marginparsep has been altered.

topmargin has been altered.

marginparwidth has been altered.

marginparpush has been altered.

The page layout violates the ICML style.

Please do not change the page layout, or include packages like geometry, savetrees, or fullpage, which change it for you.

We're not able to reliably undo arbitrary changes to the style. Please remove the offending package(s), or layout-changing commands and try again.

Improved UCRL2

Tianyi Gu, Reazul Hasan Russel, Marek Petrik
University of New Hampshire

. Improved UCRL2

Markov decision processes (MDPs) provide a versatile methodology for modeling dynamic decision problems under uncertainty. MDPs assume that transition probabilities are known precisely, but this is rarely the case in reinforcement learning. Errors in transition probabilities often results in probabilities often results in policies that are brittle and fail in real-world deployments. The agent has to learn the true dynamics of the MDP as it optimize the performance while interacts with its environment. The key to evaluate RL algorithms is to check how they balance between exploration that gains information about unknown states (actions) and exploitation to achieve near-term performance.

OFU-RL

Posterior sampling

Our work

. Problem formulation

We consider the problem of learning and solving an uncertain MDP :$(S, A, P^M a, R^M,)$

Algorithm 1 Bayesian Confidence Interval (BCI)

Distribution θ over $p^\star_{s,a}$, confidence level δ, sample count m Nominal point $\bar{p}_{s,a}$ and L_1 norm size $\psi_{s,a}$ Sample $X_1, \ldots, X_m \in \Delta^S$ from θ: $X_i \sim \theta$ Nominal point: $\bar{p}_{s,a} \leftarrow (1/m) \sum_{i=1}^m X_i$ Compute distances $d_i \leftarrow \|\bar{p}_{s,a} - X_i\|_1$ and sort *increasingly* Norm size: $\psi_{s,a} \leftarrow d_{(1-\delta)\,m}$ $\bar{p}_{s,a}$ and $\psi_{s,a}$

Algorithm 2 Bayes UCRL

Desired confidence level δ and prior distribution Policy with an optimistic return estimate num episodes

Initialize MDP: M Compute posterior: $\tilde{p} \leftarrow$ compute_posterior(prior, samples)

$s \in \mathcal{S}, a \in \mathcal{A} \bar{p}_{s,a}, \psi_{s,a} \leftarrow$ Invoke Algortihm **??** with $\tilde{p}$, δ $M \leftarrow$ add transition with $\bar{p}_{s,a}, \psi_{s,a}$

Compute policy by solving MDP: $\hat{\pi} \leftarrow$ Solve M

Collect samples by executing the policy: samples $\leftarrow$ execute $\hat{\pi}$ prior $\leftarrow$ posterior $(\pi_k, p_0^\mathsf{T} v_k)$

Algorithm 3 Bayes UCRL

Prior distribution f, $t = 1$ episodes k =1,2,...

$M_k = \{\}$

sample i sample $M_i \sim f(\cdot|H_{tk})$

$M_k = M_k \cup M_i$ compute $\mu_k \in argmax_{\mu, M \in M_k} V^M_{\mu,1}$

timestep h=1,...,H take acton $a_{kh} = \mu_k(s_{kh}, h)$

update $H_{kh+1} = H_{kh} \cup (s_{kh}, a_{kh}, r_{kh}, skh + 1)$

. OFVF and Bayes UCRL

. Experiments

We start with a simple problem with: 1 non-terminal state, 3 possible actions. Each action leads to 3 terminal states with probability [0.6,0.2,0.2],[0.2,0.6,0.2] and [0.2,0.2,0.6] respectively. The reward vector for the 3 terminal states is [10., 20., 30.]

. References

Osband, I., Russo, D., & Van Roy, B. (2013). (More) Efficient Reinforcement Learning via Posterior Sampling. , 1–10. Retrieved from `http://arxiv.org/abs/1306.0940`

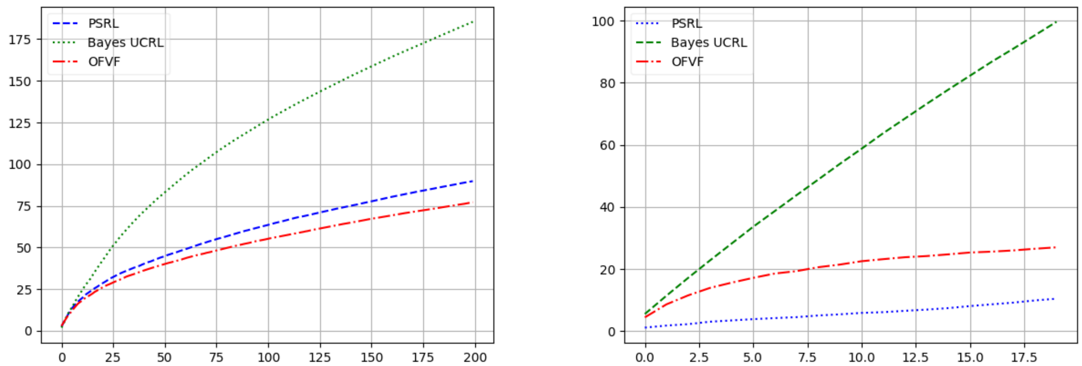

Figure 1. Cumulative regrets of PSRL and Bayes UCRL: left) above described simple problem, right) RiverSwim Problem described in (Osband et al., 2013)