Reinforcement Learning

-Sarath Chandar

5. Monte Carlo Methods

5. Monte Carlo Methods

Dynamic Programming – requires access to the transition dynamics, which is often not available.

Can we directly learn the value functions from experience?

Experience
- actual experience (by interacting with the world)
- simulated experience (by interacting with the model)

Simulated experience only requires models from which you can sample the transitions. It does not require access to the complete probability distributions of all possible transitions that is required by DP.

Monte Carlo methods:-

- Solving RL based on averaging sample returns.

- Consider episodic tasks where episodes always terminate.
- We compute returns after termination and use it to update value estimates and policies.
- This is similar to reward estimation in bandits.

In this case, we estimate returns for each state-action pair. There are multiple states and they can be treated as multiple related bandit problems. Return from a state depends not just on the action taken in current state, but also on action taken in future states.

- Since all action selections are undergoing learning, the problem becomes non-stationary from the point of view of earlier state.

Monte Carlo Prediction:-

Given a policy, estimate the state value fn.

We want to estimate $V_\pi(s)$, the value of a state 's' under policy π, given a set of episodes obtained by following 'π' and passing through 's'.

Each occurence of a state 's' in an episode → "visit" to 's'.

First-visit MC → estimates $V_\pi(s)$ as the average of the returns following first visit to 's'.

Every-visit MC → averages returns following all visits to 's'.

First-visit MC prediction, for estimating $V \approx v_\pi$

Input: a policy π to be evaluated

Initialize:

$V(s) \in \mathbb{R}$, arbitrarily, for all $s \in \mathcal{S}$

$Returns(s) \leftarrow$ an empty list, for all $s \in \mathcal{S}$

Loop forever (for each episode):

Generate an episode following π: $S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless S_t appears in $S_0, S_1, \ldots, S_{t-1}$:

Append G to $Returns(S_t)$

$V(S_t) \leftarrow$ average$(Returns(S_t))$

Both first-visit MC and every-visit MC will converge to $v_\pi(s)$ as the number of visits to 's' goes to infinity.

Backup diagram for Monte-Carlo :-

root node → state node.

DP includes only one-step transitions, which MC covers the entire episode.

DP - expected update

MC - Sample update.

Note 1: In MC, estimates of each state are independent. The estimate of one state does not build upon the estimate of any other state. So no "bootstrapping" like DP.

Note 2: MC is attractive if we want to estimate value of only one state.

Monte-Carlo estimation of action values:-

State values require access to the model to look ahead one step and choose whichever action that leads to the best combination of reward and next state.

Without a model, we need to estimate q_π.

Estimate $q_\pi(s,a)$, the expected return when starting in state 's', taking action 'a' and thereafter following policy π.

We can use same MC method we used for estimate $v_\pi(s)$. However, many state-action pairs may never be visited. If π is a deterministic policy, then we observe returns only for one of the actions from each state! These estimates are useless since we need action values of all actions to pick the best action. This is the problem of maintaining exploration.

(ES) Exploring starts assumption:- the episodes start in a state-action pair and that every pair has a non-zero prob. of being

selected as start.

ES guarantees that all state-action pairs will be visited an infinite # of times in the limit of an infinite number of episodes.

Note: ES is a very strong assumption. Not always practical.

Alternate soln: To consider only stochastic policies.

Monte Carlo Control :-

We will follow the idea of Generalized Policy Iteration (GPI).

$$\pi_0 \xrightarrow{E} q_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} q_{\pi_1} \xrightarrow{I} \pi_2 \cdots \xrightarrow{I} \pi_* \xrightarrow{E} q_*$$

Policy evaluation is done exactly as described in MC prediction.

Policy improvement is done by making the policy greedy w.r.t the current value fn.

$$\pi_{k+1}(s) = \underset{a}{\operatorname{argmax}}\ q_{\pi_k}(s,a)$$

$$q_{\pi_k}(s, \pi_{k+1}(s)) = q_{\pi_k}\left(s, \underset{a}{\operatorname{argmax}}\ q_{\pi_k}(s,a)\right)$$

$$= \max_a q_{\pi_k}(s,a)$$

$$\geq q_{\pi_k}(s, \pi_k(s))$$

$$\geq v_{\pi_k}(s)$$

There are 2 unlikely assumptions here:

① Episodes have exploring starts.

② Policy evaluation could be done with infinite number of episodes.

Easy to remove the second assumption. We don't need to wait for PE to converge to start doing policy improvement. We can do the extreme case like value iteration and improve the policy after every episode of policy

evaluation.

Monte Carlo ES (Exploring Starts), for estimating $\pi \approx \pi_*$

Initialize:
  $\pi(s) \in \mathcal{A}(s)$ (arbitrarily), for all $s \in \mathcal{S}$
  $Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$
  $Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Loop forever (for each episode):
  Choose $S_0 \in \mathcal{S}$, $A_0 \in \mathcal{A}(S_0)$ randomly such that all pairs have probability > 0
  Generate an episode from S_0, A_0, following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:
      Append G to $Returns(S_t, A_t)$
      $Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$
      $\pi(S_t) \leftarrow \text{argmax}_a Q(S_t, a)$

Note: Intuitively Monte Carlo ES should converge to the optimal policy. But the general version of this convergence is still not yet proved!

Monte Carlo Control without ES :-

To avoid ES, the agent should keep selecting all the actions. This requires stochastic/soft policies.

i.e. $\pi(a|s) > 0 \quad \forall s \in S$ and $a \in A(s)$

Example: ϵ-greedy policy.

In ϵ-greedy, non-greedy actions are given the minimal prob. of selection $\frac{\epsilon}{|\mathcal{A}(s)|}$ and the remaining bulk of prob. $1-\epsilon+\frac{\epsilon}{|\mathcal{A}(s)|}$ is given to the greedy action.

ϵ-greedy is an example for ϵ-soft policies.

ϵ-soft policies: $\pi(a|s) \geq \frac{\epsilon}{|\mathcal{A}(s)|}$ $\forall s, a,$ $\epsilon > 0.$

Among all ϵ-soft policies, ϵ-greedy is the closest to greedy.

[redacted] first-visit MC control (for ε-soft policies), estimates $\pi \approx \pi_*$

Algorithm parameter: small $\varepsilon > 0$
Initialize:
  $\pi \leftarrow$ an arbitrary ε-soft policy
  $Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$
  $Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Repeat forever (for each episode):
  Generate an episode following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:
      Append G to $Returns(S_t, A_t)$
      $Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$
      $A^* \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken arbitrarily)
      For all $a \in \mathcal{A}(S_t)$:
$$\pi(a|S_t) \leftarrow \begin{cases} 1 - \varepsilon + \varepsilon/|\mathcal{A}(S_t)| & \text{if } a = A^* \\ \varepsilon/|\mathcal{A}(S_t)| & \text{if } a \neq A^* \end{cases}$$

Any ϵ-greedy policy w.r.t. to q_π is an improvement over any ϵ-soft policy π is assured by the policy improvement theorem.

π' - ϵ-greedy policy.

$$q_\pi(s, \pi'(s)) = \sum_a \pi'(a|s)\, q_\pi(s,a)$$

$$= \frac{\epsilon}{|A(s)|} \sum_a q_\pi(s,a) + (1-\epsilon) \max_a q_\pi(s,a)$$

$$\geq \frac{\epsilon}{|A(s)|} \sum_a q_\pi(s,a) + (1-\epsilon) \sum_a \frac{\pi(a|s) - \frac{\epsilon}{|A(s)|}}{1-\epsilon}\, q_\pi(s,a)$$

$$\geq \frac{\epsilon}{|A(s)|} \sum_a q_\pi(s,a) - \frac{\epsilon}{|A(s)|} \sum_a q_\pi(s,a) + \sum_a \pi(a|s) q_\pi(s,a)$$

$$= V_\pi(s)$$

Thus by PIT, $\pi' > \pi$.

Note: Now we only achieve the best policy among the ϵ-soft policies, but on the other hand eliminated the assumption of ES.

On-policy Vs. Off-Policy methods :-

Can we learn a deterministic policy without ES assumption? However, we still need to select all the actions to learn about their values. One can use a stochastic policy to learn about an optimal policy that is not stochastic.

On-policy methods	Off-Policy methods.
- uses only one policy. It evaluates/improves the same policy that was used to take decision	- Uses two policies. 'behavior policy' - to explore 'target policy' - being learned.

Monte Carlo ES and MC with ϵ-soft policies are examples of on-policy methods.

Off-Policy methods: Use two policies, one that is learned about and that becomes the optimal policy (target policy) and one that is more exploratory and is used to generate behavior (behavior policy).

We are learning from data "off" the target policy and hence off-policy learning.

→ Off-Policy methods are more powerful and general.

→ They include on-policy methods as special case.

→ Can be applied when data is generated by a conventional non-learning controller or from human expert.

Off-Policy prediction via Importance Sampling :-

π - target policy
b - behavior policy
} fixed.

"Coverage" assumption :-

In order to use episodes from b to estimate values of π, we require that every action taken under π is also taken, at least occasionally, under b.

ie. $\pi(a|s) > 0 \Rightarrow b(a|s) > 0$.

Hence 'b' must be stochastic in states where it is not identical to π. 'b' could be ϵ-greedy.

Note: π can be deterministic.

Importance sampling - a general technique for estimating expected values under one distribution given samples from another.

Importance Sampling ratio:

Given a starting state S_t, the prob. of subsequent action trajectory under any policy π is

$$Pr\{A_t, S_{t+1}, A_{t+1} \cdots S_T \mid S_t, A_{t:T-1} \sim \pi\}$$

$$= \pi(A_t|S_t)\, p(S_{t+1} \mid S_t, A_t)\, \pi(A_{t+1}|S_{t+1}) \cdots p(S_T \mid S_{T-1}, A_{T-1})$$

$$= \prod_{k=t}^{T-1} \pi(A_k|S_k)\, p(S_{k+1} \mid S_k, A_k)$$

where 'p' - State transition prob fn.

The relative prob. of a trajectory under the target and behavior policies:

$$\rho_{t:T-1} = \frac{\prod_{k=t}^{T-1} \pi(A_k|S_k)\, p(S_{k+1} \mid S_k, A_k)}{\prod_{k=t}^{T-1} b(A_k|S_k)\, p(S_{k+1} \mid S_k, A_k)}$$

$$= \prod_{k=t}^{T-1} \frac{\pi(A_k|S_k)}{b(A_k|S_k)}$$

We wish to estimate the expected returns under the target policy. But all we have are returns G_t due to the behavior policy.

$$\mathbb{E}[G_t \mid S_t = s] = v_b(s)$$

We cannot average them to obtain v_π.

$$\mathbb{E}[\rho_{t:T-1} G_t \mid S_t = s] = v_\pi(s)$$

Let $\mathcal{J}(s)$ be set of all time steps when state 's' was visited.

$$V(s) = \frac{\sum_{t \in \mathcal{J}(s)} \rho_{t:T-1} G_t}{|\mathcal{J}(s)|}$$

↙ Ordinary importance Sampling. (OIS)

$$V(s) = \frac{\sum_{t \in \mathcal{J}(s)} \rho_{t:T-1} G_t}{\sum_{t \in \mathcal{J}(s)} \rho_{t:T-1}}$$

↙ weighted importance Sampling. (WIS)

Consider estimates of first-visit OIS and WIS after observing a single return from state s.

WIS: ratios in numerator and denominator cancels. Hence the estimate is G_t whose expected value is V_b, not V_π.

OIS: Estimate is always for V_π.

WIS is biased while OIS is unbiased, OIS has very high variance while WIS has less variance in estimation.

Incremental implementation of WIS :-

Suppose we have a sequence of returns $G_1, G_2 \dots G_{n-1}$ all starting in same state and each with a corresponding random weight W_i (e.g. $W_i = \rho_{t_i : T(t_i)-1}$).

We wish to form the estimate

$$V_n = \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_{k=1}^{n-1} W_k} \qquad n \geq 2.$$

and keep it up-to-date as we obtain a single additional return G_n. In addition to keeping track of V_n, we must maintain the cumulative sum C_n of the weights given to first 'n' returns.

$$V_{n+1} = V_n + \frac{W_n}{C_n}\left[G_n - V_n\right] \qquad n \geq 1$$

and $C_{n+1} = C_n + W_{n+1}$

where $C_0 = 0$.

Off-policy MC prediction (policy evaluation) for estimating $Q \approx q_\pi$

Input: an arbitrary target policy π
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
  $Q(s,a) \in \mathbb{R}$ (arbitrarily)
  $C(s,a) \leftarrow 0$

Loop forever (for each episode):
  $b \leftarrow$ any policy with coverage of π
  Generate an episode following b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  $W \leftarrow 1$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$, while $W \neq 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
    $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$
    $W \leftarrow W \frac{\pi(A_t|S_t)}{b(A_t|S_t)}$

Off-Policy Monte-Carlo Control:-

Off-policy MC control, for estimating $\pi \approx \pi_*$

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
  $Q(s,a) \in \mathbb{R}$ (arbitrarily)
  $C(s,a) \leftarrow 0$
  $\pi(s) \leftarrow \arg\max_a Q(s,a)$ (with ties broken consistently)

Loop forever (for each episode):
  $b \leftarrow$ any soft policy
  Generate an episode using b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  $W \leftarrow 1$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
    $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$
    $\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken consistently)
    If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)
    $W \leftarrow W \frac{1}{b(A_t|S_t)}$