CS 181 LECTURE 4/9/24

Scribe Notes

Contents

1 Admin

Midterm II is in 2 weeks, and HW 6 is out on Friday.

2 Where We Are in the Course

We are in the last phase of the course. For the next four lectures, we will be covering decision-making.

Today, we start with the idea of planning. Then, we will talk about reinforcement learning. Lastly, we will talk about Multi-Agent systems - what if there are multiple agents trying to make decisions together or competitively. The goal is to learn intuition and conceptual grounding.

3 Videos

The following videos are examples of reinforcement learning.
Google DeepMind's Deep Q-Learning playing Atari Breakout

The model is taking pixels off the screen, learning to play a game. After about 2 hours of training, the model reaches human-level of performance. Later, the model learns a strategy that is much less common for human players to succeed at. The strategy is to get the ball at the top part of the game. This showed that computation was catching up to theory and the model was actually able to learn to play the game.

Another moment in machine learning is Alpha Go. The model was able to learn by playing loads and loads of games. The model memorized games of experts. In Atari and the game of Go, you can lose a million times and get lots of experience relatively quickly.

Learning to Walk in the Real World in 1 Hour (No Simulator)

The robot learns to walk. Thirty minutes in, the robot begins to walk a bit. Then, at one hour, the robot is moving. During the perturbation period, the robot is hit with a stick and is still able to walk, proving that it did learn to walk.

4 New Notation

There is an agent sending actions into the world. The world is going to send back two things: 1) observations (ex. pixels on the screen, the joint angles of the robot) and 2) reward. The reward aspect is what makes this reinforcement learning. In a sense, reinforcement learning is intuitive because this is what humans do. For example, giving a child a piece of candy for cleaning up after themselves. Other agents, like healthcare types, may have more complicated reward structures.

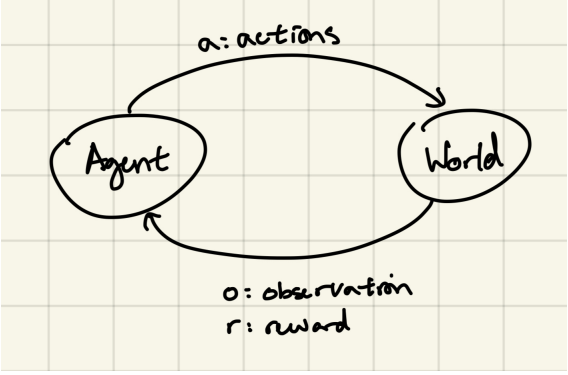

Figure 1: Reinforcement Learning.

Our goal is to maximize over the sum of expected discounted rewards:

$$\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t]$$

In the above expression:

- r_t is the reward at time t (we also write reward as a function of the state and the action, $r(s, a)$)
- γ is the discount in $[0, 1)$
- the expectation is over the world and our policy
- the max is over our policy

If you have a discount close to 1, then you are considering the rewards far into the future. This means you are confident that you will survive into the future. If you have a discount close to 0, you believe that you will not stay in the game for long. In other words, the discount can be interpreted as the probability that the agent will survive from t to $t + 1$.

In our current framework, rewards come in at every time step but what we care about is the aggregate reward. This is not the only reward framework but this works well for our computations. For example, we have exponential discounting but you could put γ in a different function to show when rewards matter more.

The reward and the discount are the **subjective parts** of planning because we pick the reward and discount factor. Then, there are the parts that the world defines. More notation/vocabulary:

- actions
- observations
- history h, which is a series of actions, observations, and rewards, ie. $\{a_0, o_0, r_0, a_1, o_1, r_1\}$
- state, which summarizes a history

The history is important when seeing what action is selected. For example, in the Atari game, just seeing where the ball is does not provide enough information. We need to see where the ball is coming from to know what direction the ball is moving in. The idea of states makes the world Markovian - we need to know that the ball is at a certain location, moving in a certain direction, but we do not need to know what happened before it. Therefore, part of the history is important but not all of it.

5 Markov Decision Process (MDP)

In an MDP, we are given the following:

$$\begin{aligned}
&\{S, A, T, \gamma, R\} \\
&S = \text{states} \\
&A = \text{actions} \\
&\gamma = \text{discount} \\
&T = \text{transition}, T(s'|s, a) \\
&R = \text{rewards}, r(s, a)
\end{aligned}$$

Our goal is to find a policy that goes from state to action, $\pi(s) = a$, such that we maximize the value of the policy, conditioned on the starting state s_0:

$$\max_{\pi} \mathbb{E}_{\pi,T}[\sum \gamma^t r_t | s = s_0]$$

"Planning" is also known as "solving an MDP." We are given how the world works. Knowing how the world works, we want to optimize decision making. In future lectures, we will no longer know T and we need to learn from history only.

Let $V^{\pi}(s_0)$ be the value of the policy given the starting state s_0.

$$\begin{aligned}
V^{\pi}(s_0) &= \mathbb{E}_{\pi,T}[\sum \gamma^t r_t | s = s_0] \\
&= \mathbb{E}[\gamma^0 r_0 + \sum_{t=1}^{\infty} \gamma^t r_t | s = s_0]
\end{aligned}$$

We know that $\gamma^0 r_0 = r(s_0, \pi(s_0))$. There is no randomness in this so we can pull it out of the expectation. We can also pull out one factor of γ out of the expectation. (We then reset time; we can think of r_1 as new r_0, r_2 as new r_1, etc. so our expectation will start from $t = 0$.)

$$\begin{aligned}
&= \gamma^0 r_0 + \gamma \mathbb{E}_{s_1} \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t | s = s_1] \\
&= r(s_0, \pi(s_0)) + \gamma \mathbb{E}_{s_1} \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t | s = s_1]
\end{aligned}$$

We also conditioned on s_1, which is the next state. We do not know which state we

will be at, so we need to take an additional expectation over s_1. Now, let us take that expectation explicitly:

$$= r(s_0, \pi(s_0)) + \gamma \sum_{s'} T(s'|s_0, \pi(s_0)) \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t | s_1 = s']$$
$$= \underbrace{r(s_0, \pi(s_0))}_{\text{current reward}} + \underbrace{\gamma}_{\text{discount}} \underbrace{\sum_{s'} T(s'|s_0, \pi(s_0))}_{\text{expected}} \underbrace{V^{\pi}(s')}_{\text{future value}}$$

This final expression is known as the Bellman Equations.
We notice that there are unknowns on both sides of this equation.

6 Solving the Bellman Equations

If the distribution is discrete, then we have a linear system of equations. This is super convenient because we can use matrix multiplication.

$$\begin{bmatrix} V(s_1) \\ ... \\ V(s_k) \end{bmatrix}$$

We observe :

$$\overrightarrow{V}^{\pi} = \overrightarrow{R}^{\pi} + \gamma T^{\pi} \overrightarrow{V}^{\pi}$$
$$\overrightarrow{V}^{\pi} = (I - \gamma T^{\pi})^{-1} \overrightarrow{R}^{\pi}$$

Alternatively, we can solve this iteratively.
We set $\overrightarrow{V}_i^{\pi}$ to $R^{\pi} + \gamma T^{\pi} \overrightarrow{V}_{i-1}^{\pi}$. We can expand this expression:

$$R^{\pi} + \gamma T^{\pi} \overrightarrow{V}_{i-1}^{\pi}$$
$$= R^{\pi} + \gamma T^{\pi} (R^{\pi} + \gamma T^{\pi} \overrightarrow{V}_{i-2}^{\pi})$$
$$= R^{\pi} + \gamma T^{\pi} R^{\pi} + \gamma^2 T^{\pi} T^{\pi} \overrightarrow{V}_{i-2}^{\pi}$$

We notice that the starting value, $\overrightarrow{V}_{i-2}^{\pi}$, is now being multiplied by γ^2. Ad we move through the steps, we are pushing this starting value further and further into the future. Therefore, eventually the value of $\overrightarrow{V}_i^{\pi}$ will converge and our starting value does not really matter.

7 Concept Check

Question to think about before next class:
If $r'(s,a) \propto r(s,a) + \beta$, does the optimal π change?