LEARNING SUBGROUP-LEVEL CAUSAL ACTIONS WITH MAXIMUM INTERVENTIONAL PROBABILITIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Data-driven decision support asks who should receive which intervention to maximize a predefined outcome, a question with wide applications in critical areas such as healthcare, public policy, and economics. While existing methods mostly focus on a single actionable variable, real-world applications often involve many, possibly dependent, actionable variables, of which decision makers want to intervene on only a few while the others respond. We study the problem of subgroup-level causal action design, which aims to decide which variables to intervene on, what values to set, and which subgroup of the population to apply the intervention to, so as to maximize the interventional success probability. Such partial interventions are hard to evaluate, since the response of the untouched variables depends on the causal graph among them, which observational data do not identify. We show that the graph is nevertheless not needed for the problem of finding the *optimal* subgroup-level action: under an approximate-partition-based model of the outcome, the problem reduces to a standard supervised learning problem, and we give a finite-sample guarantee that controls the regret of the learned action by the prediction error of the learned model. We instantiate the approach with a decision tree learner, the Subgroup-level Causal Action Tree (SCAT), and, to validate its empirical performance, construct a benchmark with multiple treatments based on the widely used ACIC 2016 semi-synthetic dataset. Compared to policy learning methods adapted to our setting, SCAT finds the subgroup-level action with the highest interventional success probability on most datasets, estimates that probability most accurately, and is substantially faster.

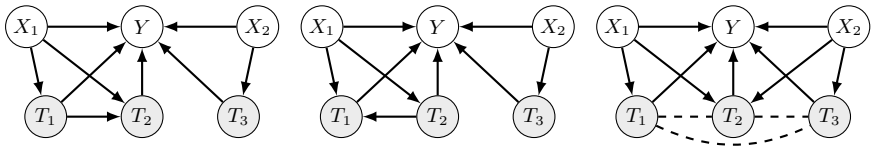
1 INTRODUCTION

In data-driven decision support for applications like healthcare, public policy, and economics, decision makers seek interventions that lead to the optimal outcome. This simultaneously concerns two questions: who should receive an intervention, and what that intervention should be, e.g., which patients should receive which treatment. Randomized experiments are the standard way to answer such questions, but are costly, slow, or unethical, and learning from observational data which action works for whom has therefore become a central problem in causal learning (Manski, 2004; Wager & Athey, 2018).

Traditional methods, known as policy learning or treatment regime learning, consider a finite set of choices and study how to assign one of them to each individual. Given a one-dimensional treatment variable T, binary or discrete with finitely many values, they decide which value of T each individual should have received (Manski, 2004; Kitagawa & Tetenov, 2018; Zhang et al., 2018; Athey & Wager, 2021; Jo et al., 2026). Some methods do formalize the problem with a multi-dimensional treatment variable (Zhou et al., 2023; Swaminathan et al., 2017; Marmarelis et al., 2024), but they always intervene on all of its coordinates at once, which we call a *full* intervention and which is equivalent to encoding the combinations of their values as the levels of one categorical T. Setting the entire treatment vector is also what allows these methods to ignore the causal relations among the treatments under common assumptions such as no hidden confounding (Pearl, 2009; Athey & Wager, 2021).

Many real-world applications, however, involve a large pool of actionable variables, and the decision maker does not necessarily intervene on all of them. A therapist can prescribe meditation, more exercise, or both, and prescribing meditation alone leaves exercise unspecified; in gene therapy, all

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) $T_1 \to T_2$
$P(t_2 \mid x, t_1)\, P(t_3 \mid x)$

(b) $T_2 \to T_1$
$P(t_2 \mid x)\, P(t_3 \mid x)$

(c) edges among T unknown
$P(t_2, t_3 \mid x, do(T_1{=}t_1)) = ?$

Figure 1: Partial interventions require the causal graph among the treatments T_1, T_2, T_3. Setting only T_1 leaves T_2 and T_3 free to respond, and $P(Y \mid x, do(T_1{=}t_1))$ is the average of $P(Y \mid x, t_1, t_2, t_3)$ over their post-intervention distribution $P(t_2, t_3 \mid x, do(T_1{=}t_1))$, which is shown under each panel. Graphs (a) and (b) are Markov equivalent, hence indistinguishable from observational data, yet give different distributions; in (c), the distribution cannot even be written down.

genes can in principle be modified, but in practice only a few are, and the remaining genes respond to the modification. These examples call for *partial* interventions, which decide on a few variables and let the rest respond, and hence strictly extend the search space of full interventions. In addition, partial interventions are cheaper, easier to interpret, and often the only ones that are admissible in practice (no one will follow a therapist who asks for ten lifestyle changes at once).

Further, a decision maker who takes responsibility for individual cases rarely wants an automatic, black-box model to decide which individual receives which treatment. They want to be supported by transparent models that reveal whether there exists an action to which a certain type of individuals responds well, together with the reasons why an individual belongs to that type. This motivates the problem of *subgroup-level causal action design*: learn from data a subgroup, the treatments to intervene on, and their values, that together maximize the probability that the target outcome is achieved under the intervention, which we call the *interventional success probability*. To meet the interpretability requirement, we consider subgroups that are described by rules, i.e., conjunctions of conditions on the covariates (Lakkaraju & Rudin, 2017; Wang & Rudin, 2022).

Compared to traditional policy learning, this problem raises two fundamental challenges. The first challenge concerns identification. Specifically, write $T = (T_A, T_{-A})$ and suppose that the intervention is on T_A only. In the do-notation of Pearl (2009), the untouched treatments T_{-A} respond to the intervention, so $P(Y \mid X, do(T_A = t_A))$ depends on their post-intervention distribution $P(T_{-A} \mid X, do(T_A = t_A))$, which in turn depends on the causal graph among the treatments, that is, on which treatments are causally connected and in which direction. Figure 1 illustrates the two obstacles this creates. On the one hand, observational data determine the causal graph only up to a Markov equivalence class (Verma & Pearl, 1991): the graphs in Figures 1(a) and (b) are indistinguishable from observational data, yet the same partial intervention has different effects in the two, so choosing between them requires prior knowledge that is rarely available. On the other hand, with more treatments, each possible graph among them yields a different identification formula (Figure 1(c)), and learning even the equivalence class from data becomes statistically fragile as the dimension of T grows (Reisach et al., 2021). The second challenge is computational and statistical at the same time. The search over subgroups, treatments to intervene on, and their values is combinatorial, with 3^m partial interventions for m binary treatments alone, and at the level of full configurations positivity fails, since most of the 2^m configurations never occur among n observations, so a method that sets all of T can only recommend configurations it has seen. A partial intervention only needs data on the subgroup and the few treatments it sets, which is exactly what makes it estimable, and why we search over partial interventions directly.

The key finding of this paper is that, even though estimating the interventional probability of an arbitrary partial intervention requires the causal graph among the treatments, the graph is *not* needed to find the *optimal* subgroup-level action. The problem reduces to a standard supervised learning problem: the best subgroup-level action can be obtained by learning a partition-based model of the outcome, such as a decision tree, with the covariates and the treatments jointly as input features, and reading the action off the learned model. We instantiate this with a decision tree learner, SCAT, for Subgroup-level Causal Action Tree, and show that the regret of its recommendation is controlled by the prediction error of the learned model, which gives a finite-sample guarantee without recovering the graph or the true partition.

Our contributions are summarized as follows. **(1) Problem formulation and identification without the graph (Sections 3.1 and 3.2).** We formalize *subgroup-level causal action design*: choose a subgroup, a subset of treatment coordinates, and their values, so as to maximize the interventional success probability, while the untouched treatments are left free to respond. Under a γ-approximate partition of the covariate-treatment space, we show that no subgroup-level action can beat the best observational cell response by more than γ, and that reading the action off a best cell attains this value up to 2γ. This reduces the problem to fitting a partition-based model of $P(Y \mid X, T)$, a standard supervised learning task. **(2) Finite-sample guarantee (Section 3.3).** For a partition learned by any penalized maximum likelihood tree and ranked on a held-out sample, we bound the regret of the recommended action by the prediction error of the learned model. Combined with an oracle inequality for penalized likelihood trees, this gives a finite-sample regret bound whose only causal ingredients are treatment overlap and the size of the selected subgroup. **(3) Method (Section 4).** We instantiate the approach suggested by the theory as SCAT, the Subgroup-level Causal Action Tree, whose learning criterion is an instance of the penalized likelihood criterion covered by our guarantee. **(4) Benchmark and experiments (Section 5).** Since no existing benchmark has multiple, possibly dependent, treatment variables with known interventional ground truth, we extend the semi-synthetic ACIC 2016 benchmark to multi-dimensional treatments. We validate our method on both one- and multi-dimensional binary treatments, and show that SCAT selects the action with the highest true interventional success probability on 31 of the 43 datasets, estimates that probability most accurately, and in the multi-treatment setting runs more than 20 times faster than the fastest competitor.

2 Related work

Policy learning from observational data. A large literature learns from observational data which treatment each individual should receive, under names such as statistical treatment rules, policy learning, and treatment regimes. Most methods proceed in two steps: they first estimate, for each treatment value, the expected outcome of each individual, e.g., by regression, inverse propensity weighting, or doubly robust scores (Dudík et al., 2011; Wager & Athey, 2018; Künzel et al., 2019), and then choose the assignment rule from a policy class that maximizes the estimated total outcome (Manski, 2004; Kitagawa & Tetenov, 2018; Athey & Wager, 2021). Black-box policy classes assign each individual an arbitrary function of its covariates, whereas interpretable classes constrain the rule to be a decision list (Lakkaraju & Rudin, 2017; Zhang et al., 2018), a tree (Athey & Wager, 2021; Zhou et al., 2023; Jo et al., 2026), or a set of hyper-boxes (disjunctive normal forms) (Tschernutter et al., 2022). A few methods go beyond a finite set of actions and consider continuous or combinatorial treatments (Kallus & Zhou, 2018; Marmarelis et al., 2024; Swaminathan et al., 2017). Whatever the policy class or the treatment domain, these methods assign an action to every individual and set the treatment in full. This is what makes them graph-free: a full intervention cuts every edge into the treatments, so without hidden confounding $P(Y \mid X, do(T = t)) = P(Y \mid X, T = t)$, and both estimation and optimization are supervised learning problems. Our goal is different: we return one subgroup and one partial intervention on a multi-dimensional discrete treatment vector, and leave the other treatments free to respond.

Causal subgroup discovery. A related research line searches for subgroups in which a *fixed* binary treatment has an enhanced effect, i.e., in which $do(T = 1)$ improves the most over $do(T = 0)$, with tree- and rule-based methods developed for clinical trials and for heterogeneous effect estimation (Lipkovich et al., 2011; Foster et al., 2011; Athey & Imbens, 2016; Wang & Rudin, 2022; Yang et al., 2026). All of these methods share our interest in an interpretable, rule-based subgroup, but take the treatment as given and one-dimensional. We instead search jointly over the subgroup, which of many, possibly dependent, treatment variables to intervene on, and the values to set, which raises the identification problem of partial interventions that a single treatment never poses.

Partial interventions with an unknown graph. When the causal graph among the variables is unknown, the effect of a partial intervention is not identifiable in general. Existing methods hence estimate the graph up to its Markov equivalence class and enumerate the effects that are consistent with the class, as IDA and its extensions do (Maathuis et al., 2009; Nandy et al., 2017; Perkovic, 2020), and therefore return a set of possible effects for one intervention rather than a ranking of interventions, let alone a subgroup. They also inherit the difficulty of structure learning in high dimensions (Spirtes et al., 2001; Zheng et al., 2018), whose reported successes can be artifacts of simulation regularities (Reisach et al., 2021). We do not estimate the graph at all and show that the

optimal subgroup-level action is identified without it. Further, when interventions can be performed, interventional data reduce the equivalence class (Hauser & Bühlmann, 2012) and causal bandits choose interventions adaptively (Lattimore et al., 2016; Lee & Bareinboim, 2018). These methods require the ability to intervene during learning, whereas we work with observational data only.

3 Theory: Maximizing the interventional success probability

We present our theory in the following two parts. First, if the observational response probability $P(Y = 1 \mid X, T)$ is approximated by a partition of the covariate-treatment space into cells, with approximation error γ, then a subgroup-level action whose interventional success probability is within 2γ of the optimum can be read off from a best cell, without any knowledge of the causal graph among the treatments (Section 3.2). Second, intuitively, if we learn a partition-based model from data that recovers the observational response probability well, γ will be small and the read-off works; we formalize this intuition as a finite-sample guarantee (Section 3.3) for penalized maximum likelihood trees, such as our MDL-based tree in Section 4.

3.1 Setup and problem statement

Setup. Let $X = (X_1, \ldots, X_p) \in \mathcal{X}$ denote the non-actionable covariates, let $T = (T_1, \ldots, T_m) \in \mathcal{T} := \prod_{j=1}^{m} \mathcal{T}_j$ denote the treatment variables, and let $Y \in \{0, 1\}$ be the target, where $Y = 1$ denotes success. Each treatment domain $\mathcal{T}_j$ is finite; the covariates may be discrete or continuous, and all spaces are standard Borel. We assume that (X, T, Y) arise from a recursive structural causal model (SCM) with mutually independent exogenous variables. The covariates and treatments may each form an arbitrary directed acyclic graph (DAG). Edges $X_i \to T_j$, $X_i \to Y$, and $T_j \to Y$ are allowed, but there are no edges from treatments to covariates and no outgoing edges from Y. Thus, $Y = f_Y(X, T, U_Y)$, where U_Y is independent of the other exogenous variables. We assume *conditional joint treatment positivity*:

$$P(T = t \mid X = x) > 0 \quad \text{for every } t \in \mathcal{T}, \quad \text{for } P_X\text{-almost every } x. \tag{1}$$

This condition allows dependence among treatments, but excludes treatment configurations that are impossible given the covariate values. Write $P_{X,T}$ for the observational joint distribution; conditional probabilities such as $P(Y = 1 \mid X = x, T = t)$ are understood up to $P_{X,T}$-null sets, which positivity makes harmless under interventions (Lemma A.1 in Appendix A).

For $A \subseteq \{1, \ldots, m\}$, write $T_A = (T_j)_{j \in A}$, $T_{-A} = (T_j)_{j \notin A}$, and $\mathcal{T}_A = \prod_{j \in A} \mathcal{T}_j$. A partial intervention fixing the coordinates in A is denoted $do(T_A = t_A)$ (Pearl, 2009); $A = \varnothing$ and $A = \{1, \ldots, m\}$ correspond to no intervention and a full intervention. The expression (t_A, t_{-A}) denotes the full treatment vector with coordinates restored to their original order.

Problem statement. We are interested in actions that fix some treatment coordinates within a subgroup of the population. Let $\mathcal{S}$ be a nonempty class of measurable subgroups $S \subseteq \mathcal{X}$ with $P(X \in S) > 0$, and let $\mathcal{A} \subseteq 2^{\{1,\ldots,m\}}$ be a nonempty family of admissible sets of treatment coordinates. In practice, each intervention comes at a cost, e.g., in public policy or in clinical practice, where only a limited number of treatments can be changed at the same time. The family $\mathcal{A}$ can encode such constraints, e.g., $\mathcal{A} = \{A : |A| \leq s\}$ for a budget of s treatments. The family may also include or exclude $\varnothing$, depending on whether "do not intervene" is an acceptable recommendation for the task at hand (e.g., in policy making or mental health therapy, it is informative for the decision makers to know that the best action for a certain subgroup is 'do not intervene'). Throughout the paper, we do not consider such constraints and hence set $\mathcal{A} = 2^{\{1,\ldots,m\}}$, so that we keep the problem in the general form; adding the constraints in practice can be easily done via restricting the search space of the subgroup-level actions. For $S \in \mathcal{S}$, $A \in \mathcal{A}$ and $t_A \in \mathcal{T}_A$, write $V(S, A, t_A) := P(Y = 1 \mid X \in S, do(T_A = t_A))$ and $V^\star := \sup_{S \in \mathcal{S},\, A \in \mathcal{A},\, t_A \in \mathcal{T}_A} V(S, A, t_A)$ for the interventional success probability of fixing T_A to t_A in the subgroup S, and the optimal value.

Definition 3.1 (Subgroup-level causal action design)**.** A *subgroup-level action* is a triple (S, A, t_A) of a subgroup $S \in \mathcal{S}$, treatment coordinates $A \in \mathcal{A}$, and values $t_A \in \mathcal{T}_A$; its *regret* is $V^\star - V(S, A, t_A) \geq 0$. We call a subgroup-level action that attains $V^\star$ an *optimal subgroup-level causal action*. The goal of subgroup-level causal action design is to find a subgroup-level action with small regret.

3.2 READING OFF THE OPTIMAL SUBGROUP-LEVEL CAUSAL ACTION FROM A PARTITION

The difficulty is that, without the causal graph among the treatments, the post-intervention distribution of the untouched T_{-A} is not identified. We overcome it through the following assumption.

Assumption 3.2 (γ-approximate partition)**.** There are a measurable partition $\{C_k\}_{k=1}^{M}$ of $\mathcal{X} \times \mathcal{T}$ with $P\{(X,T) \in C_k\} > 0$ for every k, and a constant $\gamma \in [0,1]$, such that the observational cell responses $\theta_k := P\{Y = 1 \mid (X,T) \in C_k\}$, $k = 1, \ldots, M$, satisfy

$$|P(Y = 1 \mid X = x, T = t) - \theta_k| \leq \gamma \quad \text{for } P_{X,T}\text{-almost every } (x,t) \in C_k, \quad k = 1, \ldots, M. \tag{2}$$

Since both probabilities in Equation 2 lie in $[0,1]$, the assumption always holds with $\gamma = 1$; it is informative when the partition is fine enough for γ to be small, and $\gamma = 0$ is the exact partition-based model. We focus on *hyper-box cells* $C_k = \prod_{\ell=1}^{p+m} I_{k\ell}$, where each $I_{k\ell}$ is an interval relative to its coordinate domain when that coordinate is ordered, or a subset of its levels when categorical. Such cells are described by conjunctions of single-coordinate conditions, as in axis-aligned decision trees and disjoint rule partitions.

Theorem 3.3. *Under positivity (Equation 1) and Assumption 3.2:*

(i) For every measurable $S \subseteq \mathcal{X}$ with $P(X \in S) > 0$, every $A \subseteq \{1, \ldots, m\}$, and every $t_A \in \mathcal{T}_A$, $P(Y = 1 \mid X \in S, do(T_A = t_A)) \leq \max_{1 \leq k \leq M} \theta_k + \gamma$.

(ii) Suppose, in addition, that every cell of the partition has the form

$$C_k = \{(x,t) : x \in S_k,\ t_{A_k} \in Q_k\}, \qquad S_k \subseteq \mathcal{X}, \quad A_k \in \mathcal{A}, \quad Q_k \subseteq \mathcal{T}_{A_k}, \tag{3}$$

where S_k is measurable and A_k is the smallest set of treatment coordinates for which such a representation exists. Then, for every k, $P(X \in S_k) > 0$, and $\left|V(S_k, A_k, t_{A_k}) - \theta_k\right| \leq \gamma$ for every $t_{A_k} \in Q_k$. In particular, if some $k^\star \in \arg\max_k \theta_k$ satisfies $S_{k^\star} \in \mathcal{S}$, then, for every $t_{A_{k^\star}} \in Q_{k^\star}$,

$$0 \leq V^\star - V(S_{k^\star}, A_{k^\star}, t_{A_{k^\star}}) \leq 2\gamma. \tag{4}$$

The construction in (ii) requires no knowledge of the causal graph among treatments and is optimal when $\gamma = 0$. All proofs are given in Appendix A.

A cell of the form of Equation 3 leaves the untouched T_{-A_k} free, which is what allows them to respond to the intervention without leaving the cell. Every hyper-box cell has this form, with S_k its covariate part, A_k the treatment coordinates whose factors are proper subsets of their domains, and Q_k the product of these factors (Appendix A.1). Hence a hyper-box partition directly specifies a subgroup, the treatments to intervene on, and their values. Its cell response $\theta_k = P(Y = 1 \mid X \in S_k, T_{A_k} \in Q_k)$ is an observational quantity.

3.3 FINITE-SAMPLE GUARANTEE FOR A PARTITION LEARNED FROM DATA

Theorem 3.3(ii) motivates learning a partition-based response model from data and reading the action off a best cell. Intuitively, if the learned model recovers the observational response probability well, its cells approximate $P(Y = 1 \mid X, T)$ well, so γ is small and the read-off is near-optimal. We formalize this intuition as a finite-sample guarantee, in which the closeness of the learned model to $P(Y = 1 \mid X, T)$ is measured by its average squared prediction error, which plays the role of γ in Theorem 3.3(ii), and which penalized maximum likelihood trees make small.

Learning procedure. We specialize to binary treatments $T \in \{0,1\}^m$, write $d = p + m$, and split the data into a training sample of size n and a validation sample of size n_v. On the training sample, an axis-aligned tree $\widehat{\Pi}$ with $\widehat{K}$ leaves is fitted by minimizing a penalized negative log-likelihood, $\ell_n(\Pi, \widehat{\vartheta}_\Pi) + \mathrm{pen}_n(\Pi)$, over a candidate set of trees $\mathfrak{P}_n$, where $\widehat{\vartheta}_\Pi$ are the leafwise maximum likelihood estimates and pen_n is any finite nonnegative penalty; Section 4 shows that our MDL criterion has this form. The recommended subgroup-level action $(\widehat{S}, \widehat{A}, \widehat{t}_{\widehat{A}})$ is read off, as in Theorem 3.3(ii), the leaf whose success proportion on the validation sample is largest. Appendix A.3 gives the formal definitions.

Quantitative overlap. For the statistical guarantee, we strengthen Equation 1 to $P(T = t \mid X = x) \geq \kappa > 0$ for every $t \in \mathcal{T}$ and P_X-almost every x, which controls how strongly an intervention can emphasize configurations that are rare in observational data.

Theorem 3.4 (informal; exact statement in Theorem A.11)**.** *Assume quantitative overlap, that the leaves of every candidate tree define admissible subgroup-level actions, and that some candidate tree Π° with K° leaves represents $P(Y = 1 \mid X, T)$ exactly; let S° be the subgroup read off its best leaf. Then, with probability at least $1 - 2\delta$ over the training and validation samples,*

$$0 \le V^\star - V(\widehat{S}, \widehat{A}, \widehat{t}_{\widehat{A}}) \le \sqrt{\frac{\mathcal{R}_n(\delta)}{\kappa}} \left(\frac{1}{\sqrt{P(X \in S^\circ)}} + \frac{1}{\sqrt{P(X \in \widehat{S})}} \right), \tag{5}$$

$$\textit{where} \qquad \mathcal{R}_n(\delta) = \widetilde{O}\left(\frac{\mathrm{pen}_n(\Pi^\circ)}{n} + \sqrt{\frac{(K^\circ + \widehat{K}) \log(d/\delta)}{n}} + \frac{\widehat{K}}{\delta\, n_v} \right),$$

and $\widetilde{O}$ hides logarithmic factors in n. In particular, if the tree sizes are bounded, $\mathrm{pen}_n(\Pi^\circ) = O(\log n)$, $n_v \asymp n$, and κ, $P(X \in S^\circ)$ and $P(X \in \widehat{S})$ are bounded below, the regret is $\widetilde{O}(n^{-1/4})$.

The quantity $\mathcal{R}_n(\delta)$ bounds the average squared prediction error of the validation estimate, through an oracle inequality for the penalized likelihood tree (Theorem A.8 in Appendix A). The two subgroup probabilities and the overlap constant κ are the price of not knowing the graph: accurate prediction on average need not imply accurate prediction on a small subgroup or on rare treatment configurations. Apart from these, the bound is a standard supervised learning guarantee, and it motivates our MDL-based approach. Appendix A gives the general version for a partition that only satisfies Assumption 3.2, with an additional 2γ and an approximation term; the theorem does not assert that the learned partition recovers Π°.

4 Method: Subgroup-level Causal Action Tree (SCAT)

To learn the partition-based model from data, we learn a decision tree whose leaves form the cells of the partition, using the minimum description length (MDL) principle (Rissanen, 1978; Grünwald, 2007), which selects the model that compresses the model and the data together into the fewest bits and can be regarded as a penalized maximum likelihood method (Grünwald & Roos, 2019). While our theory does not restrict which supervised learning algorithm to use, we choose an MDL-based tree because, unlike a penalty on the tree size, MDL requires no regularization coefficient to tune, which makes the method efficient (Section 5), and because MDL-based trees (Ross Quinlan & Rivest, 1989; Mehta et al., 1995), in particular the recent CDTree (Yang & van Leeuwen, 2024), are accurate and robust against spurious features in high dimensions. We build on CDTree, which only handles numeric targets, and adapt it to binary targets.

Learning criterion. The MDL-based learning criterion consists of two parts. The first part is the negative log-likelihood of Y given X and T under a fixed decision tree, with the probabilities of the target values estimated in each leaf. In our implementation, we smooth these estimates with the Laplace correction to be robust against leaves with few instances, in which the unsmoothed estimates can be exactly 0 or 1. The second part is a penalty that consists of two terms, the model encoding cost and the encoding regret (not to be confused with the regret of an action in Definition 3.1). The model encoding cost $L_{\mathrm{tree}}(\Pi \mid Z^n)$, with $Z^n = \{(X_i, T_i)\}_{i=1}^n$ the training inputs, is the code length, in bits, needed to losslessly encode the decision tree itself, including its number of nodes, its structure, and all its splitting conditions. We adopt exactly the same encoding as CDTree, which uses uniform and pseudo-uniform codes at every step. By Kraft's inequality, $\sum_{\Pi \in \mathfrak{P}_n} 2^{-L_{\mathrm{tree}}(\Pi \mid Z^n)} \le 1$, so these codes correspond to a non-informative prior over the model class of all possible decision trees. If split locations are encoded through empirical quantiles, this is a predictor-dependent tree code rather than a fixed prior on prediction functions.

The encoding regret term, under the so-called Shtarkov (or normalized maximum likelihood, NML) encoding (Shtarkov, 1987), is defined as $\log \sum_{y^n} \hat{P}(y^n \mid x^n, t^n)$, where the sum is over all possible sequences y^n and $\hat{P}(y^n \mid x^n, t^n)$ is the maximum likelihood of the sequence y^n under the tree. Because the response is binary, each leaf uses a Bernoulli model, the sum factorizes over the leaves, and each factor can be computed in linear or sub-linear time (Mononen & Myllymäki, 2008). Without smoothing, this MDL criterion has the penalized likelihood form analyzed in Section 3.3, with the tree code plus the encoding regret as the penalty $\mathrm{pen}_n(\Pi)$ (Appendix A.4). SCAT implements this approach using greedy growth, exact pruning of the grown tree, and Laplace-smoothed estimates.

Growing and pruning the tree. We grow the tree greedily, choosing at each node the split over all covariates and treatments that minimizes the code length of the tree and the data. Unlike CDTree, which stops as soon as no split decreases the code length, we grow the tree until every leaf is pure or cannot be split, and then prune it back to the subtree with the minimal code length; this avoids the horizon effect, in which a split that does not pay off by itself but enables later splits that do is never made. Since the code length of the tree structure depends on the total number of leaves, it is not additive over subtrees and bottom-up pruning is not exact, so we find the optimal subtree by dynamic programming over the number of retained leaves (Algorithms 1 and 2 in Appendix B).

Ranking the leaves and selecting the top action. Based on our theory, the subgroup-level causal action can be read off from the learned partition-based model, which requires ranking the leaves of the learned tree. Each leaf defines a candidate subgroup-level action, in which the conditions on X along its path give the subgroup and the conditions on T give the intervention, while treatment variables that do not appear on the path are left untouched. We rank the leaves by their estimated probability of $Y = 1$ on a held-out validation set, independently of the tree fitting, and return the top one. Algorithm 3 in Appendix B summarizes the full procedure.

5 Experiments

We evaluate SCAT in two settings: 1) multiple treatment variables, the setting our method is designed for, on a new benchmark that extends the semi-synthetic ACIC 2016 to multi-dimensional treatments together with three hand-built datasets that cover the basic causal structures among the treatments (Section 5.1); 2) the widely studied case of a single binary treatment, on the original ACIC 2016 benchmark (Section 5.2). All source code is available at `https://anonymous.4open.science/r/SCAT-B3E6/`. Appendix C gives the full experimental setup.

5.1 Multiple treatment variables

Datasets. As no established benchmark has multiple treatment variables, we extend the semi-synthetic ACIC 2016 benchmark (Dorie et al., 2019), which has a one-dimensional T, to multi-dimensional treatments, following the way the original benchmark is constructed; we call the result MT. We first randomly generate a causal graph among the treatments $\{T_j\}_{j=1,\dots,m}$, so that each T_j is a random function of its parents in this graph and of randomly selected covariates, and then sample Y as a random function of X and T in which the treatments affect Y through latent variables, reflecting that observed variables often act on the outcome through unobserved mechanisms. All random functions are drawn from the same pool as in ACIC 2016 (Appendix E). We use ten replicates MT01–MT10 with $m = 20$ treatments. In addition, we generate three hand-built datasets with a hand-designed graph among three treatments, covering the basic structures Chain, Fork, and Collider (Appendix E.6).

Competitor methods. Since no existing method jointly optimizes which treatments to intervene on, their values, and the subgroup, we compare against state-of-the-art policy learning methods that learn subgroup-level treatment rules, namely ListDTR (Zhang et al., 2018), PolicyTree (PT) (Zhou et al., 2023) with its Exact and Hybrid algorithms, and FlowOPT-DR (Jo et al., 2026), adapted to our setting as follows. For a fixed set of k treatments, the 2^k joint values are encoded as one categorical treatment, on which the competitor learns a policy whose assignments are decoded back into subgroup-level actions. As this requires choosing the k treatments beforehand, we rank the treatments by their doubly robust average treatment effect estimated with a causal forest (Athey et al., 2019), run each competitor on the top k for $k = 1, \dots, 4$, pool the candidate actions, and select the one with the highest estimated interventional probability on the validation set. Note that $k = 4$ equals the number of treatments that directly affect Y in every MT dataset, information that SCAT does not have, so this setup, if anything, favors the competitors. We further consider three two-step variants that first learn the Markov blanket of Y with Lasso (Meinshausen & Bühlmann, 2006), IAMB (Tsamardinos et al., 2003), or DRScreen (Athey et al., 2019) and then apply PolicyTree to the selected treatments (MB-Lasso, MB-IAMB, MB-DRScreen).

Evaluation. Each dataset is split into training, validation, and test sets (50%, 20%, 30%); every method selects the action that maximizes its estimated interventional probability on the validation set, and all metrics are computed on the test set. We repeat every experiment over five random splits and report mean and standard deviation, except for the Gurobi-based competitors (FlowOPT-DR, and

Table 1: True interventional $P(Y = 1)$ of the selected subgroup-level action on the multi-treatment datasets: mean$_{\pm \text{ sd}}$ over five splits, best in bold. †Run on the first split only (Gurobi license restriction).

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	MB-IAMB	MB-Lasso	MB-DRScreen	FlowOPT-DR†
Chain	$0.776_{\pm 0.050}$	$0.793_{\pm 0.071}$	$0.824_{\pm 0.051}$	$0.875_{\pm 0.033}$	$0.852_{\pm 0.042}$	$0.879_{\pm 0.029}$	$0.852_{\pm 0.042}$	$\mathbf{0.885}$
Fork	$\mathbf{0.899}_{\pm 0.005}$	$0.857_{\pm 0.061}$	$0.853_{\pm 0.037}$	$0.834_{\pm 0.056}$	$0.834_{\pm 0.056}$	$0.841_{\pm 0.054}$	$0.834_{\pm 0.056}$	0.884
Collider	$\mathbf{0.767}_{\pm 0.024}$	$0.715_{\pm 0.080}$	$0.675_{\pm 0.138}$	$0.699_{\pm 0.157}$	$0.643_{\pm 0.167}$	$0.612_{\pm 0.130}$	$0.693_{\pm 0.100}$	0.656
MT01	$\mathbf{0.776}_{\pm 0.069}$	$0.586_{\pm 0.040}$	$0.534_{\pm 0.082}$	$0.522_{\pm 0.071}$	$0.537_{\pm 0.078}$	$0.522_{\pm 0.071}$	$0.522_{\pm 0.071}$	0.587
MT02	$0.562_{\pm 0.034}$	$0.627_{\pm 0.077}$	$0.552_{\pm 0.026}$	$0.568_{\pm 0.025}$	$0.569_{\pm 0.025}$	$0.573_{\pm 0.028}$	$0.557_{\pm 0.034}$	$\mathbf{0.675}$
MT03	$\mathbf{0.740}_{\pm 0.048}$	$0.616_{\pm 0.125}$	$0.593_{\pm 0.048}$	$0.526_{\pm 0.036}$	$0.553_{\pm 0.064}$	$0.502_{\pm 0.115}$	$0.493_{\pm 0.096}$	0.598
MT04	$0.601_{\pm 0.038}$	$\mathbf{0.625}_{\pm 0.052}$	$0.527_{\pm 0.100}$	$0.503_{\pm 0.103}$	$0.515_{\pm 0.101}$	$0.564_{\pm 0.055}$	$0.511_{\pm 0.102}$	0.575
MT05	$0.616_{\pm 0.030}$	$0.541_{\pm 0.064}$	$\mathbf{0.621}_{\pm 0.058}$	$0.598_{\pm 0.069}$	$0.598_{\pm 0.069}$	$0.530_{\pm 0.036}$	$0.598_{\pm 0.069}$	0.595
MT06	$0.621_{\pm 0.093}$	$0.572_{\pm 0.077}$	$0.584_{\pm 0.050}$	$0.575_{\pm 0.085}$	$0.575_{\pm 0.085}$	$0.569_{\pm 0.089}$	$0.575_{\pm 0.085}$	$\mathbf{0.628}$
MT07	$\mathbf{0.750}_{\pm 0.052}$	$0.558_{\pm 0.079}$	$0.558_{\pm 0.060}$	$0.516_{\pm 0.069}$	$0.516_{\pm 0.069}$	$0.516_{\pm 0.069}$	$0.516_{\pm 0.069}$	0.552
MT08	$\mathbf{0.605}_{\pm 0.039}$	$0.552_{\pm 0.036}$	$0.492_{\pm 0.095}$	$0.504_{\pm 0.112}$	$0.484_{\pm 0.106}$	$0.484_{\pm 0.106}$	$0.489_{\pm 0.107}$	0.590
MT09	$0.678_{\pm 0.045}$	$\mathbf{0.693}_{\pm 0.133}$	$0.573_{\pm 0.042}$	$0.552_{\pm 0.056}$	$0.543_{\pm 0.059}$	$0.526_{\pm 0.069}$	$0.551_{\pm 0.056}$	0.602
MT10	$\mathbf{0.682}_{\pm 0.004}$	$0.624_{\pm 0.089}$	$0.560_{\pm 0.040}$	$0.601_{\pm 0.050}$	$0.616_{\pm 0.039}$	$0.613_{\pm 0.041}$	$0.616_{\pm 0.039}$	0.627

Table 2: Summary over the 13 multi-treatment datasets: mean over datasets and average rank (1 = best); best per row in bold. †First split only.

		SCAT	ListDTR	PT-Exact	PT-Hybrid	MB-IAMB	MB-Lasso	MB-DRScreen	FlowOPT-DR†
Mean	True $P(Y = 1) \uparrow$	**0.698**	0.643	0.611	0.606	0.603	0.595	0.600	0.650
	|Estimation error| $\downarrow$	**0.029**	0.055	0.040	0.050	0.049	0.043	0.046	0.071
	Test coverage	0.229	0.486	0.371	0.248	0.281	0.280	0.287	0.320
	Runtime (s) $\downarrow$	**39.9**	13,627	1,090	1,329	1,721	861	1,184	11,218
Rank	True $P(Y = 1)$	**2.23**	3.31	4.69	5.42	5.54	6.15	5.88	2.77
	|Estimation error|	**2.69**	5.46	3.31	5.50	5.08	3.77	4.42	5.77
	Runtime	**1.00**	7.85	3.92	4.85	5.23	2.38	3.62	7.15

IOPL in Section 5.2), which the license prevents from running in parallel and which we run on the first split only. We report four metrics: the ground-truth interventional probability $P(Y = 1)$ of the selected action on its subgroup, which is the objective in Definition 3.1; the absolute error of the method's own estimate of that probability, since decision support needs an accurate estimate and not only a ranking; the coverage of the subgroup, i.e., the fraction of test instances it contains, since a very small subgroup is of little practical use and makes the estimate unstable; and the runtime.

Results. As shown in Table 1, SCAT selects the subgroup-level causal action with the highest true interventional probability on 7 of the 13 datasets. On four of the remaining six (MT04, MT05, MT06, and MT09) it ranks second, at most 0.025 below the best competitor. This holds even though the competitors consider interventions on up to four treatments, which matches the number of treatments that directly affect Y, information SCAT does not have access to. Further, screening the treatments with a Markov blanket learner beforehand does not close the gap, as the three MB variants perform on par with or slightly below PT-Exact. SCAT is also the most stable method across data splits, with an average standard deviation over the splits of 0.041, against 0.064 to 0.076 for the other methods.

SCAT also estimates the interventional probability of the selected action most accurately, with the lowest absolute estimation error on average and on 7 of the 13 datasets (Table 2, with per-dataset results in Tables 4–6 in Appendix D). Its selected subgroups cover on average 22.9% of the test units, comparable to PT-Hybrid and the three MB variants, and large enough for their interventional probability to be estimated reliably, as the low estimation error confirms. Finally, SCAT is the fastest method on every dataset, and on average more than 20 times faster than even the fastest competitor.

5.2 One-dimensional treatment variable

Datasets. We use the original ACIC 2016 benchmark (Dorie et al., 2019), the standard semi-synthetic benchmark for a single binary treatment. Since the outcome is continuous in this benchmark, we threshold it at its observed median to obtain a binary Y. We use 30 of the 77 settings, drawn uniformly at random under a fixed seed (Appendix C).

Competitors. We compare against the following state-of-the-art policy learning methods. First, we reuse ListDTR, PolicyTree (PT), and FlowOPT-DR from the multi-dimensional setting. Second, since T is always binary in the original ACIC 2016, we additionally include two policy learning methods

Table 3: Summary over the 30 ACIC 2016 settings: mean over datasets and average rank (1 = best); best per row in bold. †First split only.

		SCAT	ListDTR	PT-Exact	PT-Hybrid	EconML	IOPL†	FlowOPT-DR†
Mean	True $P(Y=1)\uparrow$	**0.943**	0.851	0.813	0.832	0.793	0.716	0.774
	|Estimation error| $\downarrow$	**0.017**	0.027	0.033	0.029	0.034	0.033	0.029
	Test coverage	0.286	0.509	0.470	0.379	0.453	0.825	0.638
	Runtime (s) $\downarrow$	18.8	966	19.3	28.9	**2.8**	1,805	1,855
Rank	True $P(Y=1)$	**1.37**	3.37	3.87	3.17	4.70	6.33	5.20
	|Estimation error|	**2.57**	3.87	4.40	4.13	4.63	4.27	4.13
	Runtime	2.67	5.00	2.83	3.50	**1.00**	6.00	7.00

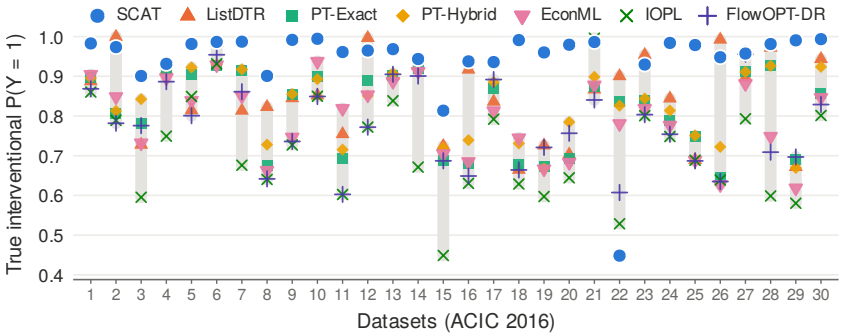


Figure 2: True interventional $P(Y=1)$ of the selected subgroup-level action on the 30 ACIC 2016 settings (indexed as in Table 7); mean over five splits, except IOPL and FlowOPT-DR ($\times$, $+$; first split only). The gray bar spans the competitors' range.

that are restricted to a one-dimensional binary T, namely EconML (Battocchi et al., 2019; Athey & Wager, 2021) and IOPL (Tschernutter et al., 2022). All results are averaged over five random splits, except for the Gurobi-based FlowOPT-DR and IOPL, which are run on the first split only.

Results. Although designed for multiple treatment variables, SCAT also performs best in the more restricted but commonly studied setting of a one-dimensional T. As shown in Figure 2, SCAT selects the subgroup-level causal action with the highest true interventional probability on 24 of the 30 datasets, improving over the best competitor by 0.096 on average on these datasets and by up to 0.29. On five of the remaining six it ranks second, within 0.044 of the best competitor, and ACIC-48 is the only dataset on which it clearly fails. Further, SCAT reaches a true interventional probability of at least 0.95 on 20 datasets, which no competitor achieves on more than 6.

As summarized in Table 3 (per-dataset results in Tables 7–10 in Appendix D), SCAT also has the lowest absolute estimation error on average, and is the most accurate method on 14 of the 30 datasets. Its selected subgroups cover on average 28.6% of the test units, which is large enough for their interventional probability to be estimated reliably, as the low estimation error confirms. Finally, SCAT runs in under 20 seconds on average, on par with PT-Exact and about 50 and 100 times faster than ListDTR and the Gurobi-based IOPL and FlowOPT-DR, respectively. Only EconML, which grows a single greedy policy tree, is faster.

6 Conclusion

We studied subgroup-level causal action design: finding a subgroup, the treatments to intervene on, and their values that maximize the interventional success probability, from observational data with many possibly dependent treatments. Although the effect of such a partial intervention depends on the causal graph among the treatments, we showed that the best action can be read off a partition-based model of $P(Y \mid X, T)$ without the graph, with a finite-sample regret bound. Our instantiation, SCAT, needs no regularization parameter and, on a new multi-treatment benchmark and on ACIC 2016, selects better actions, estimates their success probability more accurately, and is markedly faster than existing policy learning methods.

AI USE STATEMENT

In this work, we used generative AI tools for provide critical ingredients for proving of part the mathematical claims in Lemma A.7, and assist in the writing of proofs. We have *not* used generative AI tools for Generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, interpret results, and assist with translation, clean and reformat dataset, support qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for create Figure 1 (with Tikz plot in LaTeX), debug and edit software code, edit a research paper to improve readability (i.e., grammar check and polish the writing without essentially changing the content). We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness by 2 authors; proof of Lemma A.7 is carefully checked line by line by 1 author. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects or personally identifiable data. All experiments use semi-synthetic benchmarks derived from the publicly available ACIC 2016 data (Appendix E), whose outcomes and treatments are simulated, so no real treatment decisions are affected. The method is intended for decision support in domains such as healthcare and public policy, where recommendations derived from observational data should not be acted upon without expert review. Our theoretical guarantees rest on explicit assumptions, in particular no hidden confounding and treatment positivity (Section 3.1), which need to be assessed for any application. The interpretable form of the recommended actions is meant to facilitate such review. The authors declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

All source code, including the generation of the benchmark datasets, the implementation of SCAT, and the scripts used to run every competitor, is available at `https://anonymous.4open.science/r/SCAT-B3E6/`. The assumptions of all theoretical results are stated in Section 3, and complete proofs are given in Appendix A. Pseudo-code for SCAT is given in Appendix B. The adaptation of the competitors to multiple treatments, the data splits, the evaluation protocol and the metrics are described in Appendix C, per-dataset results for every metric are reported in Appendix D, and the full generating process of the multi-treatment and hand-built datasets, including the stored ground truth used for evaluation, is documented in Appendix E.

REFERENCES

Hussein Almuallim. An efficient algorithm for optimal pruning of decision trees. *Artificial Intelligence*, 83(2):347–362, 1996.

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

Susan Athey and Stefan Wager. Policy learning with observational data. *Econometrica*, 89(1): 133–161, 2021.

Susan Athey, Julie Tibshirani, and Stefan Wager. Generalized random forests. 2019.

Keith Battocchi, Eleanor Dillon, Maggie Hei, Greg Lewis, Paul Oka, Miruna Oprescu, and Vasilis Syrgkanis. EconML: A Python Package for ML-Based Heterogeneous Treatment Effects Estimation. https://github.com/py-why/EconML, 2019. Version 0.x.

Marko Bohanec and Ivan Bratko. Trading accuracy for simplicity in decision trees. *Machine Learning*, 15(3):223–250, 1994.

Vincent Dorie, Jennifer Hill, Uri Shalit, Marc Scott, and Dan Cervone. Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *Statistical Science*, 34(1):43–68, 2019.

Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. *arXiv preprint arXiv:1103.4601*, 2011.

Jared C Foster, Jeremy MG Taylor, and Stephen J Ruberg. Subgroup identification from randomized clinical trial data. *Statistics in medicine*, 30(24):2867–2880, 2011.

Peter Grünwald and Teemu Roos. Minimum description length revisited. *International Journal of Mathematics for Industry*, 11(01):1930001, 2019.

Peter D. Grünwald. *The Minimum Description Length Principle*. MIT Press, 2007.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *Journal of Machine Learning Research*, 13(1): 2409–2464, 2012.

Nathanael Jo, Sina Aghaei, Andrés Gómez, and Phebe Vayanos. Learning optimal prescriptive trees from observational data. *Management Science*, 2026.

Nathan Kallus and Angela Zhou. Policy evaluation and optimization with continuous treatments. In *International conference on artificial intelligence and statistics*, pp. 1243–1251. PMLR, 2018.

Toru Kitagawa and Aleksey Tetenov. Who should be treated? empirical welfare maximization methods for treatment choice. *Econometrica*, 86(2):591–616, 2018.

Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences*, 116(10):4156–4165, 2019.

Himabindu Lakkaraju and Cynthia Rudin. Learning cost-effective and interpretable treatment regimes. In *Artificial intelligence and statistics*, pp. 166–175. PMLR, 2017.

Finnian Lattimore, Tor Lattimore, and Mark D Reid. Causal bandits: Learning good interventions via causal inference. *Advances in neural information processing systems*, 29, 2016.

Sanghack Lee and Elias Bareinboim. Structural causal bandits: Where to intervene? *Advances in neural information processing systems*, 31, 2018.

Ilya Lipkovich, Alex Dmitrienko, Jonathan Denne, and Gregory Enas. Subgroup identification based on differential effect search—a recursive partitioning method for establishing response to treatment in patient subpopulations. *Statistics in medicine*, 30(21):2601–2621, 2011.

Marloes H Maathuis, Markus Kalisch, and Peter Bühlmann. Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37(6A):3133–3164, 2009.

Charles F Manski. Statistical treatment rules for heterogeneous populations. *Econometrica*, 72(4): 1221–1246, 2004.

Myrl G Marmarelis, Fred Morstatter, Aram Galstyan, and Greg Ver Steeg. Policy learning for localized interventions from observational data. In *International Conference on Artificial Intelligence and Statistics*, pp. 4456–4464. PMLR, 2024.

Manish Mehta, Jorma Rissanen, Rakesh Agrawal, et al. MDL-based decision tree pruning. In *KDD*, volume 21, pp. 216–221, 1995.

Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. 2006.

Tommi Mononen and Petri Myllymäki. Computing the multinomial stochastic complexity in sub-linear time. In *Proceedings of the 4th European Workshop on Probabilistic Graphical Models*, pp. 209–216, 2008.

Preetam Nandy, Marloes H Maathuis, and Thomas S Richardson. Estimating the effect of joint interventions from observational data in sparse high-dimensional settings. *The Annals of Statistics*, pp. 647–674, 2017.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.

Emilija Perkovic. Identifying causal effects in maximally oriented partially directed acyclic graphs. In *Conference on Uncertainty in Artificial Intelligence*, pp. 530–539. PMLR, 2020.

Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated DAG! Causal discovery benchmarks may be easy to game. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27772–27784, 2021.

Jorma Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

J. Ross Quinlan and Ronald L. Rivest. Inferring decision trees using the minimum description lenght principle. *Information and Computation*, 80(3):227–248, 1989. ISSN 0890-5401. doi: https://doi.org/10.1016/0890-5401(89)90010-2.

Yurii Mikhailovich Shtarkov. Universal sequential coding of single messages. *Problemy Peredachi Informatsii*, 23(3):3–17, 1987.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. The MIT press, 2001.

Adith Swaminathan, Akshay Krishnamurthy, Alekh Agarwal, Miro Dudik, John Langford, Damien Jose, and Imed Zitouni. Off-policy evaluation for slate recommendation. *Advances in Neural Information Processing Systems*, 30, 2017.

Ioannis Tsamardinos, Constantin F Aliferis, Alexander R Statnikov, and Er Statnikov. Algorithms for large scale markov blanket discovery. In *Flairs*, volume 2, pp. 376–381, 2003.

Daniel Tschernutter, Tobias Hatt, and Stefan Feuerriegel. Interpretable off-policy learning via hyperbox search. In *International Conference on Machine Learning*, pp. 21795–21827. PMLR, 2022.

Thomas Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Proceedings of Sixth Annual Conference on Uncertainty in Artificial Intelligence*, pp. 220–227, 1991.

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

Tong Wang and Cynthia Rudin. Causal rule sets for identifying subgroups with enhanced treatment effects. *INFORMS Journal on Computing*, 34(3):1626–1643, 2022.

Lincen Yang and Matthijs van Leeuwen. Conditional density estimation with histogram trees. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Lincen Yang, Zhong Li, Matthijs van Leeuwen, and Saber Salehkaleybar. Learning subgroups with maximum treatment effects without causal heuristics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27565–27573, 2026.

Yichi Zhang, Eric B Laber, Marie Davidian, and Anastasios A Tsiatis. Interpretable dynamic treatment regimes. *Journal of the American Statistical Association*, 113(524):1541–1549, 2018.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

Zhengyuan Zhou, Susan Athey, and Stefan Wager. Offline multi-action policy learning: Generalization and optimization. *Operations Research*, 71(1):148–183, 2023.

APPENDIX CONTENTS

A Proofs **14**

A.1 Population results 14

A.2 Transferring prediction error to intervention regret 17

A.3 The penalized likelihood criterion 19

A.4 The MDL criterion as a penalized likelihood 19

A.5 A uniform log-loss bound for trees 20

A.6 An oracle inequality for the penalized likelihood tree 22

A.7 Prediction error of the validation estimate 25

A.8 Exact statement and proof of Theorem 3.4 26

B Algorithms **27**

C Experimental setup **29**

C.1 Adapting the competitors to multiple treatments 29

C.2 Evaluation protocol and metrics 29

D Additional experimental results **30**

D.1 Multiple treatment variables 30

D.2 One-dimensional treatment variable 30

E Generation of the simulated datasets **35**

E.1 Covariates 35

E.2 Random function pool 35

E.3 MT's treatment model 36

E.4 MT's outcome model 37

E.5 The ten replicates used in our experiments 38

E.6 Chain, Fork and Collider 39

E.7 Stored ground truth and interventional queries 39

A PROOFS

Throughout the appendix we abbreviate the observational response probability and the propensities by

$$\mu(x,t) := P(Y=1 \mid X=x, T=t), \qquad e_t(x) := P(T=t \mid X=x),$$

both understood up to $P_{X,T}$-null sets, and we write $\mu(z) = \mu(x,t)$ for $z=(x,t)$. An expectation under an intervention refers to a new individual from the population with the subgroup and intervention held fixed, and whenever an action or a tree is learned from data, we fix the realized training data before evaluating it. Appendix A.1 proves the population results, Appendix A.2 bounds the regret of a recommended action by the prediction error of the model used to rank the cells, Appendices A.5 and A.6 bound the prediction error of the penalized likelihood tree, Appendix A.7 bounds that of the validation estimate, and Appendix A.8 combines these into Theorem 3.4.

A.1 POPULATION RESULTS

We start by presenting two lemmas that will be useful in the proofs.

Lemma A.1. *Under the setup in Section 3 and positivity (Equation 1), let $S \subseteq \mathcal{X}$ satisfy $P(X \in S) > 0$. For every $A \subseteq \{1,\dots,m\}$ and $t_A \in \mathcal{T}_A$, the post-intervention law of (X,T) conditional on $X \in S$ is absolutely continuous with respect to $P_{X,T}$. In particular, every property of (x,t) that holds $P_{X,T}$-almost surely also holds almost surely under this post-intervention law.*

Proof. The intervention changes only the structural equations of the treatments in A. Since no covariate is a descendant of a treatment, the law of X remains P_X.

Let $N \subseteq \mathcal{X} \times \mathcal{T}$ be measurable with $P_{X,T}(N) = 0$, and write $N_t = \{x : (x,t) \in N\}$. Since $\mathcal{T}$ is finite,

$$0 = P_{X,T}(N) = \sum_{t \in \mathcal{T}} \int_{N_t} e_t(x)\, P_X(dx). \tag{6}$$

Each summand is nonnegative. By positivity, $P_X(N_t) = 0$ for every t, so $N_X := \bigcup_{t \in \mathcal{T}} N_t$ also has P_X-measure zero. The event $(X,T) \in N$ implies $X \in N_X$, regardless of the treatment assignment. Therefore

$$P\{(X,T) \in N \mid X \in S, do(T_A = t_A)\} \le P(X \in N_X \mid X \in S) = 0.$$

This proves the claimed absolute continuity. □

Lemma A.2 (Conditioning removes the intervention label)**.** *Under the setup in Section 3 and positivity (Equation 1), for every A and t_A, and for the post-intervention distribution of (X, T_{-A})-almost every (x, t_{-A}),*

$$\begin{aligned} &P(Y=1 \mid X=x, T_{-A}=t_{-A}, do(T_A=t_A)) \\ &\quad = P(Y=1 \mid X=x, T_A=t_A, T_{-A}=t_{-A}). \end{aligned} \tag{7}$$

Consequently, for any subgroup S with $P(X \in S) > 0$,

$$\begin{aligned} &P(Y=1 \mid X \in S, do(T_A=t_A)) \\ &= \sum_{t \in \mathcal{T}} \int_S P(Y=1 \mid X=x, T=t) \\ &\qquad \times P(T=t \mid X=x, do(T_A=t_A))\, P_X(dx \mid X \in S). \end{aligned} \tag{8}$$

The first probability inside the integral is observational. The second specifies how treatments are distributed after the intervention. Their roles are different.

Proof. Step 1: apply Rule 2 of do-calculus. If $A = \varnothing$, there is no intervention label to remove and Equation 7 is immediate. Otherwise, Rule 2 permits replacing $do(T_A = t_A)$ by conditioning on $T_A = t_A$ if

$$Y \perp\!\!\!\perp T_A \mid X, T_{-A} \quad \text{in } G_{\underline{T_A}}, \tag{9}$$

where $G_{\underline{T_A}}$ is obtained by deleting every outgoing edge from a node in T_A.

To verify this condition, take any path from a node $T_a \in T_A$ to Y in the modified graph. Since T_a has no outgoing edges there, the path must start with an incoming edge $V \to T_a$. Under our graph restrictions, V is a covariate or another treatment. It cannot be a treatment in T_A, because then $V \to T_a$ would have been deleted. Therefore $V \in X \cup T_{-A}$, so V belongs to the conditioning set. The edge $V \to T_a$ has its tail at V, making V a non-collider on this path: the two path edges cannot both point into V. Conditioning on this non-collider blocks the path. Every such path is therefore blocked, proving Equation 9.

Rule 2 now gives Equation 7. The positivity observation preceding the lemma ensures that observational conditional probabilities are specified, up to irrelevant exceptions, at the configurations reached by the intervention.

Step 2: apply the law of total probability. First average over the untouched treatments and the covariates:

$$\begin{aligned} &P(Y = 1 \mid X \in S, do(T_A = t_A)) \\ &\quad = \sum_{t_{-A} \in \mathcal{T}_{-A}} \int_S P(Y = 1 \mid X = x, T_{-A} = t_{-A}, do(T_A = t_A)) \\ &\qquad\qquad \times P(T_{-A} = t_{-A} \mid X = x, do(T_A = t_A))\, P_X(dx \mid X \in S). \end{aligned}$$

The covariate averaging distribution is observational because treatments do not cause covariates. Conditioning further on $X \in S$ adds no information once $X = x \in S$ is given.

Replace the first probability in the integral using Equation 7. Under the intervention, $T_A = t_A$ with probability one, so

$$\begin{aligned} &P(T_{-A} = t_{-A} \mid X = x, do(T_A = t_A)) \\ &\quad = P(T = (t_A, t_{-A}) \mid X = x, do(T_A = t_A)). \end{aligned}$$

Every full treatment vector inconsistent with t_A has post-intervention probability zero. We may therefore extend the sum to all $t \in \mathcal{T}$, obtaining exactly Equation 8. □

We now turn to the proofs of theorems in the main text.

Proof of Theorem 3.3, part (i). Fix S, A, and t_A as in the statement. The intervention does not change the law of X, so $P(X \in S \mid do(T_A = t_A)) = P(X \in S) > 0$. Define the post-intervention law of (X, T_{-A}) conditional on $X \in S$ by

$$\nu(dx, dt_{-A}) := P(dt_{-A} \mid X = x, do(T_A = t_A))\, P(dx \mid X \in S). \tag{10}$$

This is a probability measure on $S \times \mathcal{T}_{-A}$; integration over the finite treatment domain is a finite sum.

The law of total probability yields

$$\begin{aligned} V(S, A, t_A) = \int_{S \times \mathcal{T}_{-A}} &P(Y = 1 \mid X = x, T_{-A} = t_{-A}, do(T_A = t_A)) \\ &\nu(dx, dt_{-A}). \end{aligned} \tag{11}$$

By Lemma A.2, the integrand equals $\mu(x, t_A, t_{-A})$ for ν-almost every (x, t_{-A}). Thus

$$V(S, A, t_A) = \int_{S \times \mathcal{T}_{-A}} \mu(x, t_A, t_{-A})\, \nu(dx, dt_{-A}). \tag{12}$$

For each cell, set

$$B_k := \{(x, t_{-A}) \in S \times \mathcal{T}_{-A} : (x, t_A, t_{-A}) \in C_k\},$$

and write

$$w_k := \nu(B_k) = P\{(X, T) \in C_k \mid X \in S, do(T_A = t_A)\}.$$

The sets B_k partition $S \times \mathcal{T}_{-A}$, so $w_k \geq 0$ and $\sum_k w_k = 1$. Splitting Equation 12 over this partition gives

$$V(S, A, t_A) = \sum_{k=1}^{M} \int_{B_k} \mu(x, t_A, t_{-A})\, \nu(dx, dt_{-A}). \tag{13}$$

The approximation assumption (Equation 2) holds observationally almost surely. Lemma A.1 ensures that its exceptional set also has probability zero under the intervention, conditional on $X \in S$. Consequently,

$$|\mu(x, t_A, t_{-A}) - \theta_k| \le \gamma \quad \text{for } \nu\text{-almost every } (x, t_{-A}) \in B_k.$$

Applying the upper bound in each integral yields

$$\begin{aligned} V(S, A, t_A) &\le \sum_{k=1}^{M} w_k(\theta_k + \gamma) \\ &= \sum_{k=1}^{M} w_k\theta_k + \gamma \\ &\le \max_k \theta_k + \gamma. \end{aligned} \tag{14}$$

This proves part (i). The unknown treatment graph affects the weights w_k, but the argument uses only that they form a probability distribution. When $\gamma = 0$, Equation 13 gives the exact identity $V(S, A, t_A) = \sum_k w_k\theta_k$. □

Proof of Theorem 3.3(ii). Fix a cell C_k of the form of Equation 3 and a value $t_{A_k} \in Q_k$. Abbreviate $A = A_k$ and $t_A = t_{A_k}$. Since membership in C_k implies $X \in S_k$,

$$P(X \in S_k) \ge P\{(X, T) \in C_k\} > 0.$$

Equation 12, applied with $S = S_k$, gives

$$V(S_k, A, t_A) = \int_{S_k \times \mathcal{T}_{-A}} \mu(x, t_A, t_{-A})\, \nu(dx, dt_{-A}), \tag{15}$$

where ν is defined by Equation 10 with $S = S_k$.

For every $x \in S_k$ and $t_{-A} \in \mathcal{T}_{-A}$, the point (x, t_A, t_{-A}) lies in C_k, because $t_A \in Q_k$ and the cell places no restriction on the untouched treatment coordinates. Thus the post-intervention distribution is concentrated on C_k, regardless of how the untouched treatments respond.

By Equation 2 and Lemma A.1,

$$\theta_k - \gamma \le \mu(x, t_A, t_{-A}) \le \theta_k + \gamma \quad \text{for } \nu\text{-almost every } (x, t_{-A}).$$

Integrating these inequalities against the probability measure ν in Equation 15 gives

$$\theta_k - \gamma \le V(S_k, A, t_A) \le \theta_k + \gamma, \tag{16}$$

which proves the first claim of part (ii).

Finally, choose $k^\star \in \arg\max_k \theta_k$ with $S_{k^\star} \in \mathcal{S}$, and any $t_{A_{k^\star}} \in Q_{k^\star}$. By assumption $A_{k^\star} \in \mathcal{A}$, so $(S_{k^\star}, A_{k^\star}, t_{A_{k^\star}})$ is a subgroup-level action in the sense of Definition 3.1. The lower bound in Equation 16 gives

$$V(S_{k^\star}, A_{k^\star}, t_{A_{k^\star}}) \ge \max_k \theta_k - \gamma. \tag{17}$$

Since part (i) holds for every $A \subseteq \{1, \dots, m\}$, taking the supremum over $S \in \mathcal{S}$, $A \in \mathcal{A}$, and $t_A \in \mathcal{T}_A$ gives $V^\star \le \max_k \theta_k + \gamma$. Subtracting Equation 17 establishes the upper bound in Equation 4, and the lower bound follows because the proposed choice is feasible. When $\gamma = 0$, this choice attains $V^\star$ and is therefore optimal. □

Hyper-box cells are cylinder cells. Every hyper-box cell $C_k = \prod_{\ell=1}^{p+m} I_{k\ell}$ has the form of Equation 3: take $S_k = \prod_{\ell=1}^{p} I_{k\ell}$, let A_k contain exactly the treatment coordinates whose factors are proper subsets of their domains, and set $Q_k = \prod_{j \in A_k} I_{k,p+j}$. Taking A_k to be the coordinates that the cell actually restricts makes it the smallest set for which such a representation exists, so a cell leaving all treatments free has $A_k = \varnothing$. For a partition learned by an axis-aligned decision tree, $A_k \in \mathcal{A}$ is enforced by restricting the treatment coordinates that each root-to-leaf path may split on, e.g., at most s of them, and at least one if $\varnothing \notin \mathcal{A}$.

A.2 TRANSFERRING PREDICTION ERROR TO INTERVENTION REGRET

Actions and prediction error. For a subgroup S with $P(X \in S) > 0$ and a partial intervention $a = do(T_A = t_A)$, we write $V(S, a) := V(S, A, t_A)$, and for a measurable $g : \mathcal{X} \times \mathcal{T} \to [0, 1]$,

$$V_g(S, a) := E[g(X, T) \mid X \in S, a] \tag{18}$$

for the success probability predicted by g, that is, the average of g over S after the intervention. The average squared prediction error of g is

$$R(g) := \int \big[g(x, t) - \mu(x, t)\big]^2 \, P_{X,T}(dx, dt), \tag{19}$$

where the integral is over a fresh observational draw with g held fixed. We also write

$$\varepsilon_g(S) := \sqrt{\frac{R(g)}{\kappa \, P(X \in S)}}, \tag{20}$$

where κ is the quantitative overlap constant of Section 3.3, restated as Equation 28 in Appendix A.3.

We first record that an action read off a cylinder cell keeps every individual in that cell. This is the argument used in the proof of Theorem 3.3(ii), stated separately because it is also needed for the learned tree.

Lemma A.3. *Let $C = \{(x, t) : x \in S_C,\ t_{A_C} \in Q_C\}$ be a cell of the form of Equation 3 with $P\{(X, T) \in C\} > 0$, and let $t_{A_C} \in Q_C$. Then $P(X \in S_C) > 0$ and*

$$P\{(X, T) \in C \mid X \in S_C,\ do(T_{A_C} = t_{A_C})\} = 1.$$

Consequently, if a measurable g equals a constant c on C, then $E[g(X, T) \mid X \in S_C, do(T_{A_C} = t_{A_C})] = c$.

Proof. Since $(X, T) \in C$ implies $X \in S_C$, we have $P(X \in S_C) \geq P\{(X, T) \in C\} > 0$. Conditioning on $X \in S_C$ and setting $T_{A_C} = t_{A_C} \in Q_C$ satisfies both constraints defining C, and C does not restrict the remaining treatments, so $(X, T) \in C$ with probability one whatever their post-intervention distribution. The claim about g follows immediately. □

The next lemma compares the prediction error of a model under a subgroup intervention with its observational prediction error. It does not require the model to recover the population partition.

Lemma A.4. *Fix an action (S, a) with $a = do(T_A = t_A)$, and let $Q_{S,a}$ denote the distribution of (X, T) given $X \in S$ under the intervention a. Suppose that, for some constant $\rho_{S,a}$,*

$$\frac{dQ_{S,a}}{dP_{X,T}} \leq \rho_{S,a} \qquad P_{X,T}\text{-almost surely}. \tag{21}$$

Then, for any measurable $g : \mathcal{X} \times \mathcal{T} \to [0, 1]$,

$$|V(S, a) - V_g(S, a)| \leq \sqrt{\rho_{S,a} \, R(g)}. \tag{22}$$

Under positivity (Equation 1) and quantitative overlap (Equation 28), condition Equation 21 holds with $\rho_{S,a} = 1/\{\kappa P(X \in S)\}$ for every action. Therefore

$$|V(S, a) - V_g(S, a)| \leq \sqrt{\frac{R(g)}{\kappa P(X \in S)}} = \varepsilon_g(S). \tag{23}$$

Proof. Write $h = g - \mu$, so that $R(g) = E_{P_{X,T}}[h^2]$. The idea is to express the error of the predicted success probability as an average of h under $Q_{S,a}$, and then to move this average back to the observational distribution. By Equation 8, the true success probability is the average of μ under $Q_{S,a}$,

$$V(S, a) = \sum_{t \in \mathcal{T}} \int_S \mu(x, t) \, P(T = t \mid X = x, a) \, P_X(dx \mid X \in S) = E_{Q_{S,a}}[\mu],$$

and the predicted one is the same average of g, so that $V_g(S,a) - V(S,a) = E_{Q_{S,a}}[h]$. Jensen's inequality and Equation 21 then give

$$\big(E_{Q_{S,a}}[h]\big)^2 \le E_{Q_{S,a}}[h^2] = E_{P_{X,T}}\Big[h^2\,\frac{dQ_{S,a}}{dP_{X,T}}\Big] \le \rho_{S,a}\,R(g),$$

and taking square roots proves Equation 22.

It remains to bound the density ratio under quantitative overlap. By positivity, $e_t(x) > 0$ for every t and almost every x, and writing $P_{X,T}(dx,t) = e_t(x)\,P_X(dx)$ and $Q_{S,a}(dx,t) = \mathbf{1}\{x \in S\}\,P(T = t \mid X = x, a)\,P_X(dx)/P(X \in S)$, we obtain

$$\frac{dQ_{S,a}}{dP_{X,T}}(x,t) = \frac{\mathbf{1}\{x \in S\}}{P(X \in S)} \cdot \frac{P(T = t \mid X = x, a)}{e_t(x)} \le \frac{1}{\kappa\,P(X \in S)},$$

using $P(T = t \mid X = x, a) \le 1$ in the numerator and quantitative overlap in the denominator. This gives Equation 23. □

Remark A.5. Only Equation 21 enters the proof, and the crude bound $P(T = t \mid X = x, a) \le 1$ is where the unknown post-intervention distribution of T_{-A} is handled. When more is known about this distribution, a smaller $\rho_{S,a}$ can be used, and Proposition A.6 holds with $\varepsilon_g(S)$ replaced by $\sqrt{\rho_{S,a} R(g)}$.

The following proposition is the general form of the guarantee for a learned partition. It applies to any leaf-constant score g used to rank the cells, e.g., the fitted values $\widehat{f}$ or the validation estimate $\bar{f}$.

Proposition A.6. *Assume the setup of Section 3 and positivity (Equation 1), together with the following conditions.*

- *(i)* Overlap. *Quantitative overlap (Equation 28) holds.*
- *(ii)* Reference cell. *Assumption 3.2 holds with a partition satisfying the conditions of Theorem 3.3(ii), and some best cell $k^\star \in \arg\max_k \theta_k$ has $S^\circ := S_{k^\star} \in \mathcal{S}$. Let (S°, a°) be any action read off this cell as in Theorem 3.3(ii).*
- *(iii)* Learned partition. *The realized learned partition consists of hyper-box cells of positive observational probability, whose covariate projections belong to $\mathcal{S}$ and whose restricted treatment coordinates belong to $\mathcal{A}$.*
- *(iv)* Recommendation. *g is constant on the cells of the learned partition, and $(\widehat{S}, \widehat{a})$ is read off a cell $\widehat{L}$ on which g is largest, as in Theorem 3.3(ii).*

Then, for every realized learned partition and score,

$$0 \le V^\star - V(\widehat{S}, \widehat{a}) \le 2\gamma + \varepsilon_g(S^\circ) + \varepsilon_g(\widehat{S}) = 2\gamma + \sqrt{\frac{R(g)}{\kappa}}\left(\frac{1}{\sqrt{P(X \in S^\circ)}} + \frac{1}{\sqrt{P(X \in \widehat{S})}}\right). \tag{24}$$

Proof. We fix the training and validation data, and hence the learned partition, g, the selected cell $\widehat{L}$ and the learned action $(\widehat{S}, \widehat{a})$, so that every step below is deterministic. Write $\widehat{V} = V_g$ and $\varepsilon = \varepsilon_g$. Since the learner compares actions only through g, we move from true to predicted success probabilities at the reference action, compare the two actions under g, and move back at the learned action:

$$\begin{aligned} V^\star &\overset{(A)}{\le} V(S^\circ, a^\circ) + 2\gamma \\ &\overset{(B)}{\le} \widehat{V}(S^\circ, a^\circ) + \varepsilon(S^\circ) + 2\gamma \\ &\overset{(C)}{\le} \widehat{V}(\widehat{S}, \widehat{a}) + \varepsilon(S^\circ) + 2\gamma \\ &\overset{(D)}{\le} V(\widehat{S}, \widehat{a}) + \varepsilon(\widehat{S}) + \varepsilon(S^\circ) + 2\gamma. \end{aligned} \tag{25}$$

Subtracting $V(\widehat{S}, \widehat{a})$ gives the upper bound in Equation 24, so it remains to justify the four links.

Link (A) is the population result: by (ii), Equation 4 of Theorem 3.3(ii) applies to the reference cell and gives $V^\star - V(S^\circ, a^\circ) \le 2\gamma$. Links (B) and (D) are two applications of Lemma A.4 to g, at the reference action and at the learned action, using the two directions of its absolute-value bound; both subgroups have positive probability by Lemma A.3, since the cells they are read off have positive probability.

Link (C) is where the choice of the recommendation enters. Let $\widehat{m}$ be the value of g on $\widehat{L}$, which by (iv) is the largest value of g. Under any action $\widehat{V}$ is an average of $g \le \widehat{m}$, so $\widehat{V}(S^\circ, a^\circ) \le \widehat{m}$, while Lemma A.3 applied to $\widehat{L}$ shows that the learned action keeps every individual in $\widehat{L}$, so that $\widehat{V}(\widehat{S}, \widehat{a}) = \widehat{m}$. This is also the only place where the hyper-box form of the learned partition is needed.

Finally, the regret is nonnegative because $\widehat{S} \in \mathcal{S}$ and $\widehat{A} \in \mathcal{A}$ by (iii), so the learned action is one of the admissible actions in Definition 3.1. □

A.3 The penalized likelihood criterion

We collect the definitions used in Section 3.3. The training sample is $D_n = \{(Z_i, Y_i)\}_{i=1}^n$ with $Z_i = (X_i, T_i)$, and the validation sample D'_{n_v} consists of $n_v \ge 1$ further independent copies. An axis-aligned tree Π has splits of the form $z_j \le c$ and K_Π leaves, and $f_{\Pi,\vartheta}(z) = \vartheta_k$ on leaf k; continuous split thresholds and leaf probabilities are allowed. The candidate set $\mathfrak{P}_n$ is a nonempty set of such trees, which may depend on $Z_1, \dots, Z_n$, e.g., through empirical split quantiles. The negative conditional log-likelihood of (Π, ϑ) on the training sample is

$$\ell_n(\Pi, \vartheta) = -\sum_{i=1}^{n} \left[Y_i \log f_{\Pi,\vartheta}(Z_i) + (1 - Y_i) \log\{1 - f_{\Pi,\vartheta}(Z_i)\} \right], \tag{26}$$

in nats, with the usual conventions at zero and one, and $\widehat{\vartheta}_\Pi$ is the leafwise maximum likelihood estimate: the empirical success proportion in each nonempty leaf and $1/2$ in an empty leaf. For a finite nonnegative penalty $\mathrm{pen}_n(\Pi)$, the fitted tree $\widehat{\Pi}$ satisfies

$$\ell_n(\widehat{\Pi}, \widehat{\vartheta}_{\widehat{\Pi}}) + \mathrm{pen}_n(\widehat{\Pi}) \le \inf_{\Pi \in \mathfrak{P}_n} \{\ell_n(\Pi, \widehat{\vartheta}_\Pi) + \mathrm{pen}_n(\Pi)\} + \xi_n, \tag{27}$$

where $\xi_n \ge 0$ bounds the optimization error of the search; exact minimization gives $\xi_n = 0$. We write $\widehat{f} := f_{\widehat{\Pi}, \widehat{\vartheta}_{\widehat{\Pi}}}$ for the fitted predictor. Quantitative overlap is the condition

$$P(T = t \mid X = x) \ge \kappa > 0 \quad \text{for every } t \in \mathcal{T}, \quad \text{for } P_X\text{-almost every } x. \tag{28}$$

The validation estimate $\bar{f}$ equals, on each leaf of $\widehat{\Pi}$, the success proportion of the validation observations in that leaf, and $1/2$ if there are none. The explicit deviation bound $\Delta_n(K, \delta)$ of Lemma A.7 below is of order

$$\Delta_n(K, \delta) = O\left(\log(n+2) \sqrt{\frac{K \log\{d(n+2)\} + \log\{K/\delta\}}{n}} \right). \tag{29}$$

A.4 The MDL criterion as a penalized likelihood

With n_k observations in leaf k, the Bernoulli NML normalizer is

$$\mathcal{C}_{N,2} = \sum_{s=0}^{N} \binom{N}{s} \left(\frac{s}{N}\right)^s \left(1 - \frac{s}{N}\right)^{N-s}, \qquad N \ge 1, \tag{30}$$

with $0^0 = 1$ and $\mathcal{C}_{0,2} = 1$, and the regret of a tree Π with K_Π leaves is $\sum_{k=1}^{K_\Pi} \log \mathcal{C}_{n_k,2}$. The criterion of Section 4 is therefore Equation 27 with

$$\mathrm{pen}_n(\Pi) = L_{\mathrm{tree}}(\Pi \mid Z^n) + \sum_{k=1}^{K_\Pi} \log \mathcal{C}_{n_k,2}, \tag{31}$$

where $L_{\mathrm{tree}}(\Pi \mid Z^n)$ is the tree code length converted to nats. The NML regret accounts for fitting the leaf probabilities, so no separate code for those probabilities is needed. Since this penalty is nonnegative and the candidate trees are axis-aligned, Theorem A.8, and hence Theorem 3.4, apply to this unsmoothed criterion under exact minimization ($\xi_n = 0$); the greedy search of SCAT enters the bound only through the optimization error ξ_n. For a simple explicit bound, each summand in Equation 30 is a binomial probability and hence at most one, so $0 \le \log \mathcal{C}_{N,2} \le \log(N+1)$ and $\mathrm{pen}_n(\Pi) \le L_{\mathrm{tree}}(\Pi \mid Z^n) + K_\Pi \log(n+1)$.

A.5 A UNIFORM LOG-LOSS BOUND FOR TREES

For $u \in (0,1)$ and $y \in \{0,1\}$, write $\ell_u(y) = -y \log u - (1-y)\log(1-u)$, and for a predictor f define its population and training losses

$$L(f) = E[\ell_{f(Z)}(Y)], \qquad L_n(f) = \frac{1}{n}\sum_{i=1}^{n} \ell_{f(Z_i)}(Y_i),$$

where the expectation in $L(f)$ is over a new observation with f held fixed; the criterion Equation 27 uses the total loss $nL_n(f)$. The population Bernoulli divergence of f measures how far its fitted probabilities are from the true observational probabilities,

$$\mathcal{K}(f) := \int \left[\mu(z)\log\frac{\mu(z)}{f(z)} + \{1-\mu(z)\}\log\frac{1-\mu(z)}{1-f(z)}\right] P_{X,T}(dz). \tag{32}$$

At a fixed z, the expected log loss of f exceeds that of the true probability $\mu(z)$ by exactly the displayed divergence, so averaging over Z gives

$$L(f) - H(Y \mid Z) = \mathcal{K}(f), \tag{33}$$

where $H(Y \mid Z) = -\int \sum_{y\in\{0,1\}} P(Y=y \mid Z=z)\log P(Y=y \mid Z=z)\, P_{X,T}(dz) \in [0, \log 2]$ is the conditional entropy. Since $H(Y \mid Z)$ is the same for every predictor, it cancels whenever population losses are compared.

Because the fitted tree is chosen using the data, we need a deviation bound that holds for all trees at once. Let $\eta_n = (n+2)^{-1}$ and $B_n = -\log \eta_n = \log(n+2)$, and for $K \ge 1$ let $\mathcal{F}_K$ be the class of axis-aligned trees with at most K leaves and arbitrary leaf values in $[\eta_n, 1-\eta_n]$. The leaf values in $\mathcal{F}_K$ are not estimated from data, so a bound that is uniform over $\mathcal{F}_K$ applies to any rule for choosing splits and leaf values, including rules that use the training sample, as long as the values stay in this interval, which also keeps every loss in $[0, B_n]$. The leaf MLE may equal zero or one and is therefore not in $\mathcal{F}_K$; the proof of Theorem A.8 handles this by smoothing. The next lemma gives the required bound over $\mathcal{F}_K$.

Lemma A.7. *Let $J_n = n(n+2)$. For each integer $K \ge 1$, define*

$$G_n(K) := K\{4d(2n+1)\}^{K-1}(J_n+1)^K, \tag{34}$$

$$\Delta_n(K,\delta) := B_n\sqrt{\frac{8}{n}\log\left\{\frac{4K(K+1)G_n(K)}{\delta}\right\}} + \frac{2}{n}. \tag{35}$$

With probability at least $1-\delta$, simultaneously for every $K \ge 1$ and every $f \in \mathcal{F}_K$,

$$|L(f) - L_n(f)| \le \Delta_n(K,\delta). \tag{36}$$

This holds for arbitrary real thresholds and leaf values in $[\eta_n, 1-\eta_n]$, and therefore for any tree in $\mathcal{F}_K$ selected using the sample, including trees from sample-dependent candidate sets.

Proof. We follow the classical Vapnik–Chervonenkis symmetrization argument, with $G_n(K)$ in the role of the growth function. As both thresholds and leaf values range over continua, we first reduce the leaf values to a finite grid and count the predictions that grid-valued trees can make on finitely many points; a second sample and random signs then reduce the problem to a union bound over these finitely many predictions.

1) We first reduce the leaf values to a finite grid. On finitely many points, the predictions of a tree are determined by the leaf each point falls in, which takes only finitely many forms as shown in step 2), and by the values attached to the leaves, which do not. Let

$$\mathcal{G}_n = \{\eta_n + (1-2\eta_n)j/J_n : j = 0, \dots, J_n\}$$

be the $J_n + 1$ equally spaced points from η_n to $1 - \eta_n$, and let $\mathcal{F}_K^{\mathrm{grid}} \subseteq \mathcal{F}_K$ consist of the trees with leaf values in $\mathcal{G}_n$. Rounding the leaf values of any $f \in \mathcal{F}_K$ to $\mathcal{G}_n$, while keeping the splits, gives $f^{\mathrm{grid}} \in \mathcal{F}_K^{\mathrm{grid}}$ with $|f(z) - f^{\mathrm{grid}}(z)| \le 1/J_n$ for every z. The derivative of $u \mapsto \ell_u(y)$ is $-1/u$ or $1/(1-u)$, which is at most $1/\eta_n$ in absolute value on $[\eta_n, 1-\eta_n]$, so the mean value theorem gives

$$|\ell_{f(z)}(y) - \ell_{f^{\mathrm{grid}}(z)}(y)| \le \frac{1}{\eta_n J_n} = \frac{1}{n}. \tag{37}$$

This is why we take $J_n = n/\eta_n$, and the fine grid is cheap, as it enters the count below only through $K \log(J_n + 1)$. Averaging Equation 37 and using the triangle inequality,

$$|L(f) - L_n(f)| \le |L(f^{\mathrm{grid}}) - L_n(f^{\mathrm{grid}})| + \frac{2}{n}, \tag{38}$$

so it suffices to control the deviation over $\mathcal{F}_K^{\mathrm{grid}}$. The grid is only used in the proof; the learner does not round.

2) Next, we count the predictions of grid-valued trees on N fixed points. Consider trees with exactly k leaves, which have $k-1$ internal nodes. There are at most 4^{k-1} ordered tree shapes, a bound on the Catalan number that follows by recording one bit per node in preorder, since the $2k-1$ bits, the first of which is fixed, give at most 2^{2k-2} strings. At each internal node one of d coordinates is chosen, and because the assignment of the N points changes only when the threshold crosses one of their values, there are at most $d(N+1)$ routings per node. Assigning one of the $J_n + 1$ grid values to each leaf and summing over $k \le K$, where each term is at most the one for $k = K$, we obtain

$$\#\{\text{prediction vectors on the } N \text{ points}\} \le K\{4d(N+1)\}^{K-1}(J_n+1)^K. \tag{39}$$

For fixed labels a prediction vector determines the loss vector, so this also bounds the number of loss vectors, and for $N = 2n$, which is the case needed in step 4), the bound equals $G_n(K)$.

3) The population loss is then replaced by the loss on a second sample. Let D_n' be an independent copy of the sample with average loss $L_n'(f)$; it is a device of the proof and requires no extra data. Here and in step 4), suprema are over $\mathcal{F}_K^{\mathrm{grid}}$. For a fixed f, $E[L_n'(f)] = L(f)$, and since each loss lies in $[0, B_n]$, $\mathrm{Var}(L_n'(f)) \le B_n^2/(4n)$ by Popoviciu's inequality $\mathrm{Var}(W) = E[(W - B_n/2)^2] - (EW - B_n/2)^2 \le B_n^2/4$. Chebyshev's inequality gives

$$P\{|L_n'(f) - L(f)| > t/2\} \le \frac{B_n^2/(4n)}{t^2/4} = \frac{B_n^2}{nt^2}, \tag{40}$$

which is at most $1/2$ when $nt^2 \ge 2B_n^2$. Now suppose that some f has $|L(f) - L_n(f)| > t$, and choose such an f using D_n only. Given D_n, this tree is fixed with respect to D_n', so with conditional probability at least $1/2$ we have $|L_n'(f) - L(f)| \le t/2$ and hence $|L_n'(f) - L_n(f)| > t/2$. Therefore

$$P\{\sup_f |L(f) - L_n(f)| > t\} \le 2P\{\sup_f |L_n'(f) - L_n(f)| > t/2\}, \tag{41}$$

where the right-hand side depends on each tree only through its predictions at the $2n$ observed points.

4) Random signs now turn the two-sample deviation into a finite union bound. Pair observation i of D_n with observation i of D_n', and exchange the two members of each pair independently with probability $1/2$; equivalently, independent random signs ε_i decide which member of pair i plays the training role. Because the $2n$ observations are i.i.d., these exchanges leave the joint distribution of the two samples unchanged. With

$$a_i(f) = \ell_{f(Z_i')}(Y_i') - \ell_{f(Z_i)}(Y_i), \qquad |a_i(f)| \le B_n,$$

the difference of empirical losses after the exchanges is $S(f) = n^{-1}\sum_{i=1}^n \varepsilon_i a_i(f)$, so that

$$P\Big\{\sup_f |L_n'(f) - L_n(f)| > t/2\Big\} = P\Big\{\sup_f |S(f)| > t/2\Big\}, \tag{42}$$

where the probability on the right is over both the data and the signs.

We now condition on the $2n$ observations, so that only the signs are random and each $a_i(f)$ is a fixed number; write P_ε for probability over the signs. For a single tree, the summands $\varepsilon_i a_i(f)$ are independent with mean zero and range $2B_n$, and Hoeffding's inequality gives, for $u > 0$,

$$P_\varepsilon\left\{\left|\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i a_i(f)\right| > u\right\} \le 2\exp\left\{-\frac{2n^2u^2}{n(2B_n)^2}\right\} = 2\exp\left\{-\frac{nu^2}{2B_n^2}\right\}. \tag{43}$$

The event $\{\sup_f |S(f)| > u\}$ is the union of the events $\{|S(f)| > u\}$ over all trees, and given the observations, $S(f)$ depends on f only through its loss vector at the $2n$ points. Trees with the same loss vector thus define the same event, and by step 2) there are at most $G_n(K)$ distinct ones, so a union bound with $u = t/2$ gives

$$P_\varepsilon\Big\{\sup_f |S(f)| > t/2 \;\Big|\; \text{observations}\Big\} \le 2G_n(K)\exp\left\{-\frac{nt^2}{8B_n^2}\right\}. \tag{44}$$

The conditioning is needed here, as which trees share a loss vector depends on where the observations fall. Since the right-hand side is the same for every realization of the observations, the tower property shows that it also bounds the unconditional probability in Equation 42, and together with Equation 41,

$$P\{\sup_f |L(f) - L_n(f)| > t\} \le 4G_n(K)\exp\left\{-\frac{nt^2}{8B_n^2}\right\}. \tag{45}$$

5) Finally, we choose the deviation level for every tree size. As the size of the fitted tree is not known in advance, we spread the failure probability over sizes by assigning $\delta_K = \delta/[K(K+1)]$ to size K; these sum to δ because $1/[K(K+1)] = 1/K - 1/(K+1)$ telescopes. The choice

$$t_K = B_n\sqrt{\frac{8}{n}\log\{4G_n(K)/\delta_K\}}$$

satisfies $nt_K^2 \ge 2B_n^2$, as required in step 3), and makes the right-hand side of Equation 45 equal to δ_K. Hence, with probability at least $1-\delta$, every $f \in \mathcal{F}_K^{\mathrm{grid}}$ satisfies $|L(f) - L_n(f)| \le t_K$ for every K, and by Equation 38 every $f \in \mathcal{F}_K$ satisfies

$$|L(f) - L_n(f)| \le t_K + \frac{2}{n} = \Delta_n(K,\delta).$$

Since this event covers every tree in $\mathcal{F}_K$, it also covers a tree chosen after observing the data.

Technical detail about suprema. The selections and suprema above can be taken over a countable class. Approximating each real threshold from above by rational thresholds reproduces every split decision in the limit, including at points equal to the threshold, so predictions converge pointwise and, as the losses are bounded by B_n, so do their expectations. Rational-threshold trees therefore give the same suprema as arbitrary real thresholds, and the tree in step 3) can be chosen as the first one in an enumeration whose deviation exceeds t. This imposes no restriction on the learner's thresholds. □

A.6 An oracle inequality for the penalized likelihood tree

Set $\eta_n = (n+2)^{-1}$ and $b_n = -\log(1-2\eta_n) + 2\eta_n^2$.

Theorem A.8. *For every $0 < \delta < 1$ and $n \ge 2$, with probability at least $1-\delta$ over D_n, the estimator satisfying Equation 27 obeys*

$$R(\widehat{f}) \le \mathcal{B}_n(\delta) := \inf_{\substack{\Pi\in\mathfrak{P}_n \\ \vartheta\in[\eta_n,1-\eta_n]^{K_\Pi}}} \left\{\mathcal{K}(f_{\Pi,\vartheta}) + \frac{\mathrm{pen}_n(\Pi)}{n} + \Delta_n(K_\Pi,\delta)\right\} + \Delta_n(\widehat{K},\delta) + \frac{\xi_n}{n} + b_n. \tag{46}$$

The bound is an oracle inequality: the infimum is over comparison trees whose leaf probabilities may be chosen with knowledge of the truth, and each such tree contributes its approximation error $\mathcal{K}(f_{\Pi,\vartheta})$, its penalty and its deviation term. The remaining terms account for the size of the selected

tree, the optimization error and a smoothing step used only in the proof, with $b_n \le 2/n + 2\eta_n^2$. Since $\widehat{K}$ is random, so is $\mathcal{B}_n(\delta)$, but the statement holds on a single event of probability at least $1-\delta$. The bound is uniform over tree thresholds and therefore applies when the partition is learned on the same sample used to fit the leaf probabilities. The restriction on ϑ is only for comparison in the proof: the fitted MLEs may equal zero or one, and no clipping or discretization is added to the algorithm. A nonnegative penalty alone does not ensure a small bound: the penalty of a useful candidate, the selected tree size, and the optimization error must also be controlled, and a weak penalty that allows an excessively large selected tree can make the bound uninformative.

Proof. The proof mirrors that of Proposition A.6. The learner compares trees by penalized training loss, so we move from population to training loss at the selected tree, compare it with an arbitrary comparison tree as the learner did, and move back to population loss at the comparison tree, with Lemma A.7 paying for each move. Since the lemma only covers probabilities in $[\eta_n, 1-\eta_n]$ while the leaf MLE may equal zero or one, the fitted tree is smoothed slightly and the comparison tree is given fixed probabilities in this interval. The excess log loss is finally converted into squared prediction error.

1) We first fix the event on which the losses of all trees are controlled. By Lemma A.7, there is an event $\mathcal{E}$ with $P(\mathcal{E}) \ge 1-\delta$ on which

$$|L(f) - L_n(f)| \le \Delta_n(K,\delta) \qquad \text{for every } K \ge 1 \text{ and every } f \in \mathcal{F}_K.$$

We work on $\mathcal{E}$ from now on, so the sample, the candidate set and the fitted tree are fixed, and since $\mathcal{E}$ does not depend on the tree under consideration, the comparison tree below can be chosen freely at no further cost in probability. Write $\eta = \eta_n$ and $c_\eta = -\log(1-2\eta)$; since $n \ge 2$, we have $0 < \eta \le 1/4$ and $c_\eta = \log\{(n+2)/n\} \le 2/n$.

2) The fitted tree is then smoothed so that the lemma applies to it. The predictor

$$\widetilde{f}(z) = (1-2\eta)\widehat{f}(z) + \eta$$

shrinks $\widehat{f}$ slightly toward $1/2$ on the same tree, and satisfies

$$\eta \le \widetilde{f}(z) \le 1-\eta, \qquad |\widetilde{f}(z) - \widehat{f}(z)| = \eta|1 - 2\widehat{f}(z)| \le \eta,$$

so that $\widetilde{f} \in \mathcal{F}_{\widehat{K}}$. Both of its predicted probabilities are at least $1-2\eta$ times those of $\widehat{f}$,

$$\widetilde{f}(z) \ge (1-2\eta)\widehat{f}(z), \qquad 1-\widetilde{f}(z) \ge (1-2\eta)(1-\widehat{f}(z)),$$

so taking negative logarithms shows that every loss increases by at most c_η,

$$\ell_{\widetilde{f}(z)}(y) \le \ell_{\widehat{f}(z)}(y) - \log(1-2\eta) = \ell_{\widehat{f}(z)}(y) + c_\eta,$$

and averaging over the training observations gives

$$L_n(\widetilde{f}) \le L_n(\widehat{f}) + c_\eta. \tag{47}$$

This comparison is finite even if $\widehat{f}$ takes the values zero or one, because the leaf MLE predicts zero only in a leaf where every observed label is zero, and one only where every label is one, while empty leaves contribute nothing. The smoothed tree is only used in the analysis; the estimator and its action are unchanged.

3) The optimization criterion now compares the fitted tree with any comparison tree. Take any $\Pi \in \mathfrak{P}_n$ and $\vartheta \in [\eta, 1-\eta]^{K_\Pi}$, and let $f = f_{\Pi,\vartheta}$. We use these fixed probabilities instead of the candidate's own MLE, which may equal zero or one, so that $f \in \mathcal{F}_{K_\Pi}$ and $\mathcal{K}(f)$ is finite. This costs nothing, since the MLE has the smaller training loss, $\ell_n(\Pi, \widehat{\vartheta}_\Pi) \le \ell_n(\Pi,\vartheta) = nL_n(f)$, and combined with Equation 27 it gives

$$\begin{aligned} nL_n(\widehat{f}) + \mathrm{pen}_n(\widehat{\Pi}) &\le \ell_n(\Pi, \widehat{\vartheta}_\Pi) + \mathrm{pen}_n(\Pi) + \xi_n \\ &\le nL_n(f) + \mathrm{pen}_n(\Pi) + \xi_n. \end{aligned}$$

Dividing by n,

$$L_n(\widehat{f}) \le L_n(f) + \frac{\mathrm{pen}_n(\Pi) - \mathrm{pen}_n(\widehat{\Pi}) + \xi_n}{n}. \tag{48}$$

This is the only place where the fitting procedure enters.

4) Both training losses are then replaced by population losses. Applying $\mathcal{E}$ at the selected tree, then steps 2) and 3), and finally $\mathcal{E}$ at the comparison tree,

$$\begin{aligned} L(\widetilde{f}) &\le L_n(\widetilde{f}) + \Delta_n(\widehat{K}, \delta) \\ &\le L_n(\widehat{f}) + c_\eta + \Delta_n(\widehat{K}, \delta) \\ &\le L_n(f) + \frac{\mathrm{pen}_n(\Pi) - \mathrm{pen}_n(\widehat{\Pi}) + \xi_n}{n} + c_\eta + \Delta_n(\widehat{K}, \delta) \\ &\le L(f) + \frac{\mathrm{pen}_n(\Pi) - \mathrm{pen}_n(\widehat{\Pi}) + \xi_n}{n} + c_\eta + \Delta_n(\widehat{K}, \delta) + \Delta_n(K_\Pi, \delta), \end{aligned}$$

which is where the two deviation terms, one for each tree, come from. Subtracting $H(Y \mid Z)$ from both sides and using Equation 33 turns the excess losses into divergences,

$$\begin{aligned} \mathcal{K}(\widetilde{f}) \le \mathcal{K}(f) &+ \frac{\mathrm{pen}_n(\Pi) - \mathrm{pen}_n(\widehat{\Pi}) + \xi_n}{n} \\ &+ c_\eta + \Delta_n(\widehat{K}, \delta) + \Delta_n(K_\Pi, \delta). \end{aligned} \tag{49}$$

5) It remains to convert divergence into squared prediction error. For $u \in [0,1]$ and $v \in (0,1)$, Pinsker's inequality for Bernoulli distributions states that

$$u \log \frac{u}{v} + (1-u) \log \frac{1-u}{1-v} \ge 2(u-v)^2. \tag{50}$$

For completeness, if $H(u)$ denotes the left-hand side minus the right-hand side for fixed v, then $H(v) = H'(v) = 0$ and $H''(u) = 1/\{u(1-u)\} - 4 \ge 0$ on $(0,1)$, so $H \ge 0$, and continuity covers $u \in \{0,1\}$. Applying Equation 50 with $u = \mu(z)$ and $v = \widetilde{f}(z)$ and averaging over Z gives $2R(\widetilde{f}) \le \mathcal{K}(\widetilde{f})$. To return to $\widehat{f}$, we use $(a+b)^2 \le 2a^2 + 2b^2$ with $a = \widetilde{f}(z) - \mu(z)$ and $b = \widehat{f}(z) - \widetilde{f}(z)$, where $|b| \le \eta$ by step 2), and average again:

$$R(\widehat{f}) \le 2R(\widetilde{f}) + 2\eta^2 \le \mathcal{K}(\widetilde{f}) + 2\eta^2. \tag{51}$$

6) Collecting the terms gives the oracle bound. Substituting Equation 49 into Equation 51 and dropping $-\mathrm{pen}_n(\widehat{\Pi})/n \le 0$,

$$\begin{aligned} R(\widehat{f}) \le \mathcal{K}(f) &+ \frac{\mathrm{pen}_n(\Pi)}{n} + \Delta_n(K_\Pi, \delta) \\ &+ \Delta_n(\widehat{K}, \delta) + \frac{\xi_n}{n} + c_\eta + 2\eta^2. \end{aligned}$$

This holds on $\mathcal{E}$ for every comparison tree and probability vector, so we may take the infimum, and since $c_\eta + 2\eta^2 = b_n$, the right-hand side becomes $\mathcal{B}_n(\delta)$. Note that the proof uses neither the causal assumptions nor treatment overlap. □

Corollary A.9. *Suppose the candidate set contains a tree Π° with K° leaves that represents μ exactly for $P_{X,T}$-almost every (x,t). On the event of Theorem A.8,*

$$\begin{aligned} R(\widehat{f}) &\le \frac{\mathrm{pen}_n(\Pi^\circ) + \xi_n}{n} + \Delta_n(K^\circ, \delta) + \Delta_n(\widehat{K}, \delta) - 2\log(1 - 2\eta_n) + 2\eta_n^2 \\ &\le \frac{\mathrm{pen}_n(\Pi^\circ) + \xi_n}{n} + \Delta_n(K^\circ, \delta) + \Delta_n(\widehat{K}, \delta) + \frac{5}{n}. \end{aligned} \tag{52}$$

Consequently, if K° and $\widehat{K}$ are bounded, d and δ are fixed, and $\mathrm{pen}_n(\Pi^\circ) + \xi_n = O(\log n)$, then $R(\widehat{f}) = O\big((\log n)^{3/2} n^{-1/2}\big)$.

Proof. We evaluate the infimum in Equation 46 at the exact tree Π°. Its true leaf probabilities θ_k° may equal zero or one, so we smooth them as in the preceding proof, setting

$$\vartheta_k^\circ = (1-2\eta)\theta_k^\circ + \eta \in [\eta, 1-\eta], \qquad f_\eta^\circ = f_{\Pi^\circ,\vartheta^\circ},$$

with $\eta = \eta_n$, so that $f_\eta^\circ(z) = (1-2\eta)\mu(z) + \eta$ for almost every z. As before, each predicted probability of f_η° is at least $1-2\eta$ times the true one, so every log ratio in the divergence is at most $-\log(1-2\eta)$, and since the true label probabilities sum to one, with zero-probability labels contributing zero,

$$\mathcal{K}(f_\eta^\circ) \le -\log(1-2\eta).$$

Inserting this comparison tree into Equation 46 gives

$$\begin{aligned}\mathcal{B}_n(\delta) \le \underbrace{-\log(1-2\eta)}_{\ge \mathcal{K}(f_\eta^\circ)} + \frac{\mathrm{pen}_n(\Pi^\circ)}{n} + \Delta_n(K^\circ, \delta) \\ + \Delta_n(\widehat{K}, \delta) + \frac{\xi_n}{n} + \underbrace{\left\{-\log(1-2\eta) + 2\eta^2\right\}}_{b_n},\end{aligned}$$

where the first logarithmic term is the smoothing cost of the comparison tree and the second that of the fitted tree. Combining them and using $R(\widehat{f}) \le \mathcal{B}_n(\delta)$ gives the first inequality in Equation 52. For the second, $-\log(1-2\eta_n) = \log(1+2/n) \le 2/n$ and $2\eta_n^2 \le 2/n^2 \le 1/n$ for $n \ge 2$.

For the rate, note that for fixed d and bounded K every factor of $G_n(K)$ is polynomial in n,

$$\log G_n(K) \le \log K + (K-1)\log\{4d(2n+1)\} + K\log\{n(n+2)+1\} = O(\log n),$$

so for fixed δ the logarithm in Equation 35 is $O(\log n)$, and together with $B_n = O(\log n)$ both deviation terms satisfy

$$\Delta_n(K, \delta) = O\left((\log n)\sqrt{\frac{\log n}{n}}\right) = O\left(\frac{(\log n)^{3/2}}{\sqrt{n}}\right).$$

The remaining terms are $O(\log n/n)$ by assumption, so the right-hand side of Equation 52 is $O((\log n)^{3/2}/\sqrt{n})$. □

A.7 Prediction error of the validation estimate

Lemma A.10. *Fix the training data, and hence the fitted tree $\widehat{\Pi}$ with leaves $L_1, \dots, L_{\widehat{K}}$ of positive observational probability. Let $\bar{f}$ be the validation estimate, i.e., the success proportion of the n_v independent validation observations falling in each leaf, with $\bar{f} = 1/2$ in an empty leaf. Then, with probability at least $1-\delta$ over the validation sample,*

$$R(\bar{f}) \le R(\widehat{f}) + \frac{\widehat{K}}{2\delta\, n_v}. \tag{53}$$

Proof. Write $p_k = P\{(X,T) \in L_k\} > 0$ and $\bar{\theta}_k = P\{Y=1 \mid (X,T) \in L_k\} = E[\mu(X,T) \mid (X,T) \in L_k]$ for the response of the learned leaves, and let $f_{\bar{\theta}}$ be the predictor equal to $\bar{\theta}_k$ on L_k. For any predictor g that equals a constant c_k on L_k, expanding the square around $\bar{\theta}_k$ and using $E[\mu(X,T) - \bar{\theta}_k \mid (X,T) \in L_k] = 0$ gives

$$R(g) = \sum_{k=1}^{\widehat{K}} p_k\,(c_k - \bar{\theta}_k)^2 + R(f_{\bar{\theta}}).$$

Applied to $g = \widehat{f}$ this shows $R(f_{\bar{\theta}}) \le R(\widehat{f})$, and applied to $g = \bar{f}$,

$$R(\bar{f}) \le R(\widehat{f}) + \sum_{k=1}^{\widehat{K}} p_k\,(\bar{f}_k - \bar{\theta}_k)^2, \tag{54}$$

where $\bar{f}_k$ is the validation estimate on L_k. It remains to bound the expectation of the sum over the validation sample. Let $N_k \sim \mathrm{Binomial}(n_v, p_k)$ be the number of validation observations in L_k. Given $N_k = N \geq 1$, their labels are independent Bernoulli$(\bar{\theta}_k)$, because each such observation has (X, T) distributed as $P_{X,T}$ conditional on L_k; hence $E[(\bar{f}_k - \bar{\theta}_k)^2 \mid N_k = N] = \bar{\theta}_k(1-\bar{\theta}_k)/N \leq 1/(4N)$, while for $N = 0$ the error is $(1/2 - \bar{\theta}_k)^2 \leq 1/4$. In both cases the conditional expectation is at most $1/(2(N_k+1))$, and the binomial identity

$$E\Big[\frac{1}{N_k+1}\Big] = \sum_{j=0}^{n_v} \binom{n_v}{j} \frac{p_k^j (1-p_k)^{n_v-j}}{j+1} = \frac{1-(1-p_k)^{n_v+1}}{(n_v+1)p_k} \leq \frac{1}{(n_v+1)p_k}$$

gives $E[p_k(\bar{f}_k - \bar{\theta}_k)^2] \leq 1/(2(n_v+1))$. Summing over the leaves, the expectation of the sum in Equation 54 is at most $\widehat{K}/(2(n_v+1))$, and Markov's inequality bounds the sum by $\widehat{K}/(2\delta(n_v+1)) \leq \widehat{K}/(2\delta n_v)$ with probability at least $1-\delta$. □

A.8 Exact statement and proof of Theorem 3.4

Theorem A.11. *Assume the setup of Section 3.3 and quantitative overlap (Equation 28), and that the leaves of every candidate tree in $\mathfrak{P}_n$ have positive observational probability, covariate projections in $\mathcal{S}$, and restricted treatment coordinates in $\mathcal{A}$. Suppose that $\mathfrak{P}_n$ contains a tree Π° with K° leaves that represents $P(Y=1 \mid X=x, T=t)$ exactly for $P_{X,T}$-almost every (x,t), and let S° be the covariate projection of a leaf of Π° with the largest response. Then, for every $0 < \delta < 1$, with probability at least $1-2\delta$ over the training and validation samples, the recommended action satisfies*

$$0 \leq V^\star - V(\widehat{S}, \widehat{A}, \widehat{t}_{\widehat{A}}) \leq \sqrt{\frac{\mathcal{R}_n(\delta)}{\kappa}} \left(\frac{1}{\sqrt{P(X \in S^\circ)}} + \frac{1}{\sqrt{P(X \in \widehat{S})}} \right), \tag{55}$$

$$\mathcal{R}_n(\delta) := \frac{\mathrm{pen}_n(\Pi^\circ) + \xi_n}{n} + \Delta_n(K^\circ, \delta) + \Delta_n(\widehat{K}, \delta) + \frac{5}{n} + \frac{\widehat{K}}{2\delta\, n_v}, \tag{56}$$

with Δ_n as in Lemma A.7. In particular, if K° and $\widehat{K}$ are bounded, d and δ are fixed, $\mathrm{pen}_n(\Pi^\circ) + \xi_n = O(\log n)$, $n_v \asymp n$, and κ, $P(X \in S^\circ)$ and $P(X \in \widehat{S})$ are bounded below by positive constants, then the regret is $O((\log n)^{3/4} n^{-1/4})$.

For exact minimization, $\xi_n = 0$, and the informal Theorem 3.4 follows by inserting the order of Δ_n in Equation 29. For approximate minimization, the general bound retains the additional term ξ_n/n.

Proof. *Prediction error of the validation estimate.* By Corollary A.9, there is an event $\mathcal{E}$ of the training sample with $P(\mathcal{E}) \geq 1-\delta$ on which $R(\widehat{f}) \leq \{\mathrm{pen}_n(\Pi^\circ) + \xi_n\}/n + \Delta_n(K^\circ, \delta) + \Delta_n(\widehat{K}, \delta) + 5/n$. By Lemma A.10, for every realization of the training sample there is an event $\mathcal{E}'$ of the validation sample with conditional probability at least $1-\delta$ on which $R(\bar{f}) \leq R(\widehat{f}) + \widehat{K}/(2\delta n_v)$; hence $P(\mathcal{E}') \geq 1-\delta$ as well. On $\mathcal{E} \cap \mathcal{E}'$, which has probability at least $1-2\delta$,

$$R(\bar{f}) \leq \mathcal{R}_n(\delta). \tag{57}$$

From prediction error to regret. We apply Proposition A.6 with the score $g = \bar{f}$ and with the population partition taken to be the leaf partition of Π°. Since Π° represents μ exactly, the response of each leaf equals the constant value of μ on it, so Assumption 3.2 holds with $\gamma = 0$. The leaves of Π° are hyper-boxes, hence of the form of Equation 3, and because $\Pi^\circ \in \mathfrak{P}_n$ they have positive probability, covariate projections in $\mathcal{S}$, and restricted treatment coordinates in $\mathcal{A}$; in particular a best leaf has covariate projection $S^\circ \in \mathcal{S}$, which is condition (ii). Condition (iii) holds for $\widehat{\Pi} \in \mathfrak{P}_n$ for the same reason, (i) is quantitative overlap, which also implies positivity (Equation 1), and (iv) is the validation-based ranking. Proposition A.6 therefore gives, for every realization of the data,

$$0 \leq V^\star - V(\widehat{S}, \widehat{A}, \widehat{t}_{\widehat{A}}) \leq \sqrt{\frac{R(\bar{f})}{\kappa}} \left(\frac{1}{\sqrt{P(X \in S^\circ)}} + \frac{1}{\sqrt{P(X \in \widehat{S})}} \right),$$

and inserting Equation 57 proves Equation 55.

Rate. Under the stated conditions, Corollary A.9 gives $\{\mathrm{pen}_n(\Pi^\circ) + \xi_n\}/n + \Delta_n(K^\circ, \delta) + \Delta_n(\widehat{K}, \delta) + 5/n = O((\log n)^{3/2} n^{-1/2})$, and $\widehat{K}/(2\delta n_v) = O(1/n)$ when $n_v \asymp n$, so $\mathcal{R}_n(\delta) = O((\log n)^{3/2} n^{-1/2})$. With κ, $P(X \in S^\circ)$ and $P(X \in \widehat{S})$ bounded below, the regret bound is a constant times $\sqrt{\mathcal{R}_n(\delta)} = O((\log n)^{3/4} n^{-1/4})$. □

B ALGORITHMS

Notation. The data are $D = \{(x_i, t_i, y_i)\}_{i=1}^n$ with $x_i \in \mathbb{R}^p$ (categorical covariates are one-hot encoded), $t_i \in \{0,1\}^m$ and $y_i \in \{0,1\}$. We write $z_i = (x_i, t_i)$ for the $d = p + m$ variables that the tree can split on. The tree treats covariates and treatments alike, and nothing in its growth or pruning knows which variables are treatments. For a tree $\mathcal{T}$, we write $\Lambda(\mathcal{T})$ for its leaves, $\mathcal{V}(\mathcal{T})$ for its internal nodes, D_v for the instances that reach node v, and $\mathrm{path}(\ell)$ for the root-to-leaf path of ℓ. The learning criterion of Section 4 is

$$\begin{aligned} \mathrm{CL}(\mathcal{T}, D) \;=\; & \underbrace{L_{\mathbb{N}}\big(|\Lambda(\mathcal{T})|\big) + \log_2 \mathrm{Cat}_{|\Lambda(\mathcal{T})|-1}}_{\text{number and shape of the leaves}} \;+ \sum_{v \in \mathcal{V}(\mathcal{T})} \underbrace{\big(\log_2 d + \mathrm{CL}_{\mathrm{cut}}(v)\big)}_{\text{which variable, which value}} \\ & + \sum_{\ell \in \Lambda(\mathcal{T})} \underbrace{\mathrm{CL}_{\mathrm{data}}(D_\ell)}_{\text{the labels in the leaf}} , \end{aligned} \tag{58}$$

where $L_{\mathbb{N}}$ is Rissanen's code for the positive integers, Cat_k is the k-th Catalan number, $\mathrm{CL}_{\mathrm{data}}(D_\ell)$ is the NML code length of the labels in D_ℓ, i.e., their negative log-likelihood plus the regret, and $\mathrm{CL}_{\mathrm{cut}}(v)$ is the cost of naming the split value of v among the candidates at its resolution. Only the last sum is additive over subtrees. The first two terms depend on the *total* number of leaves, which is why greedy bottom-up pruning would not return the minimizer, and why Algorithm 2 carries the number of leaves through the recursion. We write $\mathcal{T}_{v \leftarrow (j,c)}$ for the tree obtained from $\mathcal{T}$ by splitting leaf v on variable j at value c.

Candidate splits. A binary variable, i.e., every treatment and every one-hot covariate, has the single candidate $c = \frac{1}{2}$, which costs no bits to name. A numeric variable is discretized into nested quantile grids of $5 \cdot 2^r$ cut points for $r = 0, \ldots, 7$. The resolution r is itself encoded, so a finer grid has to pay for its precision. The search on line 8 of Algorithm 1 scans the resolutions from coarse to fine and stops at the first one that does not improve on the best split found so far, so it returns the best split over the resolutions examined rather than over every candidate value. A candidate is admissible only if both children receive at least three instances. This is a numerical safeguard rather than a tuned hyperparameter, and it is the only constraint on the size of a leaf, as there is no maximum depth and no other minimum.

Algorithm 1 GROWFULL: growing the tree to purity

Require: node v with instances D_v
Initially: $\mathcal{T}$ is a single leaf holding all of D, and GROWFULL is called on it.
1: **if** all y_i in D_v are equal **then** ▷ pure, nothing left to encode
2: **return**
3: **end if**
4: $\mathcal{H}(v) \leftarrow$ the admissible candidates (j, c) at v ▷ both children have ≥ 3 instances
5: **if** $\mathcal{H}(v) = \varnothing$ **then**
6: **return**
7: **end if**
8: $(j^*, c^*) \leftarrow \arg\min_{(j,c) \in \mathcal{H}(v)} \mathrm{CL}\big(\mathcal{T}_{v \leftarrow (j,c)}, D\big)$ ▷ resolutions scanned from coarse to fine
9: split v on (j^*, c^*) into children v_{L} and v_{R}, and store $\mathrm{CL}_{\mathrm{cut}}(v)$
10: GROWFULL(v_{L}) and GROWFULL(v_{R})

Growth is not stopped when a split fails to lower the criterion. Unlike CDTree, which stops as soon as no split lowers CL, every node is split until it is pure or has no admissible split, and the decision of what to keep is deferred to Algorithm 2. This removes the horizon effect, in which a split that does not pay for itself is never made, although it would enable later splits that do.

Algorithm 2 PRUNE: the subtree with minimum CL, found exactly

Require: the fully grown tree $\mathcal{T}$ from Algorithm 1
Ensure: the subtree of $\mathcal{T}$ with the smallest $\mathrm{CL}(\cdot, D)$

$\Phi_v(k)$ is the smallest achievable value of the two additive sums in Equation 58 over the subtree rooted at v, given that it keeps exactly k leaves.

for each node v of $\mathcal{T}$ in post-order **do**
  $\Phi_v(1) \leftarrow \mathrm{CL}_{\mathrm{data}}(D_v)$ ▷ collapse v into a leaf
  if v is internal with children a and b **then**
    $\Phi_v(k) \leftarrow \min_{\substack{i+j=k \\ i,j\geq 1}} \Big[\Phi_a(i) + \Phi_b(j)\Big] + \log_2 d + \mathrm{CL}_{\mathrm{cut}}(v)$ for $k = 2, \ldots, |\Lambda(\mathcal{T}_v)|$
  end if
end for
$k^* \leftarrow \arg\min_k \ \Phi_{\mathrm{root}}(k) + L_{\mathbb{N}}(k) + \log_2 \mathrm{Cat}_{k-1}$ ▷ the non-additive terms, added once
return the subtree attaining $\Phi_{\mathrm{root}}(k^*)$, recovered from the minimizing pairs (i, j)

Because the structure code depends on the total number of leaves, the usual bottom-up comparison of keeping a subtree against collapsing it is not exact. Indexing the recursion by the number of retained leaves makes the remaining terms additive, and yields the minimizer exactly. This is the optimal pruning algorithm of Bohanec & Bratko (1994), with the faster variant of Almuallim (1996), applied to the MDL criterion in Equation 58.

Algorithm 3 SCAT: learning the subgroup-level causal action

Require: training data D_{tr}, validation data D_{val}
Ensure: a subgroup-level action (subgroup, intervention)

$\mathcal{T}^* \leftarrow \textsc{Prune}\big(\textsc{GrowFull}(D_{\mathrm{tr}})\big)$ ▷ the tree sees the training part only
$\mathcal{L} \leftarrow \emptyset$
for each leaf $\ell \in \Lambda(\mathcal{T}^*)$ **do**
  $s_\ell \leftarrow$ the conditions on X along $\mathrm{path}(\ell)$ ▷ subgroup
  $a_\ell \leftarrow$ the conditions on T along $\mathrm{path}(\ell)$ ▷ intervention, treatments not on $\mathrm{path}(\ell)$ are left untouched
  $D_\ell \leftarrow \{(x, t, y) \in D_{\mathrm{val}} : x \models s_\ell \text{ and } t \models a_\ell\}$
  $\hat{p}_\ell \leftarrow \big(|\{(x, t, y) \in D_\ell : y = 1\}| + 1\big) \big/ \big(|D_\ell| + 2\big)$ ▷ Laplace correction, $\alpha = 1$
  $\mathcal{L} \leftarrow \mathcal{L} \cup \{(\ell, s_\ell, a_\ell, \hat{p}_\ell)\}$
end for
return (s_{ℓ^*}, a_{ℓ^*}) with $\ell^* = \arg\max_{(\ell, \cdot, \cdot, \hat{p}_\ell) \in \mathcal{L}} \hat{p}_\ell$, ties broken towards the smaller leaf index

Remarks.

1. *The order of growth does not matter.* The split chosen at a node depends on the rest of the tree only through $L_{\mathbb{N}}(|\Lambda|) + \log_2 \mathrm{Cat}_{|\Lambda|-1}$, which shifts every candidate at that node by the same amount. Algorithm 1 therefore returns the same fully grown tree whichever traversal is used, and our implementation uses depth-first traversal.

2. *Cost.* A node v costs $O(d\,|D_v| \log |D_v|)$, as one sort and one cumulative table of class counts per variable make every candidate cut on that variable an $O(1)$ lookup. Algorithm 2 costs $O(|\Lambda(\mathcal{T})|^2)$ in total.

3. *Subgroups overlap, but actions do not.* The leaves partition (X, T) jointly rather than X alone, so several leaves can contain the same covariate vector with different interventions, which is what makes the ranking in Algorithm 3 a comparison between actions. Each validation instance still matches exactly one action (s_ℓ, a_ℓ).

4. *The test part is never used here.* It enters only the evaluation of the returned action, so the ranking cannot be tuned on it.

C EXPERIMENTAL SETUP

C.1 ADAPTING THE COMPETITORS TO MULTIPLE TREATMENTS

Existing policy learning methods are designed for a single treatment variable. Most of them assume that this variable, which we denote by T', takes values in a finite set in which each value is one treatment option, and they learn a policy that assigns one of these options to each subgroup. Our setting differs in two ways, as the treatment is a vector $T = (T_1, \ldots, T_m)$ of binary variables, and an action may intervene on only some of them while leaving the others untouched. To apply these methods, we therefore reduce our problem to theirs.

Suppose we fix a set of k treatment variables to intervene on. Every intervention on this set corresponds to one of the 2^k joint values of these variables, so we can encode each joint value as a distinct integer and treat the result as a single categorical treatment T' with 2^k options. A competitor then learns a policy over T', and each option it assigns to a subgroup is decoded back into an intervention on the k chosen variables, which gives a subgroup-level causal action of the same form as those returned by SCAT.

This reduction requires the intervened variables to be chosen beforehand, and the number of options grows exponentially in k, so it cannot be applied to all m treatment variables at once. We therefore rank the treatment variables by the absolute value of their doubly robust average treatment effect, estimated on the training set with a causal forest (Athey et al., 2019) that adjusts for X and the other treatments, and run each competitor separately on the top k treatments for $k = 1, \ldots, 4$ (for $k = 1, 2, 3$ on the hand-built datasets, which have only three treatments). We then pool the candidate subgroup-level actions from all runs and select as the final action the candidate with the highest estimated interventional probability on the validation set. Here we only consider candidates that match at least 20 validation instances, since otherwise some of the learned subgroups contain too few instances for their interventional probability to be estimated meaningfully. SCAT needs no such floor, as its MDL criterion already penalizes leaves with very few instances (Algorithm 3 in Appendix B). Note that the largest k we consider, four, equals the number of treatment variables that directly affect Y in every MT dataset. SCAT does not use this information, as it searches over which treatment variables to intervene on as part of learning, so this setup, if anything, favors the competitors.

C.2 EVALUATION PROTOCOL AND METRICS

For the single-treatment experiments we use 30 of the 77 ACIC 2016 settings, drawn uniformly at random under a fixed seed, and no ground-truth property enters this choice; we do not use all 77 settings because several competitors require a Gurobi license, which prevents running the experiments in parallel across datasets and would make the full benchmark computationally prohibitive.

We randomly split each dataset into training, validation, and test sets containing 50%, 20%, and 30% of the instances. Each method is trained on the training set, and the subgroup-level action is selected as the one that maximizes the estimated subgroup interventional probability of $Y = 1$. All metrics are computed on the test set. We repeat every experiment over five random training/validation/test splits and report the mean and standard deviation over the splits. The only exceptions are the competitors that rely on a Gurobi license (FlowOPT-DR, and IOPL in Section 5.2), since the license does not allow us to run experiments in parallel at a large scale. We therefore run these methods on the first split only and report a single value. We report the following four metrics.

First, the ground-truth interventional probability $P(Y = 1)$ of the learned causal action on its matching subgroup, which directly assesses the objective in Definition 3.1.

Second, the accuracy of the estimated interventional probability of $Y = 1$ for the learned causal action and its matching subgroup. An accurate estimate is essential for decision support, because a method is only useful and reliable if it can estimate this probability accurately, rather than only rank the learned subgroup-level causal actions.

Third, the coverage of the subgroup-level causal action, i.e., the fraction of test instances that the learned subgroup contains. A very small coverage limits practical relevance, as with a medical treatment plan that benefits only a few patients. More importantly, with a very small coverage the

estimated interventional probability becomes unstable, and the learned causal action is therefore unreliable as well.

Finally, the runtime, measured as the wall-clock time for learning the subgroup-level causal action, as an indication of how well a method scales to large datasets; on the multi-treatment datasets it includes the treatment screening of the competitors, and on ACIC 2016 the nuisance-estimation step shared by the competitors is not charged.

D Additional experimental results

D.1 Multiple treatment variables

We present further experiment results here.

Table 4: Absolute estimation error of the selected action, $|\hat{P}(Y=1) - P(Y=1)|$, i.e. the method's own estimate against the true interventional probability on the three hand-built graphs (Chain, Fork, Collider; three binary treatments each) and the ten 20-treatment replicates MT01–MT10: mean$_{\pm \text{ sd}}$ over five split seeds. The best (lowest) mean in each row is in bold; the last row is the average rank (1 = best) over the 13 datasets, ranking the methods with a value in each row. †FlowOPT-DR was run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	MB-IAMB	MB-Lasso	MB-DRScreen	FlowOPT-DR†
Chain	$0.026_{\pm 0.011}$	$0.072_{\pm 0.046}$	$0.025_{\pm 0.020}$	$0.037_{\pm 0.025}$	$0.037_{\pm 0.025}$	$0.035_{\pm 0.028}$	$0.037_{\pm 0.025}$	**0.017**
Fork	$\mathbf{0.028}_{\pm 0.012}$	$0.048_{\pm 0.027}$	$0.031_{\pm 0.025}$	$0.048_{\pm 0.039}$	$0.048_{\pm 0.039}$	$0.034_{\pm 0.041}$	$0.048_{\pm 0.039}$	0.036
Collider	$\mathbf{0.016}_{\pm 0.015}$	$0.097_{\pm 0.077}$	$0.043_{\pm 0.033}$	$0.050_{\pm 0.031}$	$0.025_{\pm 0.024}$	$0.028_{\pm 0.026}$	$0.020_{\pm 0.021}$	0.120
MT01	$0.072_{\pm 0.047}$	$0.066_{\pm 0.065}$	$\mathbf{0.031}_{\pm 0.025}$	$0.048_{\pm 0.063}$	$0.060_{\pm 0.059}$	$0.048_{\pm 0.063}$	$0.048_{\pm 0.063}$	0.032
MT02	$\mathbf{0.038}_{\pm 0.025}$	$0.074_{\pm 0.042}$	$\mathbf{0.038}_{\pm 0.015}$	$0.081_{\pm 0.077}$	$0.082_{\pm 0.076}$	$0.065_{\pm 0.079}$	$0.080_{\pm 0.078}$	0.082
MT03	$0.034_{\pm 0.022}$	$0.078_{\pm 0.054}$	$\mathbf{0.029}_{\pm 0.018}$	$0.042_{\pm 0.039}$	$0.060_{\pm 0.039}$	$0.070_{\pm 0.058}$	$0.054_{\pm 0.065}$	0.108
MT04	$0.032_{\pm 0.029}$	$0.054_{\pm 0.049}$	$0.043_{\pm 0.027}$	$0.029_{\pm 0.008}$	$0.028_{\pm 0.017}$	$0.032_{\pm 0.019}$	$\mathbf{0.023}_{\pm 0.014}$	0.073
MT05	$\mathbf{0.031}_{\pm 0.023}$	$0.042_{\pm 0.032}$	$0.040_{\pm 0.009}$	$0.047_{\pm 0.018}$	$0.047_{\pm 0.018}$	$0.044_{\pm 0.027}$	$0.047_{\pm 0.018}$	0.170
MT06	$0.041_{\pm 0.041}$	$\mathbf{0.026}_{\pm 0.019}$	$0.055_{\pm 0.045}$	$0.088_{\pm 0.093}$	$0.088_{\pm 0.093}$	$0.075_{\pm 0.095}$	$0.088_{\pm 0.093}$	0.072
MT07	$0.027_{\pm 0.014}$	$0.026_{\pm 0.023}$	$0.052_{\pm 0.044}$	$\mathbf{0.020}_{\pm 0.012}$	$\mathbf{0.020}_{\pm 0.012}$	$\mathbf{0.020}_{\pm 0.012}$	$\mathbf{0.020}_{\pm 0.012}$	0.024
MT08	$\mathbf{0.007}_{\pm 0.007}$	$0.035_{\pm 0.024}$	$0.047_{\pm 0.040}$	$0.033_{\pm 0.045}$	$0.019_{\pm 0.021}$	$0.019_{\pm 0.021}$	$0.025_{\pm 0.019}$	0.083
MT09	$\mathbf{0.015}_{\pm 0.013}$	$0.052_{\pm 0.036}$	$0.052_{\pm 0.041}$	$0.072_{\pm 0.065}$	$0.082_{\pm 0.056}$	$0.043_{\pm 0.032}$	$0.071_{\pm 0.065}$	0.064
MT10	$\mathbf{0.012}_{\pm 0.011}$	$0.044_{\pm 0.043}$	$0.035_{\pm 0.029}$	$0.052_{\pm 0.064}$	$0.041_{\pm 0.064}$	$0.041_{\pm 0.064}$	$0.041_{\pm 0.064}$	0.045
Avg. rank	2.69	5.46	3.31	5.50	5.08	3.77	4.42	5.77

Table 5: Coverage of the selected subgroup, as the fraction of test units it contains on the three hand-built graphs (Chain, Fork, Collider; three binary treatments each) and the ten 20-treatment replicates MT01–MT10: mean$_{\pm \text{ sd}}$ over five split seeds. The last row is the mean over the 13 datasets. †FlowOPT-DR was run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	MB-IAMB	MB-Lasso	MB-DRScreen	FlowOPT-DR†
Chain	$0.299_{\pm 0.042}$	$0.550_{\pm 0.332}$	$0.473_{\pm 0.294}$	$0.230_{\pm 0.165}$	$0.267_{\pm 0.183}$	$0.149_{\pm 0.185}$	$0.267_{\pm 0.183}$	0.326
Fork	$0.247_{\pm 0.010}$	$0.330_{\pm 0.150}$	$0.307_{\pm 0.152}$	$0.397_{\pm 0.268}$	$0.397_{\pm 0.268}$	$0.352_{\pm 0.266}$	$0.397_{\pm 0.268}$	0.329
Collider	$0.182_{\pm 0.044}$	$0.383_{\pm 0.273}$	$0.326_{\pm 0.385}$	$0.063_{\pm 0.033}$	$0.351_{\pm 0.386}$	$0.411_{\pm 0.350}$	$0.402_{\pm 0.239}$	0.508
MT01	$0.071_{\pm 0.014}$	$0.506_{\pm 0.318}$	$0.288_{\pm 0.137}$	$0.253_{\pm 0.135}$	$0.277_{\pm 0.151}$	$0.253_{\pm 0.135}$	$0.253_{\pm 0.135}$	0.226
MT02	$0.505_{\pm 0.246}$	$0.598_{\pm 0.234}$	$0.624_{\pm 0.149}$	$0.247_{\pm 0.107}$	$0.280_{\pm 0.144}$	$0.292_{\pm 0.154}$	$0.334_{\pm 0.080}$	0.409
MT03	$0.086_{\pm 0.024}$	$0.552_{\pm 0.387}$	$0.510_{\pm 0.248}$	$0.453_{\pm 0.090}$	$0.310_{\pm 0.165}$	$0.249_{\pm 0.211}$	$0.367_{\pm 0.199}$	0.337
MT04	$0.252_{\pm 0.211}$	$0.238_{\pm 0.134}$	$0.292_{\pm 0.114}$	$0.229_{\pm 0.079}$	$0.240_{\pm 0.086}$	$0.267_{\pm 0.029}$	$0.239_{\pm 0.086}$	0.230
MT05	$0.404_{\pm 0.225}$	$0.496_{\pm 0.351}$	$0.338_{\pm 0.098}$	$0.241_{\pm 0.106}$	$0.241_{\pm 0.106}$	$0.286_{\pm 0.106}$	$0.241_{\pm 0.106}$	0.248
MT06	$0.193_{\pm 0.188}$	$0.682_{\pm 0.424}$	$0.355_{\pm 0.233}$	$0.171_{\pm 0.099}$	$0.171_{\pm 0.099}$	$0.171_{\pm 0.099}$	$0.171_{\pm 0.099}$	0.209
MT07	$0.062_{\pm 0.017}$	$0.349_{\pm 0.285}$	$0.281_{\pm 0.199}$	$0.338_{\pm 0.154}$	$0.338_{\pm 0.154}$	$0.338_{\pm 0.154}$	$0.338_{\pm 0.154}$	0.296
MT08	$0.250_{\pm 0.177}$	$0.800_{\pm 0.111}$	$0.404_{\pm 0.199}$	$0.182_{\pm 0.140}$	$0.278_{\pm 0.286}$	$0.278_{\pm 0.286}$	$0.243_{\pm 0.265}$	0.232
MT09	$0.111_{\pm 0.035}$	$0.385_{\pm 0.404}$	$0.292_{\pm 0.230}$	$0.205_{\pm 0.143}$	$0.251_{\pm 0.209}$	$0.342_{\pm 0.358}$	$0.233_{\pm 0.198}$	0.489
MT10	$0.318_{\pm 0.018}$	$0.452_{\pm 0.360}$	$0.338_{\pm 0.112}$	$0.218_{\pm 0.135}$	$0.248_{\pm 0.183}$	$0.255_{\pm 0.190}$	$0.248_{\pm 0.183}$	0.327
Mean	0.229	0.486	0.371	0.248	0.281	0.280	0.287	0.320

D.2 One-dimensional treatment variable

Tables 7–10 report the per-dataset results on the 30 ACIC 2016 settings summarized in Table 3.

Table 6: Total runtime in seconds (fit plus treatment screening, where a method screens) on the three hand-built graphs (Chain, Fork, Collider; three binary treatments each) and the ten 20-treatment replicates MT01–MT10: mean$_{\pm \text{ sd}}$ over five split seeds. The best (lowest) mean in each row is in bold; the last row is the average rank (1 = best) over the 13 datasets, ranking the methods with a value in each row. †FlowOPT-DR was run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	MB-IAMB	MB-Lasso	MB-DRScreen	FlowOPT-DR†
Chain	$\mathbf{6.3}_{\pm 0.3}$	$135_{\pm 128}$	$36.9_{\pm 0.8}$	$24.7_{\pm 6.5}$	$21.8_{\pm 0.7}$	$22.0_{\pm 0.5}$	$22.1_{\pm 0.6}$	138
Fork	$\mathbf{6.4}_{\pm 0.3}$	$172_{\pm 105}$	$37.5_{\pm 1.0}$	$24.7_{\pm 6.2}$	$21.6_{\pm 1.2}$	$21.5_{\pm 0.8}$	$21.4_{\pm 0.8}$	141
Collider	$\mathbf{6.6}_{\pm 0.4}$	$84.9_{\pm 24.5}$	$36.8_{\pm 1.3}$	$23.8_{\pm 6.5}$	$20.9_{\pm 0.4}$	$22.0_{\pm 0.6}$	$22.0_{\pm 0.8}$	134
MT01	$\mathbf{54.1}_{\pm 6.8}$	$17,412_{\pm 502}$	$1,427_{\pm 57}$	$1,848_{\pm 206}$	$2,697_{\pm 187}$	$1,561_{\pm 853}$	$2,437_{\pm 715}$	14,534
MT02	$\mathbf{52.0}_{\pm 7.1}$	$17,568_{\pm 292}$	$1,417_{\pm 56}$	$1,803_{\pm 235}$	$1,818_{\pm 459}$	$1,530_{\pm 56}$	$1,534_{\pm 21}$	14,539
MT03	$\mathbf{49.6}_{\pm 3.5}$	$17,987_{\pm 299}$	$1,426_{\pm 47}$	$1,714_{\pm 239}$	$2,208_{\pm 643}$	$1,198_{\pm 368}$	$1,271_{\pm 330}$	14,536
MT04	$\mathbf{49.9}_{\pm 3.6}$	$17,900_{\pm 631}$	$1,414_{\pm 49}$	$1,649_{\pm 224}$	$2,422_{\pm 566}$	$1,180_{\pm 381}$	$1,598_{\pm 594}$	14,536
MT05	$\mathbf{47.6}_{\pm 3.3}$	$17,311_{\pm 181}$	$1,420_{\pm 35}$	$1,669_{\pm 223}$	$1,996_{\pm 714}$	$1,164_{\pm 872}$	$1,731_{\pm 530}$	14,538
MT06	$\mathbf{49.5}_{\pm 2.7}$	$17,289_{\pm 306}$	$1,391_{\pm 44}$	$1,678_{\pm 221}$	$2,597_{\pm 117}$	$752_{\pm 218}$	$990_{\pm 429}$	14,534
MT07	$\mathbf{48.0}_{\pm 1.4}$	$17,951_{\pm 532}$	$1,401_{\pm 45}$	$1,732_{\pm 282}$	$2,321_{\pm 754}$	$1,337_{\pm 376}$	$1,723_{\pm 736}$	14,539
MT08	$\mathbf{47.7}_{\pm 2.2}$	$17,864_{\pm 311}$	$1,399_{\pm 57}$	$1,677_{\pm 259}$	$2,236_{\pm 744}$	$796_{\pm 437}$	$1,384_{\pm 265}$	14,586
MT09	$\mathbf{51.7}_{\pm 3.8}$	$17,889_{\pm 405}$	$1,434_{\pm 48}$	$1,738_{\pm 260}$	$2,101_{\pm 830}$	$910_{\pm 614}$	$1,341_{\pm 380}$	14,540
MT10	$\mathbf{49.7}_{\pm 5.2}$	$17,589_{\pm 369}$	$1,327_{\pm 187}$	$1,692_{\pm 213}$	$1,906_{\pm 543}$	$701_{\pm 386}$	$1,319_{\pm 411}$	14,534
Avg. rank	1.00	7.85	3.92	4.85	5.23	2.38	3.62	7.15

Table 7: True interventional $P(Y = 1)$ of the selected subgroup-level action on the 30 ACIC 2016 benchmark settings: mean$_{\pm \text{ sd}}$ over five split seeds. The best (highest) mean in each row is in bold; the last row is the average rank (1 = best) over the 30 datasets, ranking the methods with a value in each row. †IOPL and FlowOPT-DR were run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	EconML	IOPL†	FlowOPT-DR†
ACIC-4	$\mathbf{0.983}_{\pm 0.009}$	$0.888_{\pm 0.012}$	$0.897_{\pm 0.017}$	$0.896_{\pm 0.016}$	$0.904_{\pm 0.048}$	0.860	0.868
ACIC-6	$0.974_{\pm 0.035}$	$\mathbf{1.000}_{\pm 0.001}$	$0.805_{\pm 0.025}$	$0.813_{\pm 0.033}$	$0.848_{\pm 0.089}$	0.789	0.782
ACIC-11	$\mathbf{0.901}_{\pm 0.092}$	$0.728_{\pm 0.086}$	$0.780_{\pm 0.092}$	$0.842_{\pm 0.079}$	$0.732_{\pm 0.061}$	0.595	0.776
ACIC-12	$\mathbf{0.931}_{\pm 0.015}$	$0.915_{\pm 0.016}$	$0.917_{\pm 0.055}$	$0.927_{\pm 0.056}$	$0.896_{\pm 0.028}$	0.749	0.886
ACIC-13	$\mathbf{0.981}_{\pm 0.008}$	$0.815_{\pm 0.078}$	$0.905_{\pm 0.044}$	$0.922_{\pm 0.042}$	$0.839_{\pm 0.065}$	0.849	0.801
ACIC-14	$\mathbf{0.987}_{\pm 0.006}$	$0.984_{\pm 0.003}$	$0.927_{\pm 0.003}$	$0.930_{\pm 0.008}$	$0.929_{\pm 0.004}$	0.931	0.954
ACIC-16	$\mathbf{0.987}_{\pm 0.004}$	$0.814_{\pm 0.009}$	$0.915_{\pm 0.048}$	$0.917_{\pm 0.050}$	$0.850_{\pm 0.060}$	0.676	0.861
ACIC-18	$\mathbf{0.901}_{\pm 0.032}$	$0.823_{\pm 0.188}$	$0.675_{\pm 0.085}$	$0.728_{\pm 0.060}$	$0.663_{\pm 0.015}$	0.640	0.642
ACIC-19	$\mathbf{0.992}_{\pm 0.006}$	$0.845_{\pm 0.083}$	$0.855_{\pm 0.015}$	$0.855_{\pm 0.016}$	$0.747_{\pm 0.059}$	0.727	0.736
ACIC-21	$\mathbf{0.994}_{\pm 0.008}$	$0.852_{\pm 0.007}$	$0.900_{\pm 0.089}$	$0.893_{\pm 0.063}$	$0.937_{\pm 0.086}$	0.850	0.849
ACIC-22	$\mathbf{0.961}_{\pm 0.026}$	$0.755_{\pm 0.110}$	$0.694_{\pm 0.025}$	$0.715_{\pm 0.028}$	$0.819_{\pm 0.160}$	0.602	0.603
ACIC-25	$0.965_{\pm 0.032}$	$\mathbf{0.997}_{\pm 0.003}$	$0.890_{\pm 0.107}$	$0.851_{\pm 0.120}$	$0.853_{\pm 0.127}$	0.772	0.772
ACIC-32	$\mathbf{0.968}_{\pm 0.019}$	$0.901_{\pm 0.002}$	$0.902_{\pm 0.028}$	$0.907_{\pm 0.033}$	$0.887_{\pm 0.036}$	0.838	0.905
ACIC-33	$\mathbf{0.944}_{\pm 0.065}$	$0.926_{\pm 0.030}$	$0.919_{\pm 0.016}$	$0.940_{\pm 0.042}$	$0.912_{\pm 0.014}$	0.671	0.901
ACIC-35	$\mathbf{0.813}_{\pm 0.022}$	$0.725_{\pm 0.067}$	$0.687_{\pm 0.028}$	$0.720_{\pm 0.047}$	$0.705_{\pm 0.026}$	0.449	0.687
ACIC-36	$\mathbf{0.938}_{\pm 0.024}$	$0.917_{\pm 0.017}$	$0.682_{\pm 0.099}$	$0.739_{\pm 0.105}$	$0.685_{\pm 0.020}$	0.630	0.649
ACIC-39	$\mathbf{0.936}_{\pm 0.026}$	$0.836_{\pm 0.064}$	$0.870_{\pm 0.027}$	$0.885_{\pm 0.024}$	$0.813_{\pm 0.038}$	0.792	0.892
ACIC-41	$\mathbf{0.991}_{\pm 0.007}$	$0.665_{\pm 0.008}$	$0.679_{\pm 0.024}$	$0.731_{\pm 0.068}$	$0.745_{\pm 0.084}$	0.629	0.664
ACIC-43	$\mathbf{0.960}_{\pm 0.013}$	$0.726_{\pm 0.062}$	$0.673_{\pm 0.023}$	$0.667_{\pm 0.024}$	$0.666_{\pm 0.050}$	0.597	0.721
ACIC-44	$\mathbf{0.980}_{\pm 0.012}$	$0.703_{\pm 0.064}$	$0.694_{\pm 0.024}$	$0.785_{\pm 0.124}$	$0.683_{\pm 0.030}$	0.644	0.757
ACIC-45	$0.986_{\pm 0.005}$	$0.867_{\pm 0.011}$	$0.875_{\pm 0.032}$	$0.899_{\pm 0.051}$	$0.878_{\pm 0.024}$	**0.996**	0.840
ACIC-48	$0.448_{\pm 0.286}$	$\mathbf{0.900}_{\pm 0.202}$	$0.836_{\pm 0.155}$	$0.826_{\pm 0.122}$	$0.781_{\pm 0.194}$	0.529	0.607
ACIC-51	$0.930_{\pm 0.075}$	$\mathbf{0.954}_{\pm 0.031}$	$0.840_{\pm 0.036}$	$0.844_{\pm 0.080}$	$0.819_{\pm 0.052}$	0.801	0.803
ACIC-54	$\mathbf{0.984}_{\pm 0.002}$	$0.844_{\pm 0.005}$	$0.788_{\pm 0.020}$	$0.814_{\pm 0.047}$	$0.777_{\pm 0.033}$	0.748	0.754
ACIC-55	$\mathbf{0.979}_{\pm 0.006}$	$0.690_{\pm 0.008}$	$0.747_{\pm 0.111}$	$0.751_{\pm 0.109}$	$0.689_{\pm 0.039}$	0.689	0.687
ACIC-60	$0.948_{\pm 0.015}$	$\mathbf{0.992}_{\pm 0.014}$	$0.646_{\pm 0.028}$	$0.722_{\pm 0.133}$	$0.627_{\pm 0.039}$	0.639	0.635
ACIC-62	$\mathbf{0.957}_{\pm 0.047}$	$0.897_{\pm 0.003}$	$0.913_{\pm 0.036}$	$0.911_{\pm 0.038}$	$0.881_{\pm 0.021}$	0.793	0.956
ACIC-70	$\mathbf{0.981}_{\pm 0.006}$	$0.972_{\pm 0.016}$	$0.928_{\pm 0.068}$	$0.926_{\pm 0.030}$	$0.748_{\pm 0.112}$	0.599	0.709
ACIC-72	$\mathbf{0.991}_{\pm 0.004}$	$0.672_{\pm 0.179}$	$0.691_{\pm 0.177}$	$0.669_{\pm 0.142}$	$0.618_{\pm 0.052}$	0.580	0.697
ACIC-75	$\mathbf{0.994}_{\pm 0.004}$	$0.944_{\pm 0.009}$	$0.856_{\pm 0.044}$	$0.924_{\pm 0.029}$	$0.846_{\pm 0.067}$	0.801	0.829
Avg. rank	1.37	3.37	3.87	3.17	4.70	6.33	5.20

Table 8: Absolute estimation error of the selected action, $|\hat{P}(Y=1) - P(Y=1)|$, i.e. the method's own estimate against the true interventional probability on the 30 ACIC 2016 benchmark settings: mean$_{\pm \mathrm{sd}}$ over five split seeds. The best (lowest) mean in each row is in bold; the last row is the average rank (1 = best) over the 30 datasets, ranking the methods with a value in each row. †IOPL and FlowOPT-DR were run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	EconML	IOPL†	FlowOPT-DR†
ACIC-4	$\mathbf{0.004}_{\pm 0.003}$	$0.021_{\pm 0.009}$	$0.019_{\pm 0.008}$	$0.015_{\pm 0.010}$	$0.023_{\pm 0.003}$	0.006	0.015
ACIC-6	$0.025_{\pm 0.026}$	$0.018_{\pm 0.004}$	$0.045_{\pm 0.034}$	$0.036_{\pm 0.036}$	$0.048_{\pm 0.026}$	**0.014**	0.042
ACIC-11	$0.033_{\pm 0.032}$	$0.029_{\pm 0.026}$	$0.070_{\pm 0.059}$	$0.022_{\pm 0.012}$	$0.065_{\pm 0.030}$	0.103	**0.009**
ACIC-12	$0.033_{\pm 0.010}$	$0.037_{\pm 0.008}$	$\mathbf{0.025}_{\pm 0.021}$	$0.025_{\pm 0.018}$	$0.033_{\pm 0.009}$	0.039	0.028
ACIC-13	$\mathbf{0.005}_{\pm 0.003}$	$0.028_{\pm 0.011}$	$0.021_{\pm 0.019}$	$0.019_{\pm 0.016}$	$0.048_{\pm 0.021}$	0.026	0.010
ACIC-14	$\mathbf{0.003}_{\pm 0.003}$	$0.004_{\pm 0.003}$	$0.015_{\pm 0.009}$	$0.017_{\pm 0.010}$	$0.016_{\pm 0.008}$	0.021	0.016
ACIC-16	$0.005_{\pm 0.002}$	$0.027_{\pm 0.014}$	$0.019_{\pm 0.013}$	$0.016_{\pm 0.011}$	$0.019_{\pm 0.016}$	**0.003**	0.018
ACIC-18	$0.021_{\pm 0.025}$	$0.031_{\pm 0.010}$	$0.026_{\pm 0.021}$	$0.024_{\pm 0.020}$	$0.021_{\pm 0.025}$	**0.002**	0.008
ACIC-19	$0.009_{\pm 0.013}$	$0.022_{\pm 0.022}$	$0.020_{\pm 0.020}$	$0.024_{\pm 0.022}$	$0.037_{\pm 0.040}$	0.012	**0.008**
ACIC-21	$\mathbf{0.013}_{\pm 0.007}$	$0.029_{\pm 0.016}$	$0.052_{\pm 0.028}$	$0.030_{\pm 0.014}$	$0.029_{\pm 0.007}$	0.040	0.042
ACIC-22	$0.017_{\pm 0.021}$	$\mathbf{0.009}_{\pm 0.007}$	$0.035_{\pm 0.039}$	$0.022_{\pm 0.030}$	$0.010_{\pm 0.007}$	0.023	0.024
ACIC-25	$0.052_{\pm 0.041}$	$\mathbf{0.008}_{\pm 0.007}$	$0.020_{\pm 0.012}$	$0.043_{\pm 0.027}$	$0.044_{\pm 0.033}$	0.088	0.088
ACIC-32	$\mathbf{0.010}_{\pm 0.005}$	$0.019_{\pm 0.007}$	$0.025_{\pm 0.010}$	$0.022_{\pm 0.010}$	$0.030_{\pm 0.035}$	0.031	0.030
ACIC-33	$0.024_{\pm 0.017}$	$0.014_{\pm 0.015}$	$0.022_{\pm 0.019}$	$0.016_{\pm 0.009}$	$\mathbf{0.009}_{\pm 0.006}$	0.050	0.009
ACIC-35	$0.044_{\pm 0.021}$	$0.061_{\pm 0.018}$	$0.078_{\pm 0.031}$	$0.062_{\pm 0.042}$	$0.056_{\pm 0.016}$	**0.024**	0.072
ACIC-36	$0.027_{\pm 0.013}$	$0.019_{\pm 0.017}$	$0.024_{\pm 0.015}$	$\mathbf{0.019}_{\pm 0.014}$	$0.020_{\pm 0.011}$	0.035	0.036
ACIC-39	$0.017_{\pm 0.007}$	$0.024_{\pm 0.028}$	$0.017_{\pm 0.009}$	$0.016_{\pm 0.011}$	$0.010_{\pm 0.009}$	0.008	**0.002**
ACIC-41	$\mathbf{0.003}_{\pm 0.003}$	$0.013_{\pm 0.010}$	$0.008_{\pm 0.005}$	$0.032_{\pm 0.039}$	$0.024_{\pm 0.014}$	0.009	0.024
ACIC-43	$\mathbf{0.006}_{\pm 0.004}$	$0.029_{\pm 0.023}$	$0.036_{\pm 0.016}$	$0.038_{\pm 0.015}$	$0.054_{\pm 0.025}$	0.040	0.016
ACIC-44	$0.018_{\pm 0.027}$	$0.014_{\pm 0.010}$	$0.012_{\pm 0.006}$	$0.017_{\pm 0.007}$	$0.017_{\pm 0.012}$	0.006	**0.004**
ACIC-45	$\mathbf{0.002}_{\pm 0.002}$	$0.013_{\pm 0.006}$	$0.010_{\pm 0.008}$	$0.014_{\pm 0.004}$	$0.009_{\pm 0.005}$	0.003	0.019
ACIC-48	$0.052_{\pm 0.037}$	$0.042_{\pm 0.051}$	$\mathbf{0.036}_{\pm 0.021}$	$0.041_{\pm 0.043}$	$0.037_{\pm 0.020}$	0.059	0.054
ACIC-51	$\mathbf{0.019}_{\pm 0.014}$	$0.033_{\pm 0.038}$	$0.064_{\pm 0.022}$	$0.059_{\pm 0.034}$	$0.065_{\pm 0.028}$	0.077	0.077
ACIC-54	$\mathbf{0.016}_{\pm 0.004}$	$0.034_{\pm 0.009}$	$0.049_{\pm 0.039}$	$0.036_{\pm 0.029}$	$0.025_{\pm 0.024}$	0.033	0.037
ACIC-55	$\mathbf{0.005}_{\pm 0.001}$	$0.073_{\pm 0.014}$	$0.059_{\pm 0.030}$	$0.062_{\pm 0.033}$	$0.095_{\pm 0.014}$	0.081	0.072
ACIC-60	$\mathbf{0.012}_{\pm 0.002}$	$0.019_{\pm 0.006}$	$0.037_{\pm 0.017}$	$0.026_{\pm 0.016}$	$0.016_{\pm 0.008}$	0.036	0.037
ACIC-62	$0.021_{\pm 0.016}$	$0.060_{\pm 0.005}$	$0.049_{\pm 0.025}$	$0.050_{\pm 0.027}$	$0.064_{\pm 0.007}$	0.071	**0.012**
ACIC-70	$0.008_{\pm 0.011}$	$0.007_{\pm 0.007}$	$0.038_{\pm 0.024}$	$0.016_{\pm 0.003}$	$0.027_{\pm 0.026}$	**0.007**	0.017
ACIC-72	$\mathbf{0.015}_{\pm 0.009}$	$0.030_{\pm 0.017}$	$0.033_{\pm 0.019}$	$0.039_{\pm 0.026}$	$0.049_{\pm 0.040}$	0.036	0.037
ACIC-75	$\mathbf{0.004}_{\pm 0.003}$	$0.029_{\pm 0.011}$	$0.014_{\pm 0.013}$	$0.025_{\pm 0.018}$	$0.029_{\pm 0.015}$	**0.004**	**0.004**
Avg. rank	2.57	3.87	4.40	4.13	4.63	4.27	4.13

Table 9: Coverage of the selected subgroup, as the fraction of test units it contains on the 30 ACIC 2016 benchmark settings: mean$_{\pm \text{ sd}}$ over five split seeds. The last row is the mean over the 30 datasets. †IOPL and FlowOPT-DR were run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	EconML	IOPL†	FlowOPT-DR†
ACIC-4	$0.740_{\pm 0.046}$	$0.929_{\pm 0.046}$	$0.938_{\pm 0.020}$	$0.736_{\pm 0.384}$	$0.604_{\pm 0.306}$	0.976	0.818
ACIC-6	$0.091_{\pm 0.023}$	$0.095_{\pm 0.020}$	$0.725_{\pm 0.383}$	$0.445_{\pm 0.447}$	$0.464_{\pm 0.334}$	0.988	0.344
ACIC-11	$0.073_{\pm 0.044}$	$0.651_{\pm 0.324}$	$0.275_{\pm 0.296}$	$0.196_{\pm 0.130}$	$0.386_{\pm 0.370}$	0.060	0.749
ACIC-12	$0.658_{\pm 0.072}$	$0.717_{\pm 0.045}$	$0.379_{\pm 0.254}$	$0.343_{\pm 0.226}$	$0.352_{\pm 0.354}$	0.981	0.787
ACIC-13	$0.299_{\pm 0.011}$	$0.591_{\pm 0.304}$	$0.275_{\pm 0.191}$	$0.292_{\pm 0.231}$	$0.221_{\pm 0.323}$	0.563	0.701
ACIC-14	$0.414_{\pm 0.039}$	$0.436_{\pm 0.012}$	$0.535_{\pm 0.019}$	$0.448_{\pm 0.102}$	$0.470_{\pm 0.047}$	0.105	0.439
ACIC-16	$0.476_{\pm 0.033}$	$0.818_{\pm 0.007}$	$0.316_{\pm 0.278}$	$0.301_{\pm 0.287}$	$0.613_{\pm 0.288}$	0.979	0.400
ACIC-18	$0.142_{\pm 0.076}$	$0.218_{\pm 0.391}$	$0.546_{\pm 0.447}$	$0.352_{\pm 0.299}$	$0.358_{\pm 0.256}$	0.977	0.295
ACIC-19	$0.281_{\pm 0.136}$	$0.579_{\pm 0.315}$	$0.319_{\pm 0.193}$	$0.286_{\pm 0.166}$	$0.154_{\pm 0.225}$	0.986	0.713
ACIC-21	$0.114_{\pm 0.130}$	$0.979_{\pm 0.030}$	$0.072_{\pm 0.042}$	$0.395_{\pm 0.458}$	$0.357_{\pm 0.441}$	0.978	0.983
ACIC-22	$0.279_{\pm 0.207}$	$0.687_{\pm 0.370}$	$0.503_{\pm 0.344}$	$0.372_{\pm 0.327}$	$0.504_{\pm 0.421}$	0.992	0.997
ACIC-25	$0.066_{\pm 0.033}$	$0.120_{\pm 0.064}$	$0.235_{\pm 0.349}$	$0.381_{\pm 0.434}$	$0.427_{\pm 0.416}$	0.976	0.997
ACIC-32	$0.467_{\pm 0.173}$	$0.802_{\pm 0.010}$	$0.592_{\pm 0.205}$	$0.577_{\pm 0.219}$	$0.684_{\pm 0.325}$	0.869	0.730
ACIC-33	$0.129_{\pm 0.042}$	$0.198_{\pm 0.093}$	$0.167_{\pm 0.059}$	$0.148_{\pm 0.051}$	$0.227_{\pm 0.014}$	0.981	0.269
ACIC-35	$0.158_{\pm 0.045}$	$0.255_{\pm 0.223}$	$0.174_{\pm 0.109}$	$0.102_{\pm 0.091}$	$0.383_{\pm 0.183}$	0.130	0.405
ACIC-36	$0.304_{\pm 0.195}$	$0.048_{\pm 0.027}$	$0.866_{\pm 0.142}$	$0.563_{\pm 0.273}$	$0.680_{\pm 0.087}$	0.985	0.779
ACIC-39	$0.435_{\pm 0.092}$	$0.773_{\pm 0.424}$	$0.410_{\pm 0.318}$	$0.354_{\pm 0.266}$	$0.532_{\pm 0.292}$	0.996	0.672
ACIC-41	$0.361_{\pm 0.062}$	$0.944_{\pm 0.003}$	$0.528_{\pm 0.325}$	$0.372_{\pm 0.365}$	$0.449_{\pm 0.331}$	0.985	0.482
ACIC-43	$0.229_{\pm 0.131}$	$0.327_{\pm 0.355}$	$0.735_{\pm 0.152}$	$0.710_{\pm 0.142}$	$0.434_{\pm 0.355}$	0.983	0.044
ACIC-44	$0.183_{\pm 0.066}$	$0.701_{\pm 0.361}$	$0.736_{\pm 0.120}$	$0.544_{\pm 0.235}$	$0.754_{\pm 0.295}$	0.951	0.533
ACIC-45	$0.430_{\pm 0.012}$	$0.699_{\pm 0.021}$	$0.500_{\pm 0.235}$	$0.342_{\pm 0.233}$	$0.530_{\pm 0.288}$	0.339	0.719
ACIC-48	$0.114_{\pm 0.025}$	$0.079_{\pm 0.076}$	$0.117_{\pm 0.031}$	$0.064_{\pm 0.023}$	$0.077_{\pm 0.021}$	0.956	0.321
ACIC-51	$0.166_{\pm 0.175}$	$0.057_{\pm 0.023}$	$0.500_{\pm 0.437}$	$0.321_{\pm 0.373}$	$0.513_{\pm 0.257}$	0.974	0.984
ACIC-54	$0.485_{\pm 0.009}$	$0.758_{\pm 0.009}$	$0.612_{\pm 0.252}$	$0.416_{\pm 0.234}$	$0.702_{\pm 0.107}$	0.929	0.982
ACIC-55	$0.397_{\pm 0.006}$	$0.983_{\pm 0.015}$	$0.629_{\pm 0.224}$	$0.591_{\pm 0.219}$	$0.423_{\pm 0.218}$	0.969	0.993
ACIC-60	$0.149_{\pm 0.020}$	$0.052_{\pm 0.010}$	$0.501_{\pm 0.250}$	$0.316_{\pm 0.245}$	$0.797_{\pm 0.090}$	0.945	0.977
ACIC-62	$0.237_{\pm 0.141}$	$0.807_{\pm 0.010}$	$0.738_{\pm 0.147}$	$0.602_{\pm 0.183}$	$0.830_{\pm 0.068}$	0.972	0.619
ACIC-70	$0.204_{\pm 0.055}$	$0.148_{\pm 0.055}$	$0.135_{\pm 0.096}$	$0.115_{\pm 0.071}$	$0.359_{\pm 0.358}$	0.970	0.482
ACIC-72	$0.081_{\pm 0.012}$	$0.704_{\pm 0.419}$	$0.480_{\pm 0.260}$	$0.373_{\pm 0.244}$	$0.193_{\pm 0.117}$	0.986	0.276
ACIC-75	$0.407_{\pm 0.037}$	$0.111_{\pm 0.071}$	$0.572_{\pm 0.224}$	$0.312_{\pm 0.232}$	$0.101_{\pm 0.057}$	0.264	0.661
Mean	0.286	0.509	0.470	0.379	0.453	0.825	0.638

Table 10: Fit time in seconds (the shared nuisance-estimation step is not charged) on the 30 ACIC 2016 benchmark settings: $\text{mean}_{\pm \text{sd}}$ over five split seeds. The best (lowest) mean in each row is in bold; the last row is the average rank (1 = best) over the 30 datasets, ranking the methods with a value in each row. †IOPL and FlowOPT-DR were run on the first split only, so a single value (no sd) is shown.

Dataset	**SCAT**	ListDTR	PT-Exact	PT-Hybrid	EconML	IOPL†	FlowOPT-DR†
ACIC-4	$38.5_{\pm 3.0}$	$1,048_{\pm 265}$	$28.2_{\pm 1.6}$	$30.3_{\pm 16.4}$	$\mathbf{2.8}_{\pm 0.1}$	$1,815$	$1,858$
ACIC-6	$30.9_{\pm 3.2}$	$1,068_{\pm 95}$	$22.5_{\pm 9.3}$	$21.4_{\pm 1.0}$	$\mathbf{2.9}_{\pm 0.1}$	$1,819$	$1,858$
ACIC-11	$34.6_{\pm 1.1}$	$1,028_{\pm 326}$	$11.4_{\pm 0.5}$	$20.1_{\pm 1.5}$	$\mathbf{2.9}_{\pm 0.1}$	$1,815$	$1,856$
ACIC-12	$42.0_{\pm 3.3}$	$967_{\pm 185}$	$11.9_{\pm 0.6}$	$19.7_{\pm 0.7}$	$\mathbf{2.9}_{\pm 0.1}$	$1,818$	$1,856$
ACIC-13	$40.7_{\pm 1.5}$	$940_{\pm 220}$	$11.9_{\pm 0.5}$	$19.2_{\pm 1.2}$	$\mathbf{2.9}_{\pm 0.1}$	$1,817$	$1,858$
ACIC-14	$31.8_{\pm 1.9}$	$1,277_{\pm 152}$	$12.3_{\pm 0.7}$	$19.0_{\pm 1.4}$	$\mathbf{2.8}_{\pm 0.1}$	$1,819$	$1,858$
ACIC-16	$21.7_{\pm 17.3}$	$891_{\pm 201}$	$13.0_{\pm 0.4}$	$19.7_{\pm 0.5}$	$\mathbf{2.4}_{\pm 0.4}$	$1,813$	$1,856$
ACIC-18	$10.8_{\pm 0.8}$	$907_{\pm 24}$	$11.6_{\pm 0.3}$	$19.8_{\pm 0.8}$	$\mathbf{2.0}_{\pm 0.0}$	$1,814$	$1,857$
ACIC-19	$10.3_{\pm 0.5}$	$741_{\pm 128}$	$12.3_{\pm 0.3}$	$17.3_{\pm 0.4}$	$\mathbf{2.1}_{\pm 0.0}$	$1,815$	$1,856$
ACIC-21	$10.5_{\pm 0.3}$	$927_{\pm 38}$	$12.1_{\pm 0.6}$	$38.1_{\pm 18.1}$	$\mathbf{2.0}_{\pm 0.0}$	$1,811$	$1,856$
ACIC-22	$10.1_{\pm 0.5}$	$847_{\pm 110}$	$15.1_{\pm 8.1}$	$49.2_{\pm 1.4}$	$\mathbf{2.4}_{\pm 0.8}$	$1,477$	$1,857$
ACIC-25	$8.2_{\pm 0.6}$	$1,173_{\pm 202}$	$26.3_{\pm 0.2}$	$39.1_{\pm 18.6}$	$\mathbf{3.6}_{\pm 0.1}$	$1,818$	$1,857$
ACIC-32	$8.8_{\pm 0.6}$	$827_{\pm 59}$	$28.0_{\pm 1.9}$	$19.1_{\pm 0.9}$	$\mathbf{3.5}_{\pm 0.3}$	$1,810$	$1,856$
ACIC-33	$9.4_{\pm 0.5}$	$655_{\pm 102}$	$29.9_{\pm 0.2}$	$34.1_{\pm 21.0}$	$\mathbf{2.9}_{\pm 0.0}$	$1,819$	$1,855$
ACIC-35	$13.2_{\pm 0.7}$	$625_{\pm 61}$	$21.6_{\pm 9.5}$	$58.6_{\pm 0.8}$	$\mathbf{2.9}_{\pm 0.0}$	$1,820$	$1,854$
ACIC-36	$22.2_{\pm 17.9}$	$909_{\pm 141}$	$11.5_{\pm 0.5}$	$58.8_{\pm 5.0}$	$\mathbf{2.9}_{\pm 0.0}$	$1,815$	$1,854$
ACIC-39	$32.9_{\pm 2.5}$	$914_{\pm 85}$	$11.9_{\pm 0.5}$	$53.5_{\pm 3.3}$	$\mathbf{2.9}_{\pm 0.1}$	$1,814$	$1,855$
ACIC-41	$13.3_{\pm 12.7}$	$1,392_{\pm 206}$	$12.1_{\pm 0.4}$	$39.2_{\pm 19.4}$	$\mathbf{3.6}_{\pm 0.1}$	$1,813$	$1,854$
ACIC-43	$8.9_{\pm 0.4}$	$1,159_{\pm 398}$	$11.9_{\pm 0.4}$	$19.5_{\pm 1.3}$	$\mathbf{3.6}_{\pm 0.0}$	$1,815$	$1,854$
ACIC-44	$8.5_{\pm 0.3}$	$1,147_{\pm 88}$	$12.7_{\pm 0.5}$	$20.0_{\pm 0.7}$	$\mathbf{3.6}_{\pm 0.1}$	$1,818$	$1,855$
ACIC-45	$5.9_{\pm 0.5}$	$709_{\pm 528}$	$12.7_{\pm 0.5}$	$20.2_{\pm 0.9}$	$\mathbf{3.1}_{\pm 0.3}$	$1,827$	$1,853$
ACIC-48	$11.5_{\pm 0.6}$	$886_{\pm 176}$	$11.6_{\pm 0.2}$	$19.9_{\pm 0.8}$	$\mathbf{2.9}_{\pm 0.1}$	$1,815$	$1,854$
ACIC-51	$8.4_{\pm 0.6}$	$960_{\pm 93}$	$12.3_{\pm 0.3}$	$20.9_{\pm 1.1}$	$\mathbf{3.0}_{\pm 0.1}$	$1,819$	$1,855$
ACIC-54	$10.4_{\pm 0.5}$	$1,414_{\pm 216}$	$15.1_{\pm 8.2}$	$22.8_{\pm 6.5}$	$\mathbf{2.9}_{\pm 0.1}$	$1,818$	$1,853$
ACIC-55	$8.4_{\pm 0.7}$	$1,176_{\pm 293}$	$29.6_{\pm 0.6}$	$22.8_{\pm 11.7}$	$\mathbf{2.9}_{\pm 0.0}$	$1,815$	$1,854$
ACIC-60	$8.5_{\pm 0.2}$	$1,063_{\pm 175}$	$29.9_{\pm 0.2}$	$43.9_{\pm 1.7}$	$\mathbf{2.9}_{\pm 0.0}$	$1,821$	$1,854$
ACIC-62	$17.5_{\pm 10.7}$	$671_{\pm 113}$	$31.3_{\pm 0.5}$	$34.0_{\pm 15.0}$	$\mathbf{2.4}_{\pm 0.3}$	$1,813$	$1,854$
ACIC-70	$36.7_{\pm 0.6}$	$800_{\pm 90}$	$30.4_{\pm 0.9}$	$17.7_{\pm 0.4}$	$\mathbf{2.2}_{\pm 0.0}$	$1,815$	$1,854$
ACIC-72	$18.5_{\pm 1.0}$	$922_{\pm 53}$	$39.2_{\pm 0.7}$	$18.2_{\pm 1.7}$	$\mathbf{2.2}_{\pm 0.0}$	$1,816$	$1,855$
ACIC-75	$29.4_{\pm 2.0}$	$924_{\pm 328}$	$39.1_{\pm 0.4}$	$30.4_{\pm 17.7}$	$\mathbf{2.2}_{\pm 0.0}$	$1,809$	$1,855$
Avg. rank	2.67	5.00	2.83	3.50	1.00	6.00	7.00

E GENERATION OF THE SIMULATED DATASETS

This appendix specifies our multi-treatment extension of the ACIC 2016 design (Dorie et al., 2019) and the two sets of datasets it produces: 1) the ten replicates MT01 to MT10, and 2) the three hand-built datasets Chain, Fork and Collider, which share its covariates and outcome coupling but are otherwise specified by hand. Three elements are inherited from ACIC 2016: a covariate matrix held fixed across every dataset, a random function pool from which all covariate-dependent functions are drawn, and the principle that ground truth is stored rather than estimated.

Based on that, we designed the following components: 1) a DAG among the treatments together with the structural equations that sample them (Appendix E.3), 2) a response surface defined over all $2^{|\mathcal{A}^\star|}$ assignments of the active set $\mathcal{A}^\star$ of treatments, calibrated on the interventional-probability-difference scale (Appendix E.4), and 3) in place of a fixed list of precomputed estimands, a released generative model from which any interventional query can be answered for every unit (Appendix E.7).

E.1 COVARIATES

One covariate matrix is generated once, from a fixed seed, and reused unmodified by every dataset: $n = 4802$ units and $p = 58$ covariates, matching the size and the type mixture of the Collaborative Perinatal Project (CPP) matrix that ACIC 2016 uses.

The construction is a latent factor model. An $n \times 6$ Gaussian factor matrix F is drawn, and every covariate is a loading on F plus independent Gaussian noise, transformed according to its type: 1) 24 continuous columns $FW + \varepsilon$ with $W_{ij} \sim \mathcal{N}(0, 0.7^2)$, of which one in four is exponentiated and one in four rounded to non-negative counts, so that skewed and discrete columns are present; 2) 20 binary columns, obtained by thresholding a second set of loadings at $\mathrm{Unif}(-0.9, 0.9)$, with the resulting proportion of ones between 0.29 and 0.66; and 3) 14 categorical columns with 3 to 5 levels, cut at equiprobable quantiles of a third set. The six factors are the only source of dependence between covariates.

The matrix is therefore designed to have the following features: 1) more of its columns are genuinely continuous, where the original ACIC 2016 covariate matrix is largely coarse integers, and 2) the heavy two-sided skew of the real matrix is absent, which makes the covariate side smoother. Our proposed data generative procedure takes any fixed covariate matrix.

Notably, comparability across datasets requires the matrix to be identical everywhere, as in ACIC 2016, which is why it is generated once rather than per replicate.

The three hand-built datasets, namely, Chain, Fork, and Collider, all use two columns of this matrix, a standardized continuous covariate x_1 and a binary one x_2.

E.2 RANDOM FUNCTION POOL

Every covariate-dependent function in the MT design comes from one generator. This includes 1) the six confounding functions f_k that make treatment assignment depend on X (see Appendix E.3), and 2) the baseline response surface g_0 and each heterogeneity modifier h_{jk} that makes a treatment's effect vary across units (see Appendix E.4).

The pool is therefore the single source of nonlinearity in the benchmark, and this section describes it in full.

Specifically, a draw is a random sparse nonlinear map $\mathbb{R}^p \to \mathbb{R}$, built in five steps.

Select covariates. Pick $p_0 = 8$ covariates uniformly without replacement from the $p = 58$ columns of X. Sparsity is built in as two functions drawn independently will typically share few covariates.

Assign a basis transformation. Each selected covariate receives a basis determined by its type, since a single formula cannot serve all three types.

Covariate type	Basis
Continuous	one of $\{ z,\ \ z^2 - 1,\ \ \min(e^{z/2},\, 20) \}$, drawn uniformly
Binary	$\mathbf{1}\{x = 1\}$
Categorical	$\mathbf{1}\{x = \ell\}$, with the level ℓ drawn uniformly from that column's levels

Notably, here z denotes the standardized column: continuous covariates are centered and scaled before the basis is applied, whereas indicators are formed from raw values.

Select interactions. All $\binom{p_0}{2} = 28$ pairwise products of the selected terms are considered, and each is kept independently with probability 0.30.

Draw coefficients. Independent $\mathcal{N}(0, 1)$ coefficients, one for each main term and one for each retained interaction.

Center and rescale. Writing $\varphi_1, \ldots, \varphi_{p_0}$ for the basis terms and $\mathcal{I}$ for the retained interaction pairs, the unnormalized function is

$$\mathrm{raw}(x) \;=\; \sum_{l=1}^{p_0} \psi_l \varphi_l(x) \;+\; \sum_{(a,b)\in\mathcal{I}} \psi_{ab}\, \varphi_a(x)\varphi_b(x),$$

and the function returned by the pool is

$$f(x) \;=\; \frac{\mathrm{raw}(x) - \overline{\mathrm{raw}}}{\sigma_{\mathrm{raw}}} \cdot \mathrm{sd}_{\text{target}}, \qquad \sigma_{\mathrm{raw}} = \mathrm{sd}(\mathrm{raw}), \tag{59}$$

with both moments taken over the n rows of the fixed covariate matrix and $\mathrm{sd}_{\text{target}} = 1$ throughout.

E.3 MT's treatment model

The treatment side of MT is built in three steps: a DAG G over the $m = 20$ treatments is drawn, two subsets of the treatments are chosen to fix which of them matter for the outcome and which of them depend on the covariates, and the treatments are then sampled one at a time along G. We describe each step in turn.

Structure of the graph. A DAG is drawn by first fixing a uniformly random permutation ζ of the treatment labels $\{1, \ldots, m\}$ and then adding edges only from earlier to later positions, so that acyclicity holds by construction and ζ is a topological order of the resulting graph.

For the treatment at position r of ζ, each of the $r - 1$ treatments before it is included as a parent independently with probability $\min(1,\, 2/r)$. The second treatment therefore always has the first as its parent, and from the third treatment on the expected in-degree is $(r - 1) \cdot 2/r \approx 2$ regardless of position.

Since every admissible edge is an independent draw, G is an Erdős–Rényi graph restricted to the order ζ, with expected in-degree 2 (across the ten datasets used in our experiments, we have 35 to 41 edges). Each edge $j \to k$ of G receives a weight $\beta_{jk} = \pm\,\mathrm{Unif}(0.5, 1.5)$ with a random sign, so that a parent may either encourage (co-prescription) or discourage (substitution) its child. We write $\mathrm{pa}(k)$ for the parents of k in G and $\mathrm{an}_G(A)$ for the ancestors of a set A.

Active set and confounded set. Two subsets of the treatments are then drawn. The *active set* $\mathcal{A}^\star$, with $|\mathcal{A}^\star| = 4$, is the set of treatments that enter the outcome model of Appendix E.4. Every treatment outside $\mathcal{A}^\star$ has zero controlled direct effect (CDE) on Y by construction. It is drawn uniformly among the 4-subsets of $\{1, \ldots, m\}$ and redrawn until at least one edge of G has both endpoints in $\mathcal{A}^\star$, because without such an edge the total and controlled direct effects of every active treatment would coincide. (If no such draw is found in 2000 attempts, an edge is added between the two members of $\mathcal{A}^\star$ that come earliest in ζ; this fallback was not needed in any of the ten replicates.)

The *confounded set* $\mathcal{B}$, with $|\mathcal{B}| = 6$, is the set of treatments whose assignment depends on the covariates through a function f_k of x, and it is constructed to overlap $\mathcal{A}^\star$ in exactly two treatments: two labels are drawn from $\mathcal{A}^\star$ and four from $\{1, \ldots, m\} \setminus \mathcal{A}^\star$, and $\mathcal{B}$ is their union. Hence two active treatments are confounded and two are randomized conditional on their treatment parents (though not marginally, since those parents may themselves lie in $\mathcal{B}$), while the four confounded treatments outside $\mathcal{A}^\star$ are associated with Y through X without a direct effect on it.

Because the latter are drawn from the complement of $\mathcal{A}^\star$ rather than of $\mathcal{A}^\star \cup \mathrm{an}_G(\mathcal{A}^\star)$, some of them may be ancestors of $\mathcal{A}^\star$ and so carry a total effect, and which ones are causally inert is determined by G rather than by a design parameter; Appendix E.5 counts the resulting classes per replicate.

Sampling the treatments. With G, $\mathcal{A}^\star$ and $\mathcal{B}$ fixed, the treatments are sampled ancestrally in the order ζ (so that the parents of T_k are realized before T_k itself) from

$$P\big(T_k{=}1 \mid X = x, T_{\mathrm{pa}(k)} = t_{\mathrm{pa}(k)}\big) = \sigma\Big(\,\mathrm{clip}_\lambda\Big(\alpha_k + f_k(x) + \textstyle\sum_{j\in\mathrm{pa}(k)} \beta_{jk}\, t_j\Big)\Big), \tag{60}$$

where $\lambda = \mathrm{logit}(0.95) \approx 2.944$ and f_k is an independent draw from the pool of Appendix E.2 if $k \in \mathcal{B}$ and $f_k \equiv 0$ otherwise, so that for $k \notin \mathcal{B}$ the conditional law of T_k given its parents does not vary with x.

The clip bounds every propensity to $[0.05, 0.95]$ and is part of the structural equation rather than a post-hoc truncation, so the stored propensities are exact.

Within each treatment, $f_k(X)$ and the parent term determine which units are more likely to be treated, while the intercept α_k shifts the whole propensity distribution and so controls the fraction treated. It is therefore not drawn but solved for: a target treatment rate $q_k \sim \mathrm{Unif}(0.25, 0.6)$ is fixed first, and α_k is set by bisection, conditional on the already-realized parents, so that the sample-average clipped propensity equals q_k. This keeps the marginal treatment rate a controlled knob, decoupled from the edge weights and the strength of confounding, which would otherwise move it as a side effect.

E.4 MT's outcome model

The outcome is binary, and the mechanism model supplies the logit:

$$\mathrm{logit}\, P(Y = 1 \mid X = x, do(T_{\mathcal{A}^\star} = t_{\mathcal{A}^\star})) \;=\; \underbrace{c_0}_{\text{intercept}} \;+\; \underbrace{g_0(x)}_{\text{baseline}} \;+\; \underbrace{s_0 \cdot \tau(t_{\mathcal{A}^\star}, x)}_{\text{treatment effect}}. \tag{61}$$

Only $t_{\mathcal{A}^\star}$ enters. The other sixteen treatments are absent from Y entirely, which is why the potential-outcome table over $2^{|\mathcal{A}^\star|} = 16$ cells is a complete description of the potential outcomes $Y(t_{\mathcal{A}^\star})$.

E.4.1 Baseline $g_0(X)$

Drawn from the same pool as the f_k (eight covariates, polynomial bases, 30% interactions), then bounded smoothly:

$$g_0 \;=\; B \tanh\!\left(\frac{s_g\, \tilde{g}}{B}\right), \qquad B = 3,$$

with the prescale s_g solved per replicate by bisection so that $\mathrm{sd}(g_0) = 1.0$ *after* the bound. The reason is the $\exp(z/2)$ basis in the pool: rescaled to unit standard deviation, one raw draw still reached a logit of $+41$, pinning $P(Y = 1)$ at one (see Appendix E.2).

E.4.2 The treatment term: the mechanism model

Active treatments act on Y through $J = 2$ unobserved mechanisms,

$$M_j \;=\; \sum_{k\in\mathcal{A}^\star} a_{jk}(X)\, T_k, \qquad \tau \;=\; \sum_{j=1}^{J} \phi_j(M_j),$$

built from three ingredients.

A sparse loading matrix. To define M_j, each active treatment is assigned to at least one mechanism, and each remaining slot is filled independently with probability 0.25. The values of all a_{jk} are drawn from $\mathcal{N}(0, 1)$.

Heterogeneous loadings. Each nonzero loading is itself a function of the covariates,

$$a_{jk}(X) \;=\; a_{jk}^{\text{base}}\,(1 + \omega\, h_{jk}(X)), \qquad \omega = 0.5,$$

where h_{jk} is a further pool draw rescaled to unit standard deviation. This is the channel that creates subgroup structure: the effect of T_k varies across units by roughly $\pm 50\%$ and can change sign in the

tails. The loadings are then rescaled so that $\mathrm{sd}(M_j) = 1$ at the observed assignment, which places M in the informative range of the nonlinear links rather than in their flat regions.

Links. Each mechanism draws ϕ_j from

$$\{\,\text{linear: } M, \quad \text{saturating: } 1.5\tanh(M/1.5), \quad \text{non-monotone: } M - 0.30M^2\,\}.$$

Saturating links give diminishing returns when several treatments load on one mechanism; non-monotone links give an interior optimum, that is, overshoot.

The term τ is evaluated for all sixteen cells $t_{\mathcal{A}^\star} \in \{0,1\}^4$ at once, yielding an $n \times 16$ table.

E.4.3 Calibrating c_0 and s_0

We then use two knobs, solved *alternately* for four rounds (as they interact through the logistic link):

- c_0 sets the marginal $P(Y = 1) = 0.4$, by bisection;
- s_0 sets the root-mean-square controlled direct effect to 0.10 on the interventional-probability-difference scale: flipping one active treatment while holding the other three at their observed values moves $P(Y = 1)$ by about ten points on average.

The search over s_0 is a geometric grid bracket followed by bisection rather than plain bisection, as saturation makes the probability difference a non-monotone function of the scale at large values, in which the plain bisection would converge to the wrong root. These two knobs replace the noise standard deviation of a continuous-outcome design.

E.4.4 Potential outcomes by monotone coupling

Writing $p_i(t_{\mathcal{A}^\star}) = \sigma(c_0 + g_0(x_i) + s_0\,\tau(t_{\mathcal{A}^\star}, x_i))$ for unit i and cell $t_{\mathcal{A}^\star} \in \{0,1\}^4$,

$$Y_i(t_{\mathcal{A}^\star}) \;=\; \mathbf{1}\{U_i < p_i(t_{\mathcal{A}^\star})\},$$

with a *single uniform U_i per unit, shared across all sixteen cells*. This is the binary analogue of drawing the noise once and reusing it across combinations, and it carries the same guarantee: a unit's sixteen potential outcomes differ only through the treatment, never through outcome noise. The observed outcome is then a lookup, $Y_i = Y_i(T_{i,\mathcal{A}^\star})$ at the unit's realized cell $T_{i,\mathcal{A}^\star}$.

All estimands are computed on the probability table $p_i(t_{\mathcal{A}^\star})$ rather than on the realized binary outcomes, so the Bernoulli draw contributes no noise whatsoever to the ground truth.

E.4.5 Rejection for detectability

The entire response surface is redrawn, up to thirty times, unless both conditions hold:

- every active treatment has $|\mathrm{CDE}_k| \geq 0.1 \cdot \mathrm{sd}(Y) \approx 0.05$, so that no active treatment is silently unrecoverable;
- $\mathrm{sd}(\tau) \leq 2.0$ on the logit scale, keeping the surface from being effectively deterministic.

The screen prevents accidental degenerated cases: for instance, random loadings of opposite sign on the two mechanisms can cancel an active treatment to nothing. The dataset MT01 used in our experiment required eight attempts.

E.5 The ten replicates used in our experiments

MT01 to MT10 are ten independent replicates of the single configuration above ($m = 20$, $|\mathcal{A}^\star| = 4$, $|\mathcal{B}| = 6$, $|\mathcal{B} \cap \mathcal{A}^\star| = 2$, expected in-degree 2), differing in the graph, the sets, and every random function.

Because inactive treatments may be parents of active ones, each replicate contains three classes of treatment: 1) the 4 active ones ($\mathcal{A}_{\mathrm{dir}} = \mathcal{A}^\star$, nonzero controlled direct effect); 2) the *indirect* ones in $\mathrm{an}_G(\mathcal{A}^\star) \setminus \mathcal{A}^\star$, which have a total but no direct effect and are therefore correct answers for a method that targets total effects and wrong answers for one that targets direct effects ($\mathcal{A}_{\mathrm{tot}} = \mathcal{A}^\star \cup \mathrm{an}_G(\mathcal{A}^\star)$); and 3) the inert ones, of which those in $\mathcal{B}$ are *distractors*, associated with Y through X but causally irrelevant. Table 11 reports the resulting structure and support.

Table 11: The ten MT replicates used in our experiments. $|\mathcal{A}_{\mathrm{tot}}|$ counts treatments with a nonzero total effect, of which $|\mathcal{A}^\star| = 4$ are active and the rest indirect. Distractors are confounded but causally inert. Within-$\mathcal{A}^\star$ edges are the edges of G with both endpoints active, which make total and direct effects differ. The last two columns are the smallest count among the 16 cells of $t_{\mathcal{A}^\star}$ and the minimum over units of $P(t_{\mathcal{A}^\star} \mid x_i)$ at the observed assignment.

Dataset	$\lvert\mathcal{A}_{\mathrm{tot}}\rvert$	indirect	distractors	within-$\mathcal{A}^\star$ edges	links	$P(Y{=}1)$	min. cell	$\min_i P(t_{\mathcal{A}^\star} \mid x_i)$
MT01	11	7	3	1	sat., lin.	0.404	84	0.0037
MT02	16	12	2	1	lin., lin.	0.392	98	0.0027
MT03	8	4	4	1	sat., lin.	0.400	92	0.0018
MT04	11	7	2	1	lin., lin.	0.410	41	0.0015
MT05	10	6	2	2	non., sat.	0.387	76	0.0037
MT06	13	9	1	1	sat., lin.	0.407	98	0.0035
MT07	10	6	2	1	sat., lin.	0.399	68	0.0020
MT08	12	8	0	2	lin., non.	0.399	68	0.0020
MT09	13	9	1	1	lin., lin.	0.415	108	0.0110
MT10	14	10	1	1	non., sat.	0.395	68	0.0020

E.6 Chain, Fork and Collider

These three datasets are the smallest instance that separates total from controlled direct effects. They use $n = 4802$ units, the two covariates of Appendix E.1, three binary treatments and one binary outcome.

However, no random function is drawn from the pool; instead, every coefficient is fixed by hand, all quantities are exact by enumeration of the $2^3 = 8$ treatment cells, and the three datasets share one outcome model,

$$\operatorname{logit}\mu(x,t) = -0.4 + 0.9\,x_1 - 0.7\,x_2 + 0.8\,t_1 + 1.0\,t_2 - 1.0\,t_3,$$

so that they differ *only* in the graph among the treatments:

$$\begin{aligned}
\textbf{Chain } (T_1 \to T_2 \to T_3): \quad & \operatorname{logit} P(T_1{=}1) = -0.2 + 0.8x_1 + 0.6x_2,\\
& \operatorname{logit} P(T_2{=}1) = -0.3 + 0.7x_1 + 2.0\,t_1,\\
& \operatorname{logit} P(T_3{=}1) = -0.4 + 0.5x_2 + 1.6\,t_2,\\
\textbf{Fork } (T_1 \leftarrow T_2 \to T_3): \quad & \operatorname{logit} P(T_2{=}1) = -0.2 + 0.8x_1 + 0.6x_2,\\
& \operatorname{logit} P(T_1{=}1) = -0.3 + 0.7x_1 + 2.0\,t_2,\\
& \operatorname{logit} P(T_3{=}1) = -0.4 + 0.5x_2 + 1.6\,t_2,\\
\textbf{Collider } (T_1 \to T_3 \leftarrow T_2): \quad & \operatorname{logit} P(T_1{=}1) = -0.2 + 0.8x_1,\\
& \operatorname{logit} P(T_2{=}1) = -0.3 + 0.7x_2,\\
& \operatorname{logit} P(T_3{=}1) = -0.4 + 0.5x_1 + 2.0\,t_1 + 1.6\,t_2.
\end{aligned}$$

Treatments are sampled in topological order and the outcome uses the same coupling as MT, one uniform per unit across the eight cells. Marginal proportions of ones for treatments are 0.46 to 0.71, and $P(Y{=}1)$ is 0.448, 0.453 and 0.391, respectively.

E.7 Stored ground truth and interventional queries

Each dataset is released as (X, T, Y) only: the mechanisms, the potential-outcome table, the propensities and the uniforms U_i are withheld. Alongside it we store the generative model in machine-readable form, i.e., the graph and its edge weights, each pool draw as a term list with the centering and scaling constants that fix the map $x \mapsto f(x)$, the loadings, links, bounds and clips, the outcome intercept and scale, and the per-unit uniforms; therefore, for any subset A of treatments,

$$P\big(Y{=}1 \mid X = x_i, do(T_A{=}t_A)\big) = \sum_{t_{\mathcal{A}^\star}} \mu(x_i, t_{\mathcal{A}^\star})\, P\big(T_{\mathcal{A}^\star}{=}t_{\mathcal{A}^\star} \mid X = x_i, do(T_A{=}t_A)\big) \tag{62}$$

can be evaluated per unit.