Deep Reinforcement Learning (Sp25)

Instructor: Dr. Mohammad Hossein Rohban

Summary of Lecture 11: Multi-armed Bandits

Summarized By: Behnia Soleymani

- **Planning with Uncertainty**
  - **Challenge**: Model inaccuracies lead to uncertainty in predictions.
  - **Solutions**:
    - Use a **distribution over deterministic models** to capture uncertainty.
    - Alternative techniques: **moment matching**, **Bayesian Neural Networks (BNNs)**, or posterior estimation for uncertainty quantification.
- **Model-Based RL with Ensembles**
  - **Example**: Ensemble models (multiple models) outperform model-free methods after **40k steps** (10 minutes of real time).
  - **Advantage**: Reduces overfitting and improves robustness via model diversity.
- **Policy Optimization Challenges**
  - **Problem with Backpropagating into Policies**:
    - **Parameter Sensitivity**: Similar to shooting methods, leading to unstable training.
    - **Vanishing/Exploding Gradients**: Analogous to training long **RNNs with BPTT** (no LSTM-like solutions here).
    - **No Dynamic Programming**: Policy parameters couple all time steps, making second-order methods (e.g., LQR) inapplicable.
  - **Workaround**: Use **derivative-free RL algorithms** (model-free) with synthetic samples generated by the model.
- **Curse of Long Rollouts**
  - **Issue**: Errors accumulate over long simulated trajectories, degrading performance.
  - **Solution**: Use **short rollouts** (e.g., 1–5 steps) to mitigate error accumulation.
- **Dyna Architecture**:
  - **Hybrid Approach**: Combines **model-free learning** (e.g., Q-learning) with **model-based planning**.
  - **Steps**:
    - Collect real-world experience (s, a, s', r).
    - Learn a dynamics model $\hat{p}(s'|s, a)$.
    - Generate synthetic trajectories via rollouts.
    - Update the policy using both real and simulated data.
  - **Advantage**: Reduces real-world interaction time by leveraging simulated data.

- **Key Equation**:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \mathbb{E}_{s',r}\left[r + \max_{a'} Q(s',a') - Q(s,a)\right]$$

- **Advanced Methods**:
  - **Model-Based Policy Optimization (MBPO)**: Limits rollout length to balance error accumulation.
  - **Model-Based Value Expansion (MVE)**: Uses short model rollouts to improve value estimates.
  - **Key Workflow**:
    - Collect real data.
    - Train dynamics model.
    - Generate short rollouts.
    - Update policy with model-free RL (e.g., policy gradient).
- **Multi-armed Bandits (MAB)**
  - **Definition**:
    - A **sequential decision-making problem** where an agent repeatedly chooses between K actions ("arms") with unknown reward distributions.
    - The goal is to **maximize cumulative reward** over time by learning which arms yield the highest rewards.
  - **Core Challenge**:
    - **Exploration vs. Exploitation Trade-off**:
      - **Exploration**: Gathering information about arms with uncertain rewards.
      - **Exploitation**: Leveraging known information to maximize immediate rewards.
    - The agent must balance these two to avoid suboptimal long-term performance.
  - **Action Value ($q_*(a)$)**:
    - The **expected reward** of selecting arm a, defined as:

$$q_*(a) = \mathbb{E}[R_t \mid A_t = a]$$

      where R_t is the reward at time t, and A_t is the action taken.
    - The agent's goal is to estimate $q_*(a)$ for each arm and select the arm with the highest value.
  - **Key Characteristics**:
    - **Stateless**: No state transitions or state space; decisions are independent.

    - **Immediate Rewards**: Rewards depend only on the chosen arm.
    - **Foundational**: Basis for complex RL algorithms.

- **Differences Between MAB and Normal RL Problems**
  - **State Space**:
    - **MAB**: Stateless; no transitions.
    - **Normal RL**: Involves state transitions and sequential decision-making.
  - **Reward Structure**:
    - **MAB**: Immediate rewards.
    - **Normal RL**: Delayed rewards (e.g., discounted cumulative rewards).
  - **Exploration vs. Exploitation**:
    - **MAB**: Focuses entirely on the trade-off for fixed arms.
    - **Normal RL**: Balances exploration-exploitation while learning state-action policies.
  - **Complexity**:
    - **MAB**: Single-step decisions.
    - **Normal RL**: Multi-step planning and state impact analysis.
  - **Applications**:
    - **MAB**: Clinical trials, A/B testing, online advertising.
    - **Normal RL**: Robotics, autonomous driving.
- **Estimating** $q_*(a)$
  - **Sample-Average Method**:
    - Estimate $q_*(a)$ by averaging rewards:

$$Q_t(a) = \frac{\text{Sum of rewards when } a \text{ is chosen}}{\text{Number of times } a \text{ is chosen}}$$

    - Converges to $q_*(a)$ as trials increase.
- **Clinical Trials as a Bandit Problem**
  - **Analogy**:
    - Arms = treatments; rewards = patient outcomes (e.g., survival).
    - Goal: Identify optimal treatments while minimizing exposure to inferior options.

- **Adaptive Design**:
  * Dynamically assign patients to better-performing treatments as data is collected.
  * Adapts unlike traditional fixed randomization.
- **Advantages**:
  * **Ethical**: Reduces exposure to inferior treatments.
  * **Efficiency**: Accelerates optimal treatment identification.
- **Challenges**:
  * Non-stationary rewards, delayed outcomes, and small sample sizes.