
Higher-Order Causal Structure Learning with Additive Models

Anonymous Author(s)

Affiliation

Address

email

Abstract

Causal structure learning has long been the central task of inferring causal insights from data. Despite the abundance of real-world processes exhibiting higher-order mechanisms, however, an explicit treatment of interactions in causal discovery has received little attention. In this work, we focus on extending the causal additive model (CAM) to additive models with higher-order interaction. This second level of modularity we introduce to the structure learning problem is most easily represented by a directed acyclic hypergraph. We introduce the necessary definitions and theoretical tools to handle the novel structure we introduce and then provide identifiability results for the hyper DAG, extending the typical Markov equivalence classes. We next provide insights into why learning the more complex hypergraph structure may actually lead to better empirical results. In particular, more restrictive assumptions like CAM correspond to easier-to-learn hyper DAGs and better finite sample complexity. We finally develop an extension of the greedy CAM algorithm which can handle the more complex hyper DAG search space and demonstrate its empirical usefulness in synthetic experiments.

1 Introduction

Causal structure learning aims to infer the underlying causal relationships among given variables from observational or interventional data [Spirtes et al., 2001], which is crucial for understanding complex systems and has been widely applied in different fields, including biology [Sachs et al., 2005] and Earth system science [Runge et al., 2019]. Various approaches have been developed for causal discovery, including constraint-based, score-based, and functional causal model-based methods [Glymour et al., 2019].

Constraint-based methods, such as the PC [Spirtes and Glymour, 1991] and FCI [Spirtes et al., 2001] algorithms, rely on conditional independence tests to identify the causal structure. Score-based methods, on the other hand, optimize a scoring function, such as the Bayesian Information Criterion (BIC) [Schwarz, 1978], to find the best causal structure [Chickering, 2002, Singh and Moore, 2005, Yuan et al., 2011, Bartlett and Cussens, 2017]. Both constraint-based and score-based approaches can only identify the underlying causal structure up to Markov equivalence [Spirtes et al., 2001], indicating that they cannot distinguish between different structures that encode the same set of conditional independence relationships.

Functional causal model-based methods address this limitation by introducing proper functional assumptions on the causal relationships, thus enabling the identification of the whole DAG. Examples include the linear non-Gaussian model [Shimizu et al., 2006], additive noise model (ANM) [Hoyer et al., 2008], post-nonlinear causal model [Zhang and Hyvärinen, 2009], heteroscedastic noise model (HNM) [Xu et al., 2022, Immer et al., 2023], and causal additive model (CAM) [Bühlmann et al., 2014]. Among these, CAM assumes that the causal relationships are additive in the variables, which,

despite being more restrictive than the general ANM framework, has been shown to achieve superior performance in various empirical studies [Lachapelle et al., 2020, Zheng et al., 2020, Ng et al., 2022, Rolland et al., 2022], partly owing to its improved statistical power.

In this work, we revisit the additive structural assumption of CAM by incorporating recent developments in training higher-order additive models, extending the functional causal model to explicitly consider the higher-order interactions within the causal mechanisms. Higher-order mechanisms are known to exist in a variety of real-world processes and are believed to be critical into modeling and understanding a number of different scientific phenomena Battiston et al. [2020], Majhi et al. [2022]. Nevertheless, previous approaches have taken an all-or-nothing approach, either (a) directly following CAM-like assumptions or (b) modeling all possible interactions between parents of a child node.

Instead, we find that a directed hypergraph can succinctly represent the necessary structure to interpolate between the simplicity of CAM and the complexity of the full ANM. Specifically, our major contributions are as follows:

1. We develop the theoretical extension from graphs to hypergraphs across three total settings (undirected graphical models, classical DAG models, and additive noise models), and prove the identifiability of the hypergraph structures we introduce.
2. We develop an algorithm for learning the hyper DAG alongside its structural equations directly from data, extending the greedy algorithm for CAM, and showing improved performance over existing approaches on data specifically containing higher-order variable interactions.

2 Hypergraph Methods

In this work we will introduce three different generalizations to existing structure learning approaches which extends the existing graphical representations (Markov networks, Bayesian networks, etc.) to their corresponding hypergraphical representations:

1. Undirected hypergraphical models
2. Directed hypergraphical models for discrete variables (classical regime)
3. Directed hypergraphical models for continuous variables (additive noise model)

We will first introduce the ‘hyper Markov property’ which will be respected by distributions which are ‘Markov’ with respect to a given hypergraph, rather than Markov with respect to a given graph. We emphasize that since hypergraphs are a strict generalization of existing graphical models, we can see this hyper DAG or HDAG structure as an intermediate level of structure between the DAG and the SEM (structural equation model). In that sense, we write:

$$\text{DAGs} \preceq \text{HDAGs} \preceq \text{SEMs} \quad (1)$$

In what follows, we will demonstrate that this more fine-grained structure is not only identifiable directly from data, but also that this perspective allows for greater insights into the identifiability of different hypergraphs (and hence graphs) using finite observations rather than the population limit.

2.1 Undirected Models

Let us write $X \in \mathbb{R}^d$ for some number of dimensions $d \in \mathbb{N}$. We will later choose to restrict to discrete, continuous, or mixed X as appropriate. We write an *undirected* graph as $\mathcal{G}' = (V, E')$ and *undirected* hypergraph as $\mathcal{H}' = (V, H')$, where we take the vertices as $V = [d] := \{1, \dots, d\}$, the undirected edges as $E' \subseteq \{(i, j) : i \neq j \in V\}$, and the undirected hyperedges as $H' \subseteq \{S \subseteq V\}$. We will sometimes abuse notation and write $(i, j) \in \mathcal{G}'$ to mean $(i, j) \in E'$ and similarly for $\mathcal{H}'$. (Note that we are reserving the unprimed versions for the directed versions.)

We will assume throughout this work that we are in the case of fully observed variables. Moreover, we will assume that the density is strictly positive to ensure (a) that there is no confusion caused by switching between the pairwise, local, and global properties; and (b) that the score-based definitions we introduce on the log-probability face no ambiguities in regions of zero density.

Definition 1. Undirected Markov Property. Let us take $N(i)$ to denote the neighbors of $i \in V$. We may say that some distribution $p_X(\mathbf{x})$ is (locally) Markov with respect to some undirected graph $\mathcal{G}'$ if it holds for any i that “ $X_i \perp\!\!\!\perp X_{V-N(i)-\{i\}} \mid X_{N(i)}$ ”, where $-$ denotes set minus. Preparing for our focus on additive models of the log probability, this can equally be required as:

$$p_X(\mathbf{x}) = p_{N(i)}(\mathbf{x}_{N(i)}) \cdot p_i(x_i | \mathbf{x}_{N(i)}) \cdot p_{V-N(i)-\{i\}}(\mathbf{x}_{V-N(i)-\{i\}} | \mathbf{x}_{N(i)}) \quad (2)$$

$$\xi_X(\mathbf{x}) := \log p_X(\mathbf{x}) = \xi_{N(i)}(\mathbf{x}_{N(i)}) + \xi_i(x_i | \mathbf{x}_{N(i)}) + \xi_{V-N(i)-\{i\}}(\mathbf{x}_{V-N(i)-\{i\}} | \mathbf{x}_{N(i)}) \quad (3)$$

where there exists some conditional probabilities p_i and $p_{V-N(i)-\{i\}}$ or some conditional log probabilities ξ_i and $\xi_{V-N(i)-\{i\}}$ such that these equations hold true. This can be additionally written in terms of the clique representation, when we write all cliques of the graph as $Cl(\mathcal{G}') = \{S \subseteq V : S \text{ is a clique in } \mathcal{G}'\}$, as follows:

$$\log p_X(\mathbf{x}) =: \xi_X(\mathbf{x}) = \sum_{S \in Cl(\mathcal{G}')} \xi_S(x_S) \quad (4)$$

Definition 2. Undirected Hyper-Markov Property. It is now straightforward to generalize this property to hypergraphs as follows:

$$\log p_X(\mathbf{x}) =: \xi_X(\mathbf{x}) = \sum_{S \in \mathcal{H}'} \xi_S(x_S) \quad (5)$$

That is, we write the hypergraph edges as specifically representing the energy terms in the log-probability function. It is straightforward to verify that this is strictly more general than hypergraphs which can be created as a result of the maximal clique structure of a typical graph. Nonetheless, in what follows we hope to focus on the identifiability as well as the usefulness of this finer-grained structure for graphical models.

2.2 Directed Classical Models

We will write a directed graph as $\mathcal{G} = (V, E)$ and a directed hypergraph as $\mathcal{H} = (V, H)$ where the directed edges are $E \subseteq \{(k, j) : k \neq j \in V\}$ and the directed hyperedges are $H \subseteq \{(S, j) : j \in V, S \subseteq (V - j)\}$. That is, we are assuming that each hyperedge has only one “out arrow” and up to $|S|$ “in arrows”. It is hoped the purpose for this is relatively clear in the context of a causal diagram which must use several parents to generate a single child. We write the ‘parents of j in $\mathcal{G}$ ’ as $\text{Pa}_{\mathcal{G}}(j) = \{k \in [d] : (k, j) \in \mathcal{G}\}$ and the ‘hyperparents of j in $\mathcal{H}$ ’ as $\text{HypPa}_{\mathcal{H}}(j) = \{S : (S, j) \in \mathcal{H}\}$, where the dependence on $\mathcal{G}$ and $\mathcal{H}$ will be dropped when obvious.

Definition 3. Directed Markov Property. Here, we may once again recall the classical Markov property with respect to a DAG to be written as:

$$\log p(\mathbf{x}) = \sum_{i=1}^d \log p(x_i | \mathbf{x}_{\text{Pa}(i)}) = \sum_{i=1}^d \theta(x_i | \mathbf{x}_{\text{Pa}(i)}) \quad (6)$$

It is very easy to see that we may rewrite this using extraneous functions as:

$$\log p(\mathbf{x}) = \sum_{i=1}^d \sum_{S \subseteq \text{Pa}(i)} \theta(x_i; \mathbf{x}_S) - \mathcal{Z}(\mathbf{x}_{\text{Pa}(i)}) \quad (7)$$

where it is now the case that we do not have the θ energy terms explicitly representing a conditional distribution, but are instead arbitrary functions which are then set to the proper normalization via the $\mathcal{Z}$ function. It can be seen that the $\mathcal{Z}$ function does not explicitly depend on the value of x_i , but normalizes to a distribution based on only the parents alone. The extraneous θ functions which are written as all subsets are useful for the next step generalizing to hypergraph structures.

Definition 4. Directed Hyper-Markov Property. We follow the structure above from the typical DAG framework, but replace the fully-connected parent structure with the more nuanced hypergraph structure. In particular, the energy terms in each of the conditional distributions are replaced with a more specific additive model structure, rather than assuming there is a generic function:

$$\log p(\mathbf{x}) = \sum_{i=1}^d \sum_{S \in \text{HypPa}(i)} \theta(x_i; \mathbf{x}_S) - \mathcal{Z}(\mathbf{x}_{\text{Pa}(i)}) \quad (8)$$

118 It is again straightforward to see that this strictly generalizes the cases which are representable by
 119 the typical DAG. In particular, taking the hyperparents to be all subsets of the parents recovers the
 120 previous functional form (Figure 1f). However, other structures mimicking CAM and LiNGAM type
 121 assumptions are also possible (Figure 1d). Further HDAGs beyond these two existing in the literature
 122 are also possible (Figure 1e). We will further assume ‘causal minimality’ of the HDAG meaning
 123 the hyperparent set is downwards closed w.r.t subsets and each maximal element has a nontrivial θ
 124 function. See the discussion and proofs in the appendix for further details.

125 It is also relatively clear to see how this *directed* hyper-Markov property overlaps with the *undirected*
 126 hyper-Markov property, perhaps moreso than the typical Markov properties. Moreover, it becomes
 127 clear that the moralized graph corresponds to including the $\mathcal{Z}$ terms whereas the skeleton corresponds
 128 to including only the θ terms, see also Table 1. We will make this connection more clear in Section
 129 3.2, where we show identifiability of the HDAG up to its hyper Markov equivalence class (HMEC).

130 2.3 Directed Additive Noise Models

131 For continuous variables, we will generate data from the additive noise model (ANM), meaning that
 132 all variables are a deterministic function of their parent variables, plus an additive noise term.

133 **Definition 5.** *Additive Noise Model.* This may be written as:

$$X_j = f_{\text{Pa}(j) \rightarrow j}(X_{\text{Pa}(j)}) + \varepsilon_j \quad (9)$$

134 Each of the ε_j are taken to be independent, mean-zero random variables.

135 **Definition 6.** *Higher-Order Additive Noise Model.* We may once again generalize to the higher-order
 136 additive model through the use of the structure encoded by the directed hypergraph.

$$X_j = \left(\sum_{S \in \text{HypPa}(j)} f_{S \rightarrow j}(x_S) \right) + \varepsilon_j \quad (10)$$

137 Specifically, we endow the generating function $f_{\text{Pa}(j) \rightarrow j}$ with an additive model structure obeying
 138 the hyperparents of the HDAG. Models like CAM or LiNGAM then correspond to using singleton
 139 hyperparents, whereas the most general ANM corresponds to using the entire block of parents as the
 140 largest hyperparent, as depicted in Figure 1. We will follow CAM Bühlmann et al. [2014] in assuming
 141 Gaussian noise for algorithmic purposes via the minimization of mean-squared error corresponding
 142 to maximizing the log-likelihood; however, surprisingly, we show in Theorem 4 that the our settings
 143 2 and 3 actually overlap in the case of additive Gaussian noise.

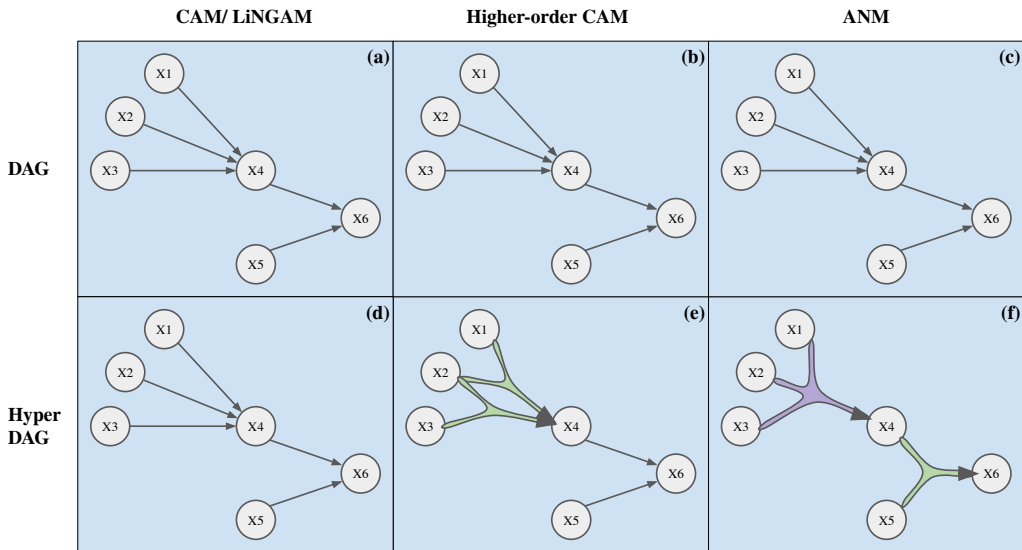


Figure 1: A depiction of the distinguishing power for hypergraphs corresponding to the same DAG.

3 Structure Identifiability

3.1 Undirected Models

First recall that under our assumptions of fully observed variables and strictly positive density, meaning that the density function is identifiable directly from the observed distribution (under mild assumptions like continuity for the continuous variable case) Rosenblatt [1956].

Importantly, then, one may only be concerned in measuring the hypergraph structure as described in Section 2.1; however, this proves to be equally straightforward. For the case of graphical models and mixed-type variables, Zheng et al. [2023] write the generalized precision matrix as:

$$\Omega_{i,j} := \left\| \frac{\partial^2}{\partial_i \partial_j} \log p(x) \right\| := \left(\mathbb{E} \left[\left| \frac{\partial^2 \log p(x)}{\partial_i \partial_j} \right|^2 \right] \right)^{\frac{1}{2}} \quad (11)$$

In the case of hypergraphical models and discrete variables, Enouen and Sugiyama [2024] similarly write the existence of ‘higher-order information’ for some subset $T \subseteq [d]$ (where $T \supsetneq \{i, j\}$ is chosen to imply higher-order) if it is the case that:

$$\Omega_T := \left\| \sum_{S \supseteq T} \theta_S(x_S) \right\| > 0 \quad \text{where} \quad \log p(x) = \sum_S \theta_S(x_S) \quad (12)$$

A straightforward combination of these approaches are sufficient for recovery of the hyper Markov network or undirected hypergraph. Nonetheless, our experiments will instead focus on identification of the directed structure as in the following two sections. Thus, for our purposes it is sufficient to say that the density and log density functions are identifiable directly from the observed distribution.

3.2 Directed Classical Models

Before our main theorem of identifiability extending the result of Verma and Pearl [1990], we must first introduce the notion of multi-dependence to extend the typical notion of conditional independence which is the workhorse of causal structure learning. We will focus on discrete and finite variables as in the classical case [Verma, 1993, Pearl, 2009]; however, most results clearly extend to continuous or mixed variables under mild conditions, and we later discuss one such special case in Theorem 4.

Definition 7. Conditional Multi-dependence. We write that X_i and X_j are dependent if the distribution $\log p(x_i, x_j)$ must be written with a 2D energy term, $\theta_{ij}(x_i, x_j)$, rather than the sum of two 1D energy terms, $\theta_i(x_i) + \theta_j(x_j)$, (corresponding to the product when the log is removed). We will write that X_i, X_j , and X_k are tri-dependent (or generally multidependent), if the distribution $\log p(x_i, x_j, x_k)$ must be written with a 3D energy term, rather than the sum of three 2D energy terms. It can be seen that this does not have a convenient product formulation like the classical case of dependence and independence because of the “mixing” or “torsion” between the three 2D terms. Nonetheless, we will attempt to prove the usefulness of such a definition in the following theorem. Generalization to conditional tests is straightforward.

Theorem 1. The HDAG is identifiable up to the hyper Markov Equivalence classes (HMECs), consisting of all HDAGs with the same “body” and the same (unshielded) “multi-colliders”, paralleling the existing result identifying DAGs up to their skeleton and (unshielded) colliders.

In the same sense that a conditional independence test can never eliminate a causal arrow between two variables, a conditional multi-independence test can never separate a higher-order causal relationship between a set of three or more variables. Removing the arrowheads from the DAG returns the DAG’s skeleton; similarly, removing the arrowheads from the HDAG returns the HDAG’s body, see Table 1 and Figure 2. In some sense, this half of the theorem about the “body identifiability” immediately states that the structure we introduced is identifiable.

Table 1: Notation for hypergraphs

DAG terms	HDAG terms
$\mathcal{G}'$, undirected graph	$\mathcal{H}'$, undirected hypergraph
$\mathcal{G}$, directed acyclic graph (DAG)	$\mathcal{H}$, hyper DAG or HDAG
moralized graph of a DAG	immoralized hypergraph of an HDAG
skeleton of a DAG	body of an HDAG
(unshielded) collider	(unshielded) multicollider

For multicolliders, recall that a collider occurs when there is a conditional dependence which is broken after marginalizing out the child, or equally a conditional independence which is broken when conditioning on the child. The multicollider of an HDAG will occur similarly via a multidependence which is broken after marginalizing out the child. Although collisions between two parents will already be covered, there are cases of three or more parents which are unshielded and can hence be identified from Theorem 1. In particular, there are cases which are not identified in the classical setting, see the RHS of Figure 2. This seeming anomaly is in part due to the historical conflation over time between what structure is recoverable from the conditional independence tests vs. what structure is recoverable from the observed distribution. Indeed, the MEC only describes what is distinguishable via the conditional independence conditions, making it unable to detect what can be seen via the conditional multi-independence test we introduce.

Another key consequence of this different perspective will be a statistical one. In particular, for a K -dimensional energy term in the body of an HDAG, we know that it requires on the order of $\mathcal{O}(n^K)$ samples to be appropriately learned. Consequently, without access to infinite samples, this places further restrictions on the HMEC classes (and hence MEC classes) of ‘distinguishability under finite samples’, whereas MECs are only able to easily represent ‘distinguishability under infinite samples’ as in the asymptotic regime.

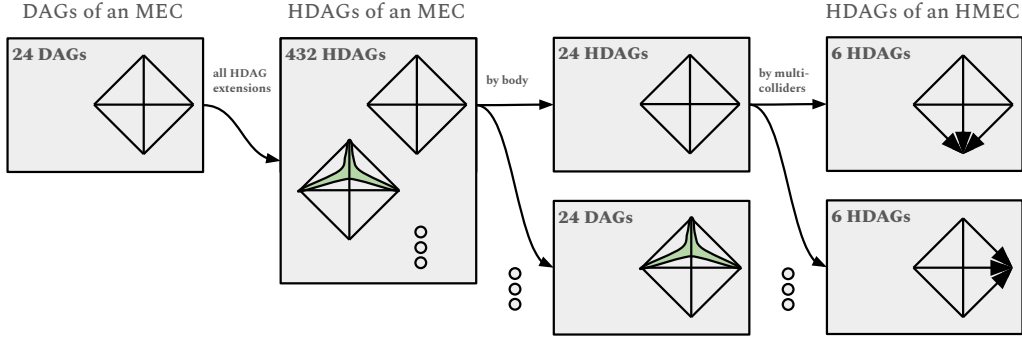


Figure 2: A gradual refinement of the DAGs within a Markov Equivalence Class (MEC) to a stronger refinement of HDAGs based on Theorem 1 to a Hyper Markov Equivalence Class (HMEC). There are $d = 4$ variables with a fully-connected DAG structure. The green triangle represents the third-degree hyperedge in the body of an HDAG. Lack of arrows indicate multiple possible orientations for different DAGs/ HDAGs of the same MEC/ HMEC.

3.3 Directed Additive Noise Models

In this section, we establish identifiability results for recovering the hyper-DAG in the ANM case. For clearer exposition, we first reproduce the arguments of Hoyer et al. [2008] which shows that, in general position, the additive noise model (ANM) is identifiable. We extend their result to a multi-dimensional result which handles the case of multiple parents rather than only the case of one parent node and one child node (slightly different from the extension in Theorem 28 of Peters et al. [2014] because it will more easily generalize to the hypergraph result).

Theorem 2. Let the joint probability densities of $\mathbf{x}$ and y be given by

$$p(x_1, \dots, x_d, y) = p_n(y - f(\mathbf{x})) \cdot p_x(x_1, \dots, x_d) \quad (13)$$

for some noise density p_n , arbitrary density p_x , and function f . Assume further that all such functions are thrice continuously differentiable. If there is also a backwards model which treats x_i as the child,

$$p(x_1, \dots, x_d, y) = p_{\tilde{n}}(x_i - g(x_{-i}, y)) \cdot p_{x_{-i}, y}(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_d, y) \quad (14)$$

for some alternate noise density $p_{\tilde{n}}$, arbitrary density $p_{x_{-i}, y}$, and function g , then it must be case that for $\nu := \log p_n$ and $\xi := \log p_x$, the following differential equation is obeyed

$$\xi''' = \xi'' \cdot \left(-\frac{\nu''' f'}{\nu''} + \frac{f''}{f'} \right) + \left(-2\nu'' f'' f' + \nu' f''' + \frac{\nu' \nu''' f' f''}{\nu''} - \frac{\nu' f'' f''}{f'} \right) \quad (15)$$

where we use ξ', ν', f' as shorthand for $\frac{\partial}{\partial x_i} \xi(x_1, \dots, x_d)$, $\nu'(y - f(\mathbf{x}))$, $\frac{\partial}{\partial x_i} f(x_1, \dots, x_d)$.

219 **Corollary 1.** Assume further that $\nu''' = 0$ and $\frac{\partial^3}{\partial x_i^3} \xi = 0$ for all $i \in [d]$. If a backwards model exists
 220 for a parent x_i , then f is linear in the argument x_i . Further, if a backwards model exists in all parents
 221 x_i , then f is a multilinear function.

222 **Remark 1.** First, this is a general position argument which says that in order to be reversible, the
 223 SEM must obey these particular constraints which usually do not occur. Although this means for
 224 an ‘arbitrary’ SEM model we have identifiability, this does not rule out well-known cases including
 225 linear+Gaussian where the existence of an equivalent backwards model is unavoidable.

226 **Remark 2.** Second, this result is applied locally to a single child node, rather than the global SCM
 227 structure. Previous work has partially explored the global structure in the population limit, see
 228 Chickering [2002] and Peters et al. [2014]; however, a priori, the constrained solutions space may
 229 grow exponentially large, leading to practical limitations in the real-world setting with finite samples.

230 We next provide the relevant extension to recover the hypergraphical structure as well. Indeed, if the
 231 DAG structure is identifiable (at least locally), then the hyper DAG structure is also identifiable (at
 232 least locally). This also implies that when the entire DAG structure is identifiable in the ANM case,
 233 the entire hyper DAG structure is also identifiable.

234 **Theorem 3.** Suppose we have two forward models given by two alternate collections $\mathcal{I} \subseteq \mathcal{P}([d])$
 235 and $\mathcal{J} \subseteq \mathcal{P}([d])$, where $\mathcal{P}$ denotes the power set, with models:

$$p(y|x_1, \dots, x_d) = p_n(y - \sum_{S \in \mathcal{I}} f_S(x_S)) \quad p(y|x_1, \dots, x_d) = p_{\bar{n}}(y - \sum_{T \in \mathcal{J}} g_T(x_T)) \quad (16)$$

236 Assume that in addition to the assumptions of Theorem 2, we also have that the functions f_S and g_T
 237 are differentiable up to order $\max\{\max_{S \in \mathcal{I}}\{|S|\}, \max_{T \in \mathcal{J}}\{|T|\}\}$, or more simply up to order d . It
 238 then follows that $\mathcal{I} = \mathcal{J}$ and thus the hypergraph structures of the two models are exactly the same.
 239 Moreover, outside of trivial modifications to the f ’s and g ’s, the two functional generating models
 240 are exactly the same.

241 Finally, we restrict further to the case of Gaussian noise variables in the ANM to directly recover a
 242 global hypergraphical result, further relating Sections 2.2 and 2.3.

243 **Theorem 4.** Suppose that all additive noise variables are drawn from a Gaussian distribution,
 244 implying that $\nu(\varepsilon) = \frac{-1}{2\sigma^2} \varepsilon^2$ (or some general quadratic form) for all $i \in [d]$. Suppose also that we
 245 generate data according to the directed hypergraph $\mathcal{H}$. The undirected, immoralized version of this
 246 hypergraph $\mathcal{H}'$ is identifiable directly from the data in the sense that $\xi(x_1, \dots, x_d) = \sum_{S \in \mathcal{H}'} \xi_S(x_S)$
 247 for some arbitrary functions ξ_S . In other words, the undirected hypergraph from setting 1 is directly
 248 identifiable from the distribution and thus settings 2 and 3 overlap for ANMs with Gaussian noise.

249 All proofs may be found in the appendix.

250 4 Algorithm

251 Our method for hypergraph discovery and structural equation modeling heavily entwines the schema
 252 of CAM [Bühlmann et al., 2014] and the higher-order interaction techniques of SIAN [Enouen and
 253 Liu, 2022]. Accordingly, we review both algorithms in much greater detail in Appendix B. Here
 254 in the main body of the paper, we briefly review the three step procedure introduced by CAM and
 255 discuss our HCAM extension to their original approach. The first stage is a preselection phase which
 256 searches for good candidate parents for each potential child node. The second stage is the bulk of the
 257 algorithm, greedily constructing the DAG via including each directed edge one at a time based on the
 258 improvement to the log-likelihood. The third stage is a final pruning stage which does not change the
 259 topological order, but simply removes parents which are no longer thought to be relevant.

260 4.1 Step 1: Candidate Search

261 The first stage of CAM Bühlmann et al. [2014] finds candidate edges/ parents by training a GAM
 262 regression on each of the d variables. In their work, this was mainly necessary for them in the
 263 high-dimensional setting, where explicit consideration of so $\mathcal{O}(d^2)$ edges when d is large poses a
 264 challenge. However, for HCAM, this stage becomes absolutely critical. That is because we not only
 265 need to look at all $(d^2 - d)$ candidate directed edges of CAM, but also all the $\frac{1}{2}(d^3 - 3d^2 + 2d)$

266 candidate directed tri-edges, as well as higher-order hyperedges, etc. To consider all hyperedges
 267 directly without any heuristic would require considering an exponential number of hyperedges.

268 Accordingly, we first use a deep neural network to ‘search’ for good hyperedges, following steps 1
 269 and 2 of SIAN [Enouen and Liu, 2022]. That is, for each of the d variables, we regress an MLP DNN
 270 to minimize the mean-squared error (Gaussian likelihood). We then use an XAI technique, called
 271 Archipelago [Tsang et al., 2020], to give an importance score for each of the feature interactions
 272 involving the other variables.

273 4.2 Step 2: Greedy Selection

274 The next phase consists of the bulk of the algorithm and can also be considered the most important
 275 part and yet most simplistic part. A greedy heuristic is taken to gradually build the DAG from the
 276 initialized empty set of vertices. For each of the possible edges (or potentially the subset selected
 277 in step 1), a new likelihood model is trained to simulate adding the edge to the DAG. Importantly,
 278 because of the independence of the ANM noise, this is simply measured as the drop in MSE with and
 279 without the additive term. Gradually, edges are added until some stopping point and the final phase of
 280 pruning begins.

281 In our case, several small adjustments must be made to deal with the case of HDAGs. Importantly,
 282 unlike CAM, HCAM must keep track of both the HDAG and the induced partial-order matrix
 283 simultaneously. Generically, we follow the exact same procedure, training higher-order additive
 284 models with each candidate hyperedge to see the improvement in MSE. We start with 10 candidates
 285 for each of the d variables, based on the ranking provided in step 1, and we replenish the candidates if
 286 there are ever less than 5 viable hyperedges per a variable. This cutdown is able to reduce the number
 287 of higher-order SIAN additive models we train, which helps in improving the overall runtime.

288 4.3 Step 3: Final Pruning

289 Finally, the full model is trained end-to-end once more with all of the included edges from step 2.
 290 The final stage simply removes edges corresponding to additive terms which are too close to zero,
 291 and thus likely to be useless in the causal model. Our extensions prunes in the exact same way,
 292 with higher-order terms corresponding to higher-order edges, but there is no practical difficulty in
 293 doing this extension. The major difficulty of this part is choosing a threshold which corresponds
 294 with a nuisance parameter. The original CAM work uses a threshold of 0.001 for the p-values
 295 provided alongside the GAM models. We use neural networks which do not provide p-values and thus
 296 somewhat similarly threshold based on the MSE of the additive term, using a threshold of $1.0e-4$.

297 5 Experiments

298 We compare across multiple synthetic datasets obeying the additive noise model (ANM) while varying
 299 the degree of the additive models. Following previous works, we generate the base DAG from an
 300 Erdos-Renyi random graph with an average of 4 edges per node. We generate 1D, 2D, and 3D
 301 additive models to distinguish different hypergraphical structures. For 1D models, we avoid the linear
 302 model to allow for identifiability and use a random Gaussian process to define the additive functions.
 303 For 2D and 3D models, we use multilinear terms, $\beta_{jk}x_jx_k$, with coefficients drawn around ± 1 , then
 304 normalizing by the total coefficient weight for a parent set. We assume the additive noise terms are
 305 coming from a Gaussian distribution and draw random variances constrained to be close to 1.0. We
 306 set $d = 30$ (number of nodes) and $n = 10000$ (number of samples) in our experiments.

307 We compare against baselines of PC [Spirtes and Glymour, 1991], GES [Chickering, 2002], BOSS
 308 [Andrews et al., 2023], RESIT [Peters et al., 2014], CAM [Bühlmann et al., 2014], and SCORE
 309 [Rolland et al., 2022]. We additionally compare against a baseline which assumes the empty graph,
 310 equivalent to assuming all of the observed variables are completely independent. We compare against
 311 the structural Hamming distance (SHD) and the structural interventional distance (SID) obtained by
 312 each method on different complexities of synthetic data. We additionally compare the Hamming
 313 distance on the hypergraph bodies, based on the assumption in Figure 1f, calling it the higher-order
 314 structural Hamming distance. Because the maximal error is on the order $2^{30} \approx$ one billion, in some
 315 cases, we do not compute exactly and report a lower bound in Table 4.

Table 2: Structural Hamming Distance for ER4 and N=10,000.

	BOSS	CAM	GES	PC	SCORE	RESIT	zero	HCAM
ER4 1D	189.67±20.24	42.67±4.50	206.67±11.90	159.67±21.17	69.33±20.24	494.67±3.30	128.67±8.38	67.67±13.82
ER4 2D	115.33± 1.70	134.67±2.49	116.00± 2.16	134.00± 4.32	126.00± 2.83	146.33±5.79	115.33±1.70	107.00± 0.82
ER4 3D	109.00± 4.24	120.33±1.89	106.67± 3.86	120.33± 3.86	117.00± 5.89	132.33±6.34	106.67±3.86	106.67± 3.86

Table 3: Structural Intervention Distance for ER4 and N=10,000.

	BOSS	CAM	GES	PC	SCORE	RESIT	zero	HCAM
ER4 1D	608.33±45.09	0.00± 0.00	646.67±54.97	748.33±66.04	105.33±47.12	646.67±54.97	726.00±64.19	525.33±64.25
ER4 2D	679.33±38.94	750.67± 7.93	687.33±39.67	713.67±60.18	745.67±16.11	734.67±21.64	679.33±38.94	661.00±31.03
ER4 3D	647.67±15.08	744.67±22.23	679.00±39.02	699.00±46.31	751.67±24.14	744.00±29.44	679.00±39.02	679.00±39.02

Table 4: Higher-Order Structural Hamming Distance for ER4 and N=10,000.

	BOSS	CAM	GES	PC	SCORE	RESIT	zero	HCAM
ER4 1D	>10,000	42.67± 4.50	>10,000	>1,000	>10,000	>10,000	128.67± 8.38	48.33± 8.18
ER4 2D	157.33± 1.70	168.00± 5.35	158.00± 1.41	183.67±10.78	171.33± 2.05	602.33±603.66	157.33± 1.70	119.00± 5.35
ER4 3D	236.00±18.38	256.33±14.66	248.67±18.15	264.67±23.23	264.00±14.76	263.00± 13.44	248.67±18.15	248.67±18.15

5.1 Results

Overall, we find that many methods are successful for the simpler 1D data obeying the CAM assumptions, especially the CAM and SCORE algorithms. HCAM does not achieve the same level of success as these algorithms but still achieves good performance on this dataset. Surprisingly, all other methods have rather great difficulty in identifying the causal structure.

On our specifically higher-order datasets, however, we find that the story is quite different. In particular, the only algorithm which is able to defeat the baseline on the 2D data is our HCAM method specifically focusing on modeling the 2D interactions. In the 3D data, none of the algorithms we run are able to find empirical success over the zero baseline. That is to say, we should have just assumed the variables were independent and not run our algorithm at all.

This lack of capability is despite the fact that we used a simple DGP (multilinear plus Gaussian) on a relatively low number of variables ($d = 30$) and provided a relatively standard number of observations ($n = 10,000$). Aligning with our hypothesis that the statistical complexity increases in the presence of higher-order interactions, we strongly believe this points to some aspects of structure discovery research which have received less attention but remain highly influenced by the presence of interactions.

6 Conclusion

We have introduced a framework for considering the impact of higher-order interactions on causal structure. After introducing the relevant definitions, we further show the identifiability of the introduced structure across multiple cases of interest. Finally, we demonstrate the potential usefulness of the hypergraphical structure in empirical case using the additive noise model, along with providing a first algorithm for adequately handling the higher-order structure directly from observed data. This perspective additionally allowed us to identify a potential blindspot of many existing structure discovery approaches: their lack of focus on statistical power and lesser ability to handle higher-order interactions.

We envision future work may continue to benefit from a perspective using higher-order interactions. Some directions of future exploration include improving on the algorithms and theoretical results presented herein, potentially solving increasingly challenging datasets constructed using the higher-order perspective on the generated variables. Extension to appropriately handling latent variables and latent confounding is a direction of serious interest. Hypergraphical structures being identifiable from the data distribution alone, extending existing MEC and identifiability results, point to the potential of hypergraphical structure across even more contexts and settings than the ones explored herein.

References

- Steen A. Andersson, David Madigan, and Michael D. Perlman. A characterization of markov equivalence classes for acyclic digraphs. *The Annals of Statistics*, 25(2):505–541, 1997. ISSN 00905364, 21688966. URL <http://www.jstor.org/stable/2242556>.
- Bryan Andrews, Joseph Ramsey, Ruben Sanchez Romero, Jazmin Camchong, and Erich Kummerfeld. Fast scalable and accurate discovery of DAGs using the best order score search and grow shrink trees. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=80g3Yql01a>.
- Mark Bartlett and James Cussens. Integer linear programming for the Bayesian network structure learning problem. *Artificial Intelligence*, 244:258–271, 2017.
- Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, Jean-Gabriel Young, and Giovanni Petri. Networks beyond pairwise interactions: Structure and dynamics. *Physics Reports*, 874:1–92, 2020. ISSN 0370-1573. doi: <https://doi.org/10.1016/j.physrep.2020.05.004>. URL <https://www.sciencedirect.com/science/article/pii/S0370157320302489>. Networks beyond pairwise interactions: Structure and dynamics.
- Peter Bühlmann, Jonas Peters, and Jan Ernest. CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526 – 2556, 2014. doi: 10.1214/14-AOS1260. URL <https://doi.org/10.1214/14-AOS1260>.
- David M. Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3(Nov):507–554, 2002.
- David Maxwell Chickering. A transformational characterization of equivalent bayesian network structures. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI’95, page 87–98, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603859.
- James Enouen and Yan Liu. Sparse interaction additive networks via feature interaction detection and sparse selection. In *Advances in Neural Information Processing Systems*, 2022.
- James Enouen and Mahito Sugiyama. A complete decomposition of kl error using refined information and mode interaction selection, 2024. URL <https://arxiv.org/abs/2410.11964>.
- Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019.
- Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2008. URL https://proceedings.neurips.cc/paper_files/paper/2008/file/f7664060cc52bc6f3d620bcdcd94a4b6-Paper.pdf.
- Alexander Immer, Christoph Schultheiss, Julia E Vogt, Bernhard Schölkopf, Peter Bühlmann, and Alexander Marx. On the identifiability and estimation of causal location-scale noise models. In *International Conference on Machine Learning*, 2023.
- Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=rklbKA4YDS>.
- Po-Ling Loh and Peter Bühlmann. High-dimensional learning of linear causal networks via inverse covariance estimation. *Journal of Machine Learning Research*, 15(88):3065–3105, 2014. URL <http://jmlr.org/papers/v15/loh14a.html>.
- Soumen Majhi, Matjaž Perc, and Dibakar Ghosh. Dynamics on higher-order networks: a review. *Journal of The Royal Society Interface*, 19(188), Mar 2022. ISSN 1742-5662. doi: 10.1098/rsif.2022.0043. URL <http://dx.doi.org/10.1098/rsif.2022.0043>.

Ignavier Ng, Shengyu Zhu, Zhuangyan Fang, Haoyang Li, Zhitang Chen, and Jun Wang. Masked gradient-based causal structure learning. In *SIAM International Conference on Data Mining (SDM)*, 2022.

Judea Pearl. *Causality*. Cambridge University Press, 2 edition, 2009.

Jonas Peters, Joris M. Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15(58):2009–2053, 2014. URL <http://jmlr.org/papers/v15/peters14a.html>.

Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russell, Dominik Janzing, Bernhard Schölkopf, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18741–18753. PMLR, 17–23 Jul 2022.

Murray Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832 – 837, 1956. doi: 10.1214/aoms/1177728190. URL <https://doi.org/10.1214/aoms/1177728190>.

Jakob Runge, Sebastian Bathiany, Erik M. Bollt, Gustau Camps-Valls, Dim Coumou, Ethan R. Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, J. Peters, Rick Quax, Markus Reichstein, Marten Scheffer, Bernhard Scholkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in Earth system sciences. *Nature Communications*, 10, 2019.

Karen Sachs, Omar Perez, Dana Pe’er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

Gideon Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

Shohei Shimizu, Patrik O. Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(72):2003–2030, 2006. URL <http://jmlr.org/papers/v7/shimizu06a.html>.

Ajit P. Singh and Andrew W. Moore. Finding optimal Bayesian networks by dynamic programming. Technical report, Carnegie Mellon University, 2005.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT press, 2nd edition, 2001.

Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review*, 9:62–72, 1991.

Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. *arXiv preprint arXiv:2006.10965*, 2020.

T. S. Verma. Graphical aspects of causal models. Technical report, UCLA Cognitive Systems Laboratory, 1993.

Thomas Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, UAI ’90*, page 255–270, USA, 1990. Elsevier Science Inc. ISBN 0444892648.

Sascha Xu, Alexander Marx, Osman Mian, and Jilles Vreeken. Causal inference with heteroscedastic noise models. In *Proceedings of the AAAI Workshop on Information Theoretic Causal Inference and Discovery*, 2022.

Changhe Yuan, Brandon Malone, and Xiaojian Wu. Learning optimal Bayesian networks using A* search. In *International Joint Conference on Artificial Intelligence*, 2011.

Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Conference on Uncertainty in Artificial Intelligence*, 2009.

- 441 Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. Learning sparse
442 nonparametric DAGs. In *International Conference on Artificial Intelligence and Statistics*, 2020.
- 443 Yujia Zheng, Ignavier Ng, Yewen Fan, and Kun Zhang. Generalized precision matrix for scalable
444 estimation of nonparametric markov networks. In *The Eleventh International Conference on*
445 *Learning Representations*, 2023. URL <https://openreview.net/forum?id=qBvBycTqVJ>.

A Proofs of Theorems

A.1 Proof of Theorem 1

To begin, we remind that while the classical MEC formulation is concerned with mapping the conditional independencies of a distribution with the Markov conditions and structure of a DAG, we are here concerned with mapping the conditional multi-independencies of a distribution with the Markov conditions and structure of an HDAG. Accordingly, it is first worth noting that because the conditional multi-independencies of a distribution is a strictly larger set of conditions than the original set of conditions, hence, we get the existing result on the DAG corresponding to the HDAG, let's call it the reduced DAG. This is done in the obvious way by taking the union of all a node's hyperparents and defining them as the parent set.

Thus, in addition to the skeletons and (unshielded) colliders of the DAG which are already identifiable from the typical conditions, we must investigate which HDAGs are further distinguishable via these conditions and which HDAGs are distinguishable via the new conditions.

We will start with the easier point about the body of an HDAG. Once again, this is defined by removing all directed arrows from the directed hypergraph. In the same way that a pair of nodes i and j are 'inseparable' if they are not conditionally independent for any conditioning set, we can say that a triple of nodes i , j , and k are inseparable if they are not conditionally multi-independent for any conditioning set. In much the same way this indicates the existence of an edge in the DAG case, this will indicate the existence of a (three-dimensional) hyperedge in the HDAG case.

It is thus straightforward to see that the existence of an inseparable triple shows the existence of a directed hyperedge (where one of the three vertices is the child). Further, this generalizes to all degrees in the exact same way. It is briefly reminded that the hierarchy constraint plays a role of convenience here in the sense that a higher-order edge is detected via a three-dimensional hyperedge even if, say, the generating equations do not make this explicit. This exactly parallels what happens in the 2D case with an inseparable pair of nodes, where the DAG edges are capturing everything 'between i and j or higher'. To be explicit, the DAG edge $i \rightarrow j$ could have a second parent of j which interacts with i when generating j . Nonetheless, it is clear from these conditions that we may directly identify the body of the HDAG, and that the body of the HDAG is strictly more informative than the skeleton of the HDAG's reduced DAG.

Now let us move on to the discussion of colliders between parents. To prepare for our generalization of colliders, we first allude to the fact that in Equation 7, we can see directly that the normalizing $\mathcal{Z}$ score over the parent set is the cause of a collider. In particular, unlike the natural θ terms which cannot be destroyed via marginalization of variables, the $\mathcal{Z}$ terms are destructible under marginalization of the child. This naturally corresponds to the more typical conditions noting that there is some smaller set (not including the child) where conditioning provides independence but conditioning on the child additionally breaks the independence. Of course, not all sets without the child included is able to marginalize out the child, namely, conditioning on any descendant of the child is also problematic.

Nonetheless, we may proceed by extending the definition in the same way. We say that a set of nodes i , j , k alongside a fourth node ℓ are a tricollider so long as there exists some conditioning set S under which the i , j , k are not conditionally tri-dependent; however, after additionally conditioning on the node ℓ (their joint child), the tri-independence breaks and i , j , k are conditionally tri-dependent when conditioning on $(S + \{\ell\})$. Equally, it can be seen that i , j , k are the joint parents of ℓ whose $\mathcal{Z}$ normalization appears only when needing to condition on ℓ .

In fact, it is now extremely straightforward to state our faithfulness condition directly. We say that a distribution observes a hypergraph structure faithfully so long as no natural theta term is destructible via marginalization; moreover, the joining of two theta terms via marginalization will lead to a new theta term with the maximal relative size (i.e. no higher-order terms magically cancel and zero out).

Finally, because of this faithfulness condition, we can equally start with the log-probability score function which obeys the Hyper Markov property and construct all possible multi-dependence tests in the backwards direction. Accordingly, no θ term will cancel via marginalization and each $\mathcal{Z}$ will only cancel via marginalization of its respective child. As a reminder, conditioning on a set is easier via implicitly pulling out the variables value in the conditional distribution, and marginalizing has been made possible via the faithfulness distribution. Together, these allow us to describe all

of the energy terms of a new conditioned distribution and one can directly read off the conditional multi-dependence via the existence of or lack of the highest-order energy term.

A.2 Proof of Theorem 2

Proof. The arguments here closely follow the original arguments for Theorem 1 of Hoyer et al. [2008].

First, recall that we will write

$$\begin{aligned}\pi(x_{[d]}, y) &= \nu(y - f(x)) + \xi(x) \\ \pi(x_{[d]}, y) &= \tilde{\nu}(x_i - g(x_{-i}, y)) + \eta(x_{-i}, y)\end{aligned}\tag{17}$$

We may first proceed with some basic calculations

$$\begin{aligned}\frac{\partial}{\partial x_i} \pi &= \nu' \cdot -\frac{\partial f}{\partial x_i} + \frac{\partial \xi}{\partial x_i} \\ &= \tilde{\nu}' \cdot 1 + 0 \\ \frac{\partial}{\partial y} \pi &= \nu' \cdot 1 + 0 \\ &= \tilde{\nu}' \cdot -\frac{\partial g}{\partial y} + \frac{\partial \eta}{\partial y}\end{aligned}$$

And further

$$\begin{aligned}\frac{\partial^2}{\partial x_i^2} \pi &= \nu'' \cdot \frac{\partial f}{\partial x_i} \cdot \frac{\partial f}{\partial x_i} - \nu' \cdot \frac{\partial^2 f}{\partial x_i^2} + \frac{\partial^2 \xi}{\partial x_i^2} \\ &= \tilde{\nu}'' \\ \frac{\partial^2}{\partial x_i \partial y} \pi &= \nu'' \cdot -\frac{\partial f}{\partial x_i} \\ &= \tilde{\nu}'' \cdot 1 \cdot -\frac{\partial g}{\partial y} + \tilde{\nu}' \cdot 0 + 0 = -\tilde{\nu}'' \cdot \frac{\partial g}{\partial y}\end{aligned}$$

So it follows from the $\tilde{\nu}$ equation that

$$\frac{\frac{\partial^2 \pi}{\partial x_i^2}}{\frac{\partial^2 \pi}{\partial x_i \partial y}} = \frac{\tilde{\nu}''}{-\tilde{\nu}'' \cdot \frac{\partial g}{\partial y}} = \frac{-1}{\frac{\partial g}{\partial y}}$$

And further

$$\frac{\partial}{\partial x_i} \left[\frac{\frac{\partial^2 \pi}{\partial x_i^2}}{\frac{\partial^2 \pi}{\partial x_i \partial y}} \right] = \frac{\partial}{\partial x_i} \left[\frac{-1}{\frac{\partial g}{\partial y}} \right] = 0$$

Plugging this back in to the equation with ν gives us

$$\frac{\partial}{\partial x_i} \left[\frac{\nu'' \cdot f' \cdot f' - \nu' f'' + \xi''}{-\nu'' f'} \right] \equiv 0$$

Application of repeated derivative rules and simplification gives the required

$$\begin{aligned}\xi''' &= \xi'' \cdot \left(-\frac{\nu''' f'}{\nu''} + \frac{f''}{f'} \right) + \\ &\quad \left(-2\nu'' f'' f' + \nu' f''' + \frac{\nu' \nu''' f' f''}{\nu''} - \frac{\nu' f'' f''}{f'} \right)\end{aligned}$$

□

512 A.3 Proof of Theorem 3

513 *Proof.* Supposing that we have two different forward models given by

$$p(x_1, \dots, x_d, y) = p_n(y - \sum_{S \in \mathcal{I}} f_S(x_S)) \cdot p_x(x_1, \dots, x_d) = p_n(y - f(x)) \cdot p_x(x_1, \dots, x_d)$$

514

$$p(x_1, \dots, x_d, y) = p_{\tilde{n}}(y - \sum_{T \in \mathcal{J}} g_T(x_T)) \cdot p_x(x_1, \dots, x_d) = p_{\tilde{n}}(y - g(x)) \cdot p_x(x_1, \dots, x_d)$$

515 We may take π as before and see that

$$\begin{aligned} \frac{\partial}{\partial y} \pi &= \nu' = \tilde{\nu}' \\ \frac{\partial}{\partial x_i} \pi &= -\nu' \cdot \left[\sum_{S \in \mathcal{I}} \frac{\partial f_S}{\partial x_i}(x_S) \mathbb{1}(i \in S) \right] = -\tilde{\nu}' \cdot \left[\sum_{T \in \mathcal{J}} \frac{\partial g_T}{\partial x_i}(x_T) \mathbb{1}(i \in T) \right] \end{aligned}$$

516 It follows as before that we may write

$$-\left[\frac{\frac{\partial}{\partial x_i} \pi}{\frac{\partial}{\partial y} \pi} \right] = \left[\sum_{S \in \mathcal{I}} \frac{\partial f_S}{\partial x_i}(x_S) \mathbb{1}(i \in S) \right] = \left[\sum_{T \in \mathcal{J}} \frac{\partial g_T}{\partial x_i}(x_T) \mathbb{1}(i \in T) \right]$$

517 Further, we have that

$$\begin{aligned} 0 &\equiv \frac{\partial}{\partial x_{R-i}} \left[0 \right] = \frac{\partial}{\partial x_{R-i}} \left[\frac{\frac{\partial}{\partial x_i} \pi}{\frac{\partial}{\partial y} \pi} - \frac{\frac{\partial}{\partial x_i} \pi}{\frac{\partial}{\partial y} \pi} \right] = \frac{\partial}{\partial x_{R-i}} \left[-\frac{\partial f}{\partial x_i} + \frac{\partial g}{\partial x_i} \right] = -\frac{\partial f}{\partial x_R} + \frac{\partial g}{\partial x_R} \\ &= -\left[\sum_{S \in \mathcal{I}} \frac{\partial f_S}{\partial x_R}(x_S) \cdot \mathbb{1}(R \subseteq S) \right] + \left[\sum_{T \in \mathcal{J}} \frac{\partial g_T}{\partial x_R}(x_T) \cdot \mathbb{1}(R \subseteq T) \right] \quad (18) \end{aligned}$$

518 Recall that we assumed that $\mathcal{I}$ and $\mathcal{J}$ are downwards closed or ‘hierarchical’ which means all $S \in \mathcal{I}$
519 have all its subsets $S' \subseteq S$ also inside of $\mathcal{I}$. If we then take $R \notin \mathcal{I}$, it implies that $R \not\subseteq S$ for
520 all $S \in \mathcal{I}$, otherwise such an S would not obey the downwards closed property. This means that
521 $\mathbb{1}(R \subseteq S) = 0$ and

$$0 \equiv -0 + \left[\sum_{T \in \mathcal{J}} \frac{\partial g_T}{\partial x_R}(x_T) \cdot \mathbb{1}(R \subseteq T) \right]$$

522 This means that for all $T \in \mathcal{J} \cap \text{up}(R)$ where $\text{up}(R) := \{T : T \supseteq R\}$ it must be the case that the
523 R -th partial derivative is zero. We will focus on the case of $T = R$, but the same holds for all T as
524 above.

525 We may take a modification of the function g_T such that it is instead represented by additive
526 functions of a lower degree. This is equivalent to saying that $g_T \equiv 0$ and hence $T \in \mathcal{J}$ was
527 actually a contradiction. Let us see that $\frac{\partial g_R}{\partial x_R} \equiv 0$, so then writing $R = \{r_1, \dots, r_{|R|}\}$, we have
528 that $\int_{x_{r_1}} \frac{\partial g_R}{\partial x_R} = C_1(x_{r_2}, \dots, x_{r_{|R|}})$ for some function C_1 which is constant with respect to x_{r_1} .
529 Further $\int_{x_{r_2}} \frac{\partial g_R}{\partial x_{R-r_1}} = \int_{x_{r_2}} C_1(x_{r_2}, \dots, x_{r_{|R|}}) = C_1(x_{r_2}, \dots, x_{r_{|R|}}) + C_2(x_{r_1}, x_{r_3}, \dots, x_{r_{|R|}})$
530 and continuing on, we may ultimately see that $g_R = \sum_{R' \subsetneq R} g_{R'}$ which means by our assumption
531 that $\mathcal{J}$ is a minimal representation with no zero additive models, that actually $T \notin \mathcal{J}$.

532 The same arguments may be taken in reverse to show that for all $R \notin \mathcal{J}$, it must be the case $R \notin \mathcal{I}$.
533 There is a mild difference in the way the contradiction is applied argument when reversing the
534 arguments because we take the perspective that $\mathcal{I}$ is the ground truth generating process and $\mathcal{J}$ is a
535 potential alternate model (i.e. the contradiction is on R not on S .) Nonetheless, this results in the
536 fact $\mathcal{I} = \mathcal{J}$ for any two forward models which are representing the same distribution, and that the
537 functional representations are moreover the same. The latter part can be seen directly from $\frac{\partial f}{\partial x_i} = \frac{\partial g}{\partial x_i}$
538 for all $i \in [d]$ and similar arguments for removing trivial terms from the additive model. The final
539 modification which may exist is up to a constant, which is resolved by the differences in the mean of
540 the variable represented by ν and $\tilde{\nu}$. This is solved by the ANM assumption that the additive noises
541 are mean-centered. $\square$

542 Altogether, this is taken to mean that whenever the DAG is locally identifiable (and thus there are
 543 only valid forward models and no potential backwards model), then the hypergraph structure of the
 544 forward model is additionally identifiable. This additionally implies that if the entirety of the DAG is
 545 identifiable, then the entirety of the hyper DAG is also identifiable.

546 A.4 Proof of Theorem 4

547 *Proof.* Under the further assumption that $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$, we have that $\nu(\varepsilon) = \log p(\varepsilon) =$
 548 $\log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{\varepsilon^2}{2\sigma^2}\right)\right) = -\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\varepsilon^2$. Further, we have that $\nu'(\varepsilon) = -\frac{1}{\sigma^2}\varepsilon$,
 549 $\nu''(\varepsilon) = -\frac{1}{\sigma^2}$, and $\nu^{(k)}(\varepsilon) = 0$ for larger k . Accordingly, we may write the entire distribution
 550 as

$$\begin{aligned}\xi(x_1, \dots, x_d) &= \log p(x_1, \dots, x_d) = \sum_i \log p(x_i | x_{\text{Pa}(i)}) \\ &= \sum_i \nu_i(\varepsilon_i) = \sum_i \nu_i\left(x_i - \sum_{(S,i) \in \mathcal{H}} f_{S \rightarrow i}(x_S)\right)\end{aligned}$$

551 Let us write $f_{\rightarrow i}$ to denote $\sum_{(S,i) \in \mathcal{H}} f_{S \rightarrow i}$ and $F_i(x) = (x_i - f_{\rightarrow i}(x))$. It is straightforward to
 552 verify through repeated applications of the chain rule and product rule that

$$\frac{\partial^n}{\partial x_1, \dots, \partial x_n} \nu_i(F_i(x)) = \nu_i \cdot \frac{\partial^n}{\partial x_1, \dots, \partial x_n} (F_i(x)) + \nu'_i \cdot \sum_{\emptyset \subsetneq A \subsetneq [n]} \frac{\partial^{|A|}}{\partial x_A} (F_i(x)) \cdot \frac{\partial^{n-|A|}}{\partial x_{[n]-A}} (F_i(x))$$

553 It can further be seen that

$$\frac{\partial^{|A|}}{\partial x_A} (F_i(x)) = \frac{\partial^{|A|}}{\partial x_A} (x_i - f_{\rightarrow i}(x)) \quad (19)$$

554 is zero whenever A is not all i 's and A is not a subset of one of the S where $(S, i) \in \mathcal{H}$. This means
 555 exactly that

$$\sum_{\emptyset \subsetneq A \subsetneq [n]} \nu'_i \cdot \frac{\partial^{|A|}}{\partial x_A} (F_i(x)) \cdot \frac{\partial^{n-|A|}}{\partial x_{[n]-A}} (F_i(x))$$

556 is barely nonzero whenever we take $[n]$ equal to $(S + i)$ for some $(S, i) \in \mathcal{H}$. Note that this is
 557 equivalent to taking some $(S + i) \in \mathcal{H}'$ where we recall $\mathcal{H}'$ is the undirected version of the directed
 558 graph $\mathcal{H}$. If we instead take some $R \notin \mathcal{H}'$, then it will be the case that this derivative is zero, because
 559 all $A \subseteq R$ will have either $\frac{\partial^{|A|}}{\partial x_A} \equiv 0$ or $\frac{\partial^{n-|A|}}{\partial x_{[n]-A}} \equiv 0$. Moreover, it is the case that the first term is
 560 clearly zero $\frac{\partial^{|R|}}{\partial x_R} (F_i(x)) \equiv 0$.

561 Since this is true for all i so long as we are taking $R \notin \mathcal{H}'$, we have that

$$\frac{\partial^{|R|}}{\partial x_R} \xi(x_1, \dots, x_d) = \frac{\partial^{|R|}}{\partial x_R} \sum_i \nu_i \left(x_i - \sum_{(S,i) \in \mathcal{H}} f_{S \rightarrow i}(x_S) \right) \equiv 0$$

562 Following the same approach as in the proof of Theorem 3, this means that we are able to write
 563 $\xi(x_1, \dots, x_d) = \sum_{S \in \mathcal{H}'} \xi_S(x_S)$ for some functions ξ_S .

564 □

565 Note that the decomposition of the likelihood function's structure does not contradict existing results
 566 saying that the directionality of the graphical model is not always identifiable from data. In particular,
 567 in the linear-Gaussian case, it may not be possible to distinguish which direction is the causal direction.
 568 Nonetheless, the graphical structure which is recovered in this purely Gaussian case corresponds to
 569 what is available in the precision matrix [Loh and Bühlmann, 2014], still identifying the undirected
 570 graphical structure underlying the distribution. Under the CAM-like assumptions of linear SEMs,
 571 the hypergraph structure is reduced to its simplest representation, which is isomorphic to a graphical
 572 representation.

B Full Algorithm Details

B.1 Causal Additive Model

CAM (Causal Additive Model) uses a three step procedure to discover a set of additive structural equations according to Equation ?? . First, a preliminary search is made over the directed edges using sparse regression to cut down on the search space. Second, a greedy algorithm gradually adds the best edges to the DAG so long as it does not create any cycles. Third, the final DAG structure’s additive models are trained once again with sparse regression to encourage the removal of extraneous edges.

Step 1 First, a preliminary search is made over all possible edges via sparse regression. For each variable $j \in [d]$, one fits an additive model based on all of the other possible directed edges (k, j) using sparse regression. This allows for a smaller subset of the quadratic number of edges to be considered, especially in the high-dimensional setting when d is large.

The mean-squared error objective is minimized based on the assumption that the noise terms are Gaussian.

$$\log p(\varepsilon_j) = -\log(2\pi\sigma_j^2) - \frac{1}{2\sigma_j^2} \cdot \varepsilon_j^2 \quad (20)$$

$$\hat{\sigma}_j^2 := \|X_j - \hat{X}_j\|^2 = \|X_j - \sum_k \hat{f}_{k \rightarrow j}(X_k)\|^2 \quad (21)$$

Step 2 Second, the bulk of the algorithm centers around a greedy approach for gradually adding directed edges which do not disagree with the partial structure which has been built up so far. Every edge from the local neighborhood determined in step 1 is considered to be added, so long as it would not create a cycle in the DAG. Each edge is ranked by its ability to improve the log-likelihood of the overall model, by training an additive model with the selected edges.

$$\hat{\sigma}_j^2(\mathcal{N}_j) := \|X_j - \sum_{k \in \mathcal{N}_j} \hat{f}_{k \rightarrow j}(X_k)\|^2 \quad (22)$$

$$(k^*, j^*) = \underset{(k,j) \text{ acyclic}}{\operatorname{argmin}} \{ \hat{\sigma}_j^2(\mathcal{N}_j \cup \{k\}) - \hat{\sigma}_j^2(\mathcal{N}_j) \} \quad (23)$$

Importantly, for $j \neq j^*$, it is not necessary to retrain the additive models to recompute the values of $\hat{\sigma}_j^2(\mathcal{N}_j)$, because they are not affected by the inclusion of edges in other parts of the graph (except that it may block an edge from being added due to the acyclicity constraint).

Step 3 Lastly, the collection of directed edges which were selected in step 2 are used to train a final model end-to-end, with additional regularization designed to shrink unnecessary edges to become sparse. Additive terms $\hat{f}_{k \rightarrow j}$ which are deemed insignificant are removed from the model completely and the final set of edges define the final DAG.

B.2 Sparse Interaction Additive Network

SIAN (Sparse Interaction Additive Network) is an approach designed to train higher-order additive models using neural network techniques. This approach also consists of three main phases. In our case, we will follow CAM’s implicit Gaussian assumption by minimizing the mean-squared error objective which corresponds with the likelihood of independent Gaussian variables.

In the first phase, a typical neural network f_θ is trained to predict an output variable in terms of the input variables.

$$\hat{\sigma}^2(\theta) := \|Y - \hat{Y}(\theta)\|^2 = \|Y - \hat{f}_\theta(X_{[d]})\|^2 \quad (24)$$

607 In the second phase, interpretability techniques are combined with a special feature interaction
 608 selection (FIS) algorithm which ensures a sufficient coverage of the complex space of interactions
 609 while avoiding the exponential blow up in complexity from exploring all higher-order interactions.
 610 The final result of the first two phases is a collection $\mathcal{I} \subseteq \mathcal{P}([d])$ which is some collection of all
 611 of the feature interactions S which are important to predicting the output variable. Finally, the set
 612 of collected higher-order interactions are then used to train a neural-network-based additive model
 613 which obeys the interaction structure determined in the selection algorithm.

$$\hat{\sigma}_{\mathcal{I}}^2(\theta) := \|Y - \hat{Y}_{\mathcal{I}}(\theta)\|^2 = \|Y - \sum_S \hat{f}_{S, \theta_S}(X_S)\|^2 \quad (25)$$

614 This final additive neural network has pleasant properties like being more interpretable as well as
 615 more robust than the original neural network. In our context, we will use these neural additive models
 616 as the major component of modeling the hypergraphical additive structure we assumed previously.

617 B.3 Higher-order Causal Additive Model

618 In our algorithm, we broadly follow the same steps as the original CAM algorithm, replacing all com-
 619 ponents which are limited to one-dimensional additive models with their higher-order counterparts.

620 In the first step of our algorithm, we must reduce the number of candidate edges which will be
 621 considered in the downstream steps. Although CAM mentions this is only necessary in the high-
 622 dimensional setting for their additive assumption, ours is absolutely necessary except in extremely
 623 small cases (perhaps $d \leq 5$). This is because instead of searching over all candidate directed edges,
 624 $\{(k, j) : k \neq j, j \in [d]\}$, we must perform a search over the much larger space of all candidate
 625 directed hyperedges, $\{(S, j) : S \subseteq ([d] \setminus j), j \in [d]\}$.

626 For this purpose, we employ the first two phases of SIAN to each of the variables. That is, for each
 627 $j \in [d]$, we train a neural network to predict X_j from X_{-j} and then run a feature interaction selection
 628 algorithm to find a neighborhood of important interactions which are useful for predicting X_j .

$$\hat{\sigma}_j^2(\theta) := \|X_j - \hat{X}_j(\theta)\|^2 = \|X_j - \hat{f}_{j, \theta}(X_{[d]-k})\|^2 \quad (26)$$

629

$$\mathcal{I}_j = \text{FeatureInteractionSelection}(\hat{f}_{j, \theta}) \quad (27)$$

630 These selected interactions are then taken as the candidate set of directed hyperedges to be used in
 631 the later parts of the algorithm.

$$\tilde{\mathcal{H}} := \{(S, j) : S \in \mathcal{I}_j, j \in [d]\} \quad (28)$$

632 Note that these hyperedges are also given an importance score from the original FIS algorithm and
 633 may be sorted by their a priori importance.

634 In the second step of our algorithm, we follow the greedy approach of including hyperedges based on
 635 the improvement to the log-likelihood. This requires training many different additive models obeying
 636 the interaction constraints imposed by the current hyper DAG. We use the additive models from the
 637 third phase of SIAN to minimize an MSE objective as before.

638 In particular, we train multiple SIAN additive models to compare the different improvements in
 639 scores coming from adding all possible hyperedges $(S, j) \in \tilde{\mathcal{H}}$. This improvement in score is again
 640 interpreted as the improvement in reducing the noise of the added Gaussian via reduction in MSE.

$$\hat{\sigma}_j^2(\mathcal{N}'_j) := \|X_j - \sum_{S \in \mathcal{N}'_j} \hat{f}_{S \rightarrow j}(X_S)\|^2 \quad (29)$$

641

$$(S^*, j^*) = \underset{(S, j) \text{ acyclic}}{\operatorname{argmin}} \{ \hat{\sigma}_j^2(\mathcal{N}'_j \cup S) - \hat{\sigma}_j^2(\mathcal{N}'_j) \} \quad (30)$$

642 Because this requires training a large amount of additive models, we make multiple concessions to
 643 allow for a more rapid selection process during step 2 which can otherwise take a significant chunk
 644 of the overall algorithm time. Because of our higher-order neighborhood selection from step 1, it is at
 645 least feasible to search over higher-order interactions without facing an exponential blowup in the
 646 number of additive models which must be trained.

647 However, in practice we further reduce the number of additive models we train to a maximum of 10
 648 interactions per each variable X_j . As tuples are selected from the candidate superset $\tilde{\mathcal{H}}$ to be actually
 649 included into the model, additional candidate hyperedges are replenished to be explored in future
 650 iterations of the step 2 loop.

651 Furthermore, instead of training these SIAN additive models until there is no further reduction in
 652 MSE, we only train each for five epochs in total. We find that this gives a strong enough measurement
 653 of the performances of the different additive models without cutting significantly into the overall time
 654 taken. Moreover, because the heuristic coming from the first two phases of SIAN used in step 1 of
 655 our algorithm is generally quite good, an approximate measure of the reduction in score from each
 656 hyperedge in step 2 seems to generally be sufficient.

657 In the third step of our algorithm, we again follow the CAM setup and train an end-to-end SIAN model
 658 which obeys the structure which was greedily added in step two of the algorithm. L1 regularization
 659 terms are used on each of the shape functions in the additive model to encourage shrinkage in the
 660 unnecessary terms of the additive model. Similar to CAM, unimportant additive terms are thresholded
 661 away and removed from the final hyper DAG.

$$\mathcal{L}(\theta) := \sum_j \|X_j - \sum_{(S,j) \in \mathcal{H}} \hat{f}_{S \rightarrow j}\|^2 + \lambda_1 \sum_{(S,j) \in \mathcal{H}} |\hat{f}_{S \rightarrow j}| \quad (31)$$

662 For comparison against other DAG-based methods, the hyper DAG is further projected onto its
 663 equivalent DAG formulation.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [NA].
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1-2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No]" or "[NA]" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist",**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: The claims in the abstract and introduction accurately reflect the paper's contributions, in our opinion.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The limitations of the empirical evaluation and algorithm choices are discussed.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: Proofs are provided for all theorems and relevant references are additionally provided.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The synthetic DGP is simple and described in detail. The algorithm is described in the appendix to sufficient detail. Additionally, code for the DGP and algorithm will be provided.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: Experiment data and code will be released to enable reproducibility.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so No is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Experiment settings are simple and provided in full detail.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Metrics are provided as mean and standard deviation across three runs.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [No]

Justification: Exact algorithm time was not computed. CAM and HCAM take on the order of several hours.

Guidelines:

- The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The authors followed the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: No immediate societal impacts worth highlighting. Consists of mostly theoretical contributions.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Not relevant.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification: Provided alongside paper and code.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: Code provided.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

974 Answer: [NA]
 975 Justification: Not applicable.
 976 Guidelines:
 977 • The answer NA means that the paper does not involve crowdsourcing nor research with
 978 human subjects.
 979 • Including this information in the supplemental material is fine, but if the main contribu-
 980 tion of the paper involves human subjects, then as much detail as possible should be
 981 included in the main paper.
 982 • According to the NeurIPS Code of Ethics, workers involved in data collection, curation,
 983 or other labor should be paid at least the minimum wage in the country of the data
 984 collector.

985 **15. Institutional review board (IRB) approvals or equivalent for research with human**
 986 **subjects**
 987 Question: Does the paper describe potential risks incurred by study participants, whether
 988 such risks were disclosed to the subjects, and whether Institutional Review Board (IRB)
 989 approvals (or an equivalent approval/review based on the requirements of your country or
 990 institution) were obtained?
 991 Answer: [NA]
 992 Justification: Not applicable.
 993 Guidelines:
 994 • The answer NA means that the paper does not involve crowdsourcing nor research with
 995 human subjects.
 996 • Depending on the country in which research is conducted, IRB approval (or equivalent)
 997 may be required for any human subjects research. If you obtained IRB approval, you
 998 should clearly state this in the paper.
 999 • We recognize that the procedures for this may vary significantly between institutions
 1000 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
 1001 guidelines for their institution.
 1002 • For initial submissions, do not include any information that would break anonymity (if
 1003 applicable), such as the institution conducting the review.

1004 **16. Declaration of LLM usage**
 1005 Question: Does the paper describe the usage of LLMs if it is an important, original, or
 1006 non-standard component of the core methods in this research? Note that if the LLM is used
 1007 only for writing, editing, or formatting purposes and does not impact the core methodology,
 1008 scientific rigor, or originality of the research, declaration is not required.
 1009 Answer: [NA]
 1010 Justification: LLMs not used in non-standard ways.
 1011 Guidelines:
 1012 • The answer NA means that the core method development in this research does not
 1013 involve LLMs as any important, original, or non-standard components.
 1014 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>)
 1015 for what should or should not be described.