> Конспект > 8 урок > Многорукие бандиты и обучение с подкреплением

> Оглавление

> Проблема эксплуатации динамического ценообразования

Представим, что:

1. Были созданы две модели динамического ценообразования.
2. Трафик был поделён между ними пополам.
3. Первая модель работает лучше (*допустим*).

Предположение: вывести в эксплуатацию первую модель.

Нюансы:

1. Трафик делили поровну: на модели, работающей хуже, теряем деньги, следовательно, хотим увеличивать долю трафика хорошей модели.
2. Возможно, вторая модель может работать лучше, если подавать ей на вход определённых покупателей в определённый период времени.

Сформулируем **проблему**:

Какую модель предсказания цены выбрать, чтобы увеличить средний доход компании, при условии, что мы не знаем точного дохода для каждой модели в определённый период времени?

Возможное решение:

A/B тесты, но мы будем терять прибыль на неэффективной модели во время теста.

Лучшее решение:

С помощью **многоруких бандитов** выбирать модель и работать напрямую с бизнес-метрикой.

> Многорукие бандиты

Проблемы:

1. N "бандитов" с неизвестным значением среднего выигрыша.
2. Какую машину/ручку выбрать в каждый момент времени?
3. Exploitation/Exploration tradeoff.

Популярные частотные подходы:

1. ϵ-жадный алгоритм (красивые графики, формальный алгоритм);
2. Upper confidence bound (UCB) (доп. источник).

> Байесовские многорукие бандиты

Обозначения:

r^a — случайная величина вознаграждения (*reward*) за действие (*action*) a

$Pr_{r^a}(\theta)$ — неизвестное распределение вознаграждения (θ — параметр).

$R(a) = E(r^a)$ — неизвестное среднее вознаграждение.

Идея:

Выразить неопределённость относительно параметра θ за счёт априорного представления распределения $Pr(\theta)$.

Затем посчитать апостериорное распределение $Pr(\theta|r_1^a, r_2^a, ...r_n^a)$ за счёт наблюдаемых $r_1^a, r_2^a, ...r_n^a$ в ответ на действие a. Для этого будем использовать теорему Байеса:

$$Pr(\theta|r_1^a, r_2^a, ...r_n^a) \propto Pr(\theta) \cdot Pr(r_1^a, r_2^a, ...r_n^a|\theta)$$

Что даёт байесовский подход:

Апостериорное распределение θ позволяет рассчитать:

- Распределение вокруг следующего значения вознаграждения r^a
- Распределение вокруг $R(a)$ при условии, что θ имеет некоторое среднее.

С точки зрения проведения экспериментов и поиска оптимальной стратегии, байесовский подход дает $Pr(R(a)|r_1^a, r_2^a, ...r_n^a)$.

> Пример с монетками

Даны две несимметричные монетки C_1 и C_2.

$$R(C_1) = Pr(C_1 = head)$$

$$R(C_2) = Pr(C_2 = head)$$

Задача: максимизировать количество "орлов" за k бросков.

Нужно решить, какую монетку выбрать.

Так как r^{C_1} и r^{C_2} изменяются в пределах $[0, 1]$, то:

$Pr(r^{C_1}|\theta_1) = \theta_1 = R(C_1)$

$Pr(r^{C_2}|\theta_2) = \theta_2 = R(C_2)$

так как распределение Бернулли параметризовано относительно среднего.

Для упрощения давайте рассмотрим Бета-распределение в качестве априорного $Pr(\theta) = Beta(\alpha, \beta)$.

Для нас очень полезным будет тот факт, что Бета-распределение является сопряжённым к распределению Бернулли, что означает, что после применения теоремы Байеса апостериорное распределение будет также Бета-распределением, но уже с изменёнными параметрами.

Априорное распределение будет выглядеть так:

$Pr(\theta) = Beta(\theta|\alpha, \beta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$, где

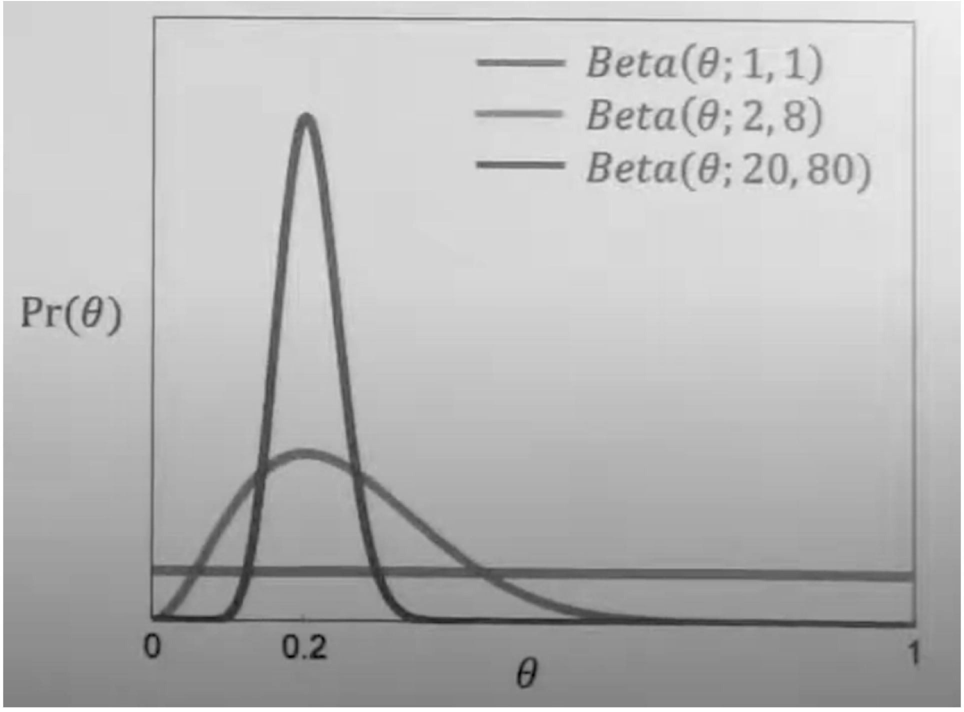
$\alpha - 1$ — количество выпавших "*орлов*"

$\beta - 1$ — количество выпавших "*решек*"

После очередного броска обновляем априорное распределение, получая апостериорное:

$Pr(\theta|C = head) = Beta(\theta|\alpha + 1, \beta)$

$Pr(\theta|C = tail) = Beta(\theta|\alpha, \beta + 1)$



При продолжительном обновлении параметров ($\alpha = 20, \beta = 80$) видим, что пик распределения приходится на $\theta = 0.2$

> Алгоритм семплирования Томпсона

Идея алгоритма:

1. Возьмём подвыборку средних вознаграждений для действия a:
$R_1(a), ...R_k(a)$
$R_j(a) \sim Pr(R_j(a)|r_1^a, r_2^a, ..., r_n^a)$, где n — количество вознаграждений, которое мы получили при a
$r_i^a \sim Pr(r^a|\theta)$

2. Рассчитаем среднее по подвыборке: $\hat{R}(a) = \frac{1}{k}\sum_{i=1}^{k} R_i(a)$, где k — количество сэмплов

3. Найдём такое a^*, где $\hat{R}(a^*)$ максимально: $a^* = \arg\max_a \hat{R}(a)$

4. Выполним a^* и получим r'

5. Обновим $Pr(R(a^*))$ на основе r'

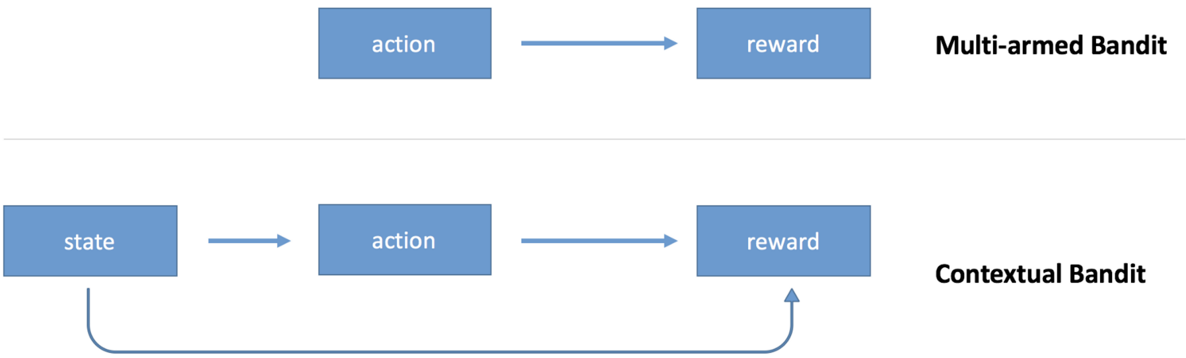
6. Повторим для горизонта наблюдений

Чем больше n, тем уже полученное распределение.

Чем больше k, тем более точная оценка среднего $\hat{R}(a)$.

> Контекстуальные бандиты

Состояние (*state*) — описание среды, которую использует агент.



В нашем случае среда состоит из множества бандитов, каждый из которых имеет несколько рук.

Состояние окружающей среды говорит, с каким бандитом мы взаимодействуем.

Цель — научиться выбирать руку, дающую наибольшее вознаграждение, для любого бандита. Агенту нужно будет научиться обусловливать свои действия состоянием окружающей среды. Если этого не сделать, со временем не будет достигнута максимально возможная награда.

Дано:

- $x^s = (x_1^s, x_2^s, ..., x_n^s)$ — вектор, определяющий пространство шагов
- $x^a = (x_1^a, x_2^a, ..., x_m^a)$ — вектор, определяющий пространство действий
- пространство вознаграждений

Задача:

Найти такое преобразование $x^s \to a$, которое максимизирует ожидаемое вознаграждение $E(r|s,a) = E(r|x^s, x^a)$.

Решение:

Найти приближение средней функции вознаграждения $\widetilde{R}(s,a) = \widetilde{R}(x^s, x^a)$

Подходы:

- $\widetilde{R}_w(x) = w^T x$ — линейный
- $\widetilde{R}_w(x) = neuralNet(x, w)$ — нелинейный

> Семплирование Томпсона для линейной модели

Алгоритм:

Возьмём шаг x^s

Для *каждого* x^a:

1. Возьмём подвыборку средних вознаграждений для каждого действия
2. Рассчитаем среднее в подвыборке: $\hat{R}(a) = \frac{1}{k}\sum_{i=1}^{k} R_i(a)$

Найдём такое a^*, где $\hat{R}(a)$ максимально

Выполним a^* и получим r'

Обновим $Pr(R(x^s, x^a))$ на основе r'

> Резюме

1. Узнали о проблеме эксплуатации динамического образования.
2. Познакомились с байесовскими многорукими бандитами.
3. Поговорили об алгоритме семплирования Томпсона.

Знакомство с контекстуальными бандитами продолжим на практическом занятии.

> Дополнительные материалы

1. CS885 Lecture 8b: Bayesian and Contextual Bandits
2. Многорукие бандиты в рекомендациях
3. CS885 Lecture 8a: Multi-armed bandits
4. Optimism in the Face of Uncertainty: the UCB1 Algorithm