

# ReforceLearning

---

## 环境

1. Python 2.7
2. Tensorflow
3. Keras
4. numpy
5. pandas
6. matplotlib
7. h5py

## 目录介绍

4个主要目录

1. QLearning 目录 包含了QLearning算法的模板，以及运用QLearning算法实现的三个gym程序。
2. DQN 目录 包含了DQN算法的模板，以及运用DQN算法实现的三个gym程序。
3. Improved DQN 目录 包含了Improved DQN算法的模板，以及运用Improved DQN算法实现的三个gym程序。
4. monitor目录保存的是三种算法训练好的模型得到的视频。

每个目录下的子目录：

1. graph：存储程序运行出来的可视化图片保存的目录，包括均值，loss值的统计，以及不同算法的对比分析
2. model：保存对应训练的模型，QLearning下的model就是普通的文件，保存的是Q表，直接文件读写即可，文件读写的函数我写在MyQLearning.py。其余两个用的是内置的接口保存的神经网络的权值model.save\_weights(model\_name)，需要用内置的接口model.load\_weights(model\_name)

## 运行方式

1. QLearning 算法相关实现在QLearning 目录中

这个目录下的MyQLearning.py是设计的QLearning算法

另外三个文件分别对应了gym的三种环境，直接运行即可，代码里包含了对程序的三种离散化模式的训练和测试。

Example command:

- python CartPole\_learning.py
- python Mountaincar\_learning.py

- python Acrobot\_learning.py

对应两个MountainCar\_compare\_three\_method.py, Acrobot\_compare\_three\_method.py文件, 是我对MountainCar-v0和 Acrobot-v1这两个环境在三种离散方式下测试后的对比图像绘制, 用于分析效果好坏的。

Example command:

- Python MountainCar\_compare\_three\_method.py
- python Acrobot\_compare\_three\_method.py

## 2. DQN算法相关是现在DQN目录中

这个目录下的MyDQN.py是设计的DQN算法

对应的文件意思和QLearning一致

Example command:

- python CartPole\_DQN.py
- python MountainCar\_DQN.py
- python Acrobot\_DQN.py

## 3. ImprovedDQN算法相关是现在ImprovedDQN目录中

这个目录下的MyImprovedDQN.py是设计的DQN算法

对应的文件意思和DQN一致

Example command:

- python CartPole\_ImprovedDQN.py
- python MountainCar\_ImprovedDQN.py
- python Acrobot\_ImprovedDQN.py