

1 Introduction

Reinforcement Learning is learning from actions.

Three central Characteristics:

- Closed-loop problems: actions of agents influence its later observations
- No direct instructions: the agent is not told which action to take.
- No initial knowledge: the agent has to figure out the consequences of its actions.

Difference to Supervised learning: 1. Agent must learn from own experience. 2. No supervisor tell what to do

Difference to unsupervised learning: 1. No predefined data. 2. Agent tries to maximize reward.

Reinforcement learning: Given: an environment and a reward signal. Find: a behavior that maximizes total reward over time.

Questions: 1. How maximize reward? 2. How find highly rewarded actions?

How to maximize total reward: exploit our knowledge about past. take currently optimal action (How do we know that?)

We need to explore the environment by taking random actions from time to time.

Dilemma: trade-off between exploration (Erkundung) and exploitation (Ausnutzung):

- to obtain reward an agent must favor actions that have proven to be beneficial in the past (Exploitation).
- to discover such actions, an agent has to try actions that it has not selected before (Exploration).

Examples:

- a animal learn to walk.
- playing chess
- trash collection robots: has to decide search more trash or go to recharge.
- Computer games
- OpenAI Gym is a python library for reinforcement learning

Commonalities

- interaction between agent and environment
- agents seek to achieve a goal in their environment, despite uncertainty
- actions affect the future state of the environment
- has to take into account indirect, delayed consequences of actions
- consequences of actions can't be fully predicted
- agents use experience
- interaction with environment is essential

1.1 Elements of Reinforcement Learning

- Environment: for example: the universe, one street, coffee machine, chess board ...
- States: represents the environments current condition: a position, temperature, etc.
- Actions: for each state there is a set of actions. E.g. turning steering wheel, move in a particular direction, ...
- Policy: A policy completely determines its behavior. E.g. if car leaves street, steer in the other direction, if pressure too high, decrease current to heating element, move to a position with high expected future reward. Is a property of the agent. An agent has a specific policy at a specific time.
- Reward: Is given out from the environment and encodes how good the agent is doing currently. Given from environment, not the agents knowledge. E.g.: winning or losing the game, car stays on the street, does not crash.
- Value functions: is compressed knowledge about the future, it encodes an agent's experience. E.g.: eat cake, feel good. Turn steering wheel 90 degrees avoid a crash.

Policy The behavior of an agent is called a policy. The policy:

- maps a state to a probability distribution over actions
- samples from this distribution to select an action

We can say an RL agent follows a policy (behaves a certain way) and updates its policy (to try and maximize reward)

The policy completely determines its behavior, deciding which action to take at a given state.

The policy is that, what the agent should learn???

Policy implementation Is a mapping from states to actions. Could be

- a lookup table
- a simple function
- a search process
- a DNN
- a combination

Reward Signal . Defines the goal in a RL problem. The reward:

- is a single real number
- is perceived by the agent at each time step
- defines what is good and what is bad
- is immediate and defines features of the problem
- primary basis for changing the policy

Value Function Defines what is good in the long run.

- is the total amount of reward an agent can expect to accumulate in the future starting from that state
- usually an estimate
- often used to choose an action

Reward vs. Value

- rewards are immediate, part of environment
- values are predictions of future reward, part of the agent
- rewards are used, in order to estimate value
- the only purpose of estimation values is to get more reward in the long run
- most RL algos focus on efficient value estimation
- a state might yield a low immediate reward but still have a high value because it is usually followed by beneficial states.

Model environment Model-based methods vs model-free methods.

Boundaries Boundary between Agent and Environment? [Example with the fisch summary](#)