Learning and Adaptivity Lecture Notes

Bastian Lang *BRS University of Applied Sciences*
email: bastian.lang@smail.inf.h-brs.de

May 24, 2016

This document contains content of the lecture "Learning and Adaptivity" from the summer term in 2016 that might be of relevance for the examination.

1 Reinforcement Learning

1.1 Definition

Reinforcement learning is a class of problems where an agent learns a behaviour through trial-and-error interactions with a dynamic environment.

1.2 Strategies for solving RL problems

There are two main strategies to tackle reinforcement learning problems:

- Search space of behaviours
- Estimate the utility of taking actions

1.3 The standard RL model

In the standard RL model an agent observes the current state of its environment, chooses an action based on its observations and receives a reinforcement signal indicating the value of this state transition. The agent tries to increase the values over the long run.

1.3.1 Formal RL Model

Given a discrete set of environment states **S**, a discrete set of agent actions **A** and a set of scalar reinforcement signals, find a policy π mapping states to actions such that it maximizes some long-run measure of reinforcement.

1.4 Models of Optimal Behaviour

1.4.1 Finite-Horizon Model

Optimize expected reward for next h steps:

$$E(\sum_{t=0}^{h} r_t) \tag{1}$$

Agent consideration of taking action is limited to h next steps.

1.4.2 Infinite-Horizon Discounted Model

Take long-run rewards into account, but discount future rewards with a discount factoy γ:

$$E(\sum_{t=0}^{\infty} \gamma^t r_t) \tag{2}$$

1.4.3 Average-Reward Model

Optimize long-run average reward:

$$\lim_{h\to\infty} E(\frac{1}{h}\sum_{t=0}^{h} r_t) \tag{3}$$

1.5 The k-Armed Bandit Problem

In a room with k gambling machines each with a different probability p_i for winning, what is the best strategy for maximizing the reward when having h pulls on all the machines.

1.6 Exploitation vs Exploration

The biggest difference to supervised learning is that in RL problems the agent has to explore its environment.

Justified Techniques:

- Dynamic Programming
- Learning Automata

Ad-hoc Techniques

- Greedy Strategies
- Randomized Strategies
- Interval-based Techniques

1.6.1 Dynamic Programming

- Agent with fixed horizon
- Use Bayesian reasoning to solve for optimal strategy
- Assume prior joint distribution for parameters $\{p_i\}$ independently uniformly distributed.
- Compute a mapping from belief states to actions