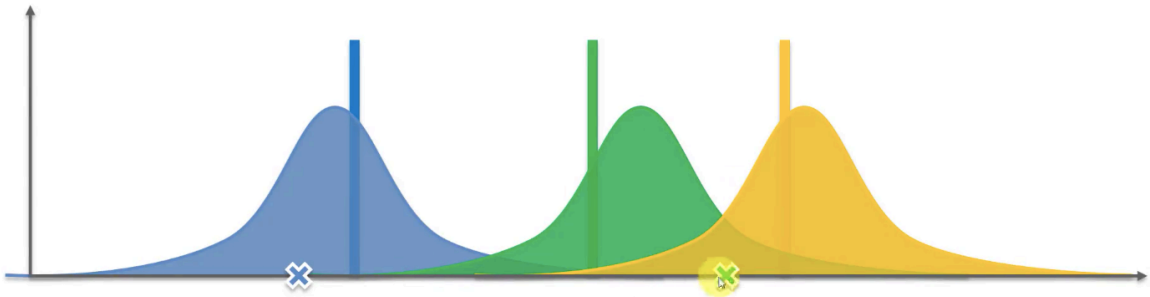
This distribution means that there is high likelihood that our μ* is on the top, but there is also a (less likelihood) chance that our μ* is at the bottom.

Thompson Sampling Algorithm

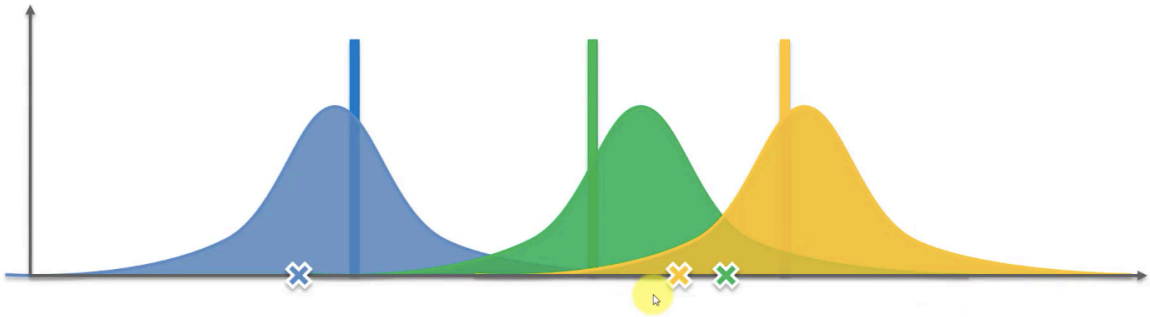


The algorithm goes and pulls out a value from each distribution.

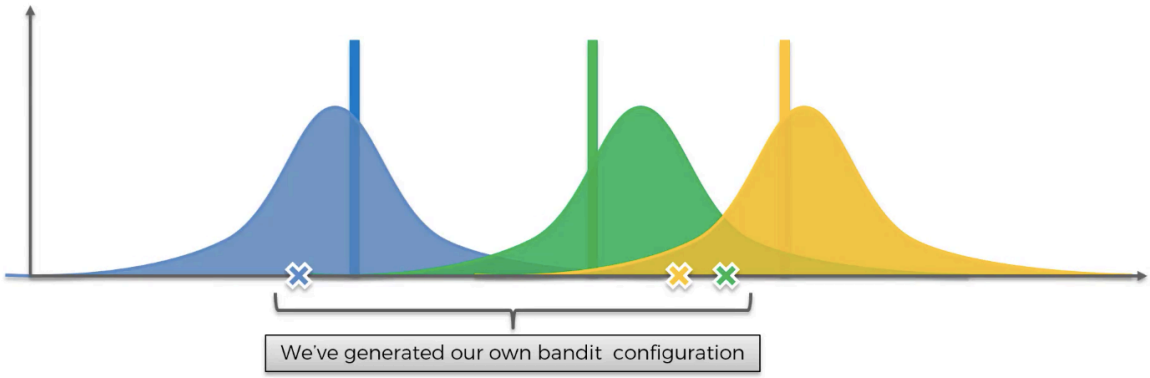
Thompson Sampling Algorithm



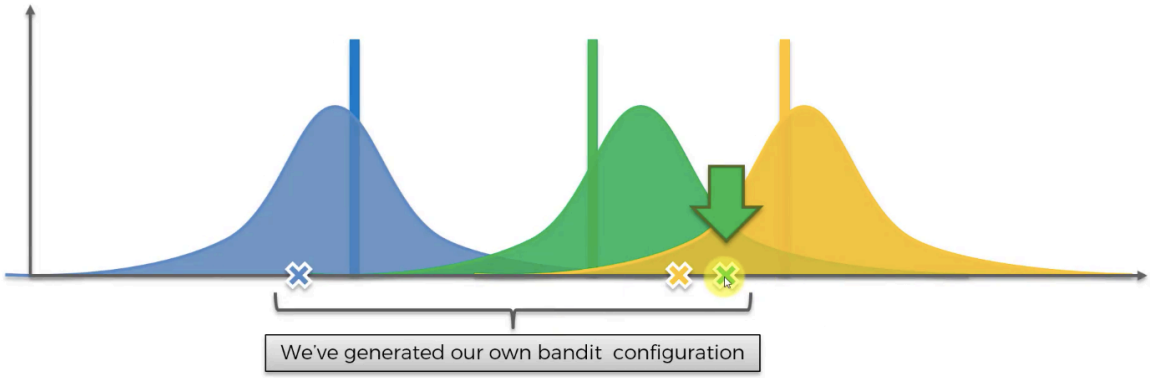
Thompson Sampling Algorithm



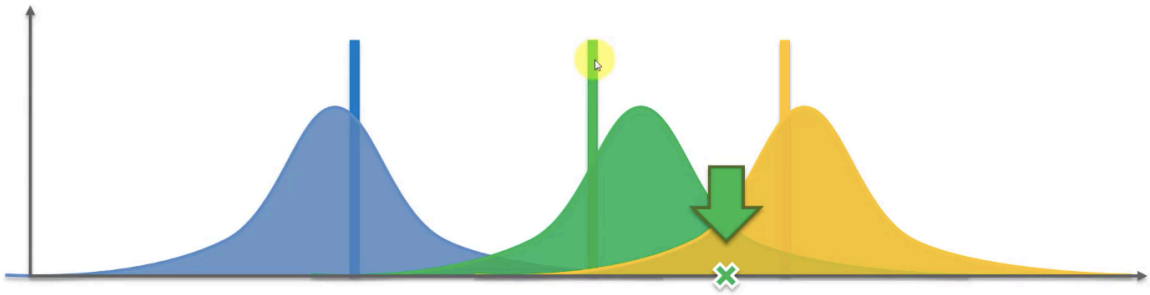
Thompson Sampling Algorithm



Thompson Sampling Algorithm



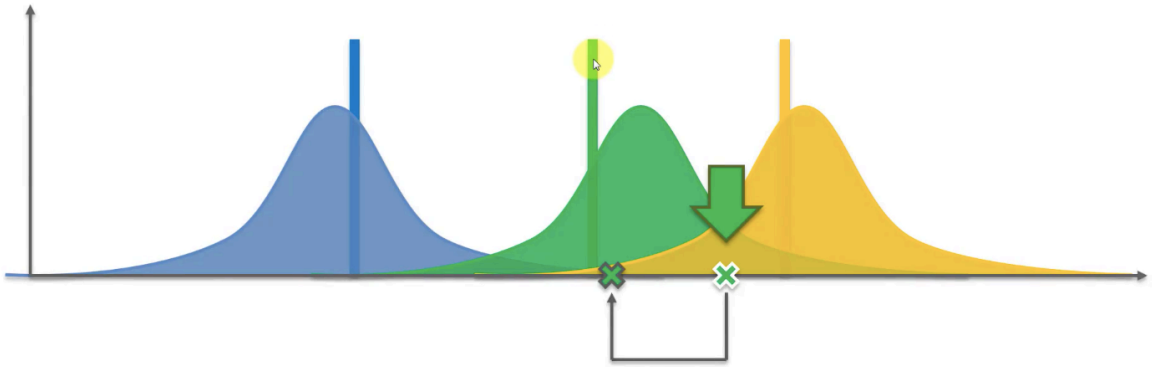
Thompson Sampling Algorithm



Machine Learning A-Z © SuperDataScience

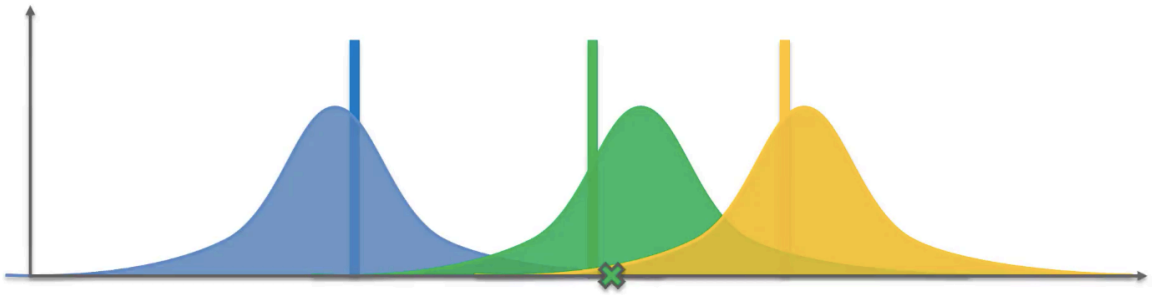
In here we choose this machine because it has the highest expected return out of these 3.

Thompson Sampling Algorithm



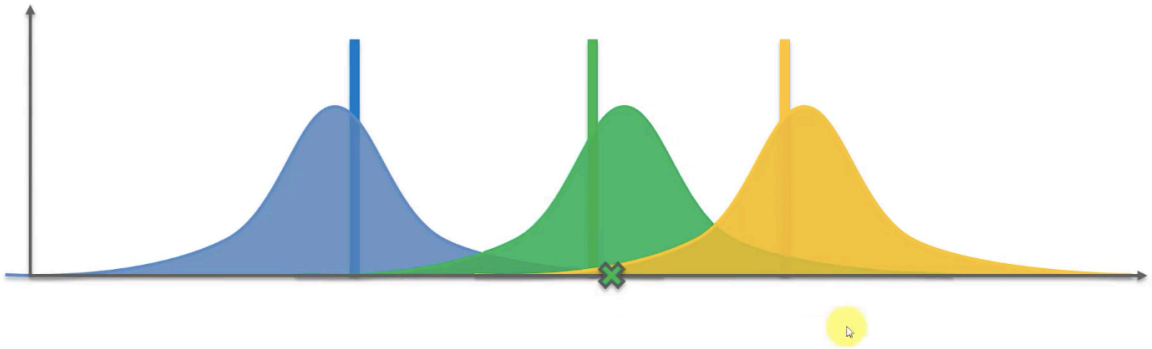
At this stage, it pulls the green lever for this machine and it gives us a value. The value is going to be based on the distribution behind that machine. we change the place of our value.

Thompson Sampling Algorithm



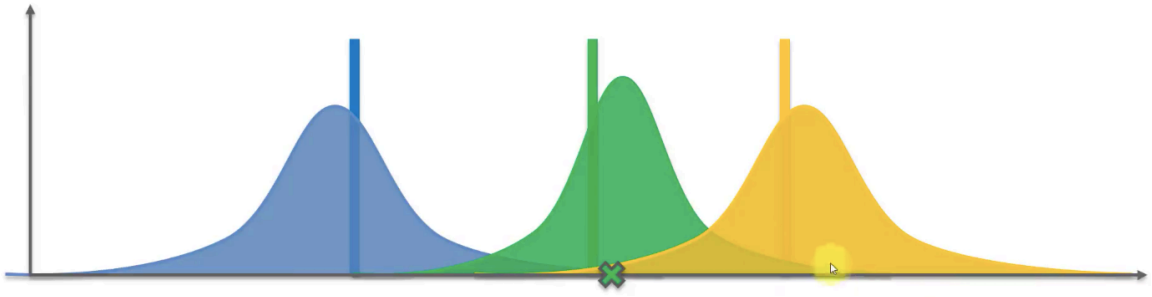
Now we have a new information and we gonna added it in and see how it changes our perception.

Thompson Sampling Algorithm



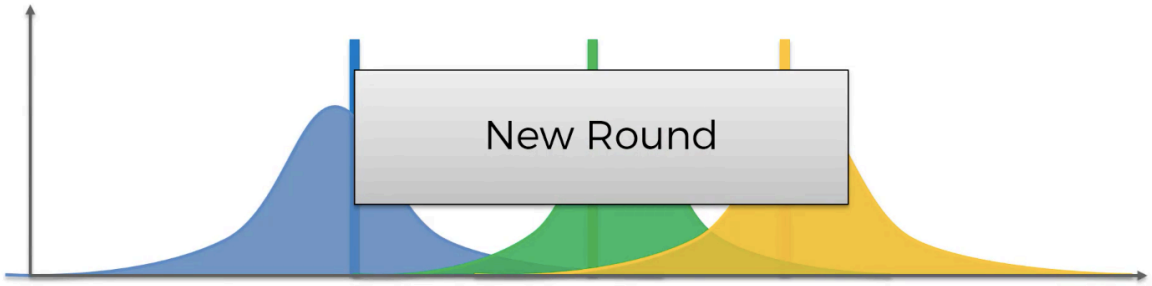
so in here, our perception has been changed and the distribution went a little to the left.

Thompson Sampling Algorithm



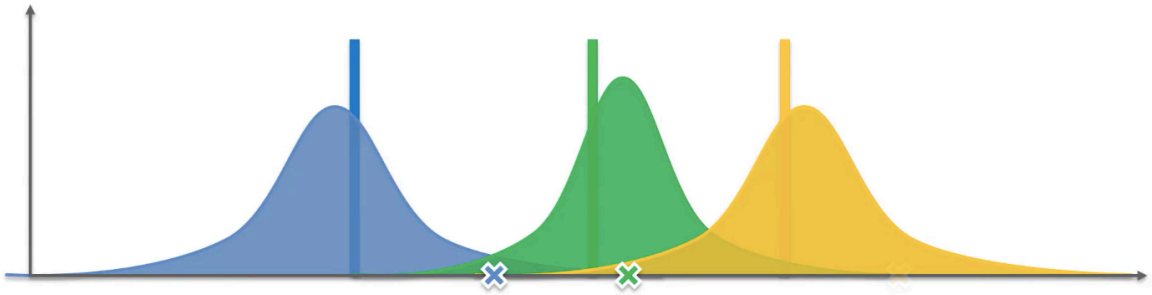
And it became narrower because we have more information and our sample size increases.

Thompson Sampling Algorithm

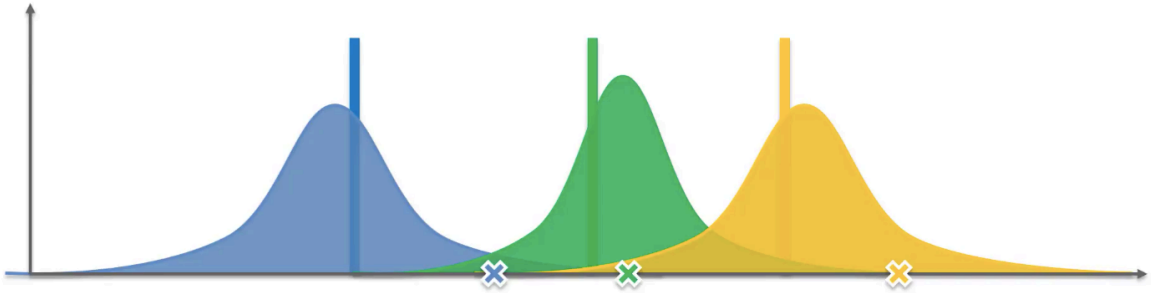


Thompson Sampling Algorithm

Thompson Sampling Algorithm

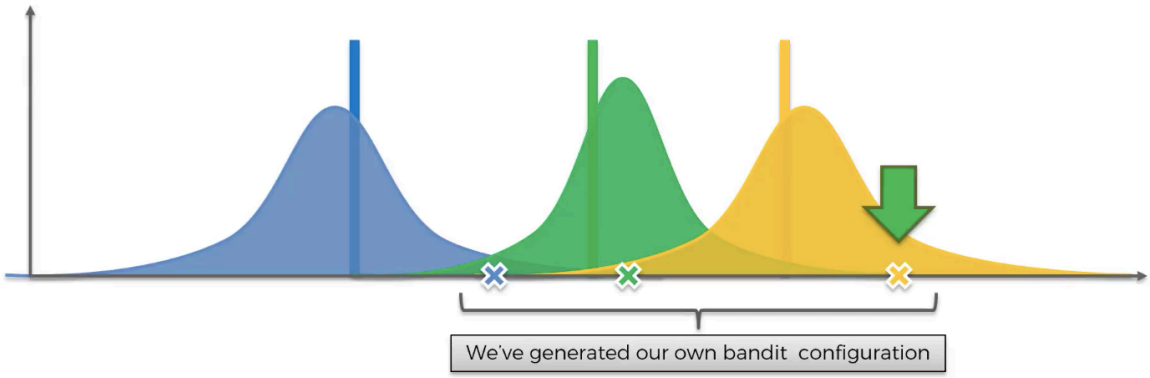


Thompson Sampling Algorithm

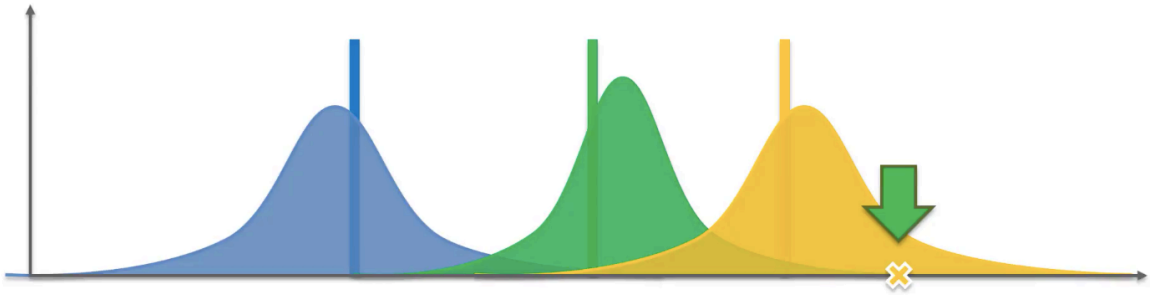


Thompson Sampling Algorithm

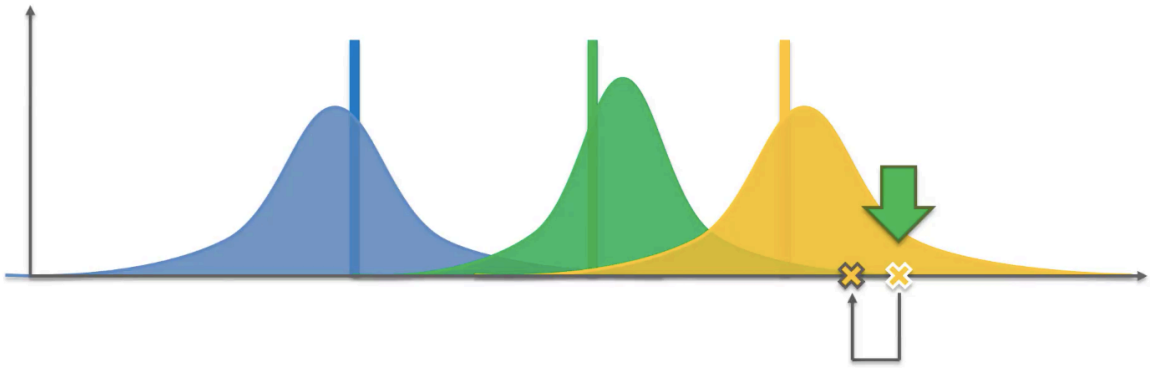
Thompson Sampling Algorithm



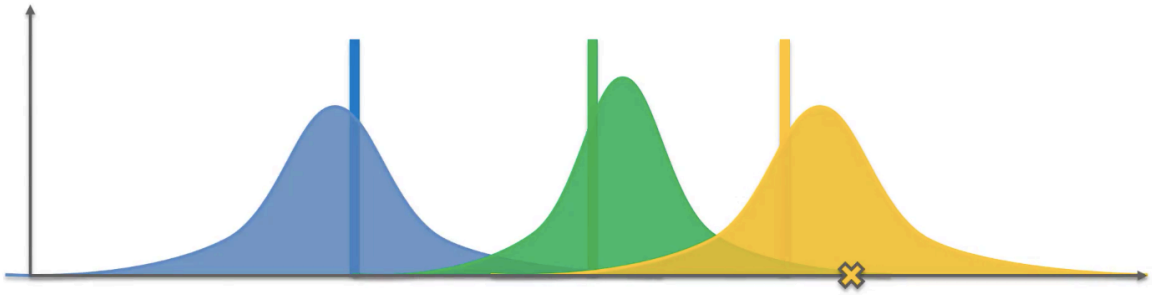
Thompson Sampling Algorithm



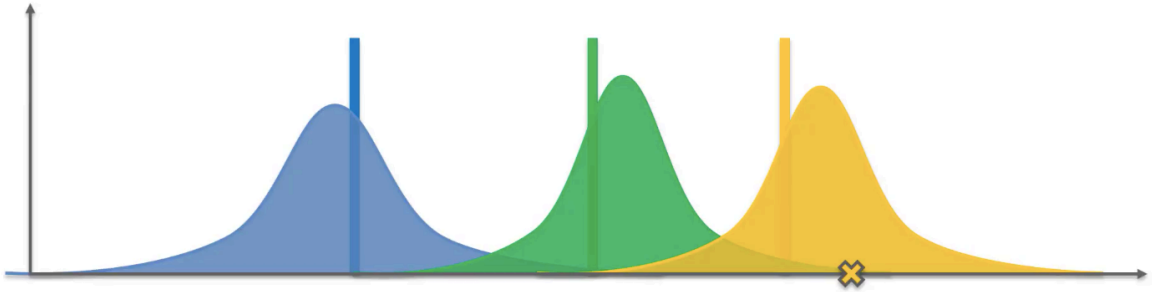
Thompson Sampling Algorithm



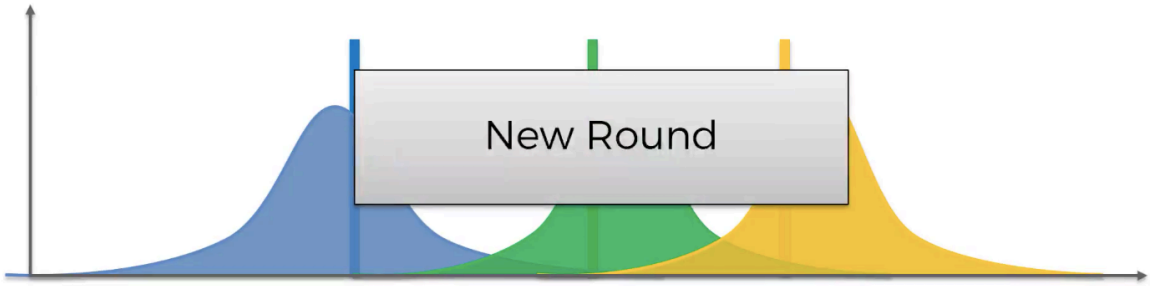
Thompson Sampling Algorithm



Thompson Sampling Algorithm

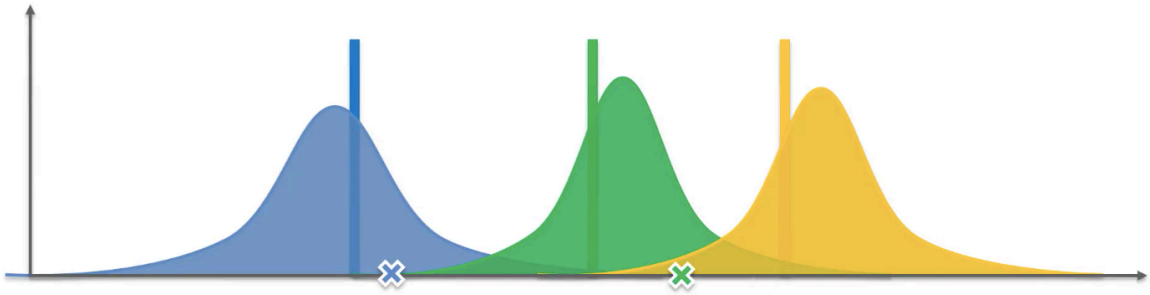


Thompson Sampling Algorithm

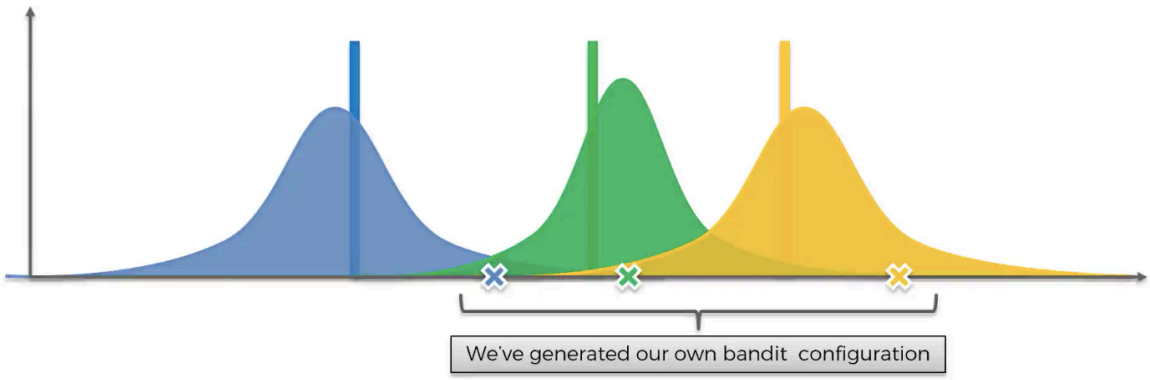


Thompson Sampling Algorithm

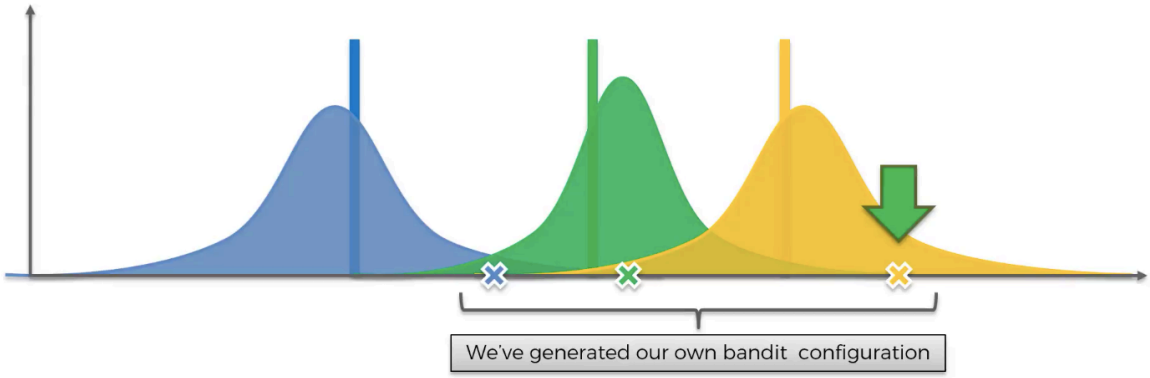
Thompson Sampling Algorithm



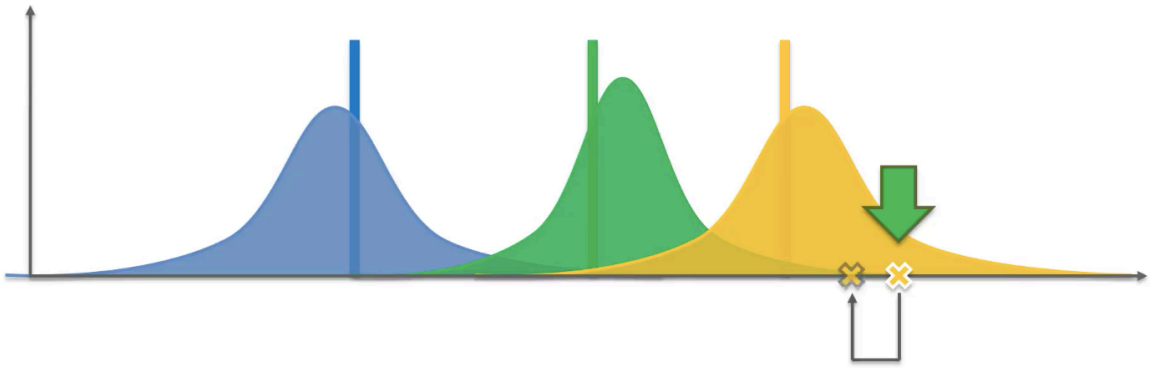
Thompson Sampling Algorithm



Thompson Sampling Algorithm

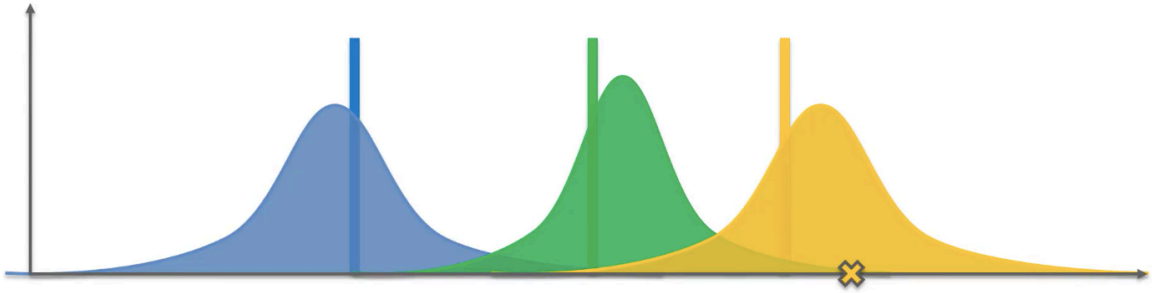


Thompson Sampling Algorithm



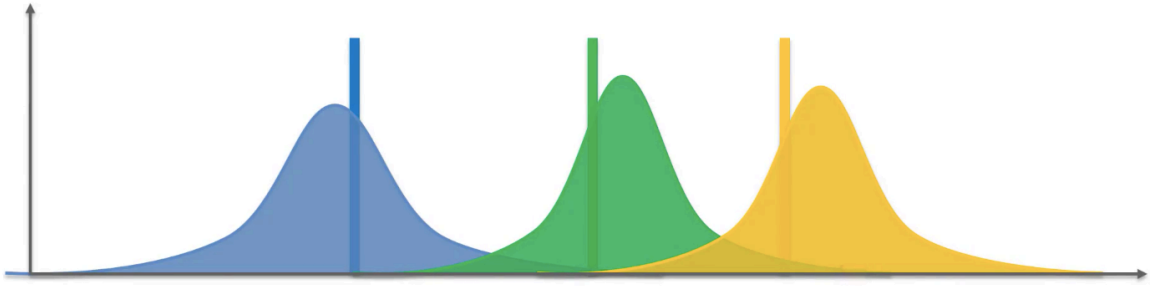
Thompson Sampling Algorithm

Thompson Sampling Algorithm



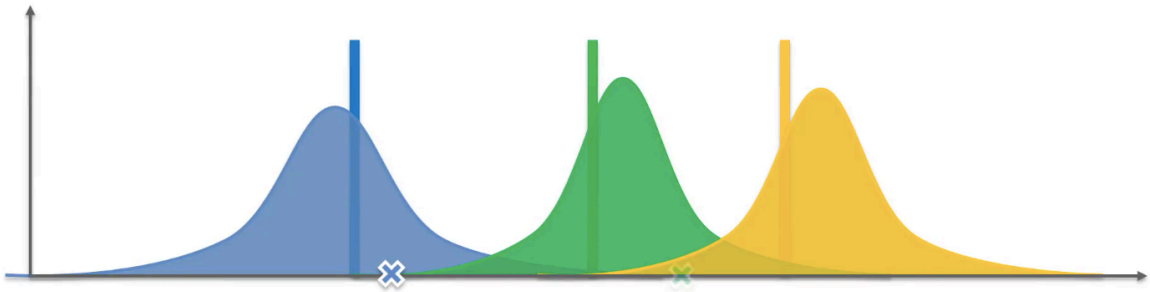
Thompson Sampling Algorithm

Thompson Sampling Algorithm



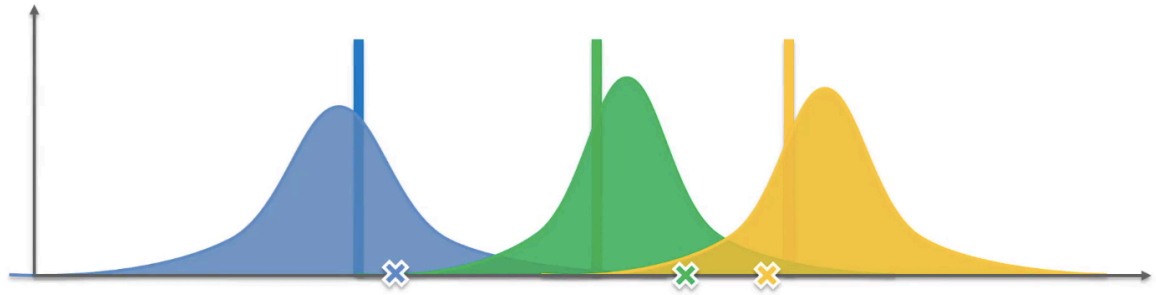
Thompson Sampling Algorithm

Thompson Sampling Algorithm

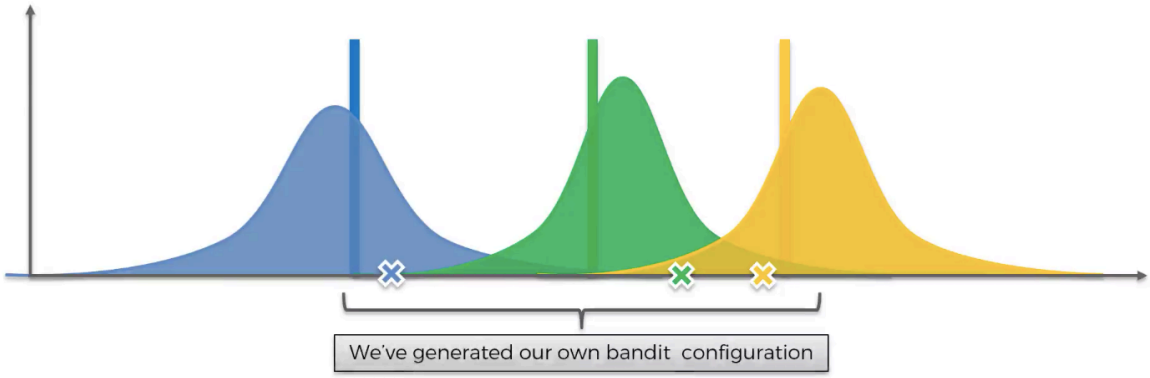


Thompson Sampling Algorithm

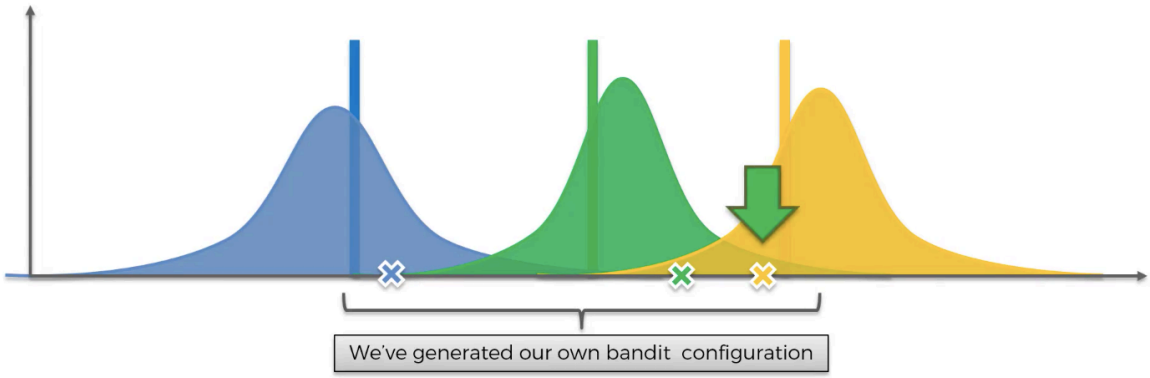
Thompson Sampling Algorithm



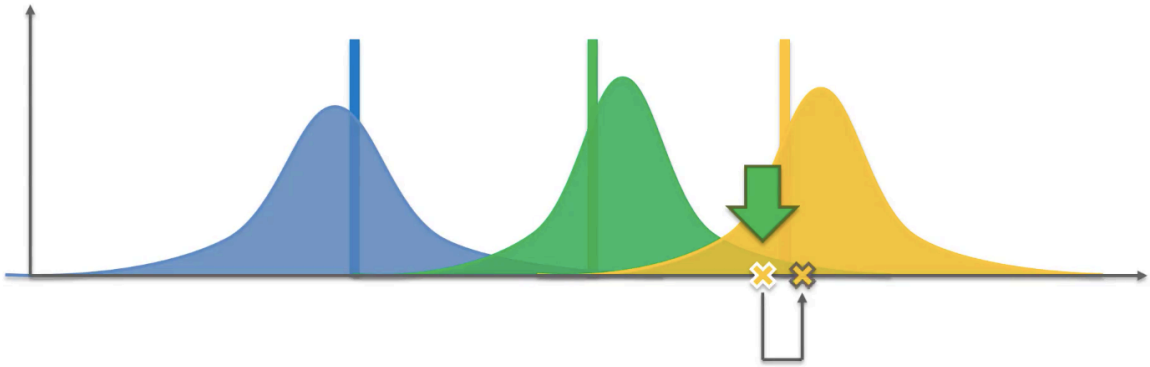
Thompson Sampling Algorithm



Thompson Sampling Algorithm

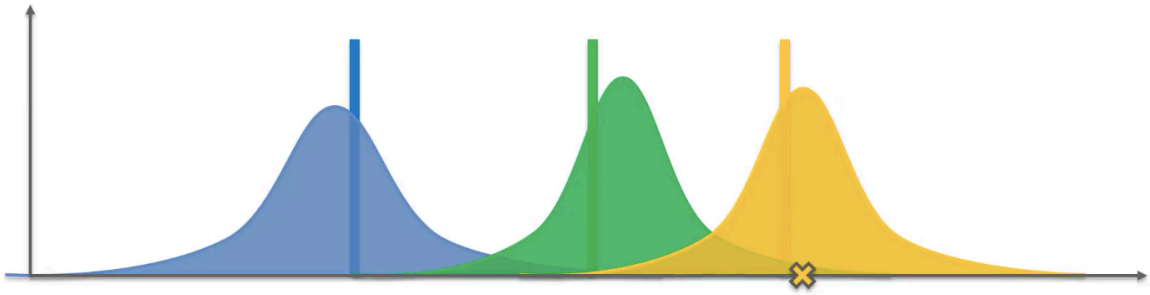


Thompson Sampling Algorithm



Thompson Sampling Algorithm

Thompson Sampling Algorithm

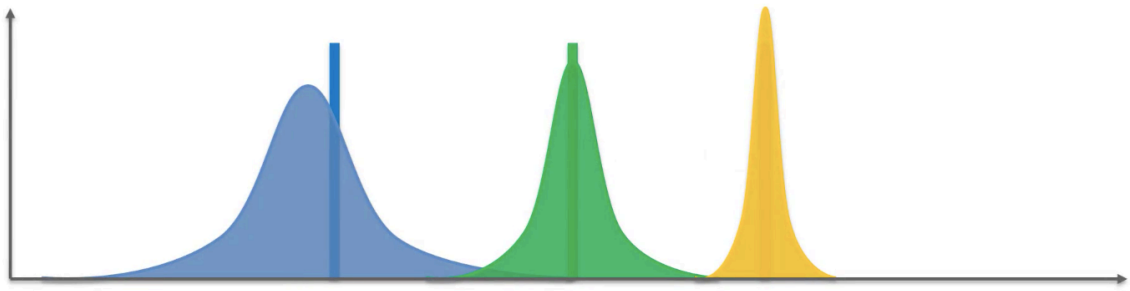


Thompson Sampling Algorithm

Thompson Sampling Algorithm

And so on...

Thompson Sampling Algorithm



Machine Learning A-Z © SuperDataScience