Budget Allocator (Optimizer)

If first day (t=0): initialize n_i and v_i to balanced values Otherwise: update $n_j(.)$ and v_j and update the total regret as $P = P + P_{1,t} + P_{2,t} + P_{3,t}$

New day begin: t= t+1

Calculate the best budget for the 3 subcampaign for day t, solving the knapsack problem

Budget for subcampaign j on day t $(y_{j,t})$

Subcampaign j

Advertising

Update the GP for the current day in the classical way $(n_i(.))$

Retrieve from the real curve the number of click, given the budget above $N_j(y_{j,t})$

> Number of clicks of the current day $N_i(y_{i,t})$

Pricing

Update the TS for the current context in the classical way, given the daily number of clicks above

> Extract the daily reward $(r_{j,t})$

Calculate the daily regret as: $P_{i,t} = (N_i(Y_{i,t}) * R_{i,t}) - (N_i(y_{i,t}) * r_{i,t})$

Prepare the value of n_i(.) and v_i for the update of the budget allocator (v_i is derived from the cumulative reward of this subcampaign)

Legend:

t = current day

Ni(.) = Real distribution function for advertising

 $n_i(.)$ = Learned (from GP) distribution function for advertising

Y_{i.t} = Real best budget allocation

y_{i,t} = Learned best budget allocation for the budget allocator

R_{j,t} = Best daily reward

 $r_{i,t}$ = Learned daily reward

v_i = Value per click

same for the other subcampaigns

New Knapsack problem to solve:

$$\max_{y_{j,t}} \sum_{j=1}^{N} v_j \ n_j(y_{j,t})$$
s.t.
$$\sum_{j=1}^{N} y_{j,t} \le \overline{y}_t$$

s.t.
$$\sum_{j=1}^{N} y_{j,t} \leq \overline{y}_t$$