ECE 285 - Deep Generative Models - Assignment 3

Saqib Azim (A59010162)
sazim@ucsd.edu

NOTE: USING 4 GRACE DAYS FOR THIS ASSIGNMENT

ASSIGNMENT REPOSITORY: https://github.com/saqib1707/ECE285_Assignment3

1 VAE Introduction

Variational Autoencoders (VAEs) belong to the set of deep generative models. With huge training data, sophisticated model architectures, and optimization algorithms, VAEs can generate highly realistic content of various kinds such as images, text, speech, etc. VAEs can be described as autoencoders whose training is regularized to avoid overfitting and ensure that the latent space has good properties to generate new synthetic and variety of content.

1.1 Dataset Information

Name: Chest X-Ray Dataset
Download Path: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

Number of Train Images (NORMAL + PNEUMONIA): 5216
Number of Validation Images (NORMAL + PNEUMONIA): 16
Number of Test Images (NORMAL + PNEUMONIA): 624

The goal of this assignment is to learn to generate new synthetic chest x-ray images from random input vectors sampled from the latent distribution space using variational autoencoder models. Since, we are required to only generate synthetic images, there is no need for classification labels. Therefore, all the images belonging to NORMAL and PNEUMONIA class have been merged together. The chest x-ray images are of varying sizes / resolution, hence all the images are resized to a fixed size (M x M) before feeding to the Vanilla VAE network.

1.2 Network Architecture

Vanilla VAE model has been used to train the chest x-ray data in order to learn to generate synthetic images. The Vanilla VAE architecture consists of an encoder and a decoder. The encoder encodes an input image of shape MxMx3 through multiple convolutional hidden layers to a latent space of dimension D_L. The output of the encoder is designed such that it represents the mean and covariance vectors of a gaussian distribution space. Then, a random vector is sampled from this gaussian distribution space and fed to the decoder which, again through a series of hidden convolutional layers, outputs an image of the same shape as input. The loss function is designed to meet 2 objectives - First, minimize the reconstruction loss between the input image and reconstructed decoded output using MSE. Second, regularize the latent space such that any random vector sampled from the latent space results in a meaningful output using KL divergence loss (for example, vectors sampled very close to each other in the latent space should result in similar decoded output). During the experiment, the following hyper-parameters and their effect on training and performance has been observed using metrics such as inception score (IS) and frauchet inception distance (FID) scores.

- input patch size (M x M)

Preprint. Under review.

- latent dimension D_L
- Number of epochs N_E
- KL divergence weight β
- learning rate α

Inception Score (IS): This metric is used for evaluating the quality of generated images by generative models (such as GAN, VAE, etc.). It uses the classification probabilities provided by a pre-trained InceptionV3 model to measure performance of a VAE. It passes each generated sample image through an InceptionV3 classification network and generates classification probability distribution. If this distribution is narrow, then this implies that the image has a distinct class. Similarly, it also computes marginal score for all the classes using all the images. If this marginal distribution is uniform, then it means the VAE network is capable of generating variety of images belonging to different classes. Using these concepts, the inception score is computed. Higher the inception score, better the generative capability of a generative network.

Frechet Inception Distance (FID): This metric calculates the distance between feature vectors extracted for real and synthetic generated images. Similar to IS, it uses a pre-trained InceptionV3 model, and computes the mean and co-variance between the feature vectors of real and generated images to measure VAE model performance. Since the FID score measures distance between the latent gaussian distribution and the standard normal distribution, lower the FID score, the better the performance of the generative model.

The chest x-ray dataset and the generated images are both MxMx3 (where M = 64, 128, 256), but the InceptionV3 model (used in FID and IS computation) requires images of minimum size 299x299x3. Hence, the dataset test images and VAE generated sample images have been interpolated to the specified required size while calculating these metrics. In addition, the IS and FID scores are reported using samples generated after the training is complete.

1.3 Common hyperparameters:

Following hyperparameters have not been modified during these experiments.

- Number of input channels: 3
- Train batch size: 64
- Validation batch size: 4 (since only 16 validation images are used)
- Leaky ReLU weight: 0.01

1.4 Experiments

1.4.1 Experiment-1

Hyperparameter settings: M = 64, $D_E = 128$, $N_E = 100$, $\beta = 0.00025$, $\alpha = 0.005$

Metrics Score: Inception Score (IS): 2.1461, Frauchet Inception Distance (FID): 2.8069

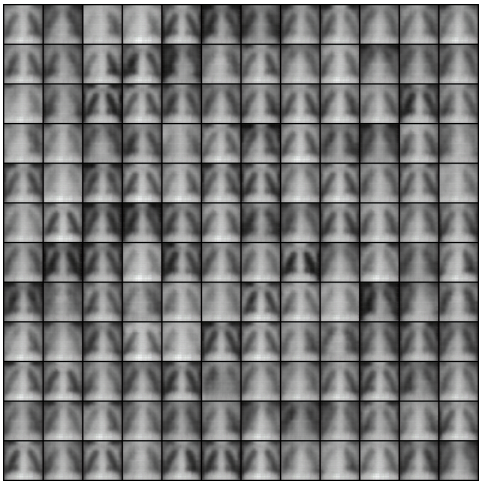

Figure 1: Generated samples at training epoch=5

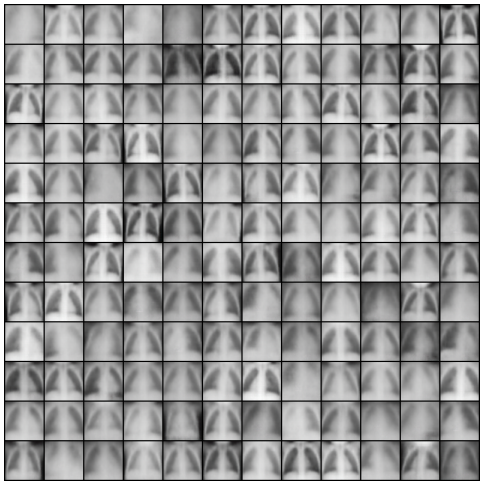

Figure 2: Generated samples at training epoch=99

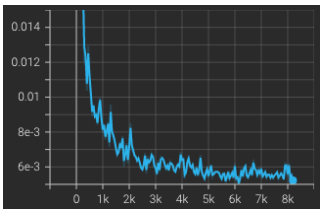

Figure 3: Reconstruction Loss Curve

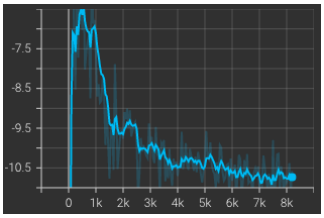

Figure 4: KLD Loss Curve

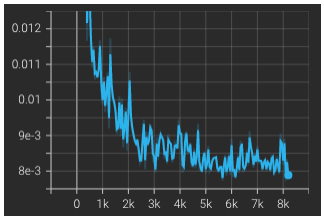

Figure 5: Overall Loss Curve

1.4.2 Experiment-2

Hyperparameter settings: M = 64, $D_E = 256$, $N_E = 100$, $\beta = 0.00025$, $\alpha = 0.005$

Metrics Score: IS: 2.0784, FID: 2.1167

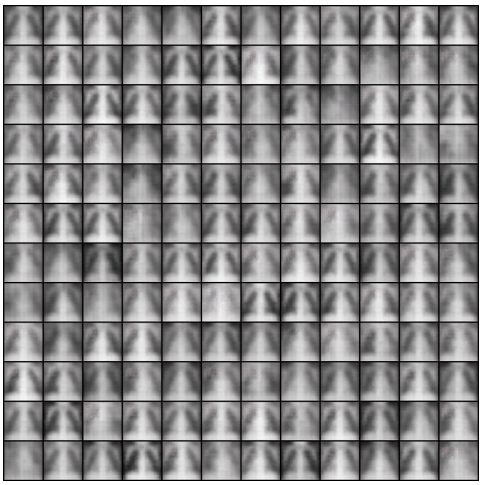

Figure 6: Generated samples at training epoch=5

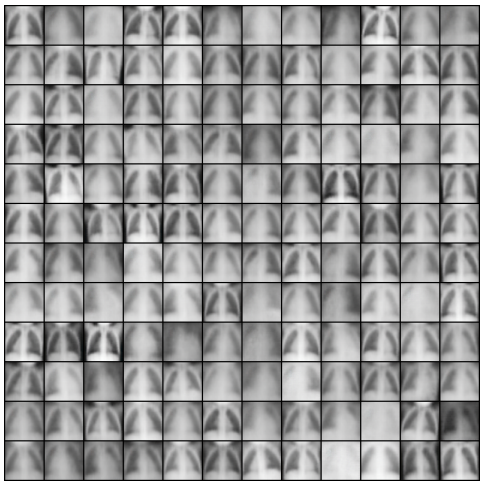

Figure 7: Generated samples at training epoch=99

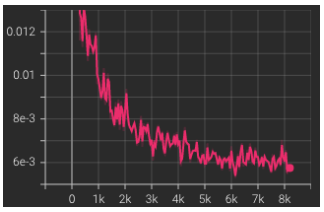

Figure 8: Reconstruction Loss Curve

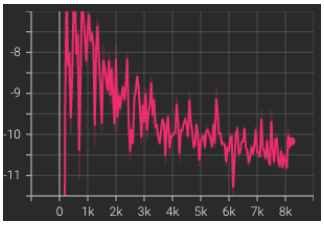

Figure 9: KLD Loss Curve

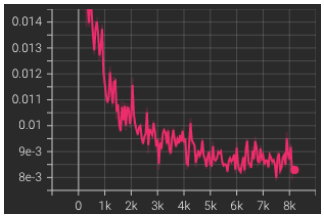

Figure 10: Overall Loss Curve

1.4.3 Experiment-3

Hyperparameter settings: M = 128, $D_L = 128$, $N_E = 100$, $\beta = 0.00025$, $\alpha = 0.01$

Metrics Score: IS: 2.1292, FID: 34.5945

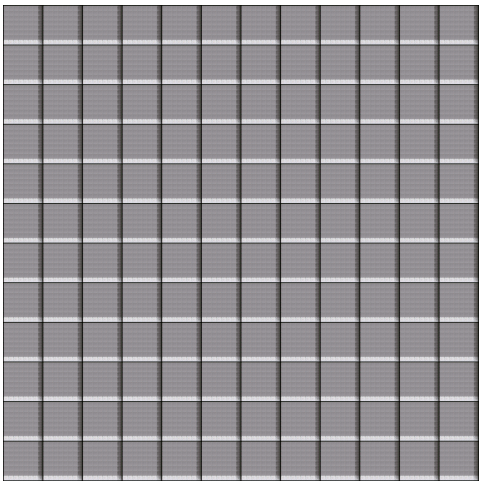

Figure 11: Generated samples at training epoch=5

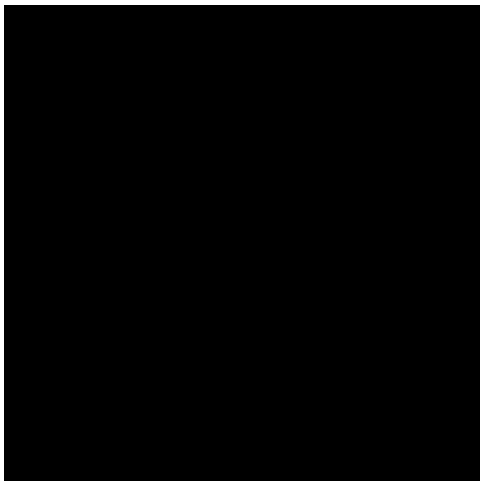

Figure 12: Generated samples at training epoch=99

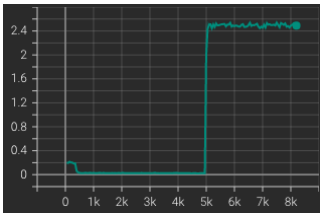

Figure 13: Reconstruction Loss Curve

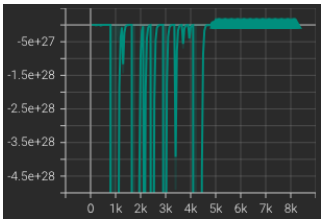

Figure 14: KLD Loss Curve

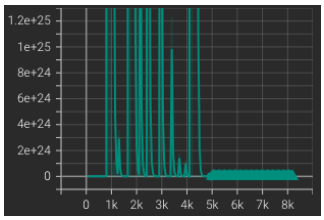

Figure 15: Overall Loss Curve

1.4.4 Experiment-4

Hyperparameter settings: $M = 64$, $D_L = 128$, $N_E = 100$, $\beta = 0.0025$, $\alpha = 0.005$

Metrics Score: IS: 1.5013 , FID: 2.9523

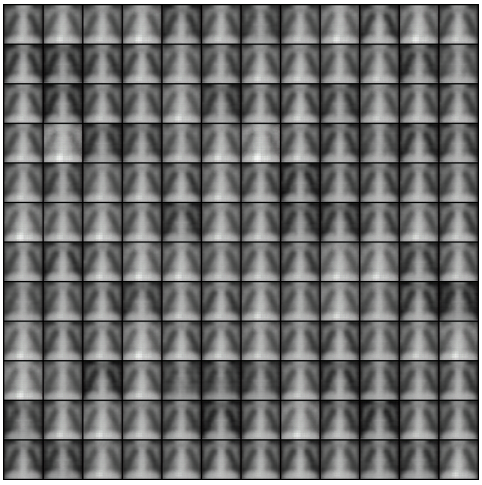

Figure 16: Generated samples at training epoch=5

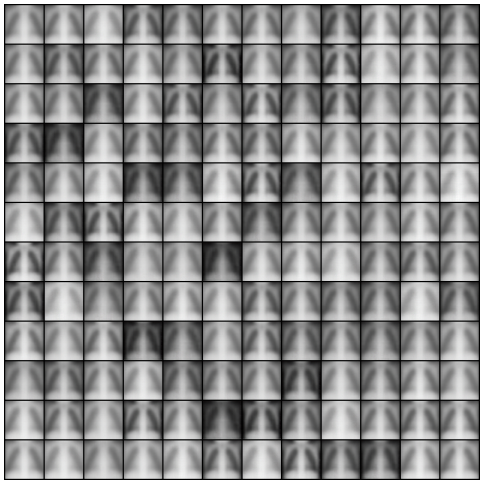

Figure 17: Generated samples at training epoch=99

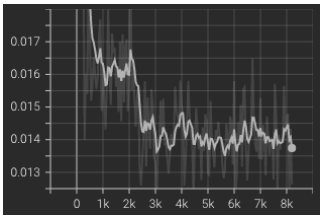

Figure 18: Reconstruction Loss Curve

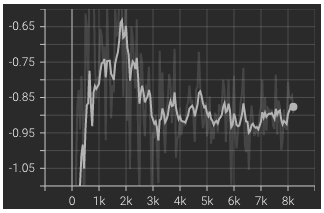

Figure 19: KLD Loss Curve

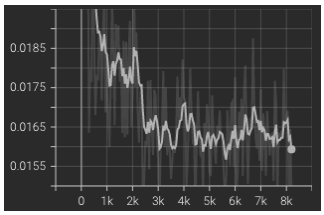

Figure 20: Overall Loss Curve

1.4.5 Experiment-5

Hyperparameter settings: $M = 64$, $D_L = 64$, $N_E = 50$, $\beta = 0.0025$, $\alpha = 0.005$

Metrics Score: IS: 1.8232 , FID: 2.4871

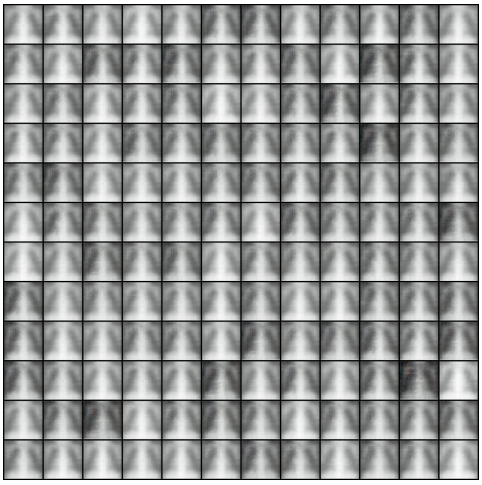

Figure 21: Generated samples at training epoch=5

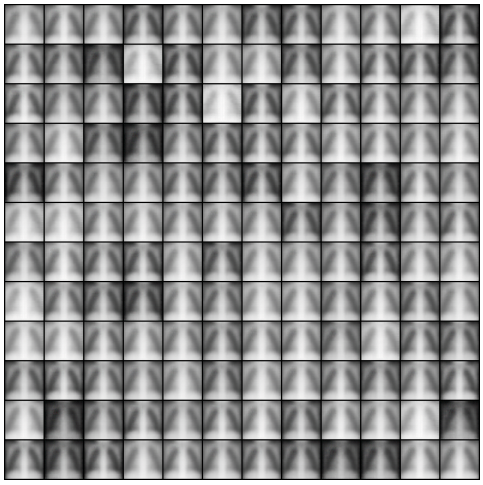

Figure 22: Generated samples at training epoch=49

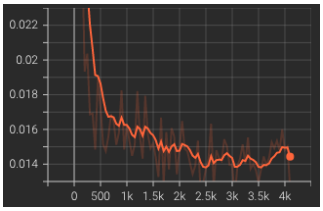

Figure 23: Reconstruction Loss Curve

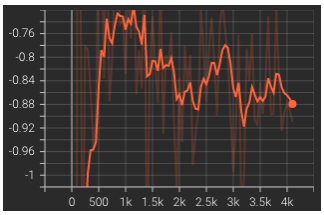

Figure 24: KLD Loss Curve

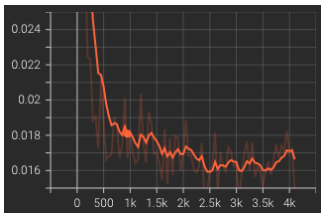

Figure 25: Overall Loss Curve

1.4.6 Experiment-6

Hyperparameter settings: $M = 256$, $D_L = 64$, $N_E = 20$, $\beta = 0.0025$, $\alpha = 0.005$

Metrics Score: IS: , FID: (due to lost connection, could not record these values)

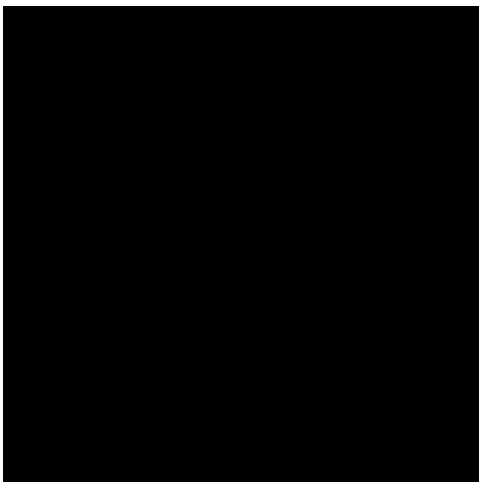

Figure 26: Generated samples at training epoch=5

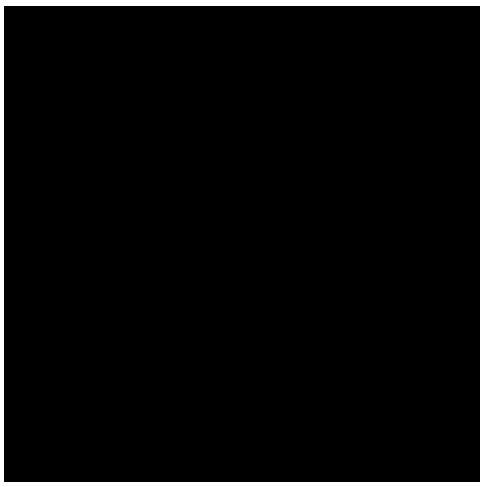

Figure 27: Generated samples at training epoch=49

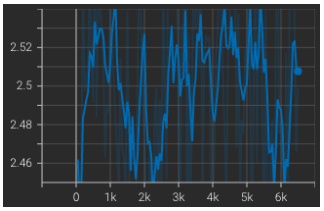

Figure 28: Reconstruction Loss Curve

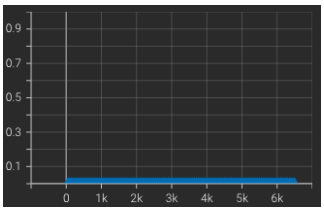

Figure 29: KLD Loss Curve

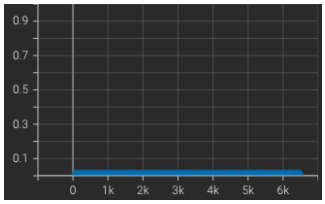

Figure 30: Overall Loss Curve

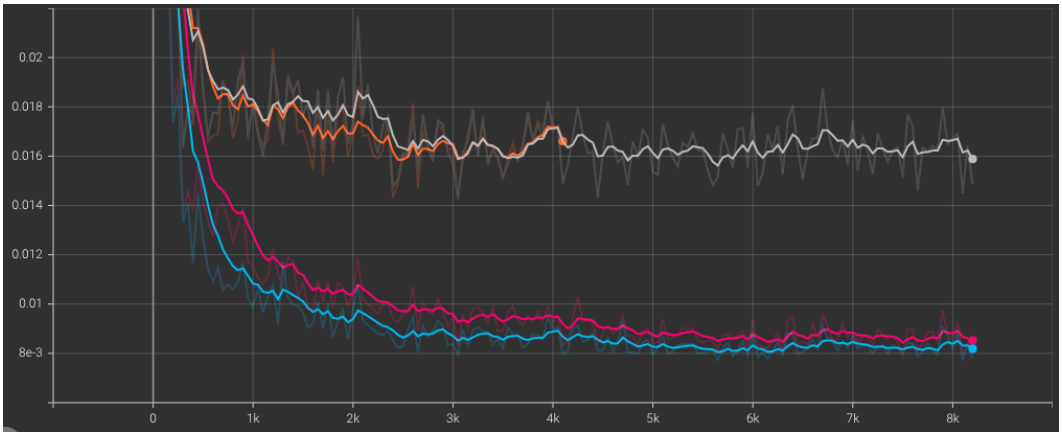

Figure 31: Combined loss curves for experiment - 1 (sky blue), 2 (pink), 4 (gray), 5 (orange), 6 (dark blue)

1.5 Summary

From the combined loss plot, one can observe that the blue curve corresponding to experiment 1 performs better compared to other choosen set of hyperparameters. The ideal set of hyperparameters that I could observe was - M = 64, $D_L = 128$, $N_E = 50/100$, $\alpha = 0.005$, $\beta = 0.0025$. Doubling the learning rate from 0.005 to 0.01 results in divergence of loss (due to large learning rate) as can be seen from experiment 3 figures. Changing the input patch size to M = 256 also results in incomprehensible output. It was observed that most of the loss curves achieved stability after 40-50 epochs. In most experiments, the IS scores starts from a value closer to 1 and as the training progresses it increases and settes between 1.5-2 (given that the loss is decreasing and sufficient epochs have been allowed to run). In contrast, the frauchet inception distance score starts from high values and decreases as the training progresses (given that the loss is decreasing). The generated synthetic images become more and more identical to the chest x-ray images with each epoch until the loss function stabilizes after which there is very minimal improvement in observation.