人工智能

实验3-强化学习

2024春

实验目的

- 理解强化学习的基本原理，通过实践探索强化学习的训练过程，从而加深对该领域核心概念的理解。
- 掌握DQN算法，理解其原理和实现方式。
- 尝试使用不同的优化策略来提高模型的学习效率和性能，通过对比评估不同优化策略对模型训练和性能的影响，进一步优化强化学习模型在实际问题中的表现。

实验内容

- 安装gymnasium，实现CartPole环境的DQN（Deep Q-Network）算法的训练。
- 尝试使用不同的优化策略，以提高Agent的学习效率和性能。
- 选做：使用mindspore框架实现CartPole环境的DQN算法的训练

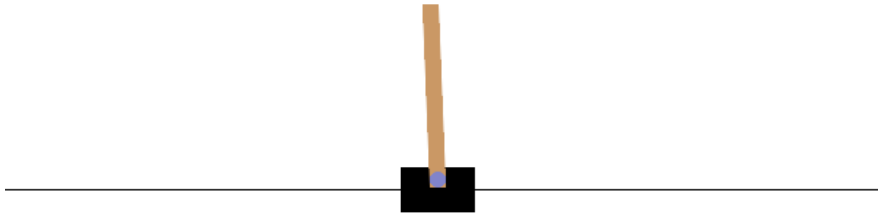
实验环境

- Python
- Matplotlib
- Pytorch
- Gymnasium：OpenAI开发的一款用于强化学习的Python库，提供了Atari、MuJoCo、Box2D等环境，包括了经典的控制问题、连续控制问题和各种强化学习任务。

实验环境

➢CartPole

- CartPolo环境是Gym库中的一个经典强化学习环境。在CartPole环境中，一个小车通过左右移动来平衡一个倒立的杆子。
- 环境的状态由四个连续值组成，分别是小车的位置、速度、杆子的角度和角速度。



Num	Observation	Min	Max
0	Cart Position	-4.8	4.8
1	Cart Velocity	-Inf	Inf
2	Pole Angle	~ -0.418 rad (-24°)	~ 0.418 rad (24°)
3	Pole Angular Velocity	-Inf	Inf

实验环境

➢CartPole

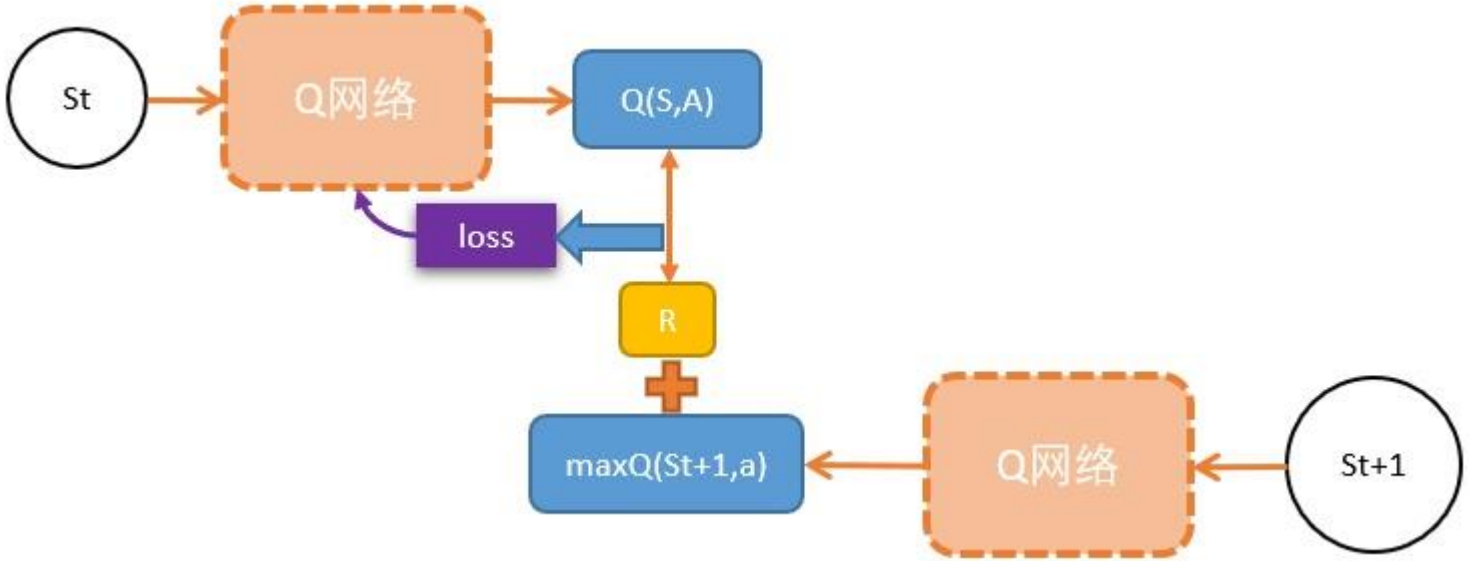
- Agent的动作空间是离散的，可以选择向左或向右推动小车。
  - 0: Push cart to the left
  - 1: Push cart to the right
- Agent需要学会通过观察环境状态并选择合适的动作来保持杆子的平衡，以获得尽可能长的平衡时间。Agent可以通过观察环境状态和奖励信号来调整自己的策略，以最大化长期累积奖励。

DQN算法：经验回放

- 经验回放实现
  - 维护一个回放缓冲区，将每次从环境中采样得到的四元组数据（状态、动作、奖励、下一状态）存储到回放缓冲区中，训练 Q 网络的时候再从回放缓冲区中随机采样若干数据。
- 目的
  - 打破样本之间的相关性，满足独立假设
  - 提高样本效率，每一个样本可以被使用多
- 回放区容量大小、每次采样的数据可根据需要调整

DQN算法：目标网络

- 目标网络
  - Q网络不断更新，会使得Q网络的学习效率比较低，而且不稳定。
  - 引入目标网络，下图中St作为输入的Q网络对应代码中的策略网络（policy net），St+1作为输入的Q网络为目标网络（target net），做梯度下降的时候，只调整策略网络的参数。在若干次学习以后才把参数复制到目标网络中。



实验步骤

1. 配置运行环境。
2. 运行gym.ipynb，安装Gymnasium, Matplotlib, Pytorch等依赖包。
3. 调整训练的num_episodes，运行reinforcement_q_learning.ipynb使得训练收敛。
4. 优化代码，可从神经网络结构的优化、超参数调优、优化经验回放区、奖励函数的设计、探索策略的设计等方面着手。要求至少进行4项优化。
5. 分析比较各种优化方法带来的结果。

实验注意事项

- 优化代码，可从神经网络结构的优化、超参数调优、优化经验回放区、奖励函数的设计、探索策略的设计等方面着手。
  - 多次执行时要注意重新初始化ReplayMemory, episode_durations等变量
  - 在强化学习中，由于多个环节存在一定的随机性，即使在相同的环境和相同的超参数下，每次训练的结果可能会略有不同。因此，如果训练的学习效果不理想，可以尝试多次重复执行，以便观察不同运行中的结果变化。这样可以更全面地了解强化学习算法的表现，有助于确定是否需要调整模型架构、超参数或者优化策略，以获得更好的学习效果。

实验提交

- 本实验不要求课上检查
- 课后将以下内容打包提交：
  - 代码
  - 实验报告（使用实验报告模板）
- 提交截止时间为14周周一，具体见作业提交系统。

同学们
请开始实验吧！