ТюмГУ

Обучение с подкреплением.

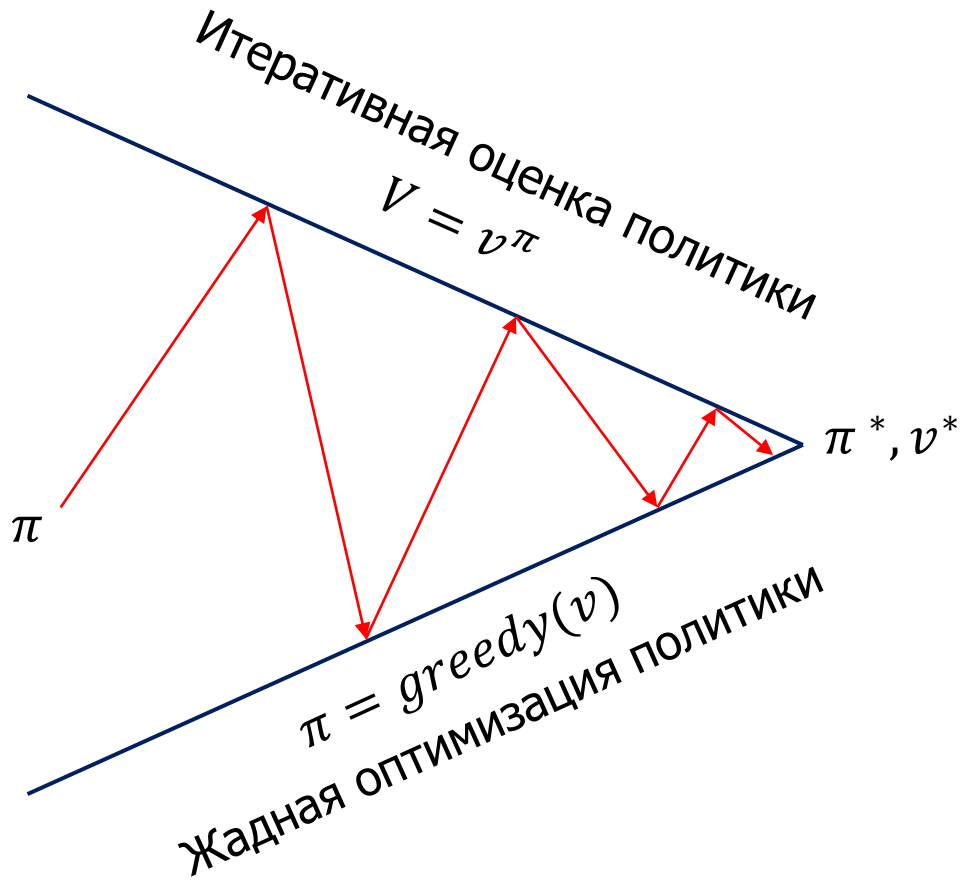
Оптимизация поведения агентов.

Сергей Аксёнов,
Доцент отделения информационных технологий
Инженерной школы информационных технологий и робототехники
Томский политехнический университет

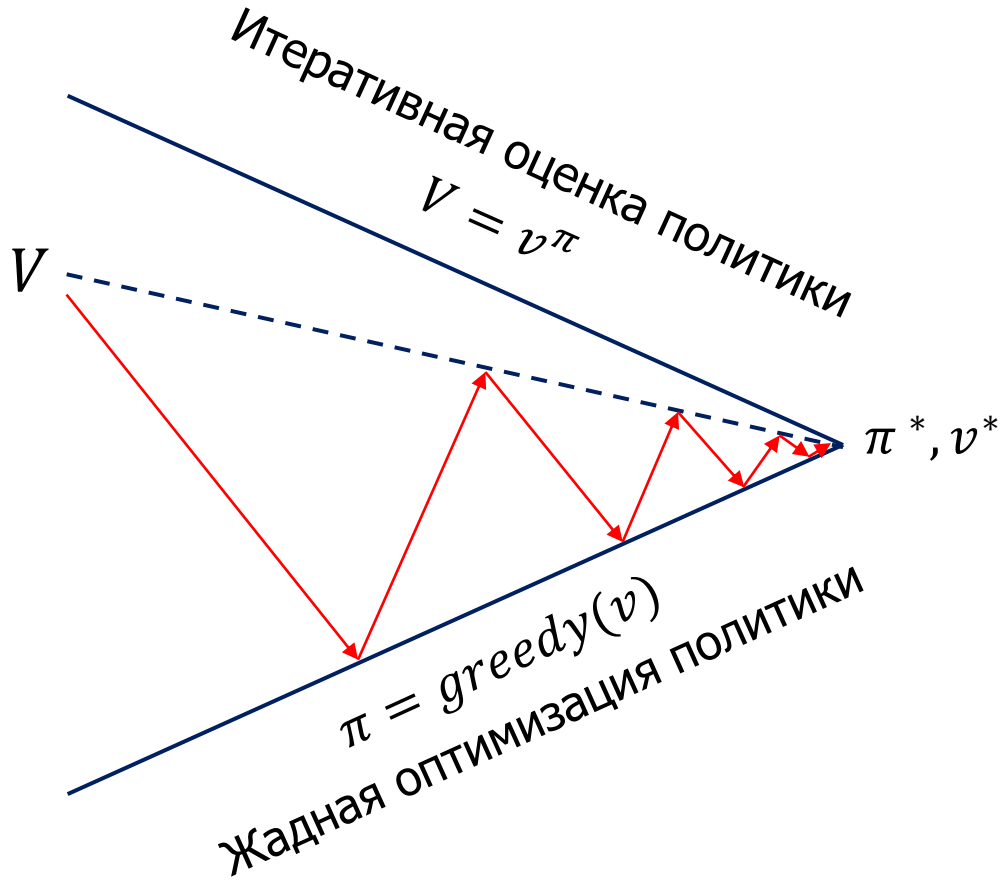
Тюмень-2024

Итерация политик и итерация ценности

Итерация политик



Итерация ценности



Метод управления Монте-Карло

Генерация траектории на основе политики

$$S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}, \dots, R_T, S_T \sim \pi_{t:T}$$

Расчет выгоды

$$G_{t:T} = R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-1} R_T$$

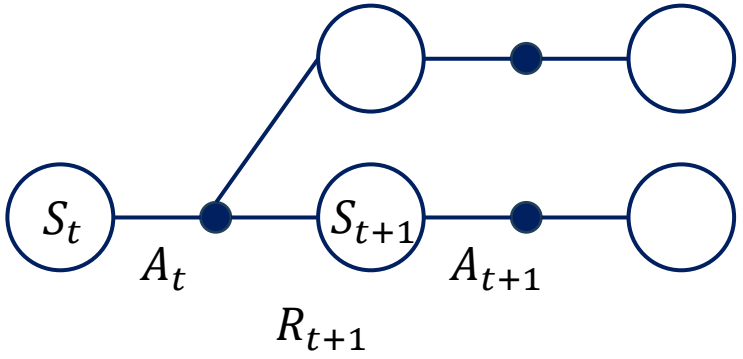
Обновление Q-функции

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha_t [G_{t:T} - Q(S_t, A_t)]$$

Агент SARSA

SARSA – State-Action-Reward-State-Action

Использование TD вместо MC



Обучение на основе политики (on-policy) – обучение на своих ошибках

Обновление Q-функции агентом SARSA

Погрешность SARSA

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha_t[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$$

Цель SARSA

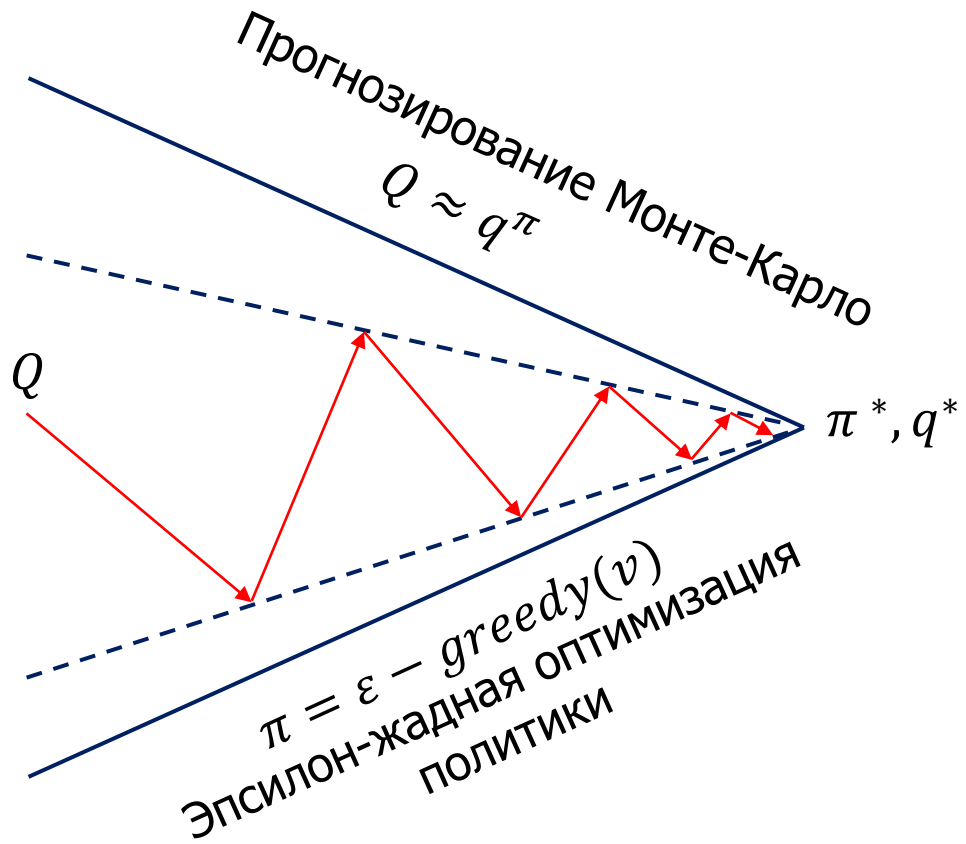
Жадность в пределе с бесконечным исследованием

Требования, которым должен следовать алгоритм on-policy RL для гарантии сходимости к оптимальной политике:

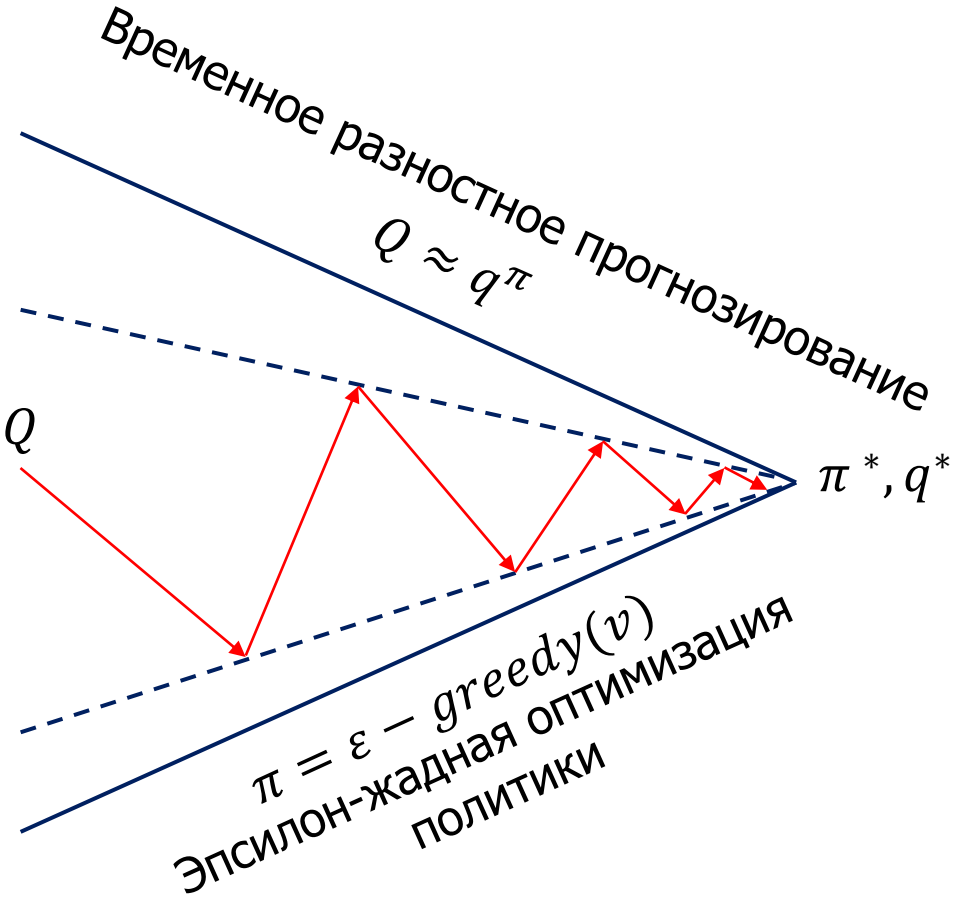
- Все пары «состояние-действие» должны исследоваться бесконечно часто
- При схождении политика должна становиться жадной

Управление Монте-Карло и SARSA

Управление Монте-Карло

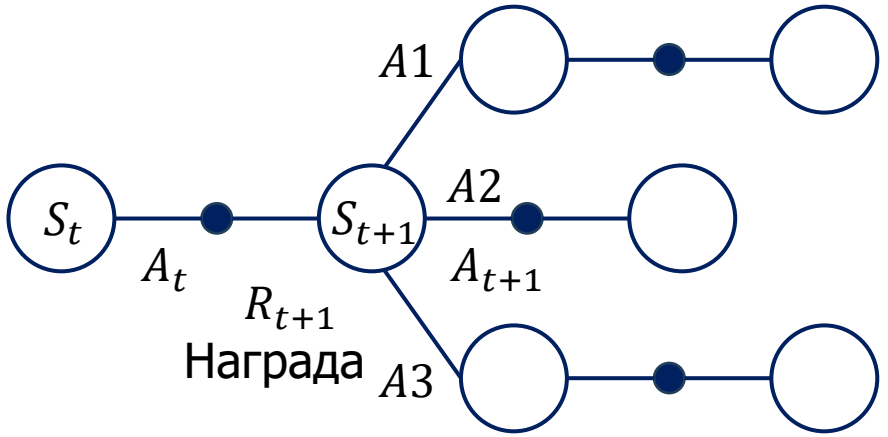


SARSA



Q-обучение

Для оценки цели используется действие с максимальной ожидаемой ценностью в следующем состоянии, независимо от того, какое действие было выбрано



$$a = \{A1, A2, A3\}$$

Обучение вне политики (off-policy) – обучение на чужих ошибках

Уравнение Q-обучения

Погрешность Q-обучения

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha_t \left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \right]$$

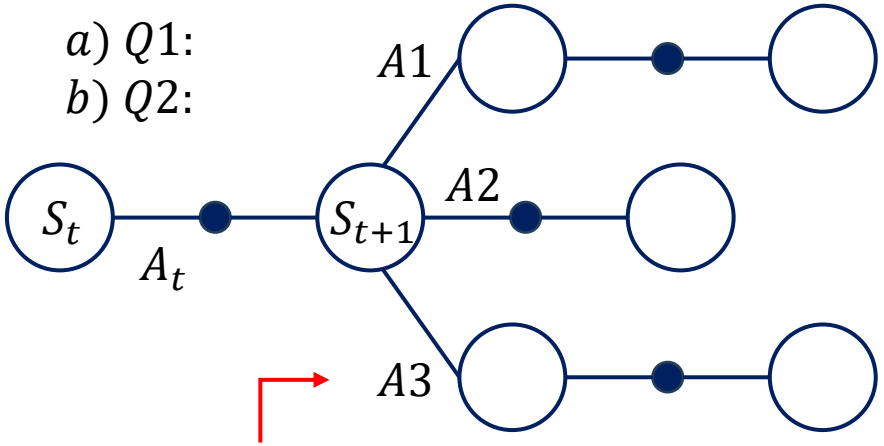
Цель Q-обучения

Двойное Q-обучение

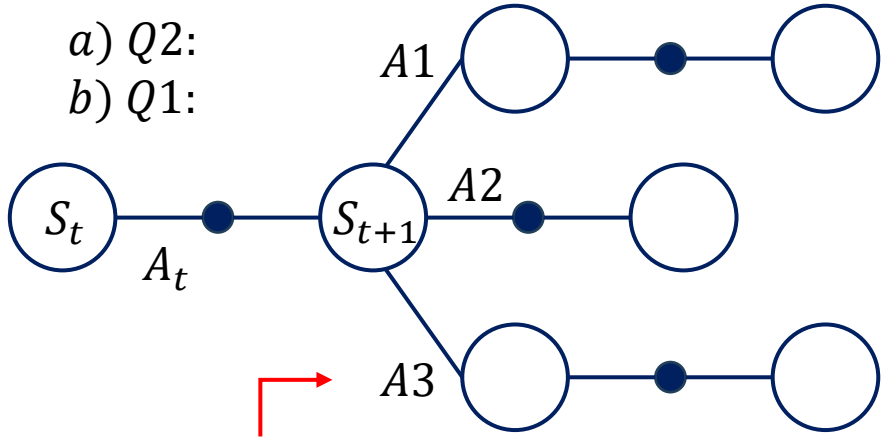
Проблема Q-обучения: Смещение максимизации (выбор максимальной функции ценности действий, ожидаемую в следующем состоянии дает слишком оптимистичный прогноз Q)

Решение: Использование 2-х функций ценности Q1 и Q2

После получения кортежа опыта случайно выбираем из Q1 и Q2, функцию, которая находит лучшее действие (максимальная ценность), но вычисление цели TD происходит на основе ценности из другой функции для выбранного действия



Действие с макс. ценностью



Оптимизация Q

Оптимальная политика: $\pi^* = argmax((Q1 + Q2)/2)$

SARSA(λ)

Вместо цели на основе одношагового бутстреппинга (цели TD) используется λ – выгода.

Вместо следов приемлемости для учета посещённых состояний анализ посещенных пар «состояние - действие».

Механизм замещающих признаков – использование для значений приемлемости максимального значения 1 (при посещении пары «состояние-действие»). Затухание значений в соответствии с величиной λ.

SARSA(λ)

Обнуление вектора приемлемости 0

$$E_0 = 0$$

Взаимодействие со средой

$$S_t, A_t, R_{t+1}, S_{t+1} \sim \pi_{t:t+1}$$

При посещении состояния, увеличиваем его приемлемость с учетом максимального значения

$$E_t(S_t, A_t) = E_t(S_t, A_t) + 1$$

$$E_t(S_t, A_t) = clip\big(1, E_t(S_t, A_t)\big)$$

Вычисление TD-погрешности

$$\delta^{TD}_{t:t+1}(S_t) = R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)$$

Обновление Q - функции

$$Q = Q + \alpha_t \delta^{TD}_{t:t+1}(S_t) E_t$$

Уменьшение приемлемости

$$E_{t+1} = E_t \gamma \lambda$$

Q(λ) Уоткинса

Обнуление вектора приемлемости 0

$$E_0 = 0$$

Взаимодействие со средой

$$S_t, A_t, R_{t+1}, S_{t+1} \sim \pi_{t:t+1}$$

Проверка, имеет ли выбранное действие максимальную ценность

$$is\_greedy = \begin{cases} 1, Q(S_t, A_t) = \max\limits_a Q(S_t, a) \\ 0, Q(S_t, A_t) \neq \max\limits_a Q(S_t, a) \end{cases}$$

Вычисление TD-погрешности

$$\delta^{TD}_{t:t+1}(S_t) = R_{t+1} + \gamma \max_a Q(S_t, a) - Q(S_t, A_t)$$

Обновление Q - функции

$$Q = Q + \alpha_t \delta^{TD}_{t:t+1}(S_t) E_t$$

Управление приемлемостью

$$E_{t+1} = \begin{cases} E_t \gamma \lambda, is\_greedy = 1 \\ 0, is\_greedy = 0 \end{cases}$$

Dyna-Q

Пример модельно-ориентированного обучения с подкреплением

Взаимодействие со средой по примеру безмодельных методов, формирующее по результатам взаимодействий модель среды (MDP)

Dyna-Q: Q-обучение + Итерация по значениям

Оценка функции перехода и вознаграждения:

- Тензор перехода: расчет количества кортежей опыта (S_t, A_t, S_{t+1}) в эпизодах
- Тензор вознаграждения: расчет средней награды в кортежах (S_t, A_t, S_{t+1})