Лекция 5

Обучение с подкреплением: награды

Дополнительные главы машинного обучения

Андрей Фильченков

09.04.2021

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

- В презентации используются материалы курсов
  «Машинное обучение с подкреплением» А.И. Панова
  CS234: Reinforcement Learning, E. Brusnkill
- Слайды доступны: **shorturl.at/wGV59**
- Видео доступны: **shorturl.at/ovBTZ**

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Значение награды

В любой постановке обучения с подкреплением оптимизируется функционал от награды.

Агент всегда в конечном итоге максимизирует награду, хотя в постановке могут быть и другие аргументы, не зависящие от награды.

Зачем тогда другие аргументы?

Значение награды

В любой постановке обучения с подкреплением оптимизируется функционал от награды.

Агент всегда в конечном итоге максимизирует награду, хотя в постановке могут быть и другие аргументы, не зависящие от награды.

Другие аргументы помогают учитывать наши априорные предположения о том, как вообще устроена функция награды и как ей лучше обучаться.

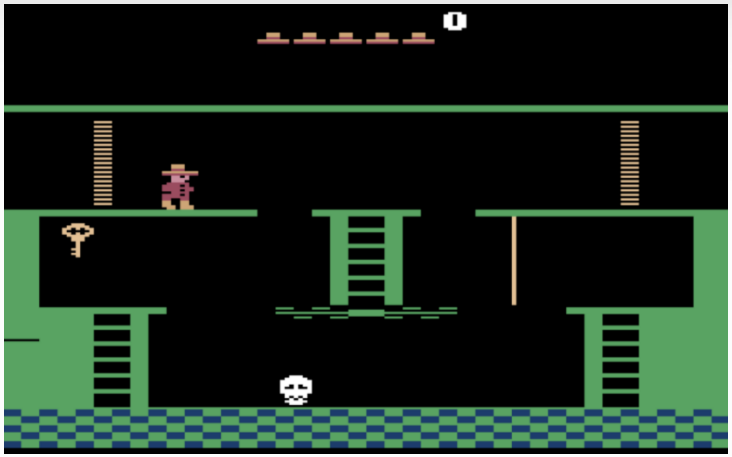
Напоминание: исследование vs использование

Чтобы максимизировать награду, нужно узнавать, как устроена функция награды

Исследование происходит при помощи стратегий. Плохие стратегии производят плохие траектории, которые уводят обучение в сторону.

Разреженная награда

- Награда приходит слишком редко
- Непонятно, как задавать промежуточную награду



Ограниченность стандартной постановки

Подходы, которые мы обсуждали, неприменимы или неэффективны в случаях, когда:

- высокая стоимость или больше время действий
- разреженная награда
- высокая цена ошибки

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Основная идея

Основная идея: попросим эксперта показать нам оптимальные траектории, которые будем учиться воспроизводить.

Фактически, мы переходим к обучению с учителем.

Пример с AlphaGo

Сначала AlphaGo обучалась на партиях мастеров, и только потом играла сама с собой

Клонирование поведение

Пусть $\mathcal{T}_{\text{expert}}$ — траектории, собранные по поведению эксперта, награда неизвестна.

Задача **клонирования поведения (behavior cloning)** состоит в обучении стратегии, воспроизводящей поведение эксперта:

$$\sum_{(s,a)\in\mathcal{T}_{\text{expert}}} \log \pi_\theta(a|s) \to \max_\theta$$

Накопление ошибки

При переходе к обучению с учителем мы начинаем игнорировать темпоральные зависимости.

Чем это грозит?

Накопление ошибки

При переходе к обучению с учителем мы начинаем игнорировать темпоральные зависимости.

Чем это грозит?

Ошибки начинают накапливаться, ошибки в начале приводят к большим ошибкам в следующие моменты времени.

Анализ

Достоинства

- Простота реализации
- Понятность
- Верифицируемость

Недостатки:

- Зависимость от экспертности эксперта
- Накопление ошибки
- Неясно, что делать, когда мы попадаем в новые состояния, которые не встречались в выборке

Сбор дополнительной разметки

Идея: будем просить эксперта доразметить те состояния, которые порождает получаемая стратегия

Это реализуется алгоритмом **DAGGER**:

последовательно дополняем обучающую выборку новыми состояниями, которые посетил агент, и действий, которые в этих состояниях совершит эксперт.

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Основная идея

Наблюдение: состояния различаются по тому, насколько критично точно угадать действие эксперта, однако в исходной постановке для нас это неотличимо.

Основная идея: будем восстанавливать функцию награды, которой «руководствовался» эксперт.

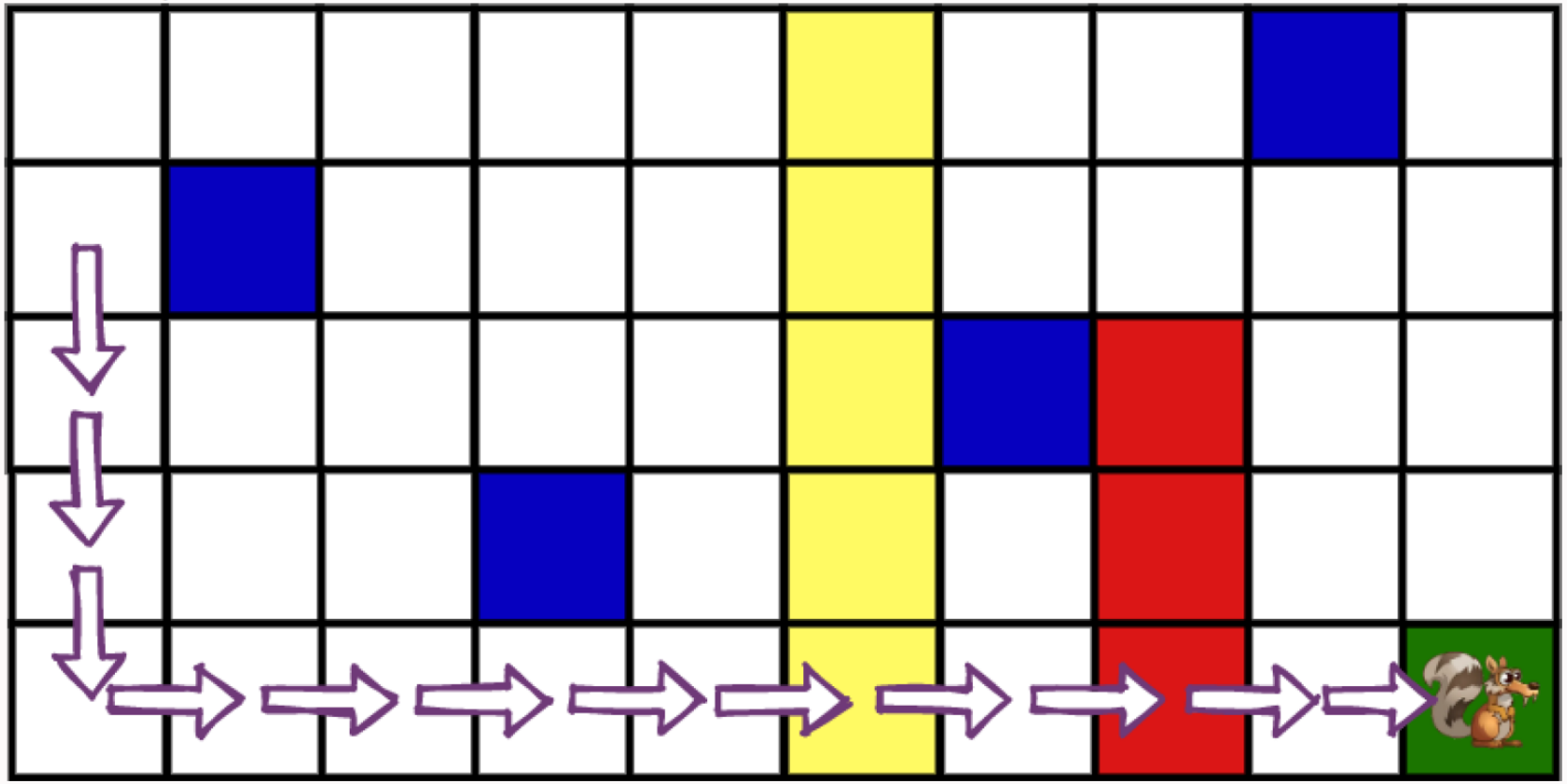
Что здесь вообще происходит?

Такая постановка задачи некорректна, поскольку имеет бесконечное число решений, причем некоторые решения тривиальны, например, $r(s, a) \equiv 0$.

Если эксперт не был оптимальным, то как нам поможет такая функция наград?

Успокаивающий пример

Пусть награда в клетках одного цвета одинакова



Оптимизация параметрической награды

Введем $r_\rho(s,a)$, $\rho \in \mathrm{P}$,

$$V_\rho^\pi = \mathrm{E}_\pi \sum_t \gamma^t R_\rho(s_t),$$

будем искать такую $r_{\rho^*}(s,a)$, что для экспертной стратегии π^*

$$V_{\rho^*}^{\pi^*} \geq V_\rho^{\pi^*}$$

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Maximum entropy RL

Основная идея: будем искать не только хорошие стратегии, но и нестационарные стратегии. Вместо

$$\mathrm{E}_{\pi_\theta} Q^{\pi_\theta}(s,a) \to \max_\theta$$

будет оптимизировать

$$J_{soft} = \mathrm{E}_{\pi_\theta} \sum_t \gamma^t [r_t + \alpha \mathcal{H}(\pi(\cdot\,|s))],$$

где $\mathcal{H}\big(\pi(a)\big) = -\mathrm{E}_{\pi_\theta} \log \pi_\theta(a)$ — энтропия.

Простой переход

$$J_{soft} = \mathrm{E}_{\pi_\theta} \sum_t \gamma^t [r_t + \alpha \mathcal{H}(\pi(\cdot\,|s))] \to \max_\theta$$

эквивалентна

$$J_{soft} = \mathrm{E}_{\pi_\theta} \sum_t \gamma^t [r_t - \alpha \log \pi_\theta(a_t|s_t)] \to \max_\theta$$

то есть мы модифицировали награду:

$$r_{soft} = r(s,a) - \alpha \log \pi_\theta(a|s)$$

Важная теорема

Теорема. Пусть π — стратегия с мягкой оценочной функцией $Q^{\pi}_{soft}(s,a)$. Тогда стратегия π^*, распределенная согласно $\exp Q^{\pi}_{soft}(s,a)$, не хуже π.

Следствие: оптимальные стратегии будут иметь вид $\exp Q_{soft}(s,a)$ или $\exp R(s)$ для заданной функции наград.

Soft actor-critic

Модифицированную награду можно использовать для обучения AC алгоритма, который тогда будет называться **Мягкий актор-критик (soft actor-critic)**. Для этого максимизируется не Q по действиям, а

$$\mathrm{KL}(\pi_\theta(s,a)||\exp Q^{\pi}_{soft}(s,a)) \to \min_\theta$$

Maximum Entropy IRL

Пусть оптимальная стратегия порождает траекторию τ с вероятностью, пропорциональной

$$\exp R_\rho(\tau).$$

Тогда **задача maximum IRL:**

$$\mathrm{KL}(p(\tau|\pi_\theta)||p_{expert}(\tau)) \to \min_\theta$$

Она эквивалентна

$$\mathrm{E}_{\pi_\theta} \sum_t \gamma^t [r_\rho(a_t, s_t) - \alpha \log \pi_\theta(a_t|s_t)] \to \max_\theta$$

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Переход к логарифму

$$p_\rho(\tau|\pi_\theta) = \frac{1}{Z_\rho}\exp R_\rho(\tau)$$

$$\log p_\rho(\tau|\pi_\theta) = R_\rho(\tau) - \log Z_\rho$$

Максимизация логарифма означает, что мы предполагаем давать высокую награду тем траекториям, которые агент посетил, и маленькую — всем остальным.

Оптимизация градиента

Для того, чтобы аппроксимировать $R_\rho(\tau)$, мы будем использовать нейронную сеть.

Теорема: градиент максимизации правдоподобия равен

$$\mathrm{E}_{\tau\sim\pi^*}\nabla_\rho R_\rho(\tau) - \mathrm{E}_{\tau\sim\pi^*_{[\rho]}}\nabla_\rho R_\rho(\tau),$$

где $\pi^*_{[\rho]}$ отпимизирует текущую функцию награды r_ρ

Интерпретация

Первое слагаемое оптимизирует награду у состояний, встретившихся у эксперта

Далее строим оптимальную стратегию $\pi^*_{[\rho]}$, собираем траектории.

Второе слагаемое минимизирует награду, которая встретилась в состояниях этой стратегии.

Когда награда станет оптимальной, оптимальная стратегия сойдется к эксперту, градиент станет нулевым.

Основная идея

Однако оптимизация на каждом шаге здесь затратна.

Основная идея: будем пытаться найти седловую точку в пространстве награды x стратегии

Мера занятия

Мера занятия (occupancy measure):

$$\omega_\pi(s,a) = \pi(a|s)d_\pi(s),$$

где $d_\pi(s)$ — частоты посещения состояний.

$$\pi(a|s) = \frac{\omega_\pi(s,a)}{\int_A \omega_\pi(s,a)\,\mathrm{d}a}$$

Значит, можно вместо стратегий искать меры занятия.

Title

После переходов, замен и сокращений перепишем оптимизируемый функционал в виде

$$\max_{\rho} \min_{\omega_\pi} \int_S \int_A \left(\omega_{\pi^*}(s,a) - \omega_\pi(s,a)\right) r_\rho(s,a) \mathrm{d}s\mathrm{d}a - \mathcal{H}(\omega_\pi)$$

GAIL

Generative adversarial imitation learning

Будем обучать дискриминатор D, который для пар (s, a) из эксперта должен вернуть 1, а для остальных – 0.

Оптимизация по D при заданном π:

$$\mathrm{E}_{\omega_{\pi^*}} \log\big(1 - D(s,a)\big) + \mathrm{E}_{\omega_{\pi}} \log D(s,a) \to \max_D$$

Оптимизация по π при фиксированном D

$$E_{\tau\sim\pi} \sum_t [-\log D(s_t, a_t) + \mathcal{H}(\pi(\cdot\,|s_t))] \to \max_\pi$$

Интерпретация

Задача обучения по экспертным траекториям свелась к поиску $\omega_\pi(s,a)$, наиболее похожего на $\omega_{\pi^*}(s,a)$, что тривиально – мы пытаемся ходить как эксперт.

Но для этого мы в итоге используем генеративно-состязательный подход

План лекции

- Проблемы с наградами
- Клонирование поведения
- Обратное обучение с подкреплением
- Максимизация энтропии
- GAIL
- Внутренняя мотивация

Награда за поиск иголки

$$r(s,a) = \begin{cases} 1, s \text{ термально} \\ -\varepsilon, \text{если нет.} \end{cases}$$

Основная идея

Основная идея: придумаем себе вспомогательную задачу, которая позволит хоть чему-то, но научиться.

- Похоже на самообучение (self-supervised learning)
- Похоже на то, как устроено школьное (и не только) образование

Функция награды как сумма

$$r(s,a) = r^{extr}(s,a) + \alpha r^{intr}(s,a)$$

Так же раскладываются все ценностные функции.

В policy- алгоритмах можно учитывать эти слагаемые отдельно.

Благодаря $r^{intr}(s,a)$ агент может прерывать эпизоды.

Исследовательские бонусы

Исследовательские бонусы (exploration bonuses) можно давать в чистом виде за исследование среды

- частотные / плотностные бонусы
- бонусы предсказуемости (новизны)

Напоминание

$$A_t = \arg\max_{a \in A} \left(Q_t(a) + \sqrt{\frac{2 \ln t}{k_t(a)}} \right)$$

RND

Пусть ϕ это случайная сеть, которая строит эмбеддинги состояний, а f обучается предсказывать значения ϕ.

$$r^{intr} = \|f(s) - \phi(s)\|_2^2$$

Любопытство

Пусть $f(s, a)$ предсказывает следующее состояние

Любопытство (curiosity):

$$r^{intr} = \|f(s, a) - s'\|_2^2$$

Проблема шумного телевизора

Проблема шумного телевизора (noisy TV): не все непредсказуемые ситуации интересны.

(Не всякая неопределенность эпистимическая)

Модель обратной динамики

Вместо предсказания состояний, будем предсказывать действия:

$$r^{intr} = \|f(s, s') - a\|_2^2$$

Что еще можно делать для борьбы с шумным телевизором

- ~~Смотреть YouTube~~
- Использовать фильтрацию состояний
- Использовать память