Computer Engineering Department

Bayesian Reinforcement Learning

Mohammad Hossein Rohban, Ph.D.

Hosein Hasani

Spring 2023

Courtesy: Most of slides are adopted from CS 885 UWaterloo.

Outline

- Bayesian RL
- Belief MDP
- Value iteration with belief model
- Thompson sampling in Bayesian RL
- Model-based Bayesian actor critic
- Model-based RL with ensembles (previously seen)

Bayesian RL

- Explicit representation of uncertainty
- Benefits
  - Balance exploration/exploitation tradeoff
  - Mitigate model bias
  - Reduce data needs
- Drawbacks
  - Complex computation
  - Poor scalability

MDP (Recap)

- **MDP** in traditional RL:
  - States: $s \in \mathcal{S}$
  - Actions: $a \in \mathcal{A}$
  - Rewards: $r \in \mathbb{R}$
  - Unknown model: $p(r, s' | s, a; \theta)$
- **Goal**:
  find policy $\pi : \mathcal{S} \to \mathcal{A}$
  or value function: $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

Belief MDP

- **Information-state MDP**:
  - Information states: $(s, b) \in \mathcal{S} \times \mathcal{B}$
    - Physical states: $s \in \mathcal{S}$
    - Belief states: b $\in \mathcal{B}$ where $b(\theta) = p(\theta)$
  - Actions:$a \in \mathcal{A}$
  - Rewards: $r \in \mathbb{R}$
  - Known model: $p(r, s', b' | s, b, a)$
- **Goal**:
  find policy $\pi: \mathcal{S} \times \mathcal{B} \rightarrow \mathcal{A}$
  or value function: $Q: \mathcal{S} \times \mathcal{B} \times \mathcal{A} \rightarrow \mathbb{R}$

Idea: augment state with distribution about unknown model parameters!

Model in Bayesian RL

- Claim: the model in Bayesian RL is known!

$$p(r, s', b' | s, b, a) = \underbrace{p(r, s' | s, b, a)}_{physical\ model} \underbrace{p(b' | r, s', s, b, a)}_{\text{belief}\ model}$$

- Idea: marginalize out unknown θ w.r.t uncertainty $b(\theta)$

$$p(r, s' | s, b, a) = \int p(r, s', \theta | s, b, a) d\theta = \int p(r, s' | s, a, \theta) b(\theta) d\theta$$

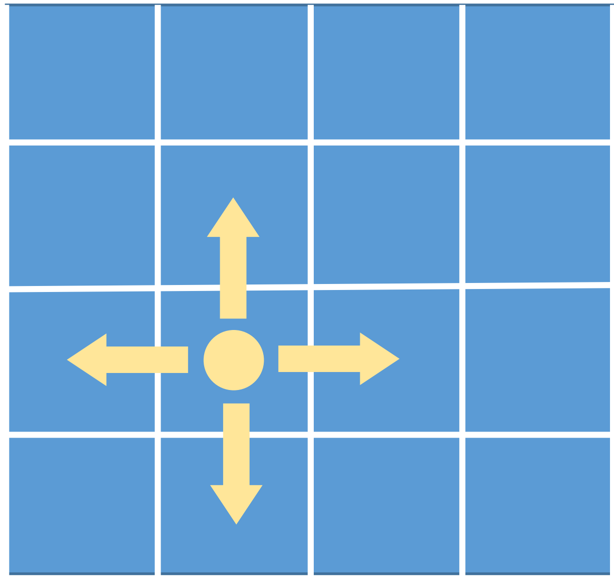
- Idea: b' is the posterior belief (deterministic for posterior update)

$$p(b' | r, s', s, b, a) = \begin{cases} 1, & if\ b'(\theta) = \boxed{p(\theta | s, a, s, r')} \\ & 0, otherwise. \end{cases}$$

$\boxed{b^{s,a,s',r}}$

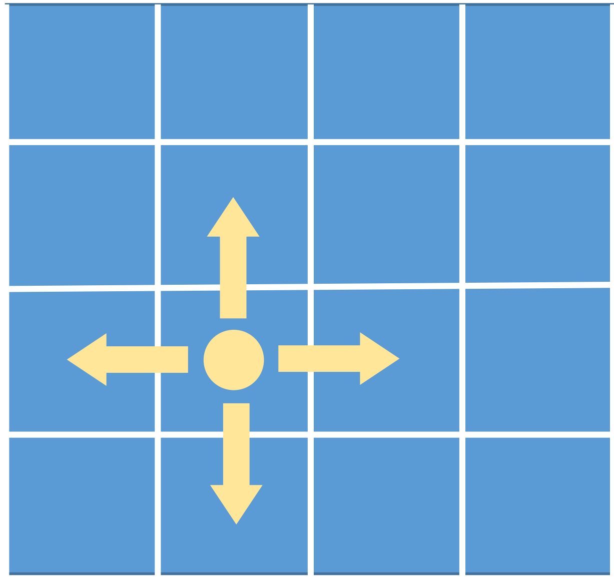
simpler notation

Belief MDP: Grid World Example



- $\mathcal{A} = \{left, right, up, down\}$
- Transitions are stochastic:
  - True direction, with probability of θ. For example: $p(dir = up|a = up) = \theta$
  - Each of other three (False) directions, with probability of $\frac{1-\theta}{3}$

Belief MDP: Grid World Example



- Belief state:

Modelling choice: Let's model our uncertainty with respect to θ by a Beta distribution

$$b(\theta) = Beta(\theta; \alpha, \beta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$$

Belief MDP: Grid World Example

- Prediction:

  Given current belief, what's the probability of going to "right" after selecting the action "left"?

- Predictive distribution:

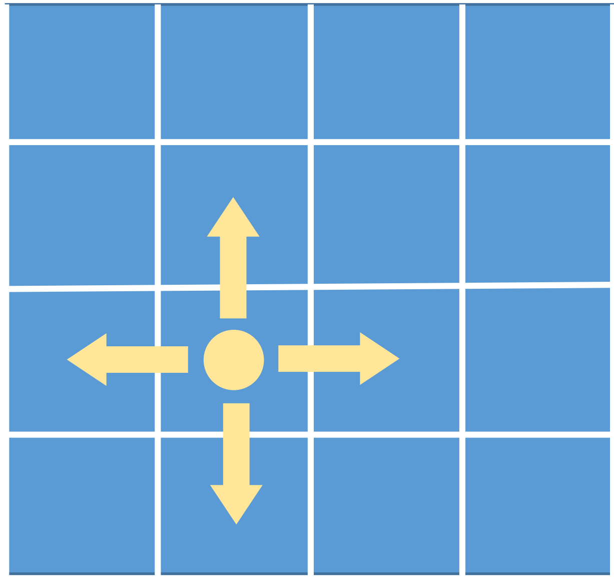
$$p(s'|s,b,a) = \int p(s'|s,a,\theta)b(\theta)d\theta$$

$$= \int p(\,dir = right|a = left,\theta)Beta(\theta;\alpha,\beta)d\theta$$

$$= \int \frac{(1-\theta)}{3} Beta(\theta;\alpha,\beta)d\theta$$

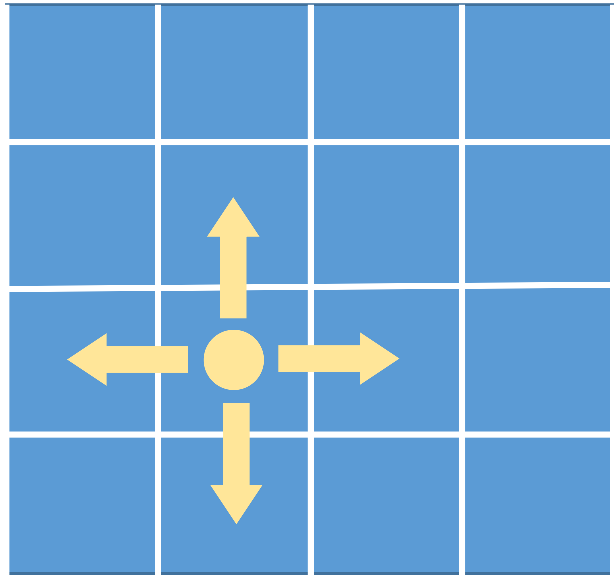
$$= \frac{1}{3}\left(1 - \mathbb{E}_{\theta\sim\beta(\alpha,\beta)}[\theta]\right)$$

$$= \frac{1}{3}\left(1 - \frac{\alpha}{\alpha+\beta}\right) = \frac{\beta}{3(\alpha+\beta)}$$



$$\mathbb{E}_{\theta\sim\beta(\alpha,\beta)}[\theta] = \frac{\alpha}{\alpha+\beta}$$

Belief MDP: Grid World Example



- Belief update:

By applying Bayes' theorem

$$b'(\theta) = b(\theta|s, a, s')$$
$$\propto p(s'|s, a, \theta)b(\theta)$$

- Belief update after a single observation:

Agent selects "left" and then goes to the "left"

$$b'(\theta) \propto b(\theta)p(dir = left|a = left)$$
$$\propto \theta^{\alpha-1}(1-\theta)^{\beta-1}\theta$$
$$\propto \theta^{\alpha}(1-\theta)^{\beta-1}$$
$$\propto Beta(\theta; \alpha+1, \beta)$$

Planning

- Since the model is known, treat Bayesian RL as an MDP
- Benefits:
  - Solve RL problem by planning (e.g., value/policy iteration)
  - Optimal exploration/exploitation tradeoff
- Drawback:
  - Complex computation
- Bellman's optimality equation:

$$V^*(s,b) = \max_a \mathbb{E}_{r,\theta}[r|s,b,a] + \sum_{s'} p(s'|s,b,a)V^*(s',b^{s,a,s'})$$

Value Iteration

- Traditional MDP

valueIteration(MDP)

$V_0^*(s) \leftarrow \max_a E[r|s,a] \ \forall s$

For $t = 1$ to h do

$$V_t^*(s) \leftarrow \max_a E[r|s,a] + \gamma \sum_{s'} \Pr(s'|s,a)\, V_{t-1}^*(s') \ \forall s$$

Return V^*

- Information state MDP

valueIteration(BayesianRL)

$V_0^*(s,b) \leftarrow \max_a E[r|s,b,a] \ \forall s$

For $t = 1$ to h do

$$V_t^*(s,b) \leftarrow \max_a E[r|s,b,a] + \gamma \sum_{s'} \Pr(s'|s,a,b)\, V_{t-1}^*(s', b^{s,a,s'}) \ \forall s$$

Return V^*

Bayesian RL

Two phases:

- Offline planning (without the environment)

Find π^* and/or V^*
by policy/value iteration or any other algorithm

- Online execution (with the environment)

Initialize $s_0, b_0, n \leftarrow 0$
Repeat
    Execute policy $a_n \leftarrow \pi(s_n, b_n)$
    receive s_{n+1} and r_n from the environment
    Belief update: $b_{n+1}(\theta) = b_n^{s_n, a_n, r_n, s_{n+1}}(\theta) = b_n(\theta | s_n, a_n, r_n', s_{n+1}')$
    $n \leftarrow n + 1$

Exploration/Exploitation Tradeoff

- Dilemma:
  - ~~Maximize immediate rewards (exploitation)?~~
  - ~~Or, maximize information gain (exploration)?~~

wrong question!

- Single objective: max expected total rewards
  - $V^{\pi}(s,b) = \sum_t \gamma^t \mathbb{E}[r_t | s_t, b_t]$
  - Optimal policy π^*: $V^{\pi^*}(s,b) \geq V^{\pi}(s,b)\ \forall s$

optimal exploration/exploitation tradeoff! (given prior knowledge)

Challenges in Bayesian RL

- Offline planning is notoriously difficult
  - Continuous information space
  - Use function approximators for V and π
  - Problem: a good plan should implicitly account for all possible environments, which is intractable
- Alternative: online partial planning
  - Thompson sampling
  - PILCO (Model-based Bayesian Actor Critic)

Thompson Sampling in Bayesian RL

Idea: Sample models θ at each step and plan for the corresponding MDP_θ

ThompsonSamplingInBayesianRL(s,b)

Repeat

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

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

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

  $a^* \leftarrow \mathrm{argmax_a}\ \hat{Q}(s,a)$

  Execute a^* and receive r, s'

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

  $s \leftarrow s'$

Model-based Bayesian Actor Critic

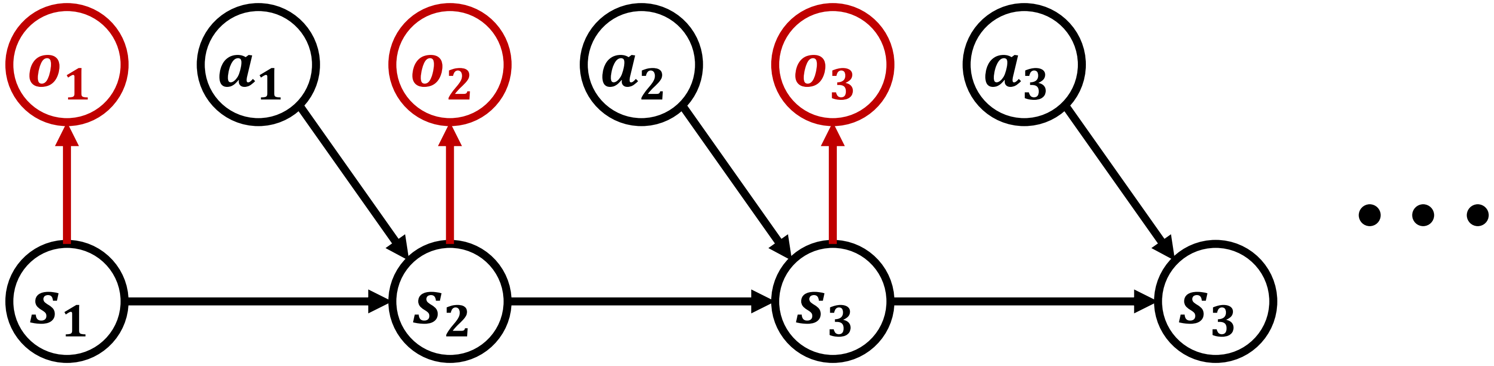
- PILCO: Deisenroth, Rasmussen (2011):
  $b(\theta)$: Gaussian Process transition model
- Deep PILCO: Gal, McCallister, Rasmussen (2016):
  $b(\theta)$: Bayesian neural network transition model

PILCO(s, b, π)
    Repeat
        Repeat
            Critic: $V_b^\pi \leftarrow policyEvaluation(b, \pi)$
            Actor: $\pi \leftarrow \pi + \alpha\, \partial V_b^\pi / \partial \pi$
        $a \leftarrow \pi(s, b)$
        Execute a and receive r, s'
        $b \leftarrow b^{s,a,r,s'}$ and $s \leftarrow s'$

Partially Observable Markov Decision Process (POMDP)

- MDP augmented with observations
- States (s_t) are not observable anymore
- Relation to Hidden Markov Models
- We can not assume Markov assumption on observations:

$$p(o_t|o_{1:t-1}) \neq p(o_t|o_{t-1})$$



Partially Observable Markov Decision Process (POMDP)

- Definition
  - States: $s \in \mathcal{S}$
  - Observations: $o \in \mathcal{O}$
  - Actions: $a \in \mathcal{A}$
  - Rewards: $r \in \mathbb{R}$
  - Transition model: $p(s_t|s_{t-1}, a_{t-1})$
  - Observation model: $p(o_t|s_t)$
  - Reward model: $p(r_t|s_t, a_t)$

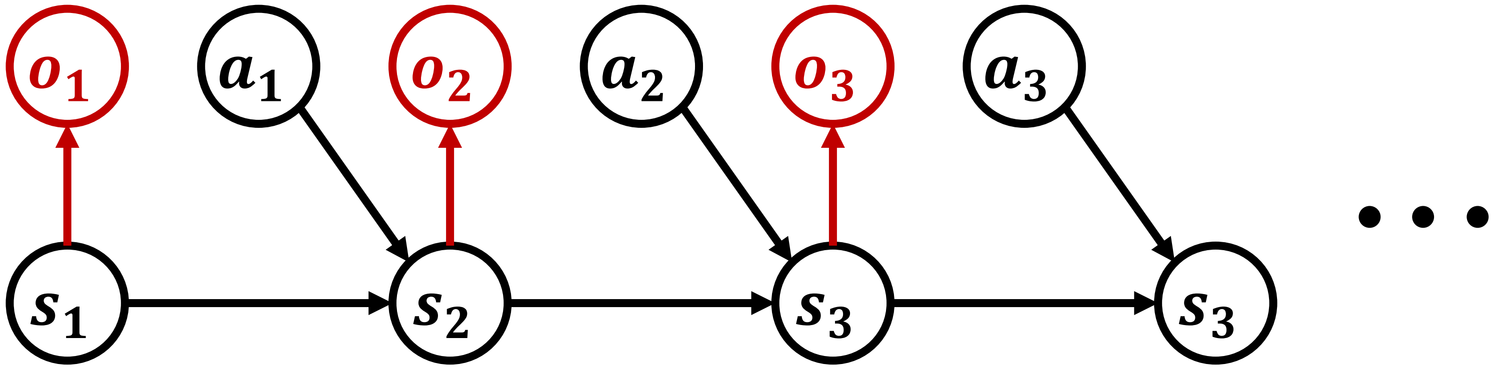
unknown models

- Goal: find optimal policy π^* such that

$$\pi^* = argmax_\pi \sum_t \mathbb{E}_\pi[r_t]$$

POMDP: Simple Heuristic

- Approximate by s_t by o_t (or finite window of previous observations: $o_{t-k:t}$)
- Use favorite RL algorithms on observations instead of states



POMDP: More Sophisticated Methods

Idea: summarize information of past observations in an array (suff. stat.)

- Probabilistic modelling
  - Belief monitoring (HMM)

$$p(s_t|o_{1:t}) = p(o_t|s_t) \sum_{s_{t-1}} p(s_t|s_{t-1})\, p(s_{t-1}|o_{1:t-1})$$

  - Variational inference: treat hidden states as latent variables
- Deep neural networks for sequential data
  - Recurrent neural network (RNNs)
  - Transformers