facebook
Artificial Intelligence Research

Exploration-Exploitation in Reinforcement Learning

Part 4 – Regret Minimization in Continuous MDPs

Mohammad Ghavamzadeh, Alessandro Lazaric and Matteo Pirotta
Facebook AI Research

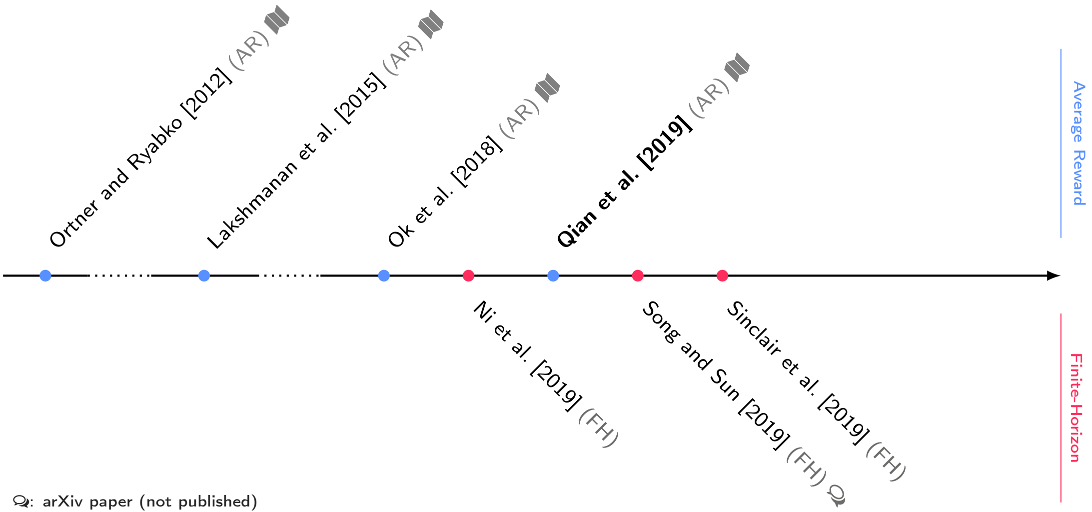
Outline

1. Smooth MDPs
   - Adaptive Q-Learning
2. Linear Structure
   - Low-Rank MDPs
   - LQR

Website

https://rlgammazero.github.io

History: Regret Minimization in Smooth MDPs



: arXiv paper (not published)
: model-based

Smooth Problems

$\mathcal{S} \times \mathcal{A}$ is a *compact metric space*

$$d[(s,a),(s',a')] \text{ is a metric on } \mathcal{S} \times \mathcal{A}$$

$Q^\star$ is a *smooth function*: $\forall (s,a,s',a')$ and $\forall h \in [H]$

$$|Q_h^\star(s,a) - Q_h^\star(s',a')| \leq L_{q,h}\, d\left[(s,a),(s',a')\right]$$

Examples

- Discrete and continuous state-action spaces
- Deterministic systems with metric structure
- Stochastic systems with regularity assumptions on the transitions

Smooth MDPs

A *smooth continuous* MDP has

- $\mathcal{S}, \mathcal{A}$ measurable spaces
- *Transitions and rewards are "smooth"*: $\forall(s, a, s', a')$ and $\forall h \in [H]$

e.g., Total Variation (TV) or Wasserstein

$$d_M \left[p_h(\cdot|s,a), p_h(\cdot|s',a')\right] \le \lambda_p \, d\left[(s,a),(s',a')\right]$$

$$|r_h(s,a) - r_h(s',a')| \le \lambda_r \, d\left[(s,a),(s',a')\right]$$

Smooth transitions and rewards $\Longrightarrow$ smooth Q-function ($\not\Longleftarrow$)

Total Variation $\mapsto$ $L_{q,h} = 2\lambda_p(H-h) + \lambda_r,$ Wasserstein $\mapsto$ $L_{q,h} = \sum_{h'=h}^{H} \lambda_r \lambda_p^{H-h'}$

Total Variation Liptschitz $\Longrightarrow$ Wasserstein Lipschitz [Gibbs and Su, 2002]

Lower-Bound in Metric Space

Theorem (Sinclair et al. [2019])

Consider a metric space $\mathcal{S} \times \mathcal{A}$ such that $diam(\mathcal{S} \times \mathcal{A}) \leq d_{\max}$. Let d_c be the covering dimension with parameter c of the space $\mathcal{S} \times \mathcal{A}$

$$d_c = \inf \left\{ d \geq 0 \mid N_r \leq c r^{-d}, \forall r \in (0, d_{\max}] \right\}$$

where N_r is the packing number.
Then there exists a distribution over problem instances such that for any algorithm, the regret is at least

$$\Omega\left(H^{3/2} K^{(d_c+1)/(d_c+2)} c^{1/(d_c+2)} \right)$$

Adapted from RL [Jin et al., 2018] and the contextual bandit case [Slivkins, 2014]

Online Learning in Smooth Problems

Model-based

- Estimate both p and r

  + Counterfactual reasoning - Computational complexity
- *Optimism*: [Ortner and Ryabko, 2012, Lakshmanan et al., 2015, Qian et al., 2019]*
- *Randomization*: ?

Model-free algorithms

- Eschew learning transitions and only focus on learning good state-action mappings

  + No need of planning - Estimate only $Q^\star$
- *Optimism*: Q-learning $\widetilde{\mathcal{O}}(H^{5/2}K^{(d+1)/(d+2)})$ [Song and Sun, 2019, Sinclair et al., 2019]
- *Randomization*: ?

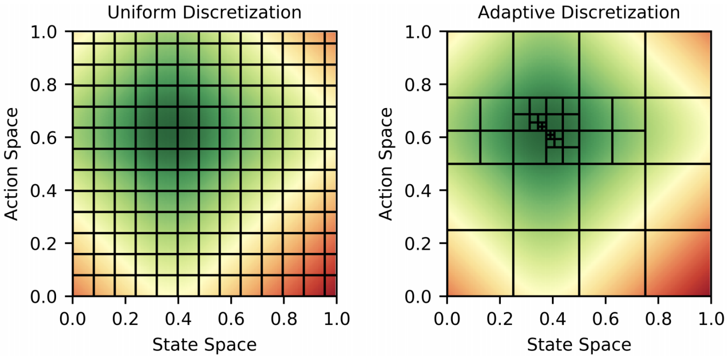
*Not reporting bounds because in infinite-horizon and/or slightly different assumptions.

Solution Methods

- Uniform discretization (e.g., ϵ-net)

  [Ortner and Ryabko, 2012, Lakshmanan et al., 2015, Qian et al., 2019, Song and Sun, 2019]
- Adaptive discretization (e.g., zooming)

  adapt the discretization over space and time in a data-driven manner [Sinclair et al., 2019]



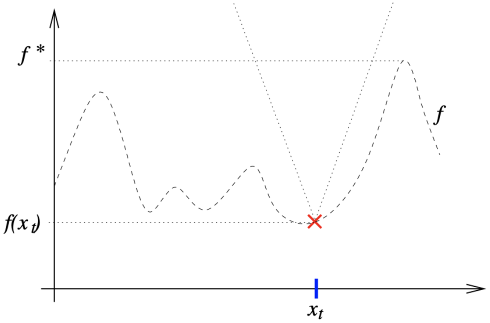
* figure by Sinclair et al. [2019]

Adaptive Partitioning in Bandits

Find *online* the maximum of a function f.
Assume f is Lipschitz: $|f(x) - f(y)| \leq d(x, y)$.

- At each time step t, select x_t
- Observe $f(x_t)$
- Goal: maximize sum of $f(x_t)$

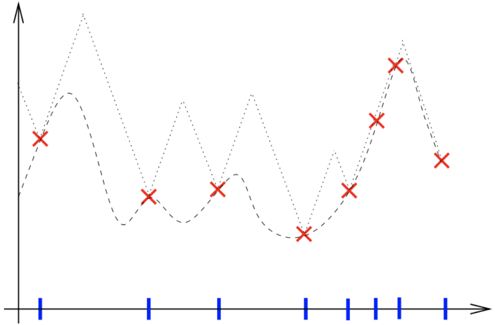
Adaptive Partitioning in Bandits



Evaluating f at a point x provides an upper-bound (f is Lipschitz)

* example from [Munos, 2013]

Adaptive Partitioning in Bandits

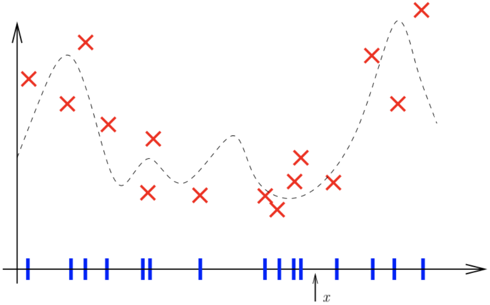


Refine upper-bound

What point to select? *Optimism*

* example from [Munos, 2013]

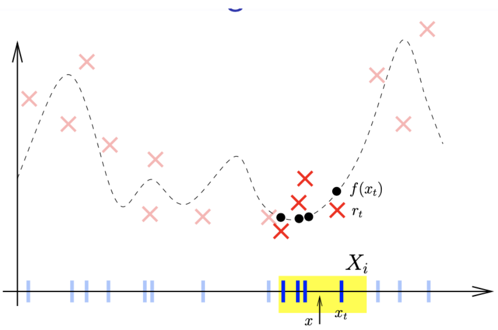
Adaptive Partitioning in Bandits



We have noisy observations. How to define high-probability upper-bound?

* example from [Munos, 2013]

Adaptive Partitioning in Bandits



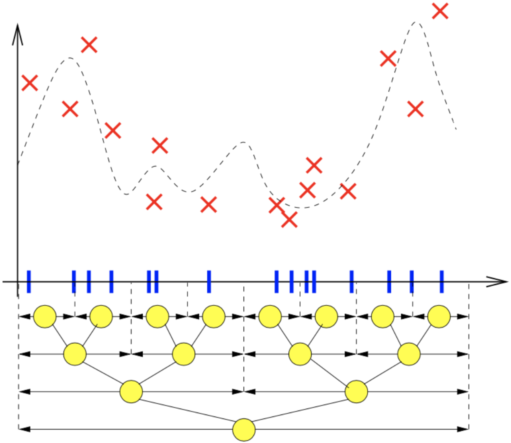
Fix a ball B_i (interval in 1D) containing n_i points $\{x_t\}$. Then, $\forall y \in B_i$

$$\frac{1}{n_i}\sum_{t=1}^{n_i} r_t + \sqrt{\frac{\log 1/\delta}{2n_i}} \geq \frac{1}{n_i}\sum_{t=1}^{n_i} f(x_t) \geq f(y) - diam(B_i)$$

since f is Lipschitz

* example from [Munos, 2013]

Adaptive Partitioning in Bandits



How to increase accuracy? Increase granularity over time (tree structure)
Split is a trade-off between confidence interval and ball radius

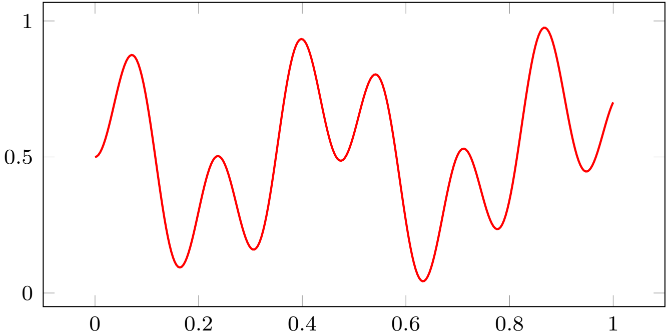
bias-variance trade-off

* example from [Munos, 2013]

Adaptive Partitioning: *Example*

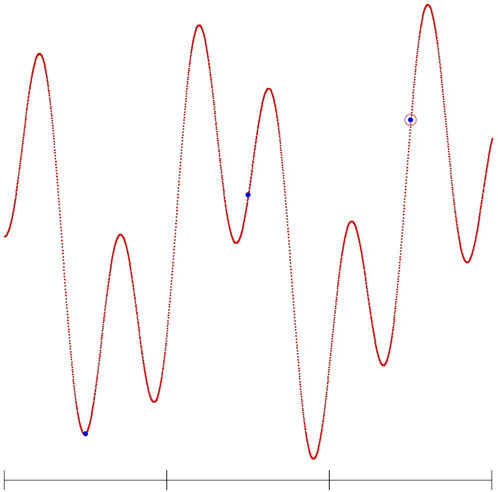
$f(x) = \frac{1}{2}(\sin(13x)\sin(27x) + 1)$ satisfies the local smoothness assumption $f(x) \geq f(x^\star) - l(x, x^\star)$ with

- $l_1(x,y) = 14|x-y|$ (i.e., f is globally Lipschitz)
- $l_2(x,y) = 222|x-y|^2$ (i.e., f is locally quadratic)



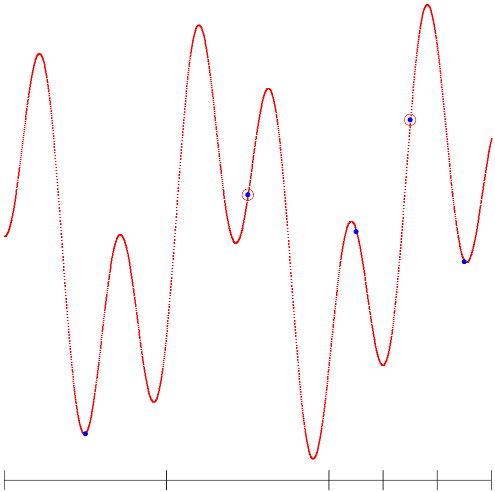
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



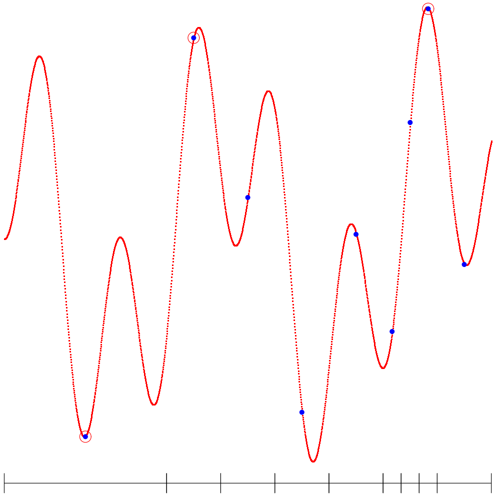
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



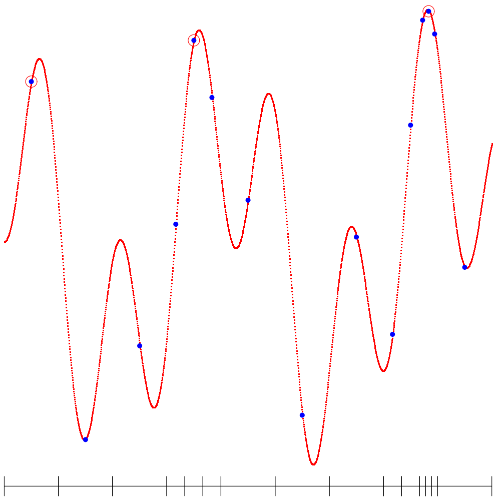
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



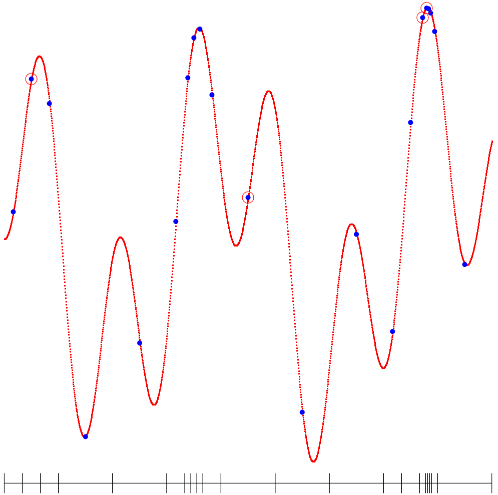
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



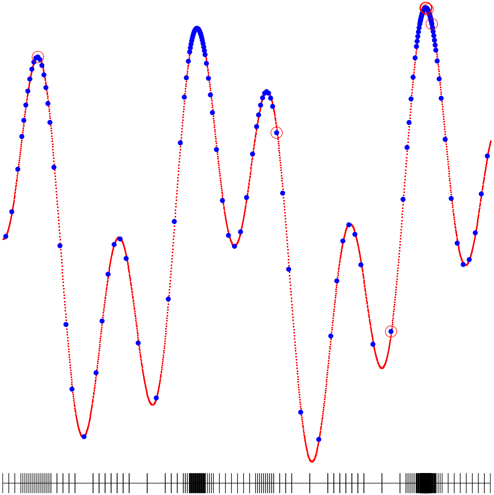
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



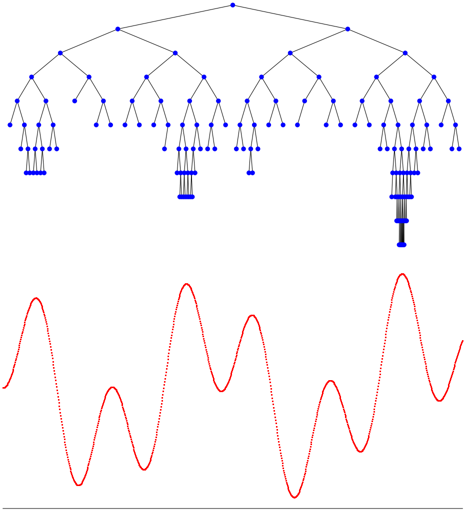
* example from [Munos, 2013]

Adaptive Partitioning: *Example*



* example from [Munos, 2013]

Adaptive Partitioning: *Example*



* example from [Munos, 2013]

Adaptive Optimistic Q-Learning (AdOpt-QL)

[Sinclair et al., 2019]

- Uses *hierachical covering* of state-action space
  - *Starts* from a *single partition* covering all the space
  - *Adapts* the granularity of the partition based on rewards and visits

$$\mathcal{P}_{hk} = \{B_i\} : \mathcal{S} \times \mathcal{A} \subseteq \bigcup_i B_i$$

- For each ball B we store
  - An optimistic estimate $Q_h(B)$ of $Q^\star$
  - A visit counter $N_h(B)$

AdOpt-QL

Input: $\mathcal{S}$, $\mathcal{A}$, $\cancel{r_h, p_h}$, L_{qh}

Initialize $Q_h(B) = H$ and $N_h(B) = 0$ for all $h = [H]$, with $B = \mathcal{S} \times \mathcal{A}$

for $k = 1, \ldots, K$ **do** `// episodes`

Observe initial state s_1 *(arbitrary)*

for $h = 1, \ldots, H$ **do**

Select region containing s_h: $B = \arg\max_{\overline{B} \in \mathsf{rel}_h(s_h)} Q_h(\overline{B})$

Execute *any action* a such that $(s_h, a) \in dom_h(B)$

Observe r_h and s_{h+1}

Set $N_h(B) = N_h(B) + 1$

Update

$$\widehat{Q}_h(B) = (1 - \alpha_t)\widehat{Q}_h(B) + \alpha_t \left(r_h + \widehat{V}_{h+1}(s_{h+1}) + b_t\right)$$

Set $\widehat{V}_h(s) = \min\left\{H, \max_{B' \in \mathsf{rel}_h(s)} \widehat{Q}_h(B')\right\}$

If $N_h(B) \geq g(B)$ **then** $\texttt{SplitBall}(B, h, k)$

end

end

As in Opt-QL

Refine state-action aggregation

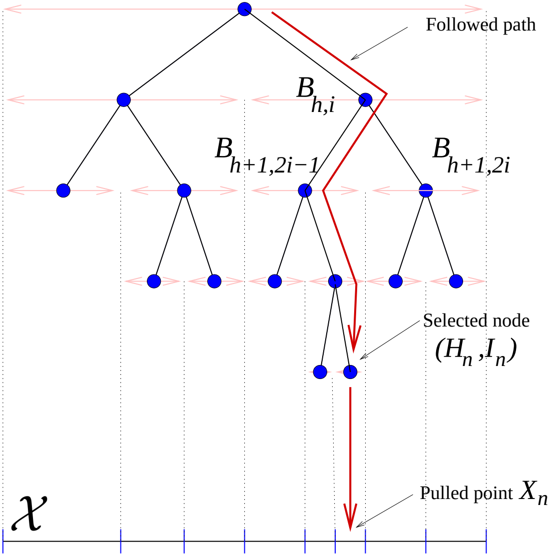
AdOpt-QL: Action Selection

- Traverse the hierachical structure based on Q and s_{hk}
- Optimistic selection of the ball

$$\color{blue}{B} = \arg\max_{\overline{B}} Q_h(\overline{B})$$

such that $s_{hk} \in B$

- Play random action in B



* figure from [Bubeck et al., 2011]

AdOpt-QL: Uncertainty

$\alpha_t = \dfrac{H+1}{H+t}$ as in [Jin et al., 2018]

$$Q_h^{k+1}(B) = (1 - \alpha_t)Q_{hk}(B) + \alpha_t(r_h + b_h(t) + V_{h+1}^k(s_{h+1}))$$

exploration bonus

$$b_h(t) = \underbrace{2\sqrt{\frac{H^3 \log(4HK/\delta)}{t}}}_{\textit{estimation error}} + \underbrace{\frac{d_{\max} L_{q,h}}{\sqrt{t}}}_{\textit{discretization error}}$$

- ! $t = N_{hk}(B) + 1$ i.e., number of visits
- ! $diam(\mathcal{S} \times \mathcal{A}) \leq d_{\max}$

AdOpt-QL: Refining the Partition

If $N_h^{k+1}(B) \geq \left(\frac{d_{\max}}{radi(B)}\right)^2$ **then**

- Split ball
- Cover $dom(B)$ using a $\frac{1}{2}radi(B)$-net

☞ when the number of samples is large, variance dominates the bias $\implies$ better to split

☞ new balls inherit properties of the parent ball

AdOpt-QL: Regret

Theorem (Thm. 4.1 by Sinclair et al. [2019])

For any smooth MDP with L_{qh}-Lipschitz Q-function and non-stationary transitions, AdOpt-QL, *with high probability, suffers a regret*

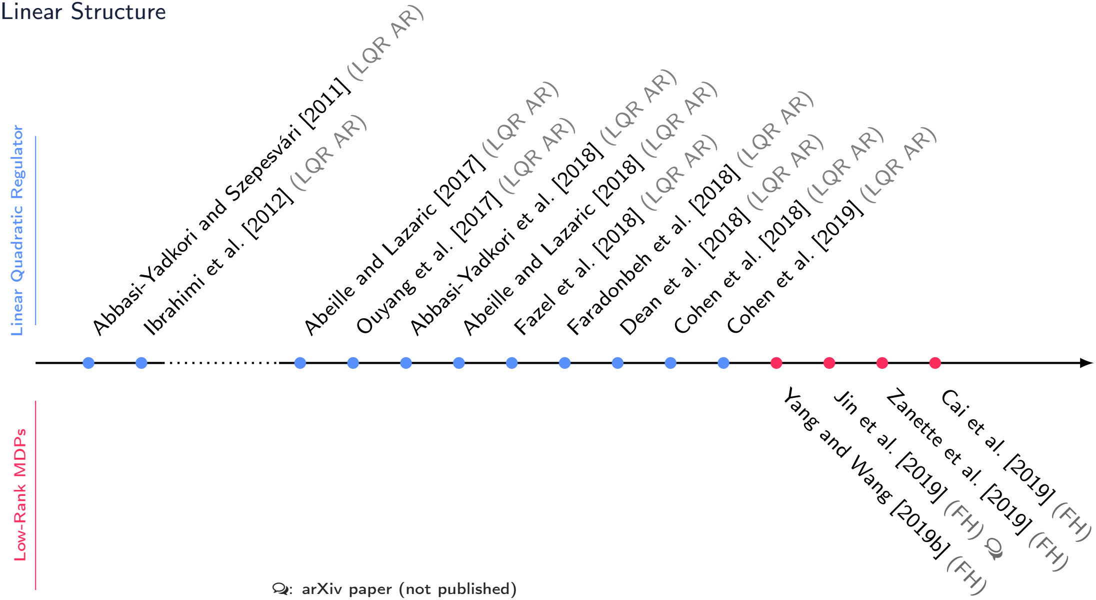
$$R(K, M^\star, \text{AdOpt-QL}) = \widetilde{\mathcal{O}}\left(c^{1/(d_c+2)}\ H^{5/2}\ K^{(d_c+1)/(d_c+2)}\right)$$

- Order optimal in c and K
- Factor H worse than the lower-bound

same bound for [Song and Sun, 2019] with uniform discretization

History: Regret Minimization

Linear Structure



Linear Function Approximation

Action-value functions

- Feature map $\phi_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$
- Function approximation $Q_h(s, a) = \phi_h(s, a)^\mathsf{T} \theta_h$

Least-Squares Value Iteration (LSVI)

Input: Dataset $\mathcal{D}_k = (s_{hi}, a_{hi}, r_{hi})_{h=1,i=1}^{H,k}$

Set $\theta_{H+1} = 0$ and $\widehat{Q}_{H+1}(s,a) = \phi_{H+1}(s,a)^\mathsf{T}\theta_{H+1}$

for $h = H, \dots, 1$ **do** `// backward induction`

Compute

Bootstrapping estimates

$$y_{hi} = r_{hi} + \max_{a \in \mathcal{A}} \widehat{Q}_{h+1,k}(s_{h+1,i}, a) = r_{hi} + \widehat{V}_{h+1,k}(s_{h+1,i}), \quad i = 1, \dots, k$$

Build regression dataset $\mathcal{D}_h^{\text{reg}} = \{\phi_h(s_{hi}, a_{hi}), y_{hi}\}_i$

Compute

$$\Sigma_{hk} = \sum_{i=1}^{k} \phi_h(s_{hi}, a_{hi})\phi_h(s_{hi}, a_{hi})^\mathsf{T} + \lambda I, \qquad \Omega_{hk} = \sum_{i=1}^{k} \phi_h(s_{hi}, a_{hi}) y_{hi}$$

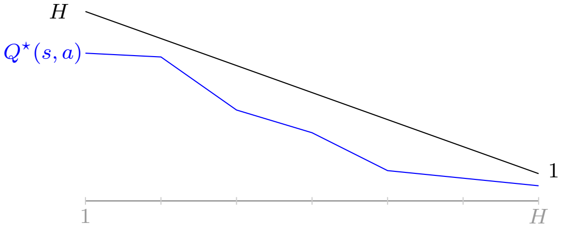
$$\widehat{\theta}_{hk} = \arg\min_{\theta} \frac{1}{k}\sum_{i=1}^{k}\left(y_{hi} - \phi_h(s_{hi}, a_{hi})^\mathsf{T}\theta\right)^2 + \lambda\|\theta\|_2^2$$

$$= \Sigma_{hk}^{-1}\,\Omega_{hk}$$

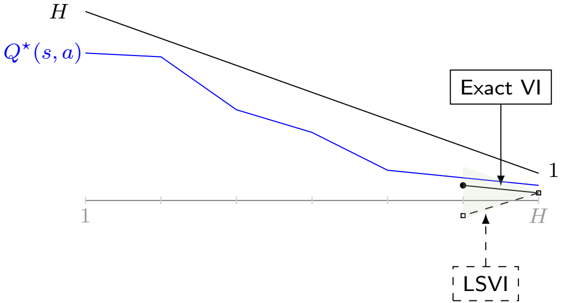
end

return $\{\widehat{\theta}_{hk}\}_{h=1}^{H}$

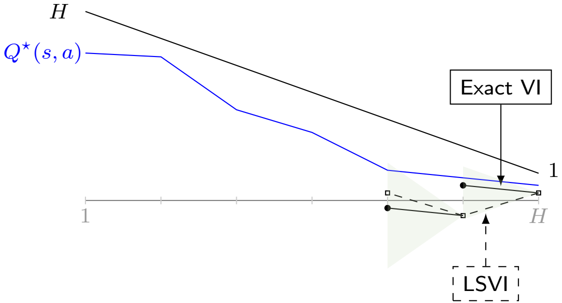
Least-Squares Value Iteration (LSVI)



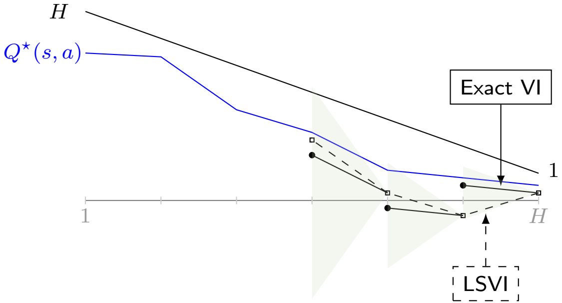
Least-Squares Value Iteration (LSVI)



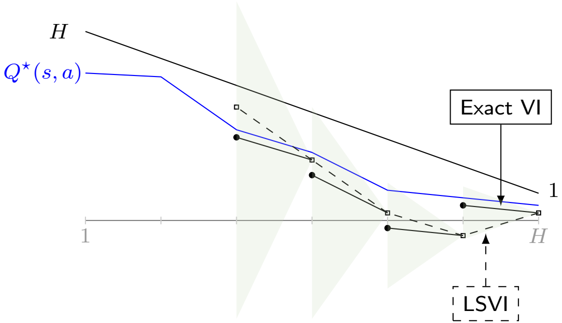
Least-Squares Value Iteration (LSVI)



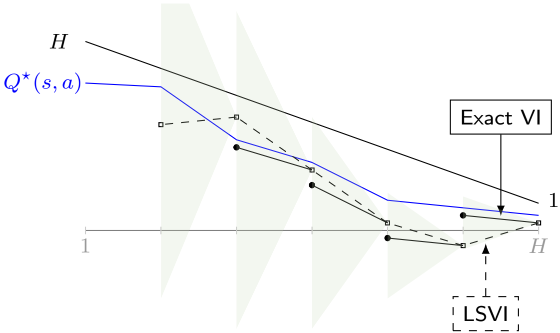
Least-Squares Value Iteration (LSVI)



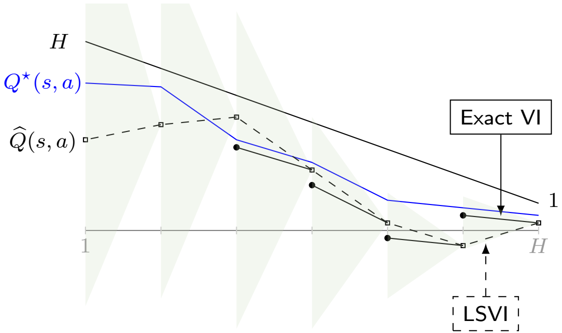
Least-Squares Value Iteration (LSVI)



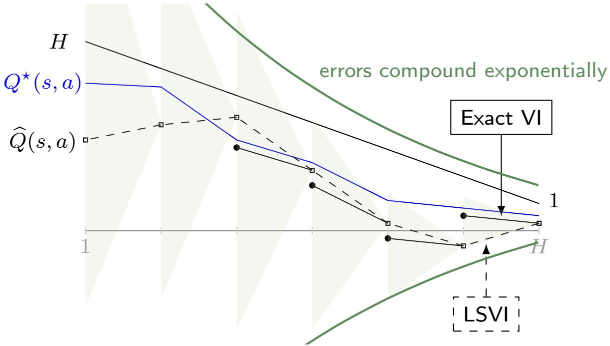
Least-Squares Value Iteration (LSVI)



Least-Squares Value Iteration (LSVI)



Least-Squares Value Iteration (LSVI)



Bad News

Theorem ([Du et al., 2019])

Let assume that ϕ_h approximates well the action-value function of any policy π

$$\min_\theta \|Q_h^\pi(\cdot) - \phi_h(\cdot)^\mathsf{T}\theta\|_\infty \leq \epsilon.$$

There exists an MDP such that any algorithm that returns a $1/2$-optimal policy with 0.9 probability requires

$$T \geq \Omega\Big(\min\{|\mathcal{S}|, 2^H, \exp(d\epsilon^2/16)\}\Big)$$

[Baird, 1995] counter-examples for LSVI-like algorithms

Not So Bad News

Theorem ([Lattimore and Szepesvári, 2019])

Let assume that ϕ_h approximates well the action-value function of any policy π

$$\min_{\theta} \|Q_h^{\pi}(\cdot) - \phi_h(\cdot)^{\mathsf{T}}\theta\|_{\infty} \leq \epsilon.$$

Approximate policy iteration using a generative model returns a $O(\epsilon\sqrt{d})$-optimal policy with

$$T \leq \widetilde{O}\Big(\frac{d}{\epsilon^2}\Big)$$

Not So Bad News

Theorem ([Lattimore and Szepesvári, 2019])

Let assume that ϕ_h approximates well the action-value function of any policy π

$$\min_\theta \|Q_h^\pi(\cdot) - \phi_h(\cdot)^\mathsf{T}\theta\|_\infty \leq \epsilon.$$

Approximate policy iteration using a generative model returns a $O(\epsilon\sqrt{d})$-optimal policy with

$$T \leq \widetilde{O}\Big(\frac{d}{\epsilon^2}\Big)$$

API vs LSVI, generative model vs RL

Some Good News

Low-Rank MDPs [Yang and Wang, 2019a]

A low-rank (linear) MDP $M = \langle \mathcal{S}, \mathcal{A}, \phi_h, r_h, p_h, H \rangle$

- $\mathcal{S} \times \mathcal{A}$ is a measurable space
- Dynamics is low rank, $\psi_h : \mathcal{S} \rightarrow \mathbb{R}^d$

$$p_h(s'|s,a) = \phi_h(s,a)^\mathsf{T} \, \psi_h(s')$$

- Reward has linear structure, $\theta_h^r \in \mathbb{R}^d$

$$r_h(s,a) = \phi_h(s,a)^\mathsf{T} \, \theta_h^r$$

examples are tabular MDPs, simplex feature space (e.g., mixture models)

This is a generalization of Linear Contextual Bandits [Lattimore and Szepesvári, 2018]

Some Good News

Low-Rank MDPs

For *every policy* $\pi = (\pi_1, \ldots, \pi_H)$ and $h \in [H]$, *Q_h^π is linear in ϕ_h*

$$\begin{aligned} Q_h^\pi(s,a) &= r_h(s,a) + \mathbb{E}_{s'|s,a}[V_{h+1}^\pi(s')] \\ &= \phi_h(s,a)^\mathsf{T}\theta_h^r + \int_{s'} \phi_h(s,a)^\mathsf{T}\psi_h(s')V_{h+1}^\pi(s')\mathrm{d}s' \\ &= \phi_h(s,a)^\mathsf{T} \underbrace{\left(\theta_h^r + \int_{s'} \psi_h(s')V_{h+1}^\pi(s')\mathrm{d}s'\right)}_{\textit{independent from } (s,a)} \\ &= \phi_h(s,a)^\mathsf{T}\theta_h^\pi \end{aligned}$$

⚠ *very strong structure!*

any function V_{h+1}^π is transformed into a linear function by the Bellman operator

Some Good News

Assumption: MDP is *approximately* low-rank w.r.t. features ϕ_h

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(H^2 d^{3/2}\sqrt{T})$ [Yang and Wang, 2019b]*
- *Randomization*: ?

Model-free

- *Optimism*: Opt-LSVI $\widetilde{\mathcal{O}}(H^{3/2} d^{3/2}\sqrt{T})$ [Jin et al., 2019]**
- *Randomization*: Opt-RLSVI $\widetilde{\mathcal{O}}(H^2 d^2\sqrt{T})$ [Zanette et al., 2019]***

*Depending on further (light) assumptions can be improved from $d^{3/2}$ to d
**If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon dHT)$
***If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon dHT(1+\epsilon dH^2)$

Some Good News

Assumption: MDP is *approximately* low-rank w.r.t. features ϕ_h

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(H^2 d^{3/2}\sqrt{T})$ [Yang and Wang, 2019b]*
- *Randomization*: **?**

Model-free

- *Optimism*: Opt-LSVI $\widetilde{\mathcal{O}}(H^{3/2} d^{3/2}\sqrt{T})$ [Jin et al., 2019]**
- *Randomization*: Opt-RLSVI $\widetilde{\mathcal{O}}(H^2 d^2\sqrt{T})$ [Zanette et al., 2019]***

continuous MDPs, approximate low-rank, model-free

*Depending on further (light) assumptions can be improved from $d^{3/2}$ to d
**If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon d H T)$
***If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon d H T (1 + \epsilon d H^2)$

Some Good News

Assumption: MDP is *approximately* low-rank w.r.t. features ϕ_h

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(H^2 d^{3/2}\sqrt{T})$ [Yang and Wang, 2019b]*
- *Randomization*: **?**

Model-free

- *Optimism*: Opt-LSVI $\widetilde{\mathcal{O}}(H^{3/2} d^{3/2}\sqrt{T})$ [Jin et al., 2019]**
- *Randomization*: Opt-RLSVI $\widetilde{\mathcal{O}}(H^2 d^2\sqrt{T})$ [Zanette et al., 2019]***

continuous MDPs, approximate low-rank, model-free

not "that" scalable, strong assumption (see later)

*Depending on further (light) assumptions can be improved from $d^{3/2}$ to d
**If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon dHT)$
***If the MDP is ϵ-low-rank, additional term $\widetilde{\mathcal{O}}(\epsilon dHT(1+\epsilon dH^2)$

Lower-Bound for Low Rank MDPs

Theorem ([Jin et al., 2019])

For any low-rank MDP $M^\star$, any algorithm $\mathfrak{A}$ suffers at least a regret

$$R(K, M^\star, \mathfrak{A}) = \Omega(d^{1/2} H \sqrt{T})$$

OptLSVI
[Jin et al., 2019]

Input: ϕ_h
Initialize $Q_{h1}(s,a) = 0$ for all $(s,a) \in \mathcal{S} \times \mathcal{A}$ and $h = 1, \dots, H$, $\mathcal{D}_1 = \emptyset$

for $k = 1, \dots, K$ **do** `// episodes`
    Observe initial state s_{1k} *(arbitrary)*
    Run LSVI with UCB on $\mathcal{D}_k$
    for $h = 1, \dots, H$ **do**
        Execute $a_{hk} = \pi_{hk}(s_{hk}) = \arg\max_a \widehat{Q}_{hk}(s_{hk}, a)$
        Observe r_{hk} and $s_{h+1,k}$
    end
    Add trajectory $(s_{hk}, a_{hk}, r_{hk})_{h=1}^{H}$ to $\mathcal{D}_{k+1}$
end

LSVI *with Upper-Confidence Bounds*

Input: Dataset $\mathcal{D}_k = (s_{hi}, a_{hi}, r_{hi})_{h=1,i=1}^{H,k}$
Set $\theta_{H+1} = 0$ and $\widehat{Q}_{H+1}(s,a) = \phi_{H+1}(s,a)^\mathsf{T}\theta_{H+1}$
for $h = H, \dots, 1$ **do** `// backward induction`

Compute

$$y_{hi} = r_{hi} + \widehat{V}_{h+1,k}(s_{h+1,i}), \quad i = 1, \dots, k$$

Build regression dataset $\mathcal{D}_h^{\text{reg}} = \{\phi_h(s_{hi}, a_{hi}), y_{hi}\}_i$
Compute

$$\Sigma_{hk} = \sum_{i=1}^{k} \phi_h(s_{hi}, a_{hi})\phi_h(s_{hi}, a_{hi})^\mathsf{T} + \lambda I, \qquad \Omega_{hk} = \sum_{i=1}^{k} \phi_h(s_{hi}, a_{hi}) y_{hi}$$

$$\widehat{\theta}_{hk} = \Sigma_{hk}^{-1}\Omega_{hk}$$

Add *uncertainty*

$$\widehat{Q}_{hk}(s,a) = \phi_h(s,a)^\mathsf{T}\widehat{\theta}_{hk} + b_{hk}(s,a)$$

Set $\widehat{V}_{hk}(s) = \min\left\{H, \max_{a\in\mathcal{A}} \widehat{Q}_{hk}(s,a)\right\}$

end
return $\{\widehat{\theta}_{hk}\}_{h=1}^{H}$

Measuring Uncertainties

Theorem (Lem. B.4-B.5 of [Jin et al., 2019])

Consider the filtration composed by the history generated by the algorithm at any point during its runtime. If $\|\phi_h(s,a)\|_2 \leq L_\phi$, $\|\theta_h^r\|_2 \leq L_r$ *and* $\int_s \|\psi_h(s)\|_2 \leq L_\psi$, *then with probability at least* $1-\delta$, *for all* (s,a,h,k), *we have*

$$\left|\phi_h(s,a)^\mathsf{T}\widehat{\theta}_h^k - Q_h^\star(s,a)\right| \leq \alpha_k \sqrt{\phi_h(s,a)^\mathsf{T}\Sigma_{hk}^{-1}\phi_h(s,a)} := b_{hk}(s,a)$$

where

$$\alpha_k \propto dH\sqrt{\log\left(\frac{dHkL_\phi L_\psi L_r \lambda}{\delta}\right)} + \sqrt{\lambda}L_\phi L_\theta$$

👉 $\|\phi_h(s,a)\|_{\Sigma_{hk}^{-1}}$ measures the correlation between $\phi_h(s,a)$ and the features observed so far

OptLSVI: Regret

Theorem

Let $\lambda = 1$, $L_\psi = L_r = \sqrt{d}$ and $L_\phi = 1$. For any ϵlow rank MDP M w.r.t. features ϕ_h, OptLSVI with $\alpha_k = \mathcal{O}(dH + \epsilon H\sqrt{dk})$, with high probability, suffers a regret

$$R(K, M^\star, OptLSVI) = \mathcal{O}\left(d^{3/2}H^{3/2}\sqrt{T} + \epsilon dHT\right)$$

- Order optimal $\sqrt{T}$
- Factor $d\sqrt{H}$ worse than the lower-bound
- Linear dependence in ϵ

👍 $\sqrt{H}$ might be saved by moving from Hoeffding to Bernstein bound see tabular RL [e.g., Azar et al., 2017]

👎 we don't know a Bernstein bound for the Least-Square estimator

Randomized Least-Squares Value Iteration (RLSVI)

[Osband et al., 2016]

Input: ϕ_h

Initialize $Q_{h1}(s,a) = 0$ for all $(s,a) \in \mathcal{S} \times \mathcal{A}$ and $h = 1, \ldots, H$, $\mathcal{D}_1 = \emptyset$

for $k = 1, \ldots, K$ **do** `// episodes`

  Observe initial state s_{1k} *(arbitrary)*

  Run RLSVI on $\mathcal{D}_k$

  for $h = 1, \ldots, H$ **do**

    Execute $a_{hk} = \pi_{hk}(s_{hk}) = \arg\max_a \widehat{Q}_{hk}(s_{hk}, a)$

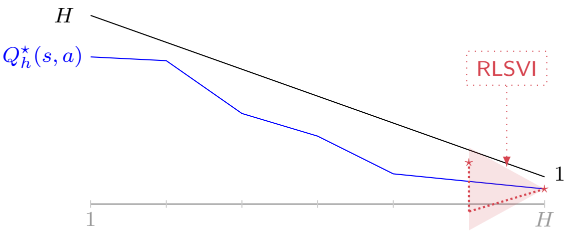
    Observe r_{hk} and $s_{h+1,k}$

  end

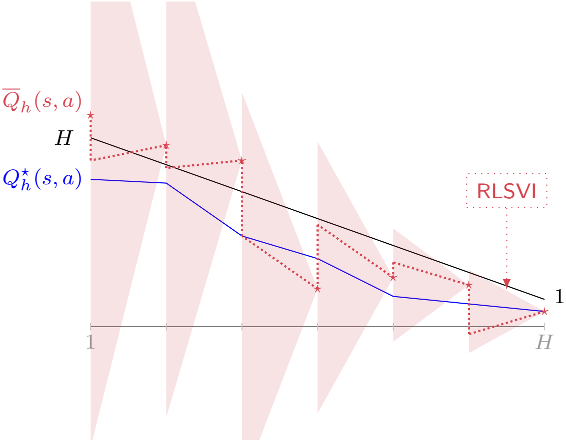
  Add trajectory $(s_{hk}, a_{hk}, r_{hk})_{h=1}^{H}$ to $\mathcal{D}_{k+1}$

end

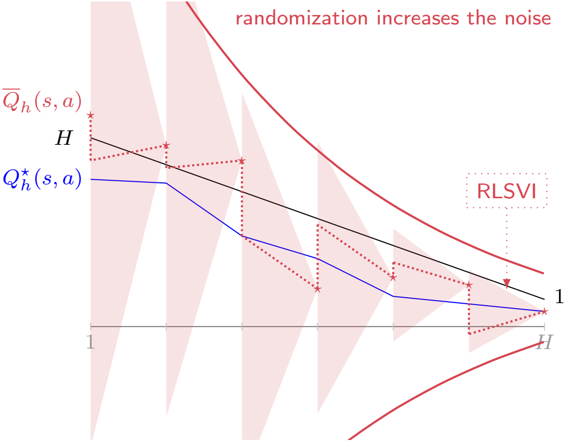
Randomized LSVI



Randomized LSVI



Randomized LSVI



Randomized Least-Squares Value Iteration (RLSVI)

Input: Dataset $\mathcal{D}_k = (s_{hi}, a_{hi}, r_{hi})_{h=1,i=1}^{H,k}$
Set $\theta_{H+1} = 0$ and $\overline{Q}_{H+1}(s,a) = \phi_{H+1}(s,a)^\mathsf{T}\theta_{H+1}$
for $h = H, \dots, 1$ **do** `// backward induction`

Compute

$$\overline{y}_{hi} = r_{hi} + \max_{a \in \mathcal{A}} \overline{Q}_{h+1,k}(s_{h+1,i}, a) = r_{hi} + \overline{V}_{h+1,k}(s_{h+1,i}) \ , \quad i = 1, \dots, k$$

Bootstrapping randomized estimates

Build regression dataset $\mathcal{D}_h^{\text{reg}} = \{\phi_h(s_{hi}, a_{hi}), \overline{y}_{hi}\}_i$
Compute

$$\widehat{\theta}_{hk} = \Sigma_{hk}^{-1}\overline{\Omega}_{hk}; \qquad \overline{\Omega}_{hk} = \sum_{i=1}^{k} \phi_h(s_{hi}, a_{hi})\overline{y}_{hi}$$

Sample $\xi_{hk} \sim \mathcal{N}(0, \sigma^2\Sigma_{hk}^{-1})$
Set $\overline{\theta}_{hk} = \widehat{\theta}_{hk} + \xi_{hk}$

end
return $\{\overline{\theta}_{hk}\}_{h=1}^{H}$

RLSVI as Regression on Perturbed Data

[Osband et al., 2019, Russo, 2019]

Bayesian Update

- True parameter is $\theta^\star \in \mathbb{R}^d$ $\Rightarrow$ we want to estimate it
- Assume *Gaussian prior* $\mathcal{N}(\overline{\theta}, \lambda I)$
- Dataset $\mathcal{D} = (x_i, y_i)_{i=1}^N$, where

$$y_i = x_i^\mathsf{T}\theta + \epsilon_i \quad , \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

- Conditional *posterior*

$$\theta^\star | \mathcal{D} \sim \underbrace{\mathcal{N}\left(\Sigma^{-1}\left(\frac{1}{\sigma^2}X^\mathsf{T}y + \frac{1}{\lambda}\overline{\theta}\right), \Sigma^{-1}\right)}_{:=\mu_p}$$

RLSVI as Regression on Perturbed Data

Perturbation
$\omega_i \sim \mathcal{N}(0, \sigma^2)$

- We can sample μ_p by fitting a least-squares estimate

$$\widehat{\theta} = \arg\min_{\theta} \frac{1}{\sigma^2} \sum_{i=1}^{N} \left(y_i + \omega_i - x_i^\mathsf{T}\theta\right) + \frac{1}{\lambda}\| \widetilde{\theta} - \theta\|_2^2$$

Sample from prior
$\widetilde{\theta} \sim \mathcal{N}(\overline{\theta}, \lambda I)$

$$\Rightarrow \widehat{\theta} \sim \mu_p$$

RLSVI as Regression on Perturbed Data

Perturbation
$\omega_i \sim \mathcal{N}(0, \sigma^2)$

- We can sample μ_p by fitting a least-squares estimate

$$\widehat{\theta} = \arg\min_{\theta} \frac{1}{\sigma^2} \sum_{i=1}^{N} \left(y_i + \omega_i - x_i^\mathsf{T} \theta \right) + \frac{1}{\lambda} \| \widetilde{\theta} - \theta \|_2^2$$

Sample from prior
$\widetilde{\theta} \sim \mathcal{N}(\overline{\theta}, \lambda I)$

$$\Rightarrow \widehat{\theta} \sim \mu_p$$

For linear models,

poster sampling = regularized least-squares on perturbed data

For tabular MDPs, $x_i = e_{s,a}$ and $\theta = Q$

backward induction on randomized rewards = RLSVI

Opt-RLSVI: Regret

Theorem (to appear at AISTATS)

For any ϵ-low rank MDP w.r.t. features ϕ_h, Mod-RLSVI with $\alpha = 1/(\sigma\sqrt{d})$ and $\sigma = O(\sqrt{Hd} + \epsilon H\sqrt{dk})$, with high probability, suffers a regret

$$R(K) = \widetilde{\mathcal{O}}\Big(H^2 d^2 \sqrt{T} + H^5 d^4 + \epsilon dHT(1 + \epsilon dH^2)\Big)$$

- Order optimal $\sqrt{T}$
- Long "warm-up" phase
- Factor $\sqrt{Hd}$ worse than OptLSVI [Jin et al., 2019]
- Linear regret depending on ϵ

Computationally Inefficient

Complexity of Opt-LSVI and Opt-RLSVI

- Space $\mathcal{O}(d^2 H + dAHK)$
- Time $\mathcal{O}(d^2 AHK^2)$

Move to *incremental* model-free
$\Rightarrow$ recursive least-squares

Issues:

1. Tracking a moving non-linear target
2. How to handle randomization

Pros and Cons

- complexity
- ...

Open Questions in Low-Rank MDPs

1. TODO

LQR

Continuous state-action space

⚠ Assumption: $\mathcal{S} \times \mathcal{A} \subseteq \mathbb{R}^d$, linear dynamics and quadratic reward

$$s_{h+1} = A_h s_h + B_h a_h + \epsilon_h$$
$$r_h(s,a) = s^\mathsf{T} Q_h s + a^\mathsf{T} R_h a$$

$\Rightarrow$ *Efficient* computation of $\pi^\star$

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Abbasi-Yadkori and Szepesvári, 2011, Cohen et al., 2018, Faradonbeh et al., 2018]
- *Randomization*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Ouyang et al., 2017, Abeille and Lazaric, 2018]*

Model-free ?

*Bayesian regret or 1-dimensional guarantees

LQR

Continuous state-action space

Assumption: $\mathcal{S} \times \mathcal{A} \subseteq \mathbb{R}^d$, linear dynamics and quadratic reward

$$s_{h+1} = A_h s_h + B_h a_h + \epsilon_h$$
$$r_h(s,a) = s^\mathsf{T} Q_h s + a^\mathsf{T} R_h a$$

$\Rightarrow$ *Efficient* computation of $\pi^\star$

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Abbasi-Yadkori and Szepesvári, 2011, Cohen et al., 2018, Faradonbeh et al., 2018]
- *Randomization*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Ouyang et al., 2017, Abeille and Lazaric, 2018]*

Model-free ?

*Bayesian regret or 1-dimensional guarantees

LQR

Continuous state-action space 👍

⚠ Assumption: $\mathcal{S} \times \mathcal{A} \subseteq \mathbb{R}^d$, linear dynamics and quadratic reward

$$s_{h+1} = A_h s_h + B_h a_h + \epsilon_h$$
$$r_h(s,a) = s^\mathsf{T} Q_h s + a^\mathsf{T} R_h a$$

⇒ *Efficient* computation of $\pi^\star$ 👍

Model-based

- *Optimism*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Abbasi-Yadkori and Szepesvári, 2011, Cohen et al., 2018, Faradonbeh et al., 2018]
- *Randomization*: $\widetilde{\mathcal{O}}(\sqrt{T})$ [Ouyang et al., 2017, Abeille and Lazaric, 2018]*

Model-free ?

👎 **exact, model-based, and strong assumption**

*Bayesian regret or 1-dimensional guarantees

Questions?

Website

`https://rlgammazero.github.io`

Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *COLT*, volume 19 of *JMLR Proceedings*, pages 1–26. JMLR.org, 2011.

Yasin Abbasi-Yadkori, Nevena Lazic, and Csaba Szepesvári. Regret bounds for model-free linear quadratic control. *CoRR*, abs/1804.06021, 2018.

Marc Abeille and Alessandro Lazaric. Thompson sampling for linear-quadratic control problems. In *AISTATS*, volume 54 of *Proceedings of Machine Learning Research*, pages 1246–1254. PMLR, 2017.

Marc Abeille and Alessandro Lazaric. Improved regret bounds for thompson sampling in linear quadratic control problems. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 1–9. PMLR, 2018.

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 263–272. PMLR, 2017.

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pages 30 – 37. Morgan Kaufmann, San Francisco (CA), 1995. ISBN 978-1-55860-377-6.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12(May):1655–1695, 2011.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. *CoRR*, abs/1912.05830, 2019.

Alon Cohen, Avinatan Hassidim, Tomer Koren, Nevena Lazic, Yishay Mansour, and Kunal Talwar. Online linear quadratic control. In *ICML*, volume 80 of *JMLR Workshop and Conference Proceedings*, pages 1028–1037. JMLR.org, 2018.

Alon Cohen, Tomer Koren, and Yishay Mansour. Learning linear-quadratic regulators efficiently with only $\sqrt{t}$ regret. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 1300–1309. PMLR, 2019.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. In *NeurIPS*, pages 4192–4201, 2018.

Simon S. Du, Sham M. Kakade, Ruosong Wang, and Lin F. Yang. Is a good representation sufficient for sample efficient reinforcement learning? *CoRR*, abs/1910.03016, 2019.

Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis. Input perturbations for adaptive regulation and learning. *CoRR*, abs/1811.04258, 2018.

Maryam Fazel, Rong Ge, Sham M. Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 1466–1475. PMLR, 2018.

Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

Morteza Ibrahimi, Adel Javanmard, and Benjamin Van Roy. Efficient reinforcement learning for high dimensional linear quadratic systems. In *NIPS*, pages 2645–2653, 2012.

Chi Jin, Zeyuan Allen-Zhu, Sébastien Bubeck, and Michael I. Jordan. Is q-learning provably efficient? In *NeurIPS*, pages 4868–4878, 2018.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I. Jordan. Provably efficient reinforcement learning with linear function approximation. *CoRR*, abs/1907.05388, 2019.

K. Lakshmanan, Ronald Ortner, and Daniil Ryabko. Improved regret bounds for undiscounted continuous reinforcement learning. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 524–532. JMLR.org, 2015.

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. Pre-publication version, 2018. URL `http://downloads.tor-lattimore.com/banditbook/book.pdf`.

Tor Lattimore and Csaba Szepesvári. Learning with good feature representations in bandits and in RL with a generative model. *CoRR*, abs/1911.07676, 2019.

Rémi Munos. Introduction to reinforcement learning and multi-armed bandits. NETADIS Summer School, 2013.

Chengzhuo Ni, Lin F. Yang, and Mengdi Wang. Learning to control in metric space with optimal regret. In *Allerton*, pages 726–733. IEEE, 2019.

Jungseul Ok, Alexandre Proutière, and Damianos Tranos. Exploration in structured reinforcement learning. In *NeurIPS*, pages 8888–8896, 2018.

Ronald Ortner and Daniil Ryabko. Online regret bounds for undiscounted continuous reinforcement learning. In *NIPS*, pages 1772–1780, 2012.

Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. In *ICML*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2377–2386. JMLR.org, 2016.

Ian Osband, Benjamin Van Roy, Daniel J. Russo, and Zheng Wen. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019. URL http://jmlr.org/papers/v20/18-339.html.

Yi Ouyang, Mukul Gagrani, and Rahul Jain. Control of unknown linear systems with thompson sampling. In *Allerton*, pages 1198–1205. IEEE, 2017.

Jian Qian, Ronan Fruit, Matteo Pirotta, and Alessandro Lazaric. Exploration bonus for regret minimization in discrete and continuous average reward mdps. In *NeurIPS*, pages 4891–4900, 2019.

Daniel Russo. Worst-case regret bounds for exploration via randomized value functions. In *NeurIPS*, pages 14410–14420, 2019.

Sean R. Sinclair, Siddhartha Banerjee, and Christina Lee Yu. Adaptive discretization for episodic reinforcement learning in metric spaces. *Proc. ACM Meas. Anal. Comput. Syst.*, 3(3), December 2019. doi: 10.1145/3366703. URL https://doi.org/10.1145/3366703.

Aleksandrs Slivkins. Contextual bandits with similarity information. *J. Mach. Learn. Res.*, 15(1):2533–2568, 2014.

Zhao Song and Wen Sun. Efficient model-free reinforcement learning in metric spaces. *CoRR*, abs/1905.00475, 2019.

Lin Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 6995–7004. PMLR, 2019a.

Lin F. Yang and Mengdi Wang. Reinforcement leaning in feature space: Matrix bandit, kernels, and regret bound. *CoRR*, abs/1905.10389, 2019b.

Andrea Zanette, David Brandfonbrener, Matteo Pirotta, and Alessandro Lazaric. Frequentist regret bounds for randomized least-squares value iteration. *CoRR*, abs/1911.00567, 2019.