Reinforcement Learning Algorithms

Lecturer Name

March 19, 2025

Q-learning

Motivation: Learn optimal action values through interaction in complex environments (e.g., game playing).
Problem: Achieving optimal decision-making when the environment is unknown.
Intuitive Solution: Use the Q-value function to evaluate the expected returns of actions.

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right] \tag{1}$$

Where:

- $Q(s,a)$: Action-value function representing the expected return for taking action a in state s.
- α: Learning rate (0 ¡ α 1), controls the magnitude of learning.
- r: Immediate reward received after transitioning to state s'.
- γ: Discount factor (0 γ ¡ 1), determining the importance of future rewards.

Deep Q-Networks (DQN)

Motivation: Extend Q-learning to manage high-dimensional inputs like images (e.g., video games).
Problem: Traditional Q-learning struggles to handle raw pixel inputs efficiently.
Intuitive Solution: Use deep neural networks to approximate the Q-value function.

$$Q(s, a; \theta) = r + \gamma \max_{a'} Q(s', a'; \theta') \tag{2}$$

Where θ represents the neural network's weights trained on the input data.

Policy Gradient Methods

Motivation: Directly optimize policy functions for continuous action spaces.
Problem: Value-based methods struggle with high-dimensional continuous actions.
Intuitive Solution: Adjust policy parameters to maximize expected rewards.

$$\nabla J(\theta) = \mathbb{E}_{s_t \sim \rho_\theta} \left[\nabla \log \pi_\theta(s_t, a_t) A(s_t, a_t) \right] \tag{3}$$

Where:

- $J(\theta)$: The objective for the policy to maximize.
- $A(s_t, a_t)$: Advantage function, measuring how much better an action performs compared to a baseline.

Actor-Critic

Motivation: Combine strengths of policy and value-based approaches for stable learning.
Problem: Balancing exploration and exploitation is essential for efficiency.
Intuitive Solution: The actor updates the policy while the critic evaluates the actions taken.

$$\theta \leftarrow \theta + \alpha \nabla J(\theta) \tag{4}$$

$$w \leftarrow w + \beta \delta_t \nabla V(s_t; w) \tag{5}$$

Where $\delta_t = r_t + \gamma V(s_{t+1}; w) - V(s_t; w)$ is the temporal difference error, indicating the difference between actual and estimated returns.

Proximal Policy Optimization (PPO)

Motivation: Ensure stability during policy updates through constrained learning.
Problem: Large policy updates can result in performance degradation.
Intuitive Solution: Use a clipped objective function to limit the policy changes.

$$L^{CLIP}(\theta) = \mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] \tag{6}$$

Where $r_t(\theta)$ compares new policy against the old one, ensuring gradual updates.

Trust Region Policy Optimization (TRPO)

Motivation: Limit harmful updates by constraining policy changes.
Problem: Aggressive updates can lead to significant performance drops.
Intuitive Solution: Optimize updates within a constrained trust region.

$$\max_{\theta} \mathbb{E}\left[\hat{A}_t \pi_\theta(a_t|s_t)\right] \tag{7}$$

Subject to:

$$\mathbb{E}\left[D_{KL}(\pi_{\theta_{old}}||\pi_\theta)\right] \leq \delta \tag{8}$$

Where δ is a tuning parameter determining allowable changes.

Asynchronous Actor-Critic Agents (A3C)

Motivation: Speed up training by using multiple agents in parallel.
Problem: Slow learning in traditional single-agent environments.
Intuitive Solution: Employ multiple agents that explore concurrent environments, improving data diversity.

$$\theta_{t+1} = \theta_t + \alpha \nabla J(\theta_t) + \beta \Delta_t \tag{9}$$

Where Δ_t captures the updates from multiple environments, leading to faster convergence.

Dueling Network Architectures

Motivation: Improve learning efficiency by separating value and advantage functions.
Problem: Action selection can be inefficient in environments with many available actions.
Intuitive Solution: Model the state value and action advantages separately.

$$Q(s,a) = V(s) + \left(A(s,a) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s,a') \right) \tag{10}$$

Where $V(s)$ is the value function, and $A(s,a)$ captures the advantage of taking action a in state s.

Hierarchical Reinforcement Learning (HRL)

Motivation: Simplify complex decision-making by breaking tasks down into sub-tasks.
Problem: Managing long-term dependencies complicates learning processes.
Intuitive Solution: Use a hierarchy of policies to handle different levels of abstraction.

$$R(s_t) = \sum_{k=1}^{K} r_k(s_t) \tag{11}$$

Where K represents the number of hierarchical levels, enabling structured learning across tasks.

Multi-Agent Reinforcement Learning (MARL)

Motivation: Facilitate cooperative behaviors among multiple agents.
Problem: Coordination among agents introduces additional complexities.
Intuitive Solution: Implement joint action policies that enhance collaboration.

$$Q_{\text{joint}}(s_t, a_1, a_2) = \sum_{i=1}^{N} Q_i(s_t, a_i) \tag{12}$$

Where N is the total number of cooperating agents within the environment.

Inverse Reinforcement Learning (IRL)

Motivation: Infer underlying reward functions from expert behavior demonstrations.
Problem: Learning the reward structure is often challenging and complex.
Intuitive Solution: Replicate observed expert behavior to infer reward functions.

$$R(s, a) = \log\left(\sum_{s'} P(s'|s, a) \cdot \pi^*(s')\right) \tag{13}$$

Where π^* represents the expert policy that the agent aims to emulate.

Distributional Reinforcement Learning

Motivation: Capture the full distribution of possible returns instead of focusing solely on expected returns.
Problem: Ignoring variability in returns can hinder optimal policy learning.
Intuitive Solution: Learn the complete distribution of returns from state-action pairs.

$$Z(s, a) = P(Q(s, a)|s, a) \tag{14}$$

Where Z is the distribution of returns, providing a deeper understanding of possible outcomes.

Sparse-Sampling for Reinforcement Learning

Motivation: Enhance agent efficiency in environments with infrequent rewards.
Problem: Rare rewards can hamper learning progression.
Intuitive Solution: Prioritize actions that yield the most informative feedback.

$$\text{Reward}(s_t) \propto P(a_t|s_t) \tag{15}$$

Focusing agent attention on actions that generate valuable information.

Meta-Reinforcement Learning

Motivation: Enable agents to quickly learn new tasks based on past experiences.
Problem: Retraining for each new task is often inefficient.
Intuitive Solution: Optimize learning strategies to minimize adaptation time to new challenges.

$$M(\theta_{new}) = \max_{\theta_{old}} \sum_{i}^{n} R_i(\theta_{new}) \tag{16}$$

Improving learning efficiency through prior knowledge application.

Apprenticeship Learning

Motivation: Reduce exploration time by imitating expert behavior.
Problem: Exhaustive exploration can be impractical in complex scenarios.
Intuitive Solution: Utilize expert demonstrations to guide the agent's learning process.

$$L(\theta) = \sum_{t=0}^{T} ||\pi(a|s;\theta) - \pi_{expert}(a|s)||^2 \tag{17}$$

Rapidly refining the agent's policy by mimicking expert decisions.

Model-Based Reinforcement Learning

Motivation: Build predictive models of the environment to simulate outcomes and optimize actions.
Problem: A precise model may not always be available or feasible.
Intuitive Solution: Construct models based on interactions to forecast and plan future actions.

$$\hat{R}(s, a) = E_{s'}[R(s, a, s')] \tag{18}$$

Where $\hat{R}$ denotes the empirical return derived from the learned model.

Q-Learning with Function Approximation

Motivation: Generalize learning across similar states by approximating the Q-value function.
Problem: Traditional Q-learning struggles with large state spaces.
Intuitive Solution: Use function approximators (e.g., neural networks) to estimate Q-values effectively.

$$Q(s, a) \approx f(s, a; \theta) \tag{19}$$

Where f serves as the function approximator and θ are its parameters.

Continuous Action Space Reinforcement Learning

Motivation: Enhance agent decisions in environments with infinite action options.
Problem: Discrete action methods become insufficient with continuous action requirements.
Intuitive Solution: Employ policy gradients or similar methods to derive actions from continuous spaces.

$$\nabla J(\theta) = \mathbb{E}\left[\nabla \log \pi_\theta(s_t, a_t) A(s_t, a_t)\right] \tag{20}$$

Where continuous action execution is enabled through adapted gradient-based approaches.

Reward Shaping

Motivation: Modify rewards to help guide learning and speed up the process.
Problem: Sparse reward signals can significantly hinder effective learning.
Intuitive Solution: Introduce intermediate rewards for achievable tasks or desired behaviors.

$$R'(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a)R(s') \tag{21}$$

Enhancing learning motivation by assigning value to sub-goals.

Curriculum Learning

Motivation: Organize the learning process by presenting tasks in a progressive manner.
Problem: Immediate exposure to complex challenges can overwhelm learners.
Intuitive Solution: Begin with simpler tasks, gradually increasing difficulty.

$$\text{Task}_{\text{complex}} \rightarrow \text{Task}_{\text{simple}} \tag{22}$$

Facilitating skill acquisition through structured learning.

Off-Policy Learning

Motivation: Learn policies using data generated by alternative behaviors.
Problem: Efficient data utilization is necessary to improve learning outcomes.
Intuitive Solution: Apply importance sampling to weigh updates based on the difference between behaviors.

$$w_t = \frac{\pi_\theta(a_t|s_t)}{\mu(a_t|s_t)} \tag{23}$$

Where μ represents the behavior policy from which samples are derived.

Exploration Methods (Epsilon-Greedy)

Motivation: Balance exploration and exploitation in decision-making processes.
Problem: Agents risk stagnation by not exploring sufficient options.
Intuitive Solution: With probability ϵ, choose a random action; otherwise, adopt the best-known action.

$$a = \begin{cases} \text{random action} & \text{with probability } \epsilon \\ \text{argmax } Q(s, a) & \text{otherwise} \end{cases} \tag{24}$$

This mechanism encourages both exploration of new strategies and exploitation of known results.

Soft Actor-Critic (SAC)

Motivation: Utilize maximum entropy reinforcement learning to encourage exploration.
Problem: Value functions can underestimate returns, especially in uncertain environments.
Intuitive Solution: Integrate entropy maximization into policy updates, balancing exploration and exploitation.

$$\mathcal{L}(\theta) = \mathbb{E}\left[\log \pi_\theta(a|s) - Q(s,a)\right] \tag{25}$$

This approach results in more robust and exploratory agent behavior.

Hindsight Experience Replay (HER)

Motivation: Improve learning efficiency in environments with sparse rewards.
Problem: Many trajectories may lead to failures without clear learning signals.
Intuitive Solution: Learn from failures by treating them as successes towards different goals.

$$\mathcal{L}^H = \mathbb{E}_{(s,a,s')}[r + \gamma V(s')] \tag{26}$$

Where s' corresponds to a desired goal state to facilitate learning.

Transfer Learning in Reinforcement Learning

Motivation: Transfer knowledge across tasks to improve learning speed.
Problem: Extensive retraining for similar tasks can be inefficient.
Intuitive Solution: Leverage previously learned knowledge to assist in new tasks.

$$Q_{\text{new}}(s, a) \approx Q_{\text{previous}}(s, a) + \gamma \sum_{s'} P(s'|s, a) Q_{\text{previous}}(s', a') \quad (27)$$

Sharing knowledge among similar tasks to enhance learning efficiency.

Action Constraints in RL

Motivation: Improve performance in environments with action limitations.
Problem: Agents may execute invalid or suboptimal actions when not constrained.
Intuitive Solution: Enforce constraints on action selection to adhere to allowed actions.

$$a' \in \mathcal{A}_{\text{valid}} \tag{28}$$

Where $\mathcal{A}_{\text{valid}}$ denotes those actions that are permissible within the given constraints.

Safe Reinforcement Learning

Motivation: Ensure agents do not engage in risky or harmful behaviors during their learning process.
Problem: High-stakes environments can result in detrimental outcomes when agents fail.
Intuitive Solution: Incorporate safety constraints in the reward structure and limit actions accordingly.

$$r_{\text{safe}}(s, a) = \begin{cases} R(s, a) & \text{if action } a \text{ is safe} \\ -\infty & \text{if action } a \text{ is unsafe} \end{cases} \tag{29}$$

Promoting safer learning practices within constrained environments.

Exploration-Exploitation Dilemma Strategies

Motivation: Achieve an effective balance between exploring new actions and exploiting known strategies.
Problem: Focusing solely on either aspect may lead to suboptimal performance.
Intuitive Solution: Employ techniques such as Upper Confidence Bound (UCB) to manage exploration limits.

$$\text{UCB}(a) = \bar{Q}(a) + \sqrt{\frac{2 \ln n}{N(a)}} \tag{30}$$

Where n is the total number of actions taken, and $N(a)$ indicates the number of times action a has been chosen.

Non-Markovian Reinforcement Learning

Motivation: Address challenges in environments that do not conform to Markov properties.
Problem: Memory limitations can impede effective decision-making in dynamic scenarios.
Intuitive Solution: Incorporate historical context in the decision-making process.

$$V(s_t|\text{history}) = E[R|s_t, \text{history}] \tag{31}$$

Utilizing historical events to enhance predictions of future rewards.

Lecture Summary

Key Takeaways:

- Understanding various reinforcement learning algorithms enhances problem-solving in AI through practical applications.
- Integrating theoretical approaches with real-world scenarios fosters robust learning strategies.
- Continuous advancements and innovations in reinforcement learning open new avenues for exploration and application in various domains, such as robotics.