**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МОЭВМ**

**ОТЧЕТ**

**по лабораторной работе №2**

**по дисциплине «Обучение с подкреплением»**

**Тема: Реализация PPO для среды MountainCarContinious-v0**

|  |  |  |
| --- | --- | --- |
| Студент гр. 0306 |  | Голубев А.Н. |
| Преподаватель |  | Глазунов С.А. |

**Цель работы**.

Реализация PPO для среды MountainCarContinuous-v0. Исследование зависимости результатов от параметров и добавления нормализации преимуществ.

**Задание.**

1. Реализация PPO
2. Добавление нормализации
3. Изменение длины траектории
4. Подбор оптимального значения clip ratio
5. Сравнение обучения при разных количествах эпох

**Выполнение работы.**

1. **Реализация PPO**

Основными созданными сущностями для алгоритма PPO являются нейронные сети Actor и Critic.

Actor рассчитывает потенциальное действие, а Critic оценивает его выгоду (advantage) на основании полученной награды. Таким образом, Critic корректирует дальнейшие действия Actor.

Входными параметрами для Actor, так как он выбирает действие, являются Observation Space среды MountainCarContinuous-v0, описание которых представлено на рисунке 1, а на выходе численный показатель движения влево или вправо в зависимости от знака числа в диапазоне от [-1 до 1]: движение влево (индекс 0) или движения в право (индекс 1).

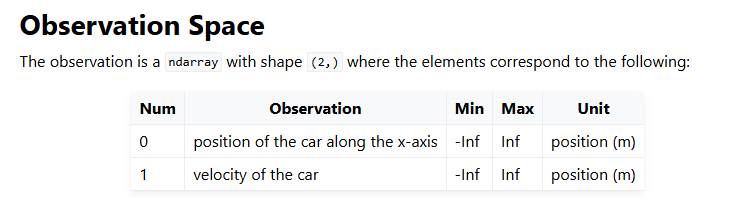


Рис. 1 – Пространство наблюдений в среде MountainCarContinuous-v0

Данная реализация алгоритма регулируется набором параметров:

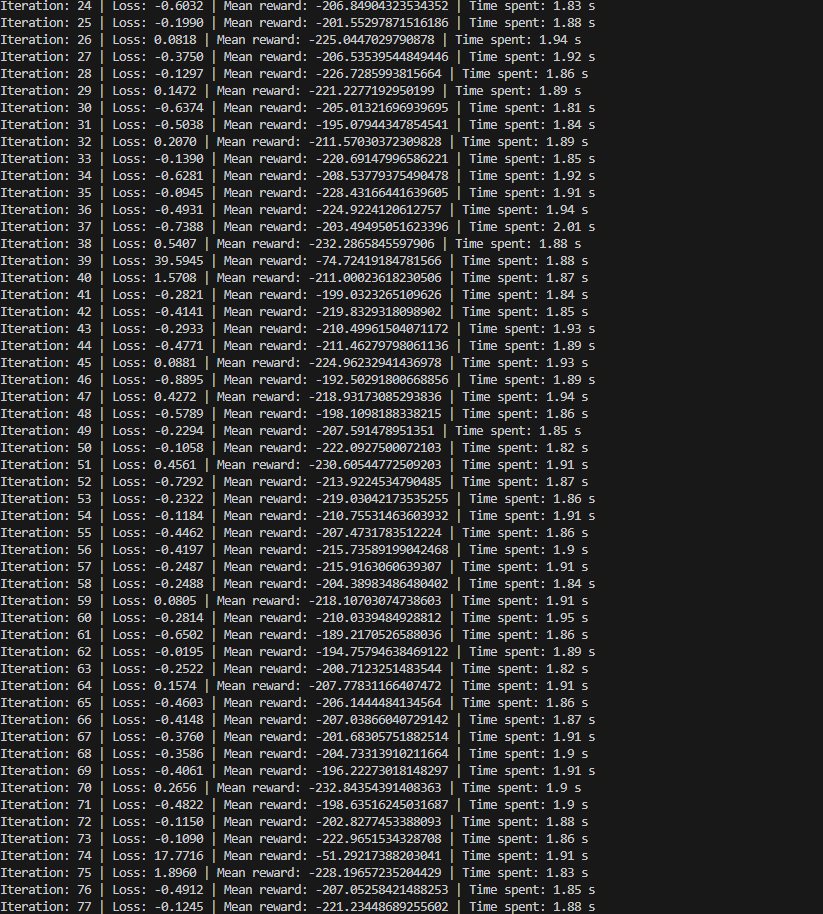
* Число итераций (iterations)
* Количество шагов для сбора одного батча (steps)
* Число эпох (epochs)
* Размер батча (mini\_batch\_size)
* Коэффициент дисконтирования (gamma)
* GAE lambda – параметр, определяющий баланс между немедленным и будущим вознаграждением
* Коэффициент обрезки PPO (clip\_ratio)
* Коэффициент потери значения (value\_coef)
* Коэффициент энтропийного бонуса (entropy\_coef)
* Шаг обучения при градиентном спуске (lr)

1. **Добавление нормализации**

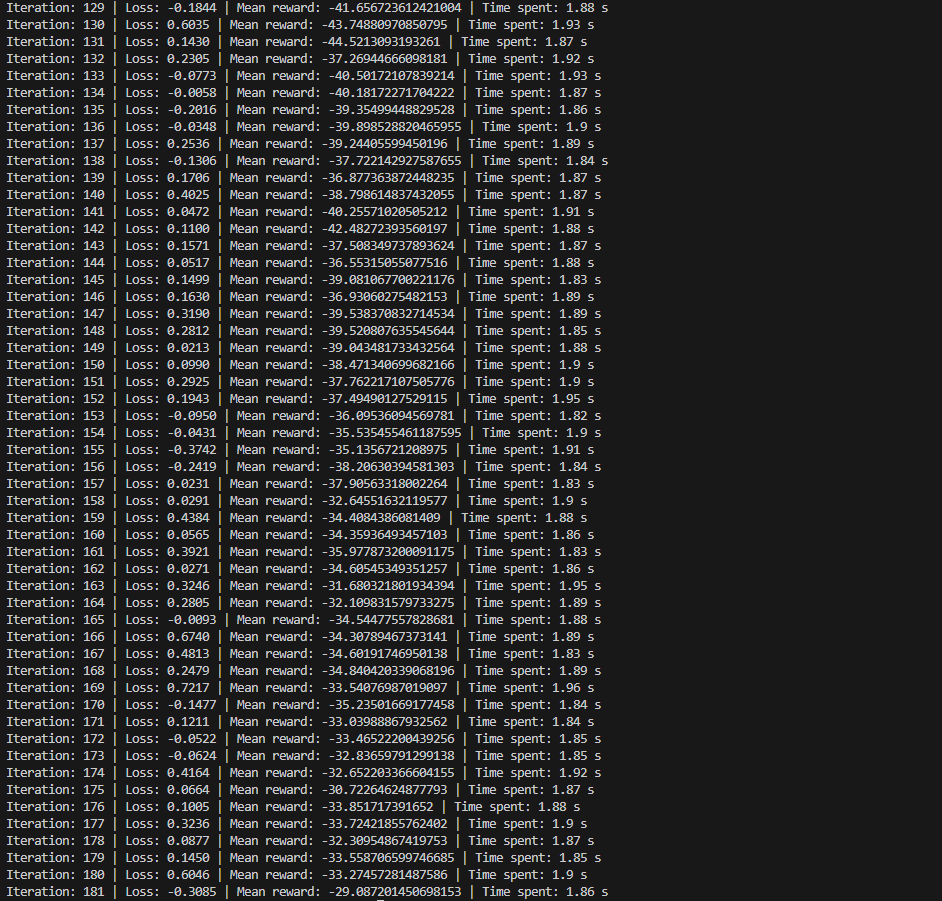
Нормализация преимуществ в алгоритме PPO позволяет стабилизировать обучение и улучшить эффективность использования данных.

В рамках данной реализации алгоритма нормализация происходит в конце метода \_\_returns\_and\_advantages() класса PPOAgent.

Сравнение результатов обучения с нормализацией и без представлено на рисунках 2 — 3.

Рис. 2 — результаты без нормализации

Как можно заметить, в случае отсутствия нормализации средняя награда в среднем принимает значения ниже -200 и со временем не увеличивается (за исключением тех итераций, где машина достигает флага, из-за чего к награде прибавляется 100 очков).

Рис. 3 — результаты при наличии нормализации

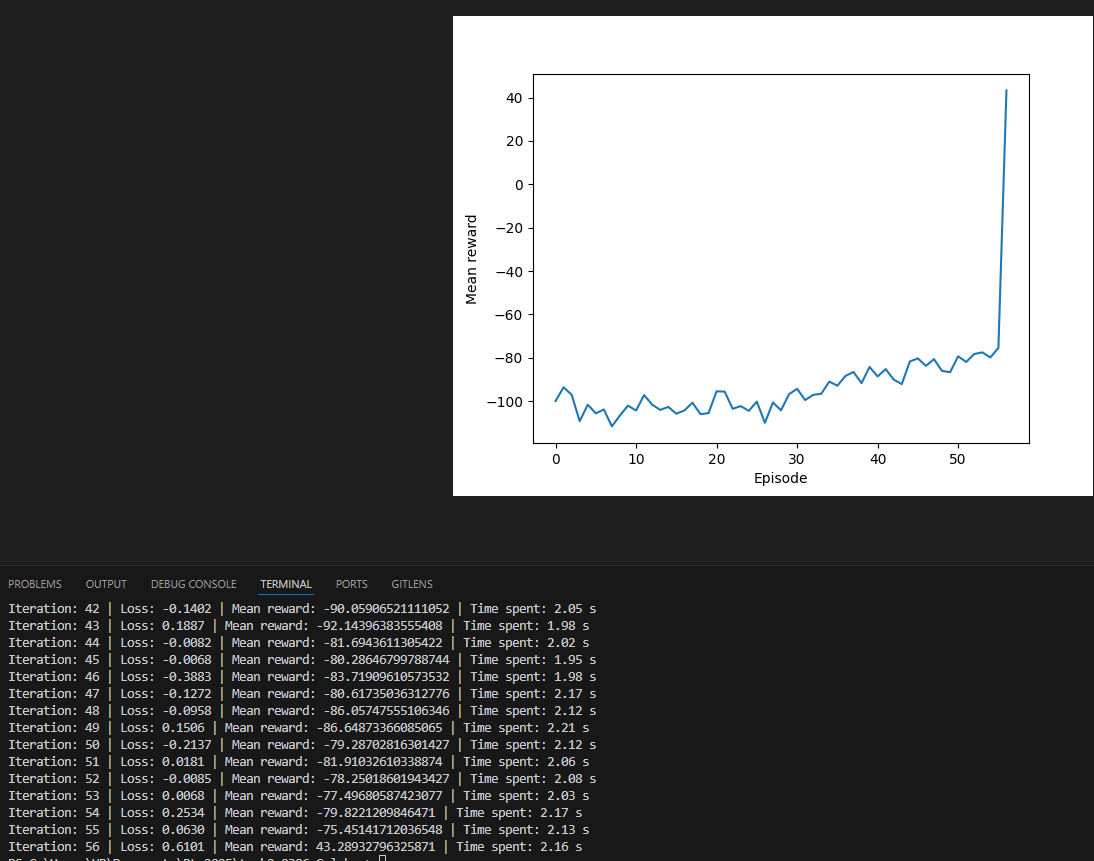
В случае наличия нормализации среднее значение награды значительно возросло. Также можно заметить, что в данном случае значения средней награды постепенно растут (на первой итерации средняя награда равнялась -115).

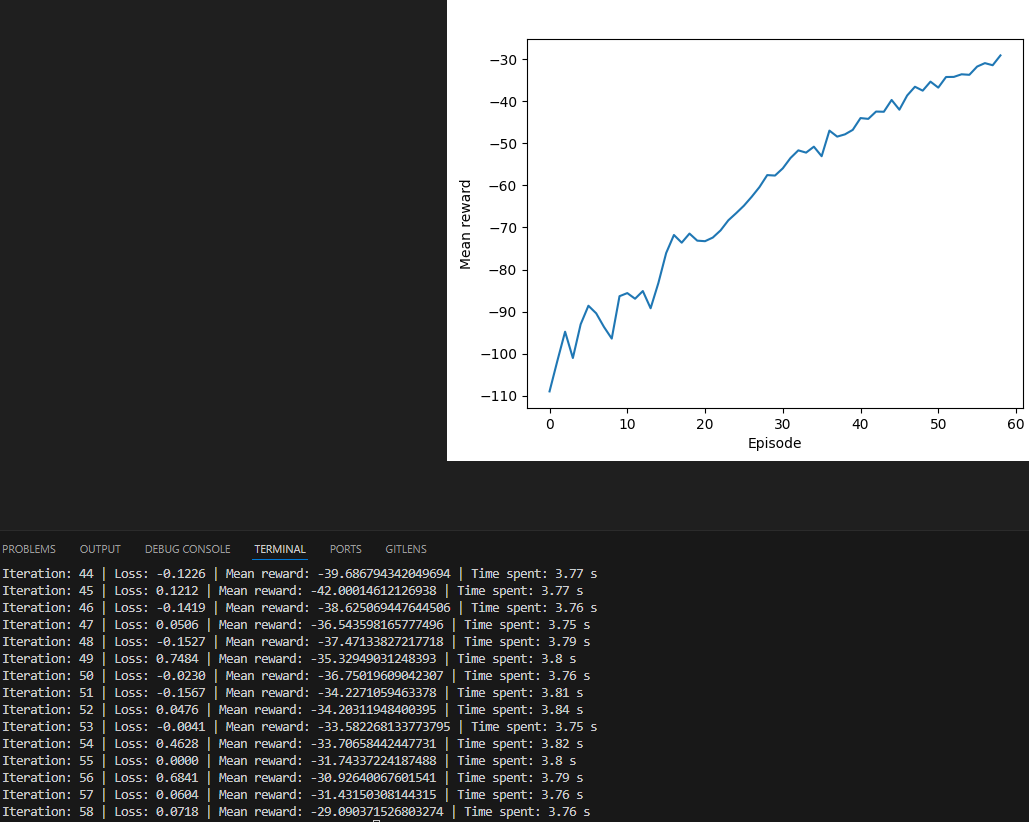
Все следующие эксперименты проводились с наличием нормализации.

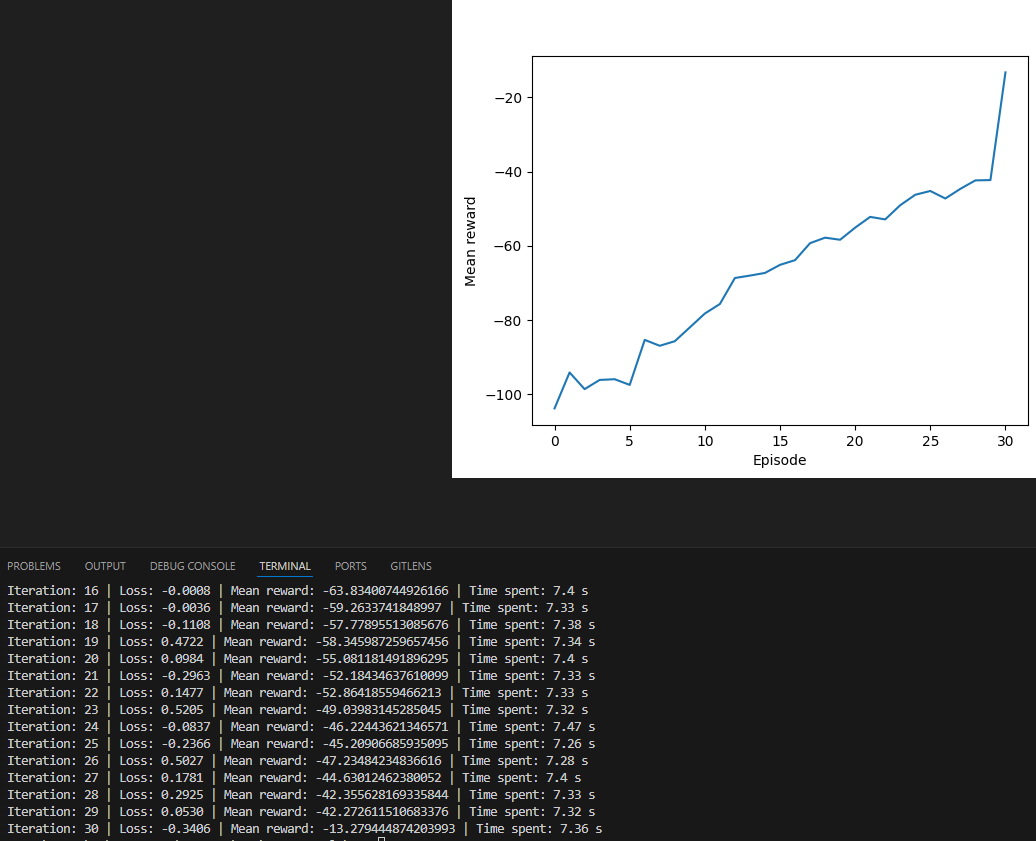
1. **Изменение длины траектории**

Для эксперимента были выбраны следующие длины траекторий: 1024, 2048, 4096.

Результаты обучения представлены на рисунках 4 — 6.

Рис. 4 — результат обучения при steps = 1024

Рис. 5 — результат обучения при steps = 2048

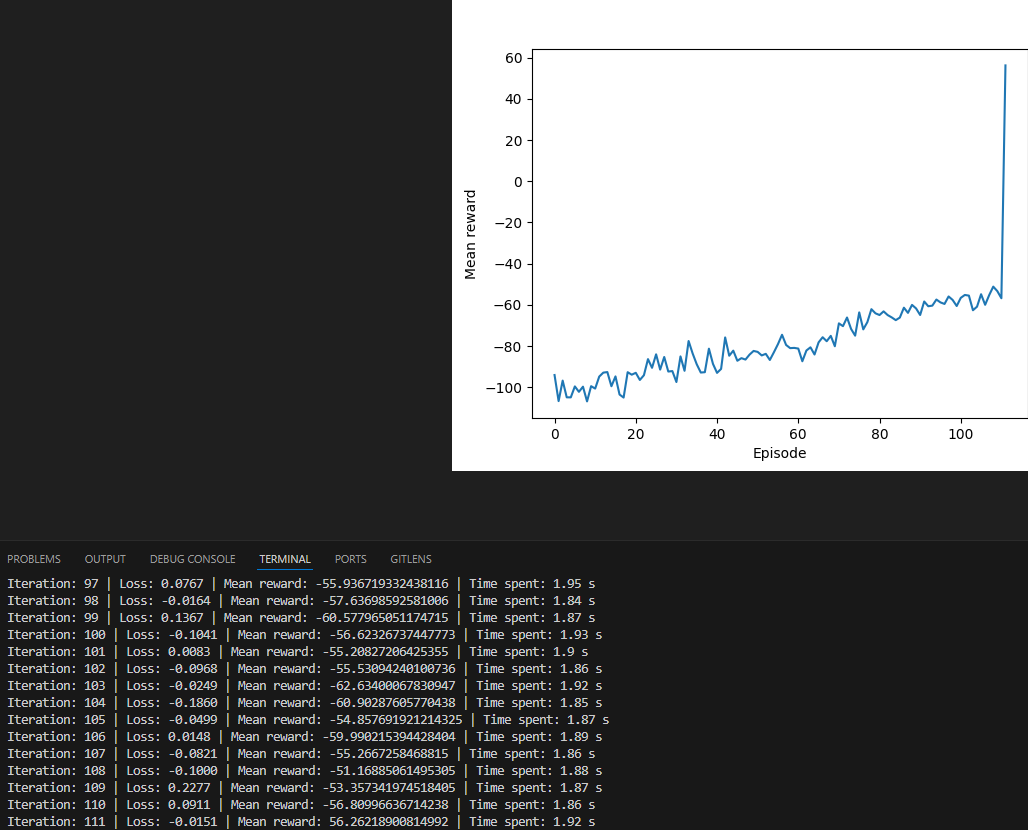
Рис. 6 — результат обучения при steps = 4096

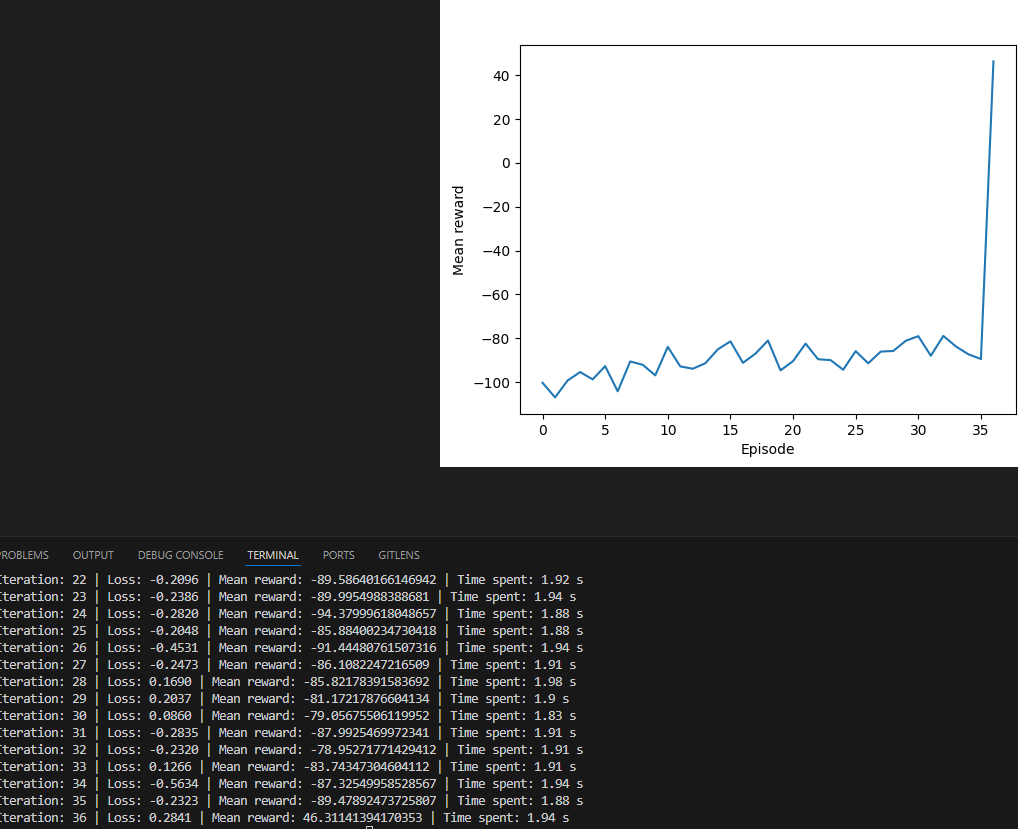
По результатам обучения можно заметить следующее:

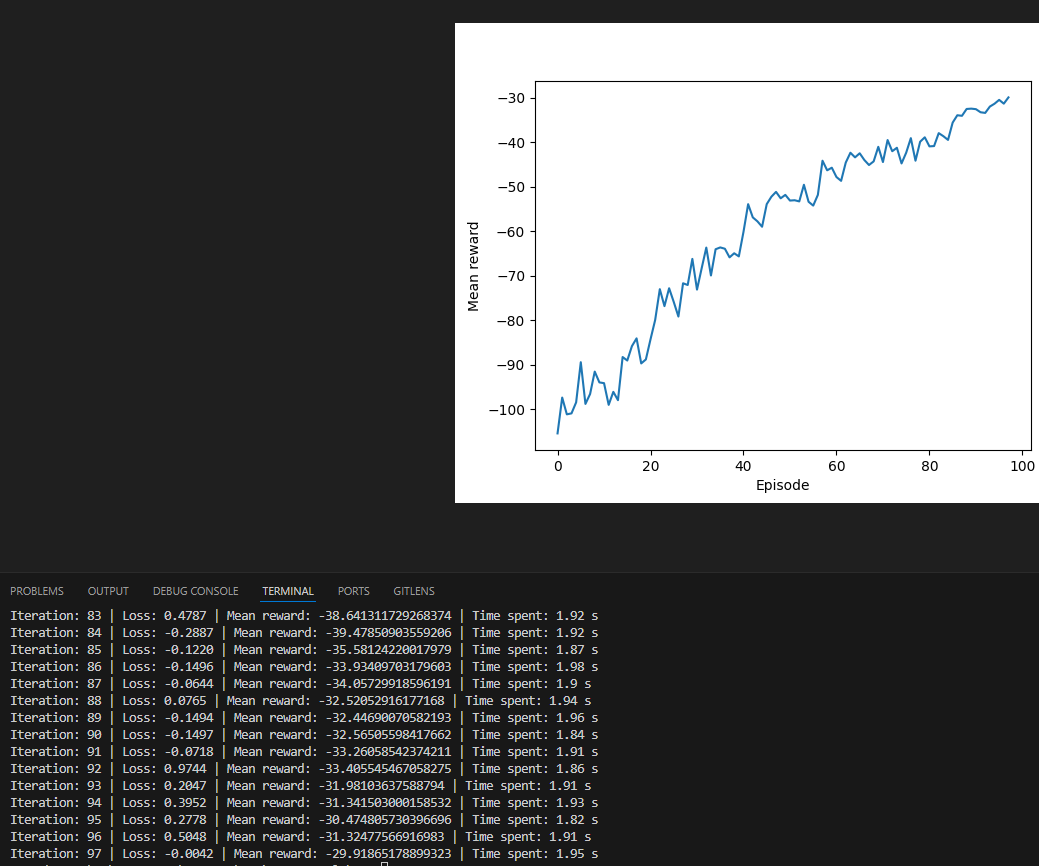
* При увеличении steps время, затрачиваемое на одну итерацию, тоже растет
* По мере увеличения steps уменьшается резкость и частота скачков в значениях средней награды

1. **Подбор оптимального значения clip ratio**

Для данного эксперимента были выбраны следующие значения clip ratio: 0.08, 0.2, 0.5. Результаты продемонстрированы на рисунках 7 — 9.

Рис. 7 — результат обучения при clip\_ratio = 0.08

Рис. 8 — результат обучения при clip\_ratio = 0.2

Рис. 9 — результат обучения при clip\_ratio = 0.5

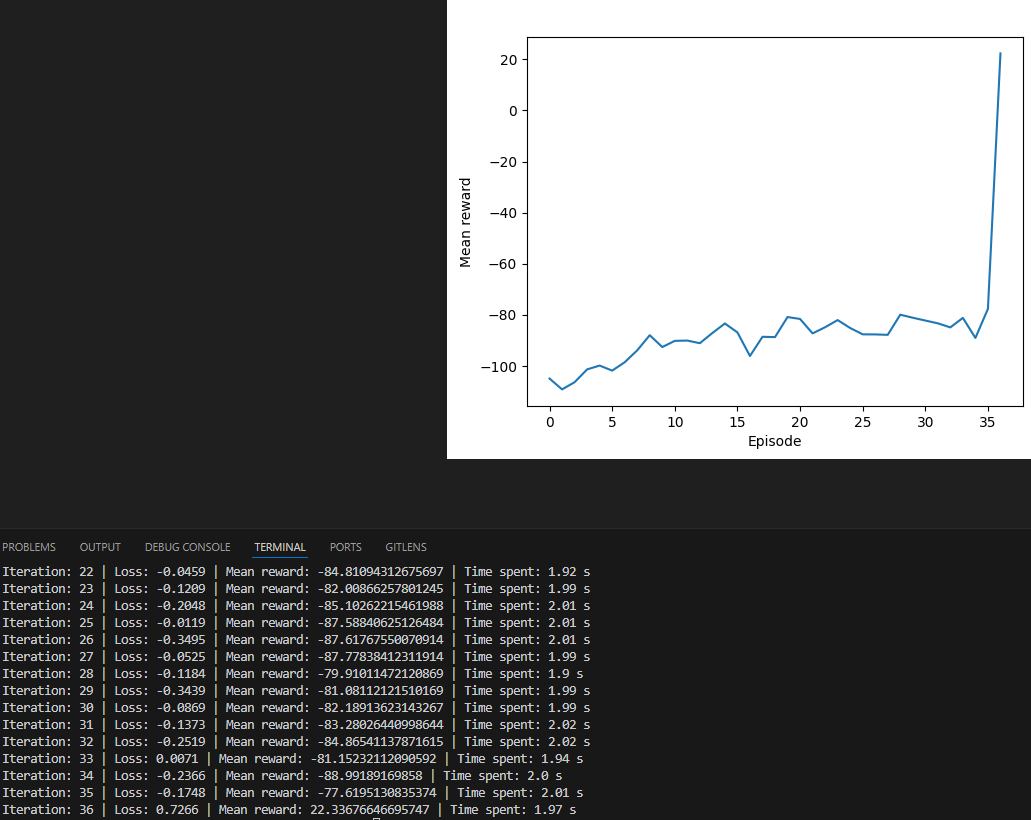
Можно сделать следующие выводы:

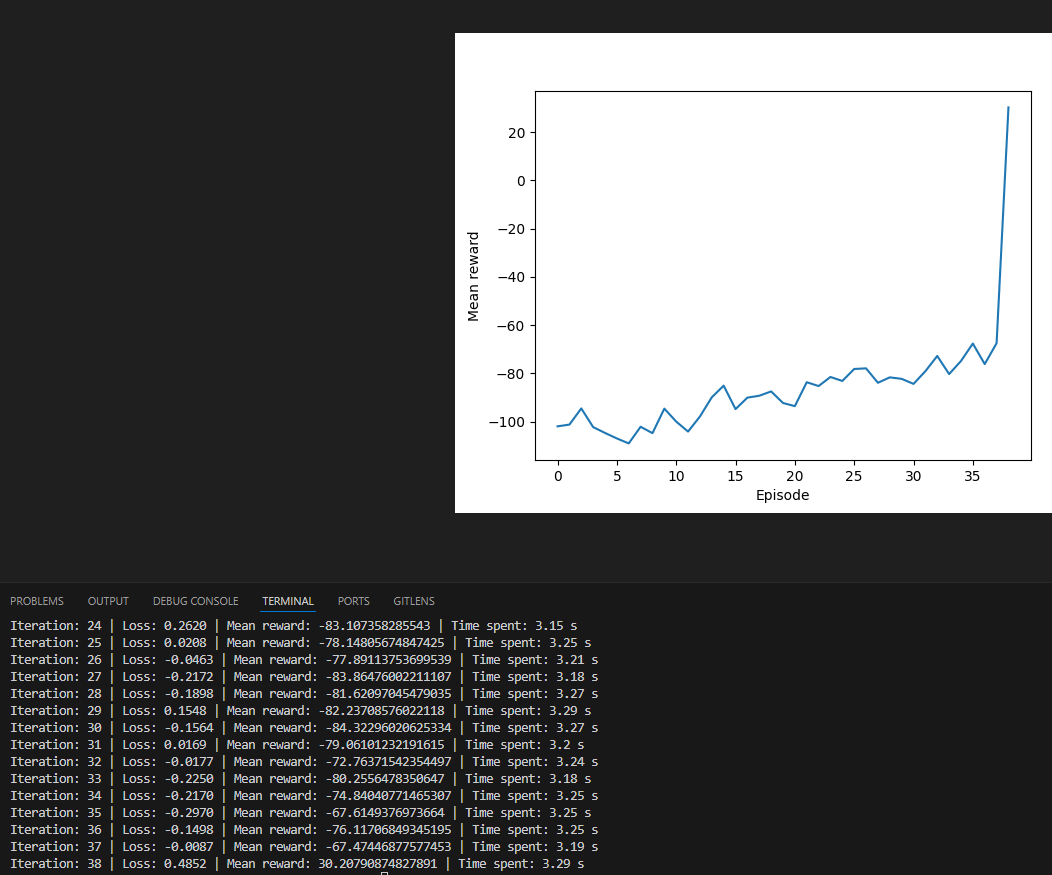
* Быстрее всего пород для окончания обучения был пройден при clip\_ratio = 0.2
* Больше всего итераций потребовалось при clip\_ratio = 0.08
* При clip\_ratio = 0.5 обучение проходило без сильных скачков в среднем значении награды
* При clip\_ratio = 0.2 машина в режиме тестирования чаще достигает цели (флага на холме)
* При clip\_ratio = 0.5 машина в режиме тестирования относительно чаще предыдущих случаев находилась в районе дна котлована

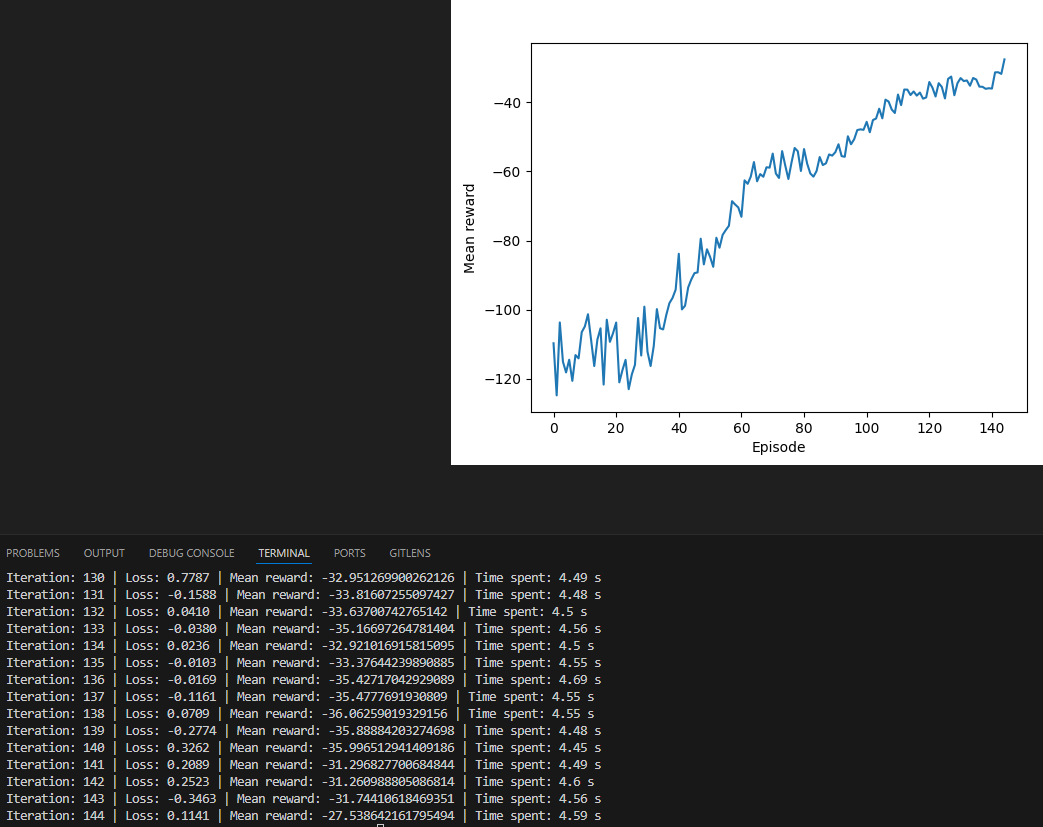
Из всего вышеперечисленного можно сделать вывод, что оптимальным является значение clip\_ratio = 0.2.

1. **Сравнение обучения при разных количествах эпох**

Были взяты следующие значения количества эпох: 10, 20, 30. Результаты представлены на рисунках 10 — 12.

Рис. 10 — результат обучения при epochs = 10

Рис. 11 — результат обучения при epochs = 20

Рис. 12 — результат обучения при epochs = 30

По результатам можно заметить следующее:

* По мере увеличения числа эпох растет среднее время итерации
* При epochs = 30 не было таких же резких скачков значений, как при epochs = 10 и epochs = 20

**Выводы.**

Была выполнена реализация PPO для среды MountainCarContinuous-v0. Было проведено исследование влияния изменения параметров алгоритма на результаты, а также эффективность применения нормализации к значениям advantages.