E-Commerce Promotions Personalization via Online Multiple-Choice Knapsack with Uplift Modeling

Javier Albert*
javier.albert@booking.com
Booking.com
Tel Aviv, Israel

Dmitri Goldenberg*
dima.goldenberg@booking.com
Booking.com
Tel Aviv, Israel

ABSTRACT

Promotions and discounts are essential components of modern e-commerce platforms, where they are often used to incentivize customers towards purchase completion. Promotions also affect revenue and may incur a monetary loss that is often limited by a dedicated promotional budget. We propose an Online Constrained Multiple-Choice Promotions Personalization framework, driven by causal incremental estimations achieved by uplift modeling. Our work formalizes the problem as an Online Multiple-Choice Knapsack Problem and extends the existent literature by addressing cases with negative weights and values as a result from causal estimations. Our real-time adaptive method guarantees budget constraints compliance achieving above 99.7% of the potential optimal impact on various datasets. It was deployed in a large-scale experimental study at Booking.com - one of the leading online travel platforms in the world. The application resulted in 162% improvement in sales while complying a zero-budget constraint, enabling long-term self-sponsored promotional campaigns.

CCS CONCEPTS

• **Information systems** → **Personalization**; **Recommender systems**; • **Applied computing** → *Multi-criterion optimization and decision-making*; • **Computing methodologies** → **Optimization algorithms**; • **Theory of computation** → *online approximation*; *online control and optimization*; Integer programming.

KEYWORDS

Uplift Modeling, Causal Inference, Promotions Personalization, Online Optimization, Multiple Choice Knapsack Problem

ACM Reference Format:
Javier Albert and Dmitri Goldenberg. 2022. E-Commerce Promotions Personalization via Online Multiple-Choice Knapsack with Uplift Modeling. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557100

*Both authors contributed equally to this research.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557100

1 INTRODUCTION

Promotions play a key role in marketing efforts of e-commerce platforms. They generate a significant uplift in sales by offering more value to the customers. Offering benefits to a customer (such as in Figure 1) creates an incentive to complete a booking and drives the business growth. While increasing the likelihood to purchase, a promotion may also incur an incremental monetary loss, if the net-revenue from sales when the promotion is offered is smaller than the net-revenue when the promotion is not offered [28]. Often, a dedicated budget is set to cover this incremental net-revenue loss, to ensure a sustainable campaign [6]. Budget constrained marketing gains recent popularity [11], in particular within promotions allocation [14]. Many solutions consider a constant cost factor [18]. However, in a presence of causal incremental estimations, the monetary impact of a promotion may result in varying, positive or negative outcomes. Existing solutions rely mainly on intent-based models, and do not accommodate for negative values from causal models, and therefore not applicable for such problems.

In this paper, we formalize the budget constrained multiple-choice promotion assignment problem, by using causal estimations via uplift modeling. We propose a novel solution for negative values and weights in Online Multiple-Choice Knapsack setup. We present the deployment of the method at Booking.com - one of the largest online travel platform and test it against various benchmarks in a real-life experimental study. Our main contributions are:

(1) Formulation of the budget-constrained promotions personalization problem as an Online-MCKP with causal estimations.
(2) A two-steps solution, relying on causal uplift modeling estimations and multiple-choice knapsack optimization.
(3) Novel extension of the Online-MCKP solution to accommodate negative values and weights.
(4) A Large-scale experimental study on real promotions offering, conducted on Booking.com platform.

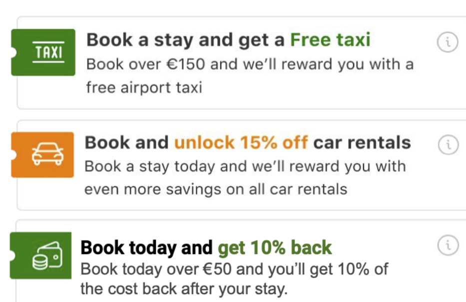


Figure 1: Examples of travel promotions.

The study proceeds as follows: Section 2 covers the related work and introduces key concepts. Section 3 formalizes the *Constrained Promotions Personalization Problem* as *Online Multiple-Choice Knapsack*. Section 4 covers the solution framework, including the uplift modeling method, linear programming solution, and suggested Online-MCKP approach. Section 5 describes a large-scale experimental study, including details on datasets and methods, the experimental setup and the deployment of the solution. Section 6 presents and discusses the experimental results. The final section concludes.

2 RELATED WORK

2.1 Causal Uplift Modeling

The effect a promotion has on the probability of completing a purchase and on the expected net revenue loss varies from customer to customer. The *Conditional Average Treatment Effect* (CATE) is defined as the expected change in a metric of interest (conversion, revenue, click rate, churn, etc.) *caused* by a treatment, given the individual's characteristics [4]. Various machine learning techniques were introduced to estimate the CATE [2], and the field is commonly known as Uplift Modeling. The input data is usually obtained from randomized controlled trials, where the customers are randomly assigned with different treatments. The collected data is used to train machine learning models that estimate the incremental causal effects of each treatment at the individual level.

A notable modeling technique for CATE estimation is the Meta-learning group of estimators, such as the two-models estimator [10], the transformed outcome estimator [2] and the X-learner estimator [13]. They allow using classical machine learning techniques to estimate CATE. Other tailored methods, such as uplift trees and neural network based approaches [27] modify existing machine learning algorithms to be suitable for CATE estimation. Over the recent years, uplift modeling has become popular in web and e-commerce applications, such as at Facebook, Amazon [17] Criteo [5], Uber [29] and Booking.com [6]. Such models can be used for personalization purposes [8] since we can use the estimations to decide if a customer should be treated or not. They are also used to target a specific segment of the customer base with a promotional offer or other types of marketing campaigns.

2.2 Budget Constrained Allocation

Our goal in promotion targeting is to maximize the incremental effect on sales while complying with a budget constraint. This optimization problem takes the form of a 0-1 knapsack problem [25]. In previous work [6] we addressed the causal estimation and binary promotion allocation problem, where we introduced the *Retrospective Estimation* technique. The proposed solution presents a dynamic system, similar to the Online Knapsack problem [16], that can decide whether a promotion should be assigned to a customer based on the CATE estimations. It allows dynamic calibration based on the overall measured impact without harming the experience of an individual customer [9]. This framework addresses the fact that a promotional offer can result in both a positive and negative incremental net revenue loss and thus introduces a knapsack problem with negative weights. Similar work on cost-sensitive uplift decision making was done on synthetic [21] and real [18] promotional marketing use-cases. Such solutions often address a single-offer treatment, which need to account for a direct comparison between two alternatives - *to treat or not to treat?*.

In a case of multiple promotions, the incremental comparison is not trivial - Should the incremental uplift be compared to zero-baseline or the second-best option? A recent work tackled the multiple-promotions uplift problem with novel meta-learning techniques [29]. Another work suggested a constrained optimization solution [17], tackling the limited marketing budget problem and assuming a constant cost per promotion. However, such solutions are limited in a realistic e-commerce setup, where the decision about promotion selection under up-to-date budget constraints needs to be made online. One possible solution for this online challenge could be achieved using constrained Bandit solutions [15, 20, 26]. Such methods might limit the ability to perform incremental estimations, such as Return on Investment (ROI) measurements and do not account for incremental negative effects on value or weight.

A potential solution for a two-steps estimation and optimization approach can arise from the multiple-choice knapsack problem (MCKP), specifically, the online MCKP [30, 31]. Similarly to the described case, the optimization goal is to pick a single promotion at each decision point (i.e., customer visit), given both the value and weight quantities. Recent work [1, 14] solve the promotions allocation in a similar manner, however the underlying On-MCKP dual-primal method does not account for incremental causal estimations of value and cost that may result in negative weights.

3 PROBLEM FORMULATION

In this work, we address the personalized assignment of promotional offers on an online e-commerce platform. The optimization target is to maximize the overall incremental number of customers completing a purchase. We are allowed to pick at most one promotion to offer each customer from a finite set of eligible promotions A global budget constrains the overall incremental net revenue loss generated by the promotions.

3.1 Treatment Effect Estimation

For each customer i in the customers set U we define Y_i - a binary random variable representing a completion of a purchase and R_i - a continuous random variable representing the net monetary revenue associated with the purchase (sum of all revenues minus all promotional costs). We adopt the Potential Outcomes framework [12], which allows us to express the causal effects of the promotions on these two variables. $Y_i(k)$ for a customer i and a promotion k represents the potential purchase if customer i is offered the promotion k, while $R_i(k)$ represents the potential net revenue if customer i is offered the promotion k. Likewise, $Y_i(0)$ and $R_i(0)$ represent the potential outcomes if no promotion is offered to customer i, an option that is always available for every customer. We can thus define the conditional average treatment effect on Y_i and R_i for a customer with pre-promotion covariates x as follows:

$$CATE_Y(i,k) = \mathbf{E}(Y_i(k) - Y_i(0) \mid X = x_i)$$

$$CATE_R(i,k) = \mathbf{E}(R_i(k) - R_i(0) \mid X = x_i)$$

$CATE_Y(i,k)$ represents the incremental effect on the expected purchase probability of customer i if presented with promotion k, while $CATE_R(i,k)$ represents the incremental effect on the expected net

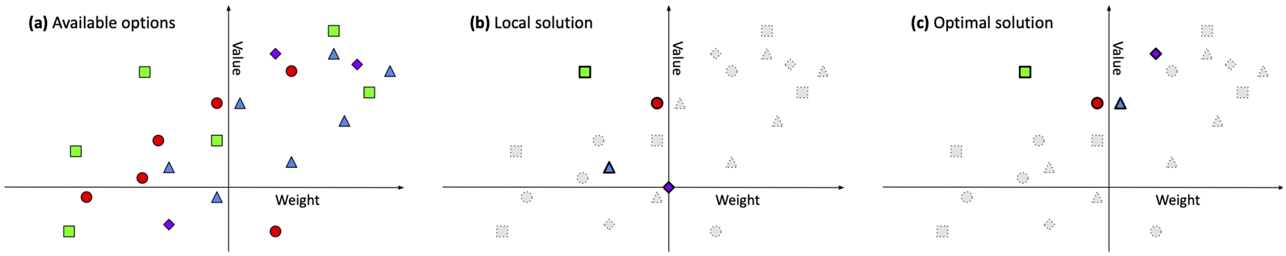


Figure 2: Toy example of promotions assignment. (a) presents all available promotion options of 4 customers (colors and shapes) on value and weight axes; (b) presents a local per-customer solution (colored); (c) presents the global optimal selection.

revenue. We assume that both $CATE_Y(i,k)$ and $CATE_R(i,k)$ can be either positive or negative. The conditional treatment effect on the expected net revenue loss $\mathcal{L}$ is defined as:

$$CATE_{\mathcal{L}}(i,k) = -CATE_R(i,k)$$

For each customer i and each promotion $k \in K_i$ we estimate $CATE_Y(i,k)$ and $CATE_{\mathcal{L}}(i,k)$ using uplift modeling on data gathered from randomized controlled trials, similarly to our previous work [6]. The base item $k = 0$ for which $CATE_Y(i,0) = 0$ and $CATE_{\mathcal{L}}(i,0) = 0$ represents no promotion. The optimization goal is to choose for every customer i a single element $k^* \in K_i$ in order to maximize the sum of all the selected $CATE_Y(i,k^*)$, while the sum of selected $CATE_{\mathcal{L}}(i,k^*)$ does not exceed the budget constraint C.

3.2 Multiple-Choice Knapsack Problem

The presented problem can be formalized as shown in Equation 1. Here, $Z_{i,k}$ is a binary assignment variable indicating whether a customer i is offered the promotion k or not. This formulation matches the Multiple-Choice knapsack problem (MCKP) [23], an extension of the 0-1 knapsack problem. In our case, the value of each item (a promotion k offered to a specific customer i) v_{ik} is $CATE_Y(i,k)$ and the weight of each item w_{ik} is $CATE_{\mathcal{L}}(i,k)$. Different from the classical setup, we allow the weights and values of the items to be negative. For practical applications, we investigate the Online-MCKP [30] variation, where customers arrive one-by-one and the decision on allocation is done in an online manner.

$$\begin{aligned}
&\textbf{Maximize} \sum_{i \in U} \sum_{k \in K_i} CATE_Y(i,k) \cdot Z_{ik} \\
&\textbf{subject to:} \\
&1.\ \sum_{i \in U} \sum_{k \in K_i} CATE_{\mathcal{L}}(i,k) \cdot Z_{ik} \le C \qquad \forall i \in U, k \in K_i \\
&2.\ \sum_{k \in K_i} Z_{ik} = 1 \qquad \forall i \in U \\
&3.\ Z_{ik} \in \{0,1\} \qquad \forall i \in U, k \in K_i
\end{aligned} \tag{1}$$

3.3 Problem Toy Example

We demonstrate the optimization problem by illustrating a toy example on Figure 2. In this example, we observe various promotional offers for four different customers (green-squares, red-circles, blue-triangles, and purple-rhombuses) given a budget of zero. In other words - we need to pick one promotion per customer, such that the total weight (expected net revenue loss $\mathcal{L}$) will not be positive. For each promotion k we present its value $CATE_Y(i,k)$ and weight $CATE_{\mathcal{L}}(i,k)$ on a two-dimensional chart, as shown in sub-figure (a). We observe promotions in all four quadrants of the axes, representing both positive and negative expected value (y-axis) and weight (x-axis). Sub-figure (b) presents a local solution to the problem - for each customer, we pick the promotion with the highest value and non-positive weight. This solution results in picking promotions from one quadrant of the axes only - where the value is positive, and weight is negative. It is important to note that there is no purple option in the quadrant, and therefore we defaulted for the base $(0,0)$ solution - where we do not offer any promotion to the customer. Sub-figure (c) presents the optimal solution to the problem. In this case, we also pick promotions with positive weight for the blue and purple customers since their weight is compensated with the negative weight of the selected green and red promotions. We can observe that the total value (overall position of selected options on the y-axis) is higher than any other possible combination within the budget constraints. While it is easy to identify the optimal combination in this toy example, in reality the promotion sets are bigger, and the data arrive in an online manner.

4 SOLUTION FRAMEWORK

We address the problem with a two-steps approach: an estimation step and an optimization step. The first step is to estimate $CATE_Y(i,k)$ and $CATE_{\mathcal{L}}(i,k)$. The second step is to use the estimations and solve the MCKP and the Online-MCKP.

4.1 CATE Estimation

CATE estimation for both Y and R are achieved using uplift modeling and data from randomized controlled trials. We consider different estimators such as two-models [10], transformed outcome [2] and X-learner [13]. Common evaluation metrics such as Qini Curves and Qini Score are used for model selection. The step results in estimations of $CATE_Y(i,k)$ and $CATE_{\mathcal{L}}(i,k)$ for every customer i and every promotion k. These quantities will serve as the input item sets K_i for the following optimization step.

4.2 MCKP Approximation Solution

Similar to the 0-1 knapsack problem, we are interested in an approximation solution to overcome the limitations of computational complexity and lack of fitness for online environments of linear programming solutions. Previous work [31] describes an MCKP approximation solution based on Lueker's algorithm [16]. We extend the solution and allow for negative values, negative weights, and possible negative budget constraints, which are essential for our business case and problem formulation. We suggest a four-steps solution that (1) eliminates dominated items; (2) calculates incremental values and weights to allow comparison between items; (3) transforms the incremental value-weight quantities into efficiency angles to allow efficiency-sorting for both positive and negative values-weights; (4) selects a single item according to an efficiency angle threshold, designed to meet the capacity constraints.

4.2.1 Dominant items. Items' dominance plays an important role in solving the MCKP, since it allows to disqualify items that would not be included in an optimal solution. Given an items set K_i we say that an item b is dominated by a if $w_{ia} \leq w_{ib}$ and $v_{ia} > v_{ib}$. Similarly, we say that b is LP-dominated by items a and c if b is dominated by a convex combination of a and c. Thus, if $w_{ia} \leq w_{ib} \leq w_{ic}$ and $v_{ia} > v_{ib} > v_{ic}$, then b is LP-dominated by a,c if:

$$\frac{v_{ic} - v_{ib}}{w_{ic} - w_{ib}} \geq \frac{v_{ib} - v_{ia}}{w_{ib} - w_{ia}}$$

The dominant items of K_i are those that are not dominated or LP-dominated. Dominated items are not expected to be part of an optimal solution since there are items with a higher value and lower weight. The approximation solution begins by identifying the set of dominant items $D_i \subseteq K_i$. The dominant items D_i form the upper-left convex hull of K_i as illustrated in Figure 3, also known as the Pareto efficiency front. The dominant items can be found in $O(|K_i| \cdot log|K_i|)$ time complexity as described in [23].

4.2.2 Incremental value and weight. For each dominant item $d \in D_i$, we compute its incremental value and weight $(\overline{v}_{id}, \overline{w}_{id})$. Incremental quantities represent the extra value and extra weight added to the knapsack by choosing item d instead of item $d-1$. This transformation is needed to build a solution that relies on a single efficiency threshold. To compute the incremental quantities we sort the dominant items by increasing weight and proceed as follows:

$$\overline{w}_{id} = \begin{cases} w_{id} & if \;\; d = 0 \\ w_{id} - w_{id-1} & else \end{cases} \qquad \overline{v}_{id} = \begin{cases} v_{id} & if \;\; d = 0 \\ v_{id} - v_{id-1} & else \end{cases}$$

Adding all the incremental values and weights $\{(\overline{v}_{id}, \overline{w}_{id})\}$ up to item $d = d'$ to the knapsack is equivalent to adding the original value and weight $(v_{id'}, w_{id'})$ of item d'. Another interesting property is that for all $d \geq 1$, the incremental efficiency $\frac{\overline{v}_{id}}{\overline{w}_{id}}$ is monotonically decreasing with d. The property does not hold between $d = 0$ and $d = 1$ due to negative weights and values, opening the possibility that for $k = 0$ the incremental efficiency might be negative. This discontinuity breaks the algorithm proposed in [31] and requires us to make a further adaptation in the solution procedure - the efficiency angle.

4.2.3 Efficiency angle. Next, we compute the efficiency angle:

$$\theta_{id} = \begin{cases} \frac{3\pi}{2} & if \;\; \overline{v}_{id} = 0 \;\wedge\; \overline{w}_{id} = 0 \\ 2\pi + atan2(\overline{v}_{id}, \overline{w}_{id}) & if \;\; \overline{v}_{id} < 0 \;\wedge\; \overline{w}_{id} \leq 0 \\ atan2(\overline{v}_{id}, \overline{w}_{id}) & else \end{cases}$$

The efficiency angle θ_{id} is the angle between the incremental quantities $(\overline{v}_{id}, \overline{w}_{id})$ and the positive X-axis as shown in Figure 3(bottom). The higher the efficiency angle, the smaller the weight of the respective dominant item, a property needed for creating an efficiency angle function as explained in the next section.

4.2.4 Efficiency angle threshold. The core of the solution method is the efficiency angle function. Its role is to map an efficiency angle θ to the expected weight of the dominant items with an efficiency angle above θ. The function encodes the distribution of the dominant items from customers we already encountered. We can use that information to decide which promotion to pick for future customers, given our updated budget constraints. The idea behind this function is to allow us to pick an efficiency angle threshold, such that the sum of all the expected weights of future dominant items above the threshold is equal to the remaining knapsack capacity. We create the efficiency angle function by taking all dominant items from already seen customers into an item set P. We sort the items $p \in P$ by decreasing efficiency angle θ_p and calculate the efficiency angle function as the cumulative average of items' weight:

$$f(\theta_0) = w_0/|P|$$
$$f(\theta_p) = f(\theta_{p-1}) + w_p/|P|$$

Therefore f is a piece-wise function and can be represented as a list of pairs $\{\theta_p, f(\theta_p)\}$. The algorithm to create the efficiency angle function is described in Algorithm 1.

Using the efficiency angle function f we find an approximate MCKP solution. Given a knapsack capacity C and efficiency angle function f, we retrieve the efficiency angle threshold $\theta^*(i)$ as follows:

$$\theta^*(i) = min_{p \in P} \left\{ \theta_p \mid f(\theta_p) \leq \frac{C}{\frac{|P|}{i} \cdot (|U| - i + 1)} \right\}$$

Here, $|U|$ is the number of customers, i is the current iteration and $|P|/i$ is the average number of dominant items per customer, based on previous data. We use the efficiency angle threshold θ^* to find the dominant item on K_i whose efficiency angle is minimal and greater than θ^*. Intuitively, selecting such item will achieve the best value, given the capacity constraints. Picking an item with a smaller angle is not expected to meet the constraint, while picking

Algorithm 1 Efficiency Angle Threshold

Input:
- Set of past dominant items P
- Knapsack capacity C
- Current customer index i
- Expected number of customers $|U|$

Sort P by decreasing angle θ_p
for $p \in P$ **do**:
  if p=0: $f(\theta_0) = w_0/|P|$
  else: $f(\theta_p) = f(\theta_{p-1}) + w_p/|P|$
end for
Return efficiency threshold θ^*:
  $\theta^* \leftarrow min_{p \in P} \left\{ \theta_p \mid f(\theta_p) \leq \frac{C}{\frac{|P|}{i} \cdot (|U| - i + 1)} \right\}$

Algorithm 2 Online MCKP

Input:
- Customer set U
- Item sets K_i
- Knapsack capacity C

$P \leftarrow \emptyset$
for $(i \in U \mid 1 \leq i \leq |U|)$ **do**
  $D_i \leftarrow$ dominant items of K_i sorted by increasing weight
  for $d \in D_i$ **do**
    Compute incremental values and weights $(\overline{v}_{id}, \overline{w}_{id})$:
    if d=0: $\overline{w}_{i0} \leftarrow w_{i0}$; $\overline{v}_{i0} \leftarrow v_{i0}$
    else: $\overline{w}_{id} \leftarrow w_{id} - w_{id-1}$; $\overline{v}_{id} \leftarrow v_{id} - v_{id-1}$
    Compute efficiency angle θ_{id}:

$$\theta_{id} = \begin{cases} \frac{3\pi}{2} & if \quad \overline{v}_{id} = 0 \ \wedge \ \overline{w}_{id} = 0 \\ 2\pi + atan2(\overline{v}_{id}, \overline{w}_{id}) & if \quad \overline{v}_{id} < 0 \ \wedge \ \overline{w}_{id} \leq 0 \\ atan2(\overline{v}_{id}, \overline{w}_{id}) & else \end{cases}$$

    $P \leftarrow P \cup (\theta_{id}, w_{id})$
  end for
  Get updated efficiency threshold θ^*:
    $\theta^* \leftarrow \mathcal{A}lgorithm1\ (P, C, i, |U|)$
  Find dominant item d^*:
    $d^* \leftarrow \text{argmin}_{d \in D_i} \{\theta_{id} \mid \theta_{id} \geq \theta^*\}$
  Update capacity:
    $C \leftarrow C - w_{id^*}$
  Pick item d^*
end for

an item with a higher angle will achieve a sub-optimal value. A full description of the overall method is presented on Algorithm 2.

This algorithm has an $O(|K_i| \cdot log|K_i|)$ runtime complexity *per evaluated customer* [23] and is applicable in an online manner. The threshold update has an $O(|U|)$ complexity, given that the list of dominant items P is maintained in a sorted data-structure. However, the efficiency angle function update can be executed separately from the real-time calculation, often in a batch-manner.

4.3 Solution Toy Example

We demonstrate the first steps of the solution method for a single customer on a toy example in Figure 3. At the top figure, we can see nine promotion items represented as $(value, weight)$ pairs. Promotions 1 to 5 are the dominant items, forming an upper-left convex hull. We can see that items 6, 7, and 9 are dominated while item 8 is LP-dominated. Dominant items 1 to 5 are sorted by increasing weight. The bottom figure shows the respective incremental values and weights of the dominant items. The efficiency angle θ starts at the positive weight axis and increases counter-clockwise. We can see that item 1 has the greatest efficiency angle while item 5 has the smallest. This property results from deriving the incremental quantities from dominant items that are sorted by increasing weight.

5 EXPERIMENTAL STUDY

5.1 Randomized Controlled Trial

Prior to performing promotions assignment optimization, we conducted a randomized controlled trial (RCT) to assess the potential impact of different promotions. The experiment took the form of an online multi-variant A/B test, in which different treatment groups were offered different promotions while the control group was offered with no promotion. The experiment was conducted on real traffic of Booking.com website and lasted for several weeks. In our experiment, the customers in the treatment groups received three different levels of discounts on Booking.com products. The target metrics - completing a purchase, promotion cost, and incremental revenue - were aggregated and compared between the control and treatment groups, producing an estimation of the average treatment effect on purchase completion and net revenue per treatment. The experiment showed a conclusively positive treatment effect on purchases for all three promotions but with a conclusive net revenue loss. Such results signal that all the promotion versions require a budget to operate and are not self-sufficient. We are interested in a promotions personalization solution that will allow a zero-budget promotional campaign to operate in the long term.

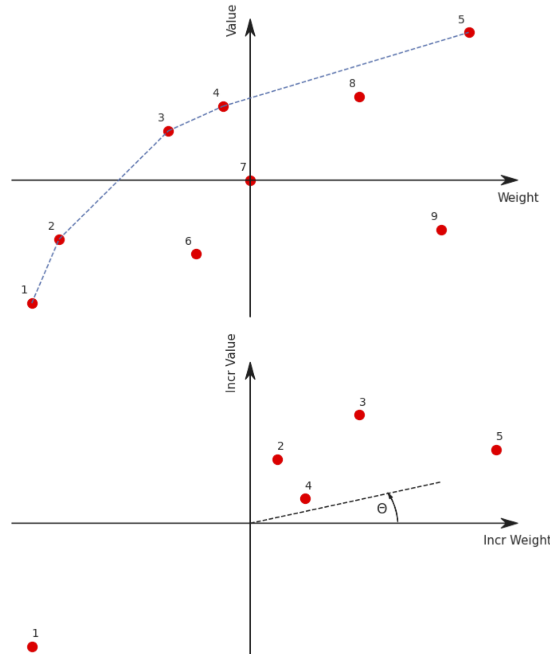


Figure 3: Dominant items selection (top) and their respective incremental items (bottom). Efficiency angle is in counter-clockwise direction from the incremental weight axis.

5.2 Uplift Modeling on Experimental Data

The RCT resulted in a dataset of more than 20 million entries. Each data point was represented by the binary variable Y indicating the completion of a purchase, the continuous variable R indicating the total net revenue (including the promotional costs), and covariates X filled with the customer characteristics. We selected the best model from a number of uplift modeling techniques in order to obtain $CATE_Y(i, k)$ and $CATE_{\mathcal{L}}(i, k)$. The models were trained on a portion of the dataset, and the predictions were made on another disjoint portion. The training on high-scale required a dedicated implementation of uplift models for Hadoop ecosystem. The modeling code, including additional implementation of evaluation metrics, was open-sourced as a public **UpliftML** package [24].

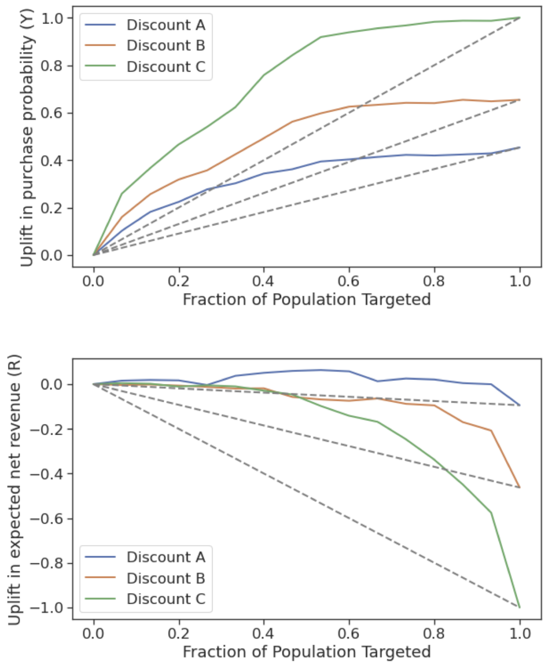


Figure 4: Qini curves of uplift modeling on Y (top) and R (bottom) for 3 discount types (normalized axes)

The best modeling technique was chosen by the highest Qini Score on the test set. The normalized Qini plots on the test set for both Y and R are presented on Figure 4. We observe that each discount has a significant effect when treating 100% of the population and that the higher the positive effect on Y, the higher the negative effect on R. We can also see that our models successfully rank the customers by predicted $CATE$ on both Y and R. This is evidenced by the shape of the Qini curves and the area below the curve compared to the random assignment line (Qini Score).

5.3 Datasets

We tested our method on the experimental, synthetic and publicly available datasets, which are described in Table 1.

5.3.1 Experimental data. Following the modeling in subsection 5.2, we generated a sub-sampled dataset of 200,000 customers. For each customer i and each potential promotion k we extrapolated the expected value ($CATE_Y(i,k)$) and the expected weight ($CATE_{\mathcal{L}}(i,k)$). This resulted in $|U| \times K_i$ rows, where the number of treatments per customer $m_i = 4 \; \forall i \in U$ (three promotion levels and a no-discount treatment) as described in the first row (Discounts) in Table 1. The joint and the marginal distributions of value and weight across three discount levels are depicted on a normalized scale in the bottom part of Figure 5. We observe items in all four quadrants of the chart, with a vast majority in the first quadrant - representing positive values weights for most customers.

Table 1: Evaluated datasets' properties

Dataset	Number of treatments	Number of customers	Source
Discounts	4	200,000	Online experiment
sim5k9	9	5,000	Simulation
sim10k9	9	10,000	Simulation
sim20k9	9	20,000	Simulation
sim30k9	9	30,000	Simulation
sim50k9	9	50,000	Simulation
sim100k9	9	100,000	Simulation
Hillstrom	3	64,000	Public dataset [22]

5.3.2 Synthetic data. We simulated data of a promotional campaign, offering nine levels of discounts between 0 to 40% in steps of 5% for different population sizes of 5-100 thousands customers. For each customer i and each promotion k we computed the expected value (conversion uplift) and expected weight (incremental net revenue loss). The conversion uplift was randomly sampled from a

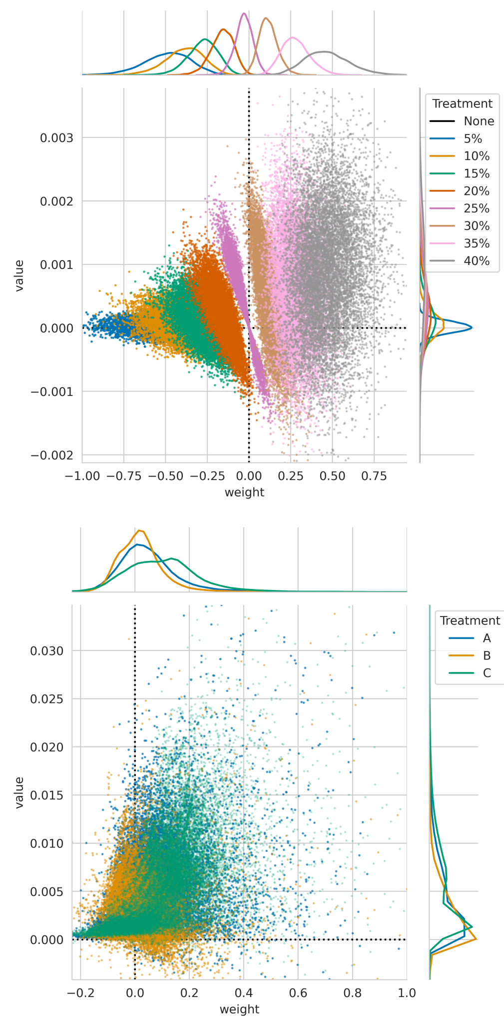


Figure 5: Distribution of value and weight of the simulated 10K dataset (top) and real discounts dataset (bottom)

normal distribution with $\mu = A * D^2$ and $\sigma^2 = S * D^2$, where A and S are global constants fitted by maximum likelihood estimators (MLE) from real data, and D is the respective discount level. We assume a positive correlation between the discount level and the conversion uplift, with a diminishing marginal effect.

The net revenue uplift was sampled from a normal distribution with $\mu = P*(C-D)$ and $\sigma^2 = S_p$ multiplied by $1+CATE_Y(i,k)$. P, C and S_p represent MLEs of price, commission and revenue standard-deviation respectively. The $(C-D)$ multiplier is the commission loss of the discount and $[1 + CATE_Y(i,k)]$ is the positive impact of conversion uplift. The net revenue loss has a positive correlation with the discount level but can result in negative quantities (profit) due to conversion uplift. Figure 5(top) depicts the distributions of value and weight of the dataset for 10,000 customers and nine discount levels(5%-40% represented by colors). We observe that higher discounts result in higher values and weights, introducing a tradeoff between the two. Moreover, we observe a higher variance with the discount level increase, similar to the experimental data.

5.3.3 Public data. We used a publicly available e-mail campaign dataset by Hillstrom [22] which is commonly used in uplift modeling literature [17]. It represents an e-mail campaign delivered to 64,000 customers. It consist of a random assignment of three treatment variations - *Men merchandise*, *Women merchandise* and *No Treatment* with information about purchase conversion and total spend. By utilizing uplift modeling, similar to subsection 5.2, we estimated the expected conversion uplift and the expected net revenue loss, which are equivalent to *value* and *weight* in our setup. While the Hillstrom dataset allows a direct application of our method, there are two fundamental differences in the nature of the data: (1) Unlike discounts that imply an associated revenue loss, e-mail campaigns do not have a direct cost factor; (2) Contrary to online web-platforms, e-mail campaigns are an offline marketing channel, and therefore the promotion assignment optimization can be pre-computed in advance.

5.4 Assignment Optimization Methods

To evaluate the effectiveness of the suggested Online-MCKP solution, we compare online and offline optimization methods on all datasets, with a zero-budget constraint, namely - the promotion campaigns are required to be self-sponsored. The suggested problem setup with negative values and weights is not feasible in the existing methods covered in the literature. Therefore such methods cannot be compared as a benchmark in our problem with negative incremental estimations. While relying on simple benchmarks, our main evaluation of the method is against the optimal solution, to understand the optimality gap. Moreover, these simple benchmark represent our previously deployed solutions, and are widely used across the industry [14]. The compared methods are:

5.4.1 Global Selection (Global). A standard solution for selecting the best flat treatment - a classical strategy assuming a near-homogeneous treatment effect. The global solution picks a single treatment, out of the possible promotion options, and suggests it to all of the customers in a non-personalized manner. The selected treatment is the one that achieves the highest value while complying the budget constraint. If none of the treatments meets the budget constraint, the solution will default to the base variant, without offering a promotion at all.

5.4.2 Local Solution (Local). The local solution solves a local optimization problem within the scope of a single customer - it picks the promotion with the highest value and non-positive weight. An example of such a solution is depicted in the toy example in Figure 2b. This basic solution relies on the individual impact estimation of each promotion per customer and can be applied online without sharing knowledge between the individual decisions.

5.4.3 Greedy Solution (Greedy). Similar to *Local*, the greedy solution solves the optimization problem within the scope of a single customer. However, it picks the promotion with the highest value and a weight within the *current capacity state*. This allows to pick items with positive-weight in cases where previously picked items increased the remaining capacity.

5.4.4 Online MCKP (On-MCKP). This method corresponds to the real-world scenario where customers arrive one at a time, and the decision of which promotion to offer must be made at each time step. At the beginning of the process, we have no information about the general weights and values distributions. The method adapts the promotion assignment decision to the remaining budget and the updated efficiency angle function as described in Algorithm 2.

5.4.5 Offline MCKP (Off-MCKP). This method uses the same underlying selection algorithm as the Online MCKP, while the items are provided in advance, allowing to fit the efficiency angle function once, prior to the assignment decision. Then the algorithm decides which promotion to offer to each customer without updating the efficiency angle function. The core of the method, the efficiency angle function, is described in Algorithm 1.

5.4.6 Integer Linear Programming (ILP). We use an ILP solver in order to find an optimal upper bound solution in an offline setup. The MCKP can be solved as an ILP using the linear formulation described in Equation 1. The budget constant C was set to zero ($C = 0$). We relied on python PuLP package [19] with CBC (Coin-or branch and cut) solver. The overall solver runtime was limited to 3-hours. On a side note, while there are more suitable solvers for knapsack problems, we only faced a single instance where the optimal solution was not reached within the limited runtime, achieving a feasible solution with an infinitesimal optimality gap.

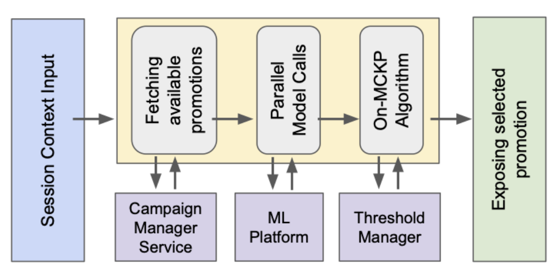
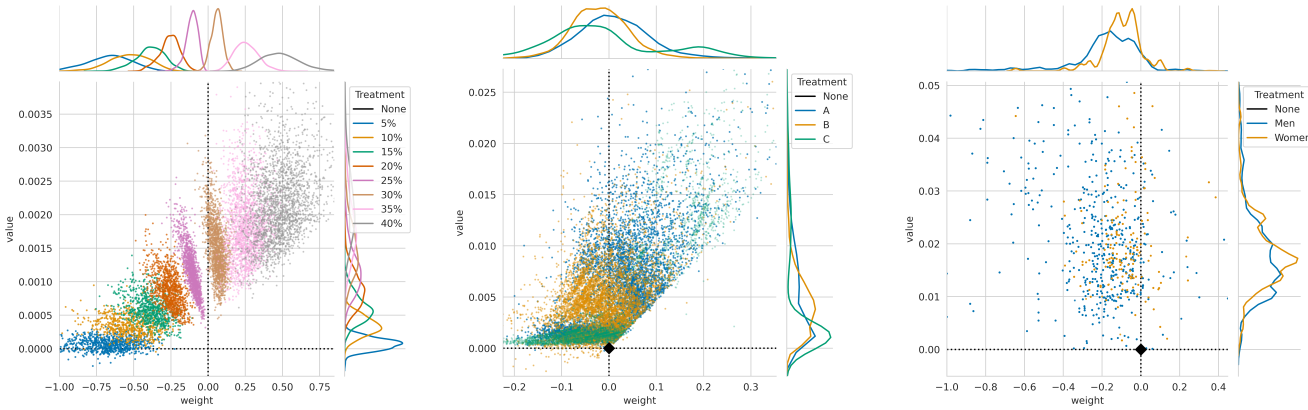


Figure 6: Multiple-Choice PPS online decision flow



(a) Simulated 10k dataset **(b) Real discounts dataset** **(c) Hillstrom dataset**

Figure 7: Selected promotions in the optimal On-MCKP solution

5.5 System Deployment

Our method was evaluated on several promotions on various products at Booking.com. The decision logic, including the uplift model, On-MCKP algorithm, threshold management and promotion selection is deployed via a dedicated Promotions Personalization System (PPS) and depicted in Figure 6. The system supports multiple optimization levers on various promotions, such as customer-targeting, multiple-choice promotions, persistency management and reward condition optimization as described in [7].

While triggered by a customer-facing system, PPS queries the promotion management system for eligible promotions in the current session. Then the incremental value and weight of each possible promotion is evaluated in-parallel via a pre-trained uplift model served by a dedicated ML-platform accounting for distributed serving and latency concerns [3]. Model results, together with an up-to-date threshold are fed into the On-MCKP algorithm, resulting in a single promotion which is presented to the customer. In order to adjust the threshold in an online manner, and retrain the uplift models, the system logs the dominant items of each iteration.

To evaluate the long-term incremental effect of a promotion and retrain the underlying models, the system allocates portions of traffic to no-treatment and flat-promotion (global) treatment lanes.

Table 2: Optimality rate of solutions (compared to optimal ILP) of evaluated methods across different datasets

	Global	Local	Greedy	On-MCKP	Off-MCKP
Discounts	0.0%	37.0%	65.79%	99.75%	>99.99%
sim5k9	16.9%	50.9%	79.09%	99.99%	>99.99%
sim10k9	25.6%	50.3%	80.42%	99.98%	99.99%
sim20k9	18.6%	46.6%	76.09%	>99.99%	>99.99%
sim30k9	26.0%	50.6%	80.59%	99.99%	>99.99%
sim50k9	26.4%	51.0%	79.94%	99.99%	>99.99%
sim100k9	25.9%	50.6%	80.42%	99.99%	>99.99%
Hillstrom	91.9%	93.3%	>99.99%	>99.99%	>99.99%

6 RESULTS

6.1 Optimization results

The complete optimization results of the *Global*, *Local*, *Greedy*, *Online* and *Offline MCKP* methods are listed in Table 2 and depicted in Figure 8. We report the optimality rate of the total value compared to the optimal ILP solution, across the datasets.

The *Off-MCKP* demonstrates a near-optimal performance with a negligible optimality gap up to $7.4{\times}10^{-5}$. The *On-MCKP* demonstrates excellent performance as well, with a maximal optimality gap of 0.245% on the real discounts dataset. We observe that the *Global*, non-personalized benchmark achieves low performance (16-26%) at the simulated instances and can not find any feasible solution for the real discounts dataset. The *Local* solution constantly outperforms the *Global*, but is limited compared to MCKP methods, delivering about 51% of the expected impact. The *Greedy* method

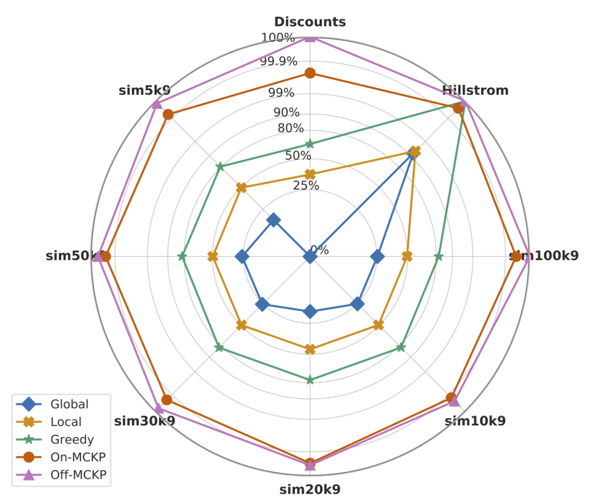


Figure 8: Achieved percentage of the optimal ILP solution across different methods and datasets. (Radial log scale)

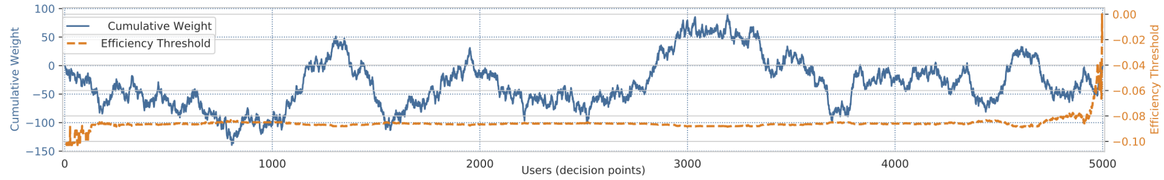


Figure 9: On-MCKP dynamics of total weight (net revenue loss) and decision threshold on Sim5K9 dataset.

results in 65%-80% optimality rates, and a near-optimal solution in Hillstorm problem, in which the capacity constraint is non-binding. The suggested MCKP solutions widely outperform the *Global*, *Local* and *Greedy* benchmarks. In the real discounts case, it plays a game-changing role in allowing the promotional campaign to become self-sponsored, with a significant improvement compared to *Local* (+170%) and *Greedy* (+51%) solutions.

6.2 Solution Properties

We observe the properties of the Online-MCKP solution on three datasets - Simulated 10K customers, Discounts and Hillstrom as depicted in Figure 7. Following the earlier visualization of the available promotions of *sim10k9* dataset in Figure 5, we present the optimal composition of promotion assignments on the same dataset in Figure 7a. We observe an interesting phenomenon where the selected promotions follow a linear value/weight slope, forming a separation bound by excluding the solutions below the line, according to efficiency threshold used by the method. Similarly, the real discounts dataset in Figure 7b forms a separation bound and depicting a clear tradeoff between value and weight. It consists of a blend of all three available promotions, using items from three possible quadrants. The Hillstrom dataset in Figure 7c has a scattered selection of promotions, since the capacity constraint is not bounding this solution. This is because the email campaign does not imply a direct revenue loss, and the optimal solution does not contain promotions with a negative values. The solutions do not include items from the fourth quadrant (negative-value/ positive-weight), since these items are dominated by the default (0,0) option. The discounts and Hillstrom solutions include the base (no-promotion) treatment (1% in Hillstrom and 19.7% in Discounts), while in the simulated dataset it was never picked over the other eight alternatives.

6.3 Online Strategies Comparison

Our On-MCKP solution was deployed and compared to offering three flat-discount levels (similar to subsection 5.1). The comparison of the normalized total transactions uplift and the incremental ROI ($\Delta Revenue \div \Delta Investment$) over several weeks is presented in Table 3. The ROI metric is used to control the long-term budget spent, when in our case a requirement of a self-sponsored campaign with zero-budget is equivalent to $ROI \geq 1$ constraint.

While the flat Discount A was the only one that is feasible to run within the budget constraint (ROI=1.13), the other discounts result in higher transaction uplift but are infeasible (ROI<1). The On-MCKP strategy converges to a feasible ROI=1.02, achieving 76% of the impact of Discount C. This is a 162% improvement compared to the second-best feasible alternative (Discount A).

Table 3: Comparison of online discount strategies

Strategy	Transactions Uplift (Normalized)	ROI	Budget Feasible
Discount A	47%	1.13*	Yes
Discount B	62%	0.55	No
Discount C	100%	0.34	No
On-MCKP	76%	1.02*	Yes

6.4 Online Dynamics

The step-by-step dynamics of On-MCKP are demonstrated on the Sim5K9 dataset through 5,000 iterations in Figure 9 (efficiency threshold in orange s and cumulative weight in blue). The cumulative weight (budget) fluctuates around the target capacity (0), exceeding the capacity constraint at certain time-steps, but quickly recovers to the zero-target. The efficiency-angle threshold has high fluctuations in the first steps, and quickly converges around $\theta = -0.085$. This angle represents the observed separation boundary in Figure 7a. The threshold changes back to near-zero levels in final rounds, since it is highly dependant on the expected weight of the remaining items. In real use-cases, we are not supposed to observe such behavior since the self-sponsored promotional campaign is expected to run for an unlimited time horizon.

7 CONCLUSION

We propose a novel solution to the *Online Constrained Multiple-Choice Promotions Personalization Problem* based on a two-steps framework with causal uplift estimation and online constrained optimization. We extend the multiple-choice knapsack problem by providing a solution for negative values and weights, allowing to incorporate causal estimations. Our experimental study found the solution highly effective and applicable for decision mechanisms in real-time e-commerce setups. The method is adaptive for dynamic environments and ensures budget constraints compliance by responding to budget offsets. While the compared benchmarks might be too simple, due to a lack of feasible solutions for negative weights in the existing state-of-the-art literature, achieving above 99.7% of the possible optimal promotional impact positions our method as a nearly optimal solution, regardless of the alternative.

The solution played a game-changing role in our business by scaling promotional campaigns. It increased the promotions' impact on sales by 162% compared to existing feasible strategies, transforming a campaign with an insufficient budget into an effective self-sponsored deployment. Our framework, alongside with the publicly open-sourced code for causal incremental estimations, lays the ground for further research on causal estimation and allocation of treatments with associated cost.

REFERENCES

[1] Meng Ai, Biao Li, Heyang Gong, Qingwei Yu, Shengjie Xue, Yuan Zhang, Yunzhou Zhang, and Peng Jiang. 2022. LBCF: A Large-Scale Budget-Constrained Causal Forest Algorithm. In *Proceedings of the ACM Web Conference 2022*. 2310–2319.

[2] Susan Athey and Guido W Imbens. 2015. Machine learning methods for estimating heterogeneous causal effects. *stat* 1050, 5 (2015), 1–26.

[3] Lucas Bernardi, Themistoklis Mavridis, and Pablo Estevez. 2019. 150 Successful Machine Learning Models: 6 Lessons Learned at Booking.com. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1743–1751.

[4] Floris Devriendt, Darie Moldovan, and Wouter Verbeke. 2018. A literature survey and experimental evaluation of the state-of-the-art in uplift modeling: A stepping stone toward the development of prescriptive analytics. *Big data* 6, 1 (2018).

[5] Eustache Diemert, Artem Betlei, Christophe Renaudin, and Massih-Reza Amini. 2018. A large scale benchmark for uplift modeling. In *KDD*.

[6] Dmitri Goldenberg, Javier Albert, Lucas Bernardi, and Pablo Estevez. 2020. Free Lunch! Retrospective uplift modeling for dynamic promotions recommendation within ROI Constraints. In *Proceedings of the 14th ACM Conference on Recommender Systems*.

[7] Dmitri Goldenberg, Javier Albert, and Guy Tsype. 2021. Optimization Levers for Promotions Personalization Under Limited Budget. In *Workshop of Multi-Objective Recommender Systems (MORS'21), in conjunction with the 15th ACM Conference on Recommender Systems, RecSys*, Vol. 21. 2021.

[8] Dmitri Goldenberg, Kostia Kofman, Javier Albert, Sarai Mizrachi, Adam Horowitz, and Irene Teinemaa. 2021. Personalization in Practice: Methods and Applications. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 1123–1126.

[9] Dmitri Goldenberg, Guy Tsype, Igor Spivak, Javier Albert, and Amir Tzur. 2021. Learning to Persist: Exploring the Tradeoff Between Model Optimization and Experience Consistency. In *Companion Proceedings of the Web Conference 2021*. Association for Computing Machinery, New York, NY, USA, 527–529.

[10] Behram J Hansotia and Bradley Rukstales. 2002. Direct marketing for multichannel retailers: Issues, challenges and solutions. *Journal of Database Marketing & Customer Strategy Management* 9, 3 (2002), 259–266.

[11] Yue He, Xiujun Chen, Di Wu, Junwei Pan, Qing Tan, Chuan Yu, Jian Xu, and Xiaoqiang Zhu. 2021. A Unified Solution to Constrained Bidding in Online Display Advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2993–3001.

[12] Guido W Imbens and Donald B Rubin. 2010. Rubin causal model. In *Microeconometrics*. Springer, 229–241.

[13] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences* 116, 10 (2019), 4156–4165.

[14] Liangwei Li, Liucheng Sun, Chenwei Weng, Chengfu Huo, and Weijun Ren. 2020. Spending Money Wisely: Online Electronic Coupon Allocation based on Real-Time User Intent Detection. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2597–2604.

[15] Romain Lopez, Chenchen Li, Xiang Yan, Junwu Xiong, Michael Jordan, Yuan Qi, and Le Song. 2020. Cost-effective incentive allocation via structured counterfactual inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 4997–5004.

[16] George S Lueker. 1998. Average-case analysis of off-line and on-line knapsack problems. *Journal of Algorithms* 29, 2 (1998), 277–305.

[17] Rahul Makhijani, Shreya Chakrabarti, Dale Struble, and Yi Liu. 2019. LORE: a large-scale offer recommendation engine with eligibility and capacity constraints. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 160–168.

[18] Alex Miller and Kartik Hosanagar. 2020. Personalized discount targeting with causal machine learning. (2020).

[19] Stuart Mitchell, Michael OSullivan, and Iain Dunning. 2011. PuLP: a linear programming toolkit for python. *The University of Auckland, Auckland, New Zealand* (2011), 65.

[20] Alessandro Nuara, Francesco Trovo, Nicola Gatti, and Marcello Restelli. 2018. A combinatorial-bandit algorithm for the online joint bid/budget optimization of pay-per-click advertising campaigns. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[21] Diego Olaya, Wouter Verbeke, Jente Van Belle, and Marie-Anne Guerry. 2021. To do or not to do: cost-sensitive causal decision-making. *arXiv preprint arXiv:2101.01407* (2021).

[22] Nicholas J Radcliffe. 2008. Hillstrom's MineThatData email analytics challenge: An approach using uplift modelling. *Stochastic Solutions Limited* 1 (2008), 1–19.

[23] Prabhakant Sinha and Andris A Zoltners. 1979. The multiple-choice knapsack problem. *Operations Research* 27, 3 (1979), 503–515.

[24] Irene Teinemaa, Javier Albert, and Nam Pham. 2021. UpliftML: A Python Package for Scalable Uplift Modeling. https://github.com/bookingcom/upliftml.

[25] Paolo Toth and Silvano Martello. 1990. *Knapsack problems: Algorithms and computer implementations*. Wiley. 14–15 pages.

[26] Long Tran-Thanh, Archie Chapman, Alex Rogers, and Nicholas Jennings. 2012. Knapsack based optimal policies for budget–limited multi–armed bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 26.

[27] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. 2018. GANITE: Estimation of individualized treatment effects using generative adversarial nets. In *International Conference on Learning Representations*.

[28] Kui Zhao, Junhao Hua, Ling Yan, Qi Zhang, Huan Xu, and Cheng Yang. 2019. A Unified Framework for Marketing Budget Allocation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

[29] Zhenyu Zhao and Totte Harinen. 2019. Uplift modeling for multiple treatments with cost optimization. In *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 422–431.

[30] Yunhong Zhou, Deeparnab Chakrabarty, and Rajan Lukose. 2008. Budget constrained bidding in keyword auctions and online knapsack problems. In *International Workshop on Internet and Network Economics*. Springer, 566–576.

[31] Yunhong Zhou and Victor Naroditskiy. 2008. Algorithm for stochastic multiple-choice knapsack problem and application to keywords bidding. In *Proceedings of the 17th international conference on world wide web*. 1175–1176.