6/21/2021

**Introduction to Reinforcement Learning**

**1/ Characteristics of RL;**

* There is no supervisor, only reward signal. Trial and error
* Feedback is delayed by many steps after your decision, not instantaneous
* Time really matters (sequential, not iid data).
* Agent’s actions affect the subsequent data it receives.

**2/ Problems with RL:**

a/ Rewards:

* A reward is a scalar feedback signal indicating how well agent is doing at step t.
* The agent’s job is to maximize the cumulative rewards.
* Ex: Fly stunt maneuvers in helicopter

+ Add reward if following desired trajectory.

+ Deduct reward if crash.

b/ Sequential Decision Making:

* Goal: select action to maximise total future reward
* Action may have long term consequences
* Rewards may be delayed
* It may be better to sacrifice immediate reward to gain more long term rewards

At each time step t:

* The agent: Execute action A\_t, Receive observation O\_t, Receive scalar reward R\_t
* The environment: Receive action A\_t, Emits observation O\_t, Emits scalar reward R\_t
* The experience is the data to train the RL agent.

c/ History and State:

* The history is a sequence of observation, actions, rewards. Not very helpful in training RL agent
* State is the information used to determine what happens next.

**+ State is a function of History**

**+ Env is not helpful since the agent does not need to see the FULL env**

* 3 definitions of states:

+ **The environment state** = the environment’s private representation = whatever data the environment uses to pick the next observation/reward. The env state is not usually visible to the agent. Even if the env state is visible, it may contain irrelevant info

* Not helpful to build an agent
* **Agent state** = the agent’s internal representation = whatever info the agent uses to pick the next action = it is the info used by the RL algo.
* **The information state / MARKOV state** contains all useful info from the history.

**+** The future is independent of the past given the present

* Once the state is known, the history may be thrown away

d/ Fully Observable Environments:

* Fully observability = agent directly observes the environment state
* Agent state = environment state = information state
* THIS IS MDP

e/ Partially Observable Environments:

* Partial Observability: agent indirectly observes environment:

+ A robot with camera vision isn’t told its absolute location

* The agent state != environment state
* POMDP = Partial Observable Markov Decision process

**3/ Major Components of an RL agent:**

* An RL agent may include one or more of these components

+ Policy = agent’s behaviour function

+ Value function = how good is each state and/or action

+ Model = the agent’s representation of the environment

* **A policy = the agent’s behaviour - Agent**

+ It is a map from state to action

+ Deterministic policy: a = pi(s)

+ Stochastic policy: a = pi(a|s)

* **A value function = a prediction of the future reward – Agent, Env**

+ Why we need this? If you got to choose 2 state, you got a metric to choose between them

+ Used to evaluate the goodness/badness of the states

* **A model = Predicts what the environment will do next – Env – OPTIONAL – Model Free**

+ This is the model of the environment

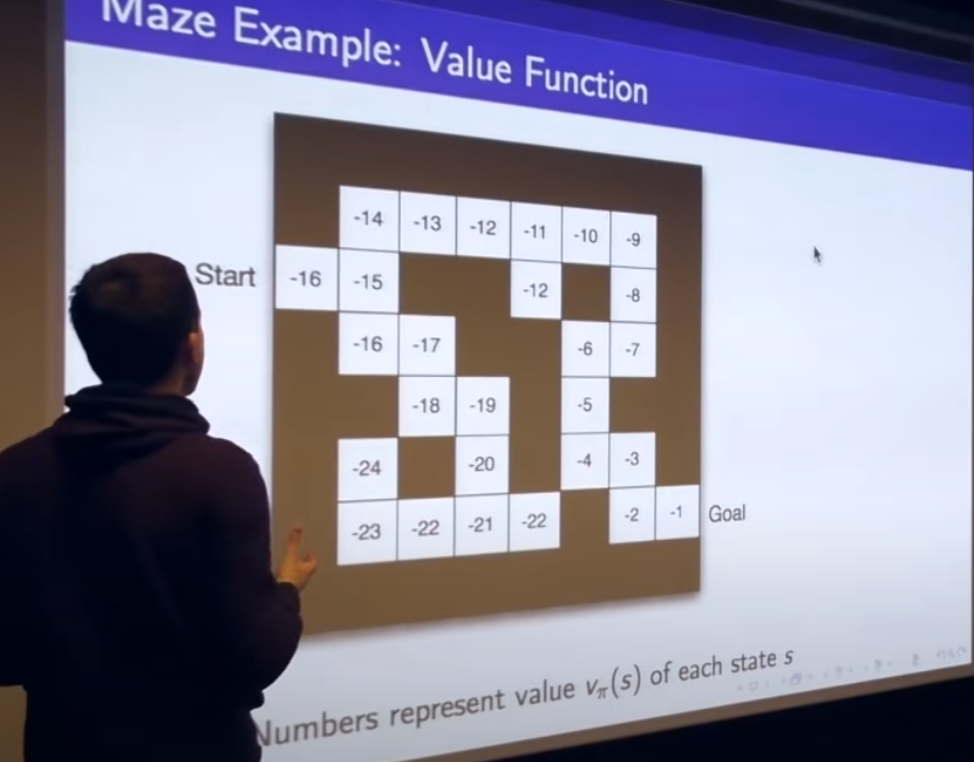
+ **Transition model:** P predicts the next state

+ **Reward model:** R predicts the next (immediate) rewards

Qr code

Description automatically generatedQr code

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

**4/ Categorizing RL Agent:**

Diagram, venn diagram

Description automatically generated

* **Value Based Algorithms / Agent** = Value Function => Look at the value function and pick the best action.
* **Policy Based Algorithms / Agent** = Look at the arrow and adjust the arrow to the best course of action.
* **Actor Critic:**

+ Policy + Value Function

* **Model Free RL:**

+ Policy and/or Value Function no model

+ We not trying to understand the environment

* **Model Based RL:**

+ Policy and/or Value Function

+ Model

+ Build the environment model and predict what will happen to the agent if it act certain way in the environment

**5/ Problems within RL:**

a/ Learning and Planning:

* Two fundamental problems in sequential decision making:

+ **RL Problem**: The environment is initially known but the agent was not told how env work. and agent improves its policy

+ **Planning**: This is the environment with all the rule. The agent performs computations with its model (without any external interaction) and agent improves its policy

Diagram

Description automatically generated Graphical user interface, text

Description automatically generated

**6/ Exploration & Exploitation:**

* RL = Trials and Error learning
* The agent should discover a good policy from its experiences of the environment without losing too much reward along the way.
* Exploration = finds more info about the env
* Exploitation = exploits the known info to maximize the reward

**7/ Predictions and Control:**

* **Prediction** = evaluate the future

+ Given a policy

* **Control** = optimize the future

+ Find the best policy