End-to-End Cost-Effective Incentive Recommendation under Budget Constraint with Uplift Modeling

Zexu Sun
Gaoling School of Artificial
Intelligence, Renmin University of
China
Beijing, China
sunzexu21@ruc.edu.cn

Hao Yang
Gaoling School of Artificial
Intelligence, Renmin University of
China
Beijing, China
yanghao@ruc.edu.cn

Dugang Liu*
Guangdong Laboratory of Artificial
Intelligence and Digital Economy (SZ)
Shenzhen, China
dugang.ldg@gmail.com

Yunpeng Weng
FiT, Tencent
Shenzhen, China
edwinweng@tencent.com

Xing Tang*
FiT, Tencent
Shenzhen, China
shawntang@tencent.com

Xiuqiang He
FiT, Tencent
Shenzhen, China
xiuqianghe@tencent.com

ABSTRACT

In modern online platforms, incentives (*e.g.*, discounts, bonus) are essential factors that enhance user engagement and increase platform revenue. Over recent years, uplift modeling has been introduced as a strategic approach to assign incentives to individual customers. Especially in many real-world applications, online platforms can only incentivize customers with specific budget constraints. This problem can be reformulated as the multi-choice knapsack problem (MCKP). The objective of this optimization is to select the optimal incentive for each customer to maximize the return on investment (ROI). Recent works in this field frequently tackle the budget allocation problem using a two-stage approach. However, this solution is confronted with the following challenges: (1) The causal inference methods often ignore the domain knowledge in online marketing, where the expected response curve of a customer should be monotonic and smooth as the incentive increases. (2) There is an optimality gap between the two stages, resulting in inferior sub-optimal allocation performance due to the loss of the incentive recommendation information for the uplift prediction under the limited budget constraint. To address these challenges, we propose a novel <u>E</u>nd-to-<u>E</u>nd Cost-<u>E</u>ffective <u>I</u>ncentive <u>R</u>ecommendation (E^{3}IR) model under the budget constraint. Specifically, our methods consist of two modules, i.e., the uplift prediction module and the differentiable allocation module. In the uplift prediction module, we construct prediction heads to capture the incremental improvement between adjacent treatments with the marketing domain constraints (*i.e.*, monotonic and smooth). We incorporate integer linear programming (ILP) as a differentiable layer input in the differentiable allocation module. Furthermore, we

conduct extensive experiments on public and real product datasets, demonstrating that our E^{3}IR improves allocation performance compared to existing two-stage approaches.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Applied computing** → **Economics**.

KEYWORDS

End-to-End Optimization, Incentive Recommendation, Budget Constraint, Uplift Modeling

1 INTRODUCTION

With the development of online platforms, online marketing has become increasingly essential and competitive [19]. Assigning customer discounts or bonuses is a critical strategy for promoting user conversion and increasing revenue. For instance, Taobao utilizes coupons to enhance user activity [18], Booking employs promotions to improve user satisfaction [3], and Meituan uses cash bonuses to stimulate user retention [32]. In an online marketing scenario, recommending a more significant incentive to customers can increase the purchase probability. It's reasonable to hypothesize that an increased incentive is unlikely to change spending behavior significantly. However, the incentive recommendation is often constrained by a limited budget, which means that only a portion of individuals can receive incentives. Therefore, the main challenge is to convert more users and generate higher revenue within the budget constraints. This problem is typically formulated as a budget allocation problem in online marketing [41].

Several recent studies have tackled this problem using a two-stage approach [3, 39, 40]. In the first stage, causal inference methods estimate uplift, and in the second stage, an integer programming formulation finds the optimal allocation based on the predicted uplift. Zhao et al. [37] employ a logit response model [26] to forecast

treatment effects and subsequently determine the optimal allocation by utilizing root-finding to satisfy the Karush-Kuhn-Tucker (KKT) conditions. Tu et al. [31] introduce several advanced estimators, including Causal Tree [7], Causal Forest [5], and Meta-Learners [17], to estimate the heterogeneous effects. They also regard the second stage as an optimization problem. There are also some works that utilize policy to address the problem of budget allocation [34, 36, 40, 41]. Xiao et al. [34] and Zhang et al. [36] have developed reinforcement learning solutions utilizing constrained Markov decision processes to learn an optimal policy directly. Zhou et al. [40] expand on the concept of decision-focused learning to accommodate multi-treatments and devise a loss function for learning decision factors for MCKP solutions.

However, there are still some limitations to these methods. For the two-stage methods, firstly, the existing causal inference methods lack interpretability and do not conform to domain knowledge, which may result in unreliable predictions in practical scenarios. As shown in Figure 1, the response curve of a user should be monotonic and smooth. For example, customers who use coupons of varying values typically yield distinct transaction revenues, with larger coupons generally resulting in higher revenues. Thus, without incorporating the marketing domain knowledge, we may get a wrong uplift prediction of the user, which results in a non-optimal incentive recommendation; Secondly, there is an optimality gap between the two stages, as their objectives are fundamentally different. For example, with a significantly low total budget, an effective allocation algorithm should distribute incentives solely among a small subset of users who exhibit high sensitivity to incentives. In such cases, a well-trained uplift model might exhibit worse marketing effectiveness if the model demonstrates higher precision across all users on average but performs inadequately for this subset of users. For the policy-based methods [20], learning complex policies solely through a model-free black-box approach, without exploiting causal information, may lead to sample inefficiency.

To solve the limitations above, in this paper, we propose an innovative End-to-End Cost-Effective Incentives Recommendation (E^3IR) model to address these limitations. Our model consists of two modules: the uplift prediction module and the differentiable allocation module. In particular, the uplift prediction module employs a reparameterization-based multi-head structure for response prediction, ensuring monotonicity and smoothness by constraining the output to be non-negative and sharing model parameters between incremental improvement prediction heads. To mitigate the optimality gap, the differentiable allocation module tackles the budget allocation problem from an end-to-end perspective using a backward pass for the Integer Linear Programming (ILP) problem. This module can learn both value terms and constraints of the ILP, enabling universal combinatorial expressivity. Additionally, we conduct extensive experiments on various datasets to evaluate the effectiveness of our E^3IR model. Our contributions can be summarized as follows:

- We propose a novel E^3IR model for the end-to-end optimization of the budget allocation problem.
- We design an uplift prediction module injected with the marketing domain knowledge, which can ensure the user response curve is monotonic and smooth.
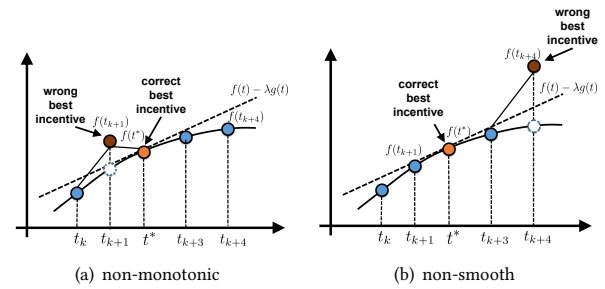

(a) non-monotonic   (b) non-smooth

Figure 1: An example of the common bad cases of the user response curve. $f(t)$ **and** $g(t)$ **are a user's response function and cost function, respectively.** t^* **is the expected best incentive level, which satisfies the** $f(t^*) - \lambda g(t^*) > f(t_i) - \lambda g(t_i), \forall t_i \neq t^*$. **In the two cases, we may find the wrong best incentive level** t_j **(i.e.,** t_{k+1} **in (a) and** t_{k+4} **in (b)) because of** $f(t^*) - \lambda g(t^*) < f(t_j) - \lambda g(t_j), \forall t_j \neq t^*$.

- We employ the differentiable ILP layer for the budget allocation, which can mitigate the performance gap in the two-stage methods.
- We conduct extensive experiments on a public dataset and a production dataset, and the results demonstrate the superiority of our method.

2 RELATED WORK

2.1 Budget Allocation

In recent years, several budget allocation methods have been proposed to address the issue of online marketing. Previous studies often utilize heuristic methods to determine the optimal incentive for users, considering predicted uplift [13, 38, 39]. However, the lack of a well-defined formulation of the optimization problem may restrict the effectiveness of these methods in achieving the marketing objective. In recent years, the most popular way of allocating budget is the two-stage methods [1, 3, 12, 37], for the first stage causal inference methods are used to predict the treatment effects. For the second stage, the integer programming is invoked to find the optimal allocation. Makhijani et al. [21] introduces the marketing goal as a Min-Cost Flow network optimization problem to enhance expressiveness. Albert et al.[3] and Ai et al. [1] employ the Multiple Choice Knapsack Problem (MCKP) to model the discrete budget allocation problem, and they propose efficient solvers based on Lagrangian duality. Despite the effectiveness of the above methods, solutions derived from these two-stage methods may be suboptimal due to misalignment between their objectives. There are also some works that examine the policy to solve this problem. Du et al. [11] and Zou et al. [41] proposed the direct learning approaches for determining the ratios between values and costs in a binary treatment setting, where treatments are first applied to users with higher scores. Zhou et al. [40] proved that the proposed loss in the studies by [11, 41] is unable to achieve the correct rank when it converges The method presented in [12] is limited in applicability as it assumes a strict budget constraint of ROI ≥ 0, which lacks

flexibility for numerous marketing campaigns on online platforms. However, the proposed method relies on the assumption of diminishing marginal utility and a presumption that is frequently not strictly upheld in practical scenarios. Our work focuses on the uplift prediction with domain knowledge, which imposes constraints on the output of user responses to be monotonic and smooth while optimizing budget allocation through an end-to-end approach.

2.2 End-to-End Optimization

End-to-end optimization is a crucial technique for mitigating the performance gap in the two-stage "predict + optimize" problem. Several recent studies have investigated the integration of predictive algorithms and optimization problems. In this context, a promising approach is to utilize neural networks to differentiate the optimization layers. Amos and Kolter [4] introduced a differentiable layer for Quadratic Programming (QP) optimization by differentiating the KKT conditions. Donti et al. [10] introduced the task-based end-to-end training process for QP problems. Studies on end-to-end training of Linear Programming (LP) problems have also been explored using quadratic regularization terms [33]. An interior-point approach (IntOpt) [22] is proposed with the homogeneous self-dual of LP problems to obtain backward gradients. Guler et al. [14] propose a divide-and-conquer algorithm for solving non-convex problems in the "predict + optimize" framework. Inspired by [25], we have developed tailored predictive optimization solutions for uplift modeling in online marketing, specifically targeting the situation with budget constraints with an end-to-end optimization, which has not been extensively explored in prior works.

3 PRELIMINARIES

In this work, we address the personalized assignment of incentives (treatments) on an online platform. The optimization target is to maximize the overall incremental number of customers completing a purchase. We can pick at most one incentive to offer each customer from a finite set of eligible incentives (see the example in Figure 1). A global budget constrains the overall incremental revenue generated by the incentive.

3.1 Cost-aware Uplift Modeling

Let $\mathcal{D}$ be the observed dataset with n samples, where each sample is represented as (x_i, t_i, y_i^c, y_i^r). Without loss of generality, $x_i \in \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^d$, is a d-dimensional feature vector. Similarly, y_i^c and y_i^r represent the conversion response variable and the revenue response variable, respectively. And they belong to the value spaces $\mathcal{Y}^c$ and $\mathcal{Y}^r$, which can be either binary or continuous. The treatment variable t_i takes values in the set $\mathcal{T} = \{0, 1, \ldots, K\}$, where $K \geq 2$. For example, $\mathcal{T}$ can represent different discounts.

We formalize the problem by following the Neyman-Rubin potential outcome framework [27], which enables us to express the uplift of incentive as follows. Let $y_i^c(k)$ and $y_i^r(0)$ represent the potential outcomes for user i when they receive incentive $t_i = k \in \{1, \ldots, K\}$ or when they are not treated, respectively. We use $\tau_{i,k}^c$ to denote the incremental uplift caused by a specific treatment k. Under certain assumptions [35], we can use the conditional average treatment effect (CATE) as an unbiased estimator for the uplift. CATE is defined as:

$$
\begin{aligned}
\tau_{i,k}^c &= \mathbb{E}\left(y_i^c(k) - y_i^c(0) \mid x_i\right), \\
\tau_{i,k}^r &= \mathbb{E}\left(y_i^r(k) - y_i^r(0) \mid x_i\right).
\end{aligned}
\tag{1}
$$

where $\tau_{i,k}^c$ represents the incremental effect on the expected conversion probability of user i with assigned treatment $t = k$. $\tau_{i,k}^r$ represents the incremental effect on the expected revenue of user i with assigned treatment $t = k$. Since most marketing incentives have a positive effect on the user response, thus we have $\tau_{i,k}^c \geq 0$ and $\tau_{i,k}^r \geq 0, \forall k \in \{1, \ldots K\}$. Specially, $k = 0$ represents the user receives no incentive, then $\tau_{i,k}^c = 0$ and $\tau_{i,k}^r = 0$.

Specially, most research studies treat the cost of implementing the treatment as fixed. When considering incentives, researchers typically use the fixed coupon value as the cost of the incentive. However, this formulation fails to accurately account for the incremental profit generated by the behavioral changes induced by the incentive. For instance, while two users might exhibit the same incremental effect, one could have a higher usage level and contribute a more significant incremental profit. This aspect can only be captured by measuring the treatment's impact on cost [11].

3.2 Budget Allocation Problem

3.2.1 Binary Treatment Budget Allocation. The binary treatment budget allocation problem is to assign the binary treatment to part of the individuals to maximize the overall revenue on the platform, but requires that the incremental cost does not exceed a limited budget B. Let the decision variables be the $z_i \in \{0, 1\}$. Therefore, this problem can be formulated as an integer programming problem.

$$
\begin{aligned}
&-\min \sum_i \tau_i^r z_i \\
&\text{s.t.} \ \sum_i \tau_i^c z_i \leq B \\
&z_i \in \{0, 1\}, \quad i = 1, \ldots, n.
\end{aligned}
\tag{2}
$$

we take an equivalent transformation from max to $-$ min for fitting the standard 0-1 knapsack problem,

3.2.2 Multi-Treatment Budget Allocation. The goal of this optimization task, which is an extension of the binary treatment budget allocation, is to select a single incentive k^* for each user i to maximize the sum of the selected $\tau_{i,k}^c$ values. However, the total sum of the selected $\tau_{i,k}^c$ should not exceed the budget constraint B.

Refer to the binary treatment budget allocation, the budget allocation in the multi-treatment scenario can be formulated as the MCKP:

$$
\begin{aligned}
&-\min \quad \sum_{i=1}^{I} \sum_{k=1}^{K} \tau_{ik}^r z_{ik} \\
&\text{s.t.} \quad \sum_{i=1}^{I} \sum_{k=1}^{K} \tau_{ik}^c z_{ik} \leq B \\
&\quad\quad \sum_{k=1}^{K} z_{ik} = 1, \quad i = 1, \ldots, n, \\
&\quad\quad z_{ik} \in \{0, 1\}, \quad i = 1, \ldots, n, \quad k = 1, \ldots, K.
\end{aligned}
\tag{3}
$$

where z_{ik} is a binary assignment variable indicating whether a user i is provided the k-th incentive or not. τ_{ik}^r represents the response uplift, τ_{ik}^c represents the cost uplift.

4 METHODOLOGY

At a high level, our E^3IR consists of two modules: the uplift prediction module and the differentiable allocation module. The first module is to predict uplifts in the user cost and response. Next, the second module utilizes the predicted response uplift and cost uplift to facilitate budget allocation through a differentiable optimization procedure based on an allocation objective. The whole structure of our E^3IR is shown in Figure 2, and we introduce the two modules in the following.

4.1 Uplift Prediction Module

In principle, uplift prediction for both $\tau_{i,k}^c$ and $\tau_{i,k}^r$ are achieved using uplift modeling and data from randomized controlled trials. To achieve the aim of this module, the previous two-stage methods use any off-the-set uplift modeling methods, such as CFRNet [29] and DragonNet [30], EUEN [15] etc. Different from them, we inject the marketing domain knowledge (*i.e.*, monotonicity, and smoothness) to design a new uplift prediction structure. Specifically, we formulate two constraints on the uplift prediction, *i.e.*, monotonic constraint and smooth constraint.

4.1.1 Monotonic Constraint. Obviously, the incentive is a direct discount on the order amount in our problem setting and does not include personalized creative content. Therefore, the higher the amount, the more attractive it is for user conversion. However, without the unique design for the model structure, we cannot guarantee the response curve for a user i is monotonic. To ensure monotonicity, we propose a multi-treatment effect estimation model with a monotonic constraint. We implicitly model the incremental effect between two adjacent treatments and obtain the predicted uplifts through accumulation.

As described in Section 3, we use $\left(x_i, t_i, y_i^c, y_i^r\right)$ to represent a user, for the online platform, x_i includes the statistical characteristics of the user's historical behavior, such as the number of transaction orders and the average amount of orders in the past n days, and user identity descriptions, such as age and membership status, consumption preference level for each business line, etc. Without loss of generality, we omit the subscript of the variables.

We adapt the commonly used share bottom structure. The output of the shared bottom L is defined as $\Phi = L(x)$, where Φ denotes the shared user representation learned from the shared bottom. It enables the efficient sharing of information across multi-treatments. As shown in Figure 2, we use $\hat{\vartheta}_i^c$ and $\hat{\vartheta}_i^r$ to represent the incremental effect of a user between two adjacent treatments. Thus, for the response prediction, to better distinguish the output of each head, we add the treatment embedding as part of the input rather than only separate the prediction. Then, the input of the prediction head includes two parts: the feature representation Φ and the treatment information δ_i^t. Especially, δ_i^t represents the embedding of k-th

treatment. Then we have:

$$\begin{aligned} \hat{\vartheta}_i^r &= h^r\left([\Phi; \delta_i]\right), \forall k = 1, \cdots, K, \\ \hat{\vartheta}_i^c &= h^c\left([\Phi; \delta_i]\right), \forall k = 1, \cdots, K. \end{aligned} \quad (4)$$

where $[\cdot; \cdot]$ denotes the concatenation of two vectors, the functions $h_i^r(\cdot)$ and $h_i^c(\cdot)$ is the incremental effect prediction head for the response and the cost, respectively.

Finally, the response prediction of a user for the k-th treatment can be formulated as:

$$\hat{y}_t^r = \begin{cases} \hat{y}_0^r, & \text{if } t = 0 \\ \hat{y}_0^r + \sum_{k=1}^t \hat{\vartheta}_i^r, & \text{if } t \geq 1. \end{cases} \quad (5)$$

Following the above equation, we can also predict the cost similarly.

Then, to implement the monotonic constraint, we only need to make the incremental effect $\hat{\vartheta}_i^c$ and $\hat{\vartheta}_i^r$ greater than 0. It is easily achieved by squaring the raw output of the last head layer, or we can turn the output of the last head layer into an exponential form. The loss function of the proposed model is defined as follows:

$$\mathcal{L}_{\text{predict}} = \frac{1}{N} \sum_{(x,t,y^c,y^r) \in \mathcal{D}} \left(\mathcal{L}_c(y^c, \hat{y}^c) + \mathcal{L}_r(y^r, \hat{y}^r)\right). \quad (6)$$

where N is the number of samples in the dataset $\mathcal{D}$, $\mathcal{L}_c$ and $\mathcal{L}_r$ are the loss functions to predict the cost and the response, which can be either cross-entropy or mean square error. $\hat{y}_c$ and $\hat{y}^r$ are the predicted cost and response for the corresponding treatment t.

4.1.2 Smooth Constraint. As illustrated in Figure 1(b), smoothness means that the model's output does not change much as the incentive level varies. To achieve the objective, inspired by the Siamese neural network [16], which is often used in contrastive learning, we share the parameter weights between different response prediction heads so as to the cost prediction heads.

Moreover, to ensure the uplift prediction can accurately reflect the influence of different treatments, we leverage Lipschitz regularization, where h is Lipschitz continuous if there exists a constant $c \geq 0$ such that:

$$\left|h\left(v_i\right) - h\left(v_j\right)\right| \leq c\left\|v_i - v_j\right\|_2 \quad (7)$$

where $v_i = \left[\Phi; \delta_i^t\right]$ is the concatenation of the user feature embedding and the treatment embedding, h can be either h_r or h_c, and intuitively, the change of outcome is bounded by constant c for smoothness. It is evident that if the neural network f_Ω is c-Lipschitz on the v, it is also c-Lipschitz on the treatment embedding δ_i^t.

The Lipschitz Regularization [24] solely depends on the weight matrices of each network layer with estimated per-layer Lipschitz upper bound c_j, the loss for regularizing the differences between treatment effects and outcomes can be written as:

$$\mathcal{L}_{\text{Lip}} = \prod_{j=1}^l \text{softplus}\left(c_j\right), \quad (8)$$

where $\text{softplus}\left(c_j\right) = \ln\left(1 + e^{c_j}\right)$ is a reparameterization strategy to avoid invalid negative estimation on Lipschitz constant c_j and l is the number of network layers.

Then, we can get the loss function of the uplift prediction module:

$$\mathcal{L}_{\text{uplift}} = \mathcal{L}_{\text{prediction}} + \alpha \mathcal{L}_{\text{Lip}}. \quad (9)$$

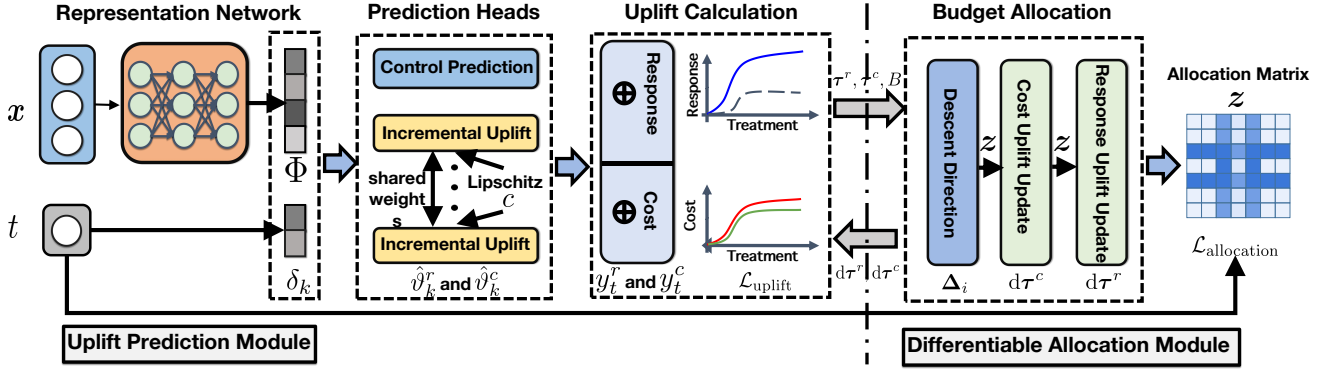where α is the hyperparameter to control the trade-off.

Figure 2: The overall structure of our E³IR. The uplift prediction module generates the predicted uplift of user responses and corresponding costs, the differentiable allocation module generates the allocation matrix, and the two loss functions $\mathcal{L}_{\text{uplift}}$ and $\mathcal{L}_{\text{allocation}}$ are jointly optimized in Eq. (20).

4.2 Differentiable Allocation Module

After we have got the predicted response uplift and the cost uplift, we aim to build a differentiable incentive recommendation layer for the end-to-end optimization, which can reduce the performance gap between the uplift prediction and the budget allocation. Especially motivated by the "predict + optimize" combination optimization problem [25], we incorporate an ILP as a differentiable module in the model structure that inputs both constraint and objective.

For ease of understanding, we omit the sum and the subscript in Section 3.2 and use the vector form (bold form) of the variables. For the differentiability problem, we can reformulate the gradient calculation as a task of determining the descent direction. We need to encounter the problem that the suggested gradient update $z - dz$ to the optimal solution z is often unattainable, meaning that $z - dz$ does not represent a feasible integer point.

4.2.1 Descent direction. During the backward pass, the gradients of the subsequent layers are obtained from the ILP solver. We need to determine the change direction for the cost and response uplift. Specifically, we want the solution of the updated ILP to move in the direction opposite to the incoming gradient, also known as the direction of descent.

Given a loss function denoted by $\mathcal{L}$, let τ^c, B, τ^r, and the incoming gradient $dz = \partial\mathcal{L}/\partial z$ at the point $z(\tau^c, B, \tau^r)$ be provided. To achieve end-to-end optimization, we must return gradients for $\partial\mathcal{L}/\partial\tau^c$, $\partial\mathcal{L}/\partial B$, and $\partial\mathcal{L}/\partial\tau^r$. Our objective is to determine the directions $d\tau^c$, dB, and $d\tau^r$ that result in the most significant decrease in distance between the updated solution $z(\tau^c - d\tau^c, B - dB, \tau^r - d\tau^r)$ and the target $z - dz$. If the mapping z is differentiable, it results in the correct gradients, such as $\partial\mathcal{L}/\partial\tau^c = \partial\mathcal{L}/\partial z \cdot \partial z/\partial\tau^c$ (similarly for B and τ^c). Then we have the following proposition:

PROPOSITION 1. *Let $y : \mathbb{R}^\ell \to \mathbb{R}^n$ be a differentiable function at $x \in \mathbb{R}^\ell$. Let $L : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function at $y = y(x) \in \mathbb{R}^n$. Denote $dy = \frac{\partial L}{\partial y}$ at y. Then the distance between $y(x)$ and $y - dy$ is minimized along the direction $\frac{\partial L}{\partial x}$, where $\frac{\partial L}{\partial x}$ is the derivative of $L(y(x))$ at x.*

PROOF. For any $\xi \in \mathbb{R}^\ell$, let $\varphi(\xi)$ represent the distance between $y(x - \xi)$ and the target $y(x) - dy$. It can be formulated as

$$\varphi(\xi) = \|y(x - \xi) - y(x) + dy\|.$$

If $dy = 0$, there is nothing to prove as $y(x) = y - dy$ and no improvements can be made. Otherwise, φ is a positive and differentiable function in the neighborhood of zero. The Frenet derivative of φ is given by:

$$\varphi'(\xi) = \frac{-[y(x - \xi) - y(x) + dy] \cdot \frac{\partial y}{\partial x}(x - \xi)}{\|y(x - \xi) - y(x) + dy\|},$$

thus,

$$\varphi'(0) = -\frac{1}{\|dy\|}\frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial x} = -\frac{1}{\|dy\|}\frac{\partial L}{\partial x},$$

where the last equality follows from the application of the chain rule, therefore, the steepest descent direction coincides with the derivative $\partial L/\partial x$, considering that $\|dy\|$ is a scalar. □

However, every ILP solution $z(\tau^c - d\tau^c, B - dB, \tau^r - d\tau^r)$ is confined to integer points, and its ability to approach the point $z - dz$ is limited unless dz itself is also an integer point. To accomplish this, we can decompose the vector dz as follows:

$$dz = \sum_{k=1}^{n} \lambda_i \Delta_i. \tag{10}$$

where $\Delta_i \in \{-1, 0, 1\}^n$ are integer points and $\lambda_i \geq 0$ are scalars. The specific choice of basis Δ_i will be discussed separately. For now, it is sufficient to understand that each point $z_i' = z - \Delta_i$ is an integer point neighboring y that indicates the direction of $-dz$. Then, we address separate problems by replacing dz with the integer updates Δ_i.

To maintain the linearity of the standard gradient mapping, our objective is to combine the gradients from the subproblems to create the final gradient linearly. It is important to note that in the budget allocation problem, B is a constant. As a result, $dB = 0$, and we do not update B in the subsequent description.

4.2.2 Cost uplift update.
To obtain a significant update for an achievable change Δ_i, we compute the gradient of a piecewise affine local mismatch function $P_{z'_i}$. The definition of $P_{z'_i}$ is derived from a geometric understanding of the underlying structure. In doing so, we depend on the Euclidean distance between a point and a hyperplane. Indeed, for any point z and a given hyperplane, parametrized by vector $\boldsymbol{\tau}^c$ and scalar B as $z \mapsto \boldsymbol{\tau}^c \cdot z - B$, we have:

$$\text{dist}(\boldsymbol{\tau}^c, B; z) = |\boldsymbol{\tau}^c \cdot z - B| / \|\boldsymbol{\tau}^c\|. \tag{11}$$

To update the constraints, we define that if $\boldsymbol{\tau}^c z'_i \leq \boldsymbol{b}$, then z'_i is feasible. Then $P_{z'_i}(\boldsymbol{\tau}^c, B)$ is formulated as:

$$P_{z'_i}(\boldsymbol{\tau}^c, B) = \begin{cases} \min_j \text{dist}\left(\boldsymbol{\tau}^c_j, B; z\right), & \text{if } z'_i \text{ is feasible and } z'_i \neq z \\ \sum_j [\![\boldsymbol{\tau}^c_j \cdot z'_i > B]\!] \text{dist}\left(\boldsymbol{\tau}^c_j, B; z'_i\right), & \text{if } z'_i \text{ is infeasible} \\ 0, & \text{if } z'_i = z \text{ or } z'_i \notin \mathcal{Z}. \end{cases} \tag{12}$$

$\mathcal{Z}$ is the value space of z. Imposing linearity and using Eq. (10), we define the gradient $d\boldsymbol{\tau}^c$ as:

$$d\boldsymbol{\tau}^c = \sum_{k=1}^n \lambda_i \frac{\partial P_{z'_i}}{\partial \boldsymbol{\tau}^c}(\boldsymbol{\tau}^c, B). \tag{13}$$

Note that the mapping $dz \mapsto d\boldsymbol{\tau}^c$ is homogeneous. It is because the whole situation is rescaled to one case (choice of basis) where the gradient is computed and then rescaled back (scalars λ_i). The most natural scale agrees when z'_i are the closest integer neighbors. This is due to the scale, which means that the entire situation is rescaled to one case (choice of basis), where the gradient is computed and subsequently rescaled. The most natural scale occurs when z'_i are the nearest integer neighbors. This ensures the situation does not collapse into a trivial solution (zero gradients) and prevents interference with distant values of z. The selection of this basis serves as a hyperparameter. In our case, we construct a valid basis explicitly and do not require optimization of any additional hyperparameters.

4.2.3 Response uplift update.
Ignoring the distinction between feasible and infeasible z'_i, the problem of updating the cost has been addressed in several prior studies. We employ a straightforward approach that defines the mismatch function in a way that the resulting update prioritizes z'_i over z in the updated optimization problem:

$$P_{z'_i}(\boldsymbol{\tau}^r) = \begin{cases} \boldsymbol{\tau}^r \cdot (z'_i - z) & \text{if } z'_i \text{ is feasible,} \\ 0 & \text{if } z'_i \text{ is infeasible or } z'_i \notin \mathcal{Z}. \end{cases} \tag{14}$$

The gradient $d\boldsymbol{\tau}^r$ is then composed similarly as in Eq. (13).

4.2.4 The choice of the basis.
Let $i_1, \ldots, i_n$ be the indices of the coordinates in the absolute values of dz in decreasing order, i.e.,

$$\left|dz_{i_1}\right| \geq \left|dz_{i_2}\right| \geq \cdots \geq \left|dz_{i_n}\right|, \tag{15}$$

and set

$$\Delta_i = \sum_{j=1}^k \text{sign}\left(dz_{i_j}\right) \boldsymbol{u}_{i_j}, \tag{16}$$

where e_i represents the k-th canonical vector. Consequently, Δ_i represents the (signed) indicator vector of the initial i dominant directions.

Let ℓ be the largest index such that $|dz_\ell| > 0$. Consequently, the initial ℓ vectors Δ_i are linearly independent and serve as a basis for their respective subspace. Thus, there are scalars λ_i that satisfy decomposition in Eq. (10).

PROPOSITION 2. *If $\lambda_j = \left|dz_{i_j}\right| - \left|dz_{i_{j+1}}\right|$ for $j = 1, \ldots, n - 1$ and $\lambda_n = \left|dz_{i_n}\right|$, then Eq. (10) holds with Δ_i's as in Eq. (16).*

PROOF. We prove that

$$\sum_{j=\ell}^n w_j \boldsymbol{u}_{i_j} = \sum_{j=\ell}^n \lambda_j \Delta_j - |w_\ell| \sum_{j=1}^{\ell-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}, \tag{17}$$

For each $\ell = 1, \ldots, n$, with the abbreviation $w_j = dz_{i_j}$, the desired equality of Eq. (10) can be derived from the special case where $\ell = 1$ in Eq. (17).

Next, we proceed with the proof of induction. To start, we will demonstrate the validity of Eq. (17) for $\ell = n$. By referring to the definition of Δ_n in Eq. (16), we can observe:

$$\lambda_n \Delta_n - |w_n| \sum_{j=1}^{n-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$= |w_n| \sum_{j=1}^n \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j} - |w_n| \sum_{j=1}^{n-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$= w_n \boldsymbol{u}_{i_n}$$

Assuming that Eq. (17) holds for $\ell + 1 \geq 2$, then we demonstrate that it also holds for ℓ:

$$\sum_{j=\ell}^n \lambda_j \Delta_j - |w_\ell| \sum_{j=1}^{\ell-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$= \sum_{j=\ell+1}^n \lambda_j \Delta_j - |w_{\ell+1}| \sum_{j=1}^\ell \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j} + \lambda_\ell \Delta_\ell$$

$$+ |w_{\ell+1}| \sum_{j=1}^\ell \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j} - |w_\ell| \sum_{j=1}^{\ell-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$= \sum_{j=\ell+1}^n w_j \boldsymbol{u}_{i_j} + (|w_\ell| - |w_{\ell+1}|) \sum_{j=1}^\ell \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$+ |w_{\ell+1}| \sum_{j=1}^\ell \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j} - |w_\ell| \sum_{j=1}^{\ell-1} \text{sign}\left(w_j\right) \boldsymbol{u}_{i_j}$$

$$= \sum_{j=\ell+1}^n w_j \boldsymbol{u}_{i_j} + \text{sign}\left(w_\ell\right) |w_\ell| \boldsymbol{u}_{i_\ell} = \sum_{j=\ell}^n w_j \boldsymbol{u}_{i_j},$$

where we use the definitions of Δ_ℓ and λ_ℓ. □

Moreover, for the binary treatment allocation problem, the value space $\mathcal{Z}$ of z is $\{0, 1\}^n$. For multi-treatment, the optimal z_{ik^*} should satisfy:

$$z_{ik^*} = \mathbb{1}\left(k^* = \arg\max_k \tau^r_{ik} - \lambda_{ik} \tau^c_{ik}\right), \tag{18}$$

where $\mathbb{1}$ is the 0/1 indicator function. Corresponding to Eq. (3), we change the decomposition results in as a vector Δ_i for the multi-choice problem, where $\Delta_i = [\Delta_{i1}, \Delta_{i2}, \cdots, \Delta_{iK}]$ and $\Delta_{ik} \in \{-1, 0, 1\} \cap \sum_k \Delta_{ik} \in \{-1, 0, 1\}$. And the value space $\mathcal{Z}$ of $z_i = [z_{i1}, z_{i2}, \cdots, z_{iK}]$ is $z_{ik} \in \{0, 1\} \cap \sum_k z_{ik} = 1$.

Moreover, inspired by the Expected Outcome Metric in [38], we design the loss function for the differentiable budget allocation module as:

$$\mathcal{L}_{\text{allocation}} = \frac{1}{N} \sum_{(x,t,y_i^c,y_i^r) \in \mathcal{D}} \mathcal{L}_d(t, z_i), \tag{19}$$

where $\mathcal{L}_d(\cdot, \cdot)$ is the cross entropy loss, z_i is the predicted incentive recommendation for each user. Combing with Eq. (6), the total loss for our E^3IR is:

$$\mathcal{L}_{E^3IR} = \mathcal{L}_{\text{predict}} + \beta \mathcal{L}_{\text{allocation}}. \tag{20}$$

where β is the hyperparameter to control the trade-off.

4.3 Discussion

This section discusses the distinctions between our E^3IR and the online MCKP and offline MCKP problems [3, 40]. In the online MCKP, the goal of optimization is to recommend a single incentive at each decision point, considering both the value and weight quantities. This online approach allows for dynamic adaptation of strategies. On the other hand, the offline MCKP employs the same underlying selection algorithm as the online MCKP but with items provided in advance. This allows the efficiency angle function to be fitted once before making recommendations. Subsequently, the algorithm determines which incentive to offer each customer without updating the efficiency angle function. In online marketing, the final objective of uplift prediction is to recommend incentives to customers. Therefore, optimizing the uplift model without incentive recommendation information may result in sub-optimal results due to the objective gap. Consequently, our E^3IR aims to provide incentive recommendation information for uplift prediction under budget constraints, aiming to mitigate the optimality gap between two-stage methods involving online or offline MCKP.

5 EXPERIMENTS

In this section, we conduct experiments to answer the following research questions:

- **RQ1:** How is the performance of our E^3IR compared with other baselines on both binary treatment and multi-treatment datasets?
- **RQ2:** How is the role that each part of our model plays?
- **RQ3:** How is the performance of our E^3IR compared with other baselines when the budget changes on the multi-treatment datasets?

5.1 Datasets

• **Hillstrom** [9]: This dataset is derived from an email merchandising campaign that involves 64,000 consumers who last purchased within twelve months. The features include recency, history_segment, history, mens, womens, zip_code, newbie, channel. The dataset contains three variables describing consumer activity in the following two weeks after email campaign delivery: visit, conversion, and spend. We select the spend as the response, the visit as the cost, and the segment as the treatment (3 types). Following [6], we also construct two binary treatment datasets *Hillstrom-Men* and *Hillstrom-Women*, which combine the mens_email treatment

group and no_email control group, the womens_email treatment group and no_email control group, respectively.

• **Production**: This dataset was obtained from one of China's biggest short video platforms. For such platforms, video sharpening is known as a valuable source for studying user experience indicators. Different degrees of video clarity can significantly influence user experiences, potentially affecting users' playback time and bringing different bandwidth costs. To investigate this, we conducted random experiments over two weeks, assigning three levels of video sharpening ($T = 1, 2, 3$) as treatment groups, while regular videos ($T = 0$) served as the control group. We quantified the impact of different treatments by tracking users' short video playback time and bandwidth costs during this period. The resulting dataset comprises over 8 million users, with 108 features capturing user-related characteristics. We present the dataset collection visualization in Figure 3. Moreover, the data statistics are shown in the appendix.
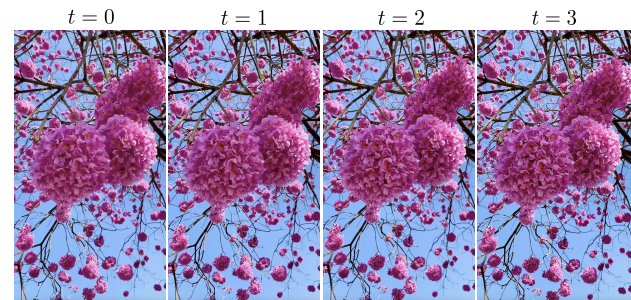

$t = 0 \qquad t = 1 \qquad t = 2 \qquad t = 3$

Figure 3: The visualization of the Production dataset collection subject to different treatments. As t increases, the clarity of the video correspondingly enhances.

5.2 Baselines and Metrics

5.2.1 Baselines. We compare our E^3IR with the following baselines:

- Cost-unaware binary incentive recommendation problem: **S-Learner** [17], **X-Learner** [17], **Causal Forest** [8], **CFR-Net** [29], **DragonNet** [30], **EUEN** [15].
- Cost-aware binary incentive recommendation problem: **TPM-SL** [37], **Direct Rank** [11], **DRP** [40].
- Cost-aware multi-incentive recommendation problem: **Multi-TPM-SL** [1], **DPM** [40].

5.2.2 Metrics. We evaluate our model with the following metrics: AUUC (Area under Uplift Curve) [28], QINI (Qini Coefficient) [23], KENDALL (Kendall's Rank Correlation) [23], AUCC (Area under Cost Curve) [11], MT-AUCC (Muti-Treatment AUCC) [40], EOM (Expected Outcome Metric) [38].

Due to the space limitation, we present the details of the baselines and metrics in the Appendix.

5.3 Implementation Details

We implement all baselines and our E^3IR based on Pytorch 1.10 and Jax 0.4.23, with Adam as the optimizer and a maximum iteration count of 30. We use the QINI as a reference to search for the best

hyper-parameters for all baselines and our model. We also adopt an early stopping mechanism with a patience of 5 to avoid over-fitting the training set. Furthermore, we utilize the hyper-parameter search library Optuna [2] to accelerate the tuning process. All experiments are implemented on NVIDIA A40 and Intel(R) Xeon(R) 5318Y Gold CPU @ 2.10GHz.

5.4 Overall Performance (RQ1)

For the uplift prediction and budget allocation on the binary treatment datasets (*i.e.*, Hillstorm-Men, Hillstorm-Women), we present the results in Table 1. From the results, we have the following observations: 1) Meta-learners (*i.e.*, the S-Learner and X-Learner) appear to perform competitively with more advanced deep learning approaches, particularly in KENDALL, which is a measure of the model's ability to rank users by the predicted uplift. Causal Forest have a better performance than CFRNet$_{wass}$ across all the metrics, which shows the effectiveness of the ensemble structure and the splitting criterion. 2) DragonNet performs best among all the cost-unaware baselines, this may because of the design of the target regularization, which can reduce the predictive error of uplift. In total, the representation learning-based methods have a higher KENDALL than other baselines, which suggests that representation learning may have potential advantages in capturing ranking consistency on the uplift prediction. 3) Our E^{3}IR exhibits superior performance on both datasets across all the three metrics, this shows that the monotonic and smoothness constraints of the user response curve in online marketing can improve the performance of the ranking metrics in uplift modeling. Moreover, the differentiable allocation module can help train the uplift prediction module to be more effective from the decision making perspective.

From the results of cost-aware binary treatment baselines, we set the budgets as 400 in both two datasets, and we have the following observations: 1) TPM-SL performs similar to the S-Learner on the basic uplift metrics (*i.e.*, AUUC, QINI and KENDALL), because the TPM-SL uses the same model structure as the S-Learner for uplift prediction. However, this method shows the worst performance on AUCC among all cost-aware binary treatment methods; this may be because using the S-Learner as the estimator of response and cost predictors will introduce a bigger prediction error than other methods, which leads to a bad AUCC. 2) Direct Rank and DRP performs better than most cost-unaware baselines, this is because that the two methods design the learning objective related to the cost, with the cost information, the model can get more accurate rank of ROI, which is benefit for the uplift prediction. DRP performs better than Direct Rank on AUCC, which leverages a factor model for mitigating the performance gap between the prediction and optimization. This is helpful for getting better budget allocation metrics. 3) Our E^{3}IR shows superior performance across diverse metrics and datasets, especially on QINI. This is due to that we use the QINI as the objective to tune the hyperparameters of all the baselines and our E^{3}IR. Compared to the cost-aware baselines, the superior performance indicates that the differentiable budget allocation module can help our E^{3}IR get better results on the ROI ranking. The results also show that the end-to-end training of our E^{3}IR can reduce the performance gap between the prediction and optimization.

Considering the multi-treatment scenarios, we evaluate the performance of the budget allocation on the Hillstorm and Production datasets with the budget as 500 and 50 thousand, respectively. We present the results in Table 2, from the results, we have the following observations: 1) DRM achieves the suboptimal performance on most results of the two datasets, this may because the design of decision factor module. Compared with solving the ILP problem by the bisection methods, the decision factor can transport the optimization information into the uplift prediction, and this can partly remove the performance gap in the two-stage method Multi-TPM-SL. 2) Our E^{3}IR achieves superior performance across the MT-AUCC and EOM in the two datasets. Expected the public dataset Hillstorm, the results of the Production dataset further verify the effectiveness of our designed structure. The consistency of E^{3}IR's performance accentuates its adaptability and potential for generalization across diverse experimental conditions.

5.5 Ablation Study (RQ2)

To analyze the role played by each proposed module, we construct the ablation study on both the binary and multi-treatment datasets. Specifically, we sequentially remove each module of our E^{3}IR, *i.e.* the monotonic constraint in the uplift prediction module (E^{3}IR w/o MC), the smooth constraint in the uplift prediction module (E^{3}IR w/o SC), the differentiable allocation module (E^{3}IR w/o DA). We present the results in Figure 4. From the results, we can see that removing any module will cause performance degradation across all the datasets and metrics. This verifies the validity of each module design in our E^{3}IR. That is, the monotonic constraint and the smooth constraint can help the model get a better ranking performance of the predicted uplift by injecting the marketing domain knowledge into the structure design. The differentiable allocation module can build the bridge between the prediction and optimization, which can bring a big performance improvement compared with other parts in our E^{3}IR.
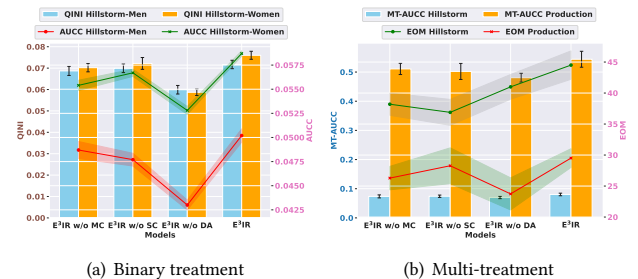

(a) Binary treatment   (b) Multi-treatment

Figure 4: Ablation study of our E^{3}IR on all the binary treatment and multi-treatment datasets.

5.6 Analysis of Budget Influence (RQ3)

After we predict the uplifts τ^r and τ^c, we can evaluate the influence of different budgets. Following the formulation in [40], we draw the curve and use incremental cost as the X-axis and incremental response as the Y-axis. Similar to the uplift curve, if a good model generates the score, then the curve should be above the benchmark
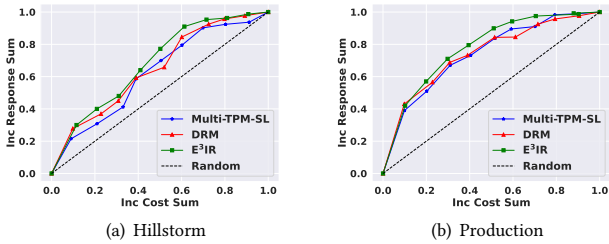
Table 1: Overall comparison between our models and the baselines on Hillstorm Men and Hillstorm Women datasets. We report the results over five random seeds. The best and second best results are bold and underlined, respectively.

Methods	Hillstorm Men				Hillstorm Women			
	AUUC	QINI	KENDALL	AUCC	AUUC	QINI	KENDALL	AUCC
S-Learner	0.4729 ± 0.0377	0.0206 ± 0.0110	0.5271 ± 0.0103	−	0.4804 ± 0.0253	0.0271 ± 0.0202	0.4809 ± 0.0197	−
X-Learner	0.4557 ± 0.0245	0.0473 ± 0.0227	0.5324 ± 0.0402	−	0.4513 ± 0.0056	0.0421 ± 0.0332	0.5723 ± 0.0288	−
Causal Forest	0.5023 ± 0.0219	0.0487 ± 0.0165	0.5689 ± 0.0402	−	0.5122 ± 0.0104	0.0523 ± 0.0211	0.5504 ± 0.0301	−
CFRNet$_{mmd}$	0.5228 ± 0.0306	0.0496 ± 0.0232	0.5451 ± 0.0243	−	0.5189 ± 0.0349	0.0519 ± 0.0251	0.5244 ± 0.0342	−
CFRNet$_{wass}$	0.4998 ± 0.0442	0.0464 ± 0.0266	0.5199 ± 0.0303	−	0.5097 ± 0.0369	0.0477 ± 0.0198	0.5308 ± 0.0314	−
DragonNet	0.5640 ± 0.0392	0.0670 ± 0.0298	0.6804 ± 0.0381	−	0.5921 ± 0.0289	0.0623 ± 0.0277	0.6597 ± 0.0392	−
EUEN	0.5723 ± 0.0122	0.0603 ± 0.0265	0.5408 ± 0.0385	−	0.5414 ± 0.0151	0.0654 ± 0.0288	0.5202 ± 0.0401	−
TPM-SL	0.4665 ± 0.0188	0.0315 ± 0.0249	0.5902 ± 0.0351	0.0476 ± 0.0056	0.4725 ± 0.0211	0.0310 ± 0.0256	0.5108 ± 0.0399	0.0502 ± 0.0064
Direct Rank	<u>0.5810</u> ± 0.0198	<u>0.0683</u> ± 0.0295	0.6406 ± 0.0299	0.0498 ± 0.0047	0.6022 ± 0.0251	<u>0.0671</u> ± 0.0265	0.6441 ± 0.0297	0.0519 ± 0.0048
DRP	0.5783 ± 0.0252	0.0667 ± 0.0210	<u>0.6811</u> ± 0.0275	**0.0545** ± 0.0044	<u>0.6211</u> ± 0.0298	0.0669 ± 0.0214	**0.6802** ± 0.0357	<u>0.0531</u> ± 0.0043
E^{3}IR	**0.5928** ± 0.0193	**0.0717** ± 0.0196	**0.7033** ± 0.0353	<u>0.0502</u> ± 0.0069	**0.6466** ± 0.0267	**0.0760** ± 0.0187	<u>0.67f54</u> ± 0.0322	**0.0587** ± 0.0027

Table 2: Overall comparison between our models and the baselines on Hillstorm and Production datasets. The EOM is represented by employing the min-max normalized responses. Moreover, the EOM of the Production dataset is scaled down by a factor of e^4. We report the results over five random seeds. The best and second best results are bold and underlined, respectively.
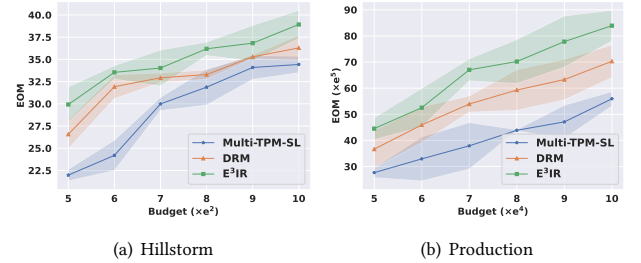
Methods	Hillstorm		Production	
	MT-AUUC	EOM	MT-AUUC	EOM
Multi-TPM-SL	0.0645 ± 0.0078	21.9767 ± 0.1088	<u>0.3907</u> ± 0.0196	27.6576 ± 0.1079
DRM	<u>0.0726</u> ± 0.0021	<u>26.5744</u> ± 0.1102	0.3601 ± 0.0258	<u>36.6544</u> ± 0.1099
E^{3}IR	**0.0803** ± 0.0044	**29.9221** ± 0.1698	**0.4639** ± 0.0311	**44.5087** ± 0.1877

line. This means a better model would select samples to achieve higher incremental value for the same incremental cost level. The results are shown in Figure 5. Corresponding to the MT-AUUC results in Table 2, our E^{3}IR has the best performance of the cost curve on both multi-treatment datasets. Moreover, we also test the EOM of different approaches when given different budgets; we show the results in Figure 6. It is still clear that our E^{3}IR can always help the platform obtain much more profits under different budgets.

(a) Hillstorm   (b) Production

Figure 6: EOM comparison with the budget changes between the cost-aware baselines and our E^{3}IR on the multi-treatment datasets.

(a) Hillstorm   (b) Production

Figure 5: Cost curve comparison between the cost-aware baselines and our E^{3}IR on the multi-treatment datasets.

two customized modules. Firstly, we incorporate marketing domain knowledge into the uplift prediction module, enabling the acquisition of a monotonic and smooth user response curve. Additionally, we utilize the differentiable optimization of the ILP problem to reduce the performance gap in two-stage methods. Extensive experiments conducted on binary treatment and multi-treatment datasets demonstrate the superiority of our method across various metrics. Future work will concentrate on reducing the time complexity of the differentiable allocation module and applying this approach to complex real-world marketing scenarios.

6 CONCLUSION

In this paper, we formulate online marketing with a specific budget constraint as a "predict + optimize" problem. To solve it, we propose an end-to-end optimization method E^{3}IR, which consists of

ACKNOWLEDGMENTS

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] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2623–2631.

[3] 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 & Knowledge Management*. 2863–2872.

[4] Brandon Amos and J Zico Kolter. 2017. Optnet: Differentiable optimization as a layer in neural networks. In *International Conference on Machine Learning*. PMLR, 136–145.

[5] Susan Athey and Stefan Wager. 2019. Estimating treatment effects with causal forests: An application. *Observational studies* 5, 2 (2019), 37–51.

[6] Artem Betlei, Eustache Diemert, and Massih-Reza Amini. 2021. Uplift modeling with generalization guarantees. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 55–65.

[7] Ph Darondeau and Pierpaolo Degano. 1989. Causal trees. In *International Colloquium on Automata, Languages, and Programming*. Springer, 234–248.

[8] Jonathan MV Davis and Sara B Heller. 2017. Using causal forests to predict treatment heterogeneity: An application to summer jobs. *American Economic Review* 107, 5 (2017), 546–550.

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

[10] Priya Donti, Brandon Amos, and J Zico Kolter. 2017. Task-based end-to-end model learning in stochastic optimization. *Advances in neural information processing systems* 30 (2017).

[11] Shuyang Du, James Lee, and Farzin Ghaffarizadeh. 2019. Improve User Retention with Causal Learning. In *The 2019 ACM SIGKDD Workshop on Causal Discovery*. PMLR, 34–49.

[12] 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*. 486–491.

[13] Leo Guelman, Montserrat Guillén, and Ana M Pérez-Marín. 2015. Uplift random forests. *Cybernetics and Systems* 46, 3-4 (2015), 230–248.

[14] Ali Ugur Guler, Emir Demirović, Jeffrey Chan, James Bailey, Christopher Leckie, and Peter J Stuckey. 2022. A divide and conquer algorithm for predict+ optimize with non-convex problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 3749–3757.

[15] Wenwei Ke, Chuanren Liu, Xiangfu Shi, Yiqiao Dai, S Yu Philip, and Xiaoqiang Zhu. 2021. Addressing exposure bias in uplift modeling for large-scale online advertising. In *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1156–1161.

[16] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, Vol. 2. Lille.

[17] 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.

[18] 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.

[19] Dugang Liu, Xing Tang, Han Gao, Fuyuan Lyu, and Xiuqiang He. 2023. Explicit Feature Interaction-aware Uplift Network for Online Marketing. *arXiv preprint arXiv:2306.00315* (2023).

[20] 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.

[21] 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.

[22] Jayanta Mandi and Tias Guns. 2020. Interior point solving for lp-based prediction+ optimisation. *Advances in Neural Information Processing Systems* 33 (2020), 7272–7282.

[23] Belbahri Mouloud, Gandouet Olivier, and Kazma Ghaith. 2020. Adapting neural networks for uplift models. *arXiv preprint arXiv:2011.00041* (2020).

[24] Adam M Oberman and Jeff Calder. 2018. Lipschitz regularized deep neural networks converge and generalize. *arXiv preprint arXiv:1808.09540* (2018).

[25] Anselm Paulus, Michal Rolínek, Vít Musil, Brandon Amos, and Georg Martius. 2021. Comboptnet: Fit the right np-hard problem by learning integer programming constraints. In *International Conference on Machine Learning*. PMLR, 8443–8453.

[26] Robert L Phillips. 2021. *Pricing and revenue optimization*. Stanford university press.

[27] Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. *J. Amer. Statist. Assoc.* 100, 469 (2005), 322–331.

[28] Piotr Rzepakowski and Szymon Jaroszewicz. 2010. Decision trees for uplift modeling. In *2010 IEEE International Conference on Data Mining*. IEEE, 441–450.

[29] Uri Shalit, Fredrik D Johansson, and David Sontag. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*. PMLR, 3076–3085.

[30] Claudia Shi, David Blei, and Victor Veitch. 2019. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems* 32 (2019).

[31] Ye Tu, Kinjal Basu, Cyrus DiCiccio, Romil Bansal, Preetam Nandy, Padmini Jaikumar, and Shaunak Chatterjee. 2021. Personalized treatment selection using causal heterogeneity. In *Proceedings of the Web Conference 2021*. 1574–1585.

[32] Chao Wang, Xiaowei Shi, Shuai Xu, Zhe Wang, Zhiqiang Fan, Yan Feng, An You, and Yu Chen. 2023. A Multi-stage Framework for Online Bonus Allocation Based on Constrained User Intent Detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5028–5038.

[33] Bryan Wilder, Bistra Dilkina, and Milind Tambe. 2019. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 1658–1665.

[34] Shuai Xiao, Le Guo, Zaifan Jiang, Lei Lv, Yuanbo Chen, Jun Zhu, and Shuang Yang. 2019. Model-based constrained MDP for budget allocation in sequential incentive marketing. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 971–980.

[35] Weijia Zhang, Jiuyong Li, and Lin Liu. 2021. A unified survey of treatment effect heterogeneity modelling and uplift modelling. *ACM Computing Surveys (CSUR)* 54, 8 (2021), 1–36.

[36] Yang Zhang, Bo Tang, Qingyu Yang, Dou An, Hongyin Tang, Chenyang Xi, Xueying Li, and Feiyu Xiong. 2021. BCORLE (λ): An Offline Reinforcement Learning and Evaluation Framework for Coupons Allocation in E-commerce Market. *Advances in Neural Information Processing Systems* 34 (2021), 20410–20422.

[37] 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*. 1820–1830.

[38] Yan Zhao, Xiao Fang, and David Simchi-Levi. 2017. Uplift modeling with multiple treatments and general response types. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 588–596.

[39] 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.

[40] Hao Zhou, Shaoming Li, Guibin Jiang, Jiaqi Zheng, and Dong Wang. 2023. Direct heterogeneous causal learning for resource allocation problems in marketing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 5446–5454.

[41] Will Y Zou, Shuyang Du, James Lee, and Jan Pedersen. 2020. Heterogeneous causal learning for effectiveness optimization in user marketing. *arXiv preprint arXiv:2004.09702* (2020).