Reinforcement Learning

Lesson 1: A brief introduction

Edoardo Fazzari, 2022

Overview

- Reinforcement Learning: basic concepts
- Examples
- Elements of Reinforcement Learning
- Tabular and Approximate Solution Methods

RL: Basic Concepts

Learning from interaction

RL is focus on goal-directed learning from interaction

- RL is learning what to do —how to map situations to actions— so as to maximize a numerical reward signal.
- The learner:
    - is not told which action to take
    - must discover which *actions* yield the most reward by trying them

May affect not only the immediate reward but also the next situation and all subsequent rewards

Problems and Solution Methods Distinction

Very important in Reinforcement Learning

- We formalize the *problem* of RL using ideas from *dynamic systems theory* **(we will see better later in Markov Decision processes)**.
- *Idea*: capture the most important aspects of the real problem facing a learning agent interacting over time with its environment to achieve a goal.
- The *agent* must:
    - Sense the state of its environment
    - Take actions that affect the state
    - Have a goal(s) relating to the state of the environment

A third type of learning

RL is different to *supervised* and *unsupervised learning*

- *Supervised Learning*: learning from a training set of labeled examples provided by a knowledgable external supervisor.
- Unsupervised Learning: typically about finding structure hidden in collections of unlabeled data (do not rely on examples of correct behavior)

RL is trying to maximize a reward signal

Exploitation and Exploration

Concepts not present in supervised and unsupervised learning

- To obtain a lot of reward, a RL agent must prefer actions that it has tried in the past and found to be effective in producing reward:
    - *Exploit* what it has already experienced
- But to discover such actions, it has to try actions that it has not selected before:
    - *Explore* in order to make better action selections in the future

Dilemma!

We need to balance exploitation and exploration

Examples

Some everyday examples

Part 1

- A master chess player makes a move. The choice is informed both by planning — anticipating possible replies and counter-replies—and by immediate, intuitive judgments of the desirability of particular positions and moves
- An adaptive controller adjusts parameters of a petroleum refinery's operation in real time. The controller optimizes the yield/cost/quality trade-off on the basis of specified marginal costs without sticking strictly to the set points originally suggested by engineers
- A mobile robot decides whether it should enter a new room in search of more trash to collect or start trying to find its way back to its battery recharging station. It makes its decision based on the current charge level of its battery and how quickly and easily it has been able to find the recharger in the past.

Some everyday examples

Part 2

- The examples share some features
    - All involve *interaction* between an active decision-making agent and its environment
    - The agent seeks to achieve a *goal* despite *uncertainty* about its environment
        - Affect the future state of the environment, the actions and opportunities available to the agent at later times
- Correct choice requires taking into account indirect, delayed consequences of actions: *foresight or planning may be required*

The effects of actions cannot be fully predicted!

The agent must monitor its environment frequently and react appropriately

Elements of RL

Policy, Reward Signal, Value Function, Model

Part 1

- *Policy*: defines the learning agent's way of behaving at a given time.
    - Mapping from perceived states of the environment to actions to be taken when those states (*Stimulus-response rules)*
    - May be *s*tochastic, specifying probability for each actions
- *Reward signal*: defines goal of a RL problem
    - The environment sends to the RL Agent a single number called *reward*
    - Indicates what is good in an immediate sense, but *the agent's sole objective is to maximize the total reward it receives over* ***the long run***
    - May be stochastic functions of the state of the environment and the action taken

Policy, Reward Signal, Value Function, Model

Part 2

- *Value function*: specifies what is good in the long run
    - State of the total amount of reward an agent can expect to accumulate over the future, starting from that state
    - *Hard to determine*: must be estimated and re-estimated from the sequence of observations an agent makes over its entire lifetime
- *Model of the environment*(optional)
    - Something that mimics the behavior of the environment
    - Allows inferences to be made about how the environment will behave

Tabular and Approximate Solutions Methods

Tabular Solution Methods

We will see it in detail!

- RL algorithms in their simples forms:
    - State and action spaces are small enough for the approximate value functions to be represented as arrays, or *tables*
    - Methods can often find exact solutions (*optimal value function* and *optimal policy*)

- *Topics*:
    - Multi-armed bandits
    - Finite Markov Decision Processes
    - Dynamic Programming
    - Monte Carlo Methods
    - Temporal-Differences Learning
    - N-step Bootstrapping
    - Planning and Learning Methods

Approximate Solution Methods

- Extend tabular methods
- To apply to problems with arbitrarily large state spaces
- We cannot expect to find an optimal policy or the optimal value function
- Our goal is to find a good approximate solution using limited computational resources
- Use of *generalization methods:*
    - *Function approximation*
- *Topics and challenges*:
    - On-policy Prediction with Approximation
    - On-policy Control with Approximations
    - Off-policy Methods with Approximation
    - Eligibility Traces
    - Policy Gradient Methods

Bibliography:

Reinforcement Learning An Introduction (Second Edition), R. S. Sutton & A. G. Barto