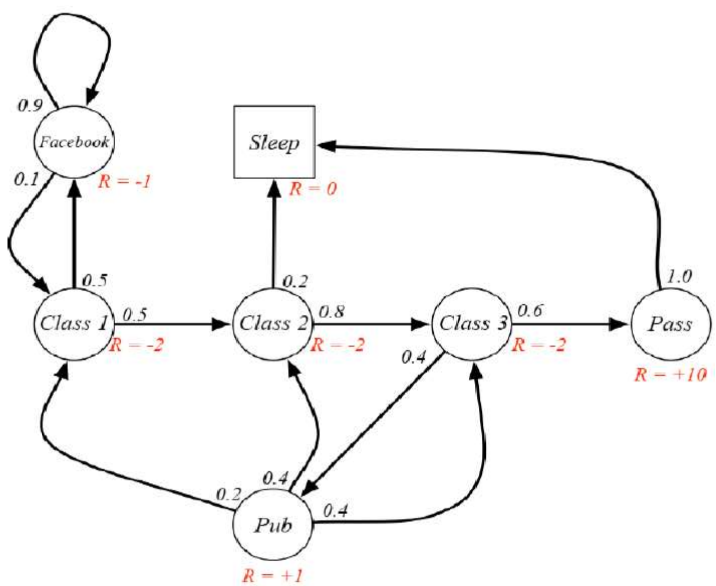
Reinforcement learning

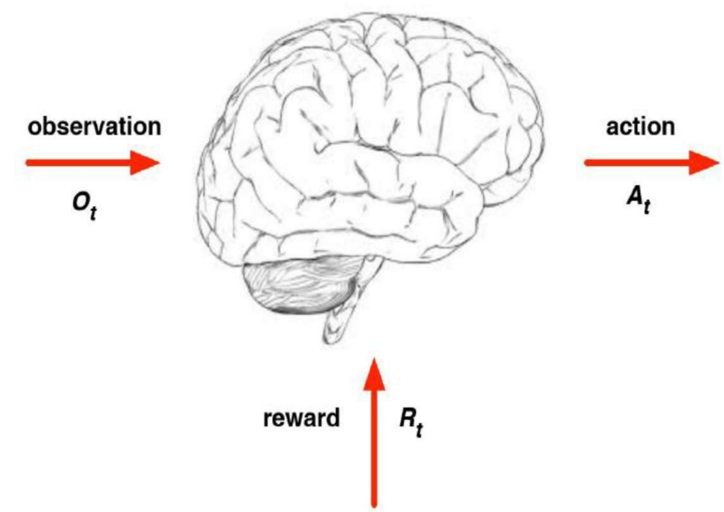
- Learning to maneuver vehicles
- Learn to control robots (walking, navigation, manipulation)
- Manage portfolios
- Play games
- Discover new molecules
- End-to-end learning with discrete structures

Markov decision process



$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Reinforcement learning



- A reward R is a scalar feedback signal
- Indicates how well agent is doing at step
- The agent’s job is to maximise cumulative reward
- Reinforcement learning is based on the reward hypothes
- All goals can be described by the maximisation of expected cumulative reward

Reinforcement learning

- Learning to drive a car (+reward for getting places safely getting places safely -reward for crashing)
- Make a humanoid robot walk (+reward for forward motion, -reward for tripping over)
- Make a robot arm manipulate objects (+reward for goal achievement, -reward for object falling)
- Manage an investment portfolio (+reward for each $ in bank)
- Play games (reward for increasing/decreasing score)
- Discover new molecules (+reward for synthesizable molecule, -reward for toxic molecule)
- Scheduling and planning
- Solve other optimization problems

Reinforcement learning

- Goal: select actions to maximise total future reward
- Actions may have long term consequences
- Reward may be delayed
- It may be better to sacrifice immediate reward to gain more long-term reward
- Examples:
  - A financial investment (may take months to mature)A financial investment (may take months to mature)
  - Refuelling a helicopter (might prevent a crash in several hours)Refuelling a helicopter (might prevent a crash in several hours)

Key components of RL

- Policy: agent’s behaviour function
- Value function: how good is each state and/or action
- Model: agent’s representation of the environment

Policy in RL

- A policy is the agent's behaviour.
- It is a map from state *s* to action *a*.

$$a = \pi(s) \qquad\qquad \pi(a|s) = P(A_t = a | S_t = s)$$

Value function and model

- The value function *v* is a predictor of future reward
- Used to evaluate the goodness/badness of states
- And therefore to select between actions, e.g.

$$V_{\pi}(\mathrm{s}) = \mathrm{E}[R_{t+1} + \gamma R_{t+2} + \gamma^{2} R_{t+2} + \cdots | S_t = s]$$

- A mode predicts what the environment will do predicts what the environment will do next
- Predict next state following an action *a:*

$$P_{ss}^{a}{}' = \mathrm{P}(S_{t+1} = s' | S_t = s, A_t = \mathrm{a})$$

- Predict next reward:

$$R_S^a = E(R_{t+1} | S_t = \mathrm{s} | A_t = \mathrm{a})$$

RL Agents

- Value Based
    - Policy Implicit
    - Value Function

- Policy Based
    - Policy

- Actor Critic
    - Policy
    - Value Function

Reinforcement learning

Two fundamental problems in sequential decision making:

Reinforcement learning:
- The environment is initially unknown
- The agent interacts with the environment
- The agent improves its policy

Planning (reasoning, introspection, search,…):
- A model of the environment is known
- The agent performs computations with its model (no external interaction)
- The agent improves its policy

Expolaration and exploitation

- Reinforcement Learning follows a trial and error process
- The agent should discover a good policy
- From its experiences of the environment
- Without losing too much reward along the way
- Exploration finds more information about the environment
- Exploitation exploits known information to maximise reward

Exploration and exloitation

- Effective reinforcement learning requires to trade between exploration and exploitation

- Game Playing:
  - Exploitation--Play the move you believe is best
  - Exploration--Play an experimental move

- Prediction: evaluate the future
  - Given a policy

- Control: optimise the future
  - Find the best policy

Deep Reinforcement learning