MITx 6.86x

Machine Learning with Python-From Linear Models to Deep Learning

Course Progress Dates Discussion Resources

Course / Unit 5. Reinforcement Learning (2 weeks) / Project 5: Text-Based Ga

< Previous ✓ ✓ ✓ ✓ ✓ ✓

1. Introduction

Bookmark this page

In this project, we address the task of learning control policies for text-based games us learning. In these games, all interactions between players and the virtual world are thro world state is described by elaborate text, and the underlying state is not directly obse descriptions of the state and respond with natural language **commands** to take actions

For this project you will conduct experiments on a small **Home World**, which mimic the house.The world consists of a few rooms, and each room contains a representative obj interact with. For instance, the kitchen has an **apple** that the player can **eat.** The goal o some quest. An example of a quest given to the player in text is **You are hungry now** . the player has to navigate through the house to reach the kitchen and eat the apple. In **hidden** from the player, who only receives a description of the underlying room. At each the text describing the current room and the quest, and respond with some command (player then receives some reward that depends on the state and his/her command.

In order to design an autonomous game player, we will employ a reinforcement learning command policies using game rewards as feedback. Since the state observable to the text, we have to choose a mechanism that maps text descriptions into vector represen approach is to create a map that assigns a unique index for each text description. How becomes difficult to implement when the number of textual state descriptions are huge is to use a bag-of-words representation derived from the text description. This project complete the following tasks:

1. Implement the tabular Q-learning algorithm for a simple setting where each text desc with a unique index.

2. Implement the Q-learning algorithm with linear approximation architecture, using bag representation for textual state description.

3. Implement a deep Q-network.

4. Use your Q-learning algorithms on the **Home World** game.

Setup:

As with the previous projects, please use Python's NumPy numerical library for handling operations; use matplotlib for producing figures and plots.

1. Note on software: For the all the projects, we will use python 3.6 augmented with the toolbox, the matplotlib plotting toolbox. For THIS project, you will also be using **PyTo** Neural Nets

‹ Previous

Next ›

Connect

Blog

Contact Us

Help Center

Security

Media Kit