Онлайн-аукционы

С. Николенко [*]

19 мая 2008 г.

Содержание

1 Постановка задачи

Вспомним задачу, которую мы решали на прошлой лекции. Тогда продавец продавал несколько копий одной вещи и N агентов хотели купить эту вещь. У агентов были свои цены x_i. При этом вещей было столько, что мы могли продать их хоть каждому агенту. Агенты называли свои ставки, и продавец предлагал им товар по некоторой своей цене.

Теперь модифицируем задачу. Агенты приходят по очереди по одному, чтобы купить вещь. Каждому прибывшему агенту мы предлагаем цену, по которой он может приобрести товар. То есть на шаге i приходит агент со ставкой b_i, и механизм решает, продать ли ему (бит d_i) и по какой цене p_i.

Как и раньше, суть в том, чтобы определять минимальную выигрывающую ставку t_i: минимальную ставку, при которой агент i выигрывает.

[*]Законспектировала О. Канжелева.

Напомним, что аукцион называют правдивым, если оптимальная стратегия любого агента - говорить правду. Тогда для правдивости нашего аукциона нужно, чтобы t_i не зависели от b_i. Для онлайновости нужно, чтобы цена (t_i) не зависела от ставок агентов $(b_{i+1}, b_{i+2} \ldots)$, которые придут позже. То есть это функция $t_i(b_1, \ldots, b_{i-1})$.

Наша цель — приблизиться к бенчмарке. На этот раз мы приближаемся к бенчмарке $\mathcal{F}(\mathbf{b})$ — прибыли от продажи вещей по одной оптимальной цене:

$$\mathcal{F}(\mathbf{x}) = \max_p p \times \{\text{к-во участников с } x_i \geq p\}.$$

И, как и прежде, будем пытаться делать сколько-нибудь-оптимальные алгоритмы.

Таким образом перед нами стоит задача:

- ставки из $[1, h]$;
- n агентов;
- на раунде i механизм
  1. выбирает цену предложения t_i,
  2. узнаёт b_i,
  3. получает $p_i = t_i$, если $t_i \leq b_i$, и 0 в противном случае;
- цель — приблизиться к $\mathcal{F}(\mathbf{b})$.

2 Machine Learning: expert advice

2.1 Формулировка задачи

В machine learning есть такая задача.

- k экспертов;
- n итераций;
- на раунде i алгоритм
  1. выбирает эксперта j,
  2. узнаёт payoffs $q_1^{(i)}, \ldots, q_k^{(i)}$,
  3. получает $q_j^{(i)}$;
- цель — приблизиться к оптимальному эксперту, к $\max_j \sum_{i=1}^n q_j^{(i)}$.

Пример 1 *Пусть у нас есть k предсказателей погоды. Каждый вечер мы будем спрашивать всех предсказателей прогноз погоды на следующий день. Из всех прогнозов выберем один, которому поверим. На следующий день мы получаем или не получаем прибыль в зависимости от точности прогноза, которому поверили. При этом мы узнаем насколько точны были прогнозы остальных предсказателей. Наша цель — максимально приблизиться к лучшему из предсказателей.*

Можно свести задачу онлайн-аукциона к задаче expert learning. Пусть у нас есть эксперты, и эксперт j на каждом шаге предлагает установить цену 2^j. Его payoff на шаге i тогда получается 2^j, когда $2^j \le b_i$, и 0 в противном случае. Пусть ставки лежат в промежутке $[0, h]$. Тогда количество экспертов $k = \log h$. Тогда получается соответствие между лучшим экспертом и лучшей ценой из тех, которые являются степенями двойки. То есть оптимальный алгоритм expert learning даст 2-оптимальный онлайн-аукцион. Разумеется, 2 можно заменить на $1+\epsilon$, а экспертов всё равно будет $O(\log h)$. Но мы в дальнейшем будем просто предполагать, что константа равна 2.

2.2 Follow the leader

Пусть общим доходом $s_j^{(i)}$ эксперта j, при условии, что уже приходило i агентов, будет сумма, которую он бы получил, предлагая цену 2^j на каждом шаге. То есть

$$s_j^{(i)} = \sum_{i'=1}^{i} q_j^{(i')}.$$

На каждом следующем раунде будем выбирать того эксперта, у которого самый большой общий доход. То есть на раунде i выберем эксперта

$$j = \mathrm{argmax}_j s_j^{(i-1)}.$$

Однако, можно привести пример, на котором алгоритм работает плохо.

Пример 2 *Есть два эксперта, доходы которых распределены так, как показано ниже.*

	1	2	3	4	...
Expert 1:	$\frac{1}{2}$	0	1	0	...
Expert 2:	0	1	0	1	...

Очевидно, что при таких данных, ничего не получится. Идею надо усовершенствовать.

2.3 Weighted Majority

Как и в алгоритме Follow the leader для каждого эксперта будем считать его общий доход $s_j^{(i)}$. Пусть доходы экспертов лежат в $[0, h]$. Теперь в раунде i будем выбирать эксперта j с вероятностью пропорциональной

$$2^{\frac{s_j^{(i-1)}}{h}}.$$

Приведем без доказательства теорему и следствие.

Теорема 1 *Для доходов экспертов в* $[0, h]$ *ожидаемый доход Weighted Majority*

$$\mathbf{E}[\text{Payoff}] \geq \frac{\text{Opt}}{2} - O(\frac{h}{k}\log k).$$

Где $\text{Opt} = \max_p p \times \{$*к-во участников с* $x_i \geq p\}$.

Следствие 1 *Алгоритм Weighted Majority для онлайн-аукционов позволяет достичь*

$$\mathbf{E}[\text{Profit}] \geq \frac{\mathcal{F}}{4} - O(h \log\log h).$$

3 Kalai's Online Learning Algorithm

3.1 Формулировка задачи

Предположим, что доходы экспертов лежат в $[0, h]$. Общий доход эксперта $s_j^{(i)}$ складывается из его реального дохода ($\sum_{i'=1}^{i} q_j^{(i')}$) и дохода-галлюцинации.

Для каждого эксперта j будем считать доходы-галлюцинации

$$s_j^{(0)} = h \times \{\text{к-во орлов подряд}\}.$$

То есть будем бросать монетку, пока не выпадет решка. Количество выпавших орлов умножим на h и получим доход-галлюцинацию. Как и прежде, на раунде i будем выбирать эксперта с максимальным доходом, то есть эксперта

$$j = \text{argmax}_{j'}(s_{j'}^{(0)} + s_{j'}^{(i-1)}).$$

3.2 Теорема

Теорема 2

$$\textit{Для алгоритма Kalai } \mathbf{E}[\text{Payoff}] \geq \frac{\text{Opt}}{2} - O(h \log k).$$

План доказательства. Для доказательства рассмотрим гипотетический алгоритм be-the-leader. Он отличается от follow-the-leader лишь тем, что на шаге i мы выбираем эксперта $j = \text{argmax}_{j'}(s_{j'}^{(0)} + s_{j'}^{(i)})$. Для be-the-leader докажем, что

$$\mathbf{E}[\text{Payoff}] \geq \text{Opt} - O(h \log k).$$

Потом докажем, что follow-the-leader не более чем вдвое хуже, чем be-the-leader.

Доказательство. Проведем доказательство для be-the-leader. Обозначим H_i доход лучшего эксперта на шаге i. То есть

$$H_0 = \max_j s_j^{(0)}; H_n = \max_j(s_j^{(0)} + s_j^{(n)}).$$

На шаге i be-the-leader обязательно получит не меньше $H_i - H_{i-1}$. В противном случае, H_i или $H_i - 1$ не были бы доходами лучших экспертов на шагах i или $i-1$ соответственно. Тогда общий доход будет $H_n - H_0$.

$$\text{Так как } H_n \geq \mathrm{Opt}, \text{ то и } \mathbf{E}[H_n] \geq \mathrm{Opt}.$$

H_0 — это максимум из h умножить на k геометрических случайных величин (количества орлов).

Упражнение 1 *Этот максимум равен* $O(\log k)$.

Бросим монеты одновременно для k экспертов. По теории вероятности, в половине экспериментов выпадет решка. Подбросим монеты для оставшихся $\frac{k}{2}$ экспертов. Из них только у половины выпадет орел, поэтому только $\frac{k}{4}$ продолжат участие в экспериментах. Эксперимент остановится на $\log k$ *шаге, когда останется один эксперт.*

Таким образом, ожидаемый доход алгоритма be-the-leader $\mathrm{Opt} - O(h \log k)$.

Докажем, что follow-the-leader не более чем вдвое хуже. Для этого докажем, что вероятность того, что лидер останется прежним, в каждом раунде по крайней мере $\frac{1}{2}$.

Рассмотрим раунд i. Будем сначала рассматривать доход экспертов без галлюцинаций, а потом добавим их определенным образом. Выберем эксперта с наименьшей суммой,

$$j = \mathrm{argmin}_{j'}(s_{j'}^{(0)} + s_{j'}^{(i)}).$$

Подбросим монетку. Если выпадет орел, то добавим ещё h к галлюцинации эксперта

$$j : s_j^{(0)} \leftarrow s_j^{(0)} + h.$$

Если выпадет решка, отбросим вообще этого эксперта. Его галлюцинация останется равной $s_j^{(0)}$, и общей суммы точно не хватит, чтобы стать лучшим на этом шаге. Далее из оставшихся экспертов снова выберем эксперта с наименьшей суммой и подбросим монетку для него. Будем продолжать эти действия, пока не останется один эксперт. При этом у лидера еще останется монетка в запасе.

Если она выпадет орлом, то эксперт, получается, будет лидировать более чем на h. Поэтому, даже если отнять от него его текущий раунд $q_j^{(i)}$, он всё равно будет лидировать (так как $q_j^{(i)} \leq h$). Значит, он и на предыдущем раунде был лидером. Таким образом лидер останется прежним с вероятностью по крайней мере $\frac{1}{2}$. А это значит, что follow-the-leader не более чем вдвое хуже be-the-leader. Таким образом теорема доказана.

3.3 Non-uniform алгоритм

Когда мы используем экспертное обучение для онлайн-алгоритмов, мы знаем, какие там выплаты, а именно 2^j. То есть мы можем уверенно говорить, что выплата эксперта j ограничена $h_j = 2^j$. Поэтому теперь будем считать, что доходы эксперта j лежат в $[0, h_j]$. Как и прежде общий доход эксперта $s_j^{(i)}$ складывается из его реального дохода ($\sum_{i'=1}^{i} q_j^{(i')}$) и дохода-галлюцинации.

Доход-галлюцинация $s_j^{(0)} = h \times \{$к-во орлов подряд$\}$. На раунде i будем выбирать эксперта с максимальным доходом, то есть эксперта

$$j = \mathrm{argmax}_{j'}(s_{j'}^{(0)} + s_{j'}^{(i-1)}).$$

Теорема 3

$$\textit{Для данного алгоритма } \mathbf{E}[\mathrm{Payoff}] \geq \frac{\mathrm{Opt}}{2} - \frac{1}{2}\sum_j h_j.$$

Доказательство проводится аналогично доказательству теоремы для алгоритма Kalai. Сначала докажем, что be-the-leader работает как $\mathrm{Opt} - \sum_j h_j$. Общий доход будет $H_n - H_0$, где $\mathbf{E}[H_n] \geq \mathrm{Opt}$ (см. доказательство теоремы Калаи). Осталось оценить H_0. Ожидаемый максимум галлюцинаций уж точно меньше, чем сумма всех галлюцинаций, то есть $\mathbf{E}[H_0] \geq \sum_j h_j$. Аналогично доказательству теоремы Kalai докажем, что follow-the-leader вдвое хуже be-the-leader.

4 Другие online аукционы

4.1 Online posted price mechanisms

Отличие этого аукциона от нами рассмторенного в том, что в каждом раунде продавец выставляет цену, но ничего не узнаёт о ставке покупателя.

То есть на каждом шаге приходит агент, чтобы купить вещь. Каждому прибывшему агенту мы предлагаем цену, по которой он может приобрести товар. А агент либо соглашается и покупает товар, либо отказывается и уходит.

4.2 Limited supply online auctions

Этот аукцион отличается лишь тем, что у продавца ограничено количество товара. Поэтому при продаже ему надо учитывать, что, если он быстро продаст весь товар, то потом могут придти агенты с более высокими ставками, которым уже будет нечего продать.