Task-specific experimental design for treatment effect estimation

Bethany Connolly [*] **Kim Moore** [*] **Tobias Schwedes** **Alexander Adam**
Gary Willis **Ilya Feige** **Christopher Frye** [1]

Abstract

Understanding causality should be a core requirement of any attempt to build real impact through AI. Due to the inherent unobservability of counterfactuals, large randomised trials (RCTs) are the standard for causal inference. But large experiments are generically expensive, and randomisation carries its own costs, e.g. when suboptimal decisions are trialed. Recent work has proposed more sample-efficient alternatives to RCTs, but these are not adaptable to the downstream application for which the causal effect is sought. In this work, we develop a task-specific approach to experimental design and derive sampling strategies customised to particular downstream applications. Across a range of important tasks, real-world datasets, and sample sizes, our method outperforms other benchmarks, e.g. requiring an order-of-magnitude less data to match RCT performance on targeted marketing tasks.

1. Introduction

Artificial intelligence makes its impact through the influence it has on actions taken in the real world. This influence is often indirect, e.g. when machine learning predictions inform human decision making. The field of causal inference studies the effect of actions on outcomes directly. Causal inference is thus central to the broader programme of engineering AI's positive effects on society.

Treatment effect estimation is a paradigm of causal inference that ranges from measuring the average treatment effect (ATE) of an intervention at the population level to the more granular prediction of individual treatment effects (ITE). Such treatment efficacy measures are useful across domains, from education (Olaya et al., 2020) and customer retention (Devriendt et al., 2021) to medicine (Qi & Tang, 2019) and marketing (Verbeke et al., 2023).

The fundamental challenge of causal inference is the inaccessibility of counterfactuals: upon observing an individual's response to one treatment, their response had they received a different treatment is not observable. Treatment effect estimation must therefore occur through observation of many individuals, and large randomised controlled trials are the hallmark of experimental design (Devriendt et al., 2018). RCTs minimise bias in cohort selection and treatment assignment to produce treatment groups representative of the population at large (Markozannes et al., 2021).

At smaller sample sizes, however, randomised treatment assignments often result in feature imbalances between treatment groups (Morgan & Rubin, 2012). This has spurred substantial research into covariate balancing between treatment groups (Greevy et al., 2004; Morgan & Rubin, 2012; Kallus, 2018). These techniques focus on treatment assignment given a fixed cohort of trial participants.

There is recent interest in optimising cohort selection as well (Qin et al., 2021; Jesson et al., 2021; Addanki et al., 2022). Such work aims to reduce the sample size required for performant treatment effect estimation. This is fundamentally an active learning (AL) (Ren et al., 2021) task wherein a population of unknown treatment response represents a pool of unlabelled data; a point is labelled when that individual's outcome is observed in a trial (Puha et al., 2020). In this context, AL can further be viewed as a multi-armed bandit problem (Deng et al., 2011).

In our view, AL techniques designed for supervised learning are ill-suited for causal modelling. Due to the fundamental challenge of causal inference, treatment effect models must be trained on loss functions that do not directly target the treatment effect. The T-learner (Gutierrez & Gérardy, 2017), for example, is the difference in two models trained separately to predict outcomes in each treatment group. Such proxy tasks are often far removed from the downstream application, e.g. ordering subjects by ITE. Loss-targeting AL methods (Roy & McCallum, 2001) are thus ill-fit for causal problems. Uncertainty-based AL (Beluch et al., 2018) is not well-suited either: in treatment effect estimation, one aims to learn the *difference* in two outcome probabilities, irrespective of their entropies.

[*]Equal contribution [1]Faculty, 160 Old Street, London, UK. Correspondence to: Christopher Frye <chris.f@faculty.ai>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

In this work, we remedy this problem by introducing a new approach to sample selection for treatment effect estimation. We take inspiration from Mindermann et al. (2022), focusing on data points that are "worth learning," though we do not employ the RHO loss presented in that work as it too is inherently supervised. Instead we derive a series of analytic sample-selection strategies tailored to specific applications.

1.1. Our contributions

In this paper, we are concerned with the optimisation of *experimental design*, i.e. the process of sampling individuals from a large population and assigning them treatments for a trial. The outcomes that result are used to model treatment effects and these predictions are then used to perform some downstream task. Our primary contributions include:

1. We introduce a novel approach to experimental design specific to the downstream task to be performed.
2. As an important example, we present an algorithm for experimental design for cases in which the area under the Qini curve (AUQ) (Radcliffe & Surry, 2011) is a good proxy for performance on the downstream task.
3. We validate our approach on several large real-world datasets and demonstrate state-of-the-art sample efficiency. In particular, on AUQ tasks our method requires about an order-of-magnitude less data to match the performance achieved by an RCT.

Our empirical analysis on large real-world datasets is itself significant, as tests on small semi-synthetic datasets are the norm in much of the related literature. We show that our method (i) scales well to large populations, (ii) leads unequivocally to statistically significant increases in performance, and (iii) works well on real-world data, whereas semi-synthetic data is often unrealistically simplistic.

2. Background

Let $\mathcal{X}$ denote a domain of features with density $p(X)$, and let $\mathcal{T} = \{0, 1\}$ represent a binary set of treatments that can be applied to individuals in the domain. Experimental design refers to the process of selecting individuals from $\mathcal{X}$, assigning treatments from $\mathcal{T}$, and observing outcomes that take values in $\mathcal{Y}$. In particular, we are interested in learning treatment effects. Provided no unobserved variables confound treatment and outcome, the individual treatment effect, or *uplift*, for $x \in \mathcal{X}$ is given by

$$u(x) = \mathbb{E}\left[Y|X = x, T = 1\right] - \mathbb{E}\left[Y|X = x, T = 0\right] \quad (1)$$

We use $\hat{u}(x)$ to denote a learnt estimate of the uplift. The *average treatment effect* can be written as:

$$\text{ATE}[u] \;= \int_{\mathcal{X}} p(x)\, u(x)\, dx \quad (2)$$

2.1. Metrics

A variety of metrics exist to judge a learnt treatment effect, each being the primary measure of performance in a specific context. In practice, one estimates these metrics using an *RCT test set*, in which individuals are sampled from $p(X)$ and their treatments T are uniformly randomised.

Mean squared error

First consider the *mean squared error* (MSE) of the model:

$$\text{MSE}[\hat{u}] \;= \int_{\mathcal{X}} p(x)\, (\hat{u}(x) - u(x))^2\, dx \quad (3)$$

In this work, the MSE principally serves as a simple example to demonstrate our method. The MSE is used primarily in the context of synthetic data where $u(x)$ is known exactly. The PEHE (precision in estimation of heterogeneous effects) is a common proxy for MSE in practice (Qin et al., 2021).

Squared ATE error

In a clinical context, the purpose of a trial is often to estimate the ATE of a particular intervention on a population. The *squared ATE error* $\left(\text{ATE}[\hat{u}] - \text{ATE}[u]\right)^2$ is therefore a useful target metric for this context. The true $\text{ATE}[u]$ can be estimated well as a simple difference in empirical means $\mathbb{E}\left[Y|T = 1\right] - \mathbb{E}\left[Y|T = 0\right]$ on an RCT test set.

Qini curve and AUQ

In the context of targeted marketing, the task is often to determine which subset of one's customer base should receive a finite budget of advertisements to maximise return on investment. A model $\hat{u}(x)$ can perform this task by selecting those customers with highest predicted uplift. The *Qini curve* $Q(f)$ measures performance at this task by tracing out the expected increase in per-capita outcomes as a function of the fraction f of customers accommodated by the budget (Radcliffe, 2007). The Qini curve is maximised as the model $\hat{u}(x)$ tends to the ground truth $u(x)$.

To write down an estimator of the Qini curve on a finite sample $\{x_1, \ldots, x_n\}$ from the underlying population, one must first order this sample according to descending predicted uplift: let $\{\hat{x}_1, \ldots \hat{x}_n\}$ denote the ordering in which $\hat{u}(\hat{x}_i) \geq \hat{u}(\hat{x}_{i+1})$ for all i. Then

$$Q(f) \approx \frac{1}{n} \sum_{i=1}^{f \cdot n} u(\hat{x}_i) \quad (4)$$

Note this estimator references the true uplift distribution; Algorithm 2 in the appendix details the procedure to evaluate the Qini curve on an RCT test set, in which $u(x)$ is of course unobserved. The *area under the Qini curve* provides

a general metric on model performance across budgets:

$$\text{AUQ}\,[\hat{u}] \;=\; \int_0^1 df\, Q(f) - \frac{\text{ATE}[u]}{2} \tag{5}$$

We subtract off the area under the straight line between $(0,0)$ and $(1, \text{ATE}[u])$ as this is the Qini curve of a random baseline model (Gutierrez & Gérardy, 2017).

In App. C.1, we describe one additional metric known as ERUPT (expected response under prescribed treatments).

3. Theory

Here we develop an approach to experimental design that optimises performance on a downstream task, as quantified by a metric $\mathcal{R}$. This target metric generally differs from the loss function used to train any model.

First, we make a simplifying assumption, taking a discrete feature space $\mathcal{X} = \{x_1, x_2, \ldots, x_k\}$. This will not limit the applicability of our method: we present an approach to discretisation in Sec. 3.3 that allows us to apply our method on continuous data. Also, for concreteness, we assume binary outcomes $\mathcal{Y} = \{0, 1\}$; footnote 1 describes how to alter our results for continuous $\mathcal{Y}$.

Second, we aim to optimise $\mathbb{E}[\mathcal{R}[\hat{U}]]$, and we need to select a representative model $\hat{U}$ to insert into this expression. We opt to work with Eq. (6) below as it allows us to derive a sampling strategy by hand. This choice does not limit the applicability of our method either: any downstream model can be trained on the sample that results from our method, and in Sec. 4 we empirically demonstrate excellent performance for a variety of model types.

To motivate our choice of $\hat{U}$, note that we would have complete knowledge of the treatment effect if we had access to $p(Y = 1|X = x, T = t) =: \theta^*_{xt}$. Let us use θ_{xt} to denote any prediction of these values (and we'll suppress subscripts when referring to the full set $\{\theta_{xt}\}$). Then, given a prior $p_\alpha(\Theta = \theta)$ with hyperparameter dependence indicated by α, and given observations $\{(x_i, t_i, y_i)\}_{i=1}^n$, there exists a posterior predictive $p_\alpha(Y \,|\, X = x,\, T = t,\, \{(x_i, t_i, y_i)\}_{i=1}^n)$. Motivated by Eq. (1), we take the difference in the means of the posterior predictives for $T = 1$ and 0 as our representative uplift model, and we place a Beta distribution prior over θ_{xt}, with hyperparameters[1] α_{xt} and β_{xt}, as it is conjugate to the Bernoulli likelihood. This results in

$$\hat{U}(x) \;=\; \sum_{t=0}^{1} (-1)^{t+1}\, \frac{\alpha_{xt} \;+\; \sum_{i=1}^{n(x,t)} Y_{i(x,t)}}{\alpha_{xt} \;+\; \beta_{xt} \;+\; n(x,t)} \tag{6}$$

In this equation, $n(x,t)$ denotes the number of samples in $\{(x_i, t_i, y_i)\}_{i=1}^n$ with $x_i = x$ and $t_i = t$, and we have capitalised $\hat{U}(x)$ and $Y_{i(x,t)}$ to emphasise that the model is a random variable through its dependence on outcomes $Y_{i(x,t)} \sim p(Y|X = x, T = t)$. The target metric $\mathcal{R}[\hat{U}]$ is a random variable as well, and our method involves calculating its expectation value by hand:

$$\mathbb{E}\big[\mathcal{R}[\hat{U}]\big] \;=:\; f_\alpha^{\mathcal{R}}\big(\{n(x,t)\}\,;\,\theta^*\big) \tag{7}$$

The function $f_\alpha^{\mathcal{R}}\big(\{n(x,t)\}\,;\,\theta^*\big)$ is computed explicitly for a few key target metrics in the following section.

Now, θ^* is unknown in practice, and $\mathcal{R}[\hat{U}]$ is generically unobserved during the experiment as well. Nonetheless, the latent value of the target metric $\mathcal{R}[\hat{U}]$ will change at each step $n(x,t) \mapsto n(x,t) + 1$, as this amounts to drawing an outcome $Y_{n(x,t)+1} \sim p(Y|X = x, T = t)$ and using it to update the model $\hat{U}$. Each improvement in $\mathcal{R}[\hat{U}]$ can be viewed as a reward. The true reward distribution belongs to a family of distributions indexed by θ, each with expectation value $f_\alpha^{\mathcal{R}}\big(\{n(x,t)\}\,;\,\theta\big)$, and every observation tightens the posterior $p_\alpha\big(\Theta = \theta \,\big|\, \{(x_i, t_i, y_i)\}_{i=1}^n\big)$.

We thus have a $2k$-armed bandit wherein pulling the (x,t) arm accrues a reward and improves the estimation of future rewards. This system can be optimised through Thompson sampling (Thompson, 1933), wherein at each step one draws a value of θ from the posterior, then selects the next (x,t) that maximises the expected reward given θ.

We therefore propose an approach to experimental design in which one cycles through the following sequence:

1. Draw: $\theta \sim p_\alpha\big(\Theta = \theta \,\big|\, \{(x_i, t_i, y_i)\}_{i=1}^n\big)$
2. Select: $\underset{(x^*,t^*)}{\arg\max}\; f_\alpha^{\mathcal{R}}\left(\begin{Bmatrix} n(x^*,t^*) + 1 & \text{at } (x^*,t^*) \\ n(x,t) & \text{otherwise} \end{Bmatrix};\, \theta\right)$
3. Observe: $Y_{n(x^*,t^*)+1} \sim p(Y|X = x^*, T = t^*)$
4. Update: $p_\alpha\big(\Theta = \theta \,\big|\, \{(x_i, t_i, y_i)\}_{i=1}^{n+1}\big)$

Thompson sampling optimises $\mathbb{E}[\mathcal{R}[\hat{U}]]$ in two ways: it results in an asymptotically (i.e. as $n \to \infty$) optimal sampling policy, and it achieves minimal regret, i.e. rewards forgone due to suboptimality at finite n are minimised (Agrawal & Goyal, 2012; Kaufmann et al., 2012).

Our full method, including discretisation (see Sec. 3.3), is detailed in Algorithm 1 and summarised in Fig. 2. It can be run in a sequential/online fashion as described here, if observations can be made between selection steps. Otherwise, it should be run in batches by iterating between steps 1-2 and deferring steps 3-4 to be performed at longer intervals; see App. A and Fig. 16.

3.1. Calculations for example target metrics

Here we provide examples of $f_\alpha^{\mathcal{R}}\big(\{n(x,t)\}\,;\,\theta^*\big)$ for a few key target metrics. The bias and variance of our representa-

[1] To accommodate the case of continuous outcomes, one could substitute $\alpha \to \mu\,\nu$ and $\beta \to \nu\,(1-\mu)$ in Eq. (6) where μ and ν are the parameters of a normal-gamma prior.

tive model, Eq. (6), arise throughout these calculations, so we collect results for these quantities here. (See App. C for all derivations of the results in this section.)

$$\begin{aligned}\mathbb{E}[\hat{U}(x) - u(x)] &= \sum_{t=0}^{1} (-1)^{t+1} \frac{\alpha_{xt} - (\alpha_{xt} + \beta_{xt})\,\theta^*_{xt}}{\alpha_{xt} + \beta_{xt} + n(x,t)} \\ \mathbb{V}[\hat{U}(x)] &= \sum_{t=0}^{1} \frac{\theta^*_{xt}\,(1-\theta^*_{xt})\,n(x,t)}{\big(\alpha_{xt} + \beta_{xt} + n(x,t)\big)^2} \qquad (8)\end{aligned}$$

In these equations, we replaced $\mathbb{E}[Y|X=x, T=t]$ and $\mathbb{V}[Y|X=x, T=t]$ with θ^*_{xt} and $\theta^*_{xt}\,(1-\theta^*_{xt})$ respectively, as Y is Bernoulli distributed.

Mean squared error

Beginning with the simplest case, consider the MSE, which takes the following expected value on $\hat{U}$:

$$\begin{aligned}\mathbb{E}\big[\mathrm{MSE}[\hat{U}]\big] &= \sum_{x\in\mathcal{X}} p(x)\,\mathbb{E}\big[(\hat{U}(x) - u(x))^2\big] \qquad (9)\\ &= \sum_{x\in\mathcal{X}} p(x)\,\Big(\mathbb{E}\big[\hat{U}(x) - u(x)\big]^2 + \mathbb{V}\big[\hat{U}(x)\big]\Big)\\ &=: -f^{\mathrm{MSE}}_{\alpha}\big(\{n(x,t)\};\,\theta^*\big)\end{aligned}$$

where one can refer to Eq. (8) to write down the explicit dependence of $f^{\mathrm{MSE}}_{\alpha}(\{n(x,t)\};\,\theta^*)$ on its arguments. We refer to the process of using Eq. (9) to iterate through our 4-step method of Sec. 3 as *MSE-optimised sampling*.

Squared ATE error

Similar to the case of the MSE, the squared ATE error takes the following expected value on $\hat{U}$:

$$\begin{aligned}&f^{\mathrm{ATE}}_{\alpha}\big(\{n(x,t)\};\,\theta^*\big) := \qquad (10)\\ &-\Big(\sum_{x\in\mathcal{X}} p(x)\,\mathbb{E}\big[\hat{U}(x) - u(x)\big]\Big)^2 - \sum_{x\in\mathcal{X}} p(x)^2\,\mathbb{V}\big[\hat{U}(x)\big]\end{aligned}$$

The explicit dependence of $f^{\mathrm{ATE}}_{\alpha}(\{n(x,t)\};\,\theta^*)$ on its arguments follows again from Eq. (8). *ATE-optimised sampling* refers to the insertion of Eq. (10) into the method of Sec. 3.

Area under Qini curve

The AUQ is more complicated but in many contexts much more relevant (Radcliffe & Surry, 2011; Rzepakowski & Jaroszewicz, 2012; Gutierrez & Gérardy, 2017). In the large n limit, $\mathbb{E}[\mathrm{AUQ}[\hat{U}]]$ takes the following value:

$$\begin{aligned}&f^{\mathrm{AUQ}}_{\alpha}\big(\{n(x,t)\};\,\theta^*\big) := \qquad (11)\\ &\sum_{\substack{x,x'\in\mathcal{X}\\ x\neq x'}} p(x)\,p(x')\,u(x')\left[\frac{1}{2} + \frac{1}{2}\,\mathrm{erf}\left(\frac{u(x') - u(x)}{\sqrt{2\,\sigma^2(x,x')}}\right)\right]\end{aligned}$$

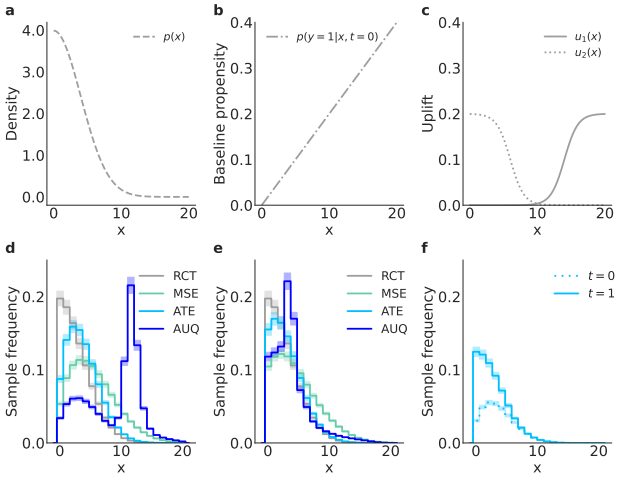

Figure 1: Synthetic density (a), baseline propensity (b), and uplift (c) distributions, along with the samples that result from our optimised methods; (d) and (e) correspond to u_1 and u_2, respectively, and (f) shows the ATE-optimised sample from (e) split by treatment and control.

where the dependence of $f^{\mathrm{AUQ}}_{\alpha}\big(\{n(x,t)\};\,\theta^*\big)$ on its arguments follows from $u(x) = \theta^*_{x1} - \theta^*_{x0}$ and the shorthand

$$\sigma^2(x,x') = \sum_{t=0}^{1} \frac{\theta^*_{xt}\,(1-\theta^*_{xt})}{n(x,t)} + (x \to x') \qquad (12)$$

We refer to the insertion of Eq. (11) into the method of Sec. 3 as *AUQ-optimised sampling*.

In App. C.6, we derive $f^{\mathcal{R}}_{\alpha}\big(\{n(x,t)\}\,;\,\theta^*\big)$ for the ERUPT metric as an additional example.

3.2. Intuition for task-specific sampling

Here we demonstrate our optimised samplers on synthetic data. (See App. B for details of this experiment.)

For this experiment, we take $\mathcal{X} = \{1, 2, \ldots, 20\}$ with a truncated Gaussian density $p(X)$ as shown in Fig. 1(a). We set the *baseline propensity*, i.e. $p(Y = 1 \,|\, X = x,\, T = 0)$, to a linear function of x as depicted in Fig. 1(b). We consider two uplift distributions, $u_1(x)$ and $u_2(x)$, mirror images of each other defined using sigmoids; see Fig. 1(c).

In this section, we aim to provide intuition for the different $f^{\mathcal{R}}_{\alpha}(\{n(x,t\};\theta^*)$'s derived in Sec. 3.1. To isolate the effect of this function, we cycle through the 4-step algorithm of Sec. 3 but with $p_{\alpha}\big(\Theta = \theta \,\big|\, \{(x_i, t_i, y_i)\}_{i=1}^{n}\big)$ set to a point mass at the correct value of θ^*. We set $\alpha_{xt} = \beta_{xt} = 0$ in $f^{\mathcal{R}}_{\alpha}(\{n(x,t\};\theta^*)$ for concreteness as well.

The resulting distributions are plotted in Fig. 1(d) for $u_1(x)$ and Fig. 1(e) for $u_2(x)$, where they are compared against an RCT sample. Fig. 1(f) shows the ATE-optimised sample from Fig. 1(e) split into its $T = 0$ and 1 components.

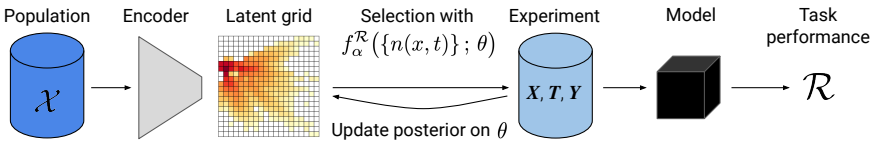

Figure 2: Schematic of our end-to-end method for task-specific experimental design for treatment effect estimation.

Our samplers generically select different distributions for treatment and control to optimise the target metric.

Note from Fig. 1 that the RCT sample is proportional to the density, whereas the MSE-optimised sample is also sensitive to the variance in outcomes (see Eq. 9) which pulls the distribution to the right. The ATE-optimised sample comes out roughly in between, as it has stronger (i.e. squared) dependence on the density (see Eq. 10). The AUQ-optimised sample is additionally sensitive to uplift differences $u(x)-u(x')$ (see Eq. 11) which leads to a strong peak coincident with the sharp gradient in $u(x)$.

3.3. Discretisation for general datatypes

Our method, as developed so far, assumes a discrete feature space. Here we describe our approach to discretisation to accommodate continuous and mixed-type data.

We discretise data using a variational autoencoder (VAE) (Kingma & Welling, 2013) trained on $\mathcal{X}$. In particular, we encode the dataset and place a rectangular grid on latent space to partition $\mathcal{X}$ into a finite set $\{x_1, x_2, \ldots, x_k\}$ as required. This is the final ingredient in our approach to experimental design; see Algorithm 1 and Fig. 2.

This approach to discretisation partitions the data in a smooth homogeneous space rather than the heterogeneous space of raw features. It also gives us control over the cardinality k, which influences the runtime of our algorithm. We thus find it useful to perform latent-space discretisation regardless of the datatypes in $\mathcal{X}$. App. C shows further that discretisation does not introduce bias into our method.

Uniform sampling in latent space

Having defined a grid on latent space, one simple but novel alternative to Algorithm 1 is to select grid cells and assign treatments both uniformly at random. We refer to this as *uniform sampling in latent space*. As it is task-agnostic by construction, one cannot expect this strategy to perform optimally across metrics. Indeed, Fig. 8 in the appendix shows poor performance for the ATE task. Despite this, we show in Sec. 4.2 that uniform sampling in latent space is remarkably effective for the AUQ task. App. D develops an importance-sampling formalism that could be used to determine when uniform sampling will perform well.

In App. A, we isolate the advantage of discretisation with a VAE by comparing uniform sampling in latent space to uniform sampling in *feature* space; see Fig. 9 in the appendix.

Algorithm 1 Task-specific experimental design

Input: Population $\mathcal{X}$; Target metric $\mathcal{R}$; Sample size n

VAE's latent representation of population $\mathcal{X}_{\text{latent}}$
Rectangular grid on latent space $\mathcal{B}$
Prior parameters α_{bt} and β_{bt} for each $b \in \mathcal{B}$ and $t \in \mathcal{T}$

Output: Sample of experimental results $\{(x_i, t_i, y_i)\}_{i=1}^n$

Initialise $n_{bt} = 0$ for $b \in \mathcal{B}$ and $t \in \mathcal{T}$
for $i = 1$ **to** n **do**
  Draw $\theta_{bt} \sim \text{Beta}(\alpha_{bt}, \beta_{bt})$ for each $b \in \mathcal{B}$ and $t \in \mathcal{T}$
  for each $b \in \mathcal{B}$ and $t \in \mathcal{T}$ **do**
    Set $n'_{b't'} = n_{b't'}$ for each $b' \in \mathcal{B}$ and $t' \in \mathcal{T}$
    Increment $n'_{bt} \leftarrow n'_{bt} + 1$
    Set $f_{bt} = f_\alpha^{\mathcal{R}}(\{n'_{b't'}\}; \theta)$ using $\mathcal{R}$-specific formula
  end for
  Select $(b_i, t_i) = \arg\max_{(b,t)} f_{bt}$
  Sample x_{latent} uniformly from $\mathcal{X}_{\text{latent}} \cap b_i$
  Select individual $x_i \in \mathcal{X}$ that corresponds to x_{latent}
  Observe $y_i \sim p(Y|X = x_i, T = t_i)$
  $n_{b_i t_i} \leftarrow n_{b_i t_i} + 1$
  $\alpha_{b_i t_i} \leftarrow \alpha_{b_i t_i} + y_i$
  $\beta_{b_i t_i} \leftarrow \beta_{b_i t_i} + (1 - y_i)$
end for
return $\{(x_i, t_i, y_i)\}_{i=1}^n$

4. Experiments

Here we present empirical analyses of our approach to experimental design. See App. B for full experimental details.

4.1. Validation on synthetic data

In Sec. 3.2 we used two synthetic datasets to show distributions resulting from our optimised samplers. Here we take the $u_1(x)$ case, use the samples from Fig. 1(d) to train uplift models according to Eq. (6), and measure model performance according to the metrics of Sec. 2.1. Fig. 3 displays the results, with one plot for each target metric and one curve for each optimised sampler. We also include ordinary RCT sampling as well as uniform sampling in latent space for comparison. Note that, across sample sizes, the

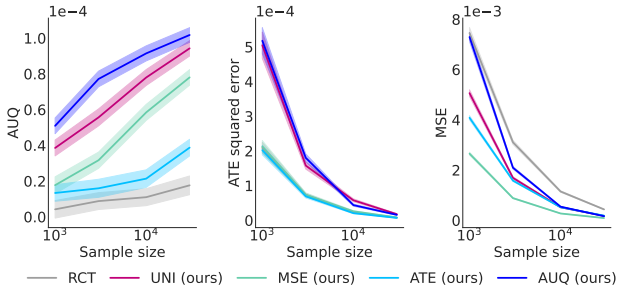

Figure 3: Performance of MSE-, ATE-, and AUQ-optimised samplers across target metrics on synthetic data, compared to RCT and uniform sampling on $\mathcal{X}$.

MSE / ATE / AUQ - optimised sampler outperforms the alternatives when judged by the test-set MSE / ATE / AUQ, respectively. This provides the first empirical validation of our approach to task-specific experimental design.

4.2. Performance on real-world data

Here we describe experiments carried out on real-world data, but first we address a subtlety in our setup. We aim to benchmark a new method of experimental design using data gathered from historical experiments that were performed as RCTs. We circumvent this apparent paradox as follows. Most of the datasets we use are very large in size; in particular, after discretisation there are many examples of most (x, t) combinations we would be interested in sampling in our experimental design. We thus treat the large pool of data as though treatments were yet unassigned, and whenever the dataset lacks an (x, t) that our algorithm selects, we select the algorithm's next preference instead.

The datasets we use for our experiments are described at length in App. B.1. In brief, we test our method on:

- STROKE: clinical trial evaluating aspirin's effect on stroke patients; our sub-selection procedure results in a dataset of size 9k (Sandercock et al., 2011).
- CRITEOVISIT & CRITEOCONVERSION: marketing trial evaluating effectiveness of email campaign on two different outcomes; we sub-select 7M rows of data (Diemert et al., 2018).
- RETAILHERO: marketing trial in which we engineered features from purchase history data for 200k individuals (see App. B.1 for references).

Next we describe performance results for two downstream applications: clinical trials and targeted marketing.

APPLICATION TO CLINICAL TRIALS

Here we take the squared ATE error as our target metric. While STROKE is our only example of a *true* clinical trial,

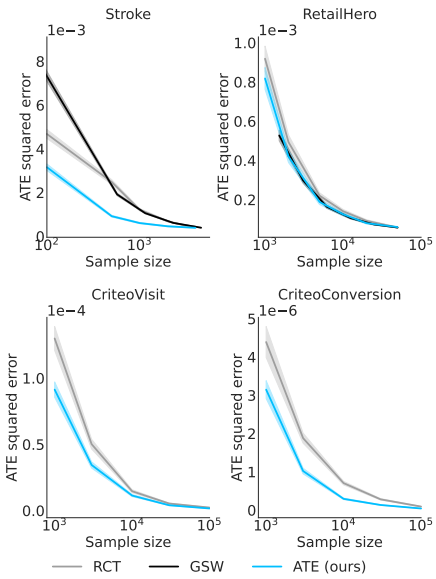

Figure 4: Performance of ATE-optimised sampler across datasets, benchmarked against RCT and Recursive GSW.

we include all our datasets in this study. In this experiment, we sample data according to different algorithms, then reweight the sample according to the population density to estimate the ATE.

In Fig. 4 we show the performance of our ATE-optimised sampler alongside RCT. On three of four datasets, our ATE-optimised sampler provides a meaningful advantage over RCT with respect to the target metric.

We also benchmark the Recursive Gram-Schmidt Walk (GSW) method of Addanki et al. (2022) though it does not outperform RCT on STROKE or RETAILHERO. Due to the quadratic scaling of the GSW algorithm with respect to population size (Harshaw et al., 2019), Recursive GSW is infeasible on both CRITEO datasets. Fig. 8 in the appendix further benchmarks the B-EMCMITE method of Puha et al. (2020). This is a loss-targeting method of supervised AL applied to causal modelling, which is only computationally feasible for very small populations. We find its ATE performance to be poor in our experiments.

Fig. 10 in the appendix benchmarks the time complexity of our method against Recursive GSW and B-EMCMITE as a function of population size. This is a very favourable result for our method, as the only step in Algorithm 1 that scales with population size is the uniform sampling of an occupant from the selected latent grid cell.

APPLICATION TO TARGETED MARKETING

Here we take the AUQ as our target metric and exclude STROKE from this study, as the AUQ requires a very large test set to estimate precisely. In this experiment, we sample data according to different algorithms, then train T-learners

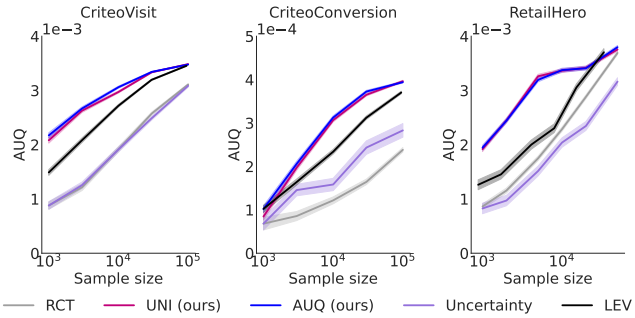

Figure 5: Performance of AUQ-optimised sampling and uniform sampling in latent space across datasets, benchmarked against RCT, Leverage Scoring, and uncertainty sampling.

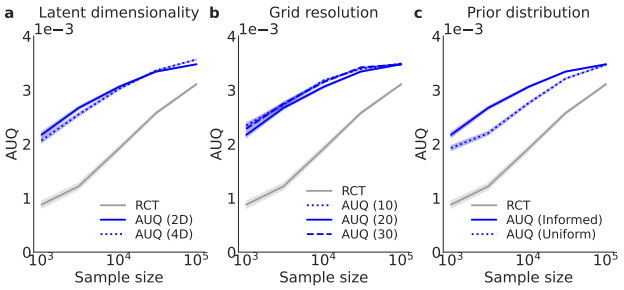

Figure 6: Hyperparameter sensitivity of AUQ-optimised sampling to latent dimension, grid resolution, and prior choice on CRITEOVISIT, with RCT as an anchor point.

(Gutierrez & Gérardy, 2017) on the samples to predict uplift, and finally measure the AUQ of each T-learner.

In Fig. 5 we show the performance of our AUQ-optimised sampler. Across all sample sizes and datasets, our method provides a substantial improvement over RCT, generally matching its performance with roughly an order-of-magnitude smaller sample.

Fig. 5 also benchmarks the Leverage Scoring method of Addanki et al. (2022), which performs between our method and RCT, as well as an uncertainty-based approach to supervised AL (Beluch et al., 2018) in which points are selected that have the highest entropy in predicted outcomes, according to the model-in-training. The latter is not tailored to causal modelling and performs commensurate with RCT.

Fig. 5 further shows that uniform sampling in latent space performs on par with AUQ-optimised sampling. This is quite remarkable, given the state-of-the-art nature of our AUQ-optimised sampler and the simplicity of our uniform sampler. We understand this as follows:

AUQ-optimised sampling targets segments of the population where uplift differences are large; this is a direct result of Eq. (11) that we demonstrated in Sec. 3.2. In addition, our sampler has the largest uncertainty on uplift in regions that have been under-sampled. Through our Thompson sampling procedure of Sec. 3, our sampler will attribute the largest swings in uplift – and thus the largest uplift differences – to the least-sampled regions of the data. This can result in a sampling strategy that is not too different from uniform. (See Fig. 11 in the appendix for an example).

App. A includes two further results that we mention here: (i) Whereas Fig. 5 displays the mean AUQ of the various samplers, Fig. 12 shows the distribution of AUQ across trials to give a sense of how often our method outperforms RCT in practice. (ii) Fig. 13 explores an additional use case, ERUPT-optimised sampling, where our method outperforms RCT, Leverage Scoring, and uniform sampling in latent space.

4.3. Sensitivity to hyperparameters

Here we evaluate the sensitivity of our method to the hyperparameters required to fully define it.

LATENT SPACE DIMENSIONALITY

Provided latent space has capacity to capture the factors of variation underlying the raw data, we would not expect strong dependence on the choice of latent dimensionality. Our experiments above were performed with latent dimension 2, and in Fig. 6(a) we test our AUQ-optimised sampler on CRITEOVISIT with latent dimension 4. The result is that performance hardly varies with this hyperparameter. (Though note that latent dimensionality would need to be tuned more carefully for higher-dimensional feature spaces.)

SAMPLING GRID RESOLUTION

Regarding the rectangular grid used to discretise latent space, one would expect a trade-off: a finer grid allows more precise targeting, while a coarser grid allows lower-variance θ_{xt} estimates (see Sec. 3). Our experiments above all used 20x20 grids, and in Fig. 6(b) we test our AUQ-optimised sampler on CRITEOVISIT with 10x10 and 30x30 grids, showing minimal dependence on this hyperparameter.

PRIOR DISTRIBUTION

The prior (in particular α_{xt} and β_{xt}) directly influences the formula used to select the training sample (see Sec. 3) so we should expect dependence on this choice.

For the ATE experiments above, we used $\alpha_{xt} = \beta_{xt} = 1$, placing uniform priors on the Bernoulli outcome probabilities. For the AUQ experiments, we defaulted to an informed prior with $\alpha_{xt} = 1 + p(x)\, n_t^+$ and $\beta_{xt} = 1 + p(x)\, n_t^-$. Here $n_t^\pm$ is a ballpark estimate of the successes/failures anticipated for treatment group t (but agnostic of x) in an experiment of the given size. Thus α_{xt} and β_{xt} reflect the anticipated base rates in the data. (These estimates need not be accurate; Fig. 14 in the appendix shows minimal effect

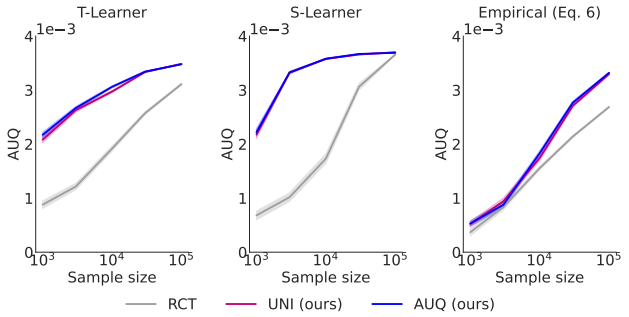

Figure 7: Variation of the downstream model trained on AUQ-optimised, uniform, and RCT samples from CRITEO-VISIT. Our methods outperform RCT across model choices.

when the n_t^+'s are doubled or halved.)

In Fig. 6(c) we show how AUQ-optimised sampling on CRITEOVISIT depends on the prior. The informed prior does provide a meaningful advantage over the uniform prior when it comes to AUQ performance, though even the uninformed prior provides a large improvement over RCT.

UPLIFT MODEL TYPE

Finally we demonstrate that our method works well for a variety of downstream uplift models. While we derived our algorithm using one specific model (Eq. 6) our AUQ experiments used a T-learner (Gutierrez & Gérardy, 2017) for the modelling task. The good performance of Fig. 5 thus provides initial validation of this claim.

In Fig. 7 we validate this further by showing the T-learner's performance trained on AUQ-optimised samples from CRITEOVISIT alongside that of an S-learner and of Eq. (6) itself (with $\alpha_{xt} = \beta_{xt} = 0$). While absolute performance varies across model types, we find that training a model on a sample selected with our method dramatically outperforms that same model trained on an RCT sample.

TARGET METRIC & BATCH SIZE

App. A includes two further results that we mention here for completeness. First, Fig. 15 shows the performance of our AUQ-, ATE-, and MSE-optimised samplers evaluated on both the AUQ and ATE-squared-error metrics on CRITEO-VISIT. This demonstrates that no sampler, not even RCT, performs well across target metrics as disparate as AUQ and ATE squared error. This result calls into question any attempt at task-agnostic experimental design and provides further justification for our general approach.

Second, Fig. 16 shows that our AUQ-optimised sampler is extremely insensitive to the batch size with which observed outcomes are used to update its posterior (cf. step 4 in Sec. 3 and the discussion below it). The ATE-optimised sampler is more sensitive but still demonstrates useful performance.

5. Related work

Here we compare our work to other studies of experimental design for treatment effect estimation.

Deng et al. (2011) frame experimental design as a multi-armed bandit problem as we do. They arrive at a formula similar to our $f_\alpha^{\text{MSE}}(\{n(x,t)\};\theta)$ with $\alpha_{xt} = \beta_{xt} = 0$, but leaving out dependence on the density $p(x)$. In comparison, we consider our approach to be more general as we consider a variety of target metrics, including the AUQ which we have not seen targeted in any other experimental design work. Deng et al. (2011) also assume the data is split into subpopulations but do not include any algorithm to do so – a hurdle to applying their method in practice.

Puha et al. (2020) devise an approach to experimental design called B-EMCMITE, which targets individuals who maximise the expected change in the uplift model's loss function, taking inspiration from supervised AL. This is downstream-task-agnostic, in contrast with our approach. We benchmark our ATE-optimised sampler against B-EMCMITE in Fig. 8, at least where computationally feasible, and find that our method strongly outperforms it. B-EMCMITE scales poorly as it requires model gradients across the entire population at every selection step; see Fig. 10.

Addanki et al. (2022) present two sampling algorithms, Recursive GSW and Leverage Scoring, intended for ATE and ITE estimation, respectively. They prove a notion of optimality under very restrictive assumptions – e.g. that outcomes, and thus the uplift itself, are linear functions of the features. In Sec. 4.2 we show that our optimised samplers outperform Recursive GSW and Leverage Scoring across tasks, datasets, and sample sizes – except for the ATE task on RETAILHERO (see Fig. 4) where results are even.

Our work relates to pool-based AL with expected-model-change querying (Ren et al., 2021). The defining distinction is that we do not query according to the uplift model's loss function; instead, our approach targets the metric that defines performance on the downstream task.

Our method of uniform sampling in latent space relates to core-set methods (Sener & Savarese, 2018) in AL. VAEs have been used elsewhere to obtain low-dimensional representations for general AL (Pourkamali-Anaraki & Wakin, 2019; Sinha et al., 2019), though we have not seen this applied to the problem of causal modelling outside our work.

6. Conclusion

We have introduced a novel method of experimental design for treatment effect learning, in which a task-specific formula governs cohort selection and treatment assignment. We derived this formula explicitly for four metrics – MSE, ATE, AUQ, ERUPT – which find relevance in distinct ap-

plications. Our method outperforms RCT and other benchmarks almost universally across the tasks, datasets, and sample sizes we studied. Most notably, on the AUQ metric prevalent in contexts in which the ITE is of primary interest, our method requires about an order-of-magnitude less data to match RCT performance.

In our view, the primary limitation of our method is that it requires by-hand calculations for each target metric, and future work could explore data-driven alternatives to this requirement. Follow-up work could also explore alternatives to Thompson sampling as the backbone of our method. More generally, we hope our work spurs further advances in the field, so that efficient causal inference can have a positive impact across a wide range of important applications.

References

Addanki, R., Arbour, D., Mai, T., Musco, C., and Rao, A. Sample constrained treatment effect estimation. *NeurIPS*, 2022.

Agrawal, S. and Goyal, N. Analysis of thompson sampling for the multi-armed bandit problem. *Conference on Learning Theory*, 2012.

Beluch, W., Genewein, T., Nürnberger, A., and Köhler, J. The power of ensembles for active learning in image classification. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Deng, K., Pineau, J., and Murphy, S. Active learning for personalizing treatment. *IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, 2011.

Devriendt, F., Moldovan, D., and Verbeke, W. A literature survey and experimental evaluation of the state-of-the-art in uplift modeling. *Big Data*, 2018.

Devriendt, F., Berrevoets, J., and Verbeke, W. Why you should stop predicting customer churn and start using uplift models. *Information Sciences*, 2021.

Diemert, E., Betlei, A., Renaudin, C., and Amini, M.-R. A large scale benchmark for uplift modeling. *KDD*, 2018.

Greevy, R., Lu, B., Silber, J., and Rosenbaum, P. Optimal multivariate matching before randomization. *Biostatistics*, 2004.

Gutierrez, P. and Gérardy, J.-Y. Causal inference and uplift modelling: A review of the literature. *International conference on predictive applications and APIs*, 2017.

Harshaw, C., Sävje, F., Spielman, D., and Zhang, P. Balancing covariates in randomized experiments with the gram–schmidt walk design. *arXiv*, 1911.03071, 2019.

Hitsch, G. and Misra, S. Heterogeneous treatment effects and optimal targeting policy evaluation. *SSRN*, 3111957, 2018.

Jesson, A., Tigas, P., van Amersfoort, J., Kirsch, A., Shalit, U., and Gal, Y. Causal-BALD: Deep bayesian active learning of outcomes to infer treatment-effects from observational data. *NeurIPS*, 2021.

Kallus, N. Optimal a priori balance in the design of controlled experiments. *Journal of the Royal Statistical Society: Series B*, 2018.

Kaufmann, E., Korda, N., and Munos, R. Thompson sampling: An asymptotically optimal finite-time analysis. *Algorithmic Learning Theory*, 2012.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. *ICLR*, 2015.

Kingma, D. and Welling, M. Auto-encoding variational bayes. *arXiv*, 1312.6114, 2013.

Künzel, S., Sekhon, J., Bickel, P., and Yu, B. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 2019.

Markozannes, G., Vourli, G., and Ntzani, E. A survey of methodologies on causal inference methods in meta-analyses of randomized controlled trials. *Systematic Reviews*, 2021.

Mindermann, S., Brauner, J., Razzak, M., Sharma, M., Kirsch, A., Xu, W., Höltgen, B., Gomez, A., Morisot, A., Farquhar, S., et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. *ICML*, 2022.

Morgan, K. and Rubin, D. Rerandomization to improve covariate balance in experiments. *The Annals of Statistics*, 2012.

Olaya, D., Vásquez, J., Maldonado, S., Miranda, J., and Verbeke, W. Uplift modeling for preventing student dropout in higher education. *Decision Support Systems*, 2020.

Pourkamali-Anaraki, F. and Wakin, M. The effectiveness of variational autoencoders for active learning. *arXiv*, 1911.07716, 2019.

Puha, Z., Kaptein, M., and Lemmens, A. Batch mode active learning for individual treatment effect estimation. In *International Conference on Data Mining Workshops*, 2020.

Qi, Y. and Tang, Q. Predicting phase 3 clinical trial results by modeling phase 2 clinical trial subject level data using deep learning. In *Machine Learning for Healthcare Conference*, 2019.

Qin, T., Wang, T.-Z., and Zhou, Z.-H. Budgeted heterogeneous treatment effect estimation. *ICML*, 2021.

Radcliffe, N. Using control groups to target on predicted lift: Building and assessing uplift model. *Direct Marketing Analytics Journal*, 2007.

Radcliffe, N. and Surry, P. Real-world uplift modelling with significance-based uplift trees. *Stochastic Solutions*, 2011.

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B., Chen, X., and Wang, X. A survey of deep active learning. *ACM Computing Surveys*, 2021.

Roy, N. and McCallum, A. Toward optimal active learning through Monte Carlo estimation of error reduction. *ICML*, 2001.

Rzepakowski, P. and Jaroszewicz, S. Decision trees for uplift modeling with single and multiple treatments. *Knowledge and Information Systems*, 2012.

Sandercock, P., Niewada, M., and Członkowska, A. The international stroke trial database. *Trials*, 2011.

Sener, O. and Savarese, S. Active learning for convolutional neural networks: A core-set approach. *ICLR*, 2018.

Sinha, S., Ebrahimi, S., and Darrell, T. Variational adversarial active learning. *IEEE/CVF International Conference on Computer Vision*, 2019.

Thompson, W. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 1933.

Verbeke, W., Olaya, D., Guerry, M.-A., and Van Belle, J. To do or not to do? Cost-sensitive causal classification with individual treatment effect estimates. *European Journal of Operational Research*, 2023.

Zhao, Y., Fang, X., and Simchi-Levi, D. Uplift modeling with multiple treatments and general response types. *International Conference on Data Mining*, 2017.

A. Additional experimental results

In this appendix, we provide the additional experimental results that were referenced throughout the paper.

Fig. 8 displays the performance of ATE-optimised sampling and uniform sampling in latent space across datasets, benchmarked against RCT, Recursive GSW (Addanki et al., 2022) and B-EMCMITE (Puha et al., 2020). Across datasets, this result shows that uniform sampling in latent space is not well-aligned with the ATE prediction task. It also shows

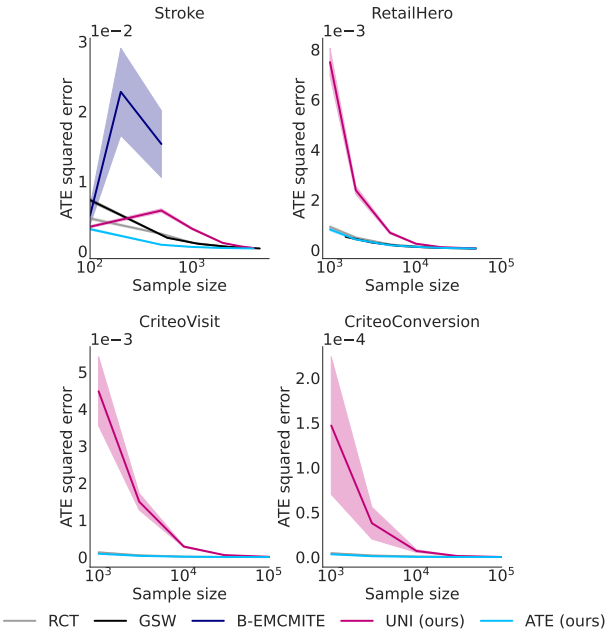

Figure 8: Performance of ATE-optimised sampler across datasets benchmarked against RCT, uniform sampling in latent space, Recursive GSW, and B-EMCMITE.

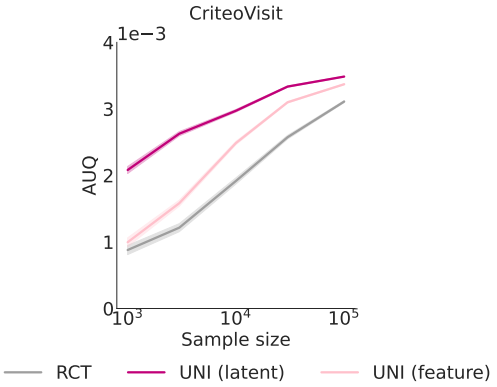

Figure 9: Comparison of uniform sampling in latent versus feature space on CRITEOVISIT.

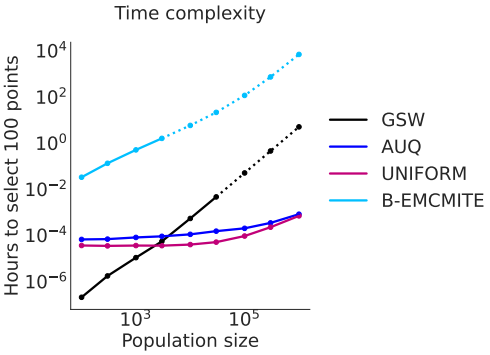

Figure 10: Empirical time complexity of our method benchmarked against Recursive GSW and B-EMCMITE.

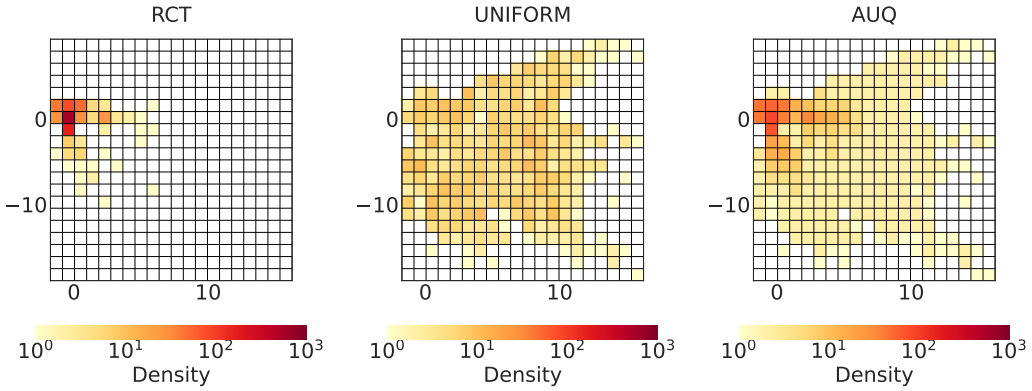

Figure 11: The first 10^3 points selected from CRITEOVISIT by RCT, uniform sampling in latent space, and our AUQ-optimised sampler displayed with respect to the rectangular grid on latent space.

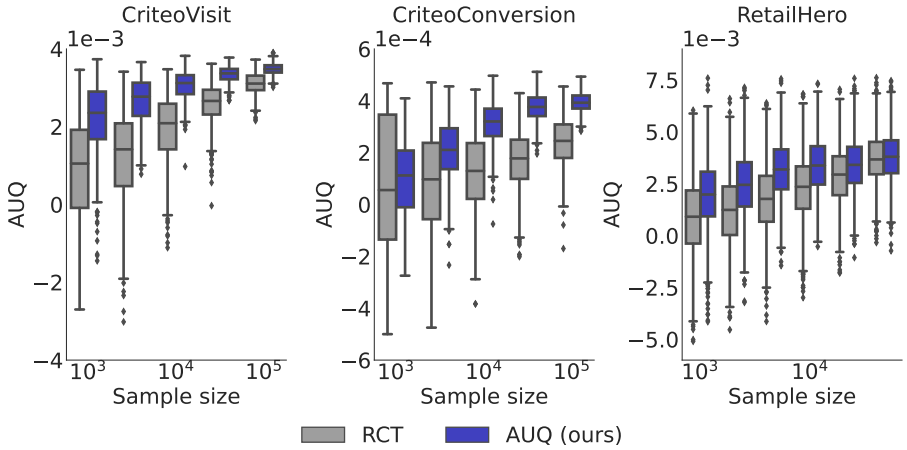

Figure 12: Box-and-whiskers plots showing the distribution in performance achieved by AUQ-optimised sampling and RCT across trials. These are the same experimental results used to compute mean performance in Fig. 5.

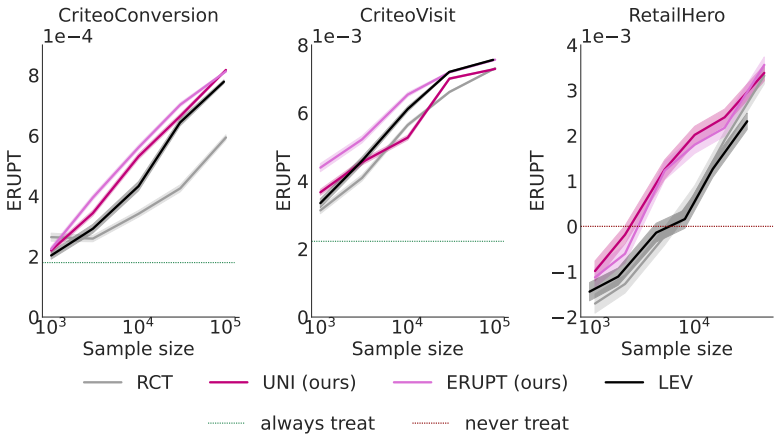

Figure 13: Performance of ERUPT-optimised sampling across datasets, benchmarked against RCT, Leverage Scoring, as well as always-treat and never-treat baselines.

that B-EMCMITE, where computationally feasible, is quite misaligned as well.

Fig. 9 demonstrates the performance of uniform sampling in latent space on CRITEOVISIT benchmarked against uniform sampling in *feature space*. For the latter, we split each raw feature at its median value, which results in $2^n = 4096$ distinct bins for this dataset with $n = 12$ features (though only 238 of them are populated). This baseline performs better than RCT but much worse than uniform sampling in latent space, demonstrating the advantage of the smooth dense representation that the VAE provides.

Fig. 10 benchmarks the time complexity of our sampling algorithms against Recursive GSW (Addanki et al., 2022) and B-EMCMITE (Puha et al., 2020) as a function of population size. For cases where it was more expensive to run the experiments at the largest population sizes, the data points were extrapolated (dashed line) from the experimentally observed results (solid line). This explains our limited application of these other methods in our experiments and demonstrates an additional advantage of our approach.

Fig. 11 shows the distributions of data points selected by RCT, uniform sampling in latent space, and our AUQ-optimised sampler. The plots display the occupancies within the rectangular grid on latent space after the first 10^3 points are drawn by each algorithm. Note the broad similarity between the cohorts selected by AUQ-optimised sampling and uniform sampling in latent space.

Fig. 12 shows the performance of our AUQ-optimised sampler, benchmarked against RCT, and displayed as a box-and-whiskers plots at each sample size. This gives visibility into the distribution of performance achieved by each method, whereas Fig. 5 showed only the mean performance over many trials.

Fig. 13 displays the performance of our ERUPT-optimised sampler, which is presented below in App. C. Our method outperforms Leverage Scoring (Addanki et al., 2022) and RCT across samples sizes and datasets. The ERUPT-optimised sampler also distinctly outperforms uniform sampling in latent space, in contrast to the results of Fig. 5 for the AUQ target. We describe both this metric and this experiment in further detail in App. C.1.

Fig. 14 explores the sensitivity of the informed prior discussed in Sec. 4.3 to inaccurate estimates of the treatment effect by systematically over- and under-estimating $n_t^{\pm}$ while keeping $n_t^+ + n_t^-$ constant. The $n_t^{\pm}$ values can be combined to predict the ATE, and we varied them in such a way as to predict both double and half the actual ATE in the test set. Fig. 14 demonstrates that our optimised samplers are robust to such substantial variations in the prior.

Fig. 15 displays the performance of our AUQ-, ATE-, and

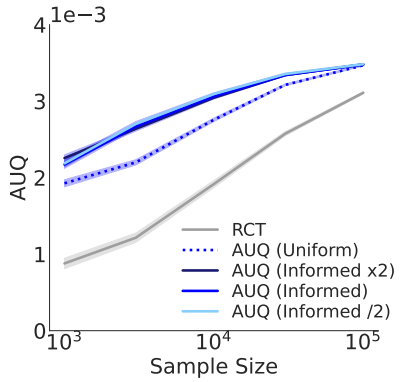

Figure 14: Dependence of AUQ-optimised sampler on the informed prior's hyperparameters on CRITEOVISIT.

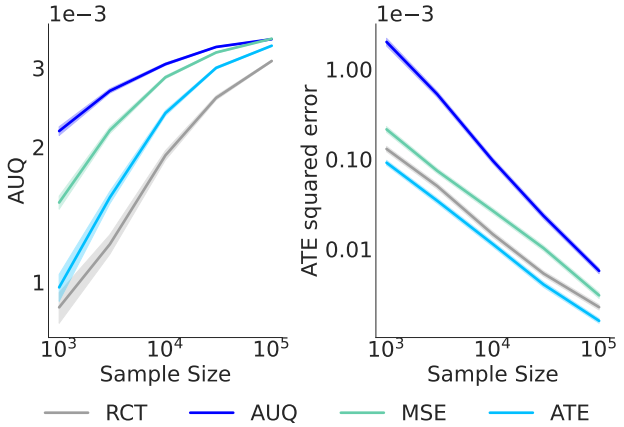

Figure 15: Performance of AUQ / ATE / MSE - optimised samplers across target metrics on CRITEOVISIT.

MSE-optimised samplers evaluated on both the AUQ and ATE-squared-error metrics on CRITEOVISIT. In a sense, this result shows that one cannot ignore the downstream task for which an experiment is being designed. No sampler, not even RCT, performs well across target metrics as disparate as AUQ and ATE squared error.

Fig. 16 displays the performance of our optimised samplers across datasets when our method is run in *batch mode*. Referring to the 4-step procedure of Sec. 3, batch mode consists of iterating between steps 1-2 (until a full batch is selected) and deferring steps 3-4 to be performed at longer intervals (between batches). This reduces the frequency at which the posterior distribution for Thompson sampling is updated with new observations. Batch sizes are indicated in parentheses in the legends of Fig. 16. This result shows positive performance even at large batch sizes.

B. Experimental details

Here we provide the details of our setup for each of our empirical analyses.

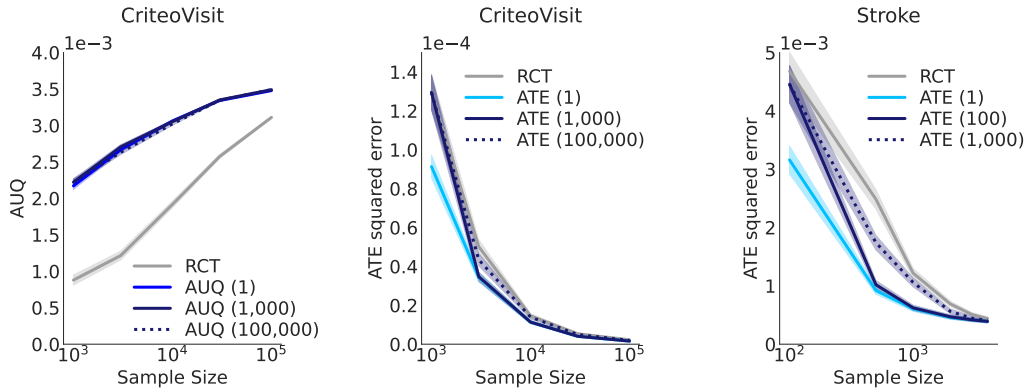

Figure 16: Performance of optimised samplers across datasets and target metrics when our method is run in batch mode (with batch sizes listed in parentheses in the legends).

B.1. Datasets

Synthetic Data

We created 1-dimensional synthetic datasets for Sec. 3.2 by sampling continuous features from a truncated normal distribution on $[0, 1]$, with mean $\mu = 0$ and $\sigma = 0.2$:

$$p(X = x) = \frac{1}{\sigma} \cdot \frac{\varphi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{1-\mu}{\sigma}\right) - \Phi\left(\frac{-\mu}{\sigma}\right)} \tag{13}$$

where φ denotes the probability density function of the standard normal distribution, and Φ is its corresponding cumulative distribution function. The data density is plotted in Fig. 1(a). We set the baseline propensity to be a linear function of the features

$$p(Y = 1|X = x, T = 0) = 0.4\,x \tag{14}$$

The baseline propensity is plotted in Fig. 1(b). We explored two sigmoidal uplift distributions, designed to replicate the situation where the largest uplift individuals are found in either the tail or the bulk of the data:

$$u_i(x) = \frac{a_i}{(1 + \exp(-b_i(x - c_i)))} \tag{15}$$

where $(a_i, b_i, c_i) = (0.7, \pm 20, 0.2)$. The two uplift functions are plotted in Fig. 1(c). Target metrics in Fig. 3 were computed on a test set of 4M points.

Stroke

We extracted this dataset from the International Stroke Trial (Sandercock et al., 2011), a large scale randomised trial studying the health outcomes of 19,435 stroke patients given different treatments upon hospital admission. We extracted 40 patient level features such as age, sex, and blood pressure which were recorded upon admittance to the hospital. To study a binary treatment effect we used patients receiving no treatment drug as the control group and patients receiving only aspirin as the treatment group. We also dropped rows from the pilot phase of the trial, as well as patients who received alternative treatments, to get a dataset of 9,208 rows. The binary outcome corresponds to whether or not the patient was discharged alive from hospital.

CriteoVisit & CriteoConversion

The Criteo Uplift Modelling dataset (Diemert et al., 2018) is a large scale benchmark for ITE estimation. It has 13,979,592 rows comprised of 12 anonymized customer features, a binary treatment indicator, and two possible binary outcome measures (Visit and Conversion). We used this dataset twice (CriteoVisit and CriteoConversion) to consider each outcome in turn. The public treatment group was randomly down-sampled to give a 50/50 balance between treatment and control groups. This resulted in a train pool of 2,992,640 rows and a test set of 4,000,000 rows.

RetailHero

We created this dataset from raw customer, sales, and marketing trial data in the scikit-uplift X5 RetailHero dataset[2] which contains raw information about previous purchases made by customers of the X5 RetailGroup. The customers were exposed to a binary treatment and their corresponding binary outcome was recorded. Customer features (e.g. age and gender) were combined with new features engineered from their purchase history e.g. total historical spend, average per transaction spend, and number of stores visited. This resulted in a dataset of 23 customer features with 100,036 train rows and 100,000 test rows. We have made this dataset and information about the engineered features publicly available.[3]

[2]Raw data: https://www.uplift-modeling.com/en/v0.3.1/api/datasets/fetch_x5.html

[3]Processed data: https://tinyurl.com/RetailHero

B.2. Models

VAE

The VAE architecture we used in our experiments is comprised of a 2-layer fully-connected encoder with 100-dimensional hidden layers, a 2-dimensional latent space, and a decoder with the same architecture as the encoder which outputs a Bernoulli probability for binary features, a softmax for categorical features, and a mean & variance for continuous features. Given the positive results this simple architecture supported, we did not tune the VAE beyond this vanilla baseline (Kingma & Welling, 2013). We trained the VAE using Adam (Kingma & Ba, 2015) with learning rate 10^{-4} and early stopping on the validation-set ELBO.

The first step of our algorithm (see Fig. 2) is to discretise the population of training data in latent space. We discretised the continuous latent representation of each dataset by taking the smallest hypercube that contains all of the data and slicing each edge of the hypercube uniformly into a number of cells (with a default of 20 unless stated otherwise). Fig. 17 provides a visualisation of the resulting latent distribution for each dataset studied in this paper.

ATE

For treatment effect estimation when targeting the ATE, we began by computing the empirical uplift in each grid cell b of discretised latent space $\mathcal{B}$ according to the sampled training data. We then estimated the ATE by weighting each of these empirical uplift values by the data density in that grid cell and summing:

$$\widehat{\text{ATE}} = \sum_{b \in \mathcal{B}} p(b) \sum_{t=0}^{1} \sum_{i=1}^{n(b,t)} \frac{(-1)^{t+1}}{n(b,t)} Y_{i(b,t)} \tag{16}$$

Here the density $p(b)$ in each cell is calculated according to our available pool of training samples.

ITE

For ITE estimation, we used custom implementations of the T-learner and the S-learner (Künzel et al., 2019). We defaulted to the T-learner in all our experiments unless stated otherwise (e.g. in Fig. 7). For either model, the core learners of the ITE estimator were XGBoost models initialised with following hyperparameters:

- `n-estimators`: 400
- `objective`: `binary:logistic`
- `eval-metric`: `rmse`
- `max-depth`: 1 (T-learner), 2 (S-learner)

Additionally during model training, the sampled data was partitioned 80/20 into training/validation sets for early stopping (with `early-stopping-rounds`: 50).

B.3. Benchmarks

We benchmarked our optimised samplers against the work of Addanki et al. (2022) which provides two sample efficient algorithms for ATE and ITE estimation, namely Recursive GSW and Leverage Scoring, respectively. The code for using these algorithms on synthetic or semi-synthetic data, where both treatment outcomes are available, is open source[4]. We modified this code in order to benchmark the methods on real data. Besides the modifications below, the algorithms were implemented exactly as reported.

The Recursive GSW algorithm assigns treatments by recursively splitting a given pool of training data $\mathcal{D}_{\text{pool}}$ into halves ($\mathcal{D}_{\text{treatment}}$ and $\mathcal{D}_{\text{control}}$) with covariates balanced between the two. The aim is to obtain a small treatment and control group which are both balanced and representative of $\mathcal{D}_{\text{pool}}$ as a whole. Since our work utilises large datasets originally obtained by RCT, we treated the two available treatment arms as independent pools of data which are already balanced. We then performed the Recursive GSW algorithm on each treatment group separately to obtain treatment and control samples.

We did not run Recursive GSW for CRITEOVISIT and CRITEOCONVERSION because the algorithm scales poorly to large datasets. See Fig. 10 for a visualisation of time for one split of a dataset of size n_{pool}.

The Leverage Scoring algorithm works by assigning sampling probabilities π to each point in $\mathcal{D}_{\text{pool}}$. The treatment and control groups are then independently sampled with probability π for each data point. To replicate this on our datasets, we again treated the available treatment and control groups as two independent pools of data with independent π. Treatment and control groups were then independently sampled as above.

Due to the stochastic nature of these batch sampling algorithms, the desired sample size is not guaranteed. Instead of reporting performance of these samplers at the target sample size (as in the original paper), we report it at the average sample size gathered over many experimental seeds which, in the case of Leverage Scoring, we found to be systematically smaller than the target size. Furthermore, for Recursive GSW we gathered sample sizes only of particular fractions of the population size ($n_{\text{pool}}/2^s$). While it is possible to collect intermediate sample sizes by random sub-sampling, this would introduce randomisation and degrade the covariate balancing that the algorithm is designed to achieve.

[4] https://github.com/raddanki/Sample-Constrained-Treatment-Effect-Estimation

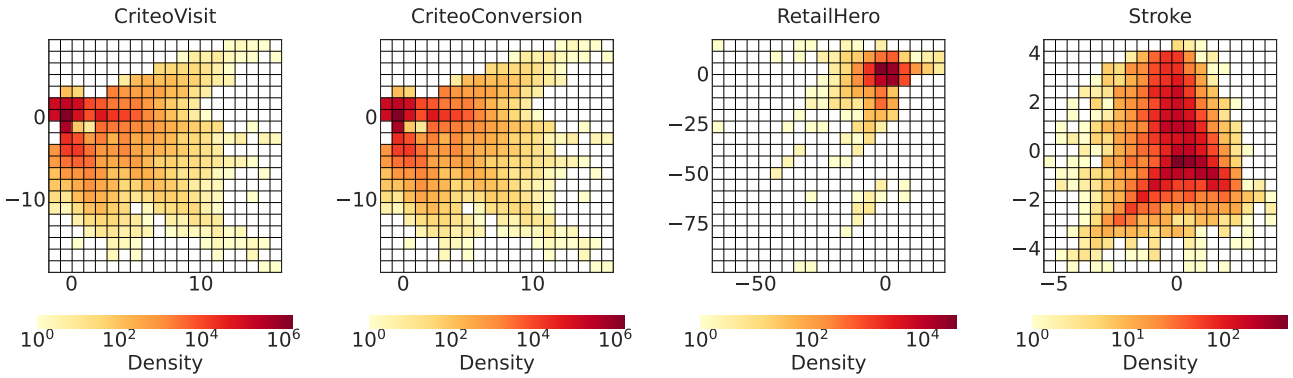

Figure 17: Discretised latent representations of the full population of each dataset studied in this paper.

B.4. Compute and empirical analysis

All experiments were performed in parallel on 96 core, 393 GB machines. For all datasets except STROKE, we performed 384 trials per experiment, and we bootstrap-resampled the test set for each trial. Because of its smaller size, experiments on STROKE each consisted of 1000 trials, and we performed a fresh train-test split for each trial. We reported the performance of each sampling method as the mean target metric achieved across these trials, with uncertainty bands representing the standard error.

C. Theoretical details

In this appendix, we provide the algorithms, derivations, and additional details mentioned throughout the paper. We begin by providing the algorithm for evaluating the Qini curve on an RCT test set in Algorithm 2.

C.1. The ERUPT metric

ERUPT[5] (Expected Response Under Proposed Treatments) (Zhao et al., 2017; Hitsch & Misra, 2018) quantifies a model's performance at the downstream task of predicting the optimal treatment to assign each member of the population. It is defined as follows.

Given an uplift model $\hat{u} : \mathcal{X} \to \mathcal{Y}$ for a binary treatment, we can use this model to decide whether to treat a member of the population x according to the rule:

$$\text{treat } x \text{ if } \hat{u}(x) > c$$

where c denotes the threshold for treatment being "worthwhile." For example, if the outcome $\mathcal{Y}$ is continuous and reflects spend, this can be interpreted as the cost of the treatment (over the baseline or control). The ERUPT metric measures the performance of such assignments by calculating the true uplift u (minus cost c) over those who were

[5]Coined by S. Weiss (2019) in an eponymous Medium article.

Algorithm 2 Qini curve estimation on RCT test set

Input: Data $\mathcal{D} = \{(x_i, t_i, y_i)\}_{i=1}^n$

Uplift model $\hat{u}$

Number of points m at which to approximate curve

Output: Qini curve values $\{(f_i, q_i)\}_{i=0}^m$

Sort $\hat{\mathcal{D}} = \{(\hat{x}_i, \hat{t}_i, \hat{y}_i)\}_{i=1}^n$ so $\hat{u}(\hat{x}_i) \geq \hat{u}(\hat{x}_{i+1})$ for all i

Set $(f_0, q_0) = (0, 0)$

for $i = 1$ **to** m **do**

$k = \lfloor n \cdot i/m \rfloor$

$k_0 = \sum_{j=1}^k (1 - \hat{t}_j)$

$k_1 = \sum_{j=1}^k \hat{t}_j$

$u = k_1^{-1} \sum_{j=1}^k \hat{t}_j \, \hat{y}_j \; - \; k_0^{-1} \sum_{j=1}^k (1 - \hat{t}_j) \, \hat{y}_j$

Set $(f_i, q_i) = (i/m, \, u \cdot i/m)$

end for

return $\{(f_i, q_i)\}_{i=0}^m$

assigned treatment by the model:

$$\text{ERUPT}[\hat{u}] = \int_{\mathcal{X}} p(x) \, \Theta(\hat{u}(x) - c) \, (u(x) - c) \, dx \quad (17)$$

where Θ is the step function defined below in Eq. (31). Larger values of ERUPT correspond to better treatment assignments by the model. ERUPT is maximal when $\hat{u} = u$.

In the experimental results of Fig. 13, the "cost" of treatment c is taken to be the ATE. This choice is motivated by the fact that with such a cost, a random baseline would break even at ERUPT$= 0$. Fig. 13 also marks the performance of "always treat" and "never treat" baselines as well.

C.2. Bias and variance of Eq. (6)

In the following sections, we provide derivations of the core functions $f_\alpha^{\mathcal{R}}\left(\{n(x,t)\}\,;\,\theta^*\right)$ that appear in Sec. 3 and

around which our method is built. We will perform these calculations explicitly for the case in which the population is partitioned into bins to ensure that the discretisation process within our method does not introduce any bias into our approach. So in this section, we begin with the derivation of the bias and variance results of Eq. (8), but under the assumption that the feature space $\mathcal{X}$ has been discretised into k bins. Let $\mathcal{B}$ represent the set of these bins. The model defined in Eq. (6) applied to bins $b \in \mathcal{B}$ then becomes:

$$\hat{U}(b) \;=\; \frac{\alpha_{b1} \;+\; \sum_{i=1}^{n(b,1)} Y_{i(b,1)}}{\alpha_{b1} \;+\; \beta_{b1} \;+\; n(b,1)} - \frac{\alpha_{b0} \;+\; \sum_{i=1}^{n(b,0)} Y_{i(b,0)}}{\alpha_{b0} \;+\; \beta_{b0} \;+\; n(b,0)} \tag{18}$$

where each $Y_{i(b,t)} \sim p(y|X = X_{i(b,t)}, T = t)$ and each $X_{i(b,t)} \sim p|_b$. Instead of estimating the uplift at a fixed $x \in \mathcal{X}$, this is now an estimate of the *average* uplift in b:

$$u(b) = \frac{1}{p(b)} \int_b \big(\mathbb{E}\left[Y|X = x, t = 1\right] - \mathbb{E}\left[Y|X = x, t = 0\right] \big)\, p(x)\, dx \tag{19}$$

where

$$p(b) := \int_b p(x)\, dx \tag{20}$$

To see this, we calculate the expected value of $\hat{U}(b)$ in the bin b. Since $\hat{U}(b)$ depends on random variables in both X and Y, we apply the law of total expectation to see that

$$\mathbb{E}\left[\sum_{i=1}^{n(l,t)} Y_i|_{X=X_i,t}\right] = \frac{n(b,t)}{p(b)} \int_b \mathbb{E}\left[Y|X = x, t\right] p(x) dx =: n(l,t)\mathbb{E}\left[Y|b,t\right] \tag{21}$$

and so

$$\mathbb{E}\left[\hat{U}(b)\right] = \frac{\alpha_{b1} \;+\; n(b,1)\,\mathbb{E}\left[Y|b,t=1\right]}{\alpha_{b1} \;+\; \beta_{b1} \;+\; n(b,1)} - \frac{\alpha_{b0} \;+\; n(b,0)\,\mathbb{E}\left[Y|b,t=0\right]}{\alpha_{b0} \;+\; \beta_{b0} \;+\; n(b,0)} \tag{22}$$

We see from this expression that in the limit of large $n(b,t)$ we have $\mathbb{E}[\hat{U}(b)] \to u(b)$. Similarly, we can apply the law of total variance to calculate

$$\mathbb{V}\left[\sum_{i=1}^{n(b,t)} Y_i|_{X=X_i,t}\right] = \frac{n(b,t)}{p(b)} \int_b \mathbb{V}\left[Y|X = x\right] p(x) dx + n(b,t)\,\mathbb{V}\left[\mathbb{E}\left[Y|X\right]\right] \tag{23}$$

where we have leveraged the pairwise independence of samples (X, Y). It transpires that

$$\mathbb{V}\left[\sum_{i=1}^{n(b,t)} Y_i|_{X=X_i,t}\right] = n(b,t)\,\sigma^2(b,t) \tag{24}$$

where

$$\sigma^2(b,t) := \mathbb{E}\left[Y^2|b,t\right] - \mathbb{E}\left[Y|b,t\right]^2 \tag{25}$$

So we can write

$$\mathbb{V}\left[\hat{U}(b)\right] = \frac{n(b,1)\sigma^2(b,1)}{(\alpha_{b1} \;+\; \beta_{b1} \;+\; n(b,1))^2} + \frac{n(b,0)\sigma^2(b,0)}{(\alpha_{b0} \;+\; \beta_{b0} \;+\; n(b,0))^2} \tag{26}$$

Writing $\theta^*_{bt} = \mathbb{E}\left[Y|b,t\right]$, we see that when $\mathcal{Y} = \{0,1\}$ we have

$$\mathbb{E}[\hat{U}(b) - u(b)] \;=\; \frac{\alpha_{b1} - (\alpha_{b1} + \beta_{b1})\,\theta^*_{b1}}{\alpha_{b1} + \beta_{b1} + n(b,1)} \;-\; (1 \to 0)$$

$$\mathbb{V}[\hat{U}(b)] \;=\; \frac{\theta^*_{b1}\,(1 - \theta^*_{b1})\,n(b,1)}{\big(\alpha_{b1} + \beta_{b1} + n(b,1)\big)^2} \;+\; (1 \to 0) \tag{27}$$

in agreement with Eq. (8), though with a slightly altered interpretation of θ^*. The model of Eq. (18) then induces the following model on $\mathcal{X}$:

$$\hat{U}(x) = \sum_{b \in \mathcal{B}} \hat{U}(b)\chi(x \in b) \tag{28}$$

where χ denotes the indicator function.

C.3. MSE-optimised sampler

Next we derive Eq. (9), but for the case in which we have discretised $\mathcal{X}$ as described above. In this case, the MSE-optimised sampler in fact aims to minimise the density-weighted squared error of the predicted average uplift in each bin $b \in \mathcal{B}$, rather than the squared error in uplift at each point $x \in \mathcal{X}$. In particular,

$$\begin{aligned}\mathbb{E}\big[\text{MSE}[\hat{U}]\big] &= \sum_{l=1}^{k} p(b)\,\mathbb{E}\big[(\hat{U}(b) - u(b))^2\big] \\ &= \sum_{b \in \mathcal{B}} p(b)\,\Big(\mathbb{E}\big[\hat{U}(b) - u(b)\big]^2 + \mathbb{V}\big[\hat{U}(b)\big]\Big) \\ &=: -f_\alpha^{\text{MSE}}\big(\{n(b,t)\};\,\theta^*\big)\end{aligned} \tag{29}$$

where dependence on prior parameters, $n(b,t)$, and θ^* can be inferred from Eq. (27).

C.4. ATE-optimised sampler

Next we derive the discretised analog of Eq. (10) for the ATE-squared error:

$$\begin{aligned}
&\mathbb{E}\Big[\big(\text{ATE}[\hat{U}] - \text{ATE}[u]\big)^2\Big] \\
&= \mathbb{E}\big[\text{ATE}[\hat{U}] - \text{ATE}[u]\big]^2 + \mathbb{V}\big[\text{ATE}[\hat{U}]\big] \\
&= \mathbb{E}\Big[\int_{\mathcal{X}} (\hat{U}(x) - u(x))p(x)\,dx\Big]^2 \\
&+ \mathbb{V}\Big[\int_{\mathcal{X}} \hat{U}(x)p(x)\,dx\Big] \\
&= \mathbb{E}\Big[\sum_{b\in\mathcal{B}} \big(p(b)\hat{U}(b) - \int_b u(x)p(x)\,dx\big)\Big]^2 \\
&+ \mathbb{V}\Big[\sum_{b\in\mathcal{B}} \hat{U}(b)p(b)\Big] \\
&= \Big(\sum_{b\in\mathcal{B}} p(b)\,\mathbb{E}\big[\hat{U}(b) - u(b)\big]\Big)^2 + \sum_{b\in\mathcal{B}} p(b)^2\,\mathbb{V}\big[\hat{U}(b)\big] \\
&=: -f_\alpha^{\text{ATE}}\big(\{n(b,t)\};\,\theta^*\big) \qquad (30)
\end{aligned}$$

where dependence on prior parameters, $n(b,t)$, and θ^* can be inferred from Eq. (27).

C.5. AUQ-optimised sampler

Next we derive Eq. (11) for the AUQ in the large $n(b,t)$ limit. We first introduce some notation. Let

$$\Theta(s) = \begin{cases} 0 & \text{if } s < 0 \\ 1/2 & \text{if } s = 0 \\ 1 & \text{if } s > 0 \end{cases} \qquad (31)$$

Then we can write

$$\begin{aligned}
\text{AUQ}[\hat{U}] = &\sum_{\substack{b,b'\in\mathcal{B} \\ b'\neq b}} p(b)\,p(b')\,u(b')\,\Theta(\hat{U}(b') - \hat{U}(b)) \\
&- \frac{1}{2}\sum_{b\in\mathcal{B}} p(b)\,u(b) \qquad (32)
\end{aligned}$$

which implies

$$\begin{aligned}
\mathbb{E}\big[\text{AUQ}[\hat{U}]\big] = &\sum_{\substack{b,b'\in\mathcal{B} \\ b'\neq b}} p(b)\,p(b')\,u(b')\,\mathbb{E}\Big[\Theta(\hat{U}(b') - \hat{U}(b))\Big] \\
&- \frac{1}{2}\sum_{b\in\mathcal{B}} p(b)\,u(b) \qquad (33)
\end{aligned}$$

We thus need to compute

$$\mathbb{E}\Big[\Theta(\hat{U}(b') - \hat{U}(b))\Big] \qquad (34)$$

We will not compute this exactly, but the central limit theorem tells us that for large $n(b,t)$ and $n(b',t)$, the difference $\hat{U}(b') - \hat{U}(b)$ is approximately normally distributed, with mean and variance given by

$$\mu(b',b) = u(b') - u(b) \qquad (35)$$

$$\sigma^2(b',b) = \frac{\sigma^2(b',1)}{n(b',1)} + \frac{\sigma^2(b',0)}{n(b',0)} + \frac{\sigma^2(b,1)}{n(b,1)} + \frac{\sigma^2(b,0)}{n(b,0)}$$

where we have used the notation Eq. (25). (The dependence on the prior parameters α_{bt} and β_{bt} drops out from $\hat{U}(b)$ in the large $n(b,t)$ limit.) Under this assumption,

$$\begin{aligned}
\mathbb{E}\Big[\Theta(\hat{U}(b') - \hat{U}(b))\Big] &\approx \int_{-\infty}^{\infty} \Theta(z)\,\varphi_{\mu,\sigma}(z)\,dz \qquad (36) \\
&= \int_0^{\infty} \varphi_{\mu,\sigma}(z)\,dz \\
&= \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{\mu(b',b)}{\sqrt{2\sigma^2(b',b)}}\right)
\end{aligned}$$

where erf denotes the error function , and we use $\varphi_{\mu,\sigma}$ to denote the normal density function with mean $\mu(b',b)$ and variance $\sigma^2(b',b)$. It follows that

$$\begin{aligned}
&\mathbb{E}\big[\text{AUQ}[\hat{U}]\big] \approx \\
&\sum_{\substack{b,b'\in\mathcal{B} \\ b'\neq b}} p(b)\,p(b')\,u(b')\left[\frac{1}{2} + \frac{1}{2}\,\text{erf}\left(\frac{u(b') - u(b)}{\sqrt{2\,\sigma^2(b',b)}}\right)\right] \\
&- \frac{1}{2}\sum_{b\in\mathcal{B}} p(b)\,u(b).
\end{aligned}$$

in agreement with Eq. (11) up to a constant independent of $n(b,t)$ that can be dropped.

C.6. ERUPT-optimised sampler

Finally we derive $f_\alpha^{\mathcal{R}}\big(\{n(x,t)\}\,;\,\theta^*\big)$ for the ERUPT metric, which turns out to be similar to the case of AUQ above:

$$\begin{aligned}
\mathbb{E}[\text{ERUPT}[\hat{U}]] &= \sum_{b\in\mathcal{B}} \mathbb{E}\Big[\Theta(\hat{U}(b) - c)\Big] \int_b (u(x) - c)\,p(x)\,dx \\
&= \sum_{b\in\mathcal{B}} p(b)\,\mathbb{E}\Big[\Theta(\hat{U}(b) - c)\Big]\,(u(b) - c)
\end{aligned}$$

In the large $n(b,t)$ limit, the central limit theorem implies that $\hat{U}(b) - c$ is approximately normally distributed, with mean $\mu_b = u(b) - c$ and variance

$$\sigma_b^2 = \frac{\sigma^2(b,1)}{n(b,1)} + \frac{\sigma^2(b,0)}{n(b,0)} \qquad (37)$$

Thus, for each $b \in \mathcal{B}$, we have that

$$\begin{aligned}\mathbb{E}\left[\Theta(\hat{U}(b)-c)\right] &= \int_{-\infty}^{\infty} \Theta(z_b)\varphi(z_b)dz_b \\ &= \int_{0}^{\infty} \varphi(z_b)dz_b \\ &= \frac{1}{2}\left[1+\text{erf}\left(\frac{\mu_b}{\sqrt{2\sigma_b^2}}\right)\right] \end{aligned} \tag{38}$$

which then implies

$$\begin{aligned}\mathbb{E}[\text{ERUPT}[\hat{U}]] &\approx \sum_{b\in\mathcal{B}} \frac{p(b)}{2}\left[1+\text{erf}\left(\frac{\mu_b}{\sqrt{2\sigma_b^2}}\right)\right](u(b)-c) \\ &=: f_\alpha^{\text{ERUPT}}(\{n(b,t)\};\theta^*) \end{aligned} \tag{39}$$

D. Connection with importance sampling

Here we briefly view our approach to task-specific experimentation from an alternative perspective. The different distributions of $n(x,t)$ values plotted in Fig. 1(d) and 1(e) might lead one to wonder whether there could be a connection between our method and importance sampling.

A weak connection between our method and importance sampling exists when the target metric can be viewed as the error on the estimation of a particular integral. This is explicitly true for the squared ATE error, and it is also true for the AUQ but not for the MSE. (Up to a shift by a constant, the AUQ can be interpreted as the difference between the estimated integral AUQ$[\hat{U}]$ and true integral AUQ$[u]$.) In this section we will assume the context of the squared ATE error.

To make the connection, note that Eq. (2) can be written as

$$\begin{aligned}\text{ATE} &= \int_{\mathcal{X},\mathcal{T},\mathcal{Y}} p(x)\,p(y|x,t)\,(-1)^{t+1}\,y\,dy\,dt\,dx \\ &= \int_{\mathcal{X},\mathcal{T},\mathcal{Y}} \frac{p(x)\,p(y|x,t)}{h(x,t)}\,(-1)^{t+1}\,y\,h(x,t)\,dy\,dt\,dx \\ &\approx \sum_{i=1}^{n} \frac{p(X_i)}{n\cdot h(X_i,T_i)}\,(-1)^{T_i+1}\,Y_i|_{X_i,T_i} \end{aligned} \tag{40}$$

where we have introduced an alternative sampling distribution $h(X,T)$ so that $h(X,T)\,p(Y|X,T)$ defines a joint probability distribution over $(\mathcal{X},\mathcal{T},\mathcal{Y})$. Eq. (40) provides an importance-sampling Monte Carlo estimate of the ATE that depends on N independent samples (X_i,T_i,Y_i) from this joint distribution. Note the similarity between Eq. (40) and ATE$[\hat{U}]$, i.e. the result of using Eq. (6) with $\alpha=\beta=0$ in order to give an estimate of Eq. (2). The only difference is that in ATE$[\hat{U}]$ the $n(x,t)$ counts are controlled manually, whereas in Eq. (40) the counts are stochastic, with expectation value $n\cdot h(x,t)$. We thus refer to Eq. (40) as ATE$[\hat{U}_h]$, as it can be interpreted as the ATE of an empirical uplift model defined using samples from $h(X,T)$.

The variance of ATE$[\hat{U}_h]$ is given by

$$\begin{aligned}&\frac{1}{n^2}\sum_{i=1}^{n}\mathbb{V}\left[\frac{p(X_i)}{h(X_i,T_i)}(-1)^{T_i+1}Y_i\right] \\ &= \frac{1}{n}\mathbb{E}\left[\frac{p(X)^2}{h(X,T)^2}Y^2\right] - \frac{1}{n}\text{ATE}^2 \\ &= \int_{\mathcal{X},\mathcal{T}} \frac{p(x)^2\,\mathbb{E}\left[Y^2|X=x,\,T=t\right]}{n\cdot h(x,t)}\,dt\,dx \;-\; \frac{\text{ATE}^2}{n} \end{aligned} \tag{41}$$

It follows from the Cauchy-Schwarz inequality that the choice of $h(X,T)$ that minimises this variance is given by

$$h^*(X,T) = \frac{1}{Z}\,p(x)\,\sqrt{\mathbb{E}\left[Y^2|X,T\right]} \tag{42}$$

where Z is a constant normalisation factor. Sampling from $h^*(X,T)$ differs from our proposed approach to task-specific experimentation presented above, both because of the stochasticity in the number of samples chosen in each discrete bin (which is $n\cdot h(x,t)$ only in expectation) and because of the weaker connection with Thompson sampling (which is described precisely in Sec. 3 for the method of this paper but which does not carry over cleanly to the importance sampling setup).

Eq. (41) can be useful though in testing whether a particular sampling distribution $h(X,T)$, e.g. an ordinary RCT or an alternative proposal (such as uniform sampling in latent space), should be expected to perform well.