MSc Data Science & Artificial Intelligence

Deep Learning
β-Variational Autoencoders (β-VAE)

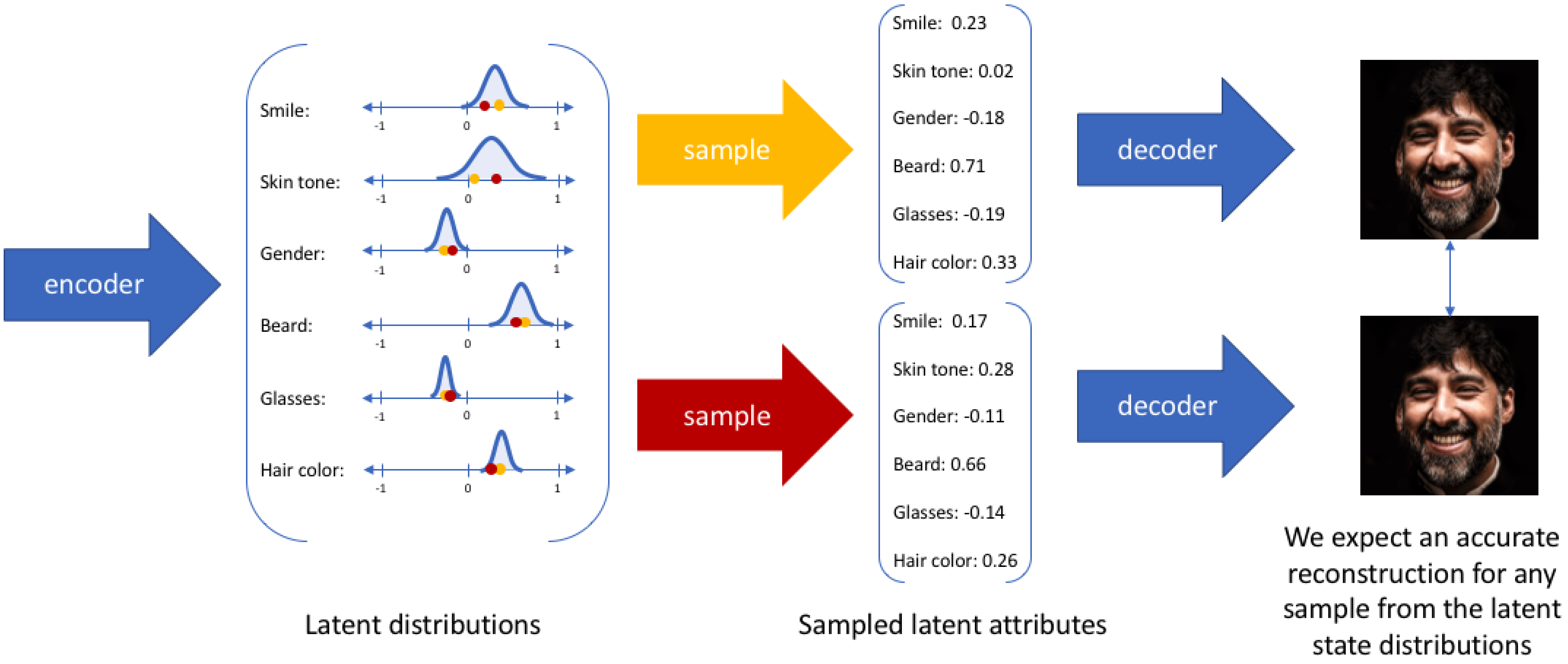
Irina Higgins et al.

Prof. Michel Riveil & Diane Lingrand

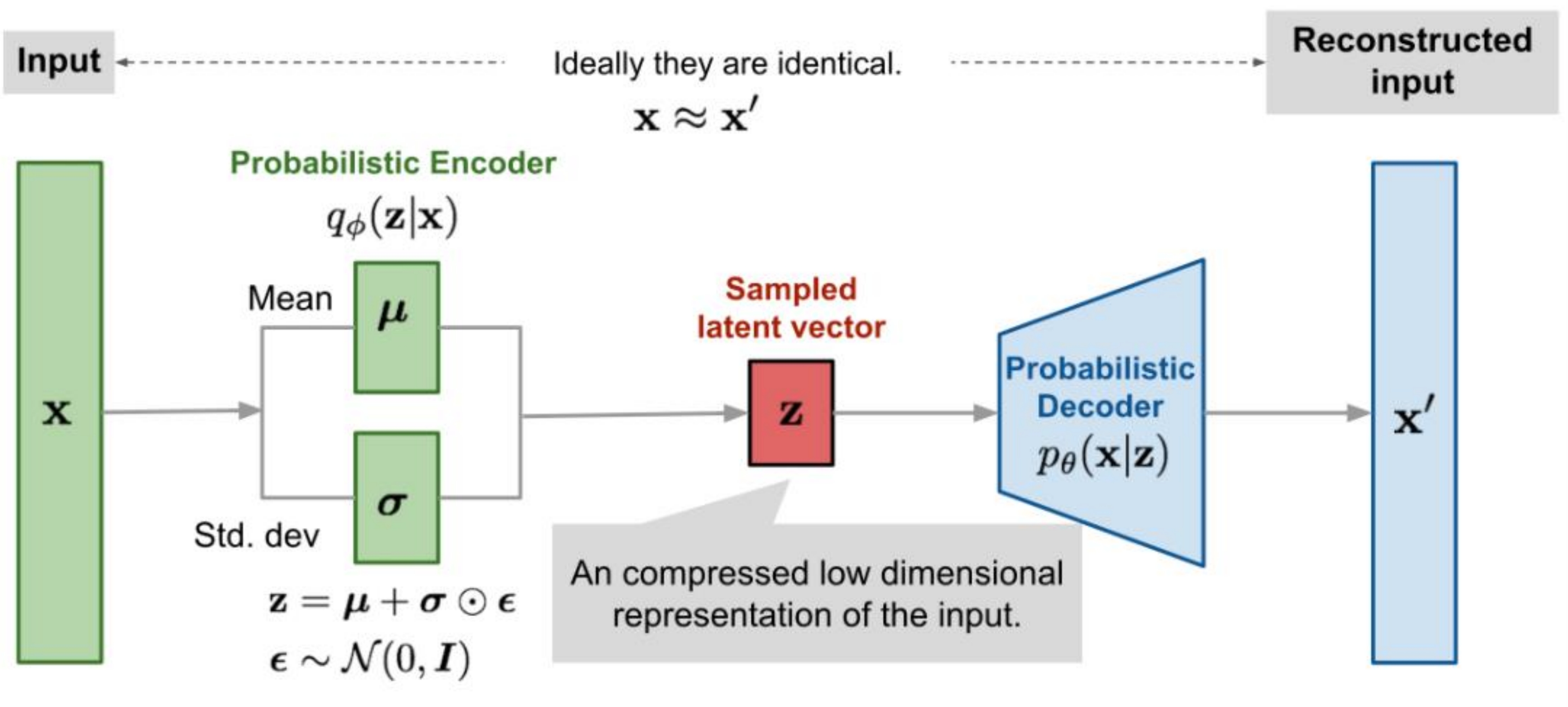
Franz Franco

Motivation

Disentangle Latent space in Variational AutoEncoders



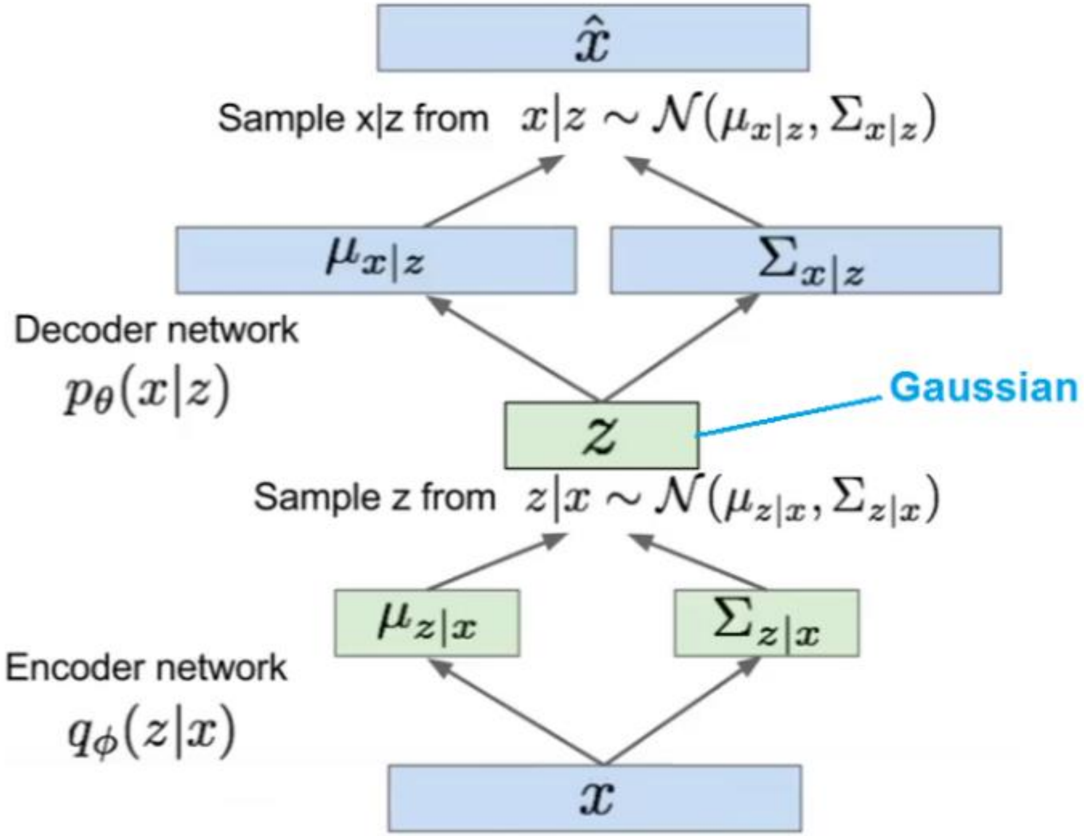
Variational AutoEncoders (VAE)



Drawback: Entangled latent space variables

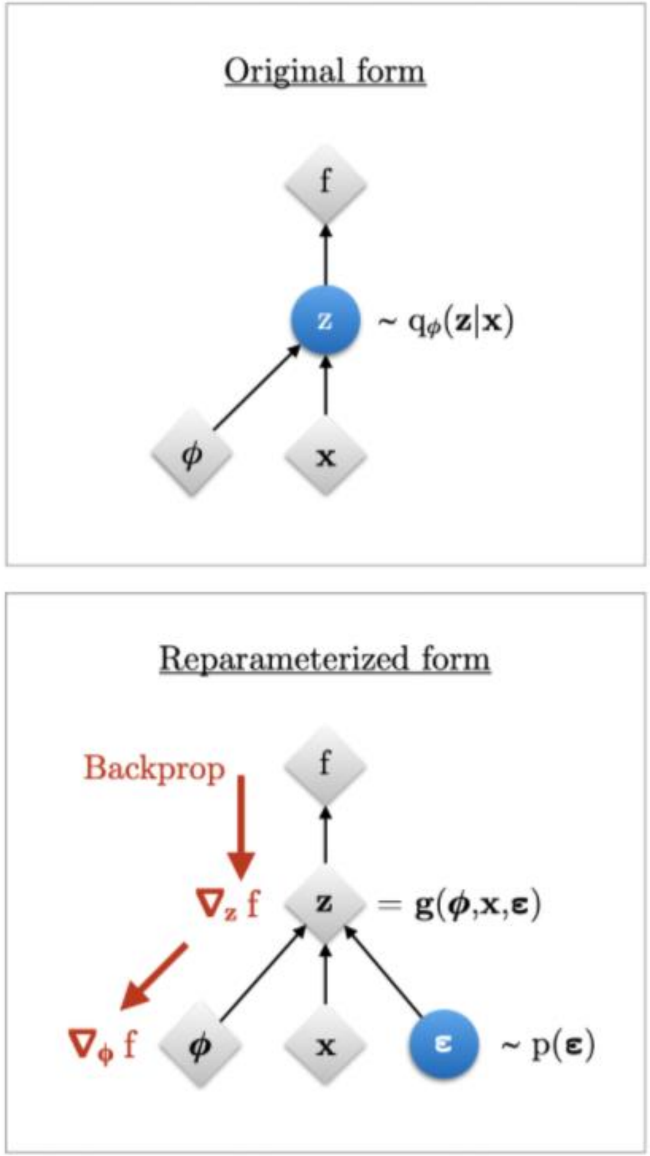
Variational AutoEncoders (VAE)

VAE make a strong assumption that the original input X and the latent vector z both have isotropic gaussian distribution



Drawback: Entangled latent space variables

Reparametrization Trick



$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$$
$$\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$$

Variational AutoEncoders (VAE)

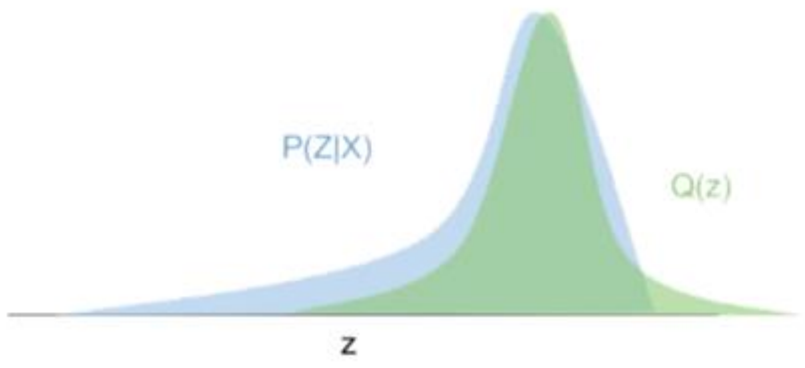
- **Variational Methods: Mean-Field Approximation**

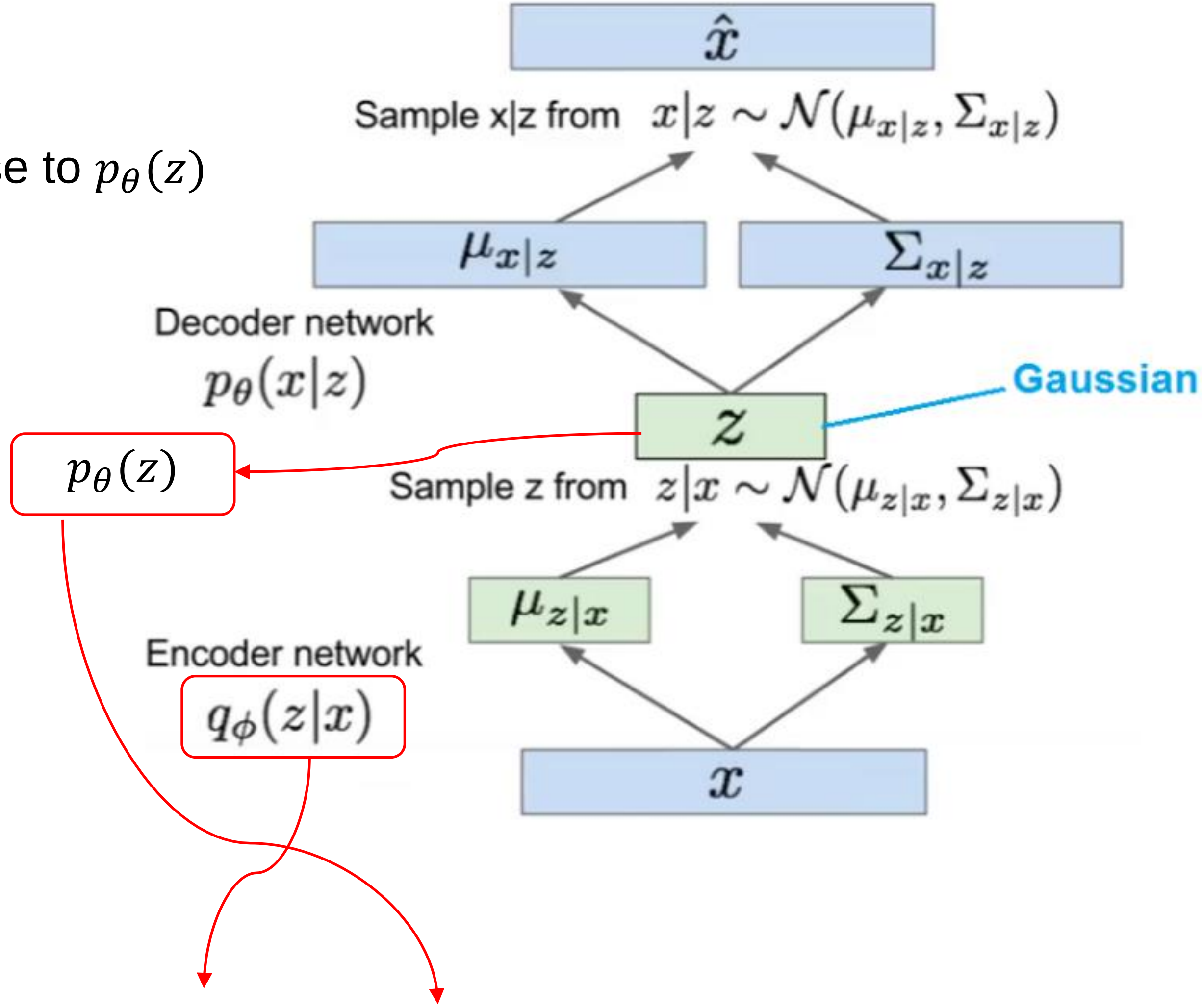
- Loss Function: Evidence Lower Bound (ELBO)

  The estimated posterior $q_\phi(z|x)$ should be very close to $p_\theta(z)$

- Kullback-Leibler divergence

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$





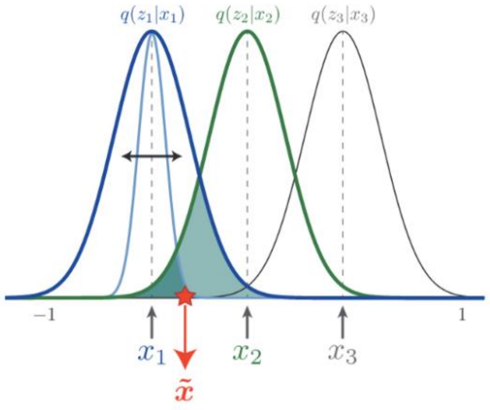
$$\mathcal{L} = \mathbb{E}_{q(z|X)}[\log p(X|z)] - D_{KL}[q(z|X)||p(z)]$$

β-Variational AutoEncoders(β-VAE)

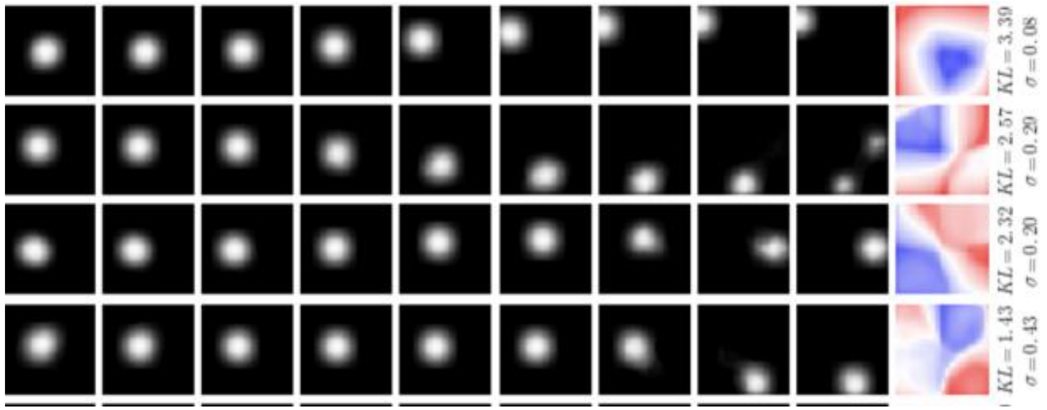
The difference between β-VAE and VAE is the use of Lagrange multiplier β on the KL divergence term in the original VAE formulation:

VAE $$\mathcal{L} = \mathbb{E}_{q(z|X)}[\log p(X|z)] - D_{KL}[q(z|X)||p(z)]$$

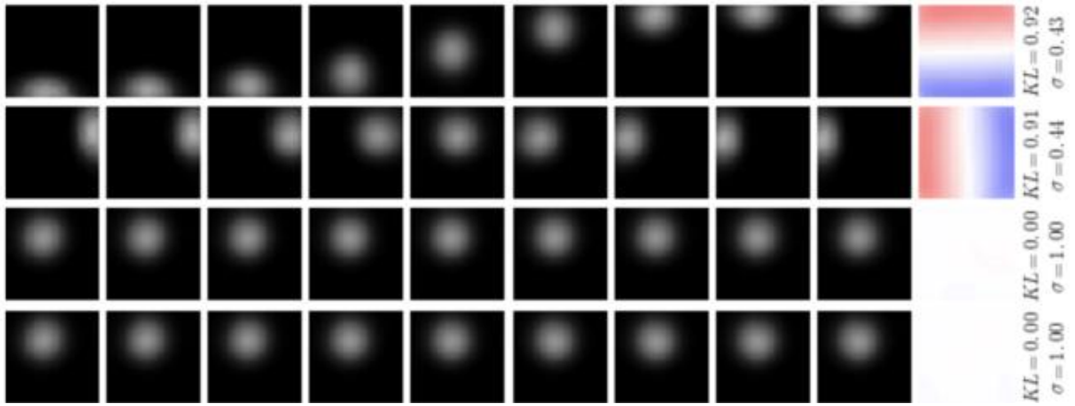
β-VAE $$\mathcal{L} = \mathbb{E}_{q(z|X)}[\log p(X|z)] - \beta D_{KL}[q(z|X)||p(z)]$$



VAE

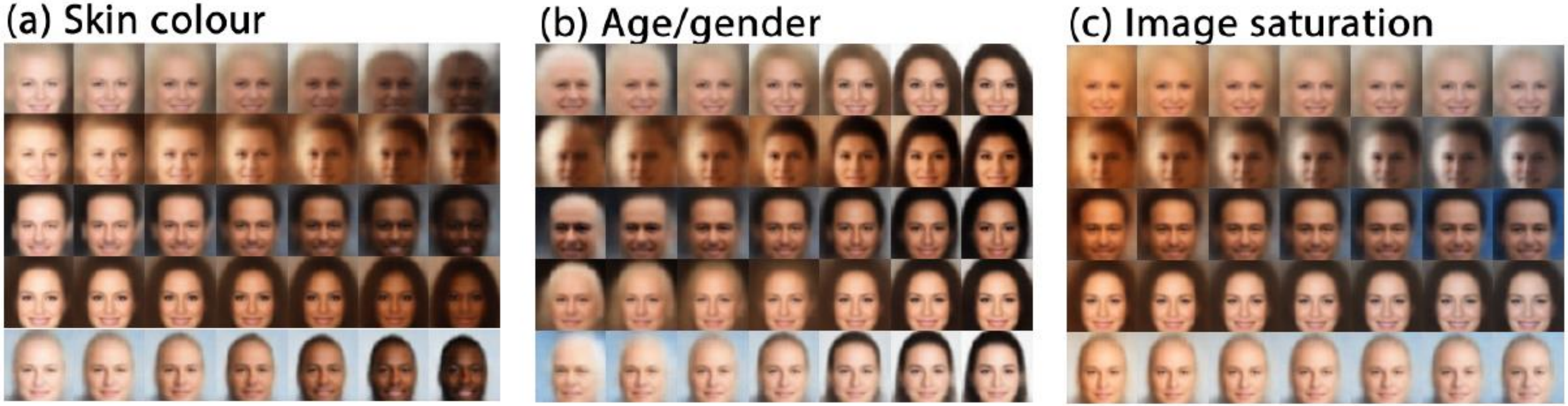


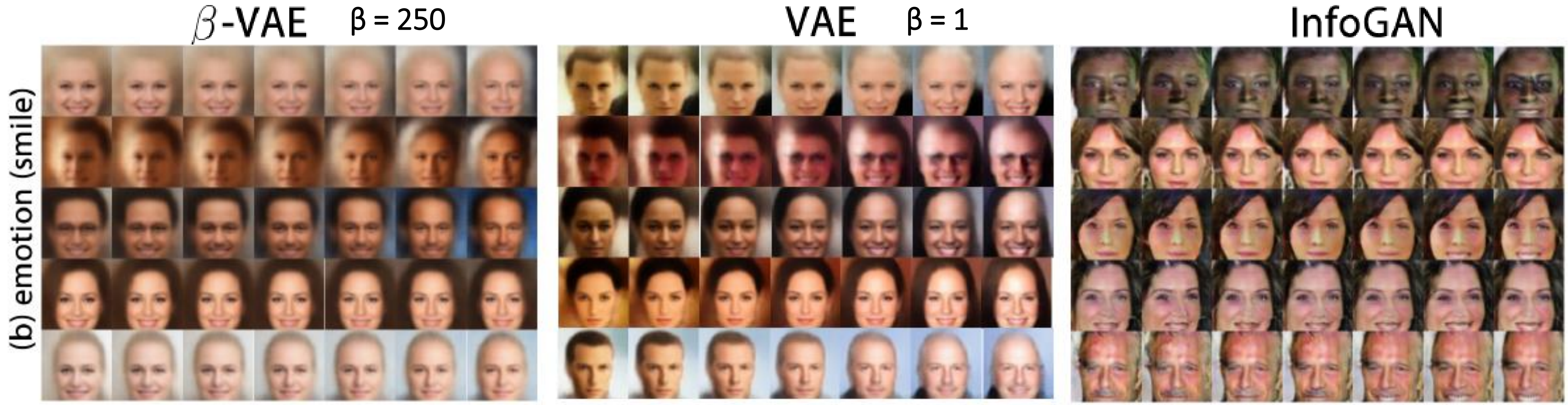
β -VAE



- higher β may create a trade-off between reconstruction quality and the extent of disentanglement.
- When β=1, it is same as VAE. When β>1, it applies a stronger constraint on the latent bottleneck and limits the representation capacity of z

β-Variational AutoEncoders (β-VAE)





Summary

- Autoencoder behaves as a deterministic system and can not go beyond our data, any other data which is not in our domain will get bad results.

- VAE can be defined as being an autoencoder whose training is regularized to avoid overfitting and ensure that the latent space has good properties that enable generative process.

- VAE make a strong assumption that the original input X and the latent vector z both have isotropic gaussian distribution.

- VAE are autoencoders that tackle the problem of the latent space irregularity by making the encoder return a distribution over the latent space instead of a single point and by adding in the loss function a regularisation term over that returned distribution in order to ensure a better organisation of the latent space

- β-VAE encourages more efficient latent encoding and further encourages the disentanglement, when $\beta=1$, it is same as VAE. When $\beta>1$, it applies a stronger constraint on the latent bottleneck and limits the representation capacity of z. For some conditionally independent generative factors, keeping them disentangled is the most efficient representation. Meanwhile, a higher β may create a trade-off between reconstruction quality and the extent of disentanglement.

APPENDIX

Variational AutoEncoders (VAE)

In our case we want to minimize: $D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))$

$$
\begin{aligned}
&= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})} d\mathbf{z} \\
&= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}) p_\theta(\mathbf{x})}{p_\theta(\mathbf{z},\mathbf{x})} d\mathbf{z} && \text{; Because } p(z|x)=p(z,x)/p(x) \\
&= \int q_\phi(\mathbf{z}|\mathbf{x}) \big(\log p_\theta(\mathbf{x}) + \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z},\mathbf{x})} \big) d\mathbf{z} \\
&= \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z},\mathbf{x})} d\mathbf{z} && \text{; Because } \int q(z|x)dz=1 \\
&= \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z})} d\mathbf{z} && \text{; Because } p(z,x)=p(x|z)p(z) \\
&= \log p_\theta(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})} - \log p_\theta(\mathbf{x}|\mathbf{z})] \\
&= \log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) - \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})
\end{aligned}
$$

Variational AutoEncoders (VAE)

So we have:

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

Rearrange a bit the left and right side of the equation:

$$\log p_\theta(\mathbf{x}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}))$$

$$\begin{aligned} L_{\mathrm{VAE}}(\theta,\phi) &= -\log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})) \\ &= -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) \\ \theta^*,\phi^* &= \arg\min_{\theta,\phi} L_{\mathrm{VAE}} \end{aligned}$$

- L_{VAE} is the lower bound of $log p_\theta(x)$:

$$\begin{aligned} -L_{\mathrm{VAE}} &= \log p_\theta(\mathbf{x}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})) \leq \log p_\theta(\mathbf{x}) \\ &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) \end{aligned}$$

Therefore by minimizing the loss, we are maximizing the lower bound of the probability of generating real data samples.