Multi-armed bandits

Part 3

Expected value

Expected value is a central concept in statistics and decision theory.

Let us toss a coin. We assume it is fair, i.e.

prob(Head) = prob(Tail) = 0.5

If we get Head, we receive 1€, while Tail is worth 0€.

The expected value of a single round of this game is worth

$$prob(Head) \times value(Head) + prob(Tail) \times value(Tail)$$
$$= 0.5 \times 1.00€ + 0.5 \times 0.00€ = 0.50€$$

perfectly according to intuition.

On the average, we will get 0.50€ per round.

[Note: the expected value formula is the same of probability of observation, the denominator in Bayes' formula. Indeed, they are the same concept]

In general, we get a decision *d*, say betting on a coin tossing:

- we have *n* possible outcomes $a_1, \ldots, a_n$
- each one with its own probability to happen $p(a_i)$
- and its own value, if happening, $v(a_i)$

The expected value of this decision is

$$EV(d) = \sum_{i=1}^{n} p(a_i)v(a_i)$$

If we have two possible decisions *d1* and *d2*, we will choose the one with the greatest expected value. The same if we have many decisions to choose among.

Expected value is the average value of possible outcomes, each one weighted with its probability.

Expected value of impressions

Until now, we assumed each advertisement in a MAB problem was remunerated with 1€ per click.

So, maximizing revenue and maximizing clicks number were the same thing.

Given that the number of impressions was independent on our choices, our problem was to maximize the CTR, the clickthrough rate.

Now we cope a more complex formulation of our problem:

Given a number of impression T, a set of advertisements $a_1, \ldots, a_n$ and a their monetary values $v_1, \ldots, v_n$, design a policy for selecting an ad at each round form 1 to T in order to maximize the overall *expected* revenue.

The word *expected* underlines the probabilistic nature of our problem.

We apply the concept of expected value.

For each ad a_i we estimate its probability to be clicked, i.e. its expected CTR.

Estimating (or forecasting, it is the same) an ad's CTR is the problem we coped with using Naïve Bayes and Logistic Regression.

Assuming we are able to solve this problem in a reliable way, we compute a weighted average of ads' values, where the weights are the estimated CTRs.

The expected value of the decision to deliver a certain ad in this round, let us call it shortly the expected revenue of an ad, is

$$EV(a_i) = \sum_{i=1}^{n} p(a_i)v(a_i)$$

We use the same MAB theory we already know, this time using the expected revenue instead of the expected clickthrough rate.

Ad	Revenue per click	Imps	Clicks	Forecasted CTR	Expected value
A	1.00€	100	5	0.05	0.05€
B	2.00€	200	6	0.03	0.06€
C	3.00€	300	3	0.01	0.03€

The forecasted CTR is computed in a simplistic way, simply taking the observed CTR. It is enough for the example, remember we can do better.

The "best" ad is *B*, though it does not have nether the best price nor the best CTR.

We can review the MAB policies taking the expected value as the criterion instead of the CTR.

Ad	Revenue per click	Imps	Clicks	Forecasted CTR	Expected value
A	1.00€	100	5	0.05	0.05€
B	2.00€	200	6	0.03	0.06€
C	3.00€	300	3	0.01	0.03€

	Score is the CTR	Score is the EV
Epsilon-Greedy	In the greedy phase, select the max CTR	In the greedy phase, select the max EV
Softmax	In exp(Q / T) the quality Q is the CTR	In exp(Q / T) the quality Q is the EV
Thompson Sampling	For each ad, sample x from Beta(trials, hits) and select max x	For each ad, sample x from Beta(trials, hits) and select max (x * Revenue)

Expected conversion rate

Until now, we were interested in clicks.

Now, we consider *conversions*.

When the user clicks on an advertisement, he/she jumps on the advertiser's site. There, she can purchase something, download a content, leave her personal data or do any other useful action (useful for the advertiser).

A conversion happened.

A click is only a particular form of conversion. The most interesting form is *sales*.

The treatment of conversions is similar to the treatment of clicks in many respects, but some important differences hold.

Ad	Imp	Click	Conv	CTR	CCR	CVR
A	100	5	1	5%	20%	1%
B	200	6	2	3%	33%	1%
C	300	3	2	1%	66%	0.66%

CTR = clicks / impressions

CCR = conversions / clicks (click conversion rate)

CVR = conversions / impressions (impression conversion rate)

The terminology is a bit confused.

We can estimate CVR with the same methods already seen, or we can separately estimate CCR and CVR then multiply them.

In the example, the outcome is the same because estimate are simplistically observed quantities.

If we use estimation moving information through cells then the two methods are different.

In general, decomposing CVR = CTR x CCR is preferable.

Ad	Imp	Click	Conv	CTR	CCR	CVR
A	100	5	1	5%	20%	1%
B	200	6	2	3%	33%	1%
C	300	3	2	1%	66%	0.66%

Conversions are in general rare events.

It is likely to have CTR 1% and CCR 1% again, which means CVR 0.01% i.e. one conversion per 10,000 impressions.

This require attention in the statistical treatment.

Simple methods can be sustainable when estimating or forecasting CTR, but seldom when applied to CVR.

CVR requires methods like Naïve Bayes or Logistic Regression.

The need to "fill more data" in cells is much more serious when treating conversions.

Expected conversion value

Conversions are very different from clicks when they have an associated monetary value which is not fixed in advance.

In that case, we cope an additional problem: forecasting conversion values.

In e-commerce, once the user has clicked on our advertisement and reached our site, he can buy or not. If buying, he can spend any amount of money, maybe 10€ maybe 1000€.

Forecasting the value of sales is very different then estimating CTR or CVR.

The point is that sales get *continuous* values, not only *binary* ones.

In order to compute the expected value of impressions, we now have to predict the worth of a conversion:

How much money can we expect this user will spend on our site, once he/she has clicked and entered the site?

If we are able to predict

v_i = expected value of conversion of this user on the site of advertiser *I*

p_i = probability of click on a_i (i.e. CTR of a_i)

Then we can compute

Expected value of selecting a_i = $EV(a_i) = p_i \times v_i$

This is what we need in order to apply MAB algorithms.