ShangtongZhang / **reinforcement-learning-an-introduction**

Watch 155 Star 1,306 Fork 540

Code Issues 2 Pull requests 0 Projects 0 Wiki Insights

Python code for Reinforcement Learning: An Introduction

reinforcement-learning artificial-intelligence

144 commits **2** branches **0** releases **9** contributors Apache-2.0

Branch: **master** New pull request Create new file Upload files Find file Clone or download

ShangtongZhang Update README Latest commit `f5f4334` on 22 Jun

chapter01	Adjust reward for tie to strengthen AI player 2, thanks **@shengchun** fo…	3 months ago
chapter02	Remove utils folder	2 months ago
chapter03	Remove utils folder	2 months ago
chapter04	Remove utils folder	2 months ago
chapter05	Remove utils folder	2 months ago
chapter06	Remove utils folder	2 months ago
chapter07	Add contributor for python3 support	4 months ago
chapter08	Remove utils folder	2 months ago
chapter09	Add contributor for python3 support	4 months ago
chapter10	Remove utils folder	2 months ago
chapter11	Fix a bug of expected ETD	2 months ago
chapter12	Sync with latest book release	a month ago
.gitignore	Sync with latest book release	a month ago
.travis.yml	Set up the Travis.CI integration environment and packaging.	7 months ago
LICENSE	Create LICENSE	7 months ago
README.md	Update README	a month ago
requirements.txt	Set up the Travis.CI integration environment and packaging.	7 months ago
setup.py	Set up for Travis.CI	7 months ago
tox.ini	Sync with latest book release	a month ago

README.md

Reinforcement Learning: An Introduction

build passing

Python code for Sutton & Barto's book *Reinforcement Learning: An Introduction (2nd Edition)*

> If you have any confusion about the code or want to report a bug, please open an issue instead of emailing me directly.

Contents

> Click to view the sample output

Chapter 1

1. Tic-Tac-Toe

Chapter 2

1. Figure 2.1: An exemplary bandit problem from the 10-armed testbed
2. Figure 2.2: Average performance of epsilon-greedy action-value methods on the 10-armed testbed
3. Figure 2.3: Optimistic initial action-value estimates
4. Figure 2.4: Average performance of UCB action selection on the 10-armed testbed
5. Figure 2.5: Average performance of the gradient bandit algorithm
6. Figure 2.6: A parameter study of the various bandit algorithms

Chapter 3

1. Figure 3.5: Grid example with random policy
2. Figure 3.8: Optimal solutions to the gridworld example

Chapter 4

1. Figure 4.1: Convergence of iterative policy evaluation on a small gridworld
2. Figure 4.2: Jack's car rental problem
3. Figure 4.3: The solution to the gambler's problem

Chapter 5

1. Figure 5.1: Approximate state-value functions for the blackjack policy
2. Figure 5.3: The optimal policy and state-value function for blackjack found by Monte Carlo ES
3. Figure 5.4: Weighted importance sampling
4. Figure 5.5: Ordinary importance sampling with surprisingly unstable estimates

Chapter 6

1. Figure 6.2: Random walk
2. Figure 6.3: Batch updating
3. Figure 6.4: Sarsa applied to windy grid world
4. Figure 6.5: The cliff-walking task
5. Figure 6.7: Interim and asymptotic performance of TD control methods
6. Figure 6.8: Comparison of Q-learning and Double Q-learning

Chapter 7

1. Figure 7.2: Performance of n-step TD methods on 19-state random walk

Chapter 8

1. Figure 8.3: Average learning curves for Dyna-Q agents varying in their number of planning steps
2. Figure 8.5: Average performance of Dyna agents on a blocking task
3. Figure 8.6: Average performance of Dyna agents on a shortcut task
4. Figure 8.7: Prioritized sweeping significantly shortens learning time on the Dyna maze task

Chapter 9

1. Figure 9.1: Gradient Monte Carlo algorithm on the 1000-state random walk task
2. Figure 9.2: Semi-gradient n-steps TD algorithm on the 1000-state random walk task
3. Figure 9.5: Fourier basis vs polynomials on the 1000-state random walk task
4. Figure 9.8: Example of feature width's effect on initial generalization and asymptotic accuracy
5. Figure 9.10: Single tiling and multiple tilings on the 1000-state random walk task

Chapter 10

1. Figure 10.1: The cost-to-go function for Mountain Car task in one run
2. Figure 10.2: Learning curves for semi-gradient Sarsa on Mountain Car task
3. Figure 10.3: One-step vs multi-step performance of semi-gradient Sarsa on the Mountain Car task
4. Figure 10.4: Effect of the alpha and n on early performance of n-step semi-gradient Sarsa

5. Figure 10.5: Differential semi-gradient Sarsa on the access-control queuing task

Chapter 11

1. Figure 11.2: Baird's Counterexample
2. Figure 11.6: The behavior of the TDC algorithm on Baird's counterexample
3. Figure 11.7: The behavior of the ETD algorithm in expectation on Baird's counterexample

Chapter 12

1. Figure 12.3: Off-line λ-return algorithm on 19-state random walk
2. Figure 12.6: TD(λ) algorithm on 19-state random walk
3. Figure 12.8: True online TD(λ) algorithm on 19-state random walk
4. Figure 12.10: Sarsa(λ) with replacing traces on Mountain Car
5. Figure 12.11: Summary comparison of Sarsa(λ) algorithms on Mountain Car

Environment

- Python2 or Python3
- Numpy
- Matplotlib
- Six
- Seaborn

Usage

```
git clone https://github.com/ShangtongZhang/reinforcement-learning-an-introduction.git
cd reinforcement-learning-an-introduction/chapterXX
python XXX.py
```

Contribution

This project contains almost all the programmable figures in the book. However, when I completed this project, the book is still in draft and some chapters are still incomplete. Furthermore, due to the limited computational capacity of my machine, I can only use limited runs and episodes for some experiments, so the sample output is much less smooth than that in the book.

If you want to contribute some exercises of the book or some missing examples, fix some bugs in existing code, provide sample outputs with higher quality, add some new interesting experiments related to RL, feel free to open an issue or make a pull request. I will appreciate it very much. Also, feel free to comment on the sample outputs, some curves are really interesting.

Following are known missing figures/examples:

- Example 3.4: Pole-Balancing
- Example 3.6: Draw Poker
- Example 5.2: Soap Bubble
- Example 8.5: Rod Maneuvering
- Figure 12.14: The effect of λ (I don't have time to replicate it for now)
- Chapter 14 & 15 are about psychology and neuroscience
- Chapter 16: Backgammon, The Acrobot, Go