Learning Disentangled Representation with Pairwise Independence

Zejian Li,[1] **Yongchuan Tang,**[1,2*] **Wei Li,**[1] **Yongxing He**[1]
[1]College of Computer Science, Zhejiang University, Hangzhou 310027, China
[2]Zhejiang Lab, Hangzhou 310027, China
{zejianlee, yctang, liwei_2014, heyongxing}@zju.edu.cn

Abstract

Unsupervised disentangled representation learning is one of the foundational methods to learn interpretable factors in the data. Existing learning methods are based on the assumption that disentangled factors are mutually independent and incorporate this assumption with the evidence lower bound. However, our experiment reveals that factors in real-world data tend to be pairwise independent. Accordingly, we propose a new method based on a pairwise independence assumption to learn the disentangled representation. The evidence lower bound implicitly encourages mutual independence of latent codes so it is too strong for our assumption. Therefore, we introduce another lower bound in our method. Extensive experiments show that our proposed method gives competitive performances as compared with other state-of-the-art methods.

1 Introduction

This paper is concerned with the unsupervised learning of disentangled representation. The disentangled representation is a distributed data representation in which latent codes represent interpretable attributes. Disjoint dimensions of the representation change independently in the variation of the data and are associated with different high-level data factors (Bengio, Courville, and Vincent 2013). One example of the disentangled representation is the task of generating hand-written digits. The hand-written digits are generated by the generator according to the latent codes, while different codes control rotation, stroke width, writing style and other different attributes. These attributes interact non-linearly in the data. However, when one factor varies but all others are fixed, the generated sequence of samples can show an interpretable change to human beings. Due to its interpretability, disentangled representations are useful in many downstream tasks such as supervised learning (Liu et al. 2018; Hadad, Wolf, and Shahar 2018) and transfer learning (Zamir et al. 2018).

Many recent works have been devoted to the supervised learning of disentangled representation. Bouchacourt, Tomioka, and Nowozin (2018) and Hadad, Wolf, and Shahar (2018) assume the group division of samples is given. Liu et al. (2018) require the predefined attributes. Adel, Ghahramani, and Weller (2018) consider side information in the learning process. However, real-world data is often raw data without labels or attributes, and thus the unsupervised learning of disentangled representation is an important and challenging problem. Most existing methods are based on the prior assumption that the learned codes should be mutually independent. It is believed that interpretable factors tend to change independently in the data, so by inferring independent codes, the model may capture those interpretable factors backward. To model this independence, Higgins et al. (2017) and Burgess et al. (2018) limit the capacity of learning model. Kumar, Sattigeri, and Balakrishnan (2018) match the code distribution to the standard normal distribution. Kim and Mnih (2018) and Chen et al. (2018) optimize the term of total correlation to enable the distribution to be factorial. Other works (Chen et al. 2016; Li, Tang, and He 2018) take the principle of mutual information minimization. Most of these methods are built on top of variational autoencoder (Kingma and Welling 2014).

However, we find interpretable factors are pairwise independent in experiments. We perform Pearson's chi-squared test on the CelebA attributes (Liu et al. 2015) and find that some attributes pairs are independent. However, only a group of three attributes is three-wise independent and no four-wise independent group is observed. Therefore, we assume the latent codes of data are pairwise independence in the design of our model. Notice that pairwise independence is different from mutual independence. A finite set of k random variables $\{Z_1, \ldots, Z_k\}$ are pairwise independent when any two of them are independent. However, they are mutually independent only when the joint cumulative distribution function is always the product of the marginal cumulative functions, namely $F_{Z_1,\ldots,Z_k}(z_1, \ldots, z_k) = \prod_{i=1}^{k} F_{Z_i}(z_i)$. Since mutual independence is a special case of pairwise independence, our assumption is more general.

The pairwise independence assumption cannot be incorporated with variational autoencoder directly. This is because variational autoencoder is based on the evidence lower bound, which implicitly encourages mutual independence among codes (Hoffman and Johnson 2016). We introduce another lower bound of log-likelihood according to the principle of variational inference. Similar to the evidence lower bound, it enables the model to recover the sample given the inferred code. However, it restricts the marginal distribution instead

*Corresponding author

of the joint distribution of the latent code. As a result, it can incorporate the pairwise independence assumption which constrains the joint distribution between code pairs. Finally, the discussed lower bound is combined with a designed pairwise independence term. Inspired by (Kingma and Welling 2014), our model is implemented with deep neural networks and trained with the stochastic optimization method and the reparameterization trick. Figure 1 shows an illustration of our model.

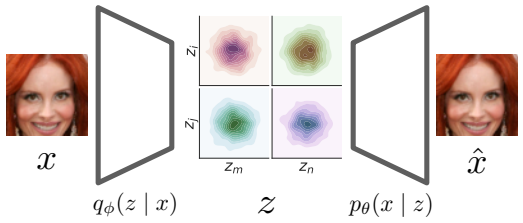

Figure 1: **The architecture of our proposed method.** Given a sample x, the encoder $q_\phi(z \mid x)$ infers the code z and the decoder $p_\theta(x \mid z)$ recovers $\hat{x}$ accordingly. The aggregated posterior distribution $q_\phi(z)$ is encouraged to be pairwise independent. To illustrate the idea, we visualize the pairwise joint distributions of $q_\phi(z)$. Here z_i, z_j, z_m and z_n are different code components. Ideally we have $q_\phi(z_i, z_m) = q_\phi(z_i)q_\phi(z_m)$ and the same holds for other code pairs. The notations are summarized in Table 1. The figure is best viewed magnified on screen.

An outline of the remainder of our paper is as follows. Section 2 gives a brief review of related works. Section 3 describes our experiments on CelebA attributes and shows the attributes tend to be pairwise independent. Based on this observation, Section 4 introduces our proposed model, which combines our discussed lower bound and the designed term to measure pairwise independence. In Section 5, we perform canonical correlation analysis between the CelebA attributes and the learned codes of different models to show how well the methods capture the attributes. Finally, Section 6 concludes our paper. Our source code is available on https://github.com/ZejianLi/Pairwise-Indepence-Autoencoder.

2 Related Works

In this part, we give a brief introduction of variational autoencoder (Kingma and Welling 2014) and its variants which learn disentangled representations.

Variational autoencoder (VAE) has been a foundational generative model to learn the latent representation. Given a k-dimensional latent code $z \in \mathcal{Z}$ sampled from a prior distribution $p(z)$, a new sample $x \in \mathcal{X}$ can be generated with $p_\theta(x \mid z)$. To increase the log-likelihood of the observed samples $\log p_\theta(x)$, VAE maximizes the evidence lower bound (ELBO) (Jordan et al. 1999) defined as:

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{p(x)}\mathbb{E}_{q_\phi(z|x)} \log \frac{p_\theta(x, z)}{q_\phi(z \mid x)} \leq \mathbb{E}_{p(x)} \log p_\theta(x). \tag{1}$$

The notations are summarized in detail in Table 1.

Table 1: Notations.

Notation	Definition
x	An observed sample from the data space $\mathcal{X}$.
z	A code in the k-dimensional latent space $\mathcal{Z}$.
$p(x)$	The ground-truth data distribution of x, assumed to be absolutely continuous.
$p(z)$	The prior distribution of z, assumed to be $\mathcal{N}(0, I)$.
$p_\theta(x \mid z)$	The distribution to generate a new sample x given z, parameterized by θ.
$p_\theta(x)$	The marginal distribution of $p_\theta(x, z) = p(z)p_\theta(x \mid z)$.
$q_\phi(z \mid x)$	The variational distribution of the posterior $p_\theta(z \mid x)$, parameterized by ϕ.
$q_\phi(z)$	The marginal distribution of $q_\phi(x, z) = p(x)q_\phi(z \mid x)$.
$\mathcal{B}$	A mini-batch of b samples $\{x_1, \ldots, x_b\}$.

VAE can disentangle factors by encouraging the latent codes to be independent (Hoffman and Johnson 2016). To see this, the ELBO is decomposed as:

$$\begin{aligned}&\mathbb{E}_{p(x)}\mathbb{E}_{q_\phi(z|x)} \log \frac{p_\theta(x, z)}{q_\phi(z \mid x)} \\ =&\mathbb{E}_{p(x)}\mathbb{E}_{q_\phi(z|x)} \log p_\theta(x \mid z) - \mathbb{E}_{q_\phi(z,x)} \log \frac{q_\phi(z \mid x)}{p(z)}.\end{aligned} \tag{2}$$

The first term is the expected log-likelihood to recover the sample x. The second term can be further decomposed as:

$$\begin{aligned}&\mathbb{E}_{q_\phi(z,x)} \log \frac{q_\phi(z \mid x)}{p(z)} \\ =&\mathbb{E}_{q_\phi(z,x)} \log \frac{q_\phi(z, x)}{q_\phi(z)p(x)} + \mathbb{E}_{q_\phi(z)} \log \frac{q_\phi(z)}{p(z)} \\ =&I_\phi(z; x) + \mathrm{KL}(q_\phi(z) \| p(z)).\end{aligned} \tag{3}$$

$I_\phi(z; x)$ is the mutual information between x and z specified by $q_\phi(z, x)$. $\mathrm{KL}(q_\phi(z) \| p(z))$ is the Kullback-Leibler divergence between $q_\phi(z)$ and $p(z)$. It guides $q_\phi(z)$ to be factorial and the marginal distributions of $q_\phi(z)$ to be Gaussian. To see this, $\mathrm{KL}(q_\phi(z) \| p(z))$ is decomposed as

$$\begin{aligned}&\mathrm{KL}(q_\phi(z) \| p(z)) \\ =&\mathbb{E}_{q_\phi(z)} \log \frac{q_\phi(z)}{\prod_{i=1}^k q_\phi(z_i)} + \sum_{i=1}^k \mathbb{E}_{q_\phi(z_i)} \log \frac{q_\phi(z_i)}{p(z_i)}.\end{aligned} \tag{4}$$

$\mathbb{E}_{q_\phi(z)} \log \frac{q_\phi(z)}{\prod_{i=1}^k q_\phi(z_i)}$ is the total correlation of the latent codes. Similar to mutual information, the total correlation is zero when $q_\phi(z_i)$ for $i = 1, \ldots, k$ are mutually independent. Therefore, VAE encourages the independence of latent codes and thus disentangles the generative factors. Recent works are mainly focused on putting more emphasis on the independence. Specifically, β-VAE (Higgins et al. 2017; Burgess et al. 2018) put more weight on $\mathbb{E}_{q_\phi(z,x)} \log \frac{q_\phi(z|x)}{p(z)}$ in (2) and thus penalize the total correlation term. FactorVAE (Kim and Mnih 2018) and β-TCVAE (Chen et

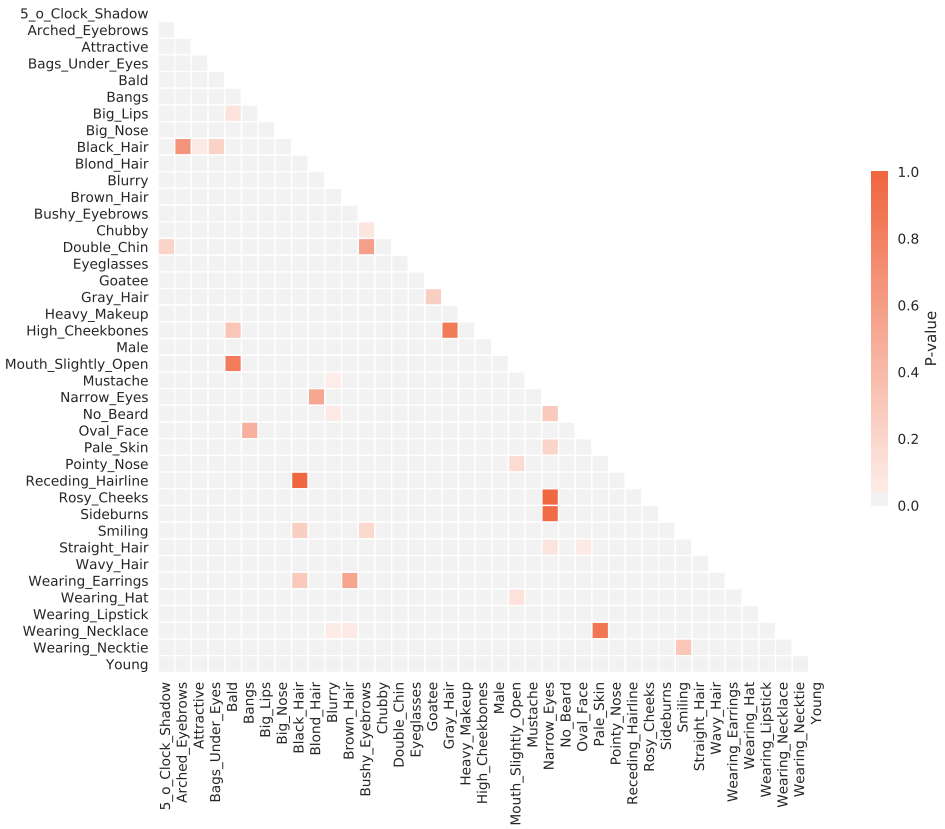

Figure 2: **The p-values in the Pearson's chi-squared test on the attribute pairs of CelebA dataset.** It is observed that 36 pairs are not significantly dependent with a significant level of 0.01. The figure is best viewed on screen.

al. 2018) augment the ELBO with the total correlation term directly. Similarly, DIP-VAE (Kumar, Sattigeri, and Balakrishnan 2018) designs an moment-matching term to minimize $\mathrm{KL}(q_\phi(z)\|p(z))$ in (3). The assumption behind VAE and these variants is that interpretable factors are mutually independent and can be captured with the factorial distribution $p(z)$. Other works (Chen et al. 2016; Li, Tang, and He 2018) encourage disentanglement by minimizing the mutual information between x and z with extra components of the model.

3 Experiment on CelebA Attributes

We tentatively argue that the interpretable factors may be pairwise independent, but not mutually independent. Our argument is supported by the following experiment findings.

We conduct the Pearson's chi-squared test on the labeled attributes of CelebA dataset (Liu et al. 2015). These 40 attributes are binary and concerned with different aspects of the faces. Notice that some attributes are intrinsically correlated, such as "brown hair" and "black hair", or "narrow eyes" and "smiling". Firstly, we perform the test on attribute pairs and find 36 pairs are not significantly dependent with a significant level of 0.01. The p-values of all pairwise tests are visualized in Figure 2. We also perform the test on groups of three and four attributes. Only the group of "Blond Hair", "Straight Hair" and "Narrow Eyes" is not significantly dependent with the p-value as 0.038, and all groups of four attributes are significantly dependent. So in this experiment, attributes in CelebA dataset are not mutually independent while some attribute pairs are independent.

The assumption of mutual independence may be too strong and interpretable factors in the real-world data tend to be pairwise independent. Intuitively, human beings can easily see whether two factors are independent or not, but mutual independence among three or more factors is not straightforward. Given three factors A, B and C, one should first consider they are pairwise independent or not and then investigate whether A and the joint distribution of (B, C) are independent. The latter investigation is involved with high-order relations between factors, which is not intuitive. However, most interpretable factors are intuitive and come easily from common sense. Therefore, we hypothesize that pairwise independent factors may be more consistent with human intuition. Our proposed method is based on the assumption of pairwise independence.

4 Method

Based on the discussion above, we propose an autoencoding framework which learns a pairwise independent latent distribution to capture disentangled factors. Specifically, we

describe our method to approximate the pairwise independence of latent codes. We also derive a variant of ELBO, which does not contain the total correlation term and constrains only the marginal distributions of the latent codes. Our proposed method combines the derived lower bound with the term of pairwise independence.

Given two code components z_i and z_j where $i \neq j$, $q_\phi(z_i)$ and $q_\phi(z_j)$ are expected to be independent in our scenario. The independence is measured by the mutual information

$$I_\phi(z_i; z_j) = \mathbb{E}_{q_\phi(z_i,z_j)} \log \frac{q_\phi(z_i, z_j)}{q_\phi(z_i) q_\phi(z_j)}.$$

To approximate $q_\phi(z_i)$, we use the Monte Carlo estimation based on a mini-batch of samples $\mathcal{B}$ from $p(x)$. Since the aggregated posterior $q_\phi(z) = \mathbb{E}_{p(x)} q_\phi(z \mid x)$, $q_\phi(z)$ can be approximated by $\frac{1}{b}\sum_{l=1}^{b} q_\phi(z \mid x_l)$, which is a mixture of Gaussian distributions as $q_\phi(z \mid x_l)$ is $\mathcal{N}(\mu_\phi(x_l), \sigma_\phi^2(x_l) I)$. To sample from $q_\phi(z_i)$, we first choose a sample x_l from $\mathcal{B}$ uniformly at random. Next we use the reparameterization trick and have $\tilde{z}_i = \mu_{\phi,i}(x_l) + \epsilon \sigma_{\phi,i}(x_l)$ where $\epsilon \sim \mathcal{N}(0, I)$. Then we have the estimator $q_\phi(\tilde{z}_i) = \frac{1}{b}\sum_{l=1}^{b} q_\phi(\tilde{z}_i \mid x_l)$. The same estimation can be applied to $q_\phi(z_j)$ and $q_\phi(z_i, z_j)$, too. This estimation is acceptable in our scenario because it is in the one-dimensional or two-dimensional space, and those high-dimensional sampling problems discussed in (Chen et al. 2018; Kim and Mnih 2018) are avoided. We define the average mutual information of code pairs as

$$PI_1(q_\phi(z)) = \frac{1}{\binom{k}{2}} \sum_{i \neq j} I_\phi(z_i; z_j). \tag{5}$$

An alternative measure of the pairwise independence is the KL-divergence between the aggregated posterior and the prior, defined as

$$\begin{aligned}
&\mathrm{KL}(q_\phi(z_i, z_j) \| p(z_i, z_j)) \\
=&\mathbb{E}_{q_\phi(z_i,z_j)} \log \frac{q_\phi(z_i, z_j)}{p(z_i, z_j)} \\
=&I_\phi(z_i; z_j) + \mathrm{KL}(q_\phi(z_i) \| p(z_i)) + \mathrm{KL}(q_\phi(z_j) \| p(z_j)).
\end{aligned}$$

This consists of the mutual information and the KL-divergences which push $q_\phi(z_i)$ to $p(z_i)$ and $q_\phi(z_j)$ to $p(z_j)$. Additionally, this is more computationally efficient because it eliminates the computation of $q_\phi(\tilde{z}_i)$ and $q_\phi(\tilde{z}_j)$ and only requires the probability of $q_\phi(\tilde{z}_i, \tilde{z}_j)$ and $p(\tilde{z}_i, \tilde{z}_j)$. Thus, we define

$$PI_2(q_\phi(z)) = \frac{1}{\binom{k}{2}} \sum_{i \neq j} \mathrm{KL}(q_\phi(z_i, z_j) \| p(z_i, z_j)). \tag{6}$$

It is not appropriate to combine the pairwise independence term with the ELBO. The ELBO contains the total correlation term and encourages mutual independence of codes, so it is too strong for the pairwise independence assumption. We introduce a different lower bound based on variational inference and design a corresponding autoencoding framework. We rewrite the expected log-likelihood as:

$$\begin{aligned}
&\mathbb{E}_{p(x)} \log p_\theta(x) \\
=&\mathbb{E}_{p(x)} \mathbb{E}_{q_\phi(z|x)} \log \frac{p_\theta(z, x)}{p_\theta(z \mid x)} \\
=&\mathbb{E}_{q_\phi(z,x)} \log p_\theta(z, x) - \mathbb{E}_{p(x)} \mathbb{E}_{q_\phi(z|x)} \log p_\theta(z \mid x).
\end{aligned}$$

The second term is the expected cross entropy over the random variable z given x, denoted as $H(q_\phi(z \mid x), p_\theta(z \mid x))$. With Jensen's inequality, we have

$$H(q_\phi(z \mid x), p_\theta(z \mid x)) \geq H(q_\phi(z \mid x)),$$

where $H(q_\phi(z \mid x))$ is the differential entropy of z. Notice that the differential entropy can be negative. However, when $H(q_\phi(z \mid x))$ is non-negative, we have

$$H(q_\phi(z \mid x), p_\theta(z \mid x)) \geq 0$$

and thus $\mathcal{L}'(\theta, \phi) = \mathbb{E}_{q_\phi(z,x)} \log p_\theta(z, x)$ is a lower bound of the log-likelihood. The bound is tight when $H(q_\phi(z \mid x)) = 0$ and $q_\phi(z \mid x)$ matches $p_\theta(z \mid x)$.

To analyze $\mathcal{L}'(\theta, \phi)$, we decompose it into two parts.

$$\begin{aligned}
\mathcal{L}'(\theta, \phi) &= \mathbb{E}_{q_\phi(z,x)} \log p_\theta(x \mid z) + \mathbb{E}_{q_\phi(z)} \log p(z) \\
&= \mathbb{E}_{p(x)} \mathbb{E}_{q_\phi(z|x)} \log p_\theta(x \mid z) + \sum_{i=1}^{k} \mathbb{E}_{q_\phi(z_i)} \log p(z_i).
\end{aligned}$$

We have $\log p(z) = \sum_{i=1}^{k} \log p(z_i)$ because $p(z)$ is $\mathcal{N}(0, I)$. The first term is the log-likelihood that $p_\theta(x \mid z)$ recovers sample x given the latent code from $q_\phi(z \mid x)$. The second term is the sum of the negative cross entropies $-H(q_\phi(z_i), p(z_i))$ for $i = 1, \ldots, k$. It restricts the marginal distributions $q_\phi(z_i)$ instead of the joint distribution $q_\phi(z)$.

Finally, we augment $\mathcal{L}'(\theta, \phi)$ with the pairwise independence term and arrive at the optimization problem of our Pairwise Independence Autoencoder (PIAE) as follows:

$$\begin{aligned}
\arg\max_{\theta,\phi} \quad & \mathcal{L}'(\theta, \phi) - \lambda \mathrm{PI}_\alpha(q_\phi(z)), \\
\text{s.t.} \quad & H(q_\phi(z \mid x)) \geq 0 \text{ for } x \in \mathcal{X}.
\end{aligned} \tag{7}$$

Here λ is the penalty parameter. When we have $\alpha = 1$ and take $\mathrm{PI}_1(q_\phi(z))$ in (5), we term our model as PIAE(MI). MI is short for mutual information. Similarly, we have PIAE(KL) when taking (6) with $\alpha = 2$.

To make the optimization easier, $q_\phi(z \mid x)$ is assumed to be $\mathcal{N}(\mu_\phi(x), \sigma_\phi^2(x) I)$. The model is trained with stochastic batches with the reparameterization trick. Notice that the differential entropy of $q_\phi(z \mid x)$ is

$$\begin{aligned}
H(q_\phi(z \mid x)) &= \frac{1}{2} \ln \det |2\pi \mathrm{e} \sigma_\phi^2(x)| \\
&= \frac{1}{2} \sum_{i=1}^{k} \ln(2\pi \mathrm{e} \sigma_{\phi,i}^2(x)).
\end{aligned}$$

$H(q_\phi(z \mid x)) \geq 0$ when $\ln(2\pi \mathrm{e} \sigma_{\phi,i}^2(x)) \geq 0$ for any i, which is equivalent to $\sigma_{\phi,i}^2(x) \geq \frac{1}{2\pi \mathrm{e}} \approx 0.0585$. To model this constraint, $\sigma_\phi^2(x)$ is defined as $\max(f_\phi(x), 0) + \frac{1}{2\pi \mathrm{e}}$, where f_ϕ is approximated by a neural network.

Unfortunately, the proposed method does not have the ability to generate high-quality new samples. This is because $q_\phi(z)$ is unknown and may lie in a low-dimensional manifold due to the pairwise independence constraint. Thus, codes sampled from $p(z)$ may be out of the support of $q_\phi(z)$.

We further show that the lower bound $\mathcal{L}'(\theta, \phi)$ is closely related to rate-distortion theory (Cover and Thomas 2006). We begin with the following optimization

$$\begin{aligned} &\max \mathbb{E}_{q_\phi(z)} \log p(z) \\ &\text{s.t. } -\mathbb{E}_{q_\phi(z,x)} \log p_\theta(x \mid z) \leq D. \end{aligned} \tag{8}$$

D is a constant. Writing (8) as a Lagrangian we have

$$\mathcal{L}'(\theta, \phi, \beta) = \mathbb{E}_{q_\phi(z)} \log p(z) + \beta \mathbb{E}_{q_\phi(z,x)} \log p_\theta(x \mid z).$$

$\mathcal{L}'(\theta, \phi)$ is a special case when $\beta = 1$. (8) has a close relation with the rate-distortion function. As $H(q_\phi(z \mid x) \geq 0$,

$$\begin{aligned} \mathbb{E}_{q_\phi(z)} \log p(z) \leq & \mathbb{E}_{q_\phi(z,x)} \log p(z) + \mathbb{E}_{p(x)} H(q_\phi(z \mid x)) \\ = & -\mathbb{E}_{q_\phi(z,x)} \log \frac{q_\phi(z \mid x)}{p(z)} \\ \leq & -I_\phi(z; x). \end{aligned}$$

In the last step we use (3) and that KL-divergence is non-negative. Thus the mutual information $I_\phi(z; x)$ is minimized when $\mathbb{E}_{q_\phi(z)} \log p(z)$ is maximized. On the other hand, when $p_\theta(x \mid z)$ is $\mathcal{N}(\mu_\theta(z), I)$, we have

$$-\mathbb{E}_{q_\phi(z,x)} \log p_\theta(x \mid z) = \mathbb{E}_{q_\phi(z,x)} \left[\frac{\|\mu_\theta(z) - x\|^2}{2} + C \right],$$

where C is a constant. Thus $-\mathbb{E}_{q_\phi(z,x)} \log p_\theta(x \mid z)$ corresponds to the squared-error distortion. To summarize, (8) is related to the following rate-distortion function

$$\begin{aligned} R(D') = &\min I_\phi(x; z) \\ &\text{s.t. } \mathbb{E}_{q_\phi(z,x)} \|\mu_\theta(z) - x\|^2 \leq D', \end{aligned}$$

where $D' = 2(D - C)$. Therefore, the optimization in (8) helps the model to find an achievable rate and distortion pair so as to learn a useful representation for reconstruction. This is the same for $\mathcal{L}'(\theta, \phi)$.

5 Experiment

In this section, we compare our proposed methods with other state-of-the-art methods. Particularly, we compare how well the methods capture the attributes in CelebA dataset by examining the maximum correlations and the prediction performances in canonical correlation analysis. We also compare the methods along subspace score (Li, Tang, and He 2018), an unsupervised disentanglement metric. Furthermore, we display rerendered sample sequences in the latent traversal as appropriate. The experiments are conducted on several image datasets, including MNIST (LeCun et al. 1998), FashionMNIST (Xiao, Rasul, and Vollgraf 2017), CelebA (Liu et al. 2015), Flower (Nilsback and Zisserman 2008), CUB (Wah et al. 2011), Chairs (Aubry et al. 2014) and CIFAR10 (Krizhevsky, Nair, and Hinton 2009).

Methods to be compared include β-VAE ($\beta = 20$) (Higgins et al. 2017), Improved β-VAE ($\beta = 30$) (Burgess et al. 2018), DIP-VAE ($\lambda = 20$) (Kumar, Sattigeri, and Balakrishnan 2018), β-TCVAE ($\beta = 20$) (Chen et al. 2018) and FactorVAE ($\gamma = 20$) (Kim and Mnih 2018). These are methods based on the mutual independence assumption. We also include comparisons with AnaVAE (Li, Tang, and He 2018) and InfoGAN (Chen et al. 2016). VAE (Kingma and Welling 2014) is also compared as a baseline. Hyperparameters are chosen as suggested in the original papers. We set $\lambda = 20$ in (7) for our PIAE(MI) and PIAE(KL).

We does not perform comparisons along the disentanglement metrics proposed in (Higgins et al. 2017; Kim and Mnih 2018; Chen et al. 2018) in our experiments. These metrics are applied on the synthetic dataset of 2D shapes (Matthey et al. 2017), whose factors are defined to be mutual independent. Therefore, they are not applicable in our scenario.

Implementation Details

Implementation details of our models are summarized here. The latent dimension of z is set as 16 in MNIST and Fashion-MNIST, and 64 in other datasets. The network architecture is designed according to DCGAN (Radford, Metz, and Chintala 2015). Specifically, the encoder borrows the major structure of the discriminator in DCGAN and the decoder is the same as the generator. The architecture guidelines introduced in (Radford, Metz, and Chintala 2015) can make the training easier and more stable. We use Adam optimizer (Kingma and Ba 2014) with a learning rate of 0.0001 and a momentum of 0.5. The batch size is 64. Different from the notation in (5) and (6), we randomly select only $k - 1$ pairs of z_i and z_j in each batch to reduce the computational cost. In the whole training process, all code pairs are constrained. Empirically, this stochastic approximation shows acceptable performance, but its robustness remains unclear and will be studied in our future work. Finally, the proposed algorithms are implemented with PyTorch (Paszke et al. 2017).

Canonical Correlation Analysis

To evaluate the learned code, we analyze the relation between the code z and the attributes y annotated in CelebA dataset. Inspired by (Adel, Ghahramani, and Weller 2018), we hypothesize that in the ideal case this relation can be described by a linear model. We use canonical correlation analysis (CCA) because CelebA attributes are correlated. The evaluation framework (Eastwood and Williams 2018) applies individual least square estimate for each attribute, implicitly assuming the attributes are uncorrelated, so it is not applicable here. Instead, CCA finds a sequence of uncorrelated linear combinations zv_m for $m = 1, \ldots, 40$ and a corresponding sequence of uncorrelated yu_m such that the correlations $\mathrm{Corr}(zv_m, yu_m)$'s are successively maximized. The leading canonical responses are those linear combinations of attributes best predicted by the codes. By investigating the leading correlation coefficients, we can see how well the attributes are captured.

We conduct the CCA analysis within a tenfold cross validation and display the average performances. Table 2 shows four leading correlations in the training set and the testing set, respectively. The larger the correlations are, the better

Table 2: **Four leading correlation coefficients in the CCA analysis.** The best performances are highlighted.

	Training set				Testing set			
	1	2	3	4	1	2	3	4
VAE	0.890	0.826	0.705	0.697	0.890	0.826	0.704	0.697
InfoGAN	0.374	0.314	0.147	0.101	0.373	0.313	0.146	0.099
β-VAE	0.751	0.729	0.649	0.648	0.751	0.729	0.647	0.649
Improved β-VAE	0.728	0.705	0.642	0.627	0.728	0.705	0.641	0.626
DIP-VAE	0.866	0.801	**0.756**	**0.707**	0.866	0.801	**0.756**	**0.707**
β-TCVAE	0.755	0.725	0.697	0.663	0.754	0.724	0.696	0.663
FactorVAE	0.884	0.825	0.721	0.693	0.883	0.825	0.720	0.693
AnaVAE	**0.893**	0.826	0.710	0.693	**0.893**	0.826	0.709	0.693
PIAE(MI)	0.892	**0.830**	0.716	0.695	0.891	**0.829**	0.716	0.695
PIAE(KL)	0.890	0.828	0.717	0.692	0.890	0.827	0.716	0.692

the codes capture the attributes. For the first correlation coefficients, AnaVAE has the largest values and our methods have marginally smaller ones. For the second correlations, our methods have the highest values. For the third and forth correlations, DIP-VAE has the largest correlations. However, the first two correlations of DIP-VAE are significantly lower than those of other methods. Generally, our methods give competitive performances.

We also investigate the prediction accuracy in CCA. A higher accuracy means the model captures the attributes better. The prediction accuracies are evaluated by R^2 score on the training and testing set, as shown in Table 3. R^2 score can be negative since the performance can be arbitrary ineffective, and its best possible value is 1. In this experiment, our method gives the best performances in both cases.

Table 3: **The average R^2 score in the CCA analysis.** The best performances are highlighted.

	Training set	Testing set
VAE	0.2045	0.2041
InfoGAN	-0.0871	-0.0873
β-VAE	0.1488	0.1483
Improved β-VAE	0.1302	0.1297
DIP-VAE	0.2027	0.2023
β-TCVAE	0.1614	0.1609
FactorVAE	0.2060	0.2057
AnaVAE	0.2035	0.2031
PIAE(MI)	**0.2130**	**0.2126**
PIAE(KL)	0.1979	0.1975

Subspace Score

In this part we present the comparison along subspace score (Li, Tang, and He 2018). Subspace score is an unsupervised disentanglement metric. It is based on two assumptions. The first one is that sample sequences generated by varying one latent code are expected to form an affine subspace, and subspaces of different latent codes are independent. This is measured by the clustering performance of a designed subspace clustering method. The second is that the union of these subspaces should be close to the majority of observed samples. This is reflected by the average distance between the samples and their projections in the subspace. The model with a higher subspace score is believed to separate independent factors better. Different from the implementation in (Li, Tang, and He 2018), we use the thresholding ridge regression (Peng, Yi, and Tang 2015) instead of orthogonal match pursuit in the subspace clustering part, because the thresholding ridge regression method is robust in capturing subspaces and more computationally efficient. We calculate the subspace score over five different sets of generated samples to get the average.

The results are shown in Table 4. DIP-VAE has the highest score in FashionMNIST and CelebA dataset and has a slightly higher score than our method in Chairs. AnaVAE has the best performance in MNIST. PIAE(KL) enjoys the best performances in CIFAR10, Flower and CUB datasets. PIAE(MI) has similar performance. In summary, our methods have competitive performances in this experiment.

Latent Traversal

In this part, we present the latent traversal to show the learned factors. The latent traversal is conducted in the following way. Given a selected example, the encoder infers the code z. Then a specific component of z is varied, and accordingly the decoder rerenders a sequence of samples. The variation of the sample sequences can visualize attributes learned by the autoencoding model.

Figure 3 shows the sample sequences of CelebA dataset. The models learn to disentangle factors including gender (a), the skin brightness (b) and the smiling of the face (c). The β-VAE and its improved variant give blurry faces, while other methods have samples of better clarity. In grid (a) of Figure 3, PIAE(MI) entangles gender with color tone slightly, which is also observed in other methods. PIAE(KL) entangles gender with the background color; the background blueness turns into the red hair. However, our method captures the correlated features of gender. In the left of the sequence the gentleman grows light beard, while in the right the lady has heavy makeup. Since these two factors are not independent of the gender factor, the model represents the combined factor in a singled component. In grid (b), PIAE(MI) and AnaVAE

Table 4: **The average subspace score.** The best performances are highlighted.

	MNIST	FashionMNIST	CIFAR10	Flower	CUB	CelebA	Chairs
VAE	0.557	0.562	0.614	0.563	0.571	0.600	0.608
InfoGAN	0.541	0.505	0.598	0.482	0.528	0.309	0.540
β-VAE	0.554	0.524	0.590	0.523	0.560	0.575	0.571
Improved β-VAE	0.554	0.517	0.589	0.522	0.551	0.572	0.561
DIP-VAE	0.560	**0.597**	0.609	0.560	0.558	**0.625**	**0.630**
β-TCVAE	0.541	0.522	0.582	0.529	0.540	0.565	0.586
FactorVAE	0.467	0.548	0.611	0.568	0.567	0.587	0.611
AnaVAE	**0.561**	0.556	0.614	0.561	0.571	0.596	0.611
PIAE(MI)	0.553	0.558	0.619	0.563	0.574	0.610	0.629
PIAE(KL)	0.551	0.557	**0.625**	**0.570**	**0.577**	0.609	0.626

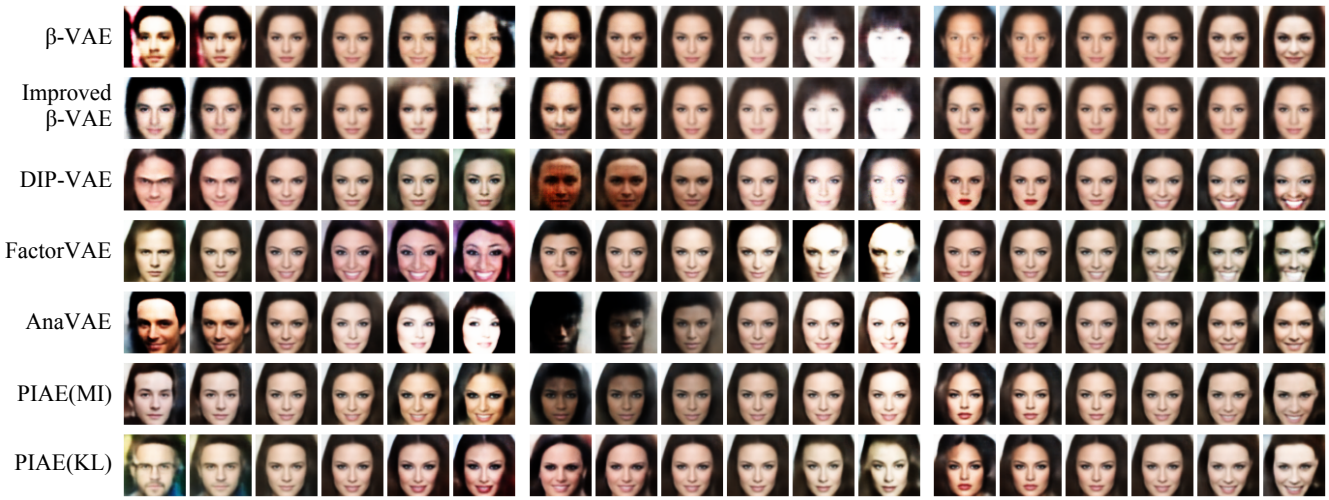

Figure 3: **Latent factors learned in CelebA dataset.** The pictures are generated by varying a component of the inferred code of a selected input image. Each figure grid shows the variation of the similar factors, and each row shows the samples generated by the same method. The models learn to disentangle factors including gender (a), the skin brightness (b) and the smiling of the face (c). The pictures are best viewed magnified on screen.

seem to confuse brightness with skin color. The difference between these two factors is subtle in image data. On the other hand, β-VAE and its variants give almost the identical sequences in grid (b), and they isolate brightness in a relatively clear way. They even infer the effect of overexposure at the end of the sequences. In grid (c), our methods and DIP-VAE entangle the smiling factor with the factor of wearing lipsticks. In general, our methods give a comparable performance in separating disentangled factors.

6 Conclusion

In this paper, we propose our Pairwise Independence Autoencoder with the attempt to learn unsupervised disentangled representation. Our method is motivated by our finding that attributes in the real-world dataset tend to be pairwise independent rather than mutually independent. A variant of the evident lower bound is introduced, which requires the variational posterior to have a non-negative differential entropy and restricts only marginal distributions. Our proposed models incorporate the lower bound with the terms of pairwise independence. Experiments show that our models can uncover interpretable factors in the data and give competitive performances as compared with other state-of-the-art methods. However, we believe not all interpretable factors are pairwise independent, and some are even correlated. As shown in Figure 3, some factors are jointly represented by one code; correlated factors may not be disentangled with the independence prior without supervised signal. Furthermore, the pairwise independence assumption may not be fully satisfied in real-world data. Therefore, we will explore the potential learning methods with a more general assumption.

Acknowledgments

This work is funded by the National Natural Science Foundation of China (NSFC) under Grant NO. 61773336 and NO. 91748127. We would like to thank the data providers.

References

Adel, T.; Ghahramani, Z.; and Weller, A. 2018. Discovering interpretable representations for both deep generative and discriminative models. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, 50–59. PMLR.

Aubry, M.; Maturana, D.; Efros, A. A.; Russell, B. C.; and Sivic, J. 2014. Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3762–3769. IEEE.

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)* 35(8):1798–1828.

Bouchacourt, D.; Tomioka, R.; and Nowozin, S. 2018. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In *AAAI Conference on Artificial Intelligence*, 2095–2102.

Burgess, C. P.; Higgins, I.; Pal, A.; Matthey, L.; Watters, N.; Desjardins, G.; and Lerchner, A. 2018. Understanding disentangling in β-VAE. *arXiv preprint arXiv:1804.03599*.

Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; and Abbeel, P. 2016. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2172–2180. Curran Associates, Inc.

Chen, T. Q.; Li, X.; Grosse, R.; and Duvenaud, D. 2018. Isolating sources of disentanglement in variational autoencoders. *arXiv preprint arXiv:1802.04942*.

Cover, T. M., and Thomas, J. A. 2006. *Elements of Information Theory (2nd Edition)*. Wiley.

Eastwood, C., and Williams, C. K. I. 2018. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representation (ICLR)*.

Hadad, N.; Wolf, L.; and Shahar, M. 2018. A two-step disentanglement method. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 772–780.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representation (ICLR)*.

Hoffman, M. D., and Johnson, M. J. 2016. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, Advances in Neural Information Processing Systems (NIPS)*.

Jordan, M. I.; Ghahramani, Z.; Jaakkola, T. S.; and Saul, L. K. 1999. An introduction to variational methods for graphical models. *Machine learning* 37(2):183–233.

Kim, H., and Mnih, A. 2018. Disentangling by factorising. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, 2654–2663. PMLR.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Kingma, D. P., and Welling, M. 2014. Auto-encoding variational bayes. In *International Conference on Learning Representation (ICLR)*.

Krizhevsky, A.; Nair, V.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, Department of Computer Science, University of Toronto.

Kumar, A.; Sattigeri, P.; and Balakrishnan, A. 2018. Variational inference of disentangled latent concepts from unlabeled observations. In *International Conference on Learning Representation (ICLR)*.

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

Li, Z.; Tang, Y.; and He, Y. 2018. Unsupervised disentangled representation learning with analogical relations. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2418–2424. International Joint Conferences on Artificial Intelligence Organization.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *International Conference on Computer Vision (ICCV)*, 3730–3738. IEEE.

Liu, Y.; Wei, F.; Shao, J.; Sheng, L.; Yan, J.; and Wang, X. 2018. Exploring disentangled feature representation beyond face identification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2080–2089.

Matthey, L.; Higgins, I.; Hassabis, D.; and Lerchner, A. 2017. dSprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/.

Nilsback, M.-E., and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image*, 722–729. IEEE.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch. In *Autodiff Workshop, Advances in Neural Information Processing Systems (NIPS)*.

Peng, X.; Yi, Z.; and Tang, H. 2015. Robust subspace clustering via thresholding ridge regression. In *AAAI Conference on Artificial Intelligence*, 3827–3833.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representation (ICLR)*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms.

Zamir, A. R.; Sax, A.; Shen, W.; Guibas, L. J.; Malik, J.; and Savarese, S. 2018. Taskonomy: Disentangling task transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3712–3722.