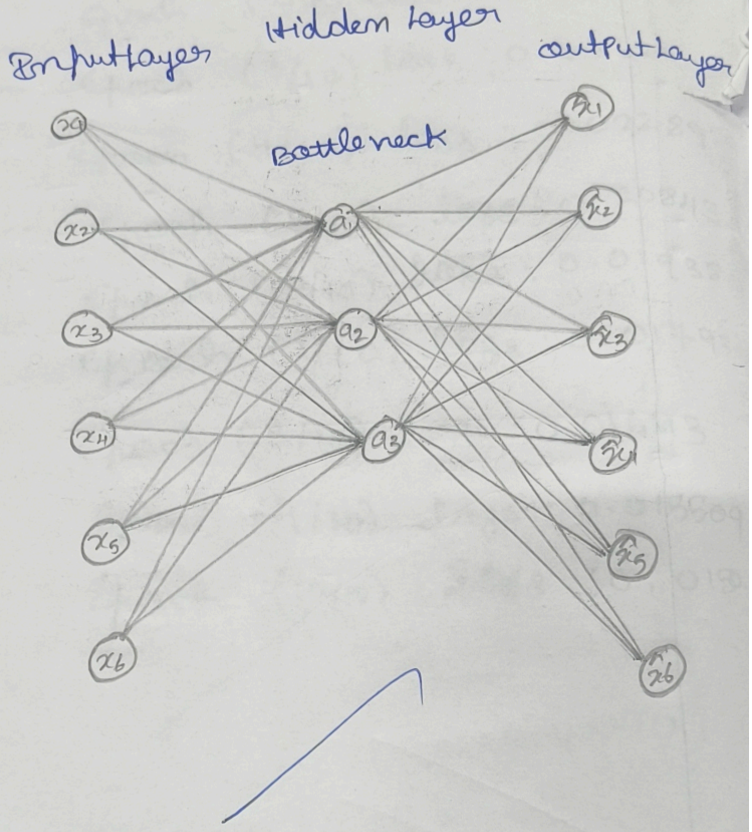
Architecture :-



11. Experiments using Variational Autoencoders

Aim :-

To implement and analyze a variational Autoencoder (VAE)

Objective:

* To understand the concept and working of variational Autoencoders (VAEs)

* To Perform unsupervised feature learnings using probabilistic latent space representation

* To train a VAE model that learns both encoding and decoding

* To visualize the latent space and generate new images by sampling from it

PSEUDOCODE:-

BEGIN

Import torch, torchvision

load MNIST Dataset and Normalize images to [0, 1]

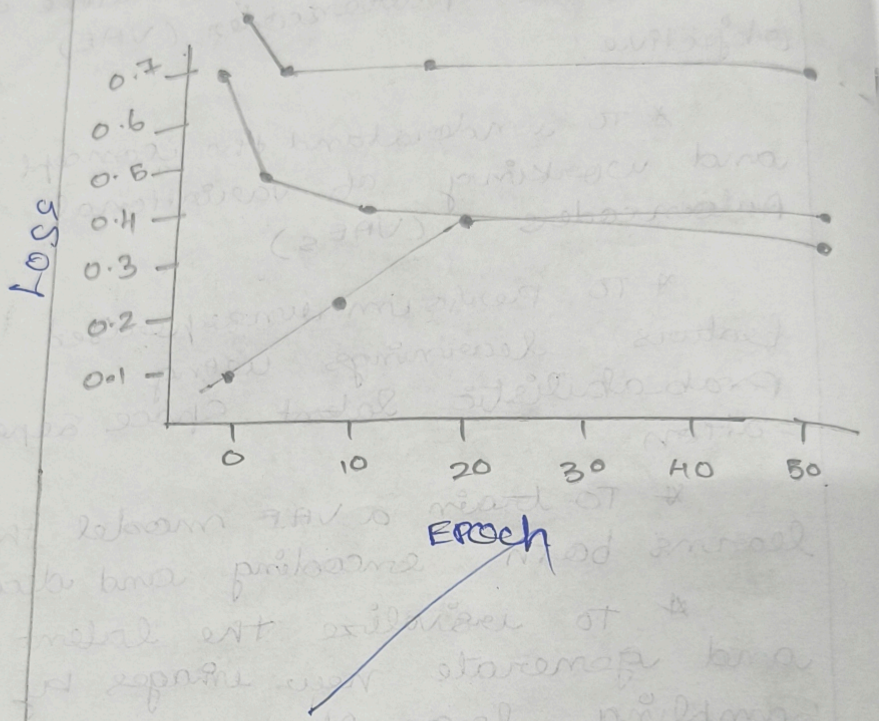
Define VAE model:

Encoder: 784 → 400 →

(mean μ: log-variance σ^2)

Reparameterise: $z = \mu + \sigma * \epsilon$ ($\epsilon \sim N(0,$

Decoder: z – 400 → 784 (sigmoid la


Loss
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
10
20
30
40
50
EPOCh

Define loss:

total loss = reconstruction loss + KL Divergence use Adam optimizer for training

for each epoch:

For each batch:

Forward Pass -> encode sample, decode

compute loss -> back Propagate -> update weights

After trainings

Reconstruct test images and generate new samples from latent space

Visualize original, reconstructed and generated images

END

OBSERVATION

* The VAE learned smooth latent space allowing continuous inter polation between different digits

* Generated samples resemble relistic handwritten digits, though slightly blurry due to probabilistic sampling

* The KL Divergence term regularized the latent space, ensuring meaning representati

* Reconstruction loss decrea gradually, indicating successful lea

Result:-

Therefore the implementation and Analyse of variational Autoencoder is successfully completed.