Обучение с подкреплением III

Сергей Николенко

Центр Речевых Технологий, 2012

Центр Речевых Технологий

Outline

Динамическое программирование

- Предположим, что агент действует на протяжении h шагов.
- Используем байесовский подход для определения оптимальной стратегии.
- Начинаем со случайных параметров $\{p_i\}$, например, равномерно распределённых, и вычисляем отображение из *belief states* (состояния после нескольких раундов обучения) в действия.
- Состояние выражается как $\mathcal{S} = \{n_1, w_1, \ldots, n_k, w_k\}$, где каждого бандита i запустили n_i раз и получили w_i единичек (считаем, что результат бинарный).

Динамическое программирование

- $V^*(\mathcal{S})$ — ожидаемый оставшийся выигрыш.
- Рекурсивно: если $\sum_{i=1}^{k} n_i = h$, то больше нечего делать, и $V^*(\mathcal{S}) = 0$.
- Если знаем V^* для всех состояний, когда осталось t запусков, сможем пересчитать и для $t+1$:

$$V^*(n_1, w_1, \ldots, n_k, w_k) =$$
$$= \max_i \left(\rho_i(1 + V^*(\ldots, n_i + 1, w_i + 1, \ldots)) + \right.$$
$$\left. (1 - \rho_i)\, V^*(\ldots, n_i + 1, w_i, \ldots)\right),$$

где ρ_i — апостериорные вероятности того, что действие i оправдается (если изначально p_i равномерно распределены, то $\rho_i = \frac{w_i+1}{n_i+2}$).

Байесовский подход к многоруким бандитам

- А теперь давайте посмотрим на многоруких бандитов в общем вероятностном виде.
- Для простоты – бинарный случай, выплата либо 1, либо 0.

Байесовский подход к многоруким бандитам

- Пусть во время t у нас состояние $\mathbf{s}_t = (s_{1t}, \ldots, s_{Kt})$ для K ручек, и мы хотим дёрнуть такую ручку, чтобы максимизировать общее ожидаемое число успехов.
- Есть функция вознаграждения $R_i(\mathbf{s}_t, \mathbf{s}_{t+1})$ – награда за дёргание ручки i (a_i), которое переводит состояние $\mathbf{s}_t$ в $\mathbf{s}_{t+1}$.
- Есть вероятность перехода $p\left(\mathbf{s}_{t+1} \mid \mathbf{s}_t, a_i\right)$.
- И мы хотим обучить стратегию $\pi(\mathbf{s}_t)$, которая возвращает, какую ручку дёргать.

Байесовский подход к многоруким бандитам

- Тогда value function в самом общем виде до горизонта T:

$$V_T(\pi, \mathbf{s}_0) = \mathrm{E}\left[R_{\pi(\mathbf{s}_0)}(\mathbf{s}_0, \mathbf{s}_1) + V_{T-1}(\pi, \mathbf{s}_1)\right] =$$
$$= \int p\left(\mathbf{s}_1 \mid \mathbf{s}_0, \pi(\mathbf{s}_0)\right)\left[R_{\pi(\mathbf{s}_0)}(\mathbf{s}_0, \mathbf{s}_1) + V_{T-1}(\pi, \mathbf{s}_1)\right] d\mathbf{s}_1.$$

- Если всё известно, и T невелико, то можно, опять же, динамическим программированием.
- Но даже подсчитать отдачу от фиксированной стратегии может быть очень дорого, не говоря уж об оптимизации.

Байесовский подход к многоруким бандитам

- Если T большой/неограниченный, логично рассмотреть

$$R = R(0) + \gamma R(1) + \gamma^2 R(2) + \ldots, \quad 0 < \gamma < 1.$$

- Теорема Гиттинса (1979): задачу поиска оптимальной стратегии

$$\pi(\mathbf{s}_t) = \arg\max_\pi V(\pi, \mathbf{s}_t = (s_{1t}, \ldots, s_{Kt}))$$

можно факторизовать и свести к

$$\pi(\mathbf{s}_t) = \arg\max_i \gamma(s_{it}).$$

- $\gamma(s_{it})$ – *индекс Гиттинса*; но его подсчитать тоже обычно трудно; есть приближения.

Оценки на regret

- Другой вариант – давайте рассчитаем приоритет каждой ручке i так, чтобы непосредственно regret ограничить.
- [Auer et al., 2002]: стратегия $\mathrm{UCB1}$. Учитывает неопределённость, «оставшуюся» в той или иной ручке, старается ограничить regret. Если мы из n экспериментов n_i раз дёрнули за i-ю ручку и получили среднюю награду $\hat{\mu}_i$, алгоритм $\mathrm{UCB1}$ присваивает ей приоритет

$$\mathrm{Priority}_i = \hat{\mu}_i + \sqrt{\frac{2 \log n}{n_i}}.$$

Дёргать дальше надо за ручку с наивысшим приоритетом.

Оценки на regret

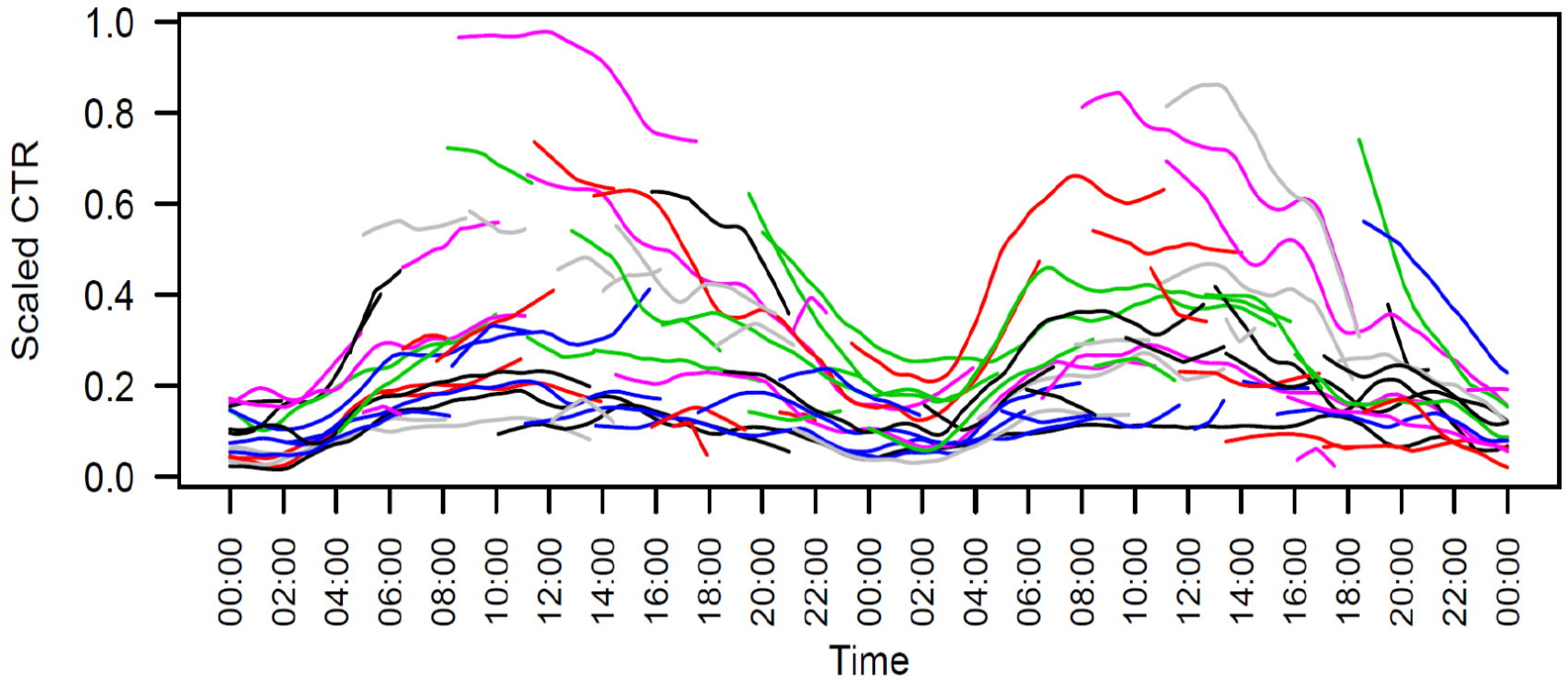
- При таком подходе субоптимальные ручки будут дёргать $O(\log n)$ раз, и regret будет $O(\log n)$, а меньше и нельзя (но тут константы тоже важны :)).
- $\mathrm{UCB1}$ – хорошая стратегия, но рассчитывает на оценку в худшем случае, может быть неоптимально.

Центр Речевых Технологий

Outline

Пример: клики на странице новостей

- Пример:



Пример: клики на странице новостей

- Мы в момент t должны распределить доли показов $(x_1, x_2, \ldots, x_K)$ так, чтобы оптимизировать CTR.
- Совсем простая ситуация: два момента времени, $t = 0$ и $t = 1$, выбор из двух объектов:
  - объект P имеет CTR p_0 в момент $t = 0$ и p_1 в момент $t = 1$, мы на этот счёт не уверены, есть распределение какое-то;
  - про объект Q всё знаем точно, q_0 и q_1.
- Надо найти x – долю показов P в момент $t = 0$; у нас есть N_0 показов для распределения в $t = 0$ и N_1 в $t = 1$.

Пример: клики на странице новостей

- Пусть мы получили c кликов после того как выбрали x; c – случайная величина.
- Мы пронаблюдаем c, и на втором этапе оптимальное решение будет понятно: дать все N_1 кликов P iff

$$\hat{p}_1(x, c) = \mathbb{E}\left[p_1 \mid x, c\right] > q_1.$$

- Т.е. мы должны оптимизировать x с точки зрения общего ожидаемого числа кликов, учитывая, что что-то новое мы узнаем про p_1 к моменту $t = 1$.

Центр Речевых Технологий

Пример: клики на странице новостей

- Ожидаемое число кликов:

$$N_0 x \hat{p}_0 + N_0 (1 - x) q_0 + N_1 \mathrm{E}_c \left[\max\{\hat{p}_1(x, c), q_1\} \right] =$$
$$= N_0 q_0 + N_1 q_1 + N_0 x (\hat{p}_0 - q_0) + N_1 \mathrm{E}_c \left[\max\{\hat{p}_1(x, c) - q_1, 0\} \right].$$

- Второе слагаемое – это и есть выгода от исследования P:

$$\mathrm{Gain}(x, q_0, q_1) = N_0 x (\hat{p}_0 - q_0) + N_1 \mathrm{E}_c \left[\max\{\hat{p}_1(x, c) - q_1, 0\} \right],$$

его мы и оптимизируем по x.

Пример: клики на странице новостей

- Если приблизить $\hat{p}_1(x, c)$ нормальным распределением (разумно по ЦПТ):

$$\text{Gain}(x, q_0, q_1) = N_0 x(\hat{p}_0 - q_0) + N_1 \left[\sigma_1(x)\Phi\left(\frac{q_1 - \hat{p}_1}{\sigma_1(x)}\right) + \right.$$
$$\left. + \left(1 - \Phi\left(\frac{q_1 - \hat{p}_1}{\sigma_1(x)}\right)\right)(\hat{p}_1 - q_1)\right],$$

$$p_1 \sim \text{Beta}(a, b) \text{ (априорное распределение)},$$

$$\hat{p}_1 = \text{E}_c\left[\hat{p}_1(x, c)\right] = \frac{a}{a + b},$$

$$\sigma_1^2(x) = \text{Var}\left[\hat{p}_1(x, c)\right] = \frac{xN_0}{a + b + xN_0} \frac{ab}{(a + b)^2(1 + a + b)}$$

Центр Речевых Технологий

Пример: клики на странице новостей

- Для нескольких вариантов ($K > 2$) задача становится гораздо сложнее; её можно несколько ослабить и свести к K независимым задачам с двумя вариантами.
- А что для нескольких временных слотов ($T > 1$)?

Центр Речевых Технологий

DGP

- Модель Dynamic Gamma–Poisson (DGP): фиксируем период времени t (небольшой) и будем считать показы и клики за время t.
- Пусть мы в течение периода t показали продукт n_t раз и получили суммарный рейтинг r_t (если это ссылки на странице, например, то будет суммарное число кликов $r_t \leq n_t$).
- Тогда нам в каждый момент t дана последовательность $n_1, r_1, n_2, r_2, \ldots, n_t, r_t$, и мы хотим предсказать p_{t+1} (доля успешных показов в момент $t+1$, CTR).

Центр Речевых Технологий

DGP

- Вероятностные предположения модели DGP:
  1. $(r_t \mid n_t, p_t) \sim \text{Poisson}(n_t p_t)$ (для данного n_t и p_t, r_t распределено по пуассоновскому распределению).
  2. $p_t = \epsilon_t p_{t-1}$, где $\epsilon_t \sim \text{Gamma}(\mu = 1, \sigma = \eta)$ (средняя доля успешных показов p_t меняется не слишком быстро, а путём умножения на случайную величину ϵ_t, которая имеет гамма-распределение вокруг единицы).
  3. Параметрами модели являются параметры распределения $p_1 \sim \text{Gamma}(\mu = \mu_0, \sigma = \sigma_0)$, а также параметр η, который показывает, насколько «гладко» может изменяться p_t.
  4. Соответственно, задача заключается в том, чтобы оценить параметры апостериорного распределения

$$(p_{t+1} \mid n_1, r_1, n_2, r_2, \ldots, n_t, r_t) \sim \text{Gamma}(\mu = ?, \sigma = ?).$$

Центр Речевых Технологий

DGP

- Можно пересчёт параметров в этой модели явно вычислить аналитически.
- Пусть на предыдущем шаге $t-1$ мы получили некоторую оценку μ_t, σ_t для параметров модели:

$$(p_t \mid n_1, r_1, n_2, r_2, \ldots, n_{t-1}, r_{t-1}) \sim \mathrm{Gamma}(\mu = \mu_t, \sigma = \sigma_t),$$

  а затем получили новую точку (n_t, r_t).
- Тогда, обозначив $\gamma_t = \frac{\mu_t}{\sigma_t^2}$ (эффективный размер выборки), сначала уточним оценки μ_t, σ_t:

$$\gamma_{t|t} = \gamma_t + n_t,$$
$$\mu_{t|t} = \frac{\mu_t \gamma_t + r_t}{\gamma_{t|t}},$$
$$\sigma^2_{t|t} = \frac{\mu_{t|t}}{\gamma_{t|t}}.$$

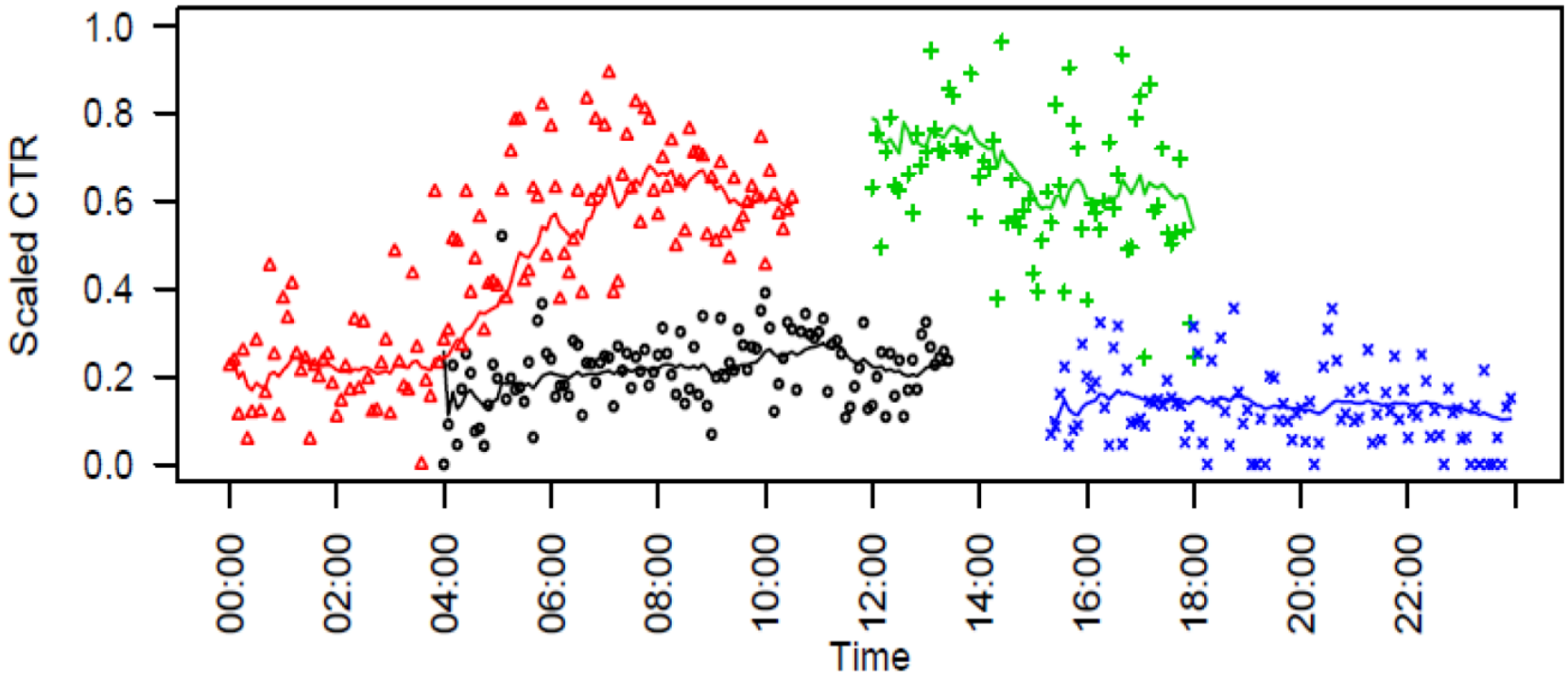
Центр Речевых Технологий

DGP

- А затем породим новое предсказание для $(p_{t+1} \mid n_1, r_1, \ldots, n_t, r_t)$:

$$\mu_{t+1} = \mu_{t|t},$$
$$\sigma^2_{t+1} = \sigma^2_{t|t} + \eta\left(\mu^2_{t|t} + \sigma^2_{t|t}\right).$$

Пример


Scaled CTR
1.0
0.8
0.6
0.4
0.2
0.0
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00
Time

Thank you!

Спасибо за внимание!