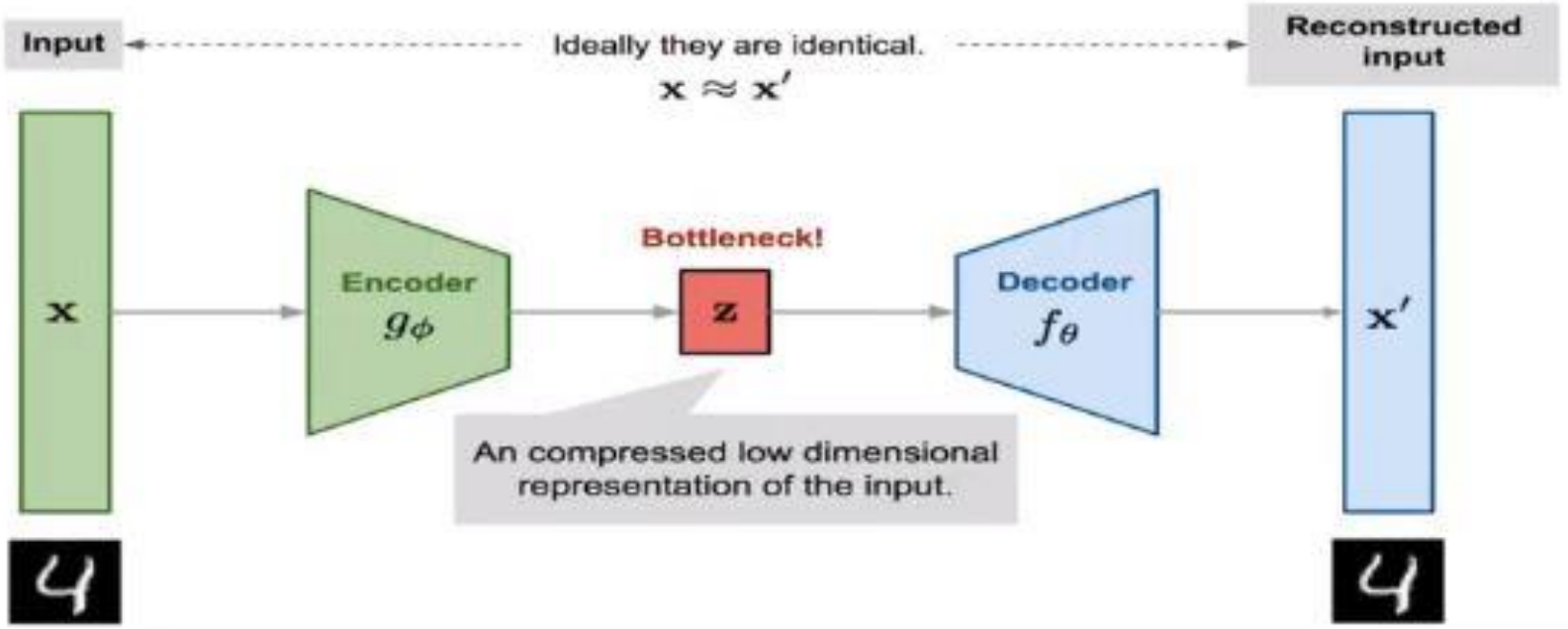
Variational Autoencoders

1. Review of Stacked Autoencoders
2. Basics of Probability
3. KL Divergence & its significance
4. Derivation of Loss function for Variational Autoencoders

Stacked Autoencoders

(Image Reconstruction)



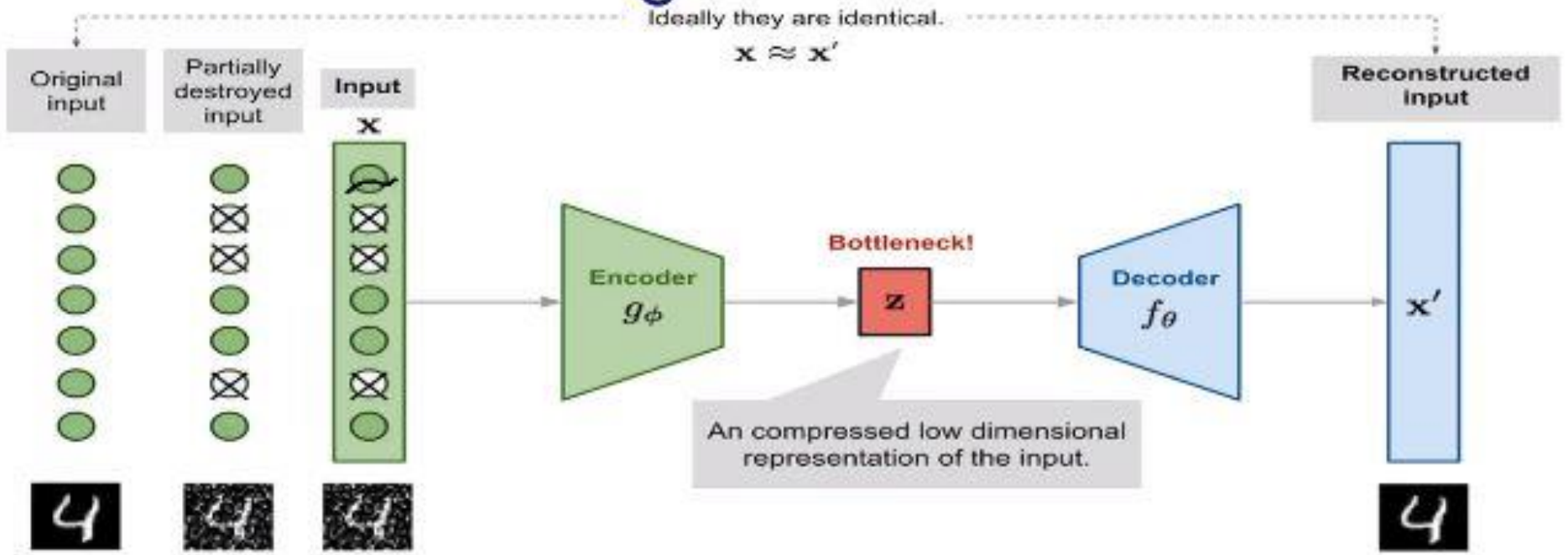
Cost function: $L(\theta, \phi) = \frac{1}{n} \sum_{i=1}^{n} [x^{(i)} - f_\theta(g_\phi(x^{(i)}))]^2$

⊛ from lilianweng github account

11

DL_16

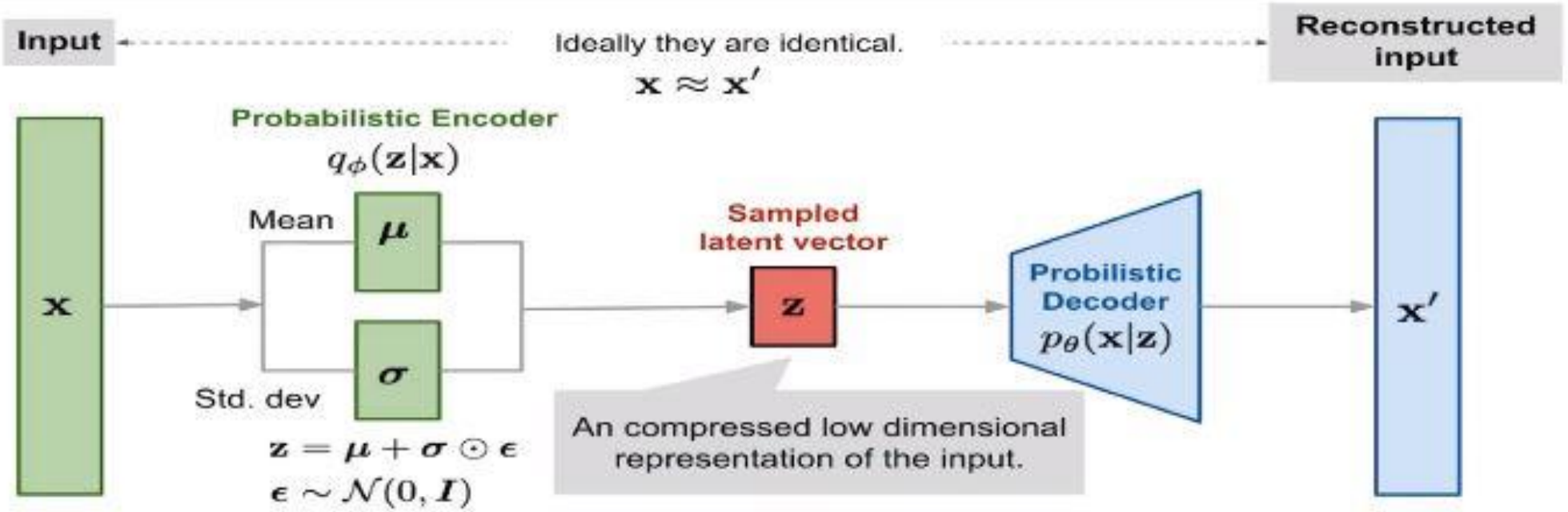
Denoising Autoencoder



$$\tilde{x}^{(i)} \sim \mathcal{N}(\tilde{x}^{(i)} \mid x^{(i)})$$

$$\text{Loss: } L(\theta, \phi) = \frac{1}{n} \sum_{i=1}^{n} \left[x^{(i)} - f_\theta(g_\phi(\tilde{x}^{(i)})) \right]^2$$

Variational Autoencoders



$$\text{Loss} = L(\theta, \phi) = -E_{z \sim q_\phi(z|x)}\left[p_\theta(x|z)\right] + D_{KL}\left(q_\phi(z|x) \| p_\theta(z)\right)$$

Pre-requisite

VA

① Probability
- $p(x)$
- $p(x/y)$
- $E[\cdot]$
- KL Divergence

$p(x)$: defines the probability of random variable x

$p(x|y)$: defines as the probability of random variable x provided Y has happened
Also called as conditional probability

likelihood ratio

prior probability

$$p(y|x) = \frac{p(x|y)\,P(y)}{P(x)} \rightarrow \text{Baye's Theorm}$$

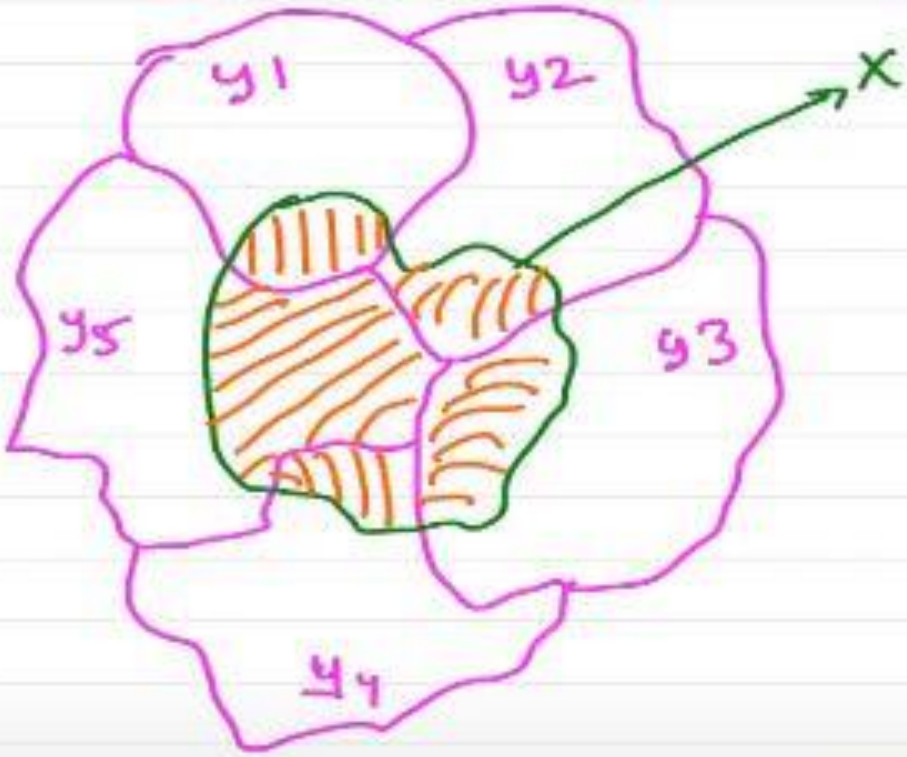
posterior probability

$$= \frac{P(x,y)}{P(x)} \rightarrow \text{joint distribution} \quad \text{—— (1)}$$

→ Theorem of Total probability.

Let $y_1, y_2, \ldots y_N$ be a set of mutually exclusive events (i.e $y_i \wedge y_j = 0$) & event X is the union of N mutually exclusive events, then

$$P(X) = \sum_{i=1}^{N} P(x/y_i)\,P(y_i) \quad \text{—— (2)}$$



$$P(x) = \sum_{i=1}^{4} P(x, y_i)$$

$$= \sum_{i=1}^{4} P(x/y_i)\, P(y_i)$$

(X)

$y_1, y_2 \cdots y_4$

So substituting ② in ① results in

$$\left\{ P(y/x) = \frac{P(x/y)\,P(y)}{\sum_{i=1}^{N} P(x/y_i)\,P(y_i)} \right\}$$

Expectation of random variable X i.e E(X)

Expected value of random variable is a weighted average of the possible values of X can take, each value being weighted according to the probability of that

event defined as

$$E(x) = \sum_{i=1}^{k} x_i \, P(X = x_i)$$

Q When a die is tossed once. What is the probability of getting 3.

Ans Sample space = $\{1, 2, 3, 4, 5, 6\}$, $P(3) = \frac{1}{6}$

Q2 In tossing a fair die, what is the probability the 3 has occured conditioned on the toss being odd.

A Since, we are given that odd number has occured the sample space reduces from $\{1, 2, 3, 4, 5, 6\}$ to $\{1, 3, 5\}$. Hence the probability of 3 in this reduced sample space is $\frac{1}{3}$. It can be observed there is increase in the probability compared to the earlie case. Why ?

Q3 Let X represent the outcome of a fair six sided die. What is the E(X)?

Ans $X = \{1, 2, 3, 4, 5, 6\}$ $P(X) = 1/6$

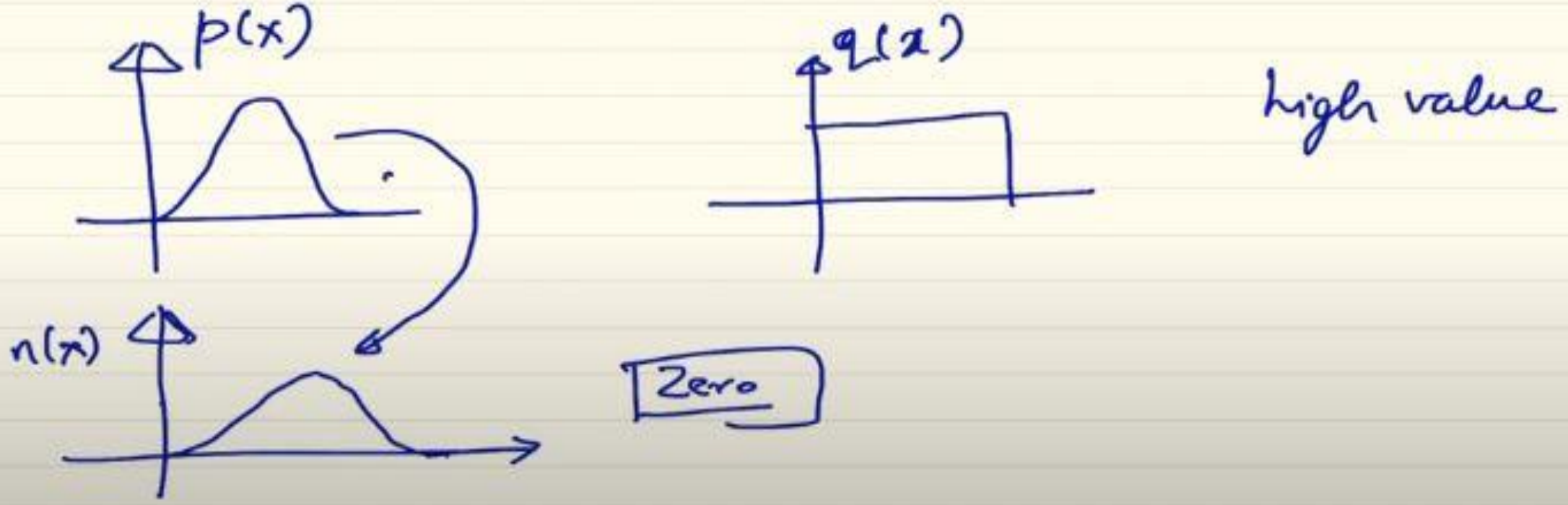
$$E(X) = \sum_{i=1}^{6} x\, P(X) = 1 \times \frac{1}{6} + 2/6 + 3 \times 1/6 + 4 \times \frac{1}{6} + 5 \times \frac{1}{6} + 6 \times \frac{1}{6} = 3.5$$

K-L Divergence

(#)

Kullback-Leibler divergence (K-L) is measure of how one probability distribution is different from the second. For the discrete probability distribution P & Q, the K-L divergence between P & Q is defined as

K-L Divergence

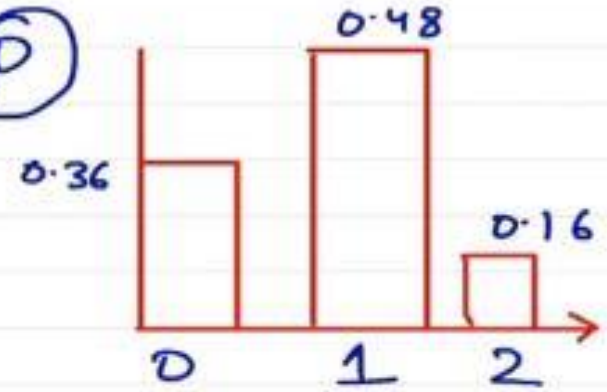


$$\rightarrow D_{KL}(P \| Q) = \sum_x P(X=x) \log\left(\frac{P(X=x)}{Q(X=x)}\right)$$

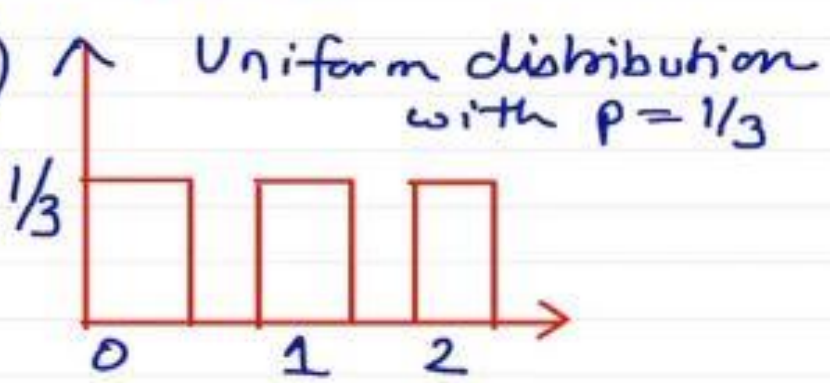
$$\equiv \sum_x P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

Example:

(P)



(Q)



$$D_{KL}(Q \| P) = \frac{1}{3} \ln\left(\frac{0.333}{0.36}\right) + \frac{1}{3} \ln\left(\frac{0.333}{0.48}\right) + \frac{1}{3} \ln\left(\frac{0.333}{0.16}\right)$$

$$= 0.09637 \text{ nats}$$

Properties :

① $KL(P||Q)$ or $KL(Q||P) \geq 0$

② $KL(P||Q) \neq KL(Q||P)$ (Not Symmetric)

Suppose we have two multivariate normal distributions defined as

$$p(x) = N(x; \mu_1, \Sigma_1)$$
$$q(x) = N(x; \mu_2, \Sigma_2)$$

where μ_1 & μ_2 are the means & Σ_1, Σ_2 are the Covariance matrix

And the multivariate normal density is defined as

$$N(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k |\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

if the two distributions have the same dimension k.

$$D_{KL}(p(x) \| q(x)) = \frac{1}{2}\left[\log \frac{|\Sigma_2|}{|\Sigma_1|} - d + \mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + (\mu_2-\mu_1)^T \Sigma_2^{-1} (\mu_2-\mu_1)\right]$$

Prove ?

Proof: We know

$$KL(P(x)||Q(x)) = \sum_x P(x) \log\left(\frac{P(x)}{Q(x)}\right) \quad \text{— (1)}$$

We know

$$P(x) = \frac{1}{\sqrt{(2\pi)^K |\Sigma_1|}} \exp\left(\frac{-(x-\mu_1)^T \Sigma_1^{-1} (x-\mu_1)}{2}\right)$$

$$\Rightarrow \log P(x) = -\frac{K}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma_1| - \frac{1}{2}(x-\mu_1)^T \Sigma_1^{-1}(x-\mu_1) \quad \text{— (2)}$$

Similarly

$$\log Q(x) = -\frac{K}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma_2| - \frac{1}{2}(x-\mu_2)^T \Sigma_2^{-1}(x-\mu_2) \quad \text{— (3)}$$

Eq ① can be rewritten as

$$KL(P(x)\,||\,Q(x)) = \sum_x P(x)\log P(x) - \log(Q(x))$$

Substituting ② & ③ in ① results in

$$KL(P(x)\,||\,Q(x)) = \sum_x p(x)\left\{-\frac{k}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma_1|\right.$$

$$-\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) + \frac{k}{2}\log(2\pi) + \frac{1}{2}\log|\Sigma_2|$$

$$\left.+\frac{1}{2}(x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\right\}$$

which on simplification results in

$$KL\left(P(x) \| Q(x)\right) =$$

$$+ \sum_x p(x) \left\{ \frac{1}{2} \log\left[\frac{\Sigma_2}{\Sigma_1}\right] + \frac{1}{2} (x-\mu_2)^T \Sigma_2^{-1} (x-\mu_2) \right.$$

$$\left. - \frac{1}{2} (x-\mu_1)^T \Sigma_1^{-1} (x-\mu_1) \right\} \quad \text{—} \quad (1)$$

Now, let consider part by part

$$\sum_x p(x) \frac{1}{2} (x-\mu_1)^T \Sigma^{-1} (x-\mu_1) \equiv E_p\left[\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right]$$

$$\rightarrow E(x^T A x) = E(tr(x^T A x)) \quad -(a)$$

$$= E(tr(A x x^T)) \quad -(c)$$

$$= tr(E(A x x^T)) \quad -(e)$$

Let's rewrite again

$$\frac{1}{2} E_p\left[\overbrace{(x-\mu_1)^T \Sigma_1^{-1} (x-\mu_1)}^{\text{scalar}}\right]$$

$$E_p\left[tr\left(\frac{1}{2}(x-\mu_1)^T \Sigma_1^{-1} (x-\mu_1)\right)\right]$$

$$E_p\left[tr\left(\frac{1}{2}(x-\mu_1)(x-\mu_1)^T \Sigma_1^{-1}\right)\right] \rightarrow (d)$$

Trace & Expectation trick.

→ if x is scalar

then $E(x) \overset{(a)}{=} E(tr(x))$

Since trace of x is scalar

→ $tr(AB) \overset{(b)}{=} tr(BA)$

→ $tr(ABC) \overset{(c)}{=} tr(BCA) \overset{(d)}{=} tr(CAB)$

→ $tr(ABC) \neq tr(ACB)$

$E(tr(x)) = tr(E(x))$ (e)

$$\rightarrow tr\left[E_p\left(\frac{1}{2}(x-\mu_1)(x-\mu_1)^T\Sigma_1^{-1}\right)\right] \quad \text{—(e)}$$

$$\rightarrow tr\left\{E_p\left[(x-\mu_1)(x-\mu_1)^T\right]\frac{1}{2}\Sigma_1^{-1}\right\}$$

$E_p\left[(x-\mu_1)(x-\mu_1)^T\right]$ → Covariance matrix

$$\Rightarrow tr\left[\Sigma_1 \frac{1}{2}\Sigma_1^{-1}\right]$$

$$tr\left[I_k\right] \equiv k \quad \text{— (2)}$$

Now Consider the second part

$$\sum_x p(x)\left[\frac{1}{2}(x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\right]$$

$$\sum_x p(x)\left\{\frac{1}{2}\left[(x-\mu_1)+(\mu_1-\mu_2)\right]^T\Sigma_2^{-1}\left[(x-\mu_1)+(\mu_1-\mu_2)\right]\right\}$$

$$\sum_x p(x)\left[\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu_1)+\frac{2}{2}(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right.$$
$$\left.+\frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

$$\Rightarrow E_p\left[\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu)+(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right.$$
$$\left.+\frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

Now Consider the second part

$$\sum_x p(x)\left[\frac{1}{2}(x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\right]$$

$$(A+B)^T\Sigma_2^{-1}[A+B]$$

$$(A^T+B^T)\Sigma_2^{-1}(A+B)$$

$$A^T\Sigma_2^{-1}A$$

$$\sum_x p(x)\left\{\frac{1}{2}[(x-\mu_1)+(\mu_1-\mu_2)]^T\Sigma_2^{-1}[(x-\mu_1)+(\mu_1-\mu_2)]\right\}$$

$$B^T\Sigma_2^{-1}B$$

$$\sum_x p(x)\left[\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu_1)+\frac{2}{2}(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)+\frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

$$\begin{cases}B^T\Sigma_2^{-1}A\\A^T\Sigma_2^{-1}B\end{cases}$$

$$\Rightarrow E_p\left[\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu)+(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)+\frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

Expanding we get

$$E_p\left\{\frac{1}{2}(x-\mu_1)^T\Sigma_2^{-1}(x-\mu_1)\right\} + E_p\left[(x-\mu_1)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

$$+ E_p\left[(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

Constant

$$= \text{tr}\left\{\frac{\Sigma_2^{-1}\Sigma_1}{2}\right\} + (\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2) + 0$$

similar to earlier derivation

$E[\text{Constant}] = \text{Constant}$

proved on next slide

(B)

$$\rightarrow E_p\left[(x-\mu_1)^T \Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

$$\left[(E_p(x)-\mu_1)^T \Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

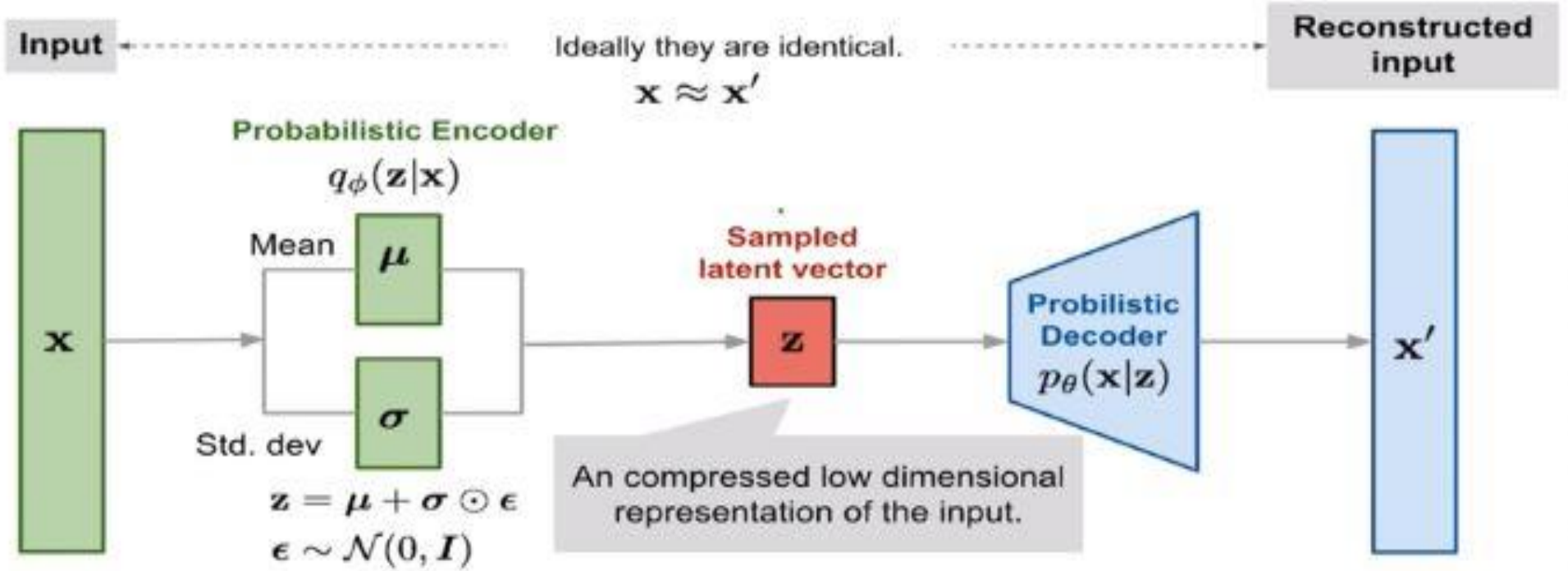
$$\equiv (\mu_1-\mu_1)^T \Sigma_2^{-1}(\mu_1-\mu_2) = 0 \quad \text{(Proved)}$$

So Substituting ②, ④ in ① we obtain

$$KL(P(x)\|Q(x)) = \frac{1}{2}\left[\log\frac{|\Sigma_2|}{|\Sigma_1|} - K + tr(\Sigma_2^{-1}\Sigma_1) + (\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)\right]$$

Proved

Variational Autoencoder

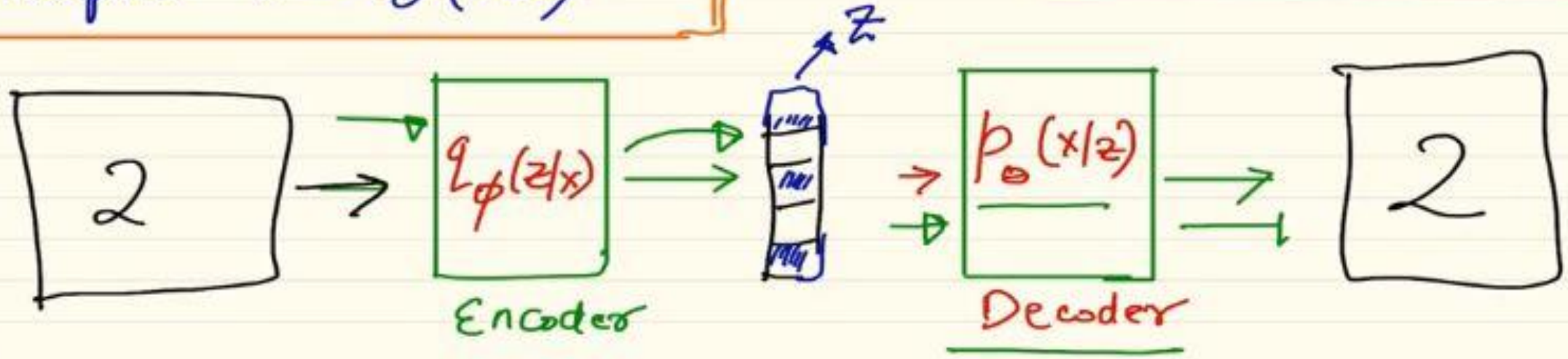


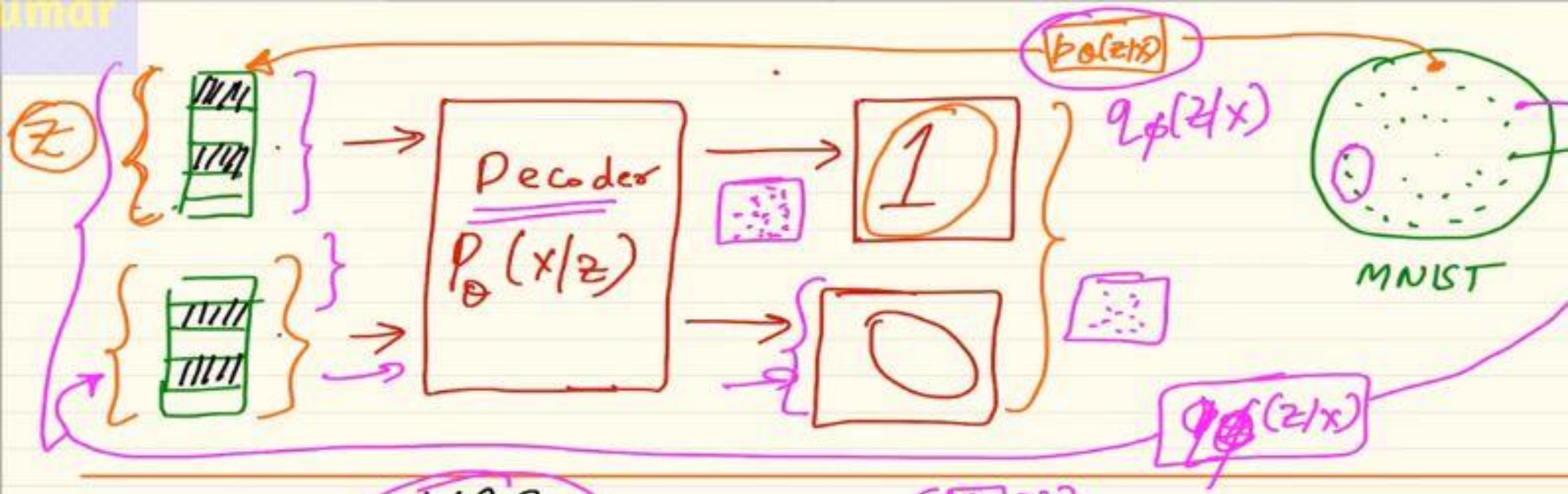
$$\text{Loss} = L(\theta, \phi) = -E_{z \sim q_\phi(z|x)}\left[p_\theta(x|z)\right] + D_{KL}\left(q_\phi(z|x) \| p_\theta(z)\right)$$

The goal of VAE

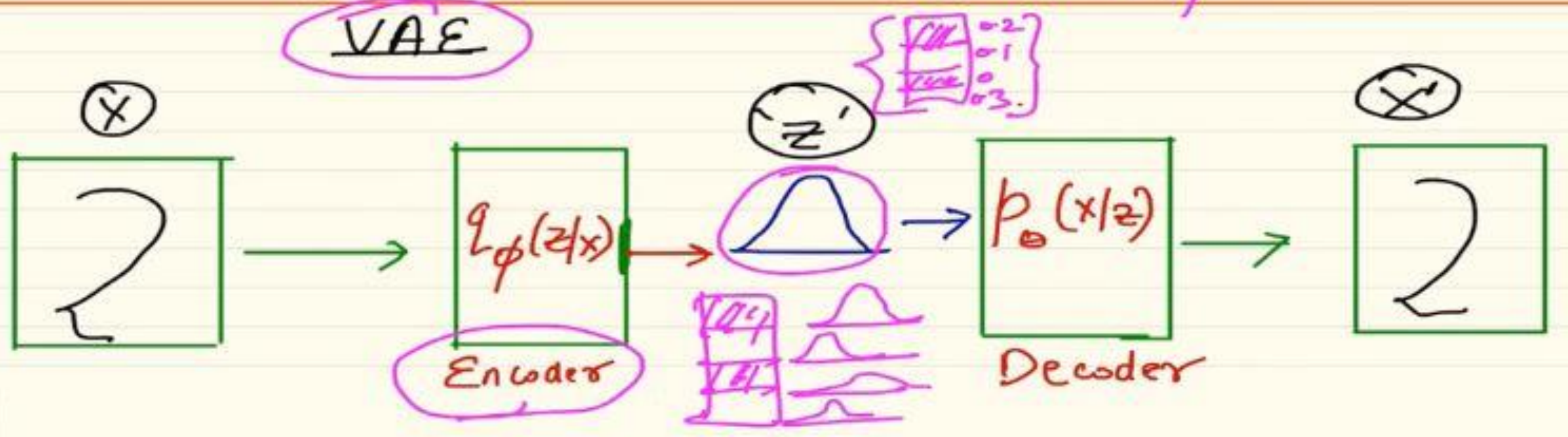
The goal of VAE is to find a distribution $q_\phi(z|x)$ of some latent variables which we can sample from $z \sim q_\phi(z|x)$ to generate new samples $x' \sim p_\theta(x|z)$

Typical Autoencoder




z
Decoder
$p_\theta(x|z)$
$p_\theta(z|x)$
$q_\phi(z|x)$
MNIST
$q_\phi(z|x)$

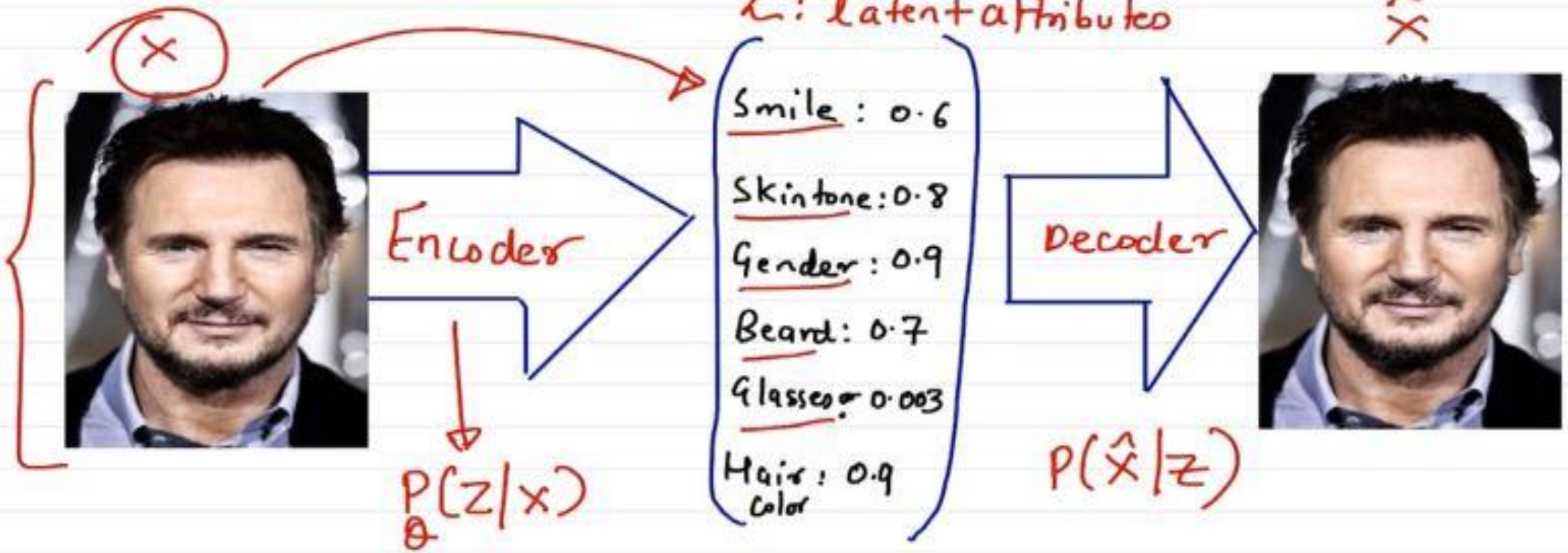
VAE


x
$q_\phi(z|x)$
Encoder
z
$p_\theta(x|z)$
Decoder
x'

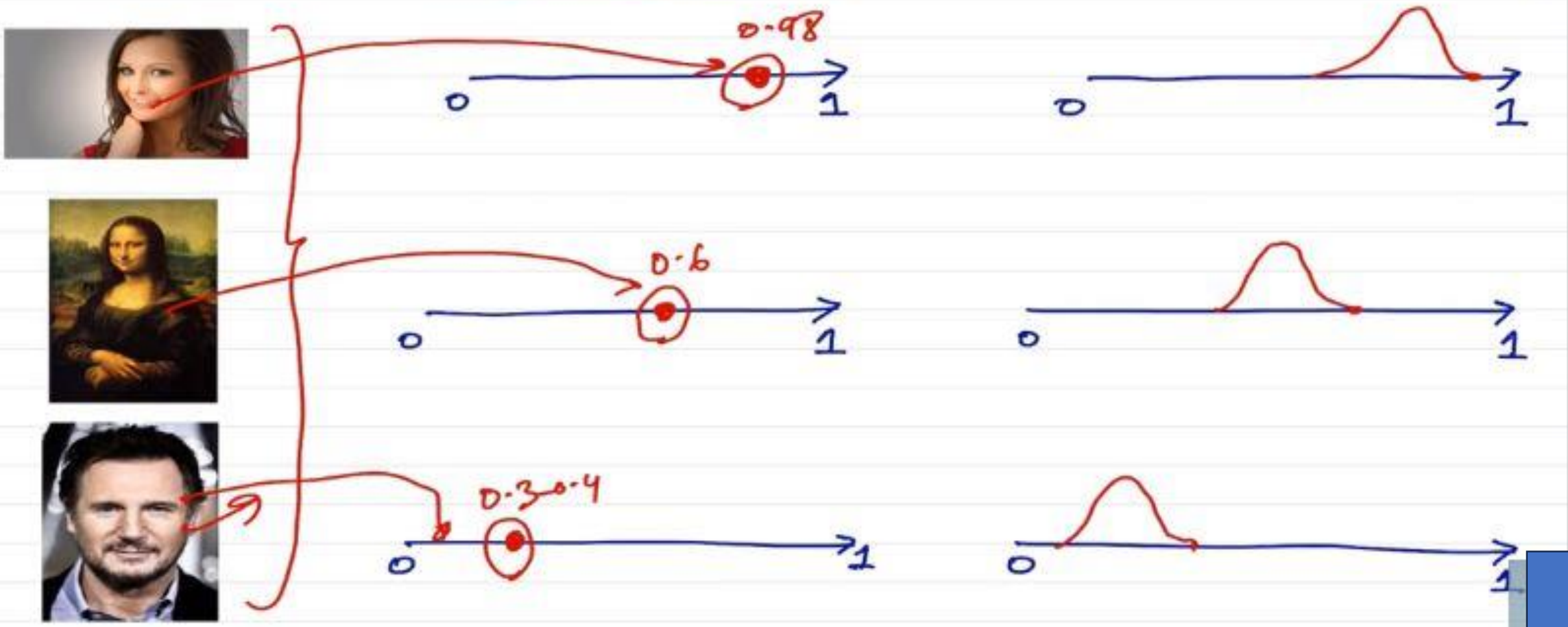
~~Latent variables can be placed in 2 categories~~

Latent (2)

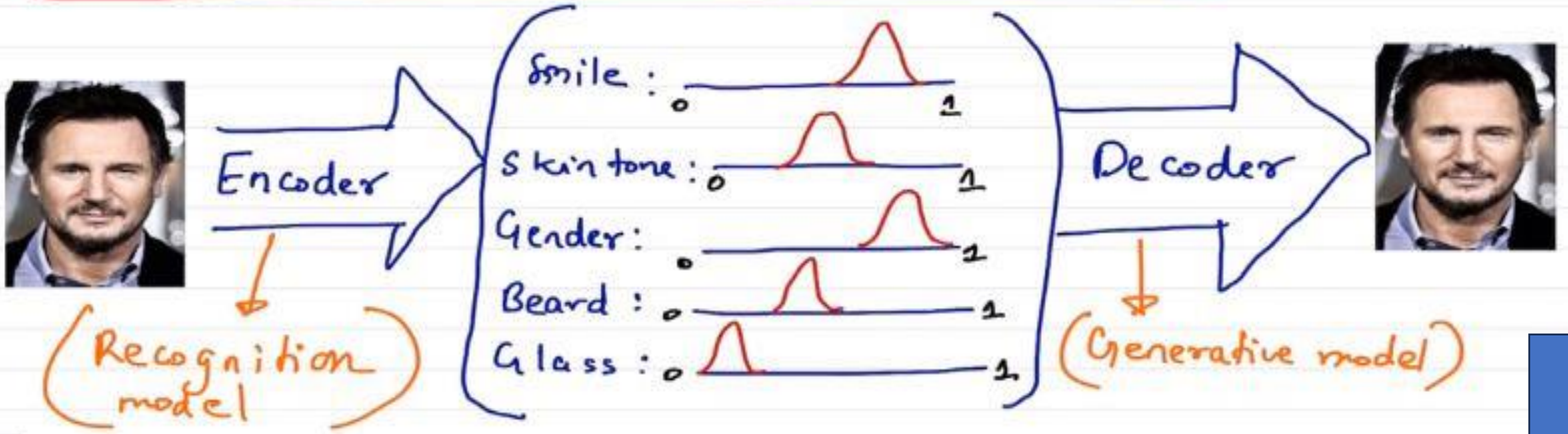
(1) Variables corresponding to a real feature of the object that have not been measured (may be technology is not available to do that)

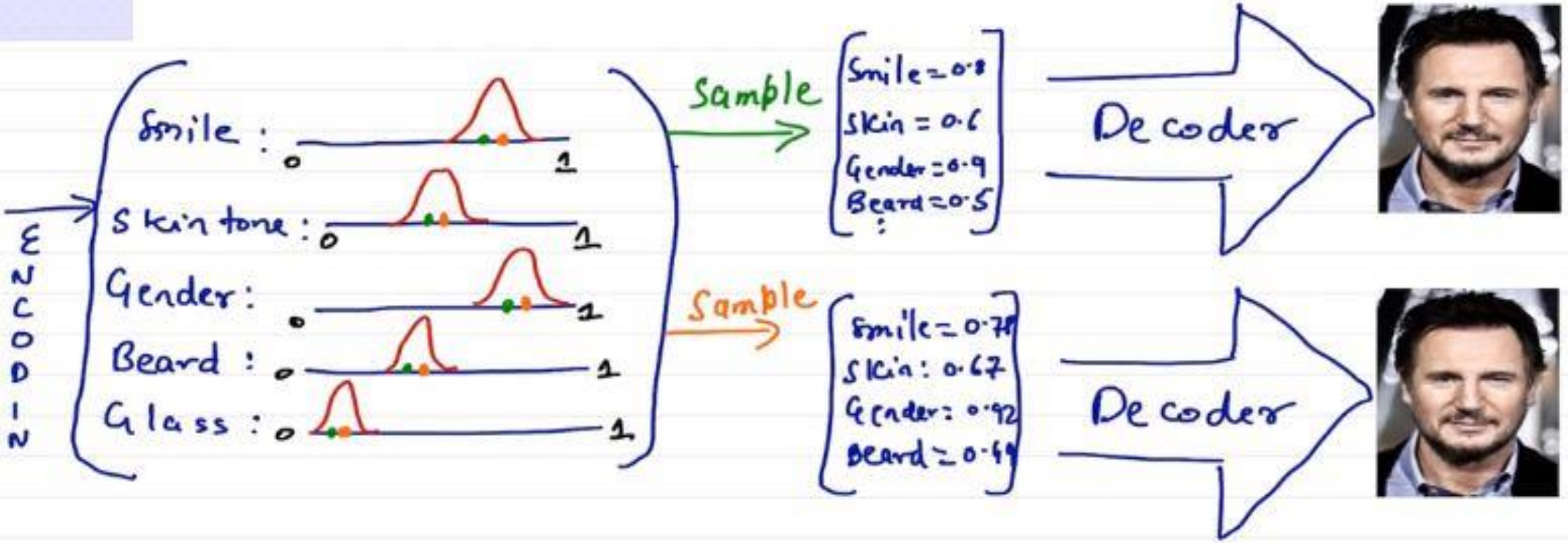


Using VE, we define latent attributes in probablistic terms.



With this approach, we now represent each latent attribute for a given input as a probability distribution. When decoding, we will randomly sample from each latent state distribution to generate a vector as i/p for our decoder model.





By Constructing our encoder model to output range of possible values (a statistically distribution) from which we will randomly Sample to feed into our decoder model. The the values which are near by each other in latent space must correspond to similar reconstruct