Function Approximation

Introduction

Marius Lindauer

Leibniz Universität Hannover

Automated Machine Learning Hannover
AutoML.org

Overview

- Last time: Control (making decisions) without a model of how the world works
  - We have to search for a well-performing policy
  - We still don't know the MDP model
  - We assume that we can model everything by table look-ups

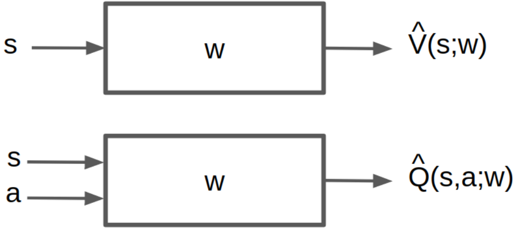
Overview

- Last time: Control (making decisions) without a model of how the world works
  - We have to search for a well-performing policy
  - We still don't know the MDP model
  - We assume that we can model everything by table look-ups
- This time: How can we learn if it does not fit into a table
  - table-based RL is often only applicable to toy problems
  - the real-world is much more complex
  - often we cannot see all kinds of states
  - Solution: we approximate the value functions by some kind of function

  $\rightsquigarrow$ First step towards deep reinforcement learning

Value Function Approximation (VFA)

- Represent a (state-action/state) value function with a parameterized function instead of a table



- For finite action spaces, often represent the Q function as a vector: takes s as input and outputs a vector with one value for each action $[Q(s, a_1), Q(s, a_2), \ldots]$

Motivation for Value Function Approximation (VFA)

- Don't want to have to explicitly store or learn for every single state a
  - Dynamics or reward model
  - Value
  - State-action value
  - Policy

Automated Machine Learning Hannover

Motivation for Value Function Approximation (VFA)

- Don't want to have to explicitly store or learn for every single state a
  - Dynamics or reward model
  - Value
  - State-action value
  - Policy
- Want more compact representation that generalizes across state or states and actions

Automated Machine Learning Hannover

Motivation for Value Function Approximation (VFA)

- Don't want to have to explicitly store or learn for every single state a
  - Dynamics or reward model
  - Value
  - State-action value
  - Policy
- Want more compact representation that generalizes across state or states and actions
- When is this possible / a reasonable thing to hope for?
  - smoothness in the state space (and action space)
    $\rightsquigarrow$ in similar states, actions should have similar effects
  - structure in the problem

Automated Machine Learning Hannover

Benefits of Generalization

- Reduce memory needed to store $(P, R)/V/Q/\pi$
- Reduce computation needed to compute $(P, R)/V/Q/\pi$
- Reduce experience needed to find a good $(P, R)/V/Q/\pi$