Введение в обучение с подкреплением

Тема 8: TD методы с разделённой стратегией

Лектор: Кривошеин А.В.

Q-learning

Q-learning — метод TD-обучения **с разделённой стратегией,** основанный на оценивании Q-функции, как и SARSA.

Но SARSA занимается приближением Q-функции конкретной стратегии q_π,

а Q-learning сразу занимается **приближением оптимальной ценности действий** q_*.

Чтобы получить формулу обновления оценки Q-функции, напомним связь V-функции и Q-функции, записав эту связь для оптимальной стратегии

$$q_*(s,a) = \sum_{r\in\mathcal{R}}\sum_{s'\in S} p(s',r\,|\,s,a)\,(r + \gamma\, v_*(s')).$$

При этом тот факт, что стратегия оптимальна, означает: $v_*(s) = \max\limits_a q_*(s,a)$.

Следовательно, уравнение оптимальности Беллмана для Q-функции имеет вид:

$$q_*(s,a) = \sum_{r\in\mathcal{R}}\sum_{s'\in S} p(s',r\,|\,s,a)\left(r + \gamma \max_a q_*(s',a)\right) \quad \text{или} \quad q_*(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a'} q_*(S_{t+1},a')\,\middle|\,S_t = s, A_t = a\right].$$

Оценку математического ожидания случайной величины $R_{t+1} + \gamma \max\limits_{a'} q_*(S_{t+1},a')$ можно получить путём усреднения фактически полученных значений этой величины для многих траекторий. Тогда обновление оценки оптимальной Q-функции можно получить в виде:

$$Q(S_t,A_t) \leftarrow Q(S_t,A_t) + \alpha\left[R_{t+1} + \gamma \max_{a'} Q(S_{t+1},a') - Q(S_t,A_t)\right]$$

Q-learning

Базовая идея метода Q-learning: в ходе взаимодействия агента со средой для каждой полученной четвёрки значений $S_t,\ A_t,\ R_{t+1},\ S_{t+1}$ обновлять значения Q-функции по формуле

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\Big[R_{t+1} + \gamma \max_{a'} Q(S_{t+1}, a') - Q(S_t, A_t)\Big].$$

При этом не важно по какой стратегии генерируются эти четвёрки значений $S_t,\ A_t,\ R_{t+1},\ S_{t+1}$. Агент может пользоваться некоторой исследовательской стратегией, формула обновления всё равно будет приближать оптимальную Q-функцию.

Сравним эту формулу обновления с обновлением по методу SARSA:

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

В **SARSA** цель обновления формировалась с помощью текущей стратегии, так как надо выбрать действие A_{t+1}.
В **Q-learning** цель обновления **не зависит** от используемой стратегии, а формируется с помощью лучшей текущей оценки Q-функции.

Это и позволяет сделать Q-learning методом обучения **с разделённой стратегией**. То есть генерировать траектории можно по исследовательской стратегии, формируя при этом оценки Q-функции для другой стратегии.

Q-learning, псевдокод

Приведём псевдокод алгоритма Q-learning для оценки оптимальной Q-функции q_*. Фиксируем размер шага обучения $\alpha > 0$, а также $\gamma \in (0,\ 1)$.

---

1. Инициализировать:

значения $Q(s,\ a)\ = 0$ для всех состояний $s \in \mathcal{S}$ и действий $a \in \mathcal{A}$,

задать поведенческую стратегию

2. Q-learning

Повторять для каждого эпизода:

Выбрать начальное состояние s

Повторять:

Выбрать a, следуя поведенческой стратегии

Наблюдать r, s'

Обновить оценку

$$Q(s,\ a)\ :=\ Q(s,\ a)\ + \alpha\left[R + \gamma \max_a Q(s',\ a)\ - Q(s,\ a)\right]$$

$s := s'$

Если s заключительное состояние, то выйти из цикла.

Вернуть стратегию π на основе найденой Q-функции.

Условия сходимости алгоритма **Q-learning**: поведенческая стратегия с потенциально бесконечным числом посещения каждой пары $(s,\ a)$ + условия Роббинса-Монро на убывание величины шага обучения.

Q-learning: особенности

Обсудим выбор поведенческой стратегии.

Если в ходе взаимодействия со средой агент просто обучается, то поведенческая стратегия может быть любой, лишь бы она обеспечивала выбор каждой пары $(s,\ a)$ потенциально бесконечное число раз (например, равновероятный выбор действий).

Если в ходе взаимодействия надо обучаться и одновременно показывать всё более лучшие результаты взаимодействия со средой, то в качестве поведенческой стратегии можно выбрать ε-жадную стратегию относительно Q. Она также обеспечивает выбор каждой пары $(s,\ a)$ потенциально бесконечное число раз, но чаще будет выбираться действие с наибольшей оценкой.

Преимуществом Q-learning является то, что обучаться можно на любом наборе сгенерированных траекторий.

Недостатком Q-learning является тот факт, что при обучении на конечном наборе данных оценки Q-функции будут смещёнными, а именно завышенными относительно истинных.

Конечно, в процессе обучения смещение будет убывать к нулю. Однако, в начале обучения завышение оценок $Q(s, a)$ может приводить к тому, что те действия a, которые агент считает лучшими, такими не являются.
Кроме того, вместе с бутстреппингом, завышенные оценки будут распространяться и в более ранние пары, внося смещение в их оценки.

О смещённых оценках

Пусть есть некоторый параметр θ, значение которого требуется оценить из набора из N наблюдений.
Можно представить, что этот набор из N наблюдений формируется несколько раз. Тогда фактически каждое наблюдение — это реализация некоторой случайной величины.

Тогда по сути нам надо оценить θ по набору из N случайных величин ξ_1, ..., ξ_N. Оценку параметра θ по набору из N случайных величин будем обозначать $\hat{\theta}_N$.

В целом, хотелось бы, чтобы оценка в среднем была бы равна истинному значению. Разницу между математическим ожиданием оценки и истинным значением называют **смещением оценки** (англ. bias):

$$\mathrm{Bias}(\hat{\theta}_N) = \theta - \mathbb{E}[\hat{\theta}_N].$$

Если смещение нулевое, то тогда действительно, оценка в среднем равна истинному значению $\mathbb{E}[\hat{\theta}_N] = \theta$. В этом случае оценку называют **несмещённой**.
Если же $\mathrm{Bias}(\hat{\theta}_N) \neq 0$, то оценку называют **смещённой**.

Если оценка смещённая, но это смещение убывают к нулю с ростом числа наблюдений, то есть

$$\lim_{N \to \infty} \mathbb{E}[\hat{\theta}_N] = \theta,$$

то оценку называют асимптотически несмещённой. Обычно от хорошей оценки требуется, чтобы она была бы несмещённой или асимптотически несмещённой.

О смещённых оценках

Однако, следующий пример показывает, что смещение не единственная важная характеристика оценки.

Пусть ξ — это случайная величина, которая выдаёт результат подбрасывания монетки, 1 или -1. Если монетка нечестная, то пусть вероятность “орла” p, а вероятность “решки” $1-p$. Тогда среднее ξ имеет вид

$$\mathbb{E}[\xi] = p + (-1)(1-p) = 2p - 1.$$

Пусть величина p неизвестна и мы хотим оценить математическое ожидание $\theta = \mathbb{E}[\xi]$. Подбросим монетку N раз, будем обозначать каждый результат за x_i, это значение является реализацией случайной величины ξ_i. Рассмотрим оценку вида

$\hat{\theta} = \xi_N$. Ясно, что $\mathbb{E}[\hat{\theta}] = \mathbb{E}[\xi_N] = 2p - 1$,

то есть оценка несмещённая. Но ясно, что такая оценка не будет хорошей. Причина в том, что значение оценки $\hat{\theta}$ будет равно 1 с вероятностью p и -1 с вероятностью $1-p$. Тогда точность оценки не улучшается с ростом числа наблюдений. В частности можно отметить, что дисперсия

$$\mathbb{D}[\hat{\theta}] = \mathbb{D}[\xi_N] = \mathbb{E}[(\xi_N - \mathbb{E}[\xi_N])^2] = \mathbb{E}[\xi_N^2] - \mathbb{E}[\xi_N]^2 = 1 - (2p-1)^2 = 4p - 4p^2$$

не убывает с ростом N.

О дисперсии оценок

Чтобы в некотором смысле оценка параметра сходилась бы с истинному значению, можно потребовать, чтобы дисперсия оценки стремилась к нулю с ростом числа наблюдений

$$\lim_{N\to\infty} \mathbb{D}[\hat{\theta}_N] = \lim_{N\to\infty} \mathbb{E}\left[\left|\hat{\theta}_N - \mathbb{E}[\hat{\theta}_N]\right|^2\right] = 0$$

В случае несмещённой оценки $\mathbb{E}[\hat{\theta}_N] = \theta$ по неравенству Чебышёва для любого $\varepsilon > 0$ верна оценка

$$\mathbb{P}\left[\left|\hat{\theta}_N - \theta\right| \geq \varepsilon\right] \leq \frac{\mathbb{D}[\hat{\theta}_N]}{\varepsilon^2}.$$

Тогда при $\mathbb{D}[\hat{\theta}_N] \to 0$ при $N \to \infty$ вероятность того, что $\hat{\theta}_N$ отличается от θ больше чем на ε, стремится к нулю.

На практике часто строятся оценки, которые обеспечивают сходимость оценки $\hat{\theta}_N$ к θ в MSE смысле, то есть

$$\lim_{N\to\infty} \mathrm{MSE}(\hat{\theta}_N) = \lim_{N\to\infty} \mathbb{E}\left[\left|\hat{\theta}_N - \theta\right|^2\right] = 0.$$ При этом верно, что $\mathrm{MSE}(\hat{\theta}_N) = \mathbb{E}\left[\left|\hat{\theta}_N - \theta\right|^2\right] = \mathbb{D}[\hat{\theta}_N] + \mathrm{Bias}(\hat{\theta}_N)^2$,

так как $\mathbb{D}[\hat{\theta}_N] + \mathrm{Bias}(\hat{\theta}_N)^2 =$

$$\mathbb{E}\left[\left|\hat{\theta}_N - \mathbb{E}[\hat{\theta}_N]\right|^2\right] + \left(\theta - \mathbb{E}[\hat{\theta}_N]\right)^2 = \mathbb{E}[\hat{\theta}_N^2] - 2\,\mathbb{E}[\hat{\theta}_N]^2 + \mathbb{E}[\hat{\theta}_N]^2 + \theta^2 - 2\,\theta\,\mathbb{E}[\hat{\theta}_N] + \mathbb{E}[\hat{\theta}_N]^2 = \mathbb{E}[\hat{\theta}_N^2] - 2\,\theta\,\mathbb{E}[\hat{\theta}_N] + \theta^2 = \mathbb{E}\left[\left|\hat{\theta}_N - \theta\right|^2\right].$$

Разные способы построения оценок дают свой баланс между смещением и дисперсией оценки.

Оценку называют **состоятельной**, если она сходится в некотором смысле, к истинному значению параметра.

Для MSE сходимости оценка состоятельна, если она асимптотически несмещённая и $\lim_{N\to\infty} \mathbb{D}[\hat{\theta}_N] = 0$.

Q-learning: особенности

Продемонстрируем эффект завышенных оценок на примере среды с одним состоянием s и некоторым набором действий, возвращающих в состояние s.

Пусть вознаграждения формируются реализацией случайной величины с нормальным распределением $\mathcal{N}(0, 1)$. Тогда истинные оптимальные ценности всех действий $q_*(s, a) = 0$.

Однако, оценки $Q(s, a)$ по методу Q-learning будет завышены относительно нуля:
проведём симуляцию для 4 действий на 100000 эпизодах, выбор действия равновероятен.

```
Episodes = 100000; SeedRandom[42]; Acts = 4;
Q = Table[0, Acts]; Num = Table[0, Acts];
For[i = 1, i < Episodes, i++,
 Rew = Table[RandomVariate[NormalDistribution[]], {k, 1, 4}];
 k = RandomChoice[Range[Acts]];
 Num[[k]] = Num[[k]] + 1;
 Q[[k]] = Q[[k]] + (Rew[[k]] + 0.9 Max[Q] - Q[[k]]) / Num[[k]];]
Q
```

{0.948804, 0.935933, 0.933287, 0.944938}

Конечно же чем больше данных доступно, тем меньше будет это смещение. В пределе смещение исчезнет.

```
For[i = 1, i < Episodes, i++,
 Rew = Table[RandomVariate[NormalDistribution[]], {k, 1, 4}];
 k = RandomChoice[Range[Acts]];
 Num[[k]] = Num[[k]] + 1;
 Q[[k]] = Q[[k]] + (Rew[[k]] + 0.9 Max[Q] - Q[[k]]) / Num[[k]];]
Q
```

{0.887875, 0.881954, 0.872695, 0.882199}

Q-learning: особенности

Если же истинные значения ценностей различны, то завышение оценок может помешать сразу выявить самое ценное действие.

Проведём симуляцию для 2 действий a_1, a_2 на 10000 эпизодах, выбирая действия по ε-жадной стратегии.

За действие a_1 вознаграждение всегда $R = 0$, $q_*(s, a) = 0$.

За действие a_2 вознаграждение $R \sim \mathcal{N}(-0.2, 1)$, $q_*(s, a) = -0.2$.

```
Episodes = 10000; SeedRandom[14]; Acts = 2;
Q = Table[0, Acts]; Num = Table[0, Acts];
For[i = 1, i < Episodes, i++,
 Rew = {0, RandomVariate[NormalDistribution[]] - 0.2};
 If[RandomReal[] > 0.01 ,
      k = First@Flatten[Position[Q, Max[Q]]],
      k = RandomChoice[Range[Acts]]];
 Num[[k]] = Num[[k]] + 1;
 Q[[k]] = Q[[k]] + (Rew[[k]] + 0.9 Max[Q] - Q[[k]]) / Num[[k]];]
Print["Оценки ценности действий:
Q(s,a1) =", Q[[1]], "
Q(s,a2) =", Q[[2]]]
```

```
Оценки ценности действий:
Q(s,a1)=0.0497841
Q(s,a2)=0.353094
```

С увеличением объёма данных истинно ценное действие будет обнаружено.

```
For[i = 1, i < 350000, i++,
 Rew = {0, RandomVariate[NormalDistribution[]] - 0.2};
 If[RandomReal[] > 0.01 ,
     k = First@Flatten[Position[Q, Max[Q]]],
     k = RandomChoice[Range[Acts]]];
 Num[[k]] = Num[[k]] + 1;
 Q[[k]] = Q[[k]] + (Rew[[k]] + 0.9 Max[Q] - Q[[k]])/Num[[k]];]
Print["Оценки ценности действий:
Q(s,a1)=", Q[[1]], "
Q(s,a2)=", Q[[2]]]
```

```
Оценки ценности действий:
Q(s,a1)=0.0379957
Q(s,a2)=0.0362189
```

Expected SARSA

Рассмотрим гибрид методов Q-обучения и SARSA называемый **Expected SARSA**. Пусть траектория взаимодействия агента со средой формируется по некоторой поведенческой стратегии π_b. Рассмотрим ещё одну стратегию π (она может и совпадать с π_b) и правило обновления:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\Big[R_{t+1} + \gamma \sum_{a \in \mathcal{A}} \pi(a \mid S_{t+1})\, Q(S_{t+1}, a) - Q(S_t, A_t)\Big]$$

В этом правиле цель обновления формируется по-новому.

Вместо максимума $\max_a Q(S_{t+1}, a)$ как в **Q-learning** или

вместо текущей оценки $Q(S_{t+1}, A_{t+1})$, где A_{t+1} формируется по стратегии π_b, как в **SARSA**,

для формирования цели обновления используется усреднение оценок $Q(S_{t+1}, a)$ по возможным действиям с вероятностями выбора этих действий в рамках стратегии π.

Что же оценивает Q-функция с таким правилом обновления? Напомним уравнение Беллмана для функции ценности действий стратегии π и запишем его с помощью математического ожидания:

$$q_\pi(s, a) = \sum_{r \in \mathcal{R}} \sum_{s' \in \mathcal{S}} p(s', r \mid s, a)\Big(r + \gamma \sum_{a' \in \mathcal{A}} \pi(a' \mid s')\, q_\pi(s', a')\Big) = \mathbb{E}\Big[R_{t+1} + \gamma \sum_{a' \in \mathcal{A}} \pi(a' \mid S_{t+1})\, q_\pi(S_{t+1}, a') \,\Big|\, S_t = s, A_t = a\Big].$$

Expected SARSA

Таким образом, Expected SARSA с указанным выше правилом обновления оценивает функцию ценности действий $q_\pi(s, a)$. При этом не важно откуда получаются отсчёты для правила обновления, усреднение проводится по всем возможным будущим вознаграждениям R_{t+1} и состояниям S_{t+1}.

То есть Expected SARSA можно запускать как метод с единой стратегией и как метод с разделённой стратегией, где поведенческая стратегия для генерации траектории более исследовательская, а целевая стратегия — это ε-жадная или жадная стратегия.

Известно, что дисперсия оценок по методу Expected SARSA меньше, чем дисперсия оценок по методу SARSA из-за того, что в SARSA цель обновления формируется по текущей оценке $Q(S_{t+1}, A_{t+1})$ и для разных действий может быть большой разброс оценок, а в Expected SARSA высчитывается ожидание по целевой стратегии.

Однако, Expected SARSA, чуть более вычислительно затратно, чем SARSA.

Expected SARSA, псевдокод

Приведём псевдокод для метода Expected SARSA с единой ε-жадной стратегией.

---

1. Инициализировать:

значения $Q(s, a) = 0$ для всех состояний $s \in \mathcal{S}$ и действий $a \in \mathcal{A}$,

стратегию π определить ε-жадно относительно Q-функции

2. Expected SARSA

Повторять для каждого эпизода:

Выбрать начальное состояние s

Повторять:

Выбрать a, следуя ε-жадной стратегии относительно Q-функции

Наблюдать r, s'

Обновить оценку

$$Q(s, a) := Q(s, a) + \alpha\left[R + \gamma \sum_{a \in \mathcal{A}} \pi(a \mid s')\, Q(s', a) - Q(s, a)\right]$$

$s := s'$

Если s заключительное состояние, то выйти из цикла.

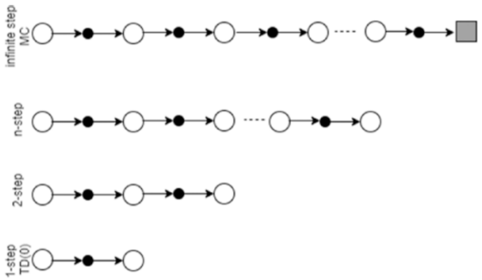
Уменьшить ε.

Вернуть стратегию π на основе найденой Q-функции.

Условия сходимости аналогичны условиям сходимости для метода SARSA (GLIE + условия Роббинса-Монро).

TD: n-шаговые методы

TD метод является 1-шаговым методом, а метод МК является ∞-шаговым методом.



Можно естественным образом привести n-шаговый алгоритм SARSA и Expected SARSA с единой стратегией, $n \geq 1$. Цель обновления для n-шагового SARSA на k-ой итерации будет иметь вид:

$$G_{t:t+n} = R_{t+1} + \gamma R_{t+2} + \ldots + \gamma^{n-1} R_{t+n} + \gamma^n Q_k(S_{t+n}, A_{t+n}), \quad \text{если} \quad 0 \leq t < T - n,$$

$$G_{t:t+n} = G_t, \quad \text{если} \; t \geq T - n.$$

Чтобы обновить оценку для текущей пары (S_t, A_t) надо подождать n шагов, сформировать цель обновления и

$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\,(G_{t:t+n} - Q(S_t, A_t))$. При $n = 1$ это обновление соответствует обычному SARSA.

TD: n-шаговые методы

Приведём псевдокод n-шагового SARSA с единой ε-жадной стратегией, $\alpha > 0,\ \varepsilon > 0,\ 0 < \gamma \le 1$, а также $n \ge 1$.

---

1. **Инициализировать:** значения $Q(s, a) = 0$ для всех состояний $s \in \mathcal{S}$ и действий $a \in \mathcal{A}$,
2. **n-шаговый SARSA**

Повторять для каждого эпизода:

  Выбрать и сохранить начальное состояние S_0, $T = \infty$

  Выбрать действие A_0 ε-жадно относительно Q-функции и сохранить

  Повторять для $t = 0, 1, 2, \ldots$:

    Если $t < T$, то:

      Сделать действие A_t, наблюдать и сохранить вознаграждение R_{t+1} и новое состояние S_{t+1}

      Если S_{t+1} заключительно, то: $T = t + 1$

      Иначе: Выбрать и сохранить действие A_{t+1} ε-жадно относительно Q-функции

    $\tau := t - n + 1$ (момент времени n шагов назад, для которого будем обновлять оценку)

    Если $\tau \ge 0$:

$$G = \sum_{i=\tau+1}^{\min(\tau+n,T)} \gamma^{i-\tau-1} R_i$$

      Если $\tau + n < T$, то $G := G + \gamma^n Q(S_{\tau+n}, A_{\tau+n})$

      Обновить оценку $Q(S_\tau, A_\tau) := Q(S_\tau, A_\tau) + \alpha[G - Q(S_\tau, A_\tau)]$

    Если $\tau = T - 1$, то выйти из цикла.

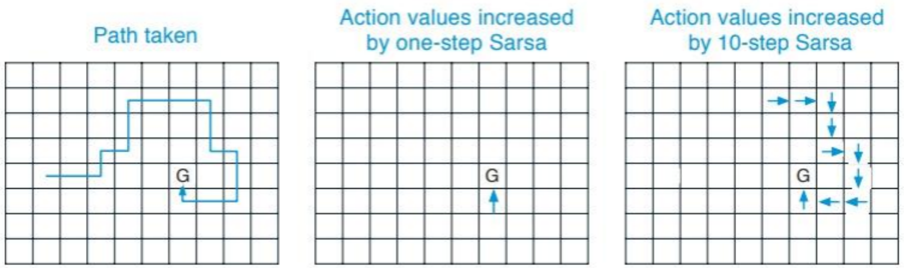
  Уменьшить ε

TD: n-шаговые методы

Условия для сходимости n-шагового SARSA аналогичны SARSA:

GLIE + условия Роббинса-Монро на шаг обучения.

Ниже демонстрируется сравнение 1-шагового и 10-шагового SARSA



Стрелки на рисунках указывают, какие ценности действий были изменены в случае одного и второго методов. Одношаговый метод повышает ценность лишь одного последнего действия. n-Шаговый SARSA обновляет ценности последних n действий, что повышает эффективность обучения в рамках одного эпизода.