OPTIMIZATION APPROACHES FOR COUNTERFACTUAL RISK MINIMIZATION WITH CONTINUOUS ACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Counterfactual reasoning from logged data has become increasingly important for a large range of applications such as web advertising or healthcare. In this paper, we address the problem of counterfactual risk minimization for learning a stochastic policy with a continuous action space. Whereas previous works have mostly focused on deriving statistical estimators with importance sampling, we show that the optimization perspective is equally important for solving the resulting nonconvex optimization problems. Specifically, we demonstrate the benefits of proximal point algorithms and soft-clipping estimators which are more amenable to gradient-based optimization than classical hard clipping. We propose multiple synthetic, yet realistic, evaluation setups, and we release a new large-scale dataset based on web advertising data for this problem that is crucially missing public benchmarks.

1 INTODUCTION

Logged interaction data is widely available in a variety of machine learning problems, ranging from drug dosage prescription in healthcare Kallus & Zhou (2018), to evaluation of recommender systems Li et al. (2012) and setting reserve prices in real-time online auctions (Bottou et al., 2013). In these settings, one would like to leverage past data in order to find a good *policy* for selecting actions (e.g., drug doses or bids) from available features (or *contexts*), rather than relying on randomized trials or sequential exploration, which may be costly or even unethical for many applications.

We consider the setting of offline logged bandit feedback data, where some logs are available, consisting of features (contexts), along with actions selected by a given *logging policy* as well as the corresponding observed rewards. This is known as *bandit feedback*, since the reward is only observed for the action chosen by the logging policy, and we do not know what it would have been under a different policy. The problem of finding a good policy thus requires a form of *counterfactual* reasoning to estimate what the rewards would have been, had we used a different policy. When the logging policy is stochastic, one may obtain unbiased estimates of the expected reward under a new policy through importance sampling with the inverse propensity scoring method (IPS, Horvitz & Thompson, 1952). Even though the variance of the IPS estimator is potentially unbounded, one may nevertheless use such a framework to optimize new policies without the need for costly experiments, and improve upon the logging policy (Bottou et al., 2013; Dudík et al., 2011; Swaminathan & Joachims, 2015a;b).

In this paper, we focus on continuous actions settings, which have received little attention in the context of counterfactual policy optimization (see Kallus & Zhou, 2018; Demirer et al., 2019, for recent work on the topic). To this end, our first contribution is to introduce comprehensive evaluation benchmarks, based on both synthetic datasets, where counterfactual performance can be explicitly evaluated, and on a new large-scale dataset of real-world advertising data with more than 100 millions of samples. We discuss the issue of counterfactual policy evaluation on such real datasets, which is difficult, but yet is often possible with appropriate statistical analysis procedures.

Our second contribution is to underline the role of optimization algorithms for counterfactual risk minimization (CRM, Bottou et al., 2013; Swaminathan & Joachims, 2015b), while previous work has mostly studied the effectiveness of *estimation* methods. Specifically, because the resulting optimization problems are typically non-convex, the choice of algorithms for obtaining good policies is crucial, which is a key focus of our work. We find that the use of proximal point algorithms (Rockafellar,

1976; Fukushima & Mine, 1981; Paquette et al., 2018), which consists of approximately solving a sequence of regularized possibly-convex subproblems for minimizing a non-convex objective, tends to dominate simpler off-the-shelf optimization approaches.

In agreement with the previous observation that optimization is a key to counterfactual policy learning, our third contribution is to introduce differentiable estimators, which are more amenable to gradient-based optimization than existing ones that have been often used for discrete action spaces. More precisely, we propose an IPS estimator based on soft-clipping the importance weights, and a differentiable estimator based on a joint nonlinear embedding of actions and contexts. The benefits of these estimators are evaluated on the proposed benchmark on both real and synthetic data.

2 Related Work

Most of counterfactual estimators under contextual bandit feedback are based on importance sampling for correcting the distribution mismatch in rewards between the logging and target policies, the simplest method being inverse propensity scoring Horvitz & Thompson (1952). While unbiased, such an estimator can have high variance as soon as the target policy deviates significantly from the logging policy, which led to new biased estimators with reduced variance, through clipping importance weights Bottou et al. (2013); Wang et al. (2017), variance regularization (Swaminathan & Joachims, 2015a), or by leveraging reward estimators through doubly robust methods (Dudík et al., 2011; Robins & Rotnitzky, 1995). In order to tackle an overfitting phenomenon that arises in counterfactual policy optimization, termed "propensity overfitting", Swaminathan & Joachims (2015b) also consider self-normalized estimators (Owen, 2013). Such estimation techniques also appear in the context of sequential learning in contextual bandits, where the goal of the agent is to find a good policy in a sequential manner while minimizing regret (e.g., Langford & Zhang, 2008; Agarwal et al.), as well as for off-policy evaluation in reinforcement learning (e.g., Jiang & Li, 2016). In contrast, the setting considered in our work is not sequential.

While most approaches for counterfactual policy optimization tend to focus on discrete actions, few works have tackled the continuous action case, again with a focus on estimation rather than optimization. In particular, propensity scores for continuous actions were considered first by Hirano & Imbens (2004). More recently, evaluation and optimization of continuous action policies was studied in a non-parametric context by Kallus & Zhou (2018) using kernel smoothing, and by Demirer et al. (2019) in a parametric setting. These methods consider deterministic policies while our focus is on stochastic ones, but we note that Foster & Syrgkanis (2019) provide an interpretation of the kernel smoothing approach of Kallus & Zhou (2018) as learning stochastic policies with a noise level depending on the kernel bandwidth—an analogy which we leverage in our work.

Optimization methods for optimizing stochastic policies have been studied mainly in the context of reinforcement learning through the policy gradient theorem (Williams, 1992; Sutton et al., 2000; Ahmed et al., 2019). Such methods typically need to observe samples from the new policy at each optimization step, which is not allowed in our off-policy setting. Other methods leverage a form of off-policy estimates during optimization for improving the policy at each step (e.g., Kakade & Langford, 2002; Schulman et al., 2017), but this still requires fresh samples at each step, while we consider objective functions that globally lead to a good policy from a fixed dataset of collected data.

3 Counterfactual Risk Minimization

In this section, we review the CRM setup of Swaminathan & Joachims (2015b), as well as classical estimators.

3.1 Background

For a stochastic policy $\pi : \mathcal{X} \to \Delta(\mathcal{A})$ over a set of actions $\mathcal{A}$, a contextual bandit environment generates i.i.d. context features $x \sim \mathcal{D}_X$ in $\mathcal{X}$, actions $a \sim \pi(\cdot|x)$ and feedbacks/losses $y \sim D_Y(\cdot|x, a)$. We denote the resulting distribution over triples (x, a, y) as $\mathcal{D}_\pi$. We consider a logged dataset $(x_i, a_i, y_i, \pi_{0,i}), i = 1, \dots, n$, where we assume $(x_i, a_i, y_i) \sim \mathcal{D}_{\pi_0}$ i.i.d. for a given stochastic logging policy π_0, and the propensities are denoted by $\pi_{0,i} := \pi_0(a_i|x_i)$. The expected loss or risk

of a policy π is then given by

$$R(\pi) = \mathbb{E}_{(x,a,y)\sim\mathcal{D}_\pi}\left[y\right]. \tag{1}$$

For the logged bandit, the task is to determine a policy π^* in a set of *stochastic* policies Π that minimizes this risk. We note that in some cases it may be desirable to enforce a minimum variance for all policies considered; for instance, one may want to continue exploring in order to perform future offline experiments in settings where the newly obtained policy is meant to be deployed. In our setting, the expectation in equation 1 cannot be computed or estimated directly for any π, as the available interaction data comes from a different distribution π_0. Multiple empirical estimators of the risk hereafter allow to derive an empirical optimal policy that is found by solving:

$$\hat{\pi} \in \arg\min_{\pi\in\Pi} \hat{R}(\pi) + \Omega(\pi), \tag{2}$$

where the objective consists of an empirical estimate of the risk and of possible data-dependent regularizers on the policy, denoted by Ω. When using counterfactual estimators for $\hat{R}$, this method has been called (regularized) *counterfactual risk minimization* (Swaminathan & Joachims, 2015a).

3.2 Counterfactual Estimators

The counterfactual approach tackles the distribution mismatch between the logging policy $\pi_0(\cdot|x)$ and a policy π in Π via importance sampling. This first method called inverse propensity scoring (IPS, Horvitz & Thompson, 1952) relies on correcting the distribution mismatch using the relation

$$R(\pi) = \mathbb{E}_{(x,a,y)\sim\mathcal{D}_{\pi_0}}\left[y\frac{\pi(a|x)}{\pi_0(a|x)}\right], \tag{3}$$

assuming π_0 has non-zero mass on the support of π, which allows us to derive an unbiased empirical estimate

$$\hat{R}_{\text{IPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} y_i \frac{\pi(a_i|x_i)}{\pi_{0,i}}. \tag{4}$$

However, the empirical estimator $\hat{R}_{\text{IPS}}(\pi)$ in Eq. equation 4 has large variance and may overfit negative feedback values y_i for samples that are unlikely under π_0, resulting in higher variances. Clipping the importance sampling weights in Eq. equation 5 as in (Bottou et al., 2013) mitigates this problem, leading to a new estimator:

$$\hat{R}^M_{\text{IPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} y_i \min\left\{\frac{\pi(a_i|x_i)}{\pi_{0,i}}, M\right\}. \tag{5}$$

Smaller values of M reduce the variance of $\hat{R}^M(\pi)$ but induce a larger bias. Swaminathan & Joachims (2015a) also propose adding an empirical variance penalty term controlled by a factor λ to the empirical risk $\hat{R}^M(\pi)$, leading to a regularized objective with hyperparameters M and λ for clipping and variance penalization, respectively.

Swaminathan & Joachims (2015b) also introduce a regularization mechanism for tackling the so-called *propensity overfitting* issue, occuring with rich policy classes, where the method would focus only on maximizing (resp. minimizing) the sum of ratios $\pi(a_i|x_i)/\pi_{0,i}$ for negative (resp. positive) losses. This effect is corrected through the following *self-normalized* (SN) estimator (see also Owen, 2013), which is equivariant to additive shifts in loss values:

$$\hat{R}_{\text{SN}}(\pi) = \frac{\sum_{i=1}^n y_i w_i^\pi}{\sum_{i=1}^n w_i^\pi}, \quad \text{with } w_i^\pi = \frac{\pi(a_i|x_i)}{\pi_{0,i}}. \tag{6}$$

Another approach is to directly fit the loss values in observed data with a direct estimator $\hat{\eta}(x,a)$, for instance by using ridge regression to fit $y_i \approx \hat{\eta}(x_i,a_i)$, and to then use the deterministic greedy policy $\hat{\pi}(x) = \arg\min_a \hat{\eta}(x,a)$. This approach, termed direct method (DM), has the benefit of avoiding the high-variance problems of IPS-based methods, but may suffer from large bias since it focuses on estimating losses mainly near actions that appear in the logged data. Nevertheless, such loss estimators can be effective when few samples are available, and may be combined with IPS estimators in the so-called doubly robust estimator (DR, see, e.g. Dudík et al., 2011). This approach consists of

correcting the bias of the DM estimator by applying IPS to the residuals $y_i - \hat{\eta}(x_i, a_i)$, thus using $\hat{\eta}$ as a control variate to decrease the variance of IPS. When actions are discrete, the estimator takes the following form:

$$\hat{R}_{\text{DR}}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\sum_{a\in\mathcal{A}}\pi(a|x_i)\hat{\eta}(x_i,a) + \frac{1}{n}\sum_{i=1}^{n}\frac{\pi(a_i|x_i)}{\pi_{0,i}}\left(y_i - \hat{\eta}(x_i,a_i)\right) \tag{7}$$

For continuous actions, we follow Foster & Syrgkanis (2019, Section 8.2) and approximate $\mathbb{E}_{a\sim\pi(\cdot|x_i)}[\hat{\eta}(x_i,a)]$ by $\pi(a_i|x_i)\hat{\eta}(x_i,a_i)$ for the first DM term.

4 Optimization-Driven Approaches for CRM

We now present optimization approaches for tackling the CRM problem, as well as new differentiable estimators.

4.1 Policy Parameterization and Modeling

For counterfactual estimators we consider a set of parameterized stochastic policies $\Pi = \{\pi_\theta, \theta \in \Theta\}$. Our experiments consider policies with parameters $\theta = (\theta_\mu, \sigma)$ where σ is the standard deviation of a chosen distribution, and the mean takes the form $\mu = f(\theta, x)$ in $\mathbb{R}$ with different models f. In particular, we consider the following choices: (i) a constant, context-independent parameterization $\mu = \theta_\mu$; (ii) a stratified piecewise constant approach where we consider a partition of the feature space $\mathcal{X}$ (for instance, these may be obtained using cross-products of quantile buckets for each feature in low dimension) and learn a specific parameter for each partition; (iii) a linear model $\mu = \theta_\mu^\top x$ on features or (iv) a non-linear one $\mu = \theta_\mu^\top \varphi(x)$, *e.g.*, based on a mapping $\varphi(x) = (x, \text{vec}(xx^\top))$, which contains all pairwise interactions between the entries of x. With the same conventions as Eq. (2), the learning problem boils down to:

$$\hat{\theta} \in \arg\min_{\theta\in\Theta} \hat{R}(\pi_\theta) + \Omega(\pi_\theta). \tag{8}$$

4.2 Proximal Point Algorithms

The objective function in Eq. (8) is often nonconvex for various reasons. Nonconvexity may come from the weight clipping strategy of IPS (4), from variance regularization as used by Swaminathan & Joachims (2015b), from the use of self-normalized estimators (6), from the data-dependent regularization used in the doubly-robust estimator (7), or simply from the policy parameterization. In general, the learning problem (2) is thus nonconvex and has been optimized with classical gradient descent methods (see, Swaminathan & Joachims, 2015a;b) such as L-BFGS (Liu & Nocedal, 1989), or by using the stochastic gradient descent approach (see, Swaminathan & Joachims, 2015a; Joachims et al., 2018).

Proximal point methods are classical approaches originally designed for convex optimization (Rockafellar, 1976), which were then found to be useful for the minimization of nonconvex functions (Fukushima & Mine, 1981; Paquette et al., 2018). In order to minimize a function $\mathcal{L}$, the main idea is to approximately solve a sequence of subproblems that are better conditioned and easier to solve than $\mathcal{L}$, such that the sequence of iterates converges towards a stationary point of the original problem. More precisely, the proximal point method consists of computing a sequence

$$\theta_k \approx \arg\min_{\theta}\left(\mathcal{L}(\theta) + \frac{\kappa}{2}\|\theta - \theta_{k-1}\|_2^2\right), \tag{9}$$

where $\mathcal{L}(\theta) = \hat{R}(\pi_\theta) + \Omega(\pi_\theta)$ and $\kappa > 0$ is a constant parameter. Therefore, instead of applying a gradient-based algorithm directly on $\mathcal{L}$, we will also consider the following proximal point strategy: we solve a few successive instances of (9), each time using θ_{k-1} as an initialization for obtaining θ_k, using a value $\kappa > 0$ for all steps except the last one, where we use $\kappa = 0$.

Note that the effect of the proximal point algorithm differs from the proximal policy optimization (PPO) strategy used in reinforcement learning (Schulman et al., 2017), even though both approaches are related. PPO consists of encouraging a new stochastic policy to be close to a previous one in

Kullback-Leibler distance. Whereas the term used in PPO modifies the objective function (and changes the set of stationary points), the proximal point algorithm optimizes and finds a stationary point of the original objective $\mathcal{L}$.

4.3 SOFT CLIPPING FOR IPS WEIGHTS

The hard clipping estimator from Eq. (5) makes the objective function non-differentiable, and also causes terms in the objective with clipped weights to have zero gradient. In other words, a trivial stationary point of the objective function is that of a stochastic policy that differs enough from the logging policy such that all weights in (4) are clipped. The main purpose of this section is to introduce an estimator, which will be helpful to escape such bad stationary points. To that effect, we propose a differentiable logarithmic soft-clipping strategy. Given a threshold parameter $M \geq 0$ and an importance weight $w_i = \pi(a_i|x_i)/\pi_{0,i}$, we consider the soft-clipped weights.

$$\zeta(w_i, M) = \begin{cases} w_i & \text{if } w_i \leq M \\ \alpha(M) \log\left(w_i + \alpha(M) - M\right) & \text{otherwise,} \end{cases} \tag{10}$$

where $\alpha(M)$ is such that $\alpha(M)\log(\alpha(M)) = M$, which yields a continuous and differentiable operator.

Then, the IPS estimator with soft clipping becomes

$$\hat{R}^M_{\text{SIPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} y_i \zeta\left(\frac{\pi(a_i|x_i)}{\pi_{0,i}}, M\right). \tag{11}$$

In the supplementary material, we prove that the variance-regularized version of this estimator enjoys a similar generalization bound to that of Swaminathan & Joachims (2015a) for the hard-clipped version, and hence provides a good optimization objective for minimizing the expected risk.

Proposition 4.1 (Generalization bound for $\hat{R}^M_{\text{SIPS}}$)**.** Assume losses y_i in $[-1, 0]$ and importance weights bounded by W. With probability $1-\delta$, we have, for all π in Π,

$$R(\pi) \leq \hat{R}^M_{\text{SIPS}}(\pi) + O\left(\sqrt{\frac{\hat{V}(\pi)C_n(\Pi,\delta)}{n}} + \frac{SC_n(\Pi,\delta)}{n}\right),$$

where $S = \zeta(W, M) = O(\log W)$, $\hat{V}$ denotes the empirical variance of the loss estimates, and $C_n(\Pi, \delta)$ is a measure of complexity of the policy class (see supplementary material).

4.4 LEVERAGING LOSS PREDICTORS

Recall that direct methods learn an estimate $\hat{\eta}(x, a)$ of the feedback y given the context x and the action a, which may lead to the deterministic policy $\hat{\pi}(x) = \arg\min_{a\in\mathcal{A}} \hat{\eta}(x, a)$. Learning a predictor of the form of $\eta(x, a) = w^\top \phi(x, a)$ where $\phi(x, a)$ is a feature map allows to capture rich joint information on $\mathcal{X} \times \mathcal{A}$. For continuous actions, we may use a kernel smoothing approach to construct such a feature map, by taking m buckets $\tilde{\mathcal{A}} = \{a_1 \dots a_m\}$ of the action space $\mathcal{A}$ (such as quantiles of the observed actions), and setting $\phi(x, a) = \left(\frac{K(a-a_i)}{\sum_{j=1}^m K(a-a_j)} x\right)_{i=1\dots m} \in \mathbb{R}^{md}$, where $d = \dim(\mathcal{X})$ and K is a kernel function such as a Gaussian kernel. Then, we begin by fitting such a linear function $\hat{\eta}$ using ridge regression on the data $(x_i, a_i, y_i)_{i=1\dots n}$, and consider the following deterministic policy:

$$\hat{\pi}_{\text{DM}}(x) = \arg\min_{a\in\tilde{\mathcal{A}}} \hat{w}^T \phi(x, a). \tag{12}$$

Stochastic direct method. While a deterministic policy may be sufficient for exploitation, stochastic policies may be needed in some situations, for instance when one wants to still encourage some exploration in a future round of data logs, perhaps for non-stationarity issues. Then, we consider a stochastic version of the direct method by adding some Gaussian noise with variance σ^2:

$$\hat{\pi}_{\text{SDM}}(\cdot|x) = \mathcal{N}(\hat{\pi}_{\text{DM}}(x), \sigma^2), \tag{13}$$

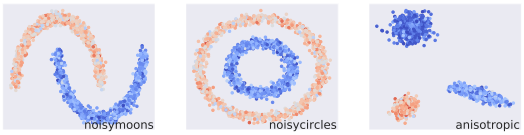

Figure 1: **Contexts (point positions in $\mathbb{R}^2$), and potentials represented by a color map for the synthetic datasets noisymoons, noisycircles, and anisotropic.** Learned policies should vary with the context to adapt to the underlying potentials.

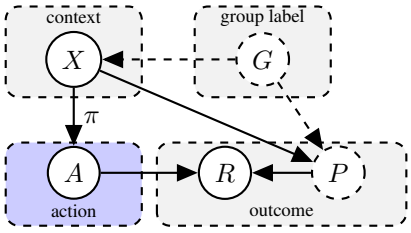

Figure 2: **Causal Graph of the synthetic setting**. A denotes the intervention action, X the feature context, G the unobserved group label, R the reward and P the unobserved potential.

Counterfactual loss predictor (CLP). While the policy obtained for the direct method may suffer from large bias since these estimators do not account for the mismatch between the obtained policy and π_0, the joint parameterization $\phi(x, a)$ of contexts and actions may provide additional expressivity compared to the context-dependent models $\phi(x)$ introduced in Section 4.1. Thus, we introduce a different parameterization of stochastic policies that leverages the same action-dependent feature map $\phi(x, a)$ as in the loss predictor, for estimating the policy mean using weighted combinations of predictions at the anchor points $a_1, \ldots, a_m$. This yields a differentiable model which may be used in the context of counterfactual optimization. Specifically, we may consider policies $\hat{\pi}_{\text{CLP}}(\cdot|x) = \mathcal{N}(\mu_\gamma(x), \sigma^2)$ with

$$\mu_\gamma(x) = \sum_{i=1}^{m} a_i \frac{\exp(\gamma \theta_\mu^\top \phi(x, a_i))}{\sum_{j=1}^{m} \exp(\gamma \theta_\mu^\top \phi(x, a_j))}.$$

μ_γ is a softmax function on the anchor points $(a_i)_{i=1\ldots m}$ that allows gradient-based optimization, γ is a hyperparameter (see supplemental material for how to choose it) and σ is the standard deviation parameter which is jointly optimized with the rest. We found that initializing θ_μ to the parameter of a reward predictor (or negative of a loss predictor) often yields improved policies compared to random initialization.

5 Experimental Setup and Evaluation

This section describes our evaluation setup. First, we propose synthetic problems that offer insights about the different policy learning methods we consider, while allowing for precise evaluation for any test policy. Then we describe a new, open dataset of very large scale that proposes a challenging problem of contextual off-policy optimization for continuous actions. We also discuss the evaluation metrics we use for assessing performance on the real-world open dataset, which is more challenging than the synthetic scenarios due to the partial information setting.

5.1 Synthetic Setting

We propose to study a simple generative setting, where we observe contexts x in $\mathcal{X}$ and rewards r in $\mathbb{R}$ given an action a in $\mathbb{R}$. The reward is generated as follows: an unobserved random group index g in $\mathcal{G}$ is drawn, which influences the drawing of a context x and of an unobserved potential p in $\mathbb{R}$, according to a joint conditional distribution $P_{X,P|G}$. The observed reward r is then a function of the context x, action a, and potential p. The causal graph corresponding to this process is given in Figure 2. Then, we generate three synthetic datasets called "noisymoons, noisycircles, and anisotropic", which are illustrated in Figure 1, with two-dimensional contexts $\mathcal{X} = \mathbb{R}^2$ and 2 or 3 groups.

Then, learning consists of finding a policy $\pi(a|x)$ that maximizes the reward by adapting to the unobserved potential. For our experiments, potentials are normally distributed conditionally on the group index, $p|g \sim \mathcal{N}(\mu_g, \sigma^2)$.

5.2 A NEW REAL-WORLD, OPEN DATASET

The open dataset comes from an online platform[1] that ran an experiment involving a randomized, continuous policy for online bidding. Each instance represents an opportunity for which a context x in $\mathbb{R}^d$ is observed, an action $a \geq 0$ is chosen according to a policy $\pi_0(a|x)$ that is logged. This logging policy was chosen to be a lognormal distribution as in (Bottou et al., 2013). The reward r in $\mathbb{R}$ is observed accordingly and can be assumed to be a noisy function of the context and action. It is notably sparse with 7% non-zero rewards. Statistics about the dataset are presented in Table 2.

A first important characteristic is the high variance of the reward signal, which is fortunately compensated by the large sample size N. Indeed, using for instance a central limit theorem argument allows to estimate the expected reward with an error $< 1.8\%$. Even though using more than $100M$ points for our problem may seem abusively large at first sight, it is in fact necessary to accurately perform the offline evaluation of the obtained policies (see Section 5.3).

Another characteristic of the dataset is the low dimensionality of contexts. For technical reasons, only 3 variables are available to the data collection process at the time when the action is drawn from the policy. Yet, the context was found to contain rich enough information to benefit from complex contextual policies and improve upon the logging policy.

To the best of our knowledge, this is the first dataset coming from a real-world randomized experiment on a machine learning system with continuous actions, for which action propensities are available. Available datasets to date are either synthetic or propose only discrete actions.

5.3 OFFLINE EVALUATION ON THE OPEN DATASET

For the synthetic datasets of Section 5.1, we can easily evaluate any policy since the reward function can be computed explicitly for any action. However, when we only have access to offline data with bandit feedback, as in the Open dataset, one cannot directly estimate rewards for a new policy, and we need to find alternative metrics for evaluation, based themselves on off-policy evaluation techniques. One may consider an IPS estimate of a policy on held-out data; however, such an estimator may be overly sensitive to the distribution of rewards in the test set, which are often zero and otherwise spread out in the Open dataset. Additionally, clipping the estimator would require tuning the clipping parameter which would be undesirable in a test metric. Therefore, we chose instead the self-normalized estimator SNIPS (Swaminathan & Joachims, 2015b), see Eq. (6), on the test set to estimate the reward on held-out logged bandit data. The SNIPS estimator is indeed more robust to the reward distribution thanks to its equivariance property to additive shifts and does not require hyperparameter tuning.

Figure 3 shows comparisons of IPS and SNIPS against an on-policy estimate of the reward for policies obtained from our experiments on the synthetic datasets, where policies can be directly evaluated online. We measure the regression slope and R^2 score to assess the quality of the estimation, and find that the SNIPS estimator is indeed more robust and gives a better fit to the on-policy estimate.

5.4 IMPORTANCE SAMPLING DIAGNOSTICS

Owen (2013) introduces diagnostics for assessing the quality of importance sampling estimates. We leverage such diagnostics to reject offline evaluations on test data for the Open dataset when they are considered not accurate enough. The use of such diagnostics was found to be crucial to allow accurate comparisons. In particular, when evaluating with SNIPS, we may consider an "effective sample size" quantity given in terms of the importance weights $w_i = \pi(a_i|x_i)/\pi_0(a_i|x_i)$ by $n_e = (\sum_{i=1}^n w_i)^2 / \sum_{i=1}^n w_i^2$. When this quantity is much smaller than the sample size n, this indicates that only few of the examples contribute to the estimate, so that the obtained value is likely a poor estimate.

[1]Name of the organization that donated the data as well as more details on the application will be released after the review process; dataset and experimental code will be available to reviewers.

6 EMPIRICAL EVALUATION

In this section, we present the empirical evaluation of the counterfactual estimators introduced above.

Evaluation protocol. For synthetic datasets, we generate training, validation, and test sets of size 10 000 each. For the Open dataset, we consider a 50%-25%-25% training-validation-test sets. We then run each method with 5 different random intializations such that the initial policy is close to the logging policy. Some of our results rely on selecting hyperparameters on the validation set, and we use a counterfactual SNIPS estimator based on the logged bandit data to evaluate the obtained policies. For estimating the final test performance and confidence intervals on synthetic datasets, we use an online estimate by leveraging the known reward function and taking a Monte Carlo average with 100 action samples per context, while for the Open dataset we use a 100-fold bootstrap procedure with the SNIPS estimator. We compute confidence intervals to discard estimators which are poorly evaluated on the Open dataset, or in order to have measures of significance when comparing pairs of estimators on synthetic datasets. In our experiments, we consider two forms of parameterizations: (i) a lognormal distribution with $\theta = (\theta_\mu, \sigma), \pi_{(\mu,\sigma)} = \log \mathcal{N}(m, s)$ with $s = \sqrt{\log\left(\sigma^2/\mu^2 + 1\right)}; m = \log(\mu) - s^2/2$, so that $\mathbb{E}_{a\sim\pi_{(\mu,\sigma)}}[a] = \mu$ and $\text{Var}_{a\sim\pi_{(\mu,\sigma)}}[a] = \sigma^2$; (ii) a normal distribution $\pi_{(\mu,\sigma)} = \mathcal{N}(\mu, \sigma)$. We add a positivity constraint for σ and add an entropy regularization term to the objective in order to encourage exploratory policies and avoid degenerate solutions.

Models. We compare the estimators IPS, IPS with classic clipping, IPS with soft clipping, DR, SNIPS, CLP. These estimators may use various parametrizations for the mean μ of our policies, as discussed in Section 4. This includes a unique parameter to learn, a stratification approach, a linear model $\phi(x) = x$, a quadratic feature map $\phi(x) = \text{vec}(xx^\top)$ (poly2) or a concatenation $\phi(x) = \text{vec}(xx^\top, x)$ (lin-poly2), or the rich joint embedding $\phi(x, a)$ of CLP.

Benefits of the proximal point algorithm. Figure 4 compares the policies obtained by optimizing with a standard L-BFGS gradient method and its proximal point variant described in Section 4, which is better suited for non-convex optimization problems. Each point compares the test metric for fixed choices of estimator and context model, as well as initialization seed and policy distribution, while optimizing the remaining hyperparameters on the validation set. We find that in most cases the proximal method improves the performance over the simpler gradient method, and reaches a near-maximal performance more often. This shows that it is beneficial for the task across a variety of configurations and more robust to various hyperparameter choices and initializations.

Benefits of the soft clipping estimator. Figure 5 shows a comparison of the soft-clipping IPS estimator and the corresponding hard-clipping version. Points correspond to different choices of the clipping parameter M, context models, intialization seeds and policy distributions, with the rest of the hyper-parameters optimized on the validation set. We can see that for most configurations, the soft-clipping estimator performs better than its hard-clipping variant, which may be attributed to the better optimization properties.

Role of contextual models. Table 1 shows a comparison of contextual models for stochastic policies with the IPS estimator. We find our new counterfactual loss predictor (CLP) parametrization, which leverages a form of action-dependent loss predictor to compute the mean of a Gaussian stochastic policy, to provide competitive performance with many other parametric models. The stratified model also performs well, perhaps because it only needs to learn a simpler constant-mean policy within each pre-specified partition of the contexts, which may be easier to optimize.

Off-policy estimates for the Open dataset. Table 3 shows estimates of the test performance for all combinations of optimization algorithm and clipping method on the clipped IPS objective for the Open dataset. We only consider the resulting policies which pass the effective sample size criterion on the importance weights described in Section 5.4, in order to discard methods for which the resulting SNIPS estimates on the test data are likely invalid. Among those methods, we show the test estimates when hyperparameters are optimized on the validation set. We find that the combination of the proposed proximal point algorithm and soft clipping yields the best results, confirming their strength beyond our previous observations on the synthetic scenarios.

REFERENCES

Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning (ICML)*.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International Conference on Machine Learning (ICML)*, 2019.

Léon Bottou, Jonas Peters, Joaquin Quiñonero Candela, Denis X. Charles, D. Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *Journal of Machine Learning Research (JMLR)*, 14(1):3207–3260, January 2013.

Mert Demirer, Vasilis Syrgkanis, Greg Lewis, and Victor Chernozhukov. Semi-parametric efficient policy learning with continuous actions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. In *International Conference on Machine Learning (ICML)*, 2011.

Dylan J Foster and Vasilis Syrgkanis. Orthogonal statistical learning. *arXiv preprint arXiv:1901.09036*, 2019.

Masao Fukushima and Hisashi Mine. A generalized proximal point algorithm for certain non-convex minimization problems. *International Journal of Systems Science*, 12(8):989–1000, 1981.

Keisuke Hirano and Guido W Imbens. The propensity score with continuous treatments. *Applied Bayesian modeling and causal inference from incomplete-data perspectives*, 226164:73–84, 2004.

D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2016.

Thorsten Joachims, Adith Swaminathan, and Maarten de Rijke. Deep learning with logged bandit feedback. In *International Conference on Learning Representations (ICLR)*, 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2002.

Nathan Kallus and Angela Zhou. Policy evaluation and optimization with continuous treatments. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

Lihong Li, Wei Chu, John Langford, Taesup Moon, and Xuanhui Wang. An unbiased offline evaluation of contextual bandit algorithms with generalized linear models. In *Proceedings of the Workshop on On-line Trading of Exploration and Exploitation 2*, 2012.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Art B. Owen. *Monte Carlo theory, methods and examples*. 2013.

C Paquette, H Lin, D Drusvyatskiy, J Mairal, and Z Harchaoui. Catalyst for gradient-based nonconvex optimization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

James M Robins and Andrea Rotnitzky. Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*, 90(429):122–129, 1995.

R Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NIPS)*, 2000.

Adith Swaminathan and Thorsten Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning (ICML)*, 2015a.

Adith Swaminathan and Thorsten Joachims. The self-normalized estimator for counterfactual learning. In *Advances in Neural Information Processing Systems (NIPS)*. 2015b.

Yu-Xiang Wang, Alekh Agarwal, and Miro Dudík. Optimal and adaptive off-policy evaluation in contextual bandits. In *International Conference on Machine Learning (ICML)*, 2017.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

A Appendix

Table 1: **Comparison of contextual parameterizations on synthetic datasets.** We show the test reward with one standard deviation estimated across contexts, after optimizing hyperparameters using SNIPS on the validation set.

	NoisyCircles	Anisotropic	NoisyMoons
CLP	**0.697** $\pm$ 0.07	**0.787** $\pm$ 0.06	0.726 $\pm$ 0.07
Unique	0.599 $\pm$ 0.00	0.598 $\pm$ 0.00	0.599 $\pm$ 0.00
Strat	0.670 $\pm$ 0.00	0.739 $\pm$ 0.02	**0.748** $\pm$ 0.00
Linear	0.611 $\pm$ 0.00	0.662 $\pm$ 0.00	0.725 $\pm$ 0.00
Poly2	0.611 $\pm$ 0.00	0.668 $\pm$ 0.00	0.668 $\pm$ 0.00
Lin-poly2	0.613 $\pm$ 0.00	0.681 $\pm$ 0.00	0.721 $\pm$ 0.00

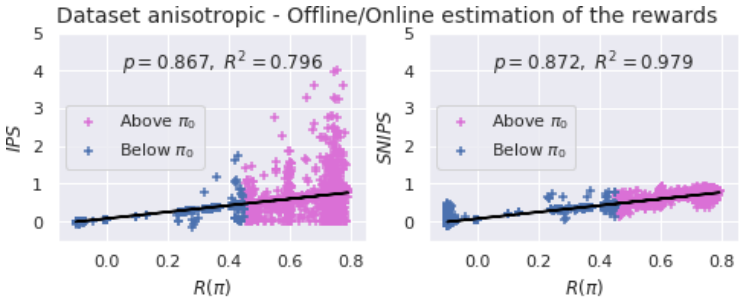

Figure 3: **Correlation between offline and online estimates on synthetic data**. Ideal fit would be $y = x$. Note how IPS (left) produces higher variance estimates compared to $SNIPS$ (right).

Table 2: **Summary statistics for Open dataset**.

N	d	$\mathbb{E}[R]$	$\mathbb{V}[R]$	$\mathbb{E}[A]$	$\mathbb{V}[A]$
119,971,451	3	11.37	9455.01	1.00	0.01

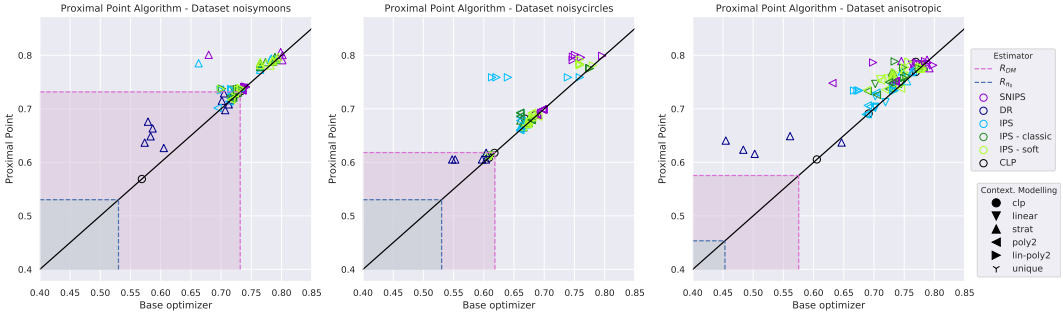

Figure 4: **Comparison of test rewards for proximal point vs the simpler gradient-based method on synthetic datasets.** For each point, all hyperparameters are optimized on the validation set except initialization seed and choice of policy distribution. The shaded areas represent test rewards below the logging policy (gray) or the best stochastic direct method (pink).

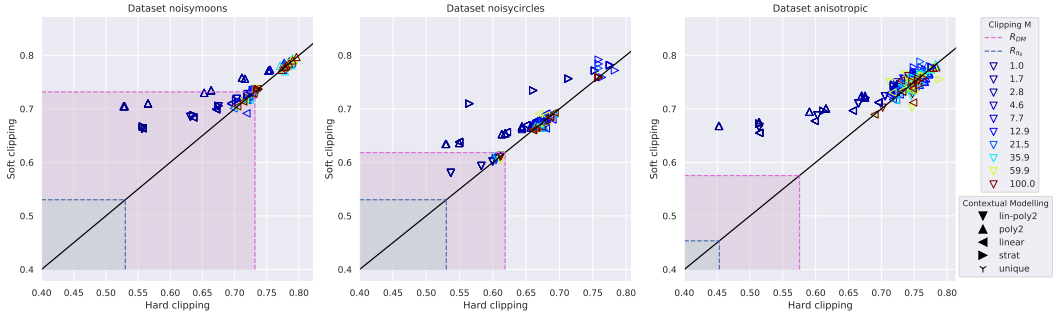

Figure 5: **Comparison of test rewards for soft- vs hard-clipping on synthetic datasets.** For each point, all hyperparameters are optimized on the validation set except initialization seed and choice of policy distribution. The shaded areas represent test rewards below the logging policy (gray) or the best stochastic direct method (pink).

Table 3: **SNIPS estimates of the test reward on the Open dataset**, when optimizing the clipped IPS objective, with hyperparameters optimized on the validation set. The logging policy baseline achieves a test reward estimate of 11.37.

	Hard-clipping	Soft-clipping
Non-Prox	11.42 ± 0.05	11.40 ± 0.05
Prox	11.55 ± 0.15	$\mathbf{11.57} \pm 0.14$