Домашняя работа #5 RL

Введение

Больше графиков в коллабе. Прошу простить за большое количество повторяющегося кода. Эксперименты долгие, и не хотелось чтобы какие-то ячейки слетали.

Первое задание

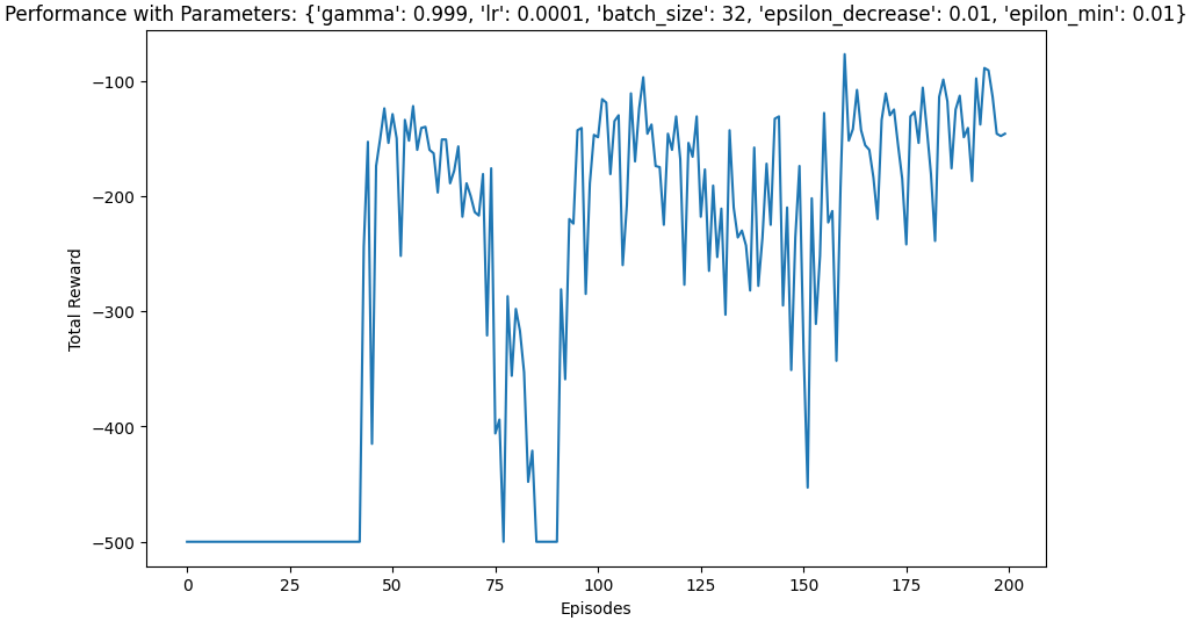
Обучение

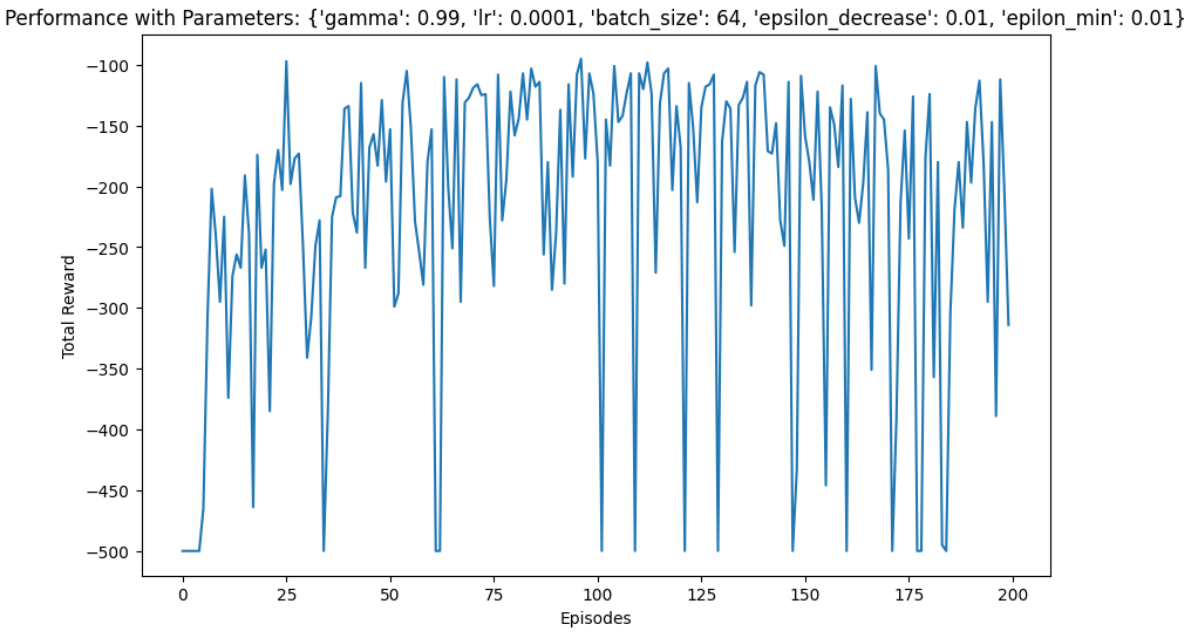
Эксперимент 1

Был написан код тестирующий разные гиперпараметры. Изначально хотелось протестировать gamma, lr, epsilon_min,epsilon_decrease, и batch_size, но вышло 3125 вариантов, и я решил сократить до gamma, lr, batch_size. Получилось 24 графика обучаемости, которые я не буду приводить здесь для краткости отчета(есть в колабе). По графикам стало ясно, что чем больше гамма тем лучше, и лучшие значения показали lr 0.001 и 0.0001. Было принято решение увеличить гамма и посмотреть что будет тогда с обучением.

Эксперимент 2

После 12 новых экспериментов почти все значения показывали хорошие результаты. За некоторым исключением:





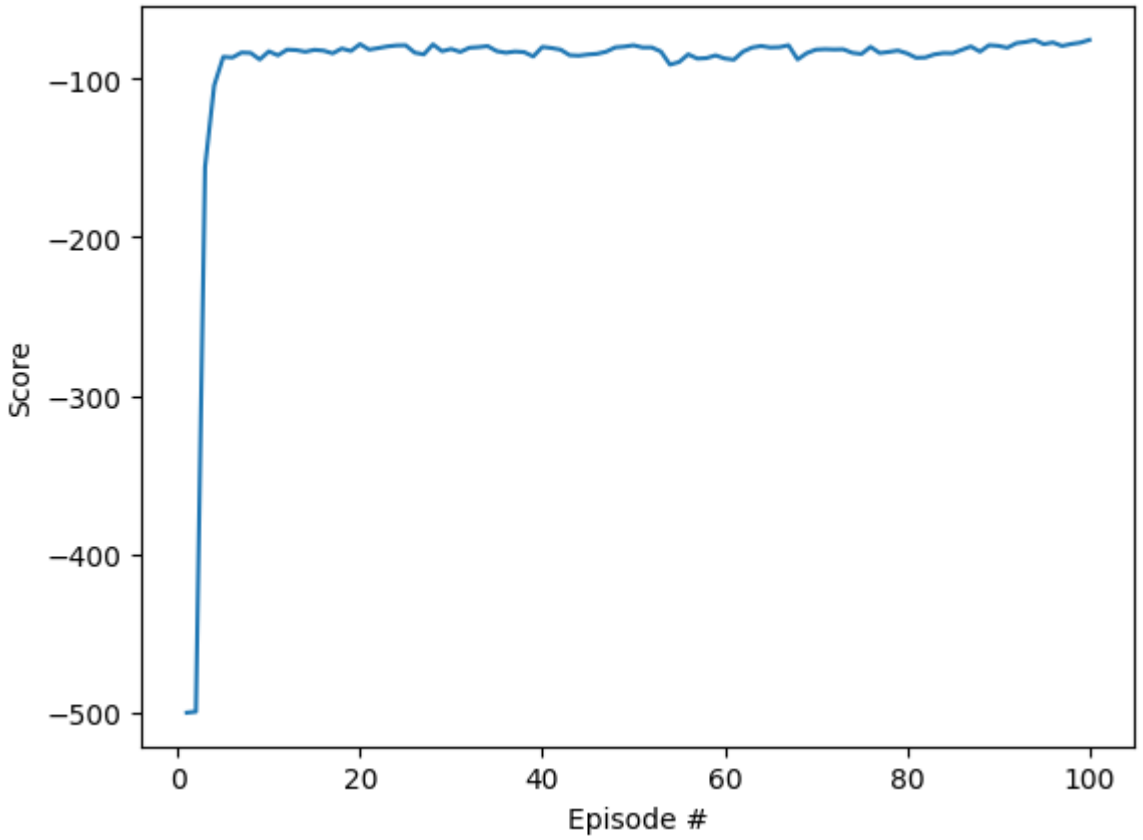
Я отсортировал лучшие значения по средней награде за последние 50 эпизодов и все эпизоды

```
Sorted by Mean Reward (Last 50):
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -95.5
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -96.1
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -109.74
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -110.52
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -111.16
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -112.72
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -115.28
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -122.26
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -127.38
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -159.92
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -166.94
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -237.5

Sorted by Mean Reward (Total):
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -102.465
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -118.905
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -120.84
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -135.83
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -137.93
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -141.645
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -144.43
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -146.715
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -180.315
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -220.285
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -221.98
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -270.12
```

Теперь попробую увеличить количество эпизодов, и повторить.

```
Sorted by Mean Reward (Last 50):
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -94.26
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -97.0
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -98.22
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -99.44
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -100.1
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -101.7
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -104.94
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -105.32
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -112.2
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -114.14
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -115.4
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -135.9

Sorted by Mean Reward (Total):
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -110.59
Parameters: {'gamma': 0.9999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -111.654
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -113.34
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -116.35
Parameters: {'gamma': 0.999, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -117.496
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -123.47
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -125.872
Parameters: {'gamma': 0.99, 'lr': 0.001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -128.528
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -132.006
Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -142.434
Parameters: {'gamma': 0.99, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -163.682
Parameters: {'gamma': 0.999, 'lr': 0.0001, 'batch_size': 32, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Total): -204.318
```

Лучшие параметры получились у Parameters: {'gamma': 0.9999, 'lr': 0.0001, 'batch_size': 64, 'epsilon_decrease': 0.01, 'epilon_min': 0.01}, Mean Reward (Last 50): -94.26.
Давайте сравним с Cross-entropy. CEM на акроботе почти с 3ьего эпизода доходит до очень хороших значений награды до которых DQN далеко даже на 500 эпизодах. Не знаю чем это обусловлено.



По идее DQN себя должен лучше показывать на более сложных средах с большим количеством действий такие как всякие Atari games.

Вывод

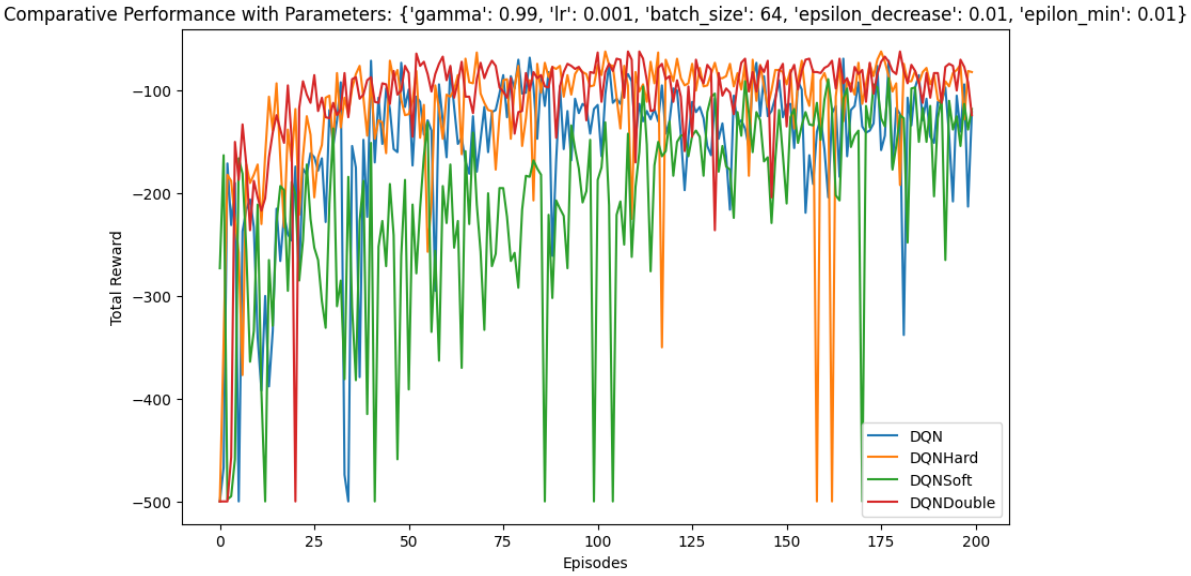
Не получилось обучить DQN лучше чем CEM. Возможно дело в конкретной среде, или же DQN не может хорошо исследовать среду акробота. Попробую во второй части обучить на улучшенных DQN, и попробую больше исследовать в процессе. Так же наверное стоит поиграться с изменением эпсилона, и посмотреть к чему это приведет.

Второе задание

Обучение

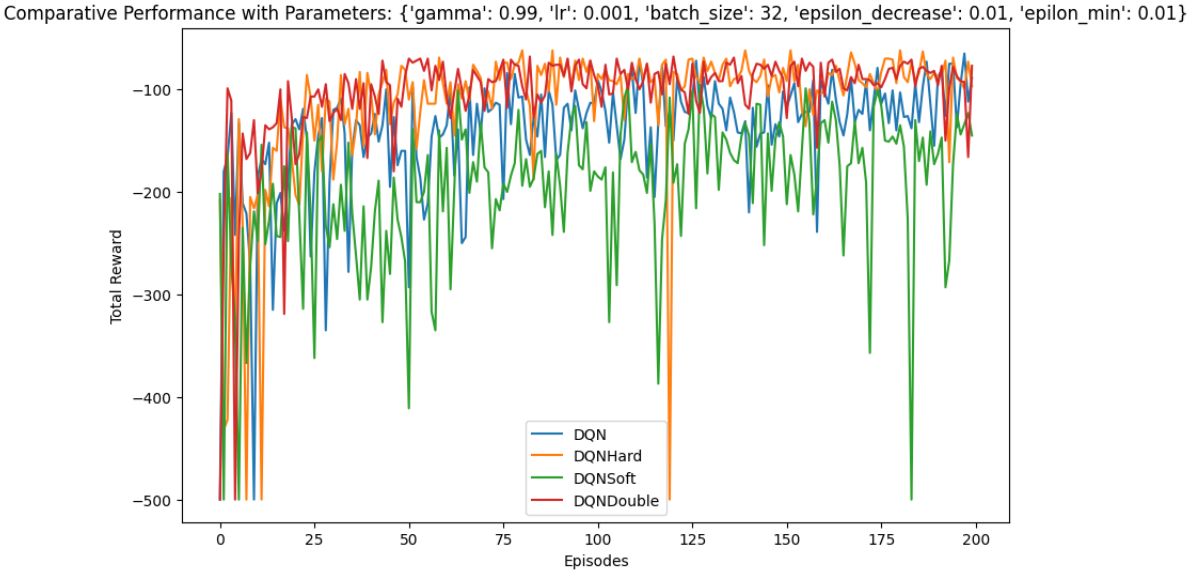
Эксперимент 1

Для начала были написаны все три агента и сравнены с обычным DQN. Я сначал сравнил общую картину и поигрался с батч сайз, гаммой, и лр как в первом эксперименте. Получилось что-то вот такое. В целом, DQNDouble показывал себя лучше всего(в колабе больше графиков).



Эксперимент 2

Понял, что это не очень интересные параметры и решил попробовать поиграться с тау, эпсилон, и батч сайз, но с другими значениями. Посмотрим, что из этого выйдет. В процессе понял почему лучше всего с большим гаммой. В случае акробота мы получаем награду единожды в конце. Поэтому всегда надо приоритизировать будущее. Для начала я запустил все классы с batch_size = 128. Стало интересно, что будет с графиками в этом случае.
Графики с 32 и 64:



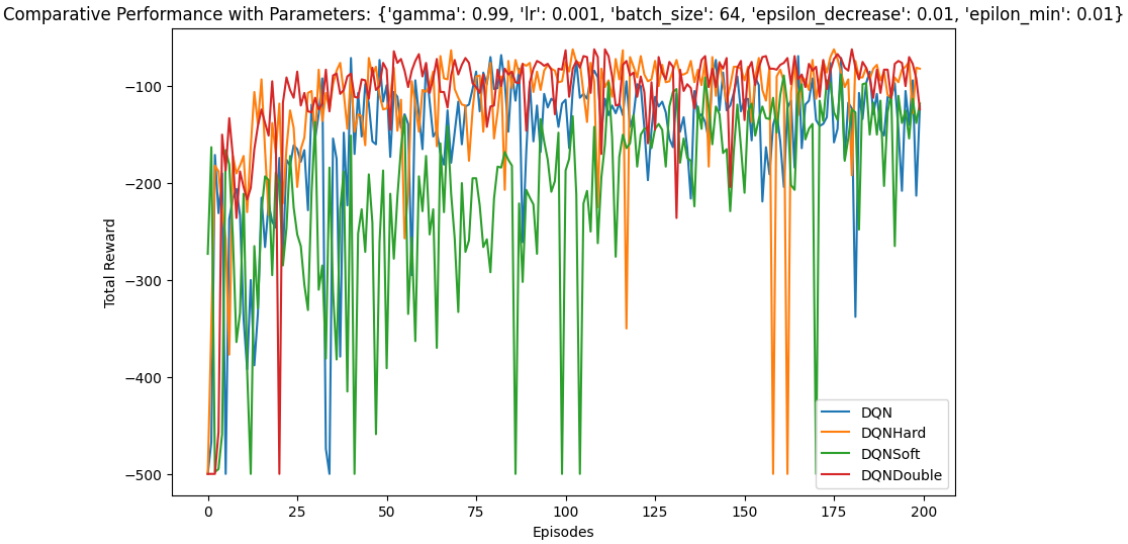
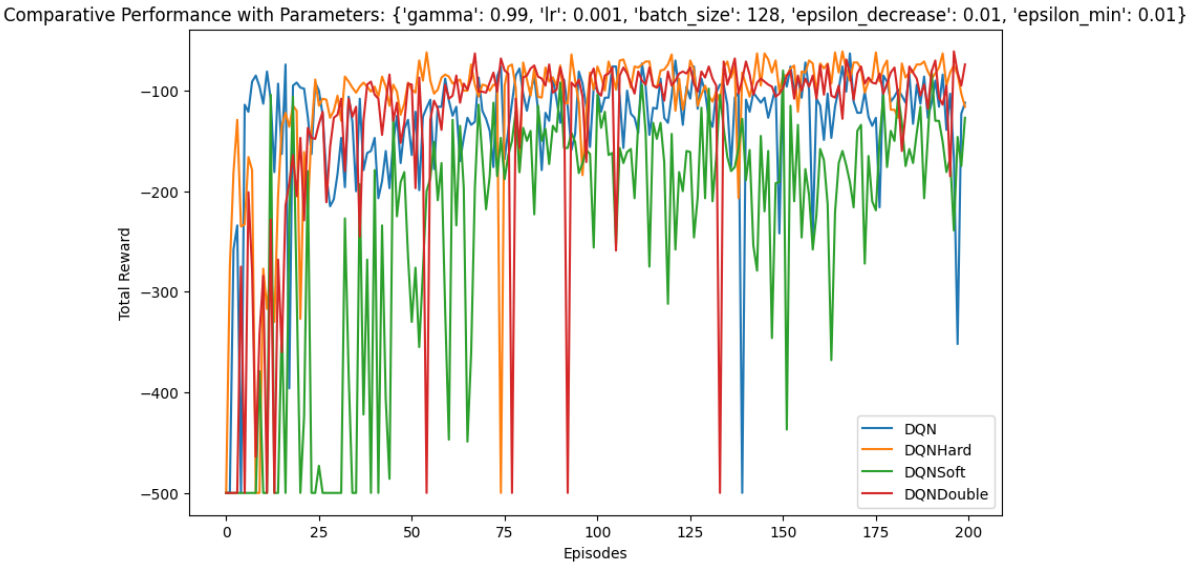


График с 128 batch_size

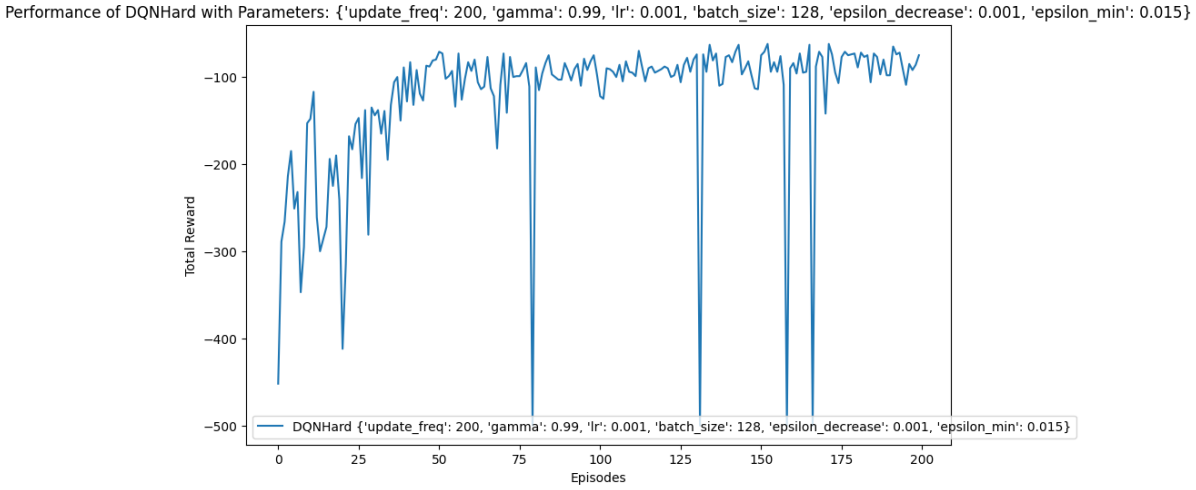


В целом какой-то существенной разницы нет.

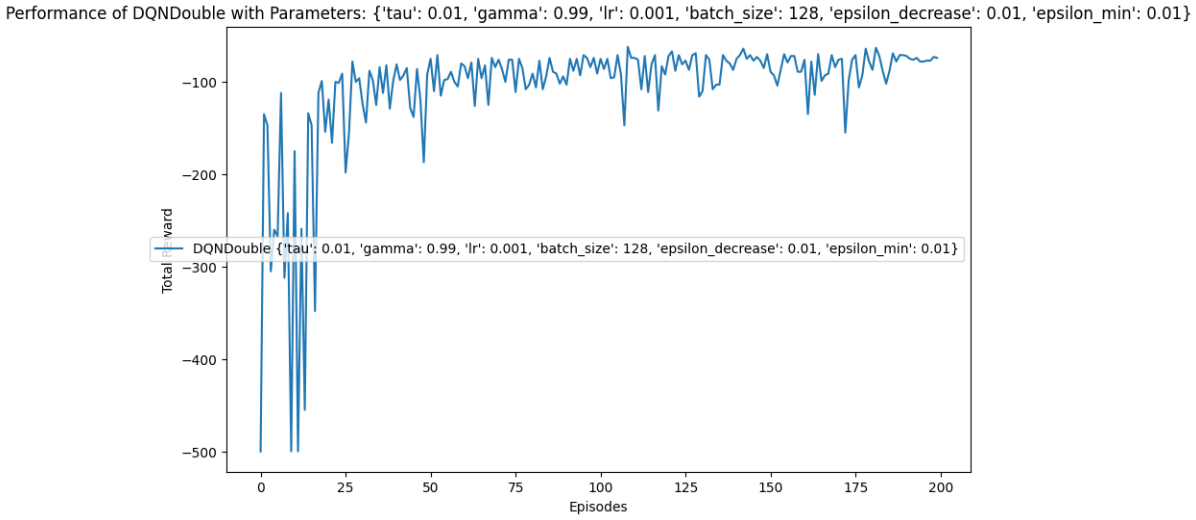
Эксперимент 3

Тут я начал экспериментировать с тау epsilon_min и epsilon_decrease, а также update_freq для DQN(больше графиков в коллабе). Некоторые из графиков(всего их около 40):

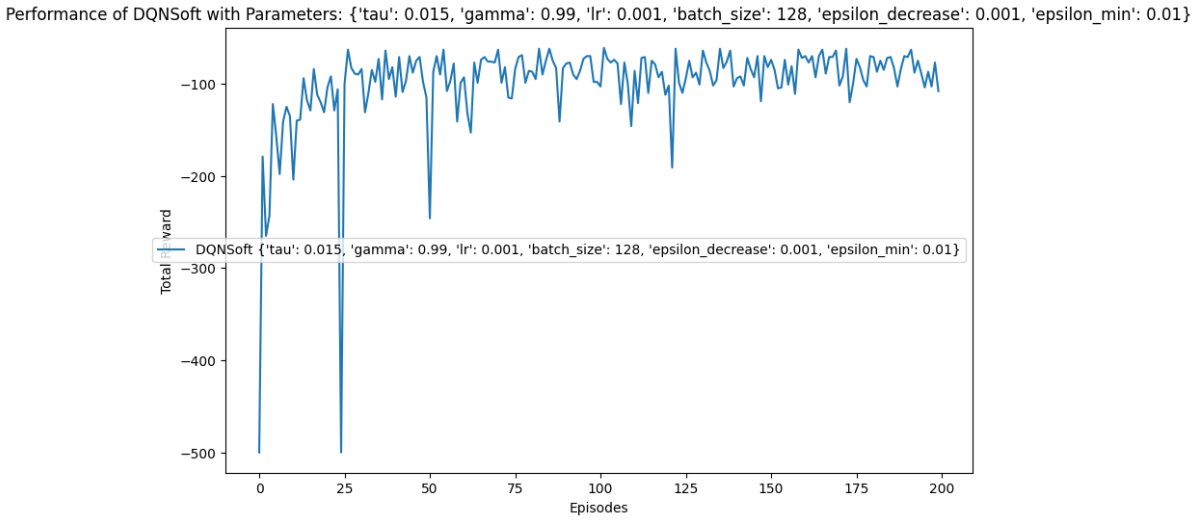
-85.46 - средняя награда за последние 50 итераций для этого агента:



-83.8



-83.48



В целом удалось достичь неплохих результатов точности, и обучение стало более стабильным.

Написал очень грубую и простую функцию корреляции, которая не совсем подходит в таких параметрах так как они все взаимосвязаны но тем не менее интересно показывает, что тау влияет очень сильно для агентов с ней. А для Hard влияет больше эпсилон. Видимо это связано с тем, что все эти параметры очень сильно влияют на обновление весов. Просто в случае с тау агентами там этот параметр влияет сильнее, а случае с Hard получается более тонкая настройка с помощью decrease/min.
Для агентов с Tay

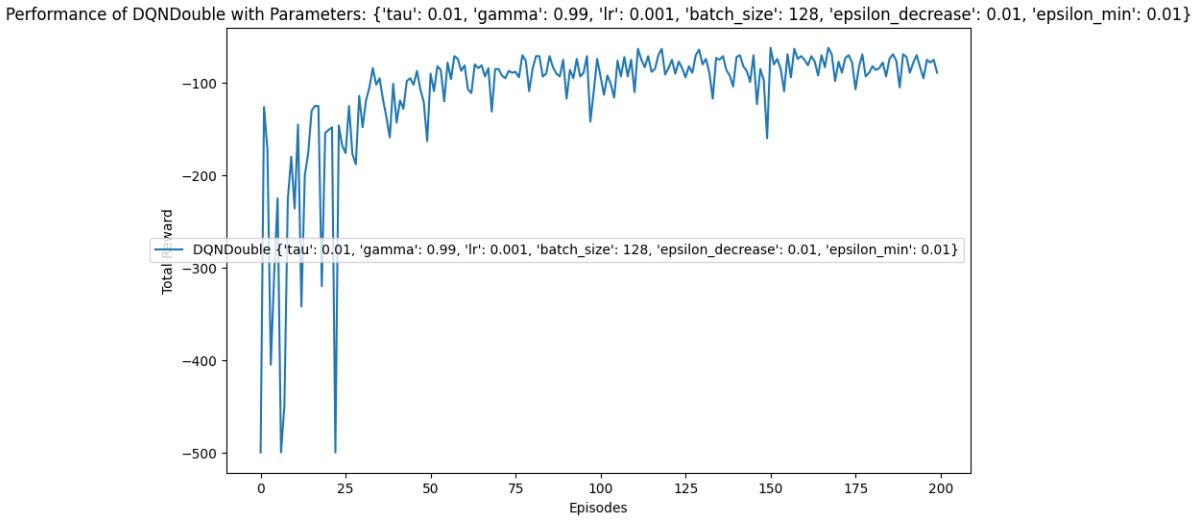
```
Correlation between tau and mean_reward: 0.6668225799949566
Correlation between gamma and mean_reward: 2.7019303097221003e-16
Correlation between lr and mean_reward: nan
Correlation between batch_size and mean_reward: nan
Correlation between epsilon_decrease and mean_reward: 0.05471851797232531
Correlation between epsilon_min and mean_reward: 0.04096774217085531
```

Для DQNHard

```
Correlation between update_freq and mean_reward: -0.014677491294584887
Correlation between gamma and mean_reward: 5.112011539241868e-16
Correlation between lr and mean_reward: nan
Correlation between batch_size and mean_reward: nan
Correlation between epsilon_decrease and mean_reward: 0.2784277941523495
Correlation between epsilon_min and mean_reward: 0.372724223447099
```

Эксперимент 4

После этого я прогнал еще эксперимент с агентом DQNDouble, и добился точности -80.4 что очень хорошо




```
Correlation between tau and mean_reward: 0.6707505655886249
Correlation between gamma and mean_reward: -5.418704989710984e-16
Correlation between lr and mean_reward: nan
Correlation between batch_size and mean_reward: nan
Correlation between epsilon_decrease and mean_reward: -0.018642513276164056
Correlation between epsilon_min and mean_reward: -0.07372300684948464
```

Корреляция была похожей. Не знаю что еще можно было бы сделать, чтобы увеличить точность. Возможно стоило протестировать с разными значениями тау, и более тонко настроить остальные параметры, но для этого потребовалось бы очень большое количество времени.

Вывод

Модицифицированные DQN в среднем дают более высокие значения награды чем немодифицированные, но все зависит от настройки гиперпараметров. Возможно более сложные стратегии выбора эпсилона, и тау дали бы лучшие результаты. Стоит отметить, что акробот очень сложная среда изза того, что награду получаешь единожды. В статьях и исследованиях обычно в играх со значительно большим количеством параметров DQN превосходят другие алгоритмы. Отдельно хочу попросить обратить внимание на графики в IPYNB отчетах. Там показаны все мои эксперименты с разными гиперпараметрами.