EE675A - IITK, 2022-23-II

Lecture 16: TD (Lambda) Backward view & Off-policy Prediction

20th March 2023

Lecturer: Subrahmanya Swamy Peruru *Scribe: Suhas, Akansh Agrawal*

1 Recap and Overview

In the previous lecture,[1] we discussed different variants of Temporal Difference (TD) updates.

TD Update:

$$V_{new}(S_t) = V_{old}(S_t) + \alpha[G_t - V_{old}(S_t)] \tag{1}$$

$$G_t = R_{t+1} + \gamma V_{old}(S_{t+1})$$

n-Step TD:

$$V_{new}(S_t) = V_{old}(S_t + \alpha[G_t^{(n)} - V_{old}(S_t)]) \tag{2}$$

$$G_t^{(n)} = R_{t+1} + \gamma R_{t+2} .... + \gamma^{n-1} R_{t+n} + \gamma^n V_{old}(S_{t+n})$$

Notice how in Eq. 2, substituting $n = 1$ gives us the TD update and $n = \infty$ gives us the Monte Carlo update. In practice, it is observed that intermediate values of n work well.

$$G_t^{(3)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 V_{old}(S_{t+3})$$

$$G_t^{(2)} = R_{t+1} + \gamma R_{t+2} + \gamma^2 V_{old}(S_{t+2})$$

$$G_t^{(2,3)} = \frac{1}{2} G_t^{(2)} + \frac{1}{2} G_t^{(3)}$$

The composite return possesses an error reduction property similar to that of individual n-step returns and thus can be used to construct updates with guaranteed convergence properties. Any set of n-step returns can be averaged in this way, even an infinite set, as long as the weights on the component returns are positive and sum to 1.[2]

1.1 TD(λ)

$$G_t^\lambda = (1 - \lambda) \sum_{n=1}^{\infty} G_t^{(n)} \lambda^{n-1}$$

$$G_t^{(n)} = G_t \quad \forall n \geq T - t$$

1.2 Issues

- Computing G_t^λ requires completion of an episode
- Online updates are not possible

In order to alleviate these issues, we make use of Eligibility Traces to make the TD update.

2 Eligibility Traces

Consider an experiment where a rat is given food at some time instant (Say $t = 5$). Now suppose, a bulb was lit at $t = 4$, and bells were rung at times $t = 1, 2, 3$. To which event should the rat attribute the event of getting food? Should it be the bulb because it happened recently or should it be the bell because it happened more number of times? The first line of thought represents a **Recency bias** whereas the second represents a **Frequency bias**.

This idea can be modelled using an eligibility trace function $E_t(s)$ that simulates short-term memory. $E_t(s)$ tells us how informative a state s is at a point in time t. Whenever we land in a state s, we increase its eligibility value. We then exponentially decay the eligibility values of all the states to simulate the passage of time. More formally, we have

$$E_0(s) = 0, \forall s$$

$$E_t(s) = \lambda E_{t-1}(s) + \mathbb{1}_{\{s_t = s\}}$$

TD Error:

$$\delta_t = R_{t+1} + \gamma V(s_{t+1}) - V(s_t)$$

3 Backward view of TD(λ)

Normal TD Update:

$$V(s_t) = V(s_t) + \alpha \delta_t$$

TD(λ):

$$V(s) = V(s) + \alpha E_t(s) \delta_t \quad ; \ \forall s$$

The important difference is that we make an update to **all** the states weighted by how informative that state is. The higher the eligibility value of a state, the higher the contribution of R_t to it.

Consider the case when $\lambda = 0$.

$$E_t(s) = \mathbb{1}_{\{s_t = s\}}$$

$$V(s) = V(s) + \alpha(\mathbb{1}_{\{s_t = s\}})\delta_t \quad ; \ \forall s$$

$$V(s) = V(s) \quad ; \ \forall s \neq s_t$$

$$V(s_t) = V(s_t) + \alpha\delta_t$$

Thus, we see how this case boils down to the Normal TD Update.

3.1 Theoretical guarantees

Consider an episode

$$S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}, \ldots, S_T$$

For the Backward view TD (λ) algorithm discussed above, it can be shown that the offline update (at the end of the episode) is equivalent to the update performed by Forward view TD (λ). More recent methods that modify the eligibility trace function are able to prove equivalence of updates even in the online case.

4 Off Policy Prediction

Off-policy prediction is a problem in RL, where an agent tries to learn the value of a policy different from the one it is currently following. This problem arises when the agent is exploring the environment and collecting data (e.g., through random actions or by following a different policy), but needs to estimate the value of a target policy in order to make decisions. Formally, if given Π , the objective is to estimate V_Π by following the other policy $\mu \neq \Pi$.

$$S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1} \sim \mu(.|S_t) \tag{3}$$

The policy Π is the target policy and the policy μ is the behaviour policy. One approach to estimate the value of the target policy is to use the off-policy prediction algorithm called importance sampling (which we will cover soon in this lecture). For now we can understand importance sampling as name suggests to be an approach which basically takes the weighted form of observed rewards and actions into account which is based on the probability of their occurrence under the target policy. In simple words, this indicates that actions and rewards that are more likely to occur under the target policy are given more weight than those that are less likely.

Consider the example: One can approximately estimate $\mathbb{E}_{x\sim p}[f(x)]$ using Monte-Carlo Approximation as:

$$\mathbb{E}_{x\sim p}[f(x)] = \sum_x f(x)p(x) \approx \frac{\sum_{i=1}^{N} f(x^{(i)})}{N} \tag{4}$$

where $x^{(i)} \sim p$. The other way we may use Monte-Carlo Approximation as:

$$\mathbb{E}_{x\sim p}[f(x)] = \sum_x f(x)p(x)q(x)/q(x) = \sum_x [f(x)p(x)/q(x)]q(x) = \mathbb{E}_{x\sim q}[f(x)p(x)/q(x)] \tag{5}$$

$$\mathbb{E}_{x\sim q}[f(x)p(x)/q(x)] \approx \frac{\sum_{i=1}^{N} f(x^{(i)})p(x^{(i)}/q(x^{(i)})}{N} \tag{6}$$

where $x^{(i)} \sim q$. We use the similar idea for achieving the target policy based on different policy via Off-Policy Monte Carlo Prediction.

4.1 Off-Policy Monte Carlo Prediction

Consider a sample episode x as follows :

$$x \triangleq (S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}, R_{t+2}, S_{t+2}...) \tag{7}$$

Say we denote the G_t by $f(x)$ of the above example, therefore :

$$G_t = f(x) \triangleq \sum_{k \geq 1} \gamma^{k-1} R_{t+k} \tag{8}$$

$$V_\Pi = \mathbb{E}_{x \sim \mu}[f(x)|S_t = s] \tag{9}$$

or,

$$V_\Pi = \mathbb{E}_{x \sim q} \; [f(x) \; p(x \text{ while following policy } \Pi \;)/p(x \text{ while following policy } \mu \;)|S_t = s]$$

In above equation, $p(x$ while following policy Π) = $\Pi(A_t|S_t = s)R^{A_t}_{S_t}P^{A_t}_{S_t,S_{t+1}}\Pi(A_{t+1}|S_{t+1})...$, where $R^{A_t}_{S_t}$ and $P^{A_t}_{S_t,S_{t+1}}$ are part of dynamics of MDP and are unknown to us.

Also one can write , $p(x$ while following policy μ) = $\mu(A_t|S_t = s)R^{A_t}_{S_t}P^{A_t}_{S_t,S_{t+1}}\mu(A_{t+1}|S_{t+1})....$

We can write the ratio of $p(x$ while following policy Π) and $p(x$ while following policy μ) using above two expressions as :

$$\rho_{\Pi/\mu} = \frac{\Pi(A_t|S_t = s)\Pi(A_{t+1}|S_{t+1})...}{\mu(A_t|S_t = s)\mu(A_{t+1}|S_{t+1})...} \tag{10}$$

This ratio is known as the importance sampling ratio. Since the ratio does not have explicit terms of $R^{A_t}_{S_t}$ and $P^{A_t}_{S_t,S_{t+1}}$ we can directly use it to get the value function as :

$$V_\Pi = \mathbb{E}_{x \sim \mu}[G^\mu_t \rho_{\Pi/\mu}] \tag{11}$$

Thus, the off-policy MC prediction takes average over multiple episodes obtained by following policy μ by appropriately weighting G_t . The ordinary importance sampling ratio we saw earlier can have very large value , so to normalize this effect we can utilise another **variant** which is **weighted importance**. It is key point to understand that the importance sampling is unbiased whereas weighted importance sampling is biased (though the bias converges asymptotically to 0). Whereas, the variance of ordinary importance sampling is in general unbounded because the variance of the ratios can take any large value (unbounded), whereas in the case of the weighted estimator the largest weight on any single return is one (because of the normalization done). For more details about it please refer to Sutton and Barto Book.

4.2 Advantages

This approach can get the value function estimate of target policy by re-using the data corresponding to older policies. Also if we wish to learn about some state which is less exploratory by the target policy then we can use some better exploratory policy (which has high chance of exploring the rare state) efficiently for this purpose. Also, this techniques can be utilised in Q-learning (which we will be discussed in future lectures) where we can learn of optimal policy using the behavior policy .

References

[1] S. S. Peruru. Lecture notes of introduction to reinforcement learning, January 2023.

[2] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.