Московский государственный технический университет

им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Системы обработки информации и управления»



**Лабораторная работа № 4**

**По курсу «методы машинного обучения в АСОИУ»**

**«Реализация алгоритма Policy Iteration»**

**Выполнил:**

студент ИУ5-24М

Ширшов А.С.

**Проверил:**

Гапанюк Ю.Е.

Подпись:

29.02.2024

Москва, 2024

**Задание**

На основе рассмотренного на лекции примера реализуйте алгоритм Policy Iteration для любой среды обучения с подкреплением (кроме рассмотренной на лекции среды Toy Text / Frozen Lake) из библиотеки Gym (или аналогичной библиотеки).

**Ход работы**

Обучение с подкреплением (англ. reinforcement learning) — один из способов машинного обучения, в ходе которого испытуемая система (агент) обучается, взаимодействуя с некоторой средой.

С точки зрения кибернетики, является одним из видов кибернетического эксперимента. Откликом среды (а не специальной системы управления подкреплением, как это происходит в обучении с учителем) на принятые решения являются сигналы подкрепления, поэтому такое обучение является частным случаем обучения с учителем, но учителем является среда или её модель.

Также нужно иметь в виду, что некоторые правила подкрепления базируются на неявных учителях, например, в случае искусственной нейронной среды, на одновременной активности формальных нейронов, из-за чего их можно отнести к обучению без учителя.

**Описание среды Taxi-v3**

**Сеточный мир:**

Среда представляет собой квадратную сетку размером 5×5.

В сетке находятся несколько фиксированных точек: начальное положение пассажира, его цель и несколько препятствий.

**Действия агента:**

Агент может выполнять одно из 6 действий:

Движение на юг (0)

Движение на север (1)

Движение на восток (2)

Движение на запад (3)

Поднять пассажира (4)

Высадить пассажира (5)

**Состояния:**

Состояние среды определяется позицией такси, позицией пассажира и целью.

Всего имеется 500500 состояний (25 позиций такси × 5 возможных позиций пассажира × 4 возможных целей).

**Вознаграждение:**

Агент получает награду -1 за каждое перемещение.

Награда -10, если агент пытается поднять пассажира в неправильной позиции или высадить его в неправильном месте.

Награда +20 за успешную высадку пассажира в нужной позиции.

**Цель:**

Перевести пассажира с начальной позиции к цели с минимальными затратами времени и энергии.

Ниже представлены основные функции алгоритма:

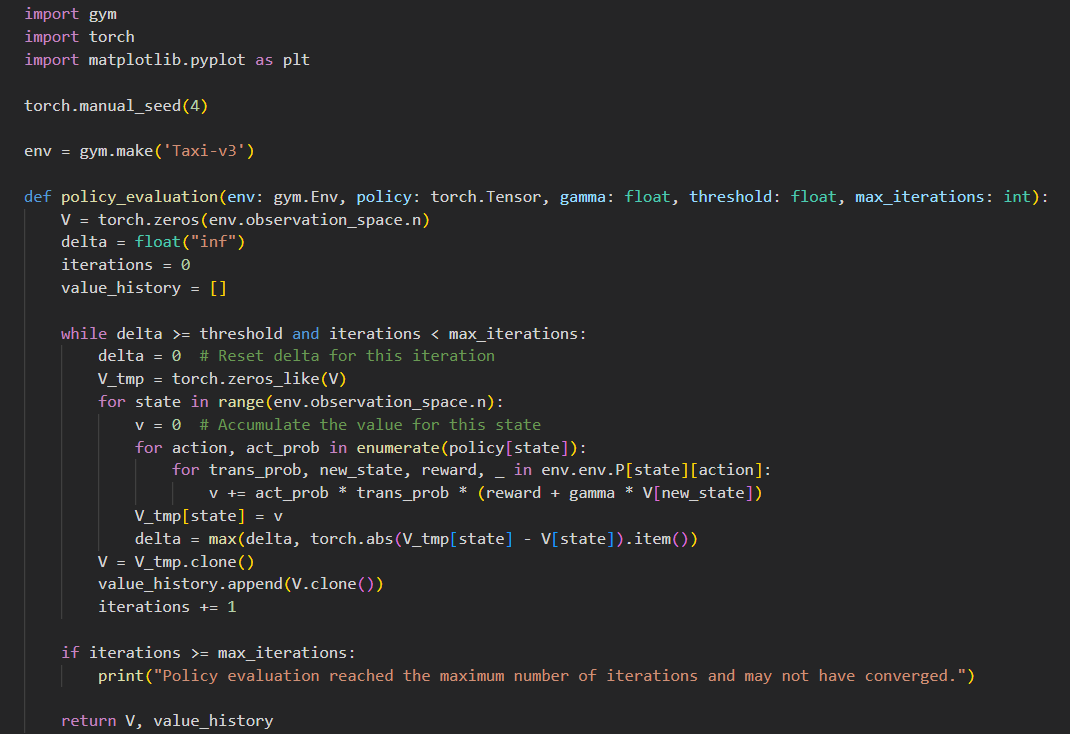


Рисунок 1 - Policy Evalution

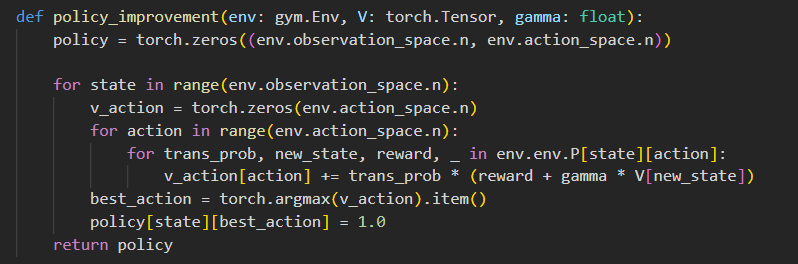


Рисунок 2 - Код функции Policy Improvement

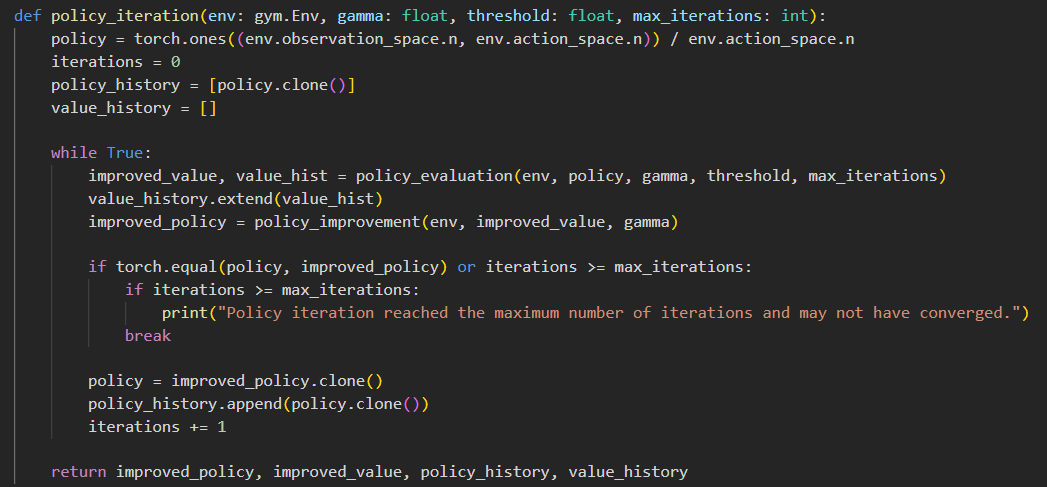


Рисунок 3 - Код функции Policy Iteration

За основу реализации был взят код с ресурса - <https://www.reddit.com/r/reinforcementlearning/comments/ubt5d9/policy_iteration_on_openai_gym_taxiv3/>

В нем были сделаны, необходимые исправления, так как он попадал в бесконечный цикл, также была добавлена визуализация результата. Она представлена на рисунке ниже. Сам код выполняется относительно долгое время, в районе 10 минут. Также было получено более 10000 итераций, что довольно много для результата, из-за визуализация получается нагроможденной.

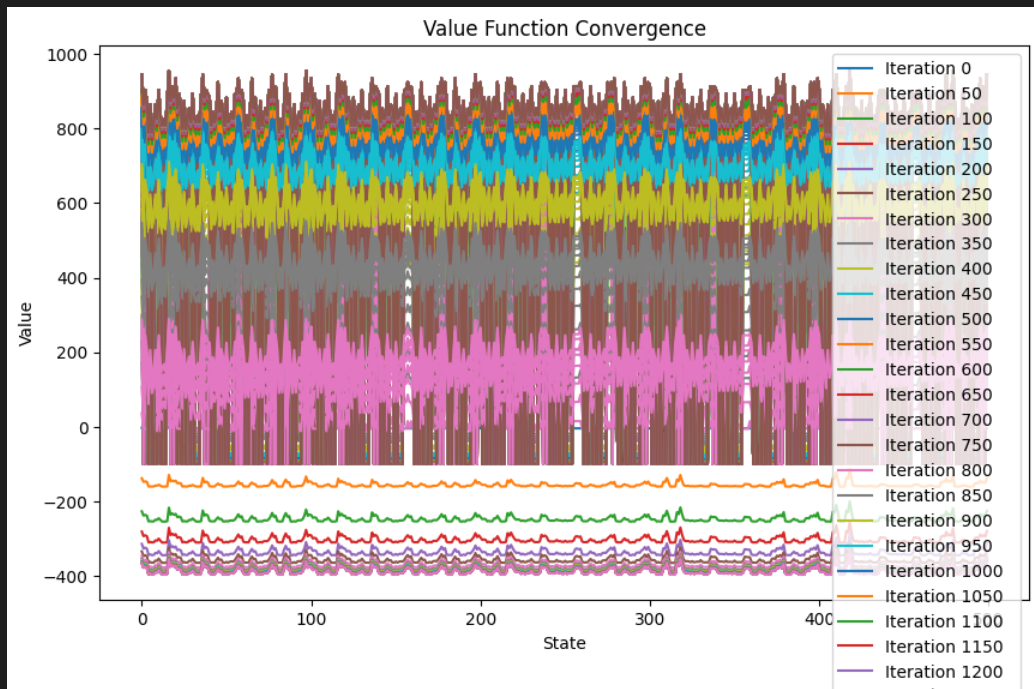


Рисунок 4 - Визуализация сходимости

**Вывод**

Для среды "Taxi-v3" из библиотеки Gym, алгоритм Policy Iteration был успешно реализован. Эта среда представляет собой задачу управления такси, которое должно доставить пассажира к его целевому местоположению в городе, избегая препятствий и неопределенных состояний.

В ходе работы были выполнены следующие шаги:

1. Импортированы необходимые библиотеки, включая Gym и NumPy.
2. Создана среда "Taxi-v3" из библиотеки Gym.
3. Реализован цикл Policy Iteration

Реализация не является наиболее оптимальной, однако передает основную суть задания и метода