LINMA2222: Stochastic optimal control and reinforcement learning

Part III: Stochastic systems

Guillaume Berger

November 28, 2025

Table of Contents

Table of Contents

Autonomous stochastic systems

System:

$$X(k+1) = F(X(k), N(k))$$

where $\{N(k)\}_{k=0}^{\infty}$ is i.i.d. (noise process), $X(k) \in \mathcal{X}$ (state space)

Solution: $\{X(k)\}_{k=0}^{\infty}$ is a stochastic process

Ergodicity: $\lim_{k\to\infty} p_{X(k)|X(0)} \to \pi \quad$ (steady-state measure)

Remark

We assume ergodicity throughout this course

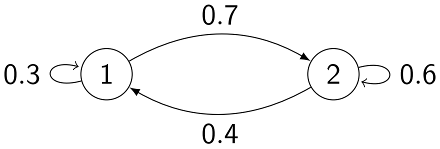
Examples

1) Linear system:

$$X(k+1) = FX(k) + N(k)$$

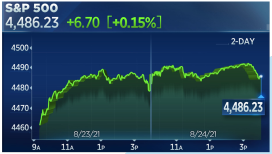
where $F \in \mathbb{R}^{n \times n}$, $\mathcal{X} = \mathbb{R}^n$, $N(k) \sim \mathcal{N}(0, \Sigma)$

2) Markov chain:

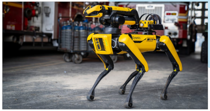


where $\mathcal{X} = \{1, 2\}$

Applications



Stochasticity
in
Systems and Control



Cost and value function – discounted case

Cost function: $c : \mathcal{X} \to \mathbb{R}_{\geq 0}$

Value function:

$$h(x) := \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k c(X(k)) \,\middle|\, X(0) = x\right]$$

(expected discounted cost) where $0 < \gamma < 1$

Course objective 1: approximate $h^\theta \approx h$

Remark

See Appendices for averaged case ($\gamma = 1$ but averaged over k)

Examples

1) $c(x) = x^\top Q x, \quad Q \succ 0$
Note: if $\gamma = 1$, $h(x) = \infty$

2) $c(1) = 0$, $c(2) = 1$
Note: if $\gamma = 1$, $h(x) = \infty$

Bellman equation

The value function satisfies the **Bellman equation**:

$$h(X(k)) = c(X(k)) + \gamma \mathbb{E}[h(X(k+1)) \mid X(k)]$$

equivalently

$$h(x) = c(x) + \gamma \mathbb{E}_N[h(F(x, N))] \quad \forall\, x \in \mathcal{X}$$

(Meyn, Eq. 9.7)

Examples

1) Let $h(x) = x^\top P x + q$. Bellman equation:

$$\begin{aligned}
&\Rightarrow\; x^\top P x + q = x^\top Q x + \gamma \mathbb{E}_N[(Fx+N)^\top P(Fx+N) + q] \\
&\Leftrightarrow\; x^\top P x + q = x^\top Q x + \gamma x^\top F^\top P F x + \gamma\,\mathrm{tr}(P\Sigma) + \gamma q \\
&\Leftrightarrow\; P = Q + \gamma F^\top P F \quad \text{and} \quad q = \frac{1}{1-\gamma}\,\mathrm{tr}(P\Sigma)
\end{aligned}$$

2) Bellman equation:

$$\begin{aligned}
h(1) &= 0 + \gamma 0.3 h(1) + \gamma 0.7 h(2) \\
h(2) &= 1 + \gamma 0.4 h(1) + \gamma 0.6 h(2)
\end{aligned}$$

E.g., with $\gamma = 0.9$, $h(1) \approx 5.78$ and $h(2) \approx 6.70$

Controlled stochastic systems

System:

$$X(k+1) = F(X(k), U(k), N(k))$$

where $\{N(k)\}_{k=0}^{\infty}$ is i.i.d. (noise process), $X(k) \in \mathcal{X}$ (state space), $U(k) \in \mathcal{U}$ (input space)

Policy: $U(k) = \phi(X(k))$ (deterministic) or $U(k) \sim \phi(\cdot \,|\, X(k))$ (randomized)

Closed-loop solution: Given a policy ϕ, $\{X(k)\}_{k=0}^{\infty}$ is a stochastic process

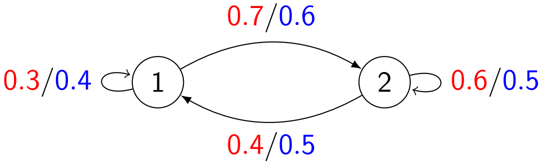
Examples

1) Linear system:

$$X(k+1) = FX(k) + GU(k) + N(k)$$

where $F \in \mathbb{R}^{n\times n}$, $G \in \mathbb{R}^{n\times m}$, $\mathcal{X} = \mathbb{R}^n, \mathcal{U} = \mathbb{R}^m$, $N(k) \sim \mathcal{N}(0, \Sigma)$

2) Markov decision process (MDP):



where $\mathcal{X} = \{1, 2\}, \mathcal{U} = \{\text{red}, \text{blue}\}$

Cost, value function and Q-function – discounted case

Cost function: $c : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_{\geq 0}$

1) Given a policy ϕ:

Value function:

$$h_\phi(x) := \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k c(X(k), U(k)) \,\Big|\, X(0) = x, \phi\right]$$

Q-function:

$$Q_\phi(x, u) := \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k c(X(k), U(k)) \,\Big|\, X(0) = x, U(0) = u, \phi\right]$$

where $0 < \gamma < 1$

Cost, value function and Q-function – discounted case

2) Optimal:

Value function:

$$h_\star(x) := \inf_{\phi} h_\phi(x)$$

Q-function:

$$Q_\star(x, u) := \inf_{\phi} Q_\phi(x, u)$$

where $0 < \gamma < 1$

Course objective 2: approximate $Q^\theta \approx Q_\phi$ or $Q^\theta \approx Q_\star$ or $\phi^\theta \approx \phi_\star$

Remark

See Appendices for averaged case ($\gamma = 1$ but averaged over k)

Bellman equations

The Q-functions satisfy the **Bellman equation**:

$$
\begin{aligned}
Q_\phi(X(k), U(k)) &= c(X(k), U(k)) + \\
&\qquad \gamma \mathbb{E}[Q_\phi(X(k+1), U(k+1)) \mid X(k), U(k), \phi] \\
Q_\star(X(k), U(k)) &= c(X(k), U(k)) + \\
&\qquad \gamma \mathbb{E}[\min_u Q_\star(X(k+1), u) \mid X(k), U(k)]
\end{aligned}
$$

(Meyn, Eq. 9.1)

Remark

Similar equations for h_ϕ and $h_\star$; omitted

Table of Contents

Problem statement

Objective: Find h^θ or Q^θ such that $h^\theta \approx h$ or $Q^\theta \approx Q_\phi$ or $Q^\theta \approx Q_\star$

Approximation space: $\mathcal{H} = \{h^\theta : \theta \in \mathbb{R}^d\}$ or $\mathcal{Q} = \{Q^\theta : \theta \in \mathbb{R}^d\}$

Most of the theory of this course:

Linear parametrizations:

- $h^\theta(x) = \theta^\top \psi(x)$ where $\psi : \mathcal{X} \to \mathbb{R}^d$
- $Q^\theta(x, u) = \theta^\top \psi(x, u)$ where $\psi : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^d$

Alternatives: kernel, neural networks, etc.

Approximation targets

Here, focus on autonomous systems (thus $h^\theta \approx h$)

1) Mean-square value error:

$$\theta^* = \arg\min_\theta \|h^\theta - h\|$$

Typically, $\|\cdot\| = \|\cdot\|_\pi$ defined by

$$\|e\|_\pi^2 = \mathbb{E}[e(X(k))^2 \mid X(k) \sim \pi]$$

2) Mean-square Bellman error:

$$B^\theta(X(k)) = -h^\theta(X(k)) + c(X(k)) + \gamma\mathbb{E}[h^\theta(X(k+1)) \mid X(k)]$$

(called Bellman error)

$$\theta^* = \arg\min_\theta \mathbb{E}[B^\theta(X(k))^2 \mid X(k) \sim \pi]$$

3) Mean-square temporal difference:

$$D^\theta(X(k), X(k+1)) = -h^\theta(X(k)) + c(X(k)) + \gamma h^\theta(X(k+1))$$

(called temporal difference)

$$\theta^* = \arg\min_\theta \mathbb{E}[D^\theta(X(k), X(k+1))^2 \mid X(k) \sim \pi]$$

4) Projected Bellman error:

Given $\{\zeta(k)\}_{k=0}^{\infty} \subseteq \mathbb{R}^d$ a stochastic process adapted to $\{X(k)\}_{k=0}^{\infty}$, find θ such that

$$\mathbb{E}[D^{\theta}(X(k), X(k+1))\zeta(k) \mid (X(k), \zeta(k)) \sim \tilde{\pi}] = 0$$

(called Galerkin approximation)

Example

$\{\zeta(k)\}_{k=0}^{\infty}$ given by

$$\zeta(k+1) = \tilde{F}(\zeta(k), X(k), N(k))$$

Table of Contents

Gradient methods

Idea: Minimize 2), 3) or 1) in *approximation targets* using SGD

Stochastic gradient descent (SGD):

To minimize $\mathbb{E}_\nu[f(\theta, \nu)]$ where ν is a random variable:

1. Compute g_k an unbiased estimator of $\nabla_\theta \mathbb{E}_\nu[f(\theta_k, \nu)]$
   E.g., sample ν and compute $g_k := \nabla_\theta f(\theta_k, \nu)$
2. Move in the direction of $-g_k$ (with stepsize α_k)

Theorem (Informal)

If stepsize sequence $\{\alpha_k\}_{k=0}^\infty$ appropriate (e.g., $\sum_k \alpha_k = \infty$ and $\sum_k \alpha_k^2 < \infty$), then convergence to stationary point

Table of Contents

Gradient Bellman error

Theorem (Meyn, Lemma 9.5)

The gradient of the mean-square Bellman error satisfies

$$
\begin{aligned}
\frac{1}{2}\nabla_\theta \mathbb{E}[B^\theta(X(k))^2 \,|\, X(k) \sim \pi] \\
&= \mathbb{E}[B^\theta(X(k)) \nabla_\theta B^\theta(X(k)) \,|\, X(k) \sim \pi] \\
&= \mathbb{E}[D^\theta(X(k), X(k+1)) \nabla_\theta B^\theta(X(k)) \,|\, X(k) \sim \pi]
\end{aligned}
$$

where

$$
\nabla_\theta B^\theta(X(k)) = \nabla_\theta h^\theta(X(k)) - \gamma \mathbb{E}[\nabla_\theta h^\theta(X(k+1)) \,|\, X(k)]
$$

Gradient Bellman error

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state
Then,

$$D^\theta(X(k), X(k+1))\nabla_\theta B^\theta(X(k))$$

is an unbiased estimator of $\frac{1}{2}\nabla_\theta \mathbb{E}[B^\theta(X(k))^2 \,|\, X(k) \sim \pi]$

Algorithm (Gradient-BE)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $g_k \leftarrow D^{\theta_k}(X(k), X(k+1))\nabla_\theta B^{\theta_k}(X(k))$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$

Return θ_k

Remark

For linear parametrizations, see LSBE in Appendices

Gradient Bellman error

Advantages:

- Conceptually simple
- Online

Limitations:

- Slow to learn
- Need double sampling to estimate

$$\nabla_\theta B^\theta(X(k)) = \nabla_\theta h^\theta(X(k)) - \gamma \mathbb{E}[\nabla_\theta h^\theta(X(k+1)) \mid X(k)]$$

- Not a good target (minimizer of MSBE is not always a useful approximation of the value function); see MSBE example in Appendices

(Sutton & Barto, Section 11.5)

Table of Contents

Gradient temporal difference

Theorem

The gradient of the mean-square temporal difference satisfies

$$\frac{1}{2}\nabla_\theta \mathbb{E}[D^\theta(X(k),X(k+1))^2 \,|\, X(k)\sim\pi] = \\ \mathbb{E}[D^\theta(X(k),X(k+1))\nabla_\theta D^\theta(X(k),X(k+1)) \,|\, X(k)\sim\pi]$$

and

$$\nabla_\theta D^\theta(X(k),X(k+1)) = \nabla_\theta h^\theta(X(k)) - \gamma\nabla_\theta h^\theta(X(k+1))$$

Gradient temporal difference

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state
Then,

$$D^{\theta}(X(k), X(k+1))\nabla_{\theta}D^{\theta}(X(k), X(k+1))$$

is an unbiased estimator of $\frac{1}{2}\nabla_{\theta}\mathbb{E}[D^{\theta}(X(k), X(k+1))^2 \,|\, X(k) \sim \pi]$

Algorithm (Gradient-TD)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $g_k \leftarrow D^{\theta_k}(X(k), X(k+1))\nabla_{\theta}D^{\theta_k}(X(k), X(k+1))$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$

Return θ_k

Remark

For linear parametrizations, see LSTD in Appendices

Gradient temporal difference

Advantages:

- Conceptually simple
- Online
- No need of double sampling (compared to gradient-BE)

Limitations:

- Slow to learn
- Not a good target (minimizer of MSTD is not always a useful approximation of the value function) (even more than MSBE); see MSTD example in Appendices

(Sutton & Barto, Section 11.5)

Table of Contents

Gradient value error

Theorem

For each k, let $\hat{h}(k)$ be an unbiased estimator of $h(X(k))$ (i.e., $\mathbb{E}[\hat{h}(k) \mid X(k)] = h(X(k))$). The gradient of the mean-square value error satisfies

$$\frac{1}{2}\nabla_\theta \mathbb{E}[\{h^\theta(X(k)) - h(X(k))\}^2 \mid X(k) \sim \pi] = \mathbb{E}[\{h^\theta(X(k)) - \hat{h}(k)\}\nabla_\theta h^\theta(X(k)) \mid X(k) \sim \pi].$$

Example

For each k, simulate $\{X'(k+\ell)\}_{\ell=0}^{T-1}$ from $X'(k) = X(k)$ with $\text{Geom}(1-\gamma)$ distribution for T and define

$$\hat{h}(k) \coloneqq \sum_{\ell=0}^{T-1} c(X'(k+\ell))$$

Gradient value error

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (Gradient-VE)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\hat{h}(k) \leftarrow$ unbiased estimator of $h(X(k))$
- $g_k \leftarrow \{h^{\theta_k}(X(k)) - \hat{h}(k)\}\nabla_\theta h^{\theta_k}(X(k))$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$

Return θ_k

Remark

For linear parametrizations, see LSVE in Appendices

Gradient value error

Advantages:

- Conceptually simple
- Converges to minimizer of value error

Limitations:

- Slow to learn
- Often not offline; difficult to have an unbiased estimator
- Unbiased estimator can have large variance

Table of Contents

Table of Contents

TD(0)

Idea: Use $\hat{h}(k) := c(X(k)) + \gamma h^{\theta_k}(X(k+1))$ as an estimator† of $h(X(k))$ and move in the direction

$$g_k := \{\hat{h}(k) - h^{\theta_k}(X(k))\}\nabla_\theta h^{\theta_k}(X(k))$$

which is the gradient of $-\frac{1}{2}(h^\theta(X(k)) - \hat{h}(k))^2$ at $\theta = \theta_k$

†Not an *unbiased* estimator!

Analysis: We will see that it zeroes the projected Bellman error with $\zeta(k) := \psi(X(k))$, for linear parametrizations

Remark

This is a form of **bootstrapping** because $h(X(k))$ is estimated from the current estimate h^θ – it is a *semi-gradient* method

TD(0)

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (TD(0))

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\delta_k \leftarrow c(X(k)) + \gamma h^{\theta_k}(X(k+1)) - h^{\theta_k}(X(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \nabla_\theta h^{\theta_k}(X(k))$

Return θ_k

TD(0) – linear parametrization

Assume linear parametrization: $h^\theta = \theta^\top \psi$
Note that $\nabla_\theta h^\theta = \psi$

Let $\{X(k)\}_{k=0}^\infty$ be in steady state

Algorithm (TD(0)-linear)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \dots,$ until stopping criterion is met:

- $A_k \leftarrow \psi(X(k))\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top$
- $b_k \leftarrow -\psi(X(k))c(X(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k(A_k\theta_k - b_k)$

Return θ_k

Soundness and convergence of TD(0)-linear

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Theorem (Meyn, Theorem 9.7(i))

The limit point θ^* of the TD(0)-linear algorithm satisfies

$$\mathbb{E}[D^{\theta^*}(X(k), X(k+1))\psi(X(k)) \,|\, X(k) \sim \pi] = 0$$

Theorem (Meyn, Theorem 9.8(i))

The matrix

$$A := \mathbb{E}[\psi(X(k))\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top \,|\, X(k) \sim \pi]$$

is Hurwitz. Hence, $\{\theta_k\}_{k=0}^\infty$ converges with probability one to $\theta^* = A^{-1}b$ where $b = \mathbb{E}[-\psi(X(k))c(X(k)) \,|\, X(k) \sim \pi]$

LSTD(0)

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T}$ be in steady state

Algorithm (LSTD(0))

For each $k = 0, 1, \ldots, T-1$:

- $A_k \leftarrow \psi(X(k))\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top$
- $b_k \leftarrow -\psi(X(k))c(X(k))$

$A \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1}b$

Soundness and convergence of TD(0)?

For nonlinear parameterizations (e.g., neural networks), the algorithm may be unstable and a fixed point may not even exist. Furthermore, if a fixed point exists, it has no more an interpretation as a Galerkin approximation (because the process $\{\zeta(k)\}_{k=0}^{\infty}$ depends on θ).

(Meyn, Section 9.4.2)

TD(0)

Advantages:

- Easy to implement
- Online
- Convergence for linear parametrization

Limitations:

- Can be too myopic, i.e., $c(X(k)) + h^{\theta_k}(X(k))$ can be biased
- Projected Bellman error may not be a good target (solution of PBE is not always a useful approximation of the value function)

Table of Contents

TD(λ)

Goal: Address myopia of TD(0)

Idea: Use

$$\hat{h}(k) := (1-\lambda)\sum_{T=1}^{\infty}\lambda^{T-1}\hat{h}_{k,T}$$

where

$$\hat{h}_{k,T} = \left(\sum_{\ell=0}^{T-1}\gamma^{\ell}c(X(k+\ell))\right) + \gamma^{T}h^{\theta_k}(X(k+T))$$

as an estimator $h(X(k))$ and move in the direction

$$g_k := \{\hat{h}(k) - h^{\theta_k}(X(k))\}\nabla_\theta h^{\theta_k}(X(k))$$

which is the gradient of $-\frac{1}{2}(h^{\theta}(X(k)) - \hat{h}(k))^2$ at $\theta = \theta_k$

TD(λ)

Need to look in the future (not online)
BUT if we "look backward", we obtain an approximation:

$$c(X(k)) \sum_{\ell=0}^{k} (\lambda\gamma)^{\ell} \nabla_\theta h^{\theta_{k-\ell}}(X(k-\ell))$$

and

$$\begin{aligned}(1-\lambda)\sum_{\ell=0} \lambda^{\ell}\gamma^{\ell+1} h^{\theta_{k-\ell}}(X(k+1)) \nabla_\theta h^{\theta_{k-\ell}}(X(k-\ell)) \\ \approx \gamma h^{\theta_k}(X(k+1)) \sum_{\ell=0} \lambda^{\ell}\gamma^{\ell} \nabla_\theta h^{\theta_{k-\ell}}(X(k-\ell)) - \\ h^{\theta_{k+1}}(X(k+1)) \sum_{\ell=1} \lambda^{\ell}\gamma^{\ell} \nabla_\theta h^{\theta_{k+1-\ell}}(X(k+1-\ell))\end{aligned}$$

TD(λ)

Hence, use

$$\tilde{g}_k := \{c(X(k)) + \gamma h^{\theta_k}(X(k+1)) - h^{\theta_k}(X(k))\} \cdot \sum_{\ell=0}^{k} (\lambda\gamma)^{\ell} \nabla_\theta h^{\theta_{k-\ell}}(X(k-\ell))$$

Analysis: For $\lambda = 0$, it corresponds to TD(0)
We will see that for $\lambda = 1$, it minimizes the value error (target 1) for linear parametrizations
For $0 < \lambda < 1$, it makes a trade-off between the two

TD(λ)

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (TD(λ))

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta h^{\theta_k}(X(k))$
- $\delta_k \leftarrow c(X(k)) + \gamma h^{\theta_k}(X(k+1)) - h^{\theta_k}(X(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

TD(λ) – linear parametrization

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Note that $\nabla_\theta h^\theta = \psi$

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (TD(λ)-linear)

$\theta_0 \leftarrow$ arbitrary

$\zeta(-1) \leftarrow 0$

For each $k = 0, 1, \dots,$ until stopping criterion is met:

- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \psi(X(k))$
- $A_k \leftarrow \zeta(k)\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top$
- $b_k \leftarrow -\zeta(k)c(X(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k(A_k\theta_k - b_k)$

Return θ_k

Soundness and convergence of TD(λ)-linear

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Theorem (Meyn, Theorem 9.7(ii))

The limit point θ^* of the TD(1)-linear algorithm satisfies

$$\theta^* = \arg\min_\theta \| h^\theta - h \|_\pi$$

Theorem (Meyn, Theorem 9.8(i))

For all $\lambda \in [0,1]$, the matrix

$$A := \mathbb{E}[\zeta(k)\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top \,|\, X(k) \sim \pi]$$

is Hurwitz. Hence, $\{\theta_k\}_{k=0}^\infty$ converges with probability one to $\theta^* = A^{-1}b$ where $b = \mathbb{E}[-\zeta(k)c(X(k)) \,|\, X(k) \sim \pi]$

LSTD(λ)

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T}$ be in steady state

Algorithm (LSTD(λ))

$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots, T-1$:

- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \psi(X(k))$
- $A_k \leftarrow \zeta(k)\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top$
- $b_k \leftarrow -\zeta(k)c(X(k))$

$A \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1}b$

Soundness and convergence of TD(λ)?

For nonlinear parameterizations (e.g., neural networks), the algorithm may be unstable and a fixed point may not even exist. Furthermore, if a fixed point exists, it has no more an interpretation as a Galerkin approximation (because the process $\{\zeta(k)\}_{k=0}^{\infty}$ depends on θ).

(Meyn, Section 9.4.2)

TD(λ)

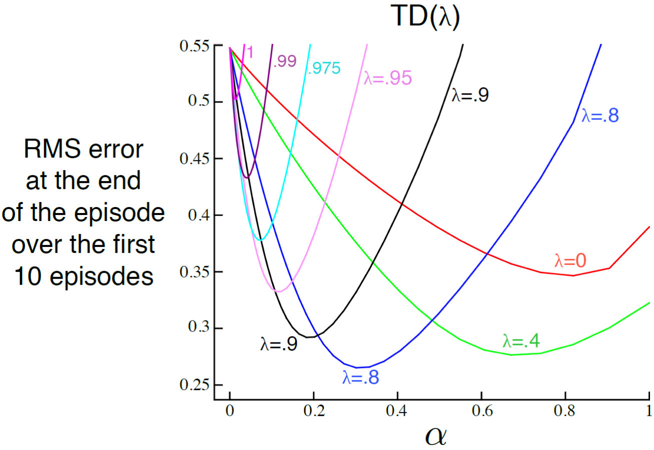
Advantages:

- Easy to implement
- Online
- Convergence for linear parametrization

Limitations:

- TD(1) can have large variance (can be better to use $0 < \lambda < 1$)

TD(λ)



TD(λ)
RMS error
at the end
of the episode
over the first
10 episodes
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0
0.2
0.4
0.6
0.8
1
α
1
.99
.975
λ=.95
λ=.9
λ=.8
λ=0
λ=.4
λ=.9
λ=.8

Table of Contents

Policy improvement (PI)

Idea: Given a policy ϕ, learn $Q^\theta \approx Q_\phi$ (e.g., using TD(λ)), then update the policy as $\phi_{\mathrm{new}}(x) \coloneqq \arg\min_u Q^\theta(x, u)$ ($+$ noise?)

Remark

Without approximation (i.e., if $Q^\theta = Q_\phi$), this guarantees to provide a better policy. However, with approximation this is not guaranteed anymore. (Sutton & Barto, Section 10.4)

TD(λ) for Q-function

Observation: Given ϕ, the system

$$\begin{cases} X(k+1) &= F(X(k), U(k), N(k)) \\ U(k+1) &= \phi(X(k), N(k)) \end{cases} \tag{1}$$

is autonomous
Moreover, Q_ϕ is the value function of (1)

Hence, we can apply TD(λ) on (1) to learn Q_ϕ

TD(λ) for Q-function

Let $\{X(k), U(k)\}_{k=0}^{\infty}$ from (1) be in steady state

Algorithm (TD(λ) for Q-function)

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta Q^{\theta_k}(X(k), U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) + \gamma Q^{\theta_k}(X(k+1), U(k+1)) - Q^{\theta_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

The soundness and convergence results still hold

PI-TD(λ)

Algorithm (PI-TD(λ))

$\phi_0 \leftarrow$ arbitrary policy (preferably stable, etc.)
For each episode $T = 0, 1, \ldots,$ until stopping criterion is met:

- Let $\{X(k), U(k)\}_{k=0}^{\infty}$ from (1) with ϕ_k be in steady state
- $Q^\theta \approx Q_{\phi_k}$ from TD(λ) for Q-function
- $\phi_{k+1}(x) \leftarrow \arg\min_u Q^\theta$ (+ exploration noise?)

Return ϕ_k

Use exploration noise (e.g., ϵ-greedy) to ensure all pairs (x, u) have a nonzero probability of being visited

Question: Is it necessary to learn $Q^\theta \approx Q_{\phi_k}$ precisely (knowing that ϕ_k will be updated anyway)? No $\rightarrow$ SARSA(λ)

Soundness and convergence of PI-TD(λ)

Results on the soundness and convergence of PI-TD(λ) are scarce. Indeed, with function approximation the policy improvement theorem is *not* satisfied. The algorithm may chatter among good policies rather than converge.

(Sutton & Barto, Section 10.4)

Table of Contents

SARSA(λ)

Idea: Instead of learning $Q^\theta \approx Q_{\phi_k}$ precisely in the PI-TD(λ) algorithm, just do one step of TD(λ):

- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta Q^{\theta_k}(X(k), U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) + \gamma Q^{\theta_k}(X(k+1), U(k+1)) - Q^{\theta_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Remark

Terminology: state – action – reward – state – action

SARSA(λ)

Algorithm (SARSA(λ))

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
$X(0), U(0) \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $X(k+1) \leftarrow F(X(k), U(k), N(k))$
- $U(k+1) \leftarrow \arg\min_u Q^{\theta_k}(X(k+1), u)$ (+ exploration noise?)
- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta Q^{\theta_k}(X(k), U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) + \gamma Q^{\theta_k}(X(k+1), U(k+1)) - Q^{\theta_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

Soundness and convergence of SARSA(λ)

Results on the soundness and convergence of SARSA(λ) are scarce. Even for linear parametrizations, it is known to have a *chattering behavior* (Gordon, 2000).

Remark

See also comments for PI-TD(λ)

SARSA(λ)

Advantages:

- Easy to implement (except policy update)
- Online

Limitations:

- Difficult to converge to the optimal policy because of exploration noise
- Need minimization in policy update (argmin)

Table of Contents

Off-policy TD(λ) for Q-function

Goal: Estimate Q_ϕ off-policy, i.e., with data generated from a *behavioral* (or *exploration*) policy $\phi_{\exp} \neq \phi$

Reminder: Bellman equation for Q_ϕ:

$$Q_\phi(X(k), U(k)) = c(X(k), U(k)) + \gamma \mathbb{E}[Q_\phi(X(k+1), U(k+1)) \mid X(k), U(k), \phi]$$

Hence, use temporal difference

$$D^\theta(X(k), U(k), X(k+1), \tilde{U}(k+1)) := c(X(k), U(k)) + \gamma Q^\theta(X(k+1), \tilde{U}(k+1)) - Q^\theta(X(k), U(k))$$

where $\tilde{U}(k+1) \sim \phi(\cdot \mid X(k+1))$

Off-policy TD(λ) for Q-function

Let $\{X(k), U(k)\}_{k=0}^{\infty}$ be in steady state from a (randomized) exploration policy $\phi_{\exp}$

Algorithm (Off-policy TD(λ) for Q-function)

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\tilde{U}(k+1) \leftarrow \phi(\cdot \,|\, X(k+1))$
- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta Q^{\theta_k}(X(k), U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) +$
  $\qquad \gamma Q^{\theta_k}(X(k+1), \tilde{U}(k+1)) - Q^{\theta_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

Soundness and convergence of off-policy TD(λ) for Q-function

There are counterexamples to soundness and convergence

(Sutton & Barto, Section 11.2)

Q(λ)-learning

Goal: Estimate $Q_\star$ off-policy, i.e., with data generated from a *behavioral* (or *exploration*) policy $\phi_{\rm exp} \neq \phi$

Reminder: Bellman equation for $Q_\star$:

$$Q_\phi(X(k), U(k)) = c(X(k), U(k)) + \\ \gamma \mathbb{E}[\min_u Q_\phi(X(k+1), u) \mid X(k), U(k), \phi]$$

Hence, use temporal difference

$$D^\theta(X(k), U(k), X(k+1)) \coloneqq c(X(k), U(k)) + \\ \gamma \min_u Q^\theta(X(k+1), u) - Q^\theta(X(k), U(k))$$

Q(λ)-learning

Let $\{X(k), U(k)\}_{k=0}^{\infty}$ be in steady state from a (randomized) exploration policy ϕ_{exp}

Algorithm (Q(λ)-learning)

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\tilde{U}(k+1) \leftarrow \arg\min_u Q^{\theta_k}(X(k+1), u)$
- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta Q^{\theta_k}(X(k), U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) + \gamma Q^{\theta_k}(X(k+1), \tilde{U}(k+1)) - Q^{\theta_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

Soundness and convergence of Q(λ)-learning

There are counterexamples to soundness and convergence

(Meyn, Section 9.11)

Q(λ)-learning

Advantages:

- Easy to implement (except policy update)
- Online
- Convergence in the tabular case (no approximation)

Limitations:

- Convergence in general can be difficult to obtain
- Need minimization in policy update (argmin)

Table of Contents

Parametrized policy

Idea: Use a parametrization of the policy
$\neq$ previous methods where only value/Q-function is parametrized and the policy is derived from it

Example

Deterministic:

- $\phi^\theta(x) = F_\theta x$ where $F_\theta \in \mathbb{R}^{n \times m}$
- $\phi^\theta(x) = \mathrm{NN}_\theta(x)$ where NN_θ is a feedforward neural network with weights and biases given by θ

Randomized:

- $\phi^\theta(\cdot \,|\, x) \sim \mathcal{N}(F_\theta x, \Sigma_\theta)$ where $F_\theta \in \mathbb{R}^{n \times m}$ and $\Sigma_\theta \in \mathbb{R}^{m \times m}$
- $\phi^\theta(u \,|\, x) = e^{h_\theta(u,x)} / \sum_{v \in \mathcal{U}} e^{h_\theta(u,x)}$

Parametrized policy

Advantages:

- Easier to represent randomized policies (sometimes needed when value/Q-function approximation is coarse)
- Inject prior knowledge/requirement about the policy (sometimes a simple policy is preferable)
- No need to do a minimization to derive policy from value/Q-function

(Sutton & Barto, Section 13.1)

Policy gradient theorem

Setting: Given θ, let

- π_θ be the steady-state distribution of the closed-loop system with policy ϕ^θ
- objective: minimize

$$J(\theta) := \mathbb{E}[c(X(k), U(k)) \,|\, X(k) \sim \pi_\theta, \phi^\theta]$$

  (averaged expected cost)
- Q_{ϕ^θ} be the associated relative Q-function (see Appendices)

Policy gradient theorem

Theorem

Assume that $\mathcal{U}$ is finite. It holds that

$$\nabla_\theta J(\theta) = \mathbb{E}\big[\{\nabla_\theta \log \mathbb{P}[U(k) \,|\, X(k), \phi^\theta]\} \cdot \\ Q_{\phi^\theta}(X(k), U(k)) \,\big|\, X(k) \sim \pi_\theta, \phi^\theta\big].$$

Remark

If $\mathcal{U} = \mathbb{R}^m$, replace $\mathbb{P}[U(k) \,|\, X(k), \phi^\theta]$ by probability density function $p_{U(k)|X(k),\phi^\theta}$

Table of Contents

REINFORCE

Corollary

For each k, let $\hat{Q}(k)$ be an unbiased estimator of $Q_{\phi^\theta}(X(k), U(k))$ (i.e., $\mathbb{E}[\hat{Q}(k) \,|\, X(k), U(k)] = Q_{\phi^\theta}(X(k), U(k))$). The gradient of $J(\theta)$ satisfies

$$\nabla_\theta J(\theta) = \mathbb{E}\big[\{\nabla_\theta \log \mathbb{P}[U(k) \,|\, X(k), \phi^\theta]\} \cdot \hat{Q}(k) \,\big|\, X(k) \sim \pi_\theta, \phi^\theta\big].$$

REINFORCE

Algorithm (REINFORCE)

$\theta_0 \leftarrow$ arbitrary
$X(0) \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $U(k) \leftarrow \phi^{\theta_k}(\cdot \mid X(k))$
- $\hat{Q}(k) \leftarrow$ unbiased estimator of $Q_{\phi^{\theta_k}}(X(k), U(k))$
- $g_k \leftarrow \{\nabla_\theta \log \mathbb{P}[U(k) \mid X(k), \phi^{\theta_k}]\} \hat{Q}(k)$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$
- $X(k+1) \leftarrow F(X(k), U(k), N(k))$

Return θ_k

REINFORCE with baseline

Let $b : \mathcal{X} \to \mathbb{R}$ be a baseline function.

Corollary

For each k, let $\hat{Q}(k)$ be an unbiased estimator of $Q_{\phi^\theta}(X(k), U(k))$ (i.e., $\mathbb{E}[\hat{Q}(k) \,|\, X(k), U(k)] = Q_{\phi^\theta}(X(k), U(k))$). The gradient of $J(\theta)$ satisfies

$$\nabla_\theta J(\theta) = \mathbb{E}\big[\{\nabla_\theta \log \mathbb{P}[U(k) \,|\, X(k), \phi^\theta]\} \cdot \{\hat{Q}(k) - b(X(k))\} \,\big|\, X(k) \sim \pi_\theta, \phi^\theta\big].$$

Example

Take $b = h^\omega$ where $h^\omega \approx h_{\phi^\theta}$, which minimizes the variance of $\hat{Q}(k) - b(X(k))$ since $h_{\phi^\theta}(X(k)) = \mathbb{E}[Q_{\phi^\theta}(X(k), U(k)) \,|\, X(k), \phi^\theta]$

REINFORCE with baseline

Algorithm (REINFORCE with baseline $b = h^{\omega}$)

$\theta_0, \omega_0 \leftarrow$ arbitrary
$X(0) \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $U(k) \leftarrow \phi^{\theta_k}(\cdot \,|\, X(k))$
- $\hat{Q}(k) \leftarrow$ unbiased estimator of $Q_{\phi^{\theta_k}}(X(k), U(k))$
- $g_k \leftarrow \{\nabla_\theta \log \mathbb{P}[U(k) \,|\, X(k), \phi^{\theta_k}]\}\{\hat{Q}(k) - h^{\omega_k}(X(k))\}$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$
- $\omega_{k+1} \leftarrow \omega_k + \beta_k\{\hat{Q}(k) - h^{\omega_k}(X(k))\}\nabla_\omega h^{\omega_k}(X(k))$
- $X(k+1) \leftarrow F(X(k), U(k), N(k))$

Return θ_k

Convergence of REINFORCE (with baseline)

The REINFORCE (with baseline) algorithm implements a stochastic gradient descent. Hence, it converges to a stationary point under the classical assumptions of SGD.

Table of Contents

Actor–critic method

Idea: Use $\hat{Q}(k) := Q^{\omega_k}(X(k), U(k))$ as an estimator† of $Q_{\phi^{\theta_k}}(X(k), U(k))$ and move in the direction

$$g_k := \nabla_\theta \log \mathbb{P}[U(k) \mid X(k), \phi^{\theta_k}]\}\{\hat{Q}(k) - b(X(k))\}$$

†Not an *unbiased* estimator!

Analysis: We will see that under some conditions on Q^ω, g_k provides an unbiased estimator of the gradient of $J(\theta)$

Remark

Use TD(λ) or any other technique to learn $Q^{\omega^k} \approx Q_{\phi^{\theta_k}}$

Actor–critic method

Algorithm (Actor–critic method with TD(λ) to learn Q^{ω})

$\theta_0, \omega_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
$X(0), U(0) \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $g_k \leftarrow \{\nabla_\theta \log \mathbb{P}[U(k) \,|\, X(k), \phi^{\theta_k}]\} Q^{\omega_k}(X(k), U(k))$
- $\theta_{k+1} \leftarrow \theta_k - \alpha_k g_k$
- $X(k+1) \leftarrow F(X(k), U(k), N(k))$
- $U(k+1) \leftarrow \phi^{\theta_{k+1}}(\cdot \,|\, X(k))$
- $\zeta(k) \leftarrow \lambda \zeta(k-1) + \nabla_\omega Q^{\omega_k}(X(k) U(k))$
- $\delta_k \leftarrow c(X(k), U(k)) + Q^{\omega_k}(X(k+1), U(k+1)) - Q^{\omega_k}(X(k), U(k))$
- $\omega_{k+1} \leftarrow \omega_k + \beta_k \delta_k \zeta(k)$

Return θ_k

Actor–critic method

Remark

The previous algorithm is presented without baseline, but a baseline (like $b = h^{\tau}$) can be used. In this case, another function (like h^{τ}) may be needed to learn.

Convergence of actor–critic method

Theorem (Meyn, Proposition 10.17)

Assume that Q^{ω} is linearly parametrized, i.e., $Q^{\omega} = \omega^{\top}\psi$. Also assume that for each θ, there is ω_{θ} such that $Q^{\omega_{\theta}} = Q_{\phi^{\theta}}$ (no approximation error on the Q-function). Assume that $\lim_{k\to\infty} \beta_k/\alpha_k = \infty$. Then, the actor-critic algorithm implements a stochastic gradient descent on ϕ^{θ} w.r.t. $J(\theta)$.

Convergence of actor–critic method

Relaxing the consistency assumption:

CFP (Compatible Feature Property): For each i, there is ω such that

$$\frac{\partial}{\partial \theta_i} \log \mathbb{P}[U(k) = u \,|\, X(k) = x, \phi^\theta] = \omega^\top \psi(x, u)$$

Theorem (Meyn, Proposition 10.19)

Assume that Q^ω is linearly parametrized, i.e., $Q^\omega = \omega^\top \psi$. Also assume that the CFP holds. Assume that $\lim_{k\to\infty} \beta_k/\alpha_k = \infty$. Then, the actor-critic algorithm with TD(1) to learn Q^ω implements a stochastic gradient descent on ϕ^θ w.r.t. $J(\theta)$.

Table of Contents

Stochastic approximation

Theorem (Meyn, Theorem 8.1)

Let $M(\theta)$ have a unique root at θ^*. Assume we can obtain measurements of the r.v. $N(\theta)$ where $\mathbb{E}[N(\theta)] = M(\theta)$. Consider the iteration

$$\theta_{k+1} = \theta_k - \alpha_k N(\theta_k),$$

where $\{a_k\}_{k=0}^{\infty}$ is a sequence of positive step sizes. It holds that $\{\theta_k\}_{k=0}^{\infty}$ converges in L^2 and with probability one to θ^*, if

- *$N(\theta)$ is uniformly bounded;*
- *$M(\theta)$ is Lipschitz continuous;*
- *$\dot{\theta} = M(\theta)$ is GAS;*
- *the sequence $\{a_k\}_{k=0}^{\infty}$ satisfies*

$$\sum_{k=0}^{\infty} \alpha_k = \infty \quad \textit{and} \quad \sum_{k=0}^{\infty} \alpha_k^2 < \infty.$$

LSBE

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T}$ be in steady state

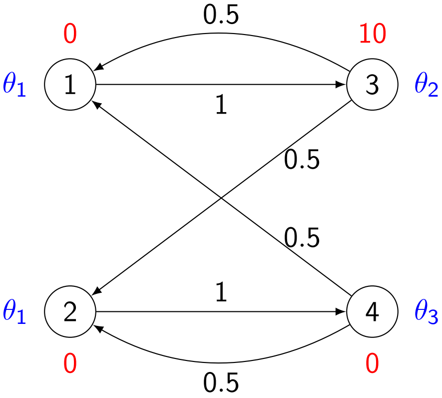
Algorithm (LSBE)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\Upsilon_k \leftarrow \gamma \mathbb{E}[\psi(X(k+1)) \mid X(k)] - \psi(X(k))$
- $A_k \leftarrow \Upsilon_k \Upsilon_k^\top$
- $b_k \leftarrow \Upsilon_k c(X(k))$

$A \leftarrow \frac{1}{T} \sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T} \sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1} b$

MSBE example



Costs are in red, $\gamma = 0.5$, parameters in blue
True values: $h(1) = \frac{35}{6}$, $h(2) = \frac{5}{6}$, $h(3) = \frac{35}{3}$, $h(4) = \frac{5}{3}$
MSBE values: $h(1) = h(2) = \frac{10}{3}$, $h(3) = \frac{32}{3}$, $h(4) = \frac{8}{3}$ (smoothing)
TD(0) values: $h(1) = h(2) = \frac{10}{3}$, $h(3) = \frac{35}{3}$, $h(4) = \frac{5}{3}$

LSTD

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T}$ be in steady state

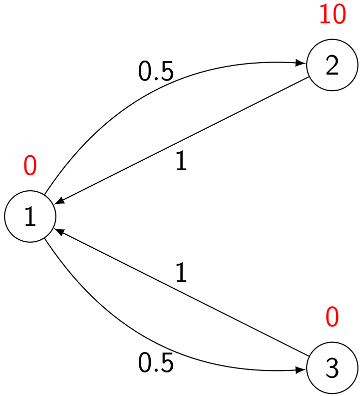
Algorithm (LSTD)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\Upsilon_k \leftarrow \gamma\psi(X(k+1)) - \psi(X(k))$
- $A_k \leftarrow \Upsilon_k \Upsilon_k^\top$
- $b_k \leftarrow \Upsilon_k c(X(k))$

$A \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1} b$

MSTD example



Costs are in red, $\gamma = 0.5$, full parametrization
True/TD(0) values: $h(1) = \frac{10}{3}$, $h(2) = \frac{35}{3}$, $h(3) = \frac{5}{3}$
MSTD values: $h(1) = \frac{10}{3}$, $h(2) = \frac{32}{3}$, $h(3) = \frac{8}{3}$ (smoothing)

LSVE

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T-1}$ be in steady state

Algorithm (LSVE)

$\theta_0 \leftarrow$ arbitrary
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\hat{h}(k) \leftarrow$ unbiased estimator of $h(X(k))$
- $A_k \leftarrow -\psi(X(k))\psi(X(k))^\top$
- $b_k \leftarrow -\psi(X(k))^\top \hat{h}(k)$

$A \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1} b$

Cost and value function – averaged case

Consider an autonomous system (ergodic)

Cost function: $c : \mathcal{X} \to \mathbb{R}_{\geq 0}$

Averaged expected cost:

$$\eta := \mathbb{E}[c(X(k)) \,|\, X(k) \sim \pi]$$

Relative value function:

$$h(x) := \mathbb{E}\left[\sum_{k=0}^{\infty} c(X(k)) - \eta \,\middle|\, X(0) = x\right]$$

Cost and value function – averaged case

Consider an autonomous system (ergodic)

Theorem

It holds that

$$\eta = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{k=0}^{T-1} c(X(k)) \,\middle|\, X(0)\right]$$

(independent of $X(0)$)

Hence, the name "averaged expected cost"

Poisson equation

Equivalent of Bellman equation for the averaged cost

Consider an autonomous system (ergodic)

The expected averaged cost η and the relative value function h satisfy the **Poisson equation**:

$$h(X(k)) = c(X(k)) - \eta + \mathbb{E}[h(X(k+1)) \mid X(k)]$$

Cost, value function and Q-function – averaged case

Consider a controlled system with policy ϕ (ergodic)

Cost function: $c : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_{\geq 0}$

Averaged expected cost:

$$\eta_\phi := \mathbb{E}[c(X(k), U(k)) \,|\, X(k) \sim \pi, \phi]$$

Relative value function:

$$h_\phi(x) := \mathbb{E}\left[\sum_{k=0}^{\infty} c(X(k), U(k)) - \eta_\phi \,\middle|\, X(0) = x, \phi\right]$$

Relative Q-function:

$$Q_\phi(x, u) := \mathbb{E}\left[\sum_{k=0}^{\infty} c(X(k), U(k)) - \eta_\phi \,\middle|\, X(0) = x, U(k) = u, \phi\right]$$

Cost and value function – averaged case

Consider a controlled system with policy ϕ (ergodic)

Theorem

It holds that

$$\eta_\phi = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{k=0}^{T-1} c(X(k), U(k)) \,\middle|\, X(0), \phi\right]$$

(independent of $X(0)$)

Hence, the name "averaged expected cost"

Poisson equation

Equivalent of Bellman equation for the averaged cost

Consider a controlled system with policy ϕ (ergodic)

The expected averaged cost η_ϕ and the relative Q-function Q_ϕ satisfy the **Poisson equation**:

$$Q_\phi(X(k), U(k)) = c(X(k), U(k)) - \eta_\phi + \mathbb{E}[Q_\phi(X(k+1), U(k+1)) \,|\, X(k), U(k), \phi]$$

Remark

Similar equation for h_ϕ; omitted

Averaged-cost TD(λ)

Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (Averaged-cost TD(λ))

$\theta_0, \rho_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \ldots,$ until stopping criterion is met:

- $\zeta(k) \leftarrow \lambda\zeta(k-1) + \nabla_\theta h^{\theta_k}(X(k))$
- $\delta_k \leftarrow c(X(k)) - \rho_k + h^{\theta_k}(X(k+1)) - h^{\theta_k}(X(k))$
- $\rho_{k+1} \leftarrow \rho_k + \beta_k\delta_k$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k\delta_k\zeta(k)$

Return θ_k

Note: Typically, $\beta_k \leq \alpha_k$

Averaged-cost LSTD(λ)

Assume linear parametrization: $h^\theta = \theta^\top \psi$

Let $\{X(k)\}_{k=0}^{T}$ be in steady state

Algorithm (Averaged-cost LSTD(λ))

$\zeta(-1) \leftarrow 0$
$\rho \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} c(X(k))$
For each $k = 0, 1, \dots, T-1$:

- $\zeta(k) \leftarrow \lambda\zeta(k-1) + \psi(X(k))$
- $A_k \leftarrow \zeta(k)\{\gamma\psi(X(k+1)) - \psi(X(k))\}^\top$
- $b_k \leftarrow \zeta(k)\{\rho - c(X(k))\}$

$A \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} A_k$
$b \leftarrow \frac{1}{T}\sum_{k=0}^{T-1} b_k$
Return $\theta = A^{-1} b$

Soundness and convergence of averaged-cost TD(λ)

Similar soundness and convergence results holds for the use of the averaged-cost TD(λ) algorithm to approximate the relative value function as for the TD(λ) algorithm for $0 \leq \lambda < 1$ and linear parametrizations. Note however that for $\lambda = 1$, it may not converge (even for linear parametrizations).

(Meyn, Theorems 9.7 and 9.8)

TD(λ) with regeneration

Goal: Address high variance when $\lambda \approx 1$

Assume $\mathcal{X}$ finite and let $\bar{x} \in \mathcal{X}$ be a recurrent state

Idea: Reset $\zeta(k)$ when $X(k) = \bar{x}$
Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

TD(λ) with regeneration

Assume $\mathcal{X}$ finite and let $\bar{x} \in \mathcal{X}$ be a recurrent state
Let $\{X(k)\}_{k=0}^{\infty}$ be in steady state

Algorithm (Regenerative TD(λ))

$\theta_0 \leftarrow$ arbitrary
$\zeta(-1) \leftarrow 0$
For each $k = 0, 1, \dots,$ until stopping criterion is met:

- if $X(k) = \bar{x}$, then $\zeta(k-1) \leftarrow 0$
- $\zeta(k) \leftarrow \lambda\gamma\zeta(k-1) + \nabla_\theta h^{\theta_k}(X(k))$
- $\delta_k \leftarrow c(X(k)) + \gamma h^{\theta_k}(X(k+1)) - h^{\theta_k}(X(k))$
- $\theta_{k+1} \leftarrow \theta_k + \alpha_k \delta_k \zeta(k)$

Return θ_k

Trust-region policy optimization (TRPO)

To do

Proximal policy optimization (PPO)

To do