Autoencoders

Autoencoder

Auto means “Self”.

Encoders means “Transform Information to another form”.

Autoencoder is a deep learning model that learns how to encode information

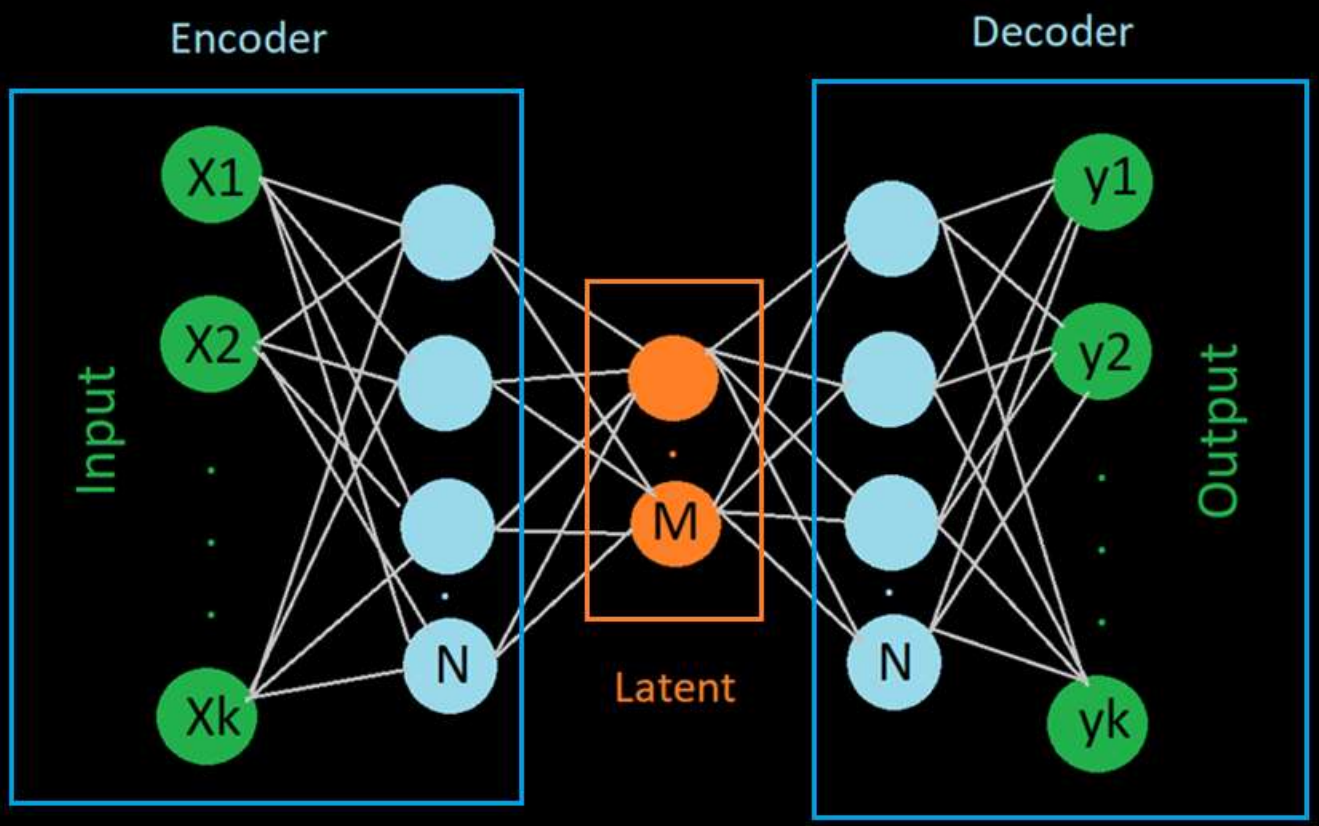
Autoencoder Architecture

An autoencoder consists of three components

- Encoder
- Code or Latent representer
- Decoder

The encoder compresses the input and produces the latent representation, the decoder then reconstructs the input only using this code.

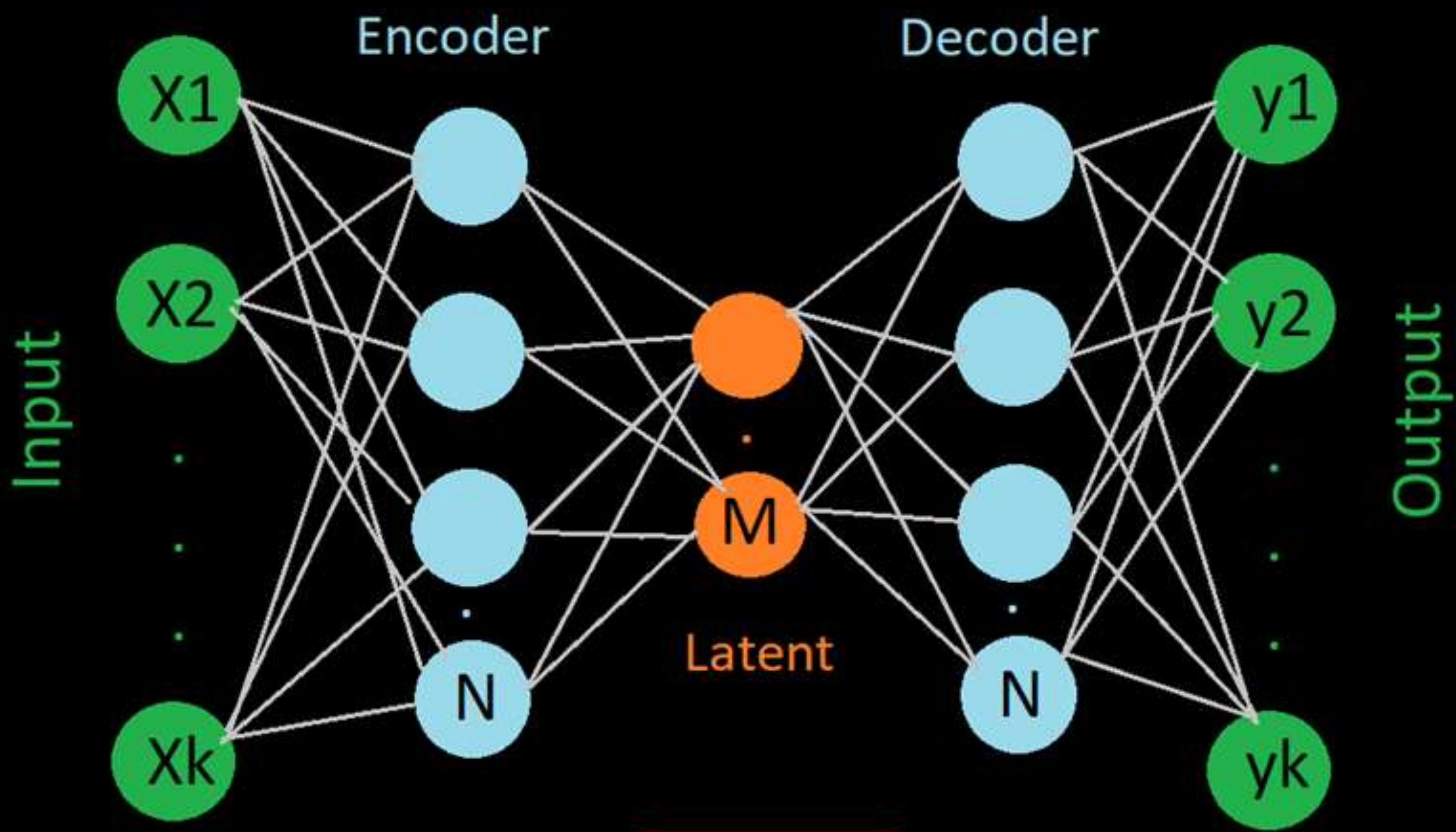
Autoencoder Architecture



Autoencoder Architecture

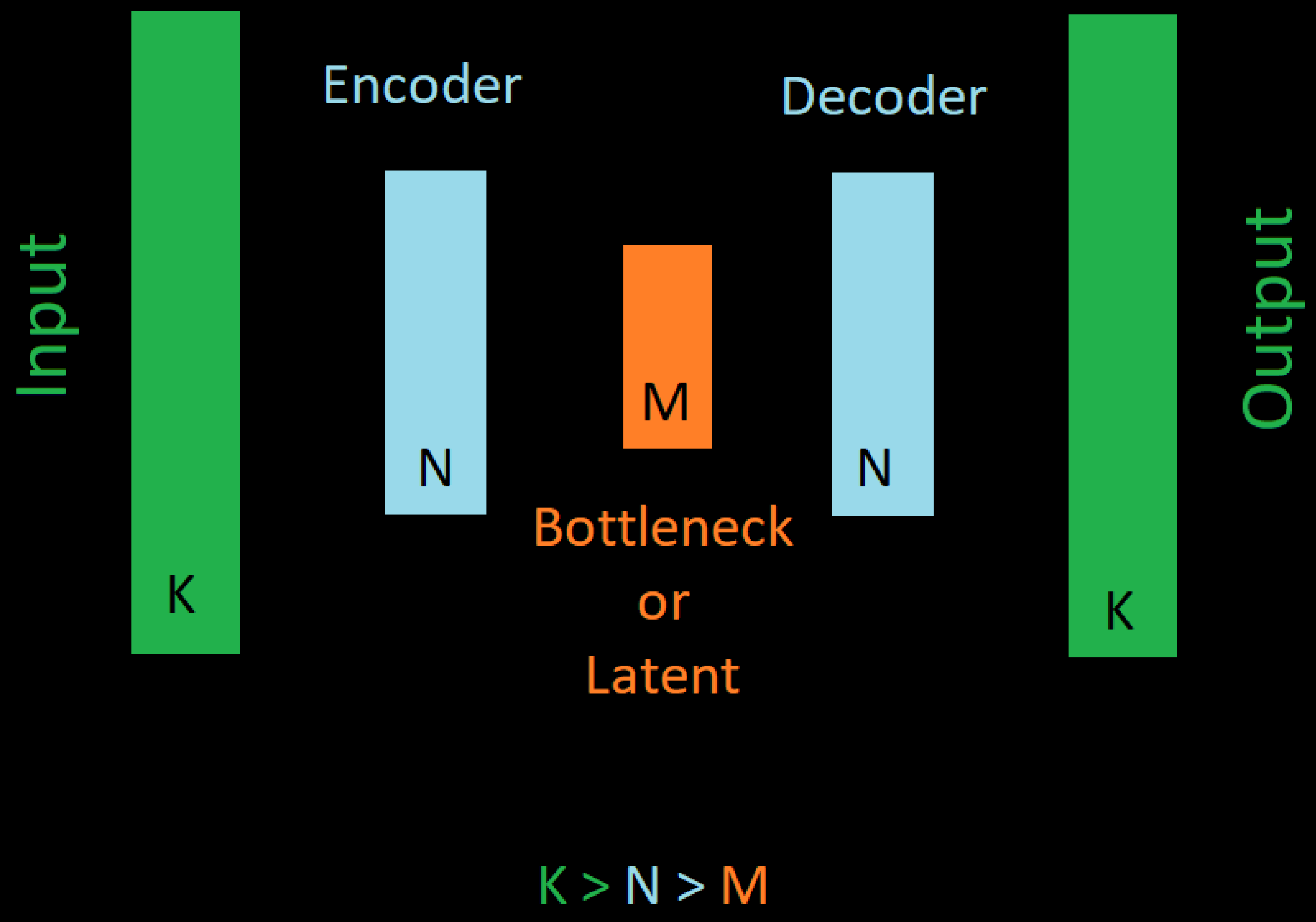
- Fully connected Autoencoder
- CNN Autoencoder
- LSTM Autoencoder
- Transformer based Autoencoder

Autoencoder Architecture

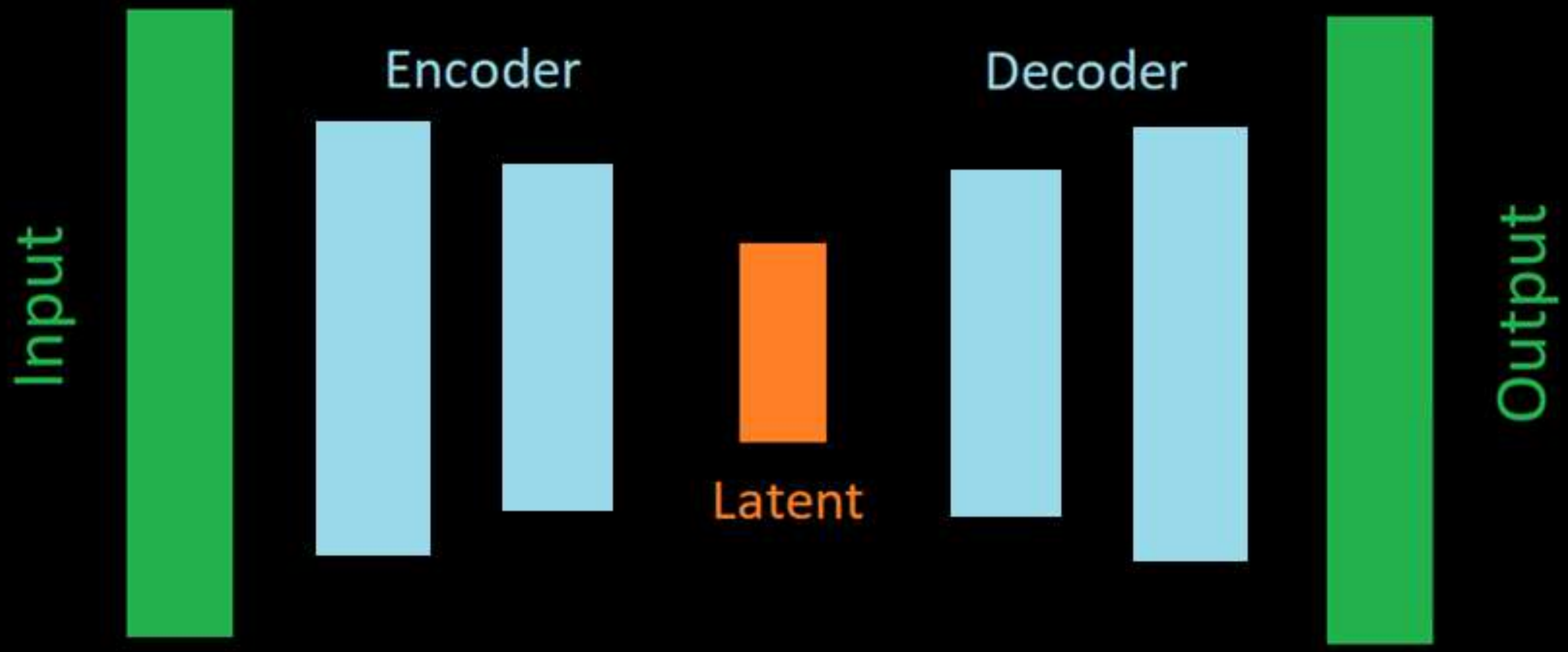


K > N > M

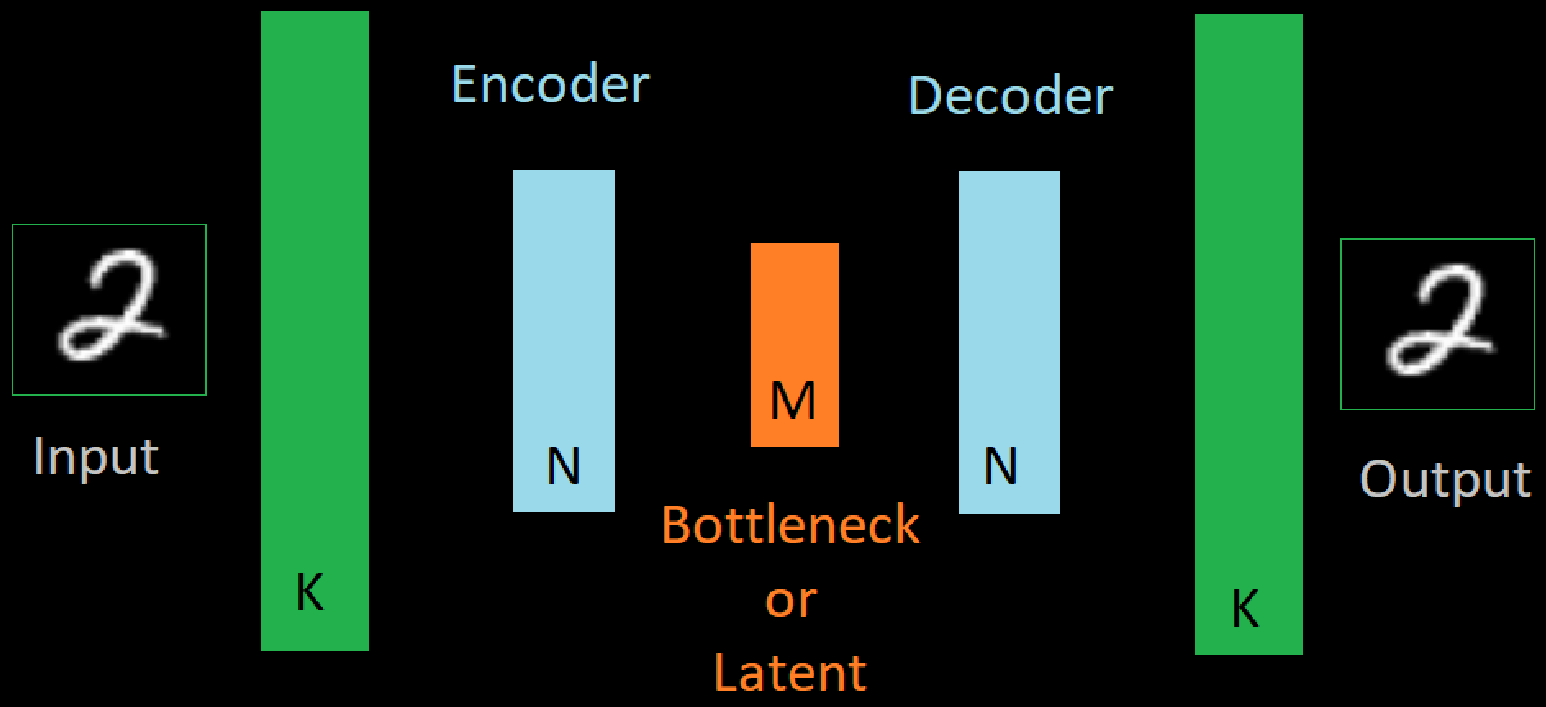
Autoencoder Architecture



Autoencoder Architecture



Working of Autoencoder



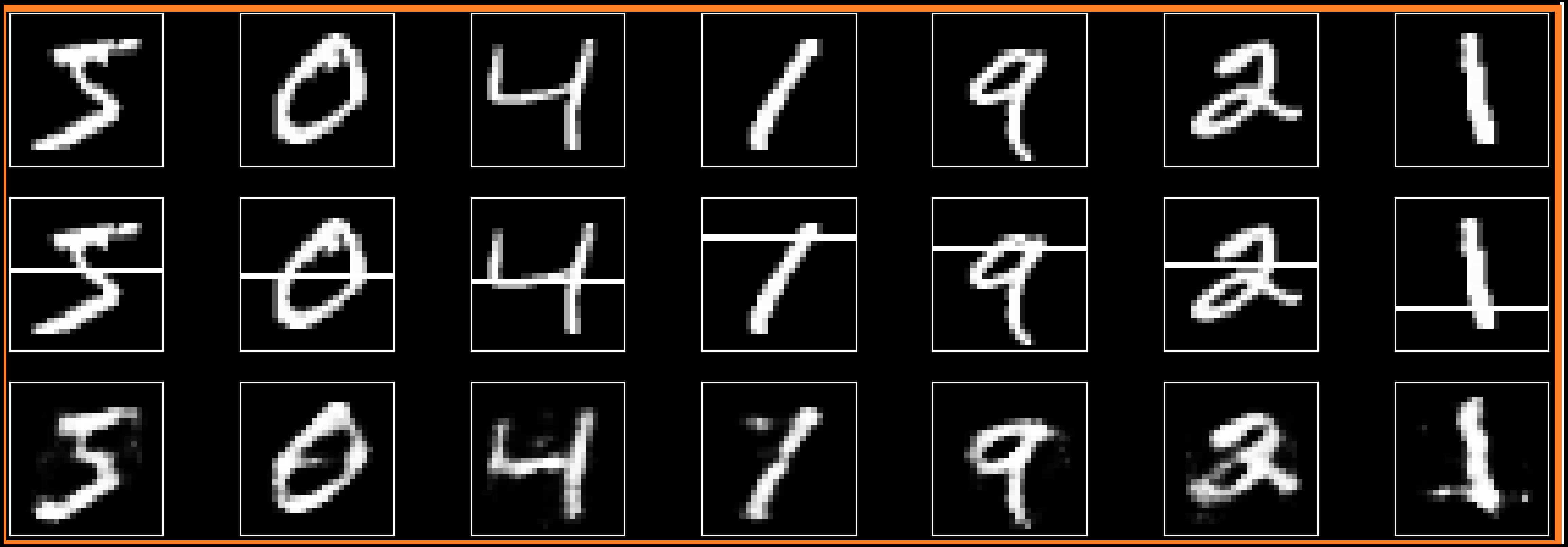
Applications of Autoencoders

- Reducing Feature dimension.
- Removing Noise from data.
- Removing Occlusion from data.
- As an Image Classifier.

Autoencoder is Noise Remover

Autoencoder is Occlusion Remover

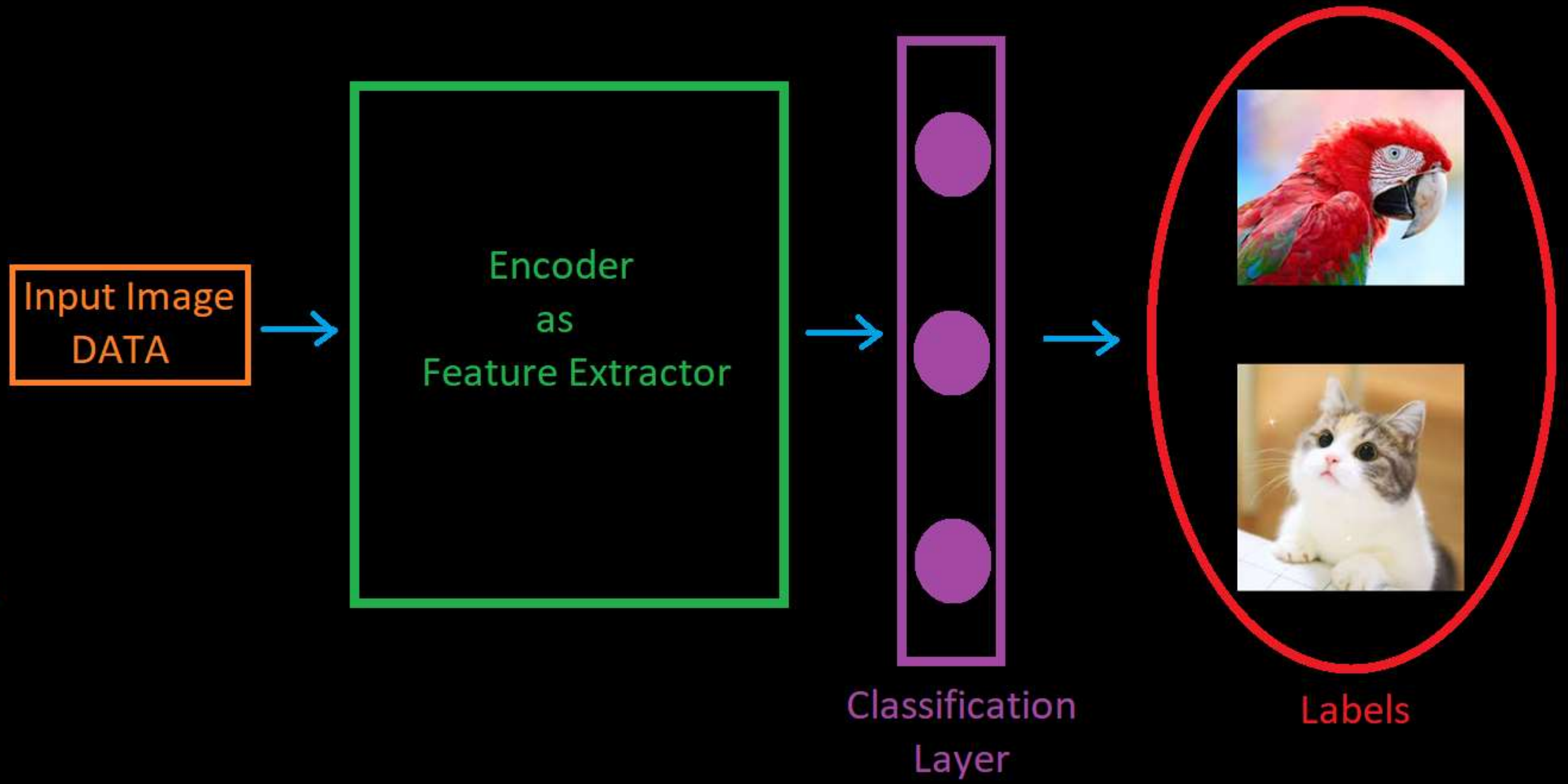
Autoencoder is Occlusion Remover



Loss Function of Autoencoder

$Loss\ function = \frac{1}{N}\sum\Big($  $-$  $\Big)^2$

Autoencoder is an Image Classifier



Variational Autoencoder (VAE)

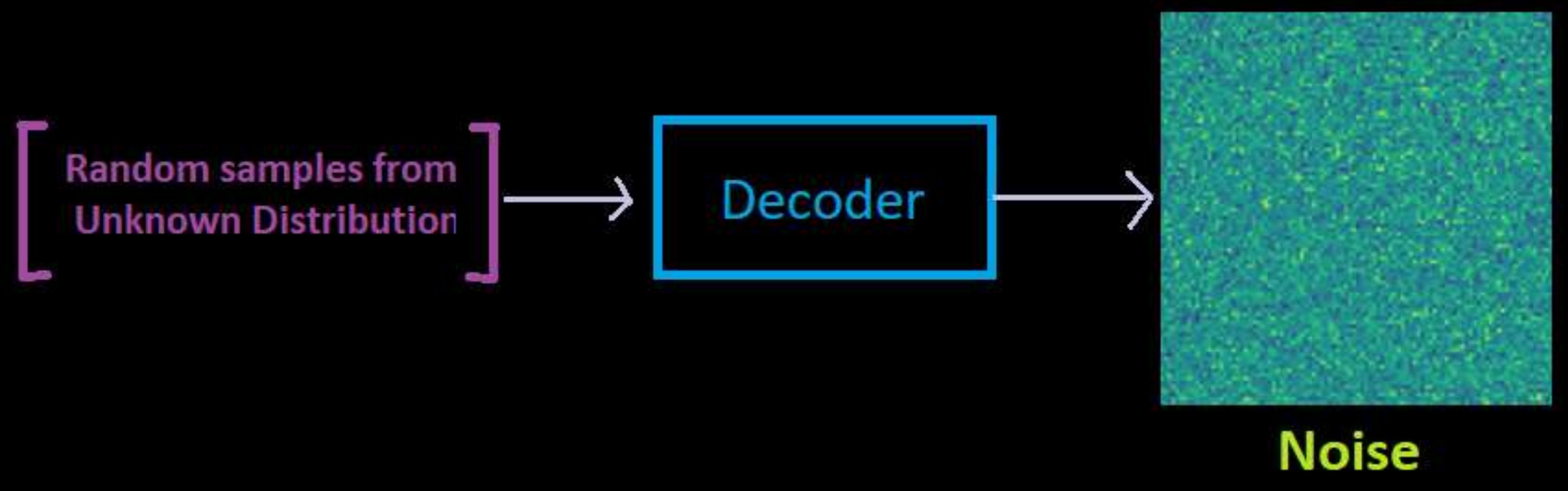
Loss Functions for Autoencoder

$$\boxed{MSE = \frac{1}{n}\sum_{i=1}^{n}(y - y_{pred})^2}$$

$$\boxed{BCE = -\frac{1}{n}\sum_{i=1}^{n}[(y_i \, log(p_i) + (1 - y_i) \, log(1 - p_i)]}$$

Problem With Autoencoder

- The compressed representation does not know to which distribution it belongs to.
- During testing, if we input a random data to the decoder, we will get a random image.



Solution of the Problem

- The solution to the problem is that it is necessary to know the distribution from which data is sampled before input to the decoder.

- Force the encoder to produce a specific probability distribution (Gaussian / Normal) over encoding.

- Decoder is then required to sample from the probability distribution and reconstruct original image.

Variational Autoencoder (VAE)

- In VAE, the last layer of encoder is used to predict two vectors i.e mean(μ) vector and variance(σ)
- We expect VAE to learn Gaussian / Normal distribution with mean(μ) and variance(σ) so that we can sample from mean(μ) and variance(σ) which VAE learn
- However the issue is that sampling is not differentiable so we can't do back propagation.

Reparameterization

We randomly sample (ϵ) from unit Gaussian and then shift it by latent distribution's mean(μ) and latent distribution's variance(σ). This allows us to perform back propagation

$$\boxed{z = \mu + \sigma \odot \epsilon}$$

Now, we can perform $\boxed{\frac{\partial z}{\partial \mu}}$ and $\boxed{\frac{\partial z}{\partial \sigma}}$

Loss Function for VAE

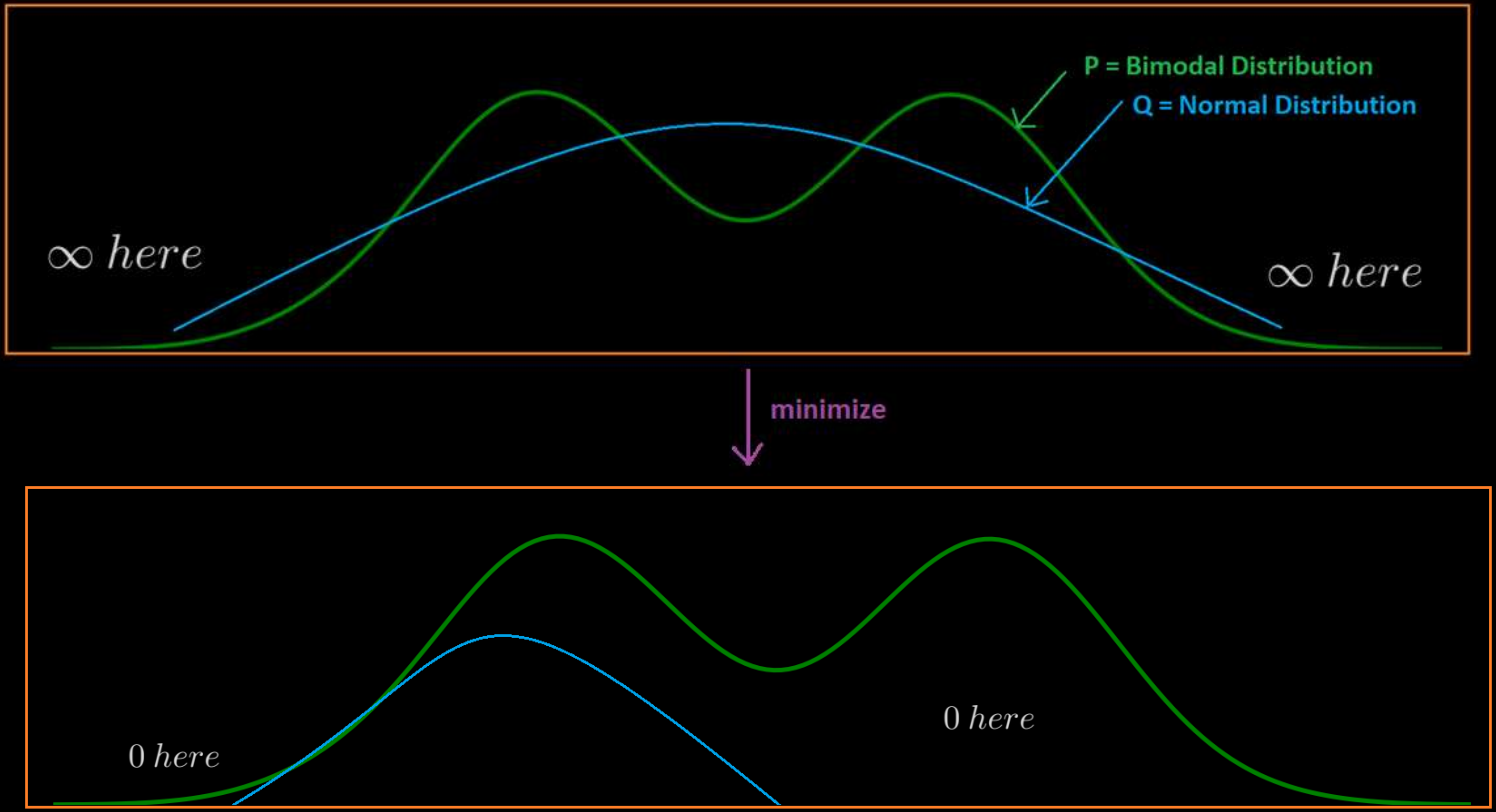
The loss function consists of two parts. First part is to minimize the reconstruction loss between the original image and reconstructed image and second is to reduce KL (Kullback-Leibler) Divergence loss. KL Divergence measures the similarities between two probability distributions such as $P(x)$ and $Q(x)$. If two distributions are different KL Divergence loss is high and if two distributions are similar, KL Divergence loss is low. Therefore, the target of KL Divergence is make two distributions equal (KL Divergence = 0)

$$\boxed{Loss\ Function = BCE + D_{KL}}$$

KL Divergence

$$D_{KL}(Q|P) = \sum Q(x) log(\frac{Q(x)}{P(x)})$$

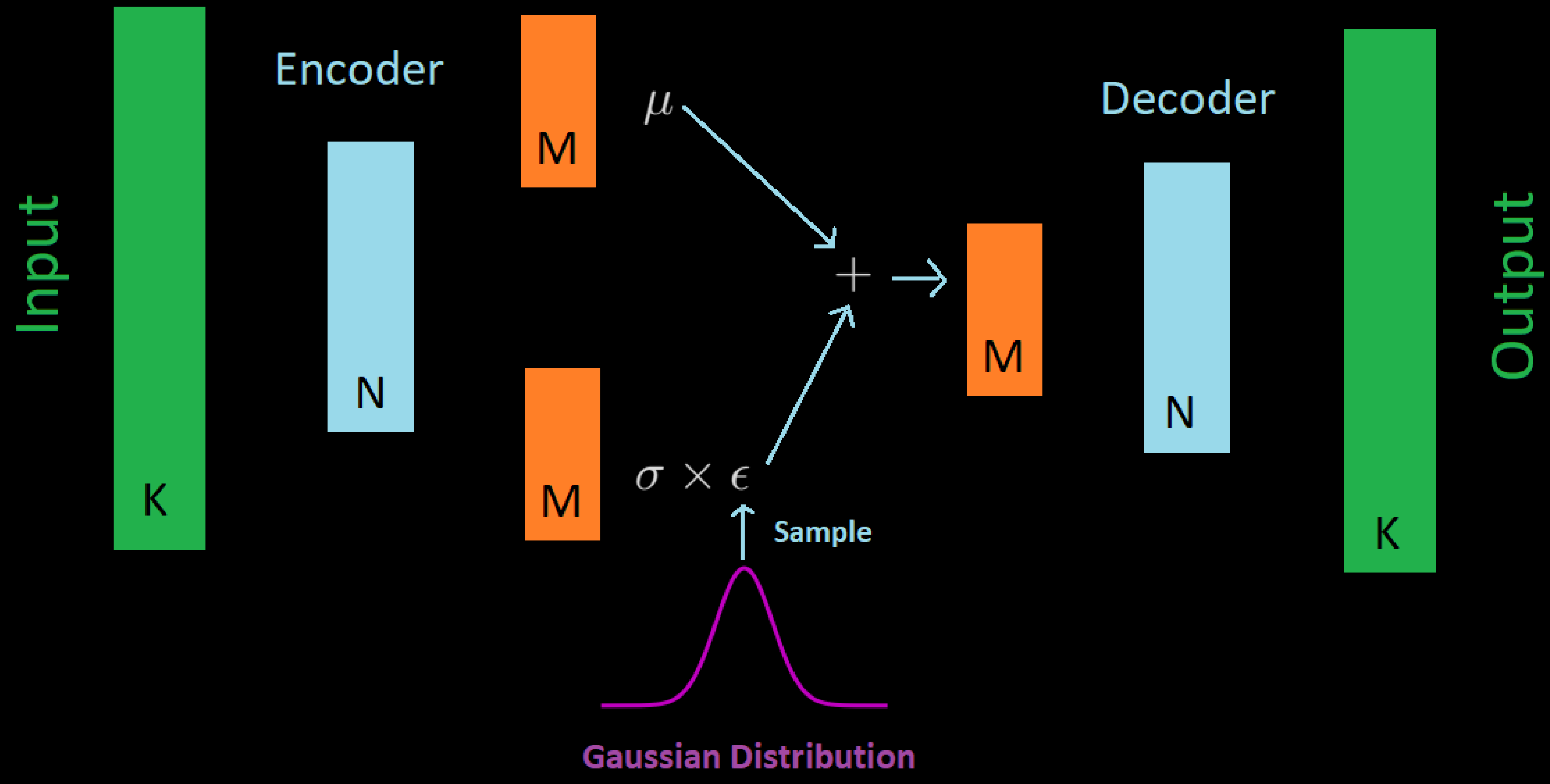
$$\boxed{D_{KL}(Q|P) = \infty \; when \; Q(x) \; has \; value \; and \; P(x) \; is \; zero}$$


P = Bimodal Distribution
Q = Normal Distribution
∞ here
∞ here
minimize
0 here
0 here

Derived KL Divergence for VAE

$$D_{KL}(Q|P) = \frac{1}{2}\sum(\exp(\Sigma(X)) + \mu^2(X) - 1 - \Sigma(X))$$

Architecture of VAE



Thank you !

Thank you !