

VAE variants

Hao Dong

Peking University

VAE variants

- Convolutional VAE
- Conditional VAE
- Representation learning {
 - β -VAE
 - IWAE
- Hierarchical representation learning {
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning {
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

VAE variants

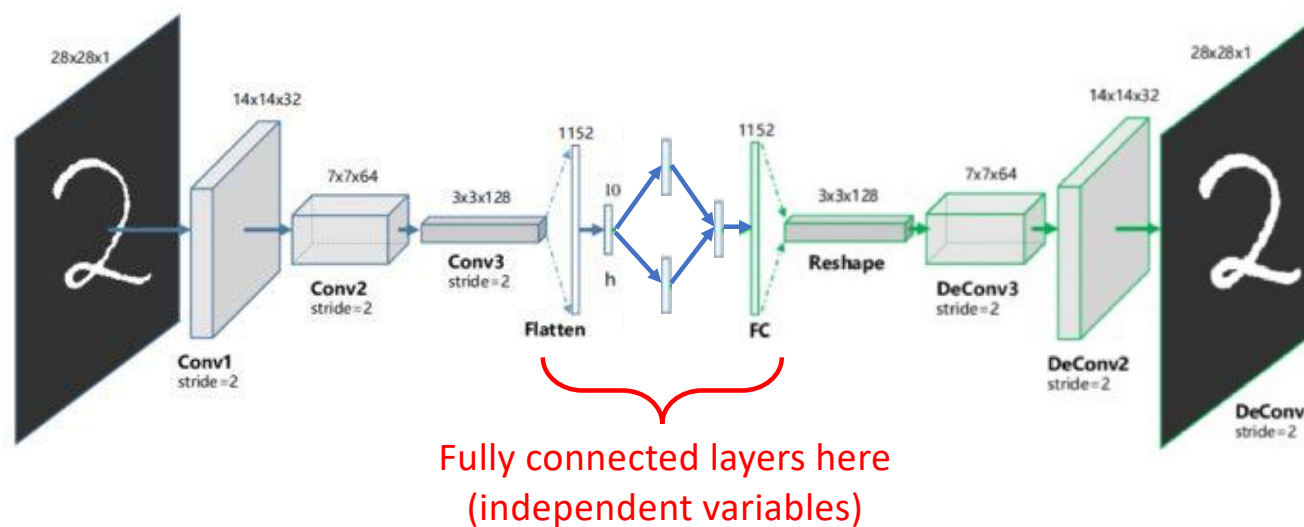
- Convolutional VAE
- Conditional VAE
- Representation learning
 - IWAE
 - β -VAE
- Hierarchical representation learning
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

Convolutional Variational Autoencoder

- **Limitations of vanilla VAE**

- The size of weight of fully connected layer == input size x output size
- If VAE uses fully connected layers only, will lead to curse of dimensionality when the input dimension is large (e.g., image).

- **Solution**

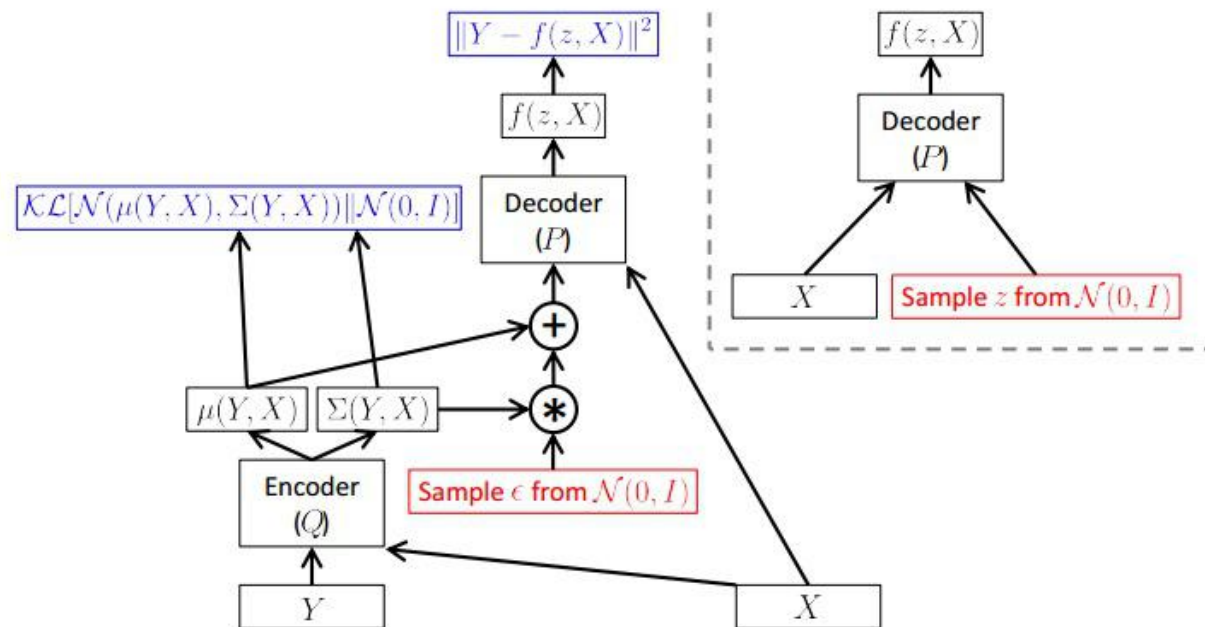


VAE variants

- Convolutional VAE
 - **Conditional VAE**
 - β -VAE
 - IWAE
 - Ladder VAE
 - Progressive + Fade-in VAE
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)
- Representation learning {
- Hierarchical representation learning {
- Temporal representation learning {

Conditional Variational Autoencoder

- Train and inference with labelled data.



Recap: Variational Autoencoder

- **Recap: Setting up the objective**
 - Maximise $P(X)$
 - Set $Q(z)$ to be an arbitrary distribution

$$\mathcal{D}[Q(z) \| P(z|X)] = E_{z \sim Q} [\log Q(z) - \log P(z|X)]$$

$$\mathcal{D}[Q(z) \| P(z|X)] = E_{z \sim Q} [\log Q(z) - \log P(X|z) - \log P(z)] + \log P(X)$$

$$\log P(X) - \mathcal{D}[Q(z) \| P(z|X)] = E_{z \sim Q} [\log P(X|z)] - \mathcal{D}[Q(z) \| P(z)]$$

$$\log P(X) - \mathcal{D}[Q(z|X) \| P(z|X)] = E_{z \sim Q} [\log P(X|z)] - \mathcal{D}[Q(z|X) \| P(z)]$$

Recap: Variational Autoencoder

- Recap: Setting up the objective

$$\log P(X) - \underbrace{\mathcal{D}[Q(z|X) \| P(z|X)]}_{\text{encoder ideal}} = \underbrace{E_{z \sim Q}[\log P(X|z)]}_{\text{reconstruction}} - \underbrace{\mathcal{D}[Q(z|X) \| P(z)]}_{\text{KLD}}$$

Conditional Variational Autoencoder

- **Setting up the objective with labels**
 - Maximise $P(Y|X)$
 - Set $Q(z)$ to be an arbitrary distribution

$$\mathcal{D}[Q(z|Y, X) \| P(z|Y, X)] = E_{z \sim Q(\cdot|Y, X)} [\log Q(z|Y, X) - \log P(z|Y, X)]$$

$$\begin{aligned} \mathcal{D}[Q(z|Y, X) \| P(z|Y, X)] = \\ E_{z \sim Q(\cdot|Y, X)} [\log Q(z|Y, X) - \log P(Y|z, X) - \log P(z|X)] + \log P(Y|X) \end{aligned}$$

$$\begin{aligned} \log P(Y|X) - \mathcal{D}[Q(z|Y, X) \| P(z|Y, X)] = \\ E_{z \sim Q(\cdot|Y, X)} [\log P(Y|z, X)] - \mathcal{D}[Q(z|Y, X) \| P(z|X)] \end{aligned}$$

Conditional Variational Autoencoder

- Setting up the objective

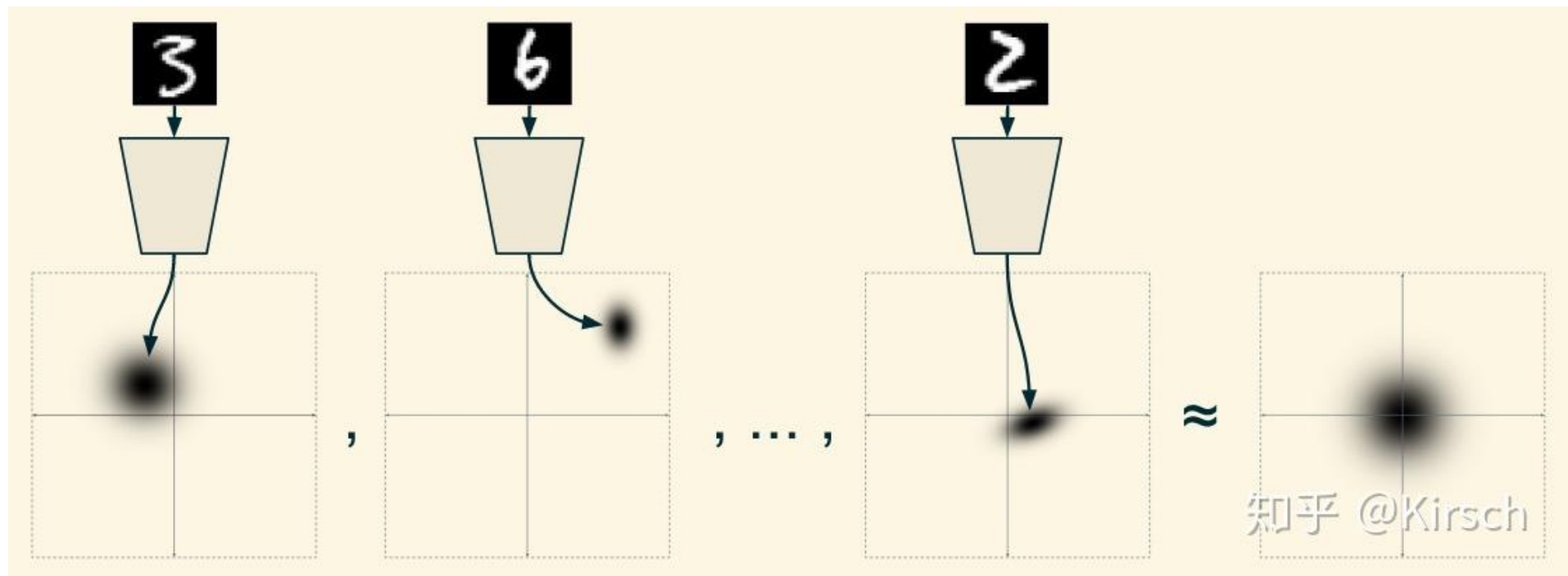
$$\log P(Y|X) - \mathcal{D} [Q(z|Y, X) \| P(z|Y, X)] =$$

$E_{z \sim Q(\cdot|Y, X)} [\log P(Y|z, X)] - \mathcal{D} [Q(z|Y, X) \| P(z|X)]$

reconstructionKLD

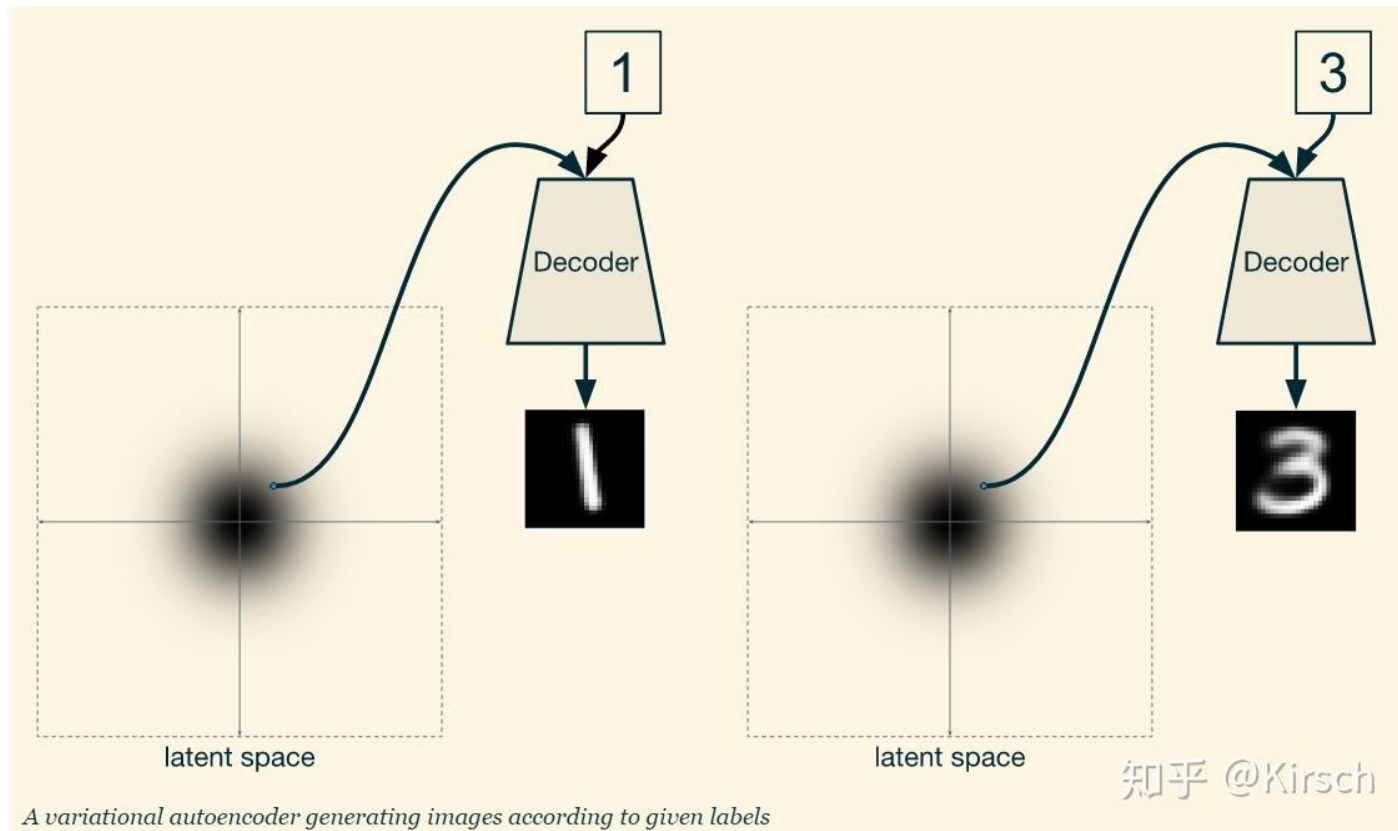
Conditional Variational Autoencoder

- Train and inference **without** labelled data i.e., vanilla VAE



Conditional Variational Autoencoder

- Train and inference **with** labelled data.



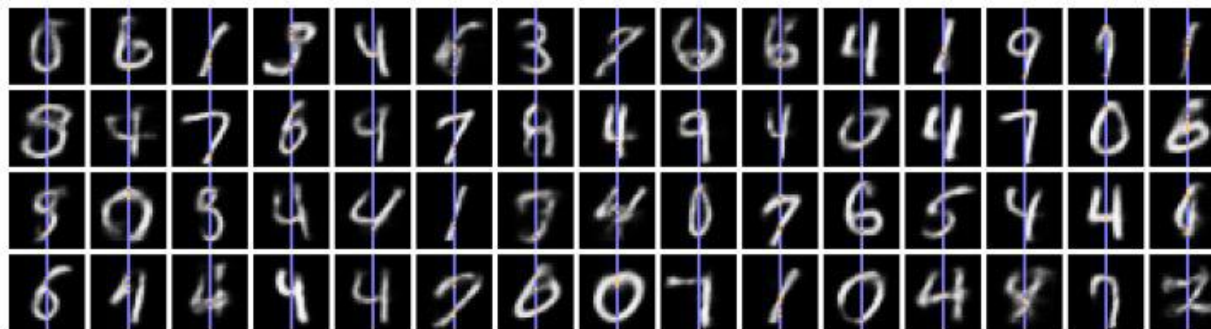
Conditional Variational Autoencoder

- Train and inference with labelled data.



Conditional Variational Autoencoder

- Train and inference with labeled data.



(a) CVAE



(b) Regressor

VAE variants

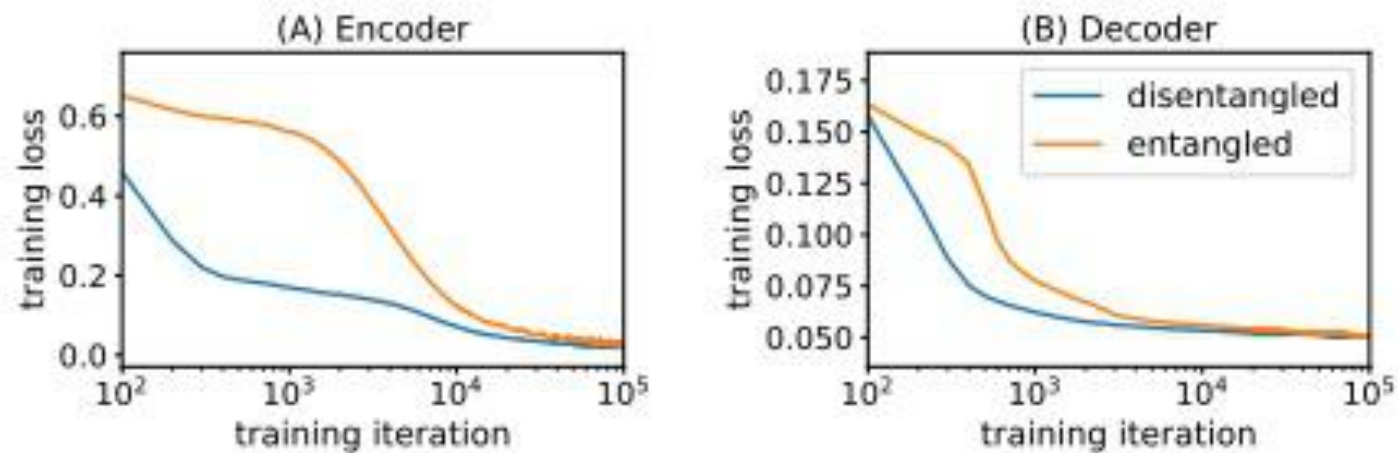
- Convolutional VAE
 - Conditional VAE
 - **β -VAE**
 - IWAE
 - Ladder VAE
 - Progressive + Fade-in VAE
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)
- Representation learning {
- Hierarchical representation learning {
- Temporal representation learning {

Before we start

- **Disentangled / Factorised representation**
 - Each variable in the inferred latent representation is only sensitive to one single generative factor and relatively invariant to other factors
 - Good interpretability and easy generalization to a variety of tasks

Before we start

- Unsupervised hierarchical representation learning



β -VAE

- **Unsupervised representation learning**
 - Augment the original VAE framework with a single hyper-parameter β that modulates the learning constraints
 - Impose a limit on the capacity of the latent information channel

β -VAE

$$\begin{aligned} & \max_{\phi, \theta} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [\mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z})] \\ & \text{subject to } D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z})) < \delta \end{aligned}$$

β -VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. ICLR 2017.

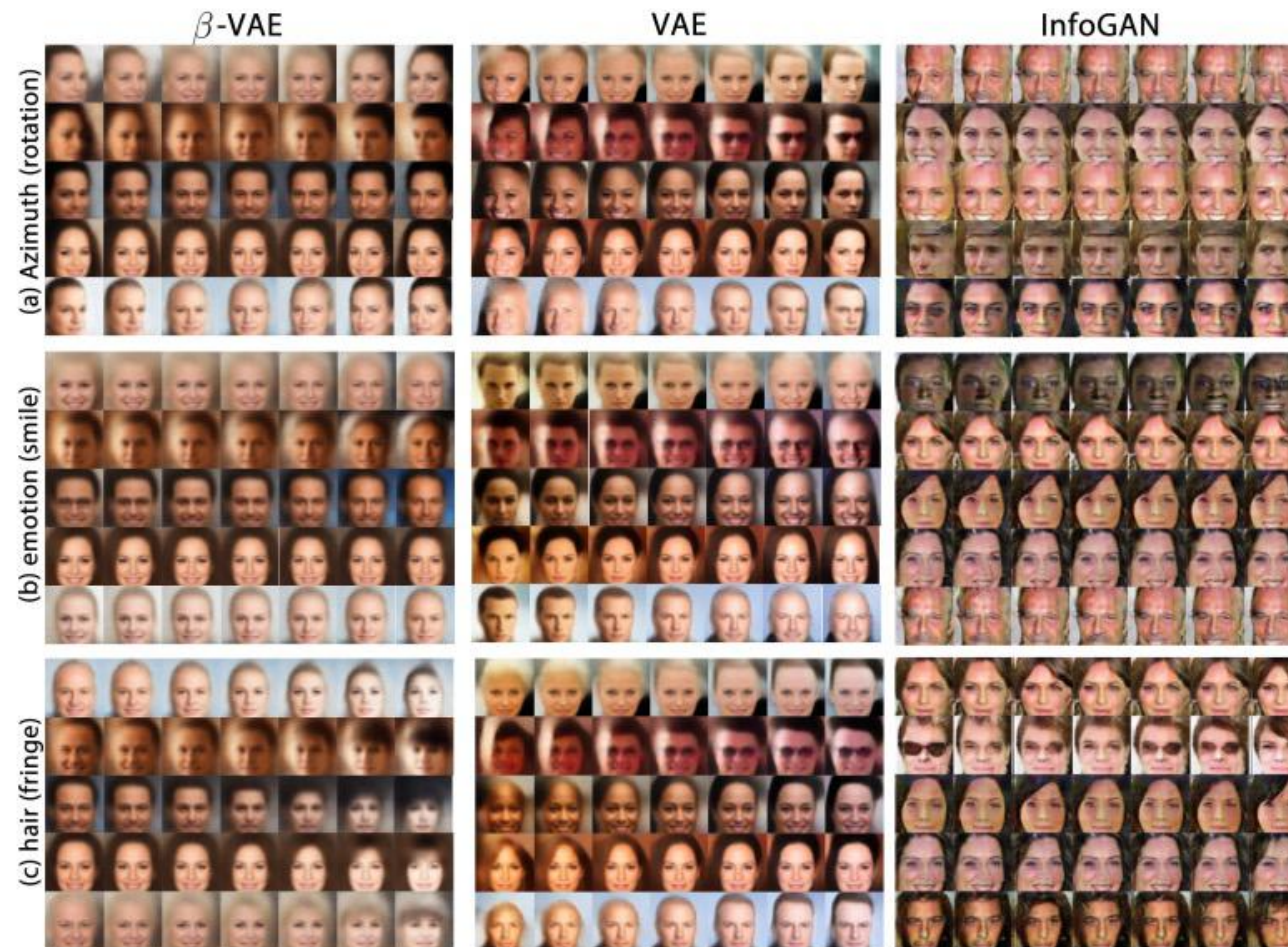
β -VAE

$$\begin{aligned}\mathcal{F}(\theta, \phi, \beta) &= \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \beta(D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}))) - \delta \\ &= \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z})) + \beta\delta \\ &\geq \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}))\end{aligned}\quad ; \text{ Because } \beta, \delta \geq 0$$

β -VAE

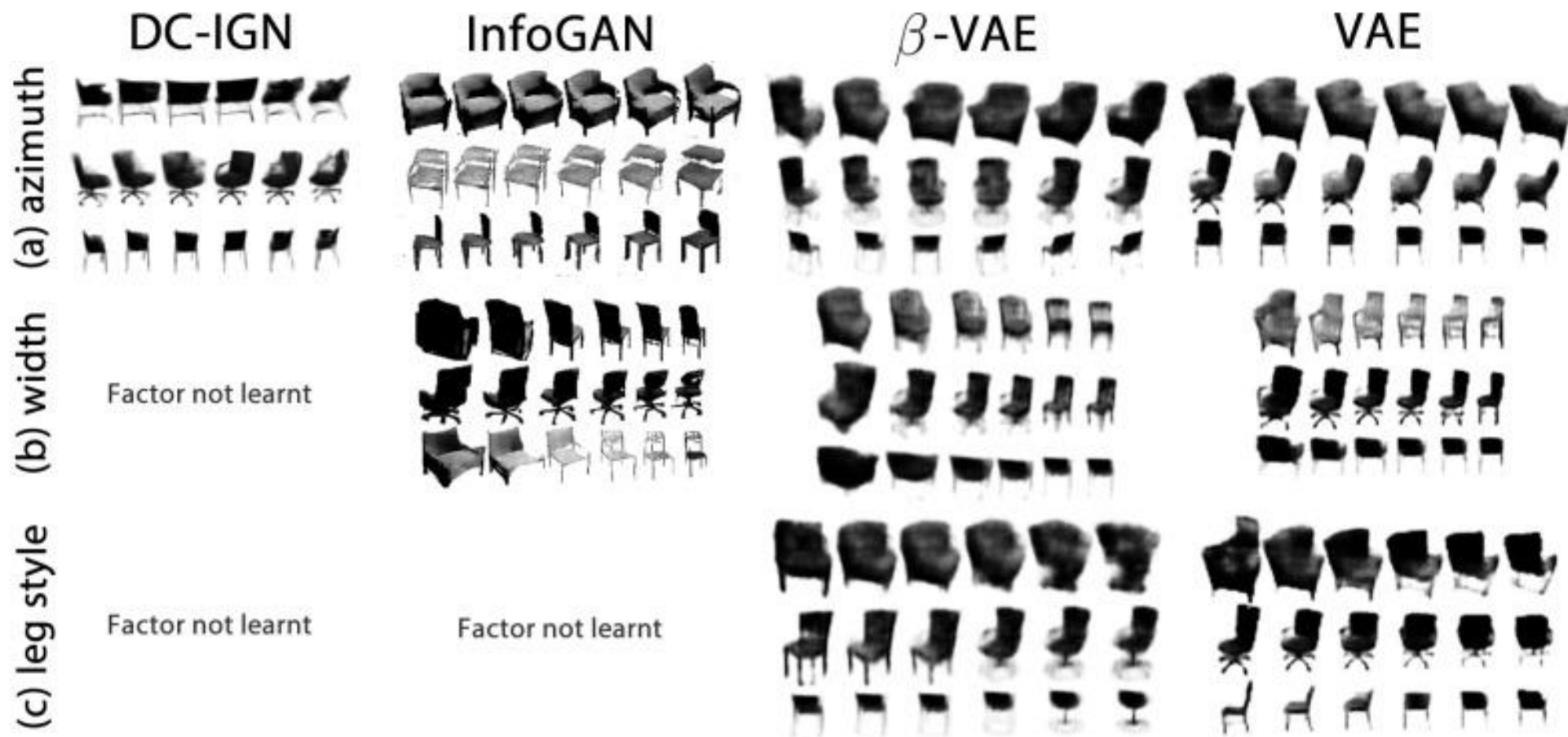
$$L_{\text{BETA}}(\phi, \beta) = -\mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) + \beta D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}))$$

β -VAE



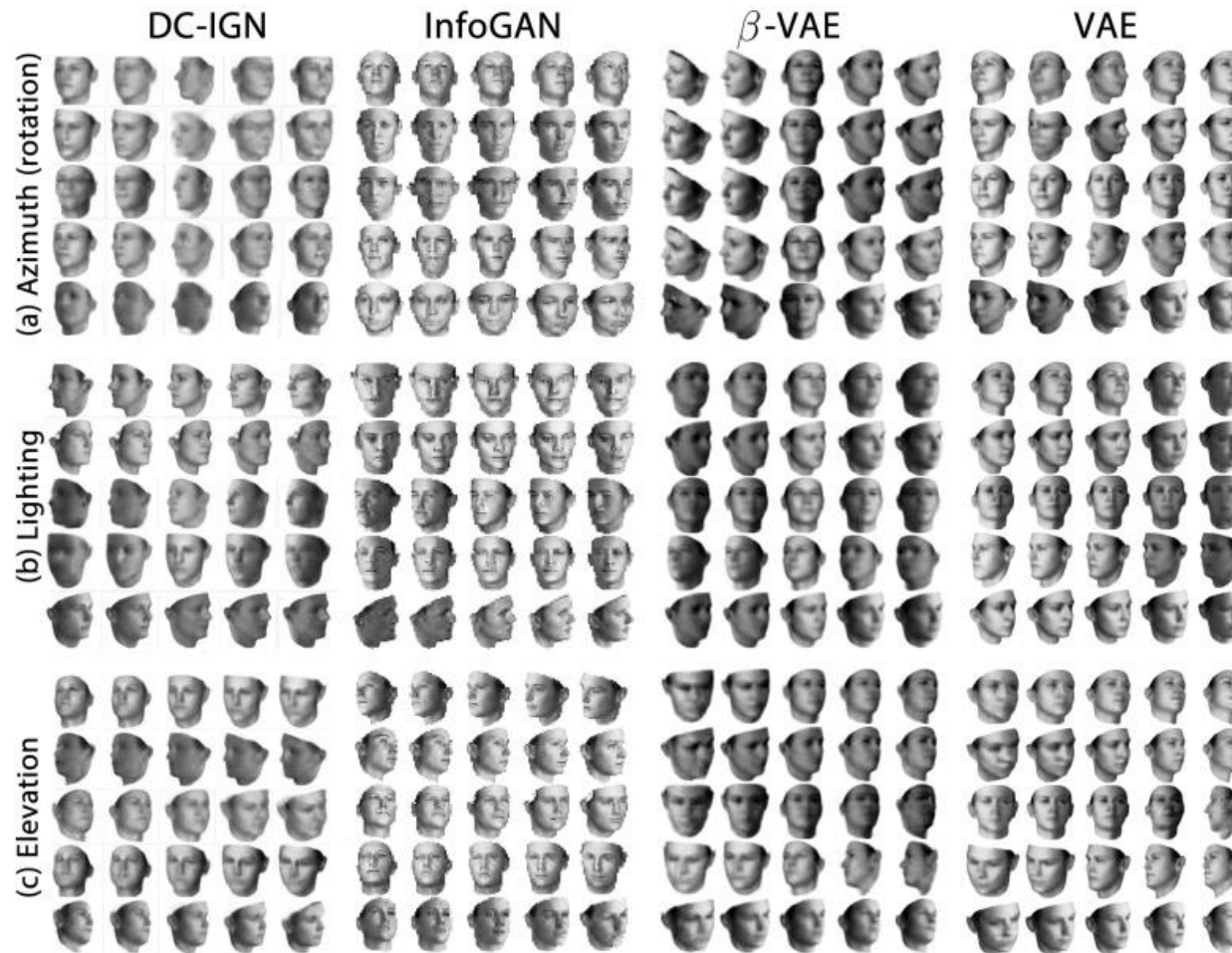
β -VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. ICLR 2017.

β -VAE



β -VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. ICLR 2017.

β -VAE



β -VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. ICLR 2017.

β -VAE

- **Discussion: It is really unsupervised?**
 - It is unsupervised/self-supervised learning, because it does not need any label data
 - It is not fully unsupervised learning, it works because of the inductive bias of the neural network model, the hierarchical design introduces prior knowledge about the data

VAE variants

- Convolutional VAE
 - Conditional VAE
 - β -VAE
 - **IWAE**
 - Ladder VAE
 - Progressive + Fade-in VAE
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)
- Representation learning {
- Hierarchical representation learning {
- Temporal representation learning {

IWAE (Importance Weighted Autoencoder)

- **Optimise a tighter lower bound than VAE**
 - VAE just optimises a lower bound of $\log P(X)$

$$\log P(X) - \underbrace{\mathcal{D}[Q(z|X) \| P(z|X)]}_{\text{encoder}} = \underbrace{E_{z \sim Q}[\log P(X|z)]}_{\text{reconstruction}} - \underbrace{\mathcal{D}[Q(z|X) \| P(z)]}_{\text{KLD}}$$

ELBO

IWAE

- Optimise a tighter lower bound than VAE

$$\log p(x) = \log \int p(x, z) dz = \log \int \frac{p(x, z)}{q(z|x)} q(z|x) dz = \log E_{q(z|x)} \left[\frac{p(x, z)}{q(z|x)} \right]$$

$$\log E_{q(z|x)} \left[\frac{p(x, z)}{q(z|x)} \right] = \log E_{z_1, z_2, \dots, z_k \sim q(z|x)} \left[\frac{1}{k} \sum_{i=1}^k \frac{p(x, z_i)}{q(z_i|x)} \right]$$

$$L_k(x) = E_{z_1, z_2, \dots, z_k \sim q(z|x)} \left[\log \frac{1}{k} \sum_{i=1}^k \frac{p(x, z_i)}{q(z_i|x)} \right] \leq \log E_{z_1, z_2, \dots, z_k \sim q(z|x)} \left[\frac{1}{k} \sum_{i=1}^k \frac{p(x, z_i)}{q(z_i|x)} \right] = \log p(x)$$

IWAE

$$ELBO(\theta) = \mathbb{E}_q[\log p(x, z)] - \mathbb{E}_q[\log q_\theta(z | x)]$$

VAE 的loss: $E_{z \sim q(z|x)} \left[\log \frac{p(x, z)}{q(z|x)} \right]$

而IWAE的loss: $E_{z_1, z_2, \dots, z_k \sim q(z|x)} \left[\log \frac{1}{k} \sum_{i=1}^k \frac{p(x, z_i)}{q(z_i|x)} \right]$

IWAE

- Why “Importance weighted”

$$\nabla_{\theta} E_{z_1, z_2, \dots, z_k \sim q(z|x, \theta)} \left[\log \frac{1}{k} \sum_{i=1}^k \frac{p(x, z_i | \theta)}{q(z_i | x, \theta)} \right] = E_{z_1, z_2, \dots, z_k \sim q(z|x, \theta)} \left[\nabla_{\theta} \log \frac{1}{k} \sum_{i=1}^k w_i \right]$$

$$\text{where } w_i = \frac{p(x, z_i | \theta)}{q(z_i | x, \theta)}$$

$$\text{VAE: } \frac{1}{k} \sum_{i=1}^k \nabla_{\theta} \log w_i$$

$$\text{IWAE: } \sum_{i=1}^k \tilde{w}_i \nabla_{\theta} \log w_i$$

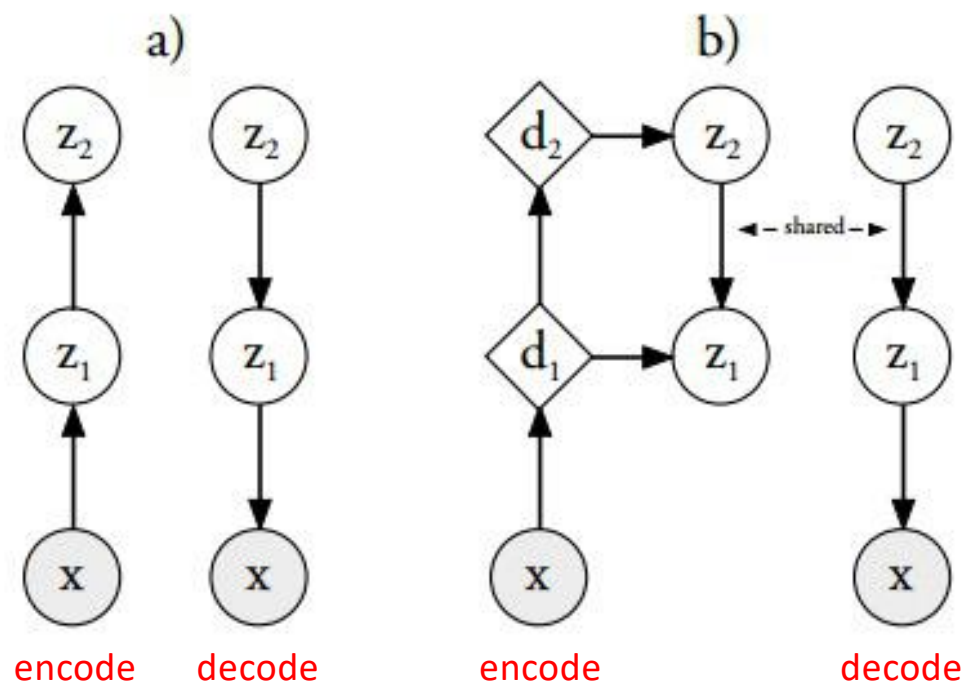
VAE variants

- Representation learning
 - Convolutional VAE
 - Conditional VAE
 - β -VAE
 - IWAE
- Hierarchical representation learning
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

Ladder VAE

- To learn **hierarchical** latent representation
- Deep models with several layers of dependent stochastic variables are difficult to train
 - Limiting the improvements obtained using these highly expressive models

Ladder VAE



Hierarchical VAE

Ladder VAE

Ladder VAE

$$\mathcal{L}(\theta, \phi; \mathbf{x})_{WU} = -\beta KL(q_{\phi}(z|x) || p_{\theta}(\mathbf{z})) + E_{q_{\phi}(z|x)} [\log p_{\theta}(\mathbf{x}|\mathbf{z})]$$

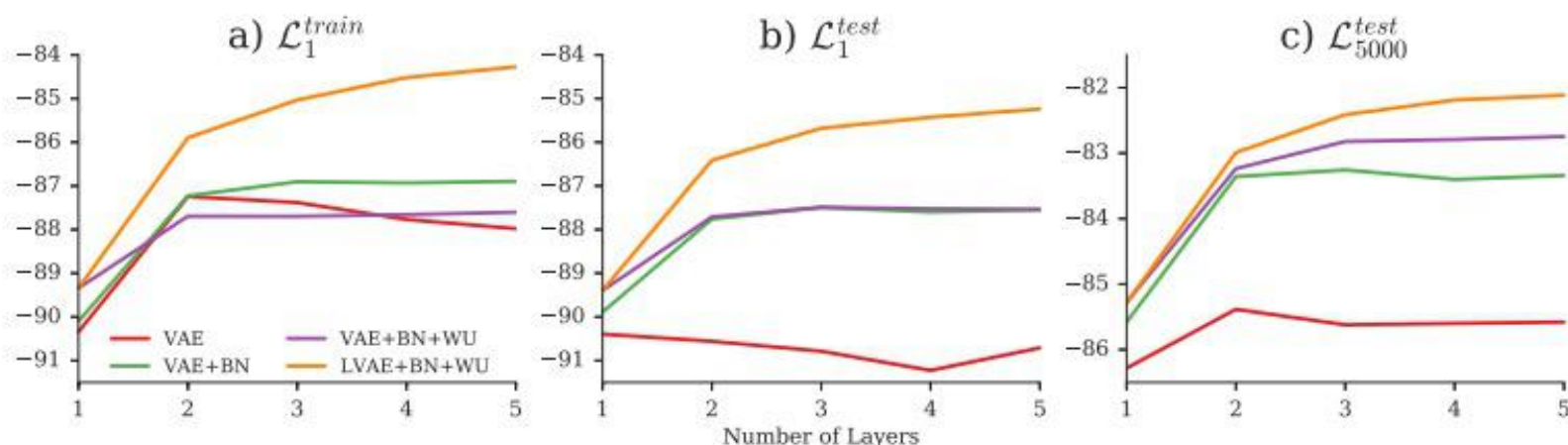


Figure 3: MNIST log-likelihood values for VAEs and the LVAE model with different number of latent layers, Batch-normalization (BN) and Warm-up (WU). a) Train log-likelihood, b) test log-likelihood and c) test log-likelihood with 5000 importance samples.

Ladder VAE

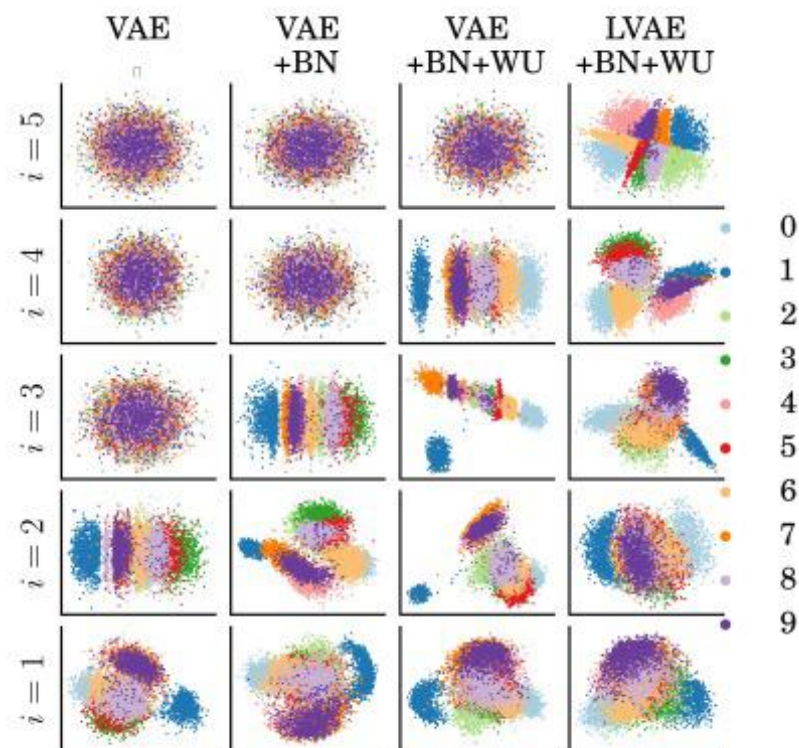
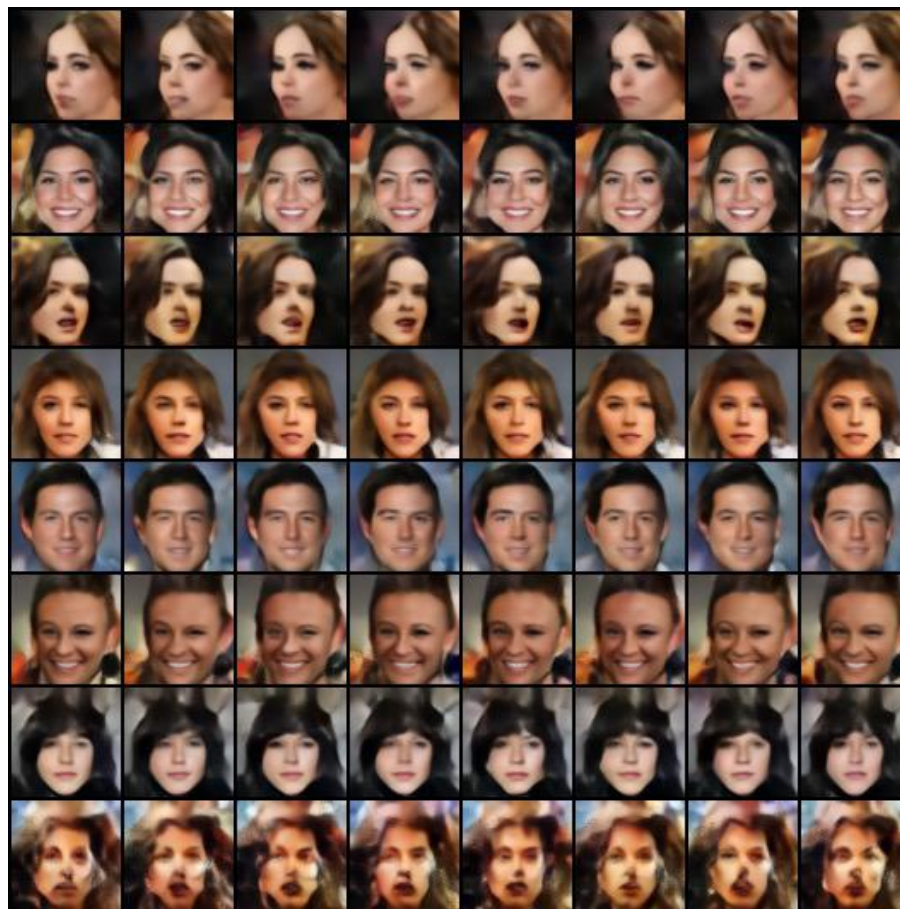


Figure 6: PCA-plots of samples from $q(z_i | z_{i-1})$ for 5-layer VAE and LVAE models trained on MNIST. Color-coded according to true class label

Ladder VAE

1 / 4

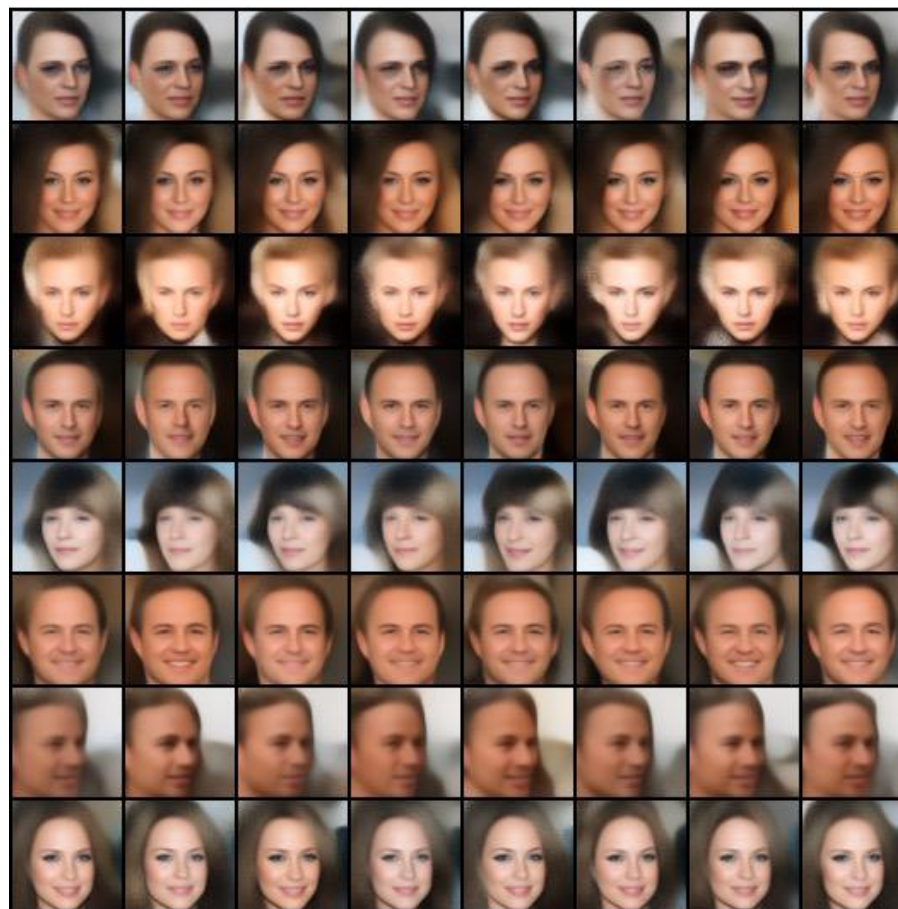
Lowest level



Ladder VAE

2 / 4

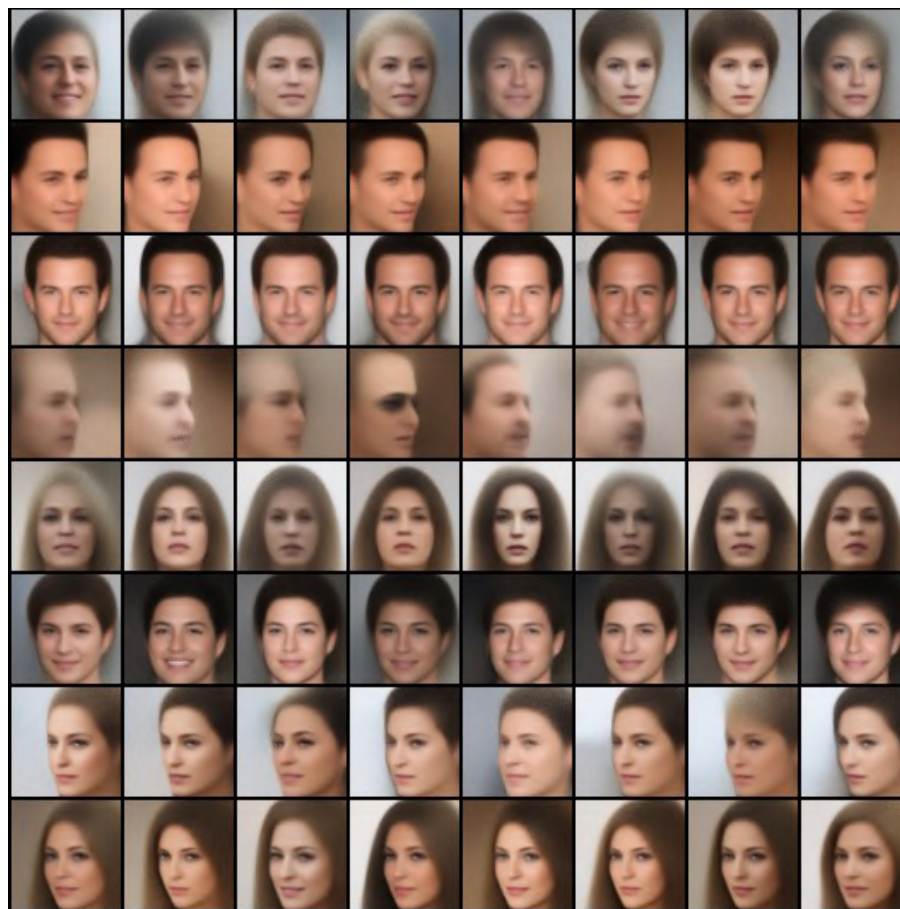
Middle level



Ladder VAE

3 / 4

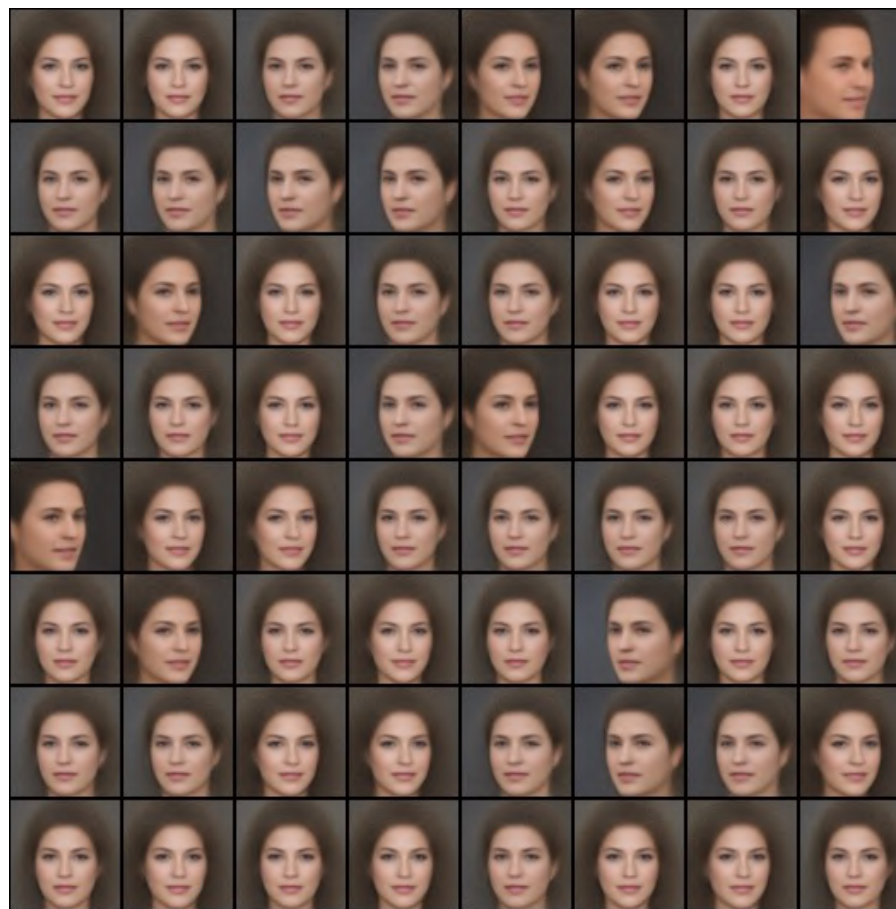
Middle level



Ladder VAE

4 / 4

High level

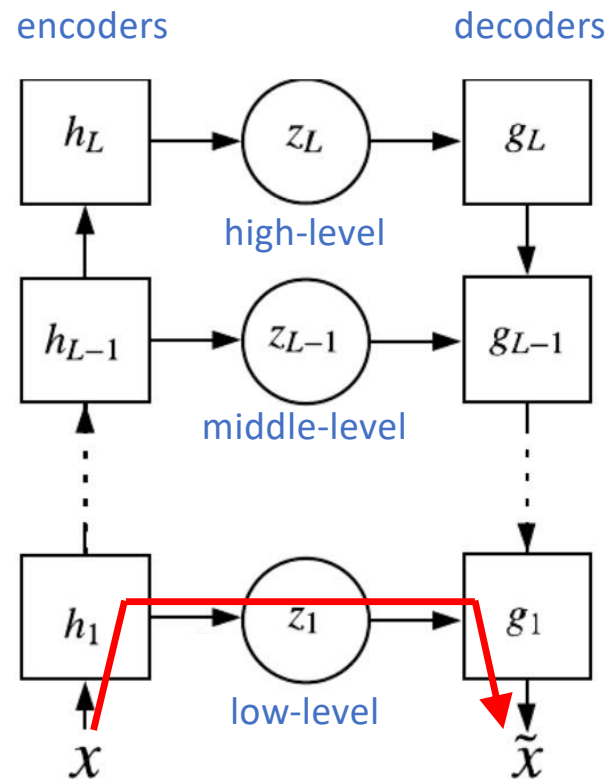


VAE variants

- Representation learning
 - Convolutional VAE
 - Conditional VAE
 - β -VAE
 - IWAE
- Hierarchical representation learning**
 - Ladder VAE
 - **Progressive + Fade-in VAE**
- Temporal representation learning
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

Progressive + Fade-in VAE

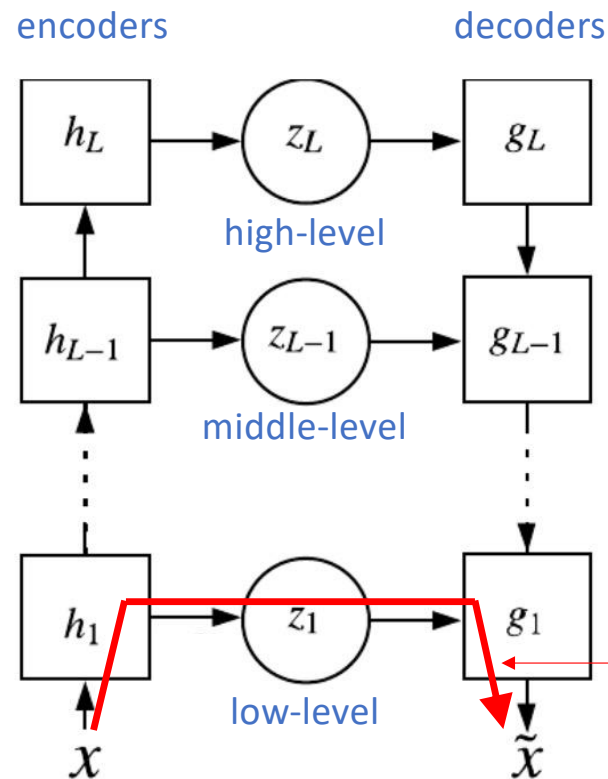
- Discussion



Can we directly train a hierarchical VAE with ladder structure like that?

Progressive + Fade-in VAE

- Discussion



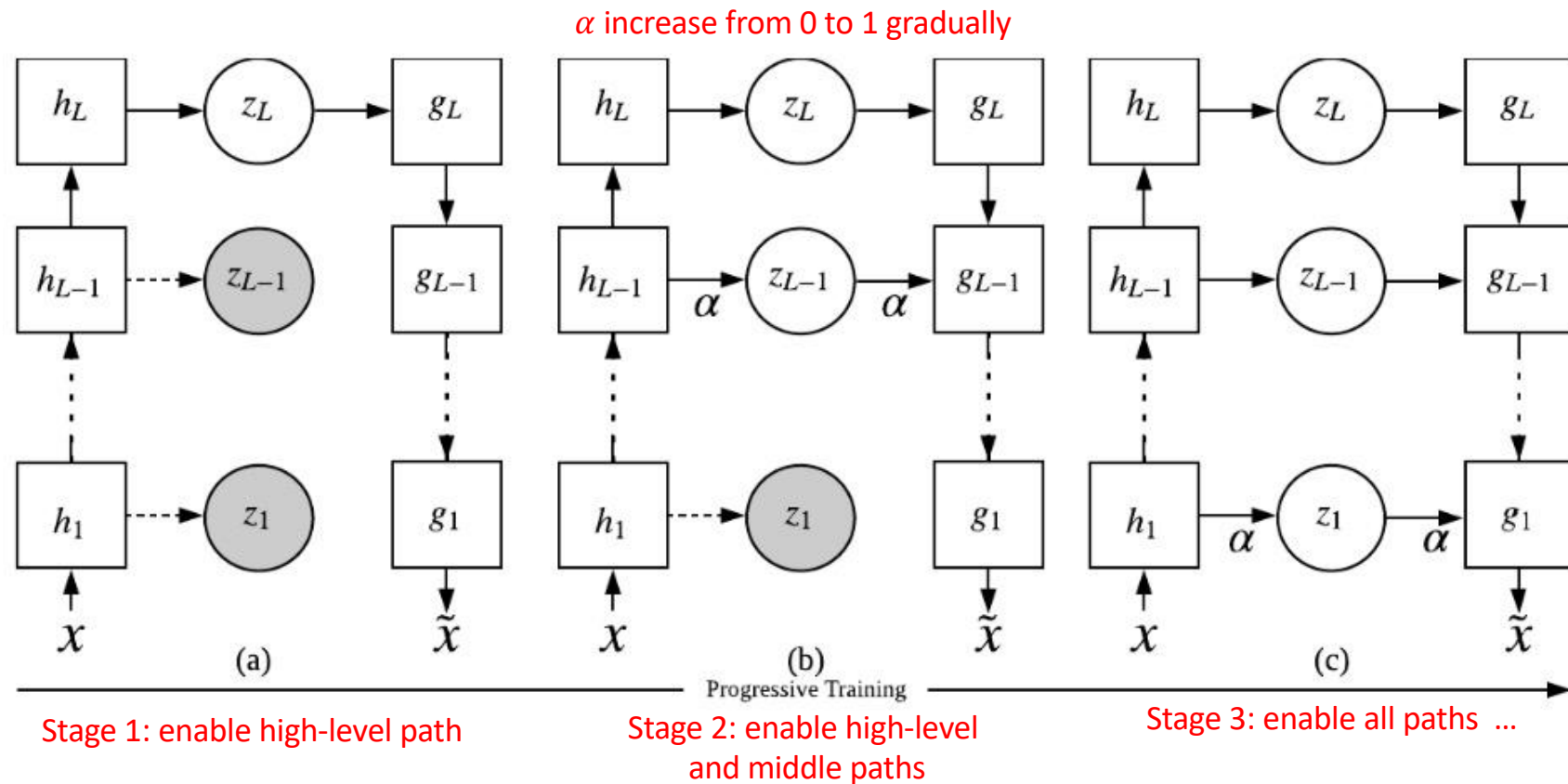
$$\begin{aligned}
 E_{q(x)}[KL(q(z|x)||p(z))] &= \int \int q(x)q(z|x)\log \frac{q(z|x)}{p(z)} dx dz \\
 &= \int \int q(x, z)\log \frac{q(z|x)q(x)q(z)}{p(z)q(x)p(z)} dx dz \\
 &= \int \int [q(x, z)\log \frac{q(x, z)}{q(x)q(z)} + q(x, z)\log \frac{q(z)}{p(z)}] dx dz \\
 &= MI_{q(x, z)}(x, z) + \int \int q(x, z)\log \frac{q(z)}{p(z)} dx dz \\
 &= MI_{q(x, z)}(x, z) + \int q(z)\log \frac{q(z)}{p(z)} dz \\
 &= \boxed{MI_{q(x, z)}(x, z)} + KL(q(z)||p(z))
 \end{aligned}$$

Information **SHORTCUT** problem

all information go through the low-level path,
other paths are ignored.
model is lazy...

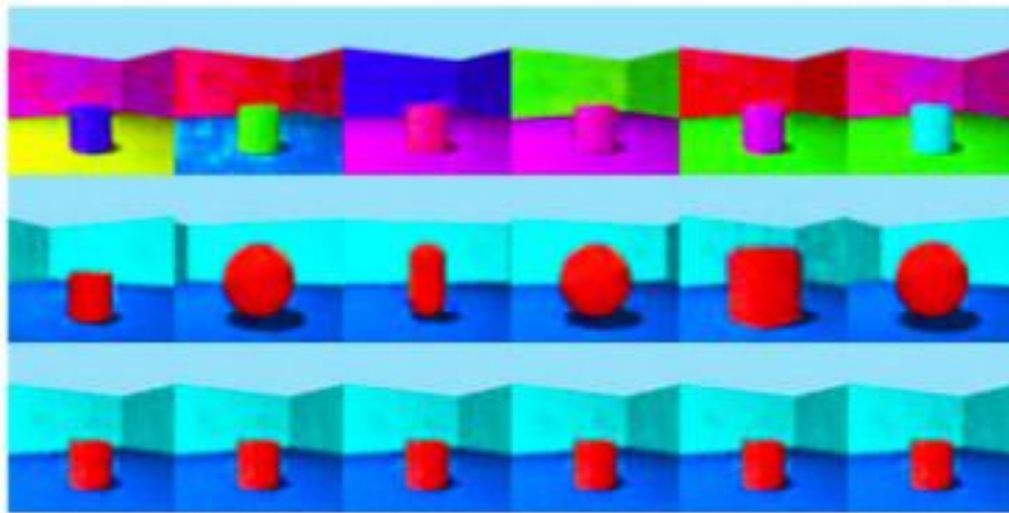
Progressive + Fade-in VAE

- Progressive + Fade-in



Progressive + Fade-in VAE

- Results



z_3 High-level: background and foreground colors

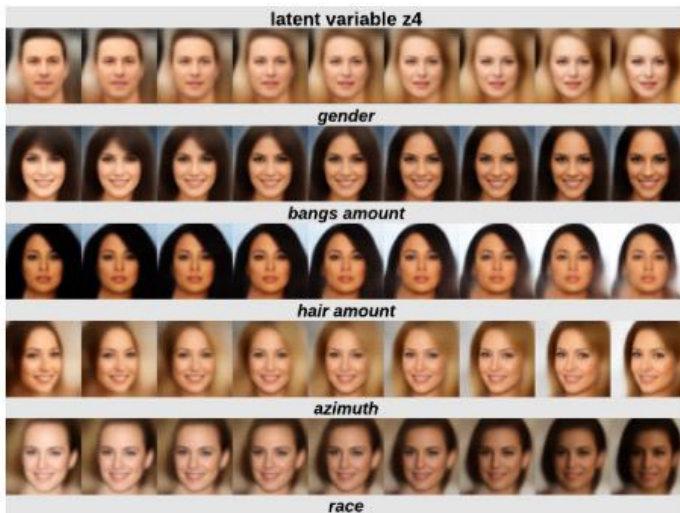
z_2 Middle-level: shape

z_1 Low-level: ...

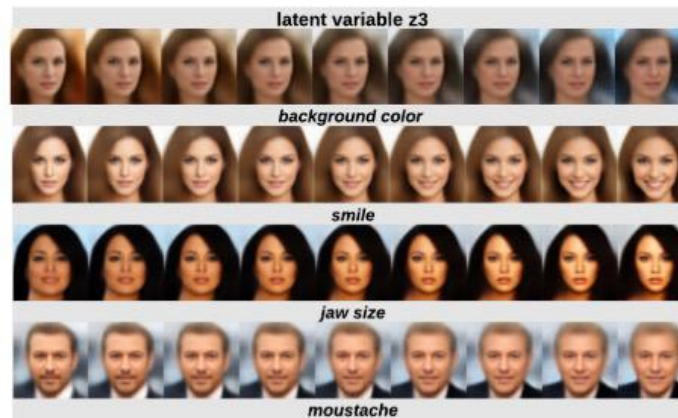
Progressive + Fade-in VAE

- Results

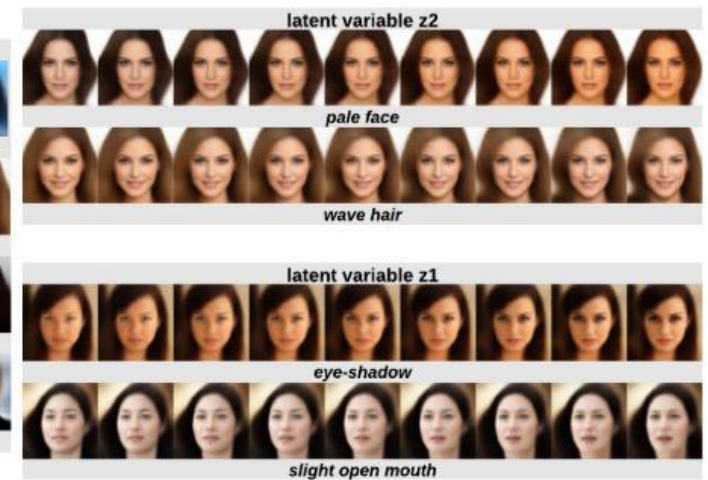
High-level Z3



Middle-level Z2



Low-level Z1

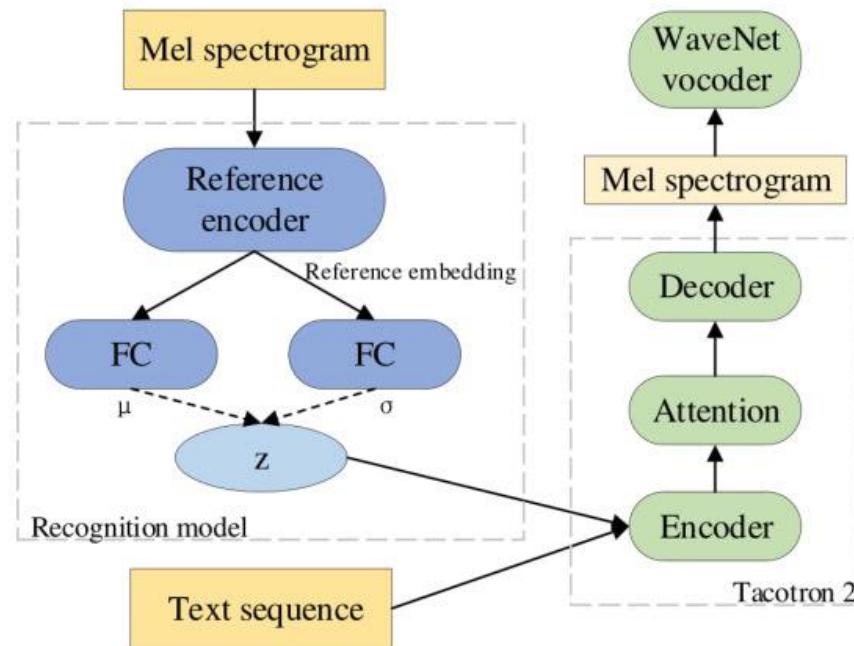


VAE variants

- Representation learning
 - Convolutional VAE
 - Conditional VAE
 - β -VAE
 - IWAE
- Hierarchical representation learning
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

VAE in speech

- Learning latent representations for style control and transfer in end-to-end speech synthesis
- RNN as encoder

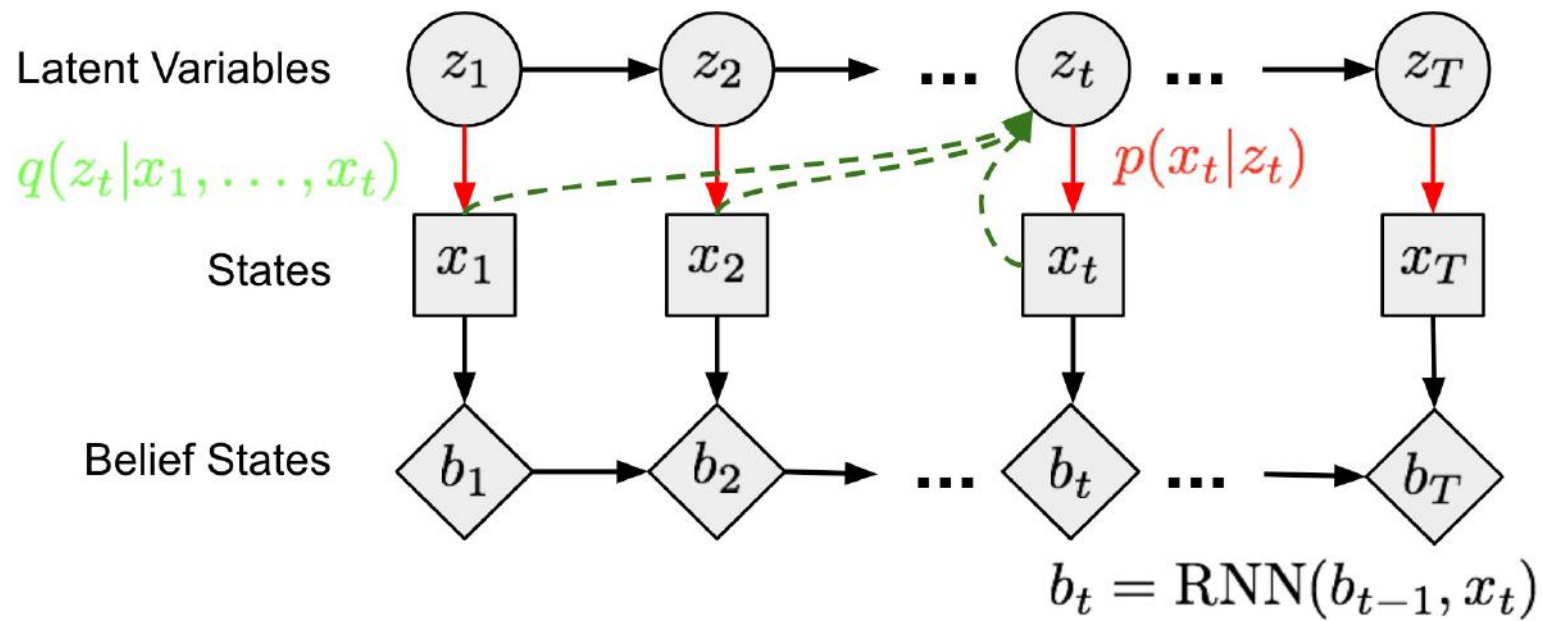


VAE variants

- Representation learning
 - Convolutional VAE
 - Conditional VAE
 - β -VAE
 - IWAE
- Hierarchical representation learning
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

TD-VAE

- State-space model as a Markov Chain model



TD-VAE

$$b_t = \text{belief}(x_1, \dots, x_t) = \text{belief}(b_{t-1}, x_t) \quad b_t = \text{RNN}(b_{t-1}, x_t)$$

$$p(x_{t+1}, \dots, x_T | x_1, \dots, x_t) \approx p(x_{t+1}, \dots, x_T | b_t)$$

TD-VAE

$$\begin{aligned}\log p(x) &\geq \log p(x) - D_{\text{KL}}(q(z|x) \| p(z|x)) \\ &= \mathbb{E}_{z \sim q} \log p(x|z) - D_{\text{KL}}(q(z|x) \| p(z)) \\ &= \mathbb{E}_{z \sim q} \log p(x|z) - \mathbb{E}_{z \sim q} \log \frac{q(z|x)}{p(z)} \\ &= \mathbb{E}_{z \sim q} [\log p(x|z) - \log q(z|x) + \log p(z)] \\ &= \mathbb{E}_{z \sim q} [\log p(x, z) - \log q(z|x)] \\ \log p(x) &\geq \mathbb{E}_{z \sim q} [\log p(x, z) - \log q(z|x)]\end{aligned}$$

TD-VAE

$$\begin{aligned}
 & \log p(x_t | x_{<t}) \\
 & \geq \mathbb{E}_{(z_{t-1}, z_t) \sim q} [\log p(x_t, z_{t-1}, z_t | x_{<t}) - \log q(z_{t-1}, z_t | x_{\leq t})] \\
 & \geq \mathbb{E}_{(z_{t-1}, z_t) \sim q} [\log p(x_t | \textcolor{red}{z}_{t-1}, z_t, \textcolor{red}{x}_{<t}) + \log p(z_{t-1}, z_t | x_{<t}) - \log q(z_{t-1}, z_t | x_{\leq t})] \\
 & \geq \mathbb{E}_{(z_{t-1}, z_t) \sim q} [\log p(x_t | z_t) + \log p(z_{t-1} | x_{<t}) + \log p(z_t | z_{t-1}) - \log q(z_{t-1}, z_t | x_{\leq t})] \\
 & \geq \mathbb{E}_{(z_{t-1}, z_t) \sim q} [\log p(x_t | z_t) + \log p(z_{t-1} | x_{<t}) + \log p(z_t | z_{t-1}) - \log q(z_t | x_{\leq t}) - \log q(z_{t-1} | z_t, x_{\leq t})]
 \end{aligned}$$

Notice two things:

- The **red** terms can be ignored according to Markov assumptions.
- The **blue** term is expanded according to Markov assumptions.
- The **green** term is expanded to include an one-step prediction back to the past as a smoothing distribution.

TD-VAE

$$\log p(x_t | x_{<t}) \geq \mathbb{E}_{(z_{t-1}, z_t) \sim q} [\log p(x_t | z_t) + \log p(z_{t-1} | x_{<t}) + \log p(z_t | z_{t-1}) - \log q(z_t | x_{\leq t}) - \log q(z_{t-1} | z_t, x_{\leq t})]$$

Precisely, there are four types of distributions to learn:

1. $p_D(\cdot)$ is the **decoder** distribution:
 - $p(x_t | z_t)$ is the encoder by the common definition;
 - $p(x_t | z_t) \rightarrow p_D(x_t | z_t)$;
2. $p_T(\cdot)$ is the **transition** distribution:
 - $p(z_t | z_{t-1})$ captures the sequential dependency between latent variables;
 - $p(z_t | z_{t-1}) \rightarrow p_T(z_t | z_{t-1})$;
3. $p_B(\cdot)$ is the **belief** distribution:
 - Both $p(z_{t-1} | x_{<t})$ and $q(z_t | x_{\leq t})$ can use the belief states to predict the latent variables;
 - $p(z_{t-1} | x_{<t}) \rightarrow p_B(z_{t-1} | b_{t-1})$;
 - $q(z_t | x_{\leq t}) \rightarrow p_B(z_t | b_t)$;
4. $p_S(\cdot)$ is the **smoothing** distribution:
 - The back-to-past smoothing term $q(z_{t-1} | z_t, x_{\leq t})$ can be rewritten to be dependent of belief states too;
 - $q(z_{t-1} | z_t, x_{\leq t}) \rightarrow p_S(z_{t-1} | z_t, b_{t-1}, b_t)$;

TD-VAE

To incorporate the idea of jumpy prediction, the sequential ELBO has to not only work on $t, t + 1$, but also two distant timestamp $t_1 < t_2$. Here is the final TD-VAE objective function to maximize:

$$J_{t_1, t_2} = \mathbb{E}[\log p_D(x_{t_2} | z_{t_2}) + \log p_B(z_{t_1} | b_{t_1}) + \log p_T(z_{t_2} | z_{t_1}) - \log p_B(z_{t_2} | b_{t_2}) - \log p_S(z_{t_1} | z_{t_2}, b_{t_1}, b_{t_2})]$$

Summary

- Convolutional VAE
- Conditional VAE
- Representation learning {
 - β -VAE
 - IWAE
- Hierarchical representation learning {
 - Ladder VAE
 - Progressive + Fade-in VAE
- Temporal representation learning {
 - VAE in speech
 - Temporal Difference VAE (TD-VAE)

Thanks