True Online TD(λ)

Harm van Seijen
Richard S. Sutton

Presented by: Wittawat Jitkrittum
wittawat@gatsby.ucl.ac.uk

Gatsby Tea Talk

12 Sep 2014

Overview

- In RL, TD(λ) is a core algorithm for value function estimation.
  - Conceptually simple forward view.
  - Can be implemented online with backward view.
- But, forward view = backward view only for the **offline** version.
- Existing TD(λ) is not truely online.

Harm van Seijen, Richard S. Sutton
True Online TD(λ).
ICML 2014

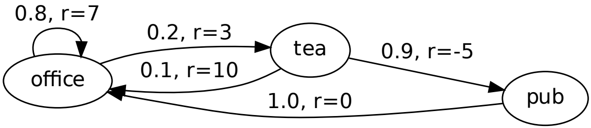
- This paper:[van Seijen and Sutton, 2014]:
  - New variant of TD(λ) such that forward view = backward view for **online** version.

MRP

Markov Reward Process

A discrete-time Markov reward process (MRP) is a tuple $\langle \mathcal{S}, p, r, \gamma \rangle$

- $\mathcal{S}$: finite set of states
- $p(s'|s)$: state transition probability
- $r(s, s')$: expected reward for transiting from s to s'
- $\gamma \in [0, 1]$: discount factor (weights for future rewards)



- MRP trajectory: $S_0, R_1, S_1, R_2, S_2, \ldots$
- MDP trajectory: $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, \ldots$

Value Function

Return from time t

$G(t) = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{i=1}^{\infty} \gamma^{i-1} R_{t+i}$

Value Function

$v(s) = \mathbb{E}\left[G(t) \mid S_t = s\right] = \mathbb{E}\left[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s\right]$

- $v(s) =$ expected total return starting from s
- Often linear approximation is used to represent v.

$$\hat{v}_t(s) = \hat{v}(s, \theta_t) = \theta_t^\top \phi(s)$$

- $\phi(s_t) := \phi_t$ is a vector representation of s_t.
- θ : parameter of $\hat{v}$ to learn

Value Function Estimation

$\approxeq$ Stochastic gradient descent

$$\begin{aligned}\theta_{t+1} &= \theta_t + \alpha\left(U_t - \hat{v}_t(S_t)\right)\nabla_{\theta_t}\hat{v}_t(S_t) \\ &= \theta_t + \alpha\left(U_t - \hat{v}_t(S_t)\right)\phi_t \end{aligned} \tag{1}$$

- U_t : update target
  - Monte Carlo : $U_t = G(t)$ (not online)
  - TD(0) : $U_t = R_{t+1} + \gamma\hat{v}_t(S_{t+1})$
- Two update schemes
  - Online update : Do Eq.1 at each t.
  - Offline update : After episode k, do

$$\begin{aligned}\Delta_t &= \alpha\left(U_t - \hat{v}(S_t)\right)\nabla_{\theta^{(k)}}\hat{v}(S_t) \\ \theta^{(k+1)} &= \theta^{(k)} + \sum_{t=1}^{T}\Delta_t\end{aligned}$$

n-Step Return & λ-Return

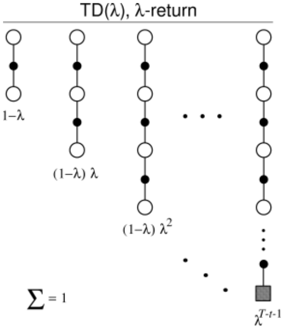
n-step return

$$G_\theta^{(n)}(t) := \left(\sum_{i=1}^{n} \gamma^{i-1} R_{t+i}\right) + \gamma^n \theta^\top \phi_{t+n}$$

$$n = 1 \qquad G_\theta^{(1)}(t) = R_{t+1} + \gamma \hat{v}(S_{t+1})$$

$$n = 2 \qquad G_\theta^{(2)}(t) = R_{t+1} + \gamma R_{t+2} + \gamma^2 \hat{v}(S_{t+2})$$

$$n = \infty \qquad G^{(\infty)}(t) = G(t)$$



λ-return

$$L_\theta^\lambda(t) := (1-\lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_\theta^{(n)}(t)$$

- Setting $\lambda = 0$ gives TD(0) i.e.,
  $U_t = G_\theta^{(1)}(t)$

Classical TD(λ)

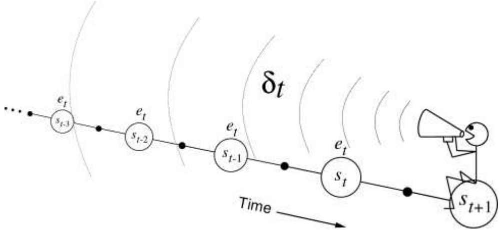
- Forward-view TD(λ) (not online):

$$\theta_{t+1} = \theta_t + \alpha \left(L^{\lambda}_{\theta}(t) - \hat{v}(S_t) \right) \phi_t$$

- Backward-view TD(λ) (can be updated online):

$$\begin{aligned} \text{(TD error) } \delta_t &= R_{t+1} + \gamma \overbrace{\theta_t^\top \phi_{t+1}}^{\hat{v}_t(S_{t+1})} - \overbrace{\theta_t^\top \phi_t}^{\hat{v}_t(S_t)} \\ \boldsymbol{e}_t &= \gamma \lambda \boldsymbol{e}_{t-1} + \alpha \phi_t \\ \theta_{t+1} &= \theta_t + \delta_t \boldsymbol{e}_t \end{aligned}$$

- $\boldsymbol{e}_t$ is called eligibility traces. Contain footprints of recently visited states. $\boldsymbol{e}_0 = \mathbf{0}$.



Equivalence of Forward and Backward TD

Theorem

The sum of ***offline*** *updates is identical for forward-view and backward-view TD*(λ)

$$\sum_{t=1}^{T} \delta_t \boldsymbol{e}_t = \sum_{t=1}^{T} \alpha \left(L_\theta^\lambda(t) - \hat{v}(S_t) \right) \phi_t$$

where T *is the last time step in the episode.*

- Only approximately equal for online updates. ☹
- Another variant of online TD(λ) that matches the forward view exactly ?

Truncated λ-Return

Truncated λ-Return

$$L^\lambda(t,t') := (1-\lambda)\sum_{n=1}^{t'-t-1} \lambda^{n-1} G^{(n)}_{\theta_{t+n-1}}(t) + \lambda^{t'-t-1} G^{(t'-t)}_{\theta_{t'-1}}(t)$$

- Examples:

$$\begin{aligned}
L^\lambda(1,3) &= (1-\lambda)\, G^{(1)}_{\theta_1}(1) + \lambda G^{(2)}_{\theta_2}(1) \\
L^\lambda(1,4) &= (1-\lambda) G^{(1)}_{\theta_1}(1) + (1-\lambda)\lambda G^{(2)}_{\theta_2}(1) + \lambda^2 G^{(3)}_{\theta_3}(1) \\
L^\lambda(3,6) &= (1-\lambda) G^{(1)}_{\theta_3}(3) + (1-\lambda)\lambda G^{(2)}_{\theta_4}(3) + \lambda^2 G^{(3)}_{\theta_5}(3)
\end{aligned}$$

- $L^0(t,t') = G^{(1)}_{\theta_t}(t) = R_{t+1} + \gamma\theta_t^\top \phi_{t+1} \Rightarrow$ TD(0)

New forward view

- At each time t', previous truncated λ-returns are updated such that they are now truncated at t'.

Forward View of True Online TD(λ)

$$\theta_{t,k} = \theta_{t,k-1} + \alpha_{k-1}\left(L^\lambda(k-1,t) - \theta_{t,k-1}^\top \phi_{k-1}\right)\phi_{k-1}$$

Expanded:

$$\begin{aligned}
\theta_{0,0}: \quad & \theta_{0,0} = && \theta_{init} = \theta_0 \\
\theta_{1,1}: \quad & \theta_{1,0} = && \theta_{init} \\
& \theta_{1,1} = && \theta_{1,0} + \alpha_0\left(L^\lambda(0,1) - \theta_{1,0}^\top \phi_0\right)\phi_0 = \theta_1 \\
\theta_{2,2}: \quad & \theta_{2,0} = && \theta_{init} \\
& \theta_{2,1} = && \theta_{2,0} + \alpha_0\left(L^\lambda(0,2) - \theta_{2,0}^\top \phi_0\right)\phi_0 \\
& \theta_{2,2} = && \theta_{2,1} + \alpha_1\left(L^\lambda(1,2) - \theta_{2,1}^\top \phi_1\right)\phi_1 = \theta_2 \\
\theta_{3,3}: \quad & \theta_{3,0} = && \theta_{init} \\
& \theta_{3,1} = && \theta_{3,0} + \alpha_0\left(L^\lambda(0,3) - \theta_{3,0}^\top \phi_0\right)\phi_0 \\
& \theta_{3,2} = && \theta_{3,1} + \alpha_1\left(L^\lambda(1,3) - \theta_{3,1}^\top \phi_1\right)\phi_1 \\
& \theta_{3,3} = && \theta_{3,2} + \alpha_2\left(L^\lambda(2,3) - \theta_{3,2}^\top \phi_2\right)\phi_2 = \theta_3
\end{aligned}$$

- for $k = 0, 1, \ldots, t$
- Require storage of all observed states, rewards and $\{\theta_i\}_{i=1}^{t-1}$.

Backward View of True Online TD(λ)

Classical TD(λ):

$$\begin{aligned}\delta_t &= R_{t+1} + \gamma\theta_t^\top \phi_{t+1} - \theta_t^\top \phi_t \\ \boldsymbol{e}_t &= \gamma\lambda \boldsymbol{e}_{t-1} + \alpha_t \phi_t \\ \theta_{t+1} &= \theta_t + \delta_t \boldsymbol{e}_t\end{aligned}$$

True Online TD(λ):

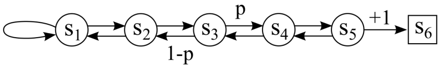
$$\begin{aligned}\delta_t &= R_{t+1} + \gamma\theta_t^\top \phi_{t+1} - \theta_{t-1}^\top \phi_t \\ \boldsymbol{e}_t &= \gamma\lambda \boldsymbol{e}_{t-1} + \alpha_t \phi_t - \alpha_t \gamma\lambda \left(\boldsymbol{e}_{t-1}^\top \phi_t\right) \phi_t \\ \theta_{t+1} &= \theta_t + \delta_t \boldsymbol{e}_t + \alpha_t \left(\theta_{t-1}^\top \phi_t - \theta_t^\top \phi_t\right) \phi_t\end{aligned}$$

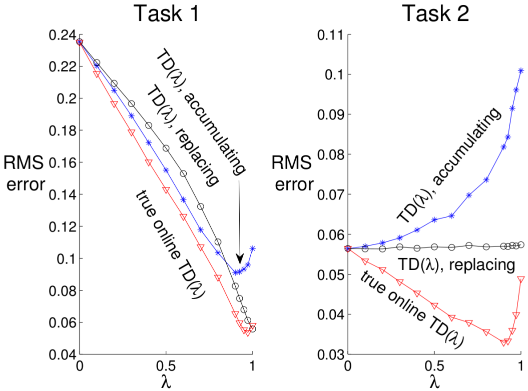
- Need to keep track of θ_t and θ_{t-1}.
- Same computational complexity.

Theorem ([van Seijen and Sutton, 2014])

θ_t (from backward update) $= \theta_{t,t}$ (from forward update) for all t.

Random-Walk Task

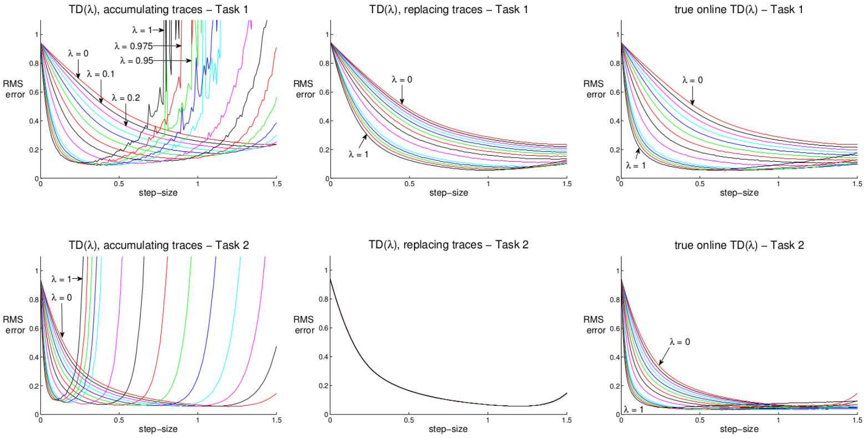




		s_1	s_2	s_3	s_4	s_5	s_6
Task 1	ϕ_1	1	$1/\sqrt{2}$	$1/\sqrt{3}$	0	0	0
	ϕ_2	0	$1/\sqrt{2}$	$1/\sqrt{3}$	$1/\sqrt{3}$	0	0
	ϕ_3	0	0	$1/\sqrt{3}$	$1/\sqrt{3}$	$1/\sqrt{3}$	0
	ϕ_4	0	0	0	$1/\sqrt{3}$	$1/\sqrt{3}$	0
	ϕ_5	0	0	0	0	$1/\sqrt{3}$	0
Task 2	ϕ_1	1	$1/\sqrt{2}$	$1/\sqrt{3}$	$1/\sqrt{4}$	$1/\sqrt{5}$	0
	ϕ_2	0	$1/\sqrt{2}$	$1/\sqrt{3}$	$1/\sqrt{4}$	$1/\sqrt{5}$	0
	ϕ_3	0	0	$1/\sqrt{3}$	$1/\sqrt{4}$	$1/\sqrt{5}$	0
	ϕ_4	0	0	0	$1/\sqrt{4}$	$1/\sqrt{5}$	0
	ϕ_5	0	0	0	0	$1/\sqrt{5}$	0

- Random-walk. $N = 11$ states in the experiment.
- "RMS error of state values at the end of each episode, averaged over the first 10 episodes, as well as 100 independent runs, for different values of λ at the best value of α."

Random-Walk Task



- "True online TD(λ) is the only method that achieves performance benefit on both tasks."

Conclusion

- True online TD(λ)
- A a new variant of TD(λ) allowing exact online updates.
- Same computational complexity as classical TD(λ).
- Empirically true online TD(λ) outperforms classical TD(λ).

Some Results

Recall

$$\begin{aligned}\delta_t &= R_{t+1} + \gamma\theta_t^\top \phi_{t+1} - \theta_t^\top \phi_t \\ \boldsymbol{e}_t &= \gamma\lambda \boldsymbol{e}_{t-1} + \alpha_t \phi_t\end{aligned}$$

Lemma 1

$L^\lambda(t, t')$ used by the forward-view true online TD(λ) is related to δ_t by

$$L^\lambda(t, t'+1) - L^\lambda(t, t') = (\gamma\lambda)^{t'-t}\,\delta_{t'}.$$

Lemma 2

$$\theta_{t+1,t} - \theta_{t,t} = \gamma\lambda\delta_t \boldsymbol{e}_{t-1}$$

References I

van Seijen, H. and Sutton, R. S. (2014).
True online td(lambda).
In Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, pages 692–700.