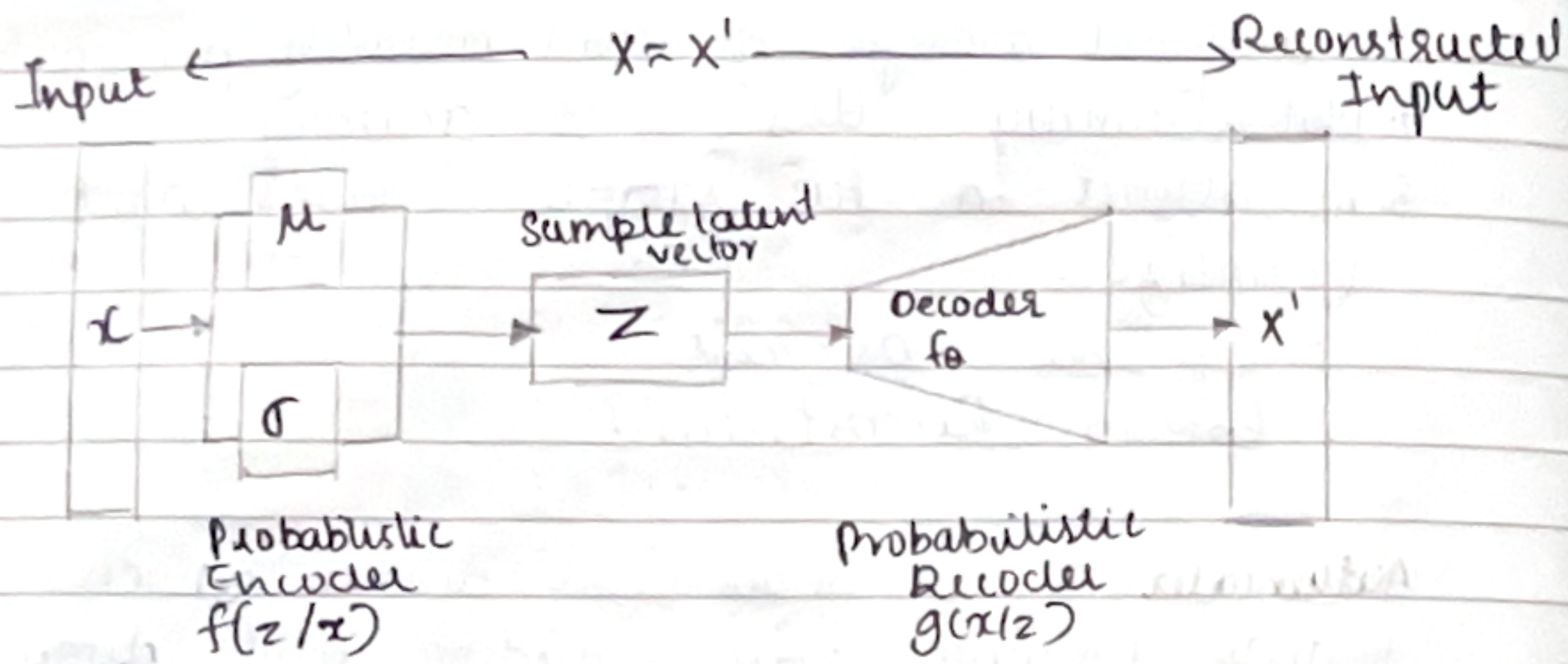
Variational Autoencoder Architecture



Z is the compressed low dimensional representation of input X

- → VAE is a probabilistic generative model that learns to generate new samples from the same distribution as the training data
- → VAE learn latent variables as distributions (mean & variance) instead of fixed vectors
- → Reparameterization: Ensures gradients can flow through the random sampling step by step computing

  $z = \mu + \sigma * \epsilon$, where $\epsilon \sim N(0,1)$
- → Reconstruction Loss → ensures accurate image reconstruction.
- → KL Divergence → Regularizes the latent space to follow a normal distribution
- → VAE can generate new images by sampling random latent vectors

17/10/25

LAB 11:

Experiment using Variational Autoencoder (VAE)

AIM:

To implement and train a VAE using MNIST dataset.

OBJECTIVES:

1. To understand the concept of VAE - a generative model that learns data distributions
2. To implement encoder and decoder networks that generate latent variables as probability distributions
3. To compute ~~both~~ and minimize both reconstruction loss and KL divergence loss.

PSEUDOCODE:

1. Import required libraries
2. Load dataset
3. Define VAE model:
   - Encoder
     - * Linear layers → ReLU activations
     - * Outputs: mean (μ) and log variance ($\log\sigma^2$)
   - Reparameterization:
     - * Sample $z = \mu + \sigma * \epsilon$ ($\epsilon \sim N(0,1)$)
   - Decoder:
     - * Linear layers → sigmoid activation to reconstruct image

OUTPUT:

Epoch 1/10 Loss : 45.1720

Epoch 2/10 Loss : 34.9497

Epoch 3/10 Loss : 33.0392

Epoch 4/10 Loss : 32.1979

Epoch 5/10 Loss : 31.6449

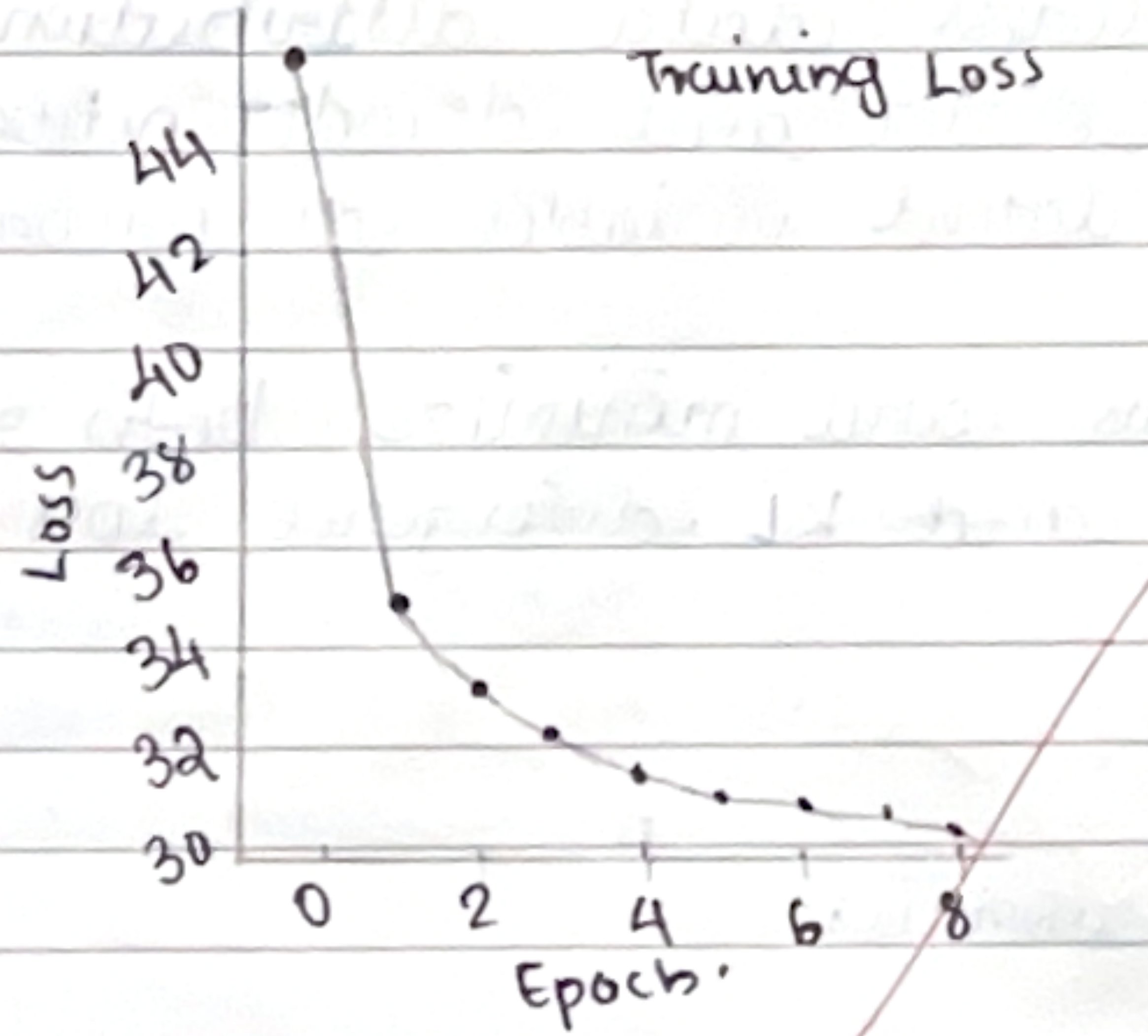
Epoch 6/10 Loss : 31.3058

Epoch 7/10 Loss : 31.0876

Epoch 8/10 Loss : 30.8796

Epoch 9/10 Loss : 30.7358

Epoch 10/10 Loss : 30.5795



OBSERVATION:

→ The loss decreases steadily with each epoch, showing that the model is learning both reconstruction and latent space structure

→ Latent space learned by the model represents the digit features in a continuous form.

4. Define the VAE loss:
   - Reconstruction loss (MSE)
   - KL divergence : $-0.5 * \Sigma(1 + \log\sigma^2 - \mu^2 - \sigma^2)$
   - Total loss = Reconstruction + KL divergence.
5. Initialize optimizer (Adam) and train model:
   - Forward pass → compute loss. → back prop → update weights
6. Plot training loss.

~~MSTN~~

RESULT:

The variational autoencoder successfully implemented and trained ~~MSTN~~ MNIST dataset.