1 Question2

Build a GAN from scratch for CIFAR-10 dataset using PyTorch. Use CNN models for generator and discriminator networks. Optimize the learning rate and mention the best learning rate for the model. Plot generator results before, during (for few or all epochs) and after training.

Solution: First let us understand what GANs are and how they works . GANs or Generative Adversive Networks are a type of deep neural networks that consists of two parts a generator and a discriminator. The generator generates fake images while the discriminator tries to distinguish between the real and fake images.

The training process of a GANs involves a minimax game where the generator tries to generate images that can fool the discriminator and the discriminator tries to correctly. classify the images as real or fake. The ultimate goal of GANs is to generate images that are indistinguishable from real images.

To build a GAN for CIFAR-10n dataset using PyTorch, we need to follow these steps:

- Load the CIFAR-10 dataset using the torchvision library and preprocess the data
- Define the generator and discriminator networks. We can use CNN models for both generator and discriminator networks.
- Define the loss functions for both the generator and discriminator.
- Train the discriminator and generator alternately. First, train the discriminator on real and fake images, then train the generator to generate images that can fool the discriminator.
- Optimize the learning rate by trying different values and selecting the one that gives the best results. We can use a learning rate scheduler to adjust the learning rate during training
- Finally, using the trained generator , plot the results.

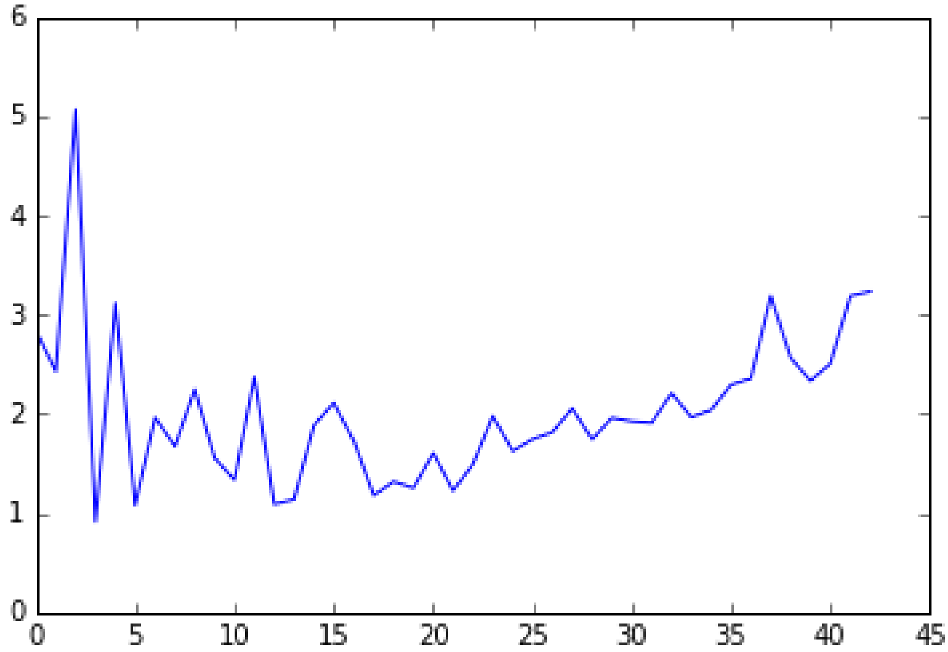

Figure 1: Generator loss

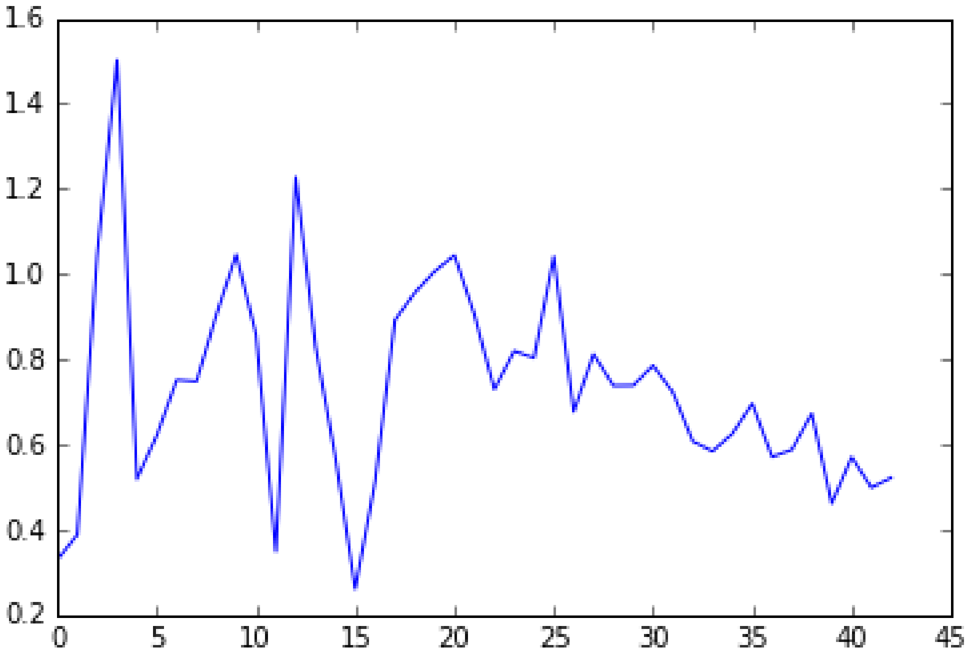

Figure 2: Discriminator loss

2 Question 1

Style GAN

2.1 Question 1.4

Explain the concept of style mixing in StyleGAN. How does it differ from traditional style transfer methods? How does it affect the generated images?

Solution: Style mixing is a technique used in the StyleGAN to generate new images by blending the styles of two or more randomly generated latent vectors. In Traditional style transfer methods, a style transfer methods, a single style is transferred from a reference image to a content image, resulting in a new image that combines the content of the original image with the style of the reference image. Style mixing, on the other hand, allows for the creation of new images that combine multiple styles in a controlled manner.

The basic idea behind style mixing in StyleGAN is to take two or more latent vectors, and to interpolate between their style vectors at a chosen layer in the generator network. The resulting vector is then passed through the generator network to produce a new image. By varying the degree of interpolation, we can control the amount of each style that is present in the resulting image.

The effect of style mixing on the generated images can be quite dramatic, depending on the chosen styles and the degree of mixing. For example, mixing two very different styles (e.g., a cartoon style and a photorealistic style) can result in images that are surreal and unusual, while mixing two similar styles (e.g., two different styles of landscape photography) can result in images that are more subtle and nuanced.

Overall, style mixing in StyleGAN allows for a high degree of control over the generated images, and can be used to create new and interesting styles that are not present in the original dataset. It differs from traditional style transfer methods in that it allows for the creation of entirely new images, rather than simply transferring the style of a single reference image to a content image.

2.2 Question 1.5

How does the depth of the chosen layer affect the quality of the generated images in style mixing? Why?
Solution: The depth of the chosen layer can have a significant impact on the quality of the generated images in style mixing. In general, deeper layers tend to capture more abstract and high-level features, while shallower layers capture more detailed and low-level features.

When performing style mixing, we are essentially blending the styles of two or more latent vectors at a chosen layer in the generator network. The depth of this layer will determine which features are being mixed and how they are combined. If the layer is too shallow, the resulting images may be too similar to the original dataset and lack diversity. On the other hand, if the layer is too deep, the resulting images may be too abstract and lack detail.

In practice, it is often useful to experiment with different layers to find the best balance between abstractness and detail. In general, intermediate layers tend to work well for style mixing, as they capture a good balance of both high-level and low-level features. However, the best layer to use will depend on the specific application and the desired style of the generated images.

2.3 Question1.1

Load a pretrained StyleGAN model and print its architecture. Choose a layer of interest to visualize and print its weights.
Solution The Style Generative Adversarial Network, or StyleGAN for short, is an extension to the GAN architecture that proposes large changes to the generator model, including the use of a mapping network to map points in latent space to an intermediate latent space, the use of the intermediate latent space to control style at each point in the generator model, and the introduction to noise as a source of variation at each point in the generator model. The other part of this question are explained in the question1.py file.

2.4 Question 1.2,1.3,1.6

SolutionThere solution are provided in the code file question1.py

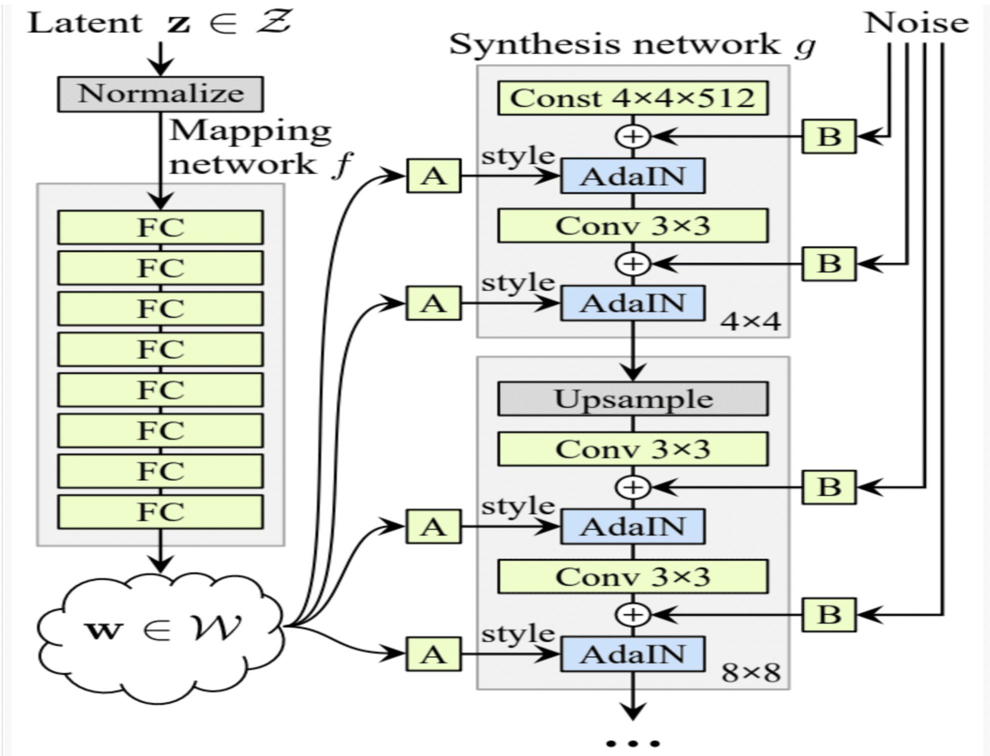

Figure 3: StyleGAN architecture