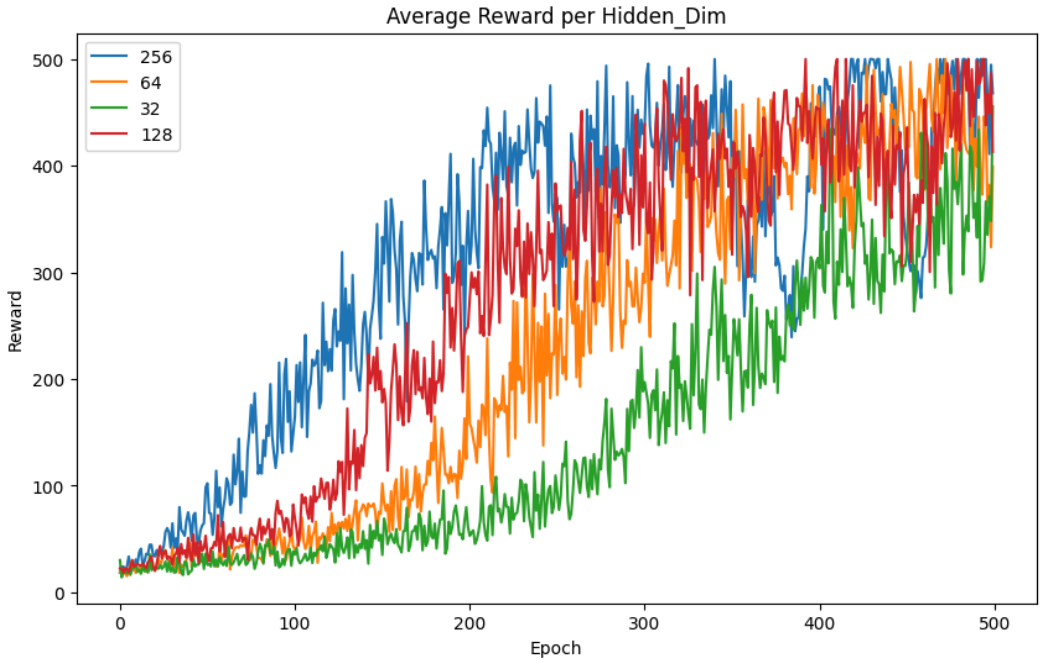
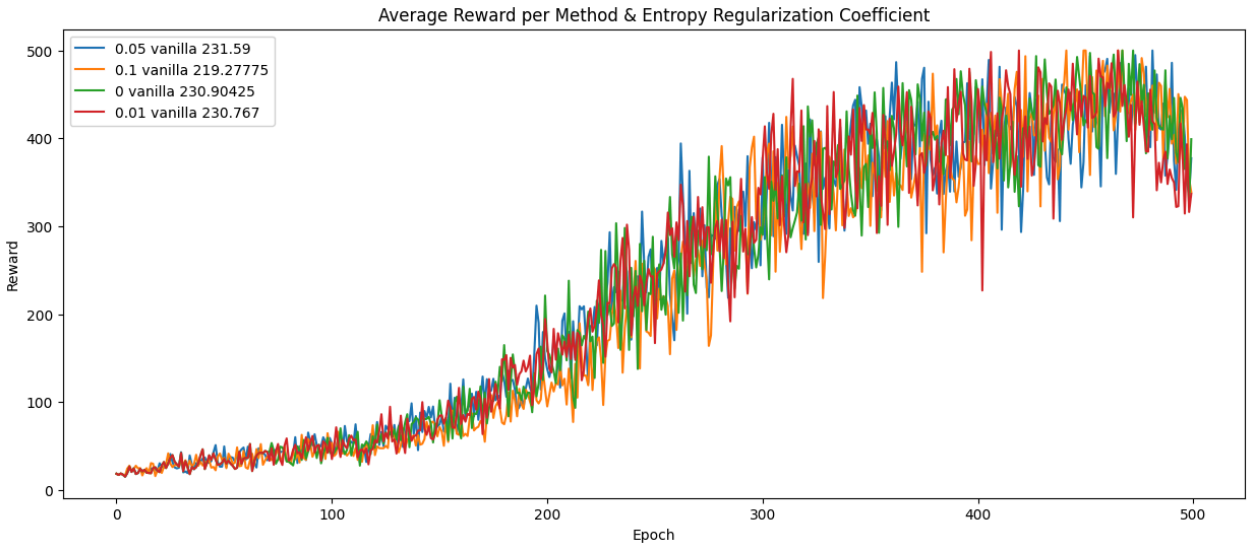
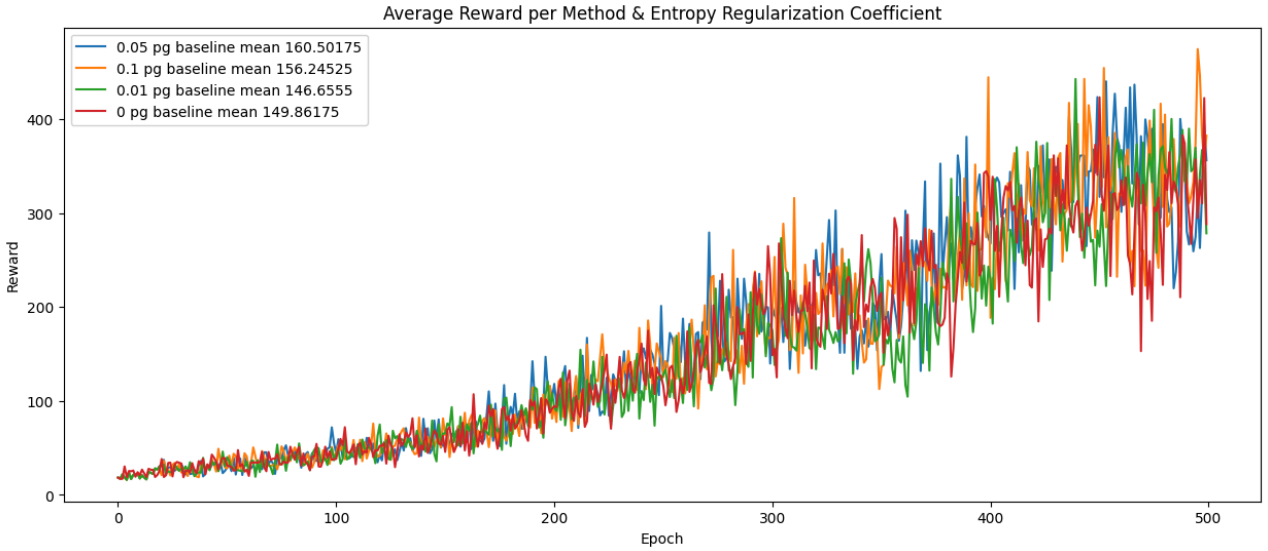
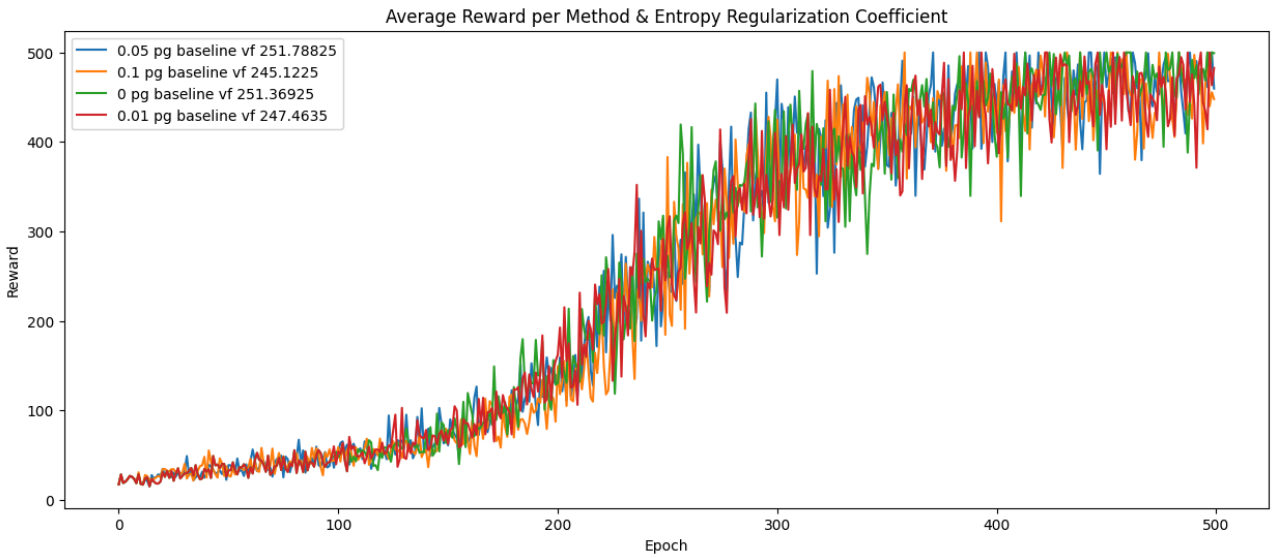
Сначала были произведены эксперименты по подбору гиперпараметров (hidden_dim) у MLP. На графике видно, что hidden_dim = 32 заметно отстает по reward от других значений и сходится медленнее. У остальных hidden_dim финальные результаты по reward особо не разнятся – все достигают так или иначе оптимума. Хотя чем больше hidden_dim, тем быстрее растет reward с эпохой, выберем 64 как компромисс между качеством и вычислительной эффективностью.

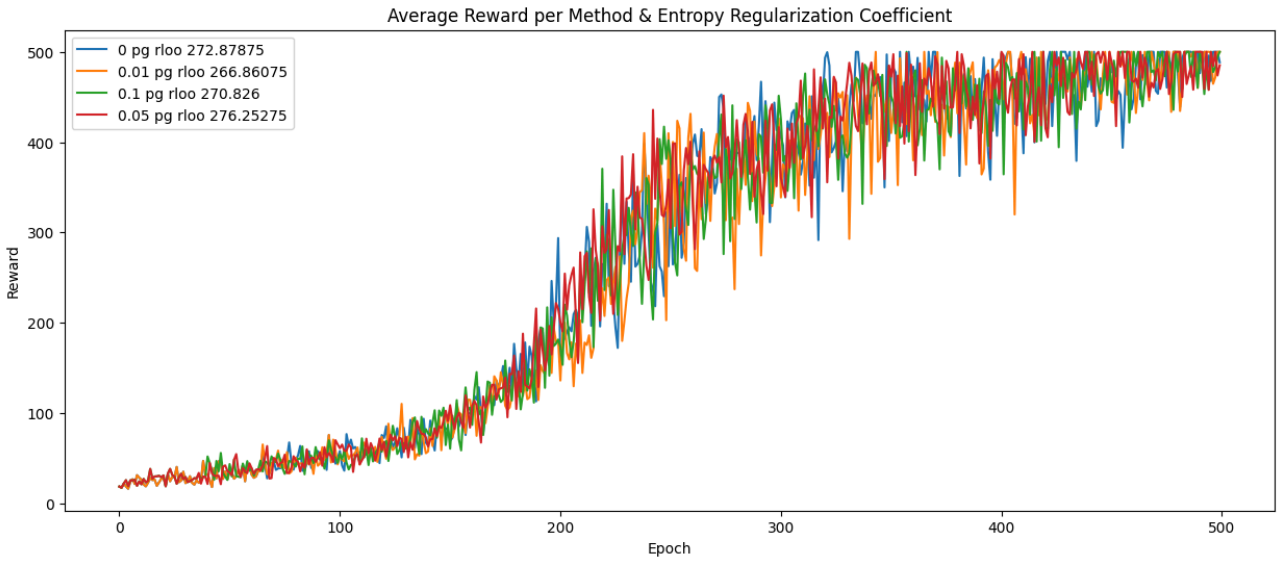


Далее идет серия экспериментов на сравнение функций потерь + применение регуляризации энтропию с разным параметром.


Average Reward per Method & Entropy Regularization Coefficient
0.05 vanilla 231.59
0.1 vanilla 219.27775
0 vanilla 230.90425
0.01 vanilla 230.767
Reward
Epoch
0
100
200
300
400
500


Average Reward per Method & Entropy Regularization Coefficient
0.05 pg baseline mean 160.50175
0.1 pg baseline mean 156.24525
0.01 pg baseline mean 146.6555
0 pg baseline mean 149.86175
Reward
Epoch
0
100
200
300
400
500


Average Reward per Method & Entropy Regularization Coefficient
0.05 pg baseline vf 251.78825
0.1 pg baseline vf 245.1225
0 pg baseline vf 251.36925
0.01 pg baseline vf 247.4635
Reward
Epoch
0
100
200
300
400
500



При применении любой функции потерь значимого результата от значения гиперпараметра при энтропии не наблюдается.

Сами методы отличаются по качеству между собой. Справа в легенде указано среднее по всем avg_rewards, где avg_reward – среднее в эпизоде. Чем больше это значение, тем лучше.

Сравнивая по нему и по графикам, видно, что хуже всего обучался PG с бейзлайном средняя награда (все показатели в диапазоне 149-160). И все кривые не достигают значения 500. Такой метод приводит к *занижению* advantages на ранних этапах, замедляя обучение.

Далее по качеству идет vanilla с reward 219-230. Видно, что на эпохах 400-500 сильный разброс от эпохе к эпохе. Зато не требует дополнительных гиперпараметров и прост в реализации.

Далее PG с Value Function, тут уже достаточно много достигаем max награды = 500. Методы с VF и RLOO быстрее сходятся к оптимуму (скачок раньше происходит). Даёт более точную оценку advantages. Требует обучения второй сети, т.е. x2 ко времени.

И самый лучший вариант – RLOO. Меньше всего разброс и больше всего достигается max награда. Для выполнения задания из блока 2 возьмем политику обученную RLOO с entropy coef = 0.05 (максимальный показатель по средней награде). Leave-One-Out стратегия эффективно снижает bias при малом batch size. Наиболее стабильный: минимальный "откат" среди всех методов после достижения оптимума. Но требует батч из 2+ эпизодов, что занимет чуть больше памяти.