Рекомендательные системы

Филиппенко Павел

May 2024

1 Многорукие бандиты

Всего рассматривается 3 товара. Запишем их характеристики в таблицу 1. Сразу внесем столбец с нормированными оценками.

1.1 Greed policy

Посчитаем ε-жадную стратегию.

Algorithm 1 ε-greed policy

if $\xi \leq \varepsilon$ **then**
    $a = \underset{a \in \mathcal{A}}{\texttt{random}}$
else
    $a = \underset{a \in \mathcal{A}}{\texttt{argmax}} Q_t(a)$
end if

В соответствии с этой стратегией получаем $\pi = [0, 0, 1]^T$ с вероятностью $1 - \varepsilon = 0.99$ и $\pi = [0.33, 0.33, 0.33]^T$ с вероятностью $\varepsilon = 0.01$. Таким образом, получаем следующую политику $\pi_\varepsilon = [0.0033, 0.0033, 0.9933]^T$.

1.2 UCB

Найдем политику UCB. Основную идею этого метода можно описать следующей формулой

$$a_t = \underset{a \in \mathcal{A}}{\texttt{argmax}} Q_t(a) + U_t(a) \tag{1}$$

Item number	reward	normalized reward	bought amount
1	4.6	0.92	181
2	4.3	0.86	21
3	4.7	0.96	384

Таблица 1: Характеристики товаров

То есть, подобная политика предполагает использование некого доверительного интервала при выборе очередного действия, чтобы более грамотно организовать exploration (исследование среды).

Для вывода классической формулы UCB используется неравенство Хёффдинга. Здесь важно заметить, что в этом неравенстве предполагается использование величин, отнормированных в отрезке $[0, 1]$, поэтому здесь мы будем использовать отнормированные значения оценок.

Запишем неравенство Хёффдинга

$$P[\mathbf{X} > \overline{X}_t + u] \leq e^{-2tu^2} \tag{2}$$

Перепишем неравенство с использованием оценок действий бандита

$$P[Q(a) - \overline{Q}_t > U_t(a)] \leq e^{-2tU_t^2(a)} \Rightarrow$$

$$U_t(a) = \sqrt{\frac{-\log p}{2N_t(a)}}$$

Возьмем $p = t^{-4}$ (USB1) и получим

$$U_t(a) = \sqrt{\frac{t}{N_t(a)}}$$

Тогда общая формула для UCB-политики

$$\pi_{ucb} = \underset{a \in \mathcal{A}}{\texttt{argmax}} Q(a) + \alpha \sqrt{\frac{t}{N_t(a)}} \tag{3}$$

где α – некоторый вес смещения $U_t(a)$

Примем за t суммарное количество покупок каждого товара + 1 (так как вычисляем стратегию для текущего действия), тогда t = 587. Подставим все оценки в формулу и получим:

$$\pi_{ucd} = \texttt{argmax} \begin{pmatrix} 0.92 + \alpha\sqrt{\frac{\log 587}{181}} \\ 0.86 + \alpha\sqrt{\frac{\log 587}{21}} \\ 0.96 + \alpha\sqrt{\frac{\log 587}{384}} \end{pmatrix}$$

Подставляя различные значения $\alpha \in \{0.1, 0.5, 1\}$ получаем, что при $\alpha = 0.1$ $\pi_{ucd} = a_3$, при $\alpha = 0.5$ $\pi_{ucd} = a_2$, при $\alpha = 1$ $\pi_{ucd} = a_2$. То есть, таким образом, мы получаем детерминированную стратегию выбора товара на каждом шаге.

1.3 Thompson Sampling

Смысл этой техники заключается в том, чтобы дать априорное распределение на каждую целевую величину (в нашем случае – на товары), а затем варьировать параметры этого распределения в зависимости от награды, получаемой на каждом шаге, приближая таким образом априорное распределение к реальному. Для того, чтобы применить в данной задачи томсоновское сэмплирование, как минимум, необходимо знать историю выдачи наград (историю оценивания каждого товара) на каждом предыдущем шаге, то есть $\forall t \in [0, T]$ знать $r_a(t)$.

Проблема данной задачи заключается в том, что сдесь мы имеем дело не с бернулевской величиной, когда мы имеем только 2 типа наград – 0 или 1, а значит использовать в качестве начального априорного распределения двупараметрическое β-распределение скорее всего будет некорректно. Есть два варианта, как можно выйти из этого положения:

Первый вариант – привести систему выдачи наград к бернулевской. Например – $\alpha = 1$ если $r_a(t) \geq 4$ и $\beta = 1$ если $r_a(t) < 4$. Другой вариант – $\alpha = 1$, если $r_a(t) \neq 0$ (пользователь выбрал товар), $\beta = 1$, если пользователь выбрал какой-то другой товар (где α – количество раз, когда агент получил награду, а β – количество раз, когда агент не получил награду).

Второй вариант – если мы знаем или предполагаем, по какому распределению агент получает награды при выборе каждого действия (например это может быть нормальное распределение), то мы можем найти для этого распределения так называемое **сопряженное априорное распределение**. Для нормального распределения сопряженным априорным является так же – номарльное распределение. Нормальное распределение описывается двумя параметрами – дисперсией и матожиданием. Перейдем от дисперсии к ее обратной величине – точности $\tau = \frac{1}{\sigma}$. Тогда при выборе очередного действия и получении очередной награды можно менять параметры распределения следующим образом:

$$\tau_0 \leftarrow \tau_0 + n\tau$$

$$\mu_0 \leftarrow \frac{\tau_0 \mu_0 + \tau \sum_{i=1}^{n} x_i}{\tau_0 + n\tau}$$

где τ_0 – точность априорного распределения, μ_0 – матожидание априорного распределения, tau – точность реального результата, n – количетво действий агента, x_i – полученное вознаграждение.

2 Counterfactual evaluation

2.1 Logging policy

Посчитаем logging policy как частоту выбора каждого товара $\pi_0 = [181, 21, 384]^T / 586 = [0.308, 0.035, 0.655]^T$.

2.2 Оценка стратегий

Оценим стратегию $\pi_1 = [0.3, 0.04, 0.66]^T$ как матожидание награды $\hat{V}(\pi_1, \mathcal{D}) = \mathbf{E}_{p(x)\pi_1(a|x)p(r|x,a)}[r]$ (полагаю, что в данном случае x – состояние агента). Заметим, что в случае многорукого бандита у нас только одно состояние, а значит для вычисления оценки нужно вычислить матожидание награды с учетом всех доступных дейтсвий.

$$\hat{V}(\pi_1, \mathcal{D}) = 0.3 \cdot 4.6 + 0.04 \cdot 4.3 + 0.66 \cdot 4.7 = 4.654$$

Аналогично оценим стратегию $\pi_2 = [0.3, 0.66, 0.04]^T$

$$\hat{V}(\pi_2, \mathcal{D}) = 0.3 \cdot 4.6 + 0.66 \cdot 4.3 + 0.04 \cdot 4.7 = 4.406$$

Оценим ε-жадную стратегию, которую мы составили в первой задаче

$$\hat{V}(\pi_\varepsilon, \mathcal{D}) = 0.0033 \cdot 4.6 + 0.0033 \cdot 4.3 + 0.9933 \cdot 4.7 = 4.698$$

Как видно, наибольшую оценку дает ε-жадная стратегия стратегия π_ε.

2.3 Оценка IPS

Посчитаем для приведенных стратегий оценку IPS (иначе зачем, собственно нам нужна была logging policy). Возмем logging policy как базовую стратегию π_0. Тогда оценка целевой стратегии π_{test}

$$\hat{V}_{IPS}(\pi_{test}, \mathcal{D}) = \frac{1}{n} \sum_{i=1}^{n} \frac{\pi_{test}(a_i|x_i)}{\pi_0(a_i|x_i)} r_i \tag{4}$$

Попробуем посчитать оценку для этих стратегий по приведенной выше формуле.

$$\hat{V}_{IPS}(\pi_1, \mathcal{D}) = \frac{1}{3}\left(\frac{0.3}{0.308} \cdot 4.6 + \frac{0.04}{0.035} \cdot 4.3 + \frac{0.66}{0.655} \cdot 4.7\right) = 4.71$$

$$\hat{V}_{IPS}(\pi_2, \mathcal{D}) = \frac{1}{3}\left(\frac{0.3}{0.308} \cdot 4.6 + \frac{0.66}{0.035} \cdot 4.3 + \frac{0.04}{0.655} \cdot 4.7\right) = 28.62$$

$$\hat{V}_{IPS}(\pi_\varepsilon, \mathcal{D}) = \frac{1}{3}\left(\frac{0.033}{0.308} \cdot 4.6 + \frac{0.0033}{0.035} \cdot 4.3 + \frac{0.9933}{0.655} \cdot 4.7\right) = 2.52$$

Как видно, с последними двумя оценками творится что-то странное. Все из-за того, что стратегии π_2 и π_ε очень сильно отличаются от стратегии π_0, из-за этого в сумме начинают появляться слишком маленькие или слишком большие веса. Для решения этой проблемы существует 2 пути. Первый вариант – при суммировании ввести некоторую верхнюю границу λ на веса

$$\hat{V}_{CIPS}(\pi_{test}, \mathcal{D}, \lambda) = \frac{1}{n}\sum_{i=1}^{n} min\left(\frac{\pi_{test}(a_i|x_i)}{\pi_0(a_i|x_i)}, \lambda\right) r_i \tag{5}$$

однако, в этом случае нам придется подбирать параметр λ. Второй способ – нормировать выражение под суммой не на n, а на сумму соответствующих весов

$$\hat{V}_{SNIPS}(\pi_{test}, \mathcal{D}) = \frac{\sum_{i=1}^{n} \frac{\pi_{test}(a_i|x_i)}{\pi_0(a_i|x_i)} r_i}{\sum_{i=1}^{n} \frac{\pi_{test}(a_i|x_i)}{\pi_0(a_i|x_i)}} \tag{6}$$

Вычислим оценки стратегий, используя последний способ

$$\hat{V}_{SNIPS}(\pi_1, \mathcal{D}) = 4.5$$

$$\hat{V}_{SNIPS}(\pi_2, \mathcal{D}) = 4.3$$

$$\hat{V}_{SNIPS}(\pi_\varepsilon, \mathcal{D}) = 4.67$$

3 Несмещенность IPS

Докажем, что оценивание стратегий через IPS несмещенное:

$$\mathbf{E}_{\mathcal{D}}\left[\hat{V}_{ips}(\pi, \mathcal{D})\right] = V(\pi) = \mathbf{E}_{p(x)\pi(a|x)p(r|x,a)} \tag{7}$$

$$\mathbf{E}_{\mathcal{D}}\left[\hat{V}_{ips}(\pi, \mathcal{D})\right] = \frac{1}{n}\sum_{x_1,a_1}\cdots\sum_{x_n,a_n}\left[\sum_{i=1}^{n}\frac{\pi(a_i|x_i)}{\pi_0(a_i|x_i)}r(x_i,a_i)\right]\pi_0(a_1|x_1)\ldots\pi_0(a_n|x_n)p(x_1)\ldots p(x_n) =$$

$$= \frac{1}{n}\sum_{x_1,a_1}\pi_0(a_1|x_1)p(x_1)\cdots\sum_{x_n,a_n}\pi_0(a_n|x_n)p(x_n)\left[\sum_{i=1}^{n}\frac{\pi(a_i|x_i)}{\pi_0(a_i|x_i)}r(x_i,a_i)\right]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\sum_{x_1,a_1}\pi_0(a_1|x_1)p(x_1)\cdots\sum_{x_n,a_n}\pi_0(a_n|x_n)p(x_n)\left[\frac{\pi(a_i|x_i)}{\pi_0(a_i|x_i)}r(x_i,a_i)\right]$$

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{x_i,a_i}\pi_0(a_i|x_i)p(x_i)\left[\frac{\pi(a_i|x_i)}{\pi_0(a_i|x_i)}r(x_i,a_i)\right] =$$

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{x_i,a_i}\pi_0(a_i|x_i)p(x_i)r(x_i,a_i) = \frac{1}{n}\sum_{i=1}^{n}V(\pi) = V(\pi)$$

В этом доказательстве предполагается: π – политика, которую мы оцениваем, π_0 – политика на которую мы обуславливаемся (logging policy). Для того, чтобы оценка IPS была несмещенной необходимо, чтобы выполнялось:

$$\pi_0(a) > 0 \quad \forall a : \quad \pi(a) > 0 \tag{8}$$