Scalable Causal Structure Learning via Amortized Conditional Independence Testing

James Leiner JLEINER@STAT.CMU.EDU
Department of Statistics and Data Science, Carnegie Mellon University

Brian Manzo BMANZO@UMICH.EDU
Department of Statistics, University of Michigan

Aaditya Ramdas ARAMDAS@STAT.CMU.EDU
Department of Statistics and Data Science, Carnegie Mellon University
Machine Learning Department, Carnegie Mellon University

Wesley Tansey TANSEYW@MSKCC.ORG
Computational Oncology, Memorial Sloan Kettering Cancer Center

Editors: Biwei Huang and Mathias Drton

Abstract

Controlling false positives (Type I errors) through statistical hypothesis testing is a foundation of modern scientific data analysis. Existing causal structure discovery algorithms either do not provide Type I error control or cannot scale to the size of modern scientific datasets. We consider a variant of the causal discovery problem with two sets of nodes, where the only edges of interest form a bipartite causal subgraph between the sets. We develop Scalable Causal Structure Learning (SCSL), a method for causal structure discovery on bipartite subgraphs that provides Type I error control. SCSL recasts the discovery problem as a simultaneous hypothesis testing problem and uses discrete optimization over the set of possible confounders to obtain an upper bound on the test statistic for each edge. Semi-synthetic simulations demonstrate that SCSL scales to handle graphs with hundreds of nodes while maintaining error control and good power. We demonstrate the practical applicability of the method by applying it to a cancer dataset to reveal connections between somatic gene mutations and metastases to different tissues.

Keywords: Causal Inference, Hypothesis Testing, Directed Acyclic Graphs

1. Introduction

Many scientific applications can be posed as a causal discovery problem where a directed acylic graph (DAG) is learned that models the causal dependencies within an observational dataset (e.g. Tennant et al. (2020), Ogburn et al. (2022)). In order to ensure the reliability of these findings, controlling the error rate on the set of causal discoveries is critical. Given finite data and noisy observations, exact determination of causal arrows in a DAG is impossible. As such, the causal discovery task is typically cast through the lens of statistical hypothesis testing for conditional independencies (Spirtes et al., 2000).

Although the problem of large-scale multiple hypothesis testing and uncertainty quantification is well studied in settings where purely associative relationships are the target discoveries (Benjamini and Hochberg, 1995; Goeman and Solari, 2014), large-scale causal learning produces a unique set of challenges. The number of possible DAGs scales superexponentially with the number of nodes in a graph, making causal learning an especially difficult problem when the number of variables in a dataset is large. Moreover, existing causal structure learning algorithms learn a graph with either no attempt to provide frequentist error guarantees (Chickering, 2003; Ramsey et al., 2017a; Zheng et al., 2018; Cundy et al., 2021b; Annadani et al., 2021; Cundy et al., 2021a) or require parametric assumptions on the data generating distribution which are unknown in practice (Strobl et al., 2019).

A feature of many scientific datasets that can be exploited to decrease the computational burden of this task is *temporal separation* of variables. If a dataset has a sequence of variables that measure quantities that came into existence at different times, this corresponds to a priori knowledge that some edges can only be oriented in a particular direction. This allows us to reduce the number of conditional independence tests required to draw an edge on a causal graph.

In this paper, we introduce Scalable Causal Structure Learning (SCSL), a method for large-scale causal hypothesis testing that can scale to problems with hundreds of variables and thousands of potential edges for causal graphs with temporally separated sets of nodes. SCSL enables the use of black box machine learning models for hypothesis testing, requires no parametric assumptions, and returns a p-value for each edge under consideration. To scale to large graphs, SCSL recasts the causal search process as a discrete optimization problem. It then amortizes the dominant cost in causal structure identification: conditional independence testing over the combinatorial set of possible parent nodes in the causal graph. This avoids the combinatorial explosion in the candidate conditioning sets and reduces the search to a series of parallelizable optimization problems for each edge. We validate SCSL through semi-synthetic experiments using a cancer dataset that pairs genomic mutations at the primary tumor location of a cancer patient with information about metastases that have developed elsewhere in a patient's body (Nguyen et al., 2022). These simulation studies indicate that the method has high power, controls Type I error rate at the target level, and scales to larger graphs than existing methods.

Background and related work Classical algorithms for causal discovery like the SGS and PC algorithms (Spirtes et al. (2000)) convert causal structure learning into queries of conditional independence between nodes. The SGS algorithm draws an edge between two nodes if they are conditionally dependent given any possible subset of the remaining nodes. The PC algorithm simplifies this process by first deleting edges in the causal graph based on marginal independence testing. The size of the conditioning subsets is then allowed to increase by one for each subsequent round of CI testing, but fewer CI tests are required in each round as the algorithm only considers conditioning subsets of nodes that have remained adjacent to each other. The computational complexity of SGS matches the complexity of the PC in the worst case, but in most settings where the causal graph is reasonably sparse, the PC algorithm significantly reduces the number of independence tests needed to learn a causal graph. However, both methods are based on perfect knowledge of the CI structure and are computationally intractable beyond a few dozen nodes. Related work used the PC algorithm as a starting point and then relaxes assumptions around faithfulness (e.g. Ramsey et al. (2006)) or causal sufficiency (e.g. Spirtes (2001)), but these methods also rely on having perfect knowledge of the graph's CI structure. The work of Strobl et al. (2019) extends the PC algorithm to generate edge-specific p-values, but only with provable Type I error control under the assumption of zero Type II error during the edge deletion stage of skeleton discovery.

Other approaches to causal graph are score-based methods which maximize a score function such as BIC (D and Heckerman, 1997) or BDeau (Heckerman et al., 1995) over the space of all possible DAGs. Since the number of DAGs increases superexponentially with the number of nodes, approximate algorithms based on greedy search (Chickering, 2003; Ramsey et al., 2017a), coordinate descent (Aragam and Zhou, 2015; Fu and Zhou, 2013) or other methods are required. Other methods impose more stringent modeling assumptions such as linearity (e.g. Shimizu et al. (2011)) simplify the task. Follow-up work (Ramsey et al., 2017b; Solus et al., 2017) improves several of these methods to scale to datasets with thousands of nodes and potential edges using parallelization. However, none of these methods output edge specific p-values and are therefore not directly comparable to our proposed method.

More recently, differentiable approaches to causal discovery have been proposed (Zheng et al., 2018), many of which enable Bayesian inference of the posterior distribution over DAGs (Cundy et al., 2021b; Annadani et al., 2021; Cundy et al., 2021a). Unfortunately, these methods focus primarily on continuous rather than discrete data, limiting the application to datasets like the cancer dataset we consider in our simulations. Further, Bayesian uncertainty requires accurate model specification for valid posterior coverage. When the model is misspecified, Bayesian credible intervals can massively inflate Type I error. Similarly, Peters et al. (2016) weaken the faithfulness assumptions, but require repeated observations of variables across different intervention settings instead of relying on purely observational data. Overall, no method exists that is capable of providing Type I error control on individual edges across a large graph without making stringent parametric assumptions.

2. Methodology

Consider a directed acyclic graph (DAG) $\mathcal{G}$ with two sets of nodes, $\mathcal{X}$ and $\mathcal{Y}$, where we want to learn the directed edges between $\mathcal{X}$ and $\mathcal{Y}$ from observational data. We assume that we have prior knowledge that no edge is directed from $\mathcal{Y}$ to $\mathcal{X}$ such as temporal separation between the sets of nodes. For instance, in the cancer dataset, $\mathcal{Y}$ would represent metastasis events that occur after the tumors represented by $\mathcal{X}$ were sequenced.

The problem reduces to learning which $X_j \in \mathcal{X}$ are connected to which $Y_k \in \mathcal{Y}$. A naive approach might be to infer an edge based on a conditional independence test of $X_j \perp\!\!\!\perp Y_k | \{X_{-j}, Y_{-k}\}$ where $X_{-j} := \mathcal{X} \backslash X_j$ and $Y_{-k} := \mathcal{Y} \backslash Y_k$. However, a collider being present in $\mathcal{Y}$ could introduce dependence even when there is no edge present as Figure 1 demonstrates.

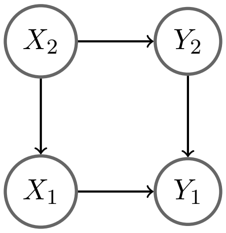

Figure 1: $\mathcal{X} = \{X_1, X_2\}$ and $\mathcal{Y} = \{Y_1, Y_2\}$. Y_1 is a collider, so a conditional independence test would indicate X_1 and Y_2 are conditionally dependent given $\{X_2, Y_1\}$. However, X_1 and Y_2 are not conditionally independent given X_2. Querying all dependence relations prevents an edge from being drawn erroneously.

We make the following assumptions about the graph and distribution of the data which are common in the literature (Spirtes et al., 2000).

Assumption 1 (Global directed Markov property) *For a graph $\mathcal{G}$, if two nodes U and V are d-separated given a disjoint set nodes W, then U and V are conditionally independent given W.*

Assumption 2 (d-separation Faithfulness) *For a graph $\mathcal{G}$, if two nodes U and V are conditionally independent given a disjoint set of nodes W, then U and V are d-separated given W.*

Assumption 3 (Causal sufficiency) *The graph $\mathcal{G}$ includes all common causes for any pair of nodes contained in $\mathcal{G}$.*

These assumptions lead to Proposition 1, which enables us to reduce queries about edge presence to queries about conditional independence between nodes.

Proposition 1 *Assume $\mathcal{G}$ satisfies the global directed Markov property and the probability distribution is d-separation faithful. Furthermore, assume that edges may not be directed from any element in $\mathcal{Y}$ to any element in $\mathcal{X}$. Then there is an edge between two vertices $X_j \in \mathcal{X}$ and $Y_k \in \mathcal{Y}$ if and only if X_j and Y_k are conditionally dependent given $S \cup X_{-j}$ for all $S \subseteq Y_{-k}$.*

In order to construct a p-value, we consider a hypothesis test for each pair (X_j, Y_k) as follows:

$$\begin{aligned} H_0 &: X_j \to Y_k \text{ is absent,} \\ H_1 &: X_j \to Y_k \text{ is present.} \end{aligned} \tag{1}$$

Proposition 1 lets us restate the null and alternative as

$$\begin{aligned} &H_0 : \exists S \subseteq \mathcal{Y} \backslash Y_k \text{ such that } X_j \perp\!\!\!\perp Y_k | S, X_{-j}, \\ &H_1 : X_j \text{ is not independent of } Y_k \text{ given } \{S, X_{-j}\} \text{ for all } S \subseteq Y_{-k}. \end{aligned} \tag{2}$$

Let $p_{X_j \perp\!\!\!\perp Y_k | S}$ denote the p-value corresponding to a test for conditional independence between X_j and Y_k given a set of nodes S. Equation (2) implies that the p-value $p_{X_j \to Y_k}$ can be bounded by

$$p_{X_j \to Y_k} \leqslant \max_{S \subseteq Y_{-k}} p_{X_j \perp\!\!\!\perp Y_k | S, X_{-j}}. \tag{3}$$

Constructing a p-value to test H_0 thus reduces to computing p-values for a series of conditional independence tests.

2.1. Conditional Independence Testing

To test the null hypothesis that $X_j \perp\!\!\!\perp Y_k | S, X_{-j}$, we employ the generalized covariance measure (GCM) of Shah and Peters (2020). The GCM recasts the problem of conditional independence into one about functional estimation of the expected conditional covariance

$$\begin{aligned} \mathbb{E}\left[\text{Cov}\left(X_j, Y_k | S, X_{-j}\right)\right] :=& \mathbb{E}[\mathbb{E}[X_j Y_k | S, X_{-j}] \\ &- \mathbb{E}[X_j | S, X_{-j}]\mathbb{E}[Y_k | S, X_{-j}]], \end{aligned} \tag{4}$$

and then tests whether this quantity is equal to 0. Any set of joint densities for $(\mathcal{X}, \mathcal{Y})$ that are null will have this property, although it is possible for X_j and Y_k to be conditionally dependent but with 0 covariance. Therefore, this measure will only have power against the set of alternatives which have non-zero conditional covariance, which is likely to be the case in most practical settings.

The procedure constructs a test for the null hypothesis of 0 conditional covariance by first finding estimates $\hat{X}_j$ for the conditional expectation $\mathbb{E}[X_j | S, X_{-j}]$ and $\hat{Y}_k$ for the conditional expectation $\mathbb{E}[Y_k | S, X_{-j}]$. Any type of predictive modeling to construct this estimate is valid, for example construction of a neural net, so long as the product of the mean square errors of the two quantities is

$o(n^{-1})$. Given n samples $\{X_j^i, Y_k^i, S^i, X_{-j}^i\}_{i=1}^n$, we denote $R_i = \left(X_j^i - \widehat{X}_j^i\right)\left(Y_k^i - \widehat{Y}_k^i\right)$, and the test statistic

$$T^{(n)} = \frac{\sqrt{n} \cdot \frac{1}{n}\sum_{i=1}^n R_i}{\left(\frac{1}{n}\sum_{i=1}^n R_i^2 - \left(\frac{1}{n}\sum_{r=1}^n R_r\right)^2\right)^{1/2}}, \tag{5}$$

which will be distributed asymptotically as $\mathcal{N}(0,1)$ under the null, allowing us to construct a valid p-value. For a precise account of the technical conditions for the theorem, see Theorem 6 of Shah and Peters (2020) which we recall in the Appendix for completeness.

We note that our method is adaptable to other CI testing procedures such as conditional randomization tests (e.g. Candes et al. (2016), Tansey et al. (2022), Liu et al. (2021)). We focus on the GCM in this work because it requires less stringent modeling assumptions on the conditional distribution $X|Z$ and is less computationally burdensome.

2.2. Amortized Predictive Modeling

The GCM only requires the computation of a single test statistic per conditional independence query. While the computation of the individual test statistics is straightforward, this approach may still become computationally onerous when testing for the presence of an edge using Equation (3) due to the fact that two separate predictive models need to be created for each of the $2^{|\mathcal{Y}|-1}$ CI tests corresponding to H_0 which will cause this methodology to scale poorly as $|\mathcal{Y}|$ increases.

To make this process more computationally tractable, we will create a single predictive model for each element $Y_k \in \mathcal{Y}$ that has the flexibility to take in a choice of conditioning set $S \subseteq Y_{-k}$ as well as a choice of element $X_j \in \mathcal{X}$ and outputs an estimate of $\mathbb{E}[Y_k|X_{-j}, S]$. We call this *amortized predictive modeling* because instead of creating bespoke models for each S, we localize the cost of training into a single flexible model. Formally, we wish to train a function for each Y_k denoted as $\pi_k : \mathbb{R}^{|\mathcal{X}|} \times \mathbb{R}^{|\mathcal{Y}|} \times \{0,1\}^{|\mathcal{Y}|-1} \times [|\mathcal{X}|] \to \mathbb{R}$ where $[|\mathcal{X}|] := \{1, ..., |\mathcal{X}|\}$. The first two inputs into the function are the realized data points corresponding to $\mathcal{X}$ and $\mathcal{Y}$ while the second two inputs are user-specified masks which correspond to the set of nodes that the user wishes to include in the conditioning set. For the third input, which we label Y^{mask}, we interpret $S := \{Y_i^{\text{mask}} \text{ s.t. } Y_i^{\text{mask}} = 1\}$. For the fourth input, we can interpret the choice as corresponding to an element $X_j \in \mathcal{X}$ to *not* include in the conditioning set X_{-j}.

When the predictive model is trained by minimizing a loss function ℓ through gradient descent, our strategy will be to amortize the model by randomly masking the inputs when using mini-batching. In particular, for each mini-batch, we can sample $B_k \in \text{Ber}(p)$ and $M \in \text{Cat}(|\mathcal{X}|, q)$ where the probabilities q are just $\frac{1}{|\mathcal{X}|}$ for each component. Then, the sampled data X_j^i and Y_k^i are replaced with perturbed updates:

$$\widetilde{Y}_k^i := Y_k^i \times B_k, \text{ and } \widetilde{X}_j^i := X_j^i \times M_i,$$

where $M_i := \mathbb{1}_{M \neq j}$. The parameter p can be chosen by the user based on the amount of dependency they believe is present within $\mathcal{Y}$. For sparser graphs, the user may wish to choose larger values of p because there is less potential for colliders so a larger conditioning set is more appropriate. If the user suspects $\mathcal{Y}$ to be less sparse, they can choose smaller values of p to bias towards smaller conditioning sets. This process is summarized in Algorithm 1 and can be visualized in Figure 2.

For simplicity, we assume all data is binary-valued. This allows us to only consider interactions between the generated masks and the elements of $\mathcal{X}$ and $\mathcal{Y}$. In the continuous setting, this can be

Algorithm 1 Amortized predictive model training

```
Input: Data: $\mathcal{D}_{\mathcal{X}} \in \{-1,1\}^{n\times|\mathcal{X}|}$; $\mathcal{D}_{\mathcal{Y}} \in \{-1,1\}^{n\times|\mathcal{Y}|}$; $n_{ep}$ (number of epochs), $n_{batch}$ (batch size), $p$ (masking parameter), loss function $\ell\left(\theta, \mathcal{X}, Y_{-k}, Y_k\right)$
for $i = 1$ to $n_{ep}$ do
    for $i = 1$ to $[n/n_{batch}]$ do
        Draw $n_{batch}$ new samples
        Draw $B_m \sim$ Ber$(p)$ for each $Y_m \in Y_{-k}$
        Draw $M \sim$ Cat$(\mathcal{X}, q)$ with equally-weighted probabilities
        Construct $\tilde{Y}_{-k}$ by taking each $Y_m^i$ in the sample and replacing with $\tilde{Y}_k^i := Y_m^i \times B_m$
        Construct $\tilde{\mathcal{X}}$ by taking each $X_j^i$ and replacing with $\tilde{X}_j^i := X_j^i \times \mathbb{1}_{M\neq k}$
        Compute $\frac{\partial \ell(\tilde{\mathcal{X}}, \tilde{Y}_{-k}, Y_k)}{\partial \theta}$
        $\theta \leftarrow \theta - \eta \frac{\partial \ell(\tilde{\mathcal{X}}, \tilde{Y}_{-k}, Y_k)}{\partial \theta}$
    end for
end for
```

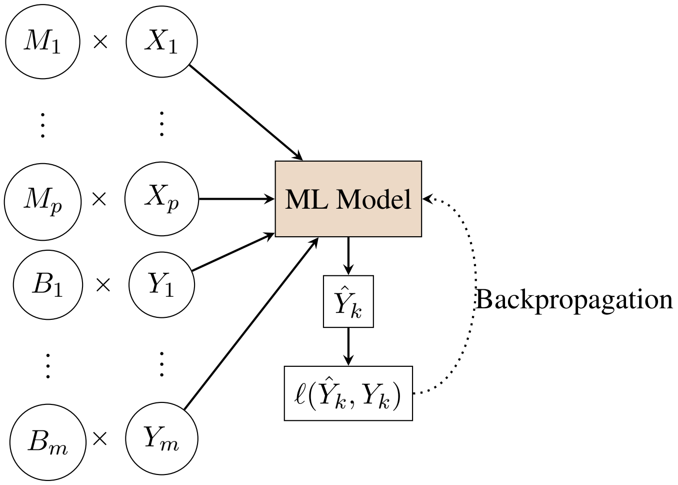

Figure 2: Illustration of masking procedure with mini-batch gradient descent. During model training, masks $B_k \sim$ Ber(p) and $M \in$ Cat$(|\mathcal{X}|, q)$ are sampled during mini-batching to randomly hide nodes within $\mathcal{Y}$. This simulates the process of a user choosing a conditioning subset during model evaluation.

generalized by including *both* the interactions and the masks themselves as inputs into the predictive model. Additional details and empirical results for the continuous case are included in the Appendix.

Note that the above process describes a methodology for computing an estimate of $\mathbb{E}[Y_k|X_{-j}, S]$, but we need a similar procedure for computing an estimate of $\mathbb{E}[X_j|X_{-j}, S]$ for every $X_j \in \mathcal{X}$. We proceed in much the same way as before, but now let $\pi_j : \{-1,1\}^{|\mathcal{X}|} \times \{-1,1\}^{|\mathcal{Y}|} \times \{0,1\}^{|\mathcal{Y}|-1} \to \mathbb{R}$. The user only needs to choose $S \subseteq \mathcal{Y}$ because X_{-j} is conditioned on by default. Algorithm 1 can then be modified to only draw Bernoulli variables and ignore the categorical variable used to mask the elements of $\mathcal{X}$. We again make this precise in the Appendix.

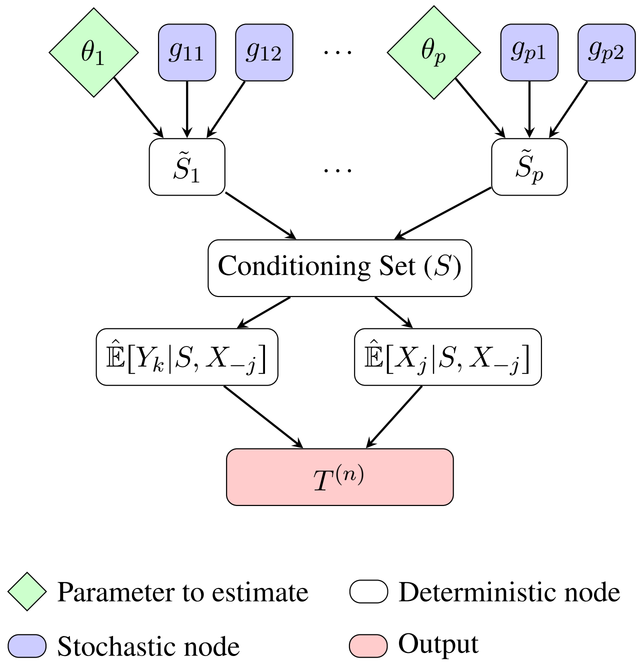

Figure 3: Optimization procedure. Stochastic nodes are sampled from a Gumbel distribution, allowing for back propagation of $\frac{\partial T^{(n)}}{\partial \theta_i}$.

2.3. P-value optimization

In order to find the maximal p-value corresponding to Equation (3), one approach would be to exhaustively search over all possible subsets. However, this will not be computationally tractable for large graphs as the number of conditional independence tests needed to compute a p-value for a single edge scales exponentially with the number of nodes. The approach that we pursue is to learn the conditioning subset that minimizes the test statistic in Equation (5) through numerical optimization. However, since the conditioning set is a discrete rather than continuous variable, standard techniques such as gradient descent do not immediately apply. To overcome this, we use the the Gumbel-Softmax reparamaterization trick (Jang et al., 2017) to express the gradient with respect to the discrete variables we wish to optimize over with continuous relaxations.

Formally, we are learning the parameter $\theta := (\theta_1, ..., \theta_m)$ where the conditioning subset is sampled as $\mathbb{1}_{Y_i \in S} \sim \text{Ber}(\theta_i)$. We then search for the value of θ which minimizes the expected value of Equation (5) through gradient descent. To do this, we approximate $\frac{\partial T^{(n)}}{\partial S} \approx \frac{\partial T^{(n)}}{\partial \tilde{S}}$ where $\tilde{S}$ is constructed as

$$\tilde{S}_i = \frac{\exp\left(\left(\log \theta_i + g_{i1}\right)/\tau\right)}{\exp\left(\log \theta_i + g_{i1}\right) + \exp\left(\log(1-\theta_i) + g_{i2}\right)},$$

with $g_{i1}, g_{i2} \sim \text{Gumbel}(0, 1)$. This approximates a discrete variable, with the quality of the approximation increasing as $\tau \to 0$. There's a trade-off between the quality of the approximation and the variance of the gradient, so we set τ to be large at first and then anneal it over subsequent iterations. The above process is summarized in Figure 3.

As a final step, after some number of iterations where the parameter θ is learned, we need to define a procedure for converting the probabilities to a discrete set of choices $\hat{S}$. A successful search generates a $\hat{S}$ corresponding to a test statistic at least as small as the statistic for any d-separating set $S^\star$. We investigate two different approaches. First, we simply let $\hat{S} = \{i : \hat{\theta}_i > 0.5\}$ after q iterations. This is labeled **Gumbel-Softmax optimization (GSO)** in simulations.

For the second approach, after q_1 iterations, we sample from conditioning subsets without replacement, but with the probability of sampling a set being proportional to its probability given $\hat{\theta}$. Precisely, the weights for each subset S are chosen as $w_S = \prod_i \hat{\theta}_i^{\mathbb{1}_{i \in S}}(1 - \hat{\theta}_i)^{\mathbb{1}_{i \notin S}}$ for every $S \in \mathcal{P}(Y_{-k})$. After q_2 samples, $\hat{S}$ is taken to be the subset yielding the minimal test statistic over the q_2 samples. This is labeled **hybrid approach** in simulations. The hybrid approach tackles the search process in two stages. The first is an exploration stage where a probability distribution over the combinatorial space is learned. The second step explores this space efficiently by prioritizing sets that are most likely under the learned distribution. Empirically, this approach has better results than only using Gumbel-Softmax optimization or the more simplistic approach of sampling without replacement from all possible sets using equal weights, which we explore in detail in Figure 4.

In our experiments, we choose fixed values for the parameters q, q_1, and q_2 — this corresponds to learning parameters with gradient descent and terminating the process after a fixed number of iterations. In principle, the performance of the method may be improved by using other stopping rules for gradient descent, such as terminating after the change in loss is below a threshold. We leave a detailed empirical investigation of this point an open avenue of inquiry.

Early stopping rule By definition, $p_{X_j \to Y_k} \leqslant 1$. If during the search process, we find that there exists an S such that $\hat{p}_{X_j \perp\!\!\!\perp Y_k | S, X_{-j}} > \alpha$ for some pre-determined level α that we know we are not interested in rejecting above, we end the search early by setting $\hat{p}_{X_j \to Y_k} := 1$. This decreases the computational cost of computing p-value for null edges without impacting Type I error control and at the cost of power only at rejection thresholds that are of no practical significance.

3. Results

We perform a benchmark and simulation study centered on a $n = 22,352$ dataset that pairs metastatic events with pre-metastasis tumor mutational info (Nguyen et al. (2022)). Previous studies have looked at the genomic landscape of metastases in different tissues such as brain (Brastianos et al., 2015)) and breast (Brown et al., 2017) metastases. However, this dataset is the only large database that pairs metastatic events with pre-metastasis tumor mutational info across a range of different sites in the body. In total, 234 genes were sequenced for each patient along with 23 secondary metastatic tissue sites (e.g. colon, breast, brain, etc.). Although records are collated across dozens of primary tumor site locations, we focus on the 10 sites with the most patient records availability (breast, colon, liver, lung, ovary, pancreas, rectum, skin, and uterus) to ensure adequate sample size.

This dataset presents an opportunity for statistical models to discover new genomic biomarkers of metastatic potential. For each sequenced gene in a given tumor, we wish to know whether a mutation in that gene has a causal effect on metastasis to another site. Somatic mutations in genes are often highly correlated (Cheng et al., 2015), eliminating the use of simple association tests. Further, tumor colonization in one site may cause eventual metastasis to another site, such as liver metastases in colon cancer (Paschos et al., 2009). It is therefore important to discover biomarkers with direct causal effect from the gene mutation to the specific metastatic site, such that intervention on that biomarker will have a positive effect on patient outcomes. Further, to ensure that discovered biomarkers are reliable, we wish to control the statistical error rate on reported causal links.

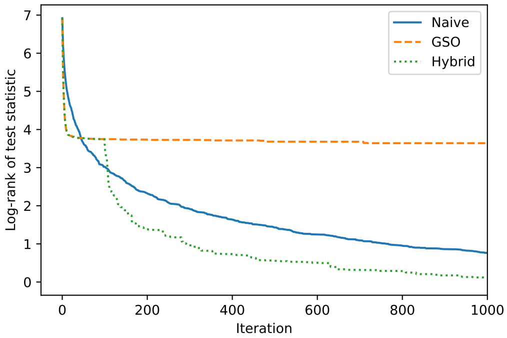

Figure 4: Comparison of the log-rank of the test statistic for each of the methodologies described in Section 2.3 when run on a semi-synthetic dataset created as described in Section 3.2 with $|\mathcal{X}| = 47$ and $|\mathcal{Y}| = 12$. The hybrid approach dominates both the naive and GSO approaches after a sufficient number of iterations.

Since the secondary tumor sites developed only after the primary tumor location has been sequenced, we know a priori that one set of variables (gene mutations in the primary tumor site) cannot be caused by a second set of variables (secondary metastatic events). We therefore can apply the methodology developed in this paper directly, letting $\mathcal{X}$ denote the mutations that have been sequenced and $\mathcal{Y}$ denote the potential secondary metastatic locations.

Modelling approach The underlying predictive models used to construct the GCM test statistics for all the results in this section are logistic regressions with L_2 regularization. Although we experimented with other approaches such as neural networks for learning the regression functions, we found these methods to have similar or lower power compared to a more parsimonious logistic model so these results were omitted. See the Appendix for further comparisons on test statistics constructed from alternative predictive models.

Baselines Our primary baseline is the PC-p algorithm (Strobl et al. (2019)) as it is the only other existing methodology designed for frequentist error coverage. We use the same GCM test statistics described to perform CI tests for this method. We also employ the same simplifying assumptions used by SCSL to streamline the number of conditional independence tests that need to be evaluated — namely, by conditioning on all elements of $\mathcal{X}$ by default and orienting all edges between $\mathcal{X}$ and $\mathcal{Y}$ away from genetic sites and towards metastases.

We also compare to other causal search algorithms implemented by the TETRAD project. Although these methods are not direct competitors to SCSL because they do not produce p-values and therefore cannot be used for false discovery rate control, empirical results suggests that SCSL's performance is still competitive. More information about the benchmark methodologies provided by TETRAD are included in the Appendix.

3.1. Semi-synthetic simulations with real-world confounding

In order to test the method, we need to produce a dataset that matches the structure of the actual data source, but with ground truth knowledge of the causal structure so that performance can be measured. To this end, we propose the construction of a semi-synthetic dataset. The method for data

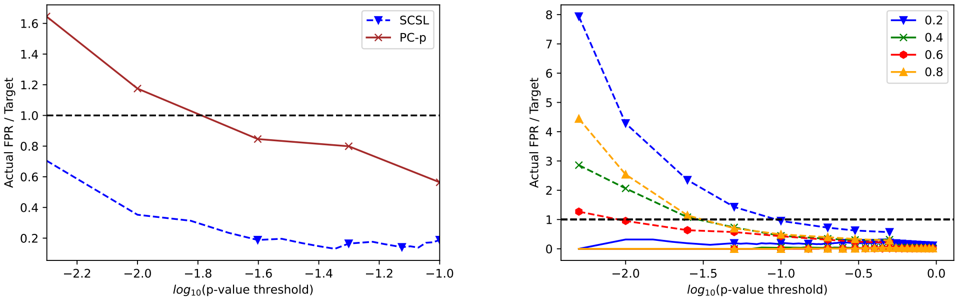

Figure 5: Left hand side shows this ratio on a semi-synthetic dataset generated with real-world confounding which matches the complexity of the real dataset ($|\mathcal{X}| = 47$ and $|\mathcal{Y}| = 23$). Right hand side shows the ratio of actual false positive rate to target rate on the PC-p algorithm (dashed) compared with the proposed methodology (solid) on semi-synthetic datasets generated with synthetic confounding and confoundedness parameter $p \in \{0.2, 0.4, 0.6, 0.8\}$.

construction outlined in this section preserves the joint distribution of $\mathcal{X}$ and $\mathcal{Y}$, but allows for a conditional distribution $P(\mathcal{Y} \mid \mathcal{X})$ that is synthetic.

Dataset construction The algorithm takes the realized datasets $\mathcal{D}_{\mathcal{X}} \in \{0,1\}^{n \times p}$ and $\mathcal{D}_{\mathcal{Y}} \in \{0,1\}^{n \times m}$ and generates a synthetic data for $\mathcal{Y}$ which we denote $\tilde{\mathcal{D}}_{\mathcal{Y}}$. To do this, we sample K features from $\mathcal{X}$ for every element of $\mathcal{Y}$. We let $\mathcal{X}_k^\star$ be the features of $\mathcal{X}$ chosen for a particular Y_k and sample coefficients $\beta_k \in \mathbb{R}^K$ from a $\mathcal{N}(2,1)$ distribution.

Letting $\mathcal{D}^\star_{\mathcal{X},k,i} \in \{0,1\}^K$ be the realized data points corresponding to the chosen features for the ith individual, we generate a logistic likelihood function $f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i}) := \frac{1}{1+\exp(-\beta_k^T \mathcal{D}^\star_{\mathcal{X},k,i})}$. An issue with using this likelihood function directly to generate the data is that the outcome dataset will no longer have the same dependence structure as the original. To circumvent this issue, we use the likelihood function to sample *actual rows* of $\mathcal{D}_{\mathcal{Y}}$. Specifically, we generate $\tilde{\mathcal{D}}_{\mathcal{Y},i} \sim \text{Cat}(\theta_1, ..., \theta_n)$ where $\theta_i \propto \prod_k f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i})$. In other words, each row gets sampled in proportion to its overall likelihood under the assumed model. The procedure is described more explicitly in the Appendix.

Results We first investigate how quickly the methods for optimizing the worst-case p-value described in Section 2.3 converge to the actual minimum, compared with the naive approach of randomly searching over the space in Figure 4. We see that the Gumbel-Softmax approach performs better than the naive approach at first, but then approaches an asymptote as it converges on a solution. The hybrid approach is able to dominate both approaches by allowing for both learning and exploration of the full combinatorial space. In this case, we manually choose to switch to sampling conditioning subsets without replacement at 200 iterations, though in principle it may be possible to learn an optimal time to swap procedures from the data.

Figure 5 (left) shows the Type I error for SCSL and the PC-p algorithm on the generated data. We note that the SCSL algorithm achieves Type I error control, while the PC-p algorithm inflates the Type I error rate when p-values are small. To allow for comparison with methodologies that do not aim at frequentist error control, we also track the F1 score of SCSL along with benchmark methodologies in Table 1. We note that SCSL outperforms existing methods when the dimension of the node set is large relative to the number of observations. Additional details about these experiments and more extensive empirical results are reported in the Appendix.

n	$\lvert\mathcal{X}\rvert$	$\lvert\mathcal{Y}\rvert$	F1 Score									
			SCSL	PC-p	PC	BOSS	CCD	FCI	FGES	GFCI	GRASP	GRaSP-FCI
200	5	5	**0.26**	0.24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10	10	0.07	**0.10**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	15	15	**0.09**	0.07	0.0	0.0	0.0	0.0	0.03	0.03	0.03	0.06
	20	20	0.04	0.04	0.02	**0.11**	0.02	0.02	0.04	0.04	0.06	0.06
2000	5	5	**0.71**	0.38	0.0	0.18	0.0	0.0	0.17	0.17	0.0	0.17
	10	10	**0.30**	0.14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	15	15	**0.12**	0.10	0.0	0.03	0.0	0.0	0.0		0.0	0.03
	20	20	**0.08**	0.06	0.0	0.04	0.0	0.0	0.02		0.04	0.04
20,000	5	5	0.87	0.57	**0.95**	0.84	0.95	0.82	0.95	0.89	0.84	0.89
	10	10	**0.78**	0.37	0.29	0.46	0.29	0.06	0.46	0.24	0.38	0.24
	15	15	**0.49**	0.16		0.15			0.13	0	0.15	0.06
	20	20	**0.33**	0.06		0.06			0.04	0.02	0.08	

Table 1: Comparison between F1 scores of SCSL comapred to existing methods for large-scale causal discovery for synthetic datasets constructed with real-world confounding. SCSL has an advantage over competing methods when the dimension of the node set $(\mathcal{X}, \mathcal{Y})$ is high relative to the sample size. Blank entries indicate the method failed to complete running after 12 hours of computation time.

3.2. Semi-synthetic simulations with synthetic confounding

Causal structure in $\mathcal{Y}$ is the key challenge in our task, as illustrated in Figure 1. To evaluate the robustness of our method in the presence of different degrees of confoundedness, we construct a collection of semi-synthetic datasets where we stochastically control the degree of confounding, allowing us to stress test the methodology under adverse conditions. Compared with Section 3.1, these datasets have the disadvantage of not matching the structure of the actual dataset as closely. However, they are still a valuable point of comparison to assess performance across additional types of confoundedness.

Dataset construction We take as input the actual datasets $\mathcal{D}_{\mathcal{X}} \in \{0,1\}^{n\times p}$ and generates a synthetic dataset for $\mathcal{Y}$ which we label $\tilde{\mathcal{D}}_y$. For each desired Y_k, we choose K features in $\mathcal{X}$ and generate coefficients $\beta_k \in \mathbb{R}^K$ sampled from a $\mathcal{N}(2,1)$ distribution. We store the chosen features in $\mathcal{X}^\star_{Y_k}$ and denote $\mathcal{D}^\star_{\mathcal{X},k,i} \in \{0,1\}^K$ to be the corresponding realized values of the chosen features for the ith individual. We now proceed by generating the elements of $\tilde{\mathcal{D}}_y$ which we call $Y_{k,i}$ in a sequential way. For the first element, the likelihood is defined simply as $f_{Y_{1,i}}(\mathcal{D}^\star_{\mathcal{X},k,i}) := \frac{1}{1+\exp(-\beta_k^T \mathcal{D}^\star_{\mathcal{X},k,i})}$ and now $Y_{1,i} \sim \mathrm{Ber}(f_{Y_{k,1}}(\mathcal{D}^\star_{\mathcal{X},Y_{1,i}}))$ is generated.

For subsequent features $Y_{k,i}$, we also pick features Y_l for $l < k$ with probability p to contribute to the likelihood function. Denote $\mathcal{D}^\star_{\mathcal{Y},Y_k,i}$ as vector containing the chosen Y_l for an individual i and the coefficients, again sampled from a $\mathcal{N}(2,1)$ distribution, as γ_k. Now, define the likelihood as $f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i}, \mathcal{D}^\star_{\mathcal{Y},k,i}) := \frac{1}{1+\exp(-\beta_k^T \mathcal{D}^\star_{\mathcal{X},k,i} - \gamma_k^T \mathcal{D}^\star_{\mathcal{Y},Y_k,i})}$ and again sample $Y_{k,i} \sim \mathrm{Ber}\left(f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i}, \mathcal{D}^\star_{\mathcal{Y},Y_k,i})\right)$. One can think of p as a confounding parameter which makes the causal learning problem more difficult by increasing the number of dependencies in the graph. This algorithm is described in detail in the Appendix.

Results Figure 5 (right) shows the Type I error control for our method and the PC-p algorithm across four different levels of confoundedness ($p \in \{0.2, 0.4, 0.6, 0.8\}$). Across all four levels, the PC-p algorithm inflates the Type I error rate, sometimes by as much as 8x the nominal level. Comparisons in terms of F1 score and power are included in the Appendix, though we note results are broadly similar to those shown in Table 1 for the datasets constructed with real-world confounding.

3.3. Results on real dataset

Finally, we run the methodology on the real data. We stratify the dataset based on the primary tumor site location, and calculate p-values separately within each stratum to identify the secondary tumor location and gene combinations that are significant across different metastases. A rejection set is formed using the Benjamini-Hochberg (BH) (Benjamini and Hochberg, 1995) procedure with target FDR of 0.05 and is shown in Table 2. As a point of comparison, we also calculate the p-values for marginal tests of independence that the original paper used to identify connections. Figure 6 compares the marginal p-values used by the original paper with the causal p-values calculated through SCSL. Applying the BH procedure with the same target threshold results in 161 rejections of marginal p-values. However, only 6 of these 161 remained in the causal rejection set.

The causal mutations discovered have mechanistic evidence in the biology literature. For instance, CDH1 has been mechanistically investigated in breast cancer models of metastasis through its connection to asparagine (Knott et al., 2018). Colon cancers are often treated with EGFR-inhibitors, which are ineffective in the presence of KRAS mutants which continue to activate the MAPK pathway, leading to eventual metastasis (Prenen et al., 2010). TP53 mutations in certain pancreas cancers have been shown to increase fibrosis, enabling tumors to better evade the immune system and increasing metastatic potential (Maddalena et al., 2021). While these mechanistic links between the primary site, gene, and general metastasis are known, the site-specific patterns have not been investigated. Thus, our causal testing results may provide valuable guidance to scientists and clinicians considering the utility of invasive patient monitoring.

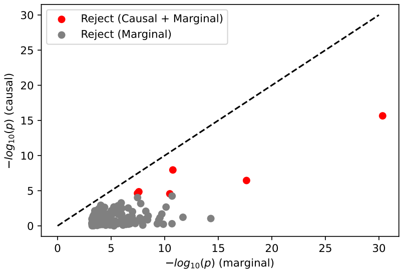

Figure 6: Plot of the 161 discoveries rejected using the BH procedure with target FDR of 0.05 on the marginal independence associations. Only 6 discoveries remain when applying the BH procedure with the same target threshold on the causal p-values.

			p-value	
Primary	Gene	Secondary	Causal	Marginal
Breast	CDH1	Lung	3.5×10^{-7}	2.3×10^{-18}
Colon	KRAS	Lung	1.4×10^{-5}	2.6×10^{-8}
Liver	TERT	Liver	2.3×10^{-5}	3.4×10^{-8}
Lung	EGFR	CNS (Brain)	2.8×10^{-5}	3.3×10^{-11}
Pancreas	KRAS	Lymph	2.2×10^{-16}	4.5×10^{-31}
Pancreas	TP53	Lymph	1.1×10^{-8}	1.7×10^{-11}

Table 2: List of all 6 genes and tumor combinations (corresponding to red markers on right graph) identified as significant at each primary site after forming a rejection set using the Benhamini-Hochberg procedure with target FDR of 0.05. The causal p-values are significantly more conservative, leading to fewer rejections.

4. Conclusion

We introduced a new algorithm for causal discovery which drastically decreases the computational burden required to compute a p-value for a causal relationship between two nodes in a directed acyclic graph with temporally separated sets of variables. We tested this methodology on semi-synthetic data constructed from a recent study on somatic tumor mutations and metastatic potential for a panel of patients and found that the methodology successfully controlled Type I error and had reasonable power across datasets of differing levels of confoundedness. When run on the dataset of Nguyen et al. (2022) , interesting connections between metastases and genes are identified.

Several avenues for follow-up work exist. From a statistical perspective, the p-values generated from the procedure are conservative and power can potentially be improved through post-hoc adjustment to the p-values, for example through the use of Empirical Bayes methods (Efron, 2008). From a computational perspective, more sophisticated probability models could be used to find the worst-case p-value to search the combinatorial space.

Other areas of improvement relate to the causal ordering assumption that edges can only be directed from $\mathcal{X}$ to $\mathcal{Y}$ due to separation in time. For datasets that cannot be partitioned in this way, many aspects of the method can still be used to improve computational and statistical efficiency such as p-value optimization and amortized predictive modeling. In addition to increasing the computational burden of the method, this would leave open the question of how to orient the edges after skeleton discovery when used on more generic datasets. Alternatively, instead of only two groups of temporally seperated nodes, it would be interesting to investigate how this methodology performs when adapted to datasets with several groups of nodes coming into existence over time. Finally, although we have motivated our method from a metastasis dataset, the methodology is general and could be applied to a number of datasets in areas like genome-wide association studies.

References

Yashas Annadani, Jonas Rothfuss, Alexandre Lacoste, Nino Scherrer, Anirudh Goyal, Yoshua Bengio, and Stefan Bauer. Variational causal networks: Approximate bayesian inference over causal structures. *arXiv preprint arXiv:2106.07635*, 2021.

Bryon Aragam and Qing Zhou. Concave penalized estimation of sparse gaussian bayesian networks. *J. Mach. Learn. Res.*, 16(1), 2015.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. of the Royal Statistical Society. Series B (Methodological)*, 57(1): 289–300, 1995.

Priscilla K Brastianos, Scott L Carter, Sandro Santagata, Daniel P Cahill, Amaro Taylor-Weiner, Robert T Jones, Eliezer M Van Allen, Michael S Lawrence, Peleg M Horowitz, Kristian Cibulskis, et al. Genomic characterization of brain metastases reveals branched evolution and potential therapeutic targets. *Cancer Discovery*, 5(11):1164–1177, 2015.

David Brown, Dominiek Smeets, Borbála Székely, Denis Larsimont, A Marcell Szász, Pierre-Yves Adnet, Françoise Rothé, Ghizlane Rouas, Zsófia I Nagy, Zsófia Faragó, et al. Phylogenetic analysis of metastatic progression in breast cancer using somatic mutations and copy number aberrations. *Nature Communications*, 8(1):1–13, 2017.

Emmanuel Candes, Yingying Fan, Lucas Janson, and Jinchi Lv. Panning for Gold: Model-free Knockoffs for High-dimensional Controlled Variable Selection. *J. of the Royal Statistical Society: Series B (Statistical Methodology)*, 80, 10 2016.

Donavan T Cheng, Talia N Mitchell, Ahmet Zehir, Ronak H Shah, Ryma Benayed, Aijazuddin Syed, Raghu Chandramohan, Zhen Yu Liu, Helen H Won, Sasinya N Scott, et al. Memorial Sloan Kettering-Integrated Mutation Profiling of Actionable Cancer Targets (MSK-IMPACT): a hybridization capture-based next-generation sequencing clinical assay for solid tumor molecular oncology. *The J. of Molecular Diagnostics*, 17(3):251–264, 2015.

David Maxwell Chickering. Optimal structure identification with greedy search. *J. Mach. Learn. Res.*, 3:507–554, 2003.

Chris Cundy, Aditya Grover, and Stefano Ermon. BCD nets: Scalable variational approaches for bayesian causal discovery. In *Advances in Neural Information Processing Systems*, volume 34, pages 7095–7110, 2021a.

Chris Cundy, Aditya Grover, and Stefano Ermon. BCD nets: Scalable variational approaches for bayesian causal discovery. In *Advances in Neural Information Processing Systems*, 2021b.

Maxwell D and David Heckerman. Efficient approximations for the marginal likelihood of bayesian networks with hidden variables. *Machine Learning*, 29:181–212, 11 1997.

Bradley Efron. Microarrays, Empirical Bayes and the Two-Groups Model. *Statistical Science*, 23(1): 1 – 22, 2008.

Fei Fu and Qing Zhou. Learning sparse causal gaussian networks with experimental intervention: Regularization and coordinate descent. *J. of the American Statistical Association*, 108(501): 288–300, 2013.

Jelle J. Goeman and Aldo Solari. Multiple hypothesis testing in genomics. *Statistics in Medicine*, 33 (11):1946–1978, 2014.

David Heckerman, Dan Geiger, and David Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017*, 2017.

Simon R. V. Knott, Elvin Wagenblast, Showkhin Khan, Sun Y. Kim, Mar Soto, Michel Wagner, Marc-Olivier Turgeon, Lisa Fish, Nicolas Erard, Annika L. Gable, Ashley R. Maceli, Steffen Dickopf, Evangelia K. Papachristou, Clive S. D'Santos, Lisa A. Carey, John E. Wilkinson, J. Chuck Harrell, Charles M. Perou, Hani Goodarzi, George Poulogiannis, and Gregory J. Hannon. Asparagine bioavailability governs metastasis in a model of breast cancer. *Nature*, 554(7692):378–381, 2018.

Wai-Yin Lam, Bryan Andrews, and Joseph Ramsey. Greedy relaxations of the sparsest permutation algorithm. In James Cussens and Kun Zhang, editors, *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pages 1052–1062. PMLR, 01–05 Aug 2022.

Molei Liu, Eugene Katsevich, Lucas Janson, and Aaditya Ramdas. Fast and powerful conditional randomization testing via distillation. *Biometrika*, 109(2):277–293, 07 2021.

Martino Maddalena, Giuseppe Mallel, Nishanth Belugali Nataraj, Michal Shreberk-Shaked, Ori Hassin, Saptaparna Mukherjee, Sharathchandra Arandkar, Ron Rotkopf, Abby Kapsack, Giuseppina Lambiase, Bianca Pellegrino, Eyal Ben-Isaac, Ofra Golani, Yoseph Addadi, Emma Hajaj, Raya Eilam, Ravid Straussman, Yosef Yarden, Michal Lotem, and Moshe Oren. Tp53 missense mutations in pdac are associated with enhanced fibrosis and an immunosuppressive microenvironment. *Proceedings of the National Academy of Sciences*, 118(23):e2025631118, 2021.

Bastien Nguyen, Christopher Fong, Anisha Luthra, Shaleigh A. Smith, Renzo G. DiNatale, Subhiksha Nandakumar, Henry Walch, Walid K. Chatila, Ramyasree Madupuri, Ritika Kundra, Craig M. Bielski, Brooke Mastrogiacomo, Mark T.A. Donoghue, Adrienne Boire, Sarat Chandarlapaty, Karuna Ganesh, James J. Harding, Christine A. Iacobuzio-Donahue, Pedram Razavi, Ed Reznik, Charles M. Rudin, Dmitriy Zamarin, Wassim Abida, Ghassan K. Abou-Alfa, Carol Aghajanian, Andrea Cercek, Ping Chi, Darren Feldman, Alan L. Ho, Gopakumar Iyer, Yelena Y. Janjigian, Michael Morris, Robert J. Motzer, Eileen M. O'Reilly, Michael A. Postow, Nitya P. Raj, Gregory J. Riely, Mark E. Robson, Jonathan E. Rosenberg, Anton Safonov, Alexander N. Shoushtari, William Tap, Min Yuen Teo, Anna M. Varghese, Martin Voss, Rona Yaeger, Marjorie G. Zauderer, Nadeem Abu-Rustum, Julio Garcia-Aguilar, Bernard Bochner, Abraham Hakimi, William R. Jarnagin, David R. Jones, Daniela Molena, Luc Morris, Eric Rios-Doria, Paul Russo, Samuel Singer, Vivian E. Strong, Debyani Chakravarty, Lora H. Ellenson, Anuradha Gopalan, Jorge S. Reis-Filho, Britta Weigelt, Marc Ladanyi, Mithat Gonen, Sohrab P. Shah, Joan Massague, Jianjiong Gao, Ahmet Zehir, Michael F. Berger, David B. Solit, Samuel F. Bakhoum, Francisco Sanchez-Vega, and Nikolaus Schultz. Genomic characterization of metastatic patterns from prospective clinical sequencing of 25,000 patients. *Cell*, 185(3):563–575, 2022.

Juan Miguel Ogarrio, Peter Spirtes, and Joe Ramsey. A hybrid causal search algorithm for latent variable models. In *Proceedings of the Eighth International Conference on Probabilistic Graphical Models*, volume 52, pages 368–379, 2016.

Elizabeth L. Ogburn, Oleg Sofrygin, Iván Díaz, and Mark J. van der Laan. Causal inference for social network data. *J. of the American Statistical Association*, 0(0):1–15, 2022.

Konstantinos A Paschos, David Canovas, and Nigel C Bird. The role of cell adhesion molecules in the progression of colorectal cancer and the development of liver metastasis. *Cellular Signalling*, 21(5):665–674, 2009.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal Inference by using Invariant Prediction: Identification and Confidence Intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 10 2016. ISSN 1369-7412.

Hans Prenen, Sabine Tejpar, and Eric Van Cutsem. New strategies for treatment of KRAS mutant metastatic colorectal cancer. *Clinical Cancer Research*, 16(11):2921–2926, 2010.

Joseph Ramsey, Peter Spirtes, and Jiji Zhang. Adjacency-faithfulness and conservative causal inference. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, page 401–408, 2006.

Joseph Ramsey, Madelyn Glymour, Ruben Sanchez-Romero, and Clark Glymour. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International J. of Data Science and Analytics*, 3:121–129, 2017a.

Joseph Ramsey, Madelyn Glymour, Ruben sanchez romero, and Clark Glymour. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International Journal of Data Science and Analytics*, 3, 03 2017b. doi: 10.1007/s41060-016-0032-z.

Thomas Richardson. A discovery algorithm for directed cyclic graphs. In *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence*, UAI'96, page 454–461, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc. ISBN 155860412X.

Rajen Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. *Annals of Statistics*, 48, 04 2020.

Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O. Hoyer, and Kenneth Bollen. DirectLiNGAM: A direct method for learning a linear non-gaussian structural equation model. *J. Mach. Learn. Res.*, 12:1225–1248, 2011.

Liam Solus, Yf Wang, Lenka Matejovicova, and Caroline Uhler. Consistency guarantees for permutation-based causal inference algorithms. *Biometrika*, 108, 02 2017. doi: 10.1093/biomet/asaa104.

Peter Spirtes. An anytime algorithm for causal inference. In *Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics*, volume R3 of *Proceedings of Machine Learning Research*, pages 278–285, 2001.

Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000.

Eric V. Strobl, Peter L. Spirtes, and Shyam Visweswaran. Estimating and controlling the false discovery rate of the pc algorithm using edge-specific p-values. 10(5), 2019.

Wesley Tansey, Victor Veitch, Haoran Zhang, Raul Rabadan, and David M. Blei. The holdout randomization test for feature selection in black box models. *J. of Computational and Graphical Statistics*, 31(1):151–162, 2022.

Peter W G Tennant, Eleanor J Murray, Kellyn F Arnold, Laurie Berrie, Matthew P Fox, Sarah C Gadd, Wendy J Harrison, Claire Keeble, Lynsie R Ranker, Johannes Textor, Georgia D Tomova, Mark S Gilthorpe, and George T H Ellison. Use of directed acyclic graphs (DAGs) to identify confounders in applied health research: review and recommendations. *International J. of Epidemiology*, 50(2): 620–632, 12 2020.

Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In *Advances in Neural Information Processing Systems*, 2018.

Appendix A. Omitted Proofs

Proof of Proposition 1 First, assume that there is an edge between X_j and Y_k. Then, X_j and Y_k are not d-separated given any set of nodes $V \subseteq X_{-j} \cup Y_{-k}$. By Assumption 2, X_j and Y_k will be conditionally dependent given V. In particular, they will be conditionally dependent for $S \cup X_{-j}$ for all $S \subseteq Y_{-k}$.

Next, assume that X_j and Y_k are conditionally dependent given $S \cup X_{-j}$ for all $S \subseteq Y_{-k}$. Since no edges are directed from any element in $\mathcal{Y}$ to any element in $\mathcal{X}$, then there are no colliders in W for any $W \subseteq X_{-j}$. This implies that X_j and Y_k are also conditionally dependent given $T \subseteq X_{-j} \cup Y_{-k}$. This implies that X_j and Y_k are not d-separated given any T. Assumption 1 and Assumption 3 then together imply that X_j and Y_k must share an edge.

Appendix B. Algorithms for construction of semi-synthetic datasets

We include the pseudocode for construction of semi-synthetic datasets in Algorithm 3 and Algorithm 2 below.

Algorithm 2 Semi-synthetic dataset with real-world confounding

```
Input: Data: $\mathcal{D}_\mathcal{X} \in \{0,1\}^{n \times p}$ and $\mathcal{D}_\mathcal{Y} \in \{0,1\}^{n \times m}$,
Shuffle rows of $\mathcal{D}_\mathcal{X}$ and store as $\tilde{\mathcal{D}}_\mathcal{X}$
Shuffle rows of $\mathcal{D}_\mathcal{Y}$
Sample $K$ features from $\mathcal{X}$ for each element of $\mathcal{Y}$ and denote $\mathcal{X}^\star_k$ the features for a particular $Y_k$
Create likelihoods for the rows of $\mathcal{Y}$ from $\mathcal{X}^\star$ by
    1. Sampling coefficients $\beta_k \in \mathbb{R}^K$ for $\mathcal{X}^\star_{Y_k}$ (e.g., from a standard normal)
    2. Let $f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i}) := \frac{1}{1+\exp(-\beta_k^T \mathcal{D}^\star_{\mathcal{X},k,i})}$ be the likelihood associated with each response $Y_k$
       and row $i$, where $\mathcal{D}^\star_{\mathcal{X},k,i} \in \{0,1\}^K$ are the realized data corresponding to $\mathcal{X}^\star_k$ for the $i$th
       individual in the dataset.
for row $i = 1$ to $n$ do
    sample $\tilde{\mathcal{D}}_{\mathcal{Y},i} \sim \text{Cat}(\theta_1, \ldots, \theta_n)$ with $\theta_i \propto \prod_k f_{Y_{k,i}}(\mathcal{D}^\star_{\mathcal{X},k,i})$
end for
Return semi-synthetic data $\tilde{\mathcal{D}}_\mathcal{X}$ and $\tilde{\mathcal{D}}_\mathcal{Y}$
```

Appendix C. Adjustments for continuous and mixed-value data

In Algorithm 4, we present a slightly augmented the methodology to train amortized predictive models to accommodate continuous valued data. The algorithm is largely the same as the one presented in the main paper, with the modification that *both* the masks (B, M) and the interactions between the mask are inputs into the loss function. In the case that binary-valued data was used with $X_j, Y_k \in \{-1, 1\}$, then this was not necessary as the interactions would simply lead to the masked versions $\tilde{X}_j, \tilde{Y}_k \in \{-1, 0, 1\}$. In the case of continuous or mixed valued data, however, there is a need to distinguish between data that is 0 because it has been masked or data that is *actually* 0.

Algorithm 3 Semi-synthetic dataset with synthetic confounding

```
Data: $\mathcal{D}_{\mathcal{X}} \in \{0, 1\}^{n \times p}$
Shuffle rows of $\mathcal{D}_{\mathcal{X}}$ and store as $\tilde{\mathcal{D}}_{\mathcal{X}}$
for $k = 1$ to $|\mathcal{Y}|$ do
  Sample $K$ features from $\mathcal{X}$ and denote $\mathcal{X}_k^\star$ the features for a particular $Y_k$
  For each elements $Y_k \in \mathcal{S}_2$ with $l < k$, add $Y_l$ to $\mathcal{Y}_{Y_k}^\star$ with probability $p$.
  Sample coefficients $\beta_k \in \mathbb{R}^K$ for $\mathcal{X}_{Y_k}^\star$ (e.g., from a standard normal) and $\gamma_k \in \mathbb{R}^K$ for $\mathcal{Y}_k^\star$.
  Let $f_{Y_{k,i}}(\mathcal{D}_{\mathcal{X},k,i}^\star, \mathcal{D}_{\mathcal{Y},k,i}^\star) := \frac{1}{1+\exp(-\beta_k^T \mathcal{D}_{\mathcal{X},k,i}^\star - \gamma_k^T \mathcal{D}_{\mathcal{Y},k,i}^\star)}$ the likelihood associated with each response $Y_k$, where $\mathcal{D}_{\mathcal{X},k,i}^\star$ is the realized data corresponding to $\mathcal{X}_k^\star$ and $\mathcal{D}_{\mathcal{Y},k,i}^\star$ is a vector corresponding to the realized data for $\mathcal{Y}_k^\star$ for the $i$th individual in the dataset
  for row $i = 1$ to $n$ do
    sample $\tilde{\mathcal{D}}_{Y_k,i} \sim \text{Ber}(\theta_{i,k})$ with $\theta_{i,k} = f_{Y_{k,i}}(\mathcal{D}_{\mathcal{X},Y_k,i}^\star, \mathcal{D}_{\mathcal{Y},Y_k,i}^\star)$
  end for
end for
Return semi-synthetic data $\tilde{\mathcal{D}}_{\mathcal{X}}$ and $\tilde{\mathcal{D}}_{\mathcal{Y}}$
```

Algorithm 4 Amortized predictive model training for continuous response data

```
Input: Data: $\mathcal{D}_{\mathcal{X}} \in \mathbb{R}^{n \times |\mathcal{X}|}$; $\mathcal{D}_{\mathcal{Y}} \in \mathbb{R}^{n \times |\mathcal{Y}|}$; $n_{ep}$ (number of epochs), $n_{batch}$ (batch size), $p$ (masking parameter), loss function $\ell(\theta, \mathcal{X}, Y_{-k}, Y_k, B, M)$
for $i = 1$ to $n_{ep}$ do
  for $i = 1$ to $[n/n_{batch}]$ do
    Draw $n_{batch}$ new samples
    Draw $B_m \sim \text{Ber}(p)$ for each $Y_m \in Y_{-k}$. Denote $B = \big(B_1, \cdots, B_{k-1}, B_{k+1}, ..., B_{|\mathcal{Y}|}\big)$.
    Draw $M \sim \text{Cat}(\mathcal{X}, q)$ with equally-weighted probabilities
    Construct $\tilde{Y}_{-k}$ by taking each $Y_m^i$ in the sample and replacing with $\widetilde{Y}_m^i := Y_m^i \times B_m$
    Construct $\tilde{\mathcal{X}}$ by taking each $X_j^i$ and replacing with $\widetilde{X}_j^i := X_j^i \times \mathbb{1}_{M \neq k}$
    Compute $\frac{\partial \ell(\theta, \tilde{\mathcal{X}}, \tilde{Y}_{-k}, Y_k, B, M)}{\partial \theta}$
    $\theta \leftarrow \theta - \eta \frac{\partial \ell(\theta, \tilde{\mathcal{X}}, \tilde{Y}_{-k}, Y_k, B, M)}{\partial \theta}$
  end for
end for
```

To construct semi-synthetic datasets to test this dataset, Algorithm 2 and Algorithm 3 can be modified slightly by letting the likelihood functions correspond to a Gaussian distribution instead of a logistic response, and then sampling appropriately.

Appendix D. Flowchart summarizing methodology

In Figure 7, we present a visual flowchart illustrating how to apply the methodology. Assumed inputs into this process are a method for computing p-values corresponding to conditional independence tests. For all of our experiments, we use amortization to efficiently compute the GCM test statistic for different conditioning subsets.

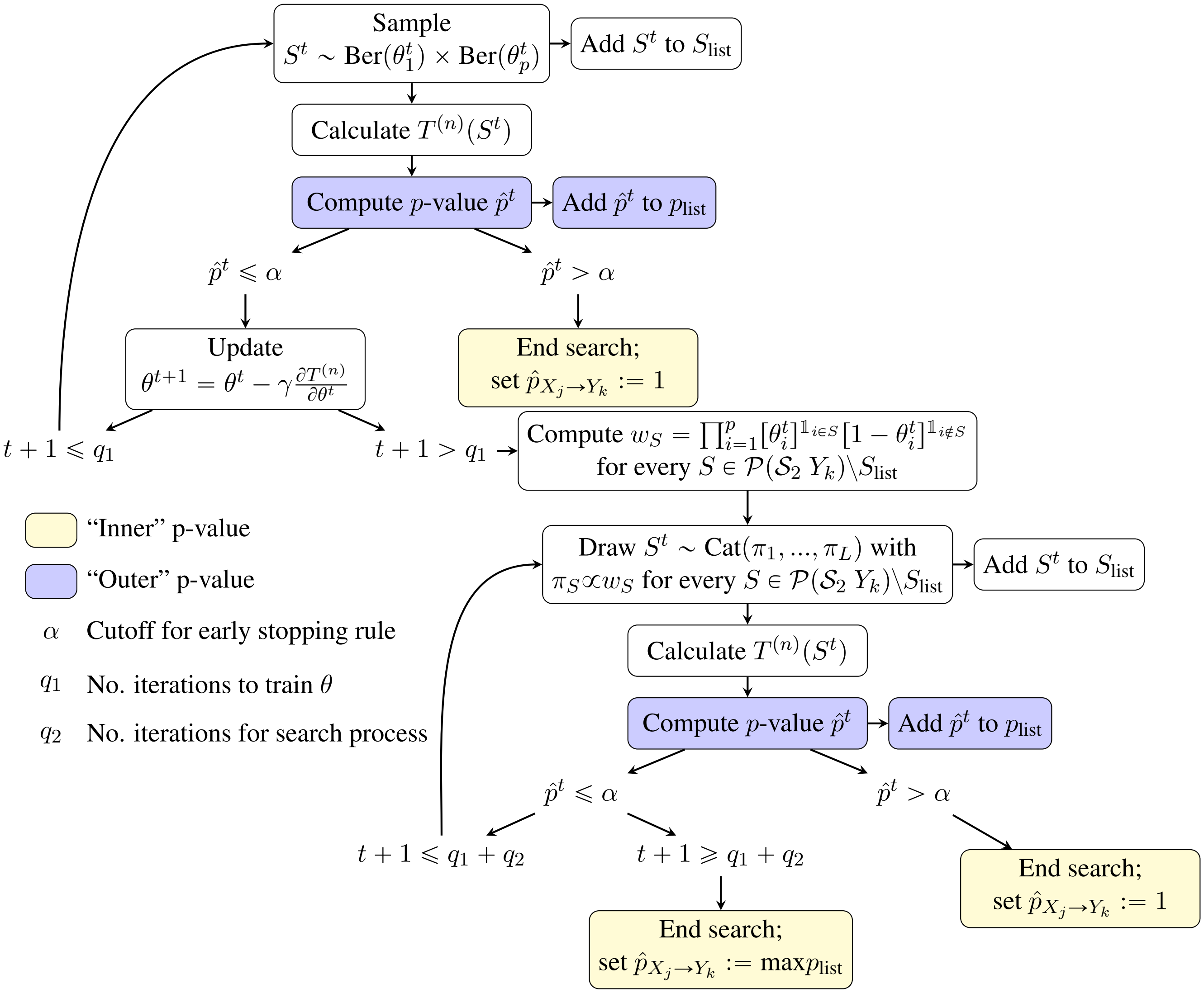
Sample
$S^t \sim \text{Ber}(\theta_1^t) \times \text{Ber}(\theta_p^t)$
Add S^t to S_{list}
Calculate $T^{(n)}(S^t)$
Compute p-value $\hat{p}^t$
Add $\hat{p}^t$ to p_{list}
$\hat{p}^t \leqslant \alpha$
$\hat{p}^t > \alpha$
Update
$\theta^{t+1} = \theta^t - \gamma \frac{\partial T^{(n)}}{\partial \theta^t}$
End search;
set $\hat{p}_{X_j \to Y_k} := 1$
$t + 1 \leqslant q_1$
$t + 1 > q_1$
Compute $w_S = \prod_{i=1}^p [\theta_i^t]^{\mathbb{1}_{i \in S}} [1 - \theta_i^t]^{\mathbb{1}_{i \notin S}}$
for every $S \in \mathcal{P}(\mathcal{S}_2\ Y_k) \backslash S_{\text{list}}$
“Inner” p-value
“Outer” p-value
α Cutoff for early stopping rule
q_1 No. iterations to train θ
q_2 No. iterations for search process
Draw $S^t \sim \text{Cat}(\pi_1, ..., \pi_L)$ with
$\pi_S \propto w_S$ for every $S \in \mathcal{P}(\mathcal{S}_2\ Y_k) \backslash S_{\text{list}}$
Add S^t to S_{list}
Calculate $T^{(n)}(S^t)$
Compute p-value $\hat{p}^t$
Add $\hat{p}^t$ to p_{list}
$\hat{p}^t \leqslant \alpha$
$\hat{p}^t > \alpha$
$t + 1 \leqslant q_1 + q_2$
$t + 1 \geqslant q_1 + q_2$
End search;
set $\hat{p}_{X_j \to Y_k} := 1$
End search;
set $\hat{p}_{X_j \to Y_k} := \max p_{\text{list}}$

Figure 7: Flowchart illustrating the methodology. Nodes colored in blue represent p-values for the individual conditional independence tests corresponding to the “inner” null that $X_j \perp\!\!\!\perp Y_k | S, X_{-j}$. Nodes colored in yellow correspond to p-values testing the “outer” null that $X_j \to Y_k$ is absent.

Appendix E. Technical exposition of Generalized Covariance Measure

We recall the details of the technical conditions described in Shah and Peters (2020) for completeness. We assume that the dataset consists of i.i.d. n samples, with individual observations denoted $\{X_j^i, Y_k^i, S^i, X_{-j}^i\}_{i=1}^n$. Let $\mathcal{P}_0$ denote the family of joint distributions corresponding to the null that $X_j \perp\!\!\!\perp Y_k | S, X_{-j}$.

Define $R_i = \left(X_j^i - \widehat{X}_j^i\right)\left(Y_k^i - \widehat{Y}_k^i\right)$ and the test statistic:

$$T^{(n)} = \frac{\sqrt{n} \cdot \frac{1}{n} \sum_{i=1}^n R_i}{\left(\frac{1}{n} \sum_{i=1}^n R_i^2 - \left(\frac{1}{n} \sum_{r=1}^n R_r\right)^2\right)^{1/2}}. \tag{6}$$

Further denote $\epsilon_{j,P} := X_j - \mathbb{E}_P[X_j|S, X_{-j}]$ and $\epsilon_{k,P} := Y_k - \mathbb{E}_P[Y_k|S, X_{-j}]$.

Fact 1 (Theorem 6 of Shah and Peters (2020)) *Define the following quantities:*

$$A_f := \frac{1}{n}\sum_{i=1}^{n}\left\{X_j^i - \widehat{X}_j^i\right\}^2, \; A_g := \frac{1}{n}\sum_{i=1}^{n}\left\{Y_k^i - \widehat{Y}_k^i\right\}^2,$$

$$B_f := \frac{1}{n}\sum_{i=1}^{n}\left\{X_j^i - \widehat{X}_j^i\right\}^2 \mathbb{E}_P\left(\epsilon_{j,P}^2|S, X_{-j}\right), \textit{ and } B_g := \frac{1}{n}\sum_{i=1}^{n}\left\{Y_k^i - \widehat{Y}_k^i\right\}^2 \mathbb{E}_P\left(\epsilon_{k,P}^2|S, X_{-j}\right).$$

1. *If for $P \in \mathcal{P}_0$, $A_f A_g = o(n^{-1})$, $B_f = o(1)$, $B_g = o(1)$, and $0 \leqslant \mathbb{E}_P\left(\epsilon_{j,P}^2\epsilon_{k,P}^2\right)$, then $\sup_{t\in\mathbb{R}}\left|\mathbb{P}_P\left(T^{(n)} \leqslant t\right) - \Phi(t)\right| \to 0$.*

2. *Let $\mathcal{P} \subseteq \mathcal{P}_0$ denote the set of null distributions such that $A_f A_g = o(n^{-1})$, $B_f = o(1)$, $B_g = o(1)$, $\inf_{\mathcal{P}\in P}\mathbb{E}_p\mathbb{E}_P\left(\epsilon_j^2\epsilon_k^2\right) \geqslant c_1$, and $\sup_{\mathcal{P}\in P}\mathbb{E}_p\mathbb{E}_P\left(\epsilon_{j,P}^2\epsilon_{k,P}^2\right) \leqslant c_2$ for some $c_1, c_2 > 0$. Then $\sup_{P\in\mathcal{P}}\sup_{t\in\mathbb{R}}\left|\mathbb{P}_P\left(T^{(n)} \leqslant t\right) - \Phi(t)\right| \to 0$.*

Appendix F. Additional experimental results

Additional results for each of the semi-synthetic simulations are shown in this section.

Additional information on alternative predictive modelling approaches In addition to using a logistic regression to compute the underlying regression functions in the GCM test statistic, we also experimented with a multi-layer perceptron neural network with with 2 hidden layers comprised of 200 nodes in each layer and dropout regularization. The performance of this method is compared and contrasted with the logistic regression used in the main paper in Figure 8. Since the logistic regression had higher power, we focused on using it as a predictive model in the main paper.

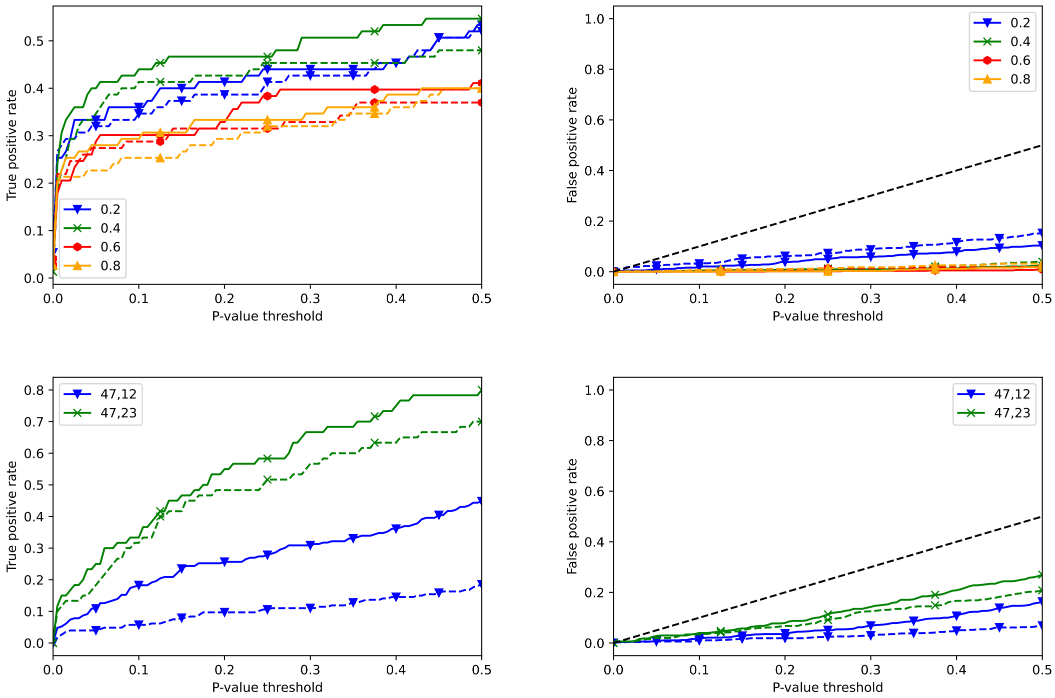

Figure 8: Type I error and Type II error rates for SCSL compared when logistic models (solid) are swapped out for a neural network (dashed). The top row shows results for the datasets with synthetic confounding while the bottom row shows results for datasets constructed with real confounding (note that the tuple (a, b) in the legend indicates the number of variables contained in each node set with $|\mathcal{X}| = a$ and $|\mathcal{Y}| = b$ respectively). The performance across all methods is comparable, though we note the logistic model has higher power in general.

Additional results on power As discussed in the results section, the PC-p algorithm does have higher power than SCSL, but this comes at the expense of Type I error control. This is a natural consequence of the fact that the PC-p algorithm draws an edge between two nodes based on a consideration set of CI tests that is a subset of the tests considered by SCSL. Since we use the same underlying CI test in our comparison of the methodologies, this necessitates that the PC-p algorithm will have strictly lower p-values than SCSL. Nonetheless, the decrease in power resulting from a move to this new methodology is not significant. We compare the power of the two methodologies side by side in Figure 9 for the same configuration of datasets discussed in the results section.

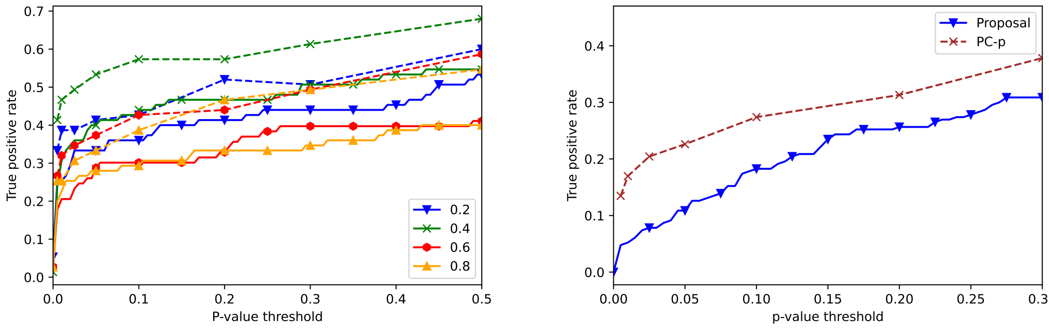

Figure 9: True positive rate for proposed methodology (solid) contrasted with PC-p method (dashed). Left-hand side shows the power for each of these methodologies run on datasets constructed by Algorithm 3 for the same sequence of confoundedness parameters discussed in the main paper with $p \in \{0.2, 0.4, 0.6, 0.8\}$. Right-hand side shows the power for these two methods on the semi-synthetic dataset constructed with Algorithm 2.

Comparisons to procedures without frequentist error guarantees Although the PC-p algorithm is the only existing method that aims at constructing p-values with frequentist error control, we also benchmark this procedure against approaches not aimed at controlling type I error that are implemented in the TETRAD project. The only two methods that scale reasonably to datasets of a similar size as the dataset of Nguyen et al. (2022) are Fast Greedy Equivalence Search (FGES) (Ramsey et al., 2017b) and the Best Order Score Search (BOSS) algorithm (Lam et al., 2022). Other methods used as baselines are Cyclic Causal Discovery (CCD, Richardson (1996)) , Fast Causal Inference (FCI, Spirtes (2001)), Greedy Fast Causal Inference (GFCI, Ogarrio et al. (2016)), Greedy relaxation of the sparsest permutation (GRaSP, Lam et al. (2022)), and GRaSP-FCI (Ogarrio et al., 2016).

We note that above methods produce output in the form of a learned graph rather than a numeric p-value. To facilitate comparisons, we compare the accuracy of the causal graph learned via these methods with a causal graph produced by taking the p-values from the SCSL and PC-p methods and drawing an edge when the p-value is below 0.1. We summarize the accuracy of these methods (in terms of ability to detect edges when present) using F1 score. The F1 scores are shown in Table 3 and overall wall time is shown in Table 4. The experiments are conducted across variety of dataset configurations described in this paper: synthetic versus real-world confounding, degree of confoundedness (p), size of dataset, and size of node set. A similar set of results for continuous-valued datasets constructed using the methods in Appendix C are shown in Table 5 and Table 6.

We note that although the primary strength of SCSL is its ability to control the error rate by producing valid p-values, the experimental results demonstrate that the method is also competitive with existing methods in terms of accuracy, power, and speed of computation. SCSL often has the highest F1 score among the tested methods and is nearly as fast as the BOSS and FGES algorithms.

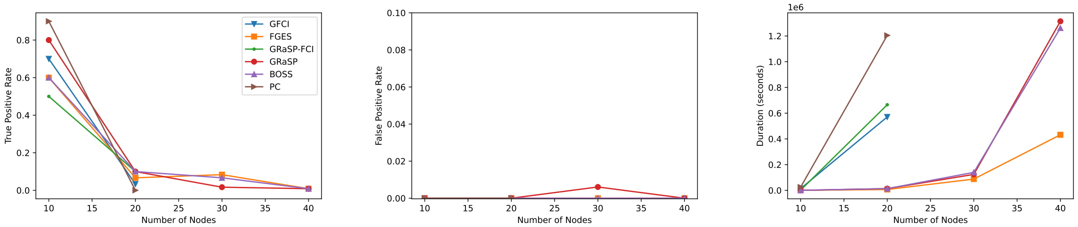

Figure 10: Performance of causal discovery algorithms implemented in TETRAD for semi-synthetic datasets constructed from Algorithm 2. Performance of all methods decreases markedly for larger graphs. Entries are missing if computation time exceeds 36 hours for any method.

						F1 Score									
Algorithm	Data Type	p	n	$\vert\mathcal{X}\vert$	$\vert\mathcal{Y}\vert$	SCSL	PC-p	PC	BOSS	CCD	FCI	FGES	GFCI	GRASP	GRaSP-FCI
Real-World	Discrete		200	5	5	**0.33**	0.29	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Discrete		200	10	10	**0.07**	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Discrete		200	15	15	**0.09**	0.07	0.00	0.00	0.00	0.00	0.03	0.03	0.03	0.06
	Discrete		200	20	20	0.04	0.04	0.02	**0.11**	0.02	0.02	0.04	0.04	0.06	0.06
	Discrete		2,000	5	5	**0.71**	0.48	0.00	0.18	0.00	0.00	0.17	0.17	0.00	0.17
	Discrete		2,000	10	10	**0.28**	0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Discrete		2,000	15	15	**0.13**	0.11	0.00	0.03	0.00	0.00	0.00		0.00	0.03
	Discrete		2,000	20	20	0.04	**0.07**	0.00	0.04	0.00	0.00	0.02		0.04	0.04
	Discrete		20,000	5	5	0.87	0.57	**0.95**	0.84	**0.95**	0.82	**0.95**	0.89	0.84	0.89
	Discrete		20,000	10	10	**0.77**		0.29	0.46	0.29	0.06	0.46	0.24	0.38	0.24
	Discrete		20,000	15	15	**0.44**			0.15		0.00	0.12	0.00	0.15	0.06
	Discrete		20,000	20	20	**0.31**			0.06			0.04	0.02	0.08	
Synthetic	Discrete	0.0	20,000	5	5	**0.77**	0.48	0.75	0.67	0.75	0.75	0.67	0.67	0.67	0.67
	Discrete	0.2	20,000	5	5	0.92	0.52	0.89	0.89	0.89	0.89	**0.95**	0.89	0.89	0.89
	Discrete	0.4	20,000	5	5	**0.71**	0.44	0.57	0.67	0.57	0.57	0.67	0.57	0.67	0.57
	Discrete	0.6	20,000	5	5	**0.75**	0.44	0.57	0.67	0.57	0.57	0.53	0.57	0.67	0.57
	Discrete	0.8	20,000	5	5	**0.82**		0.75	0.75	0.75	0.75	0.75	0.67	0.75	0.67
	Discrete	0.0	20,000	10	10	**0.82**	0.41	0.78	0.78		0.78	0.74		0.77	
	Discrete	0.2	20,000	10	10	0.55	0.31	0.59	**0.60**		0.59	0.56		**0.60**	
	Discrete	0.4	20,000	10	10	0.59	0.29	0.54	**0.70**			0.64		**0.70**	0.54
	Discrete	0.6	20,000	10	10	0.54	0.33	0.57	**0.67**			0.59		**0.67**	
	Discrete	0.8	20,000	10	10	0.57	0.34	0.54	**0.80**			0.57		**0.80**	
	Discrete	0.0	20,000	15	15	**0.67**			0.63			0.63		0.63	
	Discrete	0.2	20,000	15	15	0.68	0.32		**0.71**			0.62		0.69	
	Discrete	0.4	20,000	15	15	0.54			**0.72**			0.48		**0.72**	
	Discrete	0.6	20,000	15	15	0.52	0.23		**0.78**			0.63		0.76	
	Discrete	0.8	20,000	15	15	0.56	0.23		**0.76**			0.51		**0.76**	
	Discrete	0.0	20,000	20	20	0.76			**0.79**			0.77		**0.79**	
	Discrete	0.2	20,000	20	20	0.65			0.82			0.75		**0.83**	
	Discrete	0.4	20,000	20	20	0.42			0.71			0.53		**0.72**	
	Discrete	0.6	20,000	20	20	0.35			**0.67**			0.58		0.65	
	Discrete	0.8	20,000	20	20	0.35			**0.69**			0.63		0.69	

Table 3: Comparison between F1 scores of SCSL with existing methods for large-scale causal discovery for synthetic datasets with discrete-valued entries of $\mathcal{Y}$. Datasets with real-world confounding tend to favor SCSL, while the results are more mixed when using synthetic confounding.

Algorithm	Data Type	p	n	$\lvert\mathcal{X}\rvert$	$\lvert\mathcal{Y}\rvert$	F1 Score									
						SCSL	PC-p	PC	BOSS	CCD	FCI	FGES	GFCI	GRASP	GRaSP-FCI
Real-World	Discrete		200	5	5	**0.33**	0.29	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Discrete		200	5	5	0.00	0.01	0.00	0.00	0.00	0.00	0.00	**0.00**	0.00	0.00
	Discrete		200	10	10	0.00	0.04	0.00	0.00	0.00	0.00	0.00	**0.00**	0.00	0.00
	Discrete		200	15	15	0.00	0.16	0.00	**0.00**	0.00	0.00	0.00	0.00	0.00	0.00
	Discrete		200	20	20	0.00	0.36	0.00	0.00	0.01	0.01	0.00	**0.00**	0.00	0.00
	Discrete		2,000	5	5	0.01	0.05	0.00	**0.00**	0.01	0.00	0.00	0.00	0.00	0.00
	Discrete		2,000	10	10	0.01	0.42	0.01	**0.00**	0.04	0.02	0.00	0.00	0.00	0.00
	Discrete		2,000	15	15	0.01	1.10	0.06	**0.01**	0.17	0.05	0.01		0.01	0.01
	Discrete		2,000	20	20	0.02	1.86	0.10	0.01	0.18	0.07	0.01		0.01	**0.01**
	Discrete		20,000	5	5	0.16	0.73	1.44	0.01	2.86	2.94	**0.01**	1.66	0.01	1.13
	Discrete		20,000	10	10	0.23		9.10	0.05	13.00	8.74	**0.02**	5.29	0.05	4.97
	Discrete		20,000	15	15	0.28			0.11		22.80	**0.05**	10.94	0.11	10.50
	Discrete		20,000	20	20	0.30			0.21			**0.11**	13.86	0.19	
Synthetic	Discrete	0.00	20,000	5	5	0.07	0.19	0.13	0.01	0.53	0.22	**0.00**	0.27	0.00	0.26
	Discrete	0.20	20,000	5	5	0.11	0.22	0.35	0.01	0.91	0.84	**0.00**	1.08	0.01	0.55
	Discrete	0.40	20,000	5	5	0.08	0.16	0.26	0.01	1.23	0.63	**0.00**	0.64	0.01	0.36
	Discrete	0.60	20,000	5	5	0.09	0.19	0.30	0.01	0.97	0.55	0.01	1.36	**0.01**	0.32
	Discrete	0.80	20,000	5	5	0.17		0.21	0.01	1.47	0.46	0.01	0.70	**0.01**	0.54
	Discrete	0.00	20,000	10	10	0.13	3.91	13.35	0.15		20.22	**0.02**		0.09	
	Discrete	0.20	20,000	10	10	0.16	3.56	8.74	0.08		19.54	**0.02**		0.06	
	Discrete	0.40	20,000	10	10	0.22	2.87	7.30	0.12			**0.03**		0.12	18.81
	Discrete	0.60	20,000	10	10	0.16	5.11	10.82	0.21			**0.05**		0.22	
	Discrete	0.80	20,000	10	10	0.22	5.96	10.93	0.31			**0.08**		0.28	
	Discrete	0.00	20,000	15	15	0.30			0.59			**0.05**		0.44	
	Discrete	0.20	20,000	15	15	0.18	16.36		0.94			**0.08**		0.66	
	Discrete	0.40	20,000	15	15	0.25			1.04			**0.18**		0.79	
	Discrete	0.60	20,000	15	15	0.21	17.61		1.81			**0.18**		1.50	
	Discrete	0.80	20,000	15	15	**0.26**	18.47		2.17			0.27		1.41	
	Discrete	0.00	20,000	20	20	0.33			2.19			**0.13**		1.67	
	Discrete	0.20	20,000	20	20	**0.23**			3.04			0.24		2.52	
	Discrete	0.40	20,000	20	20	**0.25**			6.30			0.57		3.69	
	Discrete	0.60	20,000	20	20	**0.30**			5.51			0.72		4.26	
	Discrete	0.80	20,000	20	20	**0.24**			6.03			0.64		4.37	

Table 4: Comparison of wall time (in hours) scores of SCSL with existing methods for large-scale causal discovery for datasets with discrete-valued entries of $\mathcal{Y}$. Datasets with confounding tend to favor SCSL, while the results are more mixed when using confounding. Blank entries note that the method did not finish running after 12 hours of computation time.

Algorithm	Data Type	p	n	$\|\mathcal{X}\|$	$\|\mathcal{Y}\|$	F1 Score									
						SCSL	PC-p	PC	BOSS	CCD	FCI	FGES	GFCI	GRASP	GRaSP-FCI
Synthetic	Continuous	0.40	200	5	5	0.67	0.35	0.46	**0.84**	0.46	0.46	**0.84**		**0.84**	0.57
	Continuous	0.40	200	10	10	0.24	0.24	0.45	0.75	0.45	0.45	**0.77**	0.59	**0.77**	0.67
	Continuous	0.40	200	15	15	0.00	0.08	0.21	**0.75**	0.21	0.21	0.45	0.23	0.75	0.39
	Continuous	0.40	200	20	20	0.10	0.06	0.25	**0.74**	0.25	0.25	0.34	0.19	0.73	0.32
	Continuous	0.40	2000	5	5	0.89	0.55	0.82	**1.00**	0.82	0.82	**1.00**		**1.00**	0.89
	Continuous	0.40	2000	10	10	0.00	0.27	0.24	**0.90**	0.24	0.24	0.61	0.29	**0.90**	0.38
	Continuous	0.40	2000	15	15	0.12	0.20	0.38	**0.93**	0.38	0.38	0.56	0.31	0.91	0.54
	Continuous	0.40	2000	20	20	0.10	0.16	0.28	**0.89**	0.28	0.28	0.49		0.89	0.43
	Continuous	0.00	20000	5	5	0.91	0.57	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
	Continuous	0.20	20000	5	5	**1.00**	0.57	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**		**1.00**	**1.00**
	Continuous	0.40	20000	5	5	0.95	0.57	0.95	**1.00**	0.95	0.95	0.95		**1.00**	**1.00**
	Continuous	0.60	20000	5	5	0.95	0.57	0.75	**1.00**	0.75	0.75	**1.00**	0.75	**1.00**	0.82
	Continuous	0.80	20000	5	5	0.78	0.57	0.62	**1.00**	0.62	0.67	0.53	0.62	**1.00**	0.67
	Continuous	0.00	20000	10	10	**0.98**	0.46	0.95	**0.98**		0.95	0.97			0.97
	Continuous	0.20	20000	10	10	0.88	0.43	0.85	**0.98**	0.85		0.89		**0.98**	0.91
	Continuous	0.40	20000	10	10	0.58		0.75	**1.00**		0.70	0.80		**1.00**	0.75
	Continuous	0.60	20000	10	10	0.08	0.35	0.48	**0.97**	0.48	0.41	0.50		**0.97**	0.60
	Continuous	0.80	20000	10	10	0.12	0.31	0.37	**0.98**	0.37	0.32	0.48		**0.98**	0.29
	Continuous	0.00	20000	15	15	**0.97**			0.96			0.90			
	Continuous	0.20	20000	15	15	0.24		0.55	0.95			0.73		**0.96**	
	Continuous	0.40	20000	15	15	0.15		0.44	**0.98**		0.44	0.66		**0.98**	
	Continuous	0.60	20000	15	15	0.12		0.39	**0.97**		0.39	0.55		**0.97**	
	Continuous	0.80	20000	15	15	0.28		0.32	**0.99**			0.57			
	Continuous	0.00	20000	20	20	0.92			0.97			0.92		**0.98**	
	Continuous	0.20	20000	20	20	0.14			**0.98**			0.52		**0.98**	
	Continuous	0.40	20000	20	20	0.17			**0.98**			0.52		**0.98**	
	Continuous	0.60	20000	20	20	0.18			**1.00**			0.41			
	Continuous	0.80	20000	20	20	0.38			0.97			0.38		**0.98**	

Table 5: Comparison between F1 scores of SCSL with existing methods for large-scale causal discovery for synthetic datasets with continuous-valued entries of $\mathcal{Y}$. SCSL has better performance than other p-value producing methods and has comparable performance to FGES.

Algorithm	Data Type	p	n	$\lvert\mathcal{X}\rvert$	$\lvert\mathcal{Y}\rvert$	F1 Score									
						SCSL	PC-p	PC	BOSS	CCD	FCI	FGES	GFCI	GRASP	GRaSP-FCI
Synthetic	Continuous	0.40	200	5	5	0.00	0.05	0.00	0.00	0.00	0.00	0.00		**0.00**	0.00
	Continuous	0.40	200	10	10	**0.00**	0.22	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01
	Continuous	0.40	200	15	15	**0.00**	0.35	0.02	0.01	0.02	0.01	0.00	0.02	0.01	0.13
	Continuous	0.40	200	20	20	**0.00**	0.67	0.03	0.05	0.03	0.03	0.01	0.11	0.03	0.37
	Continuous	0.40	2,000	5	5	0.01	0.17	0.01	0.00	0.02	0.01	0.00		**0.00**	0.01
	Continuous	0.40	2,000	10	10	0.01	1.13	0.09	0.02	0.19	0.12	**0.01**	0.64	0.01	0.47
	Continuous	0.40	2,000	15	15	**0.01**	3.67	0.33	0.16	0.74	0.41	0.02	14.58	0.10	0.94
	Continuous	0.40	2,000	20	20	**0.01**	6.30	0.70	0.69	1.37	1.02	0.09		0.44	11.23
	Continuous	0.00	20,000	5	5	0.08	2.16	0.60	0.01	1.52	1.09	0.01	1.50	**0.01**	1.72
	Continuous	0.20	20,000	5	5	0.11	2.45	0.36	0.01	0.83	1.34	**0.01**		0.01	0.58
	Continuous	0.40	20,000	5	5	0.18	1.03	0.40	0.01	1.00	0.89	0.02		**0.01**	0.45
	Continuous	0.60	20,000	5	5	0.14	1.09	0.35	0.01	0.83	0.41	0.01	0.90	**0.01**	0.35
	Continuous	0.80	20,000	5	5	0.18	1.22	0.45	0.01	1.53	0.58	0.01	3.09	**0.01**	0.37
	Continuous	0.00	20,000	10	10	0.18	26.54	6.00	0.14		25.02	**0.04**			22.59
	Continuous	0.20	20,000	10	10	0.14	31.70	7.03	0.23	35.60		**0.04**		0.12	33.37
	Continuous	0.40	20,000	10	10	0.24		4.06	0.26		7.72	**0.05**		0.16	14.22
	Continuous	0.60	20,000	10	10	0.18	22.25	5.31	0.46	17.18	6.49	**0.12**		0.27	12.68
	Continuous	0.80	20,000	10	10	0.23	32.75	4.26	0.44	16.09	5.19	**0.10**		0.38	14.91
	Continuous	0.00	20,000	15	15	0.23			0.91			**0.07**			
	Continuous	0.20	20,000	15	15	**0.17**		29.03	1.84			0.27		1.22	
	Continuous	0.40	20,000	15	15	**0.30**		24.83	2.04		34.56	0.34		1.28	
	Continuous	0.60	20,000	15	15	**0.26**		16.28	3.29		28.29	0.41		2.10	
	Continuous	0.80	20,000	15	15	**0.21**		22.17	3.62			0.43			
	Continuous	0.00	20,000	20	20	0.22			4.31			**0.22**		1.98	
	Continuous	0.20	20,000	20	20	**0.30**			6.95			1.45		4.46	
	Continuous	0.40	20,000	20	20	**0.26**			9.94			1.55		5.56	
	Continuous	0.60	20,000	20	20	**0.31**			12.58			1.59			
	Continuous	0.80	20,000	20	20	**0.16**			14.07			1.36		11.42	

Table 6: Comparison of wall time (in hours) scores of SCSL with existing methods for large-scale causal discovery for datasets with continuous-valued entries of $\mathcal{Y}$.