ECE-6930-004 Lecture Notes
September 5, 2025

1 Markov Decision Process

Formulation

- *States* s - beginning with initial state s_0.
- *Actions* a - each state s has actions $A(s)$ available from it
- *Transition Model* $P(s'|s,a)$ - probability of moving to state s' given the state-action pair s, a.

2 The bellman equation

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s,a)U(s')$$

This gives us the optimal action given our current state. We can see this is recursive as it requires computing the utility of the state being considered s'. This allows us to make decisions which are optimal in an uncertain system. We do not have to have a perfect perception of the environment to determine our utilities.

2.1 Value vs Policy Iteration

value iteration

- state out with every $U(s) = 0$
- iterate until convergence
- at each iteration, update the value at the state using the bellman equation

policy iteration

- state with initial policy π_0 and alternate between policy evaluation and improvement.
- Evaluation involves calculating $U^{\pi_i}(s)$ for all states $s \in S$.
- Improvement involves calculating a new policy π_{i+1} based on the updated utilities.

This mutates our utility function to

$$U^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi(s))U^{\pi}(s')$$

Where $\pi(s)$ is fixed; therefore, $P(s'|s,\pi(s))$ is an $s' \times s$ matrix, therefore we can solve a linear equation to get $U^{\pi}(s)$.

This lets us evaluate the utility of the current policy. The next step is to use this further mutation

$$U^{\pi}(s) = \operatorname*{argmax}_{a \in A(s)} \sum_{s'} P(s'|s,a)U^{\pi}(s')$$

This allows us to improve our policy based on the optimal action given our current state and possible future statues.

September 8, 2025

3 On TicTacToe

In 1962, the first Tic Tac Toe simulation created and performed by hand by Donald Michie. We could just pull data from human moves, but a human may be a weak player or not represent the desired player.

Donald Michie's Machine (MENACE)

- for each board state where cross is on-move, have a "match box"labeled with that state. Each match box has a number of colored beads in it, each color represents a valid move for cross on that board.
- To make a move, pick up box with label of current state, shake it, pick random bead. Check color and make that move
- new state, wait for human counter move. Repeat until terminating condition.
- If MENACE loses, remove any beads used in sequence. If MENACE wins, add three extra copies of each bead used in that game to the appropriate match boxes. In a draw, add one extra set of beads.

This is somewhat analogous to modern Q-learning.

September 10, 2025
