

```
!nvidia-smi
```

Thu Apr 18 06:12:17 2024

NVIDIA-SMI 535.104.05			Driver Version: 535.104.05		CUDA Version: 12.2		
GPU Name		Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr.	ECC
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage	GPU-Util	Compute M.	MIG M.
0	Tesla T4		Off	00000000:00:04.0 Off		0	
N/A	63C	P8	10W / 70W	0MiB / 15360MiB	0%	Default	N/A

Processes:							
GPU	GI	CI	PID	Type	Process name	GPU Memory	Usage
ID							
No running processes found							

```
!pip install tensorflow
```

Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-packages (2.15.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (24.3.25)
Requirement already satisfied: gast!=0.5.0,!0.5.1,!0.5.2,>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.9.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (18.1.1)
Requirement already satisfied: ml-dtypes~0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: numpy<2.0.0,>=1.23.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.25.2)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (24.0)
Requirement already satisfied: protobuf<=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<5.0.0dev,>=3.20.3 in /usr/local/lib/python
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (67.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.4.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (4.11.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.36.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.62.1)
Requirement already satisfied: tensorflow<2.16,>=2.15 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.2)
Requirement already satisfied: tensorflow-estimator<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: keras<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunparse>=1.6.0->tensorflow) (0.43)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>=2.15->tensorflc)
Requirement already satisfied: google-auth-oauthlib<2,>=0.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>=2.15->te
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>=2.15->tensorflow) (3.
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>=2.15->tensorflow)
Requirement already satisfied: tensorflow-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow<2.16,>=2.15->tensorflow) (3.
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorboar
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorflow)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorflow<2.16,>=
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from google-auth-oauthlib<2,>=0.5->t
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow<2.16,>=2.1
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow<2.16
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorflow<2.16
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug>=1.0.1->tensorflow<2.16,>=2.
Requirement already satisfied: pyasn1<0.7.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn1-modules>=0.2.1->google-auth<
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-oauthlib>=0.7.0->google-auth-oa

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, Dense, Flatten, Reshape, LeakyReLU, Dropout, UpSampling2D
import tensorflow_datasets as tfds
from matplotlib import pyplot as plt

gpus = tf.config.experimental.list_physical_devices('GPU')
for gpu in gpus:
    tf.config.experimental.set_memory_growth(gpu, True)
```

```
!pip install tensorflow_datasets
```

```
Requirement already satisfied: tensorflow_datasets in /usr/local/lib/python3.10/dist-packages (4.9.4)
Requirement already satisfied: absl-py in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (1.4.0)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (8.1.7)
Requirement already satisfied: dm-tree in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (0.1.8)
Requirement already satisfied: etils[enp,epath,etree]>=0.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (1.7.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (1.25.2)
Requirement already satisfied: promise in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (2.3)
Requirement already satisfied: protobuf>=3.20 in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (3.20.3)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (5.9.5)
Requirement already satisfied: requests>=2.19.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (2.31.0)
Requirement already satisfied: tensorflow-metadata in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (1.14.0)
Requirement already satisfied: termcolor in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (2.4.0)
Requirement already satisfied: toml in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (0.10.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (4.66.2)
Requirement already satisfied: wrapt in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (1.14.1)
Requirement already satisfied: array-reCORDS>=0.5.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow_datasets) (0.5.1)
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages (from etils[enp,epath,etree]>=0.9.0->tensorflow_dataset)
Requirement already satisfied: importlib_resources in /usr/local/lib/python3.10/dist-packages (from etils[enp,epath,etree]>=0.9.0->tensorflow_da)
Requirement already satisfied: typing_extensions in /usr/local/lib/python3.10/dist-packages (from etils[enp,epath,etree]>=0.9.0->tensorflow_da)
Requirement already satisfied: zipp in /usr/local/lib/python3.10/dist-packages (from etils[enp,epath,etree]>=0.9.0->tensorflow_datasets)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests>=2.19.0->tensorflow_da)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.19.0->tensorflow_datasets) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.19.0->tensorflow_datasets)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.19.0->tensorflow_datasets)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from promise->tensorflow_datasets) (1.16.0)
Requirement already satisfied: googleapis-common-protos<2,>=1.52.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow-metadata->tensorflow_datasets)
```

```
ds = tfds.load('fashion_mnist', split='train')
```

```
→ Downloading and preparing dataset 29.45 MiB (download: 29.45 MiB, generated: 36.42 MiB, total: 65.87 MiB) to /root/tensorflow_datasets/fashion_mnist
  DL Completed...: 100%      4/4 [00:02<00:00,  2.25 url/s]
  DL Size...: 100%      29/29 [00:02<00:00, 13.04 MiB/s]
  Extraction completed...: 100%      4/4 [00:02<00:00,  2.28 file/s]
```

Dataset fashion_mnist downloaded and prepared to /root/tensorflow_datasets/fashion_mnist/3.0.1. Subsequent calls will reuse this data.

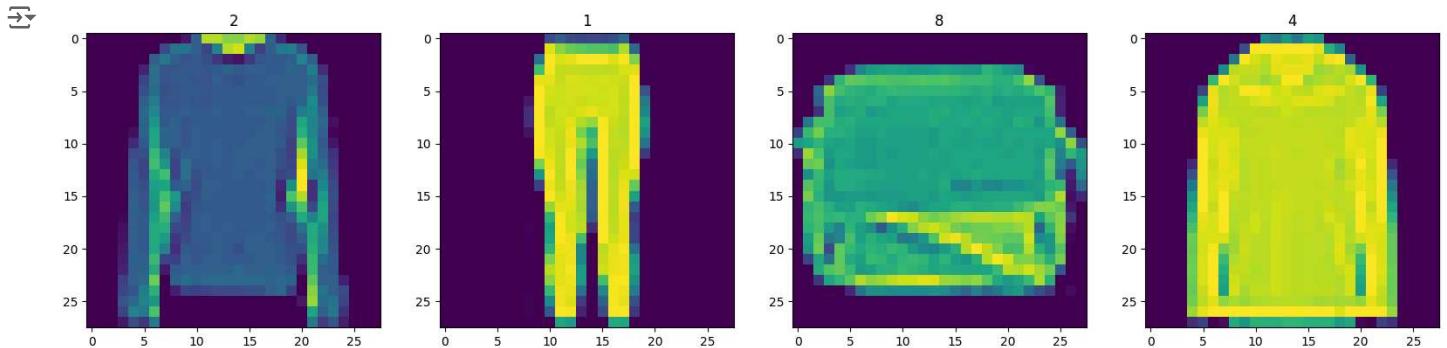
```
# Data Transformation: Scale and Vizualize Images
import numpy as np

# Setup data iterator
dataiterator = ds.as_numpy_iterator()

# Visualize some images from the dataset
fig, ax = plt.subplots(ncols=4, figsize=(20, 20))

# Loop four times and get images
for idx in range(4):
    # Grab an image and its label
    sample = dataiterator.next()
    image = np.squeeze(sample['image']) # Remove the single-dimensional entries
    label = sample['label']

    # Plot the image using a specific subplot
    ax[idx].imshow(image)
    ax[idx].title.set_text(label)
```



```
# Data Preprocessing: Scale and Batch the Images
def scale_images(data):
    # Scale the pixel values of the images between 0 and 1
    image = data['image']
    image=tf.cast(image, tf.float32) / 255.0
    return image

# Reload the dataset
ds = tfds.load('fashion_mnist', split='train')

# Apply the scale_images preprocessing step to the dataset
ds = ds.map(scale_images)

# Cache the dataset for faster processing during training
ds = ds.cache()

# Shuffle the dataset to add randomness to the training process
ds = ds.shuffle(60000)

# Batch the dataset into smaller groups (128 images per batch)
ds = ds.batch(128)

# Prefetch the dataset to improve performance during training
ds = ds.prefetch(64)

# Check the shape of a batch of images
ds.as_numpy_iterator().next().shape

# Import the Sequential API for building models
from tensorflow.keras.models import Sequential

# Import the layers required for the neural network
from tensorflow.keras.layers import (
    Conv2D, Dense, Flatten, Reshape, LeakyReLU, Dropout, UpSampling2D
)

def build_generator():
    model = Sequential()

    # First layer takes random noise and reshapes it to 7x7x128
    # This is the beginning of the generated image
    model.add(Dense(7 * 7 * 128, input_dim=128))
    model.add(LeakyReLU(0.2))
    model.add(Reshape((7, 7, 128)))

    # Upsampling block 1
    model.add(UpSampling2D())
    model.add(Conv2D(128, 5, padding='same'))
    model.add(LeakyReLU(0.2))

    # Upsampling block 2
    model.add(UpSampling2D())
    model.add(Conv2D(128, 5, padding='same'))
    model.add(LeakyReLU(0.2))

    # Convolutional block 1
    model.add(Conv2D(1, 5, padding='same'))
```

```

model.add(Conv2D(128, 4, padding='same'))
model.add(LeakyReLU(0.2))

# Convolutional block 2
model.add(Conv2D(128, 4, padding='same'))
model.add(LeakyReLU(0.2))

# Convolutional layer to get to one channel
model.add(Conv2D(1, 4, padding='same', activation='sigmoid'))

return model

# Build the generator model
generator = build_generator()
# Display the model summary
generator.summary()

```

→ Model: "sequential"

Layer (type)	Output Shape	Param #
<hr/>		
dense (Dense)	(None, 6272)	809088
leaky_re_lu (LeakyReLU)	(None, 6272)	0
reshape (Reshape)	(None, 7, 7, 128)	0
up_sampling2d (UpSampling2D)	(None, 14, 14, 128)	0
conv2d (Conv2D)	(None, 14, 14, 128)	409728
leaky_re_lu_1 (LeakyReLU)	(None, 14, 14, 128)	0
up_sampling2d_1 (UpSampling2D)	(None, 28, 28, 128)	0
conv2d_1 (Conv2D)	(None, 28, 28, 128)	409728
leaky_re_lu_2 (LeakyReLU)	(None, 28, 28, 128)	0
conv2d_2 (Conv2D)	(None, 28, 28, 128)	262272
leaky_re_lu_3 (LeakyReLU)	(None, 28, 28, 128)	0
conv2d_3 (Conv2D)	(None, 28, 28, 128)	262272
leaky_re_lu_4 (LeakyReLU)	(None, 28, 28, 128)	0
conv2d_4 (Conv2D)	(None, 28, 28, 1)	2049
<hr/>		
Total params: 2155137 (8.22 MB)		
Trainable params: 2155137 (8.22 MB)		
Non-trainable params: 0 (0.00 Byte)		

```

def build_discriminator():
    model = Sequential()

    # First Convolutional Block
    model.add(Conv2D(32, 5, input_shape=(28, 28, 1)))
    model.add(LeakyReLU(0.2))
    model.add(Dropout(0.4))

    # Second Convolutional Block
    model.add(Conv2D(64, 5))
    model.add(LeakyReLU(0.2))
    model.add(Dropout(0.4))

    # Third Convolutional Block
    model.add(Conv2D(128, 5))
    model.add(LeakyReLU(0.2))
    model.add(Dropout(0.4))

    # Fourth Convolutional Block
    model.add(Conv2D(256, 5))
    model.add(LeakyReLU(0.2))
    model.add(Dropout(0.4))

```

```
# Flatten the output and pass it through a dense layer
model.add(Flatten())
model.add(Dropout(0.4))
model.add(Dense(1, activation='sigmoid'))

return model

# Build the discriminator model
discriminator = build_discriminator()
# Display the model summary
discriminator.summary()

→ Model: "sequential_1"

```

Layer (type)	Output Shape	Param #
conv2d_5 (Conv2D)	(None, 24, 24, 32)	832
leaky_re_lu_5 (LeakyReLU)	(None, 24, 24, 32)	0
dropout (Dropout)	(None, 24, 24, 32)	0
conv2d_6 (Conv2D)	(None, 20, 20, 64)	51264
leaky_re_lu_6 (LeakyReLU)	(None, 20, 20, 64)	0
dropout_1 (Dropout)	(None, 20, 20, 64)	0
conv2d_7 (Conv2D)	(None, 16, 16, 128)	204928
leaky_re_lu_7 (LeakyReLU)	(None, 16, 16, 128)	0
dropout_2 (Dropout)	(None, 16, 16, 128)	0
conv2d_8 (Conv2D)	(None, 12, 12, 256)	819456
leaky_re_lu_8 (LeakyReLU)	(None, 12, 12, 256)	0
dropout_3 (Dropout)	(None, 12, 12, 256)	0
flatten (Flatten)	(None, 36864)	0
dropout_4 (Dropout)	(None, 36864)	0
dense_1 (Dense)	(None, 1)	36865

Total params: 1113345 (4.25 MB)
Trainable params: 1113345 (4.25 MB)
Non-trainable params: 0 (0.00 Byte)

```
# Import the Adam optimizer and Binary Cross Entropy loss function
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.losses import BinaryCrossentropy

# Define the optimizers for the generator and discriminator
g_opt = Adam(learning_rate=0.0001) # Generator optimizer
d_opt = Adam(learning_rate=0.00001) # Discriminator optimizer

# Define the loss functions for the generator and discriminator
g_loss = BinaryCrossentropy() # Generator loss function
d_loss = BinaryCrossentropy() # Discriminator loss function

from tensorflow.keras.models import Model

class FashionGAN(Model):
    def __init__(self, generator, discriminator, *args, **kwargs):
        # Pass through args and kwargs to the base class
        super().__init__(*args, **kwargs)

        # Create attributes for generator and discriminator models
        self.generator = generator
        self.discriminator = discriminator

    def compile(self, g_opt, d_opt, g_loss, d_loss, *args, **kwargs):
        # Compile with the base class
        super().compile(*args, **kwargs)
```

```

# Create attributes for optimizers and loss functions
self.g_opt = g_opt
self.d_opt = d_opt
self.g_loss = g_loss
self.d_loss = d_loss

def train_step(self, batch):
    # Get the data for real images
    real_images = batch
    # Generate fake images using the generator with random noise as input
    fake_images = self.generator(tf.random.normal((128, 128, 1)), training=False)

    # Train the discriminator
    with tf.GradientTape() as d_tape:
        # Pass real and fake images through the discriminator model
        yhat_real = self.discriminator(real_images, training=True)
        yhat_fake = self.discriminator(fake_images, training=True)
        yhat_realfake = tf.concat([yhat_real, yhat_fake], axis=0)

        # Create labels for real and fake images
        y_realfake = tf.concat([tf.zeros_like(yhat_real), tf.ones_like(yhat_fake)], axis=0)

        # Add some noise to the true outputs to make training more robust
        noise_real = 0.15 * tf.random.uniform(tf.shape(yhat_real))
        noise_fake = -0.15 * tf.random.uniform(tf.shape(yhat_fake))
        y_realfake += tf.concat([noise_real, noise_fake], axis=0)

        # Calculate the total discriminator loss
        total_d_loss = self.d_loss(y_realfake, yhat_realfake)

    # Apply backpropagation and update discriminator weights
    dgrad = d_tape.gradient(total_d_loss, self.discriminator.trainable_variables)
    self.d_opt.apply_gradients(zip(dgrad, self.discriminator.trainable_variables))

    # Train the generator
    with tf.GradientTape() as g_tape:
        # Generate new images using the generator with random noise as input
        gen_images = self.generator(tf.random.normal((128, 128, 1)), training=True)

        # Create the predicted labels (should be close to 1 as they are fake images)
        predicted_labels = self.discriminator(gen_images, training=False)

        # Calculate the total generator loss (tricking the discriminator to classify the fake images as real)
        total_g_loss = self.g_loss(tf.zeros_like(predicted_labels), predicted_labels)

    # Apply backpropagation and update generator weights
    ggrad = g_tape.gradient(total_g_loss, self.generator.trainable_variables)
    self.g_opt.apply_gradients(zip(ggrad, self.generator.trainable_variables))

    return {"d_loss": total_d_loss, "g_loss": total_g_loss}

# Create an instance of the FashionGAN model
fashgan = FashionGAN(generator, discriminator)

# Compile the model with the optimizers and loss functions
fashgan.compile(g_opt, d_opt, g_loss, d_loss)

import os

# Create the directory "images" if it doesn't exist
if not os.path.exists('images'):
    os.makedirs('images')

import os
from tensorflow.keras.preprocessing.image import array_to_img
from tensorflow.keras.callbacks import Callback

class ModelMonitor(Callback):
    def __init__(self, num_img=3, latent_dim=128):
        self.num_img = num_img
        self.latent_dim = latent_dim

    def on_epoch_end(self, epoch, logs=None):
        # Generate random latent vectors as input to the generator
        random_latent_vectors = tf.random.uniform((self.num_img, self.latent_dim, 1))

```

```

# Generate fake images using the generator
generated_images = self.model.generator(random_latent_vectors)
generated_images *= 255
generated_images.numpy()
for i in range(self.num_img):
    # Save the generated images to disk
    img = array_to_img(generated_images[i])
    img.save(os.path.join('images', f'generated_img_{epoch}_{i}.png'))

# Train the GAN model
hist = fashgan.fit(ds, epochs=20, callbacks=[ModelMonitor()])

```

Epoch 1/20
469/469 [=====] - 97s 160ms/step - d_loss: 0.4577 - g_loss: 1.2261
Epoch 2/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6780 - g_loss: 0.7725
Epoch 3/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6922 - g_loss: 0.6323
Epoch 4/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6941 - g_loss: 0.6492
Epoch 5/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6887 - g_loss: 0.6576
Epoch 6/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6897 - g_loss: 0.6422
Epoch 7/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6958 - g_loss: 0.6452
Epoch 8/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6927 - g_loss: 0.6493
Epoch 9/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6824 - g_loss: 0.6747
Epoch 10/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.5379 - g_loss: 0.5932
Epoch 11/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.3496 - g_loss: 0.0484
Epoch 12/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.3720 - g_loss: 0.0605
Epoch 13/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.3073 - g_loss: 0.0304
Epoch 14/20
469/469 [=====] - 78s 165ms/step - d_loss: 0.3394 - g_loss: 0.0285
Epoch 15/20
469/469 [=====] - 78s 165ms/step - d_loss: 0.7272 - g_loss: 0.4936
Epoch 16/20
469/469 [=====] - 77s 165ms/step - d_loss: 0.4018 - g_loss: 0.2931
Epoch 17/20
469/469 [=====] - 77s 165ms/step - d_loss: 0.3793 - g_loss: 0.0555
Epoch 18/20
469/469 [=====] - 77s 165ms/step - d_loss: 0.3149 - g_loss: 0.0810
Epoch 19/20
469/469 [=====] - 77s 165ms/step - d_loss: 0.3334 - g_loss: 0.0532
Epoch 20/20
469/469 [=====] - 78s 166ms/step - d_loss: 0.6090 - g_loss: 0.2860

```

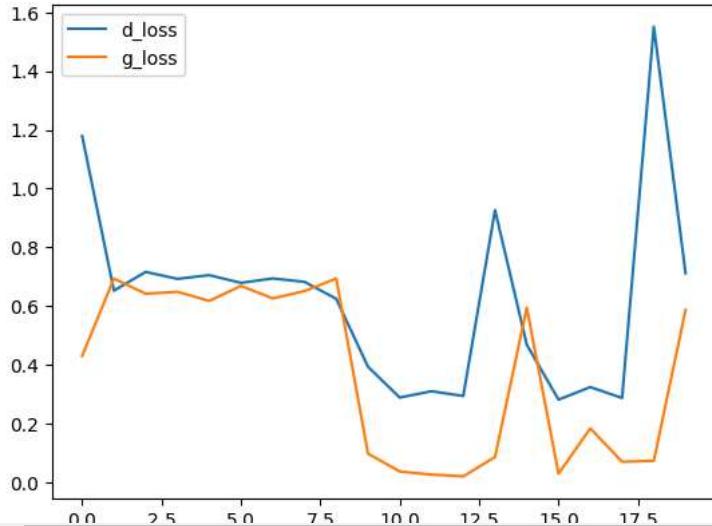
import matplotlib.pyplot as plt

# Plot the discriminator and generator losses
plt.suptitle('Loss')
plt.plot(hist.history['d_loss'], label='d_loss')
plt.plot(hist.history['g_loss'], label='g_loss')
plt.legend()
plt.show()

```



Loss



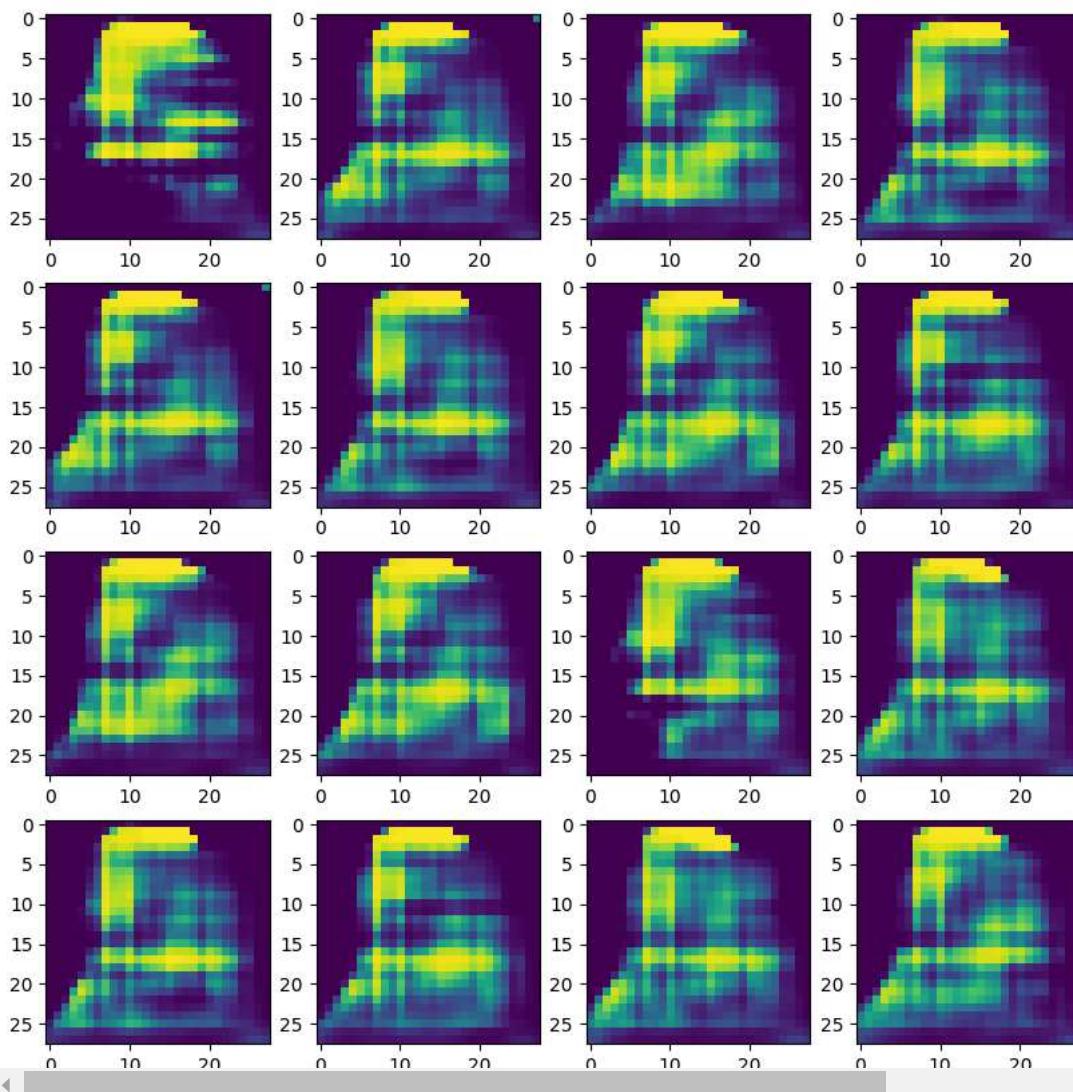
```
generator.save('generator.h5')
discriminator.save('discriminator.h5')
```

```
↳ /usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as an HDF5 file via `saving_api.save_model()
WARNING:tensorflow:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until
WARNING:tensorflow:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until
```

```
# Load the weights of the trained generator
generator.load_weights('generator.h5')
discriminator.load_weights('discriminator.h5')
# Generate new fashion images
imgs = generator.predict(tf.random.normal((16, 128, 1)))

# Plot the generated images
fig, ax = plt.subplots(ncols=4, nrows=4, figsize=(10, 10))
for r in range(4):
    for c in range(4):
        ax[r][c].imshow(imgs[(r + 1) * (c + 1) - 1])
```

1/1 [=====] - 0s 18ms/step



```
from PIL import Image
```

```
# Load the image using PIL
img_path = '/content/images/generated_img_0_0.png'
img = Image.open(img_path)

# Get the image size (width, height)
img_size = img.size
print(f"The shape of {img_path} is {img_size}")
```

The shape of /content/images/generated_img_0_0.png is (28, 28)

```
!pip install tensorflow
!pip install keras
!pip install matplotlib
```

```
Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-packages (2.15.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (24.3.25)
Requirement already satisfied: gast!=0.5.0,!>=0.5.1,!>=0.5.2,>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.9.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (18.1.1)
Requirement already satisfied: ml-dtypes>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: numpy>=2.0.0,>=1.23.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.25.2)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (24.0)
Requirement already satisfied: protobuf>=4.21.0,!>=4.21.1,!>=4.21.2,!>=4.21.3,!>=4.21.4,!>=4.21.5,<5.0.0dev,>=3.20.3 in /usr/local/lib/python
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (67.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.4.0)
```

```
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (4.11.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (0.36.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (1.62.1)
Requirement already satisfied: tensorboard<2.16,>=2.15 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.2)
Requirement already satisfied: tensorflow-estimator<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: keras<2.16,>=2.15.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunparse>=1.6.0->tensorflow) (0.43.)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflc)
Requirement already satisfied: google-auth-oauthlib<2,>=0.5 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->te)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow) (3.)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow)
Requirement already satisfied: tensorflow-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflc)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboard<2.16,>=2.15->tensorflow) (3.)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorboar)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorboard)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorboard<2.16,>=2.15)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from google-auth-oauthlib<2,>=0.5->t)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorboar)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorboard<2.16,>=2.15)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorboard<2.16)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorboard<2.16)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug>=1.0.1->tensorboard<2.16,>=2.15)
Requirement already satisfied: pyasn1<0.7.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboar)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-oauthlib>=0.7.0->google-auth-oa)
Requirement already satisfied: keras in /usr/local/lib/python3.10/dist-packages (2.15.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.5)
Requirement already satisfied: numpy>=1.20 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.25.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (24.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
```

```
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
```

```
print(generator.input_shape)
```

→ (None, 128)