The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)

A Theory of Independent Mechanisms for Extrapolation in Generative Models

Michel Besserve,[1,2] Rémy Sun,[1,3,†] Dominik Janzing[1,†] and Bernhard Schölkopf[1]

[1] Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2] Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[3] ENS Rennes, France
{michel.besserve, bs}@tuebingen.mpg.de, janzind@amazon.de, remy.sun@ens-rennes.fr

Abstract

Generative models can be trained to emulate complex empirical data, but are they useful to make predictions in the context of previously unobserved environments? An intuitive idea to promote such *extrapolation* capabilities is to have the architecture of such model reflect a causal graph of the true data generating process, such that one can intervene on each node independently of the others. However, the nodes of this graph are usually unobserved, leading to overparameterization and lack of identifiability of the causal structure. We develop a theoretical framework to address this challenging situation by defining a weaker form of identifiability, based on the principle of *independence of mechanisms*. We demonstrate on toy examples that classical stochastic gradient descent can hinder the model's extrapolation capabilities, suggesting independence of mechanisms should be enforced explicitly during training. Experiments on deep generative models trained on real world data support these insights and illustrate how the extrapolation capabilities of such models can be leveraged.

1 Introduction

Deep generative models such as Generative Adversarial Networks (GANs) (Goodfellow et al. 2014), and Variational Autoencoders (VAEs) (Kingma and Welling 2013; Rezende, Mohamed, and Wierstra 2014) are able to learn complex structured data such as natural images. However, once such a network has been trained on a particular dataset, can it be leveraged to simulate meaningful changes in the data generating process? Capturing the causal structure of this process allows the different mechanisms involved in generating the data to be intervened on independently, based on the principle of *Independence of Mechanisms* (IM) (Janzing and Schölkopf 2010; Lemeire and Janzing 2012; Peters, Janzing, and Schölkopf 2017). IM reflects a foundational aspect of causality, related to concepts in several fields, such as superexogeneity in economics (Engle, Hendry, and Richard 1983), the general concept of invariance in philosophy (Woodward 2003) and *modularity*. In particular, having the internal computations performed by a multi-layer generative model reflect the true causal structure of the data generating mechanism would thus endow it with a form of layer modularity, such that intervening on intermediate layers causes changes in the output distribution similar to what would happen in the real world. We call such ability *extrapolation*, as it intuitively involves generalizing beyond the support of the distribution sampled during training, or its convex hull.

In this paper, **we focus on the challenging case where no additional variables, besides the samples from the data to generate, are observed** (in contrast with related work, as explained below). In this unsupervised setting, generative models are typically designed by applying successive transformations to latent variables, leading to a multi-layered architecture, where neither the latent inputs nor the hidden layers correspond to observed variables. We elaborate a general framework to assess extrapolation capabilities when intervening on hidden layer parameters with transformations belonging to a given group $\mathcal{G}$, leading to the notion of $\mathcal{G}$-genericity of the chosen parameters. We then show how learning based on stochastic gradient descent can hinder $\mathcal{G}$-genericity, suggesting additional control on the learning algorithm or the architecture is needed to enforce extrapolation abilities. Although we see our contribution as chiefly conceptual and theoretical, we use toy models and deep generative models trained on real world data to illustrate our framework.

Appendix. Readers can refer to the technical appendix in the extended version of this paper[4] for supplemental figures, code resources, symbols and acronyms (Table 1), all proofs (App. A) and method details (App. B).

Related Work. Deep neural network have been leveraged in causal inference for learning causal graphs between observed variables (Lopez-Paz and Oquab 2016) and associated causal effects (Louizos et al. 2017; Shalit, Johansson, and Sontag 2017; Kocaoglu et al. 2017; Lachapelle et al. 2019; Zhu, Ng, and Chen 2019). Our ultimate goal is more akin to the use of a causal framework to enforce domain adaptation (Zhang et al. 2013; Zhang, Gong, and Schölkopf 2015) and domain shift robustness of leaning algorithms, which has been done by exploiting additional information in the context of classification (Heinze-Deml and Meinshausen 2017). Broadly construed, this also relates to zero-shot learning (Lampert, Nickisch, and Harmeling 2009) and notions of

[†] DJ contributed before joining Amazon. RS contributed before joining Thales and Sorbonne University.

[4] https://arxiv.org/abs/2004.00184

extrapolations explored in the context of dynamical systems (Martius and Lampert 2016). As an intermediary step, unsupervised disentangled generative models are considered as a way to design data augmentation techniques that can probe and enforce the robustness of downstream classification tasks (Locatello et al. 2018; Higgins et al. 2017). A causal (counterfactual) framework for such disentanglement has been proposed by Besserve et al. (2020) that leverages the internal causal structure of generative models to generate meaningful changes in their output. In order to characterize and enforce such causal disentanglement properties, the IM principle has been exploited in empirical studies (Goyal et al. 2019; Parascandolo et al. 2018) and its superiority to statistical independence has been emphasized (Besserve et al. 2020; Locatello et al. 2018). However, deriving a measure for IM is challenging in practice. Our work builds on the idea of Besserve et al. (2018) to use group invariance to quantify IM in a flexible setting and relate it to identifiability of the model in the absence of information regarding variables causing the observations. Another interesting direction to address identifiability of deep generative model is non-linear ICA, but typically requires observation of auxiliary variables (Hyvarinen, Sasaki, and Turner 2019; Khemakhem et al. 2020). Finally, our investigation of overparameterization relates to previous studies (Neyshabur et al. 2017; Zhang et al. 2016), notably arguing that Stochastic Gradient Descent (SGD) implements an *implicit regularization* beneficial to supervised learning, while we provide a different perspective in the context of unsupervised learning and extrapolation.

2 Extrapolation in Generative Models

2.1 *FluoHair*: an Extrapolation Example in VAEs

We first illustrate what we mean by extrapolation, and its relevance to generalization and generative models with a straightforward transformation: color change. "Fluorescent" hair colors are at least very infrequent in classical face datasets such as CelebA[5], such that classification algorithms trained on these datasets may fail to extract the relevant information from pictures of actual people with such hair, as they are arguably outliers.

To foster the ability to generalize to such samples, one can consider using generative models to perform data augmentation. However, highly realistic generative models also require training on similar datasets, and are thus very unlikely to generate enough samples with atypical hair attributes.

Fig. 1 demonstrates a way to endow a generative model with such extrapolation capabilities: after identifying channels controlling hair properties in the last hidden layer of a trained VAE (based on the approach of Besserve et al. (2020)), the convolution kernel k of this last layer can be modified to generate faces with various types of fluorescence (see App. B.1 for details), while the shape of the hair cut, controlled by parameters in the above layers, remains the same, illustrating layer-wise modularity of the network. Notably, this approach to extrapolation is unsupervised: no labeling or preselection of training samples is used. Importantly, in our framework hair color is not controlled by a disentangled latent variable; we rely instead on the structure of VAE/GAN to intervene on color by changing the synaptic weights corresponding to hidden units influencing hair in the last (downstream) convolution layer thereby influencing output RGB channels (see in App. B.1). Such transformation of an element of the computational graph of the generative model will guide our framework. Although this example provides insights on how extrapolations are performed, it exploits some features specific to color encoding of images. To illustrate how our framework helps address more general cases, we will use a different class of interventions that stretch the visual features encoded across a hierarchy of convolutional layers (Model 1, Fig. 2).

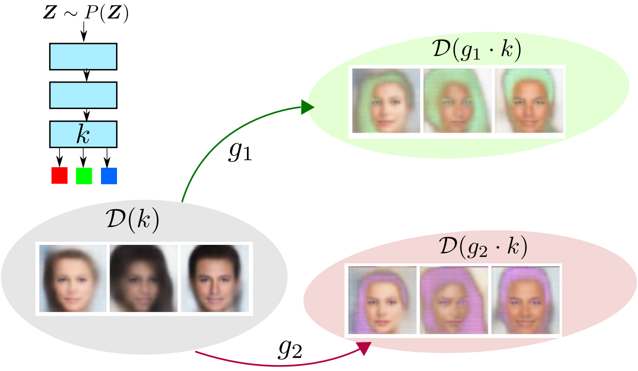


Figure 1: Illustration of *FluoHair* extrapolation for a VAE face generator (left inset). Transformations g_k modify kernel k and the sample distribution $\mathcal{D}$.

2.2 Neural Networks as Structural Causal Models

By selecting the output of a particular hidden layer as intermediate variable $\boldsymbol{V}$, we represent (without loss of generality) a multi-layer generative model as a composition of two functions $f_k^{\boldsymbol{\theta}_k}(.;\,)$, $k = 1, 2$, parameterized by $\boldsymbol{\theta}_k \in \mathcal{T}_k$, and applied successively to a latent variable $\boldsymbol{Z}$ with a fixed distribution, to generate an output random variable

$$\boldsymbol{X} = f_2^{\boldsymbol{\theta}_2}(\boldsymbol{V}) = f_2^{\boldsymbol{\theta}_2}(f_1^{\boldsymbol{\theta}_1}(\boldsymbol{Z}))\,. \qquad (1)$$

Assuming the mappings $\boldsymbol{\theta}_k \mapsto f_k$ are one-to-one, we abusively denote parameter values by their corresponding function pair. Besides pathological cases, e.g. "dead" neurons resulting from bad training initialization, this assumption appears reasonable in practice.[6]

An assumption central to our work is that the data generating mechanism leading to the random variable $\boldsymbol{Y}$ representing observations corresponds to eq. (1) with the so-called *true* parameters $\boldsymbol{\theta}^* \in \mathcal{T}$ corresponding to (f_1^*, f_2^*). More precisely both functions f_1^* and f_2^* are assumed to capture causal mechanisms such that one can interpret eq. (1) as a structural causal model (Pearl 2000) with causal graph $\boldsymbol{Z} \to \boldsymbol{V} \to \boldsymbol{X}$.

We additionally assume that a learning algorithm fits perfectly the data distribution by choosing the vector of parame-

[5] http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[6] the opposite would mean e.g. for a convolutional layer, that two different choices of tensors weights lead to the exact same response for all possible inputs, which appears unlikely

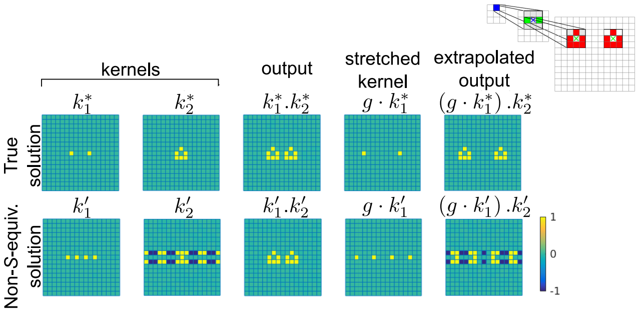


Figure 2: Numerical illustration for eye generator of Example 1. Top row: true parameters. Bottom: another solution in $S_{\boldsymbol{\theta}^*}$. (Top right inset) Illustration of the two successive convolutions (with additional striding).

ters $\widetilde{\boldsymbol{\theta}}$. This assumption allows us to focus on the theoretical underpinnings of extrapolation independent from the widely addressed question of fitting complex generative models to observations. In practical settings, this can be approached by choosing an architecture with universal approximation capabilities. Let $\mathcal{D}_{\boldsymbol{\theta}}$ denote the distribution of output $\boldsymbol{X}$ for any parameter pair $\boldsymbol{\theta}$ in $\boldsymbol{\mathcal{T}} = \mathcal{T}_1 \times \mathcal{T}_2$, then we have $\boldsymbol{Y} \sim \mathcal{D}_{\boldsymbol{\theta}^*} = \mathcal{D}_{\widetilde{\boldsymbol{\theta}}}$. The fitted parameters will thus belong to a *solution set* $S_{\boldsymbol{\theta}^*}$, defined as a set of function pairs that fit the observational distribution perfectly:

$$S_{\boldsymbol{\theta}^*} = \{(f_1, f_2) | \mathcal{D}_{(f_1,f_2)} = \mathcal{D}_{\boldsymbol{\theta}^*}\}\,. \qquad (2)$$

If $\widetilde{\boldsymbol{\theta}} = \boldsymbol{\theta}^*$, we can predict the distribution resulting from interventions on these parameters in the real world. We call such case *structural identifiability*. The IM principle at the heart of causal reasoning then allows *extrapolation* to other plausible distributions of output $\boldsymbol{Y}$ by intervening on one function while the other is kept fixed (see *FluoHair* example above). In contrast, if $S_{\boldsymbol{\theta}^*}$ is non-singleton and a value $\widetilde{\boldsymbol{\theta}} \neq \boldsymbol{\theta}^*$ is chosen by the learning algorithm, extrapolation is, in general, not guaranteed to behave like the true solution. One source on non-identifiability is the possibility that the pushforward measure of $\boldsymbol{Z}$ by two different functions $f_2 \circ f_1 = f \neq f' = f_2' \circ f_1'$ belonging to the model class may both match $\mathcal{D}_{\boldsymbol{\theta}^*}$ perfectly. In contrast, we will call *functionally identifiable* a true parameter $\boldsymbol{\theta}^*$ such that the composition $f = f_2 \circ f_1$ is uniquely determined by $\mathcal{D}_{\boldsymbol{\theta}^*}$. However, even a functionally identifiable parameter may not be structurally identifiable if f may by obtained by composing different pairs (f_1, f_2) and (f_1', f_2'). This last case is the focus of our framework, and will be illustrated using the following model.

Model 1 (Linear 2-layer convNet)**.** *Assume d, d' are two prime numbers*[7]*, $\boldsymbol{Z}$ a $(2d-1) \times (2d'-1)$ random binary latent image, such that one single pixel is set to one at each realization, and probability of this pixel to be located at (i, j) is $\boldsymbol{\pi}_{i,j}$. Let (k_1, k_2) be two invertible $(2d-1) \times (2d'-1)$ convolution kernels, and*

$$\boldsymbol{X} = k_2 \circledast \boldsymbol{V} = k_2 \circledast k_1 \circledast \boldsymbol{Z}, \qquad (3)$$

where $\circledast$ is the circular convolution (modulo $2d-1, 2d'-1$).

The reader can refer to App. B.3 for a background on circular convolution and how it relates to convolutional layers in deep networks. Such model can be used to put several copies of the same object in a particular spatial configuration at a random position in an image. The following example (Fig. 2) is an "eye generator" putting an eye shape at two locations separated horizontally by a fixed distance in an image to model the eyes of a (toy) human face. The location of this "eye pair" in the whole image may also be random.

Example 1 (Eye generator, Fig. 2)**.** *Consider Model 1 with k_2 a convolution kernel taking non-zero values within a minimal square of side $\delta < d$ encoding the eye shape, and k_1 with only two non-vanishing pixels, encoding the relative position of each eye.*

2.3 Characterization of the Solution Set

In the context of training such model from data without putting explicit constraints on each kernel, Model 1 admits "trivial" alternatives to the true parameters (k_1^*, k_2^*) to fit the data perfectly, simply by left-composing arbitrary rescalings and translations with k_1^*, and right-composing the inverse transformation to k_2^*. This is in line with observations by Neyshabur et al. (2017) in ReLU networks (incoming synaptic weights of a hidden unit can be downscaled while upscaling all outgoing weights).

To go beyond these mere observations, we systematically characterize over-parameterization entailed by composing two functions. Let $\boldsymbol{\mathcal{V}}$ be the range of $\boldsymbol{V}$, we define the subset Ω of right-invertible functions $\omega : \boldsymbol{\mathcal{V}} \to \boldsymbol{\mathcal{V}}$ such that for any pair $(f_1,\, f_2)$, $(\omega^{-1} \circ f_1, f_2 \circ \omega)$ also corresponds to a valid choice of model parameters.[8] Trivially, Ω contains at least the identity map. For any true parameter $\boldsymbol{\theta}^*$, we define the *Composed Over-parameterization Set* (COS)

$$S_{\boldsymbol{\theta}^*}^{\Omega} = \left\{ \left(\omega^{-1} \circ f_1^{\boldsymbol{\theta}_1^*},\, f_2^{\boldsymbol{\theta}_2^*} \circ \omega \right) | \, \omega \in \Omega \right\}\,. \qquad (4)$$

The COS reflects how "internal" operations in Ω make the optimization problem under-determined because they can be compensated by internal operations in neighboring layers. By definition, the COS is obviously a subset of the solution set $S_{\boldsymbol{\theta}^*}$. But if we consider *normalizing flow* (NF) models, in which f_k's are always invertible (following Rezende and Mohamed (2015)), we can show inclusion turns into equality.

Proposition 1. *For an NF model, Ω is a group and for any functionally identifiable true parameter $\boldsymbol{\theta}^*$, $S_{\boldsymbol{\theta}^*}^{\Omega} = S_{\boldsymbol{\theta}^*}$.*

Notably, this result directly applies to Model 1 (see Corollary 1 in App. B.6). We will exploit the COS group structure to study the link between identifiability and extrapolation, which we define next.

2.4 Extrapolated Class of Distributions

Humans can generalize from observed data by envisioning objects that were not previously observed, akin to our *FluoHair* example (Fig. 1). To mathematically define the notion of extrapolation, we pick interventions from a group $\mathcal{G}$ (i.e. a

[7] This will allow defining rigorously a group of transformations for extrapolation.

[8] We use the convention $A \circ \omega = \{f \circ \omega,\; f \in A\}$.

set of composable invertible transformations, see App. B.2 for background) to manipulate the abstract/internal representation instantiated by vector $\boldsymbol{V} = f_1(\boldsymbol{Z})$. Given parameter pair $\boldsymbol{\theta}^* = (f_1^*, f_2^*)$, we then define the *$\mathcal{G}$-extrapolated model class*, which contains the distributions generated by the interventions on $\boldsymbol{V}$ through the action of the group on f_1:

$$\mathcal{M}^{\mathcal{G}}_{(f_1^*, f_2^*)} = \mathcal{M}^{\mathcal{G}}_{\boldsymbol{\theta}^*} \triangleq \left\{\mathcal{D}_{(g \cdot f_1^*,\, f_2^*)},\, g \in \mathcal{G}\right\}\ , \tag{5}$$

where $g \cdot f_1^*$ denotes the group action of g on f_1^*, transforming it into another function. $\mathcal{G}$ thus encodes the inductive bias used to extrapolate from one learned model to others (when $\mathcal{G}$ is unambiguous $\mathcal{M}^{\mathcal{G}}_{\boldsymbol{\theta}^*}$ is denoted $\mathcal{M}_{\boldsymbol{\theta}^*}$). An illustration of the principle of extrapolated class, is provided in Suppl. Fig. 2.

Choosing the set of considered interventions to have a group structure allows to have an unequivocal definition of a uniform (Haar) measure on this set for computing expectations and to derive interesting theoretical results. Note this does not cover non-invertible hard interventions that set a variable to a fixed value $y = y_0$, while shifting the current value by a constant $y \to y + g$ does fit in the group framework. In the context of neural networks, this framework also allows to model a family of interventions on a hidden layer which can be restricted to only part of this layer, as done in the *FluoHair* example (see App. B.1).

The choice of the group is a form of application-dependent inductive bias. For Model 1, a meaningful choice is the multiplicative group $\mathcal{S}$ of integers modulo d (with d prime number, see App. B.5, such that the group action of stretching the horizontal image axis $\{-d+1, .., 0, d-1\}$ by factor $g \in \mathcal{S}$ turns convolution kernel k into $(g \cdot k)(m, n) = k(gm, gn)$. Such stretching is meant to approximate the rescaling of a continuous axis, while preserving group properties, and models classical feature variations in naturalistic images (see App. B.5). As an illustration for Example 1, using this group action leads to an extrapolated class that comprises models with various distances between the eyes, corresponding to a likely variation of human face properties. See Fig. 2, top row for an example extrapolation using this group. Interestingly, such spatial rescalings also correspond to frequency axis rescalings in the Fourier domain (see background in App. B.4). Indeed, let $\widehat{k}$ be the Discrete Fourier Transform (DFT) of kernel k, $(g \cdot k)(m, n) = k(gm, gn)$ corresponds to $(g \cdot \widehat{k})(u, v) = \widehat{k}(ug^{-1}, ng^{-1})$ such that the frequency axis is rescaled by the inverse of g. Due to the relationship between convolution and Fourier transform (App. B.4), several results for Model 1 will be expressed in the Fourier domain where convolution acts as a diagonal matrix multiplication.

2.5 Extrapolation Replaces Identification: $\mathcal{G}$-equivalence and $\mathcal{G}$-genericity

As elaborated above, $S_{\boldsymbol{\theta}^*}$ may not be singleton such that a solution $(\tilde{f}_1, \tilde{f}_2)$ found by the learning algorithm may not be the true pair (f_1^*, f_2^*), leading to a possibly different extrapolated class when intervening on f_1 with elements from group $\mathcal{G}$. When extrapolated classes happen to be the same, we say the solution is *$\mathcal{G}$-equivalent* to the true one.

Definition 1 ($\mathcal{G}$-equivalence)**.** *The solution $(\tilde{f}_1, \tilde{f}_2)$ is $\mathcal{G}$-equivalent to the true (f_1^*, f_2^*) if it generates the same extrapolated class through the action of $\mathcal{G}$: $\mathcal{M}^{\mathcal{G}}_{(\tilde{f}_1, \tilde{f}_2)} = \mathcal{M}^{\mathcal{G}}_{(f_1^*, f_2^*)}$.*

An illustration of $\mathcal{G}$-equivalence violation for Example 1 is shown in Fig. 2 (bottom row), and an additional representation of the phenomenon is given in Suppl. Fig. 2. Such equivalence of extrapolations imposes additional requirements on solutions. In the NF cases, such constraints rely on the interplay between the group structure of Ω (group of the COS in eq. (4)) that constrains over-parameterization, and the group structure of $\mathcal{G}$. For Model 1, in the 1D case this leads to

Proposition 2. *Assume $\widehat{\boldsymbol{\pi}}$ has no zero element and d'=1, the solution (k_1, k_2) for Model 1 is $\mathcal{S}$-equivalent to true model (k_1^*, k_2^*) if and only if there exists one $\lambda \in \mathbb{C}$ such that $(\widehat{k}_1(u)], \widehat{k}_2(u)]) = (\lambda^{-1}\widehat{k}_1^*(u), \lambda\widehat{k}_2^*(u))$ for all $u > 0$.*

This shows that at least in this model, $\mathcal{G}$-equivalence is achieved only for solutions that are very similar to the true parameters θ^* (up to a multiplicative factor), thus only slightly weaker than identifiability. As $\mathcal{G}$-equivalence requires knowledge of the true solution, in practice we resort to characterizing invariant properties of $\mathcal{M}_{\boldsymbol{\theta}^*}$ to select solutions. Indeed, if $\mathcal{M}_{\boldsymbol{\theta}^*}$ is a set that "generalizes" the true model distribution $\mathcal{D}_{\boldsymbol{\theta}^*}$, it should be possible to express the fact that some property of $\mathcal{D}_{\boldsymbol{\theta}^*}$ is *generic* in $\mathcal{M}_{\boldsymbol{\theta}^*}$. Let φ be a *contrast* function capturing approximately the relevant property of $\mathcal{D}_{\boldsymbol{\theta}^*}$, we check that such function does not change on average when applying random transformations from $\mathcal{G}$, by sampling from the Haar measure of the group $\mu_{\mathcal{G}}$,[9] leading to

Definition 2 (Contrast based $\mathcal{G}$-genericity)**.** *Let φ be a function mapping distributions on the generator output space to $\mathbb{R}$, and $\mathcal{G}$ a compact group. For any solution $(\tilde{f}_1, \tilde{f}_2)$ of the model fit procedure, we define the generic ratio*

$$\rho(\tilde{f}_1, \tilde{f}_2) = \rho(\tilde{f}_1(\boldsymbol{Z}), \tilde{f}_2) \triangleq \frac{\varphi(\mathcal{D}_{(\tilde{f}_1, \tilde{f}_2)})}{\mathbb{E}_{g \sim \mu_{\mathcal{G}}} \varphi(\mathcal{D}_{(g \cdot \tilde{f}_1, \tilde{f}_2)})}\,. \tag{6}$$

Solution $(\tilde{f}_1, \tilde{f}_2)$ is $\mathcal{G}$-generic w.r.t. φ, whenever it satisfies $\rho(\tilde{f}_1, \tilde{f}_2) = 1$.

It then follows trivially from the definition that $\mathcal{G}$-equivalence entails a form of $\mathcal{G}$-genericity.

Proposition 3. *For φ constant on $\mathcal{M}^{\mathcal{G}}_{\boldsymbol{\theta}^*}$, $\mathcal{G}$-equivalent to the true solution implies $\mathcal{G}$-generic w.r.t. φ.*

Genericity was originally defined by Besserve et al. (2018) as a measure of *independence* between cause $\boldsymbol{V} = f_1(\boldsymbol{Z})$ and mechanism f_2. In practice, genericity is not expected to hold rigorously but approximately (i.e. ρ should be close to one). In the remainder of the paper, we use interchangeably the "functional" notation $\rho(\tilde{f}_1, \tilde{f}_2)$ and the original "cause-mechanism" notation $\rho(\boldsymbol{V}, \tilde{f}_2)$.

2.6 Link Between Genericity and Direction of Causation

An interesting application of genericity is identifying the direction of causation : in several settings, if $\varphi(\tilde{f}_1(\boldsymbol{Z}), \tilde{f}_2) = 1$

[9] $\mu_{\mathcal{G}}$ is a "uniform" distribution on $\mathcal{G}$, see App. B.2

for the causal direction $\boldsymbol{V} \to \boldsymbol{X}$, reflecting a genericity assumption in a causal relation, then the *anti-causal* direction $\boldsymbol{X} \to \boldsymbol{V}$ is not generic as $\varphi(\boldsymbol{X}, \tilde{f}_2^{-1}) \neq 1$. As genericity, as measured empirically by its ratio, is only approximate (ratio not exactly equal to one), comparing genericity of the directions $\boldsymbol{Z} \to \boldsymbol{X}$ and $\boldsymbol{X} \to \boldsymbol{Z}$ can be used to support the validity of the genericity assumption. This comparison is supported by several works on identifiation of causal pairs using IM, showing identifiability can be obtained on toy examples by choosing the direction of causation that maximized genericity (Shajarisales et al. 2015; Zscheischler, Janzing, and Zhang 2011; Janzing, Hoyer, and Schölkopf 2010; Janzing et al. 2012). We use spectral independence to check genericity of neural network architectures in Sec. 4.

2.7 Scale and Spectral Independence

In the case of Example 1 and for stretching transformations, restricted to the 1D case (d'=1), one reasonable contrast is the total *Power* across non-constant frequencies, which can be written (see App. B.4)

$$\mathcal{P}(\mathbf{X}) = \frac{1}{d-1}\sum_{i\neq 0} |\widehat{k}_2(i)\widehat{k}_1(i)|^2 = \left\langle |\widehat{k}_2 \odot \widehat{k}_1|^2 \right\rangle , \quad (7)$$

where $\langle . \rangle$ denotes averaging over non zero frequencies and $\odot$ is the entrywise product. Indeed, this quantity is preserved when we stretch the distance between both eyes, as long as they do not overlap. The following result allows to exploit genericity to find a good solution:

Proposition 4 (Informal, see App. A). *For Model 1 in the 1D case, the $\mathcal{S}$-generic ratio with respect to $\varphi = \mathcal{P}$ is*

$$\rho(\boldsymbol{V}, k_2) = \frac{\langle \mathbb{E}|\widehat{k}_2 \odot \widehat{\boldsymbol{V}}|^2 \rangle}{\langle \mathbb{E}|\widehat{\boldsymbol{V}}|^2\rangle \langle |\widehat{k}_2|^2 \rangle} = \frac{\langle |\widehat{k}_2 \odot \widehat{k}_1|^2 \rangle}{\langle |\widehat{k}_1|^2\rangle \langle |\widehat{k}_2|^2 \rangle} , \quad (8)$$

Moreover, the true solution of Example 1 is $\mathcal{S}$-generic.

We call ρ the Spectral Density Ratio (SDR), as it appears as a discrete frequency version of the quantity introduced by Shajarisales et al. (2015) (baring the excluded zero frequency). We say such $\mathcal{S}$-generic solution w.r.t. ρ satisfies *scale or spectral independence*. This supports the use of SDR to check whether successive convolution layers implement mechanisms at independent scales.

3 How Learning Algorithms Affect Extrapolation Capabilities

3.1 Simplified Diagonal Model

When models are over-parameterized, the learning algorithm likely affects the choice of parameters, and thus the extrapolation properties introduced above. We will rely on a simplification of Model 1, that allows to study the mechanisms at play without the heavier formalism of convolution operations.

Model 2. *Consider the linear generative model of dimension $d-1$ with d prime number*

$$\boldsymbol{X} = \mathrm{A}\mathrm{B}\boldsymbol{Z} = \mathrm{diag}(a)\mathrm{diag}(b)\boldsymbol{Z} \quad (9)$$

with A, B square positive definite $(d-1)\times(d-1)$ diagonal matrices with diagonal coefficient vectors a and b, respectively, and $\boldsymbol{Z}$ a vector of positive independent random variables such that $\mathbb{E}|Z_k|^2 = 1, \forall k$.

Model 2 can be seen as a Fourier domain version of Model 1, with some technicalities dropped. In particular, we use real positive numbers instead of complex numbers, we drop the zero and negative frequencies by labeling dimensions as $\{1, 2, ..., d-1\}$ modulo d and considering the multiplicative action of $\mathcal{S}$ on these coordinates. We get analogous results as for Model 1 regarding the solution set and $\mathcal{S}$-equivalence (see Corol. 2 and Prop. 7 in App. B.6).

In order to measure genericity in a similar way as for Model 1, the power contrast becomes[10]

$$\tilde{\varphi}(B, A) = \tau\left[ABB^\top A^\top\right] = \frac{1}{d-1}\sum_{i=1}^{d} a_i^2 b_i^2 = \left\langle a^2 \odot b^2 \right\rangle$$

where $\tau[M]$ is the normalized trace $\frac{1}{d-1}\mathrm{Tr}[M]$. This leads to

Proposition 5. *In Model 2, the $\mathcal{S}$-generic ratio w.r.t. $\tilde{\varphi}(B, A)$ is* $\rho'(B, A) \triangleq \dfrac{\langle a^2 \odot b^2\rangle}{\langle a^2\rangle\langle b^2\rangle}$.

3.2 Drift of Over-parameterized Solutions

Consider Model 2 in the (degenerate) case of 1×1 matrices. To make the learning closer to a practical setting, we consider a VAE-like training: conditional on the latent variable $z = Z$, the observed data is assumed Gaussian with fixed variance σ^2 and mean given by the generator's output $a\cdot b\cdot z$ (e.g. in contrast to 1, noise is added after applying the second function, and one can retrieve the original setting in the limit case $\sigma^2 = 0$). To simplify the theoretical analysis, we study only the decoder of the VAE, and thus assume a fixed latent value $z = 1$, (i.e. the encoder part of the VAE infers a Dirac for the posterior of Z given the data). Assuming the true model $(a^* > 0, b^* > 0)$, we thus use data sampled from $\mathcal{N}(c = a^*b^*, \sigma^2)$, and learn (a, b) from it, assuming the data is sampled from a Gaussian with same variance and unknown mean parameter. First, considering infinite amounts of data, maximum likelihood estimation amounts to minimizing the KL-divergence between two univariate Gaussians with same variance and different mean, equivalent to:

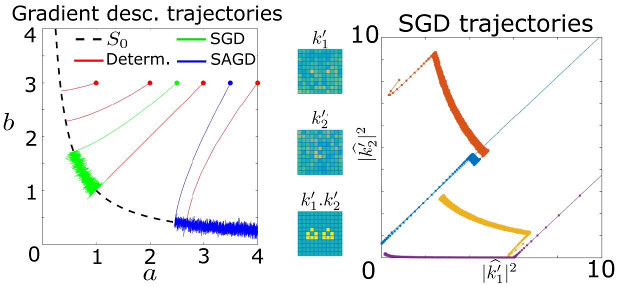
$$\underset{a,b>0}{\text{minimize}} \quad \mathcal{L}(c; (a, b)) = |c - ab|^2 . \quad (10)$$

We study the behavior of deterministic continuous time gradient descent (CTGD) in Prop. 8 of App. B.7. Typical trajectories are represented in red on Fig. 3a. We then consider the practical setting of SGD (see App. B.8) for training the VAE's decoder on the stochastic objective

$$\underset{a,b>0}{\text{minimize}}\ \ell(c_0; \omega; (a, b)) = |C(\omega) - ab|^2,\ C \sim \mathcal{N}(c_0, \sigma^2) . \quad (11)$$

The result (green sample path Fig. 3a) is very different from the deterministic case, as the trajectory drifts along S_{c_0} to asymptotically reach a neighborhood of $(\sqrt{c_0}, \sqrt{c_0})$. This

[10] This contrast is used for causal inference with the *Trace Method* (Janzing, Hoyer, and Schölkopf 2010), and relates to spectral independence Shajarisales et al. (2015).



(a)

(b)

Figure 3: (a) Gradient descent trajectories on the toy example of equation (10), $c = 1$. Thick dots indicate initial value. (b) SGD trajectories of several Fourier coefficients for Example 1. Final kernels obtained are on left.

drift is likely caused by asymmetries of the optimization landscape in the neighborhood of the optimal set S_{c_0}. This phenomenon relates to observations of an implicit regularization behavior of SGD (Zhang et al. 2016; Neyshabur et al. 2017), as it exhibits the same convergence to the minimum Euclidean norm solution. We provide a mathematical characterization of the drift in Prop. 9 (App. B.8). This result states that an SGD iteration makes points in the neighborhood of S_{c_0} evolve (on average) towards the line $\{a = b\}$, such that after many iterations the distribution concentrates around $(\sqrt{c_0}, \sqrt{c_0})$. Interestingly, if we try other variants of stochastic optimization on the same deterministic objective, we can get different dynamics for the drift, suggesting that it is influenced by the precise algorithm used (see App. B.9 for the case of Asynchronous SGD (ASGD) and example drift in blue on Fig. 3a).

We now get back to the multidimensional setting for Model 2. The above SGD results trivially apply to each component, which evolve independently from each other. Importantly, the next proposition shows that the SGD solution then drifts towards the matrix square root solution $\sqrt{A^*B^*}$ for both factors, leading to a violation of genericity.

Proposition 6. *In Model 2, assume diagonal coefficients of the true parameters A^* and B^* are i.i.d. sampled from two arbitrary non constant distributions. Then, the approximation of SGD solution $A = B = \sqrt{A^*B^*}$ satisfies*

$\rho'(B,\, A) \underset{d\to+\infty}{\longrightarrow} \mathbb{E}[c_1^2]/\mathbb{E}[c_1]^2 > 1,$ *and is thus not $\mathcal{S}$-generic.*

The solution chosen within S_c by the SGD algorithm is thus suboptimal for extrapolation.

3.3 Extension to Convolutional Model 1

We show qualitatively how the above observations for Model 2 can provide insights for Model 1. Using the same VAE-like SGD optimization framework for this case, where we consider $\boldsymbol{Z}$ deterministic, being this time a Dirac pixel at location $(0,0)$. We apply the DFT to $\boldsymbol{X}$ in Model 1 and use the Parseval formula to convert the least square optimization problem to the Fourier domain (see App. B.4). Simulating SGD of the real and imaginary parts of $\widehat{k}_1$ and $\widehat{k}_2$, we see in Fig. 3b the same drift behavior towards solutions having identical squared modulus ($|\widehat{k}_1|^2 = |\widehat{k}_2|^2$), as described for Model 2 in Sec. 3.2, reflecting the violation of $\mathcal{S}$-genericity by SGD of Prop. 6. As

$$\rho'(|\widehat{k}_1^*|^2, |\widehat{k}_2^*|^2) = \rho(k_1^*, k_2^*)\,. \tag{12}$$

this supports a violation of $\mathcal{S}$-genericity for the convolution kernels, such that the SGD optimization of Model 1 is also suboptimal for extrapolation.

3.4 Enforcing Spectral Independence

In order to enforce genericity and counteract the effects of SGD, we propose to alternate the optimization of the model parameters with SDR-based genericity maximization. To achieve this, we multiply the square difference between the SDR and its ideal value of 1 by the normalization term $\left\langle |\widehat{k}_2^i|^2 \right\rangle$ and alternate SGD steps of the original objective with gradient descent steps of the following problem

$$\underset{\widehat{k}_2}{\text{minimize}}\, (\rho(k_1^*, k_2^*) - 1)^2 \, \langle|\widehat{k}_2|^2\rangle^2 = \left\langle |\widehat{k}_2|^2 \odot \left(\frac{|\widehat{k}_1|^2}{\langle|\widehat{k}_1|^2\rangle} - 1\right)\right\rangle^2. \tag{13}$$

Performance of this procedure is investigated in App. B.12.

4 Experiments on Deep Face Generators

We empirically assess extrapolation abilities of deep convolutional generative networks, in the context of learning the distribution of CelebA. We used a plain β-VAE[11] ((Higgins et al. 2017)) and the official tensorlayer DCGAN implementation[12]. The general structure of the VAE is summarized in Suppl. Fig. 1b and the DCGAN architecture is very similar (details in Suppl. Table 2). Unless otherwise stated, our analysis is done on the generative architecture of these models (VAE encoder, GAN generator). We denote the 4 different convolutional layers as indicated in Suppl. Fig. 1b: coarse (closest to latent variables), intermediate, fine and image level. The theory developed in previous sections was adapted to match these applied cases, as explained in App. B.10.

4.1 Stretching Extrapolations

Extrapolations were performed by applying a 1.5 fold horizontal stretching transformation to all maps of a given hidden convolutional layer and compare the resulting perturbed image to directly stretching to the output sample. The extrapolated images obtained by distorting convolutional layers' activation maps are presented in the two middle rows of Fig. 4a for the VAE trained with 10000 iterations. Note the top and bottom rows respectively correspond to the original output samples, and the result of trivially applying stretching directly to them (these are only provided for comparison with respect to extrapolated samples). This affects differently features encoded at different scales of the picture: stretching the intermediate level activation maps (second row of Fig. 4a) mostly keeps the original dimensions of each eye, while inter-eye distance stretches in line with extrapolation

[11] https://github.com/yzwxx/vae-celebA

[12] https://github.com/tensorlayer/dcgan

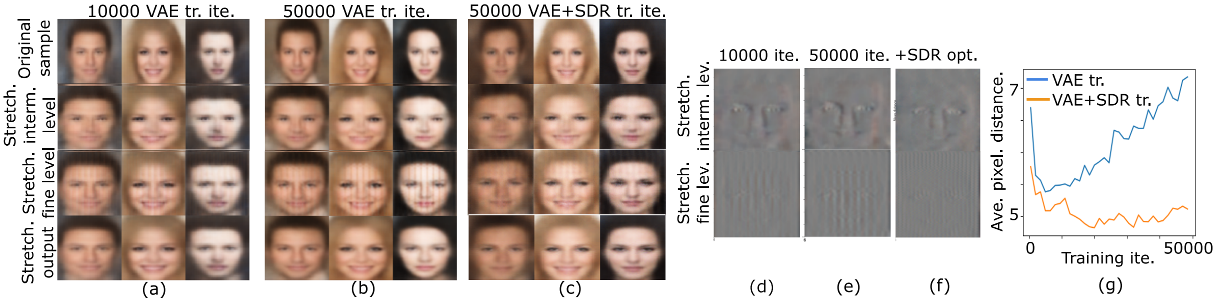


Figure 4: VAE stretching extrapolations. (a-c) VAE extrapolation samples (a-b classical VAE training, c using VAE interleaved with SDR optimization, see text). (d-f) Pixel difference between stretched output and extrapolated samples. (g) Evolution of MSE when stretching at the fine level, with and without SDR optimization. See also Suppl. Fig. 3.

ability that we introduced in Sec. 2 (Model 1). This suggests that the detrimental effect of SGD optimization investigated in Sec. 3.2 did not affect this layer. One plausible interpretation of this good extrapolation behavior is the fact that, in contrast with our toy examples, the intermediate level layer contains a large number of channels trained in parallel trough backpropagation. This may limit the propensity of the overparameterized solutions associated to a single channel to drift, due to the multiple pathways exploited during optimization. In contrast, extrapolation of the fine level activation maps (second row of Fig. 4a), results in slight vertical artifacts; a weaker extrapolation capability possibly related to the smaller number of channels in this layer. Interestingly, Fig. 4b replicating the result but after 40000 additional training iterations shows perturbed images of poorer quality for this layer. This suggests, as predicted in Section 3.2, a decrease of extrapolation capabilities with excessive training, as the drifting regime shown in Fig. 3b takes over. In particular, stronger periodic interference patterns like in Fig. 2 (bottom row) appear for the stretching of the fine level hidden layer, which comprises fewer channels, and are thus likely to undergo an earlier drift regime (compare Figs. 4b vs. 4a, 3rd row). To quantify this effect, we tracked the evolution (as the number of iterations grows) of the mean square errors for the complete picture (Fig. 4g), resulting from the stretch of the fine level convolutional layer. This difference grows as the training progresses and the same trend can be observed for the mean squared error of the complete picture.

We next investigated whether enforcing more $\mathcal{S}$-genericity between layers during optimization can temper this effect. We trained a VAE by alternatively minimizing spectral dependence of eq. (13) at image, fine and intermediate levels, interleaved with one SGD iteration on the VAE objective. Fig. 4c,g show a clear effect of spectral independence minimization on limiting the increase in the distortions as training evolves. This is confirmed by the analysis of pixel difference for 50000 iterations, as seen in Fig. 4f: perturbations of the intermediate and fine level exhibit better localization, compared to what was obtained at the same number of iterations (Fig. 4e) with classical VAE training, supporting the link between extrapolation and $\mathcal{S}$-genericity of Sec. 2.7. See also App. B.13.

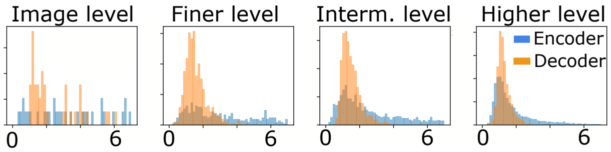


Figure 5: Superimposed SDR histograms of trained VAE decoder and encoder for different hidden layers.

4.2 Genericity of Encoder versus Decoder

The above qualitative results suggest that extrapolation capabilities are observable to some extent in vanilla generative architectures (the decoder of a VAE), but vary depending on the layer considered and can be improved by SDR optimization. We complement these qualitative observations by a validation of the genericity assumption based on the comparison with ”inverse” architecture (the encoder of a VAE, see App. B.10), in line with Sec. 2.6. We study the distribution of the SDR statistic between all possible (filter, activation map) pairs in a given layer. The result for the VAE is shown in Fig. 5, exhibiting a mode of the SDR close to 1 - the value of ideal spectral independence - for layers of the decoder, which suggests genericity of the convolution kernels between successive layers. Interestingly, the encoder, which implements convolutional layers of the same dimensions in reverse order, exhibits a much broader distribution of the SDR at all levels, especially for layers encoding lower level image features. This is in line with results stating presented in Sec. 2.6, that if a mechanism (here the generator) satisfies the principle of independent causal mechanisms, the inverse mechanism (here the encoder) will not (Shajarisales et al. 2015). In supplemental analysis, (App. B.13, Suppl. Fig. 5), we performed the same study on GANs.

Conclusion. Our framework to study extrapolation abilities of multi-layered generators based on *Independence of Mechanisms* replaces causal identifiability by a milder constraint of genericity, and shows how SGD training may be detrimental to extrapolation. Experiments are consistent with these insights and support spectral independence is a interesting indicator of IM in convolutional generative models. This provides insights to train statistical models that better capture the mechanisms of empirical phenomena.

Ethical Impact

Although this work is mostly theoretical and conceptual, we anticipate the following impact of this research direction. First, our work addresses how to enforce a causal structure in generative models trained from data. This allows developing statistical models that can better capture the outcomes of previously unseen perturbations to the system that generated the data, and as a consequence can have a positive impact on our ability to learn from observed data in context where experiments are impossible for ethical and practical reasons. Our focus on the notion of extrapolations is particularly suited to be investigate unprecedented climatic, economical and societal challenges facing humankind in the near future. Additionally, augmenting the learning algorithms of artificial systems with causal principles may allow more autonomy and robustness when facing novel environment, possibly leading to both positive and negative societal outcomes. Our approach however proposes a way to understand, formulate and control what kind of robustness should or should not be enforced, providing decision makers with information to guide their choices.

References

Besserve, M.; Mehrjou, A.; Sun, R.; and Schölkopf, B. 2020. Counterfactuals uncover the modular structure of deep generative models. In *ICLR2020*.

Besserve, M.; Shajarisales, N.; Schölkopf, B.; and Janzing, D. 2018. Group invariance principles for causal generative models. In *AISTATS*.

Engle, R. F.; Hendry, D. F.; and Richard, J.-F. 1983. Exogeneity. *Econometrica: Journal of the Econometric Society* 277–304.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

Goyal, A.; Lamb, A.; Hoffmann, J.; Sodhani, S.; Levine, S.; Bengio, Y.; and Schölkopf, B. 2019. Recurrent independent mechanisms. *arXiv preprint arXiv:1909.10893* .

Heinze-Deml, C.; and Meinshausen, N. 2017. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469* .

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR 2017*.

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

Janzing, D.; Hoyer, P.; and Schölkopf, B. 2010. Telling cause from effect based on high-dimensional observations. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*.

Janzing, D.; Mooij, J.; Zhang, K.; Lemeire, J.; Zscheischler, J.; Daniušis, P.; Steudel, B.; and Schölkopf, B. 2012. Information-geometric approach to inferring causal directions. *Artificial Intelligence* 182–183: 1–31.

Janzing, D.; and Schölkopf, B. 2010. Causal inference using the algorithmic Markov condition. *Information Theory, IEEE Transactions on* 56(10): 5168–5194.

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

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

Kocaoglu, M.; Snyder, C.; Dimakis, A. G.; and Vishwanath, S. 2017. Causalgan: Learning causal implicit generative models with adversarial training. *arXiv preprint arXiv:1709.02023* .

Lachapelle, S.; Brouillard, P.; Deleu, T.; and Lacoste-Julien, S. 2019. Gradient-based neural dag learning. *arXiv preprint arXiv:1906.02226* .

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2009. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 951–958. IEEE.

Lemeire, J.; and Janzing, D. 2012. Replacing Causal Faithfulness with Algorithmic Independence of Conditionals. *Minds and Machines* 1–23. doi:10.1007/s11023-012-9283-1.

Locatello, F.; Bauer, S.; Lucic, M.; Gelly, S.; Schölkopf, B.; and Bachem, O. 2018. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359* .

Lopez-Paz, D.; and Oquab, M. 2016. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545* .

Louizos, C.; Shalit, U.; Mooij, J. M.; Sontag, D.; Zemel, R.; and Welling, M. 2017. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, 6446–6456.

Martius, G.; and Lampert, C. H. 2016. Extrapolation and learning equations. *arXiv preprint arXiv:1610.02995* .

Neyshabur, B.; Tomioka, R.; Salakhutdinov, R.; and Srebro, N. 2017. Geometry of optimization and implicit regularization in deep learning. *arXiv preprint arXiv:1705.03071* .

Parascandolo, G.; Kilbertus, N.; Rojas-Carulla, M.; and Schölkopf, B. 2018. Learning Independent Causal Mechanisms. In *ICML*, 4036–4044.

Pearl, J. 2000. *Causality: models, reasoning and inference*, volume 29. Cambridge Univ Press.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of Causal Inference – Foundations and Learning Algorithms*. MIT Press.

Rezende, D. J.; and Mohamed, S. 2015. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770* .

Rezende, D. J.; Mohamed, S.; and Wierstra, D. 2014. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082* .

Shajarisales, N.; Janzing, D.; Schölkopf, B.; and Besserve, M. 2015. Telling cause from effect in deterministic linear dynamical systems. In *ICML 2015*.

Shalit, U.; Johansson, F. D.; and Sontag, D. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 3076–3085. JMLR. org.

Woodward, J. F. 2003. *Making Things Happen, a Theory of Causal Explanation*. Oxford University Press.

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2016. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530* .

Zhang, K.; Gong, M.; and Schölkopf, B. 2015. Multi-source domain adaptation: A causal view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Zhang, K.; Schölkopf, B.; Muandet, K.; and Wang, Z. 2013. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, 819–827. PMLR.

Zhu, S.; Ng, I.; and Chen, Z. 2019. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477* .

Zscheischler, J.; Janzing, D.; and Zhang, K. 2011. Testing whether linear equations are causal: A free probability theory approach. In *UAI 2011*.