RL: Introduction

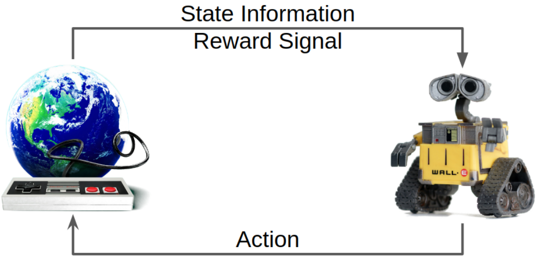
In a Nutshell

Marius Lindauer

Leibniz Universität Hannover

Automated Machine Learning Hannover

Components of RL Problems



[1]

- Data: Self-acquired observations + rewards
- Task: Learn how to behave s.t. reward is maximized

[1] Image source: Morning Brew and Marius Haakestad on Unsplash

State?

- We constantly observe our environment (and our own state)
- Mostly via sensors
  - images
  - sound
  - feeling by touch
  - feeling of acceleration
  - feeling of balance
  - ...
- Sometimes we are also presented by explicit information from our env
  - Documents
  - Scores
  - ...

⇝ We never observe the full state, but only an abstraction of it

⇝ some distinguish between states s and observations o

Automated
Machine Learning
Hannover

Actions

- In a given state, an action will (potentially) change the state
- Types of actions:

  continuous The value domain is continuous and often bounded by some range (e.g., $[0, 1]$)

  - Examples: velocity, angles, probabilities

  categorical and discrete The action is to choose from a set of possible options (i.e., potentially no ordering between actions)

  - Examples: button on a game controller, set of strategies, discrete position on a board

Transitions

- Given state s and action a, in which state do we end up?
- Either deterministic: We will end up exactly in one state
  - Examples: board games like Go or Chess
- Or non-deterministic: There is probability distribution over in which states we will end up.
  - Examples: games with randomized events (e.g., many card games), robotics – often because the control over our robot is not perfect
- Challenges:
  - Was the action responsible for the stochasticity or the environment?
  - Harder to learn in such environment since you have a different notion of reproducibility

Automated
Machine Learning
Hannover

Rewards

- Feedback on whether we did something "good" or "bad"
- Either immediate (or dense) reward: We directly get a reward signal after each transition
- Or delayed (or sparse) reward: We have to wait some states to observe the reward
  - Examples: Saving for retirement or Finding a key in video game Montezuma's revenge
  - Extreme case: we get only feedback at the end of an episode (e.g., who won a board game match)
- Introduces two challenges
  - When planning: decisions involve reasoning about not just immediate benefit of a decision but also its longer term ramifications
  - When learning: temporal credit assignment is hard (what caused later high or low rewards?)

Episode

- An episode is sequence of state-action(-reward) pair (i.e., steps)
- The end of an episode is called an horizon
- Finite horizon: We have a finite amount of steps until the episode ends
- Infinite horizon: The episode will never end (unless we abort it)