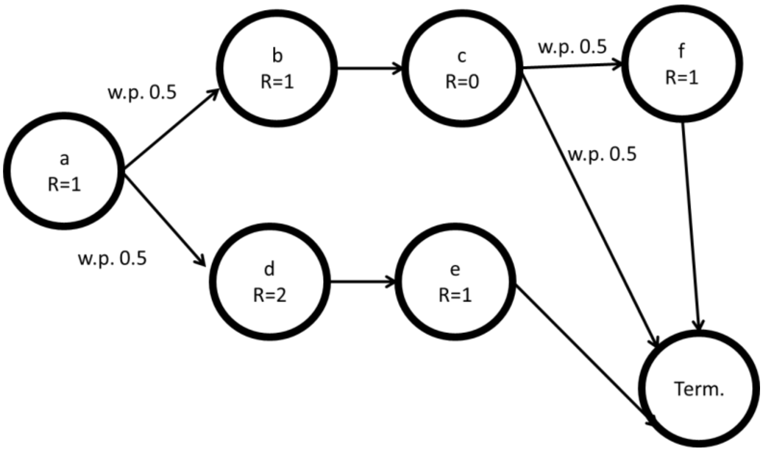
Class Tutorial 9

In this tutorial we will consider the task of *evaluating the value of a policy* without having access to the model of the environment. We will solely assume access to sampled data and explore the TD(0), Monte-Carlo and TD(λ) algorithms.

1. The TD(0) Algorithm

Consider the following (stochastic shortest path) MDP:



There is only one (trivial) policy here, which we denote by π.

a. Compute the values $V^{\pi}(s)$ for each state.

b. Consider running N trajectories $Data = \{s_0^1, \dots s_{T_1}^1\}, \dots \{s_0^N, \dots s_{T_N}^N\}$ from the MDP using π and starting from $s_0 = a$, for example:

$$\{s_0^1, \dots s_{T_1}^1\} = \{a, b, c, f, T\},$$
$$\{s_0^2, \dots s_{T_2}^2\} = \{a, d, e, T\},$$
$$\{s_0^3, \dots s_{T_3}^3\} = \{a, d, e, T\} \dots$$

Suggest an offline Monte-Carlo algorithm for estimating the values $V^{\pi}(s)$ for each state, using $Data$.

c. Consider running the TD(0) algorithm with the same data, starting from $\hat{V}_{TD}(s) = 0$ for all states. Write down the execution of the algorithm for the first few iterations. Choose a step size $a_n = 1/(no.of\ visits\ to\ s_n)$.

d. Consider running the TD(0) algorithm again, but now assume that the values of states $b, \dots, f$ start from their **true** values, and do not change during the run of the algorithm. Show that $\hat{V}_{TD}(a)$ converges to its true value.

Solution

a. $V^\pi(s)=E^\pi\left[\sum_{t=0}^{T} r(s_t)\,|\,s_0=s\right]$, and $V^\pi(s)=E^\pi\left[r(s)+V^\pi(s')\right]$, therefore

$V^\pi(f)=1, V^\pi(c)=0.5, V^\pi(e)=1, V^\pi(b)=1.5, V^\pi(d)=3, V^\pi(a)=1+(1.5+3)/2.$

b. For each state s, let $D_s=\left\{s,\ldots s_{T_1}^1\right\},\ldots,\left\{s,\ldots s_{T_{N_s}}^{N_s}\right\}$ denote parts of the trajectories that start at s, out of all the trajectories in D that path through s. Then the MC estimate is

$$\hat{V}_{MC}(s)=\frac{1}{N_s}\sum_{i=1}^{N_s}\left(r(s)+\ldots+r(s_{T_i}^i)\right) .$$

c. TD(0):

$$\hat{V}_{TD}(s_n):=\hat{V}_{TD}(s_n)+\alpha_n\cdot\left(r(s_n)+\hat{V}_{TD}(s_{n+1})-\hat{V}_{TD}(s_n)\right)$$

Following the state sequences in the example:

$$\hat{V}_{TD}(a):=\hat{V}_{TD}(a)+\alpha_1\cdot\left(r(a)+\hat{V}_{TD}(b)-\hat{V}_{TD}(a)\right)=1\cdot(1+0-0)$$
$$\hat{V}_{TD}(b):=\hat{V}_{TD}(b)+\alpha_2\cdot\left(r(b)+\hat{V}_{TD}(c)-\hat{V}_{TD}(b)\right)=1\cdot(1+0-0)$$
$$\hat{V}_{TD}(c):=\hat{V}_{TD}(c)+\alpha_3\cdot\left(r(c)+\hat{V}_{TD}(f)-\hat{V}_{TD}(c)\right)=1\cdot(0+0-0)$$
$$\hat{V}_{TD}(f):=\hat{V}_{TD}(f)+\alpha_4\cdot\left(r(f)+\hat{V}_{TD}(T)-\hat{V}_{TD}(f)\right)=1\cdot(1+0-0)$$
$$\hat{V}_{TD}(a):=\hat{V}_{TD}(a)+\alpha_5\cdot\left(r(a)+\hat{V}_{TD}(d)-\hat{V}_{TD}(a)\right)=1+\frac{1}{2}(1+2-1)$$

...

d. We have in this case

$$\hat{V}_{TD}(a)=r(a)+\frac{1}{N_a}\sum_{i=1}^{N_a}V^\pi(s_2^i)\to E^\pi\left[r(a)+V^\pi(s_2)\right]=V^\pi(a)$$

2. The TD(λ) Algorithm

Recall that in the TD(0) algorithm without function approximation the update for the value function is $V_{t+1}(s_t)=V_t(s_t)+\alpha_t\left(r(s_t)+\gamma V_t(s_{t+1})-V_t(s_t)\right)$, where the intuition behind it is that $r(s_t)+\gamma V_t(s_{t+1})$ is an estimate for $V_t(s_t)$, and $\delta_1=r(s_t)+\gamma V_t(s_{t+1})-V_t(s_t)$ is thus the 1-step error term. One may similarly take $r(s_t)+\gamma r(s_{t+1})+\gamma^2 V_t(s_{t+2})$ as an estimate for $V_t(s_t)$, and define a 2-step error term $\delta_2=r(s_t)+\gamma r(s_{t+1})+\gamma^2 V_t(s_{t+2})-V_t(s_t)$. Similarly, we may define $\delta_3,\delta_4,\ldots$.

The TD(λ) error is defined as a weighted average of $\delta_1,\delta_2,\ldots$ as follows:

$$\delta^{\lambda} = (1-\lambda)\sum_{i=1}^{\infty}\lambda^{i-1}\delta_i$$

a. For a state sequence of length 4: s_1, s_2, s_3, s_4 write down the TD(λ) errors explicitly for each state. Explain how a TD(λ) policy evaluation may be implemented.

One difficulty with the implementation above is that it seems difficult to implement as an online algorithm. However, a simple trick allows to stream-line the calculation as follows.

b. Show that for a long sequence of states $s_1, s_2, s_3, \ldots$ the error term $\delta^{\lambda}(s_1)$ may be written as a combination of $\delta_0(s_1), \delta_0(s_2), \ldots$

c. Propose an online implementation of the TD(λ) algorithm.

Solution

a.

$$\begin{aligned}
\delta^{\lambda}(s_1) &= (1-\lambda)\big(r(s_1)+\gamma V(s_2)-V(s_1)\big) \\
&\quad +\lambda(1-\lambda)\Big(r(s_1)+\gamma r(s_2)+\gamma^2 V(s_3)-V(s_1)\Big) \\
&\quad +\lambda^2(1-\lambda)\Big(r(s_1)+\gamma r(s_2)+\gamma^2 r(s_3)+\gamma^3 V(s_4)-V(s_1)\Big) \\
\delta^{\lambda}(s_2) &= (1-\lambda)\big(r(s_2)+\gamma V(s_3)-V(s_2)\big) \\
&\quad +\lambda(1-\lambda)\Big(r(s_2)+\gamma r(s_3)+\gamma^2 V(s_4)-V(s_2)\Big) \\
\delta^{\lambda}(s_3) &= (1-\lambda)\big(r(s_3)+\gamma V(s_4)-V(s_3)\big)
\end{aligned}$$

b. note that

$$\begin{aligned}
\delta^{\lambda}(s_1) &= -V(s_1)+(1-\lambda)\big(r(s_1)+\gamma V(s_2)\big) \\
&\quad +\lambda(1-\lambda)\Big(r(s_1)+\gamma r(s_2)+\gamma^2 V(s_3)\Big) \\
&\quad +\lambda^2(1-\lambda)\Big(r(s_1)+\gamma r(s_2)+\gamma^2 r(s_3)+\gamma^3 V(s_4)\Big) \\
&\quad +\ldots
\end{aligned}$$

By taking out $r(s_1)$ we see that its coefficients sum up to 1. Doing this for the other terms gives:

$$
\begin{aligned}
\delta^{\lambda}(s_1) = -V(s_1) \\
&+ (\gamma\lambda)^0 \left(r(s_1) + \gamma V(s_2) - \gamma\lambda V(s_2) \right) \\
&+ (\gamma\lambda)^1 \left(r(s_2) + \gamma V(s_3) - \gamma\lambda V(s_3) \right) \\
&+ \dots \\
&= (\gamma\lambda)^0 \left(r(s_1) + \gamma V(s_2) - V(s_1) \right) \\
&+ (\gamma\lambda)^1 \left(r(s_2) + \gamma V(s_3) - V(s_2) \right) \\
&+ \dots \\
&= \sum_{t=1}^{\infty} (\gamma\lambda)^{t-1} \delta_0(s_t)
\end{aligned}
$$

c.

$$
\begin{aligned}
&V_{t+1}(s) = V_t(s) + \alpha_t \delta_t e_t(s) && \forall s \\
&\delta_t = r(s_t) + \gamma V_t(s_{t+1}) - V_t(s_t) \\
&e_t(s) = \gamma\lambda e_{t-1}(s) + \mathbb{1}\{s_t = s\} && \forall s
\end{aligned}
$$