

МЕТОДЫ МОНТЕ-КАРЛО В RL

Сергей Николенко

СПбГУ — Санкт-Петербург

22 апреля 2022 г.

Random facts:

- 22 апреля 1418 г. закончился Констанцкий собор, завершивший Великий западный раскол в Римской церкви в 1378—1417 годах, когда сразу два (а с 1409 года — три) претендента объявили себя истинными папами; собор низложил всех троих и избрал четвёртого, Мартина V
- 22 апреля 1793 г. Джордж Вашингтон обнародовал Декларацию о нейтралитете США в войне европейских стран против революционной Франции, а 22 апреля 1864 г. Конгресс США принял закон, по которому на всех американских банкнотах появилась надпись «In God We Trust»
- 22 апреля 1838 г. в гавань Нью-Йорка прибыл английский пассажирский колёсный пароход «Сириус»; впервые пароход без парусов пересёк Атлантический океан
- 22 апреля 1943 г. Альберт Хофманн сделал первое сообщение о своей поездке на велосипеде
- 22 апреля 1993 г. вышла первая версия веб-браузера Mosaic

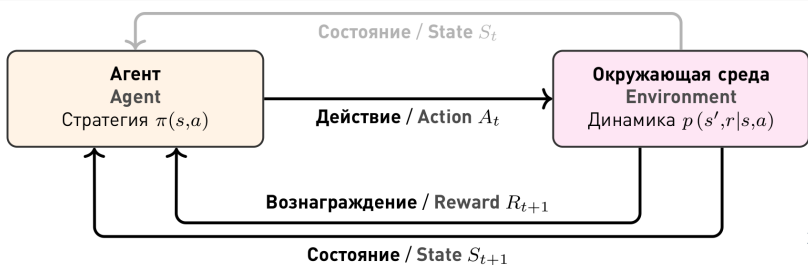
ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ

- В прошлый раз мы ввели основные понятия динамики марковских процессов принятия решений:
 - собственно динамику процесса:

$$p(s', r | s, a) = p(S_t = s', R_t = r | S_{t-1} = s, A_{t-1} = a |;)$$

- награды за каждый эпизод, начиная со времени t :

$$G_t = R_{t+1} + \gamma G_{t+1} = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1};$$

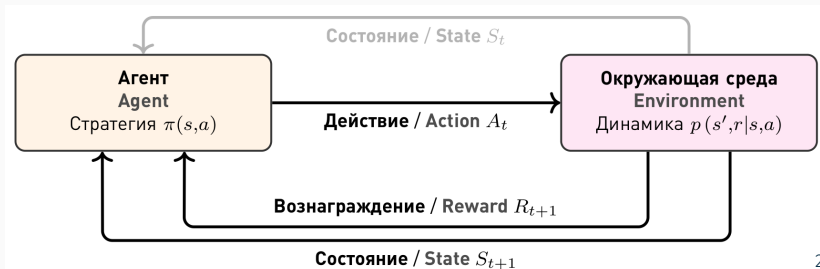


- Определили функции значений V и Q

$$V_{\pi}(s) = \mathbb{E}_{\pi} [G_t \mid S_t = s] = \mathbb{E}_{\pi} \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s \right],$$

$$Q_{\pi}(s, a) = \mathbb{E}_{\pi} [G_t \mid S_t = s, A_t = a] = \mathbb{E}_{\pi} \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a \right]$$

- Выписали уравнения Беллмана и научились их решать.



ОСНОВНЫЕ ЗАДАЧИ

- Теоретически всё готово, но у нас много проблем:
 - уравнения знаем, но пока не знаем, как их решать, то есть как найти V^π для данного π ?
 - разных стратегий очень, очень много — как найти оптимальную стратегию поведения агента в данной модели и соответствующие V^* ?
 - но уравнений тоже не знаем — в реальности обычно P и R не даны, их тоже нужно обучить; как?
 - более того, их обычно даже записать не получится, слишком уж много состояний в любой реальной задаче... что делать?



- Давайте есть слона по частям...

ИТЕРАТИВНОЕ РЕШЕНИЕ (ПО СТРАТЕГИЯМ)

- Ищем оптимальную стратегию итеративным алгоритмом.
- **PolicyIteration** – инициализировать π , потом, пока $\pi \neq \pi'$, повторять:

- вычислить значения состояний для стратегии π , решив систему линейных уравнений

$$V^\pi(s) := \sum_a \pi(s, a) \sum_{s' \in S} P_{ss'}^a (R_{ss'}^a + \gamma V^\pi(s'));$$

- улучшить стратегию на каждом состоянии:

$$\pi'(s) := \arg \max_a Q^\pi(s, a) = \arg \max_a P_{ss'}^a (R_{ss'}^a + \gamma V^\pi(s'));$$

- Почему оно сходится?

ИТЕРАТИВНОЕ РЕШЕНИЕ (ПО СТРАТЕГИЯМ)

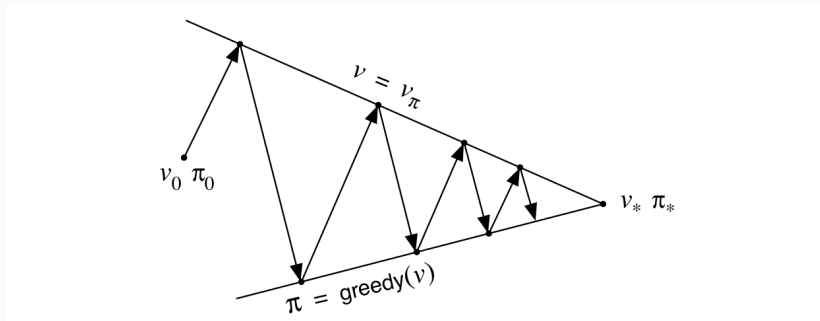
- Сходится, т.к. на каждом шаге строго улучшаем целевую функцию, а всего существует конечное число ($|A|^{|S|}$) стратегий.
- Но, конечно, это медленно, надо V^π пересчитывать; проще делать на каждой итерации ровно один шаг пересчёта V^π , а потом сразу выбирать жадную стратегию:

$$V_{k+1}(s) := \max_a \sum_{s' \in S} P_{ss'}^a (R_{ss'}^a + \gamma V_k(s')).$$

- Это называется value iteration.

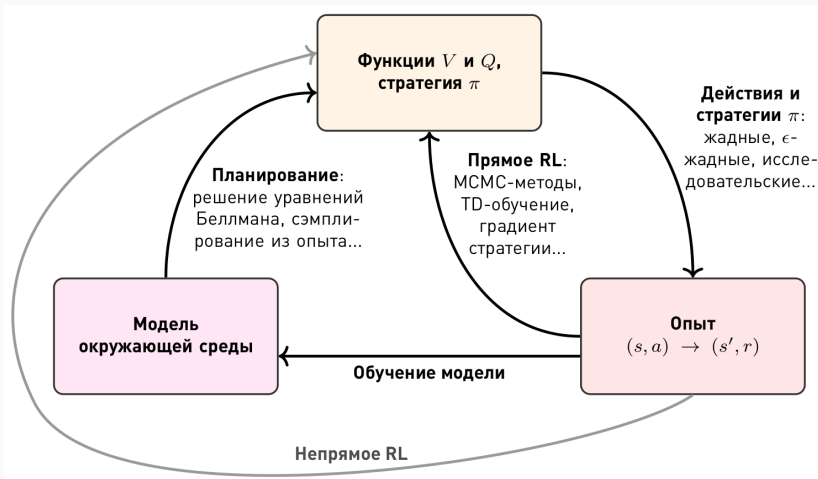
ИТЕРАТИВНОЕ РЕШЕНИЕ (ПО СТРАТЕГИЯМ)

- Есть другие похожие методы – их всех объединяет подход, основанный по сути на чём-то вроде EM-алгоритма с динамическим программированием.



- Это может быть достаточно эффективно даже для больших задач (с трюками, позволяющими не всё пространство исследовать).

ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ



МЕТОДЫ МОНТЕ-КАРЛО

- В прошлый раз мы выписали уравнения Беллмана на V и Q и научились их решать.
- Теперь будем обучать одновременно и модель, и оптимальную стратегию; вознаграждения и переходы не даны.
- Начнём со стохастических алгоритмов (метода Монте-Карло); но начнём опять с простой задачи.
- Как обучить вознаграждения $V^\pi(s)$, ожидаемые от состояния s в эпизодической задаче?

- Да очень просто: будем накапливать данные и усреднять.

Алгоритм Monte Carlo estimation:

- инициализировать случайно π и $V(s)$, пустые списки $\text{Ret}(s)$;
- повторять до сходимости:
 - сгенерировать эпизод $S_0, A_0, R_1, S_1, A_1, \dots, S_T$ по стратегии π ;
 - $G := 0$
 - для каждого $t = T - 1, T - 2, \dots, 0$:
 - $G := \gamma G + R_{t+1}$
 - если надо, то добавить G в $\text{Ret}(S_t)$ и обновить $V(S_t) := \text{Avg}(\text{Ret}(S_t))$.
- «Если надо» скрывает тонкую разницу между first-visit и every-visit Monte Carlo.
- На выходе этот алгоритм выдаст V_π для данной π , которой порождаются эпизоды.

- Но вообще без модели гораздо удобнее оценивать Q_{π} . Сразу можно и стратегию обновлять, тот же policy iteration.

Алгоритм Monte Carlo control with exploring starts:

- инициализировать случайно π и $Q(s)$, пустые списки $\text{Ret}(s)$;
- повторять до сходимости:
 - выбрать S_0, A_0 случайно так, чтобы $\forall (s, a) \quad p(s, a) > 0$;
 - сгенерировать эпизод $S_0, A_0, R_1, S_1, A_1, \dots, S_T$ по стратегии π ;
 - $G := 0$
 - для каждого $t = T - 1, T - 2, \dots, 0$:
 - $G := \gamma G + R_{t+1}$
 - если надо, то добавить G в $\text{Ret}(S_t, A_t)$ и обновить:

$$Q(S_t, A_t) := \text{Avg}(\text{Ret}(S_t, A_t)),$$

$$\pi(S_t) := \arg \max_a Q(S_t, a).$$

- Этот алгоритм выдаст π_* и соответствующую ей функцию Q_* .
- Здесь важно предположение exploring starts, без него мы не исследуем все действия.

- А если оно не выполняется, придётся исследовать самим.

Алгоритм on-policy Monte Carlo control с мягкими стратегиями:

- инициализировать случайно ϵ -мягкую π и $Q(s)$, пустые $\text{Ret}(s)$;
- повторять до сходимости:
 - выбрать S_0, A_0 случайно так, чтобы $\forall (s, a) \quad p(s, a) > 0$;
 - сгенерировать эпизод $S_0, A_0, R_1, S_1, A_1, \dots, S_T$ по стратегии π ;
 - $G := 0$
 - для каждого $t = T - 1, T - 2, \dots, 0$:
 - $G := \gamma G + R_{t+1}$
 - если надо, то добавить G в $\text{Ret}(S_t, A_t)$ и обновить:

$$Q(S_t, A_t) := \text{Avg}(\text{Ret}(S_t, A_t)),$$

$$a_* := \arg \max_a Q(S_t, a),$$

$$\pi(a | S_t) := \begin{cases} 1 - \epsilon + \frac{\epsilon}{|A(S_t)|}, & \text{если } a = a_*, \\ \frac{\epsilon}{|A(S_t)|}, & \text{если } a \neq a_*. \end{cases}$$

- Этот алгоритм умеет искать оптимальную мягкую π .

- На самом деле для ϵ -мягких стратегий тоже верен аналог policy improvement теоремы, и для ϵ -мягких стратегий метод policy iteration тоже вполне работает.
- Но это on-policy алгоритм, он найдёт мягкую стратегию, а в реальности шахматист, который играет как Магнус Карлсен 90% ходов, а 10% ходов делает случайно, вряд ли продвинется сильно дальше третьего разряда.
- Но ведь и исследовать тоже нужно! Хорошо было бы научиться исследовать по одной стратегии, а оценивать другую...

- ...и такой трюк действительно можно сделать!
- Вспомним сэмплирование со значимостями (importance sampling): если мы умеем брать сэмплы по распределению $q(\mathbf{x})$, а оценивать хотим ожидание по распределению $p(\mathbf{x})$, то можно сделать так:

$$\mathbb{E}_{p(\mathbf{x})} [f(\mathbf{x})] = \int f(\mathbf{x})p(\mathbf{x})d\mathbf{x} = \int f(\mathbf{x})\frac{p(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x} = \mathbb{E}_{q(\mathbf{x})} \left[f(\mathbf{x})\frac{p(\mathbf{x})}{q(\mathbf{x})} \right].$$

- А у нас в качестве p и q выступают распределения на траекториях:

$$\begin{aligned} p(\text{Traj} | \pi) &= p(A_t, S_{t+1}, A_{t+1}, \dots, S_{T-1}, A_{T-1}, S_T | \pi) = \\ &= \pi(A_t | S_t) p(S_{t+1} | S_t, A_t) \dots \pi(A_{T-1} | S_{T-1}) p(S_T | S_{T-1}, A_{T-1}) = \\ &= \prod_{k=t}^{T-1} \pi(A_k | S_k) p(S_{k+1} | S_k, A_k). \end{aligned}$$

- И получается, что для двух стратегий π, b (от слова behaviour) определить веса

$$\begin{aligned}\rho_{t:T-1}^{\pi,b} &= \frac{p(\text{Traj} | \pi)}{p(\text{Traj} | b)} = \frac{\prod_{k=t}^{T-1} \pi(A_k | S_k) p(S_{k+1} | S_k, A_k)}{\prod_{k=t}^{T-1} b(A_k | S_k) p(S_{k+1} | S_k, A_k)} = \\ &= \frac{\prod_{k=t}^{T-1} \pi(A_k | S_k)}{\prod_{k=t}^{T-1} b(A_k | S_k)},\end{aligned}$$

то неизвестные вероятности сократятся и останется, что когда мы порождаем эпизоды по b , нужно просто усреднять не G_t , а $\rho_{t:T-1}^{\pi,b} G_t$, и будут получаться оценки π !

- Единственное условие – покрытие (coverage): должно быть верно, что если $\pi(a|s) > 0$, то и $b(a | s) > 0$.

- Ещё есть тонкая разница между обычным importance sampling и взвешенным (weighted importance sampling):
 - в обычном мы берём оценку среднего через сэмплы

$$V(s) = \frac{1}{N} \sum_{t=1}^N \rho_{t:T-1}^{\pi,b} G_t,$$

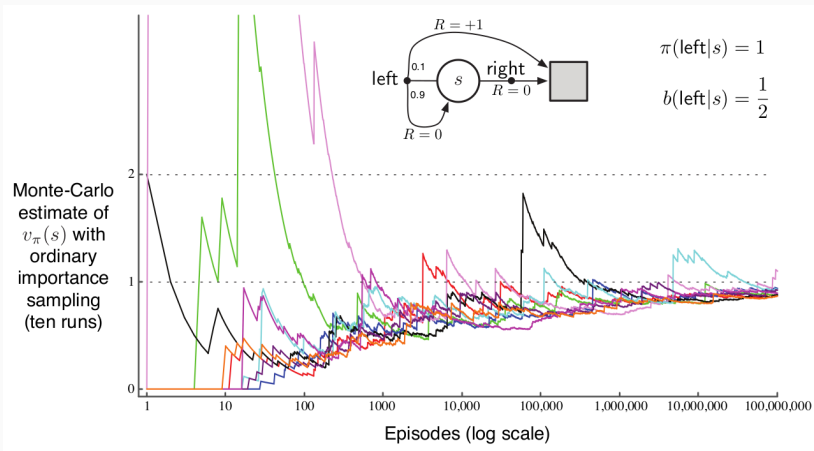
- а во взвешенном ещё нормируем суммой весов

$$V(s) = \frac{\sum_{t=1}^N \rho_{t:T-1}^{\pi,b} G_t}{\sum_{t=1}^N \rho_{t:T-1}^{\pi,b}}.$$

- Вторая оценка смещённая, но сходится куда надо и у неё нормальная человеческая дисперсия.

МЕТОД МОНТЕ-КАРЛО

- А у первого варианта дисперсия очень большая, и может быть даже бесконечная! Пример:



- Итого вот какой алгоритм получается

Алгоритм on-policy Monte Carlo control с мягкими стратегиями:

- инициализировать ϵ -мягкую b , π , $Q(s)$, пустые $\text{Ret}(s)$ и $c(s)$;
- повторять до сходимости:
 - сгенерировать эпизод $S_0, A_0, R_1, S_1, A_1, \dots, S_T$ по мягкой стратегии b ;
 - $G := 0, W := 1$
 - для каждого $t = T - 1, T - 2, \dots, 0$:
 - $G := \gamma G + R_{t+1}$
 - $c(S_t, A_t) := c(S_t, A_t) + W$
 - $Q(S_t, A_t) := Q(S_t, A_t) + \frac{W}{c(S_t, A_t)} (G - Q(S_t, A_t))$
 - $\pi(a | S_t) := \arg \max_a Q(S_t, a)$
 - если $A_t \neq \pi(S_t)$, то перейти к следующему эпизоду
 - $W := \frac{W}{b(A_t | S_t)}$.
- А если убрать $\arg \max$, то получится просто алгоритм оценки данной стратегии π .

Спасибо за внимание!