BipedalWalker

Непрерывные действия дискретизированы на 4 корзины, в каждой взято центральное значение. Общее количество действий – 256.

ReplayBuffer - 10^6 элементов, с циклическим обновлением.

Нейронная сеть: 2 полносвязанных слоя с активацией ReLU с размерами 1500 и 700, и выходной полносвязный слой размер количество действий.

Оптимизационный алгоритм – Adam с learning rate 0.0001.

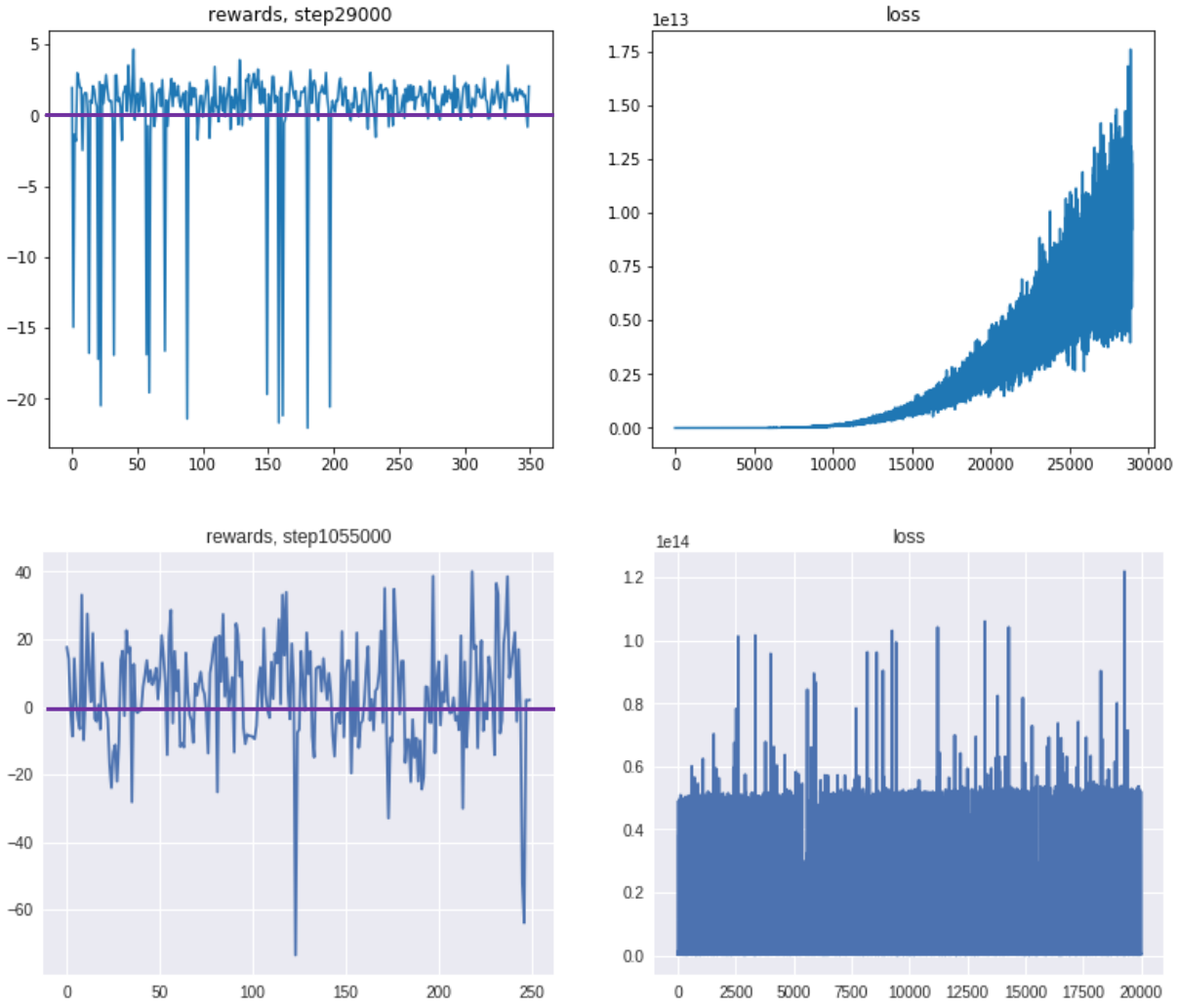
Для выбора действия использовалась epsilon-greedy политика, с начальным значением epsilon=0.95 и затуханием на каждом шаге равном 0.99991. Минимальное значение epsilon – 0.05.

После каждого шага в среде модель обучалась на случайном batch из ReplayBuffer размером 64.

Так как агент в процессе обучения в любом случае падает и получает за это -100, то в качестве критерия качества рассматривается выражение reward+100, которое проще анализировать.

Количество шагов до значимого улучшения наград около миллиона.

Видно, что агент часто получает перед падением положительную награду, что подтверждает факт обучения.



Найденные гиперпараметры использовались в качестве стартовых при обучении модель в Hardcore окружении.

BipedalWalkerHardcore

Непрерывные действия дискретизированы на 6 корзины, в каждой взято центральное значение. Общее количество действий – 1296. Для получения более мелкого шага для действий применено сжатие максимального диапазона до 0.8 от начального.

Нейронная сеть: 2 полносвязанных слоя, активация PReLU с размерами 1500 и 1000 и выходной полносвязный слой размер количество действий.

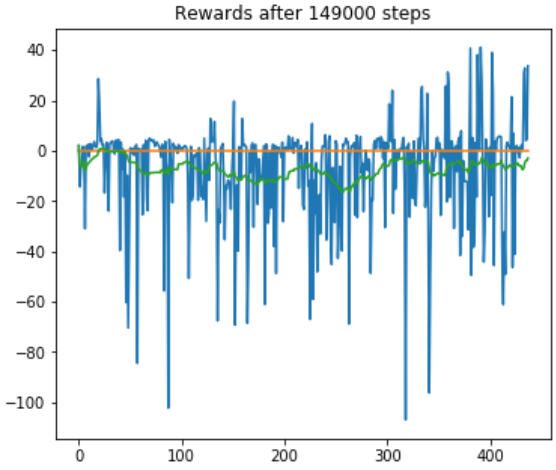
Оптимизационный алгоритм – Adam с learning rate 0.0001.

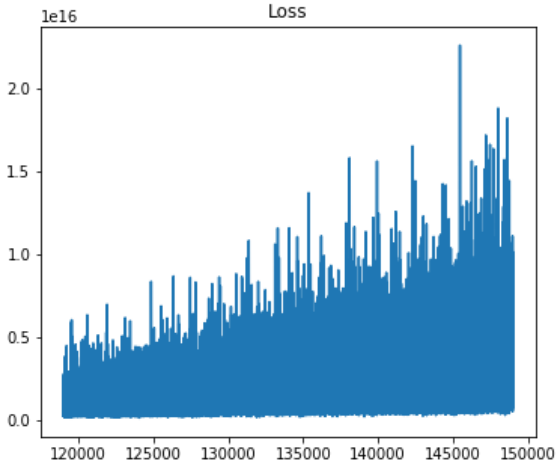
ReplayBuffer - 10^6 элементов, с циклическим обновлением.

Для выбора действия использовалась epsilon-greedy политика, с начальным значением epsilon=0.95 и затуханием на каждом шаге равном 0.99991. Минимальное значение epsilon – 0.01. Уменьшено для улучшения результатов.

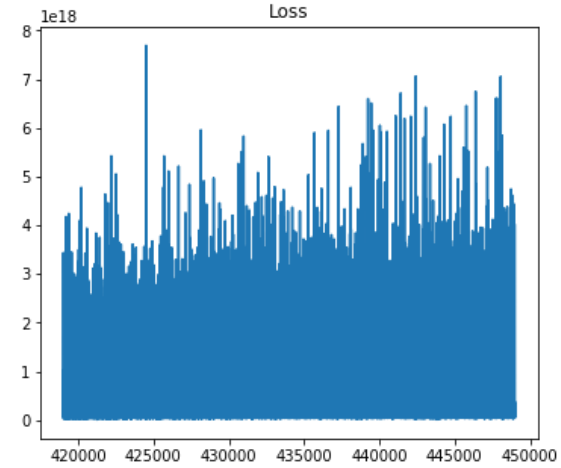
После каждого шага в среде модель обучалась на batch-е размером 128.

В качестве reward'а за эпизод выводится выражение reward+100, которое более явно показывает, какую награду получил агент до падения.





Награда после ~300000 шагов – 450 эпизодов.



Награда после ~450000 шагов – 750 эпизодов. Зеленая линия - средняя награда за последние 40 эпизодов.

По графикам видно, что в результате обучения агент стал получать награду больше ноля значительно чаще чем до обучения. Средняя награда за 40 эпизодов стала чаще превышает ноль. Количество эпизодов с наградой меньше -40 сократилась.

Для надежного подтверждения необходимо запустить обучение несколько с различными значениями seed для генератора случайных чисел.

Для улучшения качества и ускорения обучения можно продолжить поиск оптимальных гиперпараметров и выполнить нормализацию состояний:

- нормализация состояний
  формально параметры состояние неограниченны и выполнить их предварительное сэмплирование для определения среднего и стандартного отклонения невозможно. После обучения модели можно выполнить нормализацию на основе состояний из ReplayBuffer. При рассмотрении буфера размером 450000 элементов можно заметить, что компоненты вектора состояния ограничены и не превосходят по модулю 4. В силу ограниченности и сопоставимости компонент можно предположить, что улучшения после нормализации не будут принципиальными.
- количество шагов дискретизации
  В работе https://robintyh1.github.io/papers/tang2018discretizing.pdf показано, что в принципе разбиения на 7 шагов в данной задаче уже достаточно для достижения значительных результатов
- learning rate, batch_size, размер скрытых слоев, активационные функции, оптимизатор
  подбор этих параметров производился во многом интуитивно, что оставляет возможность для значительного улучшения при более систематическом подходе.