Supplementary Material of CausalVAE: Disentangled Representation Learning via Neural Structural Causal Models

Mengyue Yang[1,2], Furui Liu[1,*], Zhitang Chen[1], Xinwei Shen[3], Jianye Hao[1], Jun Wang[2]
[1] Noah's Ark Lab, Huawei, Shenzhen, China
[2] University College London, London, United Kingdom
[3] The Hong Kong University of Science and Technology, Hong Kong, China
{yangmengyue2,liufurui2,chenzhitang2,haojianye}@huawei.com
xshenal@connect.ust.hk
jun.wang@cs.ucl.ac.uk

A. Proof of Proposition 1

Write the KL term in ELBO defined in Eq. 8 in the main text as

$$\begin{aligned}
&\mathcal{D}[q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\|p_\theta(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{u})] \\
=&\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})}{p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})p_\theta(\mathbf{z}|\mathbf{u})}d\boldsymbol{\epsilon}d\mathbf{z} \\
=&\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})}{p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})}d\boldsymbol{\epsilon}d\mathbf{z} \\
&+\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})}{p_\theta(\mathbf{z}|\mathbf{u})}d\boldsymbol{\epsilon}d\mathbf{z} \\
&-\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})d\boldsymbol{\epsilon}d\mathbf{z},
\end{aligned}$$

The third term in above equation could be rewritten as a constant. Details are shown as below.

$$\begin{aligned}
&-\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})d\boldsymbol{\epsilon}d\mathbf{z} \\
=&-\iint q(\epsilon|x,u)\delta(z=C\epsilon)\log q(\epsilon|x,u)d\epsilon dz \\
&-\iint q(\epsilon|x,u)\delta(z=C\epsilon)\log\delta(z=C\epsilon)d\epsilon dz \\
=&H(q_\phi(\epsilon|x,u))-0=H(\mathcal{N}(\mu_\phi(x,u),s\mathrm{I}))) \\
=&const, \qquad (1)
\end{aligned}$$

In our method, we ignore this term in ELBO expression. Then, based on Eq. 9 in the main text, we have

$$\begin{aligned}
&\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})}{p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})}d\boldsymbol{\epsilon}d\mathbf{z} \\
=&\int q_\phi(\boldsymbol{\epsilon}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon}|\mathbf{x},\mathbf{u})}{p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})}\int\delta(\mathbf{z}=\mathbf{C}\boldsymbol{\epsilon})d\mathbf{z}d\boldsymbol{\epsilon} \\
&+\int q_\phi(\boldsymbol{\epsilon}|\mathbf{x},\mathbf{u})\int\delta(\mathbf{z}=\mathbf{C}\boldsymbol{\epsilon})\log\delta(\mathbf{z}=\mathbf{C}\boldsymbol{\epsilon}))d\mathbf{z}d\boldsymbol{\epsilon} \\
=&\mathcal{D}[q_\phi(\boldsymbol{\epsilon}|\mathbf{x},\mathbf{u})\|p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})]+0 \\
=&\mathcal{D}[q_\phi(\boldsymbol{\epsilon}|\mathbf{x},\mathbf{u})\|p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})],
\end{aligned}$$

and

$$\begin{aligned}
&\iint q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{\epsilon},\mathbf{z}|\mathbf{x},\mathbf{u})}{p_\theta(\mathbf{z}|\mathbf{u})}d\boldsymbol{\epsilon}d\mathbf{z} \\
=&\int q_\phi(\boldsymbol{z}|\mathbf{x},\mathbf{u})\log\frac{q_\phi(\boldsymbol{z}|\mathbf{x},\mathbf{u})}{p_\theta(\mathbf{z}|\mathbf{u})}\int\delta(\boldsymbol{\epsilon}=\mathbf{C}^{-1}\boldsymbol{z})d\boldsymbol{\epsilon}d\mathbf{z} \\
&+\int q_\phi(\boldsymbol{z}|\mathbf{x},\mathbf{u})\int\delta(\boldsymbol{\epsilon}=\mathbf{C}\boldsymbol{z})\log\delta(\boldsymbol{\epsilon}=\mathbf{C}^{-1}\boldsymbol{z})d\boldsymbol{\epsilon}d\mathbf{z} \\
=&\mathcal{D}[q_\phi(\boldsymbol{z}|\mathbf{x},\mathbf{u})\|p_\theta(\mathbf{z}|\mathbf{u})]+0 \\
=&\mathcal{D}[q_\phi(\boldsymbol{z}|\mathbf{x},\mathbf{u})\|p_\theta(\mathbf{z}|\mathbf{u})].
\end{aligned}$$

Adding up the above two terms leads to the desired form of Proposition 1.

B. Identifiability

B.1. Proof of Theorem 1

The general logic of the proofing follows [11], but we focus on both encoder and decoder. In our setting, we has joint latent variables $\boldsymbol{\epsilon},\mathbf{z}$, and we prove identidfiabilty of both of them.

*Corresponding author.

Another different setting from iVAE is that we consider a slighter transformation matrix, since our additional observations $\mathbf{u}$ of each concepts align to each causal representations $\mathbf{z}$.

Sketch of proof:

We analyze the identifiability of $\boldsymbol{\epsilon}$ starting with $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{u}) = p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{u})$. Then we define a new invertible matrix $\mathbf{L}$ which contains additional observation u_i in causal system, and use it to prove that the learned $\widetilde{\mathbf{T}}$ is the transformation of $\mathbf{T}$. Step 2: We take the inference model into consideration and analyze the identifiablity of the inference model by relating the inference model to the generative model.

Details:

At the begining of proof, we consider a simple condition that the dimension of observation data d equals to the dimension of latent variables n.

The distribution has two sufficient statistics, the mean and variance of $\mathbf{z}$, which are denoted by sufficient statistics $\mathbf{T}(\mathbf{z}) = (\boldsymbol{\mu}(\mathbf{z}), \boldsymbol{\sigma}(\mathbf{z})) = (T_{1,1}(z_1), \ldots, T_{n,2}(z_n))$. We use these notations for model identifiability analysis in Section 5. To simplify proof process, we absorb the injective functions $\mathbf{g}(\cdot)$ into generate model $\mathbf{f}(\cdot)$ since mask layer will not influence the quality of disentangled representation $\mathbf{z}$.

$$
\begin{aligned}
& p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{u}) = p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{u}), \\
\Rightarrow & \iint_{\mathbf{z},\boldsymbol{\epsilon}} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z},\boldsymbol{\epsilon}) p_{\boldsymbol{\theta}}(\mathbf{z},\boldsymbol{\epsilon}|\mathbf{u}) d\mathbf{z} d\boldsymbol{\epsilon} \\
& = \iint_{\mathbf{z},\boldsymbol{\epsilon}} p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{z},\boldsymbol{\epsilon}) p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{z},\boldsymbol{\epsilon}|\mathbf{u}) d\mathbf{z} d\boldsymbol{\epsilon}, \\
\Rightarrow & \int_{\mathbf{z}} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{u}) d\mathbf{z} = \iint_{\mathbf{z}} p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{z}) p_{\widetilde{\boldsymbol{\theta}}}(\mathbf{z}|\mathbf{u}) d\mathbf{z}, \\
\Rightarrow & \int_{\mathbf{x}'} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{f}^{-1}(\mathbf{x}')) p_{\boldsymbol{\theta}}(\mathbf{f}^{-1}(\mathbf{x}')|\mathbf{u}) |\det(J_{\mathbf{f}^{-1}}(\mathbf{x}'))| d\mathbf{x}' \\
& = \int_{\mathbf{x}'} p_{\boldsymbol{\theta}}(\mathbf{x}|\tilde{\mathbf{f}}^{-1}(\mathbf{x}')) p_{\widetilde{\boldsymbol{\theta}}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}')|\mathbf{u}) |\det(J_{\tilde{\mathbf{f}}^{-1}}(\mathbf{x}'))| d\mathbf{x}'.
\end{aligned} \tag{2}
$$

In determining function $\mathbf{f}$, there exist a Gaussian distribution $p_{\boldsymbol{\xi}}(\boldsymbol{\xi})$ which has infinitesimal variance. Then, the $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{f}^{-1}(\mathbf{x}'))$ can be written as $p_{\boldsymbol{\xi}}(\mathbf{x} - \mathbf{x}')$. As the assumption (1) holds, this term is vanished. Then in our method, there exists the following equation:

$$
\begin{aligned}
p_{\boldsymbol{\theta}}(\mathbf{f}^{-1}(\mathbf{x}')|\mathbf{u}) |\det(J_{\mathbf{f}^{-1}}(\mathbf{x}'))| = p_{\widetilde{\boldsymbol{\theta}}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}')|\mathbf{u}) |\det(J_{\tilde{\mathbf{f}}^{-1}}(\mathbf{x}'))|, \\
\Rightarrow \widetilde{p}_{\boldsymbol{\theta}}(\mathbf{x}) = \widetilde{p}_{\widetilde{\boldsymbol{\theta}}}(\mathbf{x}).
\end{aligned} \tag{3}
$$

Adopting the definition of multivariate Gaussian distribution, we define

$$
\boldsymbol{\lambda}_s(\mathbf{u}) = \begin{bmatrix} \lambda_1^s(u_1) & & \\ & \ddots & \\ & & \lambda_n^s(u_n) \end{bmatrix}. \tag{4}
$$

There exists the following equations:

$$
\begin{aligned}
& \log|\det(J_{\mathbf{f}^{-1}}(\mathbf{x}))| - \log \mathbf{Q}(\mathbf{f}^{-1}(\mathbf{x})) + \log \mathbf{Z}(\mathbf{u}) \\
& + \sum_{s=1}^{2} \mathbf{T}_s(\mathbf{f}^{-1}(\mathbf{x})) \boldsymbol{\lambda}_s(\mathbf{u}),
\end{aligned} \tag{5}
$$

$$
\begin{aligned}
& = \log|\det(J_{\tilde{\mathbf{h}}}(\mathbf{x}))| - \log \widetilde{\mathbf{Q}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \log \tilde{\mathbf{Z}}(\mathbf{u}) \\
& + \sum_{s=1}^{2} \widetilde{\mathbf{T}}_s(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) \widetilde{\boldsymbol{\lambda}}_s(\mathbf{u}),
\end{aligned} \tag{6}
$$

where $\mathbf{Q}$ denotes the base measure. In Gaussian distribution, it is $\boldsymbol{\sigma}(\mathbf{z})$.

In learning process, $\widetilde{\mathbf{A}}$ is restricted as DAG. Thus, the $\widetilde{\mathbf{C}}$ exists which is full rank matrix. The item which is not related to u in Eq. 6 are cancelled out [**?**].

$$
\sum_{s=1}^{2} \mathbf{T}_s(\mathbf{f}^{-1}(\mathbf{x})) \boldsymbol{\lambda}_s(\mathbf{u}) = \sum_{s=1}^{2} \widetilde{\mathbf{T}}_s(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) \widetilde{\boldsymbol{\lambda}}_s(\mathbf{u}) + \mathbf{b}, \tag{7}
$$

where $\mathbf{b}$ is a vector related to $\mathbf{u}$.

In our model, there exist a deterministic relationship $\mathbf{C}$ between $\boldsymbol{\epsilon}$ and $\mathbf{z}$ where $\mathbf{C} = (\mathbf{I} - \mathbf{A}^T)^{-1}$. Thus we could get equivalent of Eq. 7 as follows,

$$
\sum_{s=1}^{2} \mathbf{T}_s(\mathbf{C}\mathbf{h}(\mathbf{x})) \boldsymbol{\lambda}_s(\mathbf{u}) = \sum_{s=1}^{2} \widetilde{\mathbf{T}}_s(\widetilde{\mathbf{C}}\widetilde{\mathbf{h}}(\mathbf{x})) \widetilde{\boldsymbol{\lambda}}_s(\mathbf{u}) + \mathbf{b}', \tag{8}
$$

where s denote the index of sufficient statistics of Gaussian distributions, indexing the mean (1) and the variance (2).

By assuming that the additional observation u_i is different, it is guaranteed that coefficients of the observations for different concepts are distinct. Thus, there exists an invertible matrix corresponding to additional information $\mathbf{u}$:

$$
\mathbf{L} = \begin{bmatrix} \boldsymbol{\lambda}_1(\mathbf{u}) & \\ & \boldsymbol{\lambda}_2(\mathbf{u}) \end{bmatrix}. \tag{9}
$$

Since the assumption that $u_i \neq 0$ holds, $\mathbf{L}$ is $2n \times 2n$ invertible and full rank diagonal matrix. Then, function of $\boldsymbol{\lambda}$ in Eq. 7 and Eq. 8 are replcaed by Eq. 9, we could get:

$$
\mathbf{L}\mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = \widetilde{\mathbf{L}}\widetilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{b}, \tag{10}
$$

$$
\mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = \mathbf{B_2}\widetilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{b_2}, \tag{11}
$$

where

$$
\mathbf{B_2} = \begin{bmatrix} \lambda_{1,1}(u_1)^{-1}\widetilde{\lambda}_{1,1}(u_1) & & \\ & \ddots & \\ & & \lambda_{n,2}(u_n)\widetilde{\lambda}_{n,2}(u_n) \end{bmatrix}. \tag{12}
$$

We replace $\mathbf{f}^{-1}$ with $\mathbf{Ch}$ and we could get the equations as below:

$$\mathbf{LT}(\mathbf{Ch}(\mathbf{x})) = \widetilde{\mathbf{L}}\widetilde{\mathbf{T}}(\tilde{\mathbf{C}}\tilde{\mathbf{h}}(\mathbf{x})) \Rightarrow \mathbf{T}(\mathbf{h}(\mathbf{x})) = \mathbf{B_1}\widetilde{\mathbf{T}}(\widetilde{\mathbf{h}}(\mathbf{x})) + \mathbf{b_1}, \tag{13}$$

where $\mathbf{B_3} = \mathbf{C}\tilde{\mathbf{C}}^{-1}$ is invertible matrix which corresponds to $\mathbf{C}$ and $\mathbf{B_1} = \mathbf{L}^{-1}\mathbf{B_3}^{-1}\widetilde{\mathbf{L}}$. The definition of $\widetilde{\mathbf{L}}$ on learning model migrates the definition of $\mathbf{L}$ on ground truth.

Then we adopt the definitions following [11]. According to the Lemma 3 in [11], we are able to pick out a pair $(\epsilon_i, \epsilon_i^2)$ such that, $(\mathbf{T}_i'(z_i), \mathbf{T}_i'(z_i^2))$ are linearly independent. Then concat the two points into a vector, and denote the Jacobian matrix $\mathbf{Q} = [J_{\mathbf{T}}(\boldsymbol{\epsilon}), J_{\mathbf{T}}(\boldsymbol{\epsilon}^2)]$, and define $\tilde{\mathbf{Q}}$ on $\tilde{\mathbf{T}}(\tilde{\mathbf{h}} \circ \mathbf{Cf}(\boldsymbol{\epsilon}))$ in the same manner. By differentiating Eq. 13, we get

$$\mathbf{Q} = \mathbf{B_1}\tilde{\mathbf{Q}}. \tag{14}$$

Since the assumptiom (2) that Jacobian of $\mathbf{f}^{-1}$ is full rank holds, it can prove that both $\mathbf{Q}$ and $\tilde{\mathbf{Q}}$ are invertible matrix. Thus from Eq. 14, $\mathbf{B_1}$ is invertible matrix. Using the same way as shown in Eq. 14, it can prove that $\mathbf{B_2}$ is invertible matrix.

Eq. 11 and Eq. 13 both hold. Combining the two results supports the identifiability result in CausalVAE.

B.2. Extension of Definition 1

In most of scenarios, latent variable is a low dimensional representation of the observation, since we are not interested in all the information in observations.

Therefore, we usually have $d > n$. We called it the reduction of dimension. We add auxiliary term as $\lambda(\mathbf{x}) = \{\lambda(\mathbf{u}), \lambda'\}$ In our model, Only n components of the latent variable are modulated, and its density has the form:

$$p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{u}) = \frac{\mathbf{Q}(\mathbf{z})}{\mathbf{Z}(\mathbf{u})} \exp \sum_{i}^{n} \mathbf{T_i}(z_i)\lambda_i(u_i) \tag{15}$$

and the term $e^{\sum_{n+1}^{d} \mathbf{T}(z_i)\lambda_i}$ is simply absorbed into $\mathbf{Q}(\mathbf{z})$. When we evaluate Eq. 6 by new definition (Eq. 15), the dimension of $p(\mathbf{z}|\mathbf{u})$ is n, because the remaining part is cancelled out.

Assume that $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{u})$ equal to $p_{\tilde{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{u})$. For all the observational pairs $(\mathbf{x}, \mathbf{u})$, let J_h denote the Jacobian matrix of the encoder function. Following the definition in Theorem 2 in i VAE [11], $\mathbf{B}$ will be indexed by 4 indicates (i, l, a, b), where $1 < i < d$ and $1 < l < s$ refer to the rows and $1 < a < d$ and $1 < b < s$ refer to the columns. We define a following equation:

$$\mathbf{v} = \tilde{\mathbf{C}} \circ \tilde{\mathbf{h}} \circ \mathbf{f}(\mathbf{z}). \tag{16}$$

The goal is to show that $v_i(\mathbf{z})$ is a function of only one z_j. We denote by $v_i^r := \frac{\partial v_i}{\partial z_r}$ and $v_i^{rt} := \frac{\partial^2 v_i}{\partial z_r \partial z_t}$. By differentiating Eq. 11 with respect to z_s, we could get:

$$T_{i,l}'(z_i) = \sum_{a=1}^{d} \sum_{b=1}^{s} B_{2,(i,l,a,b)} \tilde{T}_{a,b}'(v_a(\mathbf{z}))v_a^r(\mathbf{z}). \tag{17}$$

Lemma 1 *(from Lemma 9 in Khemakhem* et al. *[?]): Consider a distribution that follows a strongly exponential family. Its sufficient statistic $\tilde{\mathbf{T}}$ is differentiable almost surely. Then $\tilde{T}_i' \neq 0$ almost everywhere on $\mathbb{R}$ for all $1 \leq i \leq s$.*

For $r > n$, $T_{i,l}'(z_i) = 0$, according to Lemma 1, $\tilde{T}_{a,b}'(v_a(\mathbf{z})) \neq 0$, since $\mathbf{B}_2$ is an invertible matrix, we can conclude that $v_a^r(\mathbf{z}) = 0$ for all $a < n$ and $r > n$. Therefore, we can conclude that each of the first n components of $\mathbf{v}$ is only a function of one different z_j. Thus, when $d > n$, we could get the same conclusion as Theorem 1.

B.3. Identifiability of Causal Graph

Consider the identifiability analysis in Appendix B.1. For the framework of CausalVAE, in Causal Layer, the latent code $\mathbf{z}$ is identified since $\mathbf{B_2}$ is a diagonal matrix which corresponds to learnt $\tilde{\mathbf{z}}$ and $\mathbf{z}$. Since the true $\boldsymbol{\epsilon}$ and learnt $\tilde{\boldsymbol{\epsilon}}$ are linearly related, $\mathbf{B_1}$, $\mathbf{C}$ and $\tilde{\mathbf{C}}$ are in a linear equivalent class. In other words, $\mathbf{C}$ or $\mathbf{A}$ is identifiable in Causal Layer up to a linear equivalent class.

In our work, strict identifiability is guaranteed by the non-linear mask layer. Details of the Mask Layer are shown in Section 3.2 in main text. The Mask Layer uses non-linear functions and additional supervision signal $\mathbf{u}$ (non-Gaussian) to help the model to identify the true causal graph in a linear equivalent class.

C. Implementation Details

We use one NVIDIA Tesla P40 GPU as our training and inference device.

For the implementation of CausalVAE and other baselines, we extend $\mathbf{z}$ to matrix $\mathbf{z} \in \mathbb{R}^{n \times k}$ where n is the number of concepts and k is the latent dimension of each $\mathbf{z}_i$. The corresponding prior or conditional prior distributions of CausalVAE and other baselines are also adjusted (this means that we extend the multivariate Gaussian to the matrix Gaussian).

The subdimensions k for each synthetic (pendulum, water) experiments are set to be 4, and 32 for CelebA experiments. The implementation of continuous DAG constraint $H(\mathbf{A})$ follows the code of [32] [1].

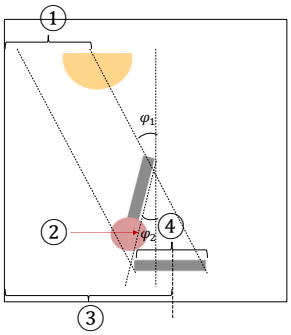


Figure 1. Generate Policy of Pendulum Simulator

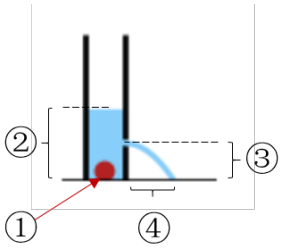


Figure 2. Generate Policy of Flow Simulator

C.1. Data Preprocessing

C.1.1 Sythetic Simulator

Fig. 1 shows our policy of generating synthetic Pendulum data. The picture includes a pendulum. The angles of pendulum and the light are changing overtime, and projection laws are used to generate the shadows. Given the light POSITION and pendulum ANGLE, we get the angles φ_1 and φ_2. Then the system can calculate the shadow POSITION and LENGTH using triangular functions. The causal graph of concepts is shown in Fig. 4 (a). In Pendulum generator, the image size is set to be 96×96 with 4 channels. We generate about 7k images (6k for training and 1k for inference), φ_1 and φ_2 are ranged in around $[-\frac{\pi}{4}, \frac{\pi}{4}]$, and they are generated independently. For each image, we provide 4 labels, which include light position, pendulum angle, shadow position and shadow length. For light position, we use the value of center of semicircle (Fig.1 ①) as supervision signal. For the pendulum angle, we use the value of ϕ_2 as supervision signal (Fig. 1 ②). For shadow position and shadow length, we use the length of Fig.1 ③ and Fig.1 ④ as supervision signal respectively.

Fig. 2 presents our policy of generating synthetic Flow data. Each image is of the $96 \times 96 \times 4$ resolution, and consists of a cup of water and a ball. The original water level, the ball size (Fig.2①) and the location of hole (Fig.2③) vary over time. Given the ball size Fig. 2 and the original water level, we determine the WATER HEIGHT (Fig.2②). Then we generate WATER FLOW according to the Parabola law, where we additionally introduce a noise from $\mathcal{N}(0, 0.01)$ to the gravitational acceleration. The causal graph of concepts is given in Fig. 4 (b). We consider four semantically meaningful concepts, BALLS SIZE, WATER HEIGHT, HOLE POSITION and WATER FLOW, whose supervised signals are given by the ball's diameter (Fig.1 ①), the length of Fig. 1 ②, the length of Fig.1 ③ and Fig.1 ④ respectively. The sample size is 8k with 6k for training and 2k for testing.

C.1.2 Data Preprocess of CelebA

CelebA dataset contains 20K human face images. We preprocess the original dataset by following two steps:

(1) We divided the whole dataset into training dataset 85% and test dataset 15%.

(2) We only focus on facial features and resize the picture to be squared (128×128 with 3 channels).

C.2. Intervention Experiments

C.2.1 Synthetic

In synthetic experiments, we train the model on synthetic data for 80 epochs, and use this model to generate latent code of representations. The hyperparameters of baselines are defined as default.

For CausalVAE, we set the $\alpha = 0.3$ and $(\beta, \gamma) = (1, 1)$. We use $\mathcal{N}(\mathbf{u}, |\mathbf{u}|)$ as the condition prior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{u})$. In the implementation of CausalVAE, $|\mathbf{z}_{\text{mean}}|$ is used as the variance of condition prior.

The details of the neural networks are shown in Table 1. We all follows the neural network design strategy of Khemakhem *et al.* [**?**] to satisfy Theorem 1 assumption (ii).

C.2.2 CelebA

We also present the DO-experiments of CausalVAE and CausalGAN. In the training of the models, we use face labels (AGE, GENDER and BEARD).

For CausalVAE, we set the $\alpha = 0.3$ and $(\beta, \gamma) = (1, 1)$. We use $\mathcal{N}(\mathbf{u}, \mathbf{I})$ as the condition prior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{u})$. For all the baseline, default hyperparameters and one common encoder and decoder structure are employed. For CausalGAN, we use the publicly available code[2].

For all the VAE-based methods, mean and variance of the distribution of the latent variable are learned during training, and the latent code z are sampled from Conditional Gaussian Distribution $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{u})$. In all experiments, we rescale the

[1]https://github.com/fishmoon1234/DAG-GNN

[2]https://github.com/mkocaoglu/CausalGAN

variance of learned representation $\mathbf{z}$ by multiplying a factor 0.1 to the original one.

Training epoches for the model is set to be 80, and our proposed CausalVAE has a pretrain step to learn causal graph $\mathbf{A}$, which takes 10 epochs.

The details of the neural networks are shown in Table 2.

C.3. The Pretrain Step for Causal Graph Learning

In our model, we need to learn the latent representation $\mathbf{z}$ and causal graph $\mathbf{A}$ simultaneously, whose optimal solution is not easy to find. Thus we adopt a pretrain stage to learn the causal graph $\mathbf{A}$ in the Mask Layer. We adopt the augmented Lagrangian to learn $\mathbf{A}$ in CausalVAE from the labels $\mathbf{u}$ in Mask Layer first. During the pretrain process, we truncate the gradient of other part of model and solve the optimization problem in Eq. 19 to learn $\mathbf{A}$.

The augmentation approach is widely used in causal discovery method, like NOTEARS [28], DAG-GNN [32]. The pretrain is a stage that learns the graph by optimizing the following objective functions:

$$\begin{aligned} &\text{minimize} \quad l_u = \mathbb{E}_{q_D} \|\mathbf{u} - \mathbf{A}^T \mathbf{u}\|_2^2 \\ &\text{subject to} \quad H(\mathbf{A}) = 0 \end{aligned} \tag{18}$$

Then, we define an augmented Lagrangian:

$$l_{pre} = l_u + \lambda H(\mathbf{A}) + \frac{c}{2} H^2(\mathbf{A}) \tag{19}$$

where λ is the Lagrangian multiplier and c is the penalty.

The following policy is used to update the λ and c:

$$\lambda_{s+1} = \lambda_s + c_s H(\mathbf{A}_s) \tag{20}$$

$$c_{s+1} = \begin{cases} c_s = \eta c_s, & if |H(\mathbf{A}_s)| > \gamma |H(\mathbf{A}_{s-1})| \\ c_s = c_s, & otherwise \end{cases}$$

where s is the iteration. In our experiments, we set $\eta = 10$ and $\gamma = \frac{1}{4}$.

D. Additional Experimental Results

In this section, we show more experimental results. Fig. 4 shows the causal graph among concepts in different dataset respectively. We here show results including experiments analyzing the properties of learned representation, intervening results and the learning process of the causal graph.

D.1. The Property of Learned Representation

We test our method and baselines on both synthetic data and benchmark human face data. In the previous section, we already show the relationships between the learned representation $\tilde{\mathbf{z}}$ and the target representation $\mathbf{z}$ (related by a linear transformation formed as a diagonal matrix). In this section, we visualize it by scatter plot.

One of the important aspect of the generative model is that whether the learned representation aligns to the conditional prior we set. Our conditional prior is generated by the true label of each concept. The results show that the learned representations align to the expected representations. In figures, points are sampled from the joint distribution, and each color corresponds to one dimension.

The additional observations (labels) of Pendulum dataset and those of CelebA dataset are different. In Pendulum, the labels are values within a fixed range The labels in CelebA dataset are discrete (in $\{-1, 1\}$). Thus the scatter plots are different.

The results show that the performance of our proposed method is better than all the baselines, including the supervised method and unsupervised method.

D.2. The Learned Graph

We demonstrate the learning process of causal graph in this section. Fig. 8 shows the graph learned process of CelebA (BEARD). In this process, we initialize all the entries in $\mathbf{A}$ as 0.5. After 5 epochs, the graph converges. We observe an almost correct graph in this group of concepts.

D.3. Intervention Results

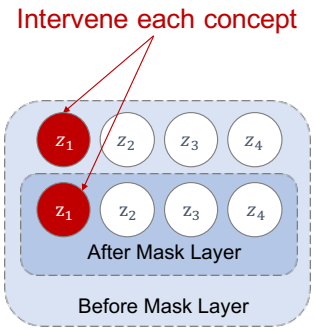


Figure 10. Intervention method

The intervention operations are as:

- For the learned model, we first put an random observed image $\mathbf{x}$ into the encoder. In this process we could get $\boldsymbol{\epsilon}$ and $\mathbf{z}$.
- Then for i-th concept, we fix the value of z_i and $g_i(\mathbf{A_i} \circ \mathbf{z})$ as constants.
- Finally, we put the new $\mathbf{z}$ into the decoder and get $\mathbf{x}'$.

Fig. 3 (a) demonstrates the intervention results of CausalVAE on Flow dataset. We see that when we intervene on the cause concept BALL SIZE, its child concepts WATER HEIGHT and WATER FLOW change correspondingly. Similarly, when the cause concept HOLE is intervened, its child

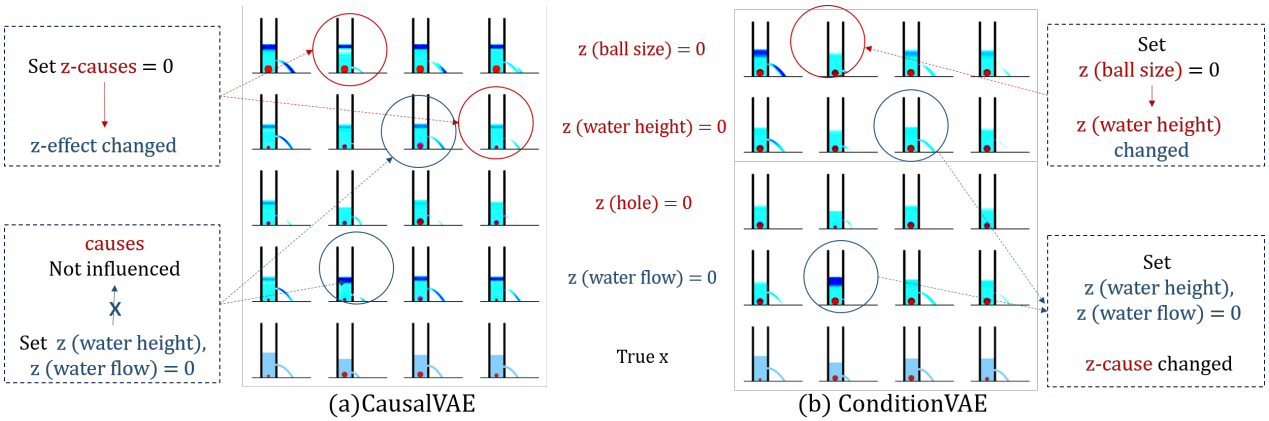


Figure 3. The results of Intervention experiments on the Flow dataset. Each row shows the result of controlling the BALL SIZE, WATER HEIGHT, HOLE, and WATER FLOW respectively. The bottom row is the original input image.

encoder	decoder
4*96*96×900 fc. 1ELU	concepts×(4× 300 fc. 1ELU)
900×300 fc. 1ELU	concepts× (300×300 fc. 1ELU)
300×2*concepts*k fc.	concepts×(300× 1024 fc. 1ELU)
-	concepts×(1024× 4*96*96 fc.)

Table 1. Network design of models trained on synthetic data.

concept WATER FLOW also changes. In contrast, intervening on effect concept WATER HEIGHT does not influence the causal concept BALL SIZE. Fig. 3(b) shows the results of ConditionVAE on Flow. We observe that when we intervene on BALL SIZE, WATER HEIGHT and WATER FLOW are affected as expected. However when we intervene on the effect concepts WATER HEIGHT and WATER FLOW, concept BALL SIZE is also influenced, which makes no sense. In general, the "do-intervention" of ConditionVAE performs worse than CausalVAE. The results support that by simply using a supervised model, one can not guarantee a causal disentangled representation.

The Fig. 11 demonstrates the result of CausalVAE on real world banchmark dataset CelebA (BEARD), with subfigures (a) (b) (c) (d) showing the intervention experiments on concepts of AGE, GENDER, BALD and BEARD respectively. The interventions perform well that when we intervened the cause concept GENDER, the BEARD changes correspondingly. Similarly, when the cause concept AGE in intervened, its child concept BALD also changes. In contrast, intervening effect concept BEARD does not influence the causal concepts GENDER and other unrelated concepts in Fig. 11 (d). Fig. 12 demonstrates the results of CausalGAN, with subfigures (a) (b) (c) (d) showing the intervention experiments on concepts CelebA (BEARD). We observe that when we intervene GENDER, the BEARD are changed. But when we intervene BEARD, concept GENDER is also changed in third line as shown by Fig. 12 (d). In general, the 'do-intervention' of CausalGAN performs worse than CausalVAE.

The Fig. 13 demonstrates the result of CausalVAE on real world banchmark dataset CelebA (SMILE), with subfigures (a) (b) (c) (d) showing the intervention experiments on concepts of GENDER, SMILE, MOUTH OPEN and EYES OPEN respectively. The interventions perform well that when we intervened the cause concept GENDER, not only the appearance of GENDER but the eyes changed. When we intervened the cause concept SMILE, not only the appearance of SMILE but the MOUTH OPEN. In contrast, intervening effect concept MOUTH OPEN does not influence the causal concepts SMILE in Fig. 13 (d). Fig. 14 demonstrates the results of CausalGAN, with subfigures (a) (b) (c) (d) showing the intervention experiments on concepts CelebA (SMILE). We find that when we control SMILE, the mouth is changed, as shown in the second line of Fig. 14 (b). But we find sometimes the control of SMILE influence other unrelated concepts like GENDER (shown in first line of Fig. 14 (b)). In this concepts group, CausalGAN also shows relatively unstable intervention experiments compared to that of ours.

encoder	decoder
-	(1×1 conv. 128 1LReLU(0.2), stride 1)
4×4 conv. 32 1LReLU (0.2), stride 2	(4×4 convtranspose. 64 1LReLU (0.2), stride 1)
4×4 conv. 64 1LReLU (0.2), stride 2	(4×4 convtranspose. 64 1LReLU (0.2), stride 2)
4×4 conv. 64 1LReLU(0.2), stride 2	(4×4 convtranspose. 32 1LReLU (0.2), stride 2)
4×4 conv. 64 1LReLU (0.2), stride 2	(4×4 convtranspose. 32 1LReLU (0.2), stride 2)
4×4 conv. 256 1LReLU (0.2), stride 2	(4×4 convtranspose. 32 1LReLU (0.2), stride 2)
1×1 conv. 3, stride 1	(4×4 convtranspose. 3 , stride 2)

Table 2. Network design of models trained on CelebA.

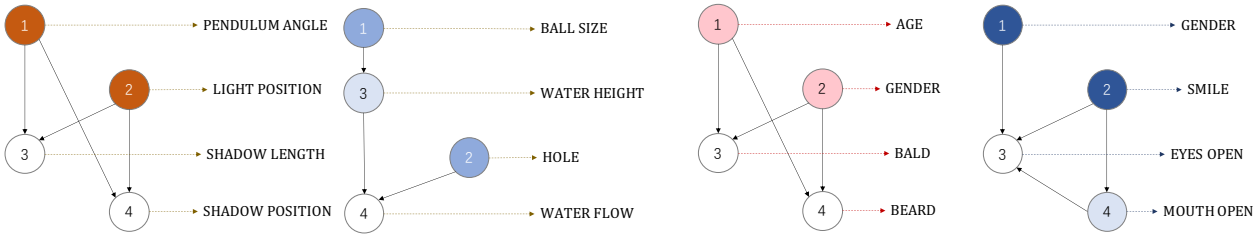


Figure 4. Causal graphs of three dataset. (a) shows the causal graph in pendulum dataset. The concepts are PENDULUM ANGLE, light POSITION, SHADOW POSITION and SHADOW LENGTH. (b) shows the causal graph in CelebA, on concepts AGE, GENDER and BEARD and BALD. (c) shows the causal graph in CelebA, on concepts GENDER, SMILE, EYES OPEN and MOUTH OPEN.

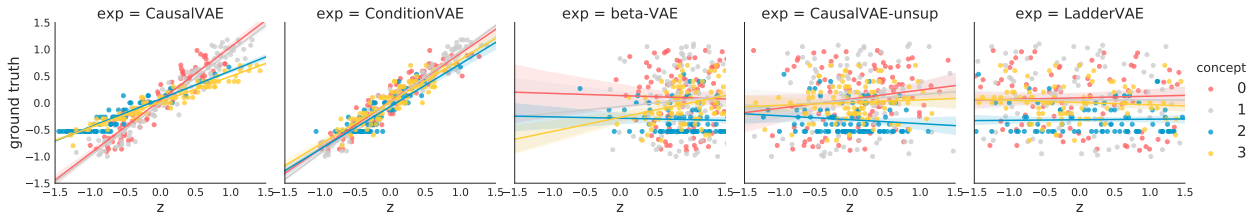


Figure 5. The figure shows the alignment of ground truth $p(\mathbf{z}|\mathbf{u})$ and the learned latent factors $q(\mathbf{z}|\mathbf{x},\mathbf{u})$ on pendulum experiments. Although ConditionVAE is also the supervised method, our proposed CausalVAE shows a better performance.

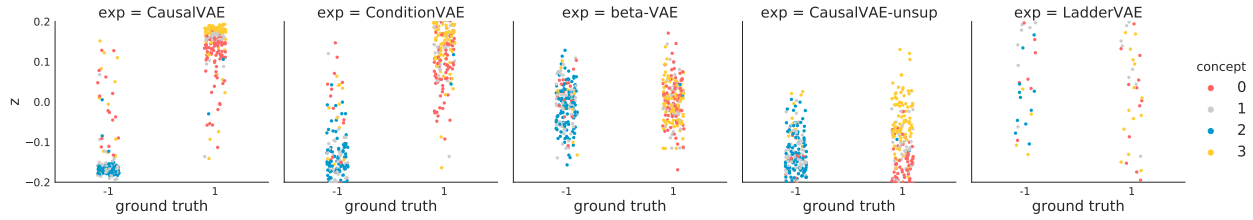


Figure 6. The figure shows the alignment of ground truth $p(\mathbf{z}|\mathbf{u})$ and the learned latent factors $q(\mathbf{z}|\mathbf{x},\mathbf{u})$ on CelebA for the concepts (BEARD). The ground truth is a discrete distribution over $\{-1, 1\}$, and the color of the points indicates different dimensions. The factors learned by CausalVAE show the best alignment among all.

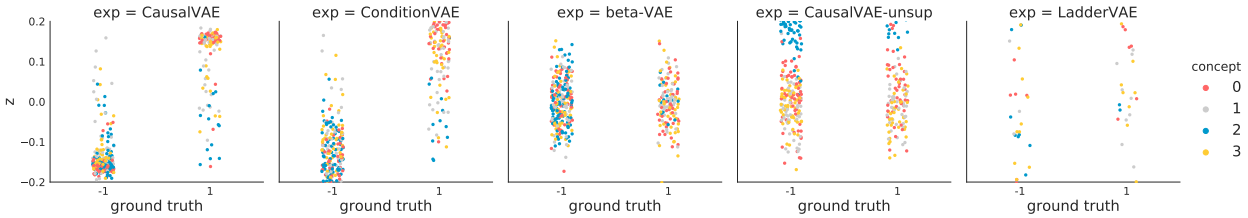
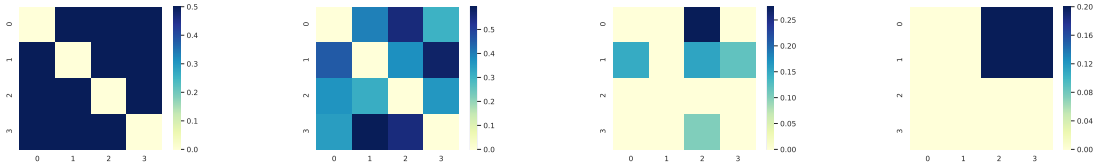
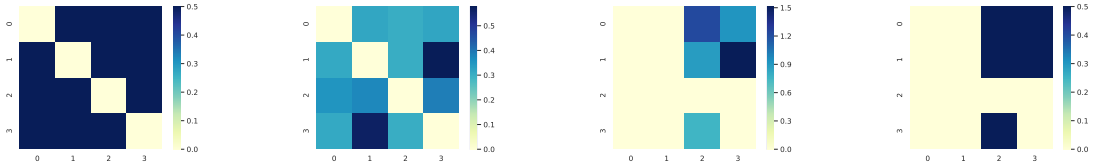


Figure 7. The figure shows the alignment between ground truth $p(\mathbf{z}|\mathbf{u})$ and the learned latent factors $q(\mathbf{z}|\mathbf{x},\mathbf{u})$ on CelebA for 5 methods (CausalVAE, ConditionVAE, β-VAE, CausalVAE-unsup, LadderVAE from left to right). The ground truth is a distribution with mean taken from $\{-1, 1\}$, and the color of the points indicates different dimensions. The factors learned by CausalVAE show the best alignment among all. The concepts include: 1 GENDER; 2 SMILE; 3 EYES OPEN; 4 MOUTH OPEN.



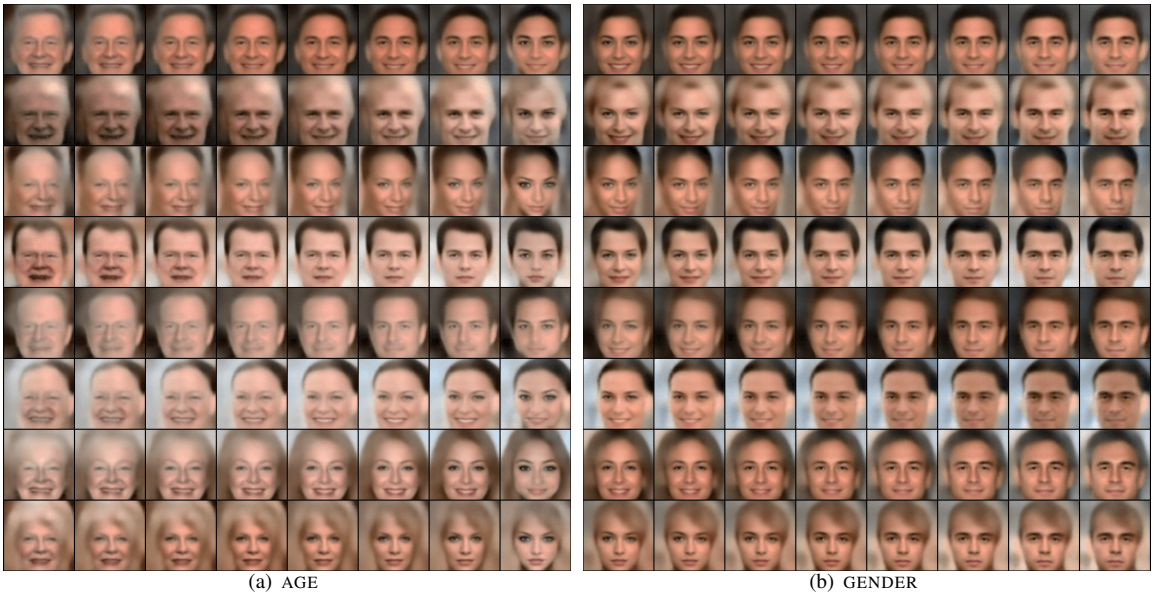
(a) Initialize (b) After 1 epoch (c) After 5 epochs (d) The true

Figure 8. Learning process of causal graph $\mathbf{A}$ in CelebA (BEARD). The concepts include: 1 AGE; 2 GENDER; 3 BALD; 4 BEARD.



(a) Initialize (b) After 1 epoch (c) After 5 epochs (d) The true

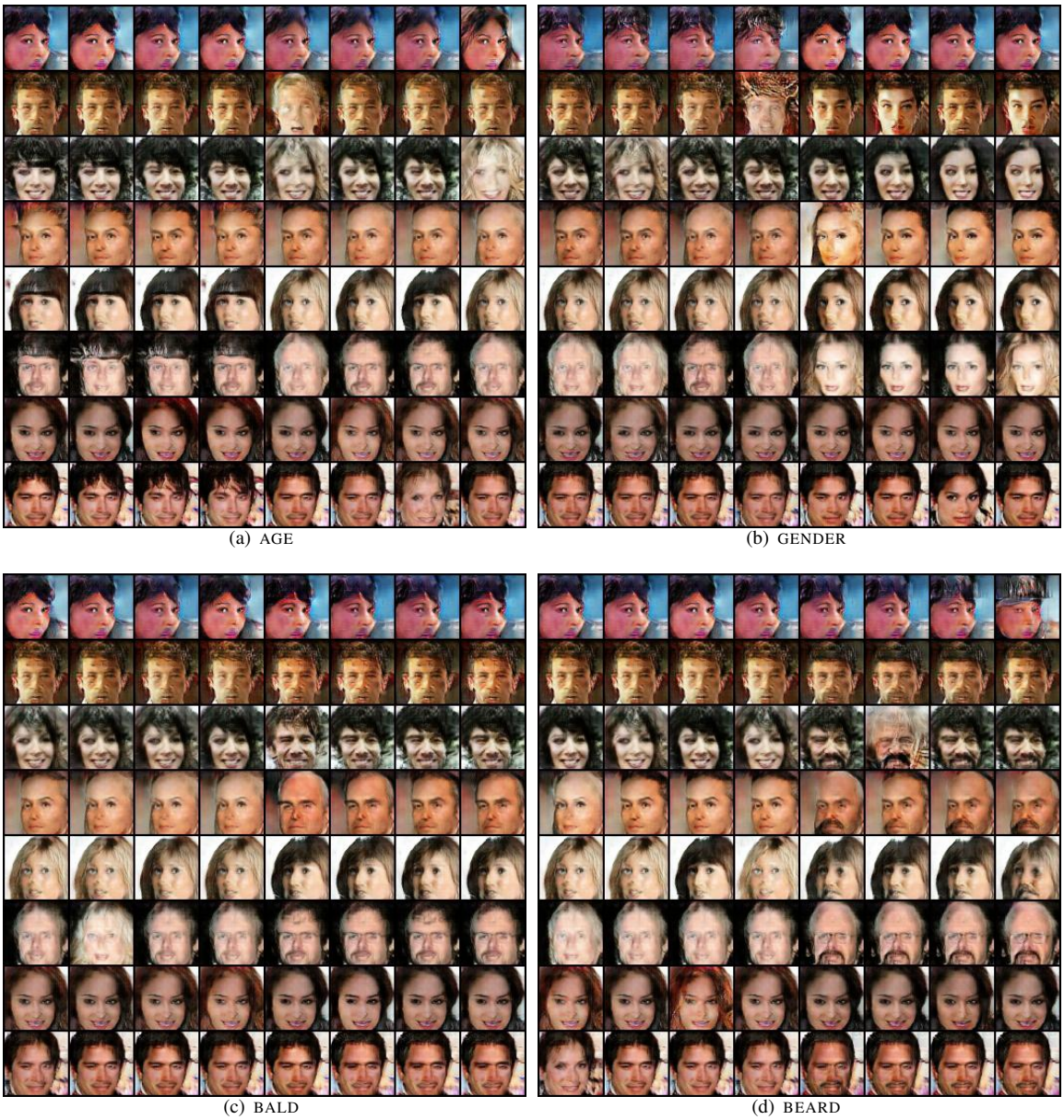
Figure 9. Learning process of causal graph $\mathbf{A}$ in CelebA (SMILE). The concepts include: 1 GENDER; 2 SMILE; 3 EYES OPEN; 4 MOUTH OPEN.



(a) AGE (b) GENDER

(c) BALD (d) BEARD

Figure 11. Results of CausalVAE model on CelebA (BEARD). The captions of the subfigures describe the controlled factors. From left to right, the pictures are results obtained by varying the value of the controlled factors.



(a) AGE (b) GENDER

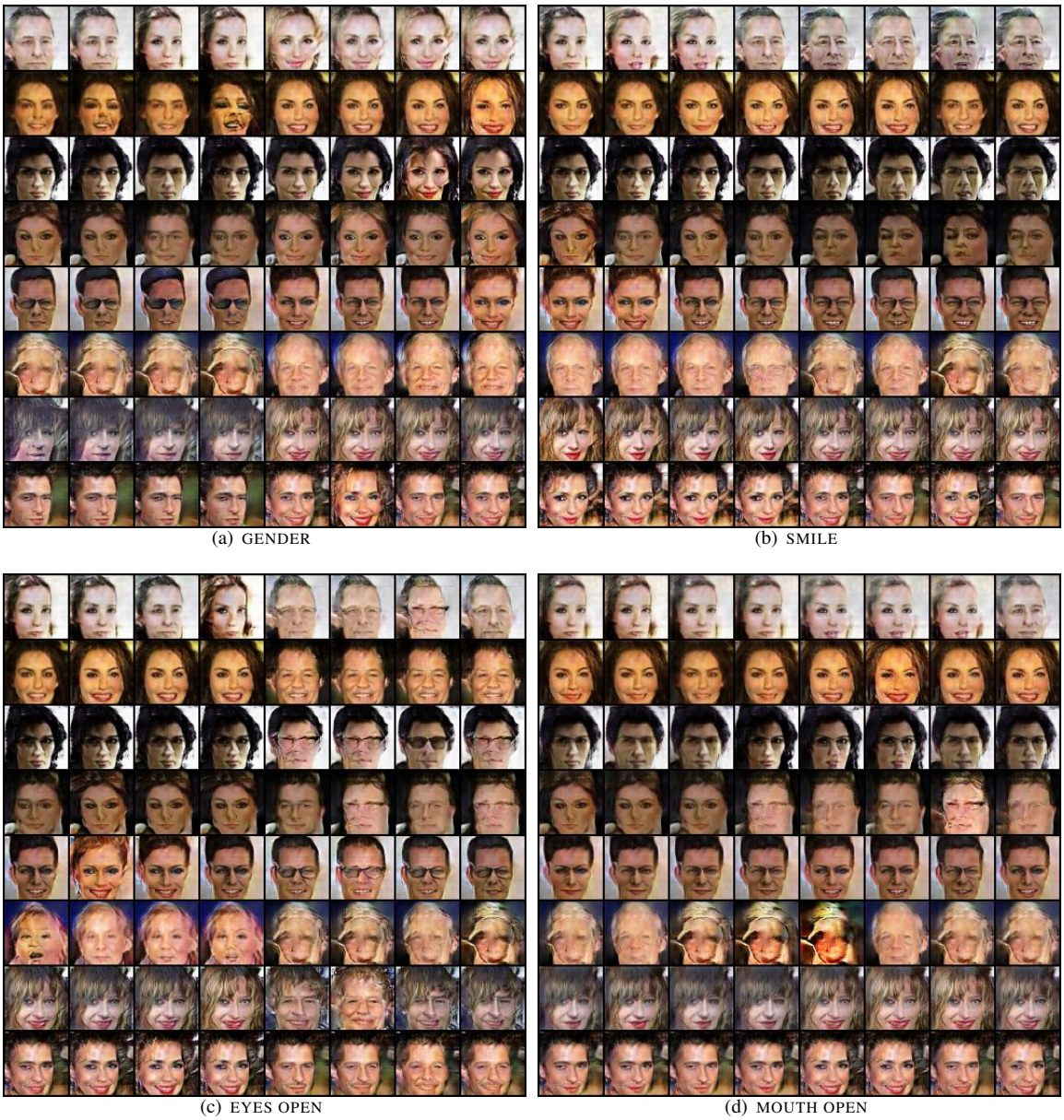
(c) BALD (d) BEARD

Figure 12. Results of CausalGAN [14] model on CelebA (BEARD). The captions of the subfigures describe the controlled factors. From left to right, the pictures are results obtained by varying the value of the controlled factors.

(a) GENDER (b) SMILE

(c) EYES OPEN (d) MOUTH OPEN

Figure 13. Results of CausalVAE model on CelebA (SMILE). The captions of the subfigures describe the controlled factors. From left to right, the pictures are results obtained by varying the value of the controlled factors.



(a) GENDER (b) SMILE

(c) EYES OPEN (d) MOUTH OPEN

Figure 14. Results of CausalGAN model on CelebA (SMILE). The captions of the subfigures describe the controlled factors. From left to right, the pictures are results obtained by varying the value of the controlled factors.

References

[1] Michel Besserve, Rémy Sun, and Bernhard Schölkopf. Counterfactuals uncover the modular structure of deep generative models. *arXiv preprint arXiv:1812.03253*, 2018.

[2] Philemon Brakel and Yoshua Bengio. Learning independent features with adversarial nets for non-linear ica. *arXiv preprint arXiv:1710.05050*, 2017.

[3] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in $beta$-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[4] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2610–2620, 2018.

[5] David Heckerman, Dan Geiger, and David Maxwell Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *CoRR*, abs/1302.6815, 2013.

[6] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *Iclr*, 2(5):6, 2017.

[7] Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in neural information processing systems*, pages 689–696, 2009.

[8] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li F Fei-Fei, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *Advances in Neural Information Processing Systems*, pages 517–526, 2018.

[9] Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in neural information processing systems*, pages 1878–1889, 2017.

[10] Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stefan Bauer, Hugo Larochelle, Chris Pal, and Yoshua Bengio. Learning neural causal models from unknown interventions. *CoRR*, abs/1910.01075, 2019.

[11] Ilyes Khemakhem, Diederik P. Kingma, and Aapo Hyvärinen. Variational autoencoders and nonlinear ICA: A unifying framework. *CoRR*, abs/1907.04809, 2019. 1, 3

[12] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. *arXiv preprint arXiv:1802.05983*, 2018.

[13] Justin B Kinney and Gurinder S Atwal. Equitability, mutual information, and the maximal information coefficient. *Proceedings of the National Academy of Sciences*, 111(9):3354–3359, 2014.

[14] Murat Kocaoglu, Christopher Snyder, Alexandros G. Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. *CoRR*, abs/1709.02023, 2017. 10

[15] Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *Advances in neural information processing systems*, pages 2539–2547, 2015.

[16] Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors. *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 2016.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 3730–3738, 2015.

[18] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in

the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359*, 2018.

[19] Francesco Locatello, Michael Tschannen, Stefan Bauer, Gunnar Rätsch, Bernhard Schölkopf, and Olivier Bachem. Disentangling factors of variation using few labels. *arXiv preprint arXiv:1905.01258*, 2019.

[20] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. Learning disentangled representations for recommendation. In *Advances in Neural Information Processing Systems*, pages 5712–5723, 2019.

[21] Emile Mathieu, Tom Rainforth, N Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. *arXiv preprint arXiv:1812.02833*, 2018.

[22] Ignavier Ng, Zhuangyan Fang, Shengyu Zhu, and Zhitang Chen. Masked gradient-based causal structure learning. *arXiv preprint arXiv:1910.08527*, 2019.

[23] Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *CoRR*, abs/1911.07420, 2019.

[24] Judea Pearl. *Causality*. Cambridge university press, 2009.

[25] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference using invariant prediction: identification and confidence intervals. *arXiv preprint arXiv:1501.01332*, 2015.

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

[27] Bernhard Schölkopf. Causality for machine learning. *arXiv preprint arXiv:1911.10500*, 2019.

[28] Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(Oct):2003–2030, 2006. 5

[29] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.

[30] Raphael Suter, Dorde Miladinović, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. *arXiv preprint arXiv:1811.00007*, 2018.

[31] Robert E. Tillman and Peter Spirtes. Learning equivalence classes of acyclic models with latent and selection variables from multiple datasets with overlapping variables. In Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, volume 15 of *JMLR Proceedings*, pages 3–15. JMLR.org, 2011.

[32] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. Dag-gnn: Dag structure learning with graph neural networks. *arXiv preprint arXiv:1904.10098*, 2019. 3, 5

[33] Kun Zhang and Aapo Hyvarinen. On the identifiability of the post-nonlinear causal model. *arXiv preprint arXiv:1205.2599*, 2012.

[34] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, pages 9472–9483, 2018.

[35] Shengyu Zhu and Zhitang Chen. Causal discovery with reinforcement learning. *CoRR*, abs/1906.04477, 2019.

[36] Shengyu Zhu, Ignavier Ng, and Zhitang Chen. Causal discovery with reinforcement learning. In *International Conference on Learning Representations(ICLR)*, 2020.