Postgraduate Course
Reinforcement Learning (MSc)

Instructor Information

Santiago Zazo

E-mail: santiago.zazo@upm.es
Work Phone: +34 667451802

Julián Cabrera

E-mail: julian.cabrera@gti.ssr.upm.es
Work Phone: +34 91 549 57 00 Ext: 4088

Course Information

Course Description

This course introduces the reinforcement learning problem, that is a branch of machine learning in which the goal is to make a system learn to adapt its behaviour in order to achieve some long term goal. This is a state-of-the-art approach to planning with multiple applications in robotics, video games or business intelligence. We will start from the basics and will cover several important extensions. Although well mathematically grounded, our emphasis will be on algorithms and applications.

Prerequisites

- Elementary concepts of probability, such as probability distributions and expectations of random variables.
- Linear algebra, such as vector spaces, matrix computations and decompositions.
- Elementary knowledge of optimization theory and algorithms, such as convexity of a problem and the gradient descent algorithm. Some exposure to stochastic optimization ideas, such as the stochastic gradient descent, will be useful.
- In addition, a working knowledge of Java or Python is required.

Course Goal

To develop an understanding of the concepts, algorithms and mathematical techniques that underlie reinforcement learning.

Summary of intended course outcomes

The students will understand the main concepts related reinforcement learning, such as exploration-exploitation tradeoff, bootstrapping, off-policy learning or feature approximation. They will implement a number of important algorithms that are widely used in robotics, business intelligence and other domains. Moreover, we will briefly deal with standard mathematical tools, like contraction mappings and stochastic approximations. By the end of the course, students should be able to:

- Identify problems that are suitable for a reinforcement learning approach.
- Choose, implement and tune practical reinforcement learning algorithms.
- Understand state-of-the-art research papers on the topic.

Syllabus

Introduction

1) Introduction to learning decision making under uncertainty:
This module has two blocks:

1. Description of the reinforcement learning problem and its associated concepts, like agent, environment, actions, states, reward, state transitions.
2. Introduction to exploration vs. exploitation tradeoff. Explanation of the multiarmed bandit problem, the optimism in the face of uncertainty principle, some of its applications and the UCB algorithm.

2) Markov Decision Processes
Introduction to Markov chains: description, properties and stationary distributions. Markov decision processes and the infinite-horizon optimal control problem.

3) Planning by Dynamic Programming
Introduction to value functions, Bellman operator, Bellman optimality principle and the Bellman equation for a generic policy.
We will also introduce the concept of contraction mapping and to prove convergence of Value and Policy Iteration for bounded rewards.

4) Model-Free Prediction and Control
Introduce TD learning as a combination of Monte Carlo methods and dynamic programming for estimating the value function.
After dealing with policy evaluation, we will show the extension of TD to controlwith the Q-learning algorithm, emphasizing its off-policy feature and the need for continuous exploration.

5) Generalization: Replacing states by features (linear approximation)
In problems with very large state-space (i.e., some board games, like chess, have more than 10^40 possible configurations), learning to evaluate a policy from samples becomes impractical. The reason is that we have to take several samples per state in order to accurately approximate the value function.
In other problems, the agent has no access to the actual state, but just to some variables that provide some information of the state (e.g., a measurement).
Thus, in these cases, huge state space or only access to state-related variables, it is more convenient to build a parametric approximation of the value function from features, so that it generalizes well across states that are near each other.
In this module we introduce linear parametric approximations and some related algorithms: GTD, LSPI and fitted-Q.
We also introduce some common feature basis: tile-coding, Gaussian and Fourier.

6) Neural network approximation to reinforcement learning
One major problem when leanring a linear approximation of the value function is to find good features that generalize well across the problem. In this module we combine neural networks with reinforcement learning algorithms. In particular, we will introduce two useful algorithms that have shown to work well across multiple domains: Neural-Fitted-Q and Deep-Q-learning.

7) Policy search
So far, we have mainly focused on a dynamic programming approach. In this module, we follow a different approach and search the optimal policy in a suitable policy space. Since the policy is a mapping from states to actions (or distribution over actions), we have to search in a space of functions. A simpler work around is to consider only parametric policies, so that we just have to search in the parameter space.
We will present standard parametric approximations. Then, we will introduce Policy Gradient and Actor-Critic methods and their underlying theory.
Finally, we will recall ideas from multiarmed bandit learning and present some novel algorithms for policy search.

Textbooks:

R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, The MIT Press 1998. Draft 2nd Edition temporarily available online https://www.dropbox.com/s/b3psxv2r0ccmf80/book2015oct.pdf?dl=1.

Recommended reading material:

1. L. Busoniu, R. Babuska, B. De Schutter and D. Ernst, Reinforcement Learning and Dynamic Programming using Function Approximators, CRC Press 2010. This is a very practical book that explains some state-of-the-art algorithms (i.e., useful for real world problems) like fitted-Q-iteration and its variations. Freely available online https://orbi.ulg.ac.be/bitstream/2268/27963/1/book-FA-RL-

DP.pdf

2. C. Szepesvari, *Algorithms for Reinforcement Learning,* Morgan Claypool 2013 (freely available online). Compendium of algorithms with pseudocode ready to be implemented. Freely available online http://www.ualberta.ca/~szepesva/papers/RLAlgsInMDPs.pdf.
3. D. Bertsekas, *Dynamic Programming and Optimal Control,* Vol 2, 4th Ed. Athena Scientific 2012. This book includes very detailed mathematical analysis of optimal control and chapter 6 is devoted to reinforcement learning, a.k.a. approximate dynamic programming. Chapter 6 freely available online http://web.mit.edu/dimitrib/www/dpchapter.pdf.
4. Reading papers.

Student Assessment Criteria

Final Exam	30%
Assignments	70%

This is a practical (rather than theoretical) course in which numerous graded projects, corresponding with the implementation of multiple algorithms, will be assigned throughout the semester. These projects will provide the students with the knowledge for working as research engineers in the field.