Reinforcement Learning

Satyanarayana Gajjarapu
AI24BTECH11009
Department of Artificial Intelligence
ai24btech11009@iith.ac.in

1 Introduction and Examples

Reinforcement learning (RL) is a form of learning which includes how to map situations to actions that maximizes numerical reward signal. The agent is not told which actions to take, but instead it discovers which actions yield the most reward by trying them. A learning agent must be able to sense the environment and also should have goals about it. RL is the third machine learning paradigm, different from supervised and unsupervised learning. The main challenge in reinforcement learning is the trade-off between **exploration** and **exploitation**. The agent has to exploit what it has already experienced in order to obtain reward, but it also has to explore in order to make better action selections in the future. Unlike supervised and unsupervised learning, RL tackles the full problem of a goal-directed agent interacting with an uncertain environment. Reinforcement learning research is certainly part of the swing back toward simpler and fewer general principles of artificial intelligence.

Some examples to understand reinforcement learning are:

- An adaptive controller adjusts parameters of a petroleum refinery's operation in real time.
- A gazelle calf struggles to its feet minutes after being born. Half an hour later it is running at 20 miles per hour.
- A mobile robot decides whether it should enter a new room in search of more trash to collect or start trying to find its way back to its battery recharging station.

2 Elements

Four main elements of a reinforcement learning system are $policy$, $reward\ signal$, $value\ function$, and $model\ of\ the\ environment$. A policy defines the agent's way of behaving at a given time. It is a mapping from perceived states of the environment to actions to be taken when in those states. A reward signal defines the goal of a reinforcement learning problem. On each time step, the environment sends to the agent a single number called the reward. The agent's sole objective is to maximize it's total reward. Analogously rewards might be thought as experiences of pleasure or pain. A value function specifies what is good in the long run. The value of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state. The model of the environment is something that mimics the behavior of the environment. Model based methods are used for solving reinforcement learning problems and planning whereas model free methods are explicitly trial-and-error learners.

Preprint. Under review.

3 Limitations and Scope

Reinforcement learning centers around the cncept of state as input for decision-making, emphasizing value function estimation. Evolutionary methods, while solving RL problems through optimization, don't interact with the environment or learn from individual state-action experiences. Though effective in certain cases, they are generally less efficient than methods that learn through interaction.

4 Tic-Tac-Toe

Reinforcement learning in tic-tac-toe utilizes a value function to estimate the probability of winning from any given state. The value function is updated as the game progresses using the update rule:

$$V(S_t) \leftarrow V(S_t) + \alpha[V(S_{t+1}) - V(S_t)]$$

Here, $V(S_t)$ is the value of the current state, $V(S_{t+1})$ is the value of the next state, and α is the step-size parameter, controlling the learning rate. This update rule is an example of a temporal-difference learning method. By adjusting the values through multiple games, the agent improves its strategy over time, converging toward optimal play.

Evolutionary methods evaluate entire policies based on their win frequencies rather than individual state transitions. While both approaches seek to improve gameplay, reinforcement learning is more adaptable to different complexities, such as large state spaces, continuous environments, or partial observability of states and opponents. Moreover, reinforcement learning naturally incorporates exploratory moves to ensure that a variety of state experiences are encountered during training, which is a significant advantage in dynamic or complex game settings.

A model can be improved by the following exercises

1. **self play**: playing against itself
2. **symmetries**: taking advantage of symmetries in the game
3. **greedy play**: move made to the position which is rated best
4. **exploration**: making exploratory moves and altering step size parameter.