Efficient Causal Inference from Combined Observational and Interventional Data through Causal Reductions

Maximilian Ilse[1] **Patrick Forré**[1] **Max Welling**[1] **Joris M. Mooij**[2]

[1]Amsterdam Machine Learning Lab, University of Amsterdam, Amsterdam, the Netherlands
[2]Korteweg-De Vries Institute for Mathematics, University of Amsterdam, Amsterdam, the Netherlands

Abstract

Unobserved confounding is one of the main challenges when estimating causal effects. We propose a novel causal reduction method that replaces an arbitrary number of possibly high-dimensional latent confounders with a single latent confounder that lives in the same space as the treatment variable without changing the observational and interventional distributions entailed by the causal model. After the reduction, we parameterize the reduced causal model using a flexible class of transformations, so-called normalizing flows. We propose a learning algorithm to estimate the parameterized reduced model jointly from observational and interventional data. This allows us to estimate the causal effect in a principled way from combined data. We perform a series of experiments on data simulated using nonlinear causal mechanisms and find that we can often substantially reduce the number of interventional samples when adding observational training samples without sacrificing accuracy. Thus, adding observational data may help to more accurately estimate causal effects even in the presence of unobserved confounders.

1 INTRODUCTION

Estimating causal effects of interventions is one of the fundamental problems in causal inference. The gold standard for studying causal relationships between interventions and outcomes are controlled experiments, for example in the form of randomized controlled trials (RCTs) in medicine or A/B-testing in psychology. However, the acquisition of experimental data is often time consuming, costly, or comes with logistic difficulties and ethical issues. To give an idea about the economic relevance within the discipline of medicine alone, according to Grand Review Research, "the global clinical trials market size was estimated at 44.3 billion US dollars in 2020". The high costs of clinical trials and, more importantly, the low availability of diseased subjects for such trials present frequent obstacles for the development of drugs for rare diseases by pharmaceutical companies.

In this work, we propose a novel principled approach for causal effect estimation that can efficiently combine observational and interventional samples. We show that this method can potentially reduce the required RCT sample size when sufficient observational samples are available (e.g., in the form of electronic health records). Recent real-world examples that could benefit from such an approach are the COVID-19 vaccine trials. The majority of the vaccines require two dosages. For example, the interval during the vaccine trials was 21 days between doses for the Pfizer vaccine and 28 days for the Moderna vaccine. However, due to a shortage of supplies and logistical challenges the second dosage is delayed in many countries. The question then arises: What is the effect of the time between the first and the second dosage on the vaccine efficacy? In the absence of any large randomized controlled trials that provide a definite answer to this question, one may hope to estimate this by combining the few available clinical trial data with massive global observational data collected as a part of the different vaccination campaigns performed worldwide. The method we propose here provides a principled approach for such causal inference problems.

The main complication when estimating causal effects is the potential presence of observed and unobserved *confounders*, i.e., common causes of the cause and the effect. Our key technical contribution, which we believe to be a useful tool on its own, is a construction that typically *reduces* the size of the latent confounder space in a structural causal model (or causal Bayesian network with latent confounders). Our approach makes only mild assumptions, namely the absence of causal feedback between cause and effect, and that the data was not subject to selection bias (due to implicit conditioning on common effects of treatment and outcome).

Submitted to the 37th Conference on Uncertainty in Artificial Intelligence (UAI 2021). **To be used for reviewing only**.

This *causal reduction* operation shows that without loss of generality, one only needs to model a single latent confounding variable that lives in the same space as the treatment variable, even if in reality there could be many latent confounders and their joint space might be much larger. In particular, for a real-valued, one-dimensional treatment variable, a real-valued, one-dimensional confounder suffices. This is a key step towards a parsimonious joint parameterization of the observational and interventional distributions.

For the linear-Gaussian case, we prove that our reduced parameterization implies that the observational and interventional distributions are not independent, but are related by certain equality constraints. This complements existing work on inequality constraints in the case of discrete treatment and outcome variables (Bell [1964], Balke and Pearl [1997], Wolfe et al. [2019]). We conjecture that such dependencies between the observational and interventional distribution hold more generally (i.e., not only in the linear-Gaussian or discrete settings), and provide empirical support for this conjecture.

To make progress in the general non-linear setting, we parameterize the reduced causal model using a flexible class of transformations, so-called normalizing flows [Tabak and Turner, 2013, Rezende and Mohamed, 2015]. This enables the use of a simple multi-task-like maximum-likelihood approach to the estimation of the reduced model parameters, where one can now combine observational and interventional training data.

We perform a series of experiments on data simulated using nonlinear causal mechanisms. We find that we can significantly reduce the number of interventional samples required to achieve a certain fit when adding sufficient observational training samples. We observe that parameter sharing allows one to learn a more accurate model from a combination of data than when learning from either of the two subsets individually. This suggests that this approach successfully exploits the conjectured dependence between the observational and interventional distributions, and opens up practical applications and further theoretical questions regarding the precise nature of the relationship between observational and interventional distributions.

In summary, our three main contributions are: (i) A causal reduction method that replaces arbitrary confounders with a single confounder that lives in the same space as the treatment variable, without changing the observational and interventional distributions entailed by the causal model; (ii) A flexible parameterization of the reduced model using normalizing flows, which enables us to estimate the observational and interventional distributions by jointly learning from observational and interventional data without making strong parametric assumptions; (iii) A derivation of equality constraints between interventional and observational distributions entailed by linear Gaussian causal models.

2 RELATED WORK

Prior work on combining observational and interventional data by Rosenman et al. [2018] relies on the assumptions that all confounders are measured. This assumption was removed in Rosenman et al. [2020], however, their new method requires a minimum of four strata and shared average treatment effects. Furthermore, Kallus et al. [2018] make the additional assumption that the hidden confounder has a parametric structure that can be modeled effectively and Athey et al. [2020] depend on observed short-term and long-term outcome variables. In contrast, our approach makes only mild assumptions, namely the absence of causal feedback between cause and effect, and that the data was not subject to selection bias.

There exists a plethora of related work on estimating causal effects solely from observational data. The majority of methods assume that there exists a set of observed variables that can be used to adjust for all confounding factors [Colnet et al., 2020]. Unfortunately one can never test this assumption, and the reliability of the conclusions of such observational studies is debated [Madigan et al., 2014]. While much work is focused on the case in which the treatment is binary, Hirano and Imbens [2005] generalize the propensity score for continuous treatment variables.

An approach that sidesteps the strong untestable assumption of no unobserved confounding is to bound the causal effect in terms of properties of observational data [Balke and Pearl, 1997, Pearl, 1995]. While these bounds are valid in the presence of arbitrary unobserved confounding, they are often too loose to be of practical relevance and only hold for discrete treatment variables. Recently, Wolfe et al. [2019] introduced a technique called inflation that can be used to derive tighter bounds.

Furthermore, methods that do not rely on bounds or an adjustment set have to make other untestable assumptions on the causal mechanism. For example, Angrist et al. [1996] relies on the existence of instrumental variables that are not affected by unobserved confounders. Miao et al. [2018] and Louizos et al. [2017] assume proxy variables that while being correlated with unobserved confounders do not confound the treatment and outcome themselves. Last, the deconfounder of Wang and Blei [2019] builds on the assumptions that there are no unobserved single-cause confounders.

3 THEORY

For simplicity of exposition, we will make some assumptions regarding the types of variables below, but the construction of the causal reduction can be done for any standard measurable spaces. Furthermore, we will focus on perfect interventions as originally introduced by Strotz and Wold [1960] and popularized by Pearl [2009].

3.1 REDUCTION OF THE LATENT SPACE

Consider a treatment variable $\mathbf{X} \in \mathscr{X} = \mathbb{R}^N$ and an outcome variable $\mathbf{Y} \in \mathscr{Y} = \mathbb{R}^M$. We assume that the outcome does not cause the treatment. Furthermore, let there exist K latent confounders $Z_1, \dots, Z_K$, where $Z_i \in \mathscr{Z}_i = \mathbb{R}$, with an arbitrary dependency structure, see Figure 1 (a) for the corresponding directed acyclic graph (DAG). Without loss of generality, we can summarize the K latent confounders $Z_1, \dots, Z_K$ with arbitrary dependency structure using a single latent confounder $\mathbf{Z} \in \mathscr{Z} = \mathbb{R}^K$:

$$p(\mathbf{x}, \mathbf{y}) = \int_{\mathscr{Z}_1} \cdots \int_{\mathscr{Z}_K} p(\mathbf{x}, \mathbf{y}, z_1, \dots, z_K) dz_1 \dots dz_K \quad (1)$$

$$= \int_{\mathscr{Z}} p(\mathbf{x}, \mathbf{y}, \mathbf{z}) d\mathbf{z}. \quad (2)$$

The resulting causal Bayesian network is shown in Figure 1 (b), which has the following factorization:

$$p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{x}|\mathbf{z}) p(\mathbf{z}). \quad (3)$$

We aim to replace the above causal Bayesian network with one that is interventionally equivalent with respect to interventions on $\mathbf{X}$ and $\mathbf{Y}$, but where the latent confounder space $\mathscr{Z}$ is lower-dimensional.

First, we generate a copy $\mathbf{W} := \mathbf{X}$ of the treatment variable $\mathbf{X}$. We will interpret $\mathbf{W}$ as a latent variable and $\mathbf{X}$ as an observed deterministic effect of $\mathbf{W}$, via the function $\mathbf{X} = \mathrm{id}(\mathbf{W})$. We obtain the Bayesian Network in Figure 1 (c):

$$p(\mathbf{x}, \mathbf{y}, \mathbf{w}, \mathbf{z}) = p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{w}|\mathbf{z}) p(\mathbf{z}), \quad (4)$$

where $p(\mathbf{w}|\mathbf{z}) := p(\mathbf{x}|\mathbf{z})|_{\mathbf{x}=\mathbf{w}}$ is a copy of the Markov kernel from above evaluated in $\mathbf{w}$ rather than in $\mathbf{x}$. Furthermore, $p(\mathbf{x}|\mathbf{w}) := \delta_{\mathbf{w}}(\mathbf{x})$ is the delta peak centered at $\mathbf{w}$, representing the deterministic identity map from $\mathbf{W}$ to $\mathbf{X}$. If we marginalize out $\mathbf{W}$ we arrive at the initial causal Bayesian network in Figure 1 (b) again. Since interventions on observed variables commute with the marginalizing over latent variables the Bayesian networks in Figure 1 (b) and (c) are interventionally equivalent with respect to interventions on $\mathbf{X}$ and $\mathbf{Y}$ [Bongers et al., 2020]. Note that while copying $\mathbf{X}$ can be understood as an inflation [Wolfe et al., 2019], we eventually will reduce the Bayesian network shown in Figure 1 (a).

Second, we refactorize the latent distribution as shown in Figure 1 (c), (d) and (e):

$$p(\mathbf{x}, \mathbf{y}, \mathbf{w}, \mathbf{z}) = p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{w}|\mathbf{z}) p(\mathbf{z}) \quad (5)$$

$$= p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{w}, \mathbf{z}) \quad (6)$$

$$= p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{z}|\mathbf{w}) p(\mathbf{w}). \quad (7)$$

The causal Bayesian networks representing these three factorizations are interventionally equivalent, as we only factor the latent distributions differently and do not consider interventions on the latent variables.

Last, we can marginalize over $\mathbf{Z}$ and obtain:

$$p(\mathbf{x}, \mathbf{y}, \mathbf{w}) = p(\mathbf{y}|\mathbf{x}, \mathbf{w}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{w}), \quad (8)$$

where we used the following composed Markov kernel:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{w}) := \int p(\mathbf{y}|\mathbf{x}, \mathbf{z}) p(\mathbf{z}|\mathbf{w}) \, d\mathbf{z}. \quad (9)$$

Again, since marginalizing over latent variables and interventions commute, the final Bayesian network in Figure 1 (f) is interventionally equivalent to the ones in Figure 1 (a–e) with respect to interventions on $\mathbf{X}$ and $\mathbf{Y}$.

Since $\mathbf{W}$ is a copy of $\mathbf{X}$, we successfully reduced the dimensionality of the latent confounder from K to N. In the common case of one-dimensional $\mathbf{X}$, we expect $N = 1 \ll K$ and therefore achieve a significant reduction of the latent space.

We formulate the conclusion as a theorem:

Theorem 3.1 (Causal Reduction)**.** *Let $\mathscr{M}$ be a causal Bayesian network with observed variables $\mathbf{X} \in \mathscr{X}, \mathbf{Y} \in \mathscr{Y}$ and latent variables, $Z_1 \in \mathscr{Z}_1, \dots, Z_K \in \mathscr{Z}_K$ such that $\mathbf{Y}$ is not an ancestor of $\mathbf{X}$. Then there exists a causal Bayesian network $\mathscr{M}^*$ with observed variables $\mathbf{X} \in \mathscr{X}$ and $\mathbf{Y} \in \mathscr{Y}$ and a single latent variable $\mathbf{Z} \in \mathscr{X}$ (that takes values in the same space as $\mathbf{X}$) such that $\mathscr{M}^*$ is interventionally equivalent to $\mathscr{M}$ with respect to perfect interventions on the observed variables $\mathbf{X}$ and $\mathbf{Y}$, i.e.,*

$$p_{\mathscr{M}}(\mathbf{x}, \mathbf{y}) = p_{\mathscr{M}^*}(\mathbf{x}, \mathbf{y})$$

$$p_{\mathscr{M}}(\mathbf{x} \mid \mathrm{do}(\mathbf{y})) = p_{\mathscr{M}^*}(\mathbf{x} \mid \mathrm{do}(\mathbf{y}))$$

$$p_{\mathscr{M}}(\mathbf{y} \mid \mathrm{do}(\mathbf{x})) = p_{\mathscr{M}^*}(\mathbf{y} \mid \mathrm{do}(\mathbf{x}))$$

We call the causal Bayesian network $\mathscr{M}^*$ a *causal reduction* of $\mathscr{M}$ since it will typically be the case that the latent space will be reduced, yet the causal semantics are preserved by construction. The single latent confounder $\mathbf{Z}$ in $\mathscr{M}^*$ will parsimoniously represent the causal influence of *all* latent confounders of $\mathbf{X}$ and $\mathbf{Y}$ in $\mathscr{M}$. For example, a single binary confounder suffices for a binary treatment variable. Extending the derivation to *simple* structural causal models (an extension of causal Bayesian networks that can represent feedback loops [Bongers et al., 2020]) is straightforward, as long as $\mathbf{X}$ and $\mathbf{Y}$ are not part of a causal cycle (although the other variables might be involved in cycles).

3.2 FROM BAYESIAN NETWORKS TO STRUCTURAL CAUSAL MODELS

Whereas in the previous section we relied on causal Bayesian networks to conduct our reduction, we now move to structural causal models (SCMs) [Pearl, 2009, Bongers et al., 2020] to obtain convenient parameterizations. We make use of the exogenous variables $\mathbf{U}, \mathbf{V}$ to represent the

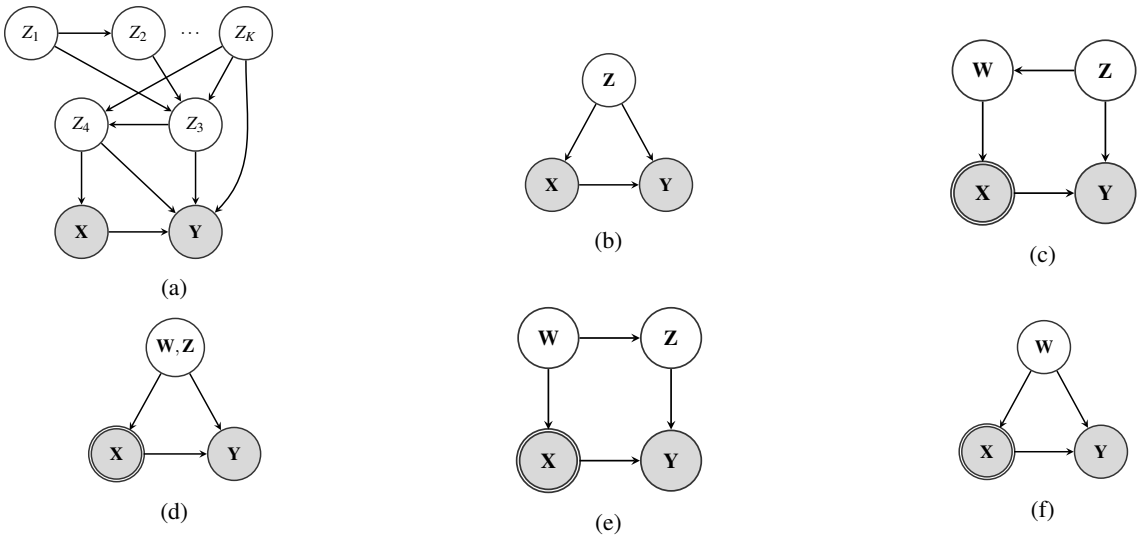

Figure 1: A graphical explanation of our causal reduction technique. (a) We assume a treatment variable $\mathbf{X}$, an outcome variable $\mathbf{Y}$, and K latent confounders $Z_1, \ldots, Z_K$ with an arbitrary dependency structure. (b) We represent the K latent confounders $Z_1, \ldots, Z_K$ by $\mathbf{Z} \in \mathscr{Z} = \mathbb{R}^K$. (c) We create a copy of $\mathbf{X}$ called $\mathbf{W}$. We use a double circle to indicate that a variable is a deterministic function of its parents. (d, e) Instead of using the factorization from (c), $p(\mathbf{w}, \mathbf{z}) = p(\mathbf{w}|\mathbf{z})p(\mathbf{z})$, we choose $p(\mathbf{w}, \mathbf{z}) = p(\mathbf{z}|\mathbf{w})p(\mathbf{w})$. (f) Last, we marginalize over $\mathbf{Z}$. Note that at every step (a–f) the DAGs are interventionally equivalent with respect to interventions on $\mathbf{X}$ and $\mathbf{Y}$.

noise in the reduced causal model. This in turn allows us to express all causal relationships as deterministic functions. Estimating the model then boils down to estimating these functions, as we will illustrate in Section 4.

Theorem 3.2. *Let $P(\mathbf{X}|\mathbf{Y})$ be a Markov kernel (e.g. a conditional probability distribution) of a $\mathbb{R}^M$-valued variable $\mathbf{X}$ with components X_m, $m = 1, \ldots, M$ and with argument $\mathbf{Y}$ that can take values in any measurable space. Then there is a M-dimensional standard normal random variable $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$ independent of $\mathbf{Y}$ and a deterministic measurable map F such that:*

$$\mathbf{X} = F(\mathbf{Z}, \mathbf{Y}) \quad a.s. \tag{10}$$

Furthermore, the map F is 'well-behaved', in the sense that it is composed out of (inverse) conditional cumulative distribution functions.

The proof is provided in the Appendix 7.2. Theorem 3.2 enables us to obtain a reduced SCM from the reduced Bayesian Network in Equation 8 with structural equations

$$\mathbf{X} = F(\mathbf{U}), \tag{11}$$
$$\mathbf{Y} = G(\mathbf{U}, \mathbf{V}, \mathbf{X}), \tag{12}$$

where $\mathbf{U} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N) \perp\!\!\!\perp \mathbf{V} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$, and F, G are deterministic maps. This allows us to parameterize the reduced causal model in terms of the two functions F and G. The corresponding DAG is shown in Figure 2.

Figure 2: DAG of the reduced SCM with $\mathbf{U} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ and $\mathbf{V} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$. Left: Reduced SCM. Right: Reduced SCM after an intervention on $\mathbf{X}$.

3.3 PARAMETER SHARING IN THE LINEAR GAUSSIAN CASE

We now consider the case where all causal relationships in Figure 1 (a) are linear, and all distributions are Gaussian. We can then guarantee that the reduced causal model is linear Gaussian as well.

Theorem 3.3 (Reduced linear Gaussian model)**.** *Consider a linear Gaussian SCM (or causal Bayesian network with possible latent variables) with observed variables $\mathbf{X}$ and $\mathbf{Y}$ such that $\mathbf{Y}$ is not ancestor of $\mathbf{X}$. Then this causal model is interventionally equivalent to a reduced linear Gaussian causal model with the following structural equations:*

$$\mathbf{X} = \mathbf{a} + B\boldsymbol{u},$$
$$\boldsymbol{Y} = \mathbf{c} + D\mathbf{X} + E\mathbf{u} + F\mathbf{v}, \tag{13}$$

with vectors $\mathbf{a}$, $\mathbf{c}$ and matrices B, D, E, F, where B and F can be chosen to be lower-triangular with non-negative diagonal entries, and where $\mathbf{u}$ is a standard Gaussian latent

variable of the same dimension as $\mathbf{x}$ *and where* $\mathbf{v}$ *is a standard Gaussian latent variable of the same dimension as* $\mathbf{y}$ *that is independent of* $\mathbf{u}$.

The proof of Theorem 3.3 can be found in the Appendix 7.3. Next, we use the reduced linear Gaussian model from Theorem 3.3 to prove that the parameters of the observational distribution are constrained by the parameters of the interventional distribution.

Theorem 3.4 (Linear Gaussian parameter constraints)**.** *Consider a linear-Gaussian SCM (or causal Bayesian network with possible latent variables) with two observed variables* $\mathbf{X}$ *and* $\mathbf{Y}$ *such that* $\mathbf{Y}$ *is not ancestor of* $\mathbf{X}$*. The entailed observational and interventional distributions are Gaussian. Modeling* $p(\mathbf{x}), p(\mathbf{y}|\mathbf{x})$ *and* $p(\mathbf{y}|\operatorname{do}(\mathbf{x}))$ *independently from each other could be done with the following parameterization:*

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\alpha,\Sigma), \tag{14}$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|\gamma+\Delta\mathbf{x},\Pi), \tag{15}$$

$$p(\mathbf{y}|\operatorname{do}(\mathbf{x})) = \mathcal{N}(\mathbf{y}|\widetilde{\gamma}+\widetilde{\Delta}\mathbf{x},\widetilde{\Pi}), \tag{16}$$

with covariance matrices Σ*,* Π*,* $\widetilde{\Pi}$*. However, using the reduced causal model from Theorem 3.3 we find that these parameters are constrained by the following relations:*

$$(\widetilde{\gamma}-\gamma)+(\widetilde{\Delta}-\Delta)\alpha = 0, \tag{17}$$

$$(\widetilde{\Delta}-\Delta)\Sigma(\widetilde{\Delta}-\Delta)^\top+\Pi = \widetilde{\Pi}. \tag{18}$$

From Equation 18 we can easily see that $\widetilde{\Pi}-\Pi$ is positive semidefinite. Furthermore, we see that these constraints lead to a reduced parameter count, M parameters for Equation 17 and $M(M+1)/2$ parameters for Equation 18, assuming $\mathbf{y}$ to be M-dimensional. In total, we have reduced the parameter count by $M(M+3)/2$ by modeling the parameters of the observational and interventional distributions jointly. The proof of Theorem 3.4 can be found in Appendix 7.4.

Now consider the task of learning the parameters of our reduced model. In the linear Gaussian case, the reduced causal model tells us exactly how many parameters we need to model the observational and interventional distribution, and which of the parameters are shared. Since the parameters $\mathbf{c}, D, E$ and, F are shared between the observational and interventional distribution, we can estimate them jointly using observational and interventional data. This effectively leads to a reduced sample complexity when trying to model the interventional distribution, which is beneficial for causal effect estimation when we are assuming that we only have access to a small number of interventional samples and a large number of observational samples. In the Appendix 7.6, we experiment on observational and interventional data generated with linear causal mechanisms, giving a linear parameterization of the reduced linear model that can learn linear causal mechanisms.

In Section 4, we propose a parameterization of the reduced causal model using normalizing flows that is suitable for the general (non-linear, non-Gaussian) setting. We conjecture that the form of the reduced structural equations (Equations 11 and 12) also imposes constraints on the observational and interventional distribution in the more general setting. In Section 5, we conduct a series of experiments where we simulate observational and interventional data using nonlinear causal mechanisms and estimate the interventional distribution $p(\mathbf{y}|\operatorname{do}(\mathbf{x}))$. We show that we are indeed able to reduce the sample complexity by training a single flow model with observational and interventional data.

3.4 REDUCTION WITH OBSERVED CONFOUNDERS

There are many scenarios where we are interested in estimating the conditional causal effect of interventions given additional covariates $\mathbf{C}$ that might confound treatment and outcome, for example when estimating the efficacy of a vaccine depending on age, weight or gender. We consider an additional set of L *observed* confounders $C_1,\dots,C_L$ of $\mathbf{X}$ and $\mathbf{Y}$, taking values in arbitrary measurable spaces, and with an arbitrary joint distribution $p(\mathbf{c})$. In the following, we summarize all L observed confounders using a single variable $\mathbf{C} \in \mathscr{C}$. We provide a more detailed derivation in the Appendix 7.5 and give only a short sketch here. First, we follow the same steps as in Section 3.1 to derive a reduced causal model of the following form

$$p(\mathbf{x},\mathbf{y},\mathbf{w},\mathbf{c}) = p(\mathbf{y}|\mathbf{x},\mathbf{w},\mathbf{c})p(\mathbf{x}|\mathbf{w})p(\mathbf{w}|\mathbf{c})p(\mathbf{c}). \tag{19}$$

Again, at every step, the Bayesian network is interventionally equivalent to the ones before, for interventions on $\mathbf{X}$ and $\mathbf{Y}$.

Then, we use a similar approach as in Section 3.2 but also marginalize out $\mathbf{W}$ to convert the causal Bayesian Network into an SCM with structural equations of the form

$$\mathbf{X} = F(\mathbf{U},\mathbf{C}), \tag{20}$$

$$\mathbf{Y} = G(\mathbf{U},\mathbf{V},\mathbf{X},\mathbf{C}), \tag{21}$$

where $\mathbf{U} \sim \mathcal{N}(\mathbf{0},\mathbf{I}_N) \perp\!\!\!\perp \mathbf{V} \sim \mathcal{N}(\mathbf{0},\mathbf{I}_M)$ and F and G are two deterministic maps. Again, this preserves the observational and interventional distributions for interventions on $\mathbf{X}$ and $\mathbf{Y}$. The corresponding DAG is shown in Figure 3.

4 PRACTICAL IMPLEMENTATION

Now that we have successfully reduced the model complexity, we will parameterize the functions F and G so we can learn the model from data. While this can be done in many different ways, we make use of diffeomorphisms, i.e., mappings that are differentiable and have a differentiable inverse. By using the change-of-variables formula we can

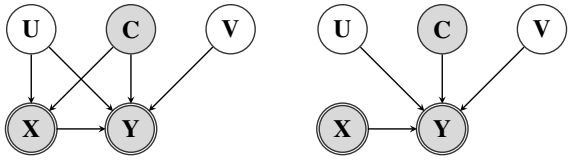

Figure 3: DAG of reduced SCM with observed confounder $\mathbf{C}$, $\mathbf{U} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ and $\mathbf{V} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$. Left: Reduced SCM for observational data. Right: Reduced SCM after intervention on $\mathbf{X}$.

derive a maximum-likelihood estimator for the mappings' parameters which can be efficently optimized through backpropagation. In the deep learning community those invertible and differentiable mappings are called (normalizing) flows and much recent research went into finding flexible and easily invertible mappings, see Pawlowski et al. [2020] and Khemakhem et al. [2020] for other recent applications of normalizing flows to approximate nonlinear causal mechanisms. Our flow model consists of two flows that are jointly trained using observational and interventional data. In the following, we derive the loss function for observational and interventional data separately. For the remaining part of this paper we focus on one-dimensional treatment outcome pairs, i.e. $x \in \mathcal{X} = \mathbb{R}$ and $y \in \mathcal{Y} = \mathbb{R}$, and (optionally) a L-dimensional observed confounder $\mathbf{c} \in \mathcal{C} = \mathbb{R}^L$.

4.1 OBSERVATIONAL DATA

Starting from the DAG in Figure 2 the joint-likelihood $p(x,y)$ can be factorized as follows

$$\log p(x,y) = \log p(y|x) + \log p(x). \tag{22}$$

We now use the following bijective transformations between observed variables x, y and latent variables u, v

$$u = f_\phi(x), \tag{23}$$
$$v = g_{x,u;\theta}(y), \tag{24}$$

where the function $g(.)$ is invertible with respect to v.

Without loss of generality we assume independent, standard Gaussian distributions for u, v: $p_U(u) = \mathcal{N}(0,1) \perp\!\!\!\perp p_V(v) = \mathcal{N}(0,1)$. The transformations defined above allow us to rewrite Equation 22 using the change of variable formula

$$\log p(x,y) = \log p_V(g_{x,u;\theta}(y)) + \log \left| \frac{\delta g_{x,u;\theta}(y)}{\delta y} \right| + \log p_U(f_\phi(x)) + \log \left| \frac{\delta f_\phi(x)}{\delta x} \right|. \tag{25}$$

Last we use $u = f_\phi(x)$ in order to replace u in $g_{x,u;\theta}(y)$. The final $\log p(x,y)$ is given by

$$\log p(x,y) = \log p_V(g_{x,f_\phi(x);\theta}(y)) + \log \left| \frac{\delta g_{x,f_\phi(x);\theta}(y)}{\delta y} \right| + \log p_U(f_\phi(x)) + \log \left| \frac{\delta f_\phi(x)}{\delta x} \right|. \tag{26}$$

The parameters ϕ and θ are jointly updated by minimizing $\sum_{o=1}^{N_O} -\log p(x_o, y_o)$ given N_O observational training samples.

4.2 INTERVENTIONAL DATA

In contrast to the observational setting, we only have to consider the conditional likelihood $p(y|\,\mathrm{do}(x))$ in the interventional case and treat $p(\mathrm{do}(x))$ as a constant. Since we cannot use $f_\phi(x)$ to impute u, we instead marginalize over u

$$\log p(y \mid \mathrm{do}(x)) = \log \int p(y|\,\mathrm{do}(x), u) p(u) du. \tag{27}$$

Inserting the bijective mapping $v = g_{x,u;\theta}(y)$ in Equation 27, we obtain

$$\log p(y \mid \mathrm{do}(x)) = \log \int p_V(g_{x,u;\theta}(y)) \left| \frac{\delta g_{x,u;\theta}(y)}{\delta y} \right| p(u) du, \tag{28}$$

where we use the trapezoidal rule to compute a numerical approximation of the integral. The parameter θ can be updated by minimizing $\sum_{i=1}^{N_I} -\log p(y_i|\,\mathrm{do}(x_i))$ given N_I interventional training samples.

4.3 JOINT OPTIMIZATION

Assuming we have N_O observational samples and N_I interventional samples, the full loss is given by

$$loss = \sum_{o=1}^{N_O} -\log p(x_o, y_o) + \alpha \sum_{i=1}^{N_I} -\log p(y_i|\,\mathrm{do}(x_i)). \tag{29}$$

The parameters ϕ and θ of the transformation f and g are learned by minimizing the loss using gradient descent. In the case of $N_0 \neq N_I$, we find it beneficial to introduce α to balance the two loss terms and thus scale the gradients. We find $\alpha = N_O/N_I$ to work well and will use it throughout the rest of the paper.

4.4 GENERATING SAMPLES

After training we are able to generate observational and interventional samples with a single flow model. The sampling procedure for observational samples consists of the following steps: $v \sim \mathcal{N}(0,1), u = f_\phi(x_o)$, and $y_o = g^{-1}_{x_o,u;\theta}(v)$, where we assume $x_o \in \mathbb{R}$ to be observed. If we instead want to generate an interventional sample, the sample procedure follows: $v \sim \mathcal{N}(0,1), u \sim \mathcal{N}(0,1)$, and $y_i = g^{-1}_{x_i,u;\theta}(v)$, where we assume $x_i \in \mathbb{R}$ to be observed.

4.5 ADDITIONAL OBSERVED CONFOUNDERS

In order to parameterize the DAG in Figure 3 we simply have to replace the functions $f(.)$ and $g(.)$ by $u = f_{\mathbf{c};\phi}(x)$ and $v = g_{x,u,\mathbf{c};\theta}(y)$ where $\mathbf{c} \in \mathbb{R}^L$ is assumed to be observed. The optimization procedure does not change.

5 EXPERIMENTS

Following the analysis in Section 3.3, we perform a series of experiments on simulated data, where the causal relationships between all variables are nonlinear, showing that we are able to significantly reduce the number of interventional samples required to estimate the interventional distribution $p(y|\,\mathrm{do}(x))$ by training jointly with observational and interventional samples. Throughout this section, we are using the parameterization described in Section 4, where we use linear rational spline flows [Dolatabadi et al., 2020]. For a detailed description of this choice see the Appendix 7.7. We perform two sets of experiments: (1) We consider K latent confounders $Z_1, \ldots, Z_K \in \mathbb{R}$ with an arbitrary dependency structure, as shown in Figure 4. (2) We consider L additional, observed confounders $C_1, \ldots, C_L \in \mathbb{R}$ with an arbitrary dependency structure. All flow models are implemented with the automatic differentiation packages Pytorch [Paszke et al., 2019] and Pyro [Bingham et al., 2019]. All code is available under https://github.com/max-ilse/CausalReduction.

5.1 WITHOUT OBSERVED CONFOUNDERS

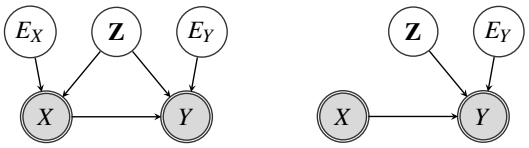

Figure 4: DAG of the data generating process.

We simulate cause and effect pairs following the DAG in Figure 4. The left DAG in Figure 4 is used to generate observational samples and the right DAG in Figure 4 is used to generate interventional samples. They share the same underlying causal process, $Y = G(X, E_Y, \mathbf{Z})$. A single dataset consists of observational and interventional samples.

All causal relationships are simulated using fully connected neural networks with a single hidden layer, where the weights are randomly initialized. The activation functions are rectified linear units (ReLUs). This ensures that the causal mechanisms simulated by $X = F(E_X, \mathbf{Z})$ and $Y = G(X, E_Y, \mathbf{Z})$ are nonlinear. The values of $E_X, E_Y, \mathbf{Z}$ and $\mathrm{do}(X)$ are sampled from a random distribution, as seen in Mooij et al. [2016]. A detailed step-by-step description of the simulation procedure is given in the Appendix 7.8. Following the process described above, we simulate 100 datasets while varying the number of dimensions K of the unobserved confounder $\mathbf{Z}$ and the random seed that is, among others, controlling the initialization of the neural networks used to model the causal mechanisms. We choose K between 1 and 10 since for $K > 10$ the joint distribution $p(x, y)$ becomes increasingly more Gaussian due to the central limit theorem. Next, we manually select ten datasets with the smallest overlap of observational and interventional samples to select cases with "strong" confounding. Note that we choose these ten datasets before training a single flow model. A scatter plot of 1000 observational and 1000 interventional samples for each of the ten datasets can be found in Section 7.10.

In this experiment we are interested in estimating the interventional distribution $p(y|\,\mathrm{do}(x))$. For each dataset we train three variants of our reduced causal model parameterized with normalizing flows. The first flow model is trained using only observational data, see Section 4.1. The second flow model is trained using only interventional data, see Section 4.2. The third flow model is trained using observational and interventional data jointly, see Section 4.3. For each of the ten datasets, we keep the number of observational samples constant at 1000 and use an increasing number of interventional samples 50, 100, 250, 500, 750, 1000, resulting in six experiments per dataset. For example in the case of 50 interventional and 1000 observational samples, the first flow model is trained with 1000 observational samples, the second flow model is trained with 50 interventional samples, and the third flow model is jointly trained with 1000 observational **and** 50 interventional samples. Motivated by the work of Oliver et al. [2018] on the realistic evaluation of semi-supervised learning algorithms we use the same number of samples for training and validation. In every case we use 1000 interventional samples for testing.

In order to compare the performance of the three flow models, we calculate the negative log-likelihood averaged over the test set, $-\frac{1}{N_I}\sum_{i=1}^{N_I} \log p(y_i \mid \mathrm{do}(x_i))$. To have a fair comparison, the same training procedure, architecture, optimizer, and hyperparameters are used for all flow models in all experiments. We use Adam [Kingma and Ba, 2015] with a learning rate of 0.001 and the default values for β_1, β_2. We train for 10000 epochs. The training is terminated early when the validation loss did not improve for 1000 epochs. We perform full batch gradient descent, where for the third flow model we alternate between batches of observational and interventional samples. For the linear rational spline flows we use 32 bins and set the bound $B = 6$. For the conditional version of the linear rational spline flows, we use a fully connected neural network with two hidden layers and ReLU activations.

In the Appendix 7.10, we provide extensive visualizations of the results of all experiments, including scatter plots of training data, samples from the trained flow models, negative log-likelihood values for all flow models on the interven-

Table 1: Comparison of a flow model trained with interventional samples only and a flow model trained with interventional and observational samples. We calculate the ratio N_I^*/N_I, where N_I^* is the number of interventional samples necessary to match the interventional test log-likelihood of a flow model trained with N_I interventional and 1000 observational samples. E.g. in the case of dataset 3 and $N_I = 100$, if we were to use only interventional samples, we would require twice as many interventional samples compared to using 100 interventional and 1000 observational samples. For dataset 11 to 15, we simulate an additional observed confounder $\mathbf{C}$. Note that if a large number of interventional samples ($250 < N_I \leq 1000$) are available the improvements become smaller as shown in the Appendix 7.10.

dataset	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ratio for $N_I = 50$	1.4	1.8	2.2	1.2	0.2	2.2	2.1	1.7	1.9	1.6	3.2	3.2	2.2	2.7	3.2
ratio for $N_I = 100$	0.8	2.6	2.0	1.5	0.3	2.1	2.0	2.5	2.0	2.1	3.2	2.9	2.5	3.0	2.5
ratio for $N_I = 250$	1.0	1.5	1.8	1.6	0.5	1.7	1.1	1.5	1.2	1.7	2.4	2.3	2.3	2.1	1.7

tional and observational test sets. To summarize our findings, we calculate the ratio of samples required to reach the same performance, measured in averaged negative log-likelihood when only using interventional samples. In Table 1 we see that in the case of dataset 3 and $N_I = 100$, we need two times the number of interventional samples (in the absence of observational training samples) in order to achieve the same performance as a flow model that is jointly trained with 100 interventional and 1000 observational samples. In eight of the ten datasets, we can substantially reduce the number of interventional samples required when using additional 1000 observational samples.

Only in the case of dataset 5, we find that we need substantially more interventional samples in order to train our flow model jointly with observational and interventional data. We argue that in the case of dataset 5 the interventional distribution resembles a standard Gaussian distribution which can easily be estimated from very few interventional samples. Last, the results in Table 1, dataset 1 to 10, are in agreement with qualitative results in the Appendix 7.10. We find that samples from the flow model trained with interventional and observational data better resemble the training data compared to samples from a flow model trained with only interventional data.

5.2 WITH OBSERVED CONFOUNDERS

We now consider the case of an additional L-dimensional observed confounder $\mathbf{C}$. We use the same setup as in Section 5.1 to simulate treatment outcome pairs x, y. We use the following nonlinear causal mechanisms to generate treatment X and outcome Y: $X = f(E_X, \mathbf{Z}, \mathbf{C})$ and $Y = g(X, E_Y, \mathbf{Z}, \mathbf{C})$, a detailed description of the simulation procedure is given in the Appendix 7.8. Again, we generate 100 datasets by varying K, L between 1 and 5 as well as the random seed. We select five datasets following the same criteria as described in Section 5.1. Furthermore, we use the implementation described in Section 4.5 to estimate the SCM in Figure 3.4. For each of the five datasets, we keep the number of observational samples constant at 1000 and use an increasing number of interventional samples: 50, 100, 250, 500, 750, 1000, resulting in six experiments per dataset. We compare three flow models trained with observational, interventional, and observational plus interventional data respectively. The training details are the same as in Section 5.1. An extensive comparison of the three flow models, as well as visualizations for each dataset, can be found in the Appendix 7.12. The main result of the experiments with additional observed confounders is the following: For each of the five datasets, we are able to substantially reduce the required number of interventional samples with our flow model trained with observational and interventional data, see Table 1, dataset 11 to 15. We find that we can reduce the number of required samples by a factor of two to three when training with 1000 additional observational samples.

6 CONCLUSION

We propose a causal reduction technique that replaces any number of (possibly high-dimensional) unobserved confounders with a single confounder, of the same dimensionality as the treatment variable, preserving the observational distributions entailed by the model as well as the interventional distributions for interventions on the treatment and outcome variable. Additionally, we show that we can perform such a reduction even in the presence of observed confounders. This allows us to derive constraints between the observational and interventional distributions in the linear Gaussian case, showing that these objects are not independent. In the nonlinear case, we propose a flexible parameterization of the reduced causal model using normalizing flows. This parameterization allows us to train a single flow model by combining observational and interventional data. In simulations, for 13 out of 15 simulated datasets we substantially reduce the required number of interventional samples if sufficient observational samples are available. Possible future work includes (i) applying the flow model to high dimensional outcome variables, e.g. medical images, (ii) using the reduction technique for causal discovery, e.g. inferring causal directions, and (iii) analyzing the relationship between the constraints in Section 3.3 and the instrumental and Bell inequalities [Pearl, 1995, Bell, 1964].

Acknowledgements

This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement 639466).

MI was funded by the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (Grant DLMedIa: Deep Learning for Medical Image Analysis).

The authors would like to thank Marco Federici, Bas Veeling, Karen Ullrich, Emiel Hoogeboom, Nick Pawlowski, Daniel C. Castro and Ben Glocker for useful discussions.

References

Joshua D. Angrist, Guido W. Imbens, and Donald B. Rubin. Identification of Causal Effects Using Instrumental Variables. *Journal of the American Statistical Association*, 91 (434):444–455, 1996.

Susan Athey, Raj Chetty, and Guido Imbens. Combining Experimental and Observational Data to Estimate Treatment Effects on Long Term Outcomes. 2020. arXiv: 2006.09676.

Alexander Balke and Judea Pearl. Bounds on Treatment Effects From Studies With Imperfect Compliance. *Journal of the American Statistical Association*, 92(439):1171–1176, 1997.

J. S. Bell. On the Einstein Podolsky Rosen paradox. *Physics, Physique, Fizika*, 1(3):6, 1964.

Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 20(28):1–6, 2019.

Stephan Bongers, Patrick Forré, Jonas Peters, Bernhard Schölkopf, and Joris M. Mooij. Foundations of structural causal models with cycles and latent variables. 2020. arXiv:1611.06221v4.

Bénédicte Colnet, Imke Mayer, Guanhua Chen, Awa Dieng, Ruohong Li, Gaël Varoquaux, Jean-Philippe Vert, Julie Josse, and Shu Yang. Causal inference methods for combining randomized trials and observational studies: a review. 2020. arXiv: 2011.08047.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017*, 2017.

Hadi Mohaghegh Dolatabadi, Sarah M. Erfani, and Christopher Leckie. Invertible generative modeling using linear rational splines. In Silvia Chiappa and Roberto Calandra, editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020*, 2020.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, 2019.

Richard D. Fuhr and Michael Kallay. Monotone linear rational spline interpolation. *Computer Aided Geometric Design*, 9(4):313–319, 1992.

Grand Review Research. Clinical trials market size, share & growth report, 2021-2028. `https://www.grandviewresearch.com/industry-analysis/global-clinical-trials-market`. Accessed: 2021-02-17.

Keisuke Hirano and Guido W. Imbens. The Propensity Score with Continuous Treatments. In *Wiley Series in Probability and Statistics*, pages 73–84. 2005.

Nathan Kallus, Aahlad Manas Puli, and Uri Shalit. Removing hidden confounding by experimental grounding. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, 2018.

Ilyes Khemakhem, Ricardo Pio Monti, Robert Leech, and Aapo Hyvärinen. Causal Autoregressive Flows. 2020. arXiv: 2011.02268.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

I. Kobyzev, S. Prince, and M. Brubaker. Normalizing Flows: An Introduction and Review of Current Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.

Christos Louizos, Uri Shalit, Joris M. Mooij, David A. Sontag, Richard S. Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 2017.

David Madigan, Paul E. Stang, Jesse A. Berlin, Martijn Schuemie, J. Marc Overhage, Marc A. Suchard, Bill Dumouchel, Abraham G. Hartzema, and Patrick B. Ryan. A systematic statistical approach to evaluating evidence from observational studies. *Annual Review of Statistics and Its Application*, 1(1):11–39, 2014.

Wang Miao, Zhi Geng, and Eric J Tchetgen Tchetgen. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105(4):987–993, 08 2018.

Joris M. Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing Cause from Effect Using Observational Data: Methods and Benchmarks. *Journal of Machine Learning Research*, 17(32):1–102, 2016.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, 2018.

George Papamakarios, Iain Murray, and Theo Pavlakou. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 2017.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. 2019. arXiv: 1912.02762.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, 2019.

Nick Pawlowski, Daniel Coelho de Castro, and Ben Glocker. Deep structural causal models for tractable counterfactual inference. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, volume 33, 2020.

Judea Pearl. On the testability of causal models with latent and instrumental variables. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, 1995.

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

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015*, 2015.

Evan Rosenman, Art B. Owen, Michael Baiocchi, and Hailey Banack. Propensity Score Methods for Merging Observational and Experimental Datasets. 2018. arXiv: 1804.07863.

Evan Rosenman, Guillaume Basse, Art Owen, and Michael Baiocchi. Combining Observational and Experimental Datasets Using Shrinkage Estimators. 2020. arXiv: 2002.06708.

Robert H. Strotz and H. O. A. Wold. Recursive vs. Nonrecursive Systems: An Attempt at Synthesis (Part I of a Triptych on Causal Chain Systems). *Econometrica*, 28 (2):417–427, 1960.

E. G. Tabak and Cristina V. Turner. A Family of Nonparametric Density Estimation Algorithms. *Communications on Pure and Applied Mathematics*, 66(2):145–164, 2013.

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

Elie Wolfe, Robert W. Spekkens, and Tobias Fritz. The inflation technique for causal inference with latent variables. *Journal of Causal Inference*, 7(2), 2019.

7 APPENDIX

7.1 THEOREM 7.1 AND PROOF

Theorem 7.1. *Let $P(X|\mathbf{Y})$ be a Markov kernel, where the variable X takes values in $\mathbb{R}$ (or $[-\infty,\infty]$) and argument $\mathbf{Y}$ has values in any measurable space (e.g. $\mathbb{R}^M$). Then there exists a uniformly distributed variable $E \sim U[0,1]$ that is independent of $\mathbf{Y}$ and a deterministic function F, namely the conditional quantile function of X given $\mathbf{Y}$, such that:*

$$X = F(E|\mathbf{Y}) \quad a.s. \tag{30}$$

Proof. Consider the interpolated conditional cumulative distribution function (iccdf) of X given $\mathbf{Y}$ with $u \in [0,1]$:

$$G(x;u|\mathbf{y}) := P(X < x|\mathbf{y}) + u \cdot P(X = x|\mathbf{y}). \tag{31}$$

Furthermore, consider the conditional quantile function (cqf) of X given $\mathbf{Y}$ with $e \in [0,1]$:

$$F(e|\mathbf{y}) := \inf\{\tilde{x} \in \mathbb{R} \,|\, G(\tilde{x};1|\mathbf{y}) \geq e\}. \tag{32}$$

Then take any uniformly distributed random variable $U \sim U[0,1]$ independent of $(X,\mathbf{Y})$ and define:

$$E := G(X;U|\mathbf{Y}), \tag{33}$$

where we plugged X, U and $\mathbf{Y}$ into G. Then one can check using standard arguments for cdf and cqf that E is uniformly distributed, $E \sim U[0,1]$, which is independent of the value $\mathbf{y}$ of $\mathbf{Y}$. Furthermore, one can show that:

$$X = F(E|\mathbf{Y}) \quad \text{a.s.} \tag{34}$$

□

7.2 PROOF OF THEOREM 3.2

Proof. We use Theorem 7.1 inductively.

1. Consider the cqf F_1 of $P(X_1|\mathbf{Y})$. Then by 7.1 there is a random variable $E_1 \sim U[0,1]$ independent of $\mathbf{Y}$ such that $X_1 = F_1(E_1|\mathbf{Y})$ a.s.
2. Now consider the cqf F_2 of $P(X_2|E_1,\mathbf{Y})$. Then by 7.1 there is a random variable $E_2 \sim U[0,1]$ independent of E_1, $\mathbf{Y}$ such that $X_2 = F_2(E_2|E_1,\mathbf{Y})$ a.s.
3. Now consider the cqf F_3 of $P(X_3|E_2,E_1,\mathbf{Y})$. Then by 7.1 there is a random variable $E_3 \sim U[0,1]$ independent of E_2,E_1, $\mathbf{Y}$ such that $X_3 = F_3(E_3|E_2,E_1,\mathbf{Y})$ a.s.
4. and so on .... until:
5. $X_M = F_M(E_M|E_{M-1},\ldots,E_1,\mathbf{Y})$ a.s. with $E_M \sim U[0,1]$ independent of $E_{M-1},\ldots,E_1$,$\mathbf{Y}$.

Now we put $Z_d := \Phi^{-1}(E_d)$, where Φ is the cdf of $\mathcal{N}(0,1)$. Then $E_d = \Phi(Z_d)$ and the Z_d are $\mathcal{N}(0,1)$-distributed and $\mathbf{Z} = (Z_1,\ldots,Z_M)$ is independent $\mathbf{Y}$). So $\mathbf{Z} = (Z_1,\ldots,Z_M) \sim \mathcal{N}(\mathbf{0},\mathbf{I}_M)$ and independent of $\mathbf{Y}$. Furthermore, we have almost surely the equations:

$$X_1 = F_1(\Phi(Z_1)|\mathbf{Y}), \tag{35}$$
$$X_2 = F_2(\Phi(Z_2)|\Phi(Z_1),\mathbf{Y}), \tag{36}$$
$$\vdots \qquad \ddots \tag{37}$$
$$X_M = F_M(\Phi(Z_M)|\Phi_{(Z_{M-1})},\ldots,\Phi(Z_1),\mathbf{Y}). \tag{38}$$

□

7.3 PROOF OF THEOREM 3.3

Proof. This follows the same steps as the general construction in Equations 3, 4, 5, 6, where $p(\mathbf{x}|\mathbf{w}) = \delta_w(\mathbf{x})$ reflects the identity map. In Equation 7, note that $p(\mathbf{z}|\mathbf{w})$ is linear Gaussian by the well-known conditioning formula for jointly Gaussian distributions. We then arrive at Equation 8, where it can be checked that in Equation 9 both parts, $p(\mathbf{z}|\mathbf{w})$ and $p(\mathbf{y}|\mathbf{x},\mathbf{z})$, are linear Gaussian, thus makes $p(\mathbf{y}|\mathbf{x},\mathbf{w})$ linear Gaussian. Finally, we use the reparameterization trick together with a Cholesky decomposition, as seen in Section 3.2, to turn $p(\mathbf{w})$ into a standard Gaussian $p(\mathbf{u})$, which makes $p(\mathbf{x}|\mathbf{u})$, as a composition of identity map and linear Gaussian also a linear Gaussian. Note that $p(\mathbf{y}|\mathbf{x},\mathbf{u})$ again is linear Gaussian by similar arguments. Last we use the reparameterization trick again to obtain $p(\mathbf{y}|\mathbf{x},\mathbf{u},\mathbf{v})$ where $\mathbf{V} \sim \mathcal{N}(\mathbf{0},\mathbf{I}_M)$. □

7.4 PROOF OF THEOREM 3.4

Proof. The linear version of the reduced SCM in Equation 13 entails the following distributions over $\mathbf{x}$ and $\mathbf{y}$

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{a},BB^\top), \tag{39}$$
$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|\mathbf{c}+D\mathbf{x}+EB^{-1}(\mathbf{x}-\mathbf{a}),FF^\top), \tag{40}$$
$$p(\mathbf{y}|\,\mathrm{do}(\mathbf{x})) = \mathcal{N}(\mathbf{y}|\mathbf{c}+D\mathbf{x},EE^\top+FF^\top), \tag{41}$$

Comparing Equations 14, 15, 16 with 39, 40, 41 we immediately get the equations for the parameters:

$$\alpha = \mathbf{a}, \tag{42}$$
$$\Sigma = BB^\top, \tag{43}$$
$$\gamma + \Delta\mathbf{x} = \mathbf{c} + (D+EB^{-1})\mathbf{x} - EB^{-1}\mathbf{a}, \tag{44}$$
$$\gamma \overset{\mathbf{x}=\mathbf{0}}{=} \mathbf{c} - EB^{-1}\mathbf{a}, \tag{45}$$
$$\Pi = FF^\top, \tag{46}$$
$$\widetilde{\gamma} = \mathbf{c}, \tag{47}$$
$$\widetilde{\Delta} = D, \tag{48}$$
$$\widetilde{\Pi} = EE^\top + FF^\top. \tag{49}$$

Substituting $\mathbf{a}, \mathbf{c}, D, FF^\top$ and then subtracting Equation 45 from 44 and solving for all $\mathbf{x}$ we get the constraints:

$$\Delta = \widetilde{\Delta} + EB^{-1}, \tag{50}$$

$$\gamma = \widetilde{\gamma} - EB^{-1}\alpha, \tag{51}$$

$$\widetilde{\Pi} = \Pi + EE^\top. \tag{52}$$

With Equation 50 we see that $E = (\Delta - \widetilde{\Delta})B$, which we can just plug into Equations 51 and 52. Finally using Equation 43 to replace $BB^\top$ with Σ in Equation 52 will give the claim. □

7.5 REDUCTION WITH OBSERVED CONFOUNDERS

There are many scenarios where we are interested in estimating the conditional causal effect of interventions given additional covariates $\mathbf{C}$ that might confound treatment and outcome, for example when estimating the efficacy of a vaccine depending on age, weight or gender. We again consider a treatment variable $\mathbf{X} \in \mathscr{X} = \mathbb{R}^N$, an outcome variable $\mathbf{Y} \in \mathscr{Y} = \mathbb{R}^M$, and a set of K latent confounders $Z_1, \ldots, Z_K$ in arbitrary standard measurable spaces (e.g., $\mathbb{R}^d$ or discrete). In addition, let there be L observed confounders $C_1, \ldots, C_L$ of $\mathbf{X}$ and $\mathbf{Y}$, again in arbitrary standard measurable spaces. We allow for arbitrary causal relations and dependencies between the confounders. In the following, we summarize all observed confounders using a single variable $\mathbf{C} = (C_1, \ldots, C_L) \in \mathscr{C}$ and all latent confounders as $\mathbf{Z} = (Z_1, \ldots, Z_K) \in \mathscr{Z}$. We follow a similar sequence of steps as in Section 3.1 to derive a reduced causal model of the following form

$$p(\mathbf{x}, \mathbf{y}, \mathbf{w}, \mathbf{c}) = p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \mathbf{c}) p(\mathbf{x}|\mathbf{w}) p(\mathbf{w}|\mathbf{c}) p(\mathbf{c}). \tag{53}$$

as illustrated in Figure 5 (a–d). At every step, the Bayesian network is observationally equivalent to the ones before, and also interventionally equivalent for interventions on $\mathbf{X}$ and $\mathbf{Y}$.

We can now use a similar approach as in Section 3.2, and in addition marginalize out $\mathbf{W}$ as seen in Figure 5 (g), to convert the causal Bayesian Network into an SCM with structural equations of the form given below

$$\begin{aligned} \mathbf{X} &= F(\mathbf{U}, \mathbf{C}), \\ \mathbf{Y} &= G(\mathbf{V}, \mathbf{X}, \mathbf{U}, \mathbf{C}), \end{aligned} \tag{54}$$

where $\mathbf{U} \sim \mathscr{N}(\mathbf{0}, \mathbf{I}_N) \perp\!\!\!\perp \mathbf{V} \sim \mathscr{N}(\mathbf{0}, \mathbf{I}_M)$ and F and G are two deterministic maps. This is illustrated in Figure 5 (e–h). Again, at every step, the Bayesian network is observationally equivalent to the ones before, and also interventionally equivalent for interventions on $\mathbf{X}$ and $\mathbf{Y}$.

7.6 LINEAR EXPERIMENT

We now show the capabilities of our flow model to learn the model parameters jointly from observational and interventional data. Throughout this experiment we assume $x, y \in \mathbb{R}$. We generate training, validation and test data using the following linear SCM

Observational

$$u \sim \mathscr{N}(0,1) \perp\!\!\!\perp v \sim \mathscr{N}(0,1) \tag{55}$$

$$x_o = 2 \cdot u + 1 \tag{56}$$

$$y_o = 1.5 \cdot v - x_o - 3 \cdot u + 2 \tag{57}$$

Interventional

$$u \sim \mathscr{N}(0,1) \perp\!\!\!\perp v \sim \mathscr{N}(0,1) \tag{58}$$

$$x_i \sim \mathscr{N}(0,1) \tag{59}$$

$$y_i = 1.5 \cdot v - x_i - 3 \cdot u + 2 \tag{60}$$

Since we know that the data is generated by a linear SCM we choose the transformations in our flow model to be linear as well

$$u = f_{a,b}(x) = a \cdot x + b \tag{61}$$

$$x = f_{a,b}^{-1}(u) = \frac{1}{a} \cdot (u - b) \tag{62}$$

$$\tag{63}$$

$$v = g_{x,u;c,d,e,f}(y) = c \cdot y + d \cdot x + e \cdot u + f \tag{64}$$

$$y = g_{x,u;c,d,e,f}^{-1}(v) = \frac{1}{c} \cdot (v - d \cdot x - e \cdot u - f), \tag{65}$$

in this case the volume terms in Equation 26 are simply given by

$$\left| \frac{\delta f_{a,b}(x)}{\delta x} \right| = |a| \tag{66}$$

$$\tag{67}$$

$$\left| \frac{\delta g_{x,u;c,d,e,f}(y)}{\delta y} \right| = |c|\,. \tag{68}$$

$$\tag{69}$$

Given a dataset consisting of observational and interventional data we can optimize the following loss

$$loss = \sum_{o=1}^{N_O} -\log p(x_o, y_o) + \alpha \sum_{i=1}^{N_I} -\log p(\mathrm{do}(x_i), y_i), \tag{70}$$

where

$$\begin{aligned} \log p(x_o, y_o) = {} & \log p_V(c \cdot y_o + d \cdot x_o + e \cdot u + f) \\ & + \log|c| + \log p_U(a \cdot x_o + b) + \log|a|\,, \end{aligned} \tag{71}$$

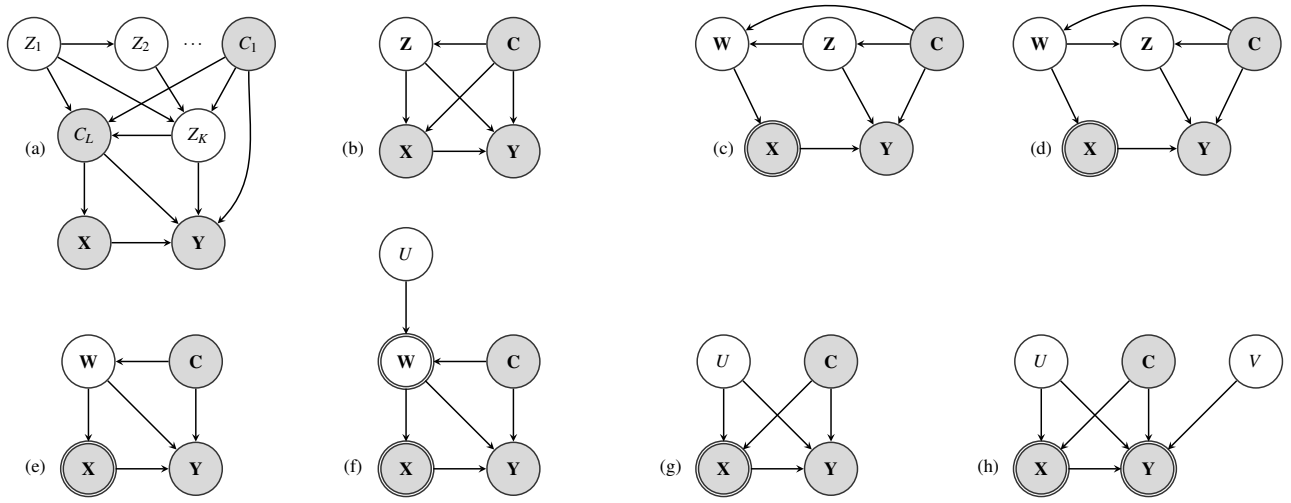

Figure 5: A graphical explanation of our reduction technique in the presence of both observed and latent confounders. (a) We assume a treatment variable $\mathbf{X} \in \mathbb{R}^N$, an outcome variable $\mathbf{Y} \in \mathbb{R}^M$, latent confounders $Z_1, \ldots, Z_K$, and observed confounders $C_1, \ldots, C_L$, with arbitrary causal and probabilistic relations between the confounders. (b) We combine the latent confounders into $\mathbf{Z} \in \mathscr{Z}$ and the observed confounders into $\mathbf{C} \in \mathscr{C}$, and factorize their joint distribution as $p(\mathbf{z} \mid \mathbf{c})p(\mathbf{c})$. (c) We create a copy of $\mathbf{X}$ called $\mathbf{W}$. (d) We refactorize $p(\mathbf{w}, \mathbf{z}, \mathbf{c})$ as $p(\mathbf{z} \mid \mathbf{w}, \mathbf{c})p(\mathbf{w} \mid \mathbf{c})p(\mathbf{c})$. (e) We marginalize over $\mathbf{Z}$. (f) We reparameterize $p(\mathbf{w} \mid \mathbf{c})$ using Theorem 3.2 as a deterministic function, introducing an independent noise variable $\mathbf{U}$. (g) We marginalize over $\mathbf{W}$. (h) We reparameterize $p(\mathbf{y} \mid \mathbf{x}, \mathbf{u}, \mathbf{c})$ with Theorem 3.2 as a deterministic function, introducing an independent noise variable $\mathbf{V}$. Note that at every step (a–h) the models remain observationally equivalent (i.e., $p(\mathbf{c}, \mathbf{x}, \mathbf{y})$ is invariant), and also interventionally equivalent with respect to interventions on $\mathbf{X}$ and $\mathbf{Y}$ (i.e., $p(\mathbf{y}, \mathbf{c} \mid \mathrm{do}(\mathbf{x}))$ and $p(\mathbf{x}, \mathbf{c} \mid \mathrm{do}(\mathbf{y}))$ are invariant).

and

$$\log p(\mathrm{do}(x_i), y_i) = \log \int p_V(c \cdot y_i + d \cdot x_i + e \cdot u + f) \, |c| \, p(u) du + \log p(x_i). \tag{72}$$

We choose $\alpha = N_O / N_I$. In Equation 71, we now use $u = f_{a,b}(x) = a \cdot x + b$ to impute u. Resulting in

$$\log p(x_o, y_o) = \log p_V(c \cdot y_o + d \cdot x_o + e \cdot (a \cdot x_o + b) + f) + \log |c| + \log p_U(a \cdot x_o + b) + \log |a| \,. \tag{73}$$

In Figure 6, we show the training data set (left) and samples from our flow model trained with 100 observational samples and 100 interventional samples (right). Our flow model correctly finds the parameters used in the SCM used to generate the data. After training our flow model is able to generate both observational and interventional samples, as described in Section 4.4.

7.7 BACKGROUND: NORMALIZING FLOWS

Normalizing flows are based on the idea of transforming samples from a simple distribution into samples from a complex distribution using the change of variable formula [Rezende and Mohamed, 2015, Tabak and Turner, 2013]:

$$p(\mathbf{x}) = p_Z(f(\mathbf{x})) \left| \det \left(\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}} \right) \right|, \tag{74}$$

where $\mathbf{z} = f(\mathbf{x})$ is a bijective map $f : \mathscr{X} \to \mathscr{Z}$, $p_Z(\mathbf{z})$ a simple prior distribution, and $\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}}$ the Jacobian with respect to $\mathbf{x}$. The transformation $f(\mathbf{x})$ is commonly composed of K transformations $f(\mathbf{x}) = f_K \circ \cdots \circ f_1(\mathbf{x})$ to increase the overall expressivity of $f(\mathbf{x})$. The choice of $f(\mathbf{x})$ is restricted by the computational complexity of calculating the Jacobian $\frac{\delta f(\mathbf{x})}{\delta \mathbf{x}}$. In recent years, a multitude of transformations with easy to compute Jacobians have been developed, for an overview see Kobyzev et al. [2020], Papamakarios et al. [2019].

In this paper we will use neural spline flows [Durkan et al., 2019, Dolatabadi et al., 2020]. Neural spline flows have two major advantages: 1. A better functional flexibility than affine transformations ($\mathbf{y} = \mathbf{s}\mathbf{x} + \mathbf{t}$), 2. A numerically stable, analytic inverse that has the same computational and space complexities as the forward operation. While Durkan et al. [2019] use quadratic, cubic, and, rational quadratic functions whose inversion is done after solving degree 2 or 3 polynomial equations, Dolatabadi et al. [2020] show that piecewise linear rational splines can perform competitively with these methods without requiring a polynomial equation to be solved in the inversion. Because of its reduced computational cost, we will use linear rational splines throughout this paper.

Consider a set of monotonically increasing points $\{(x^{(k)}, y^{(k)})\}_{k=0}^{K}$ called knots and a set of derivatives at each of the points $\{d^{(k)}\}_{k=0}^{K}$. For each bin $[x^{(k)}, x^{(k+1)}]$ we want

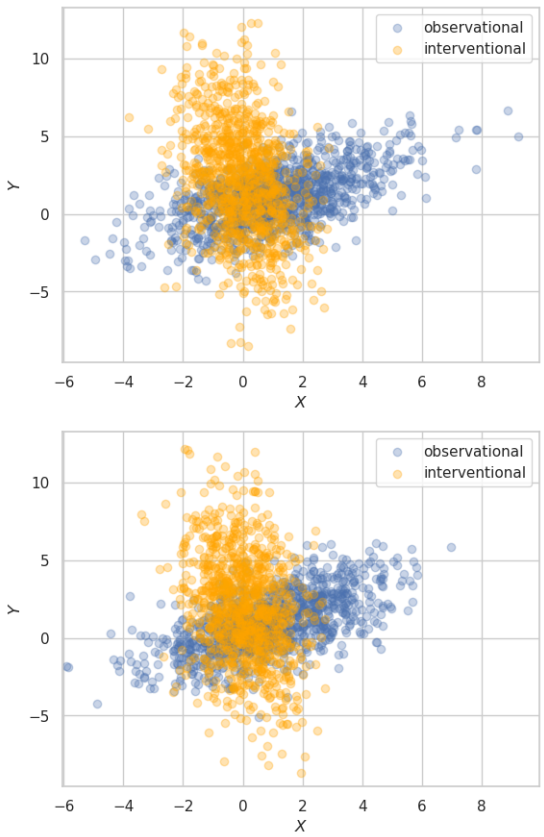

Figure 6: Top: 1000 observational and 1000 interventional samples generated from the linear SCM in Section 7.6. Bottom: 1000 observational and 1000 interventional samples generated from our flow model trained with 100 observational and 100 interventional samples.

to find a linear rational function of the form $\frac{ax+b}{cx+d}$ that fit the given points and derivatives.

Algorithm 1 Fuhr and Kallay [1992] Linear Rational Spline Interpolation of Monotonic data in the interval $\left[x^{(k)}, x^{(k+1)}\right]$

Input: $x^{(k)} < x^{(k+1)}$, $y^{(k)} < y^{(k+1)}$, $d^{(k)} > 0$, $d^{(k+1)} > 0$

1: set $w^{(k)} > 0$

2: set $0 < \lambda^{(k)} < 1$

3: $w^{(k)} = \sqrt{\frac{d^{(k)}}{d^{(k+1)}}} w^{(k)}$

4: $y^m = \frac{w^{(k)} y^{(k)} \left(1-\lambda^{(k)}\right) + w^{(k+1)} y^{(k+1)} \lambda^{(k)}}{w^{(k)} \left(1-\lambda^{(k)}\right) + w^{(k+1)} \lambda^{(k)}}$

5: $w^{(m)} = \left(\lambda^{(k)} w^{(k)} d^{(k)} + \left(1-\lambda^{(k)}\right) w^{(k+1)} d^{(k+1)}\right) \frac{x^{(k+1)} - x^{(k)}}{y^{(k+1)} - y^{(k)}}$

Return: $\lambda^{(k)}, w^{(k)}, w^{(m)}, w^{(k+1)}, y^{(m)}$

The values returned by Algorithm 1 are subsequentely used to express the following linear rational spline function

$$f(\phi) = \begin{cases} \frac{w^{(k)} y^{(k)} (\lambda^{(k)} - \phi) + w^{(m)} y^{(m)} \phi}{w^{(k)} (\lambda^{(k)} - \phi) + w^{(m)} \phi} & 0 \le \phi \le \lambda^{(k)} \\ \frac{w^{(m)} y^{(m)} (1-\phi) + w^{(k+1)} y^{(k+1)} (\phi - \lambda^{(k)})}{w^{(m)} (1-\phi) + w^{(k+1)} (\phi - \lambda^{(k)})} & \lambda^{(k)} \le \phi \le 1 \end{cases} \tag{75}$$

where $\phi = (x - x^{(k)}) / (x^{(k+1)} - x^{(k)})$.

Spline flows have two hyperparameters, the boundary B of the interval $[-B, B]$ and the number of bins K. Outside of the interval $[-B, B]$ the identity function is used. Using Equation 74 we can update the parameters of the neural spline flow using maximum-likelihood estimation in combination with gradient descent. In the case where $\mathbf{x}$ has two or more dimensions either coupling layers [Dinh et al., 2017] or autoregressive layers [Papamakarios et al., 2017] can be used.

At multiple points in this paper we are required to estimate conditional distributions, e.g. $p(\mathbf{y}|\mathbf{x})$, where we will use conditional normalizing flows to estimate conditional probabilities. We consider the mapping $f : \mathscr{X} \times \mathscr{Y} \to \mathscr{Z}$, which is bijective in $\mathscr{Y}$ and $\mathscr{Z}$, and a simple prior distribution $p_Z(\mathbf{z})$. Again, using the change of variable formula we can express the conditional distributions $p(\mathbf{y}|\mathbf{x})$ as follows

$$p(\mathbf{y}|\mathbf{x}) = p_Z(f_x(\mathbf{y})) \left| \det \left(\frac{\delta f_x(\mathbf{y})}{\delta \mathbf{y}} \right) \right|. \tag{76}$$

The conditional version of the linear rational spline transformation uses a neural network to predict the derivatives $\mathbf{d}$, width $\mathbf{w}$, height $\mathbf{h}$, and λ from $\mathbf{x}$: $\mathbf{w}, \mathbf{h}, \mathbf{d}, \lambda = NN_\theta(\mathbf{x})$.

7.8 SIMULATION DETAILS: NONLINEAR EXPERIMENTS WITHOUT OBSERVED CONFOUNDERS

The generation of observational and interventional samples follows Mooij et al. [2016]. Instead of using Gaussian processes to model the causal mechanisms we use two randomly initialized neural networks, NN_1 and NN_2.

Sampling from a random distribution
We use the following steps in order to generate samples from a random distribution

1. $X \sim \mathcal{N}(0,1)$
2. sort X in ascending order $= \overrightarrow{X}$
3. Sample from Gaussian Process: $F \sim \mathcal{N}(0, K_\theta(\overrightarrow{X}) + \sigma^2 I)$, where for the kernel K_θ we use the squared exponential covariance function with automatic relevance determination kernel
4. use the trapezoidal rule to calculate the cumulative integral of $\exp(F)$, we obtain a vector G where each element G_i corresponds to $G_i = \int_{\overrightarrow{X_1}}^{\overrightarrow{X_i}} \exp(F(x))dx$

We will denote this whole sampling procedure by $G \sim \mathscr{RD}(\theta, \sigma)$, where we sample θ from a Gamma distribution $\Gamma(a,b)$ and set $\sigma = 0.0001$.

Generate observational and interventional data
1. Sample from latent variables

$$\theta_{E_X} \sim \Gamma(a_{E_X}, b_{E_X}), \tag{77}$$
$$\theta_{E_Y} \sim \Gamma(a_{E_Y}, b_{E_Y}), \tag{78}$$
$$\theta_Z \sim \Gamma(a_Z, b_Z), \tag{79}$$
$$E_X \sim \mathscr{RD}(\theta_{E_X}, \sigma), \tag{80}$$
$$E_Y \sim \mathscr{RD}(\theta_{E_Y}, \sigma), \tag{81}$$
$$\mathbf{Z} \sim \mathscr{RD}(\theta_Z, \sigma). \tag{82}$$

2. Generate $X_{\text{observational}}$

$$X_{\text{observational}} = NN_1(E_X, \mathbf{Z}). \tag{83}$$

3. Normalize $X_{\text{observational}}$

$$X_{\text{observational}} = \frac{X_{\text{observational}} - \mathbb{E}[X_{\text{observational}}]}{\sqrt{\mathbb{V}[X_{\text{observational}}]}}. \tag{84}$$

4. Generate $Y_{\text{observational}}$

$$Y_{\text{observational}} = NN_2(X_{\text{observational}}, E_Y, \mathbf{Z}). \tag{85}$$

5. Sample from latent variables

$$E_Y \sim \mathscr{RD}(\theta_{E_Y}, \sigma) \tag{86}$$
$$\mathbf{Z} \sim \mathscr{RD}(\theta_Z, \sigma) \tag{87}$$

6. Generate $X_{\text{interventional}}$

$$\theta_X \sim \Gamma(a_X, b_X), \tag{88}$$
$$X_{\text{interventional}} \sim \mathscr{RD}(\theta_X, \sigma). \tag{89}$$

7. Normalize $X_{\text{interventional}}$

$$X_{\text{interventional}} = \frac{X_{\text{interventional}} - \mathbb{E}[X_{\text{interventional}}]}{\sqrt{\mathbb{V}[X_{\text{interventional}}]}}. \tag{90}$$

8. Generate $Y_{\text{interventional}}$

$$Y_{\text{inter}} = NN_2(X_{\text{inter}}, E_Y, \mathbf{Z}). \tag{91}$$

9. Generate noise

$$\varepsilon_{x,\text{observational}} \sim \mathcal{N}(0,1), \tag{92}$$
$$\varepsilon_{x,\text{interventional}} \sim \mathcal{N}(0,1), \tag{93}$$
$$\theta_{\varepsilon_x} \sim \Gamma(a_{\varepsilon_x}, b_{\varepsilon_x}), \tag{94}$$
$$\varepsilon_{y,\text{observational}} \sim \mathcal{N}(0,1), \tag{95}$$
$$\varepsilon_{y,\text{interventional}} \sim \mathcal{N}(0,1), \tag{96}$$
$$\theta_{\varepsilon_y} \sim \Gamma(a_{\varepsilon_y}, b_{\varepsilon_y}). \tag{97}$$

10. Add noise

$$X'_{\text{observational}} = X_{\text{observational}} + \theta_{\varepsilon_x}\varepsilon_{x,\text{observational}}, \tag{98}$$
$$X'_{\text{interventional}} = X_{\text{interventional}} + \theta_{\varepsilon_x}\varepsilon_{x,\text{interventional}}, \tag{99}$$
$$Y'_{\text{observational}} = Y_{\text{observational}} + \theta_{\varepsilon_y}\varepsilon_{y,\text{observational}}, \tag{100}$$
$$Y'_{\text{interventional}} = Y_{\text{interventional}} + \theta_{\varepsilon_y}\varepsilon_{y,\text{interventional}}. \tag{101}$$

11. Normalize Y jointly

$$Y' = [Y'_{\text{observational}}, Y'_{\text{interventional}}], \tag{102}$$
$$Y'_{\text{observational}} = \frac{Y'_{\text{observational}} - \mathbb{E}[Y']}{\sqrt{\mathbb{V}[Y']}}, \tag{103}$$
$$Y'_{\text{interventional}} = \frac{Y'_{\text{interventional}} - \mathbb{E}[Y']}{\sqrt{\mathbb{V}[Y']}}. \tag{104}$$

The two neural networks NN_1 and NN_2 are Multi-layer perceptrons with a single hidden layer. The hidden layer contains 1024 units. The input layer and the hidden layer use a ReLU activation function. The weights and biases for both neural networks are uniformly sampled from the interval $[-1,1]$. We choose the other simulation parameters as follows: $a_{E_X} = a_{E_Y} = a_Z = a_X = 5$, $a_{\varepsilon_x} = a_{\varepsilon_y} = 2$, $b_{E_X} = b_{E_Y} = b_Z = b_X = b_{\varepsilon_x} = b_{\varepsilon_y} = 0.1$, $\sigma = 0.0001$

7.9 SIMULATION DETAILS: NONLINEAR EXPERIMENTS WITH OBSERVED CONFOUNDERS

In order to simulate data with additional observed confounders, we first generate $\mathbf{C}$

$$\theta_C = \Gamma(a_C, b_C), \tag{105}$$
$$\mathbf{C} \sim \mathscr{RD}(\theta_C, \sigma), \tag{106}$$

where $a_C = 10$ and $b_C = 1$. In addition, we modify steps 2,4 and 8 as follows

$$X_{\text{observational}} = NN_1(E_X, \mathbf{Z}, \mathbf{C}), \tag{107}$$
$$Y_{\text{observational}} = NN_2(X_{\text{observational}}, E_Y, \mathbf{Z}, \mathbf{C}), \tag{108}$$
$$Y_{\text{inter}} = NN_2(X_{\text{inter}}, E_Y, \mathbf{Z}, \mathbf{C}). \tag{109}$$

7.10 NONLINEAR EXPERIMENT RESULTS WITHOUT OBSERVED CONFOUNDERS

Dataset 1: # of confounders = 1, random seed = 6

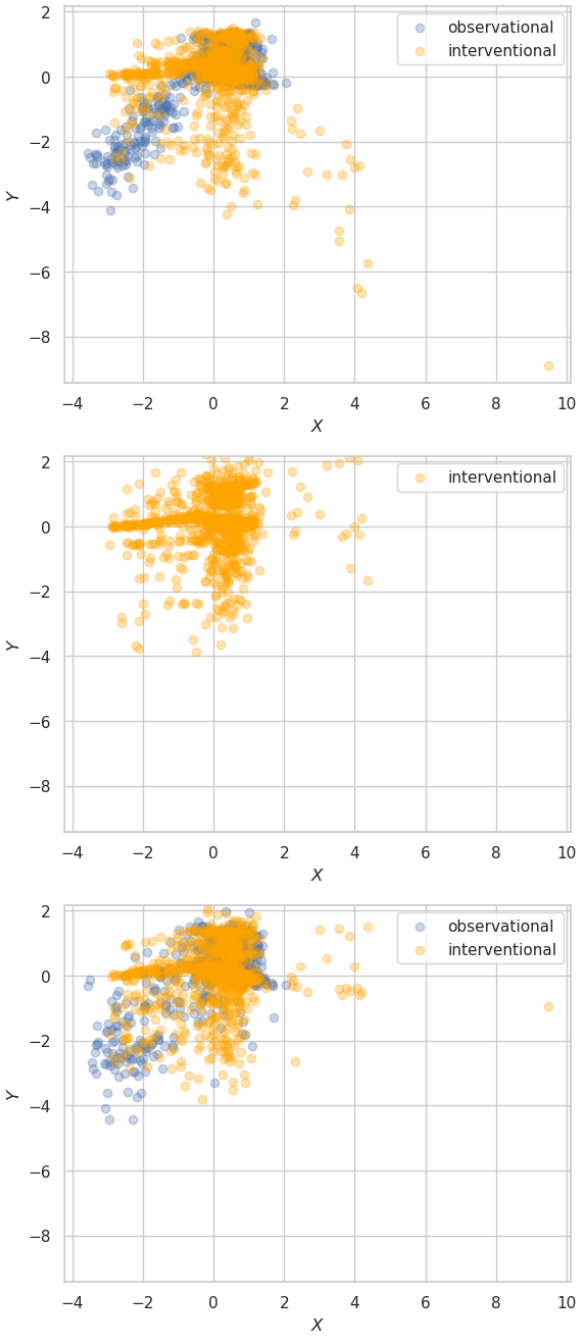

Figure 7: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a flow model trained with 50 interventional samples. Bottom: Observational and interventional samples from a flow model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

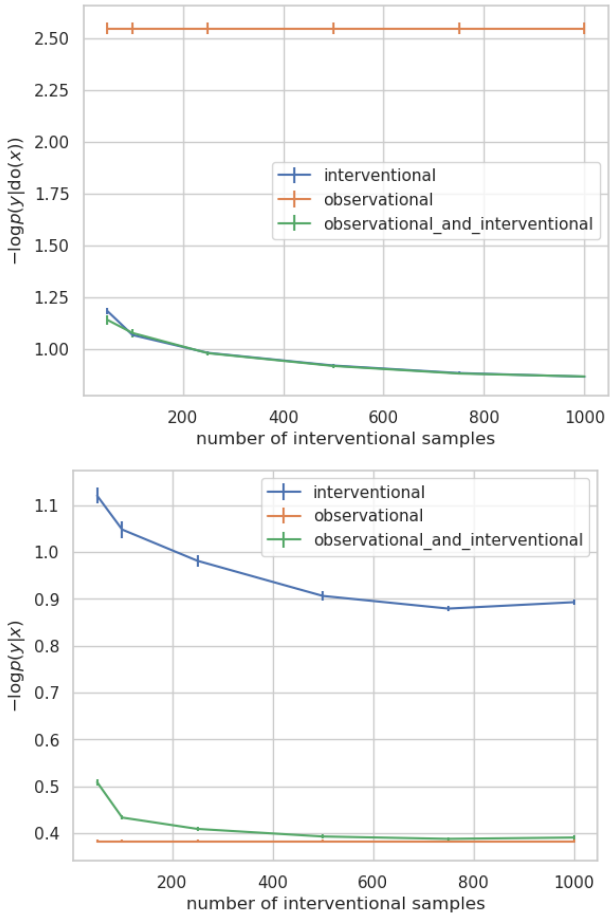

Figure 8: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 2: # of confounders = 1, random seed = 8

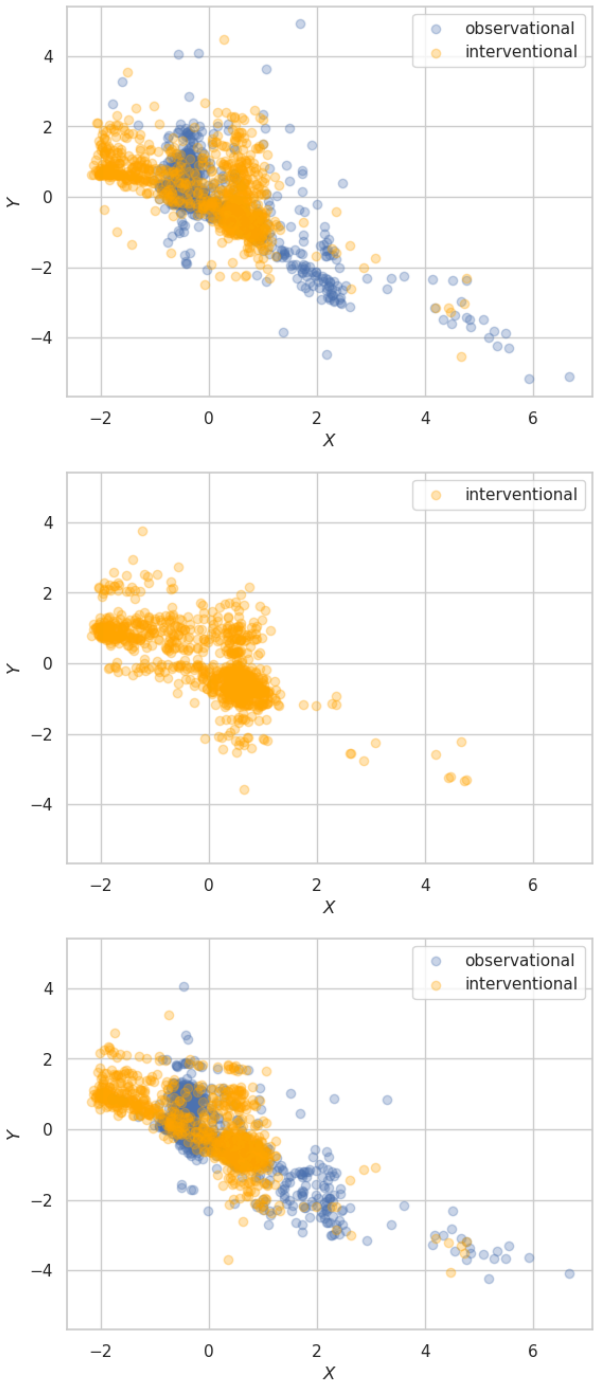

Figure 9: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

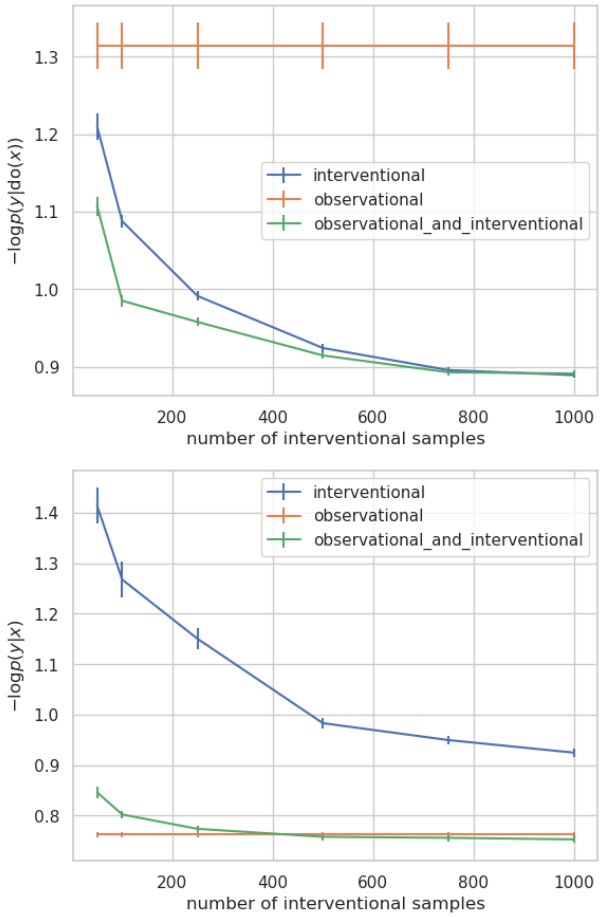

Figure 10: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 3: # of confounders = 2, random seed = 7

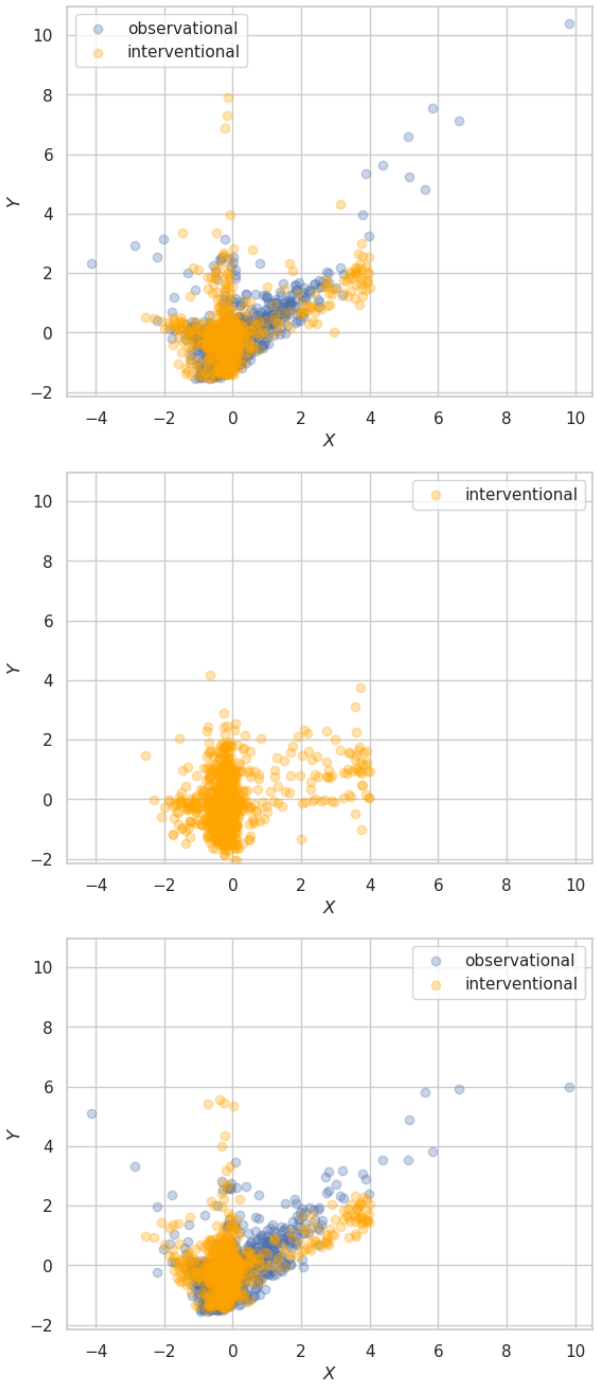

Figure 11: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

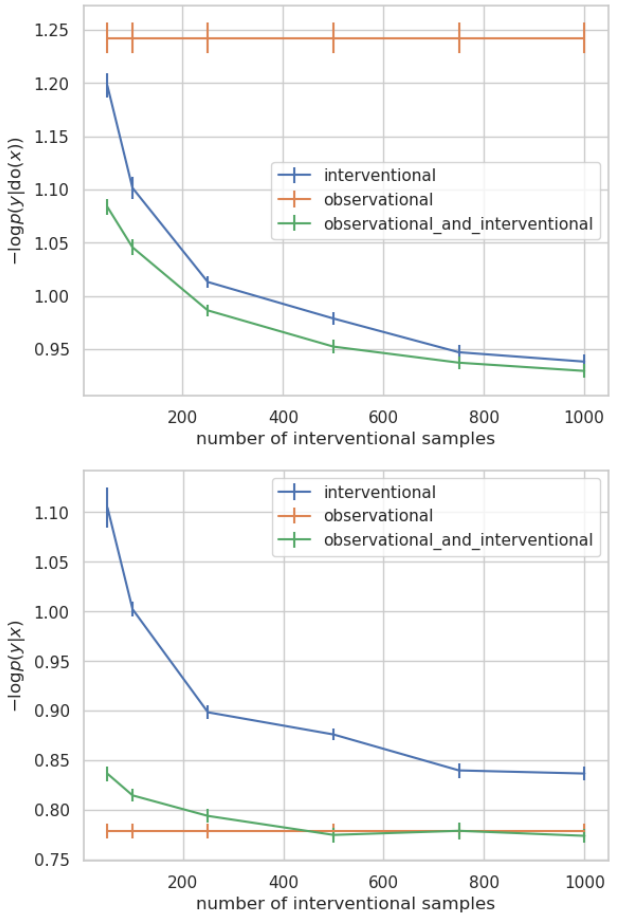

Figure 12: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

7.11 DATASET 4: 3 CONFOUNDERS, RANDOM SEED = 1

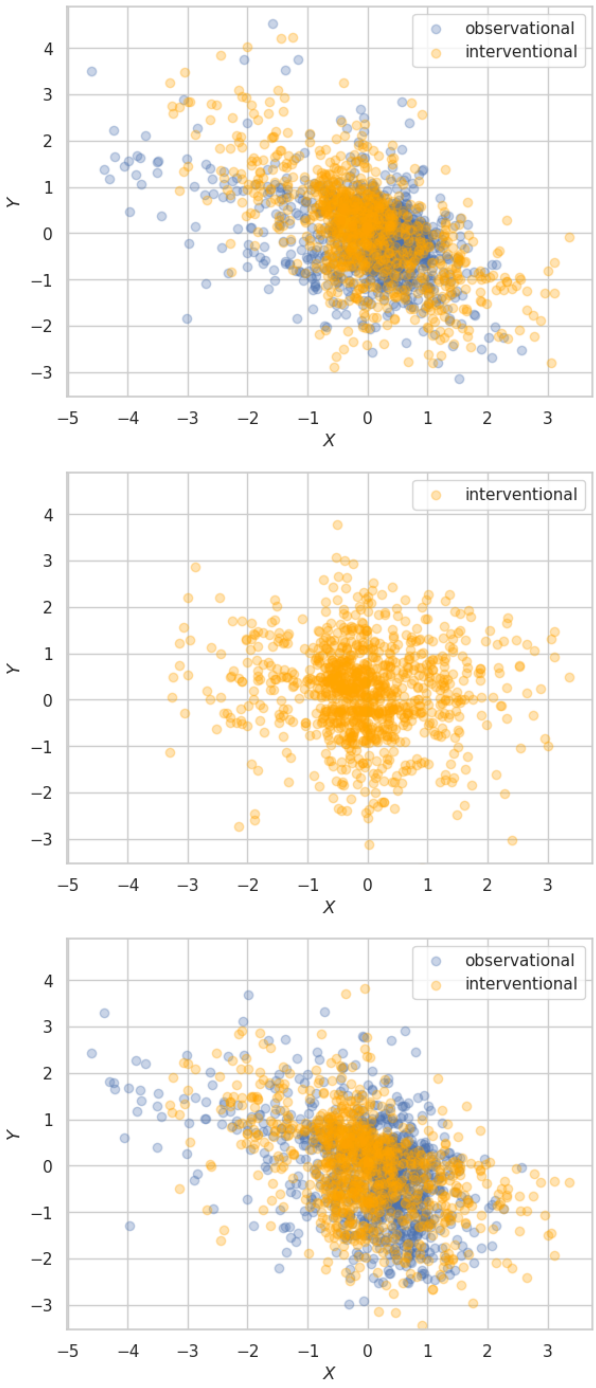

Figure 13: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

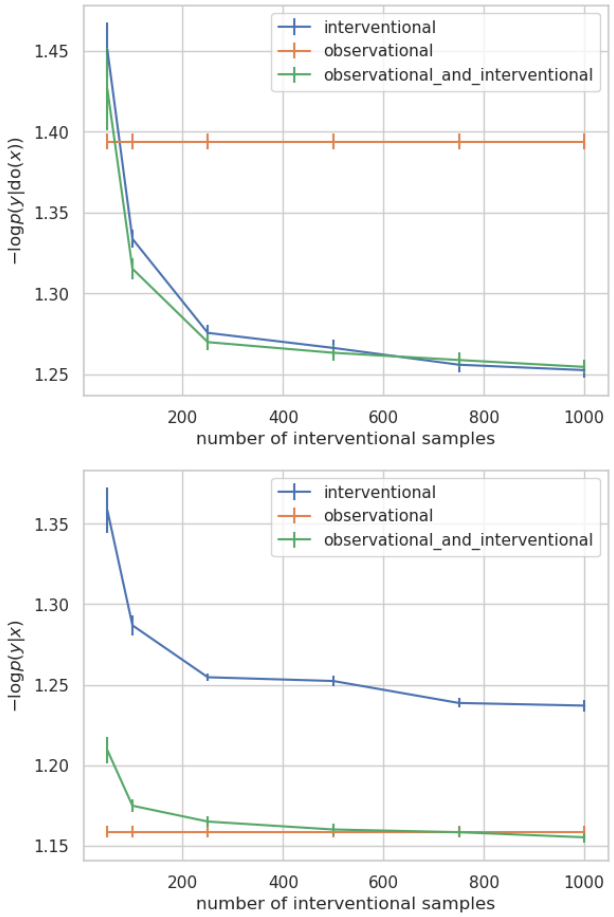

Figure 14: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 5: # of confounders = 4, random seed = 0

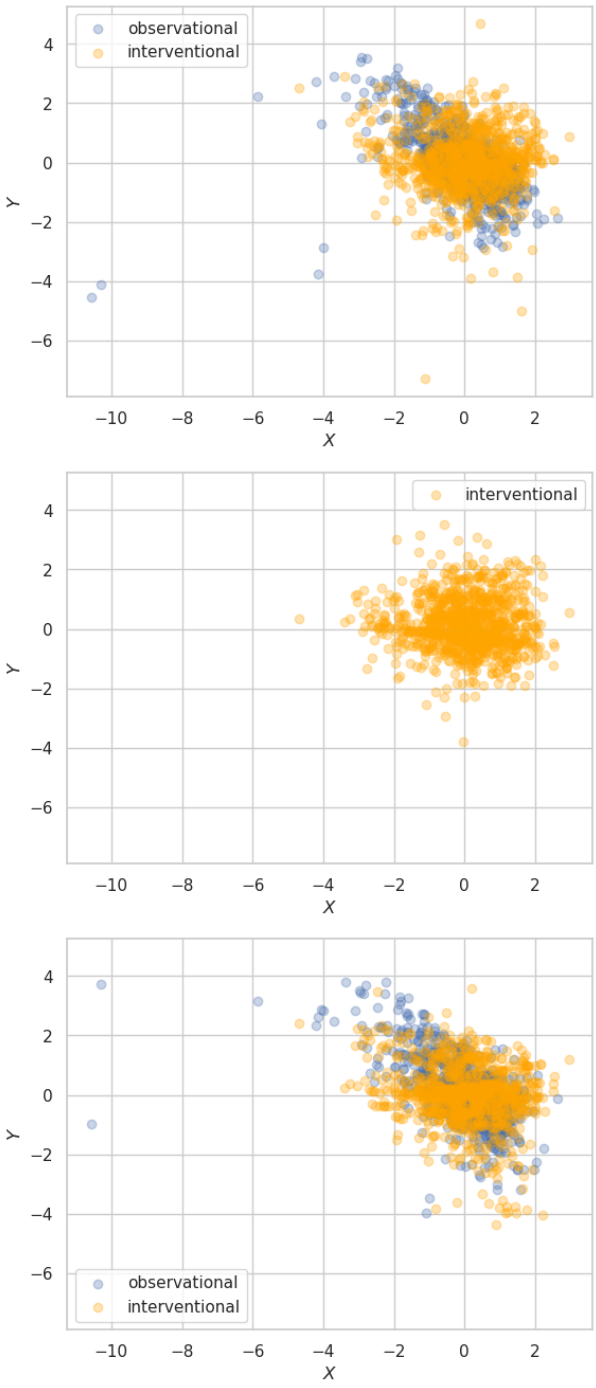

Figure 15: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

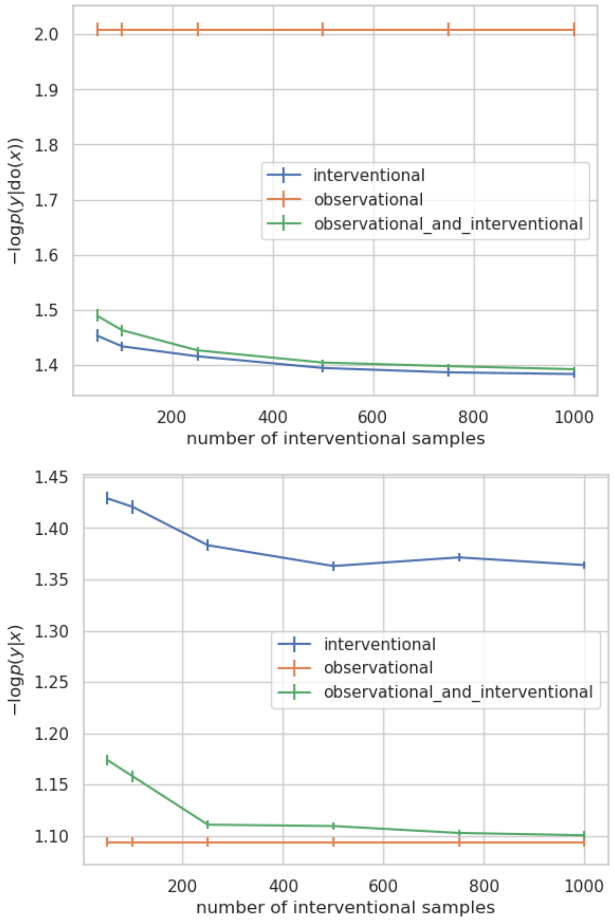

Figure 16: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 6: # of confounders = 4, random seed = 7

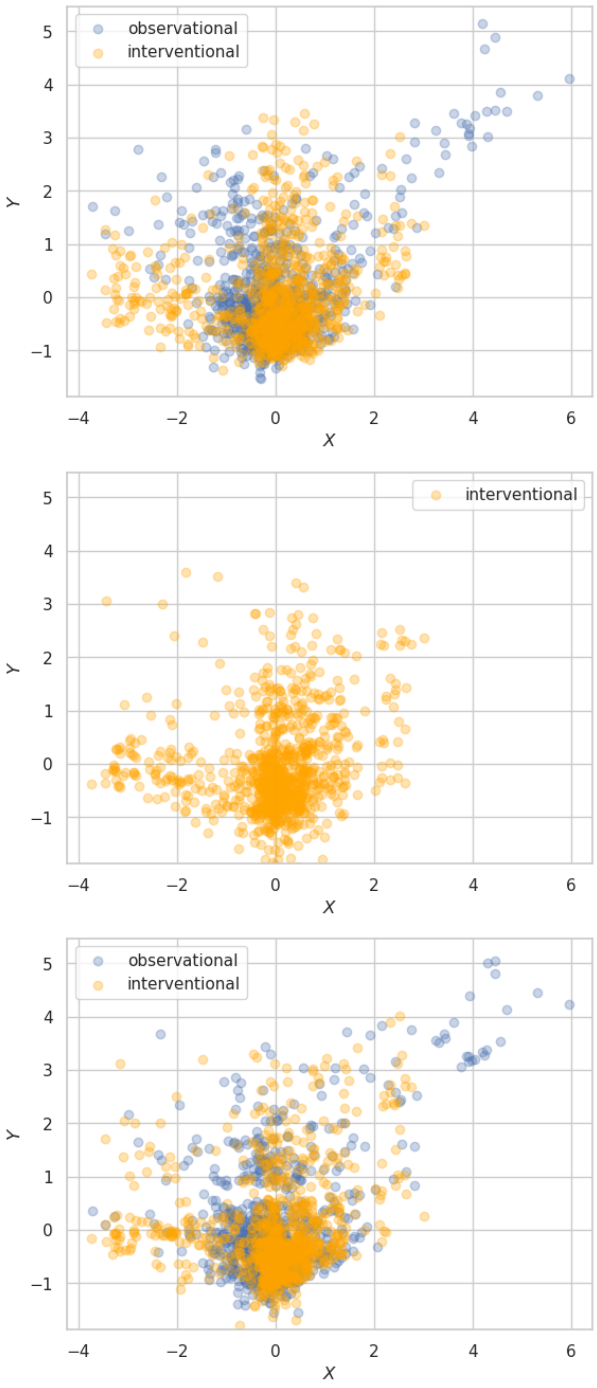

Figure 17: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

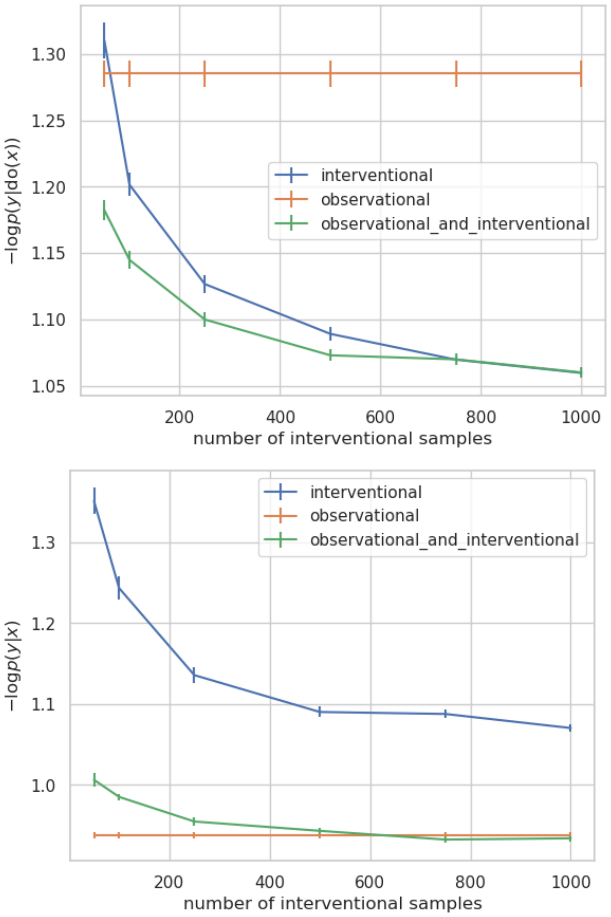

Figure 18: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 7: # of confounders = 5, random seed = 5

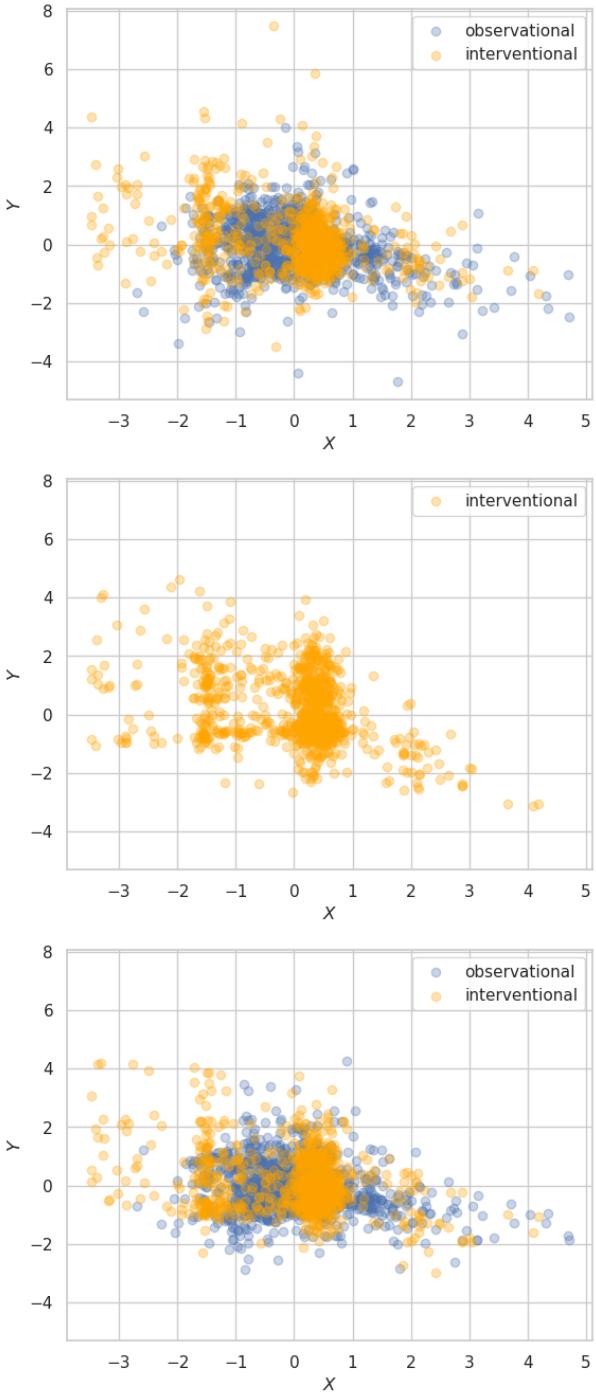

Figure 19: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

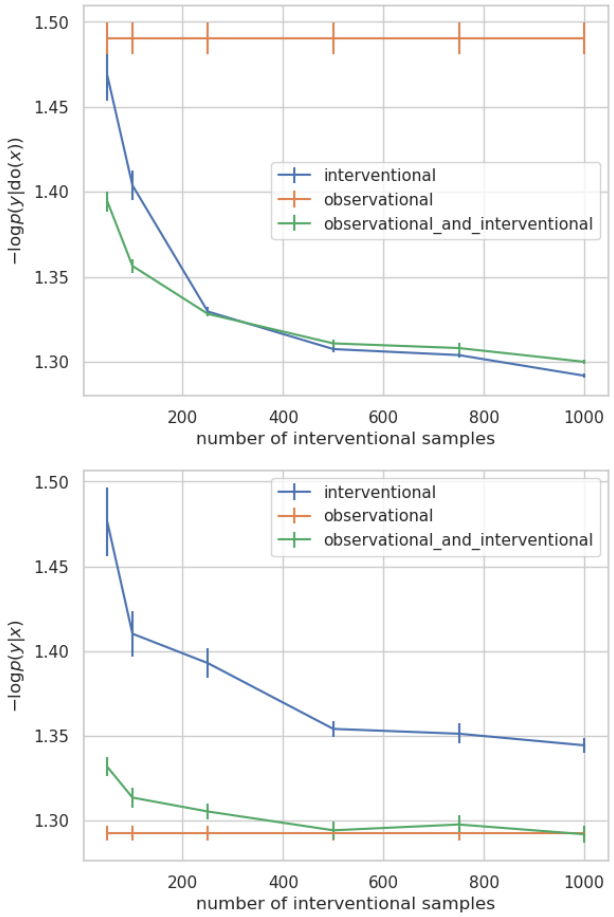

Figure 20: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 8: # of confounders = 5, random seed = 9

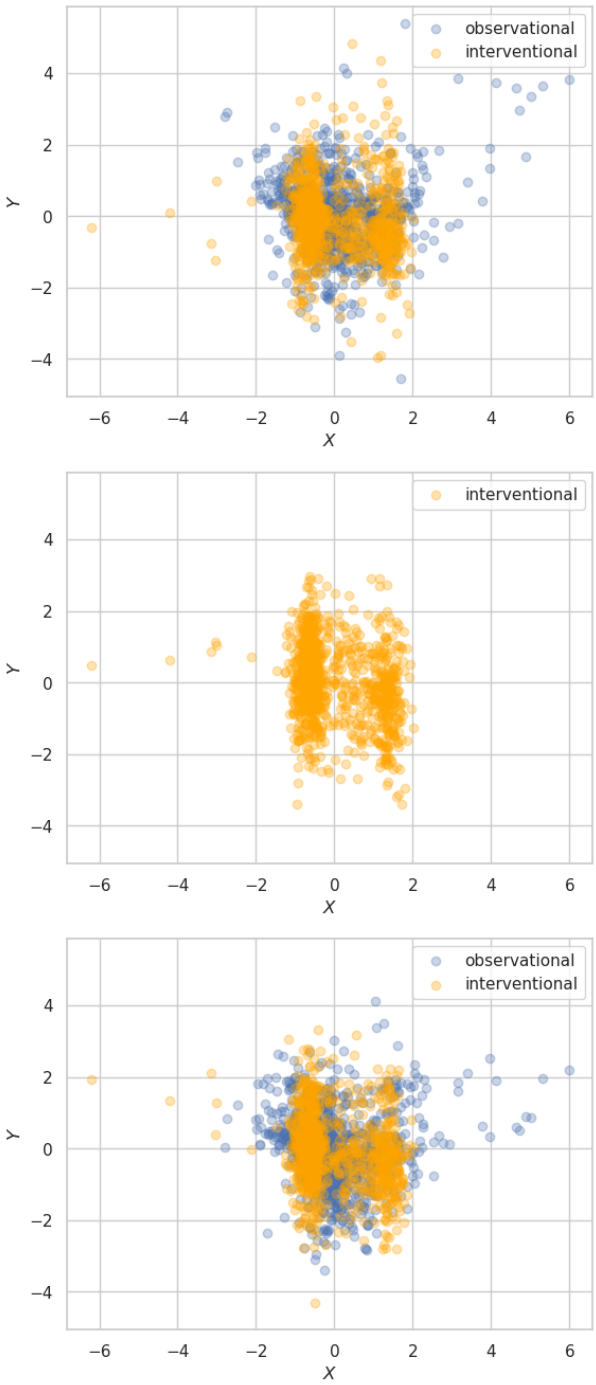

Figure 21: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

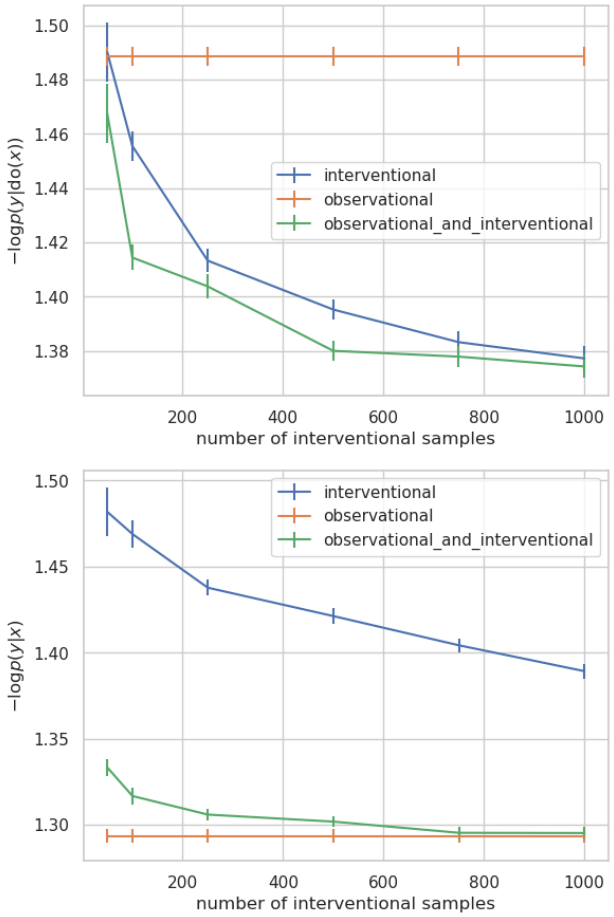

Figure 22: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 9: # of confounders = 7, random seed = 0

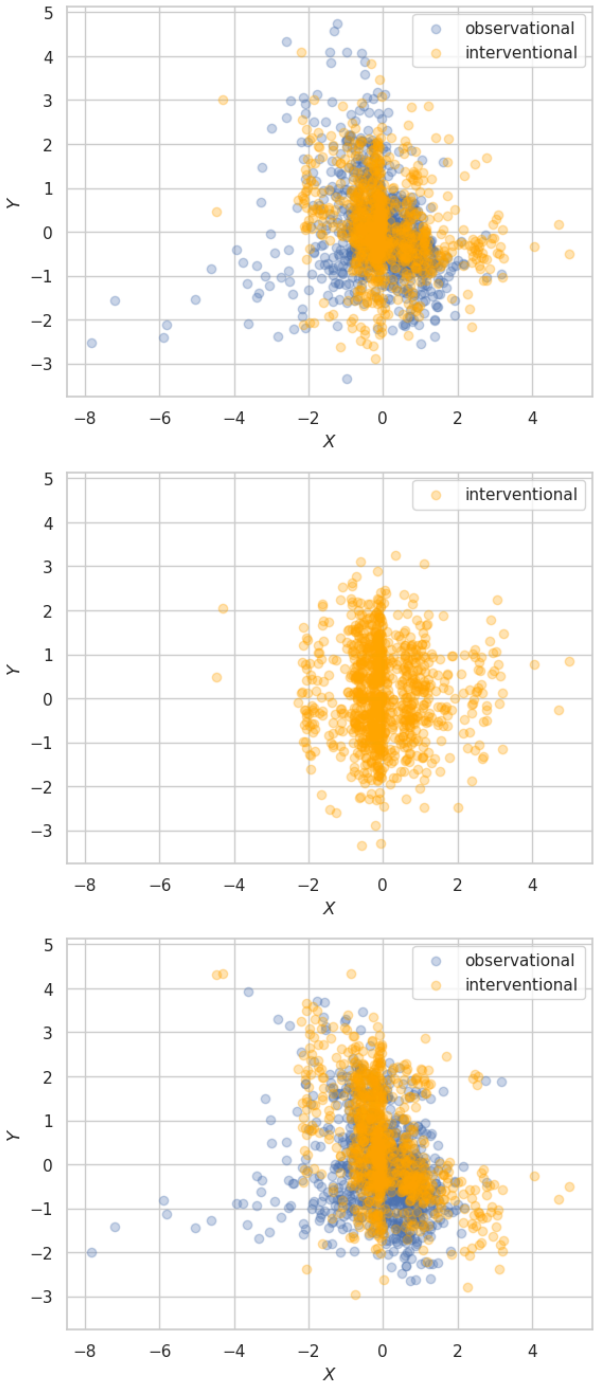

Figure 23: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

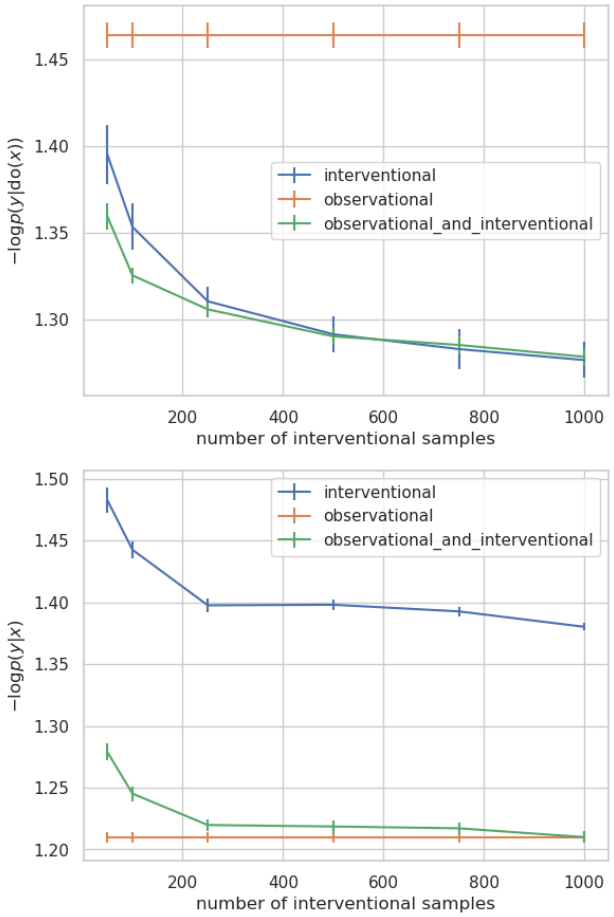

Figure 24: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 10: # of confounders = 7, random seed = 5

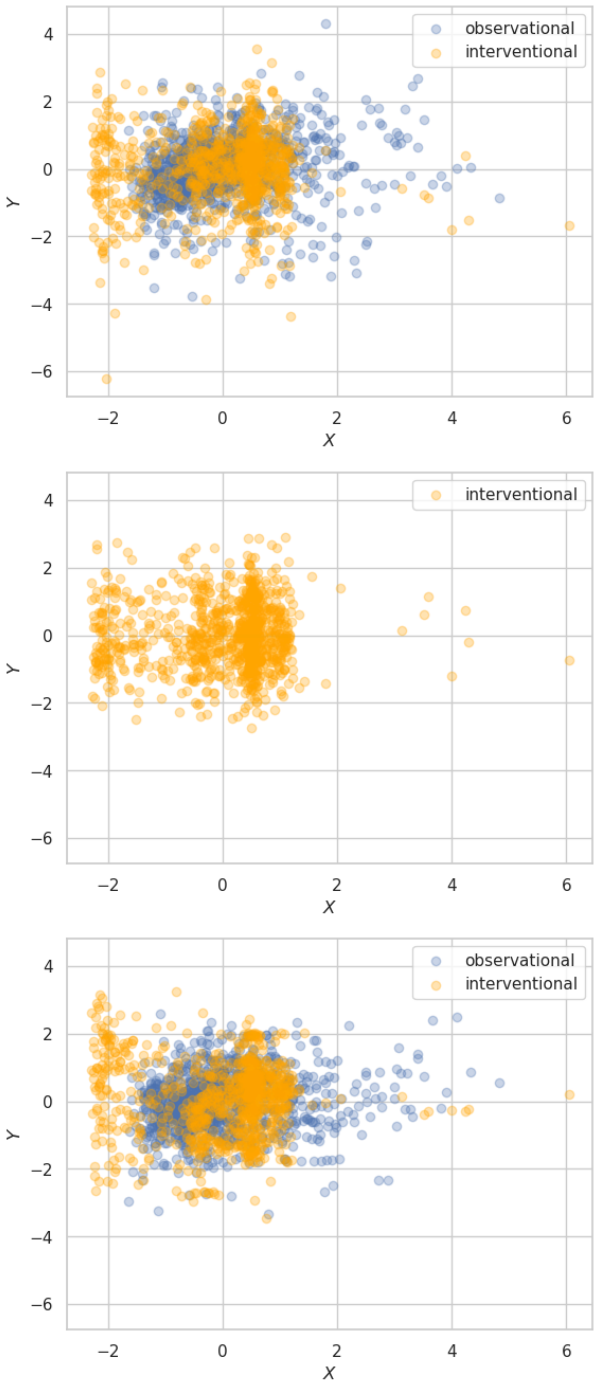

Figure 25: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

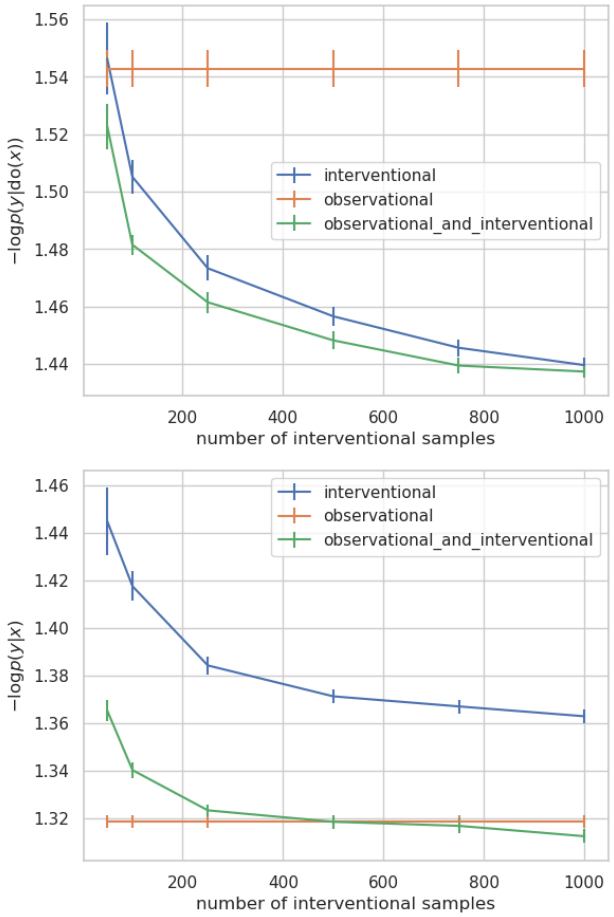

Figure 26: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

7.12 NONLINEAR EXPERIMENT RESULTS WITH OBSERVED CONFOUNDERS

Dataset 11: # of latent confounders = 1, # of observed confounders = 3, random seed = 7

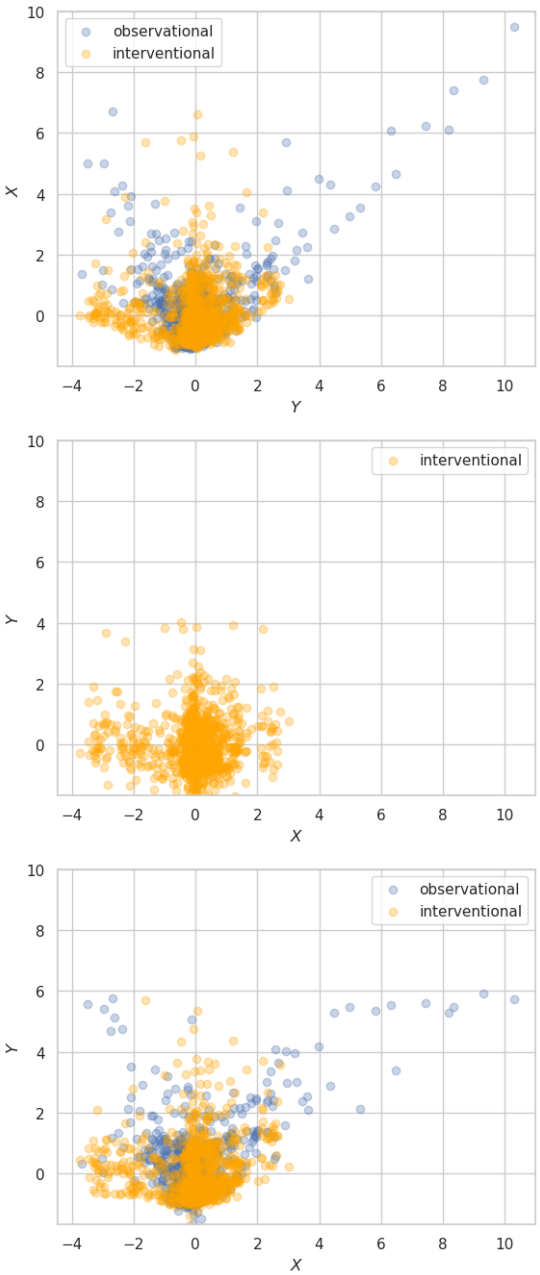

Figure 27: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

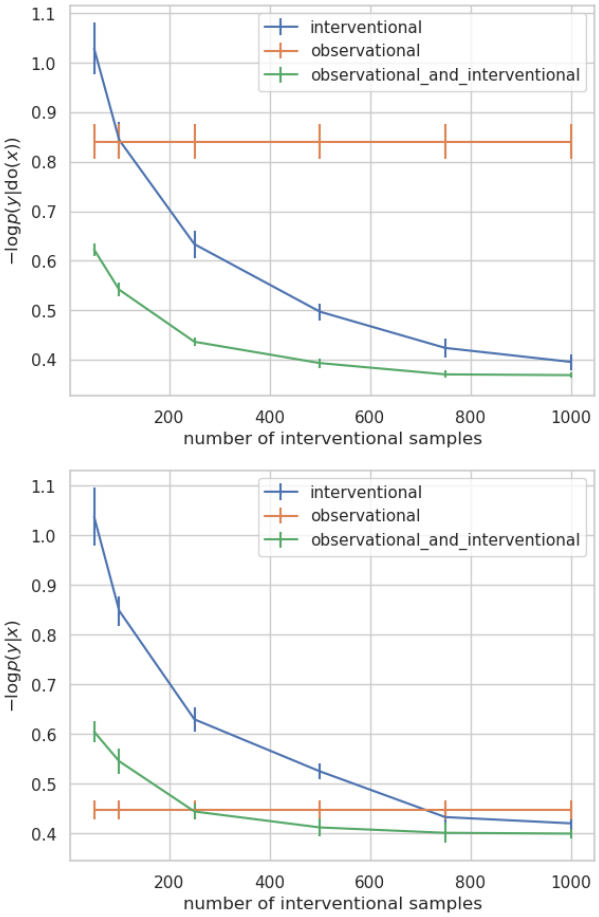

Figure 28: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 12: # of latent confounders = 1, # of observed confounders = 3, random seed = 9

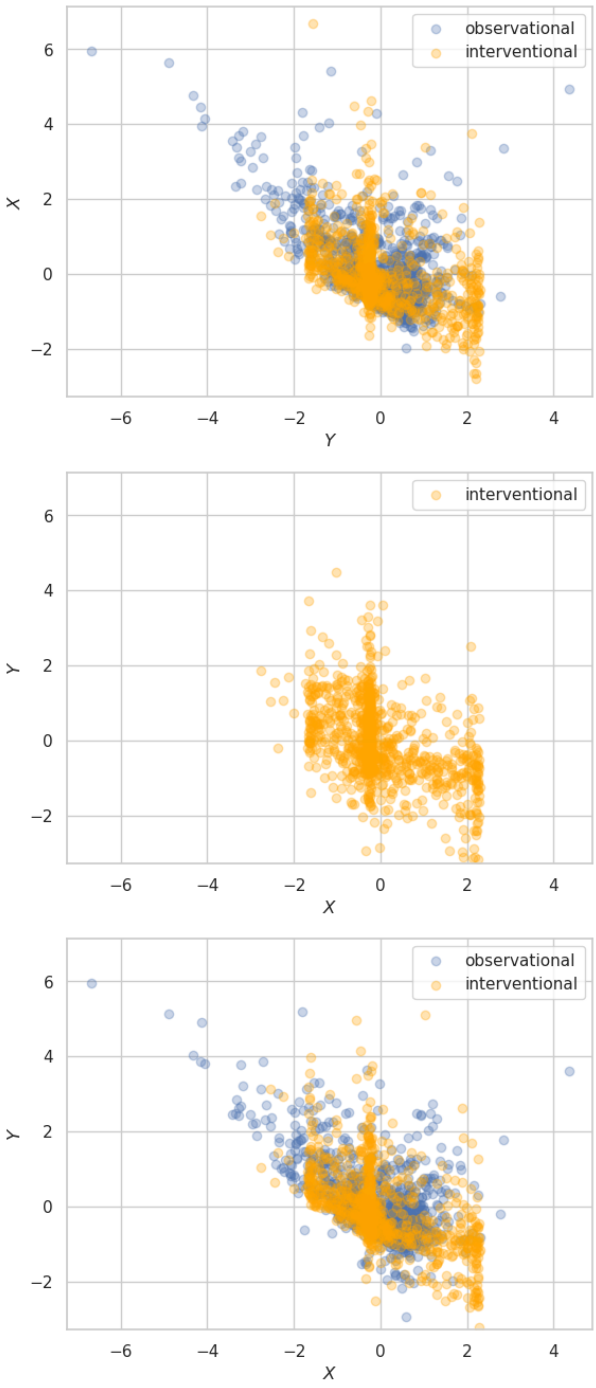

Figure 29: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

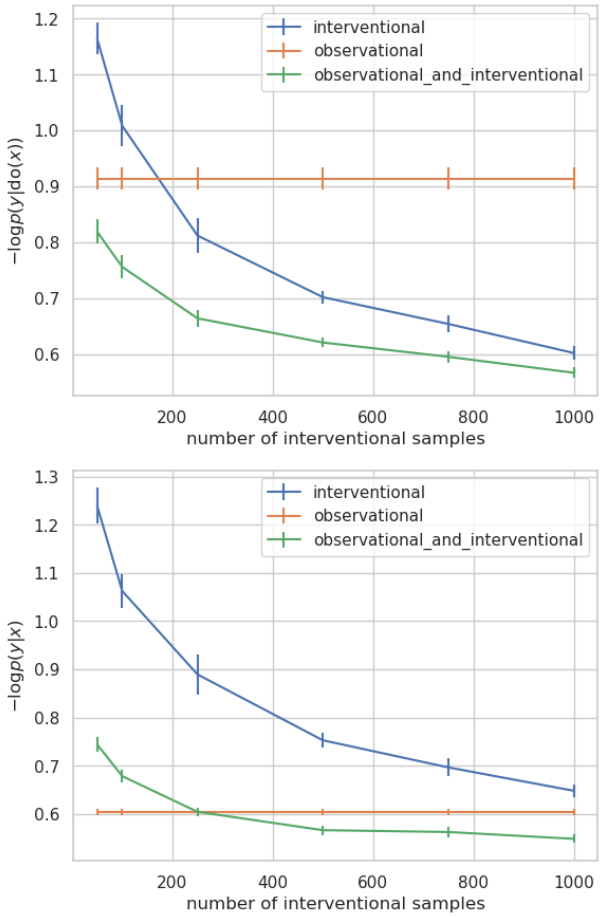

Figure 30: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 13: # of latent confounders = 2, # of observed confounders = 1, random seed = 0

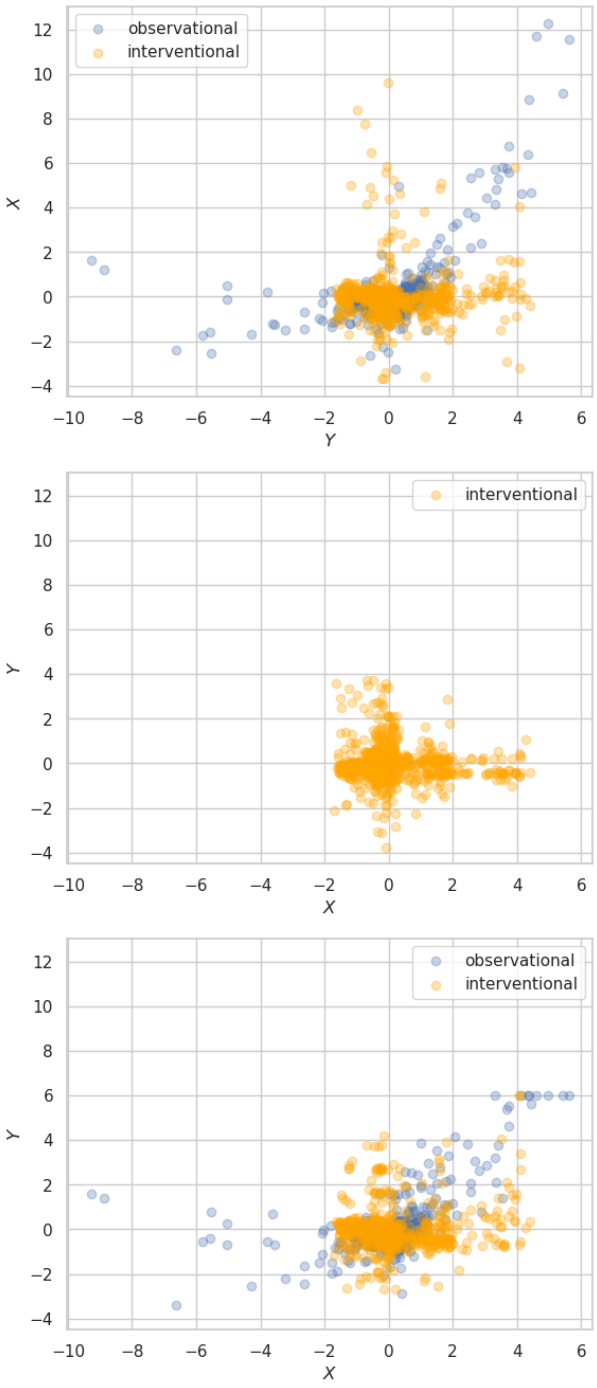

Figure 31: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

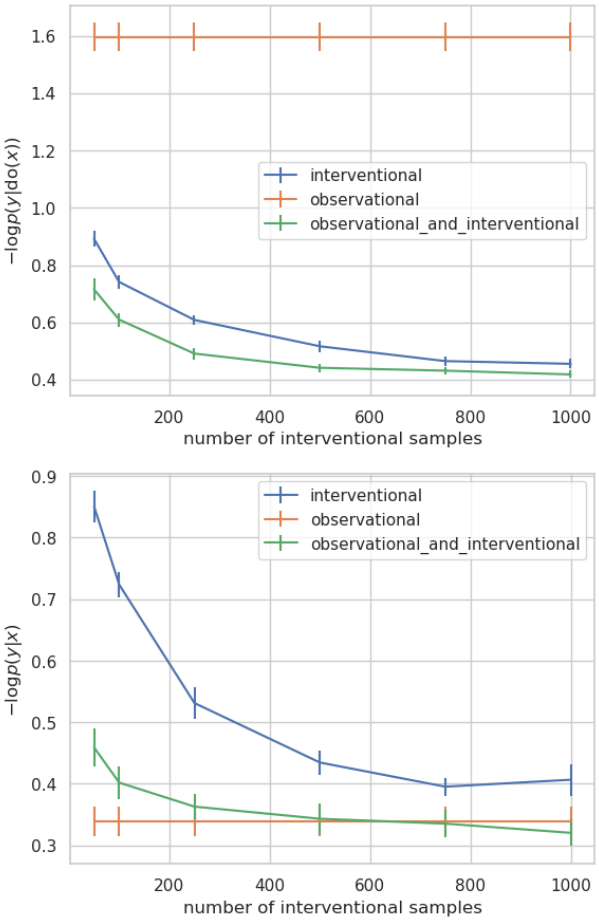

Figure 32: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 14: # of latent confounders = 3, # of observed confounders = 3, random seed = 5

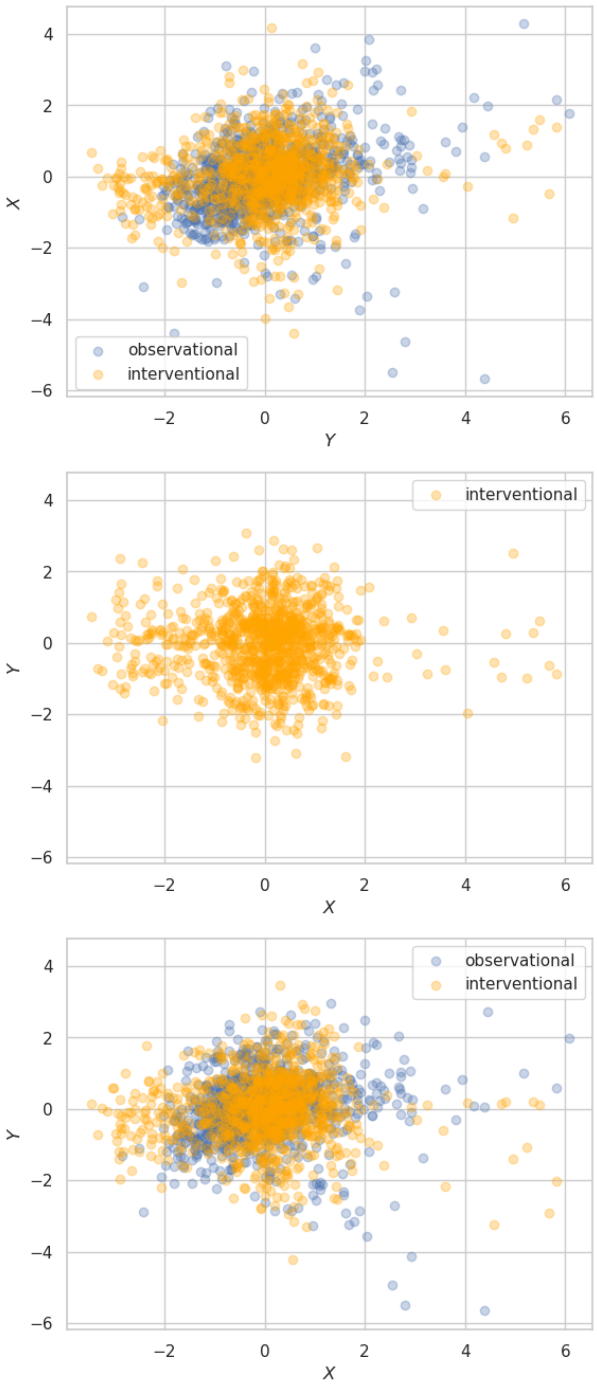

Figure 33: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

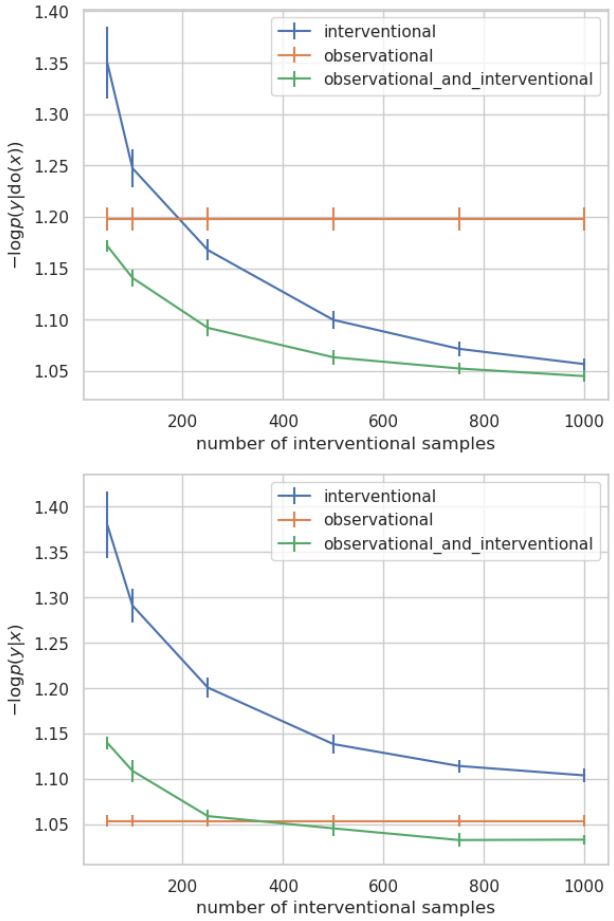

Figure 34: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.

Dataset 15: # of latent confounders = 4, # of observed confounders = 4, random seed = 2

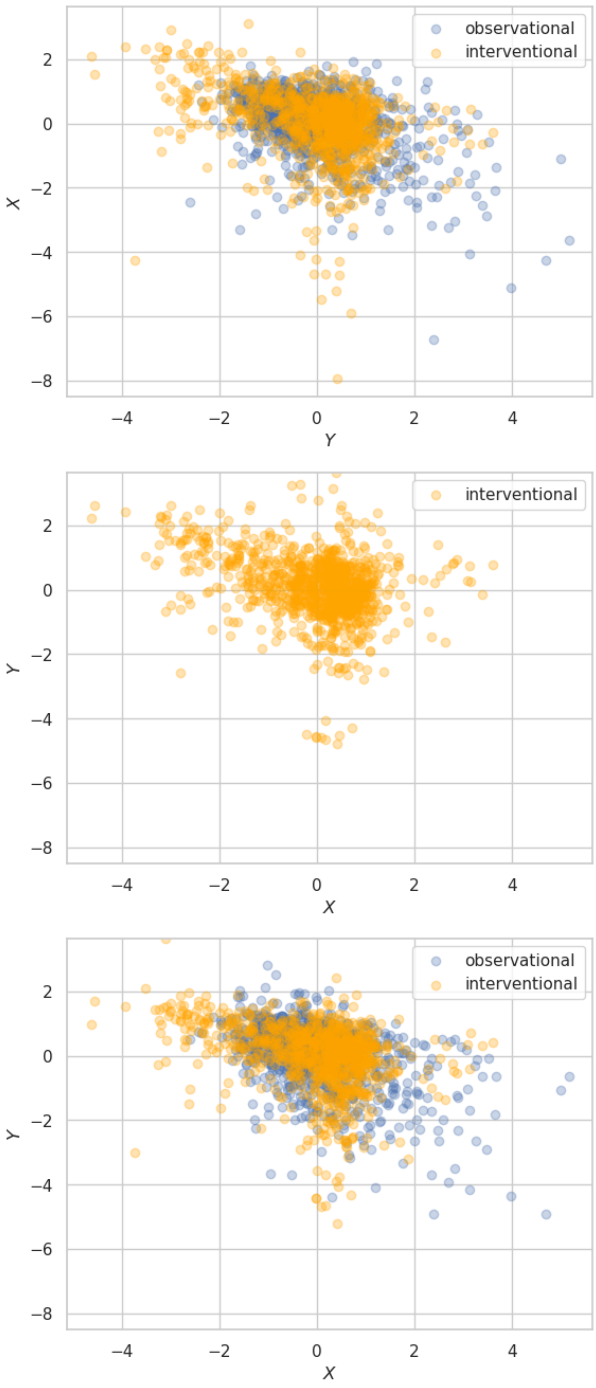

Figure 35: Interventional and observational samples. Top: Observational and interventional training samples. Center: Interventional samples from a model trained with 50 interventional samples. Bottom: Observational and interventional samples from a model trained with 50 interventional samples and 1000 observational samples. The samples are generated as described in Section 4.4.

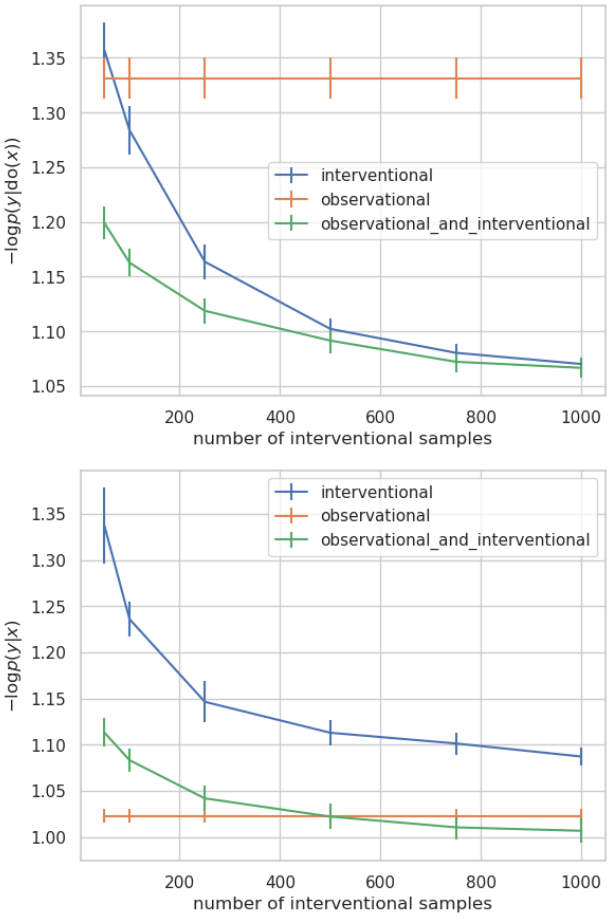

Figure 36: Performances measured in terms of negative log-likelihood on the observational and the interventional test sets, respectively. Top: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 interventional samples from the test set. Bottom: Comparison of a flow model trained with 1000 observational samples, a flow model trained with 50, 100, 250, 500, 750, 1000 interventional samples, and a flow model trained with both 1000 observational samples and 50, 100, 250, 500, 750, 1000 interventional samples. All flow models are evaluated on 1000 observational samples from the test set. We report the mean and standard error for 10 runs of each experiment.