CS285 Learning Notes on Reinforcement Learning

Created: 2024-10-12 09:40

Markov Chain and Markov Decision Process

Markov Chain

- **Definition:** A Markov Chain is a stochastic process that transitions from one state to another within a state space, where the probability of each state depends only on the previous state (Markov property).
- **Components:**
  - **State Space (S):** The set of all possible states s_i, which can be discrete (e.g., positions on a chessboard) or continuous (e.g., positions in physical space).
  - **Transition Operator (T):**
    - **Definition:** Represents the probabilities of moving from one state to another.
    - **Notation:** $T_{ij} = p(s_{t+1} = s_j \mid s_t = s_i)$, the probability of transitioning from state s_i to state s_j.
    - **Operator Property:** T acts on the state distribution μ_t to produce the next state distribution μ_{t+1}:

$$\mu_{t+1} = T\mu_t$$

    - **Interpretation:** If you know the distribution of states at time t, applying T gives you the distribution at time $t+1$.

Markov Decision Process (MDP)

- **Definition:** An MDP extends a Markov Chain by incorporating actions and rewards, modeling decision-making in stochastic environments.
- **Components:** $M = (S, A, T, r)$
  - **State Space (S):** Set of possible states.
  - **Action Space (A):** Set of possible actions the agent can take.
  - **Transition Function (T):**
    - **Definition:** $T(s_{t+1} \mid s_t, a_t) = p(s_{t+1} \mid s_t, a_t)$, the probability of transitioning to state s_{t+1} given state s_t and action a_t.

        - **Tensor Representation:** For discrete states and actions, T can be represented as a tensor of shape $|S| \times |A| \times |S|$.
    - **Reward Function (r):**
        - **Definition:** $r(s_t, a_t)$ provides the immediate reward for taking action a_t in state s_t.
        - **Purpose:** Guides the agent toward desirable outcomes.
- **State Distribution Update:**
    - **Policy (π):** A strategy defining the probability of taking action a in state s, denoted $\pi(a \mid s)$.
    - **Update Equation:**

$$\mu_{t+1}(s') = \sum_{s \in S} \sum_{a \in A} T(s' \mid s, a) \mu_t(s) \pi(a \mid s)$$

        - **Explanation:** The probability of being in state s' at time $t+1$ depends on all possible transitions from states s to s' via actions a, weighted by the probabilities $\mu_t(s)$ and $\pi(a \mid s)$.

Partially Observable Markov Decision Process (POMDP)

- **Definition:** A POMDP generalizes an MDP by accounting for situations where the agent cannot fully observe the underlying state.
- **Components:** $M = (S, A, T, r, O, \Omega)$
    - **Observation Space (O):** Set of possible observations the agent can receive.
    - **Emission Probability (Ω):**
        - **Definition:** $\Omega(o_t \mid s_t) = p(o_t \mid s_t)$, the probability of observing o_t given the true state s_t.
        - **Purpose:** Models uncertainty in perception, allowing the agent to make decisions based on observations rather than true states.

The Goal of Reinforcement Learning

- **Objective:** To find an optimal policy π^* that maximizes the expected cumulative reward over time.
- **Trajectory Probability:**
    - **Definition:** A trajectory $\tau = (s_0, a_0, s_1, a_1, \ldots)$ is a sequence of states and actions.
    - **Probability under Policy π:**

$$p_\pi(\tau) = p(s_0) \prod_{t=0}^{T-1} \pi(a_t \mid s_t) T(s_{t+1} \mid s_t, a_t)$$

- **Components:**
  - $p(s_0)$: Initial state distribution.
  - $\pi(a_t \mid s_t)$: Policy probability of action a_t in state s_t.
  - $T(s_{t+1} \mid s_t, a_t)$: Transition probability to state s_{t+1}.

Chain Rule of Probability

- **Definition:** The chain rule of probability allows us to express the joint probability of a sequence of random variables as a product of conditional probabilities.
- **Mathematical Formulation:**

$$p(x_1, x_2, \ldots, x_n) = p(x_1) \prod_{i=2}^{n} p(x_i \mid x_1, x_2, \ldots, x_{i-1})$$

- **Application in Reinforcement Learning:**
  - **Trajectory Probability:**

    In reinforcement learning, we often deal with the probability of a trajectory $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T, a_T)$ under a policy π.
    - **Using the Chain Rule:**

      The joint probability $p_\pi(\tau)$ can be decomposed using the chain rule:

$$p_\pi(\tau) = p(s_0) p(a_0 \mid s_0) p(s_1 \mid s_0, a_0) p(a_1 \mid s_1) \ldots p(s_T \mid s_{T-1}, a_{T-1}) p(a_T \mid s_T)$$

    - **Simplification Using the Markov Property:**

      Due to the Markov property (future states depend only on the current state and action), the conditional probabilities simplify:

$$p(s_{t+1} \mid s_0, a_0, s_1, a_1, \ldots, s_t, a_t) = p(s_{t+1} \mid s_t, a_t)$$

    - **Final Expression:**

      Therefore, the trajectory probability becomes:

$$p_\pi(\tau) = p(s_0) \prod_{t=0}^{T} \pi(a_t \mid s_t) T(s_{t+1} \mid s_t, a_t)$$

      - **Explanation:**
        - $p(s_0)$: Probability of starting in state s_0.

    - $\pi(a_t \mid s_t)$: Policy's probability of taking action a_t in state s_t.
    - $T(s_{t+1} \mid s_t, a_t)$: Transition probability to state s_{t+1} given s_t and a_t.
- **Understanding with an Example:**
  - **Suppose:**
    - The initial state s_0 is drawn from $p(s_0)$.
    - At each time t, the agent selects action a_t based on $\pi(a_t \mid s_t)$.
    - The environment transitions to s_{t+1} according to $T(s_{t+1} \mid s_t, a_t)$.
  - **Using the Chain Rule:**
    - The joint probability of (s_0, a_0, s_1, a_1) is:

$$p(s_0)\pi(a_0 \mid s_0)T(s_1 \mid s_0, a_0)\pi(a_1 \mid s_1)$$

    - This pattern continues for the entire trajectory.
- **Key Takeaways:**
  - The chain rule of probability is essential for decomposing complex joint probabilities into manageable conditional probabilities.
  - In reinforcement learning, it allows us to compute the likelihood of entire trajectories under a given policy by sequentially multiplying the probabilities of states and actions.
- **Relation to the Markov Property:**
  - The Markov property simplifies the chain rule by reducing dependencies to only the current state and action.
  - This simplification is crucial for computational tractability in reinforcement learning algorithms.

Infinite Horizon and Stationary Distribution

- **Stationary Distribution (μ):**
  - **Definition:** A distribution over states that remains unchanged under the transition dynamics.

$$\mu = T\mu$$

  - **Eigenvector Interpretation:** μ is an eigenvector of T with eigenvalue 1.
- **Existence and Uniqueness:**
  - **Conditions:** The stationary distribution exists and is unique if the Markov chain is **ergodic** (irreducible and aperiodic).
  - **Ergodicity:**
    - **Irreducibility:** Every state can be reached from any other state.

- **Aperiodicity:** The system does not cycle in a fixed period.
- **Importance in RL:** Understanding the long-term behavior of the state distribution is crucial for policies evaluated over an infinite horizon.

Expectations and Optimizations

- **Expectation Properties:**
  - **Linearity:** The expectation of a sum is the sum of the expectations.

$$\mathbb{E}\left[\sum_t X_t\right] = \sum_t \mathbb{E}[X_t]$$

  - **Smoothness:** While individual rewards $r(s_t, a_t)$ may be non-smooth, their expected values over trajectories can be smooth functions, enabling gradient-based optimization.
- **Optimization Objective:**
  - **Policy Optimization:** Adjust the policy π to maximize $J(\pi)$ using methods like gradient ascent.

Definition of Q-function and Value Function

- **Expanding $J(\pi)$:**
  - **Nested Expectations:**

$$J(\pi) = \mathbb{E}_{s_0 \sim p(s_0)}\left[\mathbb{E}_{a_0 \sim \pi(a_0|s_0)}\left[r(s_0, a_0) + \mathbb{E}_{s_1 \sim T(s_1|s_0,a_0)}\left[\mathbb{E}_{a_1 \sim \pi(a_1|s_1)}\left[r(s_1, a_1) + \ldots\right]\right]\right]\right]$$

  - **Interpretation:** Shows how the expected return unfolds over time through a series of actions and states.
- **Q-function ($Q^\pi(s, a)$):**
  - **Definition:** The expected cumulative reward starting from state s, taking action a, and thereafter following policy π.

$$Q^\pi(s, a) = r(s, a) + \mathbb{E}_{s' \sim T(s'|s,a)}\left[V^\pi(s')\right]$$

  - **Usefulness:** Knowing $Q^\pi(s, a)$ allows for direct policy improvement by selecting actions that maximize Q.
- **Value Function ($V^\pi(s)$):**
  - **Definition:** The expected cumulative reward starting from state s and following policy π.

$$V^\pi(s) = \mathbb{E}_{a \sim \pi(a|s)}\left[Q^\pi(s, a)\right]$$

  - **Relationship to Q-function:** $V^\pi(s)$ is the expected value of $Q^\pi(s, a)$ over all possible actions at state s according to policy π.

- **Expressing $J(\pi)$ with Value Function:**
  - **Equation:**

$$J(\pi) = \mathbb{E}_{s_0 \sim p(s_0)} \left[V^\pi(s_0) \right]$$

  - **Interpretation:** The expected return is the expected value function at the initial state.

Example: Expanding J(π) over 3 Steps

Mathematical Expansion:

$$J(\pi) = \mathbb{E}_{s_0 \sim p(s_0)} \Big[\mathbb{E}_{a_0 \sim \pi(a_0|s_0)} \Big[r(s_0, a_0) + \mathbb{E}_{s_1 \sim T(s_1|s_0,a_0)} \Big[\mathbb{E}_{a_1 \sim \pi(a_1|s_1)} \Big[r(s_1, a_1) + \mathbb{E}_{s_2 \sim T(s_2|s_1,a_1)} \Big[\mathbb{E}_{a_2 \sim \pi(a_2|s_2}$$

Using Value and Q-functions:

At t = 2:

$$V^\pi(s_2) = \mathbb{E}_{a_2 \sim \pi(a_2|s_2)} \left[Q^\pi(s_2, a_2) \right] = \mathbb{E}_{a_2} \left[r(s_2, a_2) \right]$$

At t = 1:

$$Q^\pi(s_1, a_1) = r(s_1, a_1) + \mathbb{E}_{s_2 \sim T(s_2|s_1,a_1)} \left[V^\pi(s_2) \right]$$

At t = 0:

$$V^\pi(s_0) = \mathbb{E}_{a_0 \sim \pi(a_0|s_0)} \left[Q^\pi(s_0, a_0) \right]$$

Expressing J(π):

$$J(\pi) = \mathbb{E}_{s_0 \sim p(s_0)} \left[V^\pi(s_0) \right]$$

Important Ideas in Reinforcement Learning

- **Idea 1: Policy Improvement with Q-function**
  - **Concept:** If you have a policy π and know its Q-function $Q^\pi(s,a)$, you can create a new policy π' that is at least as good by choosing actions that maximize $Q^\pi(s,a)$.
    - **Improved Policy:**

$$\pi'(a \mid s) = \begin{cases} 1, & \text{if } a = \arg\max_{a'} Q^\pi(s, a') \\ 0, & \text{otherwise} \end{cases}$$

  - **Result:** π' will perform at least as well as π, potentially better.
- **Idea 2: Policy Gradient Intuition**
  - **Concept:** Adjust the policy to increase the probability of actions that are better than average.
  - **Advantage Function ($A^\pi(s,a)$):**
    - **Definition:**

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$$

- **Interpretation:** Measures how much better action a is compared to the average action at state s.

- **Policy Adjustment:**
  - Increase $\pi(a \mid s)$ if $A^{\pi}(s,a) > 0$ (action is better than average).
  - Decrease $\pi(a \mid s)$ if $A^{\pi}(s,a) < 0$ (action is worse than average).
- **Outcome:** Over time, the policy improves by favoring better-than-average actions.

Key Takeaways

- **Understanding MDPs:** Grasp the components and how they model decision-making.
- **Goal of RL:** Maximize the expected cumulative reward by finding the optimal policy.
- **Chain Rule of Probability:** Essential for computing trajectory probabilities and understanding how policies influence outcomes.
- **Value Functions:** Central to evaluating and improving policies.
- **Policy Improvement Strategies:** Utilize the Q-function and advantage function to enhance policy performance.

References

- CS275: Lecture 4, Part 1
- CS285: Lecture 4, Part 3