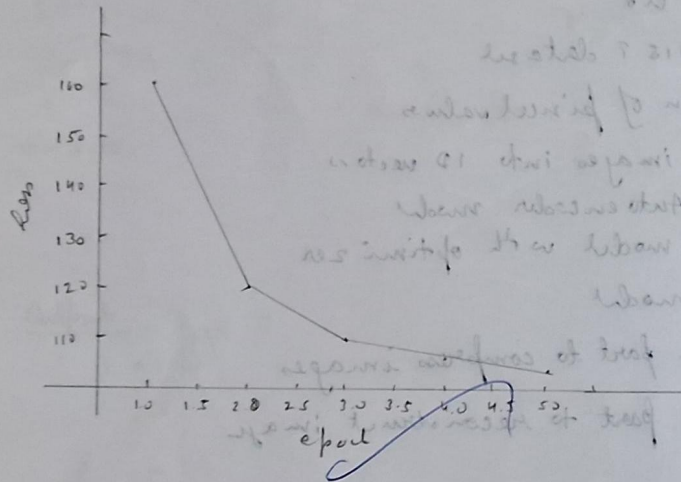


Output

Epoch (1, 5) Loss: 184.0212
Epoch (2, 5) Loss: 127.5910
Epoch (3, 5) Loss: 114.8092
Epoch (4, 5) Loss: 111.6099
Epoch (5, 5) Loss: 109.2443



(continued)

AIM: To implement and analyse a variational autoencoder for learning latent representation of MNIST dataset

Pseudo code:

Import lib
Load MNIST dataset
Define Encoder Network
Sampling step
Define decoder network
Define loss function
Train the model
→ Forward pass
Compute loss
Backpropagate and update weights
After training,
Deconv test images
Generate new images

Observation

During training, the reconstruction loss gradient dec. indicating, better learning of input

The divergence stabilizer over time showing that the latent space was learning valid distribution. The model successfully learned a smooth latent space where later two points produced

meaningful transition between digits.

Result

An variational autoencoder model successfully implemented on MNIST dataset;

Accuracy - 80.67 %

80.67

01.2M

68.551

5

2

0

01

01

03

Pseudo code

Import libraries and set device

Load CIFAR-10 dataset, normalize to $(0,1)$, create dataloader

Define VQVAE generator: series of conv layers, LeakyRelu, BatchNorm, etc.

Initialize weight (Normal with mean 0, std = 0.02)

Define loss and optimizers

For each epoch

1) Train discriminator on real image

2) Train generator to fool discriminator

3) Save generator batch of generated image for visualization

Monitor loss