Pricing & Advertising

Off-brand Hand Sanitizer

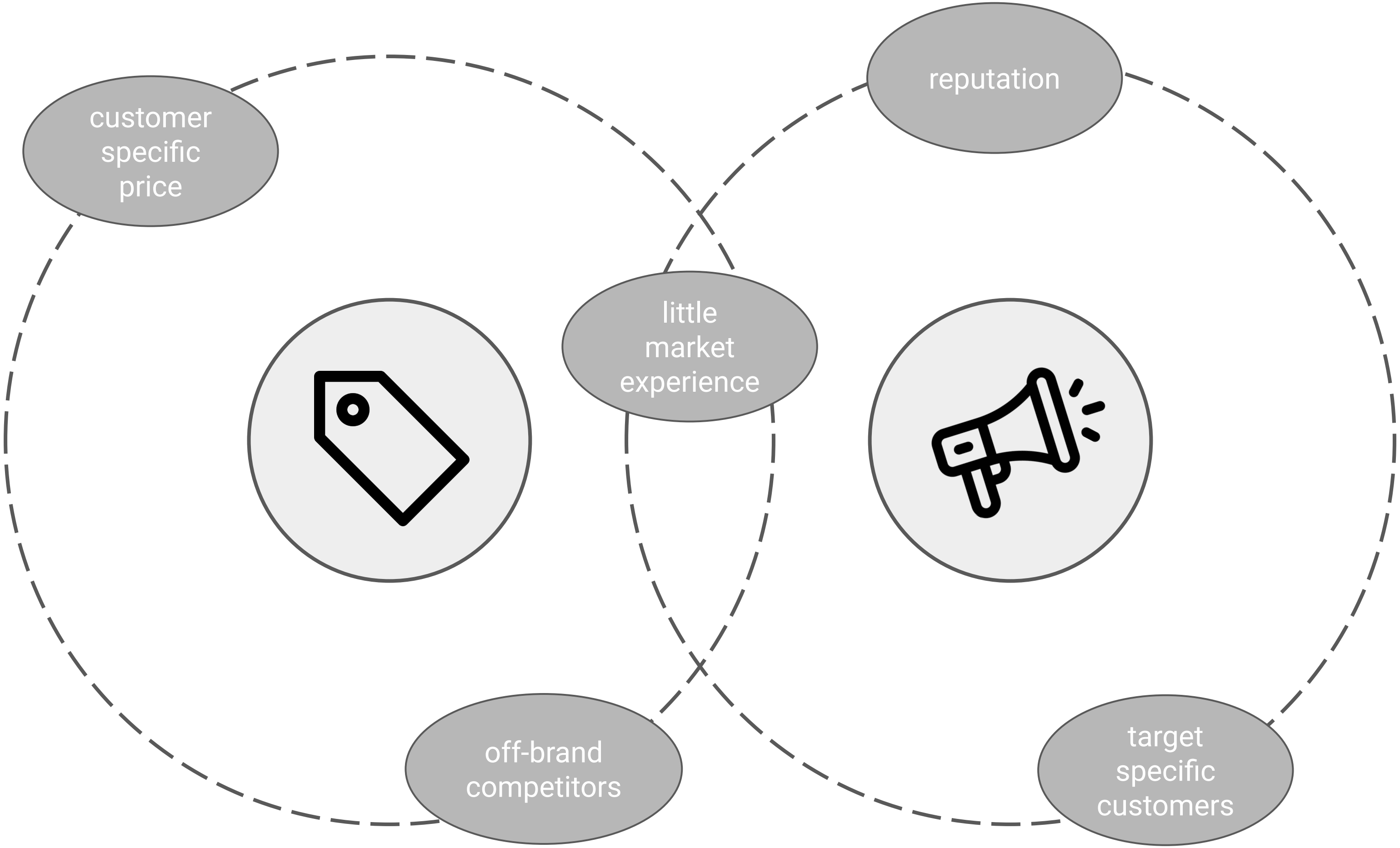
Luca Alessandrelli
Jacopo Pio Gargano
Riccardo Poiani
Tang-Tang Zhou

Data Intelligence Applications

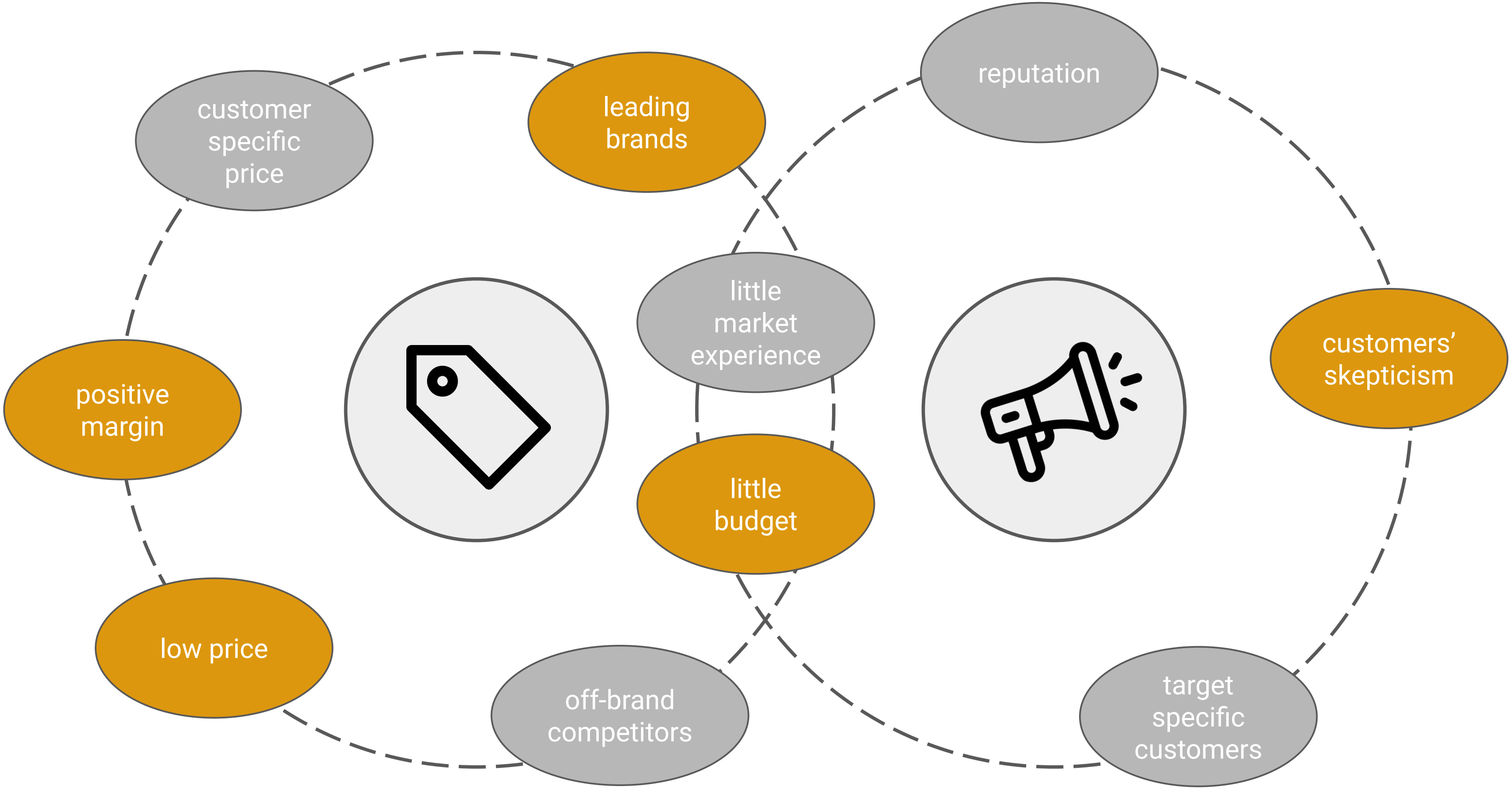
Politecnico di Milano

July 1st, 2020

// Pricing and Advertising issues



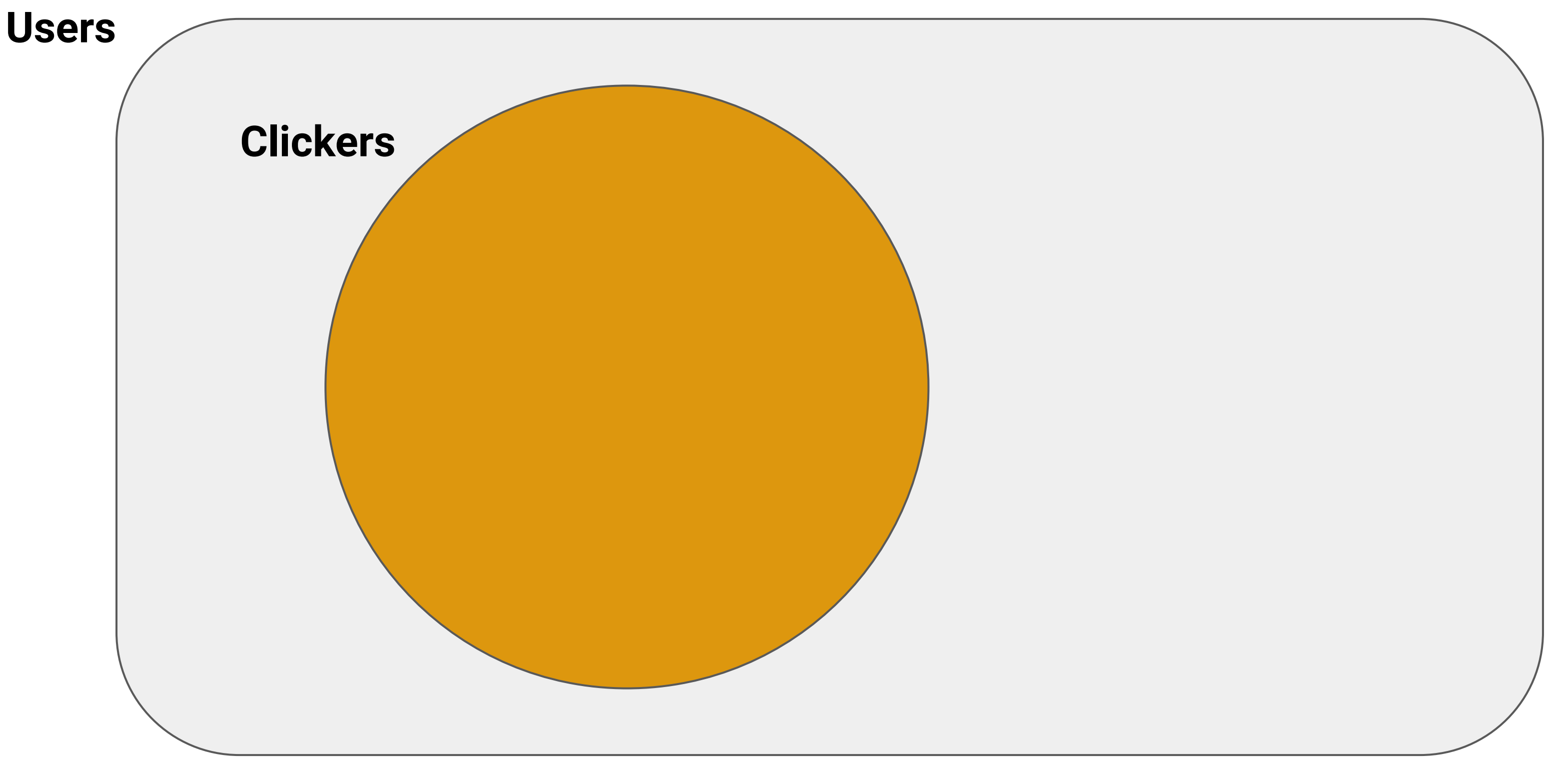
// Pricing and Advertising issues



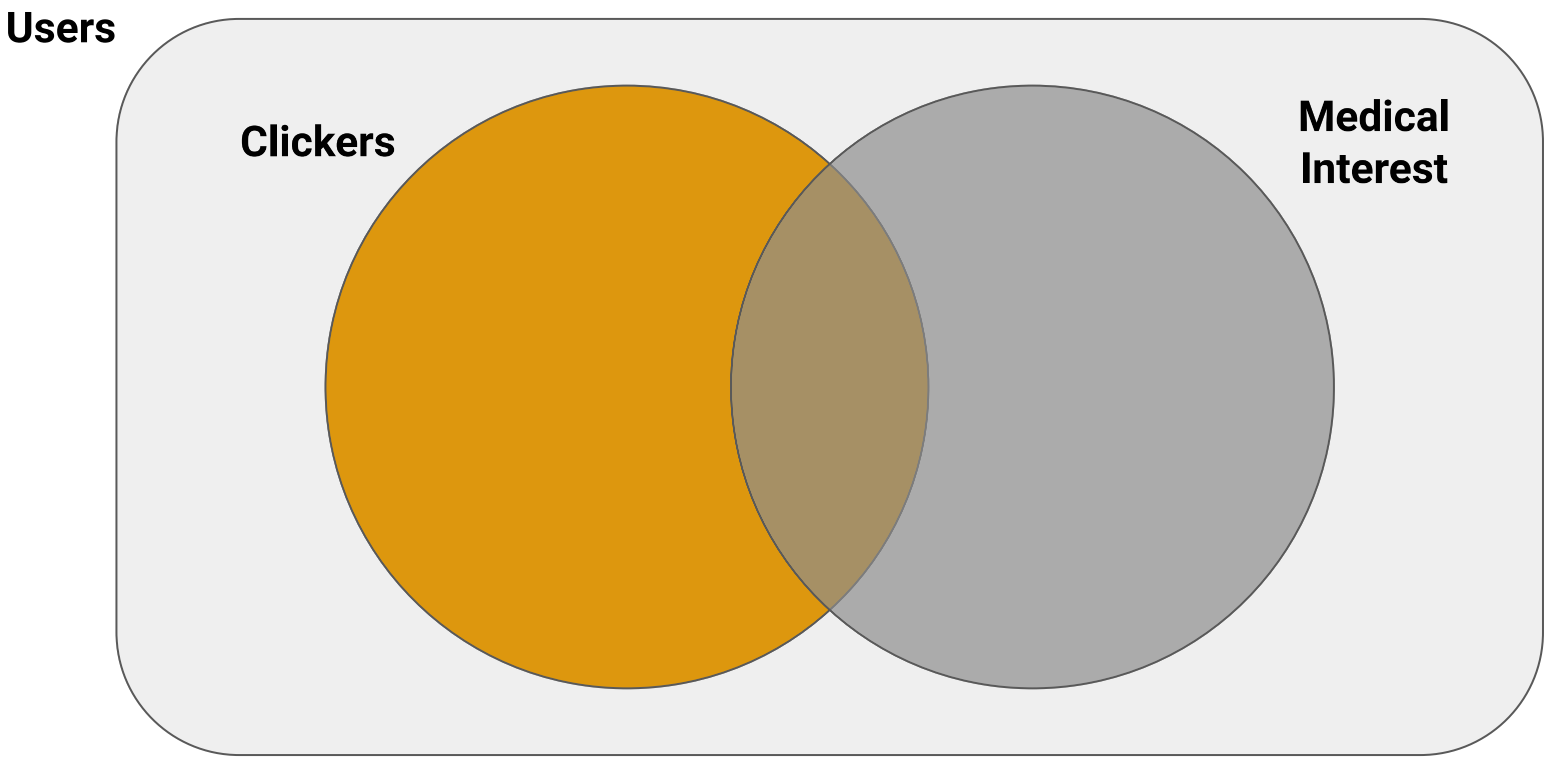
// Users Contexts

Users

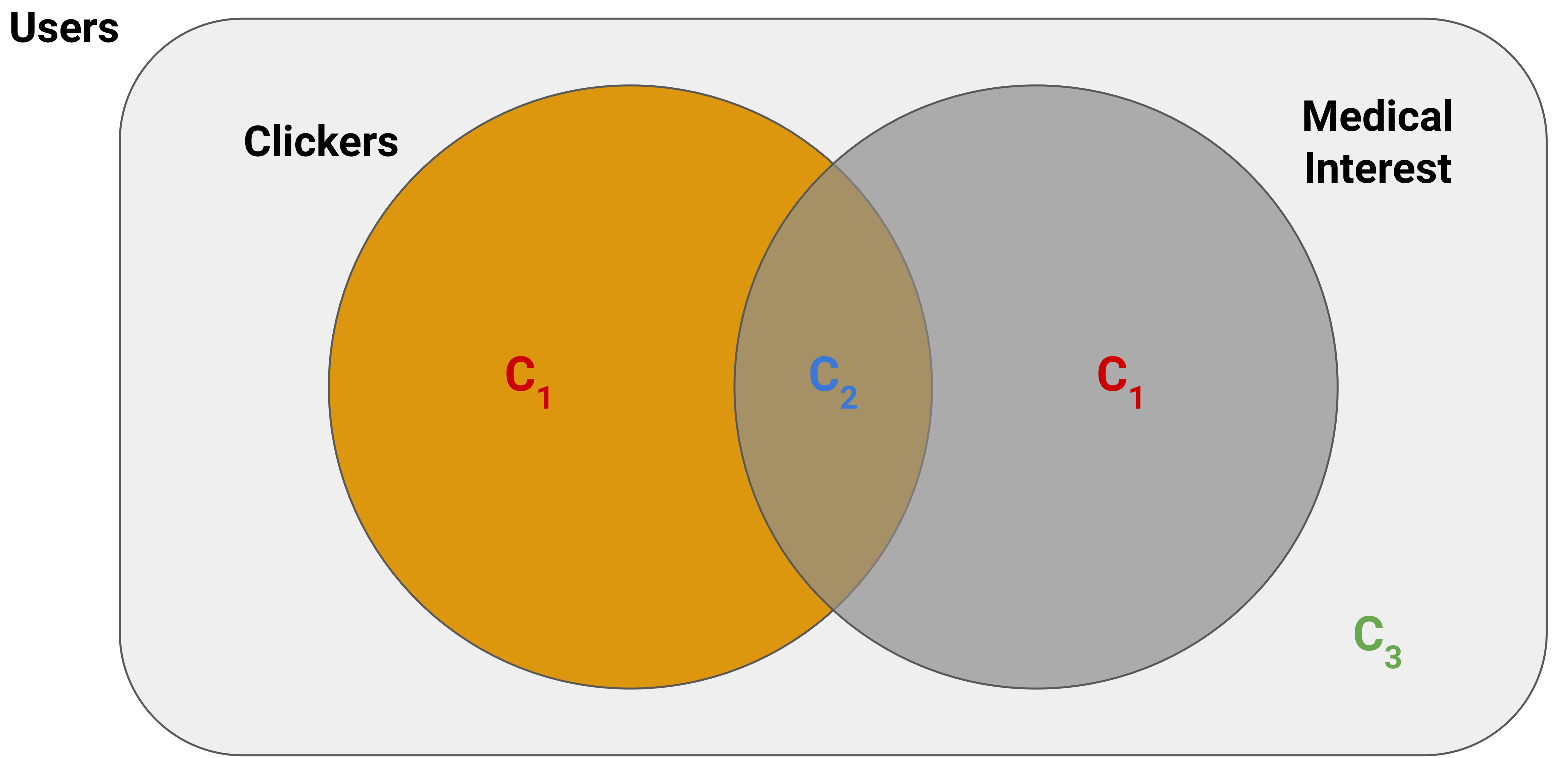
// Users Contexts



// Users Contexts



// Users Contexts



// Off-brand Hand Sanitizer

Product: 80ml 6-pack hand sanitizer

Leading brand retail price: 24 €

Production cost: 12 €

Reasonable retail price range: 15-25 €

// Off-brand Hand Sanitizer

Product: 80ml 6-pack hand sanitizer

Leading brand retail price: 24 €

Production cost: 12 €

Reasonable retail price range: 15-25 €

Facebook as advertising platform

CPC upper-bound of 0.30€

CRP upper-bound of 20%

1000 € daily total advertising budget

// Scenario linear price and linear clicks

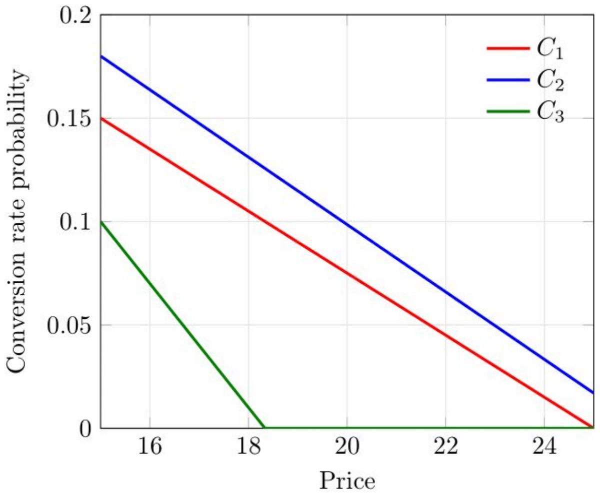
Linear CRP function

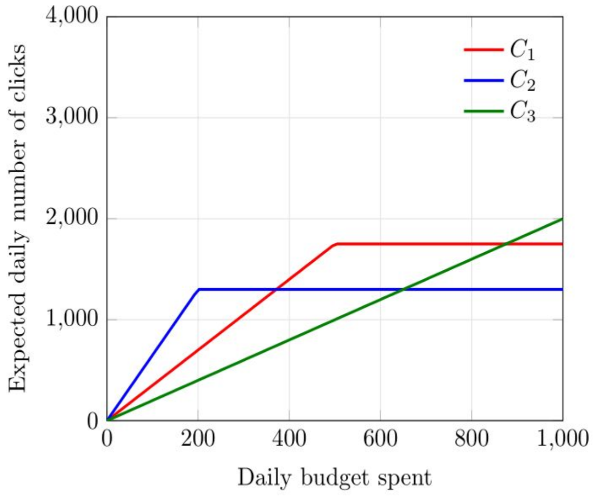
Linear number of clicks functions

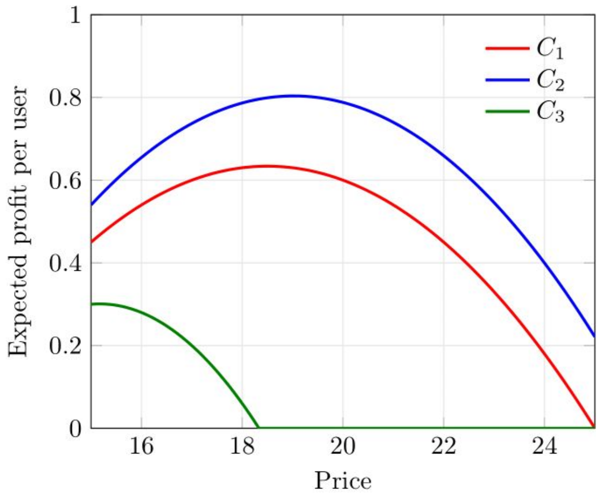
Single phase

Bernoulli modeled CRPs

Gaussian modeled number of clicks







// Scenario tanh price and linear clicks

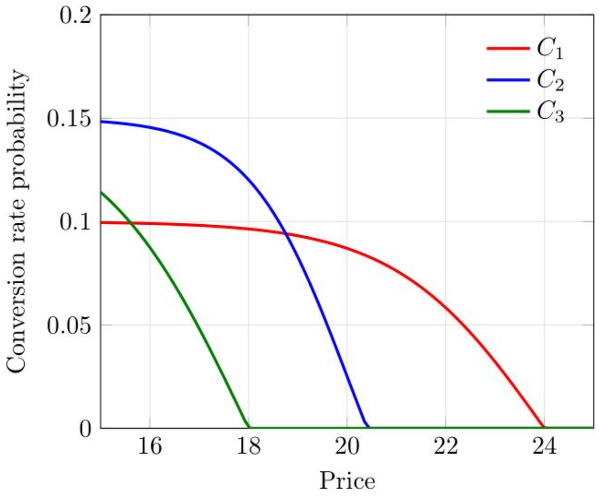
Tanh CRP function

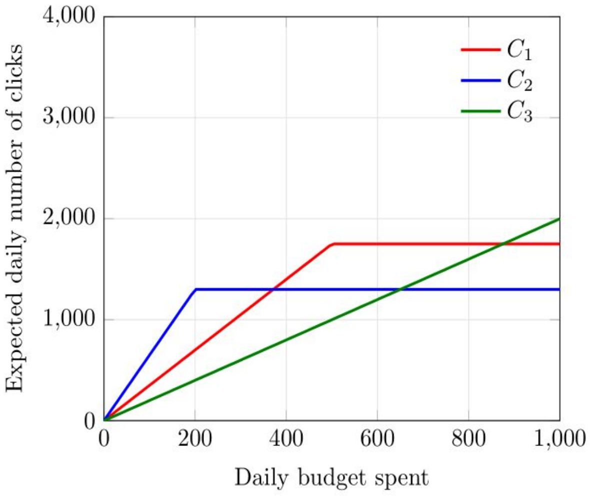
Linear number of clicks functions

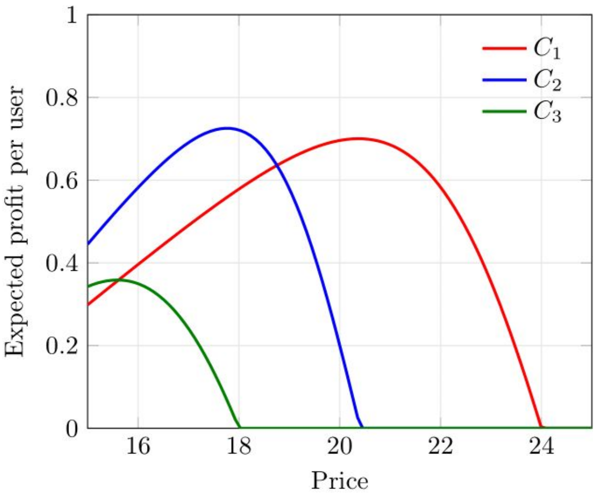
Single phase

Bernoulli modeled CRPs

Gaussian modeled number of clicks







// Scenario non-stationary clicks

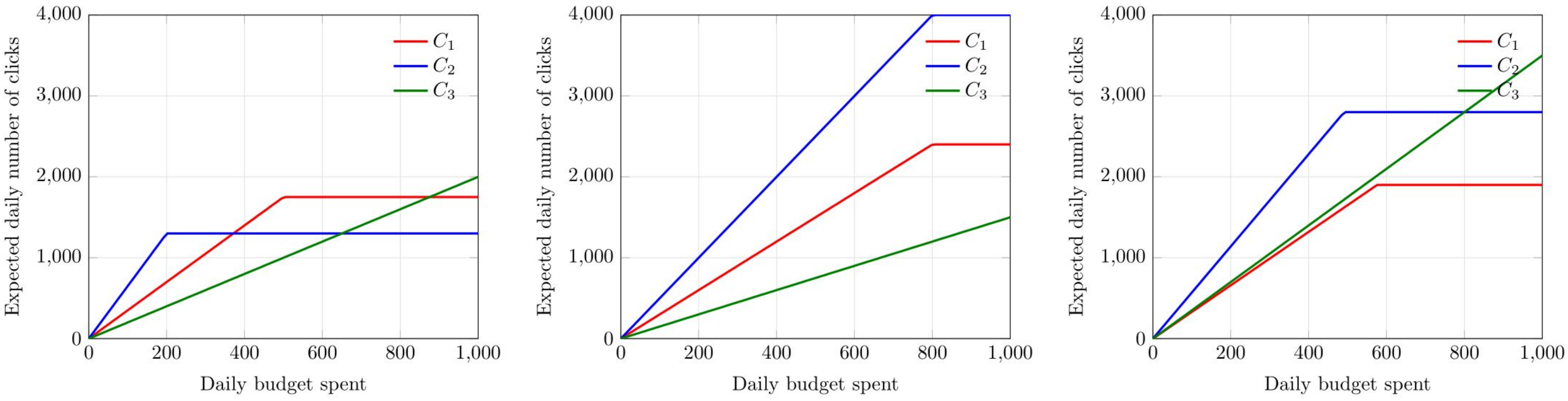
Linear CRP function

Linear number of clicks functions

Three abrupt phases

Bernoulli modeled CRPs

Gaussian modeled number of clicks



Before | During Covid-19 | After

// Stationary Advertising Problem

- Design a combinatorial bandit algorithm;
- Optimize the budget allocation over the three subcampaigns;
- Maximize the total number of clicks;
- Only one phase;
- Plot the cumulative regret.

// Stationary Advertising Algorithm Design

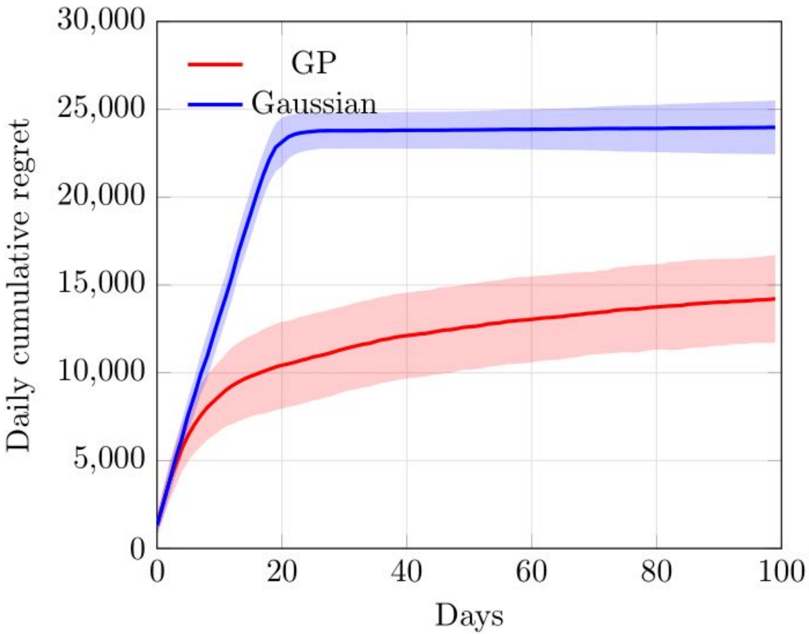
- Correlated constrained arms → combinatorial bandit
- Gaussian Process (GP) regressor VS Gaussian regressor
- Campaign budget optimization → knapsack-like dynamic programming algorithm

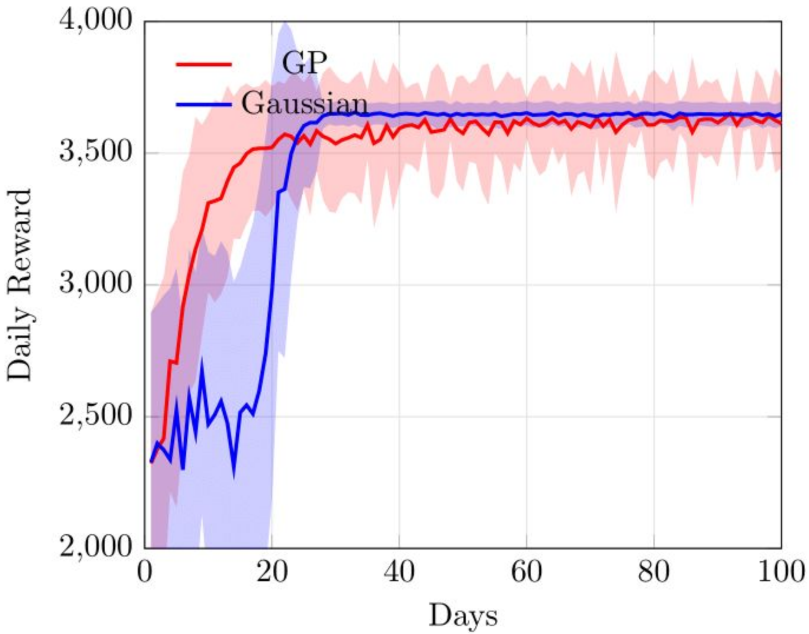
Algorithm

1. Find the best allocation of budgets
2. Observe the reward (# users clicking on ad)
3. Update the combinatorial bandit with pulled arms (budget allocation) and observed rewards
4. Train each sub-campaign regressor

// Stationary Advertising Results

Scenario linear price and linear clicks scenario



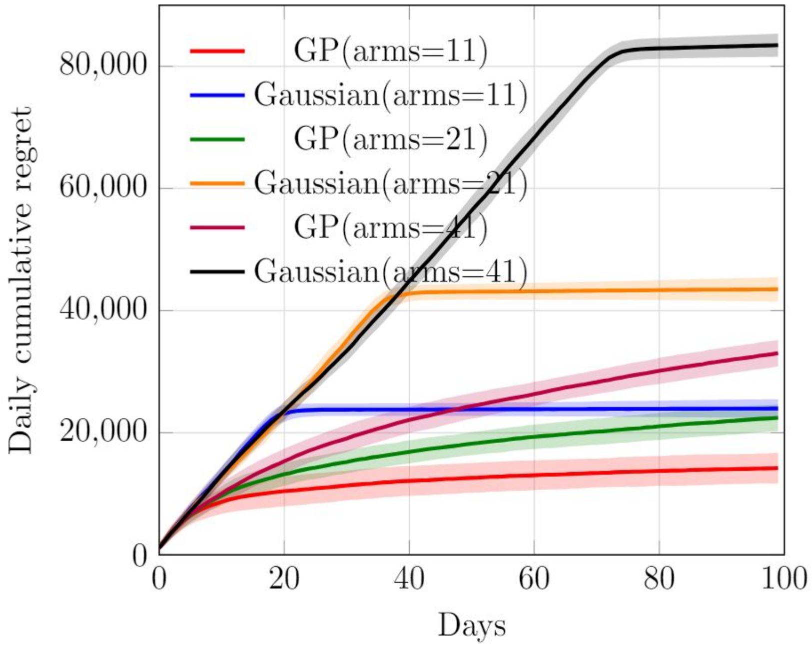


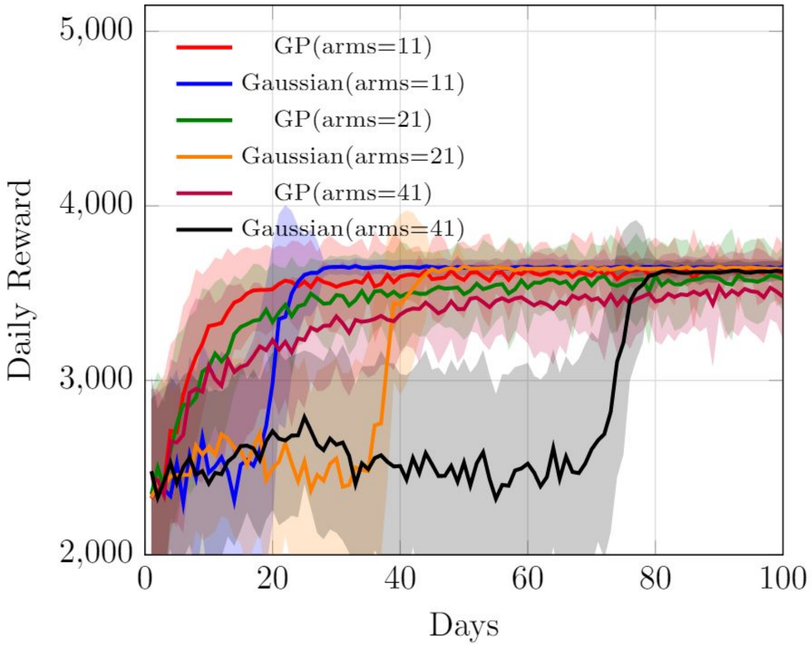
Proposed budget allocation

C_1 = 500€ C_2 = 200€ C_3 = 300€

// Stationary Advertising Results extra

Scenario linear price and linear clicks scenario

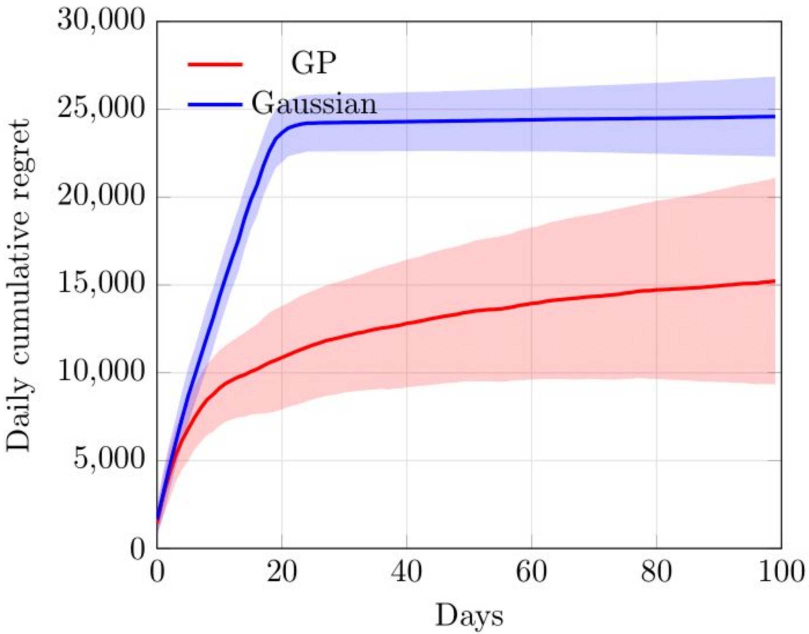


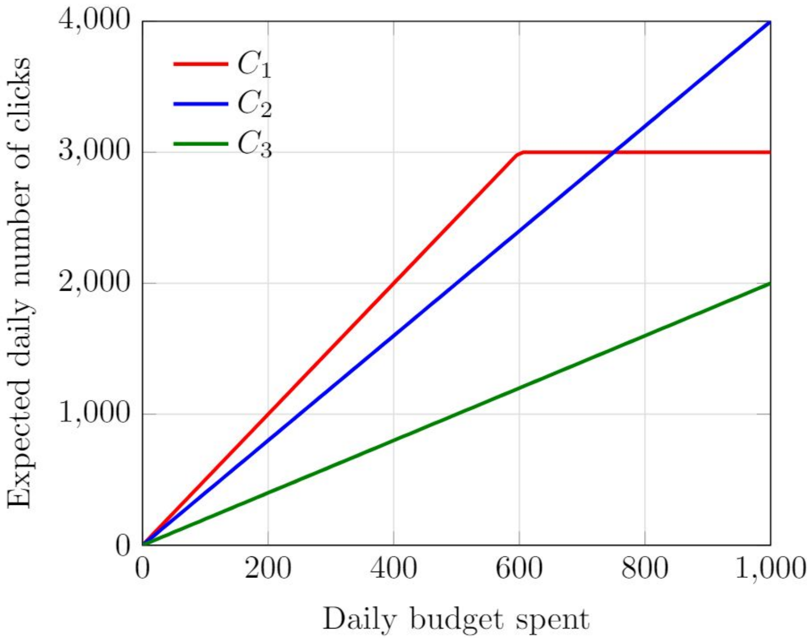


Different number of arms

// Stationary Advertising Results extra

Scenario linear price and linear clicks scenario 2





Proposed budget allocation

C_1 = 600€ C_2 = 400€ C_3 = 0€

// Non-stationary Advertising Problem

- Design a sliding-window combinatorial bandit algorithm;
- Optimize the budget allocation over the three sub-campaigns;
- Maximize the total number of clicks;
- Three phases;
- Plot and compare the cumulative regret with stationary.

// Non-stationary Advertising Algorithm Design

- Passive approach to react to changes

- Sliding window

Algorithm

1. Find the best allocation of budgets
2. Observe the reward (# users clicking on ad)
3. Update the combinatorial bandit with pulled arms (budget allocation) and observed rewards
4. Train each sub-campaign regressor using just the most recent samples

// Non-stationary Advertising Algorithm Design

- Active approach to react to changes

- Change-point detection $\rightarrow$ $\textbf{if} \, |\sum_{i=w/2+1}^{w} Y_i - \sum_{i=1}^{w/2} Y_i| > b$

- Detect distribution changes of infrequently pulled arms $\rightarrow$ randomly pull arms with prob. γ

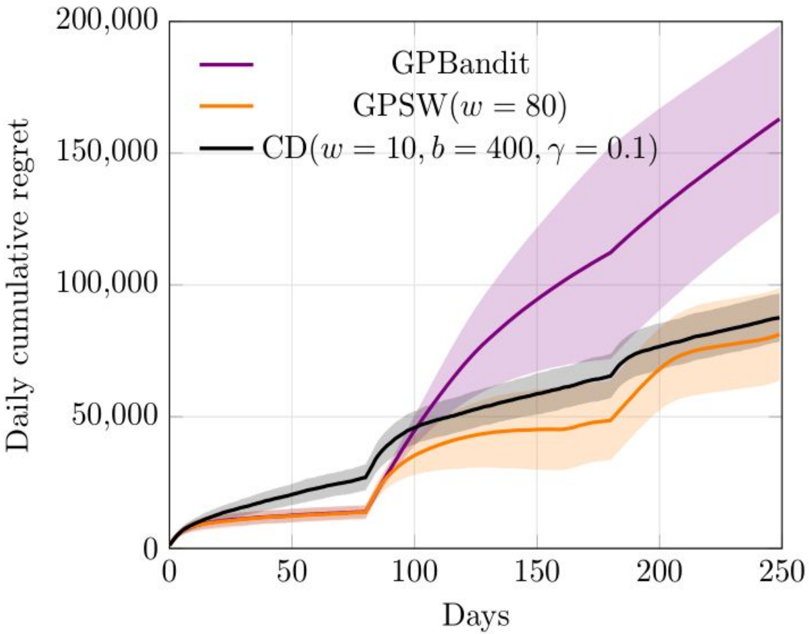
Algorithm

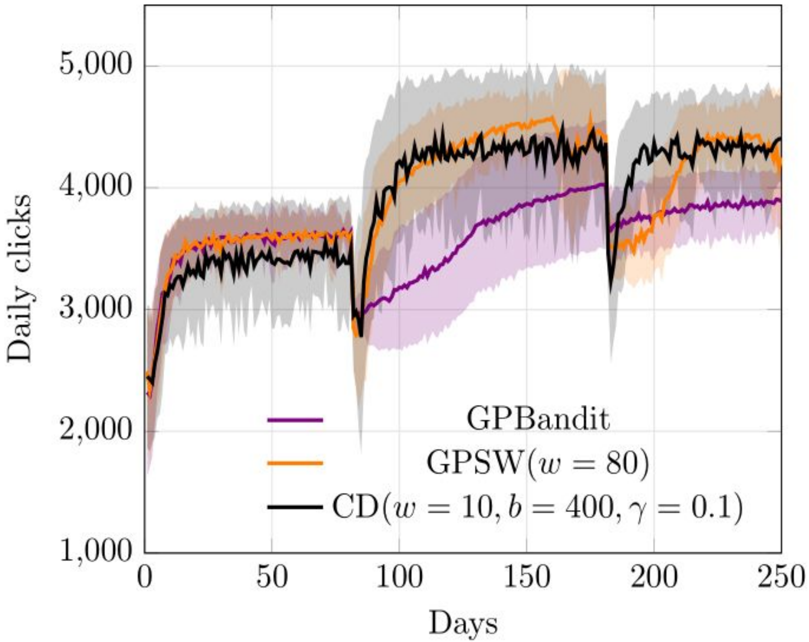
1. Find the best allocation of budgets
2. Observe the reward (# users clicking on ad)
3. Update the combinatorial bandit with pulled arms (budget allocation) and observed reward
4. Detect whether a change in an arm's distribution has happened
5. If so, reset the collected samples so far of all the arms
6. Use collected samples since the last detection to train each sub-campaign regressor

[1] Cao, Y., Wen, Z., Kveton, B., and Xie, Y. Nearly optimal adaptive procedure for piecewise-stationary bandit: a change-point detection approach. ArXiv abs/1802.03692 (2018).

// Non-stationary Advertising Results

Scenario non-stationary clicks



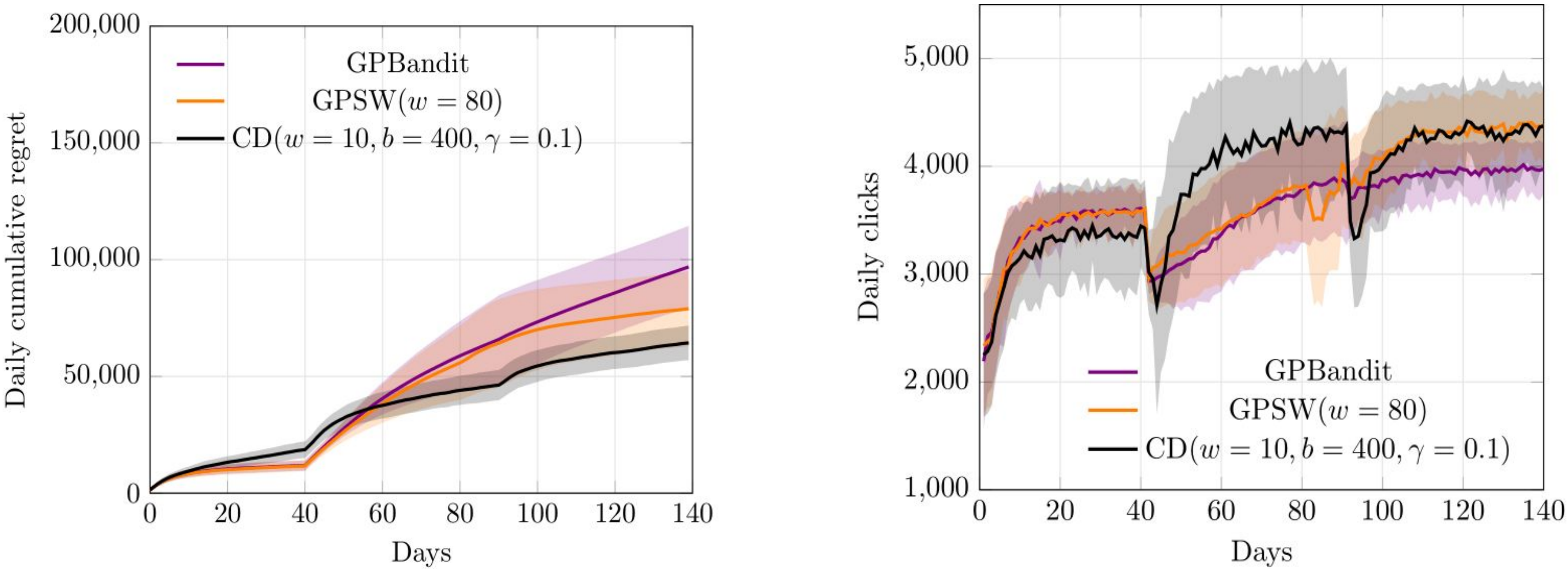


Optimal daily clicks

$p_1 = 3650$ $p_2 = 4600$ $p_3 = 4550$

// Non-stationary Advertising Results extra

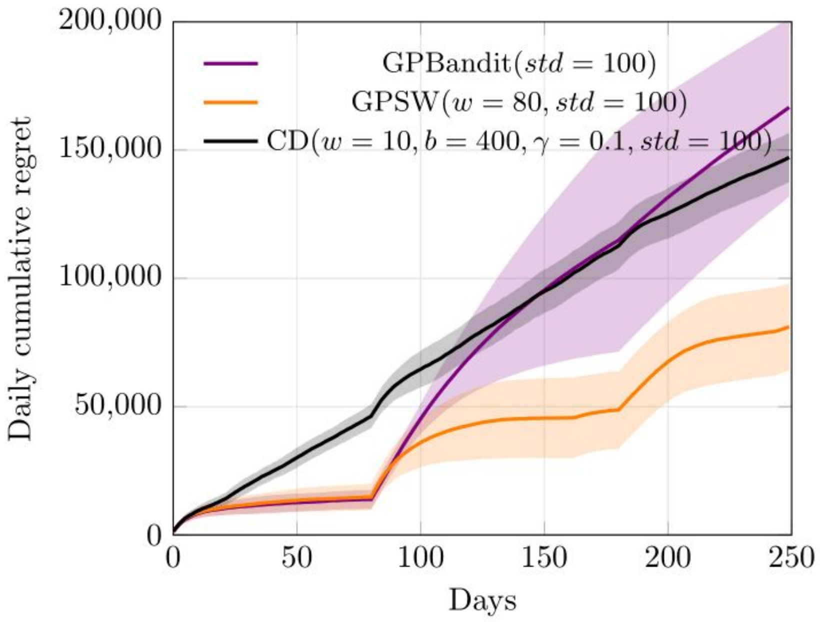
Scenario non-stationary clicks

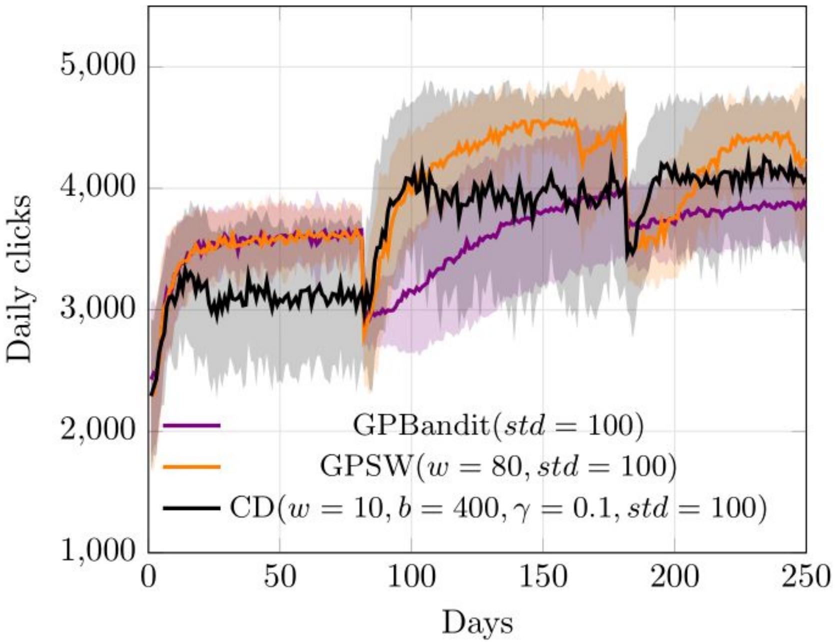


Decreased phases duration

// Non-stationary Advertising Results extra

Scenario non-stationary clicks





Increased std

// Pricing under Fixed Advertising Budget Problem

- Design a learning algorithm for pricing;
- Fixed advertising budget;
- Only one phase;
- Plot the cumulative regret.

// Pricing under Fixed Advertising Budget Algorithm Design

- Upper Confidence Bound 1 (UCB1) and variants
- Thompson Sampling (TS)
- EXP3
- Rewards are scaled Bernoulli distribution → Bound/Sample/Reward profit scaling

Algorithm

Every day:

1. Assign the fixed budget to each sub-campaign
2. For each user that visits, use the unique bandit to choose the price and then update the bandit using the observed reward

// Pricing under Fixed Advertising Budget Algorithm Design

Variants of UCB [2]

1. UCB1-M: exploits monotonicity property of the arms

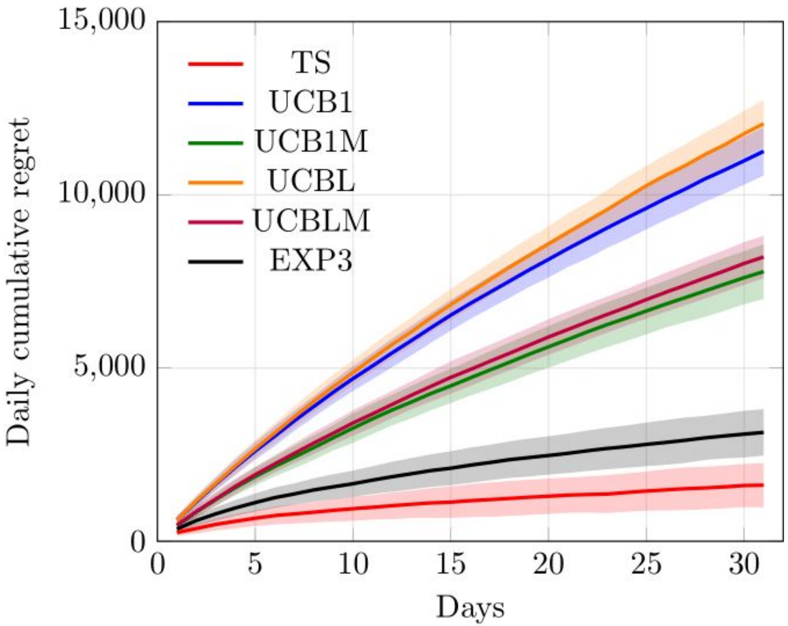
2. UCB-L: exploits the low conversion rate property

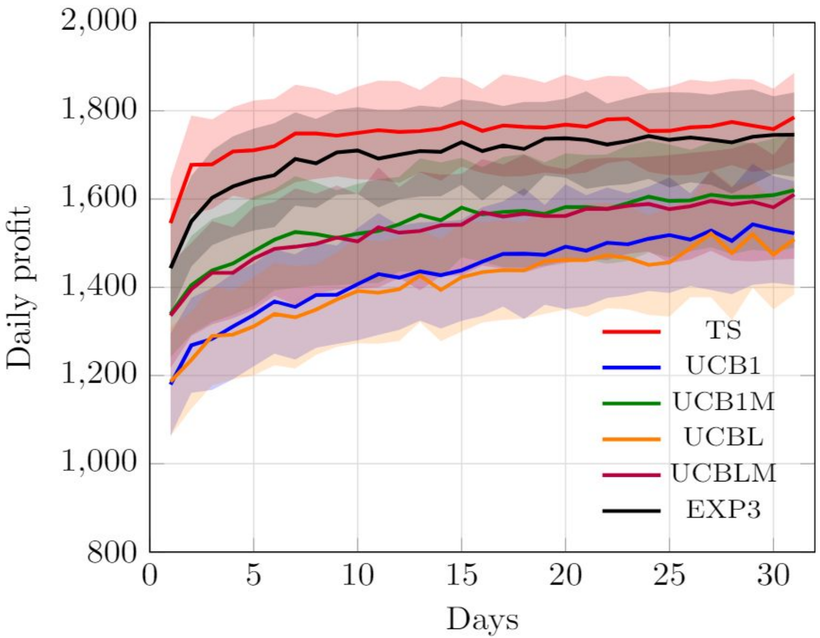
3. UCB-LM: a combination of the previous two techniques

[2] Trovò, F., Paladino, S., Restelli, M., and Gatti, N. | Improving multi-armed bandit algorithms in online pricing settings. International Journal of Approximate Reasoning 98 (2018), 196 – 235.

// Pricing under Fixed Advertising Budget Results

Scenario linear price and linear clicks scenario



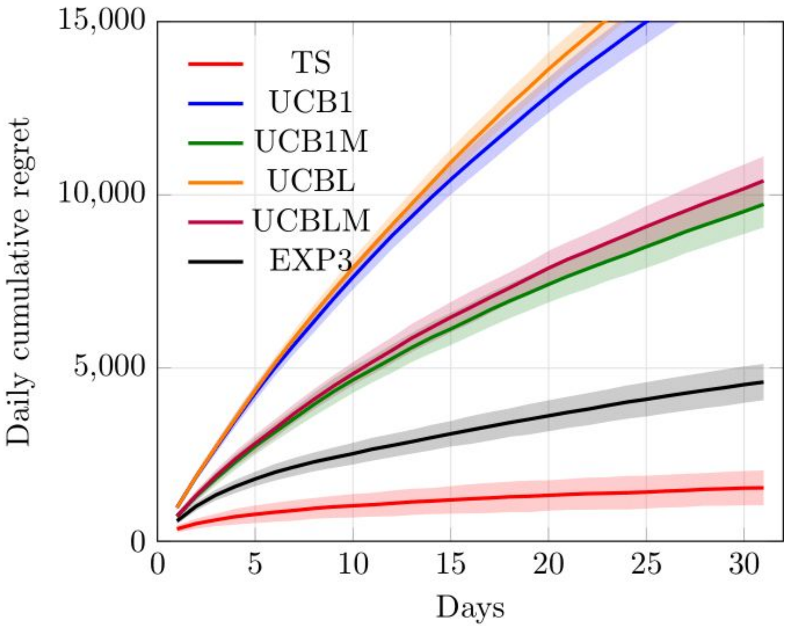


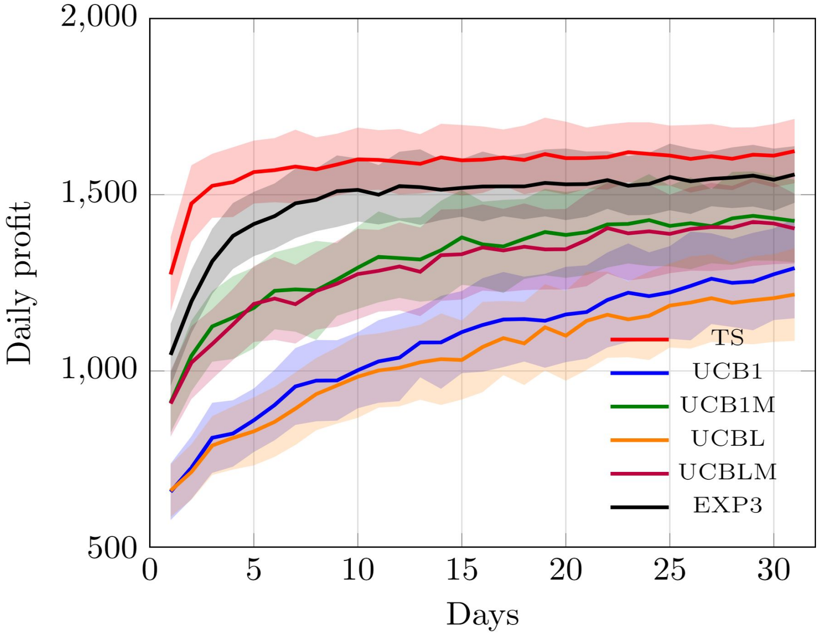
Optimal daily profit

~1797€

// Pricing under Fixed Advertising Budget Results

Scenario tanh price and linear clicks scenario



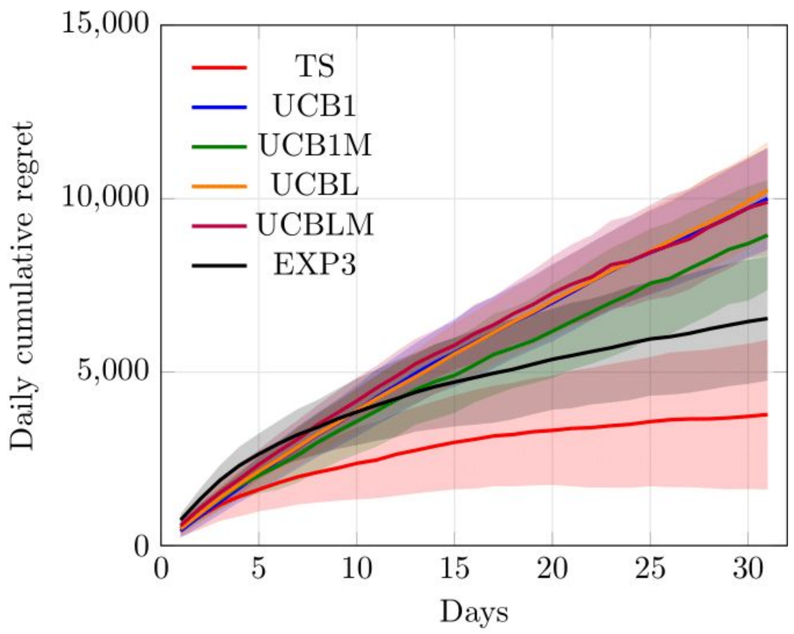


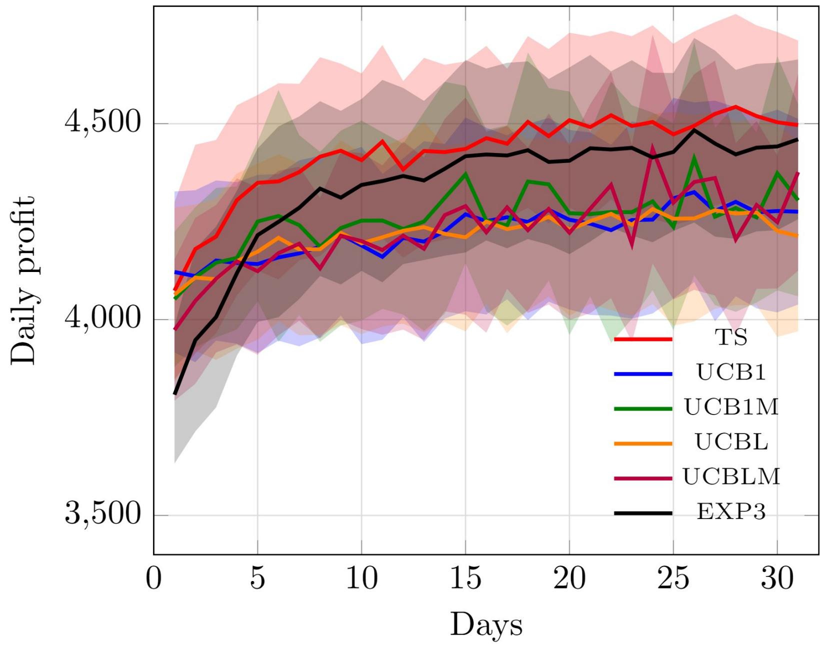
Optimal daily profit

~1630€

// Pricing under Fixed Advertising Budget Results

Scenario linear price and linear clicks scenario 3





Optimal daily profit

~4547€

// Contextual Pricing under Fixed Advertising Budget Problem

- Design a context generation algorithm for pricing;
- Fixed advertising budget;
- Context generation at the end of every week;
- Only one phase;
- Plot the cumulative regret.

// Contextual Pricing under Fixed Advertising Budget Algorithm Design

- Greedy context generation algorithm VS Bruteforce context generation algorithm
- Variant of Hoeffding lower bound
- Problem of re-aggregation in Greedy algorithm → solution: block it

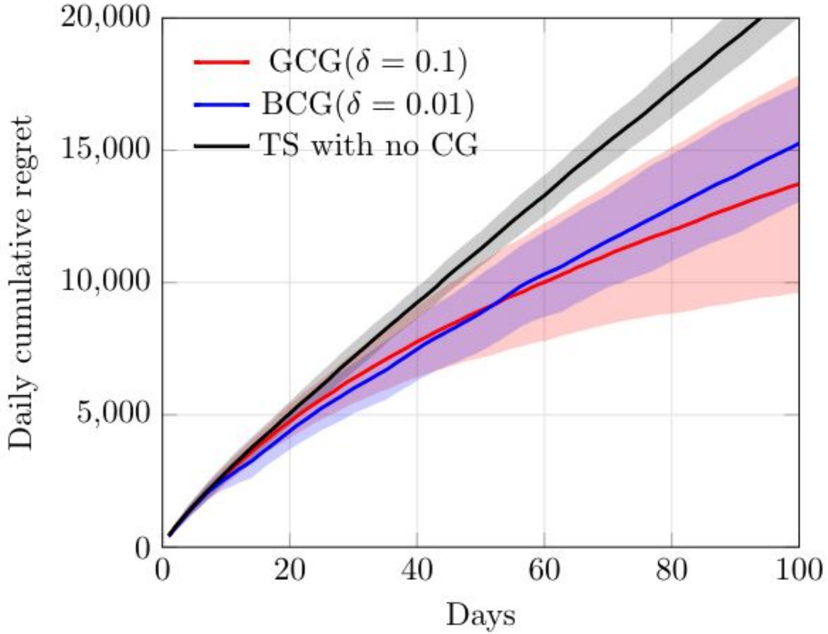
Algorithm

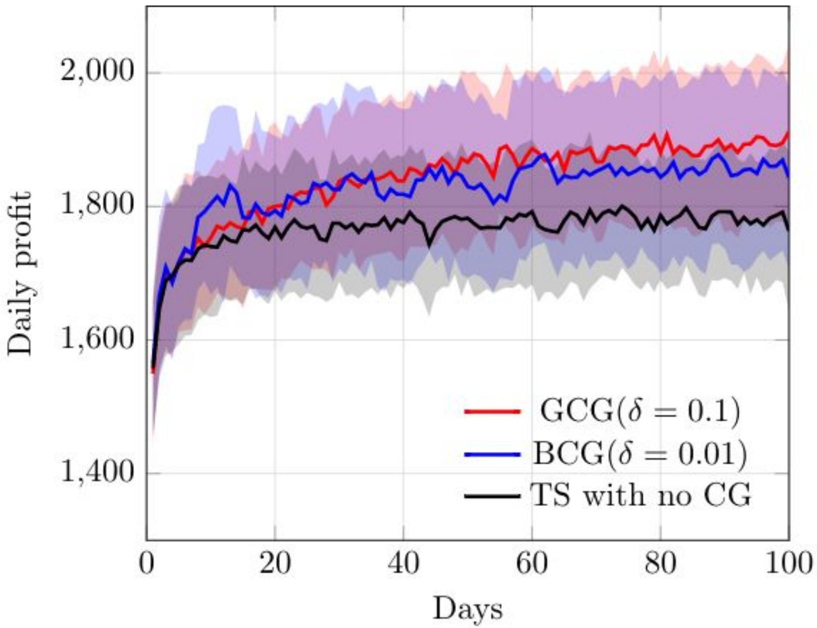
Every week:

1. Build context structure via context generation algorithm
2. Construct a bandit for each context and update its parameters by using old data
3. For each user that visits, use the bandit related to them in order to choose the price and then update the bandit using the observed reward

// Contextual Pricing under Fixed Advertising Budget Results

Scenario linear price and linear clicks scenario



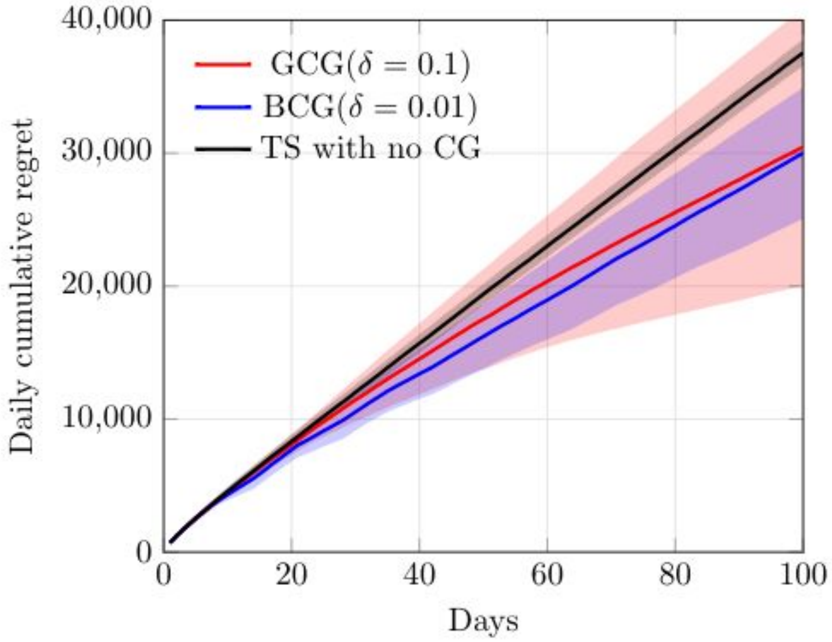


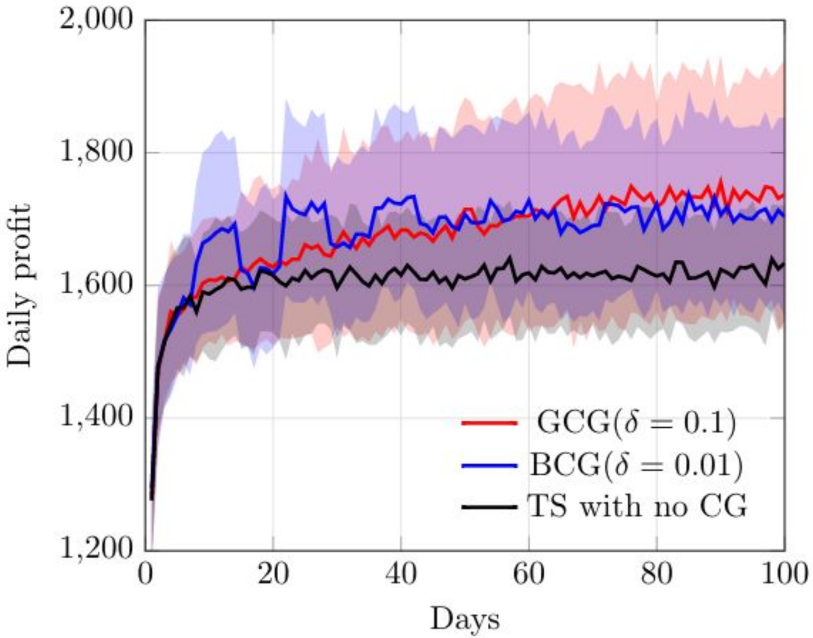
Optimal daily profit

~1979€

// Contextual Pricing under Fixed Advertising Budget Results

Scenario tanh price and linear clicks scenario



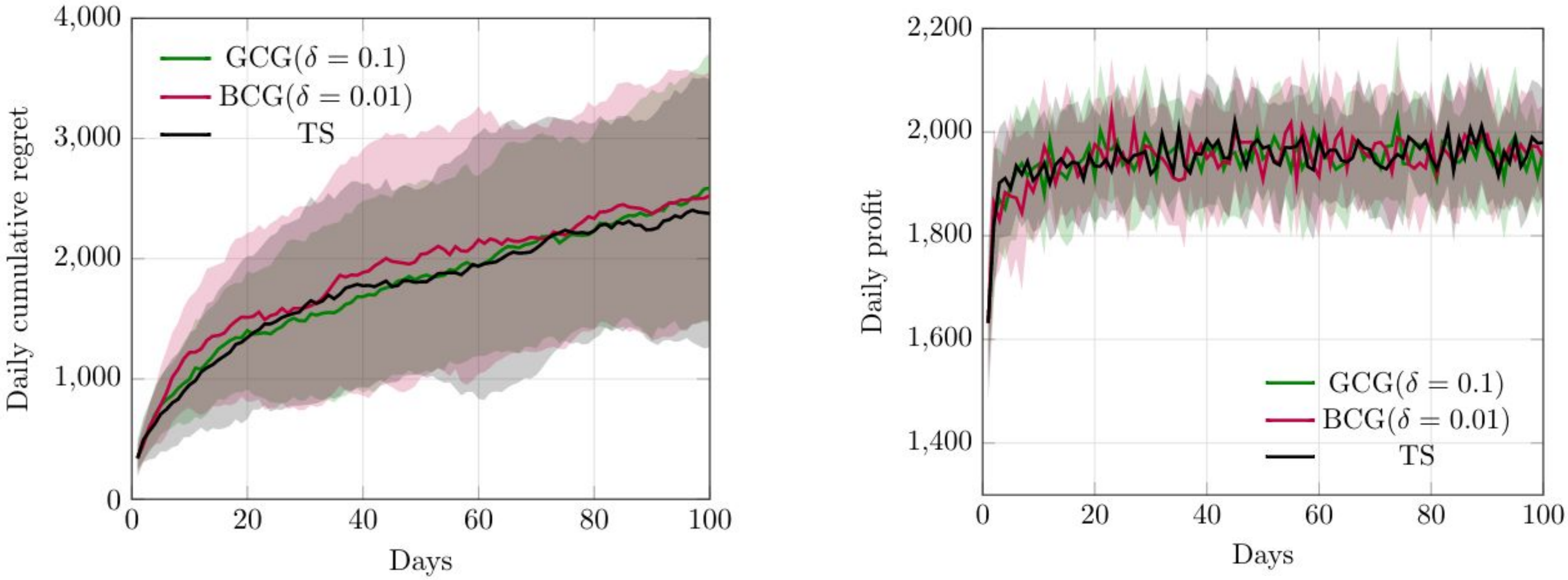


Optimal daily profit

~1982€

// Contextual Pricing under Fixed Advertising Budget Results extra

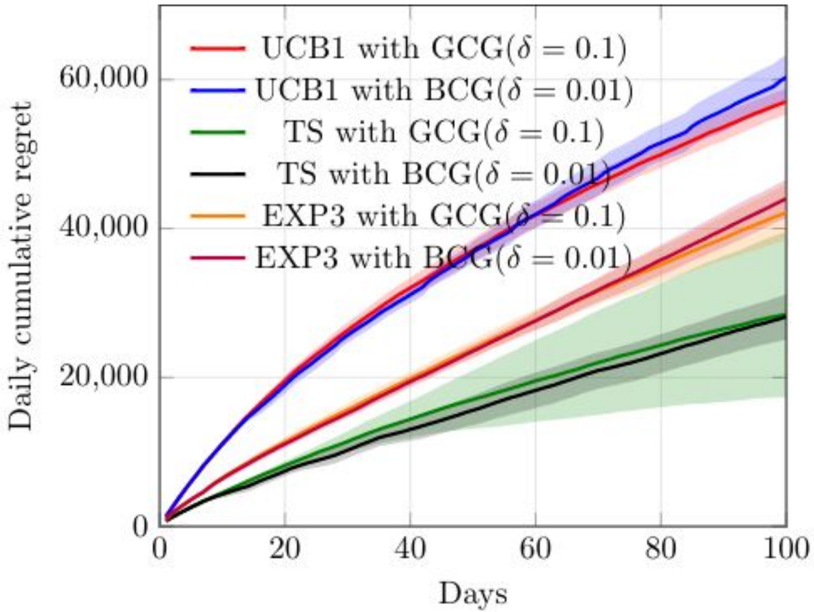
Scenario linear (“unique”) price and linear clicks scenario

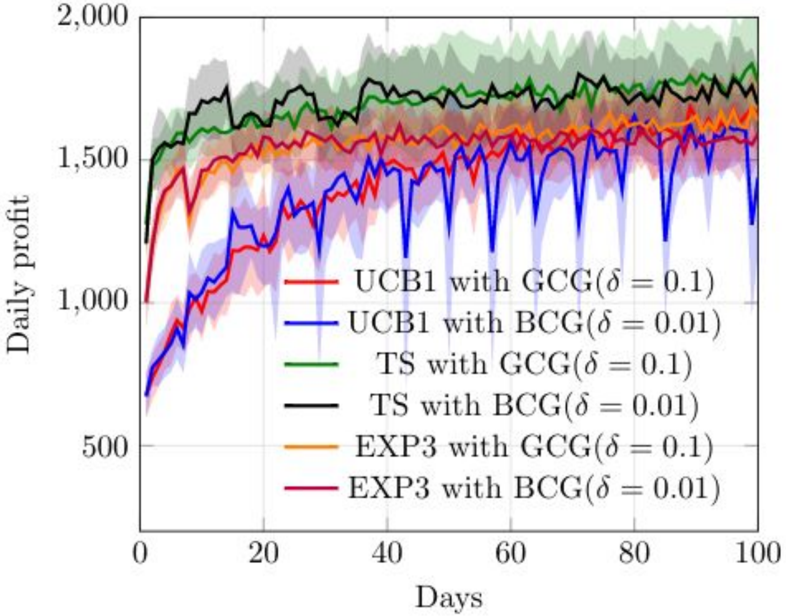


Unique price

// Contextual Pricing under Fixed Advertising Budget Results extra

Scenario tanh price and linear clicks scenario





UCB1 vs EXP3 vs TS

// Pricing & Advertising Joint Optimization Problem

- Design an optimization algorithm for pricing and advertising;
- Context-specific subcampaign and price;
- Only one phase;
- Plot the cumulative regret

// Pricing & Advertising Joint Optimization Algorithm Design

- Estimation value per click: Expectation-based VS Quantile-based method [3]
- Learn number of clicks VS Learn number of clicks $\times$ value per click
- Improved version: estimation value per click using rewards from optimal arm

Algorithm

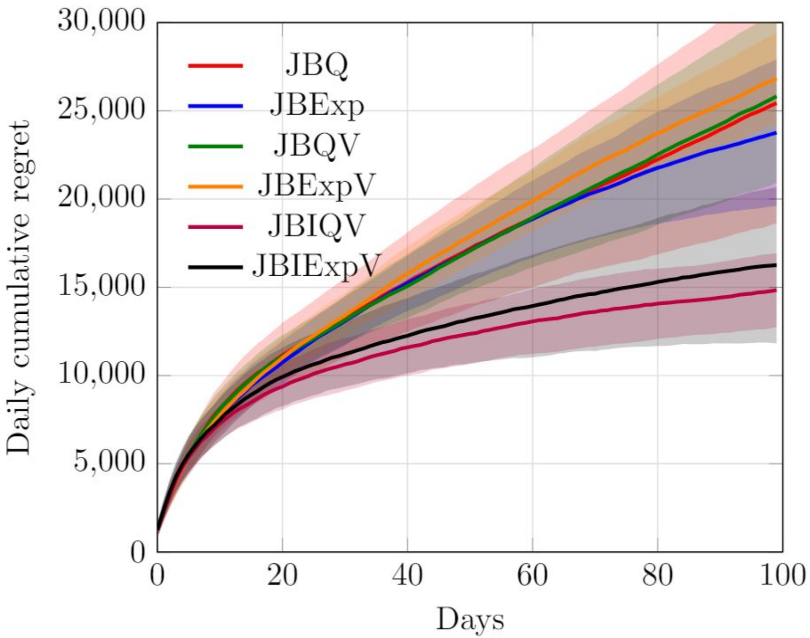
Every day:

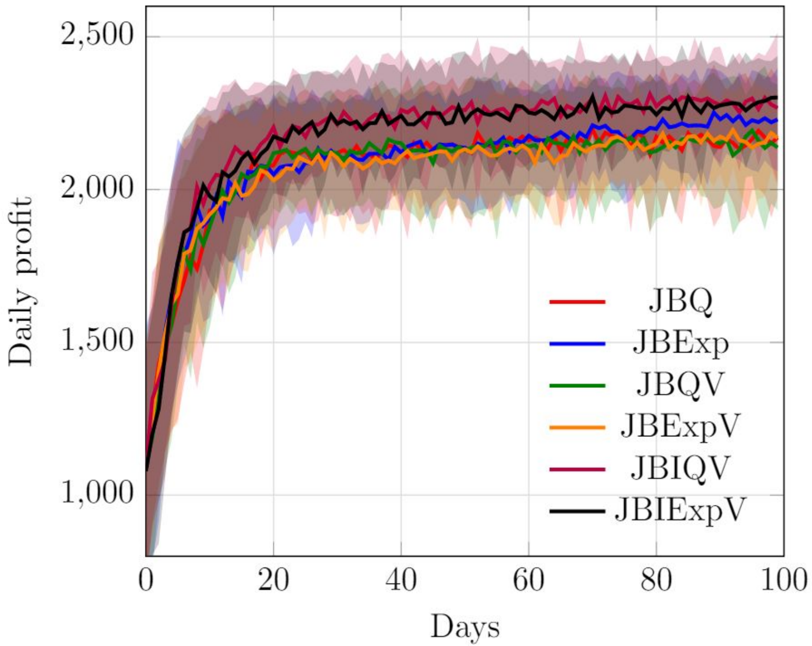
1. Solve optimization problem to select the best budget that maximize expected daily rewards
2. For each user that visits, select the price from the bandit related to that class of user, and then update the bandit using the observed reward
3. Update value per clicks using an estimation method (if necessary)

[3] Nuara, A., Trovò, F., Gatti, N., and Restelli, M. | A combinatorial-bandit algorithm for the online joint bid/budget optimization of pay-per-click advertising campaigns.

// Pricing & Advertising Joint Optimization Results

Scenario linear price and linear clicks scenario



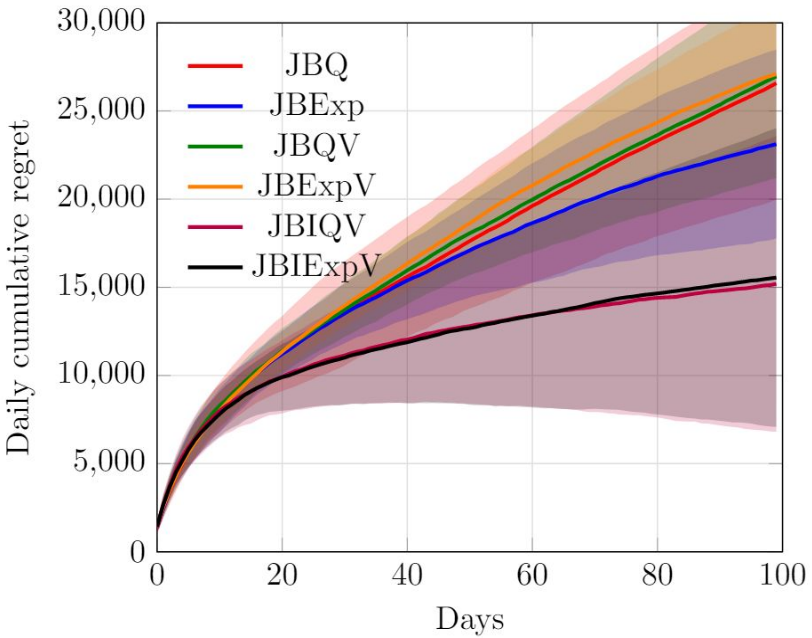


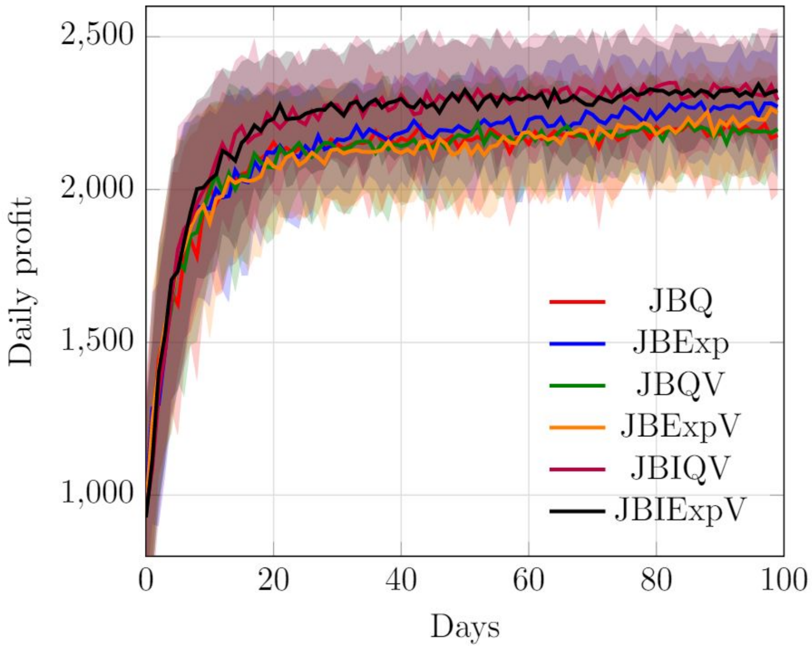
Optimal daily profit

~2327€

// Pricing & Advertising Joint Optimization Results

Scenario tanh price and linear clicks scenario



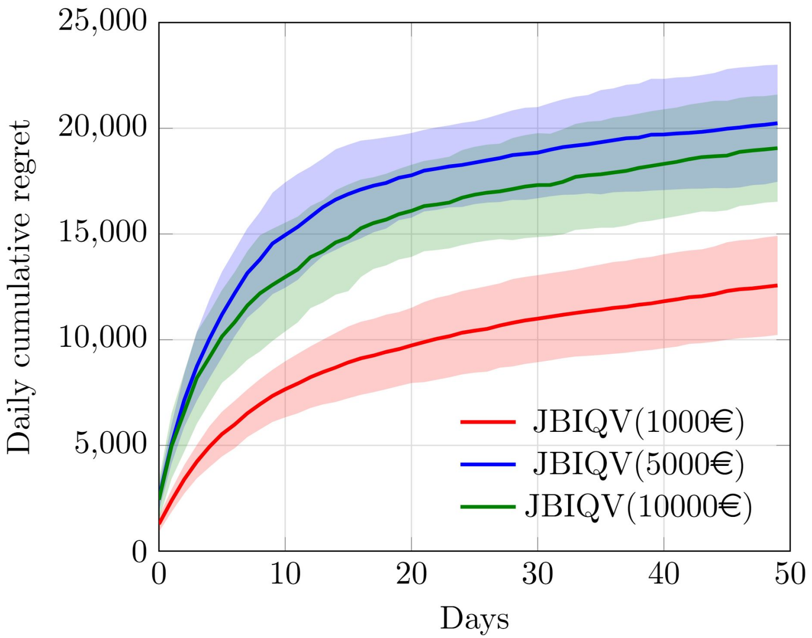


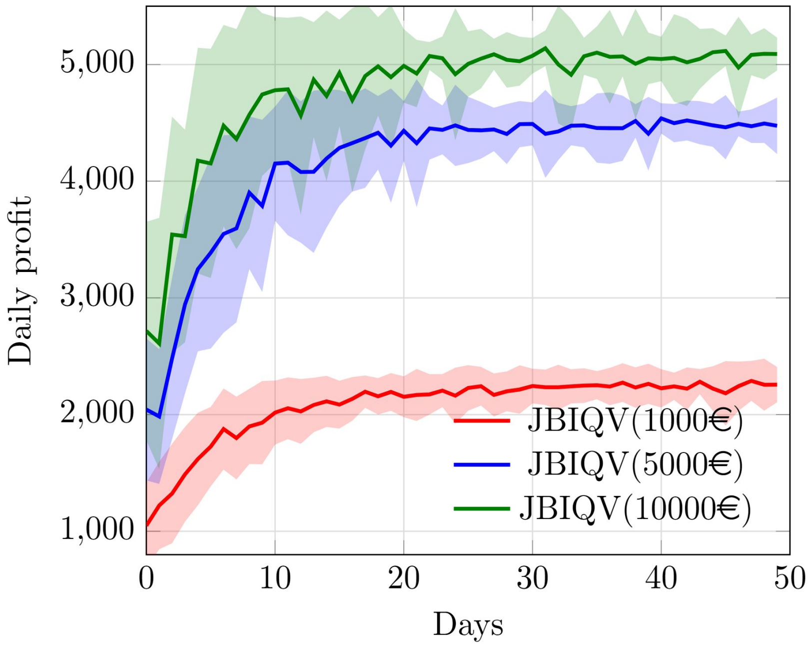
Optimal daily profit

~2365€

// Pricing & Advertising Joint Optimization Results extra

Scenario linear price and linear clicks scenario

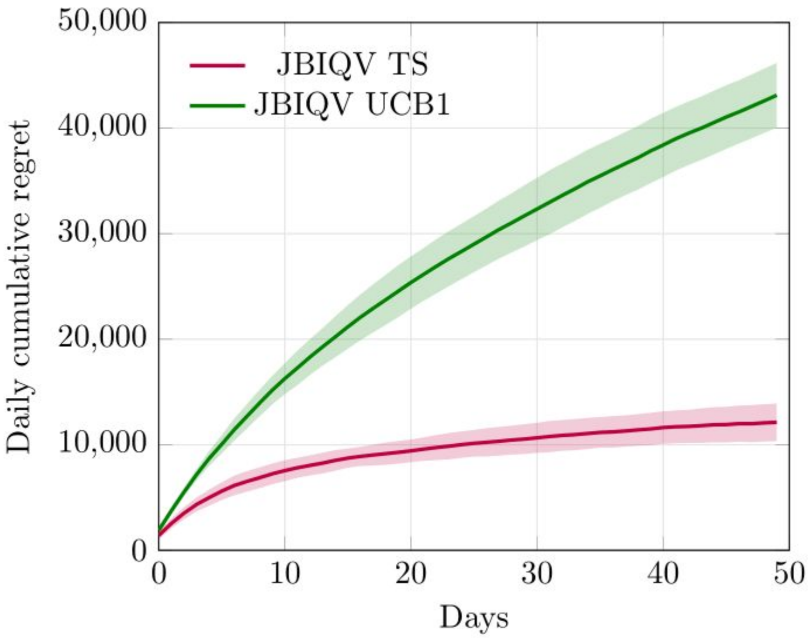


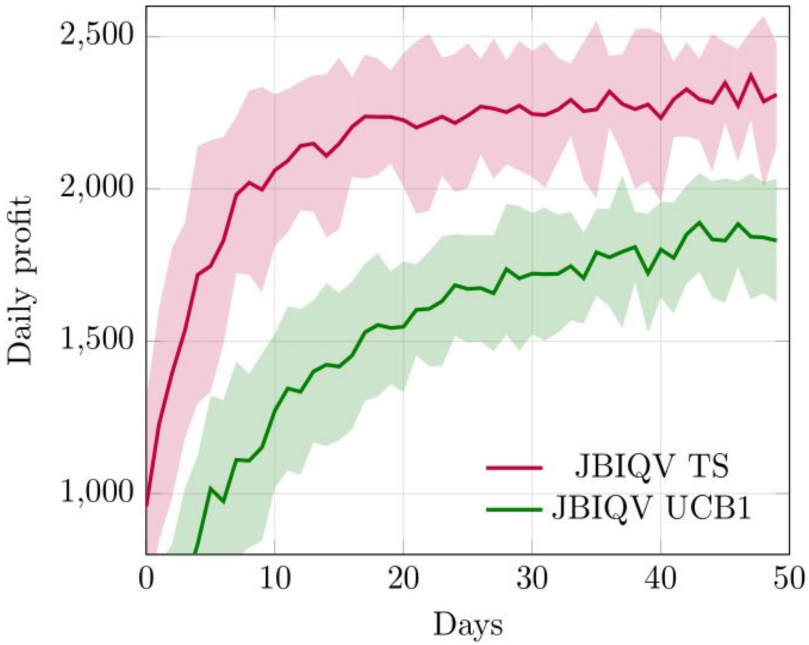


Different budget

// Pricing & Advertising Joint Optimization Results extra

Scenario tanh price and linear clicks scenario

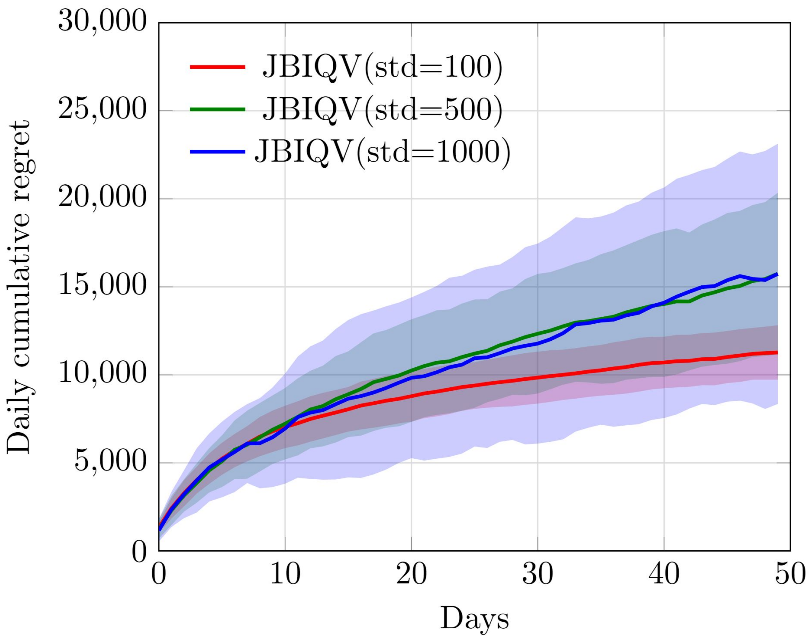


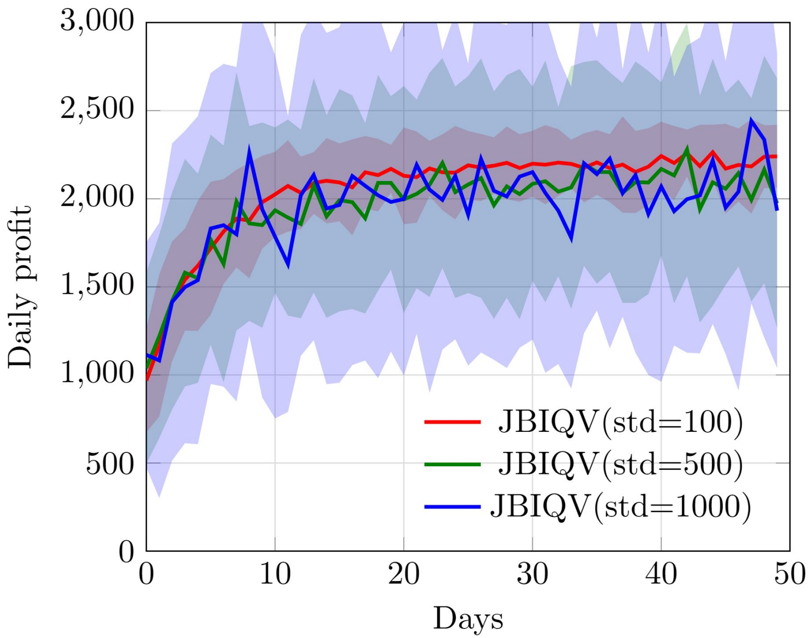


TS vs UCB1

// Pricing & Advertising Joint Optimization Results extra

Scenario tanh price and linear clicks scenario

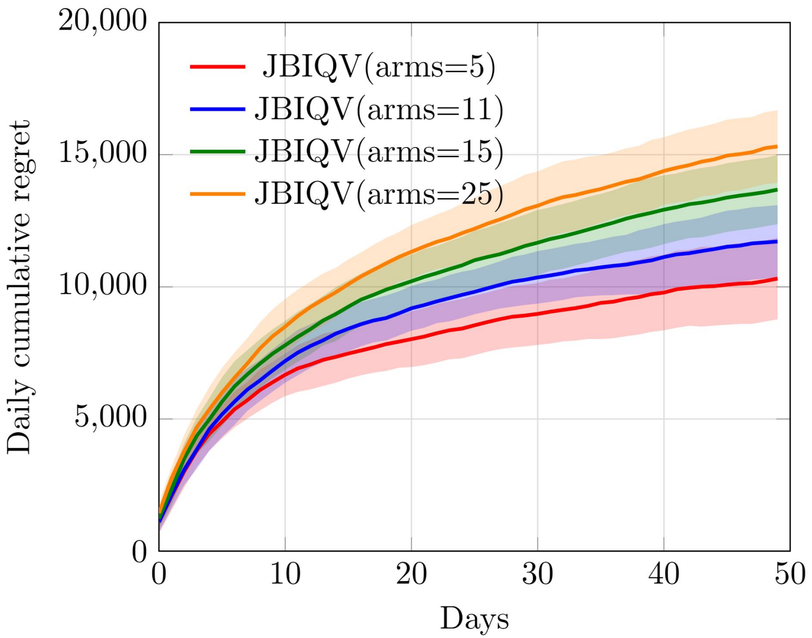




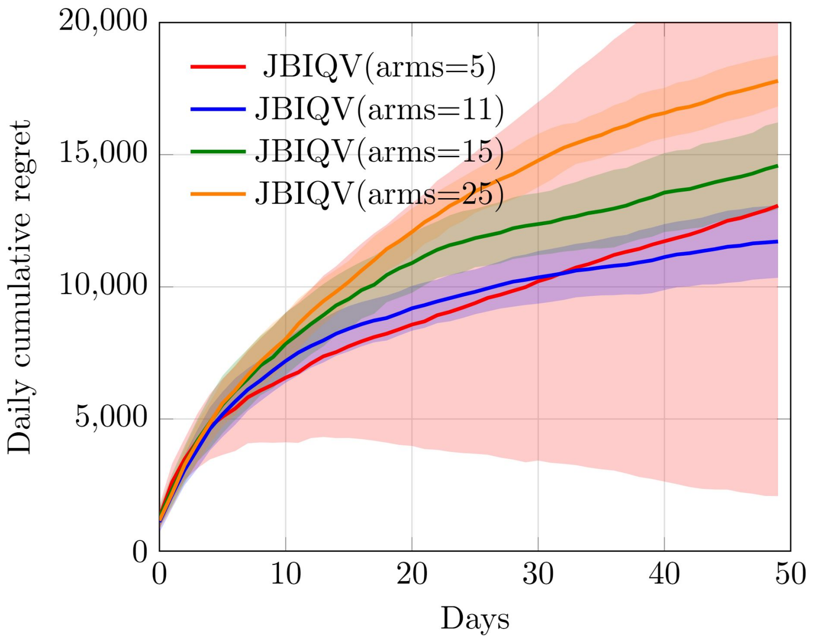
Different std

// Pricing & Advertising Joint Optimization Results extra

Scenario linear price and linear clicks scenario



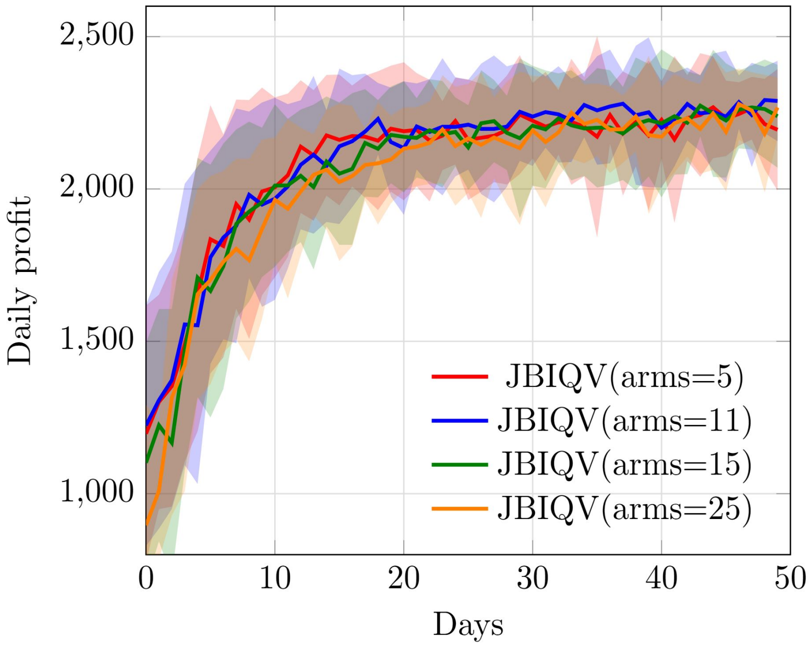
Change pricing arms



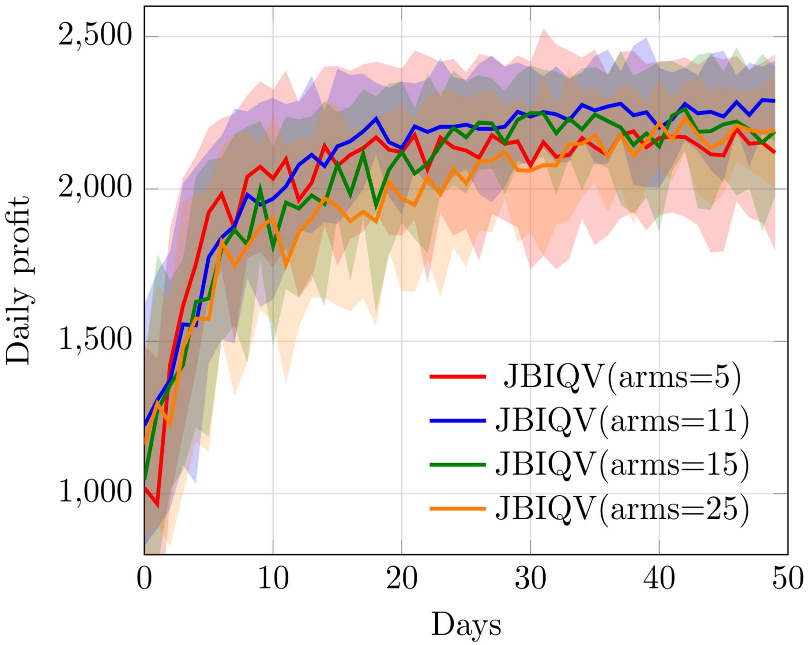
Change advertising arms

// Pricing & Advertising Joint Optimization Results extra

Scenario linear price and linear clicks scenario



Change pricing arms



Change advertising arms

// Unique Pricing & Advertising Joint Optimization Problem

- Design a joint optimization algorithm for pricing and advertising;
- Context-specific sub-campaign;
- Unique price
    - Fixed during a whole day & context-unaware
- Only one phase;
- Plot the cumulative regret

// Unique Pricing & Advertising Joint Optimization Algorithm Design

- Estimators
  - GP regressor to estimate the number of clicks in each sub-campaign
  - Beta distributions for CRP updated in a TS-like manner
- Joint optimization → repeat dynamic programming algorithm for each price

Algorithm

1. Initialize price and and budget at random
2. Observe the rewards and update beta distributions and GPs
3. Sample GPs for each sub-campaign and sample beta for each price and sub-campaign
4. Solve DP problem for each price and get the best pair (price, budget)

// Unique Pricing & Advertising Joint Optimization Algorithm Design

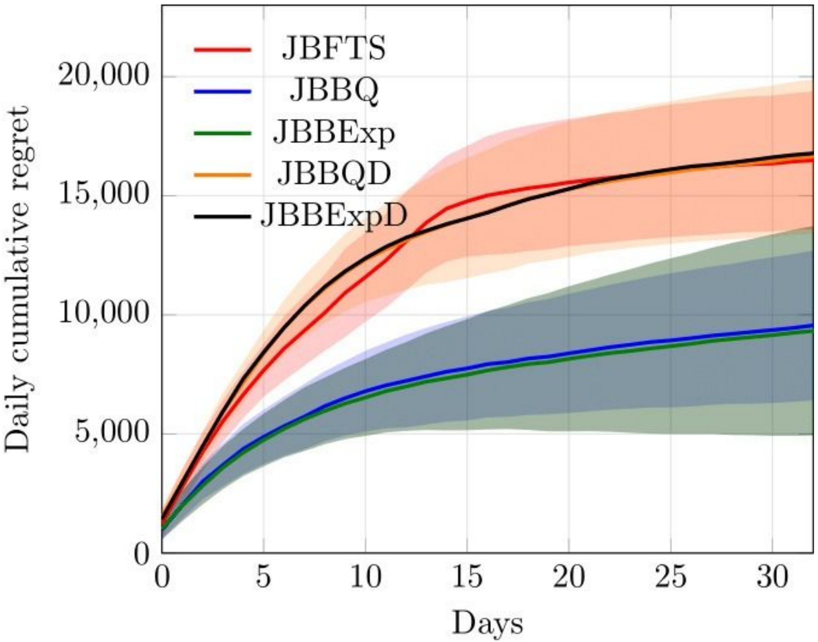
- Joint optimization → repeat dynamic programming algorithm for each price
- Distributions of users depend on advertising → balance distributions

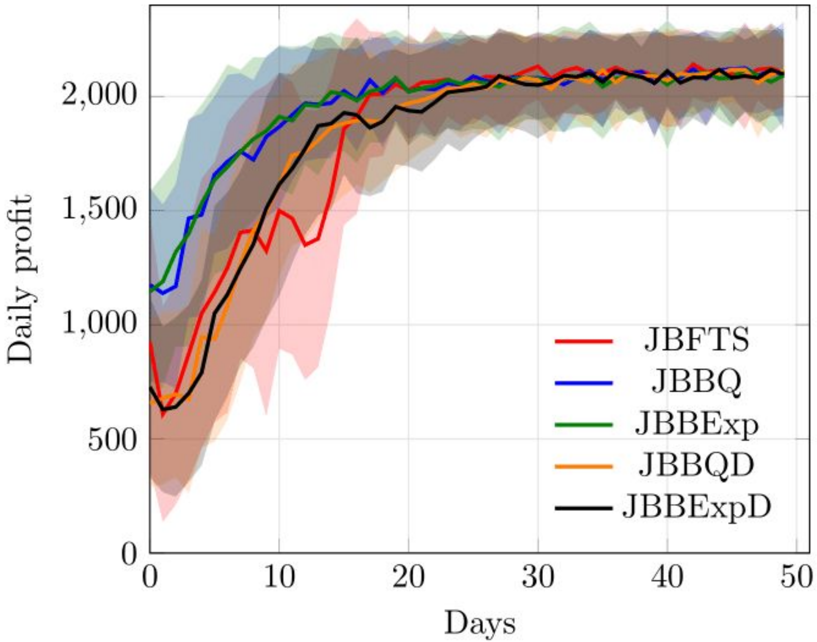
Algorithm

1. Initialize price and and budget at random
2. Observe the rewards
3. Sample GPs for each sub-campaign and updates expected value per click
4. Solve DP problem for each price and get the best pair budget
5. Re-train a bandit in an offline fashion keeping the data balanced with the expected distribution

// Unique Pricing & Advertising Joint Optimization Results

Scenario linear price and linear clicks scenario



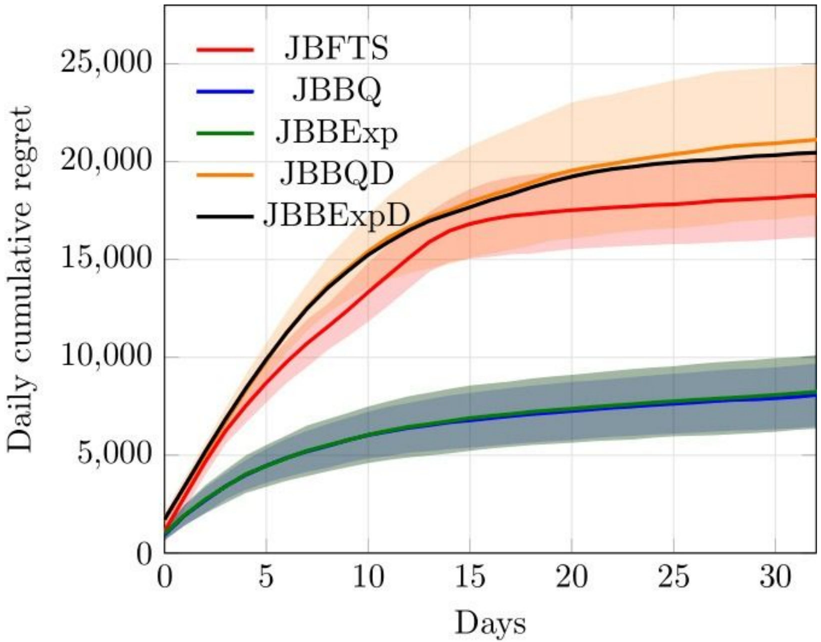


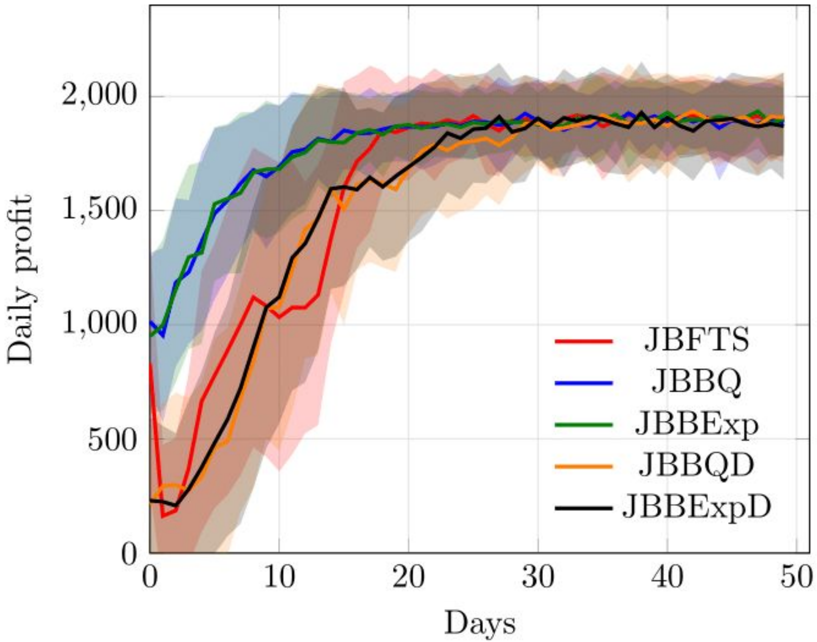
Optimal daily profit

~2161€

// Unique Pricing & Advertising Joint Optimization Results

Scenario tanh price and linear clicks scenario



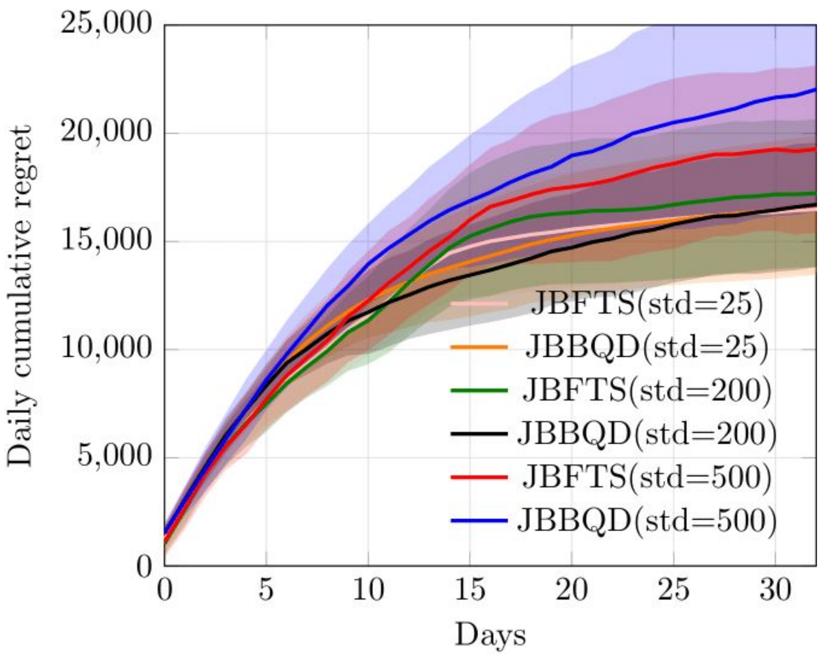


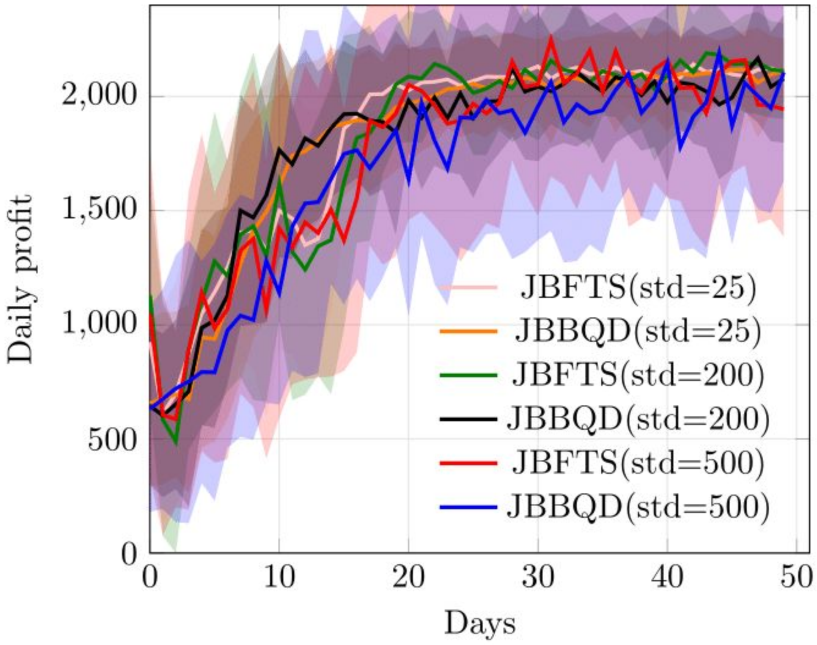
Optimal daily profit

~1949€

// Unique Pricing & Advertising Joint Optimization Results extra

Scenario linear price and linear clicks scenario

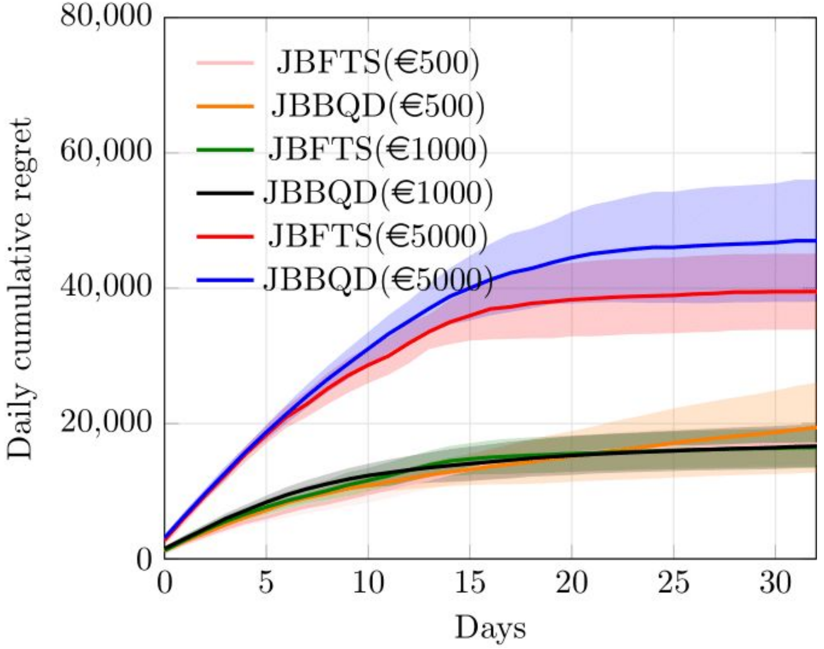


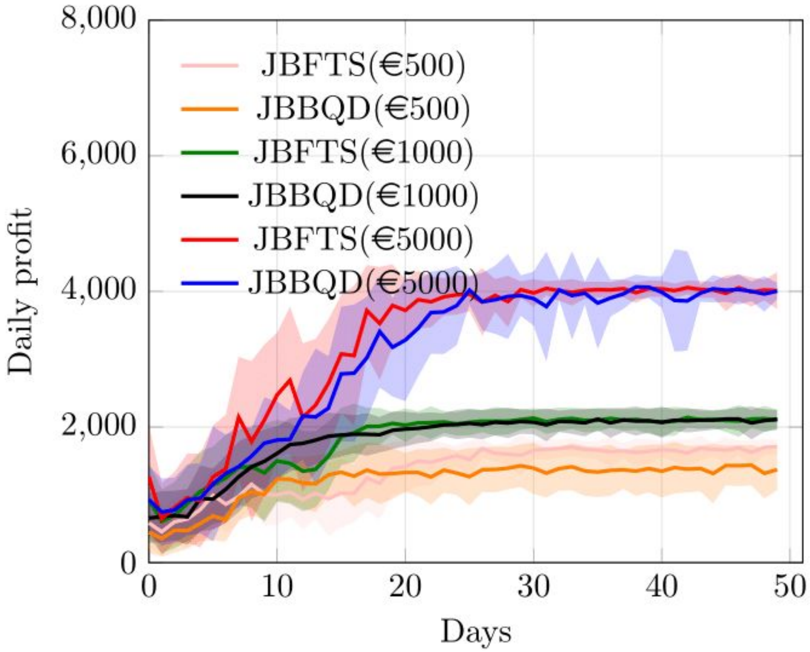


Different std

// Unique Pricing & Advertising Joint Optimization Results extra

Scenario linear price and linear clicks scenario





Different budget

Pricing & Advertising

Off-brand Hand Sanitizer

Luca Alessandrelli
Jacopo Pio Gargano
Riccardo Poiani
Tang-Tang Zhou

Data Intelligence Applications

Politecnico di Milano

July 1st, 2020