作业报告——多臂老虎机

方鸿宇 2001213098

算法说明

本算法实现了对多臂老虎机任务的策略学习。在训练过程对每个臂的平均奖金进行统计，并使用$\epsilon - greedy$的探索策略，测试过程中直接使用$greedy$策略。

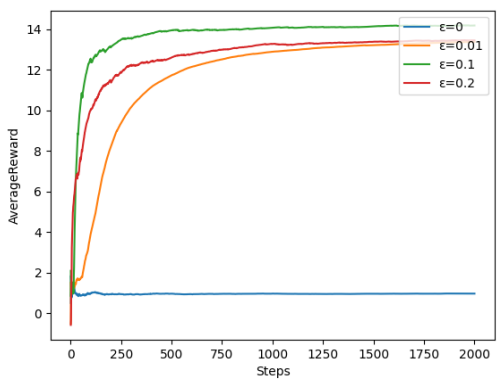
实验设置

实验中设计了臂数为15的多臂老虎机，每个臂拉一次的奖金服从高斯分布，每个臂的平均奖金为1到15的一个随机数，高斯标准差为1。实验中依次将ϵ设置为0、0.01、0.1和0.2进行实验。实验尝试次数为2000。

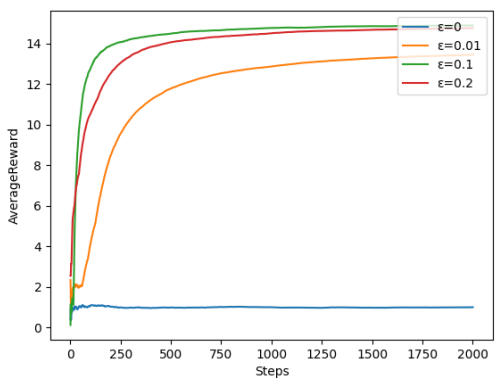
实验结果

下图展示了训练过程及测试过程中平均回报的变化曲线。当$\epsilon = 0$时，模型的平均回报几乎无法提高；$\epsilon = 0.1$时，平均回报的收敛速度快于$\epsilon = 0.01$时的速度；$\epsilon = 0.2$时，在刚开始训练时，模型收敛速度快于$\epsilon = 0.1$时的速度，然而之后逐渐慢于$\epsilon = 0.1$时的速度。

观察训练曲线，曲线$\epsilon = 0.2$最终收敛的平均回报低于曲线$\epsilon = 0.1$的收敛值，这是由于训练过程中$\epsilon - greedy$探索策略不断尝试非最佳动作的结果。测试曲线中两条曲线基本收敛至相同水平的平均回报，这是因为测试中直接使用$greedy$策略，而非$\epsilon - greedy$。



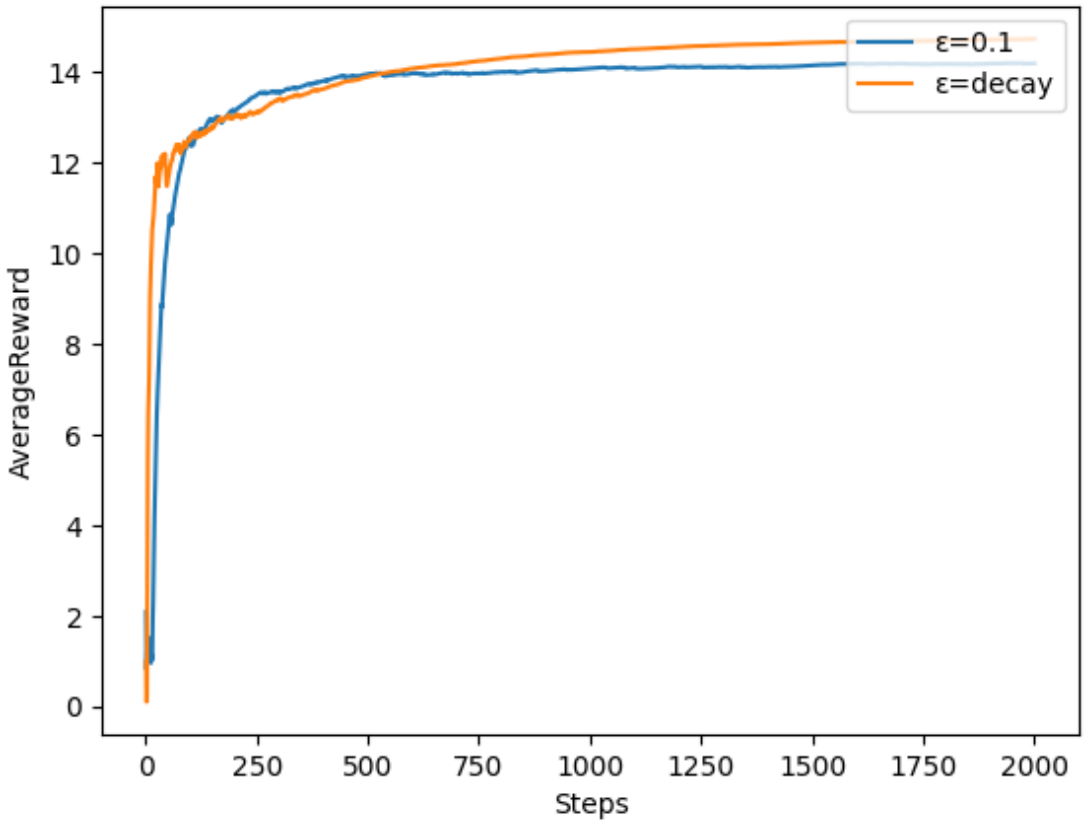
训练曲线



测试曲线

附加题

为获得更快的收敛速度，可采用ϵ衰减的$\epsilon-greedy$探索策略。初始尝试时将ϵ设置为较大值，并在训练过程中逐渐衰减至0。本实验中将ϵ的初始值设为0.3，并在训练过程中线性衰减，在第500次尝试以后衰减至0，训练曲线如下图所示，可见使用ϵ衰减策略可提高收敛速度。



代码说明

代码见`code`文件夹，包含`bandit.py`、`visualize.py`、`bandit_eps_decay.py`和`visualize_eps_decay.py`四个代码文件。

`bandit.py`进行模型训练，并将运行过程中的数据以

```
BanditID_1 Reward_1 MovingAverage50_1
BanditID_2 Reward_2 MovingAverage50_2
...
```

的格式存储，其中`MovingAverage50`表示最近50步的移动平均值。数据存储于`log/epsX.txt`和`log/test_epsX.txt`文件中，其中前者为训练过程中产生的数据，后者为测试过程中产生的数据，训练过程中的每次尝试后进行一次测试，文件名中的X表示$\epsilon-greedy$算法中的ϵ值。

`visualize.py`进行数据可视化，训练过程中的数据以"平均回报——步数"曲线表示。视可视化效果，平均回报使用了全局平均回报，即模型运行过程中所有尝试所获得的回报的平均值，未使用移动平均回报。曲线图保存为`log/log.png`和`log/test_log.png`文件，前者为训练数据曲线，后者为测试数据曲线。

`bandit_eps_decay.py`和`visualize_eps_decay.py`用于附加题中所提出的算法，代码规则与上述两部分代码相同。

代码运行方式

进入`code`文件夹，依次执行如下命令即可：

```
$ python3 bandit.py  # bandit_eps_decay.py
$ python3 visualize.py # visualize_eps_decay.py
```