Lecture #1: Introduction to Reinforcement Learning

Joel Lee
Department of Computer Science
National University of Singapore
Singapore, S117417
{joel, STUDENT2, etc.}@u.nus.edu

1 Introduction

Copy this template and use it to write up the lecture notes. Also copy over bibliography.bib and add any references you use to that file.

2 Policy for the course

The class will meet every week around 3pm to 5pm to review the material presented. Prof Min and Prof Lee Wee Sun will be giving lectures for week 3 and week 4. For subsequent weeks, participants will take turns assuming the role of questioner/lecturer.

3 Supervised Learning and Imitation Learning

In Supervised learning, we train a model to match the input to other labels, perhaps say, using mean square error or some other model to check for accuracy or loss. In reinforcement learning, we learn and reiterate using feedback. Given a state, we need to decide what action we wish to take. <Insert Image>. For greater detail, refer to https://hackernoon.com/reinforcement-learning-and-supervised-learning-a-brief-comparison-1b6d68c45ffa. The goal of reinforcement learning is more general than supervised learning and we can then couple this in a sequential method to build up a policy

4 Definitions

4.1 Policy

Given a state, what action are we going to take?

4.2 State

Action doesn't directly affect the policy. State is what is actually happening in the worl. Based on what we can see from sensory input, we can take an action. Tiger and the car(Observation wouldn't let us see it). we can't observe the tiger but our previous state allows us to know

4.3 Markov property

If I know the current state of the system I know whawt's the probability of getting to the next state. Particular state + what's the reward for taking an action.

5 Finite horizon vs infinite Horizon

Finite horizon is not stationary – it diverges. At Initial time point it may be at a stable state. Hundred or thousand points can be just of the stationary distribution. - Reward function: Stationary distribution - Reward function can discount rewards over time(Just ensure it is stable at the end) - Agent falls off the cliff and breaks then th process can't be reversed. - Ergodicity assumption doens't get us into a log of trouble - Policy is trained on stationary distribution(Policy may not be making the most optimal move)

- Optimality is with respect to the expected value - Stationary distribution: Multiply against transition proba you get exact same distribution - Infinite Horizon case: reach any state given any different state - If there is a terminal state and a non-zero probability of transitioning to it. - nonzero probability that it is going to a terminal state - Probability one in the terminal state - Reward function may not be smooth but it doesn't stop us from applying a reinforcement learning approach - SGD must be differentiable but if you take an RL approach it doesn't need to be. Reward function doesn't need ot be smooth

- Policy gradients, fit the model. - Loss function of how badly the agent is doing - Take the expectation of how well the policy performs - Whole bunch of different ways(Base/model free) - Why isn't there just one RL algorithm out there - Total reward for taking an action at a particular time at a particular state(Q value is specific to policy and it computes reward for all of the subsequent time steps)

- State of the board as you receive it as an agent - Q-function I take one or infinite or discrete number of actions how my value function will change - What is the expected value afterwards - Calculate how good a particular policy is

- Value function your reward what is your best position after making a move. How likely is it for you to win?

- Set probability to zero everywhere except for the one region where it is 0 - High variance slow learning, need many samples to estimate the gradient(How do we reduce the variance) - History/Markovian property of state - Policy - Most of this cost(Probabilistic interpretation - Sum of all the rewards over that particular trajectory - Expectation over all trajectories

6 Supervised Learning and Imitation

7 Reinforcement Learning Introduction

8 Policy Gradients Introduction

9 Actor-Critic Introduction