Bayesian Reinforcement Learning: A Survey

Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, Aviv Tamar

Presented by Jacob Nogas ft. Animesh Garg (cameo)

Bayesian RL: What

- Leverage Bayesian Information in RL problem
  - Dynamics
  - Solution space (Policy Class)
- Prior comes from System Designer

Bayesian RL: Why

- Exploration-Exploitation Trade-off
    - Posterior: current representation of world
      Max Gain wrt Current World Belief
- Regularization
    - Prior over Value, Policy (params or class) or Model results in regularization/finite sample estimation.
- Handle Parametric Uncertainty
    - Sampling based methods, aka frequentist, are computationally intractable or very conservative.

Bayesian RL: Challenges

- Selection of the correct Representation for Prior
  - How to know ahead of time?
  - Why is that knowledge not biased?

- Decision-making process over the information state
  - Dynamic Programming over large state-action spaces was hard as it is!
  - Doing this over distributions of states (beliefs) and distributions over latent dynamics model
    Computationally much harder!

- Bandits (Sec 3)
  - Bayes UCB
  - Thompson sampling
- Model-based BRL (Sec 4)
  - Offline value approximation
    - *Finite state controllers*
    - *BEETLE*
  - Online near-myopic value approximation
    - *Bayesian DP*
    - *VOI heuristic*
  - Online tree search approximation
    - *Forward search*
    - *Bayesian sparse sampling*
    - *HMDP*
    - *BFS3*
    - *Branch-and-bound search*
    - *BAMCP*
  - Exploration bonus approximation
    - *BOSS*
    - *BEB*
    - *VBRB*
    - *BOLT*
- Model-free BRL (Sec 5)
  - Value function algos
    - *GPTD*
    - *GPSARSA*
  - Policy gradient algos
    - *Bayesian Quadrature*
    - *Two Bayesian models for estimating the policy gradient*
  - Actor-Critic algos
    - *GPTD + Bayesian policy gradient*
- Risk Aware BRL (Sec 6)
  - Bias variance approximation
  - Percentile criterion
  - Min-max criterion
  - Percentile measures criteria

Preliminaries: POMDP

Model 4 (Partially Observable Markov Decision Process) Define a POMDP $\mathcal{M}$ to be a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, P, \Omega, P_0, q \rangle$ where

- $\mathcal{S}$ is the set of states,
- $\mathcal{A}$ is the set of actions,
- $\mathcal{O}$ is the set of observations,
- $P(\cdot|s,a) \in \mathcal{P}(\mathcal{S})$ is the probability distribution over next states, conditioned on action a being taken in state s,
- $\Omega(\cdot|s,a) \in \mathcal{P}(\mathcal{O})$ is the probability distribution over possible observations, conditioned on action a being taken to reach state s where the observation is perceived,
- $P_0 \in \mathcal{P}(\mathcal{S})$ is the probability distribution according to which the initial state is selected,
- $R(s,a) \sim q(\cdot|s,a) \in \mathcal{P}(\mathbb{R})$ is a random variable representing the reward obtained when action a is taken in state s.

Overview

1. **Bayesian Bandits**
   Introduction
   Bayes UCB and Thompson Sampling
2. Model-based Bayesian Reinforcement Learning
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

Multi-armed Bandits (MAB)

Model 1 (Stochastic K-Armed Bandit) Define a K-MAB to be a tuple $\langle \mathcal{A}, \mathcal{Y}, P, r \rangle$ where

- $\mathcal{A}$ is the set of actions (arms), and $|\mathcal{A}| = K$,
- $\mathcal{Y}$ is the set of possible outcomes,
- $P(\cdot|a) \in \mathcal{P}(\mathcal{Y})$ is the outcome probability, conditioned on action $a \in \mathcal{A}$ being taken,
- $r(Y) \in \mathbb{R}$ represents the reward obtained when outcome $Y \in \mathcal{Y}$ is observed.

Bayesian MAB

- In MAB model, only unknown is outcome probability P(*|a)
- Use Bayesian inference to learn the outcome probability from outcomes observed
- Parameterize outcome

$$P_{\boldsymbol{\theta}}(\cdot|a)$$

- Model our uncertainty about θ

Bayesian MAB - Bernoulli with Beta Prior

$$\theta = (\theta_1, \ldots, \theta_k)$$

$$r(Y) = Y$$

$$Y(a) \sim Bernoulli[\theta_a]$$

$$\theta_a \sim Beta(\alpha_a, \beta_a)$$

$$\theta_a | y \sim Beta(\alpha_a + y, \beta_a + y)$$

Bayesian MAB - Policy Selection

- We can represent our uncertainty about θ with posterior
- How to utilize this representation to select an adequate policy
- Want policy which minimizes regret

Overview

1. **Bayesian Bandits**
   Introduction
   Bayes UCB and Thompson Sampling
2. Model-based Bayesian Reinforcement Learning
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

UCB

- Employs optimistic policy to reduce chance of overlooking the best arm
- Starts by playing each arm once
- At time step t, plays arm a that maximizes the following (<r_a> is mean reward for arm a, t_a is number of times arm a has been played so far)

$$< r_a > + \sqrt{\frac{2 \ln t}{t_a}}$$

Bayes - UCB

- Extend UCB to Bayesian setting
- Keep posterior over expected reward of each arm
- At each step, choose the arm with the maximal posterior (1 - β_t)-quantile, where β_t is of order 1/t
- Using upper quantile instead of posterior mean serves the role of optimism, in the spirit of original UCB

Thompson Sampling

- P_{post} Is posterior over θ
- Sample a parameter $\hat{\theta}$ from posterior, and select optimal action with respect to $\hat{\theta}$
- Amounts to matching action selection probability to the posterior probability of each action being optimal

Thompson Sampling

Algorithm 1 Thompson Sampling

TS(P_{prior})

- P_{prior} prior distribution over $\boldsymbol{\theta}$

$P_{\text{post}} := P_{\text{prior}}$

for $t = 1, 2, \ldots$ **do**

  Sample $\hat{\boldsymbol{\theta}}$ from P_{post}

  Play arm $a_t = \arg\max_{a \in \mathcal{A}} \mathbf{E}_{y \sim P_{\hat{\boldsymbol{\theta}}}(\cdot|a)}[r(y)]$

  Observe outcome Y_t and update P_{post}

end for

Thompson Sampling - Beta Bernoulli

Algorithm 1 Thompson Sampling

1: $\mathbf{TS}(P_{\text{prior}})$
- P_{prior} prior distribution over $\boldsymbol{\theta}$

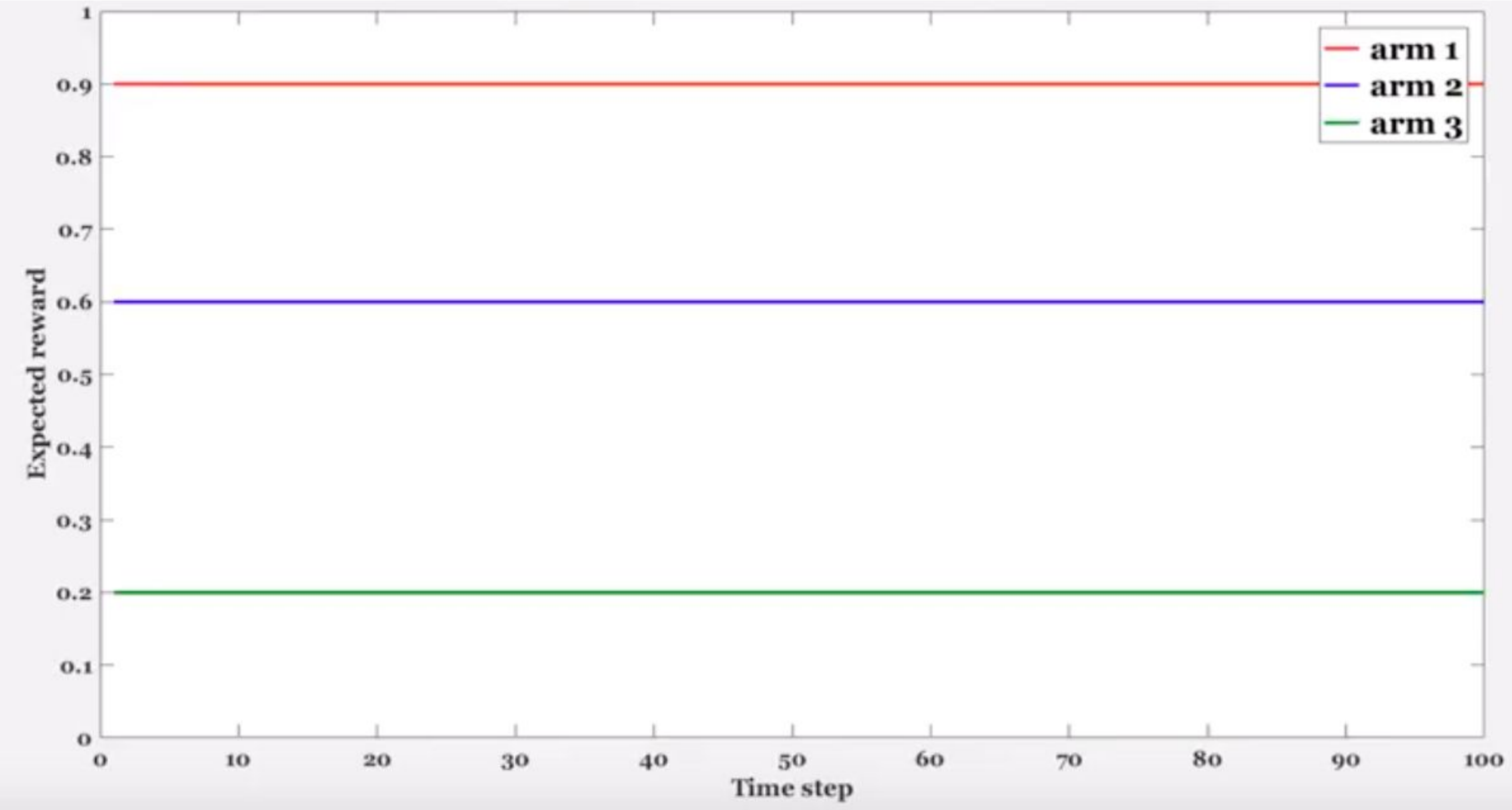
2: $P_{\text{post}} := P_{\text{prior}}$
3: **for** $t = 1, 2, \ldots$ **do**
4: Sample $\hat{\boldsymbol{\theta}}$ from P_{post}
5: Play arm $a_t = \arg\max_{a \in \mathcal{A}} \mathbf{E}_{y \sim P_{\hat{\boldsymbol{\theta}}}(\cdot|a)} [r(y)]$
6: Observe outcome Y_t and update P_{post}
7: **end for**

$$\hat{\theta} = [\hat{\theta_1}, \ldots, \hat{\theta_k}]$$

$$Y(a) \sim \text{Bernoulli}[\boldsymbol{\theta}_a]$$

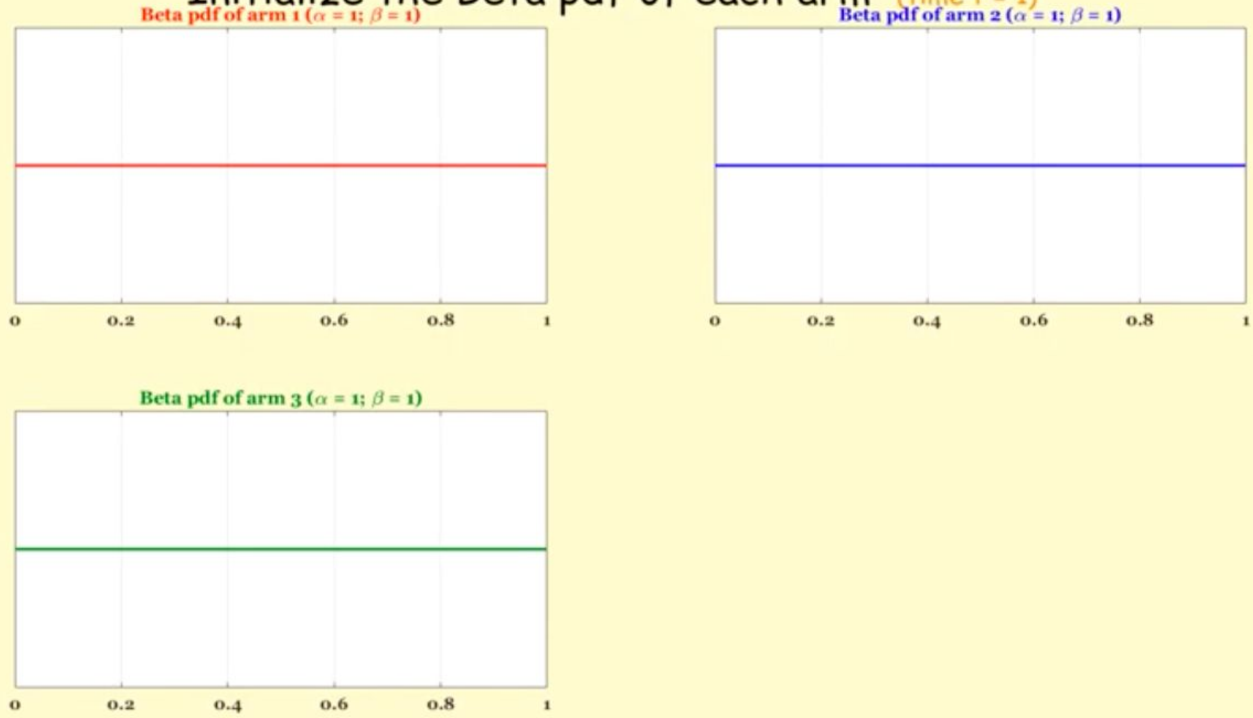
$$argmax_{a \in A}(E_{y \sim Bern[\hat{\theta_{a_1}}]}[y], E_{y \sim Bern[\hat{\theta_{a_2}}]}[y], \ldots, E_{y \sim Bern[\hat{\theta_{a_k}}]}[y]]) =$$

$$argmax_{a \in A}(\hat{\theta_{a_1}}, \hat{\theta_{a_2}}, \ldots, \hat{\theta_{a_k}})$$

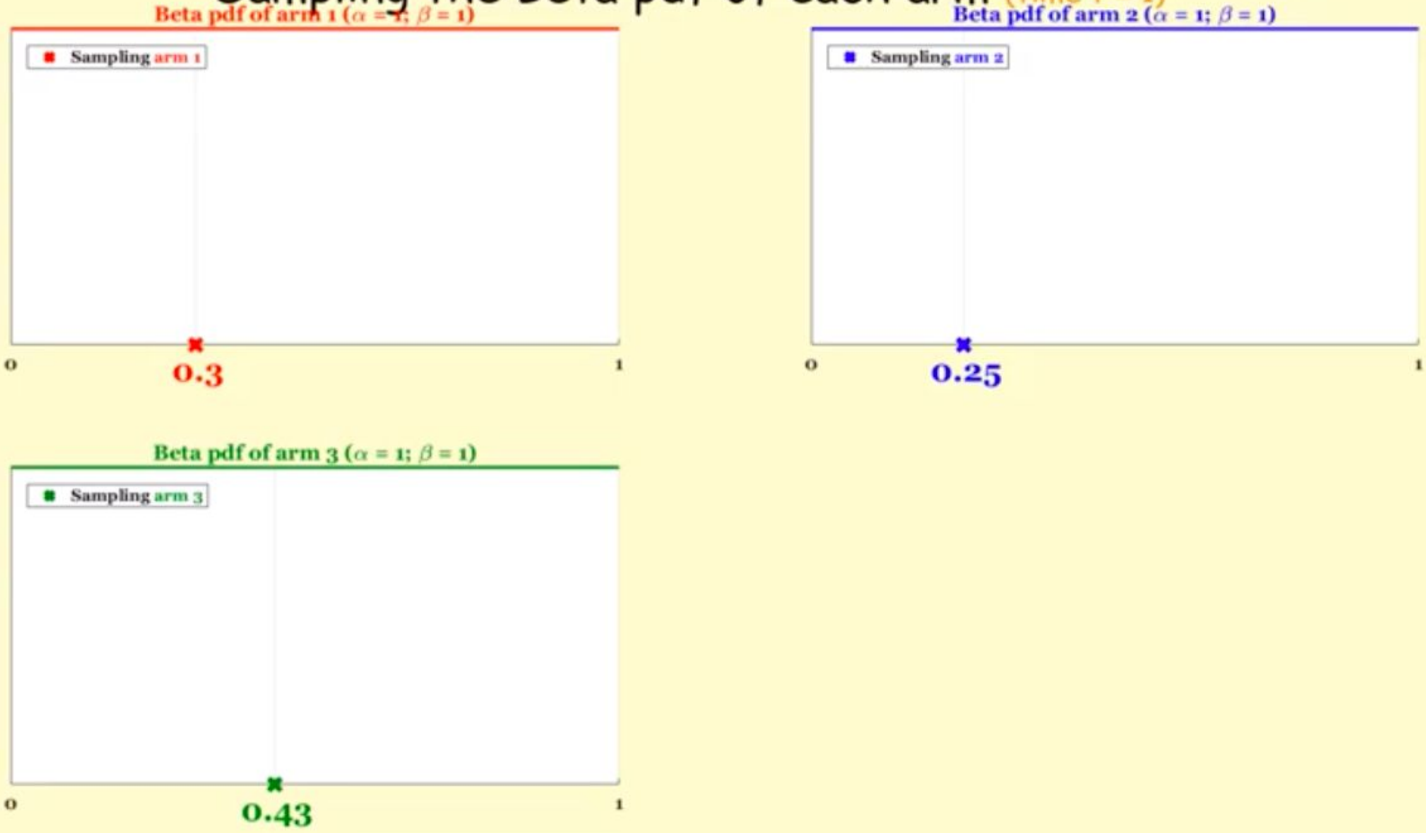


arm 1
arm 2
arm 3
Expected reward
Time step
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
10
20
30
40
50
60
70
80
90
100

Initialize the Beta pdf of each arm (time t = 1)

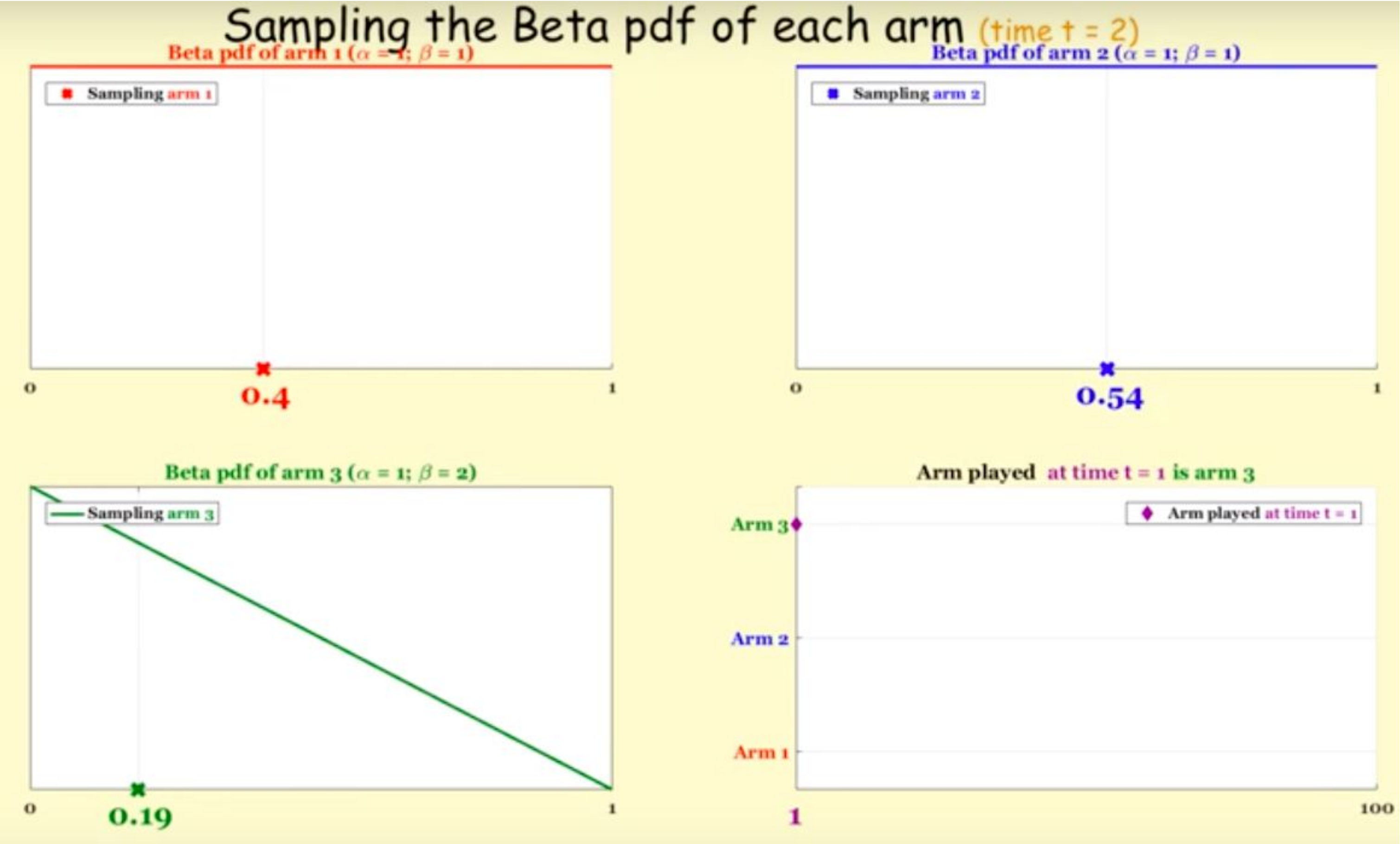


Sampling the Beta pdf of each arm (time t = 1)

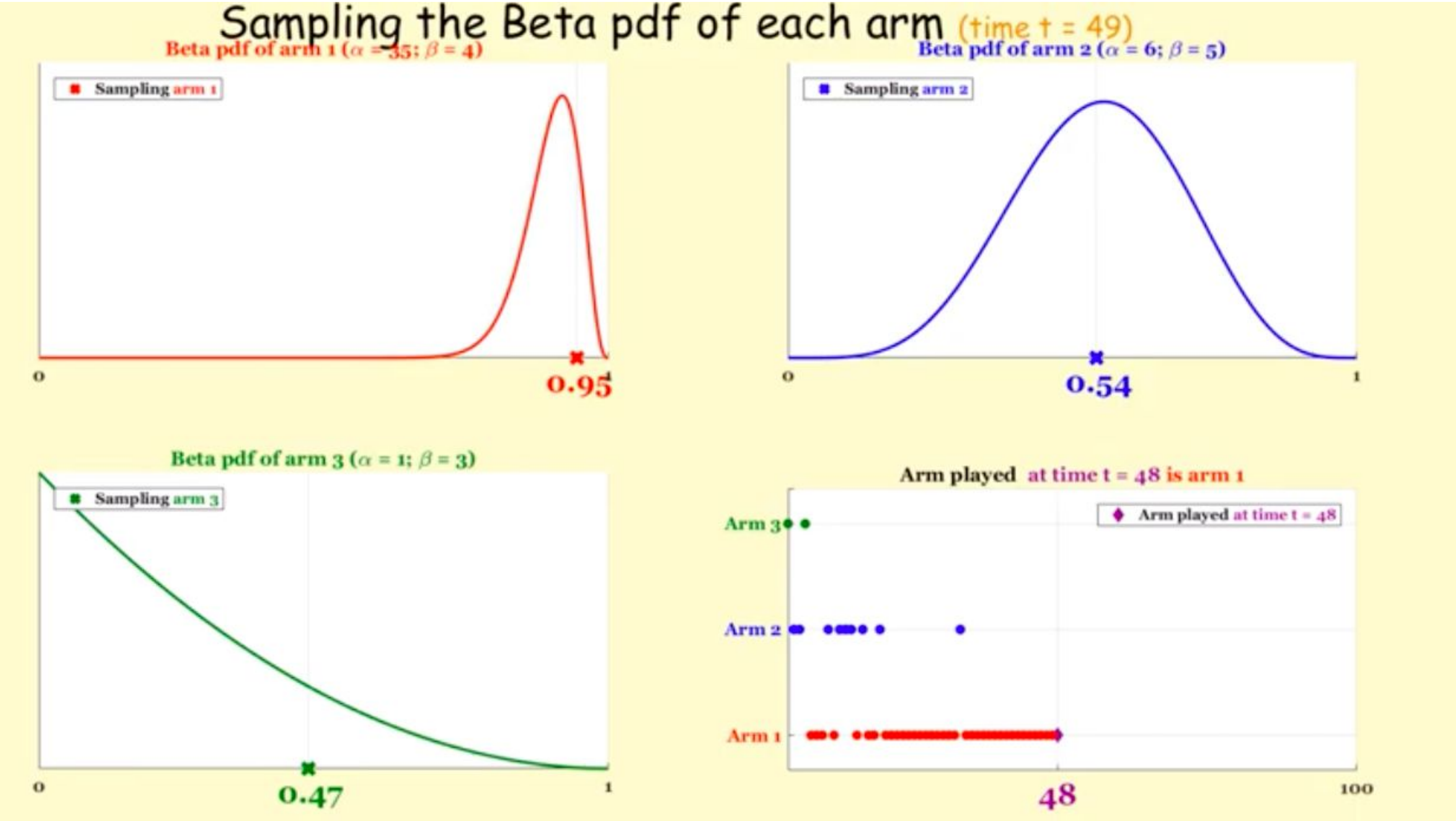


Playing arm 3 (time t = 1) and receive a reward

Update the Beta pdf of arm 3 (time t = 2)



Sampling the Beta pdf of each arm (time t = 2)
Beta pdf of arm 1 (α = 1; β = 1)
Sampling arm 1
0
0.4
1
Beta pdf of arm 2 (α = 1; β = 1)
Sampling arm 2
0
0.54
1
Beta pdf of arm 3 (α = 1; β = 2)
Sampling arm 3
0
0.19
1
Arm played at time t = 1 is arm 3
Arm played at time t = 1
Arm 3
Arm 2
Arm 1
1
100



Slides from https://www.youtube.com/watch?v=qhqAYfPv7mQ

Overview

1. Bayesian Bandits
   Introduction
   Bayes UCB and Thompson Sampling
2. **Model-based Bayesian Reinforcement Learning**
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

Model-based Bayesian Reinforcement Learning

- Represent out uncertainty in model parameters of MDP
- Can be thought of as a POMDP where parameters represent unobservable states
- Keep joint posterior over model parameters and physical state
- Derive optimal policy with respect to this posterior

Bayes-Adaptive MDP

- Assume discrete action/state sets
- Transition probabilities consist of multinomial distributions
- Represent our uncertainty with respect to the true parameters of the multinomial distribution using a Dirichlet distribution

$$(p_1, \ldots, p_k) \sim Dir(\phi_1, \ldots, \phi_k)$$

Bayes-Adaptive MDP

Model 6 (Bayes-Adaptive MDP) Define a Bayes-Adaptive MDP $\mathcal{M}$ to be a tuple $\langle \mathcal{S}', \mathcal{A}, P', P_0', R' \rangle$ where

- $\mathcal{S}'$ is the set of hyper-states, $\mathcal{S} \times \Phi$,
- $\mathcal{A}$ is the set of actions,
- $P'(\cdot|s, \phi, a)$ is the transition function between hyper-states, conditioned on action a being taken in hyper-state (s, ϕ),
- $P_0' \in \mathcal{P}(\mathcal{S} \times \Phi)$ combines the initial distribution over physical states, with the prior over transition functions ϕ_0,
- $R'(s, \phi, a) = R(s, a)$ represents the reward obtained when action a is taken in state s.

BAMDP Transition Model

- The transition model of the BAMDP captures transitions between hyper-states.
- By chain rule:

$$\Pr(s', \phi' | s, a, \phi) = \Pr(s' | s, a, \phi) \Pr(\phi' | s, a, s', \phi)$$

BAMDP Transition Model

- The transition model of the BAMDP captures transitions between hyper-states.
- By chain rule:

$$\Pr(s', \phi' | s, a, \phi) = \Pr(s' | s, a, \phi) \Pr(\phi' | s, a, s', \phi)$$

- First term: taking expectation over all possible transition functions

$$\Pr(s' | s, a, \phi) = \int_p Pr(s' | s, a, \phi, p) b(p) dp = \ldots = \frac{\phi_{s,a,s'}}{\sum_{s'' \in \mathcal{S}} \phi_{s,a,s''}}$$

BAMDP Transition Model

$$\Pr(s', \phi' | s, a, \phi) = \Pr(s' | s, a, \phi) \Pr(\phi' | s, a, s', \phi)$$

- Second Term: update of the posterior φ to φ′ is deterministic

$\Pr(\phi' | s, a, s', \phi)$ is 1 if $\phi'_{s,a,s'} = \phi_{s,a,s'} + 1$, and

0, otherwise.

BAMDP Transition Model

$$\Pr(s', \phi' | s, a, \phi) = \Pr(s' | s, a, \phi) \Pr(\phi' | s, a, s', \phi)$$

$$= \frac{\phi_{s,a,s'}}{\sum_{s'' \in \mathcal{S}} \phi_{s,a,s''}} \mathbb{I}(\phi'_{s,a,s'} = \phi_{s,a,s'} + 1)$$

BAMDP - Number of States

- Initially (at $t = 0$), there are only $|S|$ stas, one per real MDP, state (we assume a single prior $\varphi 0$ is specified).
- Assuming a fully connected state space in the underlying MDP (i.e., $P(s'|s, a) > 0, \ \forall s, a$), then at $t = 1$ there are already $|S|\times|S|$ states, since $\varphi \rightarrow \varphi'$ can increment the count of any one of its $|S|$ components. So at horizon t, there are |S|^t reachable states in the BAMDP.
- There are clear computational challenges in computing an optimal policy over all such beliefs.

BAMDP - Value Function

- Any policy which maximizes this expression is called Bayes Optimal

$$V^*(s,\phi) = \max_{a\in\mathcal{A}} \left[R'(s,\phi,a) + \gamma \sum_{(s',\phi')\in\mathcal{S}'} P'(s',\phi'|s,\phi,a)V^*(s',\phi') \right]$$

$$= \max_{a\in\mathcal{A}} \left[R(s,a) + \gamma \sum_{s'\in\mathcal{S}} \frac{\phi^a_{s,s'}}{\sum_{s''\in\mathcal{S}} \phi^a_{s,s''}} V^*(s',\phi') \right].$$

Bayes Optimal Planning

- Planning algorithms which seek a Bayes optimal policy are typically based on heuristics and/or approximations due to complexity noted above

Planning Algorithms Seeking Bayes Optimality

- Offline value approximation
  - Compute policy apriori for any possible state and posterior
  - Compute action selection strategy to optimize expected return over hyper-states of the BAMDP
  - Intractable in most domains, these methods devise approximate algorithms which leverage structural constraints

- Online near myopic value approximation
  - In practice may be fewer than |S|^t states; some trajectories will not be observed.
  - Interleave planning and execution on a step-by-step basis

- Methods with exploration bonus to achieve PAC Guarantees
  - Select actions such as to incur only a small loss compared to the optimal Bayesian policy
  - Typically employ Optimism in the Face of Uncertainty; when in doubt, an agent should act according to an optimistic model of the MDP

Overview

1. Bayesian Bandits
   Introduction
   Bayes UCB and Thompson Sampling
2. **Model-based Bayesian Reinforcement Learning**
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

Online - Bayesian Dynamic Programming

- Example of online near-myopic value approximation
- Generalization of TS
- Get estimate of Q function we would get if using transition model Pr(theta) directly
- Convergence to optimal policy is achievable
- Recent work has provided the first Bayesian regret bounds

ThompsonSamplingInBayesianRL(s,b)

Repeat

Sample $\theta_1, \ldots, \theta_k \sim \Pr(\theta) \ \forall a$

$Q^*_{\theta_i} \leftarrow solve(MDP_{\theta_i})$

$\hat{Q}(s,a) \leftarrow \frac{1}{k}\sum_{i=1}^{k} Q^*_{\theta_i}(s,a)$

$a^* \leftarrow \text{argmax}_a \hat{Q}(s,a)$

Execute a^* and receive r, s'

$b(\theta) \leftarrow b(\theta)\Pr(r, s'|s, a, \theta)$

$s \leftarrow s'$

Online - Tree Search Approximation - Forward Search

- Select actions using a more complete characterization of the model uncertainty

- Perform forward search in the space of hyper-states

- Consider current hyper-state, build fixed-depth forward search tree containing all hyper-states reachable within some fixe planning horizon, denoted d

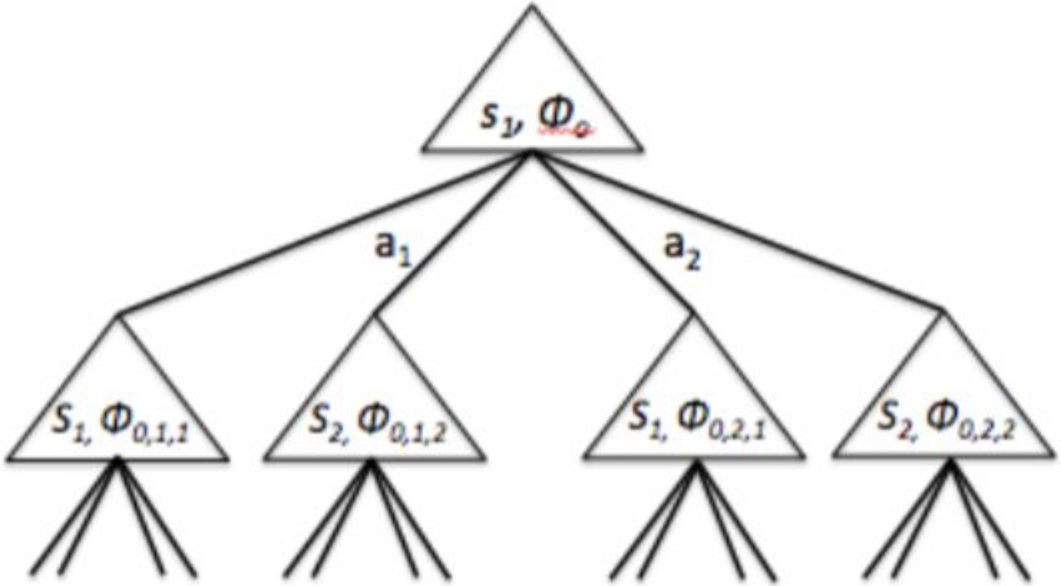
- Use dynamic programming to approximate expected return of possible actions at the root of the hyper-state

- Action with highest return is executed, and then forward search is conducted on the next hyper-state

Online - Tree Search Approximation - Forward Search

- The top node contains the initial state 1 and the prior over the model ϕ_0

- After the first action, the agent can end up in either state 1 or state 2, and updates its posterior accordingly



Online - Tree Search Approximation - Forward Search

- The main limitation of this approach is the fact that for most domains, a full forward search (i.e., without pruning of the search tree) can only be achieved over a very short decision horizon

- the number of nodes explored is $O(|S|^d)$

- Also requires specifying default value function at leaf nodes (since using dynamic programing back ups)

Online - Bayesian Sparse Sampling

- Estimates the optimal value function of a BAMDP (Equation 4.3) using Monte-Carlo sampling

- Instead of looking at all actions at each level of tree, actions are sampled according to their likelihood of being optimal, according to their Q-value distributions (as defined by Dirichlet posteriors)

- Next states are sampled according to the Dirichlet posterior on the model

- This approach requires repeatedly sampling from the posterior to find which action has the highest Q-value at each state node in the tree. This can be very time consuming, and thus, so far the approach has only been applied to small MDPs.

Overview

1. Bayesian Bandits
   Introduction
   Bayes UCB and Thompson Sampling
2. **Model-based Bayesian Reinforcement Learning**
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

Methods with exploration bonus to achieve PAC Guarantees

- Select actions such as to incur only a small loss compared to the optimal Bayesian policy
- Typically employ Optimism in the Face of Uncertainty; when in doubt, an agent should act according to an optimistic model of the MDP
- Shown to achieve bounded error in a polynomial number of steps using analysis from Probably Approximately Correct (PAC) literature

BFS3: Bayesian Forward Search Sparse Sampling

- Maintains both lower and upper bounds on the value of each state-action pair, and uses this information to direct forward rollouts in the search tree

- Consider a node s in the tree, then the next action is chosen greedily with respect to the upper-bound U(s,a)

- The next state s′ is selected to be the one with the largest difference between its lower and upper bound (weighted by the number of times it was visited)

BFS3: Bayesian Forward Search Sparse Sampling

Theorem [Asmuth, 2013]: With probability at least $1 - \delta$, the expected number of sub-ε-Bayes-optimal actions taken by BFS3 is at most $BSA(S + 1)d/\delta t$ under assumptions on the accuracy of the prior and optimism of the underlying FSSS procedure.

Overview

1. Bayesian Bandits
   Introduction
   Bayes UCB and Thompson Sampling
2. **Model-based Bayesian Reinforcement Learning**
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. Model-free Bayesian Reinforcement Learning

Offline - Bayesian Exploration Exploitation Tradeoff in LEarning (BEETLE)

- Optimal value function for a finite-horizon POMDP can be shown to be piecewise-linear and convex; can be represented by a finite set of linear segments $\alpha_0, \ldots, \alpha_m$

- The value of a given αi at a belief bt is evaluated as follows:

$$\alpha_i(b_t) = \int_S \alpha_i(s) b_t(s) ds.$$

$$V_t^*(b_t) = \max_{\alpha \in \Gamma_t} \int_S \alpha(s) b_t(s)$$

Offline - Bayesian Exploration Exploitation Tradeoff in LEarning (BEETLE)

- Hyper-states (s, ϕ) are sampled from random interactions with BAMDP model
- An equivalent continuous POMDP is solved assuming b = (s, ϕ) is a belief state in that POMDP
- The set of α-functions are constructed incrementally applying Bellman updates at the sampled hyper states using standard point-based POMDP method

Offline - Bayesian Exploration Exploitation Tradeoff in LEarning (BEETLE)

- The constructed α-functions can be shown to be multivariate polynomials
- The main computational challenge is that the number of terms in the polynomials increases exponentially with the planning horizon
- The key to applying it in larger domains is to leverage knowledge about the structure of the domain to limit the parameter inference to a few key parameters, or by using parameter tying (whereby a subset of parameters are constrained to have the same posterior)

Overview

1. Bayesian Bandits
   Introduction
   Bayes UCB and Thompson Sampling
2. Model-based Bayesian Reinforcement Learning
   Introduction
   Online near myopic value approximation
   Methods with exploration bonus to achieve PAC Guarantees
   Offline value approximation
3. **Model-free Bayesian Reinforcement Learning**

Model-free Bayesian Reinforcement Learning