

Cost-Effective Incentive Allocation via Structured Counterfactual Inference

Romain Lopez,¹ Chenchen Li,^{2,3} Xiang Yan,^{2,3} Junwu Xiong,²
Michael I. Jordan,¹ Yuan Qi,² Le Song^{2,4}

¹Department of Electrical Engineering and Computer Sciences, University of California, Berkeley
{romain.lopez, jordan}@cs.berkeley.edu

²AI Department, Ant Financial Service Group
{junwu.xjw, yuan.qi, le.song}@antfin.com

³Department of Computer Science, Shanghai Jiao Tong University
lcc1992@sjtu.edu.cn, xyansjtu@163.com

⁴College of Computing, Georgia Institute of Technology

Abstract

We address a practical problem ubiquitous in modern marketing campaigns, in which a central agent tries to learn a policy for allocating strategic financial incentives to customers and observes only bandit feedback. In contrast to traditional policy optimization frameworks, we take into account the additional reward structure and budget constraints common in this setting, and develop a new two-step method for solving this constrained counterfactual policy optimization problem. Our method first casts the reward estimation problem as a domain adaptation problem with supplementary structure, and then subsequently uses the estimators for optimizing the policy with constraints. We also establish theoretical error bounds for our estimation procedure and we empirically show that the approach leads to significant improvement on both synthetic and real datasets.

1 Introduction

Batch Learning from Bandit Feedback (BLFB) (Swaminathan and Joachims 2015a; 2015b) is a form of counterfactual inference given only observational data (Pearl 2009). The problem arises in many real-world decision-making scenarios, including personalized medicine, where one is interested in estimating which treatment would have led to the optimal outcome for a particular patient (Xu, Xu, and Saria 2016), or online marketing, which might focus for example on placing ads to maximize the click-through-rate (Strehl et al. 2010).

In this paper, we focus on a novel flavor of BLFB, which we refer to as *cost-effective incentive allocation*. In this problem formulation we allocate economic incentives (e.g., online coupons) to customers and observe a response (e.g., whether the coupon is used or not). Each action is mapped to a cost and we further assume that the response is monotonically increasing with respect to the action’s cost. Such an assumption is natural in many of the problems domains that motivate us, including drug-response estimation and online marketing, and it has the virtuous side effect of improving generalization to test data and making the model more interpretable. We also incorporate budget constraints related

to the global cost of the marketing campaign or treatment regime. This framework can be readily applied to the problem of allocating monetary values of coupons in a marketing campaign under fixed budget constraints from the management.

Existing work in counterfactual inference using bandit feedback (Joachims, Swaminathan, and de Rijke 2018; Shalit, Johansson, and Sontag 2017) does not make use of the supplementary structure for the rewards and is limited in practice when the cardinality of the action space is large (Lefortier et al. 2016). We therefore developed a novel algorithm which incorporates such structure. The algorithm, which we refer to as *Constrained Counterfactual Policy Optimization via Structured Incentive Response Estimation* (CCPOvSIRE), has two components. First, we build on recent advances in representation learning for counterfactual inference (Johansson, Shalit, and Sontag 2016) and extend this framework to estimate the incentive response for multiple treatments while taking into account the reward structure (SIRE). Second, we rely on the estimates to optimize the coupon assignment policy under budget constraints (CCPO).

The remainder of the paper is organized as follows. First, we review existing approaches related to the problem of cost-effective incentive allocation and provide a brief background on BFLB. Then, we present our novel setting and the corresponding assumptions. In particular, we introduce structure on the action space, cost functions and budget constraints. Furthermore, we derive theoretical upper bounds on the reward estimates and present CCPOvSIRE as a natural approach to optimizing this bound. Finally, we evaluate our approach on fully-simulated data, and introduce a novel benchmarking approach based on nested classification which we apply to semi-simulated data from ImageNet (Deng et al. 2009). In both cases, we show that our method compares favorably to approaches based on BanditNet (Joachims, Swaminathan, and de Rijke 2018), CFR-Net (Shalit, Johansson, and Sontag 2017), BART (Hill 2011) and GANITE (Yoon, Jordon, and van der Schaar 2018).

2 Related Work

Constrained Policy Optimization Safety constraints in reinforcement learning (RL) are usually expressed via the level sets of a cost function. The main challenge of safe RL is that the cost of a certain policy must be evaluated with a off-policy strategy, which is a hard problem (Jiang and Li 2016). Recent work focuses on developing a local policy search algorithm with guarantees of respecting cost constraints (Achiam et al. 2017). This approach is based on a trust-region method (Schulman et al. 2015), which allows it to circumvent the off-policy evaluation step. Our setting is more akin to that of contextual multi-armed bandits, since customers are modeled as independent replicates from a unique distribution. In this scenario, checking for the satisfaction of the cost constraints is straightforward, which makes the original problem substantially easier. Most research contributions on policy optimization with budget constraints focus on the online learning setting (Ding et al. 2013) and are therefore not directly applicable to BLFB.

Counterfactual Risk Minimization (CRM) The problem of BLBF consists of maximizing the expected reward using importance sampling (IS), as follows:

$$\mathbb{E}_{x \sim \mathcal{P}(x)} \mathbb{E}_{y \sim \rho(y|x)} \left[\delta(x, y) \frac{\pi(y|x)}{\rho(y|x)} \right], \quad (1)$$

where δ is the reward function, π a parameterized policy and ρ the logging policy. Notably, the action probabilities $\rho(y|x)$ need to be either logged or estimated. In the specific case where the logging policy is unknown and learned from the data, error bounds have been derived (Strehl et al. 2010). The variance of the naive IS estimator can be reduced using a doubly robust estimator (Dudik, Langford, and Li 2011). The CRM principle (Swaminathan and Joachims 2015a) is based on empirical Bernstein concentration bounds (Maurer and Pontil 2009) of finite-sample estimates for Eq. (1). BanditNet (Joachims, Swaminathan, and de Rijke 2018) is a deep learning methods based on equivariant estimation (Swaminathan and Joachims 2015b) and stochastic optimization. An advantage of this framework is that its mathematical assumptions are weak, while a disadvantage is that it is not clear how to make use of structured rewards.

Estimation of Individualized Treatment Effect (ITE) The ITE (Alaa and van der Schaar 2018) is defined as the difference in expectation between two treatments

$$\mathbb{E}[r | x, y = 1] - \mathbb{E}[r | x, y = 0], \quad (2)$$

where x is a point in customer feature space and r is a random variable corresponding to the reward. The difficulty of estimating the ITE arises primarily from the fact that historical data do not always fall into the ideal setting of a randomized control trial. That inevitably induces an estimation bias due to the discrepancy between the distributions $\mathcal{P}(x | y = 0)$ and $\mathcal{P}(x | y = 1)$. BART (Hill 2011) is an estimation procedure based on Bayesian nonparametric methods. CFRNet (Shalit, Johansson, and Sontag 2017)

and BNN (Johansson, Shalit, and Sontag 2016) both cast the counterfactual question into a domain adaptation problem that can be solved via representation learning. Essentially, they propose to find an intermediate feature space \mathcal{Z} which embeds the customers and trades off the treatment discrepancy for the reward predictability. GANITE (Yoon, Jordon, and van der Schaar 2018) proposes to learn the counterfactual rewards using generative adversarial networks and extends this framework to the multiple treatment case via a mean-square error loss. Notably, this line of work does not require knowing the logging policy beforehand. Remarkably, out of all these contributions, only one focuses on the setting of structured rewards and only in the case of binary outcomes (Kallus 2019).

3 Background: Batch Learning from Bandit Feedback

For concreteness, we focus on the example of a marketing campaign. Let \mathcal{X} be an abstract space and $\mathcal{P}(x)$ a probability distribution on \mathcal{X} . We consider a central agent and let $(x_1, \dots, x_n) \in \mathcal{X}^n$ denote a set of customers. We assume that each customer is an independently drawn sample from $\mathcal{P}(x)$. Let \mathcal{Y} be the set of financial incentives which can be provided by the central agent and let $\mathcal{S}^{\mathcal{Y}}$ be the space of probability distributions over \mathcal{Y} . The central agent deploys a marketing campaign which we model as a policy $\pi : \mathcal{X} \rightarrow \mathcal{S}^{\mathcal{Y}}$. For simplicity, we also denote the probability of an action y under a policy π for a customer x using conditional distributions $\pi(y|x)$. In response, customers can either choose to purchase the product from the central agent or from another unobserved party. Given a context x and an action y , we observe a stochastic reward $r \sim p(r|x, y)$. In practice, the reward can be defined as any available proxy of the central agent’s profit (e.g., whether the coupon was used or not). Given this setup, the central agent seeks an optimal policy:

$$\pi^* \in \arg \max_{\pi \in \Pi} \mathbb{E}_{x \sim \mathcal{P}(x)} \mathbb{E}_{y \sim \pi(y|x)} \mathbb{E}[r | x, y]. \quad (3)$$

This problem, referred to as BLBF, has connections to causal and particularly counterfactual inference. As described in Swaminathan and Joachims (2015a), the data are incomplete in the sense that we do not observe what would have been the reward if another action was taken. Furthermore, we cannot play the policy π in real time; we instead only observe data sampled from a logging policy $\rho(y|x)$. Therefore, the collected data are also biased since actions taken by the logging policy ρ are over-represented.

4 Cost-Effective Incentive Allocation

The BLBF setting might not be suitable when actions can be mapped to monetary values, as we illustrate in the following example.

Example 1. Monetary marketing campaign. Let y denote the discount rate and x denote a customer’s profile. Let r be the customer’s response (e.g., how much he or she bought). Since the customer will be more susceptible to use the discount as its monetary value increases, we assume that for

each value of x , $r \mid x, y$ is almost surely increasing in y . Then, the trivial policy which always selects the most expensive action is a solution to problem (3).

This simple example motivates the main set of hypothesis we now formally introduce in order to better pose our problem. Notably, our added assumptions further relate actions to the reward distribution.

Assumption 1. Structured action space and rewards. *Let us note the conditional reward $\mathbb{E}[r \mid x, y]$ as $f(x, y)$. We assume there exists a total ordering $<_{\mathcal{Y}}$ over the space \mathcal{Y} . The reward distribution is compatible with the total ordering $<_{\mathcal{Y}}$ in the sense that*

$$\forall (x, y, y') \in \mathcal{X} \times \mathcal{Y}^2, y <_{\mathcal{Y}} y' \Rightarrow f(x, y) \leq f(x, y'). \quad (4)$$

Assumption 2. Budget constraints. *There exists a cost function $c : \mathcal{Y} \rightarrow \mathbb{R}^+$ monotone on \mathcal{Y} with respect to the total ordering $<_{\mathcal{Y}}$. Let m be a maximal average budget per customer. We define the set of feasible policies Π as*

$$\Pi = \{\pi : \mathcal{X} \rightarrow \mathcal{S}^{\mathcal{Y}} \mid \mathbb{E}_{x \sim \mathcal{P}(x)} \mathbb{E}_{y \sim \pi(y|x)} [c(y)] \leq m\}. \quad (5)$$

In this manuscript, we will assume that the ordering as well as the cost function are known. We subsequently tailor the BLFB problem to the constrained case as follows:

$$\max_{\pi \in \Pi} \mathbb{E}_{x \sim \mathcal{P}(x)} \mathbb{E}_{y \sim \pi(y|x)} f(x, y), \quad (6)$$

which we refer to as *Counterfactual Constrained Policy Optimization* (CCPO). Additionally, we refer to the problem of estimating the response function f as *Incentive Response Estimation* (IRE); this is a natural extension of the ITE problem in the scenario of multiple treatments as discussed in Yoon, Jordon, and van der Schaar (2018). We now claim that IRE is a statistically harder problem than CCPO in the following sense:

Proposition 1. *Let us assume that the incentive response function f is identified. Then solving Eq. (6) for any budget m reduces to a binary search over a unique Lagrange multiplier for the cost constraint. In particular, each step of the search has a complexity that is linear in the sample size.*

Proof. See Appendix A¹. □

This negative results implies that, in general, plugging in the output of an estimation (IRE) algorithm for policy optimization (CCPO) might be suboptimal compared to directly learning a policy (the aim of CRM-based methods). However, Assumption 1 may substantially reduce the complexity of the estimation problem provided the inference procedure benefits from such structure (referred to as SIRE). As a consequence, we propose an approach that we refer to as ‘‘CCPOvSIRE.’’ We expect that such a structured counterfactual inference algorithm will outperform both ITE and CRM-based methods. This should especially be true in the regime of medium to large action spaces, where CRM-based methods struggle in practice (Lefortier et al. 2016). Such tradeoffs between complexity and structure are common in

¹Please visit <https://arxiv.org/abs/1902.02495> for supplementary information

machine learning (cf. discriminative versus generative approaches to supervised learning problems (Ng and Jordan 2002) and model-free versus model-based approaches to RL (Pong et al. 2018)).

5 Constrained Counterfactual Policy Optimization via Structured Incentive Response Estimation

We adopt the following classical assumptions from counterfactual inference (Rubin 2005) which are sufficient to ensure that the causal effect is identifiable from historical data (Shalit, Johansson, and Sontag 2017). Such assumptions explicitly imply that all the factors determining which actions were taken are observed. Notably, these must come from domain-specific knowledge and cannot be inferred from data.

Assumption 3. Overlap. *There exists a scalar $\epsilon > 0$ such that $\forall (x, y) \in \mathcal{X} \times \mathcal{Y}, \rho(y \mid x) > \epsilon$.*

Assumption 4. No unmeasured confounding. *Let $r^{\mathcal{Y}} = (r_y)_{y \in \mathcal{Y}}$ denote the vector of possible outcomes in the Rubin-Neyman potential outcomes framework (Rubin 2005). We assume that the vector $r^{\mathcal{Y}}$ is independent of the action y given x .*

We now turn to the problem of estimating the function f from historical data, using domain adaptation learning bounds. To this end, we first write the estimation problem with a general population loss. Let $L : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ be a loss function and let \mathcal{D} a probability distribution on the product $\mathcal{X} \times \mathcal{Y}$ (which we refer to as a *domain*). As in Shalit, Johansson, and Sontag (2017), we introduce an abstract feature space \mathcal{Z} and an invertible mapping Λ such that $z = \Lambda(x)$. For technical developments, we need to assume that

Assumption 5. $\Lambda : \mathcal{X} \rightarrow \mathcal{Z}$ is a twice-differentiable one-to-one mapping.

This allows us to identify each domain \mathcal{D} with a corresponding domain on $\mathcal{Z} \times \mathcal{Y}$. Let $(\hat{g}_{\psi})_{\psi \in \Psi}$ be a parameterized family of real-valued functions defined on $\mathcal{Z} \times \mathcal{Y}$. We define the *domain-dependent* population risk $\epsilon_{\mathcal{D}}(\hat{g}_{\psi})$ as

$$\epsilon_{\mathcal{D}}(\hat{g}_{\psi}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} [L(\hat{g}_{\psi}(\Lambda(x), y), f(x, y))]. \quad (7)$$

In the historical data, individual data points (x, y) are sampled from the so-called *source* domain $\mathcal{D}_S = \mathcal{P}(x)\rho(y \mid x)$. However, in order to perform off-policy evaluation of a given policy π , we would ideally need data from $\mathcal{P}(x)\pi(y \mid x)$. The discrepancy between the two distributions will cause a strong bias in off-policy evaluation which can be quantified via learning bounds from domain adaptation (Blitzer et al. 2007). In particular, we wish to bound how much an estimate of f based on data from the source domain \mathcal{D}_S can generalize to an abstract target domain \mathcal{D}_T (yet to be defined).

Proposition 2. *Let $\psi \in \Psi$, and $\mathcal{D}_S, \mathcal{D}_T$ be two domains. Let \mathcal{H} be a function class. Let $IPM_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T)$ denote the integral probability metric (IPM) between distributions \mathcal{D}_S and \mathcal{D}_T with function class \mathcal{H}*

$$IPM_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) = \sup_{h \in \mathcal{H}} |\mathbb{E}_{\mathcal{D}_S} h(x, y) - \mathbb{E}_{\mathcal{D}_T} h(x, y)|, \quad (8)$$

In the case that \mathcal{H} is the set of test functions

$$\mathcal{H} = \{(x, y) \mapsto L(\hat{g}_\psi(\Lambda(x), y), f(x, y)) \mid \psi \in \Psi\},$$

then, it is possible to bound the population risk on the target domain by

$$\epsilon_{\mathcal{D}_T}(\hat{g}_\psi) \leq \epsilon_{\mathcal{D}_S}(\hat{g}_\psi) + \text{IPM}_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T). \quad (9)$$

Proof. See Theorem 1 of Blitzer et al. (2007). \square

This result explicitly bounds the population risk on the target domain based on the risk on the source domain and the additional complexity term (IPM). We now explain how to compute both terms in Eq. (9) and detail how we can make use of the structure from both theoretical and implementation perspectives.

Source domain risk minimization

From the algorithmic perspective, we can directly make use of the historical data to approximate the population risk on the source domain. In order to further benefit from the known structure about the reward function, we need to restrain the search space for g_ψ to functions that are increasing in y for each fixed transformed feature z . This problem has been studied in neural network architecture engineering (MIN-MAX networks) (Daniels and Velikova 2010), and extensions are available to of architectures such as DLNs (You et al. 2017) or ICNNs (Amos, Xu, and Kolter 2017) could also be useful for other types of shape constraints (e.g., convexity). In our implementation, we rely on enforcing positivity on the weights of the last hidden layer as for the MIN-MAX networks.

From an approximation theory standpoint, Eq. (9) only refers to the population risk. However, benefits from the structure are expected in finite sample prediction error (Wainwright 2019), which can be linked back to population risk using uniform law arguments. Because analyzing prediction errors with neural networks may be a hard problem, we rely on well-studied nonparametric least-square estimators. However, while most research focuses on extending unidimensional results to multivariate function classes with the same assumptions (i.e., convexity in Guntuboyina and Sen (2013), monotonicity of functions in Gao and Wellner (2007) and matrices in Chatterjee, Guntuboyina, and Sen (2018)), asymmetry in the regularity assumptions between the different dimensions of a function class (such as partial monotony) has not received as much attention. We therefore derive such learning bounds in Appendix B. In particular, we exhibit a better rate with respect to the sample size for a multivariate Sobolev space (continuous action space), under smoother assumptions. In the case of discrete actions $\mathcal{Y} = \{1, \dots, K\}$, we use empirical processes theory to improve the dependence with respect to K . Namely, we investigate the case of functions with null variation, for which we present the result here.

Proposition 3. *Let \mathcal{F} denotes all functions defined on $[0, 1] \times \{1, \dots, K\}$ and taking values in $[0, 1]$. Let us consider the following function classes*

$$\begin{aligned} \mathcal{F}_c &= \{f \in \mathcal{F} \mid \exists a \in [0, 1]^K : \forall y : f(\cdot, y) = a_y\} \\ \mathcal{F}_m &= \{f \in \mathcal{F}_c \mid \forall x \in [0, 1], f(x, \cdot) \text{ increasing}\} \end{aligned}$$

Let f^* in \mathcal{F}_m be the incentive response, let $n \in \mathbb{N}$, $\sigma > 0$ and let us assume the noise model

$$\forall (i, j) \in \{1, \dots, n\} \times \{1, \dots, K\}, r_{ij} = f^*(i/n, j) + \sigma w_{ij},$$

with $w_{ij} \sim \text{Normal}(0, 1)$. Then with high probability, the least squares estimate \hat{f}_m over \mathcal{F}_m satisfies the bound

$$\|\hat{f}_m - f^*\|_n^2 \leq \frac{\sigma^2 \log K}{nK},$$

where $\|\cdot\|_n$ denotes the empirical norm. Conversely, it is known that the least squares estimate \hat{f}_c over \mathcal{F}_c satisfies with high probability the bound

$$\|\hat{f}_c - f^*\|_n^2 \leq \frac{\sigma^2}{n}.$$

Proof. See Appendix B. \square

Since our observation model here assume we observe for each n the outcome of all the actions, we artificially inflated $n = n'K$. Intuitively, by treating this bound as a function of n' , we recover for \mathcal{F}_c the expected parametric rate with a linear dependency in K . Such a rate is known to be minimax optimal (meaning the upper bound is indeed tight). Therefore, we proved that adding the structure on the rewards improves the dependency from K to $\log K$, which is significant. Quantifying the benefit of structured rewards for more complex function classes is left as an open problem.

IPM estimation via kernel measures of dependency

Thanks to the added structure, let us notice that the function class over which the supremum is taken inside the IPM of Eq. (9) may be substantially smaller (i.e., restricted to partially monotonic functions). This implies that weaker regularization or weaker kernels may provide satisfactory results in practice. Such a result is also suggested by Theorem 1 of Alaa and van der Schaar (2018). However, more systematic quantification of such an improvement is still an open problem.

That said, Eq. (9) assumes that we have at our disposal a target domain \mathcal{D}_T from which the estimated incentive response \hat{g}_ψ would generalize to all policies for offline evaluation. We now explain how to design such a domain in the case of multiple actions. In the work of Shalit, Johansson, and Sontag (2017), the domain that is central to the problem of binary treatment estimation is the mixture $\mathcal{D}_T = \mathcal{P}(z)\mathcal{P}(y)$, where $\mathcal{P}(y) = \int_x \rho(y \mid x) d\mathcal{P}(x)$ is the marginal frequency of actions under the logging policy ρ . Let us note that in this target domain, the treatment assignment y is randomized. We now show that choosing the same target domain in the setting of multiple treatments allows efficient estimation of the IPM. Indeed, the general problem of computing IPMs is known to be NP-hard. For binary treatments, CFRNet (Shalit, Johansson, and Sontag 2017) estimated this IPM via maximum mean discrepancy (MMD) (Gretton et al. 2012). For multiple treatments, recent work (Atan, Zame, and Van Der Schaar 2018) focused on the \mathcal{H} -divergence (Ganin et al. 2016). In this work, we propose instead a different nonparametric measure

of independence—the Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al. 2005; 2008)—which also yields an efficient estimation procedure for the IPM.

Proposition 4. *Let us assume that \mathcal{Z} and \mathcal{Y} are separable metric spaces. Let $k : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}$ (resp. $l : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$) be a continuous, bounded, positive semi-definite kernel. Let \mathcal{K} (resp. \mathcal{L}) be the corresponding reproducing kernel Hilbert space (RKHS). Let us assume that the function space \mathcal{H} is included in a ball of radius κ of the tensor space $\mathcal{K} \otimes \mathcal{L}$. Then, one can further bound the IPM in Eq. (9) as follows:*

$$IPM_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) \leq \kappa \text{HSIC}(\mathcal{P}(z)\rho(y | \Lambda^{-1}(z))). \quad (10)$$

Proof. See Section 2.3 of Smola et al. (2007), Definition 2 from Gretton et al. (2012) and Definition 11 from Sejdinovic et al. (2013). \square

Intuitively, the HSIC is a norm of the covariance operator for the joint historical data distribution (x, z) and measures the dependence between the observed action y under the logging policy and z . Remarkably, the HSIC reduces to the MMD for a binary set of actions (Proposition 2 of Lopez et al. (2018)) and our theory extends Shalit, Johansson, and Sontag (2017) for multiple as well as continuous actions sets \mathcal{Y} . We provide more context about the derivation of the HSIC and its theoretical properties in Appendix C. Crucially, the HSIC can be directly estimated via samples from (x, z) (Gretton et al. 2008) as

$$\begin{aligned} \widehat{\text{HSIC}}_n &= \frac{1}{n^2} \sum_{i,j}^n k(z_i, z_j) l(x_i, x_j) \\ &+ \frac{1}{n^4} \sum_{i,j,k,l}^n k(z_i, z_j) l(x_k, x_l) \\ &- \frac{2}{n^3} \sum_{i,j,k}^n k(z_i, z_j) l(x_i, x_k), \end{aligned} \quad (11)$$

where n is the number of samples. Such an estimator can be computed in $\mathcal{O}(n^2)$ operations.

Implementation

By putting together Eq. (9) and Eq. (10), we optimize a generalization bound for estimating the incentive function f . In our implementation, Λ and g_ψ are parameterized by neural networks and we minimize the following loss function during the SIRE step

$$\mathcal{L}(\psi, \Lambda) = \frac{1}{n} \sum_{i=1}^n L(g_\psi(\Lambda(x_i), y_i), r_i) + \kappa \text{HSIC}((\Lambda(x_i), y_i)). \quad (12)$$

Notably, in order to use stochastic optimization, the HSIC is computed on minibatches of datapoints. We use a linear kernel for all experiments. An important detail is that in practice, the parameter κ is unknown. We used a cross-validation procedure to select κ in our implementation. Although cross-validation is expected to be biased in this setting, our results suggest a reasonable performance in practice. Then, we use the estimates to solve the CCPO problem.

According to Proposition 1, this can be solved easily via a search procedure on the Lagrange multipliers. Also, in the case of discrete values for the cost function, the problem can be solved in time $\mathcal{O}(nM)$ via dynamic programming where M is the total budget.

6 Experiments

In our experiments, we consider only the case of a discrete action space. We compare our method CCPOvSIRE to a simple modification of BanditNet (Joachims, Swaminathan, and de Rijke 2018) which handles constrained policy optimization (further details in Appendix D). We also compare to GANITE (Yoon, Jordon, and van der Schaar 2018) and to ITE procedures such as BART (Hill 2011) and CFR-Net (Shalit, Johansson, and Sontag 2017). Binary treatment estimation procedures were adapted for multiple treatments following (Yoon, Jordon, and van der Schaar 2018). All estimation procedures were employed for CCPO via Proposition 1. As an ablation study, we also compare to two modified algorithms: CCPOvIRE, which is a variant of our algorithm which does not exploit the reward structure and a variant with $\kappa = 0$ (no HSIC). To establish statistical significance, we run each experiment 100 times and report confidence intervals. For our methods, GANITE and CFRNet, we performed a grid search over the hyperparameters as indicated in Appendix E. As a further sensitivity analysis regarding the choice of the parameter κ , we report the results of our algorithm for a large range of values for all experiments in Appendix F. We ran our experiments on a machine with a Intel Xeon E5-2697 CPU and a NVIDIA GeForce GTX TITAN X GPU.

Fully-simulated data

Since it is difficult to obtain a realistic dataset meeting all our assumptions and containing full information, for benchmarking purposes we first construct a synthetic dataset of the form (x, y, r) , with five discrete actions and monotonic rewards (further details in Appendix G). We also derive a dataset with binary actions for ITE benchmarking.

To train CCPOvSIRE, we used stochastic gradient descent as a first-order stochastic optimizer with a learning rate of 0.01 and a three-layer neural network with 512 neurons for each hidden layer. First, we report results on the task of estimating the reward function (IRE). For the binary treatment dataset, we report the *Precision in Estimation of Heterogeneous Effect* (PEHE) (Johansson, Shalit, and Sontag 2016). For the multiple treatments experiments, we report the *Roots of Mean Squared Error* (RMSE) (Yoon, Jordon, and van der Schaar 2018). We then apply all the algorithms to policy optimization for a fixed budget and report the expected reward. In particular, we used the dynamic programming algorithm to solve for the optimal action.

Experimental results on the synthetic dataset for a binary treatment and for multiple treatments, as well as the results for policy optimization with a fixed budget, are shown in Table 1. In the binary treatments experiments results, we show that our estimation procedure (IRE) yields the best PEHE. Notably, we observe an improvement over the no-HSIC version, which shows that our HSIC regularization improves the

	Fully-simulated data (mean \pm std)		
	Binary Action	Multiple Actions	
	PEHE	RMSE	Reward ($m = 3$)
CCPOvSIRE	NA	0.0970 \pm 0.0038	0.6082 \pm 0.0013
CCPOvSIRE [†]	NA	0.1115 \pm 0.0135	0.6069 \pm 0.0020
CCPOvIRE	0.0385 \pm 0.0011	0.1059 \pm 0.0165	0.6074 \pm 0.0011
CCPOvIRE [†]	0.0404 \pm 0.0011	0.1238 \pm 0.0132	0.6061 \pm 0.0009
GANITE	0.1077 \pm 0.0086	0.2263 \pm 0.0394	0.5964 \pm 0.0054
BART	0.0674 \pm 0.0026	0.2057 \pm 0.0027	0.6002 \pm 0.0001
CFRNet	0.0447 \pm 0.0004	0.1069 \pm 0.0060	0.5958 \pm 0.0010
BanditNet	NA	NA	0.5899 \pm 0.0076

Table 1: Performance with simulated datasets. [†]: baselines with no HSIC ($\kappa = 0$). Bold indicates the method with the best performance for each dataset.

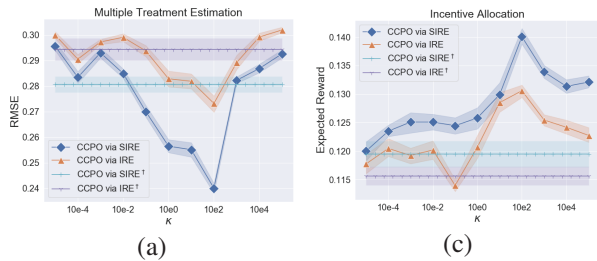


Figure 1: Benchmarking on the ImageNet dataset. Solid lines refer to mean, shadowed areas correspond to one standard deviation. (a) Estimation error with respect to κ . (b) Policy optimization performance under a budget constraint of 2 with respect to κ .

estimation of the reward. In the case of the multiple treatments experiments, we see that CCPOvIRE performs similarly than CFRNet, while CCPOvSIRE improves over all the baselines for reward estimation as well as policy optimization. In particular, we outperform BanditNet by a significant margin on this dataset.

Simulating Structured Bandit Feedback from Nested Classification

Although algorithms for BLFB can be evaluated by simulating bandit feedback in a supervised learning setting (Agarwal et al. 2014), this approach is not compatible with our structured feedback setting. We therefore propose a novel approach to evaluate cost-effective incentive allocation algorithms. In particular, we use a model of nested classification with bandit-type feedback which brings ordered actions and structured feedback. While the vanilla setting relies on disjoint labels and therefore lacks an ordering between the actions, we simulate nested labels $(\mathcal{Y}_i)_{i \in \mathcal{I}}$, which are therefore monotonic with respect to the set inclusion. We construct a concrete example from the ImageNet dataset (Deng et al. 2009), by randomly selecting images with labels “Animal,” “Plant,” and “Natural object” and focusing on the nested labels “Animal” \supset “Arthropod” \supset “Invertebrate” \supset “Insect”. Let us now derive a compatible reward function. In this particular example of nested classification, say we observe a

dataset $(x_i, y_i^*) \sim \mathcal{P}$ where x_i is a pixel value and $y_i^* \in [K]$ is the perfect descriptor for this image among the K labels. Supervised learning aims at finding a labeling function $\Psi^{\text{SL}} : \mathcal{X} \rightarrow [K]$ that maximizes $\mathbb{E}_{(x, y^*) \sim \mathcal{P}}[\mathbb{1}_{\Psi^{\text{SL}}(x) = y^*}]$. Here, we are interested in a different setting where labels are nested and partial reward should be given when the labels is *correct* but not *optimal*, which corresponds to the reward $\mathbb{E}_{(x, y^*) \sim \mathcal{P}}[\mathbb{1}_{\Psi(x) \geq y^*}]$ (after appropriately permuting the labels). Without any additional constraints, the trivial labeling function that returns “Animal” yields maximal reward (as in Example 1). By adding a cost function c and budget constraint m , the learner will have to guess what is the optimal decision to take (i.e., with an optimal ratio of reward versus cost). Overall, the incentive allocation problem can be written as

$$\begin{aligned} \max_{\Psi} \mathbb{E}_{(x, y^*) \sim \mathcal{P}}[\mathbb{1}_{\Psi(x) \geq y^*}] \\ \text{such that } \mathbb{E}_{(x, y^*) \sim \mathcal{P}} c(\Psi(x)) \leq m. \end{aligned} \quad (13)$$

A logging policy, with the exact same form as in the fully-simulated experiment, is used to assign one of these four labels for each image. The reward function is 1 if the label is correct, or 0 otherwise. The corresponding costs for selecting these labels are $\{3, 2, 1, 0\}$. The dataset has 4608 samples in total, randomly split into training, validation and testing with ratio 0.6: 0.2: 0.2. The ratio of the positive and negative samples is equal to 1:10. Images are uniformly pre-processed, cropped to the same size and embedded into \mathbb{R}^{2048} with a pre-trained convolutional neural network (VGG-16). All results are obtained using the same parameters except the number of neurons for the hidden layers that is doubled. In particular, we searched for the optimal Lagrange multiplier in the space $\{1, 2, \dots, 10\}$ and returned the largest reward policy within budget constraint.

We report estimation errors as well as expected rewards for two different budgets in Table 2. As we already noticed in the simulations, the bias-correction step via HSIC significantly contributes to improving both the estimation and the policy optimization. Also, we note that after adopting the structured response assumption, both results are improved, which shows the benefit of exploiting the structure. We report results for different values κ for treatment estimation and policy optimization with $m = 2$ in Figure 1.

	ImageNet semi-simulations (mean \pm std)		
	RMSE	Reward ($m = 1$)	Reward ($m = 2$)
CCPOvSIRE	0.2408 \pm 0.0183	0.0934 \pm 0.0225	0.1408 \pm 0.0154
CCPOvSIRE [†]	0.2740 \pm 0.0255	0.0718 \pm 0.0063	0.1186 \pm 0.0198
CCPOvIRE	0.2712 \pm 0.0341	0.0839 \pm 0.0137	0.1304 \pm 0.0112
CCPOvIRE [†]	0.2943 \pm 0.0345	0.0674 \pm 0.0112	0.1157 \pm 0.0127
GANITE	0.3449 \pm 0.0236	0.0679 \pm 0.0217	0.0968 \pm 0.0367
BART	0.2867 \pm 0.0302	0.0492 \pm 0.0217	0.0927 \pm 0.0362
CFRNet	0.2480 \pm 0.0168	0.0861 \pm 0.0220	0.1335 \pm 0.0197
BanditNet	NA	0.0654 \pm 0.0265	0.0997 \pm 0.0367

Table 2: Performance with the ImageNet dataset. †: baselines with no HSIC ($\kappa = 0$). Bold indicates the method with the best performance for each dataset.

We notice that the range of parameters for which CCPOvSIRE improves over ablation studies is large. Furthermore, it is comforting that the same κ leads to the smallest estimation error as well as the best performance for policy optimization. Overall, the approach that does not exploit the reward structure (CCPOvIRE) performs similarly to CFRNet and BanditNet for policy optimization. However, CCPOvSIRE, which exploits the structure, outperforms all methods. Finally, it is interesting that the margin between the expected rewards changes for different values of budget. This may be attributable to discrepancies in hardness of detecting the different labels in images, which effectively modulates the difficulty of the incentive allocation depending on the budget.

7 Discussion

We have presented a novel framework for counterfactual inference based on BLBF scenario but with additional structure on the reward distribution as well as the action space. For this problem setting we have proposed CCPOvSIRE, a novel algorithm based on domain adaptation which effectively trades off prediction power for the rewards against estimation bias. We obtained theoretical bounds which explicitly capture this tradeoff and we presented empirical evaluations that show that our algorithm outperforms state-of-the-art methods based on ITE and CRM approaches. Since the provided plugin approach is not unbiased, further work introducing doubly-robust estimators (Dudik, Langford, and Li 2011) should lead to better performance.

Our framework involves the use of a nonparametric measure of dependence to debias the estimation of the reward function. Penalizing the HSIC as we do for each mini-batch implies that no information is aggregated during training about the embedding z and how it might be biased with respect to the logging policy. On the one hand, this is positive since we do not have to estimate more parameters, especially if the joint estimation would require solving a minimax problem as in (Yoon, Jordon, and van der Schaar 2018; Atan, Zame, and Van Der Schaar 2018). On the other hand, that approach could be harmful if the HSIC could not be estimated with only a mini-batch. Our experiments show this does not happen in a reasonable set of configurations. Trading a minimax problem for an estimation problem does not come for free. First, there are some computational consider-

ations. The HSIC is computed in quadratic time but linear-time estimators of dependence (Jitkrittum, Szabó, and Gretton 2017) or random-feature approximations (Pérez-Suay and Camps-Valls 2018) should be used for non-standard batch sizes.

Following up on our work, a natural question is how to properly choose the optimal κ , the regularization strength for the HSIC. In this manuscript, such a parameter is chosen with cross-validation via splitting the datasets. However, in a more industrial setting, it is reasonable to expect the central agent to have tried several logging policies which once aggregated into a mixture of deterministic policies enjoy effective exploration properties (e.g., in Strehl et al. (2010)). Future work therefore includes the development of a *Counterfactual Cross-Validation*, which would exploit these multiple policies and prevent propensity overfitting compared to vanilla cross-validation.

Another scenario in which our framework could be applied is the case of continuous treatments. That application would be natural in the setting of financial incentive allocation and has already been of interest in recent research (Kallus and Zhou 2018). The HSIC would still be an adequate tool for quantifying the selection bias since kernels are flexible tools for continuous measurements.

Aknowledgements

We thank Jake Soloff for useful discussions on nonparametric least squares estimators, which considerably helped framing Proposition 3 of this manuscript.

References

- Achiam, J.; Held, D.; Tamar, A.; and Abbeel, P. 2017. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*.
- Agarwal, A.; Hsu, D.; Kale, S.; Langford, J.; Li, L.; and Schapire, R. 2014. Taming the monster: A fast and simple algorithm for contextual bandits. In *Proceedings of the 31st International Conference on Machine Learning*.
- Alaa, A., and van der Schaar, M. 2018. Limits of estimating heterogeneous treatment effects: Guidelines for practical algorithm design. In *Proceedings of the 35th International Conference on Machine Learning*.
- Amos, B.; Xu, L.; and Kolter, J. Z. 2017. Input convex neural networks. In *International Conference on Machine Learning*.

- Atan, O.; Zame, W. R.; and Van Der Schaar, M. 2018. Counterfactual Policy Optimization Using Domain-Adversarial Neural Networks. In *ICML CausalML workshop*.
- Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Wortman, J. 2007. Learning Bounds for Domain Adaptation. In *Advances in Neural Information Processing Systems*.
- Chatterjee, S.; Guntuboyina, A.; and Sen, B. 2018. On matrix estimation under monotonicity constraints. *Bernoulli*.
- Daniels, H., and Velikova, M. 2010. Monotone and partially monotone neural networks. *IEEE Transactions on Neural Networks*.
- Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *Conference on Computer Vision and Pattern Recognition*.
- Ding, W.; Qin, T.; Zhang, X.-D.; and Liu, T.-Y. 2013. Multi-armed bandit with budget constraint and variable costs. In *AAAI*.
- Dudik, M.; Langford, J.; and Li, L. 2011. Doubly Robust Policy Evaluation and Learning. In *Proceedings of the 28th International Conference on Machine Learning*.
- Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*.
- Gao, F., and Wellner, J. A. 2007. Entropy estimate for high-dimensional monotonic functions. *Journal of Multivariate Analysis*.
- Gretton, A.; Bousquet, O.; Smola, A.; and Schölkopf, B. 2005. Measuring statistical dependence with Hilbert-Schmidt norms. In *Algorithmic Learning Theory*.
- Gretton, A.; Fukumizu, K.; Teo, C. H.; Song, L.; Schölkopf, B.; and Smola, A. J. 2008. A kernel statistical test of independence. In *Advanced in Neural Information Processing Systems*.
- Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. 2012. A Kernel Two-Sample Test. *Journal of Machine Learning Research*.
- Guntuboyina, A., and Sen, B. 2013. Covering numbers for convex functions. *IEEE Transactions on Information Theory*.
- Hill, J. L. 2011. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*.
- Jiang, N., and Li, L. 2016. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of The 33rd International Conference on Machine Learning*.
- Jitkrittum, W.; Szabó, Z.; and Gretton, A. 2017. An adaptive test of independence with analytic kernel embeddings. In *Proceedings of the 34th International Conference on Machine Learning*.
- Joachims, T.; Swaminathan, A.; and de Rijke, M. 2018. Deep Learning with Logged Bandit Feedback. In *International Conference on Learning Representations*.
- Johansson, F. D.; Shalit, U.; and Sontag, D. 2016. Learning Representations for Counterfactual Inference. In *Proceedings of the 33rd International Conference on Machine Learning*.
- Kallus, N., and Zhou, A. 2018. Policy evaluation and optimization with continuous treatments. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*.
- Kallus, N. 2019. Classifying treatment responders under causal effect monotonicity. In *Proceedings of the 36th International Conference on Machine Learning*.
- Lefortier, D.; Swaminathan, A.; Gu, X.; Joachims, T.; and de Rijke, M. 2016. Large-scale validation of counterfactual learning methods: A test-bed. In *What If workshop: NIPS*.
- Lopez, R.; Regier, J.; Jordan, M. I.; and Yosef, N. 2018. Information constraints on auto-encoding variational bayes. In *Advances in Neural Information Processing Systems*.
- Maurer, A., and Pontil, M. 2009. Empirical Bernstein Bounds and Sample Variance Penalization. In *Computational Learning Theory Conference*.
- Ng, A. Y., and Jordan, M. I. 2002. On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes. In *Advances in Neural Information Processing Systems*.
- Pearl, J. 2009. *Causality: Models, Reasoning and Inference*. Cambridge University Press.
- Pérez-Suay, A., and Camps-Valls, G. 2018. Sensitivity maps of the Hilbert-Schmidt independence criterion. *Applied Soft Computing Journal*.
- Pong, V.; Gu, S.; Dalal, M.; and Levine, S. 2018. Temporal Difference Models: Model-Free Deep RL for Model-Based Control. In *International Conference On Learning Representations*.
- Rubin, D. B. 2005. Causal inference using potential outcomes. *Journal of the American Statistical Association*.
- Schulman, J.; Levine, S.; Moritz, P.; Jordan, M. I.; and Abbeel, P. 2015. Trust Region Policy Optimization. In *Proceedings of the 32th International Conference on Machine Learning*.
- Sejdinovic, D.; Sriperumbudur, B.; Gretton, A.; and Fukumizu, K. 2013. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The Annals of Statistics*.
- Shalit, U.; Johansson, F. D.; and Sontag, D. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning*.
- Smola, A.; Gretton, A.; Song, L.; and Schölkopf, B. 2007. A hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*.
- Strehl, A.; Langford, J.; Kakade, S.; and Li, L. 2010. Learning from Logged Implicit Exploration Data. In *Advances in Neural Information Processing Systems*.
- Swaminathan, A., and Joachims, T. 2015a. Counterfactual risk minimization: Learning from logged bandit feedback. In *Proceedings of the 32nd International Conference on Machine Learning*.
- Swaminathan, A., and Joachims, T. 2015b. The Self-Normalized Estimator for Counterfactual Learning. In *Advances in Neural Information Processing Systems*.
- Wainwright, M. J. 2019. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press.
- Xu, Y.; Xu, Y.; and Saria, S. 2016. A non-parametric bayesian approach for estimating treatment-response curves from sparse time series. In *Proceedings of the 1st Machine Learning for Healthcare Conference*.
- Yoon, J.; Jordon, J.; and van der Schaar, M. 2018. GANITE: Estimation of individualized treatment effects using generative adversarial nets. In *International Conference on Learning Representations*.
- You, S.; Ding, D.; Canini, K.; Pfeifer, J.; and Gupta, M. 2017. Deep Lattice Networks and Partial Monotonic Functions. In *Advances in Neural Information Processing Systems 30*.