Supplementary Material for Learning Discrete and Continuous Factors of Data via Alternating Disentanglement

Yeonwoo Jeong   Hyun Oh Song

A. Proofs

A.1. Proof of Proposition 1

Proposition 1. *The mutual information between one dimension of a random variable and the rest can be factorized as*

$$I(z_{1:i-1}; z_i) = TC(z_{1:i}) - TC(z_{1:i-1})$$

Proof. First recall the definition of total correlation,

$$TC(z_{1:i}) = D_{\mathrm{KL}}\left(p(z_{1:i}) \parallel \prod_{j=1}^{i} p(z_j)\right)$$

Then, we have

$$\begin{aligned}
&TC(z_{1:i}) - TC(z_{1:i-1}) \\
&= \int p(z_{1:i}) \log \frac{p(z_{1:i})}{\prod_{j=1}^{i} p(z_j)} dz_{1:i} \\
&\quad - \int p(z_{1:i-1}) \log \frac{p(z_{1:i-1})}{\prod_{j=1}^{i-1} p(z_j)} dz_{1:i-1} \\
&= \int p(z_{1:i}) \log \frac{p(z_{1:i})}{\prod_{j=1}^{i} p(z_j)} dz_{1:i} \\
&\quad - \int p(z_{1:i}) \log \frac{p(z_{1:i-1})}{\prod_{j=1}^{i-1} p(z_j)} dz_{1:i} \\
&= \int p(z_{1:i}) \log \frac{p(z_{1:i})}{p(z_{1:i-1})p(z_i)} dz_{1:i} \\
&= I(z_{1:i-1}; z_i)
\end{aligned}$$

□

A.2. Proof of Proposition 2

Proposition 2. *The mutual information between x and partitions of $z = [z_1, z_2]$ can be factorized as,*

$$I(x; [z_1, z_2]) = I(x; z_1) + I(x; z_2) - I(z_1; z_2)$$

Proof. Recall the conditional independence of the latent variables $p(z_1, z_2|x) = p(z_1|x)p(z_2|x)$,

$$\begin{aligned}
&I(x; [z_1, z_2]) \\
&= \int p(x, z_1, z_2) \log \frac{p(x, z_1, z_2)}{p(x)p(z_1, z_2)} dz_1 dz_2 dx \\
&= \int p(x, z_1, z_2) \log \left(\frac{p(x, z_1, z_2)}{p(x)p(z_1, z_2)} \cdot \frac{p(x)p(z_1)}{p(x, z_1)} \right. \\
&\qquad \left. \cdot \frac{p(x)p(z_2)}{p(x, z_2)} \cdot \frac{p(z_1, z_2)}{p(z_1)p(z_2)} \right) dz_1 dz_2 dx \\
&\quad + \int p(x, z_1, z_2) \log \frac{p(x, z_1)}{p(x)p(z_1)} dx dz_1 dz_2 \\
&\quad + \int p(x, z_1, z_2) \log \frac{p(x, z_2)}{p(x)p(z_2)} dx dz_1 dz_2 \\
&\quad - \int p(x, z_1, z_2) \log \frac{p(z_1, z_2)}{p(z_1)p(z_2)} dx dz_1 dz_2 \\
&= \int p(x, z_1, z_2) \log \frac{p(x, z_1, z_2)}{p(x)} \cdot \frac{p(x)}{p(x, z_1)} \cdot \frac{p(x)}{p(x, z_2)} dz_1 dz_2 dx \\
&\quad + \int p(x, z_1) \log \frac{p(x, z_1)}{p(x)p(z_1)} dx dz_1 \\
&\quad + \int p(x, z_2) \log \frac{p(x, z_2)}{p(x)p(z_2)} dx dz_2 \\
&\quad - \int p(z_1, z_2) \log \frac{p(z_1, z_2)}{p(z_1)p(z_2)} dz_1 dz_2 \\
&= \int p(x)p(z_1, z_2|x) \log \frac{p(z_1, z_2|x)}{p(z_1|x)p(z_2|x)} dz_1 dz_2 dx \\
&\quad + I(x; z_1) + I(x; z_2) - I(z_1; z_2) \\
&= \mathbb{E}_{x \sim p(x)} [\int p(z_1, z_2|x) \log \frac{p(z_1, z_2|x)}{p(z_1|x)p(z_2|x)} dz_1 dz_2] \\
&\quad + I(x; z_1) + I(x; z_2) - I(z_1; z_2) \\
&= I(x; z_1) + I(x; z_2) - I(z_1; z_2)
\end{aligned}$$

□

B. Implementation details

We follow the Network architecture in (Dupont, 2018). We use $[0, 1]$ normalized image data. Appendix B is the model architecture for 64×64 images (Chairs and dSprites). MNIST and FashionMNIST (which is 28×28) is resized to 32×32 and architecture in Appendix B was used. Batch size for training is fixed with 64. β_h is fixed with 10.0 for our experiments.

Encoder	Decoder
4×4 conv 32,ReLU, stride 2	input dim $\times$ 256 fully connected, ReLU
4×4 conv 32,ReLU, stride 2	$256 \times 64 * 4 * 4$ fully conncted, ReLU
4×4 conv 64,ReLU, stride 2	4×4 conv transpose 64, ReLU, stride 2
4×4 conv 64,ReLU, stride 2	4×4 conv transpose 32, ReLU, stride 2
$64 * 4 * 4 \times 256$ fully connected, ReLU	4×4 conv transpose 32, ReLU, stride 2
$256 \times$ output dim fully connected	4×4 conv transpose 1, Sigmoid, stride 2

Table 1. Encoder and decoder architecture for Dsprites and Chairs data

Encoder	Decoder
4×4 conv 32,ReLU, stride 2	input dim $\times$ 256 fully connected, ReLU
4×4 conv 32,ReLU, stride 2	$256 \times 64 * 4 * 4$ fully conncted, ReLU
4×4 conv 64,ReLU, stride 2	4×4 conv transpose 32, ReLU, stride 2
$64 * 4 * 4 \times 256$ fully connected, ReLU	4×4 conv transpose 32, ReLU, stride 2
$256 \times$ output dim fully connected	4×4 conv transpose 1, Sigmoid, stride 2

Table 2. Encoder and decoder architecture for MNIST and FashionMNIST

B.1. dSprites

- Dimension of discrete : 3
- Optimizer: Adam with learning rate 3e-4
- λ' : 0.001
- r : 2e4
- t_d : 1e5
- Iterations : 3e5

B.2. MNIST

- Dimension of discrete : 10
- Optimizer : Adam with learning rate 3e-4
- λ' : 0.1
- r : 1e4
- t_d : 0
- Iterations : 1.2e5

B.3. FashionMNIST

- Dimension of discrete : 10
- Optimizer : Adam with learning rate 1e-4
- λ' : 0.1
- r : 1e4
- t_d : 0
- Iterations : 1.2e5

B.4. Chairs

- Dimension of discrete : 3
- Optimizer: Adam with learning rate 1e-4
- λ' : 0.01
- r : 2e4
- t_d : 6e4
- Iterations : 1.5e5

C. Disentanglement score

We follow the disentanglement score details from (Kim & Mnih, 2018) and (Dupont, 2018). At first, we prune out all latent dimensions where variational posterior collapses to the priror. Concretely, we prune the continous latent dimension z_j where

$$\mathbb{E}_{x\sim p(x)} D_{\mathrm{KL}}(q_\phi(z_j \mid x) \parallel p(z_j)) < 0.1 \ \ .$$

We evaluate disentanglement score with the surviving dimensions. We choose a factor k from K factors (*i.e.* postion x, position y, rotation, scale, shape). Then, we obtain the representations from $L\ (=100)$ data of which factor k is fixed and the other factors are randomly chosen. We take the empirical variance of each latent dimensions and normalize with each empirical variance over the full data[1]. Concretely, the empirical variance on j latent dimension [2], is defined as

$$\widehat{\mathrm{Var}}_j = \frac{1}{2N(N-1)} \sum_{p,q=1}^{N} d(x_p, x_q),$$

where $d(x_p, x_q) = \begin{cases} \mathbb{I}(x_p \neq x_q) & \text{if } j = m+1 \\ (x_p - x_q)^2 & \text{otherwise} \end{cases}$. This procedure generates a vote (j, k) where

$$j = \underset{j^*}{\operatorname{argmin}} \frac{1}{v_{j^*}} \widehat{\mathrm{Var}}_{j^*}.$$

We generate $M\ (=\ 800)$ votes $(a_i, b_i)_{i=1}^{M}$. Let $V_{jk} = \sum_{i=1}^{M} \mathbb{I}(a_i = j, b_i = k)$. Concretely, the disentanglement score is

$$\frac{1}{M} \sum_j \max_k V_{jk}.$$

Random chance algoirhtm takes $\frac{1}{K}$ as a accuracy.

[1] We denote the empirical variance of latent dimension j on full data, v_j.

[2] For convenience, $z_{m+1} = d$.

References

Dupont, E. Learning disentangled joint continuous and discrete representations. In *NIPS*, 2018.

Kim, H. and Mnih, A. Disentangling by factorising. In *ICML*, 2018.