RL: Базовые алгоритмы

План

- ❑ Монте Карло
- ❑ Временные различия
- ❑ Обобщенный подход
- ❑ Безмодельное управление
- ❑ Обучение по чужому опыту
- ❑ Q обучение

Монте - Карло

- Работают напрямую по эпизодам взаимодействия со средой
- Безмодельный подход (model-free)
- Обучается по полным эпизодам
- Полезность равно средней отдаче
- Может применен только для эпизодических MDP

Монте – Карло оценка стратегии

- Цель: построить V^{π} по эпизодам взаимодействия со стратегией π:

$$s_1, a_1, r_1, \dots, s_k \sim \pi$$

- Отдача – суммарное вознаграждение:

$$R_t = r_{t+1} + \gamma r_{t+2} + \cdots \gamma^{T-1} r_T$$

- Функция полезности это мат ожидание отдачи:

$$V^{\pi}(s) = E_{\pi}[R_t | s_t = s]$$

- Оценка стратегии использует эмпирическое среднее отдачи вместо мат.ож

Монте – Карло оценка стратегии с первым посещением

- Оцениваем состояние s:
- Для первого по времени посещения состояния s в эпизоде:

$$N(s) \leftarrow N(s) + 1$$

$$S(s) \leftarrow S(s) + R_t$$

- Полезность оцениваем как среднюю отдачу: $V(s) = S(s)/N(s)$

$$V(s) \xrightarrow{N(s)\to\infty} V^{\pi}(s)$$

Несмещенная состоятельная оценка с высокой дисперсией

- Оценка стратегии использует эмпирическое среднее отдачи вместо мат.ож

Монте – Карло оценка стратегии с каждым посещением

- Оцениваем состояние s:
- Для первого по времени посещения состояния s в эпизоде:

$$N(s) \leftarrow N(s) + 1$$

$$S(s) \leftarrow S(s) + R_t$$

- Полезность оцениваем как среднюю отдачу: $V(s) = S(s)/N(s)$

$$V(s) \xrightarrow{N(s)\to\infty} V^{\pi}(s)$$

Смещенная состоятельная оценка с низкой дисперсией

Среднее приращение

- Средние значения могут быть вычислены последовательно:

$$\mu_k = \frac{1}{k}\sum_{i=1}^{k} x_i = \frac{1}{k}\left(x_k + \sum_{i=1}^{k-1} x_i\right) =$$

$$= \frac{1}{k}\left(x_k + (k-1)\mu_{k-1}\right) \Rightarrow$$

$$\Rightarrow \mu_k = \mu_{k-1} + \frac{1}{k}\left(x_k - \mu_{k-1}\right)$$

Монте Карло для приращений

- Обновим V(s) с приращением для эпизода $s_1, a_1, r_1, \dots, s_T$:
- Для каждого состояния s_t с отдачей R_t:

$$N(s_t) \leftarrow N(s_t) + 1$$

$$V(s_t) \leftarrow V(s_t) + \frac{1}{N(s_t)}(R_t - V(s_t))$$

- Для нестационарных задач м.б. полезно отслеживать текущее среднее:

$$V(s_t) \leftarrow V(s_t) + \alpha(R_t - V(s_t))$$

Обучение на основе временных различий

- Метод временных различий (temporal-difference, TD) обучается напрямую по эпизодам взаимодействия со средой
- TD – безмодельный подход (model-free): модель переходов MDP и функция вознаграждения не известны
- TD обучается по неполным эпизодам, используя бутстреп
- TD приближает значения на основе предыдущего приближения

MC и TD

- Цель: построить V^{π} интерактивно (online) по эпизодам взаимодействия со стратегией π
- MC с каждым посещением для приращений: обновляем $V(s_t)$ на основе текущей отдачи R_t

$$V(s_t) \leftarrow V(s_t) + \alpha(R_t - V(s_t))$$

- Подход TD:
  - Обновляем на основе ожидаемой отдачи $r_t + \gamma V(s_{t+1})$

$$V(s_t) \leftarrow V(s_t) + \alpha(r_t + \gamma V(s_{t+1}) - V(s_t))$$

  - $r_t + \gamma V(s_{t+1})$ называется TD показателем
  - $\delta_t = r_t + \gamma V(s_{t+1}) - V(s\_t)$ называется TD ошибкой

- TD приближает значения на основе предыдущего приближения

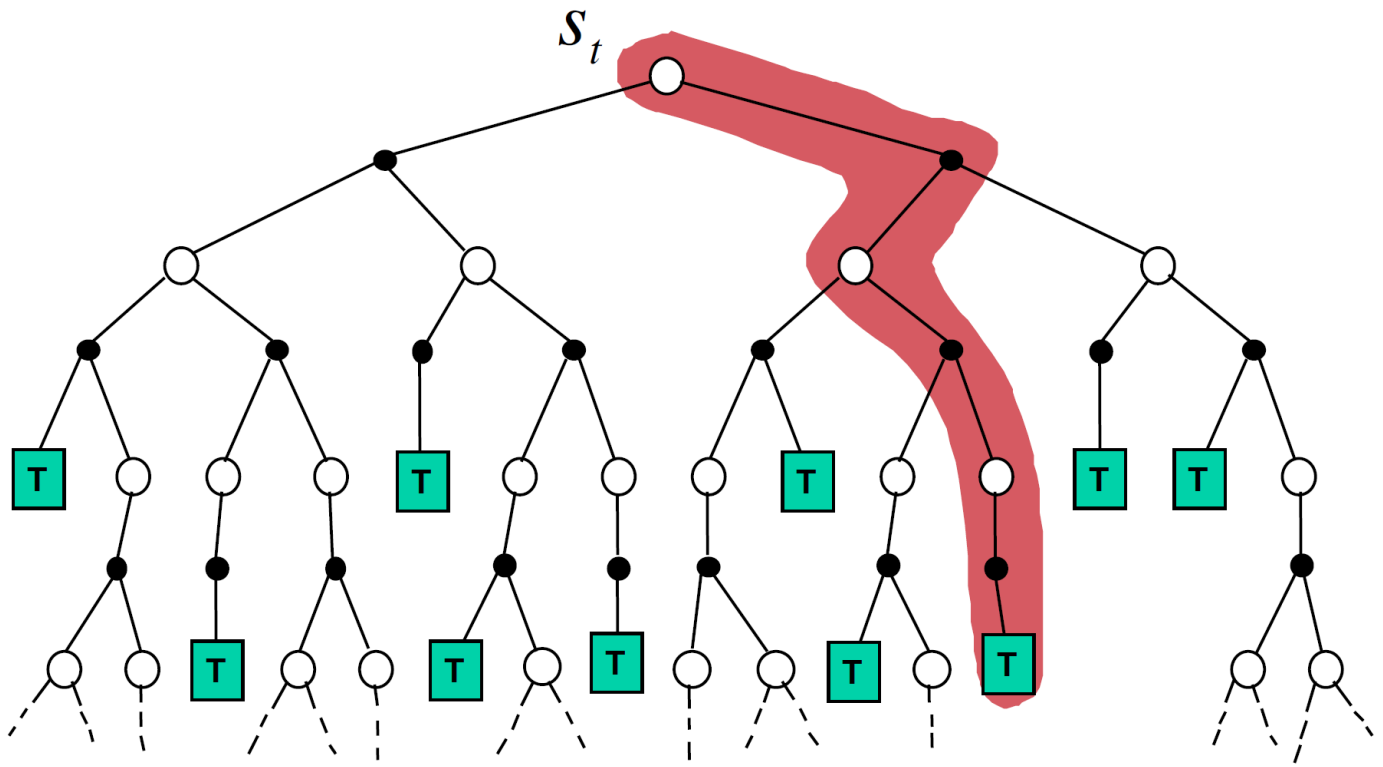
Преимущества и недостатки MC и TD

- TD может обучаться до того, как стала известна итоговая отдача:
  - TD может обучаться интерактивно на каждом шаге
  - MC должен дожидаться окончания эпизода, когда становиться известна отдача
- TD может обучаться без информации об итоговой отдаче:
  - TD может обучаться на неполных эпизодах
  - MC может обучаться только на полных эпизодах
  - TD работает и для бесконечных (без терминального состояния) окружений
  - MC работает только для эпизодических окружений

Пакетные методы MC и TD

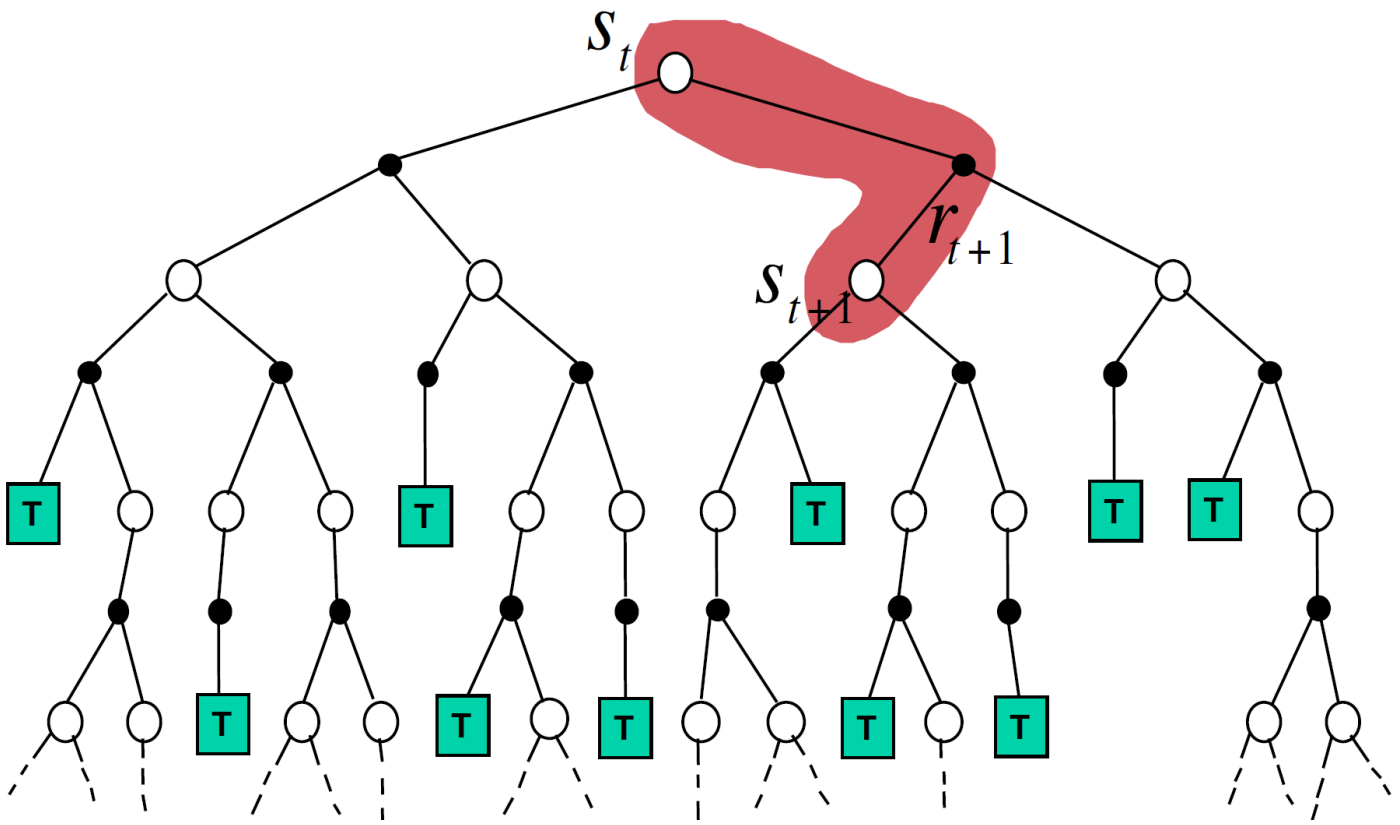
MC обновление

$$V(S_t) \leftarrow V(S_t) + \alpha\,(G_t - V(S_t))$$



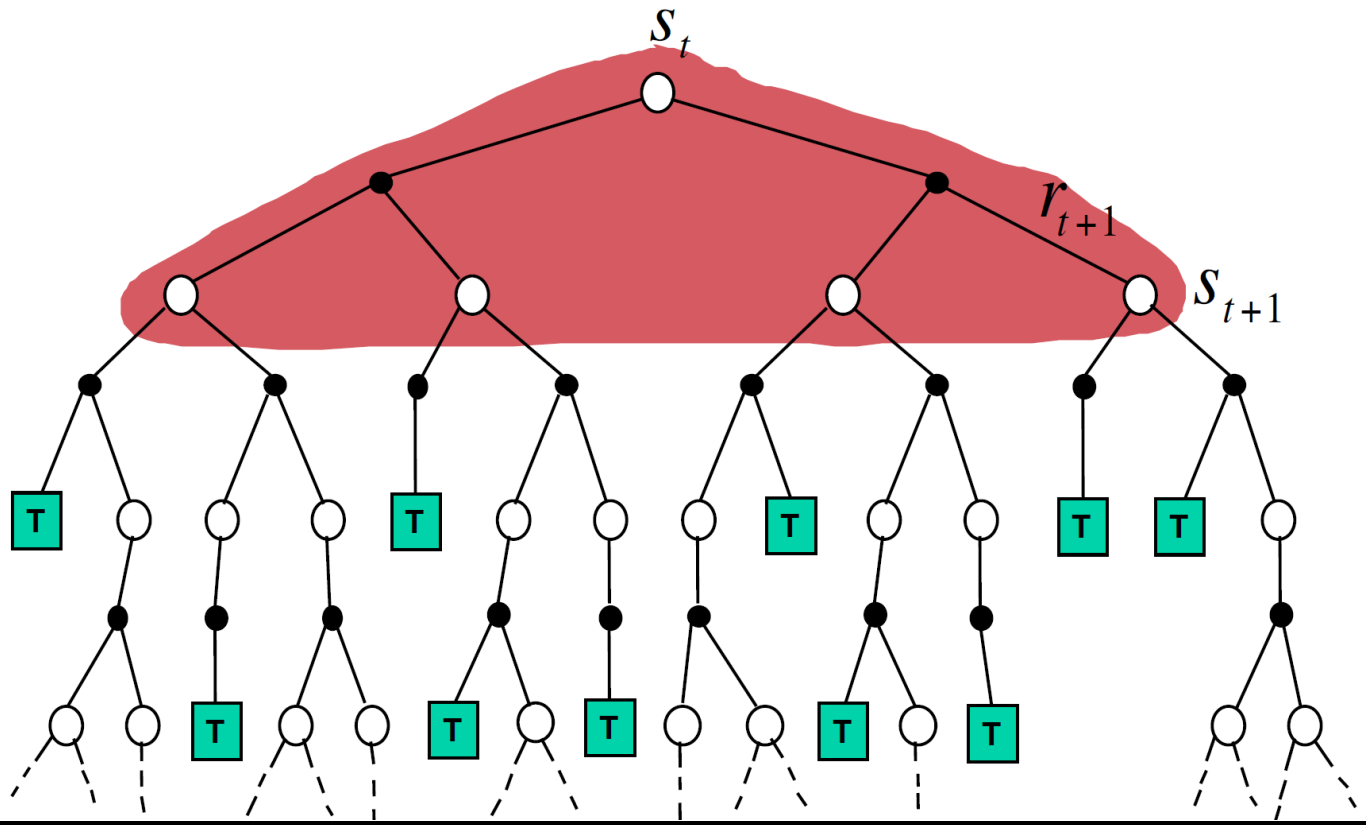
TD обновление

$$V(S_t) \leftarrow V(S_t) + \alpha \left(R_{t+1} + \gamma V(S_{t+1}) - V(S_t) \right)$$



Динамическое программирование

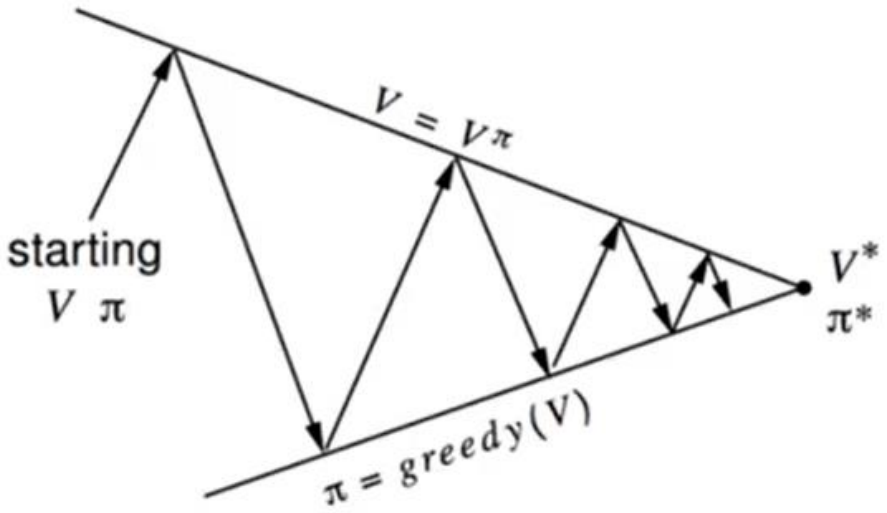
$$V(S_t) \leftarrow \mathbb{E}_\pi \left[R_{t+1} + \gamma V(S_{t+1}) \right]$$



Обучение на собственном и чужом опыте

- Обучение по собственному опыту (on-policy):
  - Обучение на ходу
  - Обучение стратегии π по опыту, полученному на основе π
- Обучение по чужому опыту (off-policy):
  - Обучение на чужих ошибках
  - Обучение стратегии π по опыту, полученному на основе μ

Обобщенные итерация по стратегиям

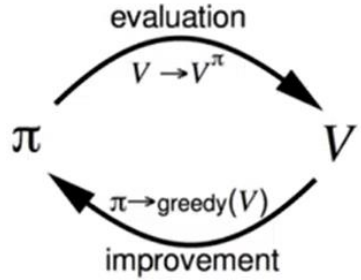


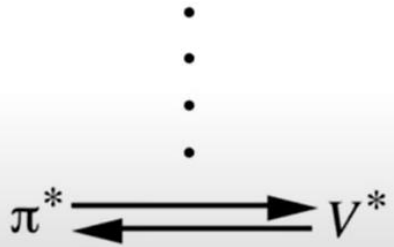
Оценка стратегии - вычисление V^{π}

Итеративная оценка стратегии

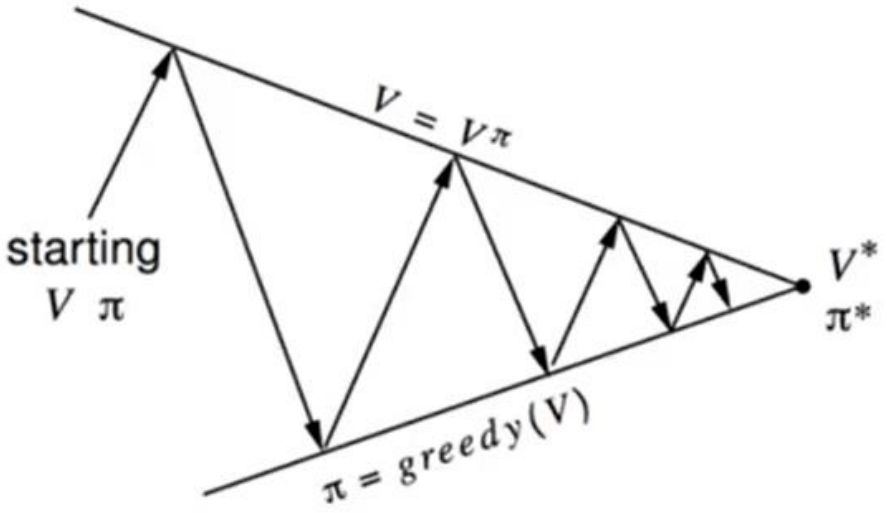
Улучшение стратегии – генерация $\pi' \geq \pi$

Жадное обновление стратегии





Обобщенные итерация по стратегиям с МС оценкой



Оценка стратегии - вычисление V^{π}

Монте Карло оценка стратегии по $V = V^{\pi}$

Улучшение стратегии

Жадное обновление стратегии?

Безмодельная итерация по стратегиям с функцией полезности действия

- Жадное улучшение стратегии по V(s) требует знания модели MDP:

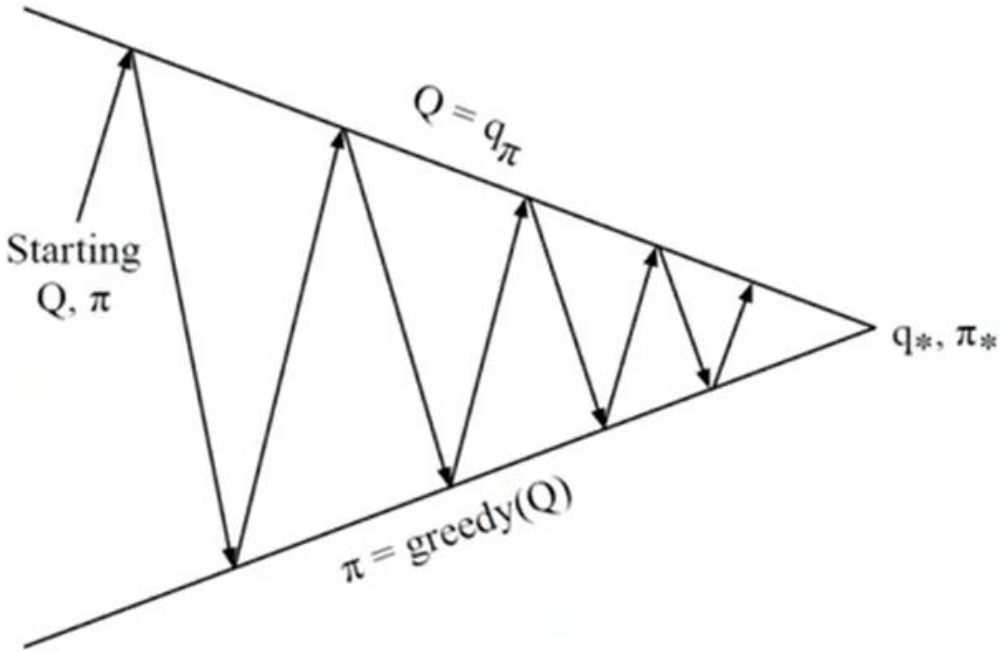
$$\pi'(s) = \underset{a \in A}{\operatorname{argmax}}(R_s^a + P_{ss'}^a V(s'))$$

  - Обучение стратегии π по опыту, полученному на основе π

- Жадное обновление стратегии по $Q(s,a)$ не требует знания модели

$$\pi'(s) = \underset{a \in A}{\operatorname{argmax}}\, Q(s,a)$$

Обобщенные итерация по стратегиям с функцией полезности действия



Оценка стратегии -

Монте Карло оценка стратегии по $\mathrm{Q} = \mathrm{Q}^{\pi}$

Улучшение стратегии

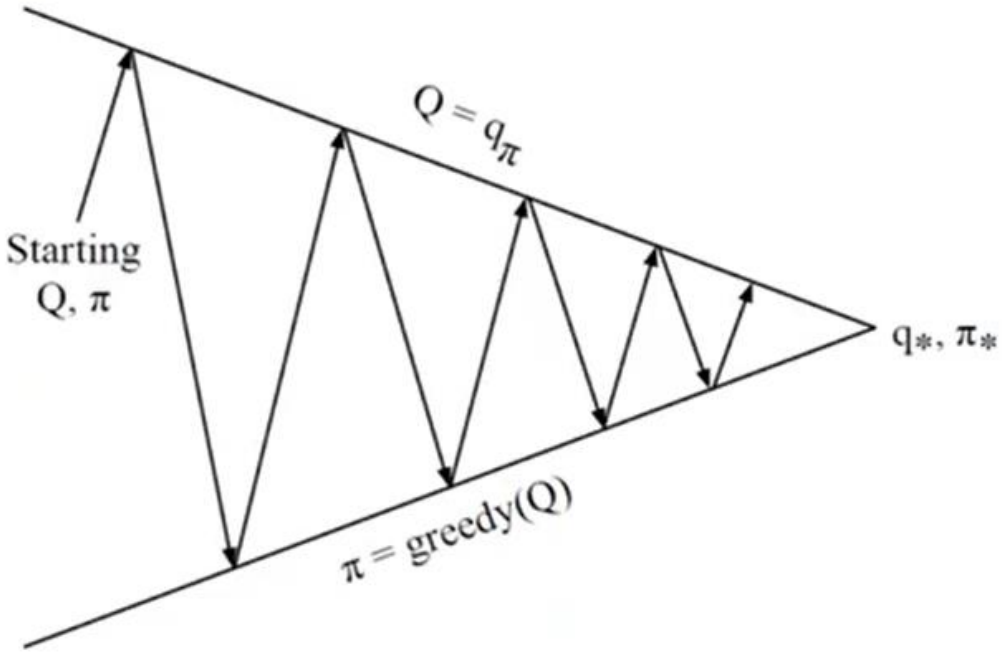
Жадное обновление стратегии?

ε жадное исследование

- Простейшая идея, обеспечивающая постоянное исследование среды
- Все действия выбираются с ненулевой вероятностью
- С вероятностью $\mathbf{1} - \boldsymbol{\varepsilon}$ выбираем действие жадно
- С вероятностью $\boldsymbol{\varepsilon}$ выбираем действие случайно

$$\pi\,(a|s) = \begin{cases} 1 - \dfrac{\varepsilon}{m}, \text{если } a^* = \max\limits_{a \in A} Q(s,a) \\ \dfrac{\varepsilon}{m}, \qquad \text{иначе} \end{cases}$$

MC итерация по стратегиям



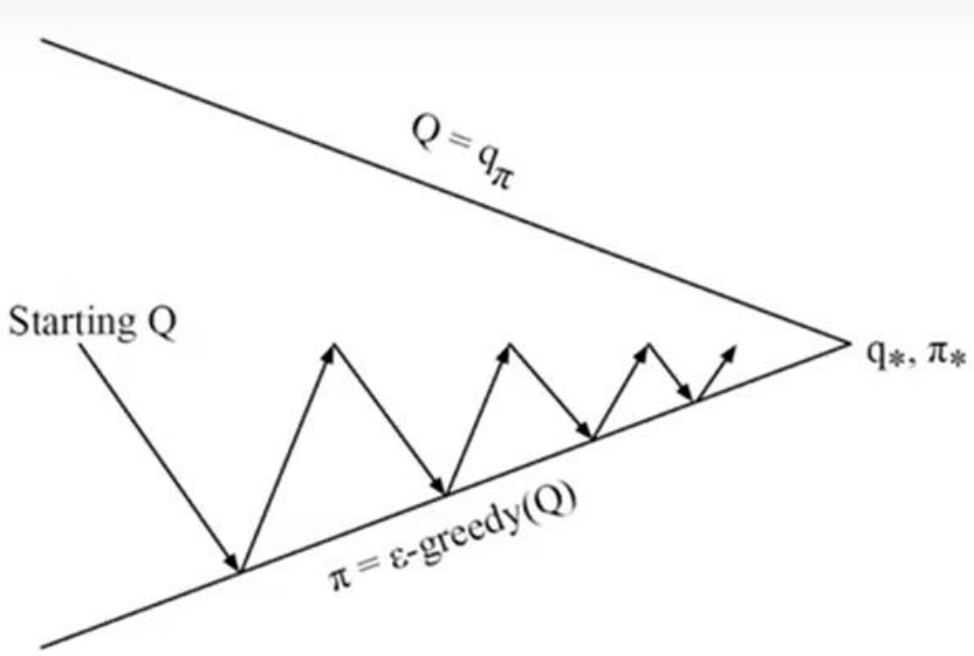
Оценка стратегии -

Монте Карло оценка стратегии по $Q = Q^\pi$

Улучшение стратегии

ε жадное обновление стратегии

MC управление



Для каждого эпизода:

Оценка стратегии -

Монте Карло оценка стратегии по $Q = Q^\pi$

Улучшение стратегии

ε жадное обновление стратегии

MC управление

- Выбираем k-й эпизод $(s_1, a_1, r_2, \dots, s_T) \sim \pi$
- Для каждой пары состояние и действия в эпизоде выполняем:

$$N(s_t, a_t) \leftarrow N(s_t, a_t) + 1$$

$$\mathrm{Q}(s_t, a_t) \leftarrow Q(s_t, a_t) + \frac{1}{N(s_t, a_t)} (R_t - Q(s_t, a_t))$$

- Улучшаем стратегию на основе новых значений полезности

$$\varepsilon \leftarrow 1/k$$

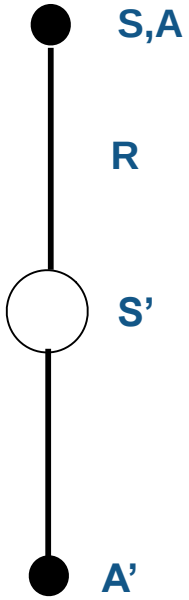
$$\pi \leftarrow \varepsilon \text{ жадная}(Q)$$

- Этот алгоритм сходится к оптимальному решению

MC vs TD управление

- TD имеет несколько преимуществ относительно MC
  - Меньше дисперсия,
  - Интерактивное
  - Неполные последовательности
- Следовательно, мы можем использовать TD вместо MC для управления
  - Применить TD к Q(s,a)
  - Использовать ε жадное улучшение
  - Обновлять на каждом шаге

Обновление функции полезности по SARSA



$$\mathrm{Q}(s,a) \leftarrow Q(s,a) + \alpha(r + \gamma Q(s',a') - Q(s,a))$$

Обучение по чужому опыту

- Строим целевую стратегию $\pi(a|s)$ для вычисления $V^{\pi}(s)$ или $Q^{\pi}(s,a)$ по следующей актуальной стратегии $\mu(a|s)$:

$$(s_t, a_t, r_t, \dots) \sim \mu$$

В чем разница?

- Обучение по наблюдениям за человеком или другими агентами
- Переиспользование опыта полученного по старым стратегиям $\pi_1, \pi_2, \dots, \pi_{t-1}$
- Конструирование оптимальной стратегии, следуя поисковой стратегии
- Конструирование нескольких стратегий, следуя одной

Выборка по значимости

- Оценка матожидания другого распределения:

$$\mathrm{E}_{X\sim P}[f(X)] = \sum P(X)f(X) =$$

$$= \sum Q(X)\frac{P(X)}{Q(X)}f(X) = E_{X\sim Q}\left[\frac{P(X)}{Q(X)}f(X)\right]$$

Выборка по значимости для МС по чужому опыту

- Изучаем отдачи полученные по μ, для вычисления π
- Взвешиваем отдачу R_t в соответствии со сходством стратегий
- Считаем поправки для выборки значимости по всему эпизоду:

$$R_t^{\pi/\mu} = \frac{\pi(a_t, s_t)\pi(a_{t+1}, s_{t+1})}{\mu(a_t, s_t)\mu(a_{t+1}, s_{t+1})} \dots \frac{\pi(a_T, s_T)}{\mu(a_T, s_T)} R_t$$

- Обновляем стратегию на основе скорректированной отдачи:

$$V(s_t) \leftarrow V(s_t) + \alpha(R_t^{\pi/\mu} - \mathrm{V}(\mathrm{s_t}))$$

- Не применяем если μ нулевая, а π не нулевая
- Выборка по значимости может существенно увеличить дисперсию

Выборка по значимости для TD по чужому опыту

- Изучаем TD показатели полученные по μ, для вычисления π
- Взвешиваем TD показатель $R + \gamma V(s')$ в соответствии с выборкой по значимости
- Необходима только одна поправка по значимости:

$$V(s_t) \leftarrow V(s_t) + \alpha \left(\frac{\pi(a_t, s_t)}{\mu(a_t, s_t)} (r_{t+1} + \gamma V(s_{t+1})) - V(s_t) \right)$$

- На много более низкая дисперсия по сравнению с MC
- Стратегии должны быть схожи только на одном шаге итерации

Q обучение

- Рассмотрим обучение по чужому опыту для функции Q(s,a)
- Выборка по значимости здесь не требуется
- Следующее действие a_{t+1} выбираем по стратегии $\mu(\cdot\,|s_t)$
- Но мы рассматриваем следующее действие a' по $\pi(\cdot\,|s_t)$
- Обновляем $Q(s_t, a_t)$ в соответствии с полезностями альтернативного действия

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1} + \gamma Q(s_{t+1}, a') - Q(s_t, a_t))$$

Контроль по чужому опыту с Q-обучением

- Будем улучшать как целевую, так и актуальную стратегию
- Целевая стратегия π улучшается жадно с учетом $Q(s,a)$:

$$\pi(s_{t+1}) = \underset{a'}{\operatorname{argmax}}\, Q(s_{t+1}, a')$$

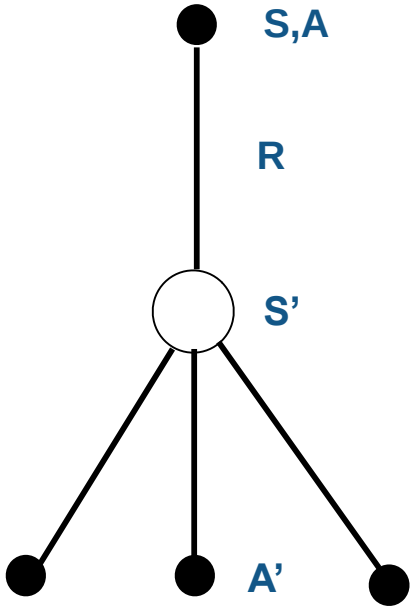
- Актуальная стратегия μ, например, ε – жадно, также с учетом $Q(s,a)$
- Показатель Q обучения упрощается:

$$\mathrm{r}_{t+1} + \gamma Q(s_{t+1}, a') =$$

$$\mathrm{r}_{t+1} + \gamma Q\left(s_{t+1}, \underset{a'}{\operatorname{argmax}}\, Q(s_{t+1}, a')\right) =$$

$$\mathrm{r}_{t+1} + \max_{a'} \gamma Q(s_{t+1}, a')$$

Алгоритм управления с Q обучением (SARSAMAX)



$$\mathrm{Q}(s,a\,) \leftarrow Q(s,a\,) + \alpha(r + \gamma \max_{a'} Q(s',a') - Q(s,a\,))$$