Language Models as Causal Effect Generators

Lucius E.J. Bynum
New York University
lucius@nyu.edu

Kyunghyun Cho
New York University
kyunghyun.cho@nyu.edu

Abstract

In this work, we present sequence-driven structural causal models (SD-SCMs), a framework for specifying causal models with user-defined structure and language-model-defined mechanisms. We characterize how an SD-SCM enables sampling from observational, interventional, and counterfactual distributions according to the desired causal structure. We then leverage this procedure to propose a new type of benchmark for causal inference methods, generating individual-level counterfactual data to test treatment effect estimation. We create an example benchmark consisting of thousands of datasets, and test a suite of popular estimation methods for average, conditional average, and individual treatment effect estimation. We find under this benchmark that (1) causal methods outperform non-causal methods and that (2) even state-of-the-art methods struggle with individualized effect estimation, suggesting this benchmark captures some inherent difficulties in causal estimation. Apart from generating data, this same technique can underpin the auditing of language models for (un)desirable causal effects, such as misinformation or discrimination. We believe SD-SCMs can serve as a useful tool in any application that would benefit from sequential data with controllable causal structure.

1 Introduction

Reasoning about counterfactuals plays a fundamental role in understanding cause and effect, both in theory and in practice. Unfortunately, counterfactuals are also fundamentally unobservable (Holland, 1985) and must always be simulated. In this work, we leverage language models (LMs) to help simulate counterfactual data in a user-controlled manner. To achieve this, we borrow the conditional distributions of a pre-trained LM in order to parameterize a structural causal model, based on an input directed acyclic graph (DAG) over variables expressed in natural language. This procedure allows us to simulate true counterfactual data — to observe both potential outcomes — but, crucially, *without manually specifying functional relationships between variables*. Instead, the specification of structural equations becomes data-driven. We explore how this data-driven approach can enable the specification of causal models for complex settings with less reliance on human expertise or creativity to manually specify relationships between variables.

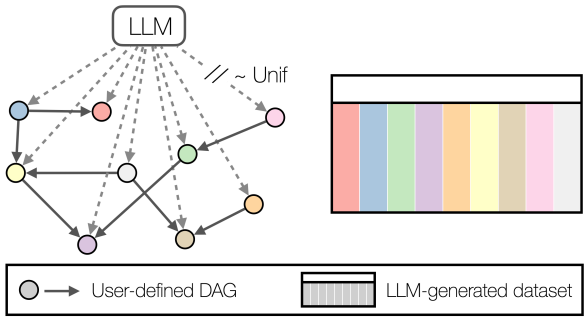


Figure 1: Illustration of a sequence-driven structural causal model (SD-SCM), which uses a language model to sample data according to a user-specified DAG. Any variables whose values are sampled from the language model will potentially share the language model as a common cause (dashed arrows), unless sampled manually, e.g., uniformly.

Many use-cases are possible for sequential data (like text) with controllable causal structure. The main use-case we explore in this work is the development of a new type of benchmark for causal inference — a benchmark for conditional average and individual treatment effect estimation, where neither the counterfactual outcomes $\tilde{y}$ nor the treatment assignments $\tilde{t}$ are manually generated. This stands in contrast to existing causal inference benchmarks that must always manually generate $\tilde{y}$ or $\tilde{t}$, even if covariates are based on real data (see, e.g., Louizos et al. (2017)). We find that data generated using our procedure is indeed useful for this task and chal-

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 2096–2115
November 4-9, 2025

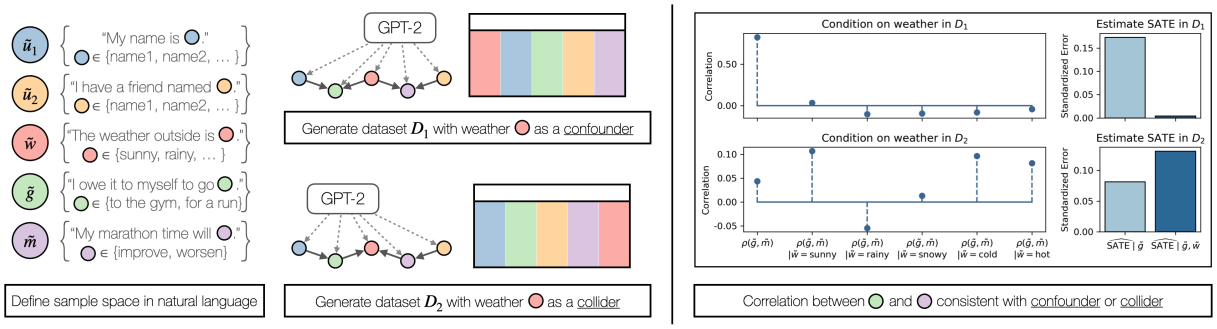


Figure 2: Toy example showing SD-SCMs that use GPT-2 (Radford et al., 2019) to generate observational and counterfactual data corresponding to user-specified DAGs. In one case, the red node (weather, $\tilde{w}$) is a confounder. In the other case, $\tilde{w}$ is a collider. Plots on the right show that despite possible effects from dashed arrows, and DAGs that may contradict what we expect to happen in the real world, the generated data are indeed consistent with $\tilde{w}$ as a confounder or a collider.

lenges state-of-the-art estimation methods across both conditional average treatment effect (CATE) and individual treatment effect (ITE)[1] estimation.

1.1 Contribution.

1. We define **a procedure for turning any language model and DAG into a sequence-driven structural causal model (SD-SCM)**. Section 3 characterizes how an SD-SCM provides access to observational, interventional, and counterfactual distributions over sequential data according to the desired DAG.
2. In Section 4, we use SD-SCMs to create **an example benchmark for causal effect estimation** and test a suite of popular estimation methods across CATE and ITE estimation. We find that our benchmark challenges state-of-the-art estimation methods. All benchmark datasets as well as code for SD-SCM-based data generation is available on GitHub.[2]
3. Section 5 demonstrates how **this same technique can underpin auditing language models for (un)desirable causal effects**.

Before describing our framework formally, we provide a toy example that illustrates the main points.

Example 1 (Improving your marathon time at the gym)**.** *In this toy example, we use a language model to sample observational and counterfactual data corresponding to two imagined scenarios, each represented by a DAG. The variables we consider will be represented via sets of sequences, where each set can be viewed as a sample space:*

- $\tilde{u}_1$ *sample space: "My name is* x*." for all* $x \in$ *{John, Jane, Alice, Bob, Charlie}*
- $\tilde{u}_2$ *sample space: "I have a friend named* x*." for all* $x \in$ *{John, Jane, Alice, Bob, Charlie}*
- `weather` $(\tilde{w})$ *sample space: "The weather outside is* x*." for all* $x \in$ *{sunny, rainy, snowy, cold, hot}*
- `gymOrRun` $(\tilde{g})$ *sample space: "I owe it to myself to go* x*." for* $x \in$ *{to the gym, for a run outside}*
- `marathonTime` $(\tilde{m})$ *sample space: "After this, my marathon time will* x*." for* $x \in$ *{improve, worsen}*

The difference between the two scenarios is what we choose to do with the weather variable $\tilde{w}$*. In the first case, we choose DAG* $\mathcal{G}_1$ *where* w *is a confounder* ($\tilde{u}_1 \to \tilde{g} \leftarrow \tilde{w} \to \tilde{m} \leftarrow \tilde{u}_2$)*. In the second case, we choose DAG* $\mathcal{G}_2$ *where* $\tilde{w}$ *is instead a collider* ($\tilde{u}_1 \to \tilde{g} \to \tilde{w} \leftarrow \tilde{m} \leftarrow \tilde{u}_2$)*. Notice we have full control over the DAG we choose, regardless of what we might expect to happen in the real world or be encoded by the language model (where, for example, we would not expect going to the gym to have any impact on the weather). Each of these DAGs, by definition, induces a corresponding factorization of the joint distribution across the 5 variables. The factorization for* $\mathcal{G}_1$ *is* $P(\tilde{g}|\tilde{w},\tilde{u}_1)P(\tilde{m}|\tilde{w},\tilde{u}_2)P(\tilde{w})P(\tilde{u}_1)P(\tilde{u}_2)$ *and the factorization for* $\mathcal{G}_2$ *is* $P(\tilde{w}|\tilde{g},\tilde{m})P(\tilde{g}|\tilde{u}_1)P(\tilde{m}|\tilde{u}_2)P(\tilde{u}_1)P(\tilde{u}_2)$.

Simulating an observational study: *Our procedure will use a language model to define each of these conditional distributions, instead of defining them manually. In order to observe data that fol-*

[1] It is common to draw no distinction between ITEs and CATEs (Vegetabile, 2021), but we emphasize these two quantities as distinct: the CATE is the conditional expectation of the ITE, which typically will not explain all ITE variation (Lei and Candès, 2020).

[2] https://github.com/lbynum/sequence-driven-scms

lows the correct structure, we iteratively sample each variable in ancestral order according to the desired DAG, ***allowing each variable to see only the text of its parents as input****. Doing this allows us to use whatever correlations the language model has encoded to define structural equations. For example, to sample a single observation corresponding to $\mathcal{G}_1$, the following five phrases (one for each covariate) are sampled using the language model, where [bracketed text] is filled in by querying the model across the corresponding sample space:*

- *$\tilde{u}_1$ sample: "My name is [Charlie]."*
- *$\tilde{u}_2$ sample: "I have a friend named [Alice]."*
- *$\tilde{w}$ sample: "The weather outside is [cold]."*
- *$\tilde{g}|\tilde{u}_1, \tilde{w}$ sample: "My name is Charlie. The weather outside is cold. I owe it to myself to go [to the gym]."*
- *$\tilde{m}|\tilde{u}_2, \tilde{w}$ sample: "I have a friend named Alice. The weather outside is cold. After this, my marathon time will [improve]."*

These five text completions correspond to a single observation, where possible values in each sample space are represented by their index. In other words, we have just observed the data point ($\tilde{u}_1$, $\tilde{u}_2$, $\tilde{w}$, $\tilde{g}$, $\tilde{m}$) = (4, 2, 3, 0, 0) sampled using DAG $\mathcal{G}_1$. Figure 2 shows the result of repeating this process for 1000 observations with $\mathcal{G}_1$ and 1000 observations with $\mathcal{G}_2$ using GPT-2 (Radford et al., 2019) as the language model. For $\mathcal{G}_1$, we would expect the magnitude of correlation ρ between $P(\tilde{g} = 1)$ and $P(\tilde{m} = 1)$ to decrease if we condition on confounder $\tilde{w}$. By contrast, for $\mathcal{G}_2$, where $\tilde{w}$ is instead a collider, we would expect the magnitude of ρ to instead increase if we condition on $\tilde{w}$. Figure 2 shows that this is indeed the case — our sampled data reflects the desired causal structure.

Simulating counterfactual data: *We can use a similar procedure to simulate interventions instead of observations: we intervene by manually setting an action (in this case, the value of covariate $\tilde{g}$), and* ***we create a counterfactual outcome by additionally setting exogenous variables $\tilde{u}_1, \tilde{u}_2$ and any observed non-descendants of $\tilde{g}$ — $\{\tilde{w}\}$ for $\mathcal{G}_1$ and $\varnothing$ for $\mathcal{G}_2$****. In Section 3, we formally define the correspondence of this process to counterfactual versus interventional distributions. This allows us to directly simulate a counterfactual outcome for each of the observed units, choosing $\tilde{g} = 1$ or $\tilde{g} = 0$ during sampling to generate each unit's potential outcomes. We can then, for example, test how well a treatment effect estimation method will perform* ***if the estimation method is given only the observational data****, i.e., data without any intervention. The right side of Figure 2 shows prediction error in standard deviation units when using a random forest to predict the sample average treatment effect (SATE) with $P(\tilde{m} = 1)$ as the outcome, either using treatment $\tilde{g}$ as the only covariate, or using both $\tilde{g}$ and $\tilde{w}$. As we would expect, including a confounder leads to more accurate effect estimation, while including a collider does not. This demonstrates in a simple way the utility of controlled causal data generation — we can benchmark effect estimation approaches in different settings of interest.*

Benchmarking CATE and ITE estimation: *Many realistic datasets exist for benchmarking estimation of average treatment effects (ATEs), because ATEs are often feasible to isolate with proper study design. However, there is a lack of such data for benchmarking CATE and ITE estimation, where either the outcomes or treatment assignments must always be manually generated.* ***The key benefit of simulating data this way is that individual-level counterfactual data are observable and controllable****. This allows us to not only test ATE estimation methods like in Figure 2, but more importantly to benchmark individual-level effect estimation.*

In the remaining sections, we formalize our procedure beyond this toy example and demonstrate how it can be used to generate more complex data that challenges state-of-the-art causal inference methods across both CATE and ITE estimation.

2 Related work

Causal inference benchmarks and evaluation. Curth et al. (2021) lay out four categories of commonly-used methods for semi-synthetic data generation with known causal effects: (1) simulating treatment effects using real baseline outcomes (Knaus et al., 2021); (2) using real covariates but simulating response surfaces (Wendling et al., 2018; Franklin et al., 2014; Hill, 2011); (3) performing biased sampling of randomized data (Gentzel et al., 2021; Dehejia and Wahba, 1999); and (4) constructing (proxies of) counterfactuals and interventions from real or empirical data (Louizos et al., 2017; Gentzel et al., 2019). The paradigm of fitting models to real data and then sampling synthetic data from the fit models is common in many works (Schuler et al., 2017; Neal et al., 2020). In this area, the most closely related works to ours in spirit are those that fit

generative models to real datasets such that treatments, outcomes, and covariates — in effect, entirely new datasets — can be sampled, such as Athey et al. (2024) and Neal et al. (2020). While such methods are similar in that they rely on generative models, they are fundamentally different from ours, as they are based on individual datasets that already exist (and already have a fixed causal structure), rather than allowing for arbitrary causal structures to be imagined by a user and then parameterized by a generative model. Our setup is akin to a high-fidelity simulation environment (McDuff et al., 2022) that provides empirical counterfactual data (Gentzel et al., 2019), but without needing to manually design all aspects of the simulation, and in a manner that is instead based on natural language. This work is also loosely related to methods that parameterize structural causal models (SCMs) with generative models or other deep learning components, such as Pawlowski et al. (2020); Sanchez and Tsaftaris (2022), but such methods are geared towards counterfactual inference and learning causal relationships from existing data, rather than flexible data generation.

Language models and causal inference. Our work is not the first to suggest that language models can generate outputs that have casual structure. Many works aim to augment language models with the ability to generate counterfactual data (Chatzi et al., 2024; Li et al., 2023; Betti et al., 2023; Hao et al., 2021; Gat et al., 2023). Counterfactuals and causal reasoning are useful across various natural language processing (NLP) tasks, making this capability of particular interest for ongoing LM research (Wang et al., 2024), and language models with causal reasoning capabilities have a wide variety of applications both within and beyond NLP (Vashishtha et al., 2023; Jin et al., 2023a; Zecevic et al., 2023; Liu et al., 2024; Feder et al., 2021; Kıcıman et al., 2023; Jin et al., 2023b; Gat et al., 2023). We are also not the first to point out that counterfactual data generation with language models is useful for understanding the internal 'world model' constructed by an LM and auditing for bias (Fryer et al., 2022). The most similar works to ours that we know of are the contemporaneous works Chatzi et al. (2024) and Ravfogel et al. (2024), which also model counterfactuals in LMs using SCMs. These works focus on how to generate counterfactual strings after *network interventions* within the LM itself. To achieve this, they leverage the Gumbel-Max trick to infer the noise responsible for generating an input and reuse the same noise (or an inferred noise distribution) to generate a corresponding counterfactual output. Our work is fundamentally different in two key ways. First, we consider *semantic interventions* rather than network interventions, i.e., modeling causal relationships and counterfactuals all within a semantically meaningful simulation based on a fixed LM. Second, we control the causal structure of the data generation process, taking a DAG as input and generating data according to that DAG.

In more general terms, *we focus on how to generate data given a desired causal structure*. This capability has important use-cases for downstream tasks like the ones we demonstrate here — generating treatment effects to benchmark effect estimation methods and testing for encoded effects. But more broadly, we provide a generalization of how sequence data and structural causal models can be combined in order to flexibly generate observational, interventional, and counterfactual data for whatever purpose it might be useful.

3 Controlled causal data generation via language model

In this section, we briefly describe how SD-SCMs enable sampling from observational, interventional, and counterfactual distributions according to the desired causal structure. The full set of definitions, notation, and algorithms for SD-SCMs using structural causal models can be found in Appendix A.

We define a **sequence variable** $\tilde{x}$ as a random variable whose sample space $\Omega_{\tilde{x}}$ is a set of sequences. We then define an SD-SCM as a 5-tuple $\mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}, \tau)$, where $\mathbf{V}$ is a set of finite-domain endogenous/observed sequence variables and $\mathbf{U}$ a set of finite-domain exogenous/unobserved sequence variables; $\mathcal{G}$ is a DAG over the variables $\tilde{x}_i$ in $\mathbf{V} \cup \mathbf{U}$ where $\mathbf{PA}_i \subseteq (\mathbf{V} \cup \mathbf{U}) \setminus \{\tilde{x}_i\}$; $\mathcal{P}$ is a language model trained on prior inputs C whose vocabulary $\mathbb{V}$ contains all tokens used in $\Omega_{\mathbf{V}}, \Omega_{\mathbf{U}}$; and τ is an arbitrary fixed topological ordering of $\mathbf{V} \cup \mathbf{U}$ consistent with $\mathcal{G}$.

The general procedure for sampling data from an SD-SCM relies on two simple ideas: (1) creating concatenated prior inputs for each variable using only the sequences of its parents, which we term **parent-only concatenation**, and (2) restricting the domain of the LM over the current variable's sample space, termed **domain-restricted sampling**.

Sampling proceeds in topological order according to τ, which is required in order to break ties between parents, since LMs can be sensitive to even small changes in phrasing. The key difference between an SCM and an SD-SCM is that all variables have at least one common ancestor — the prior inputs C that were used to train the language model.[3]

Observational samples. Observational data are sampled using parent-only concatenation and domain-restricted sampling for each variable according to τ (Appendix A.2).

Interventional samples. The sequence-driven interventional distribution, given $\text{do}(\tilde{v}_i = v)$ as the intervention, samples data in the same manner as observational sampling, but now with variable $\tilde{v}_i$ replaced by value v during sampling (Definition A.8).

Algorithm 1 A single SD-SCM sample from the counterfactual distribution given observation $\mathbf{s}_{\text{obs}}$

Inputs: $\mathbf{s}_{\text{obs}} = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1, \ldots, v_{|\mathbf{V}|}\right)$
$\text{do}(\tilde{v}_i = v), \mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}(\cdot), \tau)$
Returns: $\mathbf{s}^* = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1^*, \ldots, v_{|\mathbf{V}|}^*\right)$
$\mathbf{s}^* \leftarrow (u_1, \ldots, u_{|\mathbf{U}|})$
$\text{ND}_i \leftarrow$ non-descendants of $\tilde{v}_i$ in $\mathcal{G}$
for $\tilde{x}_t \in \tau \backslash \mathbf{U}$ **do**
  if $\tilde{x}_t \equiv \tilde{v}_i$ **then**
    $x_t \leftarrow v$
  end
  else if $\tilde{x}_t \in ND_i$ **then**
    $x_t \leftarrow \mathbf{s}_{\text{obs}}[t]$
  end
  else
    $\text{PA}_\tau \leftarrow \{t' : \tilde{x}_{t'} \in \text{PA}_{\tilde{x}_t}\}$ ordered by τ
    $x_{\text{PA}_\tau} \leftarrow \bigoplus_{x \in \mathbf{s}^*[\text{PA}_\tau]} x$
    $\mathbf{p}_{\tilde{x}_t} \leftarrow []$
    for $k \in 1, \ldots, |\Omega_{\tilde{x}_t}|$ **do**
      $x \leftarrow \Omega_{\tilde{x}_t}[k]$
      $\mathbf{p}_{\tilde{x}_t}[k] \leftarrow \mathcal{P}\left(x_{\text{PA}_\tau} \oplus x\right)$
    end
    $P_{\text{tot}} \leftarrow \sum_k \mathbf{p}_{\tilde{x}_t}[k]$
    $j \sim \text{Multinomial}(\mathbf{p}_{\tilde{x}_t} / P_{\text{tot}})$
    $x_t \leftarrow \Omega_{\tilde{x}_t}[j]$
  end
  $\text{append}(\mathbf{s}^*, x_t)$
end
return $\mathbf{s}^*$

[3] It would also be possible to *train* an LM to induce distributions over the desired variables given this setup, which we leave to future work.

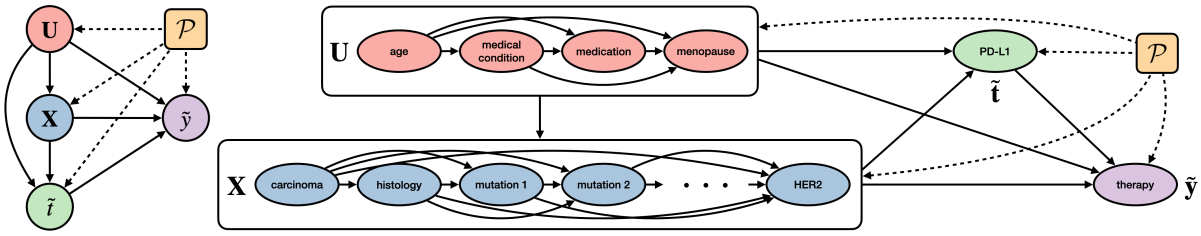
Counterfactual samples. Counterfactual samples require some additional steps. In order to admit unique answers to counterfactual queries, we define abduction for an SD-SCM given evidence $\mathbf{Z} = \mathbf{z}$ as the setting of values $\mathbf{U} = \mathbf{u}$ as well as any evidence in $\mathbf{Z}$ upstream of the intervention. In order to obtain such values $\mathbf{u}$, one needs access to more than just the endogenous variables $\mathbf{V}$ and language model $\mathcal{P}$ — obtaining $\mathbf{u}$ requires performing bookkeeping *during the data generation process*.[4] Because our primary application of SD-SCMs in this work is data *generation*, such bookkeeping is possible in all our use cases. Algorithm 1 shows our procedure for sampling a counterfactual for intervention $\text{do}(\tilde{v}_i = v)$ given observed unit $\mathbf{s}_{\text{obs}} = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1, \ldots, v_{|\mathbf{V}|}\right)$ (see also Definition A.9 for additional discussion).

4 Generating a benchmark for causal effect estimation

To design an SD-SCM-generated benchmark, we focus on the fully sequential DAG structure shown in Figure 3a. Exogenous variables $\mathbf{U}$ precede covariates $\mathbf{X}$, which in turn precede treatment $\tilde{t}$. All variables precede outcome $\tilde{y}$. Recall that the presence of an edge in a DAG allows for *the possibility of a relationship*, but it is the structural equations that determine whether or not a given relationship is meaningful. The strongest assumptions encoded by a DAG, then, are those edges that are *not present*. Our goal here is to have a language model $\mathcal{P}$ make as many 'decisions' about the data generating process as possible. We thus choose this fully-connected structure as a means of letting $\mathcal{P}$ define whichever structural equations are meaningful or not given a topological order, and focus on the edge $\tilde{t} \rightarrow \tilde{y}$ as the target for effect estimation. The key criterion we consider for a useful benchmark is that the datasets we generate require the use of causal reasoning (e.g., controlling for confounding) to recover the effect of $\tilde{t}$ on $\tilde{y}$. Specifically, we aim to generate data for which the observational and interventional distributions are different, i.e., $P^{\mathfrak{B}}_{\tilde{y}|\tilde{t}=t} \neq P^{\mathfrak{B};\text{do}(\tilde{t}=t)}_{\tilde{y}}$. This criterion is not directly in our control given a fixed language model $\mathcal{P}$.[5] However, even with fixed $\mathcal{P}$ and a fixed DAG, we

[4] This is a restatement of the fact that computing point counterfactuals in SCMs requires causal mechanisms that are invertible with respect to the noise variables in order to uniquely reconstruct the noise that produced a given observation.

[5] We discuss applications of this same idea to *training* a model in Section 6.



(a) (b)

Figure 3: **(a)** A useful DAG structure for an SD-SCM-generated estimation benchmark. **(b)** Visual depiction of the structure in (a) used to create the breast cancer SD-SCM in Section 4.1.

find we are able to achieve it through our choice of sample spaces $\Omega_{\mathbf{U}}$, $\Omega_{\mathbf{X}}$, $\Omega_{\tilde{t}}$, $\Omega_{\tilde{y}}$.

4.1 Breast cancer SD-SCM

We define an SD-SCM over 14 variables in order to explore the effect of a tumor's PD-L1 expression levels on different breast cancer therapy plans. Our goal with this SD-SCM is to induce causal structure that can challenge estimation algorithms. Covariates in the breast cancer SD-SCMs are defined in full detail in Appendix B and correspond to the DAG in Figure 3b. For each covariate, 10 different phrasings are considered, resulting in a sample space of 10^{14} possible sequences.[6] We consider 50 different SD-SCM variations, where the sample space for a given SD-SCM is defined by choosing a randomly sampled phrasing from among the possible phrasings for each of the 14 covariates. Then, for each of the 50 SD-SCMs, 20 datasets of size 1,000 are sampled, for a total of 1,000 datasets per language model. We show results for GPT-2 (Radford et al., 2019) and Llama-3-8b (Dubey et al., 2024), but we emphasize that the language model is a fully modular component, and thus other language models can be used. For the results shown here, we use $\text{logP}(\tilde{y} = 0)$ as the outcome, as there are frequently individual-level effects in probability space. See Appendix B for example plots of features, propensity scores, and ITE distributions of the generated data using different possible outcomes. We find that how similar $P_{\tilde{y}}^{\mathfrak{B};\text{do}(\tilde{t}=t)}$ is to $P_{\tilde{y}|\tilde{t}=t}^{\mathfrak{B}}$ varies across SD-SCMs, which we explore further by comparing the performance of observational versus casual estimation approaches.

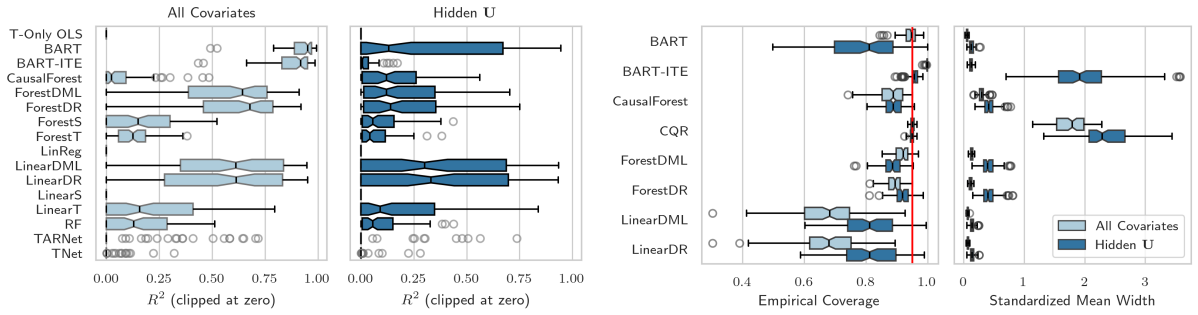
4.2 Effect estimation results

We compare the performance of several effect estimation algorithms. As a naive baseline, ordinary least squares using only the treatment $\tilde{t}$ is considered (**T-Only OLS**). Against this baseline, we consider several causal inference methods of different types, including the causal forest (Wager and Athey, 2018; Athey et al., 2019) (**CausalForest**) and two double machine learning methods for CATE estimation, one linear (**LinearDML**) and one non-parametric (**ForestDML**) (Chetverikov et al., 2016; Athey et al., 2019; Nie and Wager, 2021; Chernozhukov et al., 2017; Foster and Syrgkanis, 2023; Mackey et al., 2018; Battocchi et al., 2019). We also include two doubly robust meta-learning methods (Künzel et al., 2019), again, one linear (**LinearDR**) and one non-parametric (**ForestDR**), and add Bayesian additive regression trees (**BART**) (Hill, 2011; Chipman et al., 2008) as a widely-used Bayesian non-parametric example. To represent simpler methods we include linear and non-parametric S- and T-learners (**LinearS**, **LinearT**, **ForestS**, **ForestT**). As points of reference for NN-based CATE estimation methods, we include an NN-based T-learner (**TNet**), and the NN-based **TARNet** (Shalit et al., 2017). Additional baselines include a random forest baseline (**RF**) that fits a single response surface and directly predicts treatment effects for each unit, and a linear regression baseline (**LinReg**) that takes the conditional mean difference (the fit coefficient on $\tilde{t}$) to be the effect. Finally, we include two methods that target ITEs specifically. One method uses BART posterior draws specifically for ITEs instead of CATEs (**BART-ITE**), and the other is conformalized counterfactual quantile regression (**CQR**) (Lei and Candès, 2020), which provides conformal inference-based interval estimates of ITEs.

All methods are fit using the default settings of their publicly-available implementations.[7] While

[6] Language models are also used to generate the phrasings, but we leave full automation of this process to future work.

[7] The causal forest, DML, and DR implementations are provided by (Battocchi et al., 2019), the BART methods by (Dorie and Hill, 2020), the NN-based methods by (Curth et al., 2021; Curth and van der Schaar, 2021b,a) and CQR by (Lei and Can-



(a) (b)

Figure 4: CATE and ITE estimation on SD-SCM datasets of size 10,000 generated using Llama-3-8b. **(a)** R^2 values across all methods that provide point estimates. **(b)** Empirical coverage ($\alpha = 0.05$) and interval width (in outcome standard deviation units) for methods that provide intervals. Nominal coverage of 95% is indicated by the red line.

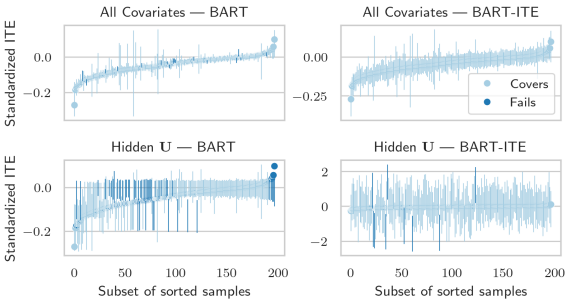


Figure 5: Interval estimates of CATEs/ITEs from BART versus BART-ITE on an example SD-SCM dataset.

additional hyperparameter tuning, etc. could be performed for several methods on a case-by-case basis, this section demonstrates what estimation results we get off-the-shelf.

A note on identification. Estimation algorithms are designed to work when identification assumptions are met, many of which are untestable. In this section, we demonstrate how SD-SCMs can provide a playground to empirically test how algorithms perform not only in ideal conditions but also when untestable assumptions are not met. This is particularly relevant for CATEs and ITEs, where, for example, we might not expect to measure all relevant covariates for each individual unit. In other words, in practice we might not expect to satisfy ignorability. We consider two settings to explore this question empirically. In the 'All Covariates' setting, all 14 covariates are observed. In 'Hidden U,' $\mathbf{U} = \{\tilde{u}_1, \tilde{u}_2, \tilde{u}_3, \tilde{u}_4\} = \{$age, medical conditions, medication, menopausal status$\}$ is hidden.

dès, 2020). All code to reproduce this benchmark is available at https://github.com/lbynum/sequence-driven-scms.

4.2.1 Average treatment effects

Though we focus on CATE and ITE estimation, we first confirm in Appendix C.1 that methods can recover the ATE. We find that (1) there is a meaningful gap between casual and observational methods and that (2) estimation performance does indeed drop significantly when $\mathbf{U}$ is hidden.

4.2.2 CATE and ITE estimation

To lessen the impact of finite-sample issues, we test on datasets of size 10,000, aggregated within each SD-SCM variation. We show results for Llama-3-8b-generated data in this section, but find similar trends with GPT-2 as well as with dataset size 1,000 in Appendix C. Figure 4a shows R^2 values clipped at zero across all methods that provide point estimates for CATEs. When all covariates are observed, BART explains the most CATE variation, while DML and DR methods do as well at times but with a much lower average. However, CATE estimation becomes much more challenging for all methods with hidden $\mathbf{U}$, where no methods perform well. Figures 11 and 12 in Appendix C show the same results in terms of PEHE (Precision in Estimating Heterogeneous Effects) (Hill, 2011), revealing that with no clipping, BART-ITE shows large outliers with hidden $\mathbf{U}$ and NN methods show large outliers in both settings.

When the ITE varies due to covariates not conditioned on in the CATE, as in the hidden $\mathbf{U}$ setting, the two quantities are distinct. In such cases, uncertainty is especially important. Figure 4b shows empirical coverage results for all estimators that provide intervals. With all covariates, empirical coverage is under nominal for all methods that target CATE, except BART. Hidden $\mathbf{U}$ increases uncertainty, but also brings coverage closer to nominal

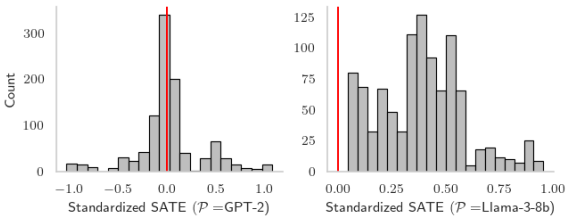


Figure 6: SATEs across all breast cancer SD-SCMs for outcome $P(\texttt{therapy} = 2)$.

for several methods, like LinearDML and both DR methods. CQR remains at or above nominal coverage, but with much wider intervals, as does BART-ITE in the ITE setting. Figure 5 demonstrates this further, comparing intervals for BART targeting CATE versus BART-ITE on an example dataset. With all covariates (top row), intervals from either method are informative about individual-level effects. However, under hidden $\mathbf{U}$ (bottom row), the tighter intervals of BART targeting the CATE are overconfident with variable coverage, and the wider intervals of BART-ITE are so wide as to be vacuous, even if we want just a ranking of the ITEs.

Takeaways. We summarize a few takeaways for off-the-shelf estimation performance in this example. The first is that **linear and tree-based methods are often able to perform well**. Second, in a real-world setting, where a method might often be used with its default parameters, **stability can be important** (e.g., the NN method performance suffers often due to lack of stability). The third takeaway is that **hidden confounding has a big impact**, across all methods. Even methods that perform particularly well with all covariates (like BART) suffer significantly under hidden $\mathbf{U}$. Finally, **ITE intervals can be unstable and/or vacuous for decision making**, especially with hidden variables, and should thus be used carefully.

5 Auditing language models for (un)desirable causal effects

The same framework we use to generate causal effects and benchmark effect estimation methods can allow us to inspect what causal information has been encoded semantically in an LM. For example, we can ask, *"Given a world-view described by an input DAG, what causal conclusion is implied by the language model?"* Our collection of breast cancer SD-SCMs is already set up to explore the effect of PD-L1 on chosen therapy plans, while allowing us to marginalize out an important source of variability: phrasing. Essentially, this amounts to reverse engineering the decision-making process of clinicians, as learned from whatever data the language model was trained on.

Figure 6 shows one example where the two language models *strongly disagree* on what the causal effect is. The effect in this case is the change in probability of choosing the second therapy plan, "start a regimen of trastuzumab and pertuzumab" (shown in standard deviation units). GPT-2 has encoded that on average, an increase in PD-L1 expression levels has neither a positive nor negative impact on choosing this therapy plan. However, Llama-3-8b has encoded instead that an increase in PD-L1 always increases the likelihood of this therapy plan. This discrepancy indicates that **these two language models have encoded two meaningfully different causal effects**. We believe the same procedure can underpin more thorough auditing of LMs for misinformation or discrimination, enabling, e.g., path-specific counterfactual fairness analysis (Kusner et al., 2017; Chiappa, 2018).

6 Conclusion and Future Work

In this work, we have introduced sequence-driven structural causal models (SD-SCMs) as a framework for specifying SCMs with user-defined structure and LM-defined mechanisms. We demonstrate an important use-case for SD-SCMs by creating a benchmark for causal effect estimation. In this proof of concept, we focused on estimation in the presence of confounding, but there are many other settings to explore for effect estimation, such as instrumental variables (Angrist et al., 1993; Hernán and Robins, 2006). Using SD-SCMs to additionally test *causal discovery* is of immediate interest, for example, allowing us to test whether a structure learning method can identify whether one variable is causally upstream or downstream of another (Krämer et al., 2013). Another significant area of future work is to use SD-SCMs or similar as a means of specifying causal structure over sequential data *during learning* (Im et al., 2024); rather than use pre-trained LMs to generate effects, a model can be trained or fine-tuned to handle tasks that require causal reasoning, including complex confounding and sequential decision making.

In short, we believe SD-SCMs can serve as a stepping stone for any application that would benefit from sequential data with controllable causal structure.

7 Limitations

A key difficulty in generating data via SD-SCM for a use-case like benchmarking causal inference methods is to ensure the data have meaningful structure (e.g., significant non-trivial relationships between variables). The reason for this challenge is, in part, by design: the user does not directly specify structural equations. Instead, the structural equations are determined by whatever the language model $\mathcal{P}$ has already encoded. This reliance on what has been previously encoded by a pre-trained language model can sometimes be limiting, motivating a direction of future extensions of SD-SCMs focused on *training* LMs to induce relationships between variables while following an input causal structure, rather than using pre-trained LMs.

A related limitation in the current work is the need to manually account for the sensitivity of generated data to input variable phrasings. For example, in the breast cancer SD-SCMs, we manually create many different phrasings of each sequence variable in order to account for this source of variability. This can be a tedious process as the number of variables grows and could be automated end-to-end in future work.

Risks and societal consequences. There are many potential societal consequences of our work, which are essentially those shared by any model-agnostic application of language models. Pre-trained language models often come with inherent biases and inaccuracies. Generated data may still include such biases or inaccuracies, whether intentional or not. Any future work that builds on this work for the purposes of auditing language models will also inherit the limitations of all tools for model explainability: model explanations always have the potential to be misleading or over-simplified.

Acknowledgments

This research was supported in part by National Science Foundation (NSF) award No. 1922658 and the Samsung Advanced Institute of Technology (under the project Next Generation Deep Learning: From Pattern Recognition to AI). We are grateful to Roman Mutel, Yulia Maksymiuk, and Julia Stoyanovich for involvement in early discussions of this work and feedback on data storage.

References

Joshua David Angrist, Guido Imbens, and Donald B. Rubin. 1993. Identification of causal effects using instrumental variables.

Susan Athey, Guido W. Imbens, Jonas Metzger, and Evan Munro. 2024. Using wasserstein generative adversarial networks for the design of monte carlo simulations. *Journal of Econometrics*, 240(2):105076.

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

Alexander Balke and Judea Pearl. 1994. Probabilistic evaluation of counterfactual queries. *Probabilistic and Causal Inference*.

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

Lorenzo Betti, Carlo Abrate, Francesco Bonchi, and Andreas Kaltenbrunner. 2023. Relevance-based infilling for natural language counterfactuals. *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*.

Ivi Chatzi, Nina Corvelo Benz, Eleni Straitouri, Stratis Tsirtsis, and Manuel Gomez-Rodriguez. 2024. Counterfactual token generation in large language models. *ArXiv*, abs/2409.17027.

Victor Chernozhukov, Matt Goldman, Vira Semenova, and Matt Taddy. 2017. Orthogonal machine learning for demand estimation: High dimensional causal inference in dynamic panels. *arXiv*, pages arXiv–1712.

D Chetverikov, M Demirer, E Duflo, C Hansen, WK Newey, and V Chernozhukov. 2016. Double machine learning for treatment and causal parameters. 2016.

Silvia Chiappa. 2018. Path-specific counterfactual fairness. In *AAAI Conference on Artificial Intelligence*.

Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. 2008. Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4:266–298.

Alicia Curth, David Svensson, James Weatherall, and Mihaela van der Schaar. 2021. Really doing great at estimating cate? a critical look at ml benchmarking practices in treatment effect estimation. In *NeurIPS Datasets and Benchmarks*.

Alicia Curth and Mihaela van der Schaar. 2021a. Nonparametric estimation of heterogeneous treatment effects: From theory to learning algorithms. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR.

Alicia Curth and Mihaela van der Schaar. 2021b. On inductive biases for heterogeneous treatment effect estimation.

Rajeev H Dehejia and Sadek Wahba. 1999. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. *Journal of the American statistical Association*, 94(448):1053–1062.

Vincent Dorie and Jennifer L. Hill. 2020. Causal inference using bayesian additive regression trees [r package bartcause version 1.0-4].

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Amir Feder, Katherine A. Keith, Emaad A. Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E. Roberts, Brandon M Stewart, Victor Veitch, and Diyi Yang. 2021. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *Transactions of the Association for Computational Linguistics*, 10:1138–1158.

Dylan J Foster and Vasilis Syrgkanis. 2023. Orthogonal statistical learning. *The Annals of Statistics*, 51(3):879–908.

Jessica M Franklin, Sebastian Schneeweiss, Jennifer M Polinski, and Jeremy A Rassen. 2014. Plasmode simulation for the evaluation of pharmacoepidemiologic methods in complex healthcare databases. *Computational statistics & data analysis*, 72:219–226.

Zee Fryer, Vera Axelrod, Ben Packer, Alex Beutel, Jilin Chen, and Kellie Webster. 2022. Flexible text generation for counterfactual fairness probing. *ArXiv*, abs/2206.13757.

Yair Ori Gat, Nitay Calderon, Amir Feder, Alexander Chapanin, Amit Sharma, and Roi Reichart. 2023. Faithful explanations of black-box nlp models using llm-generated counterfactuals. *ArXiv*, abs/2310.00603.

Amanda Gentzel, Dan Garant, and David Jensen. 2019. The case for evaluating causal models using interventional measures and empirical data. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Amanda M Gentzel, Purva Pruthi, and David Jensen. 2021. How and why to use experimental data to evaluate methods for observational causal inference. In *International Conference on Machine Learning*, pages 3660–3671. PMLR.

Changying Hao, Liang Pang, Yanyan Lan, Yan Wang, Jiafeng Guo, and Xueqi Cheng. 2021. Sketch and customize: A counterfactual story generator. *ArXiv*, abs/2104.00929.

Miguel A. Hernán and James M. Robins. 2006. Instruments for causal inference: An epidemiologist's dream? *Epidemiology*, 17:360–372.

Jennifer L. Hill. 2011. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20:217 – 240.

Paul Holland. 1985. Statistics and causal inference. *Journal of the American Statistical Association*, 81:945–960.

Daniel Jiwoong Im, Kevin Zhang, Nakul Verma, and Kyunghyun Cho. 2024. Using deep autoregressive models as causal inference engines. *ArXiv*, abs/2409.18581.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. 2023a. Cladder: A benchmark to assess causal reasoning capabilities of language models. *ArXiv*, abs/2312.04350.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona T. Diab, and Bernhard Scholkopf. 2023b. Can large language models infer causation from correlation? *ArXiv*, abs/2306.05836.

Michael C Knaus, Michael Lechner, and Anthony Strittmatter. 2021. Machine learning estimation of heterogeneous causal effects: Empirical monte carlo evidence. *The Econometrics Journal*, 24(1):134–161.

Andreas Krämer, Jeff Green, Jack Pollard, and Stuart Tugendreich. 2013. Causal analysis approaches in ingenuity pathway analysis. *Bioinformatics*, 30:523 – 530.

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

Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. *ArXiv*, abs/1703.06856.

Emre Kıcıman, Robert Osazuwa Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and large language models: Opening a new frontier for causality. *ArXiv*, abs/2305.00050.

Lihua Lei and Emmanuel J. Candès. 2020. Conformal inference of counterfactuals and individual treatment effects. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 83.

Yongqi Li, Mayi Xu, Xin Miao, Shen Zhou, and Tieyun Qian. 2023. Prompting large language models for counterfactual generation: An empirical study. In *International Conference on Language Resources and Evaluation*.

Xiaoyu Liu, Paiheng Xu, Junda Wu, Jiaxin Yuan, Yifan Yang, Yuhang Zhou, Fuxiao Liu, Tianrui Guan, Haoliang Wang, Tong Yu, Julian McAuley, Wei Ai, and Furong Huang. 2024. Large language models and causal inference in collaboration: A comprehensive survey. *ArXiv*, abs/2403.09606.

Christos Louizos, Uri Shalit, Joris M. Mooij, David A. Sontag, Richard S. Zemel, and Max Welling. 2017. Causal effect inference with deep latent-variable models. In *Neural Information Processing Systems*.

Lester Mackey, Vasilis Syrgkanis, and Ilias Zadik. 2018. Orthogonal machine learning: Power and limitations. In *International Conference on Machine Learning*, pages 3375–3383. PMLR.

Daniel McDuff, Yale Song, Jiyoung Lee, Vibhav Vineet, Sai Vemprala, Nicholas Alexander Gyde, Hadi Salman, Shuang Ma, Kwanghoon Sohn, and Ashish Kapoor. 2022. Causalcity: Complex simulations with agency for causal discovery and reasoning. In *Conference on Causal Learning and Reasoning*, pages 559–575. PMLR.

Brady Neal, Chin-Wei Huang, and Sunand Raghupathi. 2020. Realcause: Realistic causal inference benchmarking. *ArXiv*, abs/2011.15007.

Xinkun Nie and Stefan Wager. 2021. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319.

Nick Pawlowski, Daniel Coelho de Castro, and Ben Glocker. 2020. Deep structural causal models for tractable counterfactual inference. *Advances in neural information processing systems*, 33:857–869.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. Elements of causal inference: Foundations and learning algorithms.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Shauli Ravfogel, Anej Svete, Vésteinn Snæbjarnarson, and Ryan Cotterell. 2024. Gumbel counterfactual generation from language models. *Preprint*, arXiv:2411.07180.

Pedro Sanchez and Sotirios A. Tsaftaris. 2022. Diffusion causal models for counterfactual estimation. In *CLEaR*.

Alejandro Schuler, Ken Jung, Robert Tibshirani, Trevor Hastie, and Nigam Shah. 2017. Synth-validation: Selecting the best causal inference method for a given dataset. *arXiv preprint arXiv:1711.00083*.

Uri Shalit, Fredrik D Johansson, and David Sontag. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pages 3076–3085. PMLR.

Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N. Balasubramanian, and Amit Sharma. 2023. Causal inference using llm-guided discovery. *ArXiv*, abs/2310.15117.

Brian G. Vegetabile. 2021. On the distinction between "conditional average treatment effects" (cate) and "individual treatment effects" (ite) under ignorability assumptions. *ArXiv*, abs/2108.04939.

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

Yongjie Wang, Xiaoqi Qiu, Yu Yue, Xu Guo, Zhiwei Zeng, Yuhong Feng, and Zhiqi Shen. 2024. A survey on natural language counterfactual generation. *ArXiv*, abs/2407.03993.

Thierry Wendling, Kenneth Jung, Alison Callahan, Alejandro Schuler, Nigam H Shah, and Blanca Gallego. 2018. Comparing methods for estimation of heterogeneous treatment effects using observational data from health care databases. *Statistics in medicine*, 37(23):3309–3324.

M. Zecevic, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. 2023. Causal parrots: Large language models may talk causality but are not causal. *ArXiv*, abs/2308.13067.

A Formal definition of sequence-driven structural causal models

We first introduce notation preliminaries in Appendix A.1 before formally defining our procedure in Appendix A.2.

A.1 Preliminaries

Let lowercase letter with tilde $\tilde{v}$ denote a random variable, where $\tilde{v} = v$ denotes the value it obtains. Let boldface capital letter $\mathbf{V} = \{\tilde{v}_1, \ldots, \tilde{v}_n\}$ denote a set of variables with value $\mathbf{V} = \mathbf{v}$, capital $P_{\tilde{v}}$ denote the cumulative distribution function of $\tilde{v}$, and lowercase $p_{\tilde{v}}$ denote the density (or mass) function. Let $P_{\tilde{v}|\tilde{x}=x}$ denote the conditional distribution of $\tilde{v}$ given $\tilde{x} = x$ and $P_{\tilde{v}|\tilde{x}}$ denote the collection of $P_{\tilde{v}|\tilde{x}=x}$ for all x. A sequence, or string, is an ordered collection of tokens. We represent this either as a tuple (e.g., sequence $v = (w_1, \ldots, w_T)$ has tokens w_t), or interchangeably as a single string (e.g., $v = w_{1:T} \equiv \bigoplus_{t=1}^{T} w_t$, where $\oplus$ represents string concatenation).

Definition A.1 (Language model)**.** Given a vocabulary $\mathbb{V}$ of possible tokens, we define a language model $\mathcal{P}$ as a joint distribution over any sequence of tokens $v = (w_1, \ldots, w_T) \in \times_{t=1}^{T} \mathbb{V}$, where $\mathcal{P}(v) = \prod_{t=1}^{T} \mathcal{P}(w_t \mid w_{1:(t-1)})$.

Definition A.2 (Structural causal model)**.** We define a structural causal model (SCM) as a 4-tuple $\mathfrak{C} = (\mathbf{V}, \mathbf{U}, \mathbf{F}, P_{\mathbf{U}})$. In this tuple, $\mathbf{V}$ is a set of observed variables, $\mathbf{U}$ a set of unobserved (exogenous) variables, $\mathbf{F}$ a set of functions $\{f_i\}_{i=1}^{|\mathbf{V}|}$ for each $\tilde{v}_i \in \mathbf{V}$ such that $\tilde{v}_i = f_i(\mathbf{PA}_i, \mathbf{U}_i)$ where $\mathbf{PA}_i \subseteq \mathbf{V} \backslash \{\tilde{v}_i\}$ represents the causal parents of $\tilde{v}_i$ and $\mathbf{U}_i \subseteq \mathbf{U}$, and $P_{\mathbf{U}}$ a distribution over $\mathbf{U}$. A causal model can be represented visually as a directed acyclic graph (DAG) with nodes for $\mathbf{U}, \mathbf{V}$ and directed edges for $\mathbf{F}$. SCMs entail an observational distribution $P^{\mathfrak{C}}$ across variables $\mathbf{V} \cup \mathbf{U}$.

Definition A.3 (Interventional distribution)**.** An SCM $\mathfrak{C}$ also entails the distribution of any subset of variables in $\mathbf{V} \cup \mathbf{U}$ following atomic intervention $I = \text{do}(\tilde{v}_i := v)$, which replaces the structural mechanism f_i with fixed value v. Interventions can also be extended to general modifications of f_i. We denote an SCM after intervention I as $\mathfrak{C}^{\text{do}(I)}$ and the resulting distribution as $P^{\mathfrak{C};\text{do}(I)}$.[8]

Counterfactual distributions are computed in a similar fashion, but first conditioning $P_{\mathbf{U}}$ on a particular context before performing an intervention. Where ambiguous, we use an asterisk to denote counterfactual versions $\mathbf{V}^*$ of factual variables $\mathbf{V}$ (Balke and Pearl, 1994).

Definition A.4 (Counterfactual distribution)**.** Counterfactual variable $\mathbf{Y}^*$ given a factual observation $\mathbf{z}$ and intervention $\text{do}(I)$ (where $\mathbf{Y}, \mathbf{Z} \subseteq \mathbf{V}$) can be computed via a three-step procedure often referred to as 'abduction, action, prediction.' Abduction uses observed evidence to obtain $P_{\mathbf{U}|\mathbf{Z}=\mathbf{z}}$ from $P_{\mathbf{U}}$. Action performs intervention $\text{do}(I)$ to obtain modified SCM $\mathfrak{C}^{\text{do}(I)}$. Prediction computes the probability of $\mathbf{Y}^*$ from $\mathfrak{C}^{\text{do}(I)}$ and $P_{\mathbf{U}|\mathbf{Z}=\mathbf{z}}$. For general intervention I and observed assignment $\mathbf{Z} = \mathbf{z}$, we denote the counterfactual distribution $P^{\mathfrak{C}|\mathbf{Z}=\mathbf{z};\text{do}(I)}$.

A.2 Sequence-driven structural causal models

Consider a collection of ordered random variables $(\tilde{v}_1, \tilde{v}_2, \tilde{v}_3, \ldots)$, whose sample spaces $\Omega_{\tilde{v}_i}$ each consist of sets of sequences. We define $\tilde{v}_{1:m} \equiv \bigoplus_{t=1}^{m} \tilde{v}_t$ as the concatenation of the sequences themselves. The sample space for the concatenation of sequences is the cartesian product of the constituent sample spaces $\times_{t=1}^{m} \Omega_{\tilde{v}_t}$. For brevity, we will use the term **sequence variable** to refer to a random variable whose sample space is a set of sequences. Two straightforward abstractions allow us to define SD-SCMs: domain-restricted sampling and parent-only concatenation.

[8] Our notational conventions for interventional and counterfactual distributions follow (Peters et al., 2017).

Algorithm 2 A single SD-SCM sample from the observational distribution

Inputs: $\mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}(\cdot), \tau)$
Returns: $\mathbf{s} = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1, \ldots, v_{|\mathbf{V}|}\right)$
$\mathbf{s} \leftarrow ()$
for $\tilde{x}_t \in \tau$ **do**
  $\text{PA}_\tau \leftarrow \{t' : \tilde{x}_{t'} \in \text{PA}_{\tilde{x}_t}\}$ ordered by τ
  $x_{\text{PA}_\tau} \leftarrow \bigoplus_{x \in \mathbf{s}[\text{PA}_\tau]} x$
  $\mathbf{p}_{\tilde{x}_t} \leftarrow []$
  for $k \in 1, \ldots, |\Omega_{\tilde{x}_t}|$ **do**
    $x \leftarrow \Omega_{\tilde{x}_t}[k]$
    $\mathbf{p}_{\tilde{x}_t}[k] \leftarrow \mathcal{P}\left(x_{\text{PA}_\tau} \oplus x\right)$
  end
  $P_{\text{tot}} \leftarrow \sum_k \mathbf{p}_{\tilde{x}_t}[k]$
  $j \sim \text{Multinomial}(\mathbf{p}_{\tilde{x}_t} / P_{\text{tot}})$
  $x_t \leftarrow \Omega_{\tilde{x}_t}[j]$
  append$(\mathbf{s}, x_t)$
end
return $\mathbf{s}$

Definition A.5 (Domain-restricted sampling)**.** Given language model $\mathcal{P}$, some prior inputs C, and a sequence variable $\tilde{v}_i$ with sample space $\Omega_{\tilde{v}_i}$, domain-restricted sampling defines a distribution $\mathcal{P}_{\tilde{v}_i|C}$ over sample space $\Omega_{\tilde{v}_i}$ simply by tabulating and subsequently normalizing the output probabilities for each possible $v \in \Omega_{\tilde{v}_i}$ conditional on prior inputs C: $\mathcal{P}_{\tilde{v}_i|C}(v) \equiv \frac{\mathcal{P}(v|C)}{\sum_{v' \in \Omega_{\tilde{v}_i}} \mathcal{P}(v'|C)}$.

Definition A.6 (Parent-only concatenation)**.** Given DAG $\mathcal{G}$ over m sequence variables $(\tilde{v}_1, \ldots, \tilde{v}_m)$ and a topological ordering τ consistent with $\mathcal{G}$, parent-only concatenation defines $(\tilde{v}_i \mid \mathbf{PA}_i) \equiv \left(\bigoplus_{t \in \mathbf{PA}_i} \tilde{v}_t\right) \oplus \tilde{v}_i$, where $\mathbf{PA}_i$ are the parents of $\tilde{v}_i$ in $\mathcal{G}$ ordered according to τ.

Given a DAG $\mathcal{G}$ and a language model $\mathcal{P}$, a corresponding sequence-driven SCM defines a sample space of sequences for each variable in $\mathcal{G}$ and provides access to observational, interventional, and counterfactual distributions as follows.

Definition A.7 (Sequence-driven structural causal model (SD-SCM))**.** We define a sequence-driven structural causal model as a 5-tuple $\mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}, \tau)$, where

- $\mathbf{V}$ is a set of finite-domain endogenous/observed sequence variables and $\mathbf{U}$ a set of finite-domain exogenous/unobserved sequence variables;

- $\mathcal{G}$ is a DAG over the variables $\tilde{x}_i$ in $\mathbf{V} \cup \mathbf{U}$ where $\mathbf{PA}_i \subseteq (\mathbf{V} \cup \mathbf{U}) \backslash \{\tilde{x}_i\}$;
- $\mathcal{P}$ is a language model trained on prior inputs C whose vocabulary $\mathbb{V}$ contains all tokens used in $\Omega_{\mathbf{V}}, \Omega_{\mathbf{U}}$; and
- τ is an arbitrary fixed topological ordering of $\mathbf{V} \cup \mathbf{U}$ consistent with $\mathcal{G}$.

An SD-SCM uses $\mathcal{P}$ to define an **observational distribution** over the variables in $\mathbf{V} \cup \mathbf{U}$ that factorizes according to $\mathcal{G}$ via domain-restricted ancestral sampling and parent-only concatenation with τ: $P^{\mathfrak{B}} \equiv \prod_{\tilde{x}_t \in \tau} \mathcal{P}_{\tilde{x}_t | C, \mathbf{PA}_t}$. This procedure is shown in Algorithm 2.

The key difference between an SCM and an SD-SCM is that all variables have at least one common ancestor — the prior inputs C that were used to train the language model, if any. It is however possible to train the LM to induce distributions over the desired variables given this setup. As with the observational distribution, domain-restricted ancestral sampling and parent-only concatenation also allow us to define interventional and counterfactual distributions.

Definition A.8 (Sequence-driven interventional distribution)**.** An SD-SCM $\mathfrak{B}$ entails the distribution of any subset of variables in $\mathbf{V} \cup \mathbf{U}$ following intervention $I = \text{do}(\tilde{v}_i = v)$ by replacing variable $\tilde{v}_i$ with value v, and otherwise sampling in the same manner. As with an SCM, we denote an SD-SCM after intervention I as $\mathfrak{B}^{\text{do}(I)}$ and the resulting interventional distribution as $P^{\mathfrak{B};\text{do}(I)}$. This is computed for intervention $\text{do}(\tilde{v}_i = v)$ as follows: $P^{\mathfrak{B};\text{do}(\tilde{v}_i = v)} \equiv \prod_{\tilde{x}_t \in \tau} \mathcal{P}_{\tilde{x}_t | C, \tilde{v}_i = v, \mathbf{PA}'_t}$, where $\mathbf{PA}'_t = \mathbf{PA}_t \backslash \{\tilde{v}_i\}$. This procedure is shown in Algorithm 3.

In order to admit unique answers to counterfactual queries, we define abduction for an SD-SCM given evidence $\mathbf{Z} = \mathbf{z}$ as the setting of values $\mathbf{U} = \mathbf{u}$ and any evidence upstream of the intervention, rather than a distribution $P_{\mathbf{U}|\mathbf{Z}=\mathbf{z}}$.[9] In order to obtain such values $\mathbf{u}$, one needs access to more than just the observed data and language model $\mathcal{P}$ — obtaining $\mathbf{u}$ requires performing bookkeeping *during the data generation process*. This is a restatement of the fact that computing point counterfactuals in SCMs requires causal mechanisms that are invertible with respect to the noise variables in order to uniquely reconstruct the noise

[9] Other choices can be explored here, which we leave to future extensions of SD-SCMs.

Algorithm 3 A single SD-SCM sample from the interventional distribution for $\text{do}(\tilde{v}_i = v)$

Inputs: $\text{do}(\tilde{v}_i = v), \mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}(\cdot), \tau)$
Returns: $\mathbf{s} = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1, \ldots, v_{|\mathbf{V}|}\right)$

$\mathbf{s} \leftarrow ()$
for $\tilde{x}_t \in \tau$ **do**
  if $\tilde{x}_t \equiv \tilde{v}_i$ **then**
    $x_t \leftarrow v$
  end
  else
    $\text{PA}_\tau \leftarrow \{t' : \tilde{x}_{t'} \in \text{PA}_{\tilde{x}_t}\}$ ordered by τ
    $x_{\text{PA}_\tau} \leftarrow \bigoplus_{x \in \mathbf{s}[\text{PA}_\tau]} x$
    $\mathbf{p}_{\tilde{x}_t} \leftarrow []$
    for $k \in 1, \ldots, |\Omega_{\tilde{x}_t}|$ **do**
      $x \leftarrow \Omega_{\tilde{x}_t}[k]$
      $\mathbf{p}_{\tilde{x}_t}[k] \leftarrow \mathcal{P}\left(x_{\text{PA}_\tau} \oplus x\right)$
    end
    $P_{\text{tot}} \leftarrow \sum_k \mathbf{p}_{\tilde{x}_t}[k]$
    $j \sim \text{Multinomial}(\mathbf{p}_{\tilde{x}_t} / P_{\text{tot}})$
    $x_t \leftarrow \Omega_{\tilde{x}_t}[j]$
  end
  append$(\mathbf{s}, x_t)$
end
return $\mathbf{s}$

that produced a given observation. Because our primary application of SD-SCMs in this work is data *generation*, such bookkeeping is possible in all our use cases.

Definition A.9 (Sequence-driven counterfactual distribution)**.** Counterfactual sequence variable $\mathbf{Y}^*$ given factual evidence $\mathbf{z}$ and intervention $\text{do}(\mathbf{X} = \mathbf{x})$ (where $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \subseteq \mathbf{V}$) can be computed for an SD-SCM $\mathfrak{B}$ whenever the exogenous setting $\mathbf{u} = \{u_1, u_2, \ldots, u_{|\mathbf{U}|}\}$ that generated evidence $\mathbf{z}$ is known. As with an SCM, for intervention I and observed $\mathbf{Z} = \mathbf{z}$, we denote the counterfactual distribution $P^{\mathfrak{B}|\mathbf{Z}=\mathbf{z};\text{do}(I)}$. This is computed for evidence $\mathbf{z}$, exogenous conditions $\mathbf{u}$, and intervention $\text{do}(\tilde{v}_i = v)$ as follows: $P^{\mathfrak{B}|\mathbf{Z}=\mathbf{z};\text{do}(\tilde{v}_i=v)} \equiv \prod_{\tilde{x}_t \in \tau} \mathcal{P}_{\tilde{x}_t | C, \mathbf{U}=\mathbf{u}, \mathbf{Z}'=\mathbf{z}', \tilde{v}_i = v, \mathbf{PA}''_t}$, where $\mathbf{Z}' \subseteq \mathbf{Z}$ contains all non-descendants of $\tilde{v}_i$ present in $\mathbf{Z}$, and $\mathbf{PA}''_t = \mathbf{PA}_t \backslash (\mathbf{U} \cup \mathbf{Z}' \cup \{\tilde{v}_i\})$. This procedure is shown in Algorithm 4.

In plain terms, counterfactual sampling sets not only an intervention $\text{do}(\tilde{v}_i = v)$ but also exogenous variables $\mathbf{U} = \mathbf{u}$ and upstream evidence in order to sample a hypothetical alternative that corresponds to the particular unit in question.

In summary, data can be generated from an

Algorithm 4 A single SD-SCM sample from the counterfactual distribution given observation $\mathbf{s}_{\text{obs}}$

Inputs: $\mathbf{s}_{\text{obs}} = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1, \ldots, v_{|\mathbf{V}|}\right)$
$\text{do}(\tilde{v}_i = v), \mathfrak{B} = (\mathbf{V}, \mathbf{U}, \mathcal{G}, \mathcal{P}(\cdot), \tau)$
Returns: $\mathbf{s}^* = \left(u_1, \ldots, u_{|\mathbf{U}|}, v_1^*, \ldots, v_{|\mathbf{V}|}^*\right)$
$\mathbf{s}^* \leftarrow (u_1, \ldots, u_{|\mathbf{U}|})$
$\text{ND}_i \leftarrow$ non-descendants of $\tilde{v}_i$ in $\mathcal{G}$
for $\tilde{x}_t \in \tau \backslash \mathbf{U}$ **do**
  if $\tilde{x}_t \equiv \tilde{v}_i$ **then**
    $x_t \leftarrow v$
  end
  else if $\tilde{x}_t \in ND_i$ **then**
    $x_t \leftarrow \mathbf{s}_{\text{obs}}[t]$
  end
  else
    $\text{PA}_\tau \leftarrow \{t' : \tilde{x}_{t'} \in \text{PA}_{\tilde{x}_t}\}$ ordered by τ
    $x_{\text{PA}_\tau} \leftarrow \bigoplus_{x \in \mathbf{s}^*[\text{PA}_\tau]} x$
    $\mathbf{p}_{\tilde{x}_t} \leftarrow []$
    for $k \in 1, \ldots, |\Omega_{\tilde{x}_t}|$ **do**
      $x \leftarrow \Omega_{\tilde{x}_t}[k]$
      $\mathbf{p}_{\tilde{x}_t}[k] \leftarrow \mathcal{P}\left(x_{\text{PA}_\tau} \oplus x\right)$
    end
    $P_{\text{tot}} \leftarrow \sum_k \mathbf{p}_{\tilde{x}_t}[k]$
    $j \sim \text{Multinomial}(\mathbf{p}_{\tilde{x}_t} / P_{\text{tot}})$
    $x_t \leftarrow \Omega_{\tilde{x}_t}[j]$
  end
  $\text{append}(\mathbf{s}^*, x_t)$
end
return $\mathbf{s}^*$

SD-SCM by domain-restricted forward sampling variables in topological order, and, with adequate bookkeeping, both interventional and counterfactual samples can also be drawn. The key difficulty in generating data this way that is also useful for benchmarking causal inference methods is to ensure it has meaningful structure. In short, it is easy to generate data, but more difficult to generate *useful* data. The reason for this challenge is that we do not directly specify the structural equations; rather, *the structural equations are determined by whatever the language model $\mathcal{P}$ has already encoded.*

B Full description of the breast cancer SD-SCMs

The 14 covariates in the breast cancer SD-SCMs are defined generally below. For each covariate, 10 different phrasings are considered, resulting in a sample space of 10^{14} possible sequences. For example, for the covariate $\tilde{u}_1$ that represents 'age,' with $\Omega_{\tilde{u}_1} = (25, 35, 45, 55, 65, 75, 85)$, two possible phrasings are:

1. *A $\tilde{u}_1$-year-old woman seeks consultation at the oncology clinic after being recently diagnosed with invasive breast cancer.*
2. *At the oncology clinic, a $\tilde{u}_1$-year-old woman is evaluated following a recent diagnosis of invasive breast carcinoma.*

We consider 50 different variations of this SD-SCM, where the sample space for a given SD-SCM is defined by choosing a randomly sampled phrasing from among the possible phrasings for each of the covariates. For each of the 50 SD-SCMs, 20 datasets (each of size 1000) are sampled. Each covariate and corresponding (ordered) sample space is defined as follows.

1. $\tilde{u}_1$: age, $\Omega_{\tilde{u}_1}$ = (25, 35, 45, 55, 65, 75, 85)
2. $\tilde{u}_2$: medical condition, $\Omega_{\tilde{u}_2}$ = (hypertension, type 2 diabetes mellitus, hyperlipidemia, osteoporosis)
3. $\tilde{u}_3$: medications, $\Omega_{\tilde{u}_3}$ = (lisinopril, metformin, atorvastatin, calcium carbonate)
4. $\tilde{u}_4$: menopausal status, $\Omega_{\tilde{u}_4}$ = (pre-menopausal, post-menopausal)
5. $\tilde{x}_1$: type of carcinoma, $\Omega_{\tilde{x}_1}$ = (invasive ductal carcinoma (IDC), invasive lobular carcinoma, medullary carcinoma, tubular carcinoma)
6. $\tilde{x}_2$: histology grade, $\Omega_{\tilde{x}_2}$ = (grade 1, grade 2, grade 3)
7. $\tilde{x}_3$: genetic mutation 1, $\Omega_{\tilde{x}_3}$ = (TP53, PIK3CA, BRCA1, BRCA2)
8. $\tilde{x}_4$: genetic mutation 2, $\Omega_{\tilde{x}_4}$ = (TP53, PIK3CA, BRCA1, BRCA2)
9. $\tilde{x}_5$: level of hormone receptor expression, $\Omega_{\tilde{x}_5}$ = (low, moderate, high)
10. $\tilde{x}_6$: genomic instability score, $\Omega_{\tilde{x}_6}$ = (low, high)
11. $\tilde{x}_7$: chromosomal aberration strength, $\Omega_{\tilde{x}_7}$ = (significant, minor)
12. $\tilde{x}_8$: HER2 status, $\Omega_{\tilde{x}_8}$ = (positive, negative)
13. $\tilde{t}$: PD-L1 expression levels, $\Omega_{\tilde{t}}$ = (low, high)
14. $\tilde{y}$: therapy plan, $\Omega_{\tilde{y}}$ = (initiate an aromatase inhibitor therapy, administer a combination of a PARP inhibitor and chemotherapy, start a regimen of trastuzumab and pertuzumab, begin treatment with a checkpoint inhibitor such as pembrolizumab)

The following is an example sequence randomly sampled from one possible choice of phrasings:

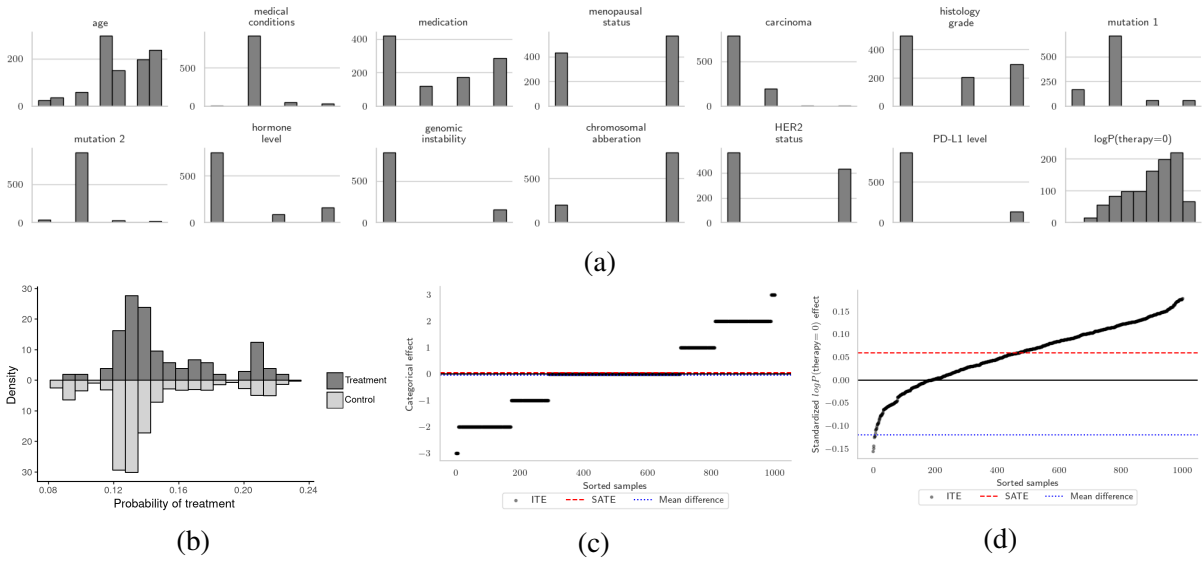


Figure 7: An example dataset generated by the breast cancer SD-SCM using Llama-3-8b, showing **(a)** features, **(b)** propensity scores, **(c)** categorical outcome ITEs, and **(d)** continuous outcome ITEs.

> A 65-year-old woman comes to the oncology department with a recent diagnosis of invasive breast carcinoma. Her prior medical history includes hyperlipidemia. This has been managed with lisinopril. This post-menopausal woman has no prior history of breast surgeries or hormone replacement therapy. Following a detailed assessment with imaging and biopsy, the results from the biopsy were analyzed and disclosed the following. The pathology report indicates the tumor is tubular carcinoma. The tumor's histology is rated as grade 3. The tumor shows an elevated mutation burden, with particular mutations detected in the BRCA2 gene in addition to the TP53 gene. The immunohistochemistry results display robust positive staining for estrogen receptor (ER) and progesterone receptor (PR), indicating high levels of expression. The level of genomic instability in the tumor is described as low. This implies that chromosomal aberrations are minor. Immunohistochemistry reveals HER2 as negative while FISH confirms that HER2 amplification is not present. Programmed death-ligand 1 (PD-L1) expression in the tumor is low with no distant metastases found in the imaging studies. Considering the comprehensive findings and the patient's health and treatment history, which treatment strategies are most suitable for this patient? The best option is to begin treatment with a checkpoint inhibitor such as pembrolizumab.

With each sample space indexed according to the order of their values above (with indexes starting at zero), the above text sequence corresponds to the observation

$$(\tilde{u}_1, \ldots, \tilde{u}_4, \tilde{x}_1, \ldots, \tilde{x}_8, \tilde{t}, \tilde{y}) = (4, 2, 0, 1, 3, 2, 3, 0, 2, 0, 1, 1, 0, 3).$$

The full set of possible examples and code to generate this SD-SCM and corresponding data (in our case generated using V100 and RTX8000 GPUs) is available in our repository at https://github.com/lbynum/sequence-driven-scms.

Figure 7 shows plots of the features (7a), propensity scores (7b), categorical ITEs (7c), and continuous ITEs (7d) for a single generated dataset using Llama-3-8b. Because the outcome $\tilde{y}$ has $|\Omega_{\tilde{y}}| = 4$ possible values, we can consider several possible outcomes, including the observed outcome (categorical), log probabilities for each outcome value (continuous), or probabilities for each outcome value (continuous). This creates, in effect, nine possible targets for each dataset. For benchmarking purposes, we find using probabilities and/or log probabilities as the outcome to be the most useful — there is frequently an effect (even at the individual level) in probability space, even if the sampled outcomes do not change. Comparing Figure 7c to Figure 7d demonstrates this, where for the 400 or so observations where the categorical ITE is zero, the continuous ITE is instead nonzero.

Figure 7d also demonstrates that we are able to satisfy our main criterion for meaningful benchmark: the observational distribution $P^{\mathfrak{B}}_{\tilde{y}|\tilde{t}=t}$ and interventional distribution $P^{\mathfrak{B};\text{do}(\tilde{t}=t)}_{\tilde{y}}$ are different enough that the SATE and the observed mean difference in outcomes between the treatment and control group are not only different in value, but *they also disagree in sign*. This is particularly meaningful in a causal inference setting — the treatment appears to lower the outcome, when in fact, its effect is to increase the outcome.

C Additional estimation results

C.1 ATE results

For all implementations that directly support ATE estimation, we report the R^2 and root-mean-squared-error (RMSE) across the 1000 datasets for each language model in two settings, using $\text{logP}(\tilde{y} = 0)$ as the outcome. The first setting is with estimation using all 14 covariates (all 12 confounders, the treatment, and the outcome). This is

Table 1: SATE prediction results for methods that directly target ATEs.

Method	$\mathcal{P}$ = GPT-2				$\mathcal{P}$ = Llama-3-8b			
	R^2		RMSE		R^2		RMSE	
	All Cov.	Hidden	All Cov.	Hidden	All Cov.	Hidden	All Cov.	Hidden
T-Only OLS	0.6047	0.6047	0.0172	0.0172	0.5082	0.5082	0.0091	0.0091
BART	**0.9999**	**0.8794**	**0.0003**	**0.0095**	**0.9967**	**0.8476**	**0.0007**	**0.0051**
ForestDML	0.9941	0.8686	0.0021	0.0099	0.9608	0.8129	0.0026	0.0056
ForestDR	$\leqslant 0$	$\leqslant 0$	6.4268	29.0210	0.9581	0.8179	0.0027	0.0055
ForestS	0.9771	0.8777	0.0041	0.0096	0.8243	0.8286	0.0054	0.0054
ForestT	0.9793	0.8588	0.0039	0.0103	0.9454	0.8126	0.0030	0.0056
LinReg	0.9146	0.8646	0.0080	0.0101	0.6538	0.5599	0.0076	0.0086
LinearDML	0.979	0.8655	0.0040	0.0100	0.9608	0.8216	0.0026	0.0055
LinearDR	$\leqslant 0$	$\leqslant 0$	11.0736	29.5414	0.9589	0.8176	0.0026	0.0055
LinearS	0.9146	0.8632	0.0080	0.0101	0.6538	0.4869	0.0076	0.0093
LinearT	0.9181	0.8688	0.0078	0.0099	0.6395	0.5385	0.0078	0.0088
RF	0.976	0.8766	0.0042	0.0096	0.8122	0.8295	0.0056	0.0054

Table 2: Mean R^2 of methods estimating CATEs/ITEs, comparing estimation with datasets of size 1,000 versus 10,000.

Method	All Covariates				Hidden $\mathbf{U}$			
	$\mathcal{P}$ = GPT-2		$\mathcal{P}$ = Llama-3-8b		$\mathcal{P}$ = GPT-2		$\mathcal{P}$ = Llama-3-8b	
	1,000	10,000	1,000	10,000	1,000	10,000	1,000	10,000
T-Only OLS	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
BART	**0.7162**	**0.9691**	**0.5221**	**0.9214**	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
BART-ITE	0.7102	0.9344	0.5183	0.8823	0.0054	0.0185	0.0011	0.0169
CausalForest	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	0.0313	$\leqslant 0$	0.1399
ForestDML	$\leqslant 0$	0.6605	$\leqslant 0$	0.5551	$\leqslant 0$	**0.0608**	$\leqslant 0$	0.1850
ForestDR	$\leqslant 0$	0.7220	$\leqslant 0$	0.5968	$\leqslant 0$	0.0503	$\leqslant 0$	**0.1931**
ForestS	$\leqslant 0$	0.0661	$\leqslant 0$	0.1263	0.0026	0.0371	**0.0063**	0.0936
ForestT	0.1031	0.3202	$\leqslant 0$	0.1319	**0.0076**	0.0366	0.0054	0.0719
LinReg	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
LinearDML	0.2381	0.5309	0.1491	0.5404	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	0.1483
LinearDR	0.2329	0.5900	0.1562	0.5293	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	0.1734
LinearS	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
LinearT	0.0116	0.1470	$\leqslant 0$	0.1355	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
RF	$\leqslant 0$	0.0327	$\leqslant 0$	0.1038	0.0024	0.0373	0.0060	0.0940
TARNet	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$
TNet	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$

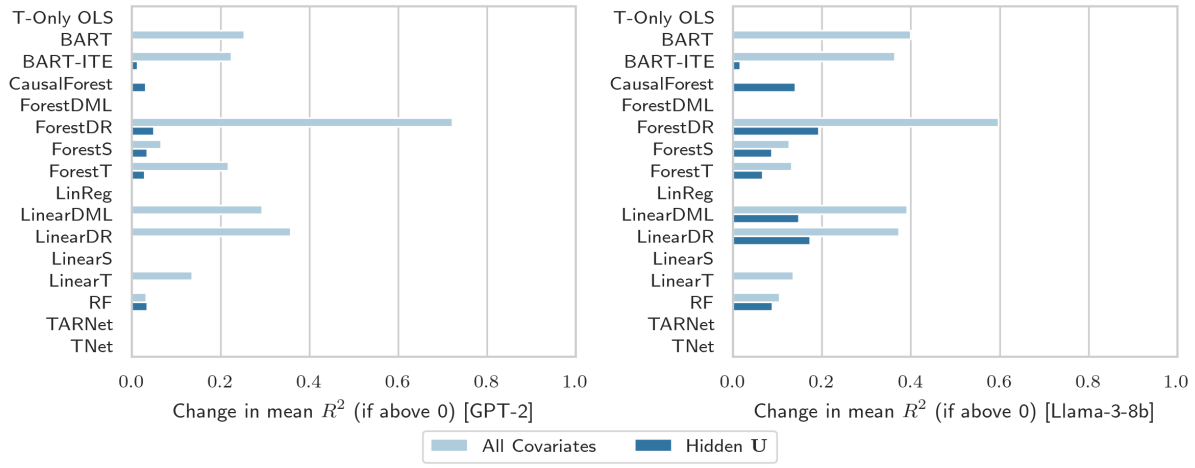


Figure 8: Change in mean R^2 (if above 0) of methods estimating CATEs and ITEs after a tenfold increase in dataset size from $1,000$ to $10,000$.

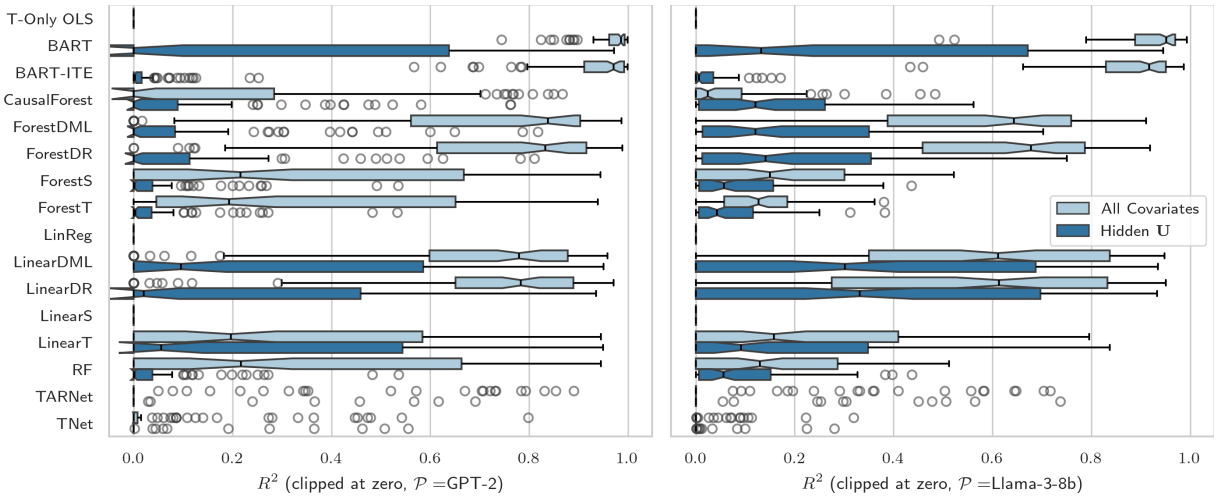


Figure 9: R^2 values across all methods that provide point estimates of CATEs/ITEs for datasets of size 10,000 generated by GPT-2 (left) and Llama-3-8b (right).

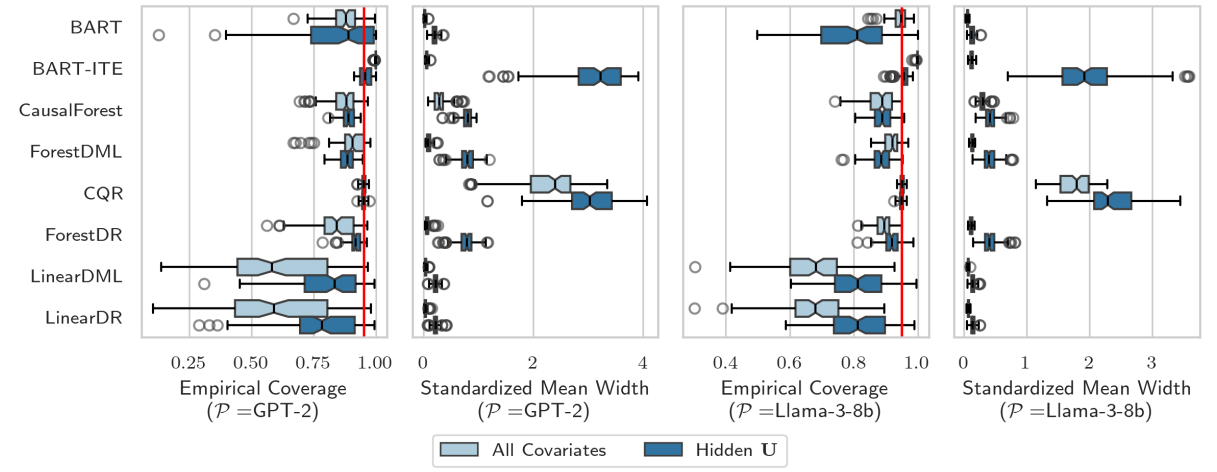


Figure 10: CATE/ITE empirical coverage ($\alpha = 0.05$) and interval width (in outcome standard deviation units) for methods that provide intervals. Nominal coverage of 95% is indicated by the red line. Shown for datasets of size 10,000 generated by GPT-2 (left) and Llama-3-8b (right).

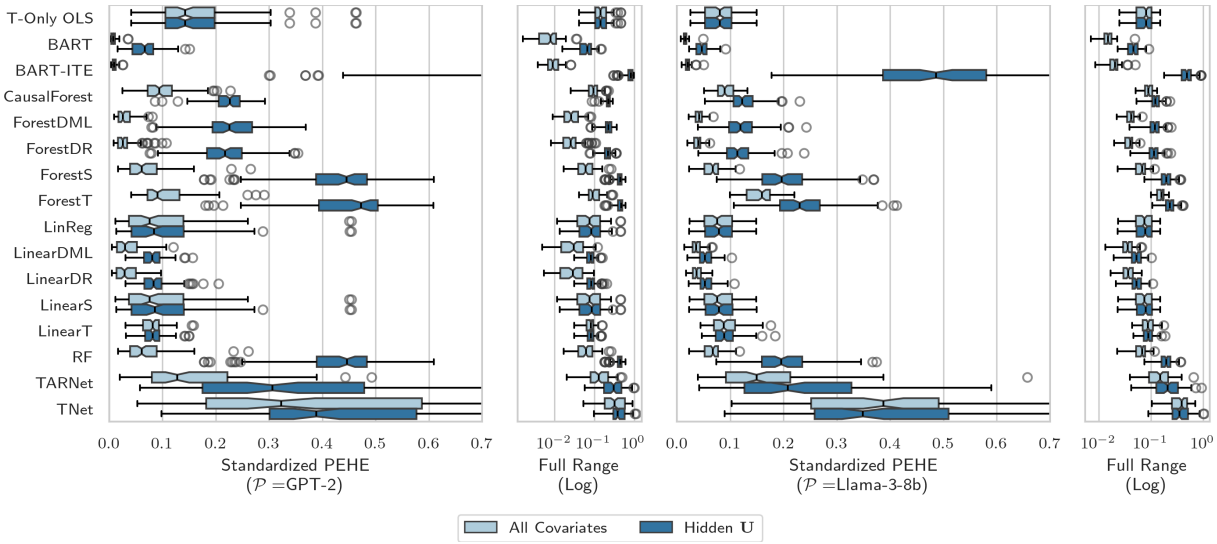


Figure 11: PEHE across all methods that provide point estimates of CATEs/ITEs, shown in standard deviation units of the outcome $\tilde{y}$. Results shown for datasets of size 10,000 generated by GPT-2 (left) and Llama-3-8b (right).

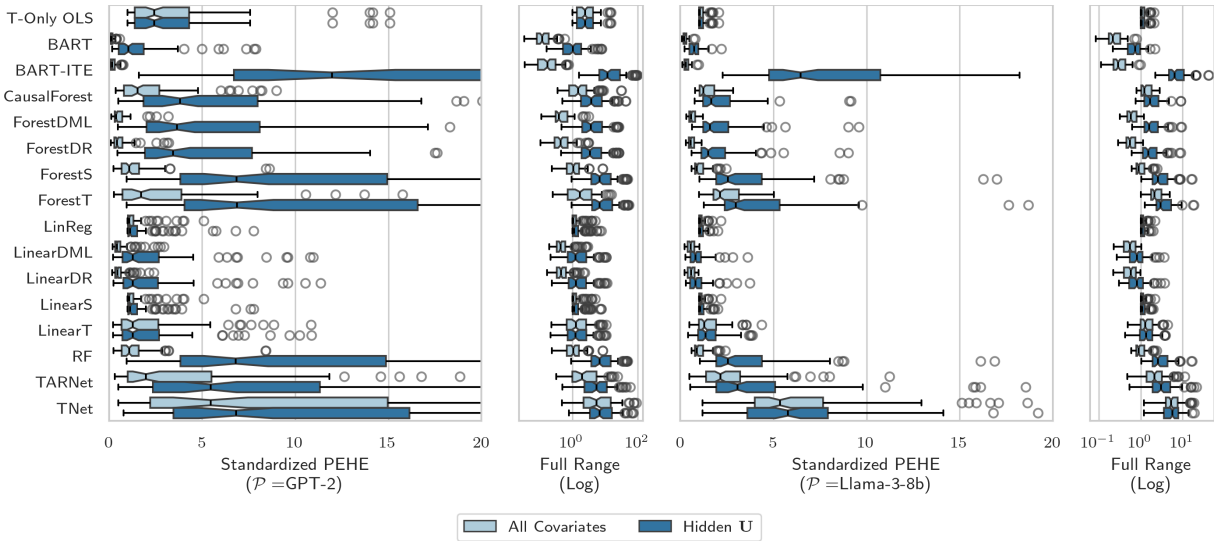


Figure 12: PEHE across all methods that provide point estimates of CATEs/ITEs, shown in units of ITE standard deviation. Results shown for datasets of size 10,000 generated by GPT-2 (left) and Llama-3-8b (right).

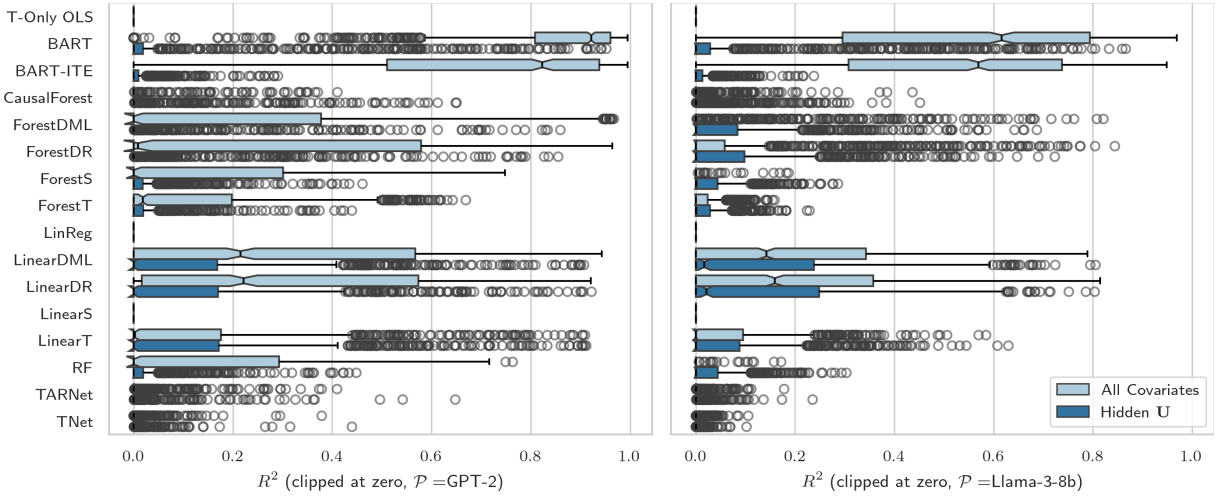


Figure 13: R^2 values across all methods that provide point estimates of CATEs/ITEs for datasets of size 1,000 generated by GPT-2 (left) and Llama-3-8b (right).

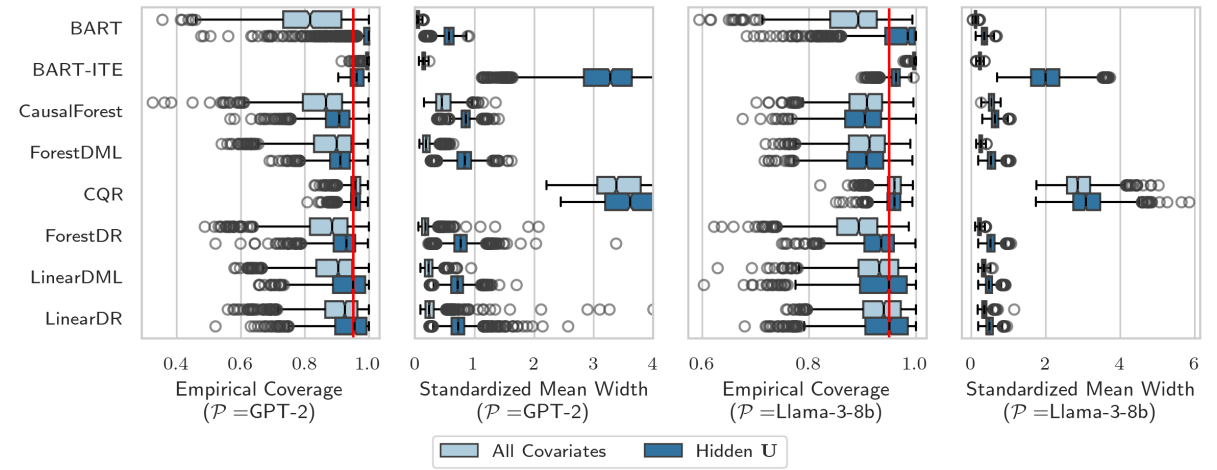
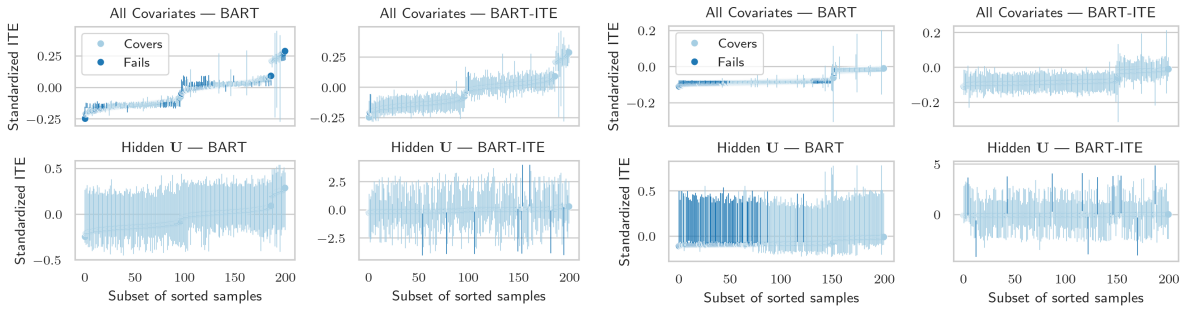


Figure 14: CATE/ITE empirical coverage ($\alpha = 0.05$) and interval width (in outcome standard deviation units) for methods that provide intervals. Nominal coverage of 95% is indicated by the red line. Shown for datasets of size 1,000 generated by GPT-2 (left) and Llama-3-8b (right).



(a) (b)

Figure 15: Interval estimates from BART versus BART-ITE across two example datasets of size 1,000, (a) and (b).

denoted **All Cov.** in Table 1. The second setting is with the variables $\mathbf{U} = \{\tilde{u}_1, \tilde{u}_2, \tilde{u}_3, \tilde{u}_4\} = \{$`age`, `medical conditions`, `medication`, `menopausal status`$\}$ hidden, denoted **Hidden**. Table 1 shows that ATE estimation is more or less challenging depending on which language model is used. In this case, Llama-3-8b produces ATEs that are more challenging to estimate, with the exception of GPT-2 for the doubly robust methods, whose R^2 and RMSE suffer significantly due to several large outlying estimates. Across all methods, performance tends to drop significantly in the 'Hidden' setting, suggesting that $\mathbf{U}$ are indeed hidden confounders. Across methods, BART shows the strongest performance in all settings in Table 1.

C.2 Comparison of dataset size 1,000 and 10,000

Table 2 and Figure 8 show that CATE and ATE estimation remain difficult even after a tenfold increase in dataset size (from $N = 1,000$ to $N = 10,000$), especially in the Hidden $\mathbf{U}$ setting. Across estimation methods, performance tends to increase as sample size increases, especially if the method originally achieved R^2 above zero with $N = 1,000$. In other words, methods that do reasonably well at $N = 1,000$ show improvement with more data, as we would expect. However, several methods struggle in both settings, even with ten times more data. For example, TARNet, TNet, and CausalForest still remain unstable and inaccurate in both the All Covariates and the Hidden $\mathbf{U}$ settings across both sample sizes. Overall, these results indicate that CATE and ITE estimation in this case are not challenging due only to small sample sizes. This is useful to know, especially when we consider that corresponding real-world use-cases often deal with even smaller sample sizes.

Additional results for each dataset size are shown individually in the following figures. Figures 9 and 10 show R^2 and coverage results on datasets of size 10,000. These correspond to the same figures in the main text, but now showing both GPT-2 and Llama-3-8b, allowing for comparison across models. Figure 11 shows the same setting using standardized Precision in Estimating Heterogeneous Effects (PEHE) (Hill, 2011), which is the RMSE of the CATE predictions across the different observed values of x, i.e.,

$$\text{PEHE}_j = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{\tau}_i - \tau_i)^2}$$

for a dataset $D_j = \{\mathbf{u}_i, \mathbf{x}_i, t_i, y_i\}_{i=1}^n$ where $\hat{\tau}_i$ is the estimated ITE for unit i and τ_i is the true ITE. We standardize PEHE using the empirical standard deviation $\hat{\sigma}_j$ of the outcomes $\{y_i\}_{i=1}^n$ in each dataset, i.e.,

$$\text{(Standardized PEHE)}_j = \sqrt{\frac{1}{n \cdot \hat{\sigma}_j^2}\sum_{i=1}^{n}(\hat{\tau}_i - \tau_i)^2}.$$

Figure 12 shows the same metric standardized instead using the (much smaller) standard deviation of the ITE.

Results in the case of dataset size $1,000$ show similar trends to those in the size $10,000$ setting. Figure 13 shows R^2 values clipped at zero across all methods that provide point estimates for CATEs. When all covariates are observed, BART does significantly better explaining CATE variation, followed by DML and DR with much lower averages, much like the size 10,000 case. Similarly, CATE estimation becomes much more challenging for all methods with hidden $\mathbf{U}$. The difference in effect estimation difficulty between Llama-3-8b and GPT-2 is also more noticeable for CATEs than it was for

ATEs. Overall, some methods show more instability in the dataset size 1,000 case than in the size 10,000 case, as expected with less data.

Figure 14 shows empirical coverage results in the dataset size 1,000 case for all estimators that provide intervals. Similar to the size 10,000 case, empirical coverage is under nominal for all methods that target CATE in the setting with all covariates. Hidden $\mathbf{U}$ generally increases uncertainty, bringing the DR methods and LinearDML median coverage near nominal. Interestingly, BART for CATE achieves higher median coverage of the ITE than BART-ITE, but with a much larger tail of poor coverage. BART-ITE, by contrast, has much less variable coverage in the ITE setting, but at the cost of much wider intervals. Figure 15 shows intervals for BART targeting the CATE versus BART-ITE across two example datasets of size 1,000, demonstrating that, as in the size 10,000 case, the tighter intervals of BART targeting the CATE can be overconfident with variable coverage, while the wider intervals of BART-ITE are too wide to be useful.