Reinforcement learning 1

Введение в RL

Бобков Денис 192

Supervised learning

Что есть:

- Набор объектов и ответов на них (x, y)
- Семейство алгоритмов $a_\theta(x) \to y$
- Функция потерь $L(y, a_\theta(x))$

Хотим:

$$\theta' = \underset{\theta}{\operatorname{argmin}} L(y, a_\theta(x))$$

Supervised learning

Но есть проблема… Что, если данных нет?

Пример:

- Мы – YouTube, хотим внедрить рекламный баннер
- Имеем признаки видео и набор банеров
- Наша цель – максимизировать кол-во кликов

Что делать?

Решение

Самая простая идея:

- Сделать наивную инициализацию
- Собрать данные
- Обучиться на данных
- Повторить процесс

Ещё пример

Хотим создать терминатора, а точнее научить ходить.

Что у нас есть:

- Очень злой робот
- Куча частей, соединённых моторчиками

Что хотим:

- ~~Уничтожить человечество~~
- Научить машину ходить

Решение (да, опять)

Самая простая идея:

- Сделать наивную инициализацию
- Собрать данные
- Обучиться на данных
- Повторить процесс

Чуть формальнее

Наблюдение
(Observation)

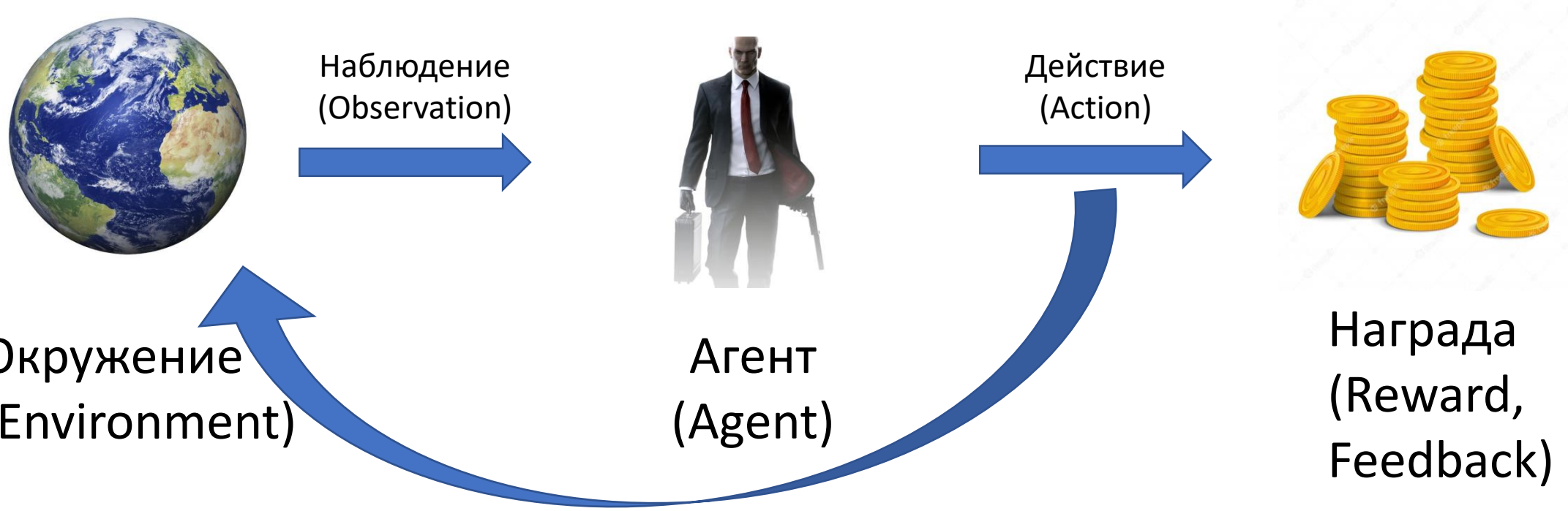
Действие
(Action)

Окружение
(Environment)

Агент
(Agent)

Награда
(Reward,
Feedback)

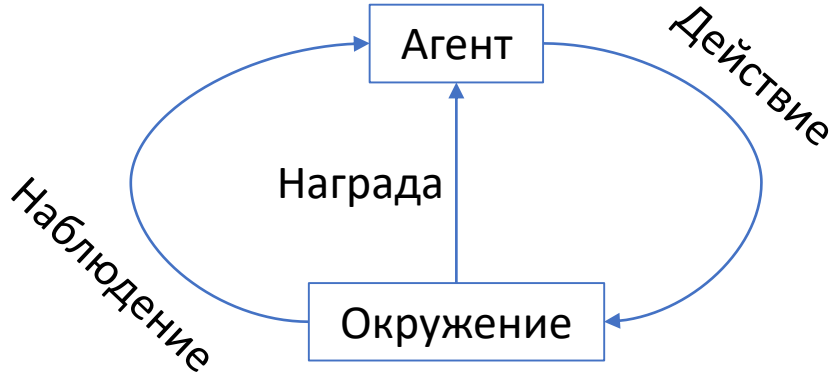
Чуть формальнее



- Feedback – может быть не дифференцируем
- Агент – наша программа
- Окружение – может быть чёрным ящиком

Decision making process

MDP

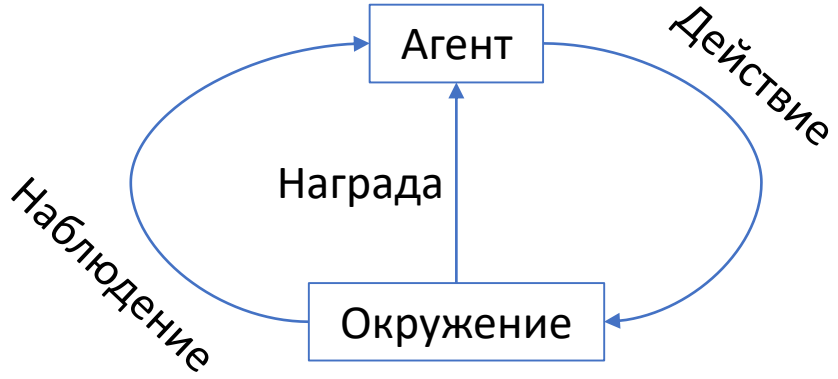


Markov Decision Process

- Environment states: $s \in S$
- Agent actions: $a \in A$
- Reward: $r \in \mathbb{R}$

- Dynamics: $P(s_{t+1}|s_t, a_t)$

MDP формализм



Markov Decision Process

- Environment states: $s \in S$
- Agent actions: $a \in A$
- Reward: $r \in \mathbb{R}$

Markov assumption

- Dynamics: $P(s_{t+1}|s_t, a_t) = P(s_{t+1}|s_t, a_t, s_{t-1}, a_{t-1})$

Model

Два основных подхода к RL

Model free

Действия оптимизируются напрямую по награде

Оптимальные решения
Маленькие нейросети
Нужно много примеров
Локальные минимумы
Необобщённость

Model based

Пытаемся спрогнозировать последующие состояния среды чтобы выбрать оптимальное действие

Мало примеров
Универсальность
Очень тяжёлая
Какие действия на вход?

POMDP

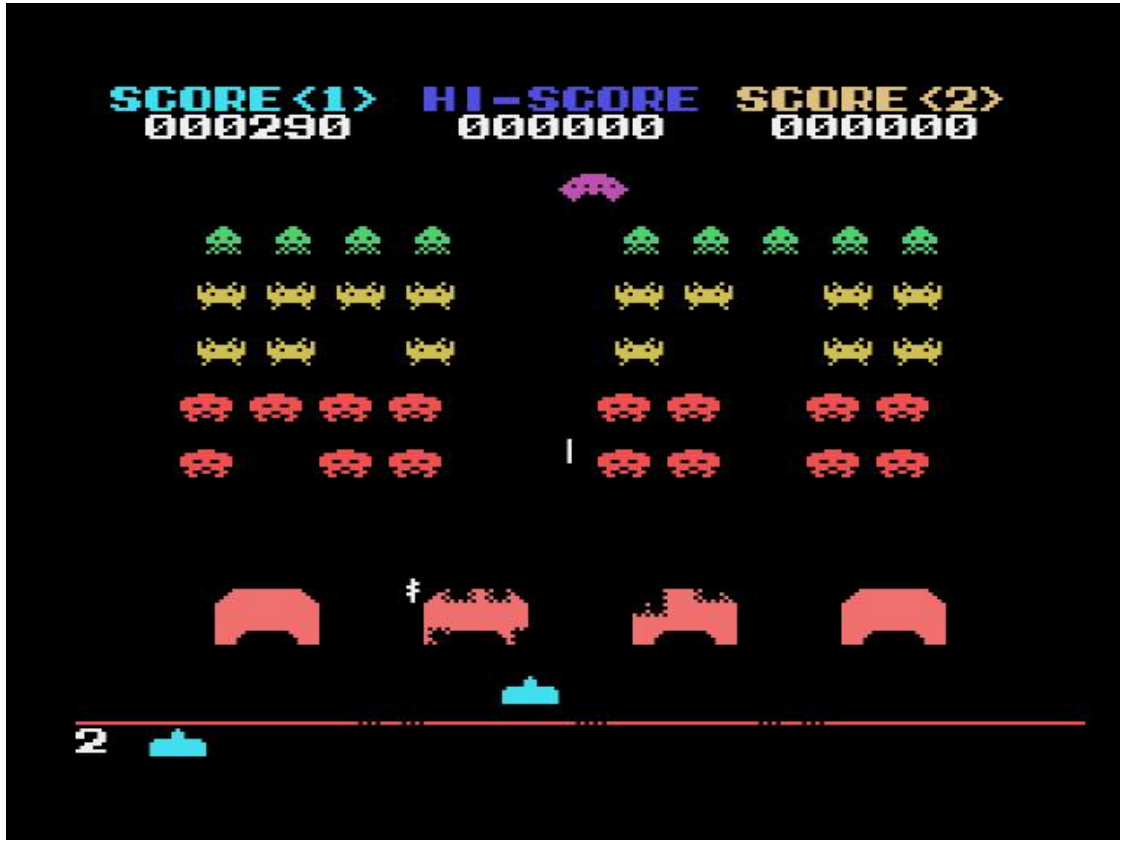
Но есть проблемы…

- Неточность самоощущения
- Неполнота видения среды
- Нестационарность среды

Проблемы MDP

Но есть проблемы…

- Неточность самоощущения
- Неполнота видения среды
- Нестационарность среды



Обобщение MDP

Решение – partially observable MDP!

Агент имеет модель датчика – распределение вероятностей полученных наблюдений при условии сделанного действия.

Total reward

Total reward for session:

$$R = \sum_t r_t$$

Agent's policy:

$$\pi(a|s) = P(\text{совершить действие } a|\text{состояние } s)$$

Хотим максимизировать матожидание R по всем возможным π.

Общий алгоритм

Сыграть несколько сессий

Обновить policy

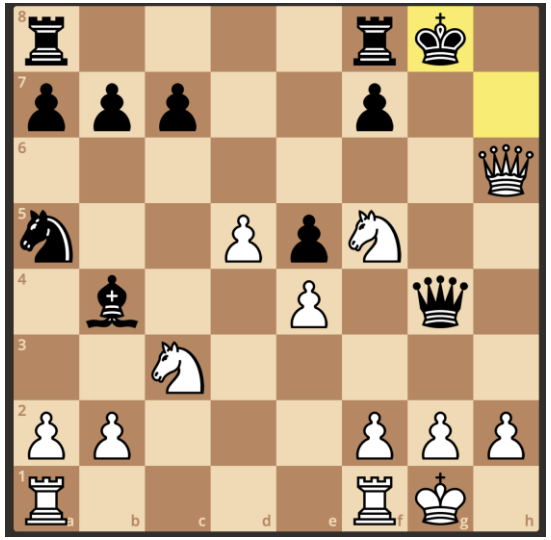
Повторить

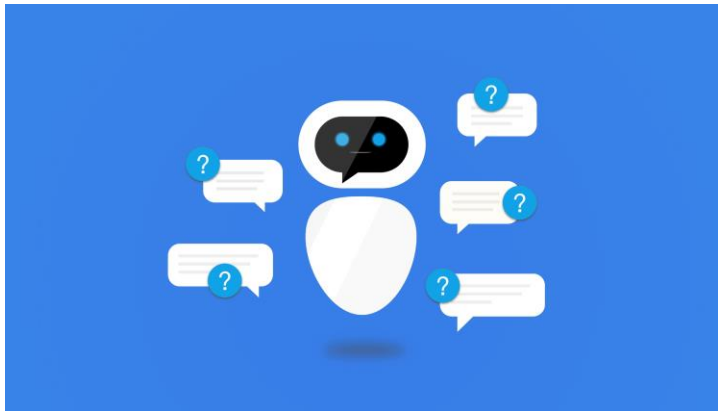
Общий алгоритм

Повторить:

- Отыграть N сессий
- Выбрать M лучших сессий, назовём их элитными
- Изменить policy в зависимости от распределения действий в элитных сессиях

А куда применять?





Спасибо за внимание!