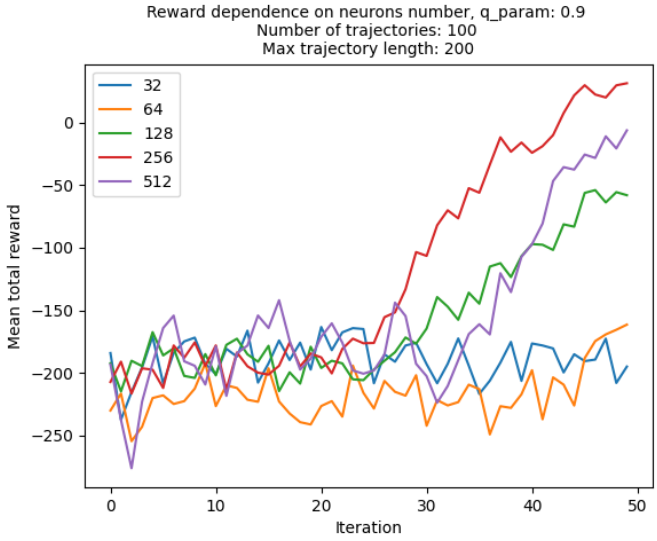
Задание 1 Lunar Lander.

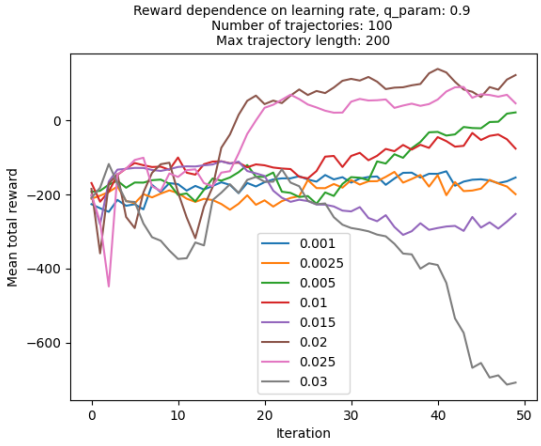
Эксперимент: зависимость награды от количества нейронов.



Вывод.

Сеть состоит из 2ух слоев, первых слой размера N который я подбирал и второй слой размером равным количеству действий. Для корректного сравнения первый слой всегда инициализировался нулями. Из полученных результатов выбрал размер 256.

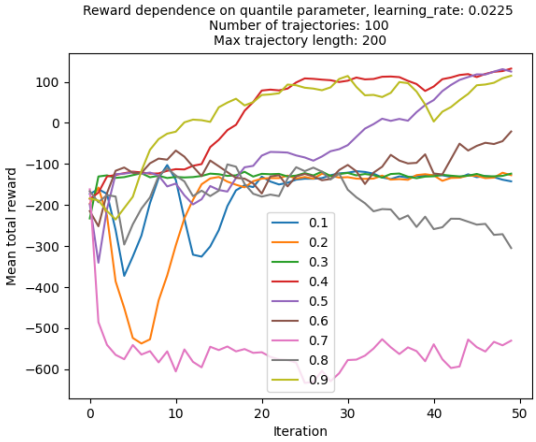
Эксперимент: зависимость награды от количества скорости обучения.



Вывод.

Оптимальный learning_rate находится между 0.02 и 0.025, для большей уверенности инициализировал веса сети нулями.

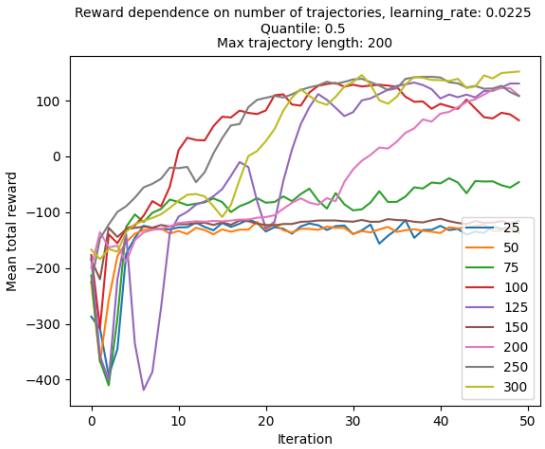
Эксперимент: зависимость награды от квантиля.



Вывод.

Для ускорения экспериментов я не стал брать много длинных траекторий, следовательно, для увеличения обучающей выборки нужно снизить границу квантиля, параметр 0.5 доводит среднюю награду до 100. Не совсем понятно почему параметры 0.7 и 0.8 показали такой плохой результат, при том что 0.9 почти равен максимальному.

Эксперимент: зависимость награды от количества траекторий.



Вывод.

Увеличение траекторий дает стабильный прирост к средней награде. Иногда достигнув пика, награда начинает падать как в случае с параметром = 100. Это похоже на переобучение, один из вариантов борьбы с этим добавить Dropout, но в данном случае это не помогло, скорее всего лучше сработает learning rate scheduler, где по мере приближения конца обучения мы плавно уменьшаем learning_rate.

Задание 2 Mountain Car Continuous.

В этой задаче не удалось получить положительный результат. К выбору действия я добавил шум который скейлился по eps = 1/N и выбирал траектории только с положительной наградой вместо отсечки по квантилям. По какой-то причине средняя награда доходила близко к 0, но положительно не становилась. Возможно дело в скейлинге шума.