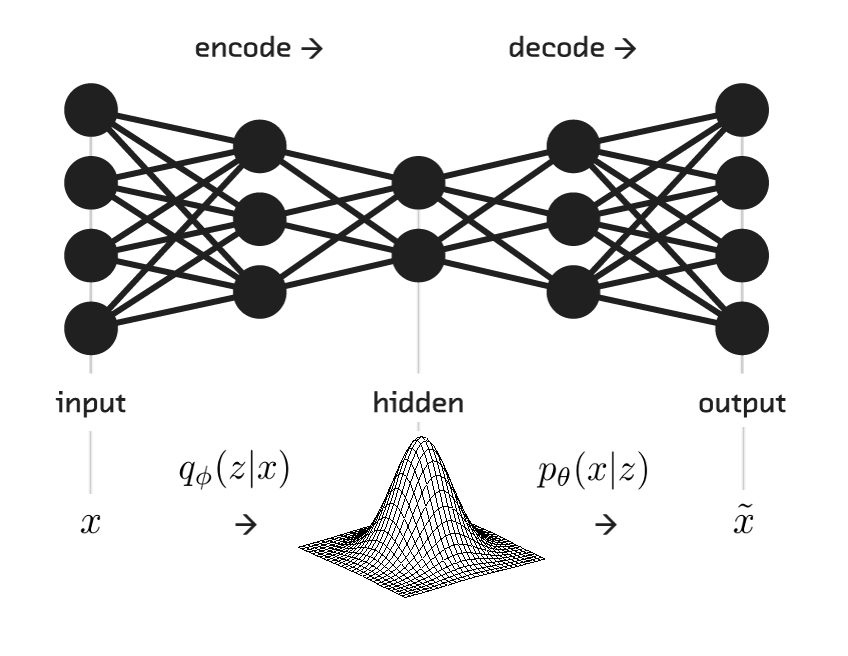
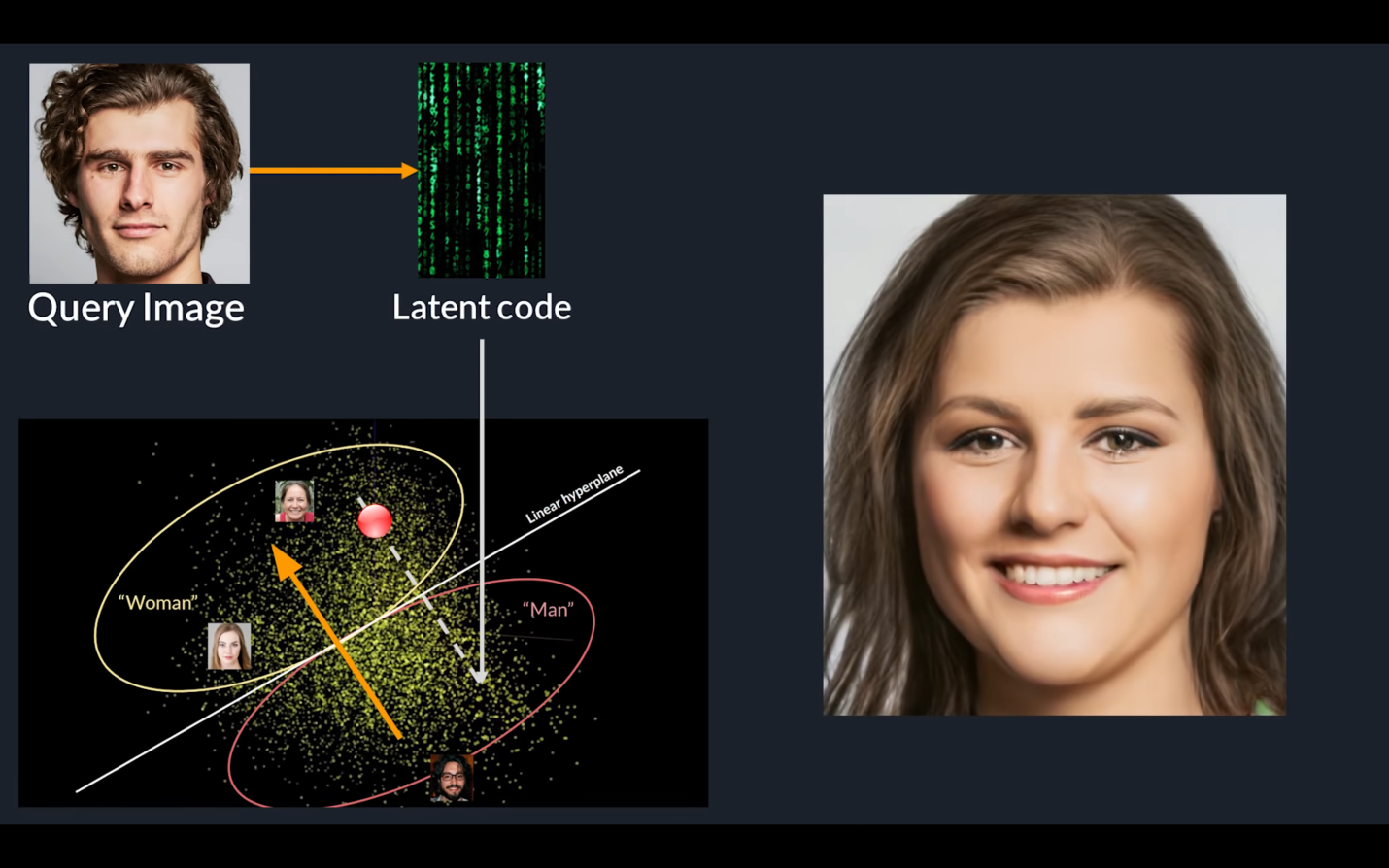
**THEORETICAL RESULT**

The goal of VAE is to find a distribution Q(ᵶ|x) of some latent variables ᵶ from which we can sample z ̴ Q(ᵶ|x) to generate new samples x’ ̴ P(x|z)

These are completely new samples but follows the existing input data distribution thus it manages to produce now samples

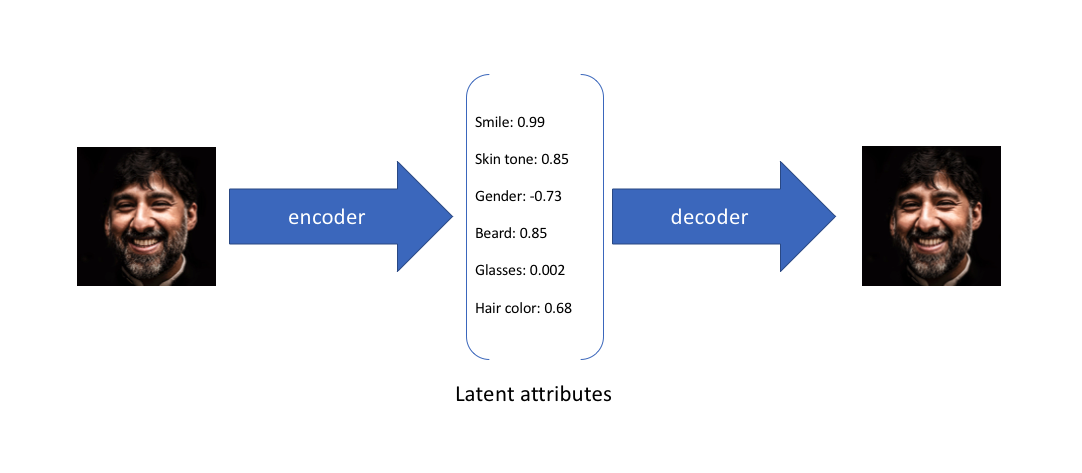
For better understanding why VAE is important we should first know what are problems that the VAE try to address which were existed in the earlier models for example auto encoders



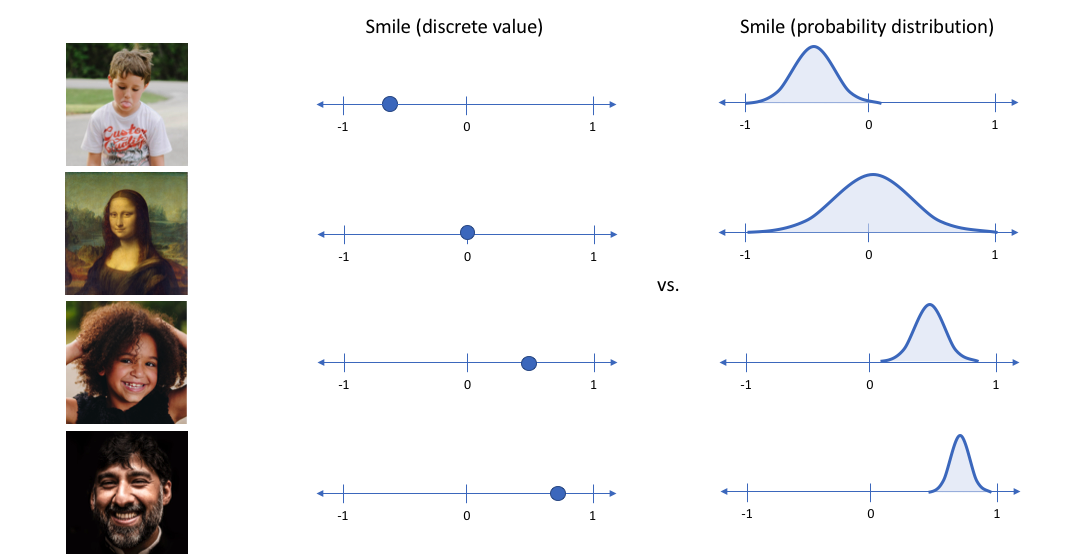
Auto encoders are the models which are used to get low level representation of data of a real data. Higher level representation of the input data is fed into the encoder model Q(ᵶ|x) which convert it into lower level representation Z (latent variable). It is very useful in term sof congestion control by the effective utilization of bandwidth in data transmission. The original image can be recovered by feeding the compressed lower representation Z into a decoder. Auto encoder are capable enough in reconstructing the original data that is already present in the distribution but has failed to render new samples which follows the fame distribution. Here comes the role of VAE. It takes a simple but effective extension of auto encoder, instead of generating a single encoding of the input samples. It generates a probability distribution over each of the encoding using the encoder. The decoder then sample from this probability distribution and generate new samples. ******

**1. LATENT VARIABLE**

It corresponds to a real feature of the object that have not been measured

In the above example, we trained auto encoder on a large dataset faces with encoding dimension of 6. An ideal autoencoder will learn the descriptive attributes of fakes like skin, smile etc… in order to describe observation in some compressed form. We have described the input image in terms of latent variable using single value to describe each attribute. Using VAE, we define the latent attributes in probabilistic manner.

With this approach we now represent each latent attribute for a given input as probabilistic distribution. While decoding, we randomly sample from each latent state distribution to render a vector as input for the decoder model.



The derivation of the Loss function can be solved using :

**2.The Problem Of Approximate Inference**

Let x be a set of observed variable and let Z be the set of latent variable with joint distribution P(ᵶ|x).

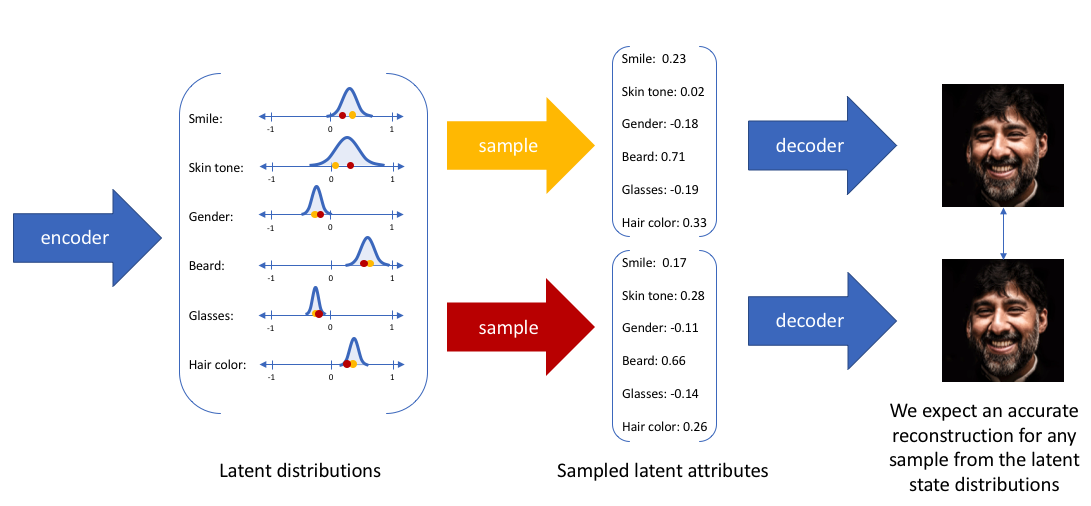
Then the inference problem will complete the conditional distribution of latent variables given the observation

P(ᵶ|x) = P(x|ᵶ) P(ᵶ) / P(x)

Evaluating the above equation is difficult because P(x) cannot be solved

**Reason :** P(x) = ʃ P(x|ᵶ) P(ᵶ) dx , this integral is not available in closed form or is intractable

(i.e., it requires exponentisl time to complete) due to multiple integral involved for latent variable vector Z.



***ALTERNATIVE:***

The alternate approach is to approximate P(ᵶ|x) by another distribution Q(ᵶ|x) what is defined in such a way that it has a tractable solution. This is done using variational inference. The main idea of V.I is to pose the inference problem as an optimization problem. It is done by modeling P(ᵶ|x) using Q(ᵶ|x) where Q(ᵶ|x) has a simple distribution such as Gaussian. This is done by minimizing the KL(Q(ᵶ|x) || P(ᵶ|x)). After rearranging and deriving, we obtain the objective function of VAE as

E z ~ Q(ᵶ|x [ log (P(x|ᵶ)) - DkL (Q(ᵶ|x)||P (ᵶ)) ]

first term represents the reconstruction likelihood and the second term ensures that our learned distribution Q is similar to the prior distribution P.

**LOSS FUNCTION = - objection function**

= - E z ~ Q(ᵶ|x [ log (P(x|ᵶ)) + DkL (Q(ᵶ|x)||P (ᵶ)) ]

So our target is to find the optimal Ø,Ѳ such that

Ѳ\* , Ø\* = argmin Ѳ,Ø  L(Ѳ,Ø)

Intution about Loss Function

L(Ѳ,Ø) = - E z ~ Q(ᵶ|x [ log (P(x|ᵶ)) + DkL (Q(ᵶ|x)||P (ᵶ)) ]

Ѳ,Ø are new work parameter which are to be learned that’s why loss function is parameterized by them so that after minimizing loss function by adjusting Ø and Ѳ we attain the model that is capable enough to generate the desired output. QØ(ᵶ|x) is neural network which is also called as recognition model which x -> ᵶ. The first term represent the log likelihood.

P(x|ᵶ) = N (ψ(ᵶ) , £(ᵶ))

So then we take log of the Gaussian, we get a square error between the data sample x and the mean of the Gaussian distribution P(x|ᵶ) is also called as the generative model us it is taking the input as z to the latent vector and mapping to the X’ the reconstructed sample.

Second term is the regularizer : PѲ(ᵶ) ~ N(0,1)

KL divergence not allows the pdf of the latent variable to collapse with zero variance but penalize the deviation from PѲ(ᵶ). Once you feed it to the decoder you get a new sample thich has the mixture of z1 & zz  which is not present in the real distribution but follows the same. 