Aggregating Counterfactual Queries with Neural Architectures to Learn Complex Action Policies

Anonymous Authors

Abstract

We study the aggregation of multiple counterfactual queries when evaluating action policies in a contextual bandit scenario with batched data. This problem is commonly encountered since personalized treatments often lead to changes to multiple outcome variables. To reason with heterogeneous queries, we propose a class of loss functions composed of policy estimators built from a causal graph. Applying a model architecture inspired by deep learning, the base parameterized model can be jointly trained for the combination of queries. The trained model makes inference decisions on each data instance to maximize the desired aggregated counterfactual changes. Our method decomposes treatment into the binary treatment decision and action policy. This allows for policy learning on complex action spaces, including a blend of continuous and discrete treatments. We demonstrate the practical significance of our methodology by applying it to real-world data, and our method outperform benchmarks on two publicly available datasets.

Introduction

Algorithm driven decisions personalized to individuals is an important application of artificial intelligence systems. In search engines, online advertising, and recommendation systems, individual context is used as input for personalization models to rank web-pages, show advertisements, and suggest content. Contextual bandits algorithms have seen rising popularity to solve these problems by evaluating the desired counterfactual changes (Li et al. 2010) (Chu et al. 2011) (Li et al. 2015) (Swaminathan and Joachims 2015a) (Su et al. 2019), and utilizing action-reward observations (Joachims, Swaminathan, and de Rijke 2018) to learn policy models.

We call to attention a common class of problems where personalized treatment interventions lead to counterfactual changes to more than one outcome variable. These problems are ubiquitous when trading-off cost with reward. Personalized treatment policy in an online advertising campaign may lead to holistic growth in product sales, but will also cause an increment in spending. The collective result of the campaign may be measured by the cost effectiveness.

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

At the heart of this problem is *counterfactual queries* (Pearl 2014) (Koller and Friedman 2009) and the aggregation of queries for *counterfactual reasoning*. For the advertising campaign, cost effectiveness can be measured by the aggregation of counterfactual increase in sales divided by the counterfactual increase in cost. In previous studies, the direct method (DM) (Rubin 1974) is applied to estimate counterfactual queries of an outcome variable, supported by doubly robust estimation (Bang and Robins 2005), and inverse propensity scoring (IPS) to work with selection bias (Austin and Stuart 2015; Imai and Van Dyk 2004). Off-policy evaluation (OPE) for contextual bandits also evaluates the single counterfactual query with flexible policy functions (Li et al. 2010) (Kallus and Zhou 2018) (Demirer et al. 2019). In both approaches, models are built to first evaluate the counterfactual query for each outcome variable. To aggregate multiple counterfactual queries, model predictions are greedily combined. The combination of multiple machine learning models could increase the bias of the overall problem (Beygelzimer and Langford 2009). From an optimization perspective, the approach could result in local minima compared with a joint optimization algorithm.

In recent times, deep learning methods (LeCun, Bengio, and Hinton 2015) (Bengio, Courville, and Vincent 2012) (Krizhevsky, Sutskever, and Hinton 2012) have been proven effective when optimizing objectives through a hierarchy of learning units. We use this perspective to construct a policy learning algorithm that optimizes for the desired overall counterfactual results with aggregated queries. Rather than using counterfactual estimators to greedily maximize single outcomes, we apply parameterized models to represent effectiveness for individuals and treatment actions. These effectiveness measures are used to build estimators (Su et al. 2019) with conditional probability rules according to the causal graph. The aggregated objective for counterfactual reasoning can be formed with estimators with differentiable operators. Eventually, gradient methods are used to find optimal parameters in the base models for best action policies personalized to individuals.

Continuous action policies have been studied through the generalized propensity score approach (Imai and Van Dyk

2004) (Kreif et al. 2015), with semi-parametric models (Athey and Wager 2017) (Demirer et al. 2019), and using kernel functions (Kallus and Zhou 2018). We build on the works in off-policy evaluation to enable learning of both discrete and continuous policies for aggregated counterfactual queries. This paper presents a formulation that considers treatment decision, and action policy as different latent variables. We utilize the causal graph rules to seamlessly integrate policy learning into the model structure. The methodology then allows the optimization of aggregated counterfactual arguments across the portfolio of individuals, in a complex action space that includes continuous treatments.

As part of the action policy learning, our algorithm utilizes not only standard covariates but also covariates of meta selections. These selections affect the type of treatment individuals receive, such as web-pages recommended to users. We build on literature on relevance ranking (Huang et al. 2013), and structure the relevance of meta selection with individual into the action policy for joint learning.

Balancing estimators through inverse propensity scoring is a cornerstone method for correcting data bias (Austin and Stuart 2015). Balancing algorithms are also incorporated for policy evaluation models (Kallus 2018). Our method supports balancing of biased data for aggregated counterfactual arguments with multiple queries by incorporating inverse propensity scoring in the model structure.

Finally, we present evidence that our algorithms have practical significance. The methods are applied on two real-world datasets where success metrics with aggregated counterfactual queries can be directly measured. We compare the aggregated counterfactual learning algorithms with prior art. It is shown that leveraging policy learning with aggregated counterfactual queries, the proposed methods outperforms previous algorithms significantly in key desired metrics and in a statistically significant manner.

Problem Setup and Notation

We first present the algorithmic setup of the problem as represented in the scenario of contextual bandits learning from logged feedback (Li et al. 2015) (Swaminathan and Joachims 2015a) (Kallus and Zhou 2018) (Kallus 2018). Given observation for an individual with covariates $x \in \mathcal{X}$. Treatment intervention[1] that can be given to this individual is represented by the boolean variable T. The action policy π governs the treatment, and represents a function mapping from the covariates x to $\rho = \pi(x)$, where ρ represents the policy action, which can be a multinomial choice $\rho = \{1, 2, ..., K\}$ or a continuous variable, e.g. $\rho \sim \mathcal{N}(\mu, \sigma^2)$. The overall policy can be a collection of independent action policies, e.g. $\pi = \{\pi_1, \pi_2, ...\}$. For the no-treatment individuals, there is no action taken on them and $T = 0$. Note no action is taken, even though $\rho = \pi(x)$ could indicate the best action that could be taken on this individual.

The outcome for this individual is denoted Y, so logged data is in the form of $\{x^{(i)}, T^{(i)}, \rho^{(i)}, Y^{(i)}\} \in \mathcal{D}$. This data is assumed to be i.i.d. distributed. From this logged data, the algorithm learns the optimal policy π, which in turn determines the optimal policy action $\rho^{(i)}$ given the individual covariates. This paper considers learning from the data in batches rather than in an online fashion.

In contextual bandit settings, the algorithm's goal is to maximize reward which can be the outcome variable, such as the click-rate of a web-page (Chu et al. 2011) (Kallus and Zhou 2018). Here, the optimal policy is $\pi^* = \arg\max_{\pi} v_\pi$.

$$v_\pi = E_\pi(Y|x) \tag{1}$$

A counterfactual query considers changes an intervention treatment incurs in an outcome variable (Koller and Friedman 2009). This can be quantified by the expected value of difference between the outcome variable given and not given the intervention. $\tau_Y(x) = E_{\pi,T=1}(Y|x) - E_{\pi,T=0}(Y|x)$. This notation is in accordance with definition of the treatment effect function (Holland 1986) (Künzel et al. 2017). The functional form is used by policy learners such as off-policy evaluation (OPE) (Wang, Agarwal, and Dudık 2017) (Demirer et al. 2019), and for treatment effect estimation algorithms (Künzel et al. 2017) (Nie and Wager 2017):

$$v_\pi = E_{\pi,T=1}(Y|x) - E_{\pi,T=0}(Y|x) = \tau_Y(x) \tag{2}$$

For off-policy evaluation, the function is fitted to the logged data $\{x^{(i)}, T^{(i)}, \rho^{(i)}, Y^{(i)}\} \in \mathcal{D}$, then can be used to estimate the treatment effect given covariates of an individual and the policy function. For our methods, we make the unconfoundedness assumption, $Y_{T=0}, Y_{T=1} \perp\!\!\!\perp T|x$, same as previous works (Nie and Wager 2017).

Instead of using a single counterfactual query for off-policy evaluation with a contextual bandits model, we propose the paradigm to form counterfactual reasoning by aggregating multiple queries. In the case where $\tau_{Y_a} > 0$, and $\tau_{Y_b} > 0$[2], the aggregated query as effectiveness for the example of the advertising campaign, can be formed by assigning Y_a to be the sales, and Y_b to be the cost. The aggregated query is defined as:

$$\omega_\pi = \frac{\tau_{Y_a}}{\tau_{Y_b}} = \frac{E_{\pi,T=1}(Y_a|x) - E_{\pi,T=0}(Y_a|x)}{E_{\pi,T=1}(Y_b|x) - E_{\pi,T=0}(Y_b|x)} \tag{3}$$

The counterfactual reasoning is that ω_π represents the cost effectiveness (incremental reward Y_a over incremental cost Y_b) if the system switches to using action policy π. The goal of the learning algorithm is to find the optimal policy that maximizes the cost effectiveness, thus $\pi^* = \arg\max_{\pi} \omega_\pi$. This optimization will result in an optimal policy model to determine the treatment actions $\rho = \pi(x)$ per individual that leads to the highest cost-effectiveness out of all possible campaign strategies. Counterfactual queries could be aggregated as long as the operators are differentiable, an example is $\omega_\pi = \tau_{Y_a}\tau_{Y_b}$.

Our approach is different from single-objective contextual bandit policy learning (Chu et al. 2011) (Li et al.

[1] Our problem setting assumes the relationship between assignment and treatment is deterministic, i.e. treatment is given with probability 1 if assigned.

[2] Division is a differentiable operator given both numerator and denominator are positive.

2010) (Swaminathan and Joachims 2015a) since we reason with aggregate counterfactual queries and optimize the policy jointly. This is also true compared with off-policy evaluation (OPE) with a single counterfactual query. Instead of evaluating change across on-off policies, our method learns the optimal policy per individual across the batched data for the aggregated counterfactual objective.

Existing Approaches and Related Work

While initially presented in the causal and counterfactual inference literature (Pearl 2009; 2014; Koller and Friedman 2009), the efficient estimation of counterfactual queries is well studied both from the contextual bandits with logged feedback (Swaminathan and Joachims 2015a; 2015b; Kallus and Zhou 2018; Kallus 2018; Su et al. 2019), and causal inference perspective, commonly know as the 'direct method' for estimating treatment effects. (Rubin 1974; Shalit, Johansson, and Sontag 2017; Künzel et al. 2017; Nie and Wager 2017; Athey and Wager 2017).

The literature for learning contextual bandits on logged feedback studies the learning principles (Dudík, Langford, and Li 2011; Wang, Agarwal, and Dudık 2017; Swaminathan and Joachims 2015a; 2015b), and constructs robust policy estimators (Su et al. 2019) for strong generalization performance. Policy evaluation and learning is approached extensively with learning generic continuous policies (Kallus and Zhou 2018), discrete action and neural network policy models (Joachims, Swaminathan, and de Rijke 2018) and effective semi-parametric policy models (Demirer et al. 2019).

The direct methods (DM) in causal inference uncovers a range of algorithms from the treatment effect estimation paradigm (Rubin 1974), to inverse propensity scoring (IPS) (Lunceford and Davidian 2004; Austin and Stuart 2015; Imai and Van Dyk 2004; Glynn, Schneeweiss, and Stürmer 2006; Wooldridge 2007; Curtis et al. 2007), to doubly robust estimation methods which combines DM and IPS (Bang and Robins 2005; Dudík, Langford, and Li 2011; Funk et al. 2011). The generalization of propensity score can be used to deal with discrete and continuous treatments (Imai and Van Dyk 2004). The DM method could be supplemented with generalized propensity score, and Super Learner to improve model selection (Kreif et al. 2015). The studies gave rise to generalizations of powerful statistical methods, named *meta-learners* (Künzel et al. 2017). This area has also seen application of effective regression-tree, decision-tree based methods (Wager and Athey 2018; Athey and Imbens 2016) together with detailed studies on tree-based policy learners (Athey and Wager 2017).

We take note the important areas of contextual bandits applications. This research area focus on efficient algorithms (Chu et al. 2011; Li et al. 2010; 2015), and calls to light the practical considerations in industry (Bottou et al. 2013), with relation to traffic and A/B experiments (Xie, Chen, and Shi 2018) commonly used in web services.

Different from prior approaches, our proposed methods focus on the counterfactual reasoning and optimization when aggregating multiple counterfactual queries. This is especially important in applications where cost-reward trade-off exists. Building on prior art, our method also details the policy learning algorithms to individually determine the best treatment. To the best of our knowledge, this paper presents a novel discussion and perform experiments to illustrate the algorithm for aggregated counterfactual queries.

It's worth discussing application of deep learning methods for contextual bandits such as (Joachims, Swaminathan, and de Rijke 2018) (Riquelme, Tucker, and Snoek 2018), balancing the representation space (Shalit, Johansson, and Sontag 2017) and adapting recurrent networks to study the effects of sequential treatments (Lim 2018). In this work, we leverage deep learning architectures, perspectives and methodology. This will be presented in detail in the next section.

Gating Networks for Measuring Effectiveness

We start with a central task for our approach: represent and parameterize the measure for effectiveness with a neural model architecture. This effectiveness is correlated with the cost effectiveness in our example, yet it is defined on the individual level. Each individual in the logged data, irrespective of their treatment T values, should be mapped into an ordinal effectiveness measure, thus can be ranked as higher or lower. Concretely, we define this measure w to correlating with the base policy model π_0: $w^{(i)} = \pi_0(x^{(i)})$. We use the softmax function to build a pair of gating networks for the treated and no-treatment groups of individuals:

$$p^{(i)}_{T=1} = \frac{\exp(w^{(i)})}{\sum_{T=1}\exp(w^{(i)})},\ p^{(j)}_{T=0} = \frac{\exp(w^{(j)})}{\sum_{T=0}\exp(w^{(j)})} \tag{4}$$

As the π_0 base policy is a functional mapping from covariates, it is defined as a parameterized representation, e.g. $\pi_0(x) = \tanh(\theta^T x)$ or $\pi_0(x) = f_\theta(x)$ where f is a functional mapping defined by a neural network. This is our base parameterized model. The pair of gating networks borrows their neural architecture from multinomial logistic regression (Böhning 1992), or neural architecture for mixture of experts (Shazeer et al. 2017; Jordan and Jacobs 1994). Leveraging these gating networks, it's possible to express the expectations in equation 3 while retaining the parametric policy model.

$$E_{\pi,T=1}(Y|x) = \sum_{T=1} p^{(i)}_{T=1} Y^{(i)},\ E_{\pi,T=0}(Y|x) = \sum_{T=0} p^{(j)}_{T=0} Y^{(j)} \tag{5}$$

In the next section, we aggregate counterfactual queries for the learning objective function, which enables learning of the parameters θ.

Aggregating Counterfactual Queries for Policy Learning

With representations in equation 5, the aggregated counterfactual query ω_π in equation 3 is expanded with the effectiveness measures $p^{(i)}$. The numerator and denominator are also confined by the soft ReLU rectifier units σ_r. This ensures the objective is differentiable.

$$\omega_{\pi_\theta} = \frac{\sigma_r(\sum_{T=1} p^{(i)}_{T=1} Y^{(i)}_a - \sum_{T=0} p^{(j)}_{T=0} Y^{(j)}_a)}{\sigma_r(\sum_{T=1} p^{(i)}_{T=1} Y^{(i)}_b - \sum_{T=0} p^{(j)}_{T=0} Y^{(j)}_b)} \tag{6}$$

The form of aggregated counterfactual queries given in equation 6 is a learning objective for individualized action policy. The objective function can be formed by considering all data points $\{x^{(i)}, T^{(i)}, \rho^{(i)}, Y^{(i)}\} \in \mathcal{D}$ with $(T = 1)$ and without $(T = 0)$ treatment in the training set. To learn the policy, we use gradient methods to find optimal parameters $\theta^* = \arg\max_\theta \omega_{\pi_\theta}$.

From (Lunceford and Davidian 2004) and causal statistics, the expected value of outcome of any instance can be written as following equations. $E_{\pi,T=1}(Y|x) = E_\pi(\frac{Y_{T=1}T}{e(x)}), E_{\pi,T=0}(Y|x) = E_\pi(\frac{Y_{T=0}(1-T)}{1-e(x)})$, where $e(x)$ is the propensity function defined as $e(x) = E(T = 1|x)$. It can be learned then applied on covariates commonly applied with inverse propensity scoring (IPS) methods (Imai and Van Dyk 2004; Glynn, Schneeweiss, and Stürmer 2006). The aggregated query objective becomes the following equation. The detailed derivation is presented in the supplementary materials.

$$\omega_\pi = \frac{\sigma_r(\hat{e}\sum_{T=1}\frac{p^{(i)}_{T=1}}{e(x)}Y^{(i)}_a - (1-\hat{e})\sum_{T=0}\frac{p^{(j)}_{T=0}}{1-e(x)}Y^{(j)}_a)}{\sigma_r(\hat{e}\sum_{T=1}\frac{p^{(i)}_{T=1}}{e(x)}Y^{(i)}_b - (1-\hat{e})\sum_{T=0}\frac{p^{(j)}_{T=0}}{1-e(x)}Y^{(j)}_b)} \tag{7}$$

The $\hat{e}$ in this equation is the example average of $E(T = 1)$, and propensity function $e(x)$ is pre-trained, then evaluated on the data-set to produce scalars in equation 7. Our method support the use of inverse propensity scoring to deal with dataset bias.

Learning Policies on Complex Action Spaces

The effectiveness measure $p^{(i)}$ can be interpreted to be a probability since the gating network architecture normalizes the measure with $\sum_{T=1} p^{(i)}_{T=1} = 1, \sum_{T=0} p^{(i)}_{T=0} = 1$. Also, we illustrate the derivation from causal statistics in the supplementary materials, showing the equivalent probability quantity of $p^{(i)} = P(X = x^{(i)}|\pi, x^{(i)}) = P(X = x^{(i)}|\rho^{(i)})$. Further, in the causal graphical model across $P(X, \rho, T, Y)$, the impact of T to Y was detailed in equation 6, while a directed edge exists from ρ to Y. Leveraging the conditional probability rules in the causal graph, we can unravel the relationship between policy action and the effectiveness of the individual using the Bayes rule.

$$p^{(i)} = \frac{P(\rho^{(i)}|x^{(i)})P(x^{(i)})}{\sum_j P(\rho^{(j)}|x^{(j)})P(x^{(j)})} \tag{8}$$

The policy action $\rho^{(i)}$ can be decomposed into multiple action spaces $\rho^{(i)} = (\rho_c, \rho_m)$. We assume ρ_c, ρ_m are independent given the covariates $x^{(i)}$.

$$P(\rho^{(i)}|x^{(i)}) = P(\rho^{(i)}_c|x^{(i)})P(\rho^{(i)}_m|x^{(i)}) \tag{9}$$

With this decomposition, π_0 could be defined as the intrinsic treatment policy function for an individual, related to the quantity $P(x^{(i)})$; To incorporate action policies, the form of π_0 can be chose as a sigmoid compressed version of arbitrary differentiable function.[3]. π_c is the continuous policy function for intensity of the treatment, relating to the quantity $P(\rho^{(i)}_c|x^{(i)})$; while π_m is the meta selection policy function for discrete treatment choices, relating to the quantity $P(\rho^{(i)}_m|x^{(i)})$. As with π_0, the parameterization of π_c and π_m could be a flexible mapping from covariates defined by a neural network, e.g. $\pi_c(x) = f_{\theta_c}(x)$, $\pi_m(x) = f_{\theta_m}x$. Here, the policy model parameters θ_c, θ_m can be jointly learned.

Continuous policy model π_c. Prior approaches for continuous policy learning applies generalizations of the propensity score (Imai and Van Dyk 2004; Kreif et al. 2015) for evaluation of counterfactual queries of treatment effect, or use infinitesimal nudges with application of regression trees (Athey and Wager 2017). Others apply kernel functions, semi-parametric forms with off-policy evaluation (Kallus and Zhou 2018; Demirer et al. 2019). Our method directly uses the relationship in equation 8 to structure the continuous policy function into the model architecture.

We could consider likelihood $P(\rho_c|x^{(i)})$ to be estimated from a distribution with with a continuous scope that can be parameterized by $\pi_c(x^{(i)})$:

$$P(\rho^{(i)}_c|x^{(i)}) \propto D(\rho_c|\pi_c(x^{(i)})) \tag{10}$$

$$\propto \sigma(\hat{\rho}^{(i)})(1 - \sigma(\hat{\rho}^{(i)})) = h_c(\hat{\rho}^{(i)})) \tag{11}$$

Here $\hat{\rho}^{(i)} = \rho - \pi_c(x^{(i)})$. In this formulation, the policy model determines hyper-parameters of the continuous distribution, which then determines the likelihood of any continuous action value. The distribution here is chosen as a bell-shaped form, the derivative of the sigmoid. Given $\pi_c(x)$, the distribution is distinct per individual, and offers measure for the goodness of policy values. For example, when π_c denotes the mean of a bell-shaped distribution, the policy model $\pi_c(x)$ should give the optimal treatment intensity for the individual. During training, if the data deviates from this optimal value, its likelihood would be penalized with respect to the amount of deviation. This is shown in Figure 1. The likelihood function can also be defined with other distributions, such as the *Beta* distribution[4].

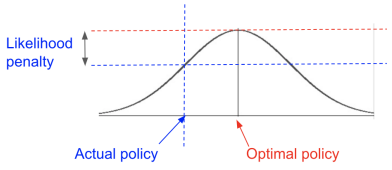

Figure 1: Penalty of a sub-optimal policy.

Meta-selection policy model π_m. Meta-selection is when we have a selection of choices to treat the individual, such as matching a user with a web-page. We utilize a helper policy

[3]With sigmoid non-linearity, the outputs are positive and are well-controlled above 0, below 1.0, useful for normalization with partition functions.

[4]This is detailed further in the supplementary materials.

function $\hat{\pi}_m$ to characterize $P(\rho_m|x^{(i)})$. Here we offer the opportunity to extend the covariates x to include the meta-selection $x = (x_o, x_m)$ so x_o is the covariates of the individual, and x_m is covariates of the meta selection. The relationship between $P(\rho_m|x_o^{(i)}, x_m^{(i)})$ and $\hat{\pi}_m$ is as follows.

$$P(\rho_m|x_o^{(i)}, x_m^{(i)}) \propto \hat{\pi}_m(x_o^{(i)}, x_m^{(i)}) \tag{12}$$

The meta selection policy model can be specified as $\pi_m(x) = \arg\max_m \hat{\pi}_m(x)$. The input covariates $x^{(i)} = (x_o^{(i)}, x_m^{(i)})$ are projected with a neural network to an embedding space: $\mathbf{e}_o = f_o(x_o^{(i)}), \mathbf{e}_m = f_m(x_m^{(j)})$. The embedding space can be learned while policy functions are learned. The algorithm for matching here is related to ranking models such as (Huang et al. 2013). We formulate the helper policy function as the cosine distance across two embeddings plus one.

$$\hat{\pi}_m(x_o^{(i)}, x_m^{(i)}) = 1 + cos(\phi) = 1 + \frac{\mathbf{e}_o \cdot \mathbf{e}_m}{\|\mathbf{e}_o \cdot \mathbf{e}_m\|} \tag{13}$$

Normalization in the neural architecture. The policy functions need to be normalized to ensure the effectiveness measures are aligned with the probabilistic quantities they represent. We do this using a deep learning model architecture with Tensorflow (Abadi et al. 2016).

Concretely, having ensured the policy functions are positive and bounded, we sum the scores together to form the partition function then encode normalization operations in the model architecture. This is done for across data instances in respective treatment and no-treatment groups. The normalization is done three times, first for normalizing h_c in equation 10 and $\hat{\pi}_m$ when combined in equation 9; the second is the combination in equation 8; finally the third normalization is performed with the softmax gating functions.

Objective for aggregated counterfactual queries. Summing up the parameterized policy model, matching model, and normalization, we can use the overall objective function[5] to optimize for parameters in all policy functions, namely π_0, π_c, and π_m. All operators are differentiable so we use gradient based numerical optimizers to solve for the policy parameters which maximizes the combined aggregated counterfactual queries. In practice, we apply improved gradient optimizers such as Adagrad (Duchi, Hazan, and Singer 2011) to find best policy parameters.

Empirical Experiments

Benchmark Models

We benchmark our method with off-policy evaluation and treatment effect estimation algorithms. The compared methods from prior work estimates multiple outcomes with separate models. In this context, two mainstream methodologies are *meta-learners* (Künzel et al. 2017) (Nie and Wager 2017) and causal trees and forests (Wager and Athey 2018). We compare with the most representative algorithm in literature, the quasi-oracle estimation algorithm, and causal forests.

Quasi-oracle estimation. We use linear regression[6] as the base estimator. Since the experiment treatments are randomly given, we use constant treatment percentage as propensity. We use the the model to learn the reward incrementality across treatment and control with an conditional average treatment effect function τ for each outcome dimension. Each sample in the test set is evaluated for the counterfactual query, and eventual metric is computed by combining the counterfactual query of all outcomes. For instance, in the case of maximizing Equation 3, we would train an estimator for each of the a, b dimensions, then for each sample in the dataset, we compute the predictions for each of τ_a, τ_b, then compute score $s = \tau^a(x)/\tau^b(x)$ for evaluation.

Causal Forest. We leverage the generalized random forest (*grf*) library in R (Wager and Athey 2018) (GRF). For details, we apply causal forest with 50 trees, 0.2 as alpha, 3 as the minimum node size, and 0.5 as the sample fraction. We train multiple forests, then apply the ratio of counterfactual queries to rank individuals. For hyperparameters, we perform search on deciles for parameters *num_trees*, *min.node.size*, and at 0.05 intervals for *alpha*, *sample.fraction* parameters. We also leverage the *tune.parameters* option for the *grf* package, eventually, we found best parameters through best performance on validation set[7].

Aggregated Ranking Model (Simplified CT Model). We use a simple parameterization for aggregated counterfactual queries as a benchmark model. We use a scoring function similar to logistic regression, i.e. $\sigma(\mathbf{w}^T x + b)$, without continuous or meta-selection policies. The model is trained without weight regularization. We use the Adam optimizer with learning rate 0.01 and default beta values. We compute gradients for entire batch of data for optimization. For hyperparameter selection, we take the best validation set performance out of 6 random initializations.

Continuous Treatment Policy Matching Model (CTPM). We implement our deep learning based models with Tensorflow (Abadi et al. 2016). The model architecture utilizes deep learning to build aggregated counterfactual queries as objectives, and continuous policy model for choosing actions. We use two-layer neural networks with the same number of first-layer units in meta-selection and continuous policy models [8]. Adam optimizer used learning rate 0.01, default beta values. We run the same number of iterations as simple CT model[9]. We take best validation set performance out of 6 random intitializations.

Datasets

Ponpare Data The Ponpare dataset is a public dataset from a coupons website (Ponpare). The dataset is well-suited to evaluate our proposed methodology since it offers multiple outcome variables, such as purchase and cost ground-truth, as well as the continuous discount levels of the coupons. The sessions also matches user with coupon as

[5]Detailed in the supplementary materials

[6]*SKLearn*'s ridge regression with zero regularization.

[7]Best parameters we experimented: *num_trees*$= 50$, *alpha*$= 0.2$, *min.node.size*$= 3$, *sample.fraction*$= 0.5$

[8]Number of hidden units is determined by validation results, 15 units for Ponpare dataset and 8 units for USCensus.

[9]Due to variance-bias trade-off across datasets, both CTPM and simple CT models are run for 2500 iterations for Ponpare dataset and 650 iterations for USCensus

meta-selection. We leverage the open-source feature engineering code provided by (Pon-Features). The causal inference scenario focuses on estimating the combined benefits when we offer a continuous and variable discount percentage given a user-coupon match. We pre-filter the sessions where customers are below the age of 45. Due to disproportion of positive and negative samples, we subsample 4.0% of sample of sessions that do not result in purchase. The eventual dataset is around $130,822$ samples, we utilize discount level as the continuous treatment policy, and use the median of the level to segment out sessions into treatment and control groups, indicated by binary variable T, Discount level is subsequently used as continuous policy ρ_c. For this dataset, we apply the aggregated counterfactual queries objective and minimize $\frac{\tau_c}{\tau_r} + \lambda\tau_m$. This is slightly changed from equation 3 with τ_c as treatment effect for absolute discount amount with reference to cost, τ_r the purchase boolean variable with reference to benefit, and τ_m the geographical distance from user to the product location for the coupon as extra cost related to delivery or travel. The λ variable is chosen to be fixed at 0.1 across all models with the goal of adding distance factor into the objective.

US Census 1990 The US Census (1990) Dataset (Asuncion & Newman, 2007 (usc) contains data for people in the census. Each sample contains a number of personal features (native language, education...). The features are prescreened for confounding variables, we left out dimensions such as other types of income, marital status, age and ancestry. This reduces features to d = 46 dimensions. Before constructing experiment data, we first filter with several constraints. We select people with one or more children (*'iFertil'* ≥ 2)[10], born in the U.S. (*'iCitizen'* = 0) and less than 50 years old (*'dAge'* < 5), resulting in a dataset with $225,814$ samples. We select 'treatment' label as whether the person works more hours than the median of everyone else, and select the income (*'dIncome1'*) as the gain dimension of outcome for τ_r, then the number of children ('iFertil') multiplied by -1.0 as the cost dimension for estimating τ_c. The hypothetical meaning of this experiment is to measure the cost effectiveness, and evaluate who in the dataset is effective to work more hours. We apply optimization problem to maximize $\tau_m(\tau_r - \lambda\tau_c)$ as comparison with Ponpare Dataset with τ_r as treatment effect in *income*, τ_c as treatment effect in negative value of *number of offspring*, and τ_q as effect on married or not as an overall weighting factor across the objective. This gives the objective hypothetical meaning of utility. The λ variable is chosen to be fixed at 3.0 across all models to add a fixed cost weighting factor across income and offspring cost.

For both datasets, we split training, validation and test with ratios 60%, 20%, 20%.

Evaluation Methodology We evaluate the algorithms in two ways. The first evaluation is Aggregated Counterfactual (Treatment Effect) To Percentage (ATETP). This measure compute the effectiveness measure on the test data-set, then take an increasing percentage of the test set as to evaluate the average treatment effect according to the pre-defined causal metric in equation 3. If the model scores the matches and treatment policies well, the ATETP should be high across the lower spectrum of percentages. We also use the ATETP area under curve (termed a-AUC) to be a numerical measure. The secondary metric is to plot a cost curve, i.e. to plot the counterfactual query on reward τ_r versus cost τ_c as we increase percentage of coverage in the test set. This measure sees cost versus reward as the main concern, and we also compute the area under curve (termed c-AUC) to numerically measure performance. [11]

Experiment Results Figure 2 and Figure 3 show results of causal models on Ponpare dataset[12]. The CTPM outperforms quasi-oracle estimation, and simplied CT model on both ATETP curve and cost-curve. With peak at 10-20% treatment, the CTPM produces ATE improvement at the most effective match instances across user and coupons. For cost curve, CTPM also outperforms other models.

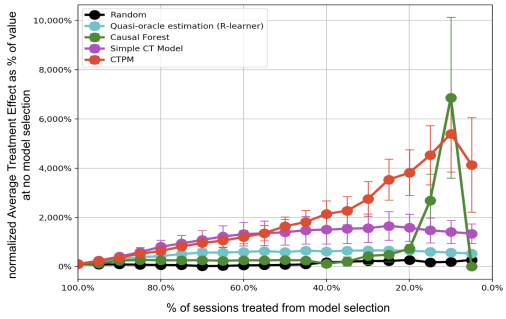

Figure 2: Aggregated counterfactual to percentage for Ponpare data.

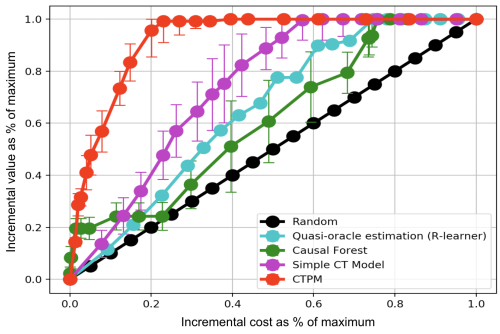

Figure 3: Secondary measure cost curve for Ponpare data.

Figure 4 and Figure 5 show results of the CTPM on US Census. We observe higher ATE for the CTPM model in high-scored instances. CTPM could identify the most incremental instances without significant differences in cost. The model outperforms baseline quasi-oracle estimation and simplified CT model significantly both on the ATETP and cost curve measures.

[10] *'iFertil'* field is off-set by 1, *'iFertil'*$= 0$ indicating ≤ 15 year old male, *'iFertil'*=1 no children.

[11] For both a-AUC and c-AUC, the higher the measure the better.

[12] Standard deviations across 6 runs are indicated for both Ponpare and USCensus

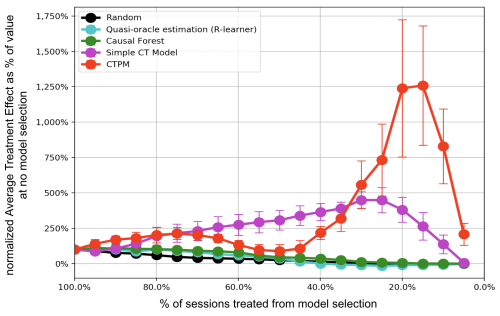

Figure 4: Aggregated counterfactual to percentage for US Census.

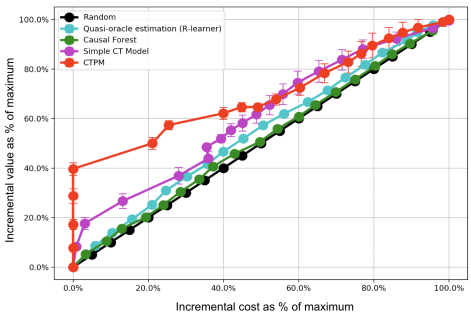

Figure 5: Secondary measure cost curve for US Census.

Table 1 summarizes results of the continuous treatment policy matching model. On Ponpare dataset, CTPM outperforms quasi-oracle estimation by more than *3*× and improves 67% upon aggregated ranking model in a-AUC. For c-AUC, CTPM improves 41% upon quasi-oracle estimation and improves *2*× upon aggregated ranking model. On US-Census dataset, CTPM performs *8*× better in terms of a-AUC than quasi-oracle estimation, and out-performs aggregated ranking model by around *42%*. CTPM is more cost effectiveness in terms of c-AUC by 28% compared with quasi-oracle estimation, and improvement 13% upon the aggregated ranking model.

Analysis and Interpretation The continuous policy model is able to predict the optimal treatment intensity. In Figure 6, we visualize the optimal discount per session for the genre *'Health'* in the Ponpare test set. Compared with original treatment intensities, the optimal intensities from model prediction shows apparent segregation of low vs high intensity recommendations. This is shown by the data clusters near zero percentage (green oval), and near full percentage (orange oval).

Table 1: Summary of results across models and datasets.

Algo/Dataset	Ponpare		USCensus	
Eval. Metric	a-AUC	c-AUC	a-AUC	c-AUC
Random	1.15	0.50	0.31	0.50
Quasi-O	5.06	0.65	0.40	0.54
Causal Forest	6.58	0.61	0.53	0.51
Simple CT	11.12	0.74	2.47	0.61
CTPM	**18.57**	**0.92**	**3.51**	**0.69**

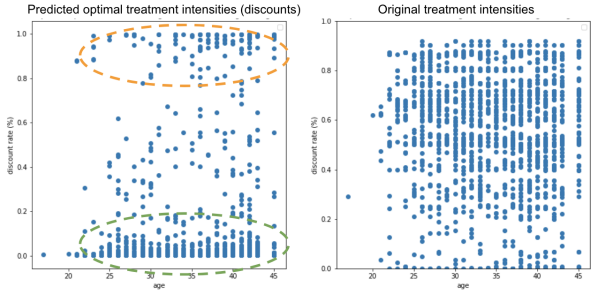

Figure 6: Scatter plot comparison across optimal predictions from model (left) and original treatment intensities (right).

Figure 7 shows the results of the learned embeddings by our model on the Ponpare dataset. The embedding space is jointly learned across the individuals and the coupons. Figure 7 plots the embeddings projected by the model using 2D t-distributed stochastic neighbor embedding (van der Maaten and Hinton 2008) (t-SNE)[13]. We can see the learned subject embeddings are organized by gender with two separable clusters.

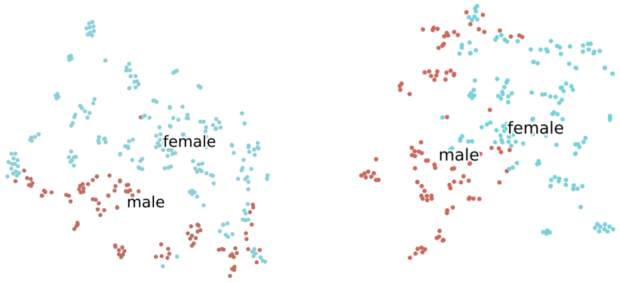

Figure 7: Visualization of CTPM user embeddings for age group 30 (left) and 44 (right) from Ponpare dataset using T-SNE with color indicating user gender.

Conclusion and Discussion

In this paper, we proposed a model that jointly optimizes aggregated counterfactual queries. This method is suitable for contextual bandits while able to learn continuous space, meta-selection policy models. This method differentiates from prior work by considering multiple counterfactual queries and utilizes deep learning architectures for joint optimization. We show the algorithm performs well on public datasets, especially handling scenarios when trading off cost with reward. For future work, our proposal offers potential to combine with other deep learning techniques such as sequential, recurrent models, generative models, and can be potentially extended and applied to other scientific domains.

[13]The parameters for t-SNE are learning rate of 30, perplexity of 20.

References

Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 265–283.

Athey, S., and Imbens, G. 2016. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences* 113(27):7353–7360.

Athey, S., and Wager, S. 2017. Efficient policy learning. *arXiv preprint arXiv:1702.02896*.

Austin, P. C., and Stuart, E. A. 2015. Moving towards best practice when using inverse probability of treatment weighting (iptw) using the propensity score to estimate causal treatment effects in observational studies. *Statistics in medicine* 34(28):3661–3679.

Bang, H., and Robins, J. M. 2005. Doubly robust estimation in missing data and causal inference models. *Biometrics* 61(4):962–973.

Bengio, Y.; Courville, A. C.; and Vincent, P. 2012. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR, abs/1206.5538* 1:2012.

Beygelzimer, A., and Langford, J. 2009. The offset tree for learning with partial labels. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 129–138.

Böhning, D. 1992. Multinomial logistic regression algorithm. *Annals of the institute of Statistical Mathematics* 44(1):197–200.

Bottou, L.; Peters, J.; Quiñonero-Candela, J.; Charles, D. X.; Chickering, D. M.; Portugaly, E.; Ray, D.; Simard, P.; and Snelson, E. 2013. Counterfactual reasoning and learning systems: The example of computational advertising. *The Journal of Machine Learning Research* 14(1):3207–3260.

Chu, W.; Li, L.; Reyzin, L.; and Schapire, R. 2011. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 208–214.

Curtis, L. H.; Hammill, B. G.; Eisenstein, E. L.; Kramer, J. M.; and Anstrom, K. J. 2007. Using inverse probability-weighted estimators in comparative effectiveness analyses with observational databases. *Medical care* S103–S107.

Demirer, M.; Syrgkanis, V.; Lewis, G.; and Chernozhukov, V. 2019. Semi-parametric efficient policy learning with continuous actions. *arXiv preprint arXiv:1905.10116*.

Du, S.; Lee, J.; and Ghaffarizadeh, F. 2019. Improve user retention with causal learning. In *The 2019 ACM SIGKDD Workshop on Causal Discovery*, 34–49.

Duchi, J.; Hazan, E.; and Singer, Y. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research* 12(7).

Dudík, M.; Langford, J.; and Li, L. 2011. Doubly robust policy evaluation and learning. *arXiv preprint arXiv:1103.4601*.

Funk, M. J.; Westreich, D.; Wiesen, C.; Stürmer, T.; Brookhart, M. A.; and Davidian, M. 2011. Doubly robust estimation of causal effects. *American journal of epidemiology* 173(7):761–767.

Glynn, R. J.; Schneeweiss, S.; and Stürmer, T. 2006. Indications for propensity scores and review of their use in pharmacoepidemiology. *Basic & clinical pharmacology & toxicology* 98(3):253–259.

GRF. Grf: Generalized random forests. https://grf-labs.github.io/grf/. Accessed: 2019-11-15.

Holland, P. W. 1986. Statistics and causal inference. *Journal of the American statistical Association* 81(396):945–960.

Huang, P.-S.; He, X.; Gao, J.; Deng, L.; Acero, A.; and Heck, L. 2013. Learning deep structured semantic models for web search using clickthrough data. ACM International Conference on Information and Knowledge Management (CIKM).

Imai, K., and Van Dyk, D. A. 2004. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association* 99(467):854–866.

Joachims, T.; Swaminathan, A.; and de Rijke, M. 2018. Deep learning with logged bandit feedback. In *International Conference on Learning Representations*.

Jordan, M. I., and Jacobs, R. A. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural computation* 6(2):181–214.

Kallus, N., and Zhou, A. 2018. Policy evaluation and optimization with continuous treatments. *arXiv preprint arXiv:1802.06037*.

Kallus, N. 2018. Balanced policy evaluation and learning. In *Advances in neural information processing systems*, 8895–8906.

Koller, D., and Friedman, N. 2009. *Probabilistic graphical models: principles and techniques*. MIT press.

Kreif, N.; Grieve, R.; Díaz, I.; and Harrison, D. 2015. Evaluation of the effect of a continuous treatment: a machine learning approach with an application to treatment for traumatic brain injury. *Health economics* 24(9):1213–1228.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Künzel, S. R.; Sekhon, J. S.; Bickel, P. J.; and Yu, B. 2017. Meta-learners for estimating heterogeneous treatment effects using machine learning. *arXiv preprint arXiv:1706.03461*.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature* 521(7553):436–444.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, 661–670. ACM.

Li, L.; Chen, S.; Kleban, J.; and Gupta, A. 2015. Counterfactual estimation and optimization of click metrics in search

engines: A case study. In *Proceedings of the 24th International Conference on World Wide Web*, 929–934.

Lim, B. 2018. Forecasting treatment responses over time using recurrent marginal structural networks. In *Advances in Neural Information Processing Systems*, 7483–7493.

Lunceford, J., and Davidian, M. 2004. Stratification and weighting via the propensity score in estimation of causal treatment effects: A comparative study.

Nie, X., and Wager, S. 2017. Quasi-oracle estimation of heterogeneous treatment effects.

Pearl, J. 2009. *Causality*. Cambridge university press.

Pearl, J. 2014. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier.

Pon-Features. Reference code for coupon purchase prediction. https://github.com/threecourse/kaggle-coupon-purchase-prediction.git. Accessed: 2020-02-06.

Ponpare. Ponpare: Coupons purchase prediction dataset. https://www.kaggle.com/c/coupon-purchase-prediction/data. Accessed: 2020-01-17.

Riquelme, C.; Tucker, G.; and Snoek, J. 2018. Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling. *arXiv preprint arXiv:1802.09127*.

Rubin, D. B. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology* 66(5):688.

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-Volume 70*, 3076–3085. JMLR. org.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Su, Y.; Wang, L.; Santacatterina, M.; and Joachims, T. 2019. Cab: Continuous adaptive blending for policy evaluation and learning. In *International Conference on Machine Learning*, 6005–6014.

Swaminathan, A., and Joachims, T. 2015a. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, 814–823.

Swaminathan, A., and Joachims, T. 2015b. The self-normalized estimator for counterfactual learning. In *advances in neural information processing systems*, 3231–3239.

Us census 1990 dataset on uci machine learning repository. https://archive.ics.uci.edu/ml/datasets/US+Census+Data+(1990). Accessed: 2019-11-15.

van der Maaten, L., and Hinton, G. 2008. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research* 2579–2605.

Wager, S., and Athey, S. 2018. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association* 113(523):1228–1242.

Wang, Y.-X.; Agarwal, A.; and Dudık, M. 2017. Optimal and adaptive off-policy evaluation in contextual bandits. In *International Conference on Machine Learning*, 3589–3597. PMLR.

Wooldridge, J. M. 2007. Inverse probability weighted estimation for general missing data problems. *Journal of econometrics* 141(2):1281–1301.

Xie, Y.; Chen, N.; and Shi, X. 2018. False discovery rate controlled heterogeneous treatment effect detection for online controlled experiments. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 876–885.