Counterfactual Identifiability of Bijective Causal Models

Arash Nasr-Esfahany [1] **Mohammad Alizadeh** [1] **Devavrat Shah** [1]

Abstract

We study counterfactual identifiability in causal models with bijective generation mechanisms (BGM), a class that generalizes several widely-used causal models in the literature. We establish their counterfactual identifiability for three common causal structures with unobserved confounding, and propose a practical learning method that casts learning a BGM as structured generative modeling. Learned BGMs enable efficient counterfactual estimation and can be obtained using a variety of deep conditional generative models. We evaluate our techniques in a visual task and demonstrate its application in a real-world video streaming simulation task.

1. Introduction

Had Cleopatra's nose been shorter, the whole face of the world would have changed.[2] *(Blaise Pascal, 1669)*

The *ladder of causation* presented by Pearl (2018) consists of three distinct layers encoding different types of concepts: *associational* ($\mathcal{L}_1$), *interventional* ($\mathcal{L}_2$), and *counterfactual* ($\mathcal{L}_3$), roughly corresponding to *seeing*, *doing*, and *imagining*, respectively. $\mathcal{L}_1$ deals with passively observed factual information, for instance, the probability of recovery in patients who take Aspirin. $\mathcal{L}_2$ deals with active interventions or the effect of actions, for example, what percentage of patients would recover if we give them Aspirin? $\mathcal{L}_3$ deals with alternative ways the world could have been including ways that might conflict with how the world currently is, e.g., if the patient took Aspirin and was cured, would the headache still be gone had they not taken Aspirin? The three levels are distinct in the sense that it is generally not possible to uniquely answer higher level queries from lower level information (Bareinboim et al., 2022, Thm. 27). In particular, although we can determine $\mathcal{L}_2$ information by conducting experiments and actively intervening in the world, we cannot answer $\mathcal{L}_3$ questions in general, even with experiments. In other words, they may be *non-identifiable* (Ibeling & Icard, 2020), or underspecified (D'Amour et al., 2020).

Nevertheless, counterfactual queries of the form "*why–?*" and "*what if–?*" are useful in defining fundamental concepts such as harm (Richens et al., 2022), fairness (Kusner et al., 2017; Zhang & Bareinboim, 2018), credit (Mesnard et al., 2021), regret, and blame. They also have applications in engineering, e.g., root cause analysis (Gan et al., 2022; Budhathoki et al., 2022), trace-driven simulation (Alomar et al., 2023; Bothra et al., 2022), and sample efficient reinforcement learning (Lu et al., 2020; Agarwal et al., 2021).

In this paper, we introduce a new class of causal models called *Bijective Generation Mechanisms (BGM)*. BGMs subsume several model classes studied in the literature, e.g., Nonlinear Additive Noise Models (ANM) (Hoyer et al., 2008), Location Scale Noise Models (LSNM) (Immer et al., 2022), and post-nonlinear causal models (PNL) (Zhang & Hyvärinen, 2009) (§4). We establish counterfactual identifiability of BGMs for three meaningful causal structures (§5), assuming a coarse knowledge of the underlying system in the form of a causal diagram.

Our identifiability results specify tractable criteria for finding the underlying BGM (up to equivalence, see Definition 6.1), enabling counterfactual estimation. We cast the search for a BGM that satisfies the specified criteria as density estimation with *structured generative networks* (§6), which has been widely studied in the learning literature (Kingma & Welling, 2013; Goodfellow et al., 2014; Dinh et al., 2016; Arjovsky et al., 2017; Song & Ermon, 2019; Ho et al., 2020).

Once the underlying BGM is learned, we use it for efficient counterfactual estimation with guarantees (§7). This is in contrast to symbolic counterfactual identification methods (Shpitser & Pearl, 2007; Correa et al., 2021) which express identifiable counterfactual queries in terms of interventional and observational distributions, statements that may not be computationally tractable. Furthermore, our work advances the growing literature on applying machine learning methods for counterfactual estimation. Unlike most prior work in

[1]Department of Electrical Engineering and Computer Science (EECS), Massachusetts Institute of Technology (MIT), Cambridge, MA 02139. Correspondence to: Arash Nasr-Esfahany <arashne@mit.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[2]Taken from Pearl & Mackenzie (2018, Ch. 8)

this space that foregoes identifiability analysis (Pawlowski et al., 2020; Sanchez & Tsaftaris, 2021), except in certain restricted settings (e.g., discrete domains (Shalit et al., 2017; Oberst & Sontag, 2019) or absence of unobserved confounders (Khemakhem et al., 2021; Sanchez-Martin et al., 2021)), our results support continuous variables, allow unobserved confounders, and extend to a more general class of generation mechanisms.

We validate our techniques using a visual task (§8.1) and demonstrate their application to a real-world system simulation task (§8.2). Our code is available in github.com/counterfactual-BGM/cf2. This work does not raise any negative societal impacts.

2. Preliminaries

We provide a brief background on the (mostly) causal concepts that we use in this work. Refer to Pearl (2009a), Pearl (2009b), and Peters et al. (2017) for more details.

Notation: We refer to a random variable with a capital letter, e.g., X, the value it obtains with a lowercase letter, e.g., x, its Probability Density Function (PDF) with P_X, and a set of random variables with boldface font, e.g., $\boldsymbol{X} = \{X_1, \dots, X_n\}$. $J_{f(\cdot)}$ denotes the Jacobian of $f(\cdot)$.

SCM and Causal Diagram: We use the framework of Structural Causal Models (SCMs) (Pearl, 2009a, Ch. 7). An SCM $\mathcal{M}$ consists of endogenous variables $\boldsymbol{V}$ determined by other variables in the model, exogenous (also called latent or background) variables $\boldsymbol{U}$ with distribution $P_{\boldsymbol{U}}(\cdot)$ determined by factors outside the model (one exogenous variable corresponding to each endogenous variable), and generation mechanisms $\mathcal{F}$. $\mathcal{F}$ contains a function f_i for each variable V_i that maps endogenous parents $\text{Pa}(V_i)$ and exogenous variable U_i to V_i. The entire $\mathcal{F}$ defines a structured mapping from $\boldsymbol{U}$ to $\boldsymbol{V}$. The prior distribution over exogenous variables, $P_{\boldsymbol{U}}(\cdot)$, together with the generation mechanisms ($\mathcal{F}$) entail an observational ($\mathcal{L}_1$) distribution over endogenous variables which we refer to as $P^{\mathcal{M}}_{\boldsymbol{V}}(\cdot)$. Each $\mathcal{M}$ induces a causal diagram $\mathcal{G}$, where every $V_i \in \boldsymbol{V}$ is a vertex, there is a directed arrow $(V_j \to V_i)$ for every $V_i \in \boldsymbol{V}$ and $V_j \in \text{Pa}(V_i)$, and there is a dashed-bidirected arrow $(V_j \dashleftarrow\dashrightarrow V_i)$ for every $V_i, V_j \in \boldsymbol{V}$ such that U_i and U_j are not independent. The bidirected arrows represent existence of unobserved confounders. A causal model satisfies the *Markovian* assumption if for every $V_i, V_j \in \boldsymbol{V}$, the corresponding U_i and U_j are independent. In other words, no bidirected arrow exists in its induced causal diagram ($\mathcal{G}$).

Interventions and the *do*-Operator: Given an SCM $\mathcal{M}$, and a set of endogenous variables $\boldsymbol{X} \subseteq \boldsymbol{V}$, an intervention $do(\boldsymbol{X} := \boldsymbol{x})$ corresponds to replacing the generation mechanisms $\mathcal{F}$ with $\mathcal{F}_{\boldsymbol{x}} = \{f_i : V_i \notin \boldsymbol{X}\} \cup \{\boldsymbol{X} := \boldsymbol{x}\}$. In words, $\mathcal{F}_{\boldsymbol{x}}$ is formed by deleting from $\mathcal{F}$ all functions f_i corresponding to members of set $\boldsymbol{X}$ and replacing them with the set of constant functions $\{\boldsymbol{X} := \boldsymbol{x}\}$. We refer to the altered SCM by $\mathcal{M}_{\boldsymbol{x}}$, and the interventional ($\mathcal{L}_2$) distribution of endogenous variables by $P^{\mathcal{M}_{\boldsymbol{x}}}_{\boldsymbol{V}}(\cdot)$ or in short $P_{\boldsymbol{V}_{\boldsymbol{x}}}(\cdot)$. The interventional (or experimental) distribution helps us analyze the effect of taking actions, i.e., what would happen if we apply an intervention?

Counterfactuals: Given an SCM $\mathcal{M}$, two sets of endogenous variables $\boldsymbol{X}, \boldsymbol{E} \subseteq \boldsymbol{V}$, observed realizations $\boldsymbol{e}$ (evidence) for $\boldsymbol{E}$, and an intervention $do(\boldsymbol{X} := \boldsymbol{x})$, the counterfactual ($\mathcal{L}_3$) distribution $P^{\mathcal{M}_{\boldsymbol{x}}}_{\boldsymbol{V}}(\cdot|\boldsymbol{E} = \boldsymbol{e})$ or in short $P_{\boldsymbol{V}_{\boldsymbol{x}}}(\cdot|\boldsymbol{E} = \boldsymbol{e})$ corresponds to the distribution entailed by $\mathcal{M}_{\boldsymbol{x}}$ using the posterior distribution $P_{\boldsymbol{U}|\boldsymbol{E}}(\cdot|\boldsymbol{e})$ over the exogenous variables. In case of deterministic counterfactuals (δ-distribution), we refer to them as $\boldsymbol{V}_{\boldsymbol{x}}|\boldsymbol{E} = \boldsymbol{e}$. Pearl (2009a, Ch. 7) proposes the following three-step procedure for counterfactual estimation. i) *Abduction:* Update $P_{\boldsymbol{U}}$ with $\boldsymbol{e}$ to obtain $P_{\boldsymbol{U}|\boldsymbol{E}}$. ii) *Action:* Update the SCM $\mathcal{M}$ to $\mathcal{M}_{\boldsymbol{x}}$. iii) *Prediction:* Use the updated distribution of exogenous variables and the updated SCM to estimate the counterfacual distribution. For generalizations of this definition, e.g., to nested counterfactuals, refer to Correa et al. (2021). Counterfactual distributions allow us to imagine hypothetical worlds where everything is fixed other than interventions.

Identifiability: Evaluating causal queries given a partial state of knowledge is a subtle problem. An $\mathcal{L}_i$ query is identifiable using $\mathcal{L}_j$ information if its answer can be expressed purely based on $\mathcal{L}_j$ distributions ($i, j \in \{1, 2, 3\}$). Formally, let $Q(\mathcal{M})$ be an $\mathcal{L}_i$ query of an SCM $\mathcal{M}$. In a class $\mathbb{M}$ of SCMs, Q is identifiable if for any pair of SCMs $\mathcal{M}_1$ and $\mathcal{M}_2$ from $\mathbb{M}$, $Q(\mathcal{M}_1) = Q(\mathcal{M}_2)$ whenever $\mathcal{M}_1$ and $\mathcal{M}_2$ match in all $\mathcal{L}_j$ queries (Pearl, 2009a, def. 3.2.3). This is always true if $1 \leq i \leq j \leq 3$. Characterizing conditions where this holds for $1 \leq j < i \leq 3$ has been subject to extensive research efforts (Spirtes et al., 2001).

3. Related Work

Interventional ($\mathcal{L}_2$) Identification and Estimation: Assuming a coarse knowledge of the underlying system in the form of a causal DAG, identification of interventional queries has been extensively studied in the literature (Pearl, 2009a, Ch. 3), including *do-calculus* (Pearl, 1995a) as a general solution. In cases where the full causal diagram is not accessible, another line of work focuses on its (partial) identification using observational data. Although this sounds compelling and amenable to data-driven and ML research practices, it is known that the causal diagram can be identified from observational data only up to its Markovian equivalence class (Spirtes et al., 2001). Investigating identifiability using only equivalence classes has thus received research attention (Zhang, 2008; Perković et al., 2018; Jaber et al.,

2019). Witty et al. (2021) investigates a simulation-based notion of identifiability with probabilistic programs (Goodman et al., 2008), using priors over parametric representations of SCMs. Once identifability of the interventional query is established, various methods exist for estimation of the causal effect, including Propensity Score for the backdoor case (Rubin, 1978; Kennedy, 2019), and other statistical methods for more relaxed settings (Fulcher et al., 2020; Jung et al., 2020). In cases where the interventional query is non-identifiable, *partial identification* methods estimate meaningful bounds (Manski, 1990; Balke & Pearl, 1997; Zhang & Bareinboim, 2021; Li & Pearl, 2022a)

Counterfactual $(\mathcal{L}_3)$ **Identification**: Using the *counterfactual graph*, Shpitser & Pearl (2007) proposes an algorithm that determines identifiability of counterfactual queries from interventional data. This was extended in Correa et al. (2021) by providing sufficient and necessary graphical conditions for identification of (nested) counterfactuals from a collection of observational and interventional distributions, given the causal diagram. In identifiable cases, these algorithms express the counterfactual query as a combination of observational and interventional quantities. However, estimating this expression may not be tractable. Shah et al. (2022) proves identifiability of counterfactuals for a specific causal diagram and provides an algorithm for their tractable estimation, assuming the joint distribution of exogenous and endogenous variables is an exponential family. In contrast, we do not restrict the joint distribution of variables.

Similar to the interventional case, partial identification methods have been proposed for non-identifiable counterfactual queries (Balke & Pearl, 1994; Tian & Pearl, 2000; Finkelstein & Shpitser, 2020; Zhang et al., 2022; Gresele et al., 2022). Furthermore, identification of specific counterfactual queries has been studied in isolation, e.g., the *effect of treatment on the treated* (Shpitser & Pearl, 2009), *path-specific effects* (Shpitser & Sherman, 2018; Zhang & Bareinboim, 2018), and *probabilities of causation* (Pearl, 1999).

Neural Methods for Causal Estimation[3]: There is a growing literature on applying neural methods for efficient estimation of causal queries. An extensive line of work focuses on estimating interventional $(\mathcal{L}_2)$ queries including Kocaoglu et al. (2018); Xia et al. (2021); Zečević et al. (2021), to name a few. However, we focus on estimating counterfactuals $(\mathcal{L}_3)$.

On the counterfactual side, a line of work uses ML techniques for estimation without any guarantees about the identifiability of the counterfactual query (Pawlowski et al., 2020; Khemakhem et al., 2021; Sanchez-Martin et al., 2021; Sanchez & Tsaftaris, 2021; Geffner et al., 2022). In contrast, we prove identifiability for several causal structures in §5. Furthermore, most existing techniques make restrictive assumptions about the causal structure or generation mechanisms. For example, Johansson et al. (2016); Shalit et al. (2017); Yao et al. (2018); Oberst & Sontag (2019); Lorberbom et al. (2021); Xia et al. (2022) consider counterfactual estimation in domains with discrete (finite-valued) endogenous variables. Several others works assume a Markovian causal structure (Pawlowski et al., 2020; Khemakhem et al., 2021; Sanchez-Martin et al., 2021; Geffner et al., 2022), i.e., the absence of unobserved confounders (bidirected edges in the causal diagram). Hartford et al. (2017); Khemakhem et al. (2021); Geffner et al. (2022) restrict the class of generation mechanisms to Nonlinear Additive Noise Models (ANM) (Hoyer et al., 2008) or Location Scale Noise Models (LSNM) (Strobl & Lasko, 2022). The models we consider are more general than these prior papers. They support continuous endogenous variables, allow unobserved confounders, and bijective generation mechanisms that include nonlinear ANM, LSNM, and post-nonlinear causal model (PNL) (Zhang & Hyvärinen, 2009) as special cases.

Disentanglement: Independent Component Analysis (ICA) (Hyvärinen & Oja, 2000) concerns the problem of recovering statistically independent source signals $\boldsymbol{S} = (S_1, \dots, S_n)$ from their observed mixtures $\boldsymbol{X} = (X_1, \dots, X_n)$, where $\boldsymbol{X} = f(\boldsymbol{S})$, and the unknown f (mixing function) is assumed to be invertible. Unlike linear mixing functions that make this problem identifiable with certain conditions, it is known to be non-identifiable for nonlinear mixing functions (Hyvärinen & Pajunen, 1999). Recently, Hyvarinen et al. (2019) proved identifiability in the presence of auxiliary variables, which make the sources conditionally independent. This has been exploited for disentangling semantically meaningful features of high-dimensional data (Locatello et al., 2019; Khemakhem et al., 2020).

We took inspiration from the non-linear ICA framework, especially for our development in §5.3. However, the problem we consider is fundamentally different: We are interested in disentangling the total contribution of the unknown sources of variation from known attributes, as opposed to disentangling the effect of individual unknown sources. A recent work (Shaham et al., 2022) considers disentangling latent variables from observed attributes, and proves that the reconstructed sources $\hat{\boldsymbol{S}}$ have the same entropy as the true sources, in distribution. However, we are interested in pointwise guarantees, i.e., $\hat{\boldsymbol{s}} = g(\boldsymbol{s})$ for some invertible function $g(\cdot)$, which are stronger than the distributional properties like $\mathcal{H}(\hat{\boldsymbol{S}}) = \mathcal{H}(\boldsymbol{S})$.

4. Problem Formulation

We assume knowledge of the causal diagram $\mathcal{G}$ that might include unobserved confounders (no Markovianity assumption), and access to observational $(\mathcal{L}_1)$ data distribution. We

[3] Appendix A provides a more detailed survey.

(a) Non-Markovian (b) Markovian

Figure 1. Causal diagram $\mathcal{G}$'s sub-graph related to generation of V

(a) Instrumental Variable (b) Backdoor Criterion

Figure 2. Instrumental Variable and Backdoor Criterion examples.

are interested in learning the data generation mechanisms $\mathcal{F}$, which can be further used for counterfactual $(\mathcal{L}_3)$ estimation. As demonstrated by Pearl's Causal Hierarchy Theorem (Bareinboim et al., 2022, Thm. 27), cross-layer inference is not possible in general settings, and we need to make further assumptions.

Bijective Generation Mechanism (BGM): As mentioned in §2, each endogenous variable V_i is generated in the following way in the SCM framework:

$$V_i \coloneqq f_i\Big(\text{Pa}(V_i), U_i\Big) \tag{1}$$

We assume that the function f_i is a bijective mapping from U_i to V_i, for each realization of $\text{Pa}(V_i)$, hence the name *Bijective Generation Mechanism (BGM)*. In other words, no information is lost in transformation from exogenous to endogenous variables. We refer to the inverse of the generation mechanism as $f_i^{-1}(\text{Pa}, \cdot)$, i.e.,

$$U_i = f_i^{-1}\Big(\text{Pa}(V_i), V_i\Big). \tag{2}$$

Note that the nonlinear ANM (Peters et al., 2014), LSNM (Immer et al., 2022), and PNL (Zhang & Hyvärinen, 2009) models from prior work are all special cases of BGMs. Specifically, the nonlinear ANM model,

$$f_i\Big(\text{Pa}(V_i), U_i\Big) = g_i\Big(\text{Pa}(V_i)\Big) + U_i \tag{3}$$

LSNM (Immer et al., 2022) model,

$$f_i\Big(\text{Pa}(V_i), U_i\Big) = l\Big(\text{Pa}(V_i)\Big) + s\Big(\text{Pa}(V_i)\Big)U_i \tag{4}$$

with s a strictly positive function on $\mathbb{R}$, and PNL causal models (Zhang & Hyvärinen, 2009),

$$f_i\Big(\text{Pa}(V_i), U_i\Big) = h_i\Big(g_i\Big(\text{Pa}(V_i)\Big) + U_i\Big) \tag{5}$$

with h_i an invertible function, are all bijective given $\text{Pa}(V_i)$.

5. Identifiability

Without loss of generality, we focus on identifying the generation mechanism f_i of a particular endogenous node V_i shown in Equation (1). For ease of exposition, we drop the subscript i, and refer to $\text{Pa}(V)$ as $\boldsymbol{X}$. The sub-graph of $\mathcal{G}$ which is related to the generation of V is depicted in Figure 1a. The dashed-bidirected arrow indicates potential existence of an unobserved confounder. All proofs are in Appendix B.

We provide three sets of constraints on f and the underlying causal structure that imply counterfactual identifiability given an observed $(\mathcal{L}_1)$ data distribution $(\mathcal{D})$, i.e., given $\mathcal{D}$, each set of constraints identifies f up to indeterminacies that do not affect counterfactual queries.

U and V can be single- or multi-dimensional in general. Our results in §5.1 and §5.2 are for the scalar case, while the result in §5.3 applies to multi-dimensional U and V.

5.1. The Markovian Case

If the exogenous variable U associated with V is independent of its parents $\boldsymbol{X}$, we end up with the causal diagram shown in Figure 1b where $\boldsymbol{X} \perp\!\!\!\perp U$.[4]

Theorem 5.1. *BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V}$ if*

1. *(Markovian) $U \perp\!\!\!\perp \boldsymbol{X}$.*
2. *For all $\boldsymbol{x}$, $f(\boldsymbol{x}, \cdot)$ is either a strictly increasing function or a strictly decreasing function.*

This theorem is similar to Lu et al. (2020, Thm. 1), and is mentioned here for completeness. In this result, $U, \boldsymbol{X}, V$ can be discrete or continuous. Note that independence of U and $\boldsymbol{X}$ by itself is not sufficient for identifiability as demonstrated with a counter-example in Appendix B.3.1 and an experiment in Appendix E.1.1. It is not clear how to generalize the monotonicity condition to BGMs with multi-dimensional V, which is a known issue in Markovain causal structures (Nasr-Esfahany & Kiciman, 2023).

5.2. Instrumental Variable (IV)

Even with an unobserved confounder, we can establish counterfactual identifiability from observational $(\mathcal{L}_1)$ data if we can find IVs relative to the pair $(\boldsymbol{X}, V)$. We define an IV

[4] It is common practice to exclude the exogenous variable U from the causal diagram.

as a set of variables independent of U that affect V only through $\boldsymbol{X}$. Figure 2a shows an example in which $\boldsymbol{I}$ is an IV with respect to $(\boldsymbol{X}, V)$.[5]

The following theorem formalizes counterfactual identifiability in this setting for discrete-valued $\boldsymbol{X}$ and $\boldsymbol{I}$, i.e., $\boldsymbol{X} \in \mathbb{X} \triangleq \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$ and $\boldsymbol{I} \in \mathbb{I} \triangleq \{\boldsymbol{i}_1, \dots, \boldsymbol{i}_n\}$.[6]

Theorem 5.2. *BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V,\boldsymbol{I}}$ if*

1. *(IV) $\boldsymbol{I} \perp\!\!\!\perp U$.*
2. *For all $\boldsymbol{x} \in \mathbb{X}$, $f(\boldsymbol{x}, \cdot)$ and $f^{-1}(\boldsymbol{x}, \cdot)$ are either strictly increasing or strictly decreasing, and two times differentiable functions.*
3. *$P(\boldsymbol{i}, \boldsymbol{x}, \cdot)$ is differentiable for every $\boldsymbol{i} \in \mathbb{I}, \boldsymbol{x} \in \mathbb{X}$.*
4. *(Positivity) $\forall u, \boldsymbol{x} \in \mathbb{X} : P_{U,\boldsymbol{X}}(u, \boldsymbol{x}) > 0$.*
5. *(Variability) $\forall u : |\det \boldsymbol{M}(u, \mathbb{I})| \geq c > 0$, where*

$$\boldsymbol{M}(u, \mathbb{I}) \triangleq \begin{bmatrix} P(\boldsymbol{x}_1|u,\boldsymbol{i}_1) & \dots & P(\boldsymbol{x}_n|u,\boldsymbol{i}_1) \\ \vdots & \ddots & \vdots \\ P(\boldsymbol{x}_1|u,\boldsymbol{i}_n) & \dots & P(\boldsymbol{x}_n|u,\boldsymbol{i}_n) \end{bmatrix}$$

Besides the technical conditions required for the proof, the variability condition in Theorem 5.2 has the following interpretation: the IV must take a sufficient number of distinct values (at least as many values as possible for $\boldsymbol{X}$), and that the IV must have a strong influence on $\boldsymbol{X}$. Positivity implies that the support of U is independent of $\boldsymbol{X}$.

5.3. The Backdoor Criterion (BC)

The second setting in which we can establish counterfactual identifiability from observational $(\mathcal{L}_1)$ data in the presence of latent confounding is when there exists a set of variables $(\boldsymbol{Z})$ that satisfy the backdoor criterion (BC) with respect to $(\boldsymbol{X}, V)$, i.e., if $\boldsymbol{Z}$ blocks every path between $\boldsymbol{X}$ and V that contains an arrow into $\boldsymbol{X}$.[7] Intuitively, we want $\boldsymbol{Z}$ to be responsible for all the spurious correlation (the dashed-bidirected edge) between $\boldsymbol{X}$ and U. Figure 2b demonstrates an example where $\boldsymbol{Z}$ satisfies the BC with respect to $(\boldsymbol{X}, V)$.

In the following, assume $U, V \in \mathbb{R}^d$.

Theorem 5.3. *BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V,\boldsymbol{Z}}$ if*

1. *(BC) $U \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$.*
2. *$\forall \boldsymbol{x} : \nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{f(\boldsymbol{x},\cdot)}|$ and $\nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{f^{-1}(\boldsymbol{x},\cdot)}|$ exist.*
3. *(Variability) $\forall u$: Instances $\boldsymbol{z}_1, \dots, \boldsymbol{z}_{d+1}$ exist such that $|\det \boldsymbol{M}(u, \boldsymbol{z}_1, \dots, \boldsymbol{z}_{d+1})| > 0$, where*

$$\boldsymbol{M}(u, \boldsymbol{z}_1, \dots, \boldsymbol{z}_{d+1}) \triangleq \begin{bmatrix} P(u|\boldsymbol{z}_1) & \nabla_u P(u|\boldsymbol{z}_1) \\ \vdots & \vdots \\ P(u|\boldsymbol{z}_{d+1}) & \nabla_u P(u|\boldsymbol{z}_{d+1}) \end{bmatrix}$$

In the above theorem, $\boldsymbol{Z}$ can be both discrete or continuous. The variability condition implies that $\mathbf{Z}$ must have a strong enough influence on U.

6. Efficient Learning of the BGM

Given an observed data distribution $(\mathcal{D})$, our goal in this section is to find a BGM $\hat{f}$ that is counterfactually equivalent to the BGM f underlying the data. First, we formalize the notion of equivalence:

Definition 6.1. *(Equivalence)* BGMs f_1 and f_2 are equivalent iff there exists an invertible function $g(\cdot)$ such that

$$\forall \boldsymbol{x}, v : f_1^{-1}(\boldsymbol{x}, v) = g\Big(f_2^{-1}(\boldsymbol{x}, v)\Big) \text{ or alternatively} \quad (6)$$

$$\forall \boldsymbol{x}, u : f_1(\boldsymbol{x}, u) = f_2\Big(\boldsymbol{x}, g^{-1}(u)\Big). \quad (7)$$

Proposition 6.2. *BGMs f_1 and f_2 produce the same counterfactuals iff they are equivalent.*

If we find an $\hat{f}$ which is equivalent to the BGM f underlying the data, we can perform Abduction-Action-Prediction (§2) using $\hat{f}$ to estimate any counterfactual quantity, which is guaranteed to produce the same counterfactuals as the true BGM f via Proposition 6.2.

The theorems in §5 (see also the lemmas in Appendix B.2) provide a tractable objective for learning a BGM that is equivalent to the ground-truth BGM for counterfactual estimation. We now describe a recipe for efficiently solving this learning problem.

To represent the search space for $\hat{f}$, we use parameterized bijective transforms $\hat{f}_\theta(\boldsymbol{x}, \cdot)$ from $\hat{U}$ to V, conditioned on $\boldsymbol{X}$. This resembles transforms used in Conditional Generative Models (CGM). Furthermore, we require certain constraints on $\hat{f}_\theta$ depending on the case, e.g., being strictly monotonic or differentiable, which we take into account to select an appropriate model family and parameterization. For example, both the IV and BC cases require differentiablity of the transform. Conditional Normalizing Flow (CNF) (Papamakarios et al., 2021) models are a good candidate for learning such functions as they are typically built using differentiable mappings with differentiable inverses (diffeomorphisms). Although our approach is applicable to

[5] There are several other IV definitions in the causal literature that capture the same concept, e.g., Bowden et al. (1990); Pearl (1995b); Angrist et al. (1996); Heckman & Vytlacil (1999). See Pearl (2009a, Sec. 7.4.5) for a discussion.

[6] If $|\mathbb{I}| > n$, pick any subset with n members.

[7] Pearl (2009a, Def. 3.3.1) requires $\boldsymbol{Z}$ to be non-descendent of $\boldsymbol{X}$ as well, but we do not need such a requirement as our goal is not to use $\boldsymbol{Z}$ for adjustment.

any CGM with the desired properties, we use CNFs in our experiments. Appendix D discusses CNFs in more detail.

Each set of constraints has an important distributional requirement (the first condition) in the form of (conditional) independence among $\hat{U}$ (Equation (2)) and other variables. One way to guide the learning to attain such conditional independence properties is to use adversarial learning for distribution matching (Li et al., 2017). However, a more elegant solution emerges from flipping the problem. Instead of passing $(\mathbf{X}, V) \in \mathcal{D}$ samples through $\hat{f}_\theta^{-1}$ (Equation (2)) and optimizing the transform to satisfy the (conditional) independence, we can sample $\hat{U}$ in a way that satisfies the (conditional) independence needed in each case, and optimize $\hat{f}_\theta$ (Equation (1)) to produce the observed conditional distribution $P_{\mathcal{D}}(V|\boldsymbol{X})$. This has two benefits: i) (Conditional) independence is guaranteed by construction. ii) Training objective simply becomes density modeling, which has been greatly studied in the literature.

The Markovian Case in §5.1 provides a simple example to explain the learning method. We use a strictly increasing conditional transformation as the search space for $\hat{f}_\theta$. We sample $\hat{U}$ and $\boldsymbol{X}$ independently from a Gaussian distribution and the dataset $\mathcal{D}$, respectively. This guarantees the first condition of Theorem 5.1. We optimize the parameters of $\hat{f}_\theta$ to match the conditional distribution $P_{\mathcal{D}}(V|\boldsymbol{X})$. In doing so, we are free to choose from a diverse set of objectives provided by years of research in generative modeling (Mohamed & Lakshminarayanan, 2016), for instance, variational loss (Kingma & Welling, 2013), adversarial loss (Goodfellow et al., 2014), likelihood maximization (Papamakarios et al., 2021), score matching (Song & Ermon, 2019) or denoising diffusion (Ho et al., 2020), etc.

In the **IV** (§5.2) and **BC** (§5.3) cases on the other hand, training is not as simple at the first glance. We cannot sample $\hat{U}$ and $\boldsymbol{X}$ independently anymore, as they do not need to be independent. Additionally, we need to satisfy the conditional independence constraints of each case.

To efficiently encode the conditional independence constraints, we use directed graphical models (DGM) (Koller & Friedman, 2009) for building *structured generative networks* which consist of CGMs as their building blocks, and inherit all conditional independencies encoded in the DGM.[8] Figure 3 demonstrates the DGMs used for satisfying the conditional independencies of IV and BC cases, as well as structured generative models built according to these DGMs. Note that the shown DGMs and structured generative networks are not the only ones that satisfy the required conditional independence properties. In the BC case, there are several alternatives DGM that encode the same conditional

[8]Here, we treat DGMs as a pure statistical objects devoid of any causal semantics.

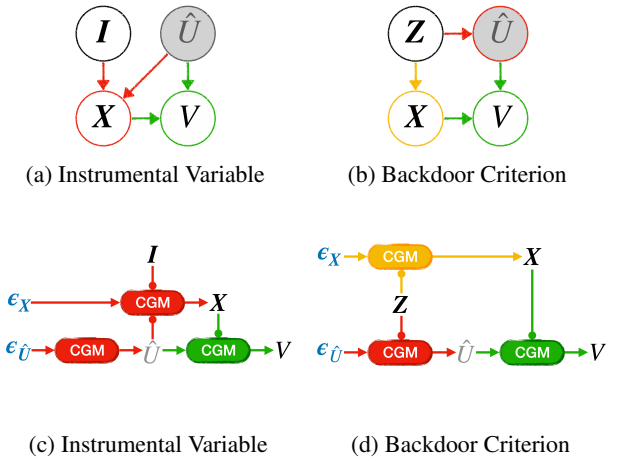


Figure 3. Graphical models (top) encode desired (conditional) independencies. Structured generative networks (bottom) for learning BGMs are constructed according to the corresponding graphical models. Blue variables are sampled independently from Gaussian distributions while black variables are observed in the dataset.

independencies and each define a distinct structured generative network that could be used to learn $\hat{f}$ (see Appendix C). Also note that the CGM used for representing $\hat{f}_\theta$ (green one) has requirements like monotonicity and differentiability, but the others do not have these restrictions.

Training: We sample the root variables, $\boldsymbol{\epsilon_X}, \epsilon_{\hat{U}}, \boldsymbol{I}$ in Figure 3c, and $\boldsymbol{\epsilon_X}, \epsilon_{\hat{U}}, \boldsymbol{Z}$ in Figure 3d, independently. The epsilon variables (in blue) are sampled from independent Gaussian distributions, and the others from the dataset $\mathcal{D}$. We train the whole structure end-to-end, with the goal of matching the distribution of $\mathcal{D}$. This process is akin to a constrained search for $\hat{f}$ where the constraints are embedded in the search space by construction. The objective function is determined by the choice of CGM. We can even mix different types of CGMs, and combine different objective functions for their end-to-end optimization.

The end goal of training is to learn the green CGM. As a result, any CGM whose training does not affect the green CGM can be removed from the network as a simplification. The yellow CGM in Figure 3d is such an example, as it is surrounded by black variables that are sampled directly from $\mathcal{D}$, which prevent its gradients from passing through and influencing training of the other CGMs.

7. Efficient Counterfactual Estimation

Once we have learned a BGM using the techniques in §6, estimating counterfactual queries is a straightforward application of the Abduction-Action-Prediction procedure (§2). We illustrate this using two examples.

Suppose we are interested in a counterfactual query

$V_{\boldsymbol{x}'}|\{\boldsymbol{X}=\boldsymbol{x}, V=v\}$, which seeks to determine the necessity of causation, e.g., given that a patient recovered ($V=1$) under a treatment $X=1$, would she have also recovered without the treatment ($X=0$). This query is in general non-identifiable from observational ($\mathcal{L}_1$) and interventional ($\mathcal{L}_2$) data (Pearl, 1999; Tian & Pearl, 2000; Li & Pearl, 2022b). However, if we can approximate the generation mechanism of V as a BGM, and if one of the identifiability conditions of §5 holds, there is hope. Specifically, we first learn the BGM from observational data (§6). Then, we do the abduction step by inverting the BGM, i.e., $\hat{u}=\hat{f}^{-1}(\boldsymbol{x}, v)$, and pass the counterfactual $\boldsymbol{x}'$ through the BGM to get the counterfactual estimate, i.e., $v'=\hat{f}(\boldsymbol{x}', \hat{u})$.

As another example, consider $P(V_{\boldsymbol{x}_2}|\boldsymbol{X}=\boldsymbol{x}_1)$ which questions the effect of treatment on the treated (Shpitser & Pearl, 2009). For the abduction step, we invert the learned BGM ($\hat{f}$) for all the observed samples (e.g., patients) assigned to $\boldsymbol{x}_1$ to obtain samples of the exogenous posterior distribution $P_{\hat{U}|\boldsymbol{x}_1}$ as $\hat{u}=\hat{f}^{-1}(\boldsymbol{x}_1, v), v\sim P(V|\boldsymbol{x}_1)$. For the prediction step, we pass the samples $\hat{u}$ through the BGM with $X=\boldsymbol{x}_2$, to obtain samples of the counterfactual distribution of interest $v'=\hat{f}(\boldsymbol{x}_2, \hat{u}), \hat{u}\sim P_{\hat{U}|\boldsymbol{x}_1}$.

The examples above considered only the generation function of particular node $V\in\boldsymbol{V}$ in the causal diagram. If all generation functions of an SCM ($\mathcal{F}$) are well-approximated by BGMs, and can also be learned in the settings described in §5, then we can answer every counterfactual query in a similar way. However, not all generation mechanisms need to be BGMs, or known, for answering a particular counterfactual query of interest. For instance, there is no need to learn BGMs for ancestors of the variables that we intervene on, or the variables that do not appear in the evidence and do not have dashed-bidirected edges to the evidence.

8. Experiments

We evaluate our techniques in two settings: (i) a simple task that allows us to visualize our approach and prior baselines, and (ii) a real-world video streaming simulation task. Appendix E has implementation details.

8.1. Counterfactual Ellipse Generation

A standard ellipse is determined by two parameters: a semi-major and semi-minor axis (a, b). If we further specify an angle, we get a single point on the ellipse. Let $U\in\mathbb{R}^2$ be the two parameters of an ellipse, $X\in(0, 2\pi)$ an angle that specifies a single point, $V\in\mathbb{R}^2$ the Cartesian coordinates of this point, and f the function that calculates these coordinates given U and X.

Suppose we are given data generated as follows. We sample $z\sim P_Z$, $u\sim P_{U|Z=z}$ and $x\sim P_{X|Z=z}$, and output the data tuple $(z, x, v:=f(u, x))$. Here, P_Z, $P_{U|Z}$, and $P_{X|Z}$ are three predefined distributions (see Appendix E.1 for details). The important point is that conditioned on Z, the ellipse parameters and angles are independent ($U\perp\!\!\!\perp X|Z$), but U and X are not independent unconditionally. For a specific pair (x, v) observed in the dataset (evidence), our goal is to draw the entire ellipse that the point v belongs to. This can be done by estimating counterfactual queries, $V_{x'}|\{X=x, V=v\}$, for $x'\in(0, 2\pi)$.

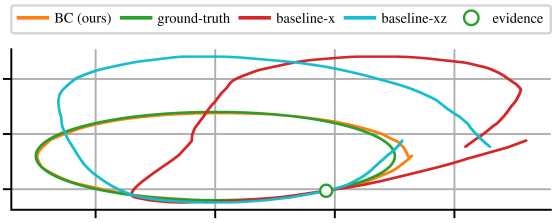


Figure 4. BC (ours) is the only scheme that generates an ellipse.

Table 1. Counterfactual Ellipse Generation Accuracy

SCHEME	BC	BASELINE-X	BASELINE-XZ
MAPE	1%	6607%	6582%

This task corresponds to our **BC** setting with Z as the back-door variable. We evaluate our method for BC visually (Figure 4) and quantitatively (Table 1). As you can see, it achieves a high accuracy as opposed to the baselines, both of which are single CGMs that do not take the causal structure into account, which is common in prior work (Lample et al., 2017; Zhu et al., 2017; He et al., 2019). **Baseline-x** models $P(V|X)$ and **baseline-xz** models $P(V|X, Z)$.

8.2. Case Study: Video Streaming Simulation

Video streaming clients use Adaptive Bitrate (ABR) algorithms to continually adapt the bitrate of a video stream based on network conditions. ABR algorithms select a bitrate for each few-second chunk of video among a finite set of available choices. These algorithms have a significant impact on user experience (e.g., video quality and stalls) and have been the subject of extensive research (Tian & Liu, 2012; Huang et al., 2014; Yin et al., 2015; Sun et al., 2016; Mao et al., 2017; Akhtar et al., 2018; Spiteri et al., 2020).

Trace-driven simulation is a common approach to design and evaluate ABR algorithms. Here, one collects traces from real video streaming sessions, with each trace providing a timeseries of observed network throughput (and possibly other player metrics) for every video chunk. At simulation time, the traces are replayed (to represent network behavior) while simulating the dynamics of video clients under new ABR algorithms. However, simply replaying a throughput trace can bias simulation outcomes (Bartulovic et al., 2017;

Table 2. Counterfactual estimation accuracy of different schemes and their training time for video streaming simulation

SCHEME	NORMALIZED MSE (%)	TIME (s)
MARKOVIAN	4.1 ± 2.1	28 ± 0
IV	10.0 ± 0.7	127 ± 1
CAUSALSIM	9.3 ± 1.0	1091 ± 1
BC	10.4 ± 7.8	37 ± 8
IV+BC	6.4 ± 2.4	49 ± 1

Alomar et al., 2023), because changing the ABR algorithm could effect the throughput that would have been achieved for the same video streaming session.

Alomar et al. (2023) formulated bias removal in this type of simulation as a counterfactual estimation problem where for download of each chunk, V is the achieved throughput, X is the chosen bitrate, U is the unobserved bottleneck link capacity, and $V = f(U, X)$. Note that in this problem, f is strictly increasing for each value of X as higher bottleneck link capacity (U) increases the achieved throughput (V). They prove identifiability for this counterfactual query assuming data collected in a Randomized Control Trial (RCT) with sufficiently diverse ABR algorithms and low-rank structure of the underlying BGM. They use a deterministic auto-encoder (Ghosh et al., 2019) equipped with adversarial learning for distribution matching for counterfactual estimation, and present experimental results (including a real-world ABR design case study) that show it significantly improves simulation accuracy compared to standard (biased) trace-driven simulators.

Next, we demonstrate how each set of conditions for identifiability (§5) translates to this real-world problem along with the efficacy of our practical algorithm (§6) using the ABR simulator provided by Alomar et al. (2023). We use the simulator to generate traces for each setting, which we use to learn the BGM. The simulator gives us ground-truth counterfactuals for error calculation. We normalize the Mean Squared Error (MSE) of our method's counterfactual estimates by the MSE of a standard (biased) video streaming simulator that assumes chosen bitrates do not affect the achieved throughput. All results are in Table 2 with mean and standard deviation calculated over ten random seeds trained until training loss convergence.

The Markovian Case: The causal structure in this problem is not Markovian because X (chosen bitrate) and V (observed throughput) are both affected by underlying network conditions (latent confounder). However, we can remove the confounding effect if the ABR algorithm used for trace collection chooses random bitrates from time to time, and we only select the subset of traces with these random decisions as $\mathcal{D}$ (Figure 1). This scheme achieves the lowest error in Table 2 as it has access to the most pristine data (perfect confounding removal by invasive randomization).

Instrumental Variable (IV): It is common for video service providers to conduct RCTs over various ABR algorithms for their comparison. If traces are collected from an RCT, their causal structure naturally fits Figure 2a where IV (I) is the algorithm identifier. This is the same setting that **CausalSim** (2023) explores, so we use their adversarial learning method as a baseline. We collect RCT data over ten different provided algorithms. Our IV method achieves almost the same accuracy as CausalSim (slight difference is not statistically significant), but converges $8.6\times$ faster since it does not require bi-level (adversarial) optimization.

Backdoor Criterion (BC): Buffer based ABR algorithms are those that only make use of the client's current playback buffer level to make bitrate decisions. They are strong ABR algorithms despite their simplicity and are widely used in practice (Yan et al., 2020). If a trace is collected using a buffer based algorithm (and includes playback buffer observations), the causal structure follows Figure 2b where Z is the buffer used for choosing the bitrate. Hence, it satisfies the backdoor criterion. Table 2 shows that our BC method applied to buffer-based traces is quite accurate even without access to RCT data (although it is slightly worse than CausalSim and IV).

IV + BC: It is possible to combine settings explored in §5 to further improve accuracy provided the problem has the appropriate causal structure. For example, in video streaming simulation, if $\mathcal{D}$ is collected in an RCT over buffer based algorithms, the BC and IV cases apply simultaneously. We refer to this case as IV+BC. Similar to the IV case, the algorithm identifier is the instrumental variable while the tuple (algorithm identifier, buffer) satisfies the backdoor criterion. Appendix E.2.1 describes the structured generative network for this case. We apply it to traces collected using an RCT over two buffer based algorithms. IV+BC decreases the average error compared to the BC scheme by almost 38%.

9. Concluding Remarks

In this work, we defined Bijective Generation Mechanisms (BGM), a class of models that contain several widely used causal models in the literature (§4). We established their counterfactual identifiability for three well-known causal structures (§5) and proposed a practical learning method that casts learning a BGM as structured generative modeling (§6). We evaluated our methodology in a visual task and demonstrated its application to a real-world video streaming simulation task (§8). Finite sample analysis of our identifiability theorems, extending the identifiable causal structures, and applications of the proposed method to real-world problems in various field, e.g., econometrics, computer systems,

causal ML, etc., are exciting directions for future work.

Acknowledgements

We would like to thank Behrooz Tahmasebi for discussions that led to the proof of Theorem 5.2. This work was supported by NSF grant 1751009, a gift from the CSAIL SystemsThatLearn (STL) Initiative, and Google, Intel, and Amazon as part of the MIT Data Systems and AI (DSAIL) lab.

References

Agarap, A. F. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

Agarwal, A., Alomar, A., Alumootil, V., Shah, D., Shen, D., Xu, Z., and Yang, C. Persim: Data-efficient offline reinforcement learning with heterogeneous agents via personalized simulators. *Advances in Neural Information Processing Systems*, 34:18564–18576, 2021.

Akhtar, Z., Nam, Y. S., Govindan, R., Rao, S., Chen, J., Katz-Bassett, E., Ribeiro, B., Zhan, J., and Zhang, H. Oboe: Auto-tuning video abr algorithms to network conditions. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pp. 44–58, 2018.

Alomar, A., Hamadanian, P., Nasr-Esfahany, A., Agarwal, A., Alizadeh, M., and Shah, D. CausalSim: A causal framework for unbiased Trace-Driven simulation. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, pp. 1115–1147, Boston, MA, April 2023. USENIX Association. ISBN 978-1-939133-33-5. URL https://www.usenix.org/conference/nsdi23/presentation/alomar.

Angrist, J. D., Imbens, G. W., and Rubin, D. B. Identification of causal effects using instrumental variables. *Journal of the American statistical Association*, 91(434): 444–455, 1996.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Balke, A. and Pearl, J. Counterfactual probabilities: Computational methods, bounds and applications. In *Uncertainty Proceedings 1994*, pp. 46–54. Elsevier, 1994.

Balke, A. and Pearl, J. Bounds on treatment effects from studies with imperfect compliance. *Journal of the American Statistical Association*, 92(439):1171–1176, 1997.

Bareinboim, E., Correa, J. D., Ibeling, D., and Icard, T. *On Pearl's Hierarchy and the Foundations of Causal Inference*, pp. 507–556. Association for Computing Machinery, New York, NY, USA, 1 edition, 2022. ISBN 9781450395861. URL https://doi.org/10.1145/3501714.3501743.

Bartulovic, M., Jiang, J., Balakrishnan, S., Sekar, V., and Sinopoli, B. Biases in data-driven networking, and what to do about them. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, pp. 192–198, 2017.

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*, 2018.

Bothra, C., Gao, J., Rao, S., and Ribeiro, B. Veritas: Answering causal queries from video streaming traces. *arXiv preprint arXiv:2208.12596*, 2022.

Bowden, R. J., Turkington, D. A., et al. Instrumental variables. *Cambridge Books*, 1990.

Budhathoki, K., Minorics, L., Blöbaum, P., and Janzing, D. Causal structure-based root cause analysis of outliers. In *International Conference on Machine Learning*, pp. 2357–2369. PMLR, 2022.

Carroll, R. J., Ruppert, D., Stefanski, L. A., and Crainiceanu, C. M. *Measurement error in nonlinear models: a modern perspective*. Chapman and Hall/CRC, 2006.

Chen, R. T., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. Residual flows for invertible generative modeling. *Advances in Neural Information Processing Systems*, 32, 2019.

Correa, J., Lee, S., and Bareinboim, E. Nested counterfactual identification from arbitrary surrogate experiments. *Advances in Neural Information Processing Systems*, 34: 6856–6867, 2021.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Dolatabadi, H. M., Erfani, S., and Leckie, C. Invertible generative modeling using linear rational splines. In *International Conference on Artificial Intelligence and Statistics*, pp. 4236–4246. PMLR, 2020.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural spline flows. *Advances in neural information processing systems*, 32, 2019.

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. Underspecification presents

challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 2020.

Finkelstein, N. and Shpitser, I. Deriving bounds and inequality constraints using logical relations among counterfactuals. In *Conference on Uncertainty in Artificial Intelligence*, pp. 1348–1357. PMLR, 2020.

Fulcher, I. R., Shpitser, I., Marealle, S., and Tchetgen Tchetgen, E. J. Robust inference on population indirect causal effects: the generalized front door criterion. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(1):199–214, 2020.

Gan, Y., Liang, M., Dev, S., Lo, D., and Delimitrou, C. Enabling practical cloud performance debugging with unsupervised learning. *SIGOPS Oper. Syst. Rev.*, 56(1):34–41, jun 2022. ISSN 0163-5980. doi: 10.1145/3544497.3544503. URL https://doi.org/10.1145/3544497.3544503.

Geffner, T., Antoran, J., Foster, A., Gong, W., Ma, C., Kiciman, E., Sharma, A., Lamb, A., Kukla, M., Pawlowski, N., et al. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022.

Ghosh, P., Sajjadi, M. S., Vergari, A., Black, M., and Scholkopf, B. From variational to deterministic autoencoders. In *International Conference on Learning Representations*, 2019.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In *NIPS*, 2014.

Goodman, N. D., Mansinghka, V. K., Roy, D., Bonawitz, K., and Tenenbaum, J. B. Church: a language for generative models. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, pp. 220–229, 2008.

Gresele, L., Kügelgen, J. V., Kübler, J., Kirschbaum, E., Schölkopf, B., and Janzing, D. Causal inference through the structural causal marginal problem. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 7793–7824. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/gresele22a.html.

Hartford, J., Lewis, G., Leyton-Brown, K., and Taddy, M. Deep iv: A flexible approach for counterfactual prediction. In *International Conference on Machine Learning*, pp. 1414–1423. PMLR, 2017.

He, Z., Zuo, W., Kan, M., Shan, S., and Chen, X. Attgan: Facial attribute editing by only changing what you want. *IEEE transactions on image processing*, 28(11):5464–5478, 2019.

Heckman, J. J. and Vytlacil, E. J. Local instrumental variables and latent variable models for identifying and bounding treatment effects. *Proceedings of the national Academy of Sciences*, 96(8):4730–4734, 1999.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

Huang, T.-Y., Johari, R., McKeown, N., Trunnell, M., and Watson, M. A buffer-based approach to rate adaptation: Evidence from a large video streaming service. In *Proceedings of the 2014 ACM conference on SIGCOMM*, pp. 187–198, 2014.

Hyvärinen, A. and Oja, E. Independent component analysis: algorithms and applications. *Neural networks*, 13:411–430, 2000.

Hyvärinen, A. and Pajunen, P. Nonlinear independent component analysis: Existence and uniqueness results. *Neural networks*, 12(3):429–439, 1999.

Hyvarinen, A., Sasaki, H., and Turner, R. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868. PMLR, 2019.

Ibeling, D. and Icard, T. Probabilistic reasoning across the causal hierarchy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 10170–10177, 2020.

Immer, A., Schultheiss, C., Vogt, J. E., Schölkopf, B., Bühlmann, P., and Marx, A. On the identifiability and estimation of causal location-scale noise models. *arXiv preprint arXiv:2210.09054*, 2022.

Jaber, A., Zhang, J., and Bareinboim, E. Causal identification under markov equivalence: Completeness results. In *International Conference on Machine Learning*, pp. 2981–2989. PMLR, 2019.

Johansson, F., Shalit, U., and Sontag, D. Learning representations for counterfactual inference. In *International*

conference on machine learning, pp. 3020–3029. PMLR, 2016.

Jung, Y., Tian, J., and Bareinboim, E. Learning causal effects via weighted empirical risk minimization. *Advances in neural information processing systems*, 33: 12697–12709, 2020.

Kennedy, E. H. Nonparametric causal effects based on incremental propensity score interventions. *Journal of the American Statistical Association*, 114(526):645–656, 2019.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Khemakhem, I., Monti, R., Leech, R., and Hyvarinen, A. Causal autoregressive flows. In *International conference on artificial intelligence and statistics*, pp. 3520–3528. PMLR, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kobyzev, I., Prince, S. J., and Brubaker, M. A. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):3964–3979, 2020.

Kocaoglu, M., Snyder, C., Dimakis, A. G., and Vishwanath, S. Causalgan: Learning causal implicit generative models with adversarial training. In *International Conference on Learning Representations*, 2018.

Koller, D. and Friedman, N. *Probabilistic graphical models: principles and techniques*. The MIT Press, 2009.

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

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. *Advances in neural information processing systems*, 30, 2017.

Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., and Ranzato, M. Fader networks: Manipulating images by sliding attributes. *Advances in neural information processing systems*, 30, 2017.

Lee, S. and Bareinboim, E. Causal identification with matrix equations. *Advances in Neural Information Processing Systems*, 34:9468–9479, 2021.

Li, A. and Pearl, J. Bounds on causal effects and application to high dimensional data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 5773–5780, 2022a.

Li, A. and Pearl, J. Probabilities of causation: Role of observational data. *arXiv preprint arXiv:2210.08874*, 2022b.

Li, C., Liu, H., Chen, C., Pu, Y., Chen, L., Henao, R., and Carin, L. Alice: Towards understanding adversarial learning for joint distribution matching. *Advances in neural information processing systems*, 30, 2017.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Lorberbom, G., Johnson, D. D., Maddison, C. J., Tarlow, D., and Hazan, T. Learning generalized gumbel-max causal mechanisms. *Advances in Neural Information Processing Systems*, 34:26792–26803, 2021.

Louizos, C., Shalit, U., Mooij, J. M., Sontag, D., Zemel, R., and Welling, M. Causal effect inference with deep latent-variable models. *Advances in neural information processing systems*, 30, 2017.

Lu, C., Huang, B., Wang, K., Hernández-Lobato, J. M., Zhang, K., and Schölkopf, B. Sample-efficient reinforcement learning via counterfactual-based data augmentation. *arXiv preprint arXiv:2012.09092*, 2020.

Lu, Y. and Huang, B. Structured output learning with conditional generative flows. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5005–5012, 2020.

Manski, C. F. Nonparametric bounds on treatment effects. *The American Economic Review*, 80(2):319–323, 1990.

Mao, H., Netravali, R., and Alizadeh, M. Neural adaptive video streaming with pensieve. In *Proceedings of the conference of the ACM special interest group on data communication*, pp. 197–210, 2017.

Meng, C., Zhou, L., Choi, K., Dao, T., and Ermon, S. Butterflyflow: Building invertible layers with butterfly matrices. In *International Conference on Machine Learning*, pp. 15360–15375. PMLR, 2022.

Mesnard, T., Weber, T., Viola, F., Thakoor, S., Saade, A., Harutyunyan, A., Dabney, W., Stepleton, T. S., Heess, N., Guez, A., et al. Counterfactual credit assignment in model-free reinforcement learning. In *International Conference on Machine Learning*, pp. 7654–7664. PMLR, 2021.

Miao, W., Geng, Z., and Tchetgen Tchetgen, E. J. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105:987–993, 2018.

Mohamed, S. and Lakshminarayanan, B. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

Monteiro, M., Ribeiro, F. D. S., Pawlowski, N., Castro, D. C., and Glocker, B. Measuring axiomatic soundness of counterfactual image models. In *The Eleventh International Conference on Learning Representations*, 2023.

Nasr-Esfahany, A. and Kiciman, E. Counterfactual (non-)identifiability of learned structural causal models, 2023. URL https://arxiv.org/abs/2301.09031.

Oberst, M. and Sontag, D. Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pp. 4881–4890. PMLR, 2019.

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):1–64, 2021.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Pawlowski, N., Coelho de Castro, D., and Glocker, B. Deep structural causal models for tractable counterfactual inference. *Advances in Neural Information Processing Systems*, 33:857–869, 2020.

Pearl, J. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

Pearl, J. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995a.

Pearl, J. On the testability of causal models with latent and instrumental variables. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pp. 435–443, 1995b.

Pearl, J. Probabilities of causation: Three counterfactual interpretations and their identification. *Synthese*, 121(1): 93–149, 1999.

Pearl, J. *Causality*. Cambridge university press, 2009a.

Pearl, J. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009b.

Pearl, J. and Mackenzie, D. *The book of why: the new science of cause and effect*. Basic books, 2018.

Perković, E., Textor, J., and Kalisch, M. Complete graphical characterization and construction of adjustment sets in markov equivalence classes of ancestral graphs. *Journal of Machine Learning Research*, 19:1–62, 2018.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15:2009–2053, 2014.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Richens, J. G., Beard, R., and Thompson, D. H. Counterfactual harm, 2022. URL https://arxiv.org/abs/2204.12993.

Rubin, D. B. Bayesian inference for causal effects: The role of randomization. *The Annals of statistics*, pp. 34–58, 1978.

Sanchez, P. and Tsaftaris, S. A. Diffusion causal models for counterfactual estimation. In *First Conference on Causal Learning and Reasoning*, 2021.

Sanchez-Martin, P., Rateike, M., and Valera, I. Vaca: Design of variational graph autoencoders for interventional and counterfactual queries. *arXiv preprint arXiv:2110.14690*, 2021.

Shah, A., Dwivedi, R., Shah, D., and Wornell, G. On counterfactual inference with unobserved confounding. In *NeurIPS 2022 Workshop on Causality for Real-world Impact*, 2022.

Shaham, U., Svirsky, J., Katz, O., and Talmon, R. Discovery of single independent latent variable. In *Advances in Neural Information Processing Systems*, 2022.

Shalit, U., Johansson, F. D., and Sontag, D. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*, pp. 3076–3085. PMLR, 2017.

Shpitser, I. and Pearl, J. What counterfactuals can be tested. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, pp. 352–359, 2007.

Shpitser, I. and Pearl, J. Effects of treatment on the treated: identification and generalization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 514–521, 2009.

Shpitser, I. and Sherman, E. Identification of personalized effects associated with causal pathways. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 2018. NIH Public Access, 2018.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Spirtes, P., Glymour, C., Scheines, R., et al. Causation, prediction, and search. *MIT Press Books*, 1, 2001.

Spiteri, K., Urgaonkar, R., and Sitaraman, R. K. Bola: Near-optimal bitrate adaptation for online videos. *IEEE/ACM Transactions on Networking*, 28(4):1698–1711, 2020.

Strobl, E. V. and Lasko, T. A. Identifying patient-specific root causes with the heteroscedastic noise model. *arXiv preprint arXiv:2205.13085*, 2022.

Sun, Y., Yin, X., Jiang, J., Sekar, V., Lin, F., Wang, N., Liu, T., and Sinopoli, B. Cs2p: Improving video bitrate selection and adaptation with data-driven throughput prediction. In *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 272–285, 2016.

Tian, G. and Liu, Y. Towards agile and smooth video adaptation in dynamic http streaming. In *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, pp. 109–120, 2012.

Tian, J. and Pearl, J. Probabilities of causation: Bounds and identification. *Annals of Mathematics and Artificial Intelligence*, 28(1):287–313, 2000.

Trippe, B. L. and Turner, R. E. Conditional density estimation with bayesian normalising flows. *arXiv preprint arXiv:1802.04908*, 2018.

Wang, Y. and Blei, D. M. The blessings of multiple causes. *Journal of the American Statistical Association*, 114(528): 1574–1596, 2019.

Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. Learning likelihoods with conditional normalizing flows. *arXiv preprint arXiv:1912.00042*, 2019.

Witty, S., Jensen, D., and Mansinghka, V. Sbi: A simulation-based test of identifiability for bayesian causal inference, 2021. URL https://arxiv.org/abs/2102.11761.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

Xia, K., Lee, K.-Z., Bengio, Y., and Bareinboim, E. The causal-neural connection: Expressiveness, learnability, and inference. *Advances in Neural Information Processing Systems*, 34:10823–10836, 2021.

Xia, K., Pan, Y., and Bareinboim, E. Neural causal models for counterfactual identification and estimation. *arXiv preprint arXiv:2210.00035*, 2022.

Yan, F. Y., Ayers, H., Zhu, C., Fouladi, S., Hong, J., Zhang, K., Levis, P., and Winstein, K. Learning in situ: a randomized experiment in video streaming. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*, pp. 495–511, 2020.

Yao, L., Li, S., Li, Y., Huai, M., Gao, J., and Zhang, A. Representation learning for treatment effect estimation from observational data. *Advances in Neural Information Processing Systems*, 31, 2018.

Yin, X., Jindal, A., Sekar, V., and Sinopoli, B. A control-theoretic approach for dynamic adaptive video streaming over http. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pp. 325–338, 2015.

Zečević, M., Dhami, D. S., Veličković, P., and Kersting, K. Relating graph neural networks to structural causal models. *arXiv preprint arXiv:2109.04173*, 2021.

Zhang, J. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16-17): 1873–1896, 2008.

Zhang, J. and Bareinboim, E. Fairness in decision-making—the causal explanation formula. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Zhang, J. and Bareinboim, E. Bounding causal effects on continuous outcome. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 12207–12215, 2021.

Zhang, J., Tian, J., and Bareinboim, E. Partial counterfactual identification from observational and experimental data. In *International Conference on Machine Learning*, pp. 26548–26558. PMLR, 2022.

Zhang, K. and Hyvärinen, A. On the identifiability of the post-nonlinear causal model. In *25th Conference on Uncertainty in Artificial Intelligence (UAI 2009)*, pp. 647–655. AUAI Press, 2009.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

A. Neural Methods for Causal Estimation

Xia et al. (2021) uses Neural Causal Models (NCMs) learned from observational data for identification and estimation of interventional ($\mathcal{L}_2$) queries, assuming knowledge of the underlying causal diagram (potentially non-Markovian) and discrete endogenous variables.

Kocaoglu et al. (2018) and Zečević et al. (2021) use adversarial training and Graph Neural Networks (GNNs) (Wu et al., 2020), respectively, to learn implicit SCMs from observational data, assuming knowledge of the causal diagram and the Markovian assumption. Learned SCMs are then used for answering interventional queries ($\mathcal{L}_2$), which are known to be identifiable given the Markovianity assumption (Bareinboim et al., 2022, Corol. 2).

Pawlowski et al. (2020) and Sanchez-Martin et al. (2021) use deep conditional generative models of various forms, structured according to the known causal diagram, to learn the SCM from observational data assuming no unobserved confounding (Markovain SCM). The learned SCMs are further used for interventional and counterfactual estimation. Sanchez & Tsaftaris (2021) uses diffusion denoising probabilistic models to learn conditional distribution of images given labeled attributes, which are further used for counterfactual image generation. However, they do not have any identifiability analysis. In fact Nasr-Esfahany & Kiciman (2023) shows counterfactual non-identifiability of generation mechanisms of multi-dimensional variables from observational data in Markovian settings. To assess the quality of non-identifiable image counterfactuals in Markovian SCMs, Monteiro et al. (2023) revisits axiomatic definition of counterfactuals by measuring their *composition*, *reversibility*, and *effectiveness*.

Xia et al. (2022) use Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) to learn proxy SCMs from observational and interventional data, assuming knowledge of (non-Markovian) causal diagram, and discrete endogenous variables. They utizile the proxy SCM for counterfactual ($\mathcal{L}_3$) identification and estimation. Gumbel-Max SCMs (Oberst & Sontag, 2019) and their generalizations (Lorberbom et al., 2021) have been used for counterfactual estimation of categorical variables from observational data. However, we focus mostly on continuous domains.

Assuming SCMs with additive noise (ANM) (Hoyer et al., 2008) and Markovianity, Geffner et al. (2022) learns both the underlying causal structure $\mathcal{G}$ and SCM from observational data, and uses them for estimation of interventional and counterfactual queries. However, their identifiability analysis is restricted to interventional queries only. Non-linear ANMs are a special case of the class of SCMs we consider in this work. Furthermore, we allow existence of unobserved confounders, and prove counterfactual identifiability of our models. Hartford et al. (2017) uses two-stage supervised learning methods to estimate counterfactual queries using Instrumental Variables (IVs) (Angrist et al., 1996), assuming SCMs with ANMs. Alomar et al. (2023) proves identifiability of counterfactual queries from Randomized Control Trial (RCT) data which is a special IV case, assuming low-rank generation mechanisms. Furthermore, they utilize a deterministic auto-encoder (Ghosh et al., 2019) equipped with adversarial learning for distribution matching to enable efficient counterfactual estimation. Our treatment does not need any assumptions about the rank of generation mechanisms.

Khemakhem et al. (2021) uses Affine Causal Autoregressive Flows (Dinh et al., 2016) for learning the underlying causal structure $\mathcal{G}$ from observational data, assuming absence of unobserved confounder. They prove identifiability of interventional queries assuming Location Scale Noise Models (LSNM) (Strobl & Lasko, 2022). Additionally, they propose using the learned SCM for counterfactual estimation, without any identifiability analysis. We allow unobserved confounders to exist, do not restrict generation functions to LSNMs, and also prove counterfactual identifiability.

Louizos et al. (2017) uses Variational Auto-Encoders (VAE) (Kingma & Welling, 2013) to estimate counterfactual queries in cases where sufficient proxy variables (Carroll et al., 2006; Kuroki & Pearl, 2014; Miao et al., 2018; Wang & Blei, 2019; Lee & Bareinboim, 2021) of unobserved confounders are available for identifiability. Johansson et al. (2016); Shalit et al. (2017); Yao et al. (2018) learn representations for estimating *Individual Treatment Effect (ITE)* assuming *strong ignorability* and binary treatments. Our work is not limited to discrete domains, and is not limited to specific counterfactual quantities like ITE.

B. Proofs

B.1. Proposition 6.2

$$\text{BGMs } f_1, f_2 \text{ produce the same counterfactuals} \iff f_1, f_2 \text{ are equivalent BGMs} \tag{8}$$

Proof. First, we prove $\Leftarrow$. Consider an arbitrary counterfactual query $V_{\boldsymbol{x}'}|\boldsymbol{X}=\boldsymbol{x}, V=v$. In the abduction step, both BGMs use the evidence $\boldsymbol{X}=\boldsymbol{x}, V=v$ to infer the exogenous variable U. We refer to f_1's inferred exogenous variable as u_1 and f_2's inferred exogenous variable as u_2. Using Equation (6) we have

$$u_1 = g(u_2). \tag{9}$$

In the prediction step, f_1 and f_2 give $f_1(\boldsymbol{x}', u_1) = f_1\Big(\boldsymbol{x}', g(u_2)\Big)$ and $f_2(\boldsymbol{x}', u_2)$ as their counterfactual estimates, respectively. Using Equation (7) we get

$$f_1\Big(\boldsymbol{x}', g(u_2)\Big) = f_2(\boldsymbol{x}', u_2). \tag{10}$$

Hence, estimated counterfactuals are equal.

Next, we prove $\Rightarrow$. Due to both f_1 and f_2 being BGMs, we can easily verify the following relationship holds between them:

$$\forall \boldsymbol{x}, u_1 : f_1(\boldsymbol{x}, u_1) = f_2\Big(\boldsymbol{x}, g^{-1}(\boldsymbol{x}, u_1)\Big) \tag{11}$$

where

$$\forall \boldsymbol{x}, u_1 : g^{-1}(\boldsymbol{x}, u_1) = f_2^{-1}\Big(\boldsymbol{x}, \Big(f_1(\boldsymbol{x}, u_1)\Big)\Big). \tag{12}$$

Now suppose we use both f_1 and f_2 for estimating the counterfactual query $V_{\boldsymbol{x}'}|\boldsymbol{X}=\boldsymbol{x}, V=v$. f_1's estimate would be $f_1(\boldsymbol{x}', u_1)$. Using Equation (11), this estimate is equal to $f_2\Big(\boldsymbol{x}', g^{-1}(\boldsymbol{x}', u_1)\Big)$. Using f_2 for estimating the same query, it infers the exogenous variable $g^{-1}(\boldsymbol{x}, u_1)$ in the abduction step. In the prediction step, its counterfactual estimate would be $f_2\Big(\boldsymbol{x}', g^{-1}(\boldsymbol{x}, u_1)\Big)$. As both the counterfactual estimates are equal, we have

$$\forall \boldsymbol{x}', \boldsymbol{x}, u_1 : f_2\Big(\boldsymbol{x}', g^{-1}(\boldsymbol{x}', u_1)\Big) = f_2\Big(\boldsymbol{x}', g^{-1}(\boldsymbol{x}, u_1)\Big). \tag{13}$$

Using invertibility property of the BGM f_2 we get

$$\forall \boldsymbol{x}', \boldsymbol{x}, u_1 : g^{-1}(\boldsymbol{x}', u_1) = g^{-1}(\boldsymbol{x}, u_1) \rightarrow \forall \boldsymbol{x}', \boldsymbol{x}, u_1 : g\Big(\boldsymbol{x}', g^{-1}(\boldsymbol{x}, u_1)\Big) = u_1. \tag{14}$$

The only way the last equality could hold for all possible $\boldsymbol{x}, \boldsymbol{x}'$ is if g^{-1} does not depend on its first argument, i.e.,

$$g^{-1}(\boldsymbol{x}, u_1) = g^{-1}(u_1), \tag{15}$$

which means that f_1 and f_2 are equivalent. □

Remark B.1. The reason why we have this indeterminacy $g(\cdot)$ is partly due to the fact that the prior distribution over exogenous variables $\Big(P(U)\Big)$ is unknown. Each choice of this prior distribution would result in a different $g(\cdot)$.

B.2. Counterfactual Equivalence Lemmas

In this section, we present three lemmas that are essential for the proof of counterfactual identifiability results in §5.

B.2.1. The Markovian Case

Lemma B.2. *BGMs f and $\hat{f}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$ are equivalent if*

1. *(Markovian) $U \perp\!\!\!\perp \boldsymbol{X}$ and $\hat{U} \perp\!\!\!\perp \boldsymbol{X}$.*
2. *for all $\boldsymbol{x}$, $f(\boldsymbol{x}, \cdot)$ and $\hat{f}(\boldsymbol{x}, \cdot)$ are either strictly increasing or strictly decreasing functions.*

Proof. We will show that BGMs f and $\hat{f}$ produce the same counterfactuals. Using Proposition 6.2, we conclude their equivalence. Suppose we are interested in the counterfactual query $V_{\boldsymbol{x}'}|\boldsymbol{X}=\boldsymbol{x}, V=v$. Without loss of generality, consider only $\boldsymbol{x}, v$ samples for which $P_{\boldsymbol{X},V}(\boldsymbol{x}, v) > 0$. Let $F(\boldsymbol{x}, v) \coloneqq P(V \leq v|\boldsymbol{X}=\boldsymbol{x})$ be the conditional Cumulative Distribution Function (CDF) and $F^{-1}(\boldsymbol{x}, \alpha)$ the quantile function, which exists where $P_{\boldsymbol{X},V}(\boldsymbol{x}, v) > 0$. In the abduction

step of counterfactual estimation, both BGMs f and $\hat{f}$ will return the $F(\boldsymbol{x}, v)^{\text{th}}$ or $\Big(1 - F(\boldsymbol{x}, v)^{\text{th}}\Big)$ quantile of their corresponding exogenous distribution $\Big(P(U|\boldsymbol{X})$ and $P(\hat{U}|\boldsymbol{X})\Big)$ if they are both increasing, or decreasing, respectively. These quantiles might in fact be two distinct values. As $U, \hat{U} \perp\!\!\!\perp \boldsymbol{X}$, the quantile function of $U, \hat{U}$ given $\boldsymbol{X}$ is independent of $\boldsymbol{X}$, and equal to the quantile function of the marginal $U, \hat{U}$. Hence, the action step would not change the estimated quantile. In the prediction step, both BGMs f and $\hat{f}$ would estimate the same value $F^{-1}\Big(\boldsymbol{x'}, F(\boldsymbol{x}, v)\Big)$ if increasing, or other similar quantile variations if decreasing. □

B.2.2. Instrumental Variable (IV)

Lemma B.3. *For $\boldsymbol{X} \in \mathbb{X} \triangleq \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$ and $\boldsymbol{I} \in \mathbb{I} \triangleq \{\boldsymbol{i}_1, \dots, \boldsymbol{i}_n\}$, BGMs f and $\hat{f}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$ are equivalent if*

1. *(IV) $\boldsymbol{I} \perp\!\!\!\perp U$ and $\boldsymbol{I} \perp\!\!\!\perp \hat{U}$.*
2. *for all $\boldsymbol{x} \in \mathbb{X}$, $f^{-1}(\boldsymbol{x}, \cdot)$ and $\hat{f}(\boldsymbol{x}, \cdot)$ are either strictly increasing or strictly decreasing, and two times differentiable.*
3. *$P_{\hat{U}}(\cdot)$ is differentiable.*
4. *$P_{\mathcal{D}}(\boldsymbol{i}, \boldsymbol{x}, \cdot)$ is differentiable for every $\boldsymbol{i}, \boldsymbol{x}$.*
5. *(Positivity) $\forall u, \hat{u}, \boldsymbol{x} \in \mathbb{X} : P_{U,\boldsymbol{X}}(u, \boldsymbol{x}) > 0$ and $P_{\hat{U},\boldsymbol{X}}(\hat{u}, \boldsymbol{x}) > 0$.*
6. *(Variability) $\forall u : |\det \boldsymbol{M}_{\mathcal{D}}(u, \mathbb{I})| \geq c$, where c is a positive constant and*

$$\boldsymbol{M}_{\mathcal{D}}(u, \mathbb{I}) \triangleq \begin{bmatrix} P_{\mathcal{D}}(\boldsymbol{x}_1|u, \boldsymbol{i}_1) & \dots & P_{\mathcal{D}}(\boldsymbol{x}_n|u, \boldsymbol{i}_1) \\ \vdots & \ddots & \vdots \\ P_{\mathcal{D}}(\boldsymbol{x}_1|u, \boldsymbol{i}_n) & \dots & P_{\mathcal{D}}(\boldsymbol{x}_n|u, \boldsymbol{i}_n) \end{bmatrix} \tag{16}$$

Proof. Define Cumulative Distribution Functions (CDF) $k(u) = P(U \leq u)$, $\hat{k}(\hat{u}) = P(\hat{U} \leq \hat{u})$. We use the CDFs to transform random variables into uniform distributions between 0 and 1. Define $Z = k(U)$, $\hat{Z} = \hat{k}(\hat{U})$. Due to Probability Integral transform we have $Z, \hat{Z} \sim Unif(0, 1)$. Furthermore $\forall \boldsymbol{x} \in \mathbb{X} : Z = s(\boldsymbol{x}, \hat{Z})$ where $s(\boldsymbol{x}, \cdot) = k(\cdot) \circ f^{-1}(\boldsymbol{x}, \cdot) \circ \hat{f}(\boldsymbol{x}, \cdot) \circ \hat{k}^{-1}(\cdot)$. Because $U \perp\!\!\!\perp \boldsymbol{I}$ (The first condition), and because Z is a deterministic function of U, we conclude $Z \perp\!\!\!\perp \boldsymbol{I}$. Using a similar argument, $\hat{Z} \perp\!\!\!\perp \boldsymbol{I}$.

$$\forall \hat{z}, \boldsymbol{i} : P_{\hat{Z}|\boldsymbol{I}}(\hat{z}|\boldsymbol{i}) = 1 \tag{17}$$

$$\rightarrow \forall \hat{z}, \boldsymbol{i} : \sum_{\ell=1}^{n} P_{\hat{Z},\boldsymbol{X}|\boldsymbol{I}}(\hat{z}, \boldsymbol{x}_\ell|\boldsymbol{i}) = 1 \tag{18}$$

$$\rightarrow \forall \hat{z}, \boldsymbol{i} : \sum_{\ell=1}^{n} P_{\hat{Z}|\boldsymbol{X},\boldsymbol{I}}(\hat{z}|\boldsymbol{x}_\ell, \boldsymbol{i}) P_{\boldsymbol{X}|\boldsymbol{I}}(\boldsymbol{x}_\ell|\boldsymbol{i}) = 1 \tag{19}$$

Using conditions $2, 3, 4, 6$ we know that $\forall \boldsymbol{x} \in \mathbb{X} : s(\boldsymbol{x}, \cdot)$ is strictly increasing and two times differentiable. Hence, using the change of variable formula we have:

$$\rightarrow \forall \hat{z}, \boldsymbol{i} : \sum_{\ell=1}^{n} P_{Z|\boldsymbol{X},\boldsymbol{I}}(s(\boldsymbol{x}_\ell, \hat{z})|\boldsymbol{x}_\ell, \boldsymbol{i}) P_{\boldsymbol{X}|\boldsymbol{I}}(\boldsymbol{x}_\ell|\boldsymbol{i}) \frac{\partial s(\boldsymbol{x}_\ell, \hat{z})}{\partial \hat{z}} = 1 \tag{20}$$

Using $Z \perp\!\!\!\perp \boldsymbol{I}$ we have:

$$P_{Z|\boldsymbol{X},\boldsymbol{I}}(z|\boldsymbol{x}, \boldsymbol{i}) P_{\boldsymbol{X}|\boldsymbol{I}}(\boldsymbol{x}|\boldsymbol{i}) = P_Z(z) P_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}|z, \boldsymbol{i}) \tag{21}$$

$Z \sim Unif(0, 1)$ thus

$$\rightarrow P_{Z|\boldsymbol{X},\boldsymbol{I}}(z|\boldsymbol{x}, \boldsymbol{i}) P_{\boldsymbol{X}|\boldsymbol{I}}(\boldsymbol{x}|\boldsymbol{i}) = P_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}|z, \boldsymbol{i}) \tag{22}$$

Now we combine Equation (20) and Equation (22):

$$\rightarrow \forall \hat{z}, \boldsymbol{i} : \sum_{\ell=1}^{n} P_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_\ell | s(\boldsymbol{x}_\ell, \hat{z}), \boldsymbol{i}) \frac{\partial s(\boldsymbol{x}_\ell, \hat{z})}{\partial \hat{z}} = 1 \tag{23}$$

We can write Equation (23) in vector product form as:

$$\forall \hat{z}, \boldsymbol{i} : \begin{bmatrix} P_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_1 | s(\boldsymbol{x}_1, \hat{z}), \boldsymbol{i}) & \dots & P_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_n | s(\boldsymbol{x}_n, \hat{z}), \boldsymbol{i}) & -1 \end{bmatrix} \begin{bmatrix} \frac{\partial s(\boldsymbol{x}_1, \hat{z})}{\partial \hat{z}} \\ \vdots \\ \frac{\partial s(\boldsymbol{x}_n, \hat{z})}{\partial \hat{z}} \\ 1 \end{bmatrix} = 0 \tag{24}$$

Now we combine all n equations of this form for $\boldsymbol{i} \in \mathbb{I}$ in matrix format:

$$\forall \hat{z} : \begin{bmatrix} p_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_1 | s(\boldsymbol{x}_1, \hat{z}), \boldsymbol{i}_1) & \dots & p_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_n | s(\boldsymbol{x}_n, \hat{z}), \boldsymbol{i}_1) & -1 \\ \vdots & \vdots & \vdots & \vdots \\ p_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_1 | s(\boldsymbol{x}_1, \hat{z}), \boldsymbol{i}_n) & \dots & p_{\boldsymbol{X}|Z,\boldsymbol{I}}(\boldsymbol{x}_n | s(\boldsymbol{x}_n, \hat{z}), \boldsymbol{i}_n) & -1 \end{bmatrix} \begin{bmatrix} \frac{\partial s(\boldsymbol{x}_1, \hat{z})}{\partial \hat{z}} \\ \vdots \\ \frac{\partial s(\boldsymbol{x}_n, \hat{z})}{\partial \hat{z}} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \tag{25}$$

As argued above, $\forall \boldsymbol{x} \in \mathbb{X} : s(\boldsymbol{x}, \cdot)$ is strictly increasing and two times differentiable. Combining this with the positivity assumption we have

$$\forall \boldsymbol{x} \in \mathbb{X} : s(\boldsymbol{x}, 0) = 0 \tag{26}$$

Consider Equation (25) in $\hat{z} = 0$. The first matrix's rank is n because of the variability condition, and has $n + 1$ columns. So its nullspace has the form $k \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}^\intercal$. This implies that

$$\forall \boldsymbol{x} \in \mathbb{X} : \frac{\partial s(\boldsymbol{x}, \hat{z})}{\hat{z}}|_{\hat{z}=0} = 1. \tag{27}$$

Next, we divide $[0, 1]$ into N pieces. We prove by induction on m that for large enough N

$$\forall m \in \{1, \dots, N\}, \boldsymbol{x} \in \mathbb{X} : |s(\boldsymbol{x}, \frac{m}{N}) - \frac{m}{N}| \leq \frac{m}{2N^2} B \tag{28}$$

where $B = \max_{\boldsymbol{x}, \hat{z}} \frac{\partial^2 s(\hat{z}, \boldsymbol{x})}{\partial \hat{z}}$.

For $m = 1$, using Taylor's theorem we have

$$\exists \xi \in [0, \frac{1}{N}] : s(\boldsymbol{x}, \frac{1}{N}) = s(\boldsymbol{x}, 0) + \frac{1}{N} \frac{\partial s(\boldsymbol{x}, \hat{z})}{\partial \hat{z}}|_{\hat{z}=0} + \frac{1}{2N^2} \frac{\partial^2 s(\boldsymbol{x}, \hat{z})}{\partial \hat{z}}|_{\hat{z}=\xi} \tag{29}$$

Using Equations (26) and (27)

$$\rightarrow \exists \xi \in [0, \frac{1}{N}] : s(\frac{1}{N}, \boldsymbol{x}) = \frac{1}{N} + \frac{1}{2N^2} \frac{\partial^2 s(\boldsymbol{x}, \hat{z})}{\partial \hat{z}}|_{\hat{z}=\xi} \tag{30}$$

$$\rightarrow |s(\boldsymbol{x}, \frac{1}{N}) - \frac{1}{N}| \leq \frac{B}{2N^2} \tag{31}$$

Now suppose that Equation (28) holds for m, we will show that it holds for $m + 1$ too. We know that $s(\boldsymbol{x}, \cdot)$ is differentiable. Combining it with assumption 6 implies that each element of matrix M is differentiable with respect to $\hat{z}$. Furthermore, determinant is a polynomial function of all elements of M which is differentiable with respect to every element. Thus determinant of M is differentiable with respect to $\hat{z}$. This means that if we perturb $\hat{z}$ in each element of M by a sufficiently small amount, M will remain full rank. As a result, for a large enough N the left matrix in Equation (25)'s rank is still n, and its null space is still one-dimensional. This means that

$$\forall \boldsymbol{x} \in \mathbb{X} : \frac{\partial s(\boldsymbol{x}, \hat{z})}{\hat{z}}|_{\hat{z}=\frac{m}{N}} = 1. \tag{32}$$

using Taylor's theorem we have

$$\exists \xi \in [\frac{m}{N}, \frac{m+1}{N}] : s(\boldsymbol{x}, \frac{m+1}{N}) = s(\boldsymbol{x}, \frac{m}{N}) + \frac{1}{N}\frac{\partial s(\boldsymbol{x}, \hat{z})}{\partial \hat{z}}|_{\hat{z}=\frac{m}{N}} + \frac{1}{2N^2}\frac{\partial^2 s(\boldsymbol{x}, \hat{z})}{\partial \hat{z}}|_{\hat{z}=\xi} \tag{33}$$

Using Equation (28) for m and Equation (32) we conclude that

$$|s(\boldsymbol{x}, \frac{m+1}{N}) - \frac{m+1}{N}| \leq \frac{m+1}{2N^2}B \tag{34}$$

This concludes the proof of Equation (28) by induction for all values of $m \in \{1, \cdots, N\}$. In other words, function $s(\boldsymbol{x}, \cdot)$ can get as close as wanted to identity in all points $\frac{m}{N}$. Using this and the fact that $\forall \boldsymbol{x} \in \mathbb{X} : s(\boldsymbol{x}, \cdot)$ is differentiable implies that $\forall \boldsymbol{x} \in \mathbb{X} : s(\boldsymbol{x}, \hat{z}) = \hat{z}$. This concludes the proof with $g(\cdot) = k(\cdot) \circ \hat{k}^{-1}(\cdot)$. □

B.2.3. BACKDOOR CRITERION (BC)

Lemma B.4. *BGMs f and $\hat{f}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$ are equivalent if*

1. *(BC) $U \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$ and $\hat{U} \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$.*
2. *For every $\boldsymbol{x}$: $\nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{f^{-1}(\boldsymbol{x}, \cdot)}|$ and $\nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{\hat{f}(\boldsymbol{x}, \cdot)}|$ both exist.*
3. *(Variability) $\forall u$: Instances $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}$ exist such that $|\det \boldsymbol{M}_{\mathcal{D}}(u, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1})| > 0$, where*

$$\boldsymbol{M}_{\mathcal{D}}(u, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}) \triangleq \begin{bmatrix} P_{\mathcal{D}}(u|\boldsymbol{z}_1) & \nabla_u P_{\mathcal{D}}(u|\boldsymbol{z}_1) \\ \vdots & \vdots \\ P_{\mathcal{D}}(u|\boldsymbol{z}_{d+1}) & \nabla_u P_{\mathcal{D}}(u|\boldsymbol{z}_{d+1}) \end{bmatrix} \tag{35}$$

Proof. Define $g(\boldsymbol{x}, \cdot) \triangleq f^{-1}(\boldsymbol{x}, \cdot) \circ \hat{f}(\boldsymbol{x}, \cdot)$. Using the change of variable formula, we get

$$\forall \boldsymbol{x}, \hat{u}, \boldsymbol{z} : P_{\hat{U}|\boldsymbol{Z}, \boldsymbol{X}}(\hat{u}|\boldsymbol{z}, \boldsymbol{x}) = P_{U|\boldsymbol{Z}, \boldsymbol{X}}\Big(g(\boldsymbol{x}, \hat{u})|\boldsymbol{z}, \boldsymbol{x}\Big)|\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}| \tag{36}$$

$$\text{(BC)} \Rightarrow \forall \boldsymbol{x}, \hat{u}, \boldsymbol{z} : P_{\hat{U}|\boldsymbol{Z}(\hat{u}|\boldsymbol{z})} = P_{U|\boldsymbol{Z}}\Big(g(\boldsymbol{x}, \hat{u})|\boldsymbol{z}\Big)|\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}| \tag{37}$$

Using chain rule of derivatives, we know that $|\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}| = |\det \boldsymbol{J}_{f^{-1}(\boldsymbol{x}, \cdot)}||\det \boldsymbol{J}_{\hat{f}(\boldsymbol{x}, \cdot)}|$ which is differentiable with respect to $\boldsymbol{x}$ according to condition 2. By differentiating Equation (37) with respect to the i^{th} element in $\boldsymbol{x}$ (x_i) we get

$$\nabla_u P_{U|\boldsymbol{Z}}\Big(g(\boldsymbol{x}, \cdot)|\boldsymbol{z}\Big) \begin{bmatrix} \frac{g_1(\boldsymbol{x}, \hat{u})}{\partial x_i} \\ \vdots \\ \frac{g_d(\boldsymbol{x}, \hat{u})}{\partial x_i} \end{bmatrix} |\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}| + P_{U|\boldsymbol{Z}}\Big(g(\boldsymbol{x}, \hat{u})|\boldsymbol{z}\Big)\frac{\partial |\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}|}{\partial x_i} = 0 \tag{38}$$

$$\Rightarrow \begin{bmatrix} \nabla_u P_{U|\boldsymbol{Z}}\Big(g(\boldsymbol{x}, \cdot)|\boldsymbol{z}\Big) & P_{U|\boldsymbol{Z}}\Big(g(\boldsymbol{x}, \hat{u})|\boldsymbol{z}\Big) \end{bmatrix} \begin{bmatrix} \frac{g_1(\boldsymbol{x}, \hat{u})}{\partial x_i} \\ \vdots \\ \frac{g_d(\boldsymbol{x}, \hat{u})}{\partial x_i} \\ \frac{\partial |\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}|}{\partial x_i} \end{bmatrix} = 0 \tag{39}$$

Stacking the equations for $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}$ we get

$$\boldsymbol{M}_{\mathcal{D}}(u, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}) \begin{bmatrix} \frac{\partial |\det \boldsymbol{J}_{g(\boldsymbol{x}, \cdot)}|}{\partial x_i} \\ \frac{g_1(\boldsymbol{x}, \hat{u})}{\partial x_i} \\ \vdots \\ \frac{g_d(\boldsymbol{x}, \hat{u})}{\partial x_i} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \tag{40}$$

Since the square matrix is full-rank due to variability condition, all elements of the vector must be zero. This means that $g(\boldsymbol{x}, \hat{u})$ does not depend on x_i. Iterating the same argument for all $i \in \{1, \ldots, d\}$ we conclude that $g(\boldsymbol{x}, \hat{u})$ does not depend on $\boldsymbol{x}$ which concludes the proof. □

B.3. Theorem 5.1

BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V}$ if

1. (Markovian) $U \perp\!\!\!\perp \boldsymbol{X}$.
2. for all $\boldsymbol{x}$, $f(\boldsymbol{x}, \cdot)$ is either a strictly increasing or a strictly decreasing function.

Proof. Let $\mathbb{F}$ be the class of BGMs that satisfy theorem's conditions. Consider any two BGMs $\hat{f}, f \in \mathbb{F}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$. Using Lemma B.2 we conclude their equivalence. As a result, they produce the same counterfactuals (Proposition 6.2), which establishes identifiability according to its definition in §2. □

B.3.1. Independence assumption is not sufficient by itself for counterfactual identifiability

In this section, we use a simple example to demonstrate that the independence assumption alone (without the monotonicity assumption) is not enough for BGM identification. This example is taken from Nasr-Esfahany & Kiciman (2023, Sec. 3) Consider the following two simple BGMs f and $\hat{f}$:

$$X \sim Bernoulli(0.5),\ \ U \sim Unif(0,1),\ X \perp\!\!\!\perp U,\ f = \begin{cases} U, & X = 1 \\ U - 1, & X = 0 \end{cases},\ \hat{f} = \begin{cases} U, & X = 1 \\ -U, & X = 0 \end{cases} \tag{41}$$

Note that f and $\hat{f}$ generate the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$, and they satisfy the first (independence) constraint. However, they give different answers to counterfactual queries. Consider the following counterfactual query: $V_1|X = 0, V = v$. f and $\hat{f}$ give $v + 1$ and $-v$ as answers, respectively.

B.4. Theorem 5.2

For $\boldsymbol{X} \in \mathbb{X} \triangleq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and $\boldsymbol{I} \in \mathbb{I} \triangleq \{\boldsymbol{i}_1, \ldots, \boldsymbol{i}_n\}$, BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V,\boldsymbol{I}}$ if

1. (IV) $\boldsymbol{I} \perp\!\!\!\perp U$.
2. for all $\boldsymbol{x} \in \mathbb{X}$, $f(\boldsymbol{x}, \cdot)$ and $f^{-1}(\boldsymbol{x}, \cdot)$ are either strictly increasing or strictly decreasing, and two times differentiable.
3. $P(\boldsymbol{i}, \boldsymbol{x}, \cdot)$ is differentiable for every $\boldsymbol{i} \in \mathbb{I}, \boldsymbol{x} \in \mathbb{X}$.
4. (Positivity) $\forall u, \boldsymbol{x} \in \mathbb{X} : P_{U,\boldsymbol{X}}(u, \boldsymbol{x}) > 0$.
5. (Variability) $\forall u : |\det \boldsymbol{M}(u, \mathbb{I})| \geq c > 0$, where

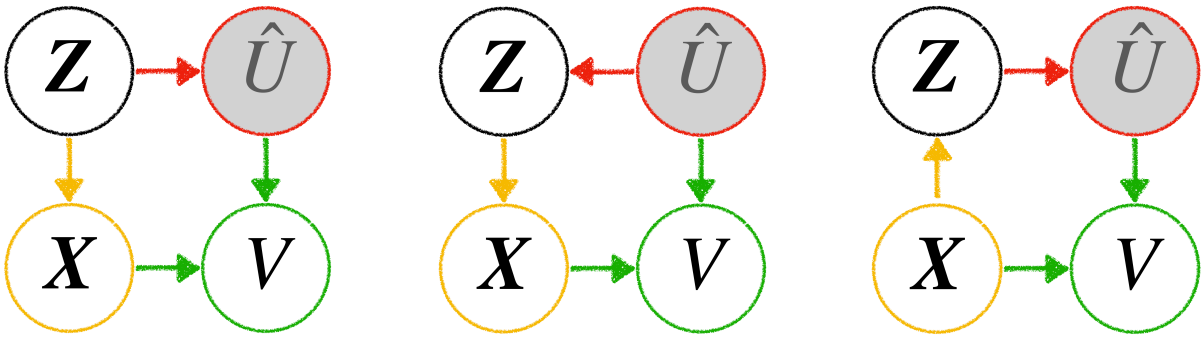
$$\boldsymbol{M}(u, \mathbb{I}) \triangleq \begin{bmatrix} P(\boldsymbol{x}_1|u,\boldsymbol{i}_1) & \ldots & P(\boldsymbol{x}_n|u,\boldsymbol{i}_1) \\ \vdots & \ddots & \vdots \\ P(\boldsymbol{x}_1|u,\boldsymbol{i}_n) & \ldots & P(\boldsymbol{x}_n|u,\boldsymbol{i}_n) \end{bmatrix}$$

Proof. Let $\mathbb{F}$ be the class of BGMs that satisfy theorem's conditions. Consider any two BGMs $\hat{f}, f \in \mathbb{F}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$. Using Lemma B.3 we conclude their equivalence. As a result, they produce the same counterfactuals (Proposition 6.2), which establishes identifiability according to its definition in §2. □

B.5. Theorem 5.3

BGM f is counterfactually identifiable given $P_{\boldsymbol{X},V,\boldsymbol{Z}}$ if

1. (BC) $U \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$.
2. $\forall \boldsymbol{x} : \nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{f(\boldsymbol{x},\cdot)}|$ and $\nabla_{\boldsymbol{x}} |\det \boldsymbol{J}_{f^{-1}(\boldsymbol{x},\cdot)}|$ exist.



(a) (b) (c)

Figure 5. Markovian equivalence class of directed graphical models for the backdoor criterion (BC) case

3. (Variability) $\forall u$: Instances $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}$ exist such that $|\det \boldsymbol{M}(u, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1})| > 0$, where

$$\boldsymbol{M}(u, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_{d+1}) \triangleq \begin{bmatrix} P(u|\boldsymbol{z}_1) & \nabla_u P(u|\boldsymbol{z}_1) \\ \vdots & \vdots \\ P(u|\boldsymbol{z}_{d+1}) & \nabla_u P(u|\boldsymbol{z}_{d+1}) \end{bmatrix}$$

Proof. Let $\mathbb{F}$ be the class of BGMs that satisfy theorem's conditions. Consider any two BGMs $\hat{f}, f \in \mathbb{F}$ that produce the same distribution $P_{\mathcal{D}}(\boldsymbol{X}, V)$. Using Lemma B.4 we conclude their equivalence. As a result, they produce the same counterfactuals (Proposition 6.2), which establishes identifiability according to its definition in §2. □

C. Directed Graphical Models

We can determine (Conditional) independencies of a joint distribution from its directed graphical model using d-seperation.

C.1. Backdoor Criterion (BC)

The distributional requirement we have in this case (§5.3) is that $\boldsymbol{X} \perp\!\!\!\perp U|\boldsymbol{Z}$. The structured generative network we build for this case (Figure 3d) resembles the structure of the directed graphical model shown in Figure 3b. As a result, it inherits all the conditional independence properties read off the graphical model using the d-separation test (Pearl, 1988; Koller & Friedman, 2009). There are two paths between U and $\boldsymbol{X}$, $\boldsymbol{X} \leftarrow \boldsymbol{Z} \rightarrow U$ which is blocked when we condition on $\boldsymbol{Z}$, and $\boldsymbol{X} \rightarrow V \leftarrow U$ which is blocked due to the v-structure at V, thus $\boldsymbol{X} \perp\!\!\!\perp U|\boldsymbol{Z}$.

Alternative viable graphical models: It is worth emphasizing that the directed graphical model used for constructing the structured generative network is purely a statistical object as opposed to a causal DAG, and its goal is solely to equip the structured generative network with the corresponding distributional constraint in each case. As a result, all directed graphical models in the Markovian equivalence class (directed graphical models that encode the same conditional independencies) of Figure 3b are valid and can be used for constructing alternative structured generative network.

To create the Markovian equivalence class, we should keep the graph's skeleton fixed, and flip the edges without creating new or removing existing v-structures. Figure 5 shows the three possible options, each of which we can use to construct a valid structured generative network.

C.2. Instrumental Variable (IV)

The structured generative network in this case is depicted in Figure 3c. It is designed to follow the directed graphical model in Figure 3a in which the two paths between $\boldsymbol{I}$ and U, $\boldsymbol{I} \rightarrow \boldsymbol{X} \leftarrow U$ and $\boldsymbol{I} \rightarrow \boldsymbol{X} \rightarrow V \leftarrow U$ are blocked by open v-structures (unconditioned X and $\boldsymbol{V}$, respectively) which implies independence of $\boldsymbol{I}$ and V As a result, the distribution produced by the structured generative network is guaranteed to satisfy the distributional constraint $U \perp\!\!\!\perp V$ (the first condition in Theorem 5.2), by construction.

In this setting, the graphical model shown in Figure 3a is the only member of its Markovian equivalence class, as flipping

any edges would change the set of (conditional) independencies.

D. Normalizing Flows (NF)

Normalizing Flows (NF) are a class of generative models with tractable distributions where both sampling and density estimation are efficient and exact. They model the data (V) as a transformation (T) of some noise variable (U) sampled from a simple base distribution (P_U), e.g., Gaussian distribution, where T is a diffeomorphism.[9] This allows for the density of V to be obtained via a change of variables:

$$P_V(v) = P_U\Big(T^{-1}(v)\Big)|\det \boldsymbol{J}_{T^{-1}}(v)| \tag{42}$$

The transform T can be tractably optimized to fit the observed distribution of V. Designing expressive transformation families with efficient inverse and Jacobin has thus been subject to research (Kingma & Dhariwal, 2018; Chen et al., 2019; Meng et al., 2022). This idea can be easily extended for modeling conditional distributions (Trippe & Turner, 2018; Winkler et al., 2019; Lu & Huang, 2020), e.g., $P_{V|\boldsymbol{X}}$ with conditional normalizing flows (CNF), by parameterizing the transform T as a function of the condition ($\boldsymbol{x}$). Refer to Kobyzev et al. (2020); Papamakarios et al. (2021) for an extensive survey of NFs.

E. Experiments

Implementation Details: We build all CGMs using NFs with linear rational splines (Dolatabadi et al., 2020) and train them with likelihood maximization using their implementation in Pyro (Bingham et al., 2018). All splines we use have 16 bins for mapping $(-3, +3)$ to $(-3, +3)$. We use affine transforms at input and output layers to calibrate the range. Condition networks are all MLPs with two hidden layers, each with 64 units. We use batch size of 2^{20} and run all experiments using A100 GPUs. We train all models using the default implementation of Adam (Kingma & Ba, 2015) in Pytorch (Paszke et al., 2019).

Empirical Relaxation of Theoretical Assumptions: Linear rational splines, although differentiable, are not necessarily two times differentiable. Two times differentiability is required in the IV case (§5.2) by the second assumption of Theorem 5.2. Furthermore, the condition network we use to condition the spline parameters based on $\boldsymbol{x}$ uses ReLU activation function (Agarap, 2018) which implies non-differentiability of our CGMs with respect to x, which is required in the IV case (§5.3) by the second assumption of Theorem 5.3. We can satisfy both of these assumptions, e.g., by using quadratic splines (Durkan et al., 2019) or GElU activations (Hendrycks & Gimpel, 2016). However, good performance in §8 suggests that these technical assumptions might not be tight, and can be relaxed. We leave their relaxation to future work.

E.1. Counterfactual Ellipse Generation

We use the following SCM for generation of the ellipse dataset:

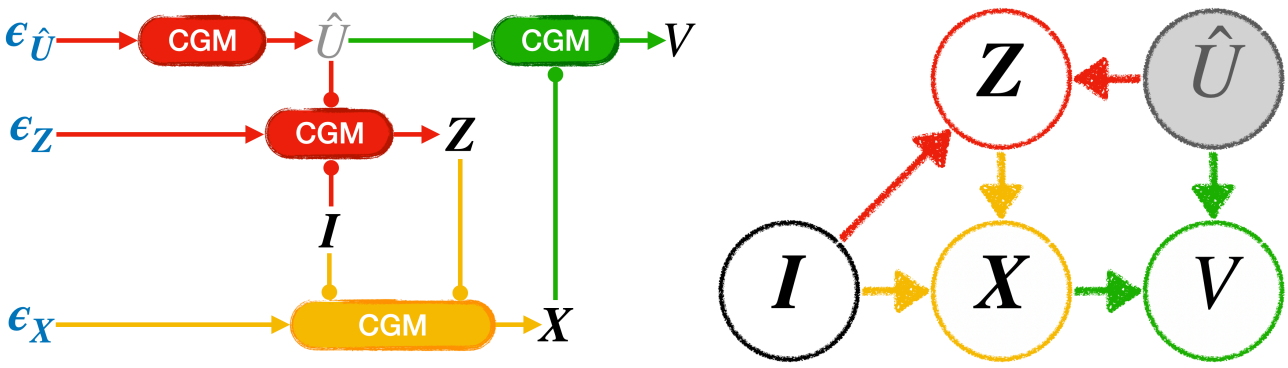
$$Z \coloneqq \epsilon_z,\ \epsilon_z \sim \text{Unif}(-0.5, 0.5) \tag{43}$$
$$X \coloneqq (1.44254843z + 0.59701923 + \epsilon_x)\ \%\ (2\pi),\ \epsilon_x \sim \text{Normal}(0, 1) \tag{44}$$
$$U_0 \coloneqq e^{1.64985274z + 0.2656131} + \epsilon_{u_0},\ \epsilon_{u_0} \sim \text{Beta}(1, 1) \tag{45}$$
$$U_1 \coloneqq U_0(1 + \epsilon_{u_1} e^{1.61323358z - 0.18070237}),\ \epsilon_{u_1} \sim \text{Exponential}(1) \tag{46}$$
$$V_0 \coloneqq U_0\Big(2 + \sin(X)\Big) \tag{47}$$
$$V_1 \coloneqq U_1\Big(2 + \cos(X)\Big) \tag{48}$$
$$\tag{49}$$

We use a sequence of three Spline transforms with coupling for all schemes.

E.1.1. Failure in the Markovian Case

In the ellipse generation taks, both U and V are two dimensional. To empirically evaluate whether or not we can lean multi-dimensional BGMs in the Markovian case, we generated a second dataset by randomly shuffling X in the previous

[9] A differentiable transform with a differentiable inverse.



(a) Structured Generative Network (b) Directed Graphical Model

Figure 6. Simultaneous Exploitation of Instrumental Variable (IV) and Backdoor Criterion (BC)

dataset. We trained a single CGM (similar to the Markovian case in §6), and used it for counterfactual estimation where it failed (MAPE $= 607$).

E.2. Video Streaming Simulation

We got the simulator and the ABR algorithms' implementations from Alomar et al. (2023). Appendix. D in this work explains the details. We use a single conditional spline for every CGM.

E.2.1. Simultaneous Exploitation of Instrumental Variable (IV) and Backdoor Criterion (BC)

Figure 6b depicts the directed graphical model we use to represent the necessary (conditional) independencies in this case. In this graphical model, all paths between $\boldsymbol{I}$ and $\hat{U}$ are blocked so $\boldsymbol{I} \perp\!\!\!\perp \hat{U}$ which is the distributional objective of the IV case (§5.2). Furthermore, conditioning on $(\boldsymbol{Z}, \boldsymbol{I})$ blocks all paths between $\boldsymbol{X}$ and U hence $\boldsymbol{X} \perp\!\!\!\perp U|(\boldsymbol{I}, \boldsymbol{Z})$. This is the distributional objective in the BC case (§5.3), where the pair $(\boldsymbol{I}, \boldsymbol{Z})$ satisfies the backdoor criterion.