DRL Course Домашнее задание 1

Алексей Матушевский

В этой практической работы мы изучаем работу алгоритма Cross Entropy Method (CEM) совмещенного с нейронными сетями для поиска оптимальных параметров в среде с непрерывным пространством с, не обязательными, непрерывными действиями. Непрерывные значения для действий и состояний не позволяют использовать предыдущие подходы и их улучшения, так как предыдущие подходы опирались на матрицы стратегий. Каждая строка такой матрицы представляло собой конкретное состояние системы, а столбец - действия. Непрерывные значения — не позволяют работать с матрицами, или попросту создают огромной матрице.

В таких случая лучше альтернатива - нейронные сети. Они подходят для работы с непрерывными значениями.

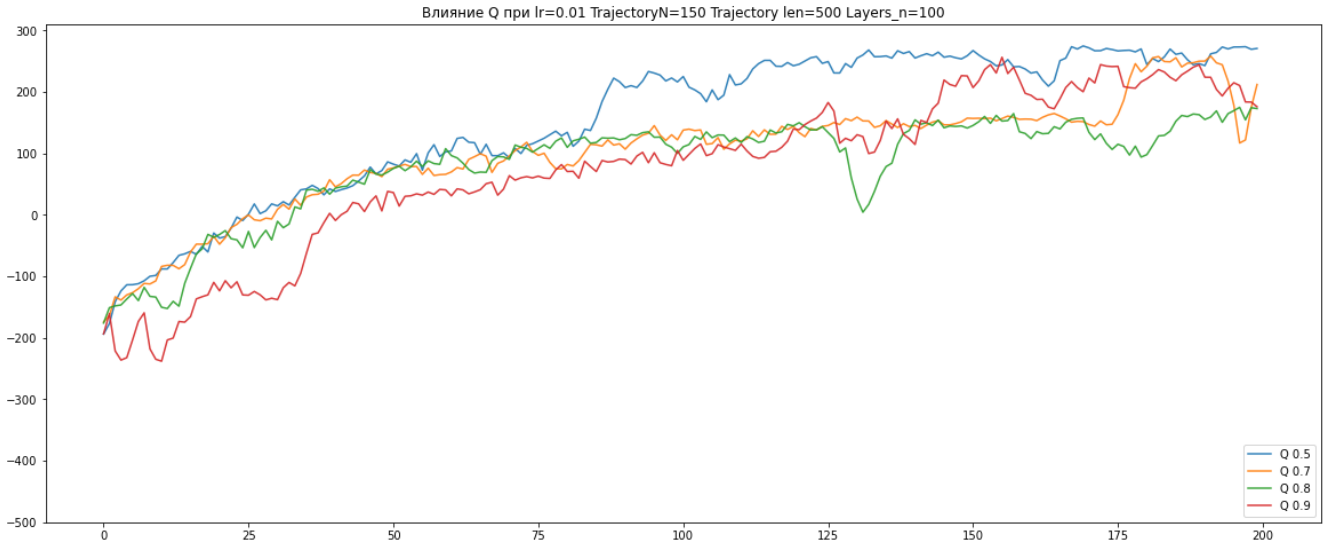
Работа в среде Lunar Lander.

В этой среде мы управляем посадочным модулем 4мя дискретными действиями. Среда описывается 8мью непрерывными состояниями.

Для решения этой задачи я использовал CEM +NN с разными наборами параметров

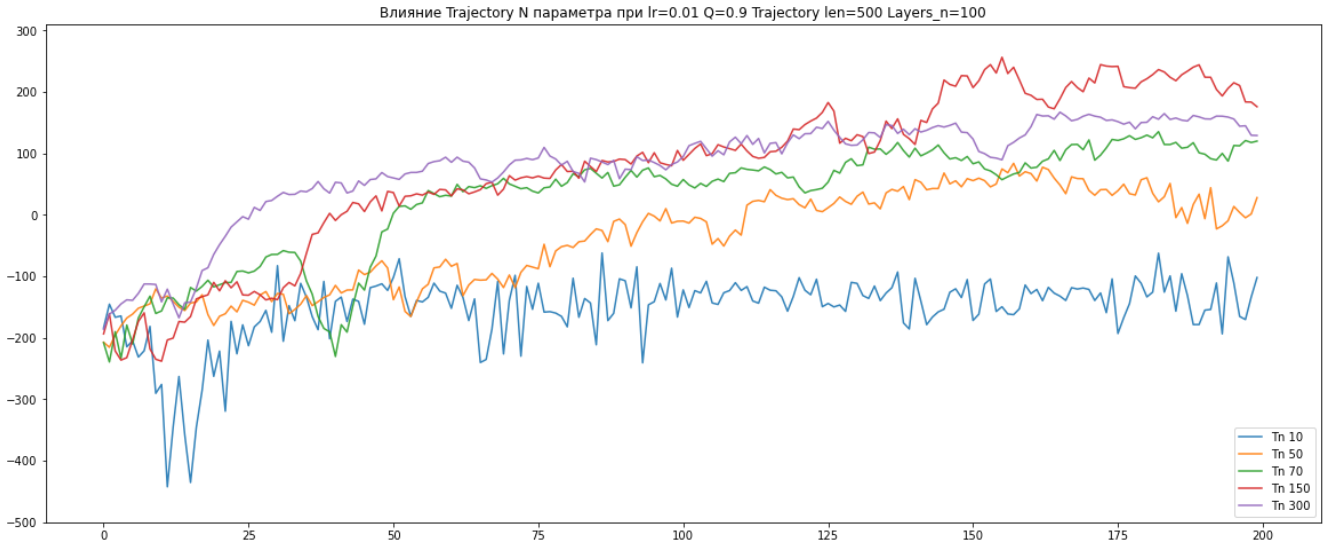
Episode Len — количество эпизодов тренировки
Q-param — квантиль отбираемые
Trajectory N — количество траекторий в одном эпизоде
Trajectory Len — длинна одной траектории
Layers N — количество нейронов в каждом слое
Learning Rate — темп изменения параметров оптимизатором

Layers N при разных Q
Проанализируем влияния Q на достижения большой награды. Как видно с увеличением Q достижение высокой награды занимает большее количество эпизодов тренировки.

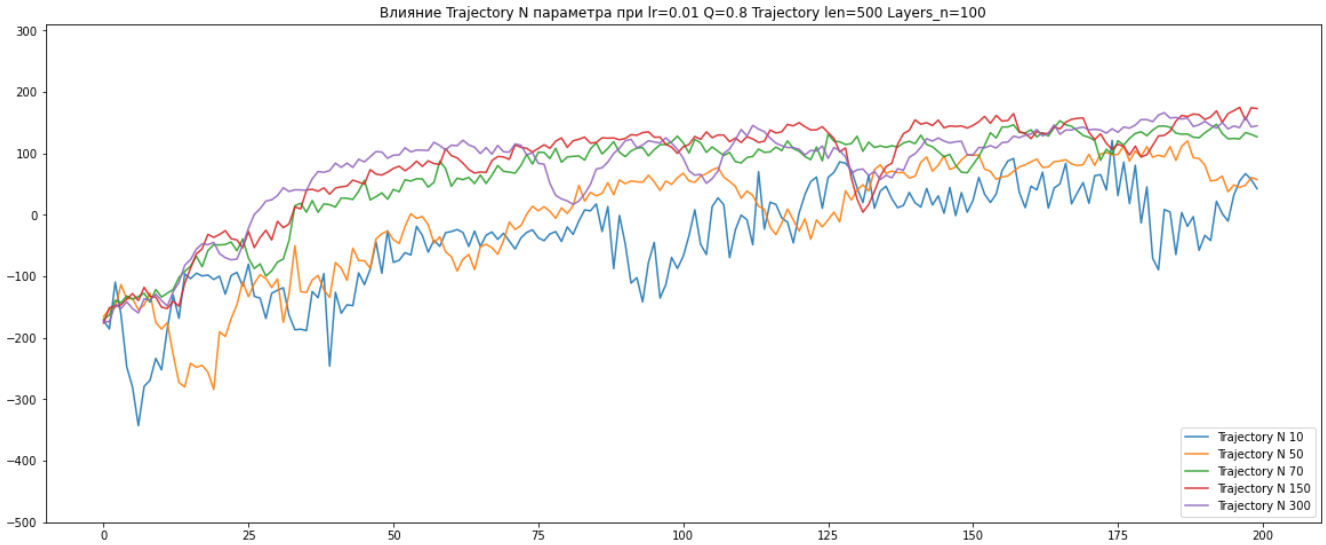


Trajectory N

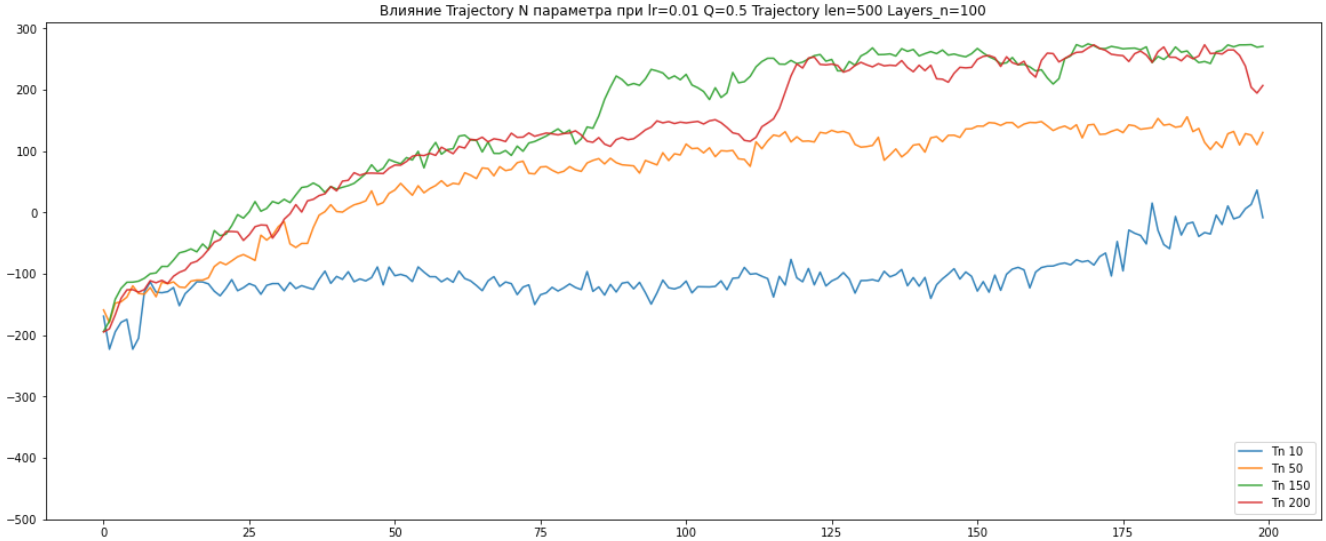
Увеличение Trajectory N при **q=0.9** Trajectory len 500. с увеличением - уменьшатся дисперсия в росте оценки.



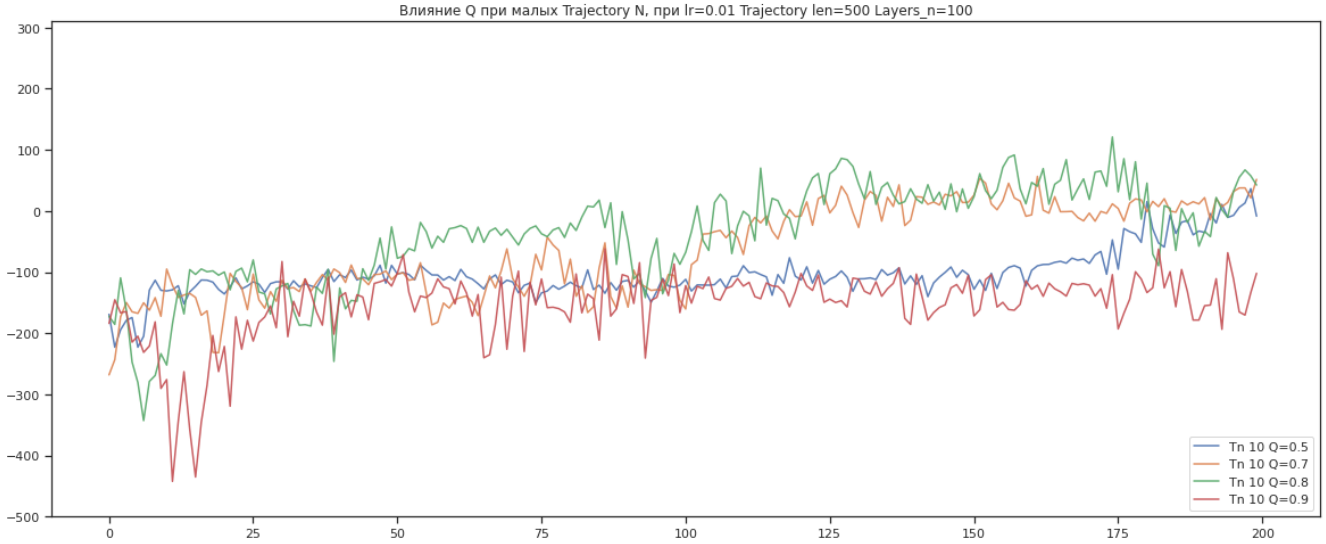
Увеличение Trajectory N при **q=0.8** Trajectory len 500. Так-же с увеличением - уменьшатся дисперсия в росте награды.



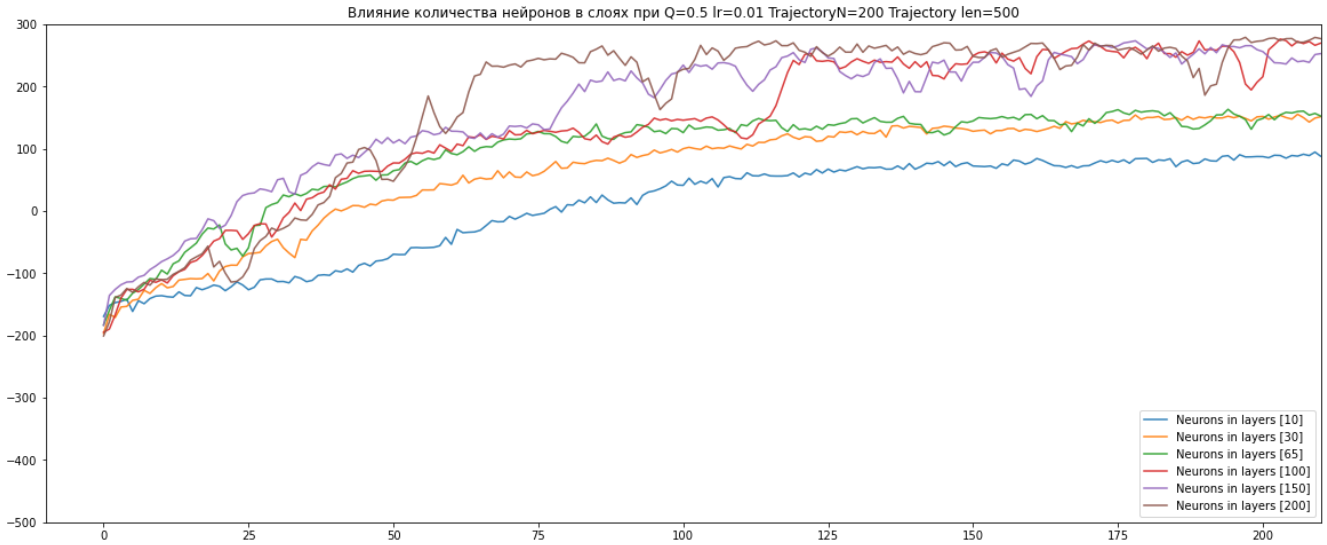
Аналогичное поведение при Q=**0.5**



При очень низких Trajectory N алгоритм не приходит к стабильному состоянию в сравнении с более высокими значениями параметра.



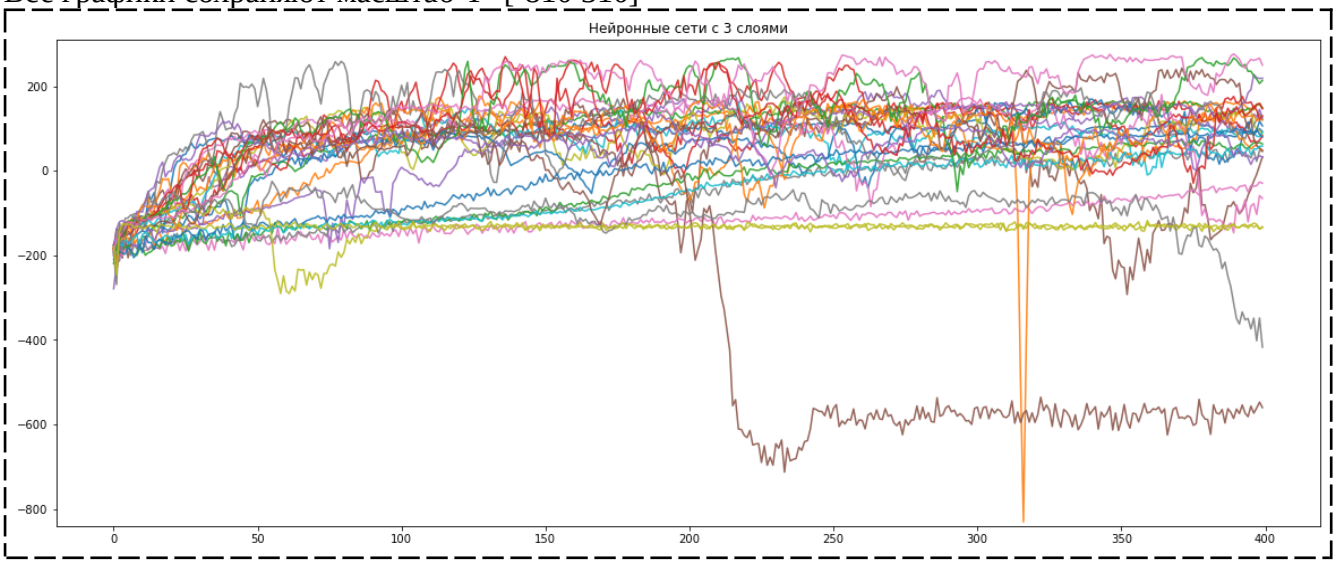
Влияние количества нейронов Layers N



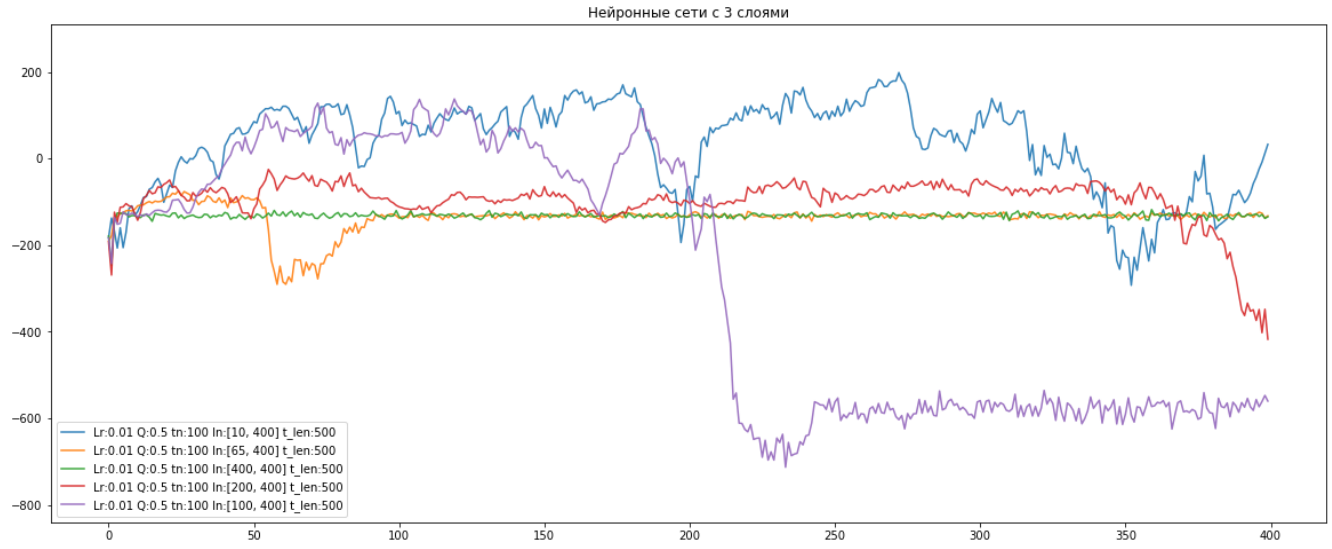
При одних и тех же параметрах Q=0.5, Trajectory N=200, и Trajectory Len=500 — создали сети с разным количеством нейронов. Как видно из графика с увеличением количества нейронов алгоритм быстрее достигает высокой награды, но при этом увеличивается дисперсия.

Увеличение количества слоев в сети.

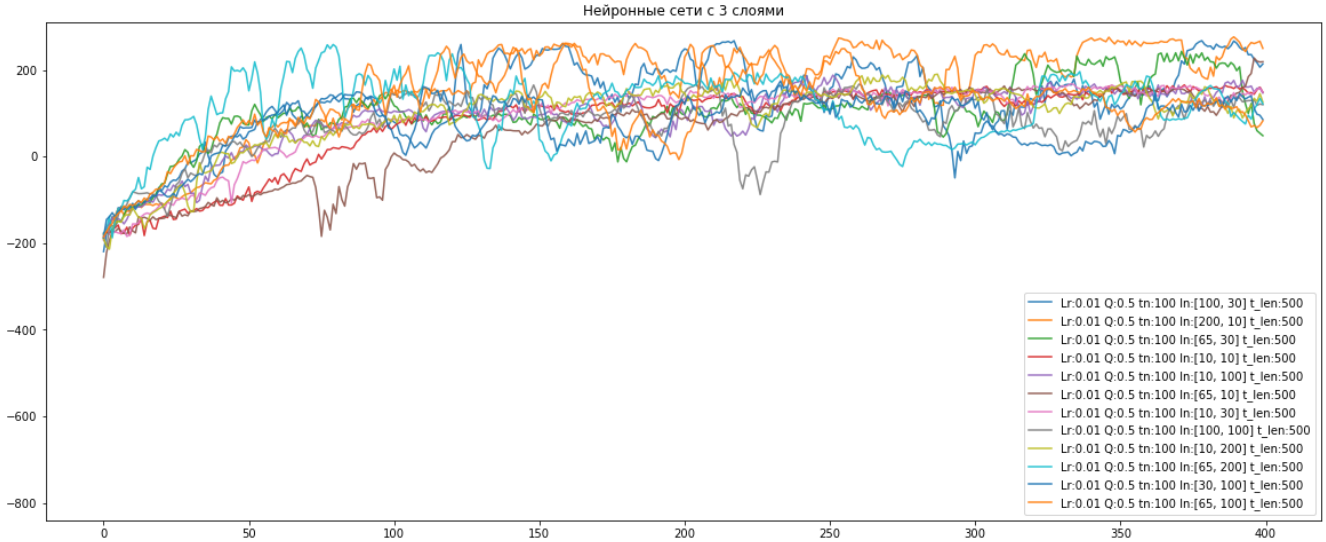
Оставляем те же параметры Q=0.5, Trajectory N = 100, Trajectory Len=500 и **400** эпизодов
Все графики сохраняют масштаб Y=[-810 310]



При очень большом количество нейронов, сеть не может удержаться на высокой награде или вообще не достигает положительной награды.

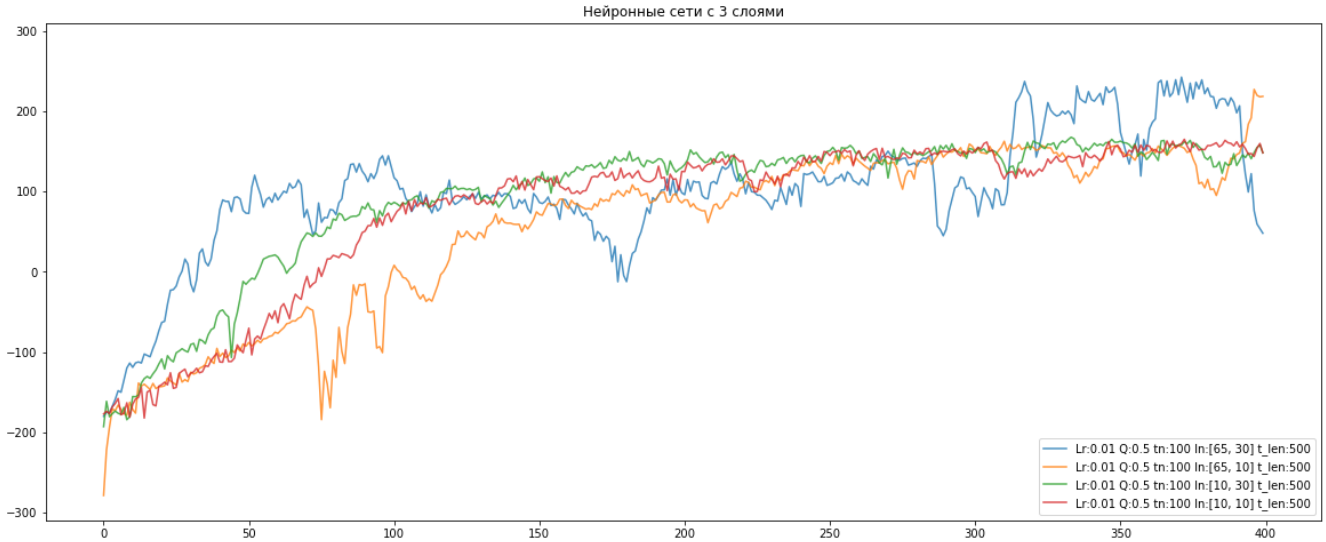


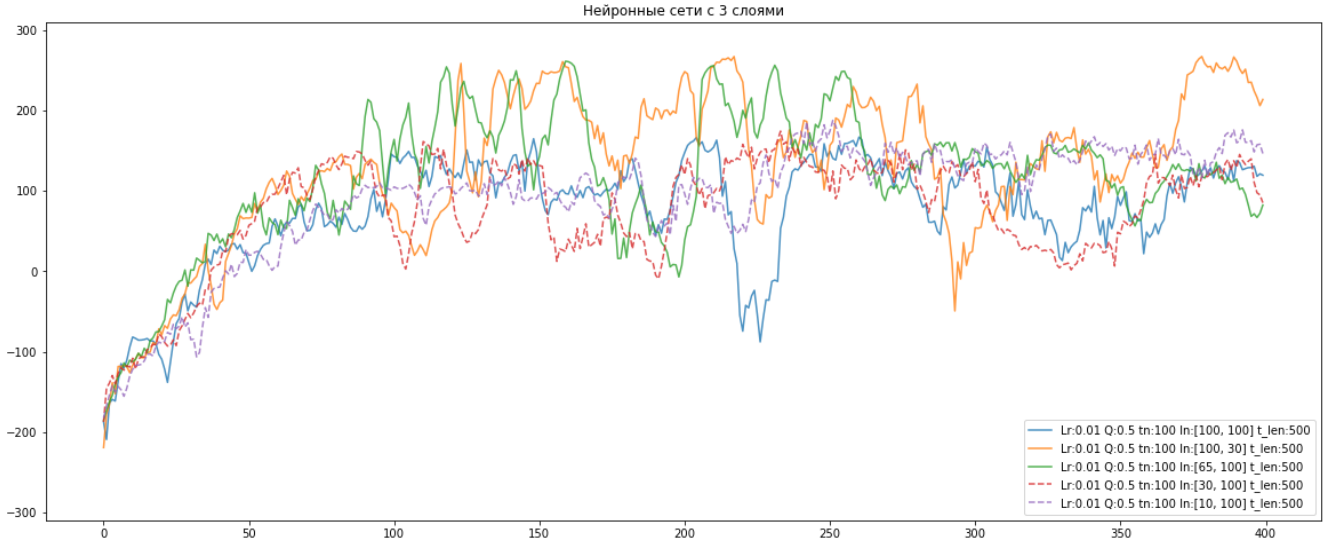
Сети которые удерживаются выше нуля, имеем слои не больше 200 нейронов в одном из слоев



Сети с меньшим в сумме количеством нейронов дают меньшую дисперсию в росте награды

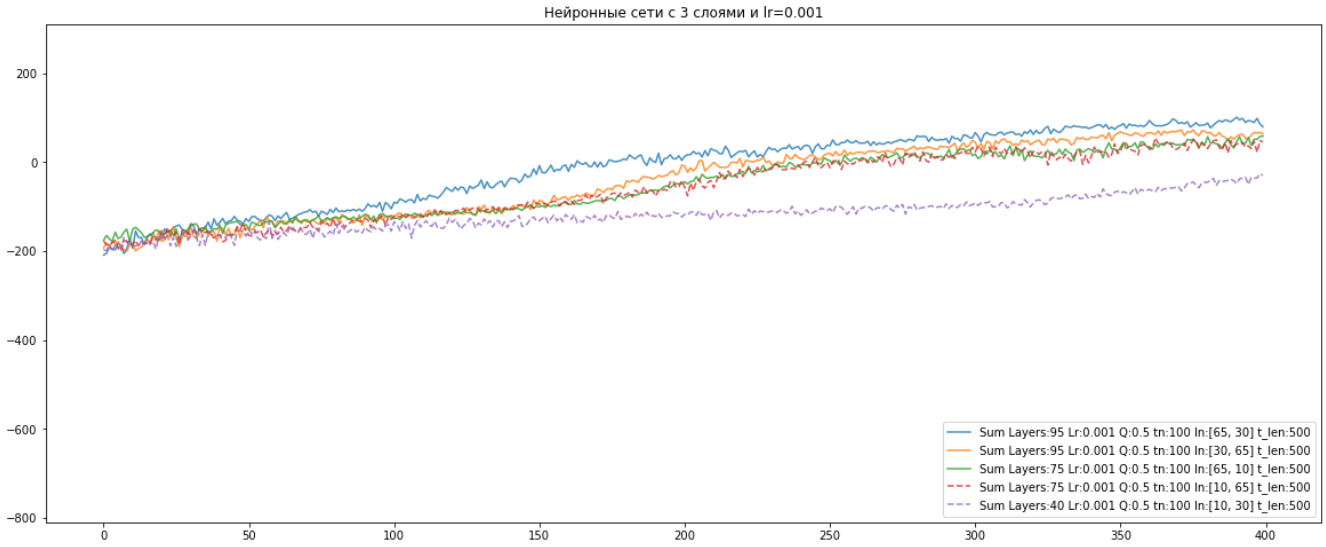
(масштаб изменен на [-310 300])





при изменении Learning Rate — скорость схождения сети уменьшается пропорционально изменению lr, сеть с меньшим количеством нейронов дает наименьшую скорость схождения

(масштаб изменен на [-810 300])



Четырехслойные сети

При увеличении количества слоев до 4 — общее поведение остается тем же. Высокое количество нейронов — дает
- больший разброс в росте награды
- новый минимум в награде

Learning Rate

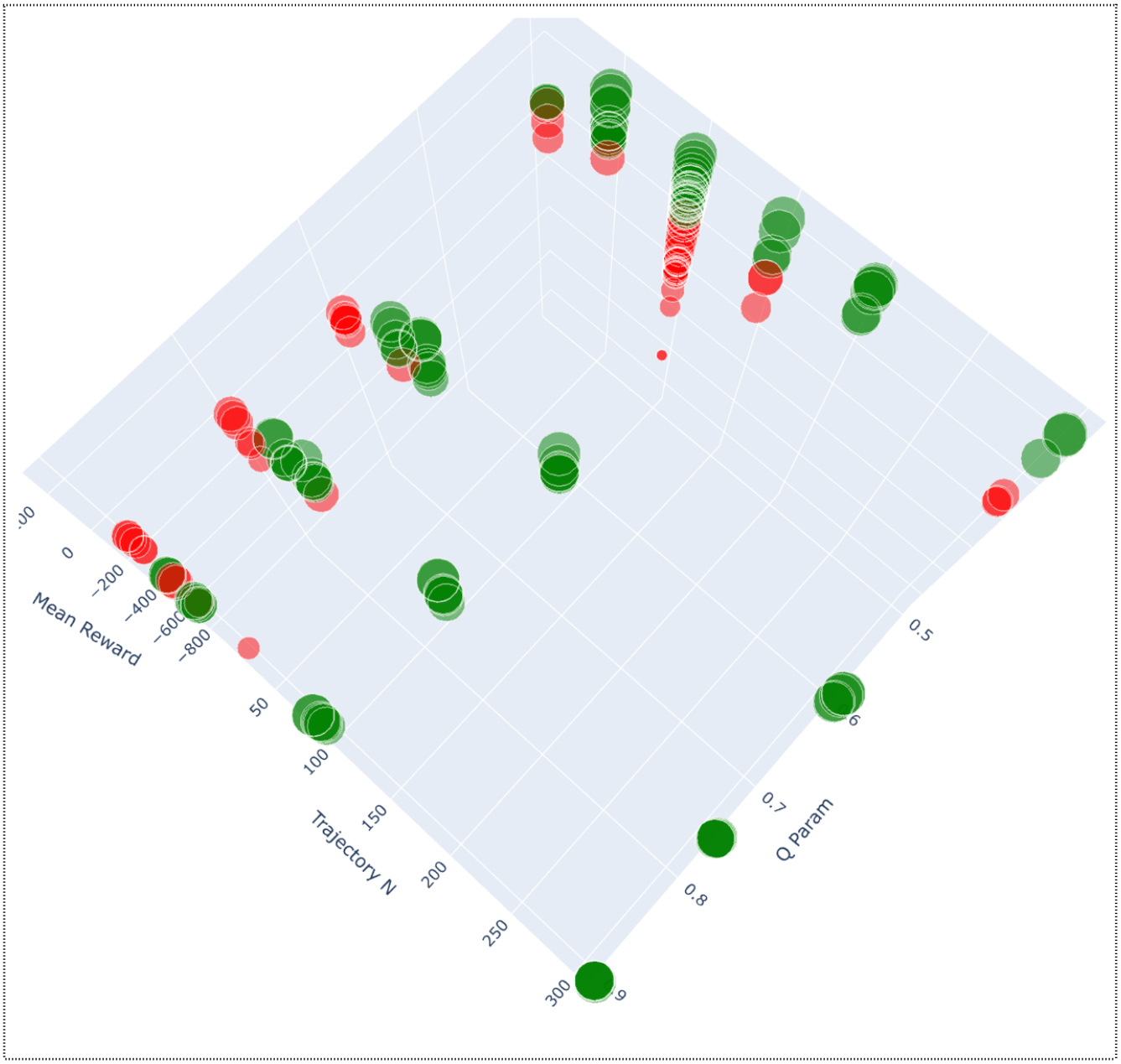
Влияние Learning Rate на разные сети. Как и в случай двухслойных чем ниже LR тем дольше сходится алгоритм

Лучшие результаты

Если выложить все параметры на 3D график,
X — Q-param
Y — Trajectory N
Z — Mean Reward

Можно увидеть что в обласит Q от 0.6-0.9 и Trajectory N [75-300] нет моделей со средними отрицательными значениями награды.

http://alexeimatusevski.com/odsrl/report_2-1-2.html
http://alexeimatusevski.com/odsrl/report_2-1-1.html

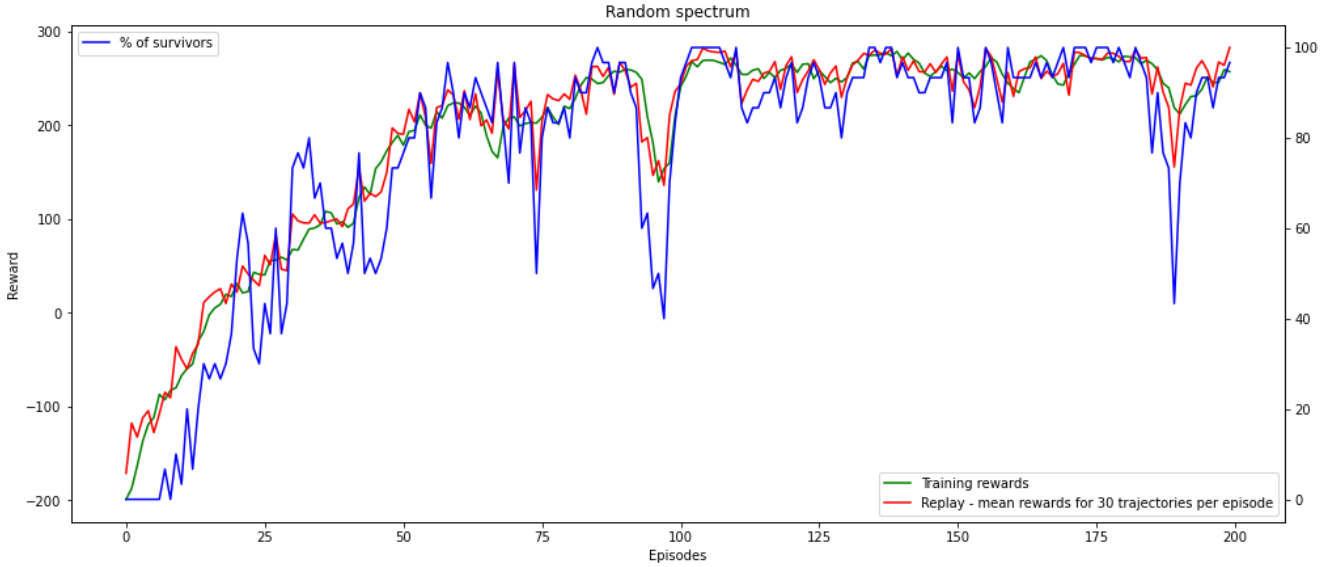


Эксперимент — это запуск алгоритма с определенными параметрами. Критерии выбора лучших результатов основываются на полученных данных из 190 экспериментов.

Для каждого эксперимента мы сохранили:
- параметра эксперимента
- состояние нейросети в каждом его эпизоде
- средние награды в течении эксперимента

Можно выбирать лучшие параметры по средней оценки во время тренировки. Графики тренировок могут содержать «провалы» в награде. Если повторить это эксперимент используя сохранные состояния весов нейросети, можно получить выборки по средним наградам а также оценить количество «выживших». Выжившие — это те траектории которые не получали награду -100 за конкретный шаг.

Из графика видно что 100% выживаемости (из 30 запусков) достигается только у нейросетей с самыми высокими средними наградами.



Минимальная награда которая дает 100% выживших (из выборки с 30ю запусками) 256. Поэтому можно предположить что эпизоды дающие такие и выше средние награды, будут подходить для лучших результатов.

Reward	Survivers
262.951434	1
269.111621	1
270.265116	1
282.396529	1
279.517435	1
278.370167	1
277.963013	1
277.992915	1
274.999818	1
280.340668	1
275.9605	1
281.941145	1
276.330687	1
282.151603	1
256.508145	1
266.308203	1
278.3295	1
277.618083	1
274.333956	1
271.442862	1
270.33336	1
277.222227	1
273.670604	1
280.847168	1

Итого список топ 7 параметров:

	lr	episode_n	trajectory_n	trajectory_len	q_param	layers_cnt	layers_n1	layers_n2	layers_n3
0	0.01	200	300	500	0.7	1	200	-	-
1	0.01	200	150	500	0.5	1	100	-	-
2	0.01	200	300	500	0.5	1	150	-	-
3	0.01	200	300	500	0.7	1	100	-	-
4	0.01	200	300	1000	0.5	1	100	-	-
5	0.01	400	200	500	0.5	1	100	-	-
6	0.01	400	200	500	0.5	1	150	-	-