Proceedings of Machine Learning Research vol 140:1–21, 2022 1st Conference on Causal Learning and Reasoning

Diffusion Causal Models for Counterfactual Estimation

Pedro Sanchez PEDRO.SANCHEZ@ED.AC.UK
Sotirios A. Tsaftaris
The University of Edinburgh

Editors: Bernhard Schölkopf, Caroline Uhler and Kun Zhang

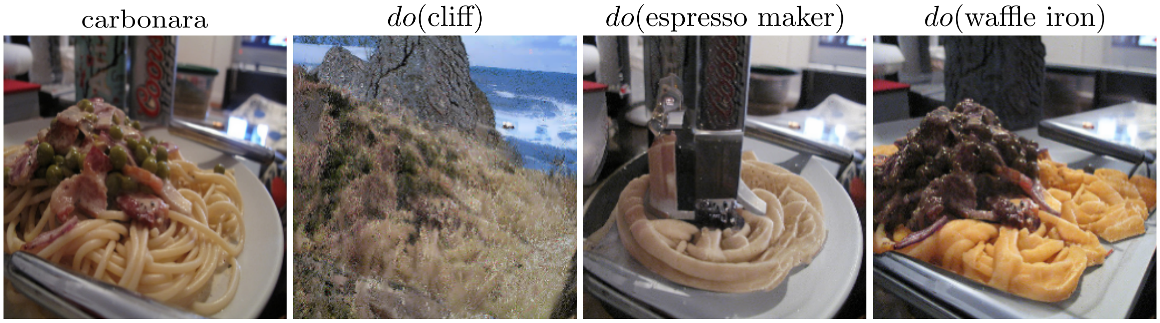


Figure 1: Counterfactuals on ImageNet 256x256 generated by Diff-SCM. *From left to right*: a random image sampled from the data distribution and its counterfactuals $do(\text{class})$, corresponding to "how the image should change in order to be classified as another class?".

Abstract

We consider the task of counterfactual estimation from observational imaging data given a known causal structure. In particular, quantifying the causal effect of interventions for high-dimensional data with neural networks remains an open challenge. Herein we propose Diff-SCM, a deep structural causal model that builds on recent advances of generative energy-based models. In our setting, inference is performed by iteratively sampling gradients of the marginal and conditional distributions entailed by the causal model. Counterfactual estimation is achieved by firstly inferring latent variables with deterministic forward diffusion, then intervening on a reverse diffusion process using the gradients of an anti-causal predictor w.r.t the input. Furthermore, we propose a metric for evaluating the generated counterfactuals. We find that Diff-SCM produces more realistic and minimal counterfactuals than baselines on MNIST data and can also be applied to ImageNet data. Code is available `https://github.com/vios-s/Diff-SCM`.

1. Introduction

The notion of applying interventions in learned systems has been gaining significant attention in causal representation learning (Scholkopf et al., 2021). In causal inference, relationships between variables are directed. An intervention on the cause will change the effect, but not the other way around. This notion goes beyond learning conditional distributions $p(\mathbf{x}^{(k)} \mid \mathbf{x}^{(j)})$ based on the data alone, as in the classical statistical learning framework (Vapnik, 1999). Building causal models implies capturing the underlying physical mechanism that generated the data into a model (Pearl, 2009). As a result, one should be able to quantify the causal effect of a given action. In particular, when an intervention is applied for a given instance, the model should be able the generate hypothetical scenarios. These are the so-called *counterfactuals*.

Building causal models that quantify the effect of a given action for a given causal structure and available data is referred to as *causal estimation*. However, estimating the effect of interventions for high-dimensional data remains an open problem (Pawlowski et al., 2020; Yang et al., 2021). While machine learning is a powerful tool for learning relationships between high-dimensional variables, most causal estimation methods using neural networks (Johansson et al., 2016; Louizos et al., 2017; Shi et al., 2019; Du et al., 2021) are only applied in semi-synthetic low-dimensional datasets (Hill, 2012; Shimoni et al., 2018). Therefore, causal estimation through learning deep neural networks for high-dimensional variables remains a desired quest. We show that we can estimate the effect of interventions by generating counterfactuals on imaging datasets, as illustrated in Fig. 1.

Herein, we leverage recent advances in generative energy based models (EBMs) (Song et al., 2021b; Ho et al., 2020) to devise approaches for causal estimation. This formulation has two key advantages: (i) the stochasticity of the diffusion process relates to uncertainty-aware causal models; and (ii) the iterative sampling can be naturally extended for applying interventions. Additionally, we propose an algorithm for counterfactual inference and a metric for evaluating the results. In particular, we use neural networks that learn to reverse a diffusion process (Ho et al., 2020) via denoising. These models are trained to approximate the gradient of a log-likelihood of a distribution w.r.t. the input. We also employ neural networks that are learned in the anti-causal direction (Schölkopf et al., 2012; Kilbertus et al., 2018) to sample via the causal mechanisms. We use the gradients of these anti-causal predictors for applying interventions in specific variables during sampling. Counterfactual estimation is possible via a deterministic version of diffusion models (Song et al., 2021a) which recovers manipulable latent spaces from observations. Finally, the counterfactuals are generated iteratively using Markov Chain Monte Carlo (MCMC) algorithms.

In summary, we devise a framework for causal effect estimation with high-dimensional variables based on diffusion models entitled Diff-SCM. Diff-SCM behaves as a structured generative model where one can sample from the interventional distribution as well as estimate counterfactuals. Our contributions: (i) We propose a theoretical framework for causal modeling using generative diffusion models and anti-causal predictors (Sec. 3.2). (ii) We investigate how anti-causal predictors can be used for applying interventions in the causal direction (Sec. 3.3). (iii) We propose an algorithm for counterfactual estimation using Diff-SCM (Sec. 3.4). (iv) We propose a metric term counterfactual latent divergence for evaluating the *minimality* of the generated counterfactuals (Sec. 5.2). We use this metric to compared our method with the selected baselines and hyperparameter search (Sec. 5.3)

2. Background

2.1. Generative Energy-Based Models

A family of generative models based on diffusion processes (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b) has recently gained attention even achieving state-of-the-art image generation quality (Dhariwal and Nichol, 2021). In particular, Denoising Diffusion Probabilistic Models (DDPMs) (Ho et al., 2020) consist in learning to denoise images that were corrupted with Gaussian noise at different scales. DDPMs are defined in terms of a forward Markovian diffusion process. This process gradually adds Gaussian noise, with a time-dependent variance $\beta_t \in [0, 1]$, to a data point $\mathbf{x}_0 \sim p_{\text{data}}(\mathbf{x})$. Thus, the latent variable $\mathbf{x}_t$, with $t \in [0, T]$, is learned to correspond to versions of $\mathbf{x}_0$ perturbed by Gaussian noise following $p\left(\mathbf{x}_t \mid \mathbf{x}_0\right) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_0, (1-\alpha_t)\,\mathrm{I}\right)$, where $\alpha_t := \prod_{j=0}^{t}\left(1-\beta_j\right)$ and I is the identity matrix.

As such, $p(\mathbf{x}_t) = \int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t \mid \mathbf{x}) \mathrm{d}\mathbf{x}$ should approximate the data distribution $p(\mathbf{x}_0) \approx p_{\text{data}}$ at time $t = 0$ and a zero centered Gaussian distribution at time $t = T$. Generative modelling is achieved by learning to reverse this process using a neural network $\boldsymbol{\epsilon}_\theta$ trained to denoise images at different scales β_t. The denoising model effectively learns the gradient of a log-likelihood w.r.t. the observed variable $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ (Hyvärinen, 2005).

Training. With sufficient data and model capacity, the following training procedure ensures that the optimal solution to $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ can be found by training $\boldsymbol{\epsilon}_\theta$ to approximate $\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t \mid \mathbf{x}_0)$. The training procedure can be formalised as

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{t, \mathbf{x}_0 \sim p_{\text{data}}, \epsilon \sim \mathcal{N}(0, \mathrm{I})} \left[(1-\alpha_t) \left\| \boldsymbol{\epsilon}_\theta(\sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\epsilon, t) - \epsilon \right\|_2^2 \right]. \tag{1}$$

Inference. Once the model $\boldsymbol{\epsilon}_\theta$ is learned using Eq. 1, generating samples consists in starting from $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$ and iteratively sampling from the reverse Markov chain following:

$$\mathbf{x}(t-1) = \frac{1}{\sqrt{1-\beta_t}} \left[\mathbf{x}_t + \beta_t\, \boldsymbol{\epsilon}_{\theta^*}(\mathbf{x}_t, t) \right] + \sqrt{\beta_t}\mathbf{z}, \quad t = T \cdots 0\,,\ \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathrm{I}). \tag{2}$$

We note that, in the DDPM setting, $\mathbf{z}$ is re-sampled at each iteration. Diffusion models are Markovian and stochastic by nature. As such, they can be defined as a stochastic differential equation (SDE) (Song et al., 2021b). We adopt the time-dependent notation from Song et al. (2021b) as it will be useful for the connection with causal models in Sec. 3.2.

2.2. Causal Models

Counterfactuals can be understood from a formal perspective using the causal inference formalism (Pearl, 2009; Peters et al., 2017; Scholkopf et al., 2021). Structural Causal Models (SCM) $\mathfrak{G} := (\mathbf{S}, p_U)$ consist of a collection $\mathbf{S} = (f^{(1)}, f^{(2)}, ..., f^{(K)})$ of structural assignments (so-called *mechanisms*), defined as

$$\mathbf{x}^{(k)} := f^{(k)}(\mathbf{pa}^{(k)}, \mathbf{u}^{(k)}), \tag{3}$$

where $X = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(K)}\}$ are the known endogenous random variables, $\mathbf{pa}^{(k)}$ is the set of parents of $\mathbf{x}^{(k)}$ (its direct causes) and $U = \{\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, ..., \mathbf{u}^{(K)}\}$ are the exogenous variables. The distribution $p(U)$ of the exogenous variables represents the uncertainty associated with variables that were not taken into account by the causal model. Moreover, variables in U are mutually independent following the joint distribution:

$$p(U) = \prod_{k=1}^{K} p(\mathbf{u}^{(k)}). \tag{4}$$

These structural equations can be defined graphically as a directed acyclic graph. Vertices are the endogenous variables and edges represent (directional) causal relationships between them. In particular, there is a joint distribution $p_{\mathfrak{G}}(X) = \prod_{k=1}^{K} p(\mathbf{x}^{(k)} \mid \mathbf{pa}^{(k)})$ which is Markov related to $\mathcal{G}$. In other words, the SCM $\mathfrak{G}$ represents a joint distributions over the endogenous variables. A graphical example of a SCM is depicted on the left part of Fig. 2. Finally, SCMs should comply to what is known as *Pearl's Causal Hierarchy* (see Appendix B for more details).

3. Causal Modeling with Diffusion Processes

3.1. Problem Statement

In this work, we build a causal model capable of estimating counterfactuals of high-dimensional variables. We will base our work on three assumptions: (i) The SCM is known and the intervention is identifiable. (ii) The variables over which the counterfactuals will be estimated need to contain enough information to recover their causes; *i.e.* an anti-causal predictor can be trained. (iii) All endogenous variables in the training set are annotated.

Notation. We use $\mathbf{x}_t^{(k)}$ is the k^{th} endogenous random variable in a causal graph $\mathfrak{G}$ at diffusion time t. $x_{t,i}^{(k)}$ is a sample $i \in [\text{CF}, \text{F}]$ (F and CF being factual and counterfactual respectively) from $\mathbf{x}_t^{(k)}$. Whenever t is omitted, it should be considered zero, *i.e.* the sample is not corrupted with Gaussian noise. $\mathbf{an}^{(k)}$ for the ancestors, with $\mathbf{pa}^{(k)} \subset \mathbf{an}^{(k)}$, and $\mathbf{de}^{(k)}$ for the descendants of $\mathbf{x}^{(k)}$ in $\mathfrak{G}$.

3.2. Diff-SCM: Unifying Diffusion Processes and Causal Models

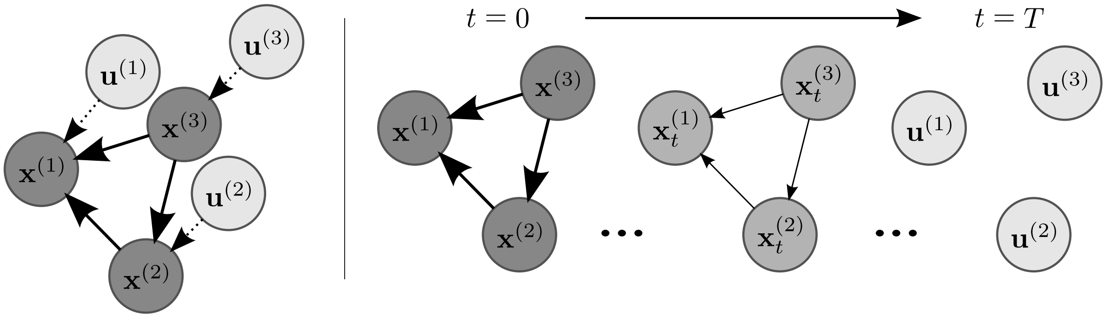


Figure 2: Illustration of a diffusion process as weakening of causal relationships. *Left:* Example of a SCM with endogenous variables $\mathbf{x}^{(k)}$ and respective exogenous variables $\mathbf{u}^{(k)}$. *Right:* The diffusion process weakens the relationship between endogenous variables until they become completely independent at $t = T$. Arrows with solid lines indicate the causal relationship between variables and direction, while the thickness of the arrow indicates strength of the relation. Note that time t is a fiction used as reference for the diffusion process and is *not* a causal variable.

SCMs have been associated with ordinary (Mooij et al., 2013; Rubenstein et al., 2018) and stochastic (Sokol and Hansen, 2014; Bongers and Mooij, 2018) differential equations as well as other types of dynamical systems (Blom et al., 2020). In these cases, differential equations are useful for modeling time-dependent problems such as chemical kinetics or mass-spring systems. From the energy-based models perspective, Song et al. (2021b) unify denoising diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) and denoising score models (Song and Ermon, 2019) into a framework based on SDEs. In Song et al. (2021b), SDEs are used for formalising a diffusion process in a continuous manner where a model is learned to reverse the SDE in order to generate images.

Here, we unify the SDE framework with causal models. Diff-SCM models the dynamics of causal variables as an Ito process $\mathbf{x}_t^{(k)}$, $\forall t \in [0, T]$ (Øksendal, 2003; Särkkä and Solin, 2019) going from an observed endogenous variable $\mathbf{x}_0^{(k)} = \mathbf{x}^{(k)}$ to its respective exogenous noise $\mathbf{x}_T^{(k)} = \mathbf{u}^{(k)}$ and back. In other words, we formulate the *forward diffusion as a gradual weakening of the causal relations between variables of a SCM*, as illustrated in Fig. 2.

The diffusion forces the exogenous noise $\mathbf{u}^{(j)}$ corresponding to a variable $\mathbf{x}^{(j)}$ of interest to be independent of other $\mathbf{u}^{(i)}$, $\forall\ i \neq j$, following the constraints from Eq. 4. The Brownian motion (diffusion) leads to a Gaussian distribution, which can be seen as a prior. Analogously, the original joint distribution entailed by the SCM $p_{\mathfrak{G}}(X)$ diffuses to independent Gaussian distributions equivalent to $p(U)$. As such, the time-dependent joint distribution $p(X_t),\ \forall t \in [0,T]$ have as bounds $p(X_T) = p(U)$ and $p(\mathbf{x}_0) = p_{\mathfrak{G}}(X)$. Note that $p(X_t)$ refers to time-dependent distribution over all causal variables $\mathbf{x}^{(k)}$.

We follow Song et al. (2021b) in defining the diffusion process from Sec. 2.1 in terms of an SDE. Since SDEs are stochastic processes, their solution follows a certain probability distribution instead of a deterministic value. By constraining this distribution to be the same as the distribution $p_{\mathfrak{G}}(X)$ entailed by an SCM $\mathfrak{G}$, we can define a deep structural causal model (DSCM) as a set of SDEs (one for each node k):

$$\begin{aligned} & d\mathbf{x}^{(k)} = -\frac{1}{2}\beta_t \mathbf{x}^{(k)} dt + \sqrt{\beta_t}\, d\mathbf{w},\ \ \forall k \in [1, K], \\ & \text{where}\ \ p(\mathbf{x}_0^{(k)}) = \prod_{j=k}^{K} p(\mathbf{x}^{(j)} \mid \mathbf{pa}^{(j)}) \text{ and } p(\mathbf{x}_T^{(k)}) = p(\mathbf{u}^{(k)}). \end{aligned} \tag{5}$$

Here, $\mathbf{w}$ denotes the Wiener process (or Brownian motion). The first part of the SDE ($-\frac{1}{2}\beta_t \mathbf{x}^{(k)}$) is known as drift function (Särkkä and Solin, 2019)[1].

The generative process is the solution of the reverse-time SDE from Eq. 6 in time. This process is done by iteratively updating the exogenous noise $\mathbf{x}_T^{(k)} = \mathbf{u}^{(k)}$ with the gradient of the data distribution w.r.t. the input variable $\nabla_{\mathbf{x}_t^{(k)}} \log p(\mathbf{x}_t^{(k)})$, until it becomes $\mathbf{x}_0^{(k)} = \mathbf{x}^{(k)}$ with:

$$d\mathbf{x}^{(k)} = \left[-\frac{1}{2}\beta_t + \beta_t\, \nabla_{\mathbf{x}_t^{(k)}} \log p(\mathbf{x}_t^{(k)}) \right] dt + \sqrt{\beta_t}\bar{\mathbf{w}}. \tag{6}$$

The reverse SDE can, therefore, be considered as the process of strengthening causal relations between variables. More importantly, the iterative fashion of the generative process (reverse SDE) is ideal in a causal framework due to the flexibility of applying interventions. We refer the reader to Song et al. (2021b) for a detailed description and proofs of SDE formulation for score-based diffusion models.

3.3. How to Apply Interventions with Anti-Causal Predictors?

An interesting result of Eq. 6 is that one only needs the gradients of the distribution entailed by the SCM $p_{\mathfrak{G}}$ for sampling. This allows learning of the anti-causal conditional distributions $p_{\mathfrak{G}^-}$ and applying interventions with the causal mechanism. This can be useful when anti-causal learning is more straightforward (Schölkopf et al., 2012). In these cases, one would train classifiers in the anti-causal direction for each edge and diffusion models for each node (over which one wants to

[1] The drift function can potentially be used to define temporal relations between variables as in Rubenstein et al. (2018) and Blom et al. (2020).

measure the effect of interventions) in the graph. Then, one might use the gradients of the classifiers and diffusion models to propagate the intervention in the causal direction over the nodes. Following this idea, proposition 1 arises as a result of Eq. 6.

Proposition 1 (Interventions as anti-causal gradient updates) *We consider the SCM $\mathfrak{G}$ and a variable $\mathbf{x}^{(j)} \in \mathbf{an}^{(k)}$. The effect observed on $\mathbf{x}^{(k)}$ caused by an intervention on $\mathbf{x}^{(j)}$, $p_{\mathfrak{G}}(\mathbf{x}^{(k)} \mid do(\mathbf{x}^{(j)} = x^{(j)}))$, is equivalent to solving a reverse-diffusion process for $\mathbf{x}_t^{(k)}$. Since the sampling process involves taking into account the distribution entailed by $\mathfrak{G}$, it is guided by the gradient of an* ***anti-causal*** *predictor w.r.t. the effect when the cause is assigned a specific value:*

$$\nabla_{x_t^{(k)}} p_{\mathfrak{G}^-}(\mathbf{x}^{(j)} = x^{(j)} \mid x_t^{(k)}). \tag{7}$$

Proposition 1 respects the principle of independent causal mechanisms (ICM)[2] (Peters et al., 2017; Schölkopf et al., 2012). It implies independence between the cause distribution and the mechanism producing the effect distribution. As shown in Eq. 7, sampling with the causal mechanism does not require the distribution of the cause $p(\mathbf{x}^{(j)})$ (Scholkopf et al., 2021).

3.4. Counterfactual Estimation with Diff-SCM

A powerful consequence of building causal models, following *Pearl's Causal Hierarchy*, is the estimation of counterfactuals. Counterfactuals are hypothetical scenarios for a given factual observation under a local intervention. Estimation of counterfactuals differentiates of sampling from an interventional distribution because the changes are applied for a given observation. As detailed in Pearl (2016), sec. 4.2.4, counterfactual estimation requires three steps: (i) abduction of exogenous noise – forward diffusion with DDIM algorithm (Song et al., 2021a) following Alg. 3 in Appendix D; (ii) action – graph mutilation by erasing the edges between the intervened variable and its parents; (iii) prediction – reverse diffusion controlled by the gradients of an anti-causal classifier.

Here, we are interested in estimating $x_{\text{CF}}^{(k)}$ based on the observed (factual) $x_{\text{F}}^{(k)}$ for the random variable $\mathbf{x}^{(k)}$ after assigning a value $x_{\text{CF}}^{(j)}$ to $\mathbf{x}^{(j)} \in \mathbf{an}^{(k)}$, *i.e.* applying an intervention $do(\mathbf{x}^{(j)} = x_{\text{CF}}^{(j)})$. It's equivalent to sample from counterfactual distribution $p_{\mathfrak{G}}(\mathbf{x}^{(k)} \mid do(\mathbf{x}^{(j)} = x_{\text{CF}}^{(j)}); \mathbf{x}^{(k)} = x_{\text{F}}^{(k)})$. We will consider a setting where only $\mathbf{x}^{(j)}$ and $\mathbf{x}^{(k)}$ are present in the graph as a simplifying assumption for Alg. 1. Considering only two variables removes the need for the graph mutilation explained above. It is also the setting used in our experiments. We will leave an extension to more complex SCMs for future work. We detail in Alg. 1 how abduction of exogenous noise and prediction is done.

Abduction of Exogenous Noise. The first step for estimating a counterfactual is the abduction of exogenous noise. Note from Eq. 3 that the value of a causal variable depends both on its parents and on its respective exogenous noise. From a deep learning perspective (Pawlowski et al., 2020), one might consider the exogenous $\mathbf{u}^{(k)}$ an inferred latent variable. The prior $p(\mathbf{u}^{(k)})$ of $\mathbf{u}^{(k)}$ in Diff-SCM is a Gaussian as detailed in Sec. 3.2.

With diffusion models, abduction can be done with a derivation done by Song et al. (2021a) and Song et al. (2021b). Both works make a connection between diffusion models and neural ODEs (Chen et al., 2018). They show that one can obtain a deterministic inference system while training

[2] The principle states that "The causal generative process of a system's variables is composed of autonomous modules that do not inform or influence each other."

with a diffusion process, which is stochastic by nature. This formulation allows the process to be invertible by recovering a latent space $u^{(k)}$ by performing the forward diffusion with the learned model. The algorithm for recovering $u^{(k)}$ is highlighted as the first box in Alg. 1.

Prediction under Intervention. Once the abduction of exogenous noise $u^{(k)}$ is done for a given factual observation $x_{\text{F}}^{(k)}$, counterfactual estimation consists in applying an intervention in the reverse diffusion process with the gradients of an anti-causal predictor. In particular, we use the formulation of guided DDIM from Dhariwal and Nichol (2021) which forms the second part of Alg. 1.

Controlling the Intervention. There are three main factors contributing for the counterfactual estimation in Alg. 1: (i) The inferred $u^{(k)}$ keeps information about the factual observation; (ii) $\nabla_{x_t^{(k)}} \log p_\phi(x_{\text{CF}}^{(j)} \mid x_t^{(k)})$ guide the intervention towards the desired counterfactual class; and (iii) $\boldsymbol{\epsilon}_\theta(x_t^{(k)}, t)$ forces the estimation to belong to the data distribution. We follow Dhariwal and Nichol (2021) in adding an hyperparameter s which controls the scale of $\nabla_{x_t^{(k)}} \log p_\phi(x_{\text{CF}}^{(j)} \mid x_t^{(k)})$. High values of s might result in counterfactuals that are too different from the factual data. We show this empirically and discuss the effects of this hyperparameter in Sec. 5.3.

Algorithm 1 Inference of **counterfactual** for a variable $\mathbf{x}^{(k)}$ from an intervention on $\mathbf{x}^{(j)} \in \mathbf{an}^{(k)}$

Models: trained diffusion model $\boldsymbol{\epsilon}_\theta$ and anti-causal predictor $p_\phi(x^{(j)} \mid x_t^{(k)})$

Input : factual variable $x_{0,\text{F}}^{(k)}$, target intervention $x_{0,\text{CF}}^{(j)}$, scale s

Output: counterfactual $x_{0,\text{CF}}^{(k)}$

```
Abduction of Exogenous Noise – Recovering $u^{(k)}$ from $x_{0,\mathbf{F}}^{(k)}$
for $t \leftarrow 0$ to $T$ do
    $x_{t+1,\text{F}}^{(k)} \leftarrow \sqrt{\alpha_{t+1}} \left( \frac{x_{t,\text{F}}^{(k)} - \sqrt{1-\alpha_t}\, \boldsymbol{\epsilon}_\theta(x_{t,\text{F}}^{(k)}, t)}{\sqrt{\alpha_t}} \right) + \sqrt{\alpha_{t+1}}\, \boldsymbol{\epsilon}_\theta(x_{t,\text{F}}^{(k)}, t)$
end
$u^{(k)} = x_{T,\text{F}}^{(k)} = x_T^{(k)}$

Generation under Intervention
for $t \leftarrow T$ to $0$ do
    $\epsilon \leftarrow \boldsymbol{\epsilon}_\theta(x_t^{(k)}, t) - s\sqrt{1-\alpha_t}\, \nabla_{x_t^{(k)}} \log p_\phi(x_{0,\text{CF}}^{(j)} \mid x_t^{(k)})$
    $x_{t-1}^{(k)} \leftarrow \sqrt{\alpha_{t-1}} \left( \frac{x_t^{(k)} - \sqrt{1-\alpha_t}\, \epsilon}{\sqrt{\alpha_t}} \right) + \sqrt{\alpha_{t-1}}\, \epsilon$
end
$x_{0,\text{CF}}^{(k)} = x_0^{(k)}$
```

4. Related Work

Generative EBMs. Our generative framework is inspired on the energy based models literature (Ho et al., 2020; Song et al., 2021b; Du and Mordatch, 2019; Grathwohl et al., 2020). In particular, we leverage the theory around denoising diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Nichol and Dhariwal, 2021). We take advantage of a non-Markovian definition DDIM (Song et al., 2021a) which allows faster sampling and recovering latent spaces from observations. Our theory connecting diffusion models and SDEs follows Song et al. (2021b), but from a different perspective. Even though Du et al. (2020) are not constrained to causal modeling, they also use the idea of guiding the generation with gradient of conditional energy models. Recently, Sinha et al. (2021) proposed a version of diffusion models for manipulable generation based on contrastive learning. Finally, Dhariwal and Nichol (2021) derive a conditional sampling process for DDIM that is used in this paper as detailed in Sec. 3.3. Here, we re-interpret their generation algorithm from a causal perspective and add deterministic latent inference for counterfactual estimation. The main, but key difference, is that we add the *abduction of exogenous noise*. Without this abduction, we cannot ensure that the resulting image will match other aspects of the original image whilst altering only the intended aspect (ie. Where we want to intervene). We can sample from a counterfactual distribution instead of the interventional distribution.

Counterfactuals. Designing causal models with deep learning components has allowed causal inference with high-dimensional variables (Pawlowski et al., 2020; Shen et al., 2020; Dash et al., 2020; Xia et al., 2021; Zečevi et al., 2021). Given a factual observation, counterfactuals are obtained by measuring the effect of an intervention in one of the ancestral attributes. They have been used in a range of applications such as (i) explaining predictions (Verma et al., 2020; Goyal et al., 2019; Looveren and Klaise, 2021; Hvilshøj et al., 2021); (ii) defining fairness (Kusner et al., 2017); (iii) mitigating data biases (Denton et al., 2019); (iv) improving reinforcement learning (Lu et al., 2020); (v) predicting accuracy (Kaushik et al., 2020); (vi) increasing robustness against spurious correlations (Sauer and Geiger, 2021). Most similar to our work, Schut et al. (2021) estimate counterfactuals via iterative updates using the gradients of a classifier. However, their method is based on adversarial updates computed via epistemic uncertainty, not diffusion processes.

5. Experiments

Ground truth counterfactuals are, by definition, impossible to acquire. Counterfactuals are hypothetical predictions. In an ideal scenario, the SCM of problem is fully specified. In this case, one would be able to verify if unrelated causal variables kept their values[3]. However, a complete causal graph is rarely known in practice. In this section, we (i) present ideas on how to evaluate counterfactuals without access to the complete causal graph nor semi-synthetic data; (ii) show with quantitative and qualitative experiments that our method is appropriate for counterfactual estimation; (iii) propose CLD, a metric for quantitative evaluation of counterfactuals; and (iv) use CLD for fine tuning an important hyperparameter of our framework.

Causal Setup. We consider a causal model $\mathfrak{G}_{image}$ with two variables $\mathbf{x}^{(1)} \leftarrow \mathbf{x}^{(2)}$ following the example in Sec. 3.3. Here, $\mathbf{x}^{(1)}$ represents an image and $\mathbf{x}^{(2)}$ a class. In practice, the gradient of the marginal distribution of $\mathbf{x}^{(1)}$ is learned with a diffusion model, which we refer as $\boldsymbol{\epsilon}_\theta$, as in Sec. 2.1. The anti-causal conditional distribution is also learned with a neural network $p_\phi(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)})$. Our experiments aim at sampling from the counterfactual distribution $p_{\mathfrak{G}}(\mathbf{x}^{(1)} \mid do(\mathbf{x}^{(2)} = x^{(2)}_{\text{CF}}); x^{(1)}_{\text{F}})$. Extra experiments on sampling from interventional distribution are in Appendix F.

Implementation. $\boldsymbol{\epsilon}_\theta$ is implemented as an encoder-decoder architecture with skip-connections, *i.e.* a Unet-like network (Ronneberger et al., 2015). For anti-causal classification tasks, we use the

[3] Remember that interventions only change descendants in a causal graph.

encoder of $\boldsymbol{\epsilon}_\theta$ with a pooling layer followed by a linear classifier. Both $\boldsymbol{\epsilon}_\theta$ and $p_\phi(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)})$ dependent on diffusion time. The diffusion model and anti-causal predictor are trained separately. Implementation details are in Appendix E.

Baselines. We consider Schut et al. (2021) and Looveren and Klaise (2021) because they (i) generate counterfactuals based on classifiers decisions; and (ii) evaluate results with metrics tailored to counterfactual estimation on images.

Datasets. Considering the causal model $\mathfrak{G}_{image}$ described above, we compare our method quantitatively and qualitatively with baselines on MNIST data (Lecun et al., 1998). Furthermore, we show empirically that our approach works with more complex, higher-resolution images from the ImageNet dataset (Deng et al., 2009). We only perform qualitative evaluations on ImageNet since the baseline methods cannot generate counterfactuals for this dataset.

5.1. Evaluating Counterfactuals: Realism and Closeness to Data Manifold

Taking into account the causal model $\mathfrak{G}_{image}$, we now employ the strategies for counterfactual estimation in Sec. 3.4. In particular, given an image $x_{\text{F}}^{(1)} \sim \mathbf{x}^{(1)}$ and a target intervention $x_{\text{CF}}^{(2)}$ in the class variable, we wish to estimate the counterfactual $x_{\text{CF}}^{(1)}$ for the image $x_{\text{F}}^{(1)}$. We use two metrics proposed by Looveren and Klaise (2021), IM1 and IM2, to measure the realism, interpretability and closeness to the data manifold based on the reconstruction loss of autoencoders trained on specific classes. See details in Appendix G.

Experimental Setup. We run Alg. 1 over the test set with randomly sampled target counterfactual classes $x_{\text{CF}}^{(2)} \sim \mathbf{x}^{(2)}$, $\forall x^{(2)} \neq x_{\text{F}}^{(2)}$. For example. we generate counterfactuals of all MNIST classes for a given factual image, as illustrated in Appendix H. We evaluate realism of Diff-SCM, Schut et al. and Looveren and Klaise using the IM1 and IM2 metrics. Diff-SCM achieves better results (lower is better) in both metrics[4], as shown in Tab. 1. We show qualitative results on ImageNet in Fig. 1 and on MNIST in Appendix H. A qualitative comparison between methods is depicted in Fig. 3(*b*).

Table 1: Quantitative comparison between Diff-SCM and baselines. Lower is better for all metrics. Results are presented with mean (μ) and standard deviation σ over the test set in the format μ_σ.

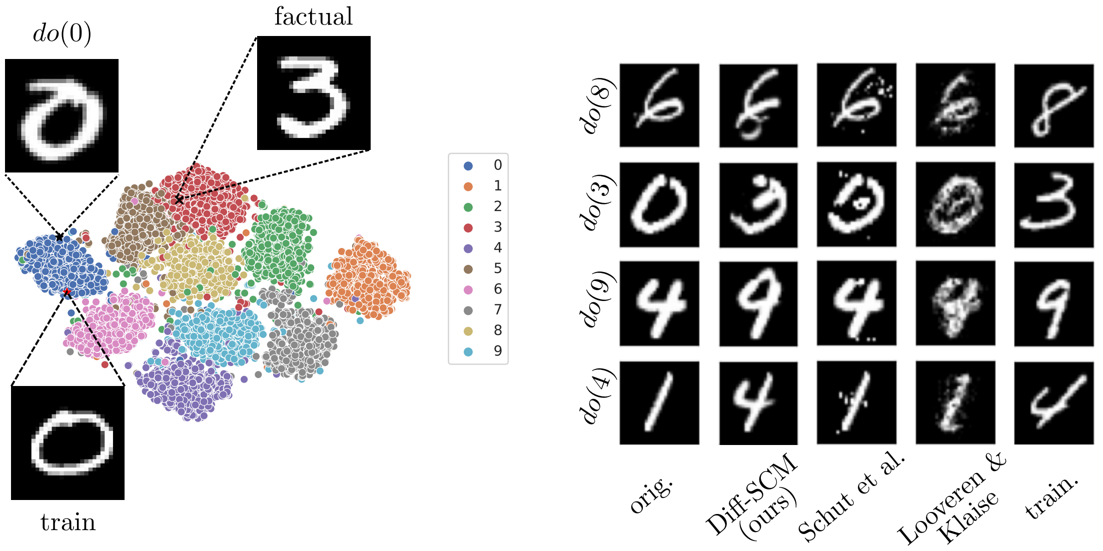
Method	IM1 ↓	IM2 ↓	CLD ↓
Diff-SCM (ours)	$\mathbf{0.94_{0.02}}$	$\mathbf{0.04_{0.00}}$	$\mathbf{1.08_{0.03}}$
Looveren and Klaise	$1.10_{0.03}$	$0.05_{0.00}$	$1.25_{0.03}$
Schut et al.	$1.05_{0.01}$	$0.10_{0.00}$	$1.19_{0.01}$

5.2. Counterfactual Latent Divergence (CLD)

Since one cannot measure changes in all variables of a real SCM, we leverage the sparse mechanism shift (SMS) hypothesis[5] (Scholkopf et al., 2021) for justifying a *minimality* property of counterfac-

[4] We highlight that our setting is slightly different from baseline works where the target counterfactual classes were similar to the factual classes. *e.g.* Transforming MNIST digits from $2 \to [3, 7]$ or $4 \to [1, 9]$. Since we are sampling target classes randomly, their metric values will look lower than in their respective papers.

[5] SMS states that a "small distribution changes tend to manifest themselves in a sparse or local way in the causal factorization, that is, they should usually not affect all factors simultaneously."



(a) CLD Intuition

(b) Qualitative comparison

Figure 3: (a) A t-SNE visualization of the 20-dimensional latent vector of a variational autoencoder VAE over all MNIST samples. Each point represents an MNIST image and colors represent the ground-truth label of each sample. CLD's goal is to estimate a relative similarity between the factual data and the counterfactual. The distance between the generated counterfactual $do(0)$ and factual observation is compared to the distances between the factual observation and all other data points from factual and counterfactual classes. (b) Qualitative comparison with baselines approaches for counterfactual estimation. Each column represents one method and each row a different intervention on digit class. The *train.* column shows training samples belonging to the target intervention class.

tuals. SMS translates, in our setting, to *an intervention will not change many elements of the observed data*. Therefore, an important property of counterfactuals is minimality or proximity to the factual observation. We suggest here a new metric entitled counterfactual latent divergence (CLD), illustrated in Fig. 3(a), that estimates minimality.

Note that the metrics IM1 and IM2 from Sec. 5.1 do not take minimality into account. In addition, previous work (Wachter et al., 2018; Schut et al., 2021) only used the mean absolute error or ℓ_1 distance in the data space for measuring minimality. However, measuring similarity at pixel-level can be challenging as an intervention might change the structure of the image whilst keeping other factors unchanged. In this case, a pixel-level comparison might not be informative about the other factors of variation.

Latent Similarity. Therefore, we choose to measure similarity between latent representation. In addition, we want a representation that captures all factors of variation on the input data. In particular, we train a variational autoencoder (VAE) (Kingma and Welling, 2014) for recovering probabilistic latent representations that capture all factors of variation in the data. The latent spaces computed with the VAE's encoder E_ϕ are denoted as $\mu_i, \sigma_i = E_\phi(x_i^{(1)})$, where subscript i means different samples from $\mathbf{x}^{(1)}$ $(t = 0)$. We use the Kullback–Leibler divergence (KL) divergence for measuring the distances between latents. The divergence for a given counterfactual estimation and factual observation pair $(x^{(1)}_{\text{CF}}, x^{(1)}_{\text{F}})$ can, therefore, be denoted as

$$div = D(x^{(1)}_{\text{CF}}, x^{(1)}_{\text{F}}), \qquad \text{with } D(x^{(1)}_i, x^{(1)}_j) = D_{\text{KL}}(\mathcal{N}(\mu_i, \sigma_i), \mathcal{N}(\mu_j, \sigma_j)). \tag{8}$$

Relative Measure. However, absolute similarity measures give limited information. Therefore, we leverage class information for measuring minimality whilst making sure that the counterfactual is far enough from the factual class. A relative measure is obtained by estimating the probability of sets of divergence measures between the factual observation and other data points in the dataset (formalized in the Eq. 9) to less or greater than div. In particular, we compare div with the set $\mathcal{S}_{\text{class}}$ of divergence measures between the factual observation $x^{(1)}_{\text{F}}$ and all data points $x^{(1)}$ in a dataset $\mathcal{D} = \{(x^{(1)}, x^{(2)}) \mid x^{(1)} \in \mathbb{R}^2, x^{(2)} \in \mathbb{N}\}$ for which the class $x^{(2)}$ is $x^{(2)}_{\text{class}}$ is denoted in set-builder notation[6] with:

$$\mathcal{S}_{\text{class}} = \{D(x^{(1)}, x^{(1)}_{\text{F}}) \mid (x^{(1)}, x^{(2)}) \in \mathcal{D} \land x^{(2)} = x^{(2)}_{\text{class}}\}. \tag{9}$$

The sets $\mathcal{S}_{\text{CF}}$ and $\mathcal{S}_{\text{F}}$ are obtained by replacing "class" in $\mathcal{S}_{\text{class}}$ with the appropriate target class of the counterfactual and factual observation class respectively.

The relative measures are: (i) $P(\mathcal{S}_{\text{CF}} \leq div)$ for comparing div with the distance between all data points of the counterfactual class and the factual image; and (ii) $P(\mathcal{S}_{\text{F}} \geq div)$ for comparing div with the distance between all other data points of the factual class and the factual image. We aim for counterfactuals with low $P(\mathcal{S}_{\text{CF}} \leq div)$, enforcing minimality, and low $P(\mathcal{S}_{\text{F}} \geq div)$, enforcing bigger distances from the factual class.

CLD. We highlight the competing nature of the two measures $P(\mathcal{S}_{\text{CF}} \leq div)$ and $P(\mathcal{S}_{\text{F}} \geq div)$ in the counterfactual setting. For example, if the intervention is too minimal – *i.e.* low $P(\mathcal{S}_{\text{CF}} \leq div)$ – the counterfactual will still resemble observations from the factual class – *i.e.* high $P(\mathcal{S}_{\text{F}} \geq div)$. Therefore, the goal is to find the best balance between the two measures. Finally, we define the counterfactual latent divergence (CLD) metric as the LogSumExp of the two probability measures. The LogSumExp operation acts as a smooth approximation of the maximum function. It also penalizes relative peak values for any of the measures when compared to a simple summation. We denote CLD as:

$$\text{CLD} = \log\left(\exp\left(P\left(\mathcal{S}_{\text{CF}} \leq div\right)\right) + \exp\left(P\left(\mathcal{S}_{\text{F}} \geq div\right)\right)\right). \tag{10}$$

We show, using the same experimental setup as in Sec. 5.1, that CLD improves counterfactual estimation when quantitatively compared with the baseline methods, as illustrated in Tab. 1.

5.3. Tuning the Hyperparameter s with CLD

We now utilize CLD, the proposed metric, for fine-tuning s, the scale hyperparameter of our framework detailed in Sec. 3.4. Incidentally, the model with hyperparameters achieving best CLD outperforms previous methods in other metrics (see Tab. 1) and output the best qualitative results (see Fig. 3(b)). This result further validate that our metric is suited for counterfactual evaluation.

[6] We use the following set-builder notation: MY_SET $= \{$function(input) $\mid$ input_domain$\}$.

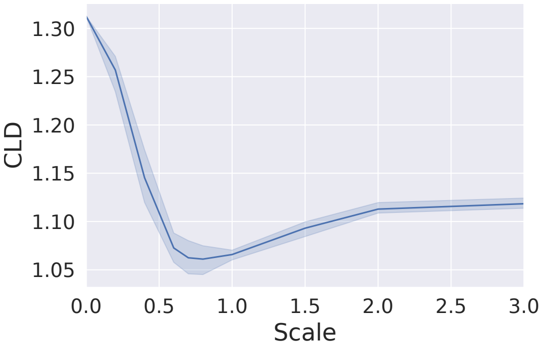


Figure 4: Scale hyperparameter search using CLD (lower is better). The line plot shows the mean and 95% confidence interval. We found that $s = 0.7$ is the best value.

Experimental Setup. We run Alg. 1 while varying the scale hyperparameter s in the $[0.0, 3.0]$ interval for MNIST data, as depicted in Fig. 4. When $s = 0$, the classifier does not influence the generation, therefore, the counterfactuals are reconstructions of the factual data; resulting in a high CLD.

When $s = 3$ (too high), the diffusion model contributes much less than the classifier, therefore, the counterfactuals are driven towards the desired class while ignoring the exogenous noise of a given observation. High values of s correspond to strong interventions which do not hold the minimality property, also resulting in a high CLD. Therefore, the optimum point for s is an intermediate value where CLD is minimum. All MNIST experiments were performed using $s = 0.7$, following this hyperparameter search. See Appendix I for qualitative results.

6. Conclusions

We propose a theoretical framework for causal estimation using generative diffusion models entitled Diff-SCM. Diff-SCM unifies recent advances in generative energy-based models and structural causal models. Our key idea is to use gradients of the marginal and conditional distributions entailed by an SCM for causal estimation. The main benefit of only using the distribution's gradients is that one can learn an anti-causal mechanism and use its gradients as a causal mechanism for generation. We show empirically how it can be applied to a two variable causal model. We leave the extension to more complex causal models to future work.

Furthermore, we present an algorithm for performing interventions and estimating counterfactuals with Diff-SCM. We acknowledge the difficulty of evaluating counterfactuals and propose a metric entitled counterfactual latent divergence (CLD). CLD measures the distance, in a latent space, between the observation and the generated counterfactual by comparison with other distances between samples in the dataset. We use CLD for comparison with baseline methods and for hyperparameter search. Finally, we show that the proposed Diff-SCM achieves better quantitative and qualitative results compared to state-of-the-art methods for counterfactual generation on MNIST.

Limitations and future work. We only have specifications for two variables in our empirical setting, therefore, applying an intervention on $\mathbf{x}^{(2)}$ means changing all the correlated variables within this dataset. Applying Diff-SCM to more complex causal models would require the use of additional techniques. For instance, consider the SCM depicted in Fig. 2, a classifier naively trained to predict $x^{(2)}$ (class) from $x^{(1)}$ (image) would be biased towards the confounder $x^{(3)}$. Therefore, the gradient of the classifier w.r.t the image would also be biased. This would make the intervention do($x^{(2)}$) not correct. In this case, the graph mutilation (removing edges from parents of node intervened on) would not happen because the gradients from the classifier would pass information about $x^{(3)}$. We leave this extension for future work.

7. Acknowledgement

We thank Spyridon Thermos, Xiao Liu, Jeremy Voisey, Grzegorz Jacenkow and Alison O'Neil for their input on the manuscript and research support. This work was supported by the University of Edinburgh, the Royal Academy of Engineering and Canon Medical Research Europe via Pedro Sanchez's PhD studentship. This work was partially supported by the Alan Turing Institute under the EPSRC grant EP N510129\1. We thank Nvidia for donating a TitanX GPU. S.A. Tsaftaris acknowledges the support of Canon Medical and the Royal Academy of Engineering and the Research Chairs and Senior Research Fellowships scheme (grant RCSRF1819\825).

References

E Bareinboim, J Correa, D Ibeling, and T Icard. On Pearl's Hierarchy and the Foundations of Causal Inference, 2020.

Tineke Blom, Stephan Bongers, and Joris M Mooij. Beyond Structural Causal Models: Causal Constraints Models. In *Proc. 35th Uncertainty in Artificial Intelligence Conference*, pages 585–594, 2020.

Stephan Bongers and Joris M Mooij. From Random Differential Equations to Structural Causal Models: the stochastic case. *arxiv pre-print*, 2018.

Ricky T Q Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems*, 2018.

Saloni Dash, Vineeth N Balasubramanian, and Amit Sharma. Evaluating and Mitigating Bias in Image Classifiers: A Causal Perspective Using Counterfactuals. *arxiv pre-print*, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. of Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009.

Emily Denton, Ben Hutchinson, Margaret Mitchell, Timnit Gebru, and Andrew Zaldivar. Image Counterfactual Sensitivity Analysis for Detecting Unintended Bias. *arxiv pre-print*, 12 2019.

Prafulla Dhariwal and Alex Nichol. Diffusion Models Beat GANs on Image Synthesis. In *Advances in Neural Information Processing Systems*, 2021.

Xin Du, Lei Sun, Wouter Duivesteijn, Alexander Nikolaev, and Mykola Pechenizkiy. Adversarial balancing-based representation learning for causal effect inference with observational data. *Data Mining and Knowledge Discovery*, 35(4):1713–1738, 12 2021.

Yilun Du and Igor Mordatch. Implicit Generation and Generalization in Energy-Based Models. In *Advances in Neural Information Processing Systems*, 12 2019.

Yilun Du, Shuang Li, Igor Mordatch, and Google Brain. Compositional Visual Generation with Energy Based Models. In *Advances in Neural Information Processing Systems*, 2020.

Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual Visual Explanations. *Proc. of 36th International Conference on Machine Learning*, pages 4254–4262, 12 2019.

Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Kevin Swersky, and Mohammad Norouzi. Your Classifier is Secretly an Energy Based Model and You Should Treat it Like One. In *Proc. of International Conference on Learning Representations*, 2020.

Jennifer Hill. Bayesian Nonparametric Modeling for Causal Inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 12 2012.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances on Neural Information Processing Systems*, 2020.

Frederik Hvilshøj, Alexandros Iosifidis, and Ira Assent. ECINN: Efficient Counterfactuals from Invertible Neural Networks. 12 2021.

Aapo Hyvärinen. Estimation of Non-Normalized Statistical Models by Score Matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

Fredrik D Johansson, Uri Shalit, and David Sontag. Learning Representations for Counterfactual Inference. In *Proc. of International Conference on Machine Learning*, 2016.

Divyansh Kaushik, Amrith Setlur, Eduard Hovy, and Zachary C Lipton. EXPLAINING THE EFFICACY OF COUNTERFACTUALLY AUGMENTED DATA, 12 2020.

N Kilbertus, G Parascandolo, and B Scholkopf. Generalization in anti-causal learning. In *NeurIPS Workshop on Critiquing and Correcting Trends in Machine Learning*, 2018.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations*, 2014.

Matt Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual Fairness. In *Advances on Neural Information Processing Systems*, 2017.

Y Lecun, L Bottou, Y Bengio, and P Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Arnaud Van Looveren and Janis Klaise. Interpretable Counterfactual Explanations Guided by Prototypes. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, volume 1907.02584, 12 2021.

Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. Causal Effect Inference with Deep Latent-Variable Models. In *Advances on Neural Information Processing Systems*, 2017.

Chaochao Lu, Biwei Huang, Ke Wang, José Miguel Hernández-Lobato, Kun Zhang, and Bernhard Schölkopf. Sample-Efficient Reinforcement Learning via Counterfactual-Based Data Augmentation. *arxiv pre-print*, 12 2020.

Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. From Ordinary Differential Equations to Structural Causal Models: The Deterministic Case. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 440–448, 2013.

Alex Nichol and Prafulla Dhariwal. Improved Denoising Diffusion Probabilistic Models. *arxiv pre-print*, 12 2021.

Bernt Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, fifth edition edition, 2003. ISBN 978-3-642-14394-6.

Nick Pawlowski, Daniel C Castro, and Ben Glocker. Deep Structural Causal Models for Tractable Counterfactual Inference. In *Advances in Neural Information Processing Systems*, 2020.

Judea Pearl. *Causality*. Cambridge University Press, 2009. doi: 10.1017/CBO9780511803161.

Judea Pearl. *Causal inference in statistics : a primer*. John Wiley & Sons Ltd, Chichester, West Sussex, UK, 2016. ISBN 978-1-119-18684-7.

Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic books, 2018.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference*. MIT Press, 2017.

O Ronneberger, P.Fischer, and T Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proc. of Medical Image Computing and Computer-Assisted Intervention*, volume 9351, pages 234–241. Springer, 2015.

Paul K Rubenstein, Stephan Bongers, uvanl Bernhard Schölkopf, and Joris M Mooij. From Deterministic ODEs to Dynamic Structural Causal Models. In *Proceedings of the 34th Annual Conference on Uncertainty in Artificial Intelligence (UAI-18).*, 2018.

Simo Särkkä and Arno Solin. *Applied Stochastic Differential Equations*, volume 10. Cambridge University Press, 2019.

Axel Sauer and Andreas Geiger. Counterfactual Generative Networks. In *Proc. of International Conference on Learning Representations*, 12 2021.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij JMOOIJ. On Causal and Anticausal Learning. In *Proc. of the International Conference on Machine Learning*, 2012.

Bernhard Scholkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward Causal Representation Learning. *Proceedings of the IEEE*, 2021.

Lisa Schut, Oscar Key, Rory McGrath, Luca Costabello, Bogdan Sacaleanu, Medb Corcoran, and Yarin Gal. Generating Interpretable Counterfactual Explanations By Implicit Minimisation of Epistemic and Aleatoric Uncertainties. In *Proc. of The 24th International Conference on Artificial Intelligence and Statistics*, pages 1756–1764, 2021.

Xinwei Shen, Furui Liu, Hanze Dong, Qing Lian, Zhitang Chen, and Tong Zhang. Disentangled Generative Causal Representation Learning. *arxiv pre-print*, 2020.

Claudia Shi, David M Blei, and Victor Veitch. Adapting Neural Networks for the Estimation of Treatment Effects. In *Proc. of Neural Information Processing Systems*, 2019.

Yishai Shimoni, Chen Yanover, Ehud Karavani, and Yaara Goldschmnidt. Benchmarking Framework for Performance-Evaluation of Causal Inference Analysis. *arxiv pre-print*, 12 2018.

Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2C: Diffusion-Decoding Models for Few-Shot Conditional Generation. *arXiv pre-print*, 2021.

Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. *Proc. of 32nd International Conference on Machine Learning*, 3:2246–2255, 12 2015.

Alexander Sokol and Niels Richard Hansen. Causal interpretation of stochastic differential equations. *Electronic Journal of Probability*, 19:1–24, 2014.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models. In *Proc. of International Conference on Learning Representations*, 2021a.

Yang Song and Stefano Ermon. Generative Modeling by Estimating Gradients of the Data Distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Yang Song Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling Through Stochastic Differential Equations. In *ICLR*, 2021b.

Vladimir N Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Sahil Verma, John Dickerson, and Keegan Hines. Counterfactual Explanations for Machine Learning: A Review. *arxiv pre-print*, 12 2020.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR. *Harvard Journal of Law & Technology*, 31 (2), 2018.

Kevin Xia, Kai-Zhan Lee, Yoshua Bengio, and Elias Bareinboim. The Causal-Neural Connection: Expressiveness, Learnability, and Inference. 12 2021.

Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. CausalVAE: Disentangled Representation Learning via Neural Structural Causal Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9593–9602, 2021.

Matej Zečevi, Devendra Singh Dhami, Petar Veličkovi, and Kristian Kersting. Relating Graph Neural Networks to Structural Causal Models. *arxiv pre-print*, 2021.

Appendix A. Theory for Training Diffusion Models

We now review with more detailed the formulation of Denoising Diffusion Probabilistic Models (DDPMs) (Ho et al., 2020). In DDPM, samples are generated by reversing a diffusion process with a neural network from a Gaussian prior distribution. We begin by defining our data distribution $x_0 \sim p(\mathbf{x}_0)$ and a Markovian noising process which gradually adds noise to the data to produce noised samples $\mathbf{x}_t$ up to $\mathbf{x}_T$. In particular, each step of the noising process adds Gaussian noise according to some variance schedule given by β_t:

$$p\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}\right) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\, \mathbf{x}_{t-1}, \beta_t \mathrm{I}\right) \tag{11}$$

In addition, it's possible to sample $\mathbf{x}_t$ directly from $\mathbf{x}_0$ without repeatedly sample from $\mathbf{x}_t \sim p\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}\right)$. Instead, $p\left(\mathbf{x}_t \mid \mathbf{x}_0\right)$ can be expressed as a Gaussian distribution by defining a variance of the noise for an arbitrary timestep $\alpha_t := \prod_{j=0}^{t}\left(1-\beta_j\right)$. We, therefore, proceed to define

$$\begin{aligned} p\left(\mathbf{x}_t \mid \mathbf{x}_0\right) &= \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha}_t \mathbf{x}_0, (1-\alpha_t)\mathrm{I}) && (12) \\ &= \sqrt{\alpha_t}\mathbf{x}_0 + \epsilon\sqrt{1-\alpha_t},\ \epsilon \sim \mathcal{N}(0, \mathrm{I}) && (13) \end{aligned}$$

However, we are interested in a generative process which consists in performing a reverse diffusion, going from noise $\mathbf{x}_T$ to data $\mathbf{x}_0$. As such, the model trained with parameters θ should correspond to conditional distribution $p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)$.

Using Bayes theorem, one finds that the posterior $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ is also a Gaussian with mean $\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$ and variance $\tilde{\beta}_t$ defined as follows:

$$\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\alpha_{t-1} - \alpha_t}}{1-\alpha_t}\mathbf{x}_0 + \frac{\alpha_t(1-\alpha_{t-1})}{\alpha_{t-1}(1-\alpha_t)}\mathbf{x}_t \qquad \tilde{\beta}_t := \frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t \tag{14}$$

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathrm{I}) \tag{15}$$

Training $p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)$ such that $p(\mathbf{x}_0)$ learns the true data distribution, the following variational lower-bound L_{vlb} for $p_\theta(\mathbf{x}_0)$ can be optimized:

$$L_{\text{vlb}} := -\log p_\theta(\mathbf{x}_0|x_1) + \sum_{t=2}^{T} D_{KL}(p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \mid\mid p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \tag{16}$$

Ho et al. (2020) considered a variational approximation of the Eq. 15 for training $p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)$ efficiently. Instead of directly parameterize $\mu_\theta(\mathbf{x}_t, t)$ as a neural network, a model $\epsilon_\theta(\mathbf{x}_t, t)$ is trained to predict ϵ from Equation 13. This simplified objective is defined as follows:

$$L_{\text{simple}} := \mathbb{E}_{t, \mathbf{x}_0 \sim p_{\text{data}}, \epsilon \sim \mathcal{N}(0,\mathrm{I})}\left[(1-\alpha_t)\left\|\boldsymbol{\epsilon}_\theta(\sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\epsilon, t) - \epsilon\right\|_2^2\right] \tag{17}$$

Appendix B. Pearl's Causal Hierarchy

Bareinboim et al. (2020) use *Pearl's Causal Hierarchy* (PCH) nonmenclature after Pearl's seminal work on causality which is well illustrated in Pearl and Mackenzie (2018) as the *Ladder of Causation*. PCH states that structural causal models should be able to sample from a collection of three distributions (Peters et al. (2017), Ch. 6) which are related to cognitive capabilities:

1. The *observational* ("seeing") distribution $p_{\mathfrak{G}}(\mathbf{x}^{(k)})$.

2. The do-calculus (Pearl, 2009) formalizes sampling from the *interventional* ("doing") distribution $p_{\mathfrak{G}}(\mathbf{x}^{(k)} \mid do(\mathbf{x}^{(j)} = x^{(j)}))$. The $do()$ operator means an intervention on a specific variable is propagated only through it's descendants in the SCM $\mathfrak{G}$. The causal structure forces that only the descendants of the variable intervened upon will be modified by a given action.

3. Sampling from a *counterfactual* ("imagining") distribution $p_{\mathfrak{G}}(\mathbf{x}^{(k)} \mid do(\mathbf{x}^{(j)} = x^{(j)}); x^{(k)})$ involves applying an intervention $do(\mathbf{x}^{(j)} = x^{(j)})$ on an given instance $\mathbf{x}^{(k)}$. Contrary to the factual observation, a counterfactual corresponds to a hypothetical scenario.

Appendix C. Example of Anti-causal Intervention

We illustrate Prop. 1 in a case with two variables, which is also used in the experiments. Consider a variable $\mathbf{x}^{(1)}$ caused by $\mathbf{x}^{(2)}$, *i.e.* $\mathbf{x}^{(1)} \leftarrow \mathbf{x}^{(2)}$. Following the causal direction, the joint distribution can be factorised as $p(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = p(\mathbf{x}^{(1)} \mid \mathbf{x}^{(2)})p(\mathbf{x}^{(2)})$. Applying an intervention with the SDE framework, however, one would only need $\nabla_{x^{(1)}} \log p_t(\mathbf{x}^{(1)} \mid \mathbf{x}^{(2)} = x^{(2)})$, as in Eq. 6. By applying Bayes' rule, one can derive $p(\mathbf{x}^{(1)} \mid \mathbf{x}^{(2)}) = p(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)})p(\mathbf{x}^{(1)})/p(\mathbf{x}^{(2)})$. Therefore, the sampling process would be done with

$$\nabla_{x^{(1)}} \log p(\mathbf{x}^{(1)} \mid \mathbf{x}^{(2)}) \propto \nabla_{x^{(1)}} \log p(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)}) + \nabla_{x^{(1)}} \log p(\mathbf{x}^{(1)}). \tag{18}$$

Appendix D. DDIM sampling procedure

A variation of the DDPM (Ho et al., 2020) sampling procedure is done with Denoising Diffusion Implicit Models (DDIM, Song et al. (2021a)). DDIM formulates an alternative non-Markovian noising process that allows a deterministic mapping between latents to images. The deterministic mapping means that the noisy term in Eq. 2 is no longer necessary for sampling. This sampling approach has the same forward marginals as DDPM, therefore, it can be trained in the same manner. This approach was used for sampling throughout the paper as explained in Sec. 3.4.

Alg. 2 describes DDIM's sampling procedure from $\mathbf{x}_T \sim \mathcal{N}(0, \boldsymbol{I})$ (exogenous noise distribution) to $\mathbf{x}_0$ (data distribution) deterministic procedure. This formulation has two main advantages: (i) it allows a near-invertible mapping between $\mathbf{x}_T$ and $\mathbf{x}_0$ as shown in Alg. 3; and (ii) it allows efficient sampling with fewer iterations even when trained with the same diffusion discretization. This is done by choosing different undersampling t in the $[0, T]$ interval.

Algorithm 2 Sampling with DDIM - Image Generation

```
Models: trained diffusion model $\boldsymbol{\epsilon}_\theta$.
Input  : $x_T \sim \mathcal{N}(0, \mathrm{I})$
Output: $x_0$ - Image
for $t \leftarrow T$ to $0$ do
    $x_{t-1} \leftarrow \sqrt{\alpha_{t-1}} \left( \frac{x_t - \sqrt{1-\alpha_t}\, \boldsymbol{\epsilon}_\theta(x_t, t)}{\sqrt{\alpha_t}} \right) + \sqrt{\alpha_{t-1}}\, \boldsymbol{\epsilon}_\theta(x_t, t)$
end
```

Algorithm 3 Reverse-Sampling with DDIM - Inferring the Noisy Latent

```
Models: trained diffusion model $\boldsymbol{\epsilon}_\theta$.
Input  : $x_0$ - Image
Output: $x_T$ - Latent Space
for $t \leftarrow T$ to $0$ do
    $x_{t+1} \leftarrow \sqrt{\alpha_{t+1}} \left( \frac{x_t - \sqrt{1-\alpha_t}\, \boldsymbol{\epsilon}_\theta(x_t,t)}{\sqrt{\alpha_t}} \right) + \sqrt{\alpha_{t+1}}\, \boldsymbol{\epsilon}_\theta(x_t, t)$
end
```

Appendix E. Implementation Details

For each dataset, we train two models that are trained separately: (i) $\boldsymbol{\epsilon}_\theta$ is implemented as an encoder-decoder architecture with skip-connections, *i.e.* a Unet-like network (Ronneberger et al., 2015). (ii) A (Anti-causal) classifier that uses the encoder of $\boldsymbol{\epsilon}_\theta$ with a pooling layer followed by a linear classifier. All models are time conditioned. Time, which is a scalar, is embedded using the transformer's sinusoidal position embedding (Vaswani et al., 2017). The embedding is incorporated into the convolutional models with an Adaptive Group Normalization layer into each residual block (Nichol and Dhariwal, 2021). Our architectures and training procedure follow Dhariwal and Nichol (2021). They performed an extensive ablation study of important components from DDPM (Ho et al., 2020) and improved overall image quality and log-likelihoods on many image benchmarks. We use the same hyperparameters as Dhariwal and Nichol (2021) for the ImageNet and define ours for MNIST. The specific hyperparameters for diffusion and classification models follow Tab. 2. We train all of our models using Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We train in 16-bit precision using loss-scaling, but maintain 32-bit weights, EMA, and optimizer state. We use an EMA rate of 0.9999 for all experiments.

We use DDIM sampling for all experiments with 1000 timesteps. The same noise schedule is used for training. Even though DDIM allows faster sampling, we found that it does not work well for counterfactuals.

dataset	ImageNet 256	ImageNet 256	MNIST	MNIST
model	diffusion	classifier	diffusion	classifier
Diffusion steps	1000	1000	1000	1000
Model size	554M	54M	2M	500K
Channels	256	128	64	32
Depth	2	2	1	1
Channels multiple	1,1,2,2,4,4	1,1,2,2,4,4	1,2,4	1,2,4,4
Attention resolution	32,16,8	32,16,8	-	-
Batch size	256	256	256	256
Iterations	$\approx$ 2M	$\approx$ 500K	30K	3K
Learning Rate	1e-4	3e-4	1e-4	1e-4

Table 2: Hyperparameters for models.

Appendix F. Sampling from The Interventional Distribution

In this section, we make sure that our method complies with the second level of *Pearl's Causal Hierarchy* (details in Appendix B). Diff-SCM can be used for efficiently sampling from the interventional distributions $p_{\mathfrak{G}_{\text{image}}}(\mathbf{x}^{(1)} \mid do(\mathbf{x}^{(2)} = x^{(2)}))$. Sampling from the interventional distribution can be done by using the second part ("Generation with Intervention") of Alg. 1 but sampling $u^{(k)}$ from a Gaussian prior, instead of inferring the latent space (using "Abduction of Exogenous Noise"). This formulation is identical to Dhariwal and Nichol (2021) with guided DDIM (Song et al., 2021a) (details in appendix D). Dhariwal and Nichol (2021) achieves state-of-the-art image quality results in generation while providing faster sampling than DDPM. Since its capabilities in image synthesis compared to other generative models are shown in Dhariwal and Nichol (2021), we restrict ourselves to present qualitative results on ImageNet 256x256.

Experimental Setup. Our experiment, depicted in Fig. 5, consists in sampling a single latent space $u^{(1)}$ from a Gaussian distribution and generating samples for different classes. Since all images are generated from the same latent, this allows visualization of the effect of the classifier guidance for different classes. This setup differs from experiments in Dhariwal and Nichol (2021), where each image presented was a different sample $u^{(1)} \sim \mathbf{u}^{(1)}$. Here, by sampling $\mathbf{u}^{(1)}$ only once, we isolate the contribution of the causal mechanism from the sampling of the exogenous noise $\mathbf{u}^{(1)}$. We use the scale hyperparameter $s = 5$ for these experiments.

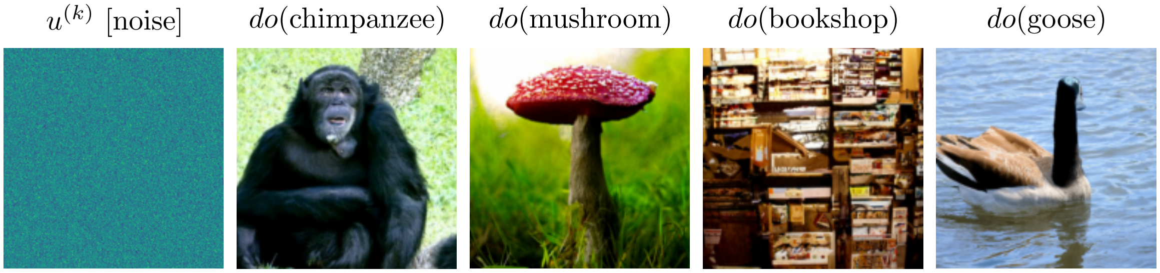


Figure 5: Sampling ImageNet images from the interventional distribution. All images originate from the same initial noise $u^{(k)}$ but different interventions are applied at inference time.

Appendix G. IM1 and IM2

Looveren and Klaise (2021) propose IM1 and IM2 for measuring the realism and closeness to the data manifold. These metrics are based on the reconstruction losses of auto-encoders trained on specific classes:

$$\text{IM1}(x_{\text{CF}}^{(1)}, x_{\text{F}}^{(2)}, x_{\text{CF}}^{(2)}) = \frac{\left\| x_{\text{CF}}^{(1)} - \text{AE}_{x_{\text{CF}}^{(2)}}(x_{\text{CF}}^{(1)}) \right\|_2^2}{\left\| x_{\text{CF}}^{(1)} - \text{AE}_{x_{\text{F}}^{(2)}}(x_{\text{CF}}^{(1)}) \right\|_2^2 + \epsilon} \tag{19}$$

$$\text{IM2}(x_{\text{CF}}^{(1)}, x_{\text{CF}}^{(2)}) = \frac{\left\| \text{AE}_{x_{\text{F}}^{(2)}}(x_{\text{CF}}^{(1)}) - \text{AE}(x_{\text{CF}}^{(1)}) \right\|_2^2}{\left\| x_{\text{CF}}^{(1)} \right\|_1 + \epsilon} \tag{20}$$

where $\mathrm{AE}_{x^{(2)}}$ denotes an autoencoder trained only on instances from class $x^{(2)}$, and AE is an autoencoder trained on data from all classes. IM1 is the ratio of the reconstruction loss of an autoencoder trained on the counterfactual class divided by the loss of an autoencoder trained on all classes. IM2 is the normalized difference between the reconstruction of the CF under an autoencoder trained on the counterfactual class, and one trained on all classes.

Appendix H. More MNIST Counterfactuals

Here, we show in Fig. 6 that we can generate counterfactuals of all MNIST classes, given factual image. We use the scale hyperparameter $s = 0.7$ for these experiments.

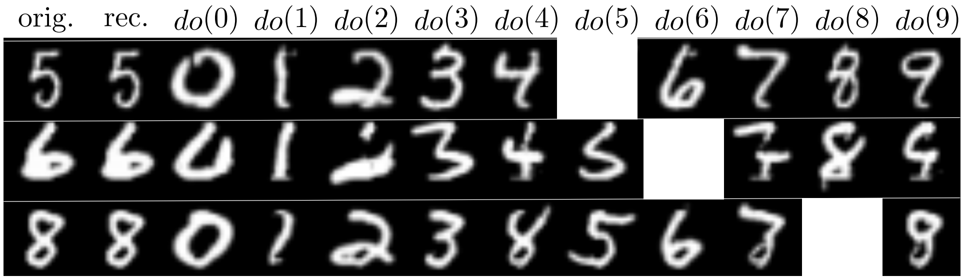


Figure 6: MNIST counterfactuals. From the left to right, one can observe the original image (*orig.*), the reconstruction (*rec.*, which entails in running the algorithm 1 without the anti-causal predictor) and the resulting counterfactuals for each of the digit classes in the dataset.

Appendix I. Qualitative influence of classifier scale

Here, we show in Fig. 7 the influence of changing the classifier's scale s quantitatively. If s is too low, the intervention will have a mild effect. On the other had, if s is too high, the intervention will neglect the information present in the exogenous noise, therefore, the counterfactual is maintain less factors from the original image.

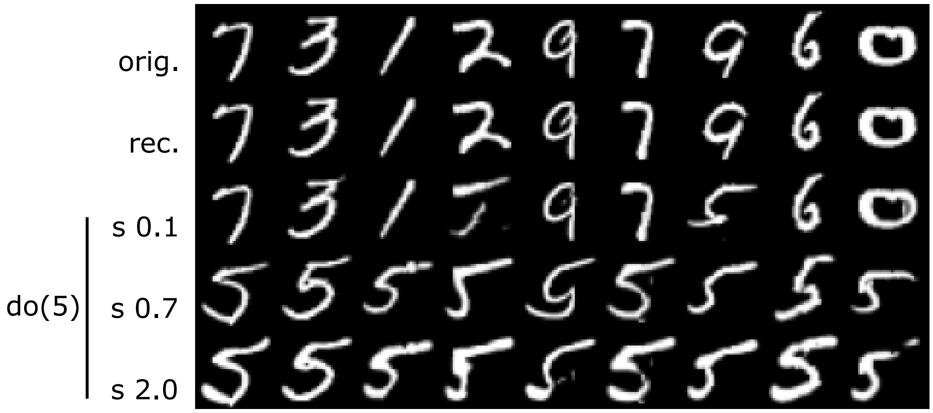


Figure 7: MNIST counterfactuals. From top to bottom, one can observe the original image (*orig.*), the reconstruction (*rec.*, and the resulting counterfactuals for the intervention $do(5)$ over three scales. As shown in Fig. 4, $s = 0.7$ is the optimal scale for MNIST data.