LECTURE 4 - MODEL FREE PREDICTION.

PLANNING = PREDICTION + CONTROL

PREDICTION: EVALUATE V_π FOR π

CONTROL: FIND π_* AND V_{π_*}.

GOAL: GIVEN A POLICY π, WE WANT TO EVALUATE THE POLICY (FIND V_π) WITHOUT KNOWLEDGE OF MDP.

MONTE-CARLO METHODS.

- DOES NOT REQUIRE KNOWLEDGE OF THE DYNAMICS OF THE MDP. INSTEAD, REQUIRES EXPERIENCE. (BEING ABLE TO OBTAIN SAMPLES).
- ONLY WORKS FOR EPISODIC TASKS that terminate NO MATTER WHAT ACTIONS WE TAKE.
- We wish to ESTIMATE $V_\pi(s) = \mathbb{E}_{\tau\sim\pi}[R_{t+1} + \gamma R_{t+2} + \dots \mid S_t = s]$.
  - $\tau \sim \pi$: TRAJECTORY SAMPLED FROM π

WE ESTIMATE $V_\pi(s)$ BY EMPIRICAL AVG.

FIRST-VISIT MONTE CARLO ON POLICY EVALUATION

$N(s)$: # times we visit state $s = 0$.

$G(s) = 0$

Loop:

- SAMPLE EPISODE i: $(S_{i0}, A_{i0}, R_{i1}, S_{i1}, A_{i1}, \dots, R_{iT}, S_{iT})$.
- Define $G_{i,t} = r_{it} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \dots$: Return from time t onwards
- For each state visited in episode i:
  - For first time step t where s is visited:
    - $N(s) \leftarrow N(s) + 1$
    - $G(s) \leftarrow G(s) + R_{it}$
    - $V_\pi(s) \leftarrow G(s)$

- This estimator is
  - UNBIASED. $\mathbb{E}[V^{\pi}(s)] = \mathbb{E}[G_t | S_t = s]$
  - CONSISTENT: By Law of Large Numbers. $V^{\pi}(s) \to \mathbb{E}[G_t | S_t = s]$ as $N(s) \to \infty$.

EVERY-VISIT MC ON POLICY EVALUATION.

- We use the SAME IDEA but we consider every sample Return we see in an EPISODE in our Average.

- This Estimator
  - is Biased (Returns are not IID).
  - is consistent.

INCREMENTAL MC ON POLICY EVALUATION.

- We do not need to wait until the END OF ALL EPISODES to obtain the Estimate of $V_{\pi}(s)$. We can UPDATE our ESTIMATE AFTER EVERY EPISODE.

Suppose we want to compute the mean of $X_1, \dots, X_m$ incrementally.

$$\overline{X}_m = \frac{1}{m} \sum_{i=1}^{m} X_i = \frac{1}{m} \left[X_m + \sum_{i=1}^{m-1} X_i \right] =$$

$$\frac{1}{m} \left[X_m + (m-1) \overline{X}_{m-1} \right] = \frac{1}{m} \left[X_m + m \overline{X}_{m-1} - \overline{X}_{m-1} \right]$$

$$= \frac{1}{m} X_m + \overline{X}_{m-1} - \frac{\overline{X}_{m-1}}{m} = \frac{1}{m} \left[X_m - \overline{X}_{m-1} \right] + \overline{X}_{m-1}$$

$$\therefore \overline{X}_m = \overline{X}_{m-1} + \frac{1}{m} \left[X_m - \overline{X}_{m-1} \right]$$

- $\overline{X}_m$: NEW ESTIMATE
- $\overline{X}_{m-1}$: OLD ESTIMATE
- $\frac{1}{m}$: STEP SIZE
- $\left[X_m - \overline{X}_{m-1} \right]$: DIFFERENCE between what I thought the mean was and what I actually saw.

- Therefore, we can UPDATE our Estimate of $V_{\pi}(s)$ AFTER EACH EPISODE!

Loop:

- SAMPLE EPISODE i : $(S_{i,0}, A_{i,0}, R_{i,1}, S_{i,1}, A_{i,1}, \ldots, R_{i,T}, S_{i,T})$.
- Define $G_{i,\tau} = r_{i,\tau} + \gamma r_{i,\tau+1} + \ldots$: Return from time τ onwards
- For each state visited in episode i:
  - For first time step τ where s is visited:
    - $N(s) \leftarrow N(s) + 1$

UPDATE ESTIMATE

$$V^{\pi}(s) \leftarrow V^{\pi}(s) + \frac{1}{N(s)}\left[G_{i,\tau} - V^{\pi}(s)\right]$$

Running mean: Non-stationary domains (policy changes over time)

- In some cases, we want to 'forget' older data.

Therefore, we would update as follows:

$$V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha\left[G_{i,\tau} - V^{\pi}(s)\right]$$

If $\alpha > \frac{1}{N(s)}$, we are giving more importance to $G_{i,\tau}$ and 'forgetting' old values.

Cons of MC Method:

- Usually high variance $\Rightarrow$ Requires lots of data.
- Only useful for episodic MDP's.

Temporal Difference Learning (TD)

- Bootstraps (uses estimates to compute an estimate)
- Samples
- Updates $V^{\pi}(s)$ after every step (s, a, r, s')
- Useful in episodic or infinite-horizon MDPs.

Aim: Estimate $V_{\pi}(s) = \mathbb{E}\left[R_{t+1} + \gamma V_{\pi}(S_{t+1}) \mid S_t = s\right]$

AIM: ESTIMATE $V_\pi(s) = \mathbb{E}_{\text{INIT.}}[R_{t+1} + \gamma V_\pi(S_{t+1}) \mid S_t = s]$

BY SAMPLING RETURNS FROM π.

$$V(S_t) \leftarrow V(S_t) + \alpha \underbrace{\left[\overbrace{R_{t+1} + \gamma V(S_{t+1})}^{\text{TD TARGET}} - V(S_t) \right]}_{\text{TD ERROR}}$$

- WE ARE NOT SAMPLING ONE FULL RETURN G_t, INSTEAD WE ARE ONLY SAMPLING 1 STEP, AND USING OUR CURRENT ESTIMATE $V(S_{t+1})$.
- $V_\pi(s) = \mathbb{E}[G_t \mid S_t = s]$ (1) $= \mathbb{E}[R_{t+1} + \gamma V(S_{t+1}) \mid S_t = s]$ (2)
- IN MONTE CARLO WE USE G_t AS A TARGET, AND IN TD-LEARNING WE USE $R_{t+1} + \gamma V(S_{t+1})$ AS A TARGET.
- IT CAN BE SHOWN, UNDER CERTAIN CONDITIONS, THAT TD WILL CONVERGE TO $V_\pi(s)$ FOR A FIXED π.

ADV/DISADV MONTE-CARLO / TD-LEARNING.

- MC IS UNBIASED, BECAUSE WE ARE TAKING AN AVERAGE OVER RETURNS SAMPLED FROM THE TRUE DISTRIBUTION.
- MC IS HIGH VARIANCE, BECAUSE IN EACH TRAJECTORY WE HAVE NOISE, AND IT STACKS UP OVER TIME STEPS
- TD IS HIGH BIAS BECAUSE WE BOOTSTRAP.
- TD IS LOW VARIANCE. ONLY HAVE 1 STEP OF NOISE
- MONTE-CARLO IS MORE DATA-EFFICIENT THAN TD-LEARNING !!!

  WHY? → LET'S SAY WE ARE IN STATE S AND WE FOLLOW A TRAJECTORY OF LENGTH L TILL THE END OF THE EPISODE. A REWARD IS ONLY GIVEN AT THE END OF THE EPISODE. W/ MONTE-CARLO, WE IMMEDIATELY PROPAGATE THAT INFO BACK TO OUR ESTIMATE OF $V_\pi(s)$, BECAUSE WE CONSIDER FULL RETURNS G_t.

  HOWEVER, W/ TD IT MIGHT TAKE L SAMPLES OVER THAT TRAJECTORY TO PROPAGATE BACK THE REWARD TO STATE S.

- MC IS EPISODIC, WHILE TD ISN'T

- MC updates only after every episode. TD after each step.
- MC does not use Markov-assumption. TD does.

Batch MC and TD learning / Data efficiency

Batch updating.

- Suppose only finite amount of experience
  10 episodes / 1000 time steps.
- Process batch ⇒ update value ⇒ ... ⇒ converge!
- We will go through the batch, apply MC or TD, but will only update the value function at the end of the batch. Repeat until convergence.

But, what do these algorithms converge to in the batch settings?

- Batch MC converges to V_θ that fits the actual data the best. I.e. $V_\theta(s) = \frac{1}{m}\sum_{i=1}^{m} G_i$

$$V_\theta(s) = \underset{y}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{m}(G_i - y)^2$$

Returns seen for state s

↳ MSE. (V_θ minimizes MSE on data)

- Batch TD converges to the value-function of the maximum-likelihood model for the MDP.