Decaf: A Deconfounding Causal Generative Model

Anonymous Authors[1]

Abstract

Causal generative models (CGMs) have recently emerged as capable approaches to simulate the causal mechanisms generating our observations, enabling causal inference. Unfortunately, existing approaches either are *overly restrictive*, assuming the absence of hidden confounders, or *lack generality*, being tailored to a particular query and graph. In this work, we introduce Decaf, a CGM that accounts for hidden confounders in a single amortized training process using only observational data and the causal graph. Importantly, Decaf can provably identify all causal queries with a valid adjustment set or sufficiently informative proxy variables. Remarkably, for the first time to our knowledge, we show that a confounded counterfactual query is identifiable, and thus solvable by Decaf, as long as its interventional counterpart is as well. Our empirical results on diverse settings—including the Ecoli70 dataset, with 3 independent hidden confounders, tens of observed variables and hundreds of causal queries—show that Decaf outperforms existing approaches, while demonstrating its out-of-the-box flexibility.

1 Introduction

Causal queries, or *what if* questions, seek to determine how changes in one variable affect another, which is crucial to evaluate the effects of interventions in fields such as healthcare (Feuerriegel et al., 2024), marketing policies (Varian, 2016) or education (Zhao & Heffernan, 2017). Importantly, when empirical trials are infeasible due to ethical, financial, or practical constraints, answering causal queries from observational data becomes essential.

To adress this challenge, causal generative models (CGMs) (Javaloy et al., 2023; Chao et al., 2023; Khemakhem et al., 2021) have recently emerged as powerful and flexible tools for modelling structural causal models (SCMs), allowing for efficiently sampling interventional and counterfactuals distributions, and enabling the estimation of any causal query of interest. However, all existing CGMs also assume *causal sufficiency*, i.e., that all confounders are observed.

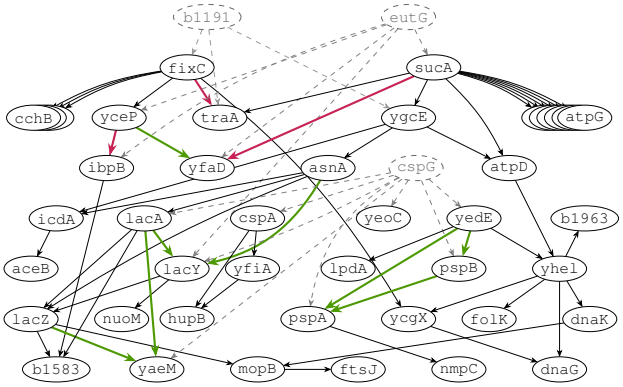


Figure 1: **Decaf can be effortlessly applied to highly complex causal graphs**, such as that of the Ecoli70 dataset (Schäfer & Strimmer, 2005), with multiple independent hidden confounders and dozens of variables. We dash hidden confounders, and highlight direct *confounded* effects that are now identifiable, or still unidentifiable, with Decaf.

However, *causal sufficiency* is rarely satisfied in practice, making *hidden confounding* a major challenge in causality, as it generally renders causal queries *unidentifiable*, i.e., that they cannot be uniquely expressed as a function of the observations. While recent advances have shown that some confounded causal queries are identifiable if there exist sufficiently informative proxies of the hidden confounders (Miao et al., 2018; 2023; Wang & Blei, 2021), these approaches are still limited to specific intervention-outcome pairs and do not allow for counterfactual estimation.

Our objective is to bridge the gap between these two lines of work. To this end, we introduce the *deconfounding causal normalizing flow* (Decaf ☕), to the best of our knowledge, the first CGM that allows the estimation of any *identifiable* causal query—including *counterfactuals*—in the presence of *hidden confounders*, with only observational data, the causal graph, and a single amortized training process. More in detail, Decaf resembles variational autoencoders (Kingma & Welling, 2014) as it is trained with an ELBO and comprises: **i)** a causal normalizing flow (CNF) (Javaloy et al.,

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2023) as "decoder", adapted to be conditioned on the (potentially many) hidden confounders; and **ii)** a conditional normalizing flow (Winkler et al., 2019) as "encoder", computing the posterior distribution of the hidden confounders.

Furthermore, we theoretically demonstrate that Decaf accurately estimates all identifiable causal queries (interventional and *counterfactual*) for which we can find a valid adjustment set or sufficiently informative proxy variables, significantly extending existing results from prior works (Miao et al., 2018; Wang & Blei, 2021; Javaloy et al., 2023).

All of the above is well illustrated in the Ecoli70 dataset (Schäfer & Strimmer, 2005), whose causal graph is depicted in Fig. 1. Specifically, by training Decaf *once* on this dataset, we can efficiently model all 43 observed variables and 3 independent hidden confounders and, most importantly, *compute any causal query on demand* during deployment. Out of all the direct causal effects (i.e., edges) in Fig. 1, Decaf can accurately estimate all unconfounded effects, as well as 8 out of the 11 confounded ones. In stark contrast with previous works, Decaf also estimates counterfactual queries, increasing the previous count to 16 identifiable queries.

In order to assist practitioners, we provide algorithms to easily check whether a particular query of interest is identifiable in our framework, and we will make our code publicly avaiable upon acceptance. Moreover, we empirically validate our claims on semi-synthetic and real-world experiments, demonstrating that Decaf outperforms existing alternatives while being widely applicable. Therefore, *Decaf offers a practical and efficient solution for causal inference in the presence of hidden confounding*, bridging the gap between general CGMs and specialized solutions.

2 Related Works

We discuss the most relevant works to put Decaf into context, and provide a more detailed literature review in App. D.

Generative causal models. In order to faithfully learn a SCM, one common approach consists modeling each variable as a function of its causal parents with an independent model, starting from the root nodes. As of the choice for modeling these functions, prior works range from simple yet well-established additive noise models (ANMs) (Hoyer et al., 2008), to more complex but powerful diffusion-based causal models (DCMs) (Chao et al., 2023), among others (Kocaoglu et al., 2018; Yang et al., 2020; Pawlowski et al., 2020; Parafita & Vitrià, 2022). Due to its nature, this approach typically is parameter-intensive, and can easily overfit and propagate errors to descendant variables.

Alternatively, recent works have explored using a single (structurally-constrained) network to model the SCM at once, e.g., using autoregressive flows (Khemakhem et al., 2021; Javaloy et al., 2023), or graph neural networks (GNNs) (Zečević et al., 2021; Sánchez-Martín et al., 2022). Among these, the causal normalizing flow (CNF) deserves special attention, given its flexibility and theoretical guarantees, which we discuss later in §3.2. Most importantly, all the approaches above assume *causal sufficiency*, i.e. the absence of hidden confounders, limiting their applicability in settings with hidden confounding.

Causal inference with latent confounders. Another line of work relies on structural assumptions for correctly answering causal queries. However, these approaches typically deal only with interventional queries (i.e., not counterfactual ones) and are tailored to a specific causal graph and a single treatment-outcome pair, requiring us to train one model per query. In particular, existing works exploit instrumental variables (IVs) (Angrist & Pischke, 2009) or mediators (Pearl, 2009) to achieve this goal and, more recently, a body of works exploit proxy variables to account for latent confounding (Allman et al., 2009; Kuroki & Pearl, 2014; Kallus et al., 2018; Louizos et al., 2017; Miao et al., 2023; 2018). Of particular interest is the Deconfounder by Wang & Blei (2021), a probabilistic model that interprets multiple treatments as null proxies to find a substitute of the hidden confounders and estimate causal queries.

3 Background

3.1 Confounded Structural Causal Models

Next, we introduce some ideas from the causality literature used throughout this work to model the causal structure of the data and answer causal queries of interest.

Definition 1. A *(confounded) Structural Causal Model (SCM)* is a triplet $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ describing a data-generating process over a set of D observed (endogenous) variables $\mathbf{x} \coloneqq (\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_D)$ as

$$\mathrm{x}_i \coloneqq f_i(\mathrm{pa}(i), \mathrm{u}_i, \mathbf{z}) \quad \text{for} \quad i = 1, 2, \dots, D\,, \qquad (1)$$
$$\text{with} \quad \mathbf{u} \coloneqq (\mathrm{u}_1, \mathrm{u}_2, \dots, \mathrm{u}_D) \sim P_{\mathbf{u}}\,,\ \mathbf{z} \sim P_{\mathbf{z}}\,,$$

and where f_i represents the structural equation to compute the i-th endogenous variable, x_i, from its observed *causal parents*, $\mathrm{pa}(i)$, the i-th exogenous variable, u_i, and the vector of *hidden confounders*, $\mathbf{z}$.[1]

Note that, while we make the dependence on the hidden confounders explicit for all observed variables in Eq. 1, we assume w.l.o.g. that a subset of them may not be directly affected by the hidden confounders. Furthermore, given a SCM $\mathcal{M}$, we denote by $\mathcal{G}$ the *faithful* causal graph that it induces, representing *only* the direct causal relationships between pairs of endogenous and hidden variables and, when necessary, also exogenous variables.

One key element in causality is the do operator (Pearl, 2012), denoted by $\mathrm{do}(\mathrm{t})$, which conceptualizes the action of ex-

[1] Bold denotes random vectors.

ternally intervening on a treatment variable t, i.e., to set t to a fixed value independently of its parents. In turn, the do operator enables the computation of interventional and counterfactual queries in SCMs (Peters et al., 2017), i.e., of population and instance-wise *what if* questions.

Definition 2. A *causal query* $Q(\mathcal{M}) \coloneqq p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c})$ is a distribution over $\mathrm{y} \in \mathbf{x}$ (the *outcome* variable), as a result of intervening upon the variable $\mathrm{t} \in \mathbf{x}$ (the *treatment* variable). Additionally, $Q(\mathcal{M})$ denotes an *interventional* or *counterfactual* query if the variable $\mathbf{c}$ is, respectively, the empty set or the vector of factual observed values, $\mathbf{x}^{\mathrm{f}}$.

However, in the presence of *hidden confounders*, one cannot simply apply the do-operator to evaluate causal queries, as the computations involve the causal parents and the unaccounted confounders would bias the results. Instead, one needs to find alternative ways to compute these quantities if possible, as we discuss in §2.

3.2 Causal Normalizing Flows

Causal normalizing flows (CNFs) (Javaloy et al., 2023) play an important role in this work, as they form the basic building blocks of Decaf, given their identifiability guarantees despite a mild set of assumptions.

Similar to Eq. 1, a CNF is defined as a pair $(T_{\boldsymbol{\theta}}, P_{\mathbf{u}})$ forming a data-generating process that yields a set of D endogenous variables as $\mathbf{x} \coloneqq T_{\boldsymbol{\theta}}^{-1}(\mathbf{u})$, where $\mathbf{u} \sim P_{\mathbf{u}}$ and $T_{\boldsymbol{\theta}} \colon \mathbb{R}^D \to \mathbb{R}^D$ is a normalizing flow (Papamakarios et al., 2021). In particular, $T_{\boldsymbol{\theta}}$ is a normalizing flow with additional structural constraints, ensuring that it induces the same causal graph as the underlying SCM.

Javaloy et al. (2023) demonstrated that CNFs form a general class of identifiable SCMs, and that they can approximate the underlying SCM as closely as required simply by maximizing the observed joint evidence, i.e., $\max_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$. Moreover, CNFs also allow for efficient sampling of any interventional and counterfactual distribution, enabling their use for complex causal-inference task.

Unfortunately, as discussed in §1, CNFs need to assume causal sufficiency to provide the above guarantees, thus limiting their applicability. In this work, we attempt to address this limitation and account for the presence of hidden confounders without losing theoretical guarantees.

4 Problem Statement

In this work, we assume the existence of an unobserved confounded SCM, $\mathcal{M}$, as in Def. 1, of which we have access to N i.i.d. observations and its induced causal graph, $\mathcal{G}$.

Our objective is therefore to design a CGM that can *faithfully answer as many causal queries from the original SCM as possible*, despite the presence of unobserved hidden confounders. In other words, to find a substitute model of $\mathcal{M}$ that we can use to accurately perform causal inference.

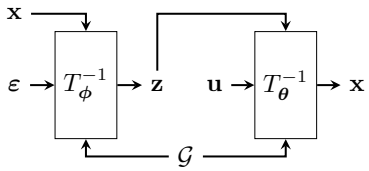


Figure 2: **Sketch of Decaf architecture**. $T_{\boldsymbol{\phi}}$ and $T_{\boldsymbol{\theta}}$ are conditional normalizing flows, with the top input as condition; $\mathcal{G}$ is the causal graph, and $\boldsymbol{\varepsilon}$ is a non-causal random variable needed by the normalizing flow to sample $\mathbf{z}$.

Assumptions. Regarding the underlying SCM $\mathcal{M}$, we simply assume that it **i)** has C^1-diffeomorphic structural equations,[2] and **ii)** induces an acyclic graph. We denote the family of SCMs meeting these assumptions by $\mathbb{M}$.

5 Deconfounding Causal Normalizing Flows

To help bridge the gap between CGMs and tailored hidden-confounding solutions, we now present the *deconfounding causal normalizing flow* (or Decaf ☕).

Intuitively, Decaf takes a well-grounded CGM such as the causal normalizing flow (Javaloy et al., 2023), which can provably approximate unconfounded SCMs and perform causal inference, and expand it such that it accounts for hidden confounding by building data-driven substitutes of these confounders, an idea that has been successfully explored in the past (Wang & Blei, 2019; 2021; Bica et al., 2020).

Decaf achieves the above by following a similar structure as that of a variational autoencoder (Kingma & Welling, 2014). That is, Decaf comprises two main components. First, an inference network which approximates the *intractable* posterior distribution of the hidden confounders, given their observed children. Second, a generative network that exploits structural constraints to accurately model the underlying SCM, given a substitute for hidden confounders. Each of these parts comes with their own challenges, however, which we now explain in detail:

Generative network. As mentioned in §3.2, we use CNFs (Javaloy et al., 2023) as our starting point. However, since our generative model needs to take in hidden confounders as conditional inputs, we adapt CNFs to use conditional normalizing flows (Winkler et al., 2019), instead of unconditional ones. The resulting model, $T_{\boldsymbol{\theta}}$, is thus an invertible transformation describing a data-generating process, conditioned on $\mathbf{z}$, which can map a set of exogenous variables $\mathbf{u}$ to our observations and vice versa, i.e.,

$$T_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = \mathbf{u} \sim P_{\mathbf{u}} \quad \text{and} \quad \mathbf{x} = T_{\boldsymbol{\theta}}^{-1}(\mathbf{u}, \mathbf{z}), \tag{2}$$

[2] That is, that $\mathbf{f}$ has inverse and both $\mathbf{f}$ and $\mathbf{f}^{-1}$ are continuously differentiable w.r.t. the exogenous variables.

where we further exploit the given causal graph to ensure that the generative process is faithful, i.e., such that

$$p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}) = \prod_{i=1}^{D} p_{\boldsymbol{\theta}}(\mathrm{x}_i \mid \mathrm{pa}(i), \mathbf{z}) \,, \tag{3}$$

defining now a process similar to that given in Def. 1. Just as in Def. 1, only the children of $\mathbf{z}$ will actually condition on $\mathbf{z}$ in Eq. 3. Furthermore, $T_{\boldsymbol{\theta}}$ allows us to write down the exact likelihood of the data given $\mathbf{z}$,

$$\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}) = p_{\mathbf{u}}(T_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}))|\det(\nabla_{\mathbf{x}} T_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}))| \,. \tag{4}$$

Deconfounding network. To model the posterior distribution of the hidden confounders given our observations, i.e., the abduction step needed to compute counterfactuals (Pearl, 2009), we use another conditional normalizing flow (Winkler et al., 2019), as it can approximate the true posterior distribution arbitrarily well. Once again, we exploit prior knowledge about the causal graph and mask the resulting network, $T_{\boldsymbol{\phi}}$, such that it models each *independent* hidden confounder $\mathbf{z}_k$ using only its observed children, i.e.,

$$q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) = \prod_{k=1}^{D_{\mathbf{z}}} q_{\boldsymbol{\phi}}(\mathbf{z}_k \mid \mathrm{ch}(\mathbf{z}_k)) \,, \tag{5}$$

where $D_{\mathbf{z}}$ is the number of independent hidden confounders.

Training process. We jointly train both networks defined above as it would be typically done in deep latent-variable models, i.e., during training we *maximize* the evidence lower bound (ELBO) (Kingma & Welling, 2014):

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})] - \mathrm{KL}[q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \| \, p(\mathbf{z})] \tag{6}$$

$$= \mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})] + \mathrm{H}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})) \,, \tag{7}$$

where $p(\mathbf{z})$ is the prior distribution of $\mathbf{z}$, KL the Kullback-Leibler divergence (Kullback & Leibler, 1951), and H the differential entropy (Kolmogorov, 1956).

The motivation for this choice is three-fold. First, we want the generative network to explain the observations given samples from $q_{\boldsymbol{\phi}}$ (first term of Eq. 6). Second, as we do not know the optimal size for $\mathbf{z}$, we need to prevent the deconfounding network from allocating information exclusive of $\mathbf{x}$ in $\mathbf{z}$ (entropy term in Eq. 7). Finally, all the theory in §6 relies on Decaf matching the data evidence, $p_{\text{data}}(\mathbf{x})$, which we encourage Decaf to do since

$$\max_{\boldsymbol{\phi}, \boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \min_{\boldsymbol{\phi}, \boldsymbol{\theta}} \mathrm{KL}[p_{\text{data}}(\mathbf{x}) \| \, p_{\boldsymbol{\theta}}(\mathbf{x})] + \mathrm{KL}[q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \| \, p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})] \,. \tag{8}$$

Causal inference. Since the tuple $(T_{\boldsymbol{\theta}}, P_{\mathbf{u}}, P_{\mathbf{z}})$ defines a confounded SCM as defined in Def. 1, we can use Decaf to efficiently sample from observational and interventional distributions by: **i)** sampling $\mathbf{z}$ from $p(\mathbf{z})$; and **ii)** sampling $\mathbf{x}$ from either $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$ or $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}, \mathrm{do}(\mathrm{t}))$, as proposed by Javaloy et al. (2023). For counterfactual inference, we can use the deconfounding network to perform the induction step, as the second KL term in Eq. 8 shows that it approximates the posterior induced by $T_{\boldsymbol{\theta}}$ (i.e., its $\mathbf{z}$-inverse given $\mathbf{x}$). Therefore, to generate counterfactual samples we simply need to: **i)** sample from $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \boldsymbol{x}^{\mathrm{f}})$; and **ii)** sample again from $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}, \mathrm{do}(\mathrm{t}))$. We provide more details about these steps and the do-operator in App. C.

6 Theoretical Results

We take advantage that our work is at intersection of CGMs and hidden-confounding solutions to leverage and expand the theory of both research fields. While we present here an intuitive summary of our main theoretical results, formal statements and derivations can be found in App. A.

Note that, throughout this section, we assume that Decaf matches the true data evidence, i.e., $p_{\text{data}}(\mathbf{x}) = p_{\boldsymbol{\theta}}(\mathbf{x})$. Given that CNFs (and hence Decaf) are universal density approximators (Papamakarios et al., 2021), we should be able to always meet this assumption, provided enough resources.

6.1 Causal Query Identifiability

First, we study which queries are identifiable with Decaf. We call a query identifiable if we are guaranteed to produce the same query distribution as the original SCM by matching the data evidence. More formally, we adopt the following definition (Pearl, 2009, Def. 3.2.4):

Definition 3. Let $Q(\mathcal{M})$ be a causal query of a model $\mathcal{M}$. We call Q *identifiable* if, for any two models $\mathcal{M}_1, \mathcal{M}_2 \in \mathbb{M}$, $Q(\mathcal{M}_1) = Q(\mathcal{M}_2)$ whenever $p_{\mathcal{M}_1}(\mathbf{x}) = p_{\mathcal{M}_2}(\mathbf{x}) > 0$.

Another relevant concept for this section is that of a *valid adjustment set* (Peters et al., 2017, Def. 6.38). In plain terms, if we were to compute a causal query, say $p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$, a valid adjustment set $\mathbf{b}$ is a subset of variables such that: **i)** it blocks all backdoor paths between y and t, and **ii)** it is independent of the variable t after severing all incoming edges in t in the associated causal graph. As a consequence, we can use $\mathbf{b}$ to apply the adjustment formula,

$$p(\mathrm{y} \mid \mathrm{do}(\mathrm{t})) = \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}) \; p(\mathbf{b}) \; \mathrm{d}\mathbf{b} \,. \tag{9}$$

Additionally, we refer to $\mathbf{b}$ as *invalid* if only **i)** holds.

6.1.1 Interventional Queries

We first look at the identifiability of interventional queries, i.e., queries of the form $Q(\mathcal{M}) = p_{\mathcal{M}}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$, where $\mathrm{y}, \mathrm{t} \in \mathbf{x}$ are any two endogenous variables. We summarize our findings in the following proposition, which we properly formalize in App. A.2:

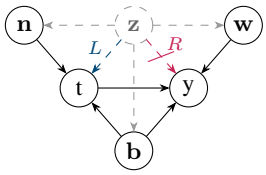


Figure 3: **Generic causal graph** where we are interested in the interventional query $Q(\mathcal{M}) = p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$. Blue and red edges play a crucial role, as their presence or absence induce different types of identifiability conditions.

Proposition 6.1 (Informal)**.** *Decaf is able to identify a given interventional causal query if one of the following exists:*

i) a valid adjustment set $\mathbf{b}$ *not containing* $\mathbf{z}$*,*

ii) an invalid one where $p(\mathbf{b} \mid do(\mathrm{t}))$ *is identifiable, or*

iii) sufficiently informative proxy and null proxy variables.

To help us go through the requirements in Prop. 6.1, let us break them down with the example depicted in Fig. 3 where, depending on the presence or absence of edges L and R, we face qualitatively different identifiability scenarios:

1. Unconfounded case, LR. If neither treatment nor outcome are directly influenced by $\mathbf{z}$, then we can always find a valid adjustment set that does not include $\mathbf{z}$. We extend the results of Javaloy et al. (2023) to show that Decaf can identify *any* interventional causal query of this type.

2. Confounded-treatment case, LR. If only the treatment is directly affected by $\mathbf{z}$, we run into two possible scenarios. First, if we are able to find a valid adjustment set e.g., $\mathbf{b}$ and $\mathbf{w}$ in Fig. 3, then Decaf can always identify the interventional query. Otherwise, Decaf could still identify the query if we find an *invalid* adjustment set where $p(\mathbf{b} \mid \mathrm{do}(\mathrm{t}))$ is still identifiable by Decaf.

3. Confounded-outcome case, LR. When only the outcome variable directly depends on $\mathbf{z}$, Decaf can identify any interventional query, as it necessarily exists a valid adjustment set not containing $\mathbf{z}$. In our running example, variables $\mathbf{n}$ and $\mathbf{b}$ would block all backdoor paths in Fig. 3, and Decaf would properly estimate the interventional query.

4. Fully-confounded case, LR. When both variables directly depend on $\mathbf{z}$, identifiability is more challenging, as any adjustment set necessarily involves the hidden confounder. In this case, we extend in Prop. A.2 the results from Miao et al. (2018) and Wang & Blei (2021) to allow for *general causal graphs and additional covariates*. In short, we find that an interventional query is identifiable if we can find: **i)** a proxy $\mathbf{w}$, independent of t, to distinguish $\mathbf{z}$ from the exogenous variables $\mathbf{u}$; and **ii)** a null proxy $\mathbf{n}$, independent of y given t and $\mathbf{z}$, to discern the correct structural equation. Additionally, as in prior works (Miao et al., 2018; Wang & Blei, 2021), $\mathbf{z}$ should be *complete* given the proxies (refer to Def. 5 for a formal definition). That is, both proxies should be sufficiently informative to accurately approximate $\mathbf{z}$.

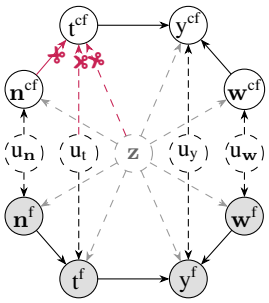


Figure 4: **Twin counterfactual network**, with observed nodes in gray. By duplicating the structural equations, we prove query identifiability in the counterfactual world while conditioning on the factual one.

6.1.2 Counterfactual Queries

We focus next on the identifiability of counterfactual queries, i.e., queries of the form $Q(\mathcal{M}) = p_{\mathcal{M}}(\mathrm{y}^{\mathrm{cf}} \mid \mathrm{do}(\mathrm{t}^{\mathrm{cf}}), \mathbf{x}^{\mathrm{f}})$, where $\mathbf{x}^{\mathrm{f}}$ is the observed factual, and where we are interested in *the distribution the outcome would have had, had we intervened on the treatment variable*. We demonstrate, for the first time to our knowledge, that counterfactual query identifiability holds for as many queries as for the interventional case. More specifically, we show that:

Proposition 6.2 (Informal)**.** *When an interventional query* $p(\mathrm{y} \mid do(\mathrm{t}))$ *is identifiable by Decaf, then it equally identifies the counterfactual query* $p(\mathrm{y}^{\mathrm{cf}} \mid do(\mathrm{t}^{\mathrm{cf}}), \mathbf{x}^{\mathrm{f}})$.

The formal result can be found in Prop. A.7. In short, our result means that, if we can identify an interventional query, then we can identify its counterfactual counterpart as well.

Our result exploits the notion of twin SCM (Balke & Pearl, 1994), which duplicates the structural equations for the factual and counterfactual worlds while sharing the exogenous variables, and the fact that Prop. A.2 allows for queries with additional covariates as long as they do not form colliders, which is always the case with $\mathbf{x}^{\mathrm{f}}$ in $p_{\mathcal{M}}(\mathrm{y}^{\mathrm{cf}} \mid \mathrm{do}(\mathrm{t}^{\mathrm{cf}}), \mathbf{x}^{\mathrm{f}})$, as we show in the example twin network from Fig. 4.

6.2 Identifying Exogenous Distributions

Besides causal query identifiability, another question of interest is whether Decaf recovers the true exogenous variables, up to component-wise transformations, disentangling the sources of variability of each endogenous variable. In App. A.1, we expand the results of Javaloy et al. (2023) to prove that Decaf identifies[3] the underlying SCM for those variables not directly affected by $\mathbf{z}$, i.e.:

Corollary 6.3 (Informal)**.** *Decaf identifies the underlying SCM, restricted to every variable other than* $\mathrm{ch}(\mathbf{z})$*, up to an*

[3] In the sense of Xi & Bloem-Reddy (2023).

element-wise transformation of the exogenous distribution.

Moreover, we conjecture that Decaf should in most cases properly disentangle the rest of exogenous variables and $\mathbf{z}$, Although we do not formally prove it, we refer to the use case of §7.3 to illustrate that the exogenous variables and the latent variables extracted by Decaf. Our intuition is that, if some children of $\mathbf{z}$ are conditionally independent, the information common to them can only be explained via $\mathbf{z}$. In addition, the entropy term in Eq. 7 discourages Decaf from using the components of $\mathbf{z}$ that are not necessary for explaining the observations. Recent works proved similar results under slightly stronger assumptions (von Kügelgen et al., 2021; Zheng et al., 2022; Brady et al., 2023).

6.3 Practical Guidelines & Implications

In this section, we outline the different aspects to consider for the successful application of Decaf to solve causal queries in real-world scenarios.

Training. One key advantage of Decaf is that it needs to train only once per dataset. However, maximizing the ELBO makes it also susceptible to posterior collapse (Wang et al., 2021), i.e., to the KL term in Eq. 6 vanishing, and hence the posterior equating the prior distribution. Fortunately, we can leverage existing solutions, e.g., implement regularization terms as the one proposed by Vahdat & Kautz (2020). Recall also that, following §6, model selection should use an observational goodness-of-fit metric as selection criterion.

Solving causal queries. Whilst Decaf can compute *any* causal query, unidentifiable causal queries may still lead to incorrect estimates. To ensure reliability, we must verify the identifiability of each specific query of interest, for which we provide algorithms that check identifiability in the causal graph in App. E. Namely, Alg. 5 checks if a query that involves a specific treatment-outcome pair, which includes average treatment effects and counterfactuals, is identifiable. If we were interested in a query on all variables, e.g., as samples from an interventional distribution, we should evaluate the identifiability of the causal effects between the treatment and all its descendants, as proposed in Alg. 6.

Limitations. Decaf relaxes the assumption of *causal sufficiency*, but it still relies on completeness for the proxies, a common condition for nonparametric identification in causal inference (D'Haultfoeuille, 2011; Chen et al., 2014). This condition is untestable with observational data alone, though collecting additional proxies can help satisfy completeness (Andrews, 2011). Moreover, we assume that the true SCM is C^1-*diffeomorphic* with respect to the exogenous variables, which precludes theoretical guarantees for modeling discrete variables, although Javaloy et al. (2023); de Vassimon Manela et al. (2024) show that, in practice, CNFs effectively approximate discrete distributions.

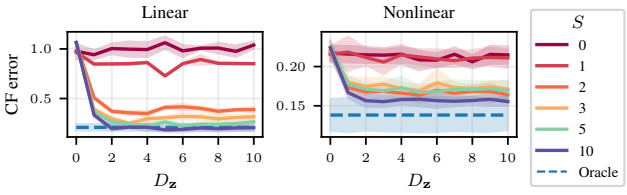


Figure 5: **Ablation**. Counterfactual error as we change the number of proxy variables and the latent dimensionality. We show means and 95 % confidence intervals over 5 realizations, intervening on the 25th, 50th, and 75th percentile of t.

7 Empirical Evaluation

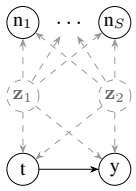
In this section, we assess the performance of Decaf comparatively to existing methods. Namely, we show that Decaf accurately estimate interventional and counterfactual queries when the requirements of Prop. 6.1 are met, and that it effectively estimates the exogenous information. We provide all experimental details in App. B.

Common evaluation. For all experiments, we estimate the performance on the interventional and counterfactual regimes via the mean absolute error (MAE) of, respectively, the average treatment effect (ATE) and the counterfactual samples, with respect to the ground-truth values. Moreover, we use as reference a CNF that *does observe* the hidden confounders, which we refer to as *oracle*. We also account for differences across observed variables by computing all errors over the standardized variables.

7.1 Ablation study

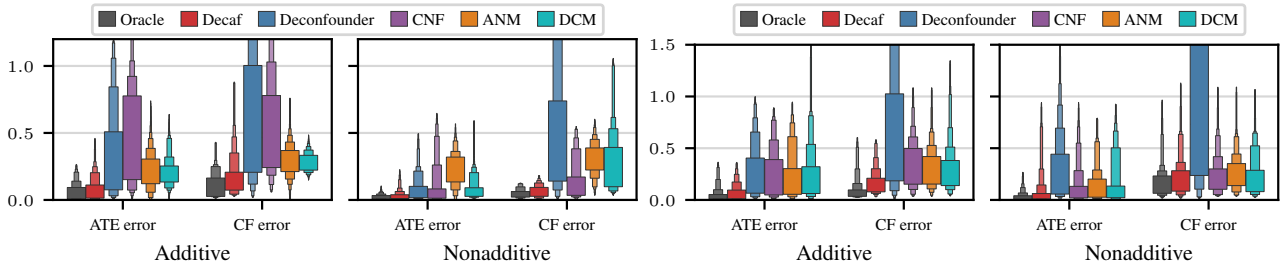
First, we conduct a simple ablation to understand how misspecifying the dimensionality of $\mathbf{z}$ may affect Decaf, as well as its sensitivity to the number of available proxies. For additional details and results, refer to App. B.1.

Experimental setup. We consider two synthetic SCMs, linear and non-linear, that follow the causal graph depicted in the inset figure, comprising two independent hidden confounders affecting every variable, and S null proxies. Then, we evaluate how well Decaf estimates the direct effect of t on y while changing the number of proxy variables, S, and the specified latent dimensionality, $D_{\mathbf{z}}$.



Results. Fig. 5 shows the counterfactual error for all cases, where we clearly observe that increasing the number of proxies reduces them, and with a drastic change as we add the second proxy, corroborating Prop. 6.1.

Similarly, we observe that underestimating $D_{\mathbf{z}}$ increases the error (especially if we assume causal sufficiency, $D_{\mathbf{z}} = 0$) while overestimating it does not. This indicates that, indeed, the entropy term in Eq. 6 prevents non-shared information from being modeled through $\mathbf{z}$, as discussed in §5.



(a) **Sachs' dataset**. (b) **Ecoli70 dataset**.

Figure 6: Error boxenplots for different CGMs, averaged over all identifiable direct effects of the Sachs (Fig. 7) **(a)** and Ecoli70 (Fig. 1) **(b)** datasets, after intervening in their 25th, 50th, and 75th percentiles in 5 random initializations.

7.2 Semi-synthetic Experiments

Next, we evaluate how Decaf performs relatively to existing approaches and, to this end, we consider semi-synthetic datasets for which we have access to the ground-truth SCMs. Additional details can be found in Apps. B.2 and B.3.

Baselines. We compare Decaf with three CGMs which assume causal sufficiency and are thus *unaware* of the hidden confounders: CNFs (Javaloy et al., 2023); ANMs (Hoyer et al., 2008); and DCMs (Chao et al., 2023); and with the Deconfounder (Wang & Blei, 2019), which uses proxies to provide unbiased ATE estimates under hidden confounding, yet it requires a model per treatment-outcome pair. We use the oracle as reference model to lower bound the error.

7.2.1 Protein-Signalling Networks

We first conduct a similar semi-synthetic experiment as that of Chao et al. (2023), based on a protein-signalling network dataset (Sachs et al., 2005). Specifically, we randomly generate a non-linear SCM that induces the same causal graph as the original dataset, depicted in Fig. 7, except for the root nodes, for which we use the original data. As a result, we have a hidden confounder with two dimensions, `PKC` and `PKA`, and three treatment variables to intervene upon, `Raf`, `Mek`, and `Erk`. We consider additive and non-additive structural equations, measure the effect of interventions on the downstream nodes and, more importantly, ensure that the randomized effect of the hidden confounder is perceptible.

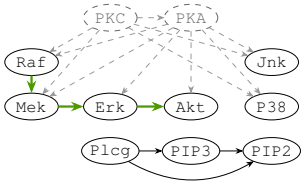


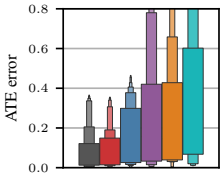
Figure 7: **Sachs' causal graph**. Green denotes identifiable confounded effects.

Results. We present a summary of the results in Fig. 6a, where we can observe that Decaf outperforms every approach in all cases, for both ATE and counterfactual errors, remaining fairly close to the oracle model. Moreover, we appreciate a great difference in performance between Decaf and CNFs, which corroborates the importance of the proposed encoder and variational training employed by Decaf, since a CNF is equivalent to Decaf with $D_{\mathbf{z}} = 0$.

7.2.2 Gene Networks

Next, we repeat a similar experiment as in the previous section, considering this time the causal graph of the Ecoli70 dataset (Schäfer & Strimmer, 2005) as reference, shown in Fig. 1, representing a gene network from E. coli data. This time, we replace root nodes with Gaussian variables.

Results. Similar to the previous case, the results presented in Fig. 6b demonstrate that Decaf is indeed able to closely match the performance of the oracle model, outperforming existing approaches. However, the non-additive case also shows significant long-tailed error distributions for all models, showing that Decaf can suffer the same problems as any data-centric approach, and that it is still needed to put attention on its effective training.



It is also worth-pointing out that the striking performance of the Deconfounder is a result of evaluating causal queries that cannot be identified by the model. As we discuss in App. B, the Deconfounder offers guarantees regarding ATE estimation and with more restrictive assumptions. If we plot instead the ATE error evaluated on only those paths that meet the assumptions placed by the Deconfounder, as shown in the inset figure, we see that it now achieves significantly lower errors that the *unaware* approaches.

Remarkably, this experiment highlights every strength of the proposed approach, since Decaf: **i)** models several hidden confounders affecting different sets of variables; **ii)** identifies all causal queries for which we have some proxy information; and **iii)** achieves the above in an agnostic manner, i.e., training out-of-the-box and *one single time*, despite the graph having 43 observed variables.

7.3 Fairness Real-world Use Case

Taking inspiration from the experiments by Kusner et al. (2017) and Javaloy et al. (2023), we aim to show how model-

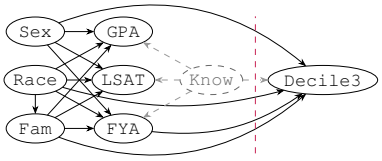


Figure 8: Causal graph assumed for the law school dataset. `Decile3` is only used by the classifiers.

Table 1: Test RMSE in `Decile3` prediction and MMD comparing different-group prediction distributions.

	Unfair	Unaware	Decaf	Fair K	Fair Add	Mean
RMSE	1.477	1.479	1.652	2.818	2.817	2.83
MMD	0.110	0.102	0.0018	10^{-6}	10^{-8}	0

ling confounded SCMs with Decaf can be leveraged beyond causal query estimation and, in particular, for counterfactual fairness prediction. See App. B.4 for further details.

Dataset and objective. Our aim is to train a predictor, using the law school dataset (Wightman, 1998) which comprises information of 21 790 law students, that remains accurate while being fair—using *demographic parity* as fairness criterion (Feldman et al., 2015)—toward the sensitive attributes of the students. In particular, we are interested in predicting the decile of a student in its 3rd year of university, given their undergraduate and 1st year grades, family income, race, and sex.

Experimental setup. First, we train Decaf assuming a causal graph such as the one in Fig. 8, excluding `Decile3`, where all grades are affected by a common "knowledge" hidden confounder. Then, we train a simple predictor using as input the hidden confounder and non-sensitive exogenous variables estimated by Decaf. If, as discussed in §6.2, Decaf successfully recovers the exogenous variables, we expect the predictor to be fair yet slightly less accurate, since `Decile3` is directly affected by the sensitive attributes.

Results. Tab 1 shows the prediction error (RMSE) and the difference between groups (MMD) for the proposed predictor using Decaf, comparing with an *unfair* predictor that uses sensitive attributes; an *unaware* predictor that excludes sensitive attributes, and two fair predictors—*Fair K* and *Fair Add*—proposed by Kusner et al. (2017).

As shown in Fig. 9, Decaf provides a much fairer predictor than the *unfair* and the *unaware* predictors at the cost of slightly higher RMSE. We can also appreciate that the other two fair approaches are so by predicting a constant value for every individual, which can be also observed comparing the RMSE obtained by these predictors with a naive predictor that predicts the mean of the distribution in Tab 1.

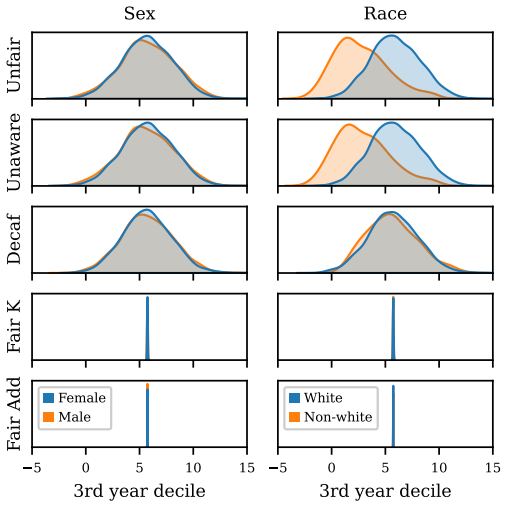


Figure 9: **Distribution of predicted `Decile3`.** Fairer predictors yield similar distributions across the two considered groups on each attribute (`Sex` and `Race`).

8 Concluding Remarks

In this work, we have bridged the current gap between CGMs, which fail to account for hidden confounders, and hidden-confounding solutions, which are tailored to a specific causal query and thus need to train once per query. To this end, we have introduced Decaf, and theoretically shown that it can accurately estimate causal queries in the presence of hidden confounders, if there exists a valid adjustment set or sufficiently informative proxies, extending prior results (Miao et al., 2018) to also consider counterfactuals. We have empirically shown that Decaf outperforms all considered baselines, better estimating confounded causal queries shown to be identifiable, and properly identifying exogenous distributions to train fair classifiers. Finally, we have provided algorithms to check the identifiability of causal queries which, along Decaf, provides practitioners with a powerful pipeline to perform causal inference in the presence of hidden confounders.

Future work. Our work opens many intriguing venues, e.g., integrating alternative identification strategies, such as instrumental variables (Hartford et al., 2017), to expand the range of identifiable queries that Decaf can estimate. We also find it interesting to apply Decaf to settings with time-varying treatments, where multiple interventions have to be performed. In real-world scenarios, it would be exciting to include interventional data during training, and seeing Decaf applied to real-world problems such as decision support systems (Sanchez et al., 2022), educational analysis (Murnane, 2010), or policy making (Fougère & Jacquemet, 2021), yet always validating them with interventional data.

Impact statement

This research contributes to advance causal inference in machine learning, particularly enhancing the ability to estimate causal effects despite unobserved variables. Thus, this work supports more informed decision-making in scenarios where controlled experimentation is impractical or unethical, such as healthcare or education. As with all advances in causal inference, practitioners should be aware of the limitations and assumptions of causal models. Particularly, in sensitive applications, where decisions are based on accurate causal conclusions, validation with interventional data should be prioritized whenever possible to ensure reliability. Overall, this work aligns with the broader goal of improving machine learning and does not introduce significant ethical risk beyond those traditionally associated with the field.

Bibliography

Allman, E. S., Matias, C., and Rhodes, J. A. Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A):3099–3132, 2009. ISSN 00905364, 21688966. URL http://www.jstor.org/stable/25662188. (Cited in pages 2, 29, and 30.)

Andrews, D. W. Examples of L2-complete and boundedly-complete distributions. 2011. (Cited in pages 6 and 15.)

Angrist, J. D. and Pischke, J.-S. *Mostly harmless econometrics: An empiricist's companion*. Princeton university press, 2009. (Cited in pages 2 and 29.)

Balke, A. and Pearl, J. Probabilistic Evaluation of Counterfactual Queries. *Probabilistic and Causal Inference*, 1994. URL https://api.semanticscholar.org/CorpusID:18845266. (Cited in page 5.)

Bica, I., Alaa, A. M., and van der Schaar, M. Time Series Deconfounder: Estimating Treatment Effects over Time in the presence of hidden confounders. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 884–895. PMLR, 2020. URL http://proceedings.mlr.press/v119/bica20a.html. (Cited in page 3.)

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *Journal of machine learning research*, 20(28):1–6, 2019. (Cited in page 27.)

Brady, J., Zimmermann, R. S., Sharma, Y., Schölkopf, B., von Kügelgen, J., and Brendel, W. Provably Learning Object-Centric representations. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 3038–3062. PMLR, 2023. URL https://proceedings.mlr.press/v202/brady23a.html. (Cited in page 6.)

Carrasco, M., Florens, J.-P., and Renault, E. Linear inverse problems in structural econometrics estimation based on spectral decomposition and regularization. *Handbook of econometrics*, 6:5633–5751, 2007. (Cited in page 16.)

Chao, P., Blöbaum, P., and Kasiviswanathan, S. P. Interventional and counterfactual inference with diffusion models. *ArXiv preprint*, abs/2302.00860, 2023. URL https://arxiv.org/abs/2302.00860. (Cited in pages 1, 2, 7, and 22.)

Chen, X., Chernozhukov, V., Lee, S., and Newey, W. K. Local identification of nonparametric and semiparametric models. *Econometrica*, 82(2):785–809, 2014. (Cited in page 6.)

de Vassimon Manela, D., Battaglia, L., and Evans, R. J. Marginal Causal Flows for Validation and Inference. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. (Cited in page 6.)

D'Haultfoeuille, X. On the completeness condition in nonparametric instrumental problems. *Econometric Theory*, 27(3):460–471, 2011. (Cited in page 6.)

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. Certifying and Removing Disparate Impact. In Cao, L., Zhang, C., Joachims, T., Webb, G. I., Margineantu, D. D., and Williams, G. (eds.), *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, pp. 259–268. ACM, 2015. doi: 10.1145/2783258.2783311. URL https://doi.org/10.1145/2783258.2783311. (Cited in page 8.)

Feuerriegel, S., Frauen, D., Melnychuk, V., Schweisthal, J., Hess, K., Curth, A., Bauer, S., Kilbertus, N., Kohane, I. S., and van der Schaar, M. Causal machine learning for predicting treatment outcomes. *Nature Medicine*, 30(4): 958–968, 2024. (Cited in page 1.)

Fougère, D. and Jacquemet, N. Policy evaluation using causal inference methods. In *Handbook of Research Methods and Applications in Empirical Microeconomics*, pp. 294–324. Edward Elgar Publishing, 2021. (Cited in page 8.)

Hartford, J. S., Lewis, G., Leyton-Brown, K., and Taddy, M. Deep IV: A Flexible Approach for Counterfactual prediction. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of*

the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, volume 70 of *Proceedings of Machine Learning Research*, pp. 1414–1423. PMLR, 2017. URL http://proceedings.mlr.press/v70/hartford17a.html. (Cited in pages 8 and 29.)

Hoyer, P. O., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pp. 689–696. Curran Associates, Inc., 2008. URL https://proceedings.neurips.cc/paper/2008/hash/f7664060cc52bc6f3d620bcedc94a4b6-Abstract.html. (Cited in pages 2 and 7.)

Javaloy, A., Sánchez-Martín, P., and Valera, I. Causal normalizing flows: from theory to practice. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/b8402301e7f06bdc97a31bfaa653dc32-Abstract-Conference.html. (Cited in pages 1, 2, 3, 4, 5, 6, 7, 15, 18, 19, 25, 27, and 28.)

Kallus, N., Mao, X., and Udell, M. Causal Inference with Noisy and Missing Covariates via Matrix factorization. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6921–6932, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/86a1793f65aeef4aeef4b479fc9b2bca-Abstract.html. (Cited in pages 2, 29, and 30.)

Kallus, N., Mao, X., and Zhou, A. Interval Estimation of Individual-Level Causal Effects Under Unobserved confounding. In Chaudhuri, K. and Sugiyama, M. (eds.), *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2281–2290. PMLR, 2019. URL http://proceedings.mlr.press/v89/kallus19a.html. (Cited in pages 29 and 30.)

Kaltenpoth, D. and Vreeken, J. Nonlinear Causal Discovery with Latent confounders. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 15639–15654. PMLR, 2023. URL https://proceedings.mlr.press/v202/kaltenpoth23a.html. (Cited in page 22.)

Khemakhem, I., Monti, R. P., Leech, R., and Hyvärinen, A. Causal Autoregressive flows. In Banerjee, A. and Fukumizu, K. (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3520–3528. PMLR, 2021. URL http://proceedings.mlr.press/v130/khemakhem21a.html. (Cited in pages 1 and 2.)

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114. (Cited in pages 1, 3, and 4.)

Kocaoglu, M., Snyder, C., Dimakis, A. G., and Vishwanath, S. Causalgan: Learning Causal Implicit Generative Models with Adversarial training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=BJE-4xW0W. (Cited in page 2.)

Kolmogorov, A. On the Shannon theory of information transmission in the case of continuous signals. *IRE Transactions on Information Theory*, 2(4):102–108, 1956. (Cited in page 4.)

Kullback, S. and Leibler, R. A. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86, 1951. (Cited in page 4.)

Kuroki, M. and Pearl, J. Measurement bias and effect restoration in causal inference. *Biometrika*, 101(2):423–437, 2014. (Cited in pages 2, 29, and 30.)

Kusner, M. J., Loftus, J. R., Russell, C., and Silva, R. Counterfactual fairness. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4066–4076, 2017. URL https://proceedings.neurips.cc/paper

/2017/hash/a486cd07e4ac3d270571622f4f316ec5-Abstract.html. (Cited in pages 7, 8, 25, 26, and 27.)

Long, C. P. and Antoniewicz, M. R. Metabolic flux analysis of Escherichia coli knockouts: lessons from the Keio collection and future outlook. *Current opinion in biotechnology*, 28:127–133, 2014. (Cited in page 23.)

Louizos, C., Shalit, U., Mooij, J. M., Sontag, D. A., Zemel, R. S., and Welling, M. Causal Effect Inference with Deep Latent-Variable models. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6446–6456, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/94b5bde6de888ddf9cde6748ad2523d1-Abstract.html. (Cited in pages 2, 29, and 30.)

Luo, R. and Zhao, H. Bayesian hierarchical modeling for signaling pathway inference from single cell interventional data. *The annals of applied statistics*, 5:725–745, 2011. doi: 10.1214/10-AOAS425. (Cited in page 22.)

Miao, W., Geng, Z., and Tchetgen Tchetgen, E. J. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105(4):987–993, 2018. (Cited in pages 1, 2, 5, 8, 16, 17, 18, 29, and 30.)

Miao, W., Hu, W., Ogburn, E. L., and Zhou, X.-H. Identifying effects of multiple treatments in the presence of unmeasured confounding. *Journal of the American Statistical Association*, 118(543):1953–1967, 2023. (Cited in pages 1, 2, 15, 29, and 30.)

Murnane, R. *Methods matter: Improving causal inference in educational and social science research*. Oxford University Press, 2010. (Cited in page 8.)

Nasr-Esfahany, A., Alizadeh, M., and Shah, D. Counterfactual Identifiability of Bijective Causal models. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25733–25754. PMLR, 2023. URL https://proceedings.mlr.press/v202/nasr-esfahany23a.html. (Cited in page 15.)

Papamakarios, G., Nalisnick, E. T., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing Flows for Probabilistic Modeling and inference. *J. Mach. Learn. Res.*, 22:57:1–57:64, 2021. URL http://jmlr.org/papers/v22/19-1028.html. (Cited in pages 3 and 4.)

Parafita, Á. and Vitrià, J. Estimand-agnostic causal query estimation with deep causal graphs. *IEEE Access*, 10: 71370–71386, 2022. (Cited in page 2.)

Pawlowski, N., de Castro, D. C., and Glocker, B. Deep Structural Causal Models for Tractable Counterfactual inference. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/0987b8b338d6c90bbedd8631bc499221-Abstract.html. (Cited in page 2.)

Pearl, J. *Causality*. Cambridge university press, 2009. (Cited in pages 2, 4, 20, and 29.)

Pearl, J. The Do-Calculus Revisited. In de Freitas, N. and Murphy, K. P. (eds.), *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, August 14-18, 2012*, pp. 3–11. AUAI Press, 2012. URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=2330&proceeding_id=28. (Cited in page 2.)

Pearl, J., Glymour, M., and Jewell, N. P. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016. (Cited in page 28.)

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017. (Cited in pages 3, 4, 16, 17, 19, and 20.)

Ranganath, R. and Perotte, A. Multiple causal inference with latent confounding. *ArXiv preprint*, abs/1805.08273, 2018. URL https://arxiv.org/abs/1805.08273. (Cited in page 30.)

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A., and Nolan, G. P. Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. *Science*, 308 (5721):523–529, 2005. doi: 10.1126/science.1105809. URL https://www.science.org/doi/abs/10.1126/science.1105809. (Cited in pages 7 and 22.)

Sanchez, P., Voisey, J. P., Xia, T., Watson, H. I., O'Neil, A. Q., and Tsaftaris, S. A. Causal machine learning for healthcare and precision medicine. *Royal Society Open Science*, 9(8):220638, 2022. (Cited in page 8.)

Sánchez-Martín, P., Rateike, M., and Valera, I. VACA: Designing Variational Graph Autoencoders for Causal queries. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022,*

The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022, pp. 8159–8168. AAAI Press, 2022. URL https://ojs.aaai.org/index.php/AAAI/article/view/20789. (Cited in page 2.)

Schäfer, J. and Strimmer, K. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical applications in genetics and molecular biology*, 4(1), 2005. (Cited in pages 1, 2, 7, and 23.)

Scutari, M. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, 35(3):1–22, 2010. doi: 10.18637/jss.v035.i03. (Cited in pages 22 and 23.)

Spirtes, P., Glymour, C., and Scheines, R. *Causation, prediction, and search*. MIT press, 2001. (Cited in page 17.)

Tchetgen, E. J. T., Ying, A., Cui, Y., Shi, X., and Miao, W. An introduction to proximal causal learning. *ArXiv preprint*, abs/2009.10982, 2020. URL https://arxiv.org/abs/2009.10982. (Cited in page 29.)

Vahdat, A. and Kautz, J. NVAE: A Deep Hierarchical Variational autoencoder. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/e3b21256183cf7c2c7a66be163579d37-Abstract.html. (Cited in page 6.)

Varian, H. R. Causal inference in economics and marketing. *Proceedings of the National Academy of Sciences*, 113 (27):7310–7315, 2016. (Cited in page 1.)

von Kügelgen, J., Sharma, Y., Gresele, L., Brendel, W., Schölkopf, B., Besserve, M., and Locatello, F. Self-Supervised Learning with Data Augmentations Provably Isolates content from style. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 16451–16467, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/8929c70f8d710e412d38da624b21c3c8-Abstract.html. (Cited in page 6.)

Wang, Y. and Blei, D. M. The blessings of multiple causes. *Journal of the American Statistical Association*, 114(528): 1574–1596, 2019. (Cited in pages 3, 7, 23, 24, and 30.)

Wang, Y. and Blei, D. M. A Proxy Variable View of Shared confounding. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10697–10707. PMLR, 2021. URL http://proceedings.mlr.press/v139/wang21c.html. (Cited in pages 1, 2, 3, 5, 15, 16, 18, 23, and 30.)

Wang, Y., Blei, D. M., and Cunningham, J. P. Posterior Collapse and Latent Variable Non-identifiability. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 5443–5455, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/2b6921f2c64dee16ba21ebf17f3c2c92-Abstract.html. (Cited in page 6.)

Wightman, L. F. Lsac National Longitudinal Bar Passage Study. LSAC Research Report Series. 1998. (Cited in page 8.)

Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. Learning Likelihoods with Conditional Normalizing flows. *ArXiv preprint*, abs/1912.00042, 2019. URL https://arxiv.org/abs/1912.00042. (Cited in pages 2, 3, and 4.)

Xi, Q. and Bloem-Reddy, B. Indeterminacy in Generative Models: Characterization and Strong identifiability. In Ruiz, F. J. R., Dy, J. G., and van de Meent, J. (eds.), *International Conference on Artificial Intelligence and Statistics, 25-27 April 2023, Palau de Congressos, Valencia, Spain*, volume 206 of *Proceedings of Machine Learning Research*, pp. 6912–6939. PMLR, 2023. URL https://proceedings.mlr.press/v206/xi23a.html. (Cited in page 5.)

Yang, M., Liu, F., Chen, Z., Shen, X., Hao, J., and Wang, J. Causalvae: Structured causal disentanglement in variational autoencoder. *ArXiv preprint*, abs/2004.08697, 2020. URL https://arxiv.org/abs/2004.08697. (Cited in page 2.)

Zečević, M., Dhami, D. S., Velivcković, P., and Kersting, K. Relating graph neural networks to structural causal models. *ArXiv preprint*, abs/2109.04173, 2021. URL https://arxiv.org/abs/2109.04173. (Cited in page 2.)

Zhao, S. and Heffernan, N. Estimating Individual Treatment Effect from Educational Studies with Residual Counterfactual Networks. *International Educational Data Mining Society*, 2017. (Cited in page 1.)

Zheng, Y., Ng, I., and Zhang, K. On the Identifiability of Nonlinear ICA: Sparsity and beyond. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/6801fa3fd290229efc490ee0cf1c5687-Abstract-Conference.html. (Cited in page 6.)

Appendix

Table of Contents

A Causal identifiability **15**
A.1 Model identifiability . . . 15
A.2 Query identifiability . . . 15
A.3 Counterfactual query identifiability . . . 20

B Experimental details and additional results **21**
B.1 Ablation study . . . 21
B.2 Semi-synthetic Sachs' dataset . . . 22
B.3 Semi-synthetic Ecoli70 dataset . . . 23
B.4 Law school fairness use-case . . . 25

C Do-operator **27**
C.1 Do-operator in causal normalizing flows . . . 27
C.2 Do-operator in interventional distributions with Decaf . . . 28
C.3 Do-operator in counterfactuals with Decaf . . . 28

D Additional details on related work of causal inference with hidden confounders **29**

E Algorithms for causal query identification **30**
E.1 Pipeline for using Decaf . . . 31

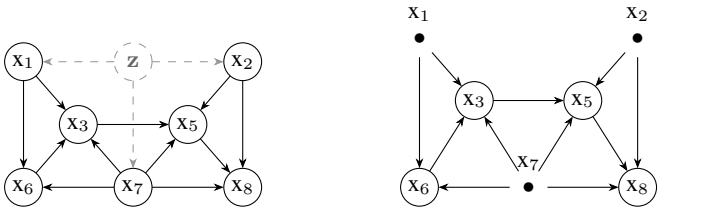
A Causal identifiability

A.1 Model identifiability

In this section, we briefly discuss the identifiability of those variables that are indirectly confounded by $\mathbf{z}$ or not confounded at all, i.e., of those variables that are not children of any hidden confounder. As we discuss now, we can reduce our SCM to a conditional SCM that only models these variables, recovering the identifiability guarantees from Javaloy et al. (2023).

To prove model identifiability, we resort to what we call the induced conditional SCM, which intuitively represents the original SCM where we restrict our view to these variables, and assume the rest of the variables are given.

Definition 4 (Induced conditional SCM)**.** Given a SCM $\mathcal{M} = (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$, and a subset of observed variables $\mathbf{x}' \subset \mathbf{x}$, we define the *induced conditional SCM of* $\mathcal{M}$ *given* $\mathbf{x}'$, denoted by $\mathcal{M}_{|\mathbf{x}'}$, to the SCM result of having observed $\mathbf{x}'$, and where causal generators and exogenous variables are restricted to only those components associated with the rest of variables.



(a) Confounded SCM. (b) Conditional unconfounded SCM.

Figure 10: Example of: **(a)** a confounded SCM $\mathcal{M}$; and **(b)** its induced conditional counterpart, $\mathcal{M}_{|\mathbf{x}'}$, when the children of the hidden confounder are observed and fixed. Note that $\mathcal{M}_{|\mathbf{x}'}$ has no hidden confounding.

We provide a visual depiction of this idea in Fig. 10. Using this definition, we can observe that, if we were to condition of the children of the hidden confounder, we would be left with a (conditional) unconfounded SCM, as the influence of the hidden confounder has been completely blocked by conditioning on its children. Now, if we have two models that perfectly match their marginal distributions, this means that they perfectly match their induced conditional SCM, no matter which value we observed for $\mathrm{ch}(\mathbf{z})$, and we can thus leverage existing results from Javaloy et al. (2023) for unconfounded SCMs.

Corollary A.1. *Assume that we have two SCMs* $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ *and* $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$ *that are compatible, i.e., they induce the same causal graph, and which coincide in their marginal distributions,* $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$*. Then, both SCMs, restricted to every variable other than* $\mathrm{ch}(\mathbf{z})$*, are equal up to an element-wise transformation of the exogenous distributions.*

Proof. The proof follows almost directly from (Javaloy et al., 2023, Theorem 1). First, note that the two induced conditional SCMs are no longer influenced by $\mathbf{z}$ once that we have observed a specific realization of $\mathrm{ch}(\mathbf{z})$, so that we can drop $\mathbf{z}$ from their structure, i.e., we can denote them by $\mathcal{M}_{|\mathrm{ch}(\mathbf{z})} = (\mathbf{f}_{|\mathrm{ch}(\mathbf{z})}, P_{\mathbf{u}|\mathrm{ch}(\mathbf{z})})$ and $\tilde{\mathcal{M}}_{|\mathrm{ch}(\mathbf{z})} = (\tilde{\mathbf{f}}_{|\mathrm{ch}(\mathbf{z})}, P_{\tilde{\mathbf{u}}|\mathrm{ch}(\mathbf{z})})$. To ease notation, let us call $\mathbf{x}^{\complement} \coloneqq \mathbf{x} \setminus \mathrm{ch}(\mathbf{z})$ the variables that are not children of $\mathbf{z}$.

Next, note that for almost every realization of $\mathrm{ch}(\mathbf{z})$, we have that $p(\mathbf{x}^{\complement} \mid \mathrm{ch}(\mathbf{z})) = \tilde{p}(\mathbf{x}^{\complement} \mid \mathrm{ch}(\mathbf{z}))$ since $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$ by assumption and $p(\mathbf{x}) = p(\mathbf{x}^{\complement} \mid \mathrm{ch}(\mathbf{z}))p(\mathrm{ch}(\mathbf{z}))$. As a result, for each realization of $\mathrm{ch}(\mathbf{z})$ we can apply Theorem 1 of Javaloy et al. (2023), which yields that the two induced conditional SCMs are equal up to an element-wise transformation of the exogenous distribution.

Finally, since the causal generators and exogenous distributions of the induced SCMs are, for almost every $\mathrm{ch}(\mathbf{z})$, identical to their counterparts in the original SCMs (as we have just discarded those components associated with $\mathrm{ch}(\mathbf{z})$), we get that the elements in the two SCMs associated with every variable except those in $\mathrm{ch}(\mathbf{z})$ are identical up to said (possibly $\mathrm{ch}(\mathbf{z})$-dependent) transformation. □

A.2 Query identifiability

We now prove the identifiability of the causal queries considered in the main text.

To this end, one key property that we will use in the following is that of completeness (see, e.g., the work of Wang & Blei (2021)). Intuitively, we say that a random variable $\mathbf{z}$ is complete given another random variable $\mathbf{n}$ if "any infinitesimal change in $\mathbf{z}$ is accompanied by variability in $\mathbf{n}$" (Miao et al., 2023), yielding enough information to recover the posterior distribution of $\mathbf{z}$. This concept is similar in spirit to that of variability in the case of discrete random variables (Nasr-Esfahany et al., 2023). In practice, completeness is more likely to be achieved the more proxies we measure (Andrews, 2011).

Definition 5 (Completeness)**.** We say that a random variable $\mathbf{z}$ is complete given $\mathbf{n}$ for all $\mathbf{c}$ if, for any square-integrable function $g(\cdot)$ and almost all $\mathbf{c}$, $\int g(\mathbf{z}, \mathbf{c}) p(\mathbf{z} \mid \mathbf{c}, \mathbf{n}) \, \mathrm{d}\mathbf{z} = 0$ for almost all $\mathbf{n}$, if and only if $g(\mathbf{z}, \mathbf{c}) = 0$ for almost all $\mathbf{z}$.

The following proposition is a generalization of the results previously presented by Miao et al. (2018) and Wang & Blei (2021), where we include an additional covariate $\mathbf{c}$ to the causal query, and make no implicit assumptions on the causal graph allowing, e.g., for the treatment and outcome variables to share some observed parents. However, note that $\mathbf{c}$ cannot be a collider (e.g., forming a subgraph of the form $\mathbf{n} \to \mathbf{c} \leftarrow \mathrm{y}$) as, otherwise, conditioning on it would make independent variables dependent (in the example, y and $\mathbf{n}$), and the causal effect of t on y would not be identifiable (Peters et al., 2017).

Proposition A.2 (Query identifiability)**.** *Assume that we have two SCMs $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ and $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$ that are compatible, i.e., they induce the same causal graph, and which coincide in their marginal distributions, $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$. Then, they compute the same causal query, $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c}) = \tilde{p}(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$, where $\mathrm{y}, \mathrm{t}, \mathbf{c} \subset \mathbf{x}$, if there exists two proxies $\mathbf{w}, \mathbf{n} \subset \mathbf{x}$ and a variable $\mathbf{b} \subset \mathbf{x}$, none of them overlapping nor containing variables from the previous subsets, such that:*

i) $\mathbf{w}$ *is conditionally independent of* $(\mathrm{t}, \mathbf{n})$ *given* $\mathbf{b}$*,* $\mathbf{z}$ *and* $\mathbf{c}$*. That is,* $\mathbf{w} \perp\!\!\!\perp (\mathrm{t}, \mathbf{n}) \mid \mathbf{b}, \mathbf{z}, \mathbf{c}$.

ii) $\mathbf{n}$ *is conditionally independent of* y *given* t*,* $\mathbf{b}$*,* $\mathbf{z}$ *and* $\mathbf{c}$*. That is,* $\mathrm{y} \perp\!\!\!\perp \mathbf{n} \mid \mathrm{t}, \mathbf{b}, \mathbf{z}, \mathbf{c}$.

iii) $(\mathbf{b}, \mathbf{z})$ *forms a valid adjustment set for the query* $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$*. That is, given* $\mathbf{c}$*, they are independent of* t *after severing any incoming edges to it,* $do(\mathrm{t}) \perp\!\!\!\perp (\mathbf{b}, \mathbf{z}) \mid \mathbf{c}$*, and they block every backdoor path from* t *to* y.

iv) $\mathbf{z}$ *is complete given* $\mathbf{n}$ *for all* t*,* $\mathbf{b}$*, and* $\mathbf{c}$*,*

v) $\tilde{\mathbf{z}}$ *is complete given* $\mathbf{w}$ *for all* $\mathbf{b}$ *and* $\mathbf{c}$*,*

and the following regularity conditions also hold:

vi) $\iint \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{w}, \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \tilde{\mathbf{z}}, \mathbf{b}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} \, \mathrm{d}\mathbf{w} < \infty$ *for all* $\mathbf{b}$*,* $\mathbf{c}$*, and*

vii) $\int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c})^2 \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{b}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} < \infty$ *for all* t*,* $\mathbf{b}$*, and* $\mathbf{c}$*.*

Proof. First, note that the first three independence assumptions hold for both models, $\mathcal{M}$ and $\tilde{\mathcal{M}}$, as they induce the same causal graph. Following the same arguments as Miao et al. (2018, Proposition 1), we have that assumptions **v)**, **vi)**, and **vii)** guarantee the existence of a function $\tilde{h}$ such that it solves the integral equation over $\tilde{\mathcal{M}}$,

$$\tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) = \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \,, \tag{10}$$

since assumption **vi)** ensures that the conditional expectation operator is compact (Carrasco et al., 2007), assumption **v)** that all square-integrable functions are in the image of the operator (i.e., the operator is surjective), and assumption **vii)** that $\tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c})$ is indeed part of the image.

We can show that $\tilde{h}$ also solves a similar integral equation, this time over the other SCM, $\mathcal{M}$, as follows:

$$\begin{aligned}
p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) &= \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) && [\textit{equal marginals}] && (11)\\
&= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} && [\textit{augment with } \tilde{\mathbf{z}}] && (12)\\
&= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} && [\textit{assumption } \textbf{\textit{ii)}}] && (13)\\
&= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} \, \mathrm{d}\mathbf{w} && [\textit{plug Eq. 10}] && (14)\\
&= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \mathbf{b}, \tilde{\mathbf{z}}, \mathrm{t}, \mathbf{n}, \mathbf{c}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} \, \mathrm{d}\mathbf{w} && [\textit{assumption } \textbf{\textit{i)}}] && (15)\\
&= \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \,. && [\textit{equal marginals}] && (16)
\end{aligned}$$

Similarly, we can relate the expression for the interventional distribution of both models:

$$\begin{aligned}
\tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) &= \int \tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\mathbf{b}, \tilde{\mathbf{z}} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\tilde{\mathbf{z}} && [\textit{augment and assumption } \textbf{\textit{iii)}}] && (17)\\
&= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\mathbf{b}, \tilde{\mathbf{z}} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\tilde{\mathbf{z}} && [\textit{backdoor criterion}] && (18)
\end{aligned}$$

$$= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c}) \tilde{p}(\mathbf{b}, \tilde{\mathbf{z}} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{w} \, \mathrm{d}\tilde{\mathbf{z}} \qquad \text{[plug Eq. 10]} \tag{19}$$

$$= \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{b}, \mathbf{w} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{w} \qquad \text{[equal marginals]} \tag{20}$$

$$= p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) \,, \tag{21}$$

where the last equality is a consequence of Eq. 16 as we will show now. More specifically, we have that

$$p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) = \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \qquad \text{[Eq. 16]} \tag{22}$$

$$= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathbf{b}, \mathbf{z}, \mathrm{t}, \mathbf{n}, \mathbf{c}) p(\mathbf{z} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \, \mathrm{d}\mathbf{z} \,, \qquad \text{[augment with } \mathbf{z}\text{]} \tag{23}$$

$$= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathbf{b}, \mathbf{z}, \mathbf{c}) p(\mathbf{z} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \, \mathrm{d}\mathbf{z} \,. \qquad \text{[assumption } \textbf{\textit{i)}}\text{]} \tag{24}$$

Similarly, we have that

$$p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) = \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{z}, \mathbf{c}) p(\mathbf{z} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{z} \qquad \text{[augment with } \mathbf{z}\text{]} \tag{25}$$

$$= \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{z}, \mathbf{c}) p(\mathbf{z} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{z} \,. \qquad \text{[assumption } \textbf{\textit{ii)}}\text{]} \tag{26}$$

Now, equating both expressions we have that

$$0 = \iint \left\{ p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{z}, \mathbf{c}) - \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathbf{b}, \mathbf{z}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \right\} p(\mathbf{z} \mid \mathrm{t}, \mathbf{b}, \mathbf{n}, \mathbf{c}) \, \mathrm{d}\mathbf{z} \,, \tag{27}$$

which, due to assumption **iv)**, implies that

$$p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{z}, \mathbf{c}) \overset{\text{a.e.}}{=} \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathbf{b}, \mathbf{z}, \mathbf{c}) \, \mathrm{d}\mathbf{w} \,. \tag{28}$$

Finally, putting all together we see that we can write the interventional distribution of the original model using $\tilde{h}$,

$$p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) = \iint p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{b}, \mathbf{z}, \mathbf{c}) p(\mathbf{b}, \mathbf{z} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{z} \qquad \text{[augment and assumption } \textbf{\textit{iii)}}\text{]} \tag{29}$$

$$= \iint p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{z}, \mathbf{c}) p(\mathbf{b}, \mathbf{z} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{z} \qquad \text{[backdoor criterion]} \tag{30}$$

$$= \iint \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{w} \mid \mathbf{b}, \mathbf{z}, \mathbf{c}) p(\mathbf{b}, \mathbf{z} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{w} \qquad \text{[Eq. 28]} \tag{31}$$

$$= \int \tilde{h}(\mathrm{y}, \mathrm{t}, \mathbf{b}, \mathbf{w}, \mathbf{c}) p(\mathbf{b}, \mathbf{w} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \, \mathrm{d}\mathbf{w} \,, \qquad \text{[equal marginals]} \tag{32}$$

which justifies the last equality in Eq. 21. □

Using a causal graph similar to the one presented by Miao et al. (2018), we now provide some intuition on the semantics of each random variable in Prop. A.2. More specifically, consider the causal graph that we depict in Fig. 11, and say that we want to identify the causal query $p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$ (that is, the same query as in Prop. A.2 but with $\mathbf{c} = \emptyset$). As it is common in the causal inference literature (Peters et al., 2017; Spirtes et al., 2001), t and y represent the treatment and outcome random variables. More specific to Prop. A.2 are $\mathbf{n}$ and $\mathbf{w}$. The variable $\mathbf{w}$ is a proxy variable whose role is that of distinguishing the information from $\mathbf{z}$ and other variables, to reconstruct the information of $\mathbf{z}$ and block the backdoor path that $\mathbf{z}$ would usually block. Similarly, the variable $\mathbf{n}$ is another proxy variable which, in this case, serves the purpose of verifying that the substitute formed

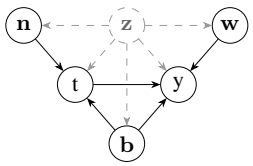


Figure 11: Example for which Prop. A.2 applies, and where $\mathbf{b}$ is not the empty set.

with $\mathbf{w}$ is indeed a good substitute. Finally, the variable $\mathbf{b}$ serves the purpose of blocking all the remaining backdoor paths that $\mathbf{z}$ may not block, so that we can apply the backdoor criterion.

Moreover, note that for all interventional queries we will let $\mathbf{c}$ be the empty set, similar to the results proved by Miao et al. (2018) and Wang & Blei (2021). We will consider cases when $\mathbf{c}$ is not empty later in App. A.3 to prove counterfactual identifiability. Note also that Prop. A.2 reduces to the existing results when we have that $\mathbf{c} = \mathbf{b} = \emptyset$.

Using this general proposition, we can now reason about causal identifiability in a wide range of scenarios, where t and y may or may not be directly caused by the hidden confounder, as we show in the following subsections.

A.2.1 Unconfounded case

First, we consider the case where neither t nor y are directly affected by the hidden confounder, i.e., $\mathbf{z} \notin \mathrm{ch}(\mathbf{z})$. In this case, the proof can be simplified and drop the requirement of finding valid proxy variables.

Corollary A.3 (Unconfounded case)**.** *Assume that we have two SCMs $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ and $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$ that are compatible, i.e., they induce the same causal graph, and which coincide in their marginal distributions, $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$. Assume that $\mathrm{y}, \mathrm{t} \notin \mathrm{ch}(\mathbf{z})$. Then, $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c}) = \tilde{p}(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$, where $\mathrm{y}, \mathrm{t}, \mathbf{c} \subset \mathbf{x}$.*

Proof. The proof follows directly by applying Prop. A.2 with the minimal subset $\mathbf{b} \subset \mathrm{pa}(\mathrm{t}) \setminus \{\mathbf{c}\}$ that blocks all the backdoor paths, and by noticing that in this case there is no need to use the variables $\mathbf{z}$ and $\tilde{\mathbf{z}}$. That is, we can go from Eq. 17 to Eq. 21 directly by using only $\mathbf{b}$ and the equal-marginals assumption:

$$\tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) = \int \tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{b} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \tag{33}$$

$$= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{b} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \tag{34}$$

$$= \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{c}) p(\mathbf{b} \mid \mathbf{c}) \, \mathrm{d}\mathbf{b} \tag{35}$$

$$= p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) \, . \tag{36}$$

□

Even though we can leverage and simplify Prop. A.2 as shown above, it is worth remarking that, for this particular case, the model identifiability results described in App. A.1 are stronger, as it provides results on the identifiability of the causal generators and exogenous distributions, and therefore of any causal query derived from them.

A.2.2 Fully confounded case

In the case where both variables are directly confounded by $\mathbf{z}$, we cannot do much but to see whether we can apply Prop. A.2 with $\mathbf{c} = \emptyset$ and a valid $\mathbf{b}$. If we manage to find two proxies $\mathbf{w}$ and $\mathbf{n}$ that hold the independence conditions from Prop. A.2 and that change the posterior of $\mathbf{z}$ enough, then we can use the proposition to ensure the identifiability of the query. Otherwise, the query is not identifiable and the model might or might not estimate the query correctly.

A.2.3 Confounded outcome case

For the case where only the outcome random variable is directly affected by the hidden variable, we can apply a similar reasoning as we did in the case with no direct confounding, although this time we cannot leverage the model identifiability results from Javaloy et al. (2023). More specifically:

Corollary A.4 (Confounded-outcome case)**.** *Assume that we have two SCMs $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ and $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$ that are compatible, i.e., they induce the same causal graph, and which coincide in their marginal distributions, $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$. Assume that $\mathrm{t} \notin \mathrm{ch}(\mathbf{z})$. Then, $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c}) = \tilde{p}(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$, where $\mathrm{y}, \mathrm{t}, \mathbf{c} \subset \mathbf{x}$.*

Proof. The proof is identical to that of Cor. A.3. □

Front-door example. While the proof above is trivial given the previous results, it is worth stressing that for them to hold it is necessary to model the hidden confounder as we do in this work with the proposed Decaf, and that other approaches may not work for all cases. As an example, consider the SCM depicted in Fig. 12, where we have that the outcome is directly confounded by $\mathbf{z}$, while t is not. In this case, a Decaf should be able to identify the true causal query $p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$, using $\tilde{\mathbf{z}}$ to model the influence of $\mathbf{b}$

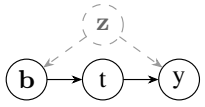


Figure 12: Textbook example of a front-door in a SCM.

onto y that is not explained through t. Other models that do not model $\mathbf{z}$ (e.g., an unaware causal normalizing flow (Javaloy et al., 2023)), would not be able to match the observed marginal likelihood as they assume that $\mathrm{y} \perp\!\!\!\perp \mathbf{b} \mid \mathrm{t}$ yet we know that $\mathrm{y} \not\perp\!\!\!\perp \mathbf{b} \mid \mathrm{t}$ in the true model. Even more, with those models we would have that $p(\mathrm{y} \mid \mathrm{do}(\mathrm{t})) = p(\mathrm{y} \mid \mathrm{t})$ which is clearly false by just looking at Fig. 12.

To be even more explicit, in this case we would have a factorization of the form

$$\tilde{p}(\mathbf{b}, \mathrm{t}, \mathrm{y}, \tilde{\mathbf{z}}) = \tilde{p}(\tilde{\mathbf{z}})\tilde{p}(\mathbf{b} \mid \tilde{\mathbf{z}})\tilde{p}(\mathrm{t} \mid \mathbf{b})\tilde{p}(\mathrm{y} \mid \mathrm{t}, \tilde{\mathbf{z}})\,. \tag{37}$$

Then, the estimated interventional distribution that a Decaf estimates as $\int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \tilde{\mathbf{z}})\,\mathrm{d}\tilde{\mathbf{z}}$ equals the true one:

$$\begin{aligned}
p(\mathrm{y} \mid \mathrm{do}(\mathrm{t})) &= \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}) p(\mathbf{b})\,\mathrm{d}\mathbf{b} && [\mathbf{b} \textit{ forms a valid adjustment set}] && (38)\\
&= \int \left\{ \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathrm{t}, \mathbf{b})\,\mathrm{d}\tilde{\mathbf{z}} \right\} \tilde{p}(\mathbf{b})\,\mathrm{d}\mathbf{b} && [\textit{latent factorization and equal marginals}] && (39)\\
&= \iint \tilde{p}(\mathrm{y} \mid \mathrm{t}, \tilde{\mathbf{z}}) \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{b}) \tilde{p}(\mathbf{b})\,\mathrm{d}\mathbf{b}\,\mathrm{d}\tilde{\mathbf{z}} && [\textit{causal graph factorization in Eq. 37}] && (40)\\
&= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \tilde{\mathbf{z}}) \tilde{p}(\tilde{\mathbf{z}})\,\mathrm{d}\tilde{\mathbf{z}} && [\textit{marginalize } \mathbf{b}] && (41)\\
&= \tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))\,. && && (42)
\end{aligned}$$

Remarkably, the identification of $p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))$ allows us to solve also the query $p(\mathrm{y} \mid \mathrm{do}(\mathbf{b}))$ leveraging the frontdoor criterion (Peters et al., 2017).

$$\begin{aligned}
p(\mathrm{y} \mid \mathrm{do}(\mathbf{b})) &= \int p(\mathrm{t} \mid \mathbf{b}) p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}))\,\mathrm{d}\mathrm{t} && [\textit{frontdoor criterion}] && (43)\\
&= \int \tilde{p}(\mathrm{t} \mid \mathbf{b}) \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \tilde{\mathbf{z}}) \tilde{p}(\tilde{\mathbf{z}})\,\mathrm{d}\tilde{\mathbf{z}}\,\mathrm{d}\mathrm{t} && [\textit{plug in Eq. 41 and equal marginals}] && (44)\\
&= \tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathbf{b})) && && (45)
\end{aligned}$$

A.2.4 Confounded treatment case

When only the treatment variable t is directly confounded, we can find two different scenarios: if we are able to find a valid adjustment set $\mathbf{b}$ blocking all confounded paths, in which case we can reason just as in the other partially confounded case, and otherwise, where we rely on the identifiability with respect to this invalid adjustment set. For example, if it happens to be a parent of y which is directly caused by the treatment variable t and the hidden confounder $\mathbf{z}$ as in Fig. 13, we cannot find a valid adjustment set for the causal query, but an invalid one may still serve us if we can identify the same query with the adjustment set as outcome.

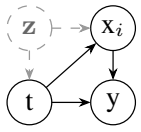


Figure 13: Case where no valid adjustment set can be found.

Corollary A.5 (Confounded-treatment case)**.** *Assume that we have two compatibleSCMs* $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ *and* $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$*, i.e., they induce the same causal graph, and which coincide in their marginal distributions,* $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$*. Assume also that* $\mathrm{y} \notin \mathrm{ch}(\mathbf{z})$*. Then,* $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c}) = \tilde{p}(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$*, where* $\mathrm{y}, \mathrm{t}, \mathbf{c} \subset \mathbf{x}$ *if there exists a subset* $\mathbf{b} \subset \mathbf{x}$ *not containing variables from the previous subsets, such that one of the following two conditions are true:*

i) $\mathbf{b}$ *forms a valid adjustment set for the query* $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$*.*

ii) $\mathbf{b}$ *forms an invalid adjustment set for the query* $p(\mathrm{y} \mid do(\mathrm{t}), \mathbf{c})$ *but the query* $p(\mathbf{b} \mid do(\mathrm{t}), \mathbf{c})$ *is identifiable. That is,* $\mathbf{b}$ *blocks all the backdoor paths, and* $p(\mathbf{b} \mid do(\mathrm{t}), \mathbf{c}) = \tilde{p}(\mathbf{b} \mid do(\mathrm{t}), \mathbf{c})$*.*

Proof. If condition **i)** holds, then we have a valid adjustment set, and the proof is identical to that of Cor. A.3.

Otherwise, if condition **ii)** holds, we have that the interventional query on y equals the observational query when conditioned on $\mathbf{b}$, but that now $\mathbf{b}$ is not independent of do(t), i.e.,

$$\tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) = \int \tilde{p}(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{b} \mid \mathrm{do}(\mathrm{t}), \mathbf{c})\,\mathrm{d}\mathbf{b} \tag{46}$$

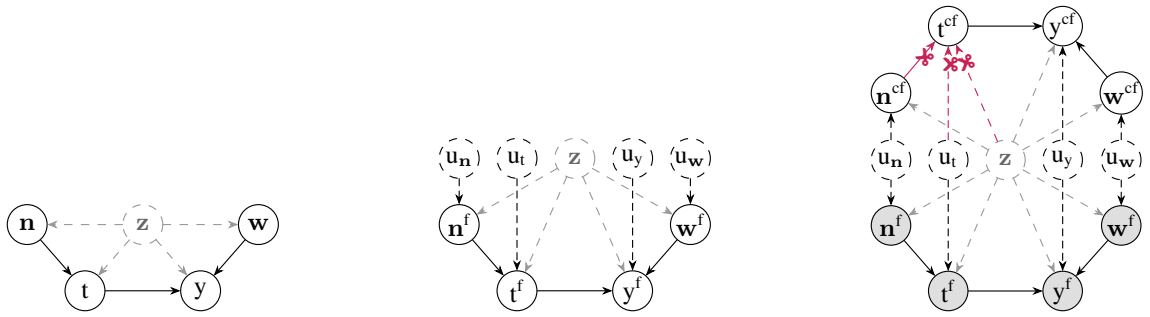


Figure 14: Example of the transition from **(a)** the regular depiction of a (confounded) SCM, to **(b)** an explicit SCM where the exogenous variables are drawn, and **(c)** a counterfactual twin SCM where the data-generating process is replicated in the "factual and counterfactual worlds". Besides, figure **(c)** also depicts which nodes are observed and which edges are severed, in order to compute a counterfactual query of the type $p(\mathrm{y}^{\mathrm{cf}} \mid \mathrm{do}(\mathrm{t}^{\mathrm{cf}}), \mathbf{x}^{\mathrm{f}})$.

$$= \int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{b} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) \, \mathrm{d}\mathbf{b} \tag{47}$$

$$= \int p(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \mathbf{c}) p(\mathbf{b} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) \, \mathrm{d}\mathbf{b} \tag{48}$$

$$= p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c}) \,, \tag{49}$$

where we needed to use that the query $p(\mathbf{b} \mid \mathrm{do}(\mathrm{t}), \mathbf{c})$ is identifiable in the third equality. □

A.3 Counterfactual query identifiability

In this section, we show that counterfactual query identifiability is a direct result of the interventional query identifiability from the previous section.

In order to formally define counterfactuals, in this section we introduce the concept of counterfactual SCMs in a somewhat novel way. Namely, we combine the concepts of twin networks from Pearl (2009) (which replicates the data-generating process) and that of counterfactual SCMs from Peters et al. (2017) (which defines a counterfactual *prior* to the intervention).

Definition 6 (Counterfactual twin SCM)**.** Given a SCM $\mathcal{M} = (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$, we define its counterfactual twin SCM as a SCM $\mathcal{M}^{\mathrm{cf}}$ where all structural equations are duplicated, and the exogenous noise is shared across replications, and where additionally one of the halves is observed ("the factual world"), and the other half is unobserved ("the counterfactual world").

We provide in Fig. 14 a more intuitive depiction on the construction of these counterfactual twin networks. From this definition, one can recover the counterfactual SCM defined by Peters et al. (2017) by just focusing on the replicated part of the counterfactual twin network, and conditioning the exogenous noise and hidden confounder on the observed half, i.e., $(\mathbf{f}, P_{\mathbf{u} \mid \mathbf{x}^{\mathrm{f}}}, P_{\mathbf{z} \mid \mathbf{x}^{\mathrm{f}}})$. Similarly, one can compute the usual counterfactual query by performing an intervention on the counterfactual twin network, i.e., by replacing the intervened equations by the constant intervened value, and computing the query conditioned on the factual variables, $p(\mathrm{y}^{\mathrm{cf}} \mid \mathrm{do}(\mathrm{t}^{\mathrm{cf}}), \mathbf{x}^{\mathrm{f}})$. This is visually represented in Fig. 14c.

In order to prove query identifiability in the counterfactual setting, we need to use the following technical result regarding the completeness of a random variable:

Lemma A.6. *If a random variable* $\mathbf{z}$ *is complete given* $\mathbf{n}$ *for all* $\mathbf{b}$*, as given by Def. 5, then it is complete given* $\mathbf{n}$ *for all* $\mathbf{b}$ *and* $\mathbf{c}$*, where* $\mathbf{c}$ *is another continuous random variable.*

Proof. We prove this result by contradiction. Assume that the result does not hold, then there must exist a non-zero measure subset of the space of $\mathbf{b} \times \mathbf{c}$for which there exists a square-integrable function $g(\cdot)$ such that $\int g(\mathbf{z}, \mathbf{b}, \mathbf{c}) p(\mathbf{z} \mid \mathbf{b}, \mathbf{c}, \mathbf{n}) \, \mathrm{d}\mathbf{z} = 0$ for almost all $\mathbf{n}$, but $g(\mathbf{z}, \mathbf{b}, \mathbf{c}) \neq 0$ for almost all $\mathbf{z}$.

Since this subset has positive measure, there must contain an ε-ball within. If we now focus on the $\mathbf{b}$-projection of this ball where we fix $\mathbf{c}$ to its value on the centre, we have that it is a subset of non-zero measure in the space of $\mathbf{b}$ (as otherwise it would be zero-measure in the Cartesian-product measure), where the function $g(\cdot, \mathbf{c})$ breaks our initial assumption of the completeness of $\mathbf{z}$. Thus, we reach a contradiction. □

Given Def. 6, it is rather intuitive that, if a causal query is identifiable in a SCM $\mathcal{M}$, then it has to be identifiable in both halves of its induced counterfactual twin SCM $\mathcal{M}^{\text{cf}}$, as they are identical. More importantly, we can now leverage again Prop. A.2, this time with $\mathbf{c} = \mathbf{x}^{\text{f}}$, to prove counterfactual query identifiability whenever we have interventional query identifiability.

Proposition A.7 (Counterfactual identifiability)**.** *Assume that we have two SCMs $\mathcal{M} \coloneqq (\mathbf{f}, P_{\mathbf{u}}, P_{\mathbf{z}})$ and $\tilde{\mathcal{M}} \coloneqq (\tilde{\mathbf{f}}, P_{\tilde{\mathbf{u}}}, P_{\tilde{\mathbf{z}}})$ that are compatible, i.e., they induce the same causal graph, and which coincide in their marginal distributions, $p(\mathbf{x}) = \tilde{p}(\mathbf{x})$. Then, if a query $p(\mathrm{y} \mid do(\mathrm{t}))$ is identifiable in the sense of Prop. A.2, where $\mathrm{y}, \mathrm{t} \subset \mathbf{x}$, then the query $p(\mathrm{y}^{\text{cf}} \mid do(\mathrm{t}^{\text{cf}}), \mathbf{x}^{\text{f}})$ is also identifiable in their induced counterfactual twin SCMs as long as the regularity conditions still hold, i.e., if:*

i) $\iint \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{w}, \mathbf{b}, \mathbf{c}) \tilde{p}(\mathbf{w} \mid \tilde{\mathbf{z}}, \mathbf{b}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} \, \mathrm{d}\mathbf{w} < \infty$ *for all* $\mathbf{b}$, $\mathbf{c}$, *and*

ii) $\int \tilde{p}(\mathrm{y} \mid \mathrm{t}, \mathbf{b}, \tilde{\mathbf{z}}, \mathbf{c})^2 \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{b}, \mathbf{c}) \, \mathrm{d}\tilde{\mathbf{z}} < \infty$ *for all* t, $\mathbf{b}$, *and* $\mathbf{c}$.

Proof. We essentially need to prove that the independence and completeness assumptions keep holding when we add the factual covariate, $\mathbf{c} = \mathbf{x}^{\text{f}}$.

For the independence, we need to show that, if we have a set of variables that fulfil the independence conditions from Prop. A.2, then this set of variables keeps holding them if we include $\mathbf{c} = \mathbf{x}^{\text{f}}$. This is, however, easy to show since factual and counterfactual variables only have "tail-to-tail" dependencies, i.e., they are connected only through the shared exogenous variables. As a result, if two variables from the same half are conditionally independent given a third set of variables, conditioning on the other half cannot change this independence.

For the completeness, we need to show that introducing the factual variable retain the completeness assumed in Prop. A.2. However, this is direct to show using the technical result in Lemma A.6. Specifically, it holds that

i) $\mathbf{z}$ is complete given $\mathbf{n}$ for all t, $\mathbf{b}$, and $\mathbf{c}$, and

ii) $\tilde{\mathbf{z}}$ is complete given $\mathbf{w}$ for all $\mathbf{b}$ and $\mathbf{c}$.

Therefore, we have shown that the requirements of Prop. A.2 hold when we append a factual variable in the twin network, and thus we can reapply all the results from the previous sections to the counterfactual cases. □

It is important to note that, while the results above provide counterfactual identifiability whenever we have interventional identifiability, we still rely on how much of a good approximation the encoder is to the inverse of the decoder in the proposed Decaf model. That is, the quality of the encoder determines how well we can perform the abduction step to compute counterfactuals. This consideration is unique to counterfactuals, as we just had to sample the latent variable as usual in the case of interventional queries.

B Experimental details and additional results

B.1 Ablation study

First, in Fig. 15 we present the ATE error committed for each combination of proxies and latent dimension, complementary to the figure of the presented in §7.1. If we observe the ATE error, we extract the same conclusion as observing counterfactual error, the causal effect is not recoverable with less than two proxies, and more proxies result in better estimates. On the other hand, the selection of the dimension of the latent space bigger than the true dimension of the latent confounders does not affect the performance negatively.

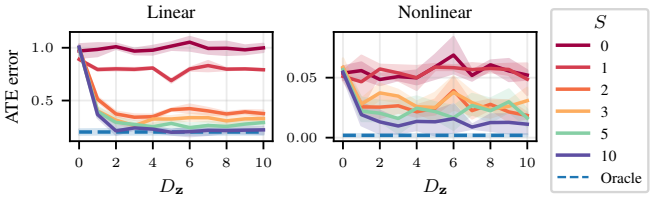


Figure 15: ATE absolute error varying the number of available proxies (S) and the dimensionality of the latent space ($D_{\mathbf{z}}$). Mean and 95% confidence interval over 5 realizations and all interventions, made in percentiles 25, 50 and 75 of $\mathbf{x}_1$. Oracle represents a causal normalizing flow that observes $\mathbf{z}$.

In addition, we show the equations that we have used for the ablation study. There exist two unobserved confounders, $\mathbf{z}_1$ and $\mathbf{z}_2$. The set of all observed proxies $\{\mathrm{x}_3, \mathrm{x}_4, ..., \mathrm{x}_{12}\}$ is represented in the graph of §7.1 as $\mathbf{x}_3$. Note that the proxies available in the nonlinear experiment are bounded or periodic, specially sigmoids and hyperbolic tangents saturate and $\max(0, \mathrm{x})$ loses all the information about the confounder for negative values and sines and cosines are periodic functions. In other words, the distributions $p(\mathbf{z} \mid \mathrm{x}_i)$ are not complete, we lose information about $\mathbf{z}$ when in the transformations to each x. However, if we add more

proxies of the confounders, the information that the proxies contain about the confounder is higher, and the causal effect of x_1 on x_2 becomes recoverable.

Linear

$$
\begin{cases}
\mathbf{z}_1 \sim P_{\mathbf{z}_1} \\
\mathbf{z}_2 \sim P_{\mathbf{z}_2} \\
\mathrm{x}_1 = 1.5 \cdot \mathbf{z}_1 + 0.5 \cdot \mathbf{z}_2 + 0.4 \cdot \mathrm{u}_1 \\
\mathrm{x}_2 = -0.75 \cdot \mathbf{z}_1 + 0.6 \cdot \mathbf{z}_2 + 0.9 \cdot \mathrm{x}_1 + 0.3 \cdot \mathrm{u}_2 \\
\mathrm{x}_3 = -0.5 \cdot \mathbf{z}_1 + 0.3 \cdot \mathbf{z}_2 + 0.5 \cdot \mathrm{u}_3 \\
\mathrm{x}_4 = 0.75 \cdot \mathbf{z}_1 - 0.4 \cdot \mathbf{z}_2 + 0.4 \cdot \mathrm{u}_4 \\
\mathrm{x}_5 = -0.85 \cdot \mathbf{z}_1 + 0.6 \cdot \mathbf{z}_2 + 0.6 \cdot \mathrm{u}_5 \\
\mathrm{x}_6 = 0.6 \cdot \mathbf{z}_1 + 0.6 \cdot \mathbf{z}_2 + 0.55 \cdot \mathrm{u}_6 \\
\mathrm{x}_7 = -0.8 \cdot \mathbf{z}_1 + 0.4 \cdot \mathbf{z}_2 + 0.4 \cdot \mathrm{u}_7 \\
\mathrm{x}_8 = 0.9 \cdot \mathbf{z}_1 - 0.7 \cdot \mathbf{z}_2 + 0.6 \cdot \mathrm{u}_8 \\
\mathrm{x}_9 = -0.72 \cdot \mathbf{z}_1 + 0.5 \cdot \mathbf{z}_2 + 0.56 \cdot \mathrm{u}_9 \\
\mathrm{x}_{10} = 0.78 \cdot \mathbf{z}_1 + 0.4 \cdot \mathbf{z}_2 + 0.58 \cdot \mathrm{u}_{10} \\
\mathrm{x}_{11} = -0.55 \cdot \mathbf{z}_1 + 0.7 \cdot \mathbf{z}_2 + 0.6 \cdot \mathrm{u}_{11} \\
\mathrm{x}_{12} = 0.88 \cdot \mathbf{z}_1 + 0.3 \cdot \mathbf{z}_2 + 0.4 \cdot \mathrm{u}_{12}
\end{cases}
$$

Nonlinear

$$
\begin{cases}
\mathbf{z}_1 \sim P_{\mathbf{z}_1} \\
\mathbf{z}_2 \sim P_{\mathbf{z}_2} \\
\mathrm{x}_1 = \dfrac{\mathbf{z}_1^2}{4} \cdot \sin\left(\dfrac{\mathbf{z}_2}{2}\right) + \mathbf{z}_1 + 0.6 \cdot \mathrm{u}_1 \\
\mathrm{x}_2 = \dfrac{\mathbf{z}_1 \cdot \mathrm{x}_1}{4} + 0.8 \cdot \mathbf{z}_2 + 0.5 \cdot \mathrm{x}_1 + \mathrm{x}_1 \cdot \mathrm{u}_2 \cdot 0.3 + 0.2 \cdot \mathrm{u}_2 \\
\mathrm{x}_3 = 0.6 \cdot \mathbf{z}_1^2 + \left(\dfrac{\mathbf{z}_2}{4}\right)^3 + 0.3 \cdot \sin\left(\dfrac{\mathbf{z}_2}{2}\right) + 0.5 \cdot \mathrm{u}_3 \\
\mathrm{x}_4 = \sin\left(\dfrac{\mathbf{z}_1}{2}\right) + \cos\left(\dfrac{\mathbf{z}_2}{3}\right) + 0.4 \cdot \mathrm{u}_4 \\
\mathrm{x}_5 = \cos\left(\dfrac{\mathbf{z}_1}{2}\right) - \tanh\left(\dfrac{\mathbf{z}_2}{3}\right) + 0.6 \cdot \mathrm{u}_5 \\
\mathrm{x}_6 = \tanh\left(\dfrac{\mathbf{z}_1}{2}\right) + \sigma\left(\dfrac{\mathbf{z}_2}{2}\right) + 0.55 \cdot \mathrm{u}_6 \\
\mathrm{x}_7 = \sigma\left(\dfrac{\mathbf{z}_1}{2}\right) + \max(0, -\mathbf{z}_2) + 0.4 \cdot \mathrm{u}_7 \\
\mathrm{x}_8 = \max(0, \mathbf{z}_1) - 0.5 \cdot \max(0, \mathbf{z}_2) + 0.6 \cdot \mathrm{u}_8 \\
\mathrm{x}_9 = \max(0, -\mathbf{z}_1) + 0.3 \cdot \max(0, -\mathbf{z}_2) + 0.5 \cdot \mathbf{z}_1 \cdot \mathrm{u}_9 \\
\mathrm{x}_{10} = 0.8 \cdot \max(0, \mathbf{z}_1) + 0.3 \cdot \max(0, \mathbf{z}_2) + 0.58 \cdot \mathrm{u}_{10} \\
\mathrm{x}_{11} = 0.75 \cdot \max(0, -\mathbf{z}_1) + 0.5 \cdot \max(0, \mathbf{z}_2) + 0.6 \cdot \mathrm{u}_{11} \\
\mathrm{x}_{12} = 0.3 \cdot \mathbf{z}_1^3 + 0.5 \cdot |\mathbf{z}_2| + 0.4 \cdot \mathrm{u}_{12}
\end{cases}
$$

B.2 Semi-synthetic Sachs' dataset

This dataset represents a network of protein-signaling in human T lymphocites. Every variable, except `PKA` and `Plcg` can be intervened upon; therefore, there is not only one causal query of interest, but tens of possible causal queries can arise in this setting. This highlights one of the strenghts of Decaf, because we only need a single trained model to answer all identifiable causal queries.

The original data contains a total of 853 observational samples; however, we have decided to evaluate our model on semi-synthetic data because of the following reasons:

- The original network of Sachs et al. (2005) contains cycles, which is violation of one of our assumptions. However, we have found different versions of the causal graph (Kaltenpoth & Vreeken, 2023; Luo & Zhao, 2011) that do not contain cycles. Therefore, we have decided to employ the causal graph that appears in the library *bnlearn* (Scutari, 2010)—a recognized library for Bayesian Nerwork learning—as ground truth causal graph. The best way to ensure that the causal graph used is the ground truth is by generating samples according to the causal graph. In addition, that causal graph is the one used by Chao et al. (2023).
- We can compare our model with one of the baseline models, DCM, with the same dataset as Chao et al. (2023) used.
- Semi-synthetic data allows us to compute all metrics to evaluate causal queries, having the ground truth.

For generating the data in this experiment, we have followed the procedure proposed by Chao et al. (2023), where they take the causal graph of Sachs et al. (2005) and the empirical distribution of the root nodes, and generate the rest of the variables with random non-linear mechanisms. In addition, exogenous variables have been included in an additive and non-additive manner, respectively.

In the following, we complement the figures presented in §7 with a table that summarizes all the interesting metrics, evaluated on the confounded identifiable causal queries shown in Fig. 7. Interventional distributions and counterfactuals have been computed intervening in percentiles 25, 50 and 75 of the intervened variable.

Since observational MMD is computed only once, the statistics given in Tab 2 are calculated *only* over 5 runs. On the other hand, we have as many interventional MMDs per run as interventions have been made. However, the statistics of

interventional MMD are computed over all the interventions of all intervened variables and 5 runs (5 runs $\times$ 3 intervened variables = 15 samples). Finally, statistics over counterfactual error and ate error aggregate all the intervention-outcome pairs over the five runs. For example, in this case we intervene on 3 variables, performing 3 different interventions and we evaluate on 3, 2, and 1 variable respectively for each intervened variable, and we have a total of (3+2+1)$\times$3$\times$5 = 90 different measurements to compute the statistics.

Table 2: Performance metrics on Sachs datasets. Mean_{std} over five runs and all causal queries of interest. Interventions on `Raf`, `Mek` and `Akt` and evaluating on confounded identifiable effects. Bold indicates significantly better results (95% CI from a Mann-Whitney U test). Lower error values indicate better performance.

		Additive				Nonadditive			
	Model	MMD obs $\times 10^4$	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$	MMD obs $\times 10^4$	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$
Oracle	CNF	$4.84_{1.84}$	$7.50_{6.17}$	$6.05_{6.83}$	$10.03_{10.29}$	$5.96_{2.37}$	$6.71_{2.97}$	$2.34_{2.02}$	$4.84_{3.43}$
Aware	Decaf	$\mathbf{2.15_{0.54}}$	$\mathbf{7.82_{9.59}}$	$\mathbf{17.89_{17.72}}$	$\mathbf{17.92_{4.01}}$	$5.12_{2.42}$	$\mathbf{5.39_{3.33}}$	$\mathbf{3.26_{4.09}}$	$\mathbf{6.82_{4.65}}$
	Deconfounder	–	–	$34.34_{33.45}$	$71.13_{86.98}$	–	–	$8.14_{10.69}$	$63.15_{79.12}$
Unaware	CNF	$5.80_{1.58}$	$73.94_{88.78}$	$44.49_{39.12}$	$56.09_{38.89}$	$5.11_{1.90}$	$12.79_{20.73}$	$9.74_{15.71}$	$15.15_{15.37}$
	ANM	$83.86_{13.41}$	$110.28_{112.43}$	$22.42_{14.06}$	$29.40_{12.22}$	$81.90_{7.21}$	$60.40_{144.08}$	$23.88_{13.94}$	$28.97_{12.44}$
	DCM	$87.80_{2.95}$	$125.59_{118.20}$	$21.21_{11.34}$	$28.25_{6.96}$	$14.23_{4.57}$	$69.74_{390.81}$	$8.44_{7.96}$	$27.50_{23.71}$

The metrics in Tab 2 indicate that Decaf outperforms all baselines across all interventional and counterfactual causal queries in both settings of the semi-synthetic datasets. However, as discussed in §8, a limitation of our empirical approach is that the differences in observational MMD, the selection criterion for CGMs, are marginal between the *oracle*, Decaf, and CNF. Notably, Decaf even achieves a lower MMD than the *oracle*. This discrepancy arises because the number of variables is large, and the MMD differences are on the order of 10^{-4}.

B.3 Semi-synthetic Ecoli70 dataset

The Ecoli 70 dataset represent the gene expression of 46 genes of the RNA-seq of *Escherichia coli* bacteria. The assumed causal graph comes from the study of (Schäfer & Strimmer, 2005), which provides insight into the regulatory mechanisms governing *E. coli* gene expression. Examples of interventions in these networks are gene knockout and gene overexpression (Long & Antoniewicz, 2014). A priori, there could be several variables in which intervening can be interesting in evaluating the effects in the cell.

For this experiment, we have generated the data in the same way as done with Sachs' dataset with random mechanisms, but in this case, since we do not have enough samples, root nodes follow standard Gaussian distributions. We have included an additive and a nonadditive ways of including exogenous variables. In this case, we have used a semi-synthetic dataset because the real dataset available in *bnlearn* (Scutari, 2010) contains only 9 samples.

In Fig. 1 is presented the causal graph of this setting.

In addition, note that Fig. 1 has been extracted from our Alg. 5 of causal effect identfiability. That is, we have specified the causal graph and the variables that are unmeasured, and our Algorithm returns (in green) all the paths that are identifiable by Decaf. Consider that black arrows are also identifiable, not only by Decaf, but also for any CGM that approximates the observed data. In red, arrows that are not identifiable by Decaf because there are not enough proxies to infer an unbiased causal effect.

A table summarizing the results obtained in the estimation confounded identifiable causal queries are presented in Tab 3. The statistics have been computed in the same way as in Sachs' dataset. In the case of ATE and CF error, they have been computed only on the *direct* confounded identifiable paths, i.e., the green paths in Fig. 1.

Decaf significantly outperforms the baselines in ATE and counterfactual estimation in the additive setting and in ATE estimation in the nonadditive setting. The MMD differences, both observational and interventional, are negligible between the *oracle*, Decaf, and CNF, likely due to the high number of variables diluting estimation bias. Counterfactual differences in the nonadditive setting are also insignificant. However, compared to the *oracle*, the gap between the *oracle* and *unaware* CGMs is smaller than in the additive case. While Decaf reaches an intermediate point, the difference remains insignificant.

B.3.1 Comment on Deconfounder results

One may realize that the errors committed by the Deconfounder of (Wang & Blei, 2019; 2021) are greater than the errors committed by the unaware models. First of all, we want to underline that, although the Deconfounder allows us to predict

Table 3: Performance metrics on Ecoli70 dataset. ATE and CF error statistics computed aggregating all causal queries and 5 runs. Intervened and evaluated on the direct confounded identifiable causal effects of Fig. 1. Bold indicates significantly better results (95% CI from a Mann-Whitney U test). Lower error values indicate better performance.

	Model	Additive				Nonadditive			
		MMD obs $\times 10^4$	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$	MMD obs $\times 10^4$	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$
Oracle	CNF	$2.34_{0.62}$	$6.05_{5.28}$	$5.04_{7.42}$	$9.91_{12.46}$	$1.49_{0.57}$	$4.05_{8.22}$	$3.51_{4.84}$	$1.67_{1.64}$
Aware	Decaf ☕	$2.97_{0.34}$	$10.00_{9.11}$	$\mathbf{7.29_{8.61}}$	$\mathbf{17.48_{12.98}}$	$1.63_{0.46}$	$9.19_{21.42}$	$\mathbf{8.72_{17.61}}$	$2.15_{2.10}$
	Deconfounder	–	–	$27.35_{26.17}$	$82.15_{116.90}$	–	–	$30.00_{33.24}$	$9.90_{9.47}$
Unaware	CNF	$2.98_{1.15}$	$10.25_{12.13}$	$23.91_{25.16}$	$34.02_{23.90}$	$1.95_{0.77}$	$10.20_{20.87}$	$12.72_{19.21}$	$2.45_{2.06}$
	ANM	$32.80_{2.81}$	$44.33_{17.62}$	$21.88_{23.89}$	$31.33_{20.64}$	$13.17_{3.95}$	$27.56_{31.57}$	$15.04_{18.18}$	$2.71_{1.88}$
	DCM	$31.65_{0.27}$	$49.50_{36.83}$	$24.45_{33.31}$	$30.22_{24.83}$	$18.78_{6.01}$	$33.37_{36.14}$	$15.07_{22.37}$	$2.36_{2.08}$

counterfactuals, the algorithm does not present any guarantees of a correct counterfactual estimation because it does not model the exogenous variables of the SCM. That is the reason of the bad performance in couterfactual estimation.

On the other hand, let us justify some of the other paths where the errors of the Deconfounder are greater than unaware models. In Sachs' datasetto model the causal effect `Ekt→Akt`, the factorization model of the deconfounder uses `Raf, Mek, Jnk` and `P38` to extract the substitute confounder; the factorization model assumes that all those variables are independent conditioned to $\tilde{z}$, while that is not the case in the true SCM and, therefore, this SCM violates the independence assumption of (Wang & Blei, 2019). The same argument is valid for the paths `yceP→yfaD`, `lacA→yaeM`, `yceP→yfaD`, `ydeE→pspA` and `pspB→pspA`.

On the other hand, the paths `lacZ→yaeM`, `asnA→lacY` are frontdoor paths that Decaf can identify because it models the hidden confounder following the true causal graph. However, the Deconfounder is not designed to model this paths. To evaluate its performance for frontdoor paths, Deconfounder uses the same variables as Decaf to extract the substitute of the confounder. However, the Deconfounder assumes independence conditioned to the substitute confounder and that is not the case; therefore, we are violating the independence assumption again.

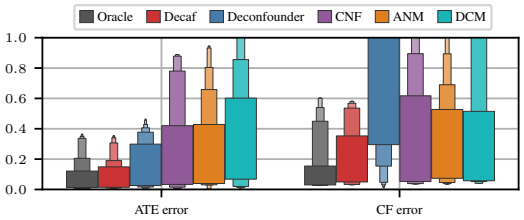


Figure 16: ATE and CF error evaluating only links where deconfounder should work in the additive case.

The only two paths that meet the Deconfounder assumptions in Fig. 1 are `lacA→lacY` and `yedE→pspB`. And we can observe that in those paths, the Deconfounder performs at least as well as unaware methods. On the other hand, all the factor models used for the Deconfounder implementation (PPCA, Deep exponential families and Variational autoencoder) assume additive noise. Therefore, interventional distributions in nonadditive settings are not computable theoretically with these models.

Table 4: Performance metrics on Ecoli70 dataset. Statistics computed an all samples over 5 runs, intervening and evaluating only in the causal effects that Deconfounder should solve. Bold indicates significantly better results (95% CI from a Mann-Whitney U test). Lower error values indicate better performance.

	Model	|ATE err| $\times 10^2$	|CF err| $\times 10^1$
Oracle	CNF	$8.31_{10.95}$	$1.49_{1.86}$
Aware	Decaf ☕	$\mathbf{9.18_{10.42}}$	$\mathbf{2.18_{2.02}}$
	Deconfounder	$14.35_{15.24}$	$12.03_{15.81}$
Unaware	CNF	$27.82_{30.17}$	$4.01_{3.62}$
	ANM	$27.63_{29.74}$	$3.64_{3.15}$
	DCM	$42.45_{54.23}$	$4.08_{4.12}$

B.3.2 METRICS ON THE OTHER PATHS

In this subsection we include a comparison between all the models in the *unconfounded* and the unidentifiable effects. For *unconfounded effects*, our expectation is to observe that all the CGMs achieve a performance comparable with the *oracle*. On the other hand, we expect to have higher errors in unidentifiable effects, since we do not have theoretical guarantees.

Unconfounded Effects. The results for *unconfounded effects* are summarized in Fig. 17 and Tab 5, considering only direct effects for ATE and counterfactual error computations. As expected, Decaf and CNF achieve metrics comparable to the *oracle* in both ATE and counterfactual estimations, particularly evident in Fig. 17, where error distributions are nearly identical. 5 does not show statistically significative differences between Decaf and CNF. Notably, architectures based on causal normalizing flows outperform ANM and DCM, which model each causal mechanism, f_i, with separate networks. This difference is crucial in settings with many variables and complex relations, where scalability is essential. Unlike ANM and DCM, which suffer from error propagation and limited scalability, causal normalizing flows leverage a single amortized model, making them more efficient in high-dimensional scenarios.

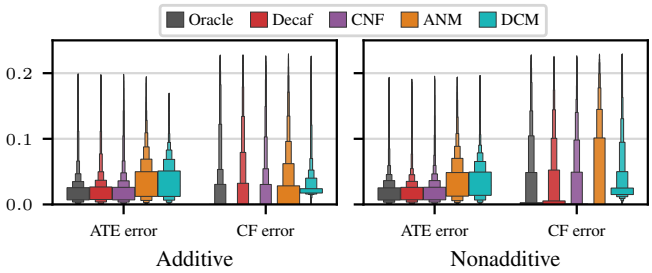


Figure 17: Error boxenplots on the Ecoli70 dataset for different CGMs, averaged over all *unconfounded* direct effects (see Fig. 1) after intervening in their 25th, 50th, and 75th percentiles and 5 random realizations of the experiment.

Finally, note that the Deconfounder has not been included in these metrics because it is not designed for *unconfounded queries* and there are many queries, while one Deconfounder model is needed for each query.

Table 5: Performance metrics on Ecoli70 dataset. Statistics computed on all *unconfounded* direct effects and 5 runs. Bold indicates significantly better results (95% CI from a Mann-Whitney U test). Lower error values indicate better performance.

		Additive			Nonadditive		
	Model	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$
Oracle	CNF	$3.72_{3.73}$	$2.00_{2.27}$	$1.27_{3.49}$	$1.94_{2.96}$	$1.92_{1.99}$	$1.76_{4.10}$
Aware	Decaf	$4.59_{5.58}$	$2.11_{2.39}$	$1.42_{3.81}$	$2.76_{7.61}$	$1.93_{1.87}$	$1.75_{4.04}$
Unaware	CNF	$4.77_{6.09}$	$2.02_{2.21}$	$1.22_{3.18}$	$2.97_{7.64}$	$1.95_{1.92}$	$1.71_{3.93}$
	ANM	$34.72_{8.56}$	$3.57_{3.02}$	$2.02_{4.09}$	$15.13_{12.57}$	$3.53_{3.15}$	$2.64_{5.34}$
	DCM	$36.23_{14.29}$	$3.48_{2.75}$	$2.69_{2.30}$	$21.22_{13.68}$	$3.42_{2.63}$	$3.00_{3.42}$

Unidentifiable Effects. The results for unidentifiable effects—causal queries that violate the assumptions in §6—are summarized in Fig. 18 and Tab 6. Notably, the *oracle* performs significantly better than the other CGMs. As seen in Fig. 18, error distributions are highly skewed, with ATE and counterfactual errors reaching extreme values—considering that metrics are computed on the standarized variables. Tab 6 shows no significant differences between the metrics achieved by Decaf and CNF.

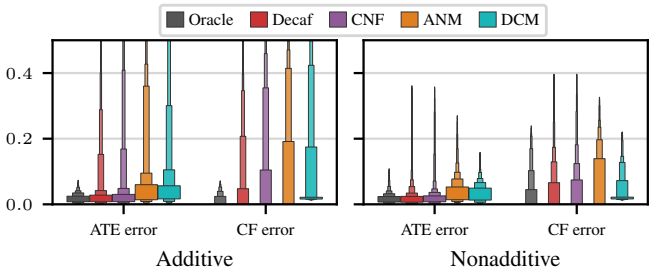


Figure 18: Error boxenplots on the Ecoli70 dataset for different CGMs, averaged over all unidentifiable direct effects (see Fig. 1) after intervening in their 25th, 50th, and 75th percentiles and 5 random realizations of the experiment.

B.4 Law school fairness use-case

The experiment with real-world data was inspired by Kusner et al. (2017) and Javaloy et al. (2023).

The purpose is to find a fair estimator of the decile that the grades of each student will occupy in their third year

Table 6: Performance metrics on Ecoli70 dataset. Statistics computed on all unidentifiable direct effects and 5 runs. Bold indicates significantly better results (95% CI from a Mann-Whitney U test). Lower error values indicate better performance

		Additive			Nonadditive		
	Model	MMD int $\times 10^4$	|ATE err| $\times 10^2$	|CF err| $\times 10^3$	MMD int $\times 10^5$	|ATE err| $\times 10^2$	|CF err| $\times 10^2$
Oracle	CNF	$3.71_{3.52}$	$1.79_{1.36}$	$5.88_{15.16}$	$16.98_{6.87}$	$1.75_{1.59}$	$1.62_{4.57}$
Aware	Decaf	$3.83_{3.93}$	$3.76_{7.90}$	$31.21_{92.38}$	$18.67_{5.64}$	$2.18_{3.73}$	$2.11_{6.13}$
Unaware	CNF	$4.54_{4.81}$	$4.75_{10.65}$	$44.76_{126.36}$	$20.22_{6.68}$	$2.32_{3.80}$	$2.13_{6.25}$
	ANM	$34.38_{5.17}$	$7.43_{12.64}$	$52.70_{137.99}$	$130.71_{41.64}$	$4.01_{3.82}$	$2.93_{7.21}$
	DCM	$35.49_{4.95}$	$7.67_{13.93}$	$67.46_{132.21}$	$198.23_{58.62}$	$3.43_{2.76}$	$3.29_{3.92}$

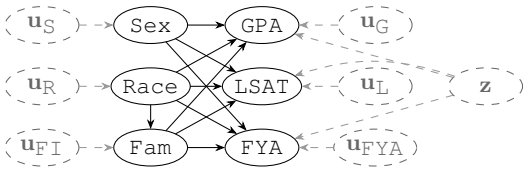


Figure 19: Confounded SCM modeled by Decaf.

of university.

The dataset contains information on $27,000$ law students who were admitted by the Law School Admissions Council (LSAC) from 1991 through 1997. We have performed an experiment similar to that carried out by Kusner et al. (2017), where race and sex were treated as sensitive attributes. We have considered the following variables to include in our study:

- `Race`: binary indicator of the race that distinguish between white and non-white.
- `Sex`: binary indicator of the sex that distinguish between male and female.
- `Fam`: family income.
- `LSAT`: the grade achieved in the Law School Admission Test (LSAT).
- `UGPA`: the undergraduate grade point average (GPA) of the student previous to the admission.
- `FYA`: first-year average grade.
- `Decile3`: the decile of the grades in the third year of university. This is the variable to predict.

For our purpose, we consider that an estimator, $\hat{\mathrm{y}}$, is fair if it meets *Demographic parity*, defined in (Kusner et al., 2017, Def. 3) as follows. A predictor $\hat{\mathrm{y}}$ satisfies demographic parity if the predicted distributions for different values of a sensitive attribute are equal: $p(\hat{\mathrm{y}} \mid \mathrm{t}=0) = p(\hat{\mathrm{y}} \mid \mathrm{t}=1)$. We evaluate the difference between predicted distributions using MMD—a lower distance between the predictions for the two groups of a sensitive attributes denotes a fairer predictor.

The assumed causal graph is slightly different from that of Kusner et al. (2017), since their purpose is to make a fair prediction `FYA` accounting only for `Race`, `Sex`, `LSAT` and `UGPA`. However, we include `Fam` and `FYA` as predictors and the task is to predict `Decile3` and the assumed causal graph is the one of Fig. 8.

Proposed fair predictor with Decaf. We propose to model the confounded SCM presented in Fig. 19, where are explicitly shown the exogenous variables, that are independent of the other variables of the graph except of their associated endogenous variable.

Afterwards, we predict the outcome, `Decile3` from the extracted latent variable that acts as substitute of the `knowledge` and the exogenous variables of `FYA` and `Fam`, following the causal graph of Fig. 8, using a generalized linear model: $\tilde{p}(\texttt{Decile3} \mid \mathbf{u}_{\texttt{FI}}, \mathbf{u}_{\texttt{FYA}}, \mathbf{z})$. Decaf models $\mathbf{z}$ and the exogenous variables as independent from `Race` and `Sex`. Therefore, the prediction of `Decile3` should be fair.

Baselines. The baselines used to compare our approach are the methods *Fair K* and *Fair add* proposed in Kusner et al. (2017).

Fair K is a fair predictor categorized in Level 2 in Kusner et al. (2017), which postulates that the student's knowledge, `know` affects `GPA`, `LSAT`, `FYA` and `Decile 3`, following the distributions described below.

$$\begin{aligned}
\texttt{Fam} &\sim \mathcal{N}\left(b_{Fam} + w^R_{Fam}\texttt{Race}, 1\right), \\
\texttt{GPA} &\sim \mathcal{N}\left(b_G + w^K_G\texttt{know} + w^R_G\texttt{Race} + w^S_G\texttt{Sex} + w^{Fam}_G\texttt{Fam}, \sigma^2_G\right), \\
\texttt{LSAT} &\sim \text{Poisson}\left(\exp(b_L + w^K_L\texttt{know} + w^R_L\texttt{Race} + w^S_L\texttt{Sex} + w^{Fam}_L\texttt{Fam})\right), \\
\texttt{FYA} &\sim \mathcal{N}\left(w^K_F\texttt{know} + w^R_F\texttt{Race} + w^S_F\texttt{Sex} + w^{Fam}_F\texttt{Fam}, 1\right), \\
\texttt{Decile3} &\sim \text{Poisson}\left(\exp(w^K_D\texttt{know} + w^R_D\texttt{Race} + w^S_D\texttt{Sex} + w^{Fam}_D\texttt{Fam})\right), \\
\texttt{know} &\sim \mathcal{N}(0, 1).
\end{aligned} \tag{50}$$

Then, the posterior distribution `know` is inferred using MonteCarlo with the probabilistic programming language Pyro (Bingham et al., 2019). The outcome is predicted using the inferred `know` using a generalized linear model: $\tilde{p}(\texttt{Decile3} \mid \texttt{know})$.

On the other hand, *Fair Add* predicts the outcome from the residuals of predicting each variable with each parent, which guarantees that these residuals are independents of `Race` and `Sex`. That is, the predictor estimates the distribution $p(\texttt{Decile3} \mid \mathbf{r}_{\texttt{Fam}}, \mathbf{r}_{\texttt{UGPA}}, \mathbf{r}_{\texttt{LSAT}}, \mathbf{r}_{\texttt{FYA}})$, where these residuals are computed as:

$$\begin{aligned}
\mathbf{r}_{\texttt{Fam}} &= \texttt{Fam} - \mathbb{E}[\texttt{Fam} \mid \texttt{Sex}, \texttt{Race}] \\
\mathbf{r}_{\texttt{UGPA}} &= \texttt{UGPA} - \mathbb{E}[\texttt{GPA} \mid \texttt{Sex}, \texttt{Race}, \texttt{Fam}] \\
\mathbf{r}_{\texttt{LSAT}} &= \texttt{LSAT} - \mathbb{E}[\texttt{LSAT} \mid \texttt{Sex}, \texttt{Race}, \texttt{Fam}] \\
\mathbf{r}_{\texttt{FYA}} &= \texttt{FYA} - \mathbb{E}[\texttt{FYA} \mid \texttt{Sex}, \texttt{Race}, \texttt{Fam}]
\end{aligned} \tag{51}$$

All predictors used are generalized linear models.

Discussion of Results. Although the *fair* methods proposed by Kusner et al. (2017) achieve significantly better *demographic parity* than our approach using Decaf (as indicated by a much lower MMD), their predictive performance is substantially inferior. Specifically, their performance is comparable to predicting the outcome using only the mean of the distribution, which serves as a baseline in Tab 1. In contrast, Decaf achieves a 98% reduction in MMD while incurring only an 11% increase in RMSE, as illustrated in Fig. 9.

These experiments demonstrate that leveraging Decaf to model confounded Structural Causal Models is beneficial beyond causal query estimation, leading to improved overall performance.

C Do-operator

We introduce here the algorithms that Decaf employ to generate interventional samples and counterfactuals. But first, we include those of Javaloy et al. (2023), since we leverage these CNFs as building blocks for Decaf. Note that the notation applied for Decaf is slightly different from the that used in the causal flows, naming the intervened variable as t, instead of x_i, in order to be consistent with the notation used in §3 and §6. However, note that both variables play the same role, and that $\text{t} \subset \mathbf{x}$.

C.1 Do-operator in causal normalizing flows

Algorithm 1 Algorithm to sample from the interventional distribution, $P(\mathbf{x} \mid \text{do}(\text{x}_i = \alpha))$. From Javaloy et al. (2023).

function SAMPLEINTERVENEDDIST(i, α)
  $\mathbf{u} \sim P_{\mathbf{u}}$ ▷ Sample a value from the observational distribution.
  $\mathbf{x} \leftarrow T_\theta^{-1}(\mathbf{u})$
  $\text{x}_i \leftarrow \alpha$ ▷ Set x_i to the intervened value α.
  $\text{u}_i \leftarrow T_\theta(\mathbf{x})_i$ ▷ Change the i-th value of $\mathbf{u}$.
  $\mathbf{x} \leftarrow T_\theta^{-1}(\mathbf{u})$
  return $\mathbf{x}$ ▷ Return the intervened sample.
end function

The computation of counterfactuals follows the steps of *abduction, action and prediction* postulated by (Pearl et al., 2016). The *abduction* step consists of using the observations to determine the value of the exogenous variables. Then, *action* is computing the intervention, modifiying the causal mechanism of the intervened variable and *prediction* consist of using the exogenous variables and the modified SCM to compute the counterfactual.

Algorithm 2 Algorithm to sample from the counterfactual distribution, $P(\mathbf{x}^{\text{cf}} \mid \text{do}(\text{x}_i = \alpha), \mathbf{x}^{\text{f}})$. From Javaloy et al. (2023).

function GETCOUNTERFACTUAL($\mathbf{x}^{\text{f}}, i, \alpha$)
  $\mathbf{u} \leftarrow T_\theta(\mathbf{x}^{\text{f}})$ ▷ **Abduction:** Get $\mathbf{u}$ from the factual sample.
  $\text{x}_i{}^{\text{f}} \leftarrow \alpha$ ▷ **Action:** Set x_i to the intervened value α.
  $\text{u}_i \leftarrow T_\theta(\mathbf{x}^{\text{f}})_i$ ▷ **Action:** Change the i-th value of $\mathbf{u}$.
  $\mathbf{x}^{\text{cf}} \leftarrow T_\theta^{-1}(\mathbf{u})$ ▷ **Prediction:** Get counterfactual
  return $\mathbf{x}^{\text{cf}}$ ▷ Return the counterfactual value.
end function

C.2 Do-operator in interventional distributions with Decaf

The sampling process consists of sampling first from the prior distribution of the latent variables and from the distribution of the exogenous variables. Then, one can use the generative network ($T_{\boldsymbol{\theta}}$) to take samples of the rest of variables, changing the components of $\mathbf{u}$ associated with t. Note that $\mathbf{z}$ is not the input of the normalizing flow, but a condition (or *context*). Therefore, $\mathbf{z}$ is transformed neither in the forward nor the reverse pass of the flow.

Algorithm 3 Algorithm to sample from the interventional distribution, $P(\mathbf{x} \mid \text{do}(\text{t} = \alpha))$ with Decaf.

function SAMPLEINTERVENEDDIST(t, α)
  $\mathbf{z} \sim P_{\mathbf{z}}$ ▷ Sample a value from the prior of $\mathbf{z}$.
  $\mathbf{u} \sim P_{\mathbf{u}}$ ▷ Sample a value from the observational distribution.
  $\mathbf{x} \leftarrow T_\theta^{-1}(\mathbf{u}, \mathbf{z})$
  $\text{t} \leftarrow \alpha$ ▷ Set t to the intervened value α.
  $\text{u}_\text{t} \leftarrow T_\theta(\mathbf{x}, \mathbf{z})_\text{t}$ ▷ Change the component of $\mathbf{u}$ associated with t.
  $\mathbf{x} \leftarrow T_\theta^{-1}(\mathbf{u}, \mathbf{z})$
  return $\mathbf{x}$ ▷ Return the intervened sample.
end function

Additionally, the process to compute the average treatment effect (ATE) involves to generate interventional distributions. For example, to compute the ATE comparing two interventions (α_1, α_2) in the variable t, we would generate samples of the interventional distributions, $p(\mathbf{x} \mid \text{do}(\text{t} = \alpha_1)), p(\mathbf{x} \mid \text{do}(\text{t} = \alpha_1))$, respectively, and approximate their expectations with MonteCarlo.

$$ATE = \mathbb{E}[\mathbf{x} \mid \text{do}(\text{t} = \alpha_2)] - \mathbb{E}[\mathbf{x} \mid \text{do}(\text{t} = \alpha_1)] \tag{52}$$

Unfortunately, if we were interested in evaluating the ATE on only one variable, y, the process would involve to generate samples of the whole interventional distribution and select only the samples of the interested variable.

C.3 Do-operator in counterfactuals with Decaf

As part of the abduction step, our model estimates the posterior distribution of hidden confounders given a factual datapoint, $q_\phi(\mathbf{z} \mid \mathbf{x}^{\text{f}})$. Therefore, the counterfactual given by the model is no longer a single point but comes from a distribution. To obtain a single sample that allows us to compare it with the true counterfactual, we estimate the mean of the posterior distribution from samples using MonteCarlo, and we use that value to generate a counterfactual.

Algorithm 4 Algorithm to sample from the interventional distribution, $P(\mathbf{x} \mid \text{do}(\text{t} = \alpha))$ with Decaf.

function GETCOUNTERFACTUAL($\mathbf{x}^{\text{f}}, \text{t}, \alpha$)
  $q_\phi(\mathbf{z} \mid \mathbf{x}^{\text{f}}) \leftarrow$ Deconfounding network($\mathbf{x}^{\text{f}}$) ▷ **Abduction:** Get $\mathbf{z}$ from the factual sample.
  $\mathbf{z} \sim \mathbb{E}_{q_\phi}[\mathbf{z} \mid \mathbf{x}^{\text{f}}]$ ▷**Abduction**: Estimate the mean of the distribution.
  $\mathbf{u} \leftarrow T_\theta(\mathbf{x}^{\text{f}}, \mathbf{z})$ ▷**Abduction:** Get $\mathbf{u}$ from the factual sample.
  $\text{t}^{\text{f}} \leftarrow \alpha$ ▷**Action:** Set t to the intervened value α.
  $\text{u}_{\text{t}} \leftarrow T_\theta(\mathbf{x}^{\text{f}}, \mathbf{z})_{\text{t}}$ ▷**Action:** Change the component of $\mathbf{u}$ associated with t.
  $\mathbf{x}^{\text{cf}} \leftarrow T_\theta^{-1}(\mathbf{u}, \mathbf{z})$ ▷ **Prediction:** compute the counterfactual
  return $\mathbf{x}^{\text{cf}}$ ▷ Return the counterfactual value.
end function

D Additional details on related work of causal inference with hidden confounders

In this section we go deeper into the methods of causal inference in scenarios where there are unobserved confounders.

First of all, we want to remark that all the following methods have been designed to address causal inferences in specific causal graphs (or subgraphs), therefore they can be used when there exists the causal relationships presented in Fig. 20.

In the following text, we assume the notation introduced in §3, where $\mathbf{z}$ is the hidden confounder, t is the intervened variable or treatment and y is the outcome, i.e. the variable where we want to evaluate the causal effects.

We have classified the different approaches depending on the graph that they are designed to address. However, there are two considerations that are common for all these approaches.

First, the methods follow a two-stage process: **i)** extracting a substitute for the unobserved confounder, $\tilde{\mathbf{z}}$, using the variables affected by the confounder or instrumental variables, and **ii)** estimating the outcome given this substitute, $\tilde{\text{y}} \sim p(\text{y} \mid \tilde{\mathbf{z}}, \text{t})$. In larger causal graphs, one predictor must be trained for each outcome, and one extractor must be trained per independent confounder.

Second, none of these methods shows the ability of identify *counterfactuals*, since they do not model exogenous variables.

Presence of null proxies independent of t **(Fig. 20a).** We say $\mathbf{n}$ to be a null proxy of $\mathbf{z}$ if it is a child of $\mathbf{z}$ independent of the outcome, y, given $\mathbf{z}$: $\mathbf{n} \perp\!\!\!\perp \text{y} \mid \mathbf{z}$. Methods for estimating causal effects were developed when null proxies of the confounder were available and those proxies are independent of the intervened variable: $\mathbf{n} \perp\!\!\!\perp \text{t} \mid \mathbf{z}$. We can use these proxies to infer a substitute. Among these, Allman et al. (2009); Kuroki & Pearl (2014) studies the case in which the confounder is categorical and uses matrix factorization to extract a substitute when there are at least three Gaussian proxies (Allman et al., 2009), when the conditional distribution of the confounder given the proxy is known or when other proxies are available (Kuroki & Pearl, 2014). Kallus et al. (2018) also employ matrix factorization to cases where the confounder is continuous and the relation with the covariates and the treatment (but not with the outcome) is linear. In addition, Kallus et al. (2019) uses kernel functions to extract the substitute confounder when the generators are nonlinear. The most relevant method based on deep generative methods is proposed by Louizos et al. (2017), consisting of a VAE to extract the substitute confounder when several null proxies are available, although there is no theoretical guarantee of its operation. Finally, Miao et al. (2023) offers a regression-based approach to estimate the unobserved confounder under *equivalence*, which assumes that any model of the joint achieves element-wise transformations of the latents, which is not feasible to check: $\tilde{p}(\text{t}, \mathbf{z} \mid \mathbf{n}) = p(\text{t}, V(\mathbf{z}) \mid \mathbf{n})$. The graph in which all these methods operate can be found in Fig. 20a.

Presence of two proxies: null and not null (Fig. 20b). When the null proxies affect treatment (see Fig. 20b: the proxy, $\mathbf{n}$, affects treatment t), Miao et al. (2018) offers theoretic guarantees of causal identifiability in the presence of another proxy, $\mathbf{w}$, and completeness conditions. The proxy $\mathbf{w}$ can be active, that is, it can directly affect y. Practically, in Tchetgen et al. (2020) the two-stage proximal least squares (P2SLS) we can find the method to infer the substitute confounder from $p(\mathbf{w} \mid \text{t}, \mathbf{n})$. P2SLS can be implemented using neural networks to achieve greater flexibility.

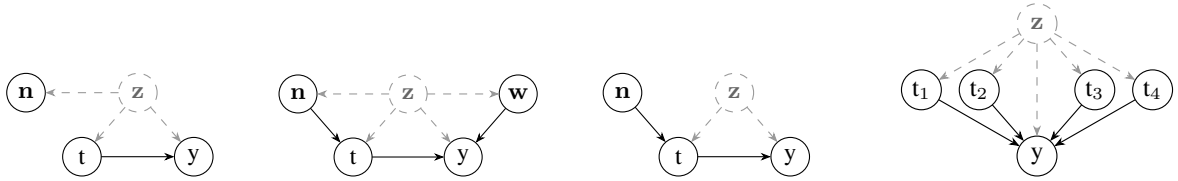
Instrumental variable (Fig. 20c). Another condition that allows causal inference is the presence of instrumental variables (IVs), i.e. variables that affect only the treatment and are independent of both the unobserved confounder and the outcome given the treatment (in Fig. 20c, $\mathbf{n}$ is an IV). In linear DGP, Pearl (2009); Angrist & Pischke (2009) demonstrates that a two-stage regression process mitigates the confounding bias as the only effect that flows from the instrumental variable to the outcome is through treatment. A substitute of the confounder is extracted by computing the conditional distribution of the treatment given the instrumental variable: $\tilde{\mathbf{z}} \sim p(\text{t} \mid \mathbf{n})$. Furthermore, (Hartford et al., 2017) develops an extension of this theory to include arbitrarily complex nonlinear DGP, designing a two-step deep approach, based on neural networks.

Multitreatment affected by a common confounder (Fig. 20d). Finally, the multitreatment scenario (Fig. 20d) has been studied by Wang & Blei (2019); Ranganath & Perotte (2018). It is called multitreatment because all covariates can be seen as a treatment over the outcome, y. It is assumed that, in the true DGP, there exist several covariates that are independent given the unobserved confounder. Therefore Wang & Blei (2019) propose to use a factorization model, such as probabilistic PCA or Poisson Matrix Factorization, to infer the substitute confounder. A factorization model assumes that the distribution of all the treatments factorizes in the following way: $p(\mathbf{t}, \mathbf{z}) = p(\mathbf{z}) \prod_{i=1}^{d} p(\mathrm{t}_i \mid \mathbf{z})$, which should allow to construct a substitute of the confounder from the posterior of $\mathbf{z}$: $\tilde{\mathbf{z}} \sim \tilde{p}(\tilde{\mathbf{z}} \mid \mathbf{t})$. However, these method only offers identifiaibility in the asymptotic setting where the number of treatments is infinite. On the other hand Ranganath & Perotte (2018) proposes a method that uses a VAE as a factorization model, adding a regularization term to reduce the additional mutual information between the estimated confounder and treatment t_j given the rest of treatments $\mathbf{t}_{-j}$. Again, the theoretical guarantees of this approach need an infinite number of treatments to achieve unbiased estimates of causal effects.

Wang & Blei (2021) connect the ideas of Miao et al. (2018) and Wang & Blei (2019) ensuring causal identification in the multitreatment setting when it is known that some of the treatments *can act as null proxies*, that is, they do not affect the outcome. This assumption allows them to provide theoretical guarantees when the number of treatments does not tend to be infinite.

How is Deconfounder Wang & Blei (2019; 2021) related to our work. As Decaf does, Deconfounder infers the posterior distribution of the substitute of the confounder from the observational data using a generative model. However, the application of a factorization model restricts the structural dependencies that we can model. For example, the Deconfounder cannot model the structural dependencies of Fig. 20b, since the factorization model assumes $\mathbf{n} \perp\!\!\!\perp \mathrm{t} \perp\!\!\!\perp \mathbf{w} \mid \mathbf{z}$. In contrast, the Decaf uses a causal flow, which does allow this dependencies because the causal graph is encoded in the flow.

We also stress that Decaf models the whole confounded SCM, including the exogenous variables. This allows to compute *counterfactuals* and train in a query-agnostic manner. In contrast, Deconfounder cannot compute counterfactuals and needs of a separate model per query.



(a) One proxy. (b) Two proxies. (c) Instrumental variable. (d) Multitreatment.

Figure 20: *Ad-hoc* graphs. **(a)** Kuroki & Pearl (2014); Louizos et al. (2017); Miao et al. (2023); Kallus et al. (2018; 2019); Allman et al. (2009) address the case where $\mathbf{n}$ is independent of t. **(b)** Miao et al. (2018) is designed for the case where there exist two proxies. **(c)** Graph with an instrumental variable, but this graph is out of the scope of our framework. **(d)** Wang & Blei (2019; 2021); Ranganath & Perotte (2018) are designed for the multitreatment setting.

E Algorithms for causal query identification

As explained in §6.3, we can ask Decaf to solve any causal query, but we do not have the guarantee that the estimation that Decaf returns is correct unless the query is identifiable. Therefore, we provide the practitioner with algorithms to check the identifiability of causal queries.

Specific treatment-outcome pair. We start presenting the Alg. 5 to identify a causal query specifying the pair treatment and outcome, which is valid for estimating the interventional distribution of the outcome—$p(\mathrm{y} \mid \mathrm{do}(\mathrm{t}), \mathbf{c})$—and the counterfactual—$p(\mathrm{y}^{\mathrm{cf}} \mid \mathrm{do}(\mathrm{t}), \mathbf{x}^{\mathrm{f}})$—, since we postulated in §6 that the latter is identifiable if the former is.

We have employed this algorithm in all the paths of Sachs and Ecoli70 datasets to check the identifiability of all the direct causal effects—where y is a child of t—, in order to get a visual representation of the identifiable queries of a complex graph. However, due to the large number of possible causal queries resulting from all edge combinations in the 43-node Ecoli70 dataset, we have not analyzed identifiability for all indirected queries.

Trivially, if one is interested in evaluating a query which involves several outcomes, $\{\mathrm{y}_1, \mathrm{y}_2, ..., \mathrm{y}_O\}$, one causal query per y_i should be evaluated.

Algorithm 5 Identification of causal queries that include intervention and outcome (t, y)

Require: Graph $\mathcal{G}$, intervention variable t, outcome variable y, covariates $\mathbf{c}$, hidden variables $\mathbf{z}$
Ensure: Boolean indicating if query is identifiable
$\mathbf{z} \leftarrow$ hidden variables that are parents of both t and y
return True **if** $\mathbf{z}$ is $\emptyset$ ▷ Unconfounded is identifiable
for all $\mathbf{z}_k \in \mathbf{z}$ **do**
  Comment: Each $\mathbf{z}_k$ is an independent component of $\mathbf{z}$
  $\mathbf{n}$-proxies $\leftarrow$ children of $\mathbf{z}_k$ d-separated from t given $(\mathbf{z}, \mathbf{c})$
  $\mathbf{w}$-proxies $\leftarrow$ children of $\mathbf{z}_k$ d-separated from y given $(\mathbf{z}, \mathbf{c})$
  if there exist $\mathbf{n} \in \mathbf{n}$-proxies and $\mathbf{w} \in \mathbf{w}$-proxies such that $\mathbf{n}$ is d-separated from $\mathbf{w}$ given $(\mathbf{z}, \mathbf{c})$ **then**
    $\mathbf{z}_k$ is deconfounded
  end if
end for
return all $\mathbf{z}_k$ are deconfounded

Evaluation on all the variables. Although the Alg. 6 consist of applying Alg. 5 iteratively, we also find it interesting to include the extension to identify causal queries evaluated on all variables in the dataset, which is useful for using Decaf as a generative model for the interventional distribution—$p(\mathbf{x} \mid \text{do}(\text{t}))$—, or offering complete counterfactual samples—$p(\mathbf{x}^{\text{cf}} \mid \text{do}(\text{t}), \mathbf{x}^{\text{f}})$—intervening in a specific variable, $\text{t} \subset \mathbf{x}$.

Algorithm 6 Identification of causal queries, intervening in t and evaluating in all variables

Require: Graph $\mathcal{G}$, intervention variable t, hidden variables $\mathbf{z}$
Ensure: Boolean indicating if the interventional distribution is identifiable
$\mathbf{z} \leftarrow$ hidden variables that are parents of t
for all $\text{x}_i \in$ descendants of t **do**
  Comment: Evaluate only on descendants of the intervention
  Check (t, x_i) identifiability with Alg. 5
end for
return all $(\text{t}, \mathbf{x}_i)$ are identifiable

E.1 Pipeline for using Decaf

Our framework provides a systematic approach to solving causal queries by integrating Decaf, a model trained on observational data, with algorithms designed for query identifiability analysis.

As depicted in the pipeline, the framework takes as input a dataset $\mathcal{D}$, a causal graph $\mathcal{G}$, and a set of N interesting queries $\{Q_i\}_{i=1}^N$. The process begins by training Decaf on $\mathcal{D}$ and $\mathcal{G}$, enabling it to learn the confounded SCM, $\mathcal{M}$.

Simultaneously, the identifiability of each causal query Q_i is assessed using dedicated algorithms (Alg. 5 and Alg. 6). If Q_i is identifiable, the trained Decaf is used to estimate $Q_i(\mathcal{M})$ (Alg. 3 and Alg. 4), yielding the estimated causal effect $\hat{Q}_i(\mathcal{M})$. If Q_i is not identifiable, the framework indicates that answering the query is not feasible given the available data and causal structure. Other causal queries can be answered by the model without retraining, provided that their identifiability is verified beforehand.

This workflow ensures a principled approach to causal inference, leveraging both data-driven modelling and theoretical guarantees on identifiability. Both the Decaf model and the algorithms for query identifiability and estimation will be included in the code that we will provide upon acceptance.

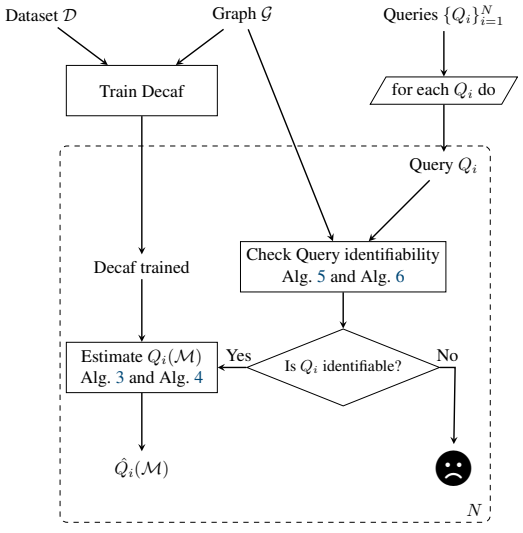


Figure 21: **Block diagram of the pipeline.**

Validation with interventional data. As a final step in the pipeline in real-world scenarios, especially in sensitive applications, we encourage practitioners to validate the framework with interventional data. Causal queries such as *average treatment effects* (ATEs) can be validated if a randomized experiment is available in which interventions are carried out on the treatment variable.

However, in cases where experiments on the required variable are not available, our framework can still be partially validated by assessing the completeness of the inferred hidden confounder given the observed proxies. This can be done by evaluating causal effects in another causal query that shares the same hidden confounder. Specifically, if a causal query Q_1 lacks interventional data, but another query Q_2 involving the same hidden confounder is identifiable, the inferred confounder from Q_2 can be postulated as a valid substitute for estimating Q_1. This indirect validation method provides a way to assess the reliability of our framework without requiring direct interventions for every confounded query.