LAB EXPERIMENT NO. 8

AIM :

Compare the performance of PCA and Autoencoders on a given dataset

THEORY:

What is Dimensionality Reduction?

In machine learning classification problems, there are often too many factors on the basis of which the final classification is done. These factors are basically variables called features. The higher the number of features, the harder it gets to visualize the training set and then work on it. Sometimes, most of these features are correlated, and hence redundant. This is where dimensionality reduction algorithms come into play. Dimensionality reduction is the process of reducing the number of random variables under consideration, by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.

Principle Component Analysis

Principle Component Analysis is an unsupervised technique where the original data is projected to the direction of high variance. These directions of high variance are orthogonal to each other resulting in very low or almost close to 0 correlation in the projected data. These features transformation is linear and the methodology to do it is:

Step 1: Calculate the Correlation matrix data consisting of n dimensions. The Correlation matrix will be of shape n*n.

Step 2: Calculate the Eigenvectors and Eigenvalues of this matrix.

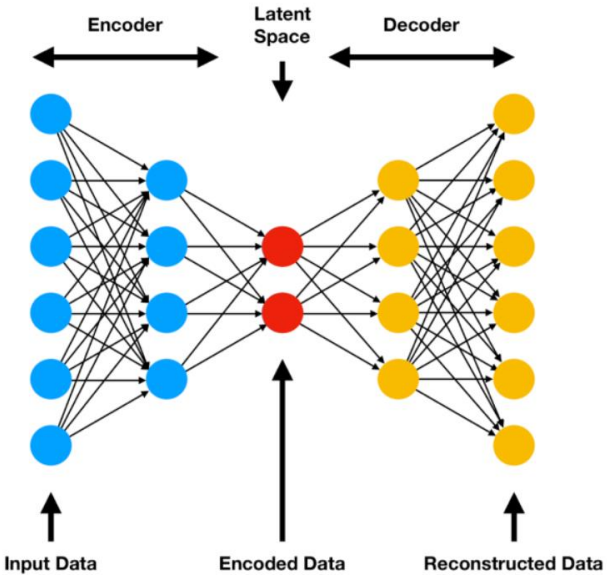
Step 3: Take the first k-eigenvectors with the highest eigenvalues.

Step 4: Project the original dataset into these k eigenvectors resulting in k dimensions where $k \leq n$.

Autoencoders

Autoencoder is an unsupervised artificial neural network that compresses the data to lower dimension and then reconstructs the input back. Autoencoder finds the representation of the data in

a lower dimension by focusing more on the important features getting rid of noise and redundancy. It's based on Encoder-Decoder architecture, where encoder encodes the high-dimensional data to lower-dimension and decoder takes the lower-dimensional data and tries to reconstruct the original high-dimensional data.



Tasks to be performed:

1. **Use the Iris Dataset present in the scikit-learn library**
2. **Create an Auto Encoder and fit it with our data using 3 neurons in the dense layer**
3. **Use encoded layer to encode the training input**
4. **Plot loss for different encoders [PCA, Linear Autoencoder, Sigmoid based Non-Linear Autoencoder, ReLU based Non-Linear Auto encoder]**

ml2-60009220202-exp8

October 12, 2024

1 NAME: AYUSHI SINGH

2 SAP: 60009220202

3 LAB 8

4 Step 1: Load the Iris Dataset

[19]:
```
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
import numpy as np

data = load_iris()
X = data['data']
y = data['target']

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

5 Step 2: Implement PCA

[20]:
```
from sklearn.decomposition import PCA

pca = PCA(n_components=3)
X_pca = pca.fit_transform(X_scaled)
```

6 Step 3: Implement Autoencoders (Linear, Sigmoid, and ReLU based)

Linear Autoencoder

[21]:
```
import tensorflow as tf
from tensorflow.keras import layers, models

input_dim = X_scaled.shape[1]
```

```
encoding_dim = 3

linear_autoencoder = models.Sequential([
    layers.Dense(encoding_dim, input_shape=(input_dim,), activation='linear'),
    layers.Dense(input_dim, activation='linear')
])

linear_autoencoder.compile(optimizer='adam', loss='mse')
history_linear = linear_autoencoder.fit(X_scaled, X_scaled, epochs=100,␣
 ↪batch_size=8, verbose=0)
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87:
UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When
using Sequential models, prefer using an `Input(shape)` object as the first
layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Sigmoid-based Non-Linear Autoencoder

[22]:
```
sigmoid_autoencoder = models.Sequential([
    layers.Dense(encoding_dim, input_shape=(input_dim,), activation='sigmoid'),
    layers.Dense(input_dim, activation='sigmoid')
])

sigmoid_autoencoder.compile(optimizer='adam', loss='mse')
history_sigmoid = sigmoid_autoencoder.fit(X_scaled, X_scaled, epochs=100,␣
 ↪batch_size=8, verbose=0)
```

ReLU-based Non-Linear Autoencoder

[23]:
```
relu_autoencoder = models.Sequential([
    layers.Dense(encoding_dim, input_shape=(input_dim,), activation='relu'),
    layers.Dense(input_dim, activation='relu')
])

relu_autoencoder.compile(optimizer='adam', loss='mse')
history_relu = relu_autoencoder.fit(X_scaled, X_scaled, epochs=100,␣
 ↪batch_size=8, verbose=0)
```

7 Step 4: Encode the Input Data Using Autoencoders

[24]:
```
linear_autoencoder.compile(optimizer='adam', loss='mse')
linear_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16,␣
 ↪shuffle=True, validation_split=0.2)


sigmoid_autoencoder.compile(optimizer='adam', loss='mse')
```

```
sigmoid_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16,␣
 ↪shuffle=True, validation_split=0.2)


relu_autoencoder.compile(optimizer='adam', loss='mse')
relu_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16,␣
 ↪shuffle=True, validation_split=0.2)
```

```
Epoch 1/50
8/8             1s 23ms/step - loss:
0.0525 - val_loss: 0.0741
Epoch 2/50
8/8             0s 6ms/step - loss:
0.0483 - val_loss: 0.0746
Epoch 3/50
8/8             0s 7ms/step - loss:
0.0429 - val_loss: 0.0746
Epoch 4/50
8/8             0s 6ms/step - loss:
0.0404 - val_loss: 0.0738
Epoch 5/50
8/8             0s 6ms/step - loss:
0.0434 - val_loss: 0.0737
Epoch 6/50
8/8             0s 5ms/step - loss:
0.0457 - val_loss: 0.0729
Epoch 7/50
8/8             0s 5ms/step - loss:
0.0454 - val_loss: 0.0725
Epoch 8/50
8/8             0s 9ms/step - loss:
0.0390 - val_loss: 0.0722
Epoch 9/50
8/8             0s 8ms/step - loss:
0.0416 - val_loss: 0.0719
Epoch 10/50
8/8             0s 10ms/step - loss:
0.0368 - val_loss: 0.0712
Epoch 11/50
8/8             0s 8ms/step - loss:
0.0376 - val_loss: 0.0706
Epoch 12/50
8/8             0s 7ms/step - loss:
0.0398 - val_loss: 0.0702
Epoch 13/50
8/8             0s 12ms/step - loss:
0.0408 - val_loss: 0.0700
```

```
Epoch 14/50
8/8             0s 10ms/step - loss:
0.0395 - val_loss: 0.0694
Epoch 15/50
8/8             0s 10ms/step - loss:
0.0408 - val_loss: 0.0692
Epoch 16/50
8/8             0s 10ms/step - loss:
0.0376 - val_loss: 0.0687
Epoch 17/50
8/8             0s 10ms/step - loss:
0.0346 - val_loss: 0.0679
Epoch 18/50
8/8             0s 11ms/step - loss:
0.0335 - val_loss: 0.0675
Epoch 19/50
8/8             0s 11ms/step - loss:
0.0466 - val_loss: 0.0671
Epoch 20/50
8/8             0s 11ms/step - loss:
0.0346 - val_loss: 0.0667
Epoch 21/50
8/8             0s 11ms/step - loss:
0.0401 - val_loss: 0.0663
Epoch 22/50
8/8             0s 11ms/step - loss:
0.0405 - val_loss: 0.0657
Epoch 23/50
8/8             0s 8ms/step - loss:
0.0371 - val_loss: 0.0654
Epoch 24/50
8/8             0s 8ms/step - loss:
0.0353 - val_loss: 0.0655
Epoch 25/50
8/8             0s 8ms/step - loss:
0.0303 - val_loss: 0.0651
Epoch 26/50
8/8             0s 14ms/step - loss:
0.0375 - val_loss: 0.0642
Epoch 27/50
8/8             0s 7ms/step - loss:
0.0322 - val_loss: 0.0638
Epoch 28/50
8/8             0s 8ms/step - loss:
0.0300 - val_loss: 0.0635
Epoch 29/50
8/8             0s 5ms/step - loss:
0.0363 - val_loss: 0.0631
```

```
Epoch 30/50
8/8             0s 6ms/step - loss:
0.0400 - val_loss: 0.0628
Epoch 31/50
8/8             0s 8ms/step - loss:
0.0362 - val_loss: 0.0625
Epoch 32/50
8/8             0s 5ms/step - loss:
0.0383 - val_loss: 0.0624
Epoch 33/50
8/8             0s 6ms/step - loss:
0.0371 - val_loss: 0.0619
Epoch 34/50
8/8             0s 6ms/step - loss:
0.0364 - val_loss: 0.0617
Epoch 35/50
8/8             0s 7ms/step - loss:
0.0333 - val_loss: 0.0612
Epoch 36/50
8/8             0s 7ms/step - loss:
0.0380 - val_loss: 0.0607
Epoch 37/50
8/8             0s 6ms/step - loss:
0.0347 - val_loss: 0.0605
Epoch 38/50
8/8             0s 6ms/step - loss:
0.0358 - val_loss: 0.0598
Epoch 39/50
8/8             0s 5ms/step - loss:
0.0317 - val_loss: 0.0596
Epoch 40/50
8/8             0s 6ms/step - loss:
0.0296 - val_loss: 0.0594
Epoch 41/50
8/8             0s 5ms/step - loss:
0.0291 - val_loss: 0.0593
Epoch 42/50
8/8             0s 8ms/step - loss:
0.0317 - val_loss: 0.0587
Epoch 43/50
8/8             0s 5ms/step - loss:
0.0299 - val_loss: 0.0583
Epoch 44/50
8/8             0s 9ms/step - loss:
0.0277 - val_loss: 0.0578
Epoch 45/50
8/8             0s 10ms/step - loss:
0.0300 - val_loss: 0.0574
```

```
Epoch 46/50
8/8             0s 8ms/step - loss:
0.0281 - val_loss: 0.0571
Epoch 47/50
8/8             0s 6ms/step - loss:
0.0316 - val_loss: 0.0567
Epoch 48/50
8/8             0s 8ms/step - loss:
0.0374 - val_loss: 0.0562
Epoch 49/50
8/8             0s 8ms/step - loss:
0.0283 - val_loss: 0.0563
Epoch 50/50
8/8             0s 8ms/step - loss:
0.0316 - val_loss: 0.0559
Epoch 1/50
8/8             1s 23ms/step - loss:
0.8354 - val_loss: 0.4930
Epoch 2/50
8/8             0s 6ms/step - loss:
0.9238 - val_loss: 0.4927
Epoch 3/50
8/8             0s 5ms/step - loss:
0.8260 - val_loss: 0.4925
Epoch 4/50
8/8             0s 7ms/step - loss:
0.8763 - val_loss: 0.4919
Epoch 5/50
8/8             0s 6ms/step - loss:
0.8378 - val_loss: 0.4914
Epoch 6/50
8/8             0s 6ms/step - loss:
0.9062 - val_loss: 0.4907
Epoch 7/50
8/8             0s 5ms/step - loss:
0.8601 - val_loss: 0.4901
Epoch 8/50
8/8             0s 5ms/step - loss:
0.9499 - val_loss: 0.4896
Epoch 9/50
8/8             0s 6ms/step - loss:
0.8981 - val_loss: 0.4889
Epoch 10/50
8/8             0s 5ms/step - loss:
0.9054 - val_loss: 0.4883
Epoch 11/50
8/8             0s 6ms/step - loss:
0.8681 - val_loss: 0.4877
```

```
Epoch 12/50
8/8             0s 5ms/step - loss:
0.9075 - val_loss: 0.4869
Epoch 13/50
8/8             0s 7ms/step - loss:
0.8791 - val_loss: 0.4861
Epoch 14/50
8/8             0s 7ms/step - loss:
0.8398 - val_loss: 0.4855
Epoch 15/50
8/8             0s 6ms/step - loss:
0.8755 - val_loss: 0.4847
Epoch 16/50
8/8             0s 10ms/step - loss:
0.8527 - val_loss: 0.4843
Epoch 17/50
8/8             0s 8ms/step - loss:
0.8915 - val_loss: 0.4835
Epoch 18/50
8/8             0s 7ms/step - loss:
0.8399 - val_loss: 0.4826
Epoch 19/50
8/8             0s 6ms/step - loss:
0.8097 - val_loss: 0.4817
Epoch 20/50
8/8             0s 9ms/step - loss:
0.8751 - val_loss: 0.4808
Epoch 21/50
8/8             0s 5ms/step - loss:
0.8905 - val_loss: 0.4799
Epoch 22/50
8/8             0s 6ms/step - loss:
0.9536 - val_loss: 0.4791
Epoch 23/50
8/8             0s 9ms/step - loss:
0.8604 - val_loss: 0.4781
Epoch 24/50
8/8             0s 7ms/step - loss:
0.8528 - val_loss: 0.4773
Epoch 25/50
8/8             0s 5ms/step - loss:
0.8627 - val_loss: 0.4766
Epoch 26/50
8/8             0s 8ms/step - loss:
0.8458 - val_loss: 0.4759
Epoch 27/50
8/8             0s 8ms/step - loss:
0.9006 - val_loss: 0.4752
```

```
Epoch 28/50
8/8             0s 6ms/step - loss:
0.8409 - val_loss: 0.4741
Epoch 29/50
8/8             0s 8ms/step - loss:
0.9425 - val_loss: 0.4732
Epoch 30/50
8/8             0s 6ms/step - loss:
0.8807 - val_loss: 0.4722
Epoch 31/50
8/8             0s 6ms/step - loss:
0.8526 - val_loss: 0.4714
Epoch 32/50
8/8             0s 7ms/step - loss:
0.7871 - val_loss: 0.4705
Epoch 33/50
8/8             0s 6ms/step - loss:
0.9123 - val_loss: 0.4695
Epoch 34/50
8/8             0s 6ms/step - loss:
0.8833 - val_loss: 0.4686
Epoch 35/50
8/8             0s 7ms/step - loss:
0.8498 - val_loss: 0.4677
Epoch 36/50
8/8             0s 6ms/step - loss:
0.7922 - val_loss: 0.4668
Epoch 37/50
8/8             0s 6ms/step - loss:
0.8785 - val_loss: 0.4661
Epoch 38/50
8/8             0s 6ms/step - loss:
0.9586 - val_loss: 0.4654
Epoch 39/50
8/8             0s 5ms/step - loss:
0.8832 - val_loss: 0.4645
Epoch 40/50
8/8             0s 6ms/step - loss:
0.9342 - val_loss: 0.4635
Epoch 41/50
8/8             0s 8ms/step - loss:
0.8449 - val_loss: 0.4626
Epoch 42/50
8/8             0s 7ms/step - loss:
0.8853 - val_loss: 0.4617
Epoch 43/50
8/8             0s 6ms/step - loss:
0.8962 - val_loss: 0.4608
```

```
Epoch 44/50
8/8             0s 6ms/step - loss:
0.9268 - val_loss: 0.4599
Epoch 45/50
8/8             0s 10ms/step - loss:
0.9273 - val_loss: 0.4590
Epoch 46/50
8/8             0s 6ms/step - loss:
0.9082 - val_loss: 0.4582
Epoch 47/50
8/8             0s 6ms/step - loss:
0.9476 - val_loss: 0.4574
Epoch 48/50
8/8             0s 6ms/step - loss:
0.8505 - val_loss: 0.4565
Epoch 49/50
8/8             0s 6ms/step - loss:
0.8344 - val_loss: 0.4558
Epoch 50/50
8/8             0s 7ms/step - loss:
0.9327 - val_loss: 0.4549
Epoch 1/50
8/8             1s 26ms/step - loss:
0.8619 - val_loss: 0.7541
Epoch 2/50
8/8             0s 8ms/step - loss:
0.9813 - val_loss: 0.7540
Epoch 3/50
8/8             0s 10ms/step - loss:
0.9485 - val_loss: 0.7540
Epoch 4/50
8/8             0s 10ms/step - loss:
0.9410 - val_loss: 0.7539
Epoch 5/50
8/8             0s 10ms/step - loss:
1.0033 - val_loss: 0.7538
Epoch 6/50
8/8             0s 10ms/step - loss:
0.9873 - val_loss: 0.7537
Epoch 7/50
8/8             0s 7ms/step - loss:
0.8637 - val_loss: 0.7536
Epoch 8/50
8/8             0s 13ms/step - loss:
0.9366 - val_loss: 0.7535
Epoch 9/50
8/8             0s 11ms/step - loss:
0.9182 - val_loss: 0.7536
```

```
Epoch 10/50
8/8             0s 7ms/step - loss:
0.9187 - val_loss: 0.7534
Epoch 11/50
8/8             0s 8ms/step - loss:
0.9899 - val_loss: 0.7533
Epoch 12/50
8/8             0s 10ms/step - loss:
0.8495 - val_loss: 0.7532
Epoch 13/50
8/8             0s 10ms/step - loss:
0.9535 - val_loss: 0.7532
Epoch 14/50
8/8             0s 11ms/step - loss:
0.9451 - val_loss: 0.7532
Epoch 15/50
8/8             0s 12ms/step - loss:
0.8759 - val_loss: 0.7531
Epoch 16/50
8/8             0s 11ms/step - loss:
0.8876 - val_loss: 0.7529
Epoch 17/50
8/8             0s 11ms/step - loss:
0.8519 - val_loss: 0.7529
Epoch 18/50
8/8             0s 11ms/step - loss:
0.9378 - val_loss: 0.7528
Epoch 19/50
8/8             0s 11ms/step - loss:
0.9979 - val_loss: 0.7529
Epoch 20/50
8/8             0s 9ms/step - loss:
0.9144 - val_loss: 0.7527
Epoch 21/50
8/8             0s 9ms/step - loss:
0.8945 - val_loss: 0.7527
Epoch 22/50
8/8             0s 11ms/step - loss:
0.9197 - val_loss: 0.7526
Epoch 23/50
8/8             0s 9ms/step - loss:
0.9597 - val_loss: 0.7526
Epoch 24/50
8/8             0s 6ms/step - loss:
0.9252 - val_loss: 0.7526
Epoch 25/50
8/8             0s 6ms/step - loss:
0.9710 - val_loss: 0.7524
```

```
Epoch 26/50
8/8             0s 6ms/step - loss:
0.9248 - val_loss: 0.7524
Epoch 27/50
8/8             0s 6ms/step - loss:
0.9112 - val_loss: 0.7523
Epoch 28/50
8/8             0s 6ms/step - loss:
0.8992 - val_loss: 0.7523
Epoch 29/50
8/8             0s 9ms/step - loss:
0.8946 - val_loss: 0.7522
Epoch 30/50
8/8             0s 6ms/step - loss:
1.0203 - val_loss: 0.7522
Epoch 31/50
8/8             0s 6ms/step - loss:
0.8728 - val_loss: 0.7520
Epoch 32/50
8/8             0s 6ms/step - loss:
0.8110 - val_loss: 0.7520
Epoch 33/50
8/8             0s 6ms/step - loss:
0.8673 - val_loss: 0.7520
Epoch 34/50
8/8             0s 7ms/step - loss:
0.8636 - val_loss: 0.7519
Epoch 35/50
8/8             0s 6ms/step - loss:
0.9693 - val_loss: 0.7518
Epoch 36/50
8/8             0s 5ms/step - loss:
0.9046 - val_loss: 0.7517
Epoch 37/50
8/8             0s 6ms/step - loss:
0.8757 - val_loss: 0.7517
Epoch 38/50
8/8             0s 8ms/step - loss:
0.9743 - val_loss: 0.7516
Epoch 39/50
8/8             0s 8ms/step - loss:
0.9430 - val_loss: 0.7515
Epoch 40/50
8/8             0s 6ms/step - loss:
0.9207 - val_loss: 0.7515
Epoch 41/50
8/8             0s 6ms/step - loss:
0.9226 - val_loss: 0.7514
```

```
Epoch 42/50
8/8             0s 8ms/step - loss:
0.9655 - val_loss: 0.7514
Epoch 43/50
8/8             0s 7ms/step - loss:
0.9571 - val_loss: 0.7513
Epoch 44/50
8/8             0s 7ms/step - loss:
0.9359 - val_loss: 0.7514
Epoch 45/50
8/8             0s 6ms/step - loss:
0.9372 - val_loss: 0.7513
Epoch 46/50
8/8             0s 6ms/step - loss:
0.9059 - val_loss: 0.7512
Epoch 47/50
8/8             0s 6ms/step - loss:
0.9200 - val_loss: 0.7511
Epoch 48/50
8/8             0s 6ms/step - loss:
0.8775 - val_loss: 0.7511
Epoch 49/50
8/8             0s 7ms/step - loss:
0.8561 - val_loss: 0.7510
Epoch 50/50
8/8             0s 8ms/step - loss:
0.8462 - val_loss: 0.7510
```

[24]: <keras.src.callbacks.history.History at 0x787ecdee7490>

[25]:
```python
from keras import models
from keras import layers
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


input_dim = X_scaled.shape[1]
input_layer = layers.Input(shape=(input_dim,))
encoded = layers.Dense(3, activation='linear')(input_layer)
decoded = layers.Dense(input_dim, activation='linear')(encoded)

linear_autoencoder = models.Model(inputs=input_layer, outputs=decoded)
linear_autoencoder.compile(optimizer='adam', loss='mse')
linear_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)

encoder_layer_linear = models.Model(inputs=input_layer, outputs=encoded)
```

```
X_encoded_linear = encoder_layer_linear.predict(X_scaled)


input_layer_sigmoid = layers.Input(shape=(input_dim,))
encoded_sigmoid = layers.Dense(3, activation='sigmoid')(input_layer_sigmoid)
decoded_sigmoid = layers.Dense(input_dim, activation='sigmoid')(encoded_sigmoid)

sigmoid_autoencoder = models.Model(inputs=input_layer_sigmoid,␣
 ↪outputs=decoded_sigmoid)
sigmoid_autoencoder.compile(optimizer='adam', loss='mse')
sigmoid_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)

encoder_layer_sigmoid = models.Model(inputs=input_layer_sigmoid,␣
 ↪outputs=encoded_sigmoid)
X_encoded_sigmoid = encoder_layer_sigmoid.predict(X_scaled)

input_layer_relu = layers.Input(shape=(input_dim,))
encoded_relu = layers.Dense(3, activation='relu')(input_layer_relu)
decoded_relu = layers.Dense(input_dim, activation='linear')(encoded_relu)

relu_autoencoder = models.Model(inputs=input_layer_relu, outputs=decoded_relu)
relu_autoencoder.compile(optimizer='adam', loss='mse')
relu_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)

encoder_layer_relu = models.Model(inputs=input_layer_relu, outputs=encoded_relu)
X_encoded_relu = encoder_layer_relu.predict(X_scaled)
```

```
Epoch 1/50
10/10             1s 3ms/step - loss:
0.8825
Epoch 2/50
10/10             0s 2ms/step - loss:
0.7532
Epoch 3/50
10/10             0s 2ms/step - loss:
0.7949
Epoch 4/50
10/10             0s 2ms/step - loss:
0.7949
Epoch 5/50
10/10             0s 2ms/step - loss:
0.6385
Epoch 6/50
10/10             0s 2ms/step - loss:
0.6170
Epoch 7/50
10/10             0s 2ms/step - loss:
```

```
0.5868
Epoch 8/50
10/10             0s 2ms/step - loss:
0.5507
Epoch 9/50
10/10             0s 2ms/step - loss:
0.5294
Epoch 10/50
10/10             0s 2ms/step - loss:
0.5479
Epoch 11/50
10/10             0s 2ms/step - loss:
0.4529
Epoch 12/50
10/10             0s 2ms/step - loss:
0.4731
Epoch 13/50
10/10             0s 2ms/step - loss:
0.4791
Epoch 14/50
10/10             0s 2ms/step - loss:
0.4131
Epoch 15/50
10/10             0s 2ms/step - loss:
0.3644
Epoch 16/50
10/10             0s 2ms/step - loss:
0.4153
Epoch 17/50
10/10             0s 2ms/step - loss:
0.3436
Epoch 18/50
10/10             0s 2ms/step - loss:
0.3940
Epoch 19/50
10/10             0s 2ms/step - loss:
0.3101
Epoch 20/50
10/10             0s 2ms/step - loss:
0.3273
Epoch 21/50
10/10             0s 2ms/step - loss:
0.3208
Epoch 22/50
10/10             0s 2ms/step - loss:
0.2800
Epoch 23/50
10/10             0s 2ms/step - loss:
```

```
0.3073
Epoch 24/50
10/10             0s 3ms/step - loss:
0.2809
Epoch 25/50
10/10             0s 3ms/step - loss:
0.2297
Epoch 26/50
10/10             0s 2ms/step - loss:
0.2794
Epoch 27/50
10/10             0s 2ms/step - loss:
0.2773
Epoch 28/50
10/10             0s 2ms/step - loss:
0.2180
Epoch 29/50
10/10             0s 3ms/step - loss:
0.2429
Epoch 30/50
10/10             0s 2ms/step - loss:
0.2400
Epoch 31/50
10/10             0s 2ms/step - loss:
0.2093
Epoch 32/50
10/10             0s 2ms/step - loss:
0.1985
Epoch 33/50
10/10             0s 2ms/step - loss:
0.2246
Epoch 34/50
10/10             0s 2ms/step - loss:
0.1908
Epoch 35/50
10/10             0s 2ms/step - loss:
0.1893
Epoch 36/50
10/10             0s 2ms/step - loss:
0.1869
Epoch 37/50
10/10             0s 2ms/step - loss:
0.1981
Epoch 38/50
10/10             0s 2ms/step - loss:
0.1978
Epoch 39/50
10/10             0s 2ms/step - loss:
```

```
0.1920
Epoch 40/50
10/10            0s 2ms/step - loss:
0.1816
Epoch 41/50
10/10            0s 3ms/step - loss:
0.1566
Epoch 42/50
10/10            0s 2ms/step - loss:
0.1859
Epoch 43/50
10/10            0s 2ms/step - loss:
0.1720
Epoch 44/50
10/10            0s 2ms/step - loss:
0.1608
Epoch 45/50
10/10            0s 2ms/step - loss:
0.1481
Epoch 46/50
10/10            0s 2ms/step - loss:
0.1448
Epoch 47/50
10/10            0s 2ms/step - loss:
0.1260
Epoch 48/50
10/10            0s 2ms/step - loss:
0.1383
Epoch 49/50
10/10            0s 2ms/step - loss:
0.1357
Epoch 50/50
10/10            0s 2ms/step - loss:
0.1174
5/5            0s 6ms/step
Epoch 1/50
10/10            1s 2ms/step - loss:
1.2957
Epoch 2/50
10/10            0s 2ms/step - loss:
1.1582
Epoch 3/50
10/10            0s 2ms/step - loss:
1.2765
Epoch 4/50
10/10            0s 3ms/step - loss:
1.2421
Epoch 5/50
```

```
10/10             0s 2ms/step - loss:
1.1575
Epoch 6/50
10/10             0s 2ms/step - loss:
1.2048
Epoch 7/50
10/10             0s 2ms/step - loss:
1.2590
Epoch 8/50
10/10             0s 2ms/step - loss:
1.1627
Epoch 9/50
10/10             0s 2ms/step - loss:
1.1686
Epoch 10/50
10/10             0s 2ms/step - loss:
1.2294
Epoch 11/50
10/10             0s 2ms/step - loss:
1.2406
Epoch 12/50
10/10             0s 2ms/step - loss:
1.1399
Epoch 13/50
10/10             0s 2ms/step - loss:
1.1635
Epoch 14/50
10/10             0s 2ms/step - loss:
1.1685
Epoch 15/50
10/10             0s 2ms/step - loss:
1.1014
Epoch 16/50
10/10             0s 2ms/step - loss:
1.0911
Epoch 17/50
10/10             0s 2ms/step - loss:
1.1609
Epoch 18/50
10/10             0s 2ms/step - loss:
1.1093
Epoch 19/50
10/10             0s 2ms/step - loss:
1.1625
Epoch 20/50
10/10             0s 2ms/step - loss:
1.1179
Epoch 21/50
```

```
10/10             0s 2ms/step - loss:
1.1833
Epoch 22/50
10/10             0s 3ms/step - loss:
1.1511
Epoch 23/50
10/10             0s 2ms/step - loss:
1.0332
Epoch 24/50
10/10             0s 2ms/step - loss:
1.0837
Epoch 25/50
10/10             0s 2ms/step - loss:
1.2054
Epoch 26/50
10/10             0s 2ms/step - loss:
1.1081
Epoch 27/50
10/10             0s 2ms/step - loss:
1.1029
Epoch 28/50
10/10             0s 2ms/step - loss:
1.1255
Epoch 29/50
10/10             0s 2ms/step - loss:
1.1455
Epoch 30/50
10/10             0s 2ms/step - loss:
1.0649
Epoch 31/50
10/10             0s 2ms/step - loss:
1.1287
Epoch 32/50
10/10             0s 2ms/step - loss:
1.0376
Epoch 33/50
10/10             0s 2ms/step - loss:
1.0842
Epoch 34/50
10/10             0s 2ms/step - loss:
1.0224
Epoch 35/50
10/10             0s 2ms/step - loss:
1.0524
Epoch 36/50
10/10             0s 2ms/step - loss:
1.0651
Epoch 37/50
```

```
10/10             0s 2ms/step - loss:
1.0845
Epoch 38/50
10/10             0s 2ms/step - loss:
1.0749
Epoch 39/50
10/10             0s 2ms/step - loss:
1.0166
Epoch 40/50
10/10             0s 3ms/step - loss:
1.0485
Epoch 41/50
10/10             0s 3ms/step - loss:
1.0537
Epoch 42/50
10/10             0s 3ms/step - loss:
1.0731
Epoch 43/50
10/10             0s 2ms/step - loss:
1.0189
Epoch 44/50
10/10             0s 3ms/step - loss:
1.0640
Epoch 45/50
10/10             0s 3ms/step - loss:
0.9899
Epoch 46/50
10/10             0s 2ms/step - loss:
1.0546
Epoch 47/50
10/10             0s 2ms/step - loss:
1.0259
Epoch 48/50
10/10             0s 2ms/step - loss:
1.0287
Epoch 49/50
10/10             0s 3ms/step - loss:
0.9786
Epoch 50/50
10/10             0s 4ms/step - loss:
0.9310
5/5             0s 9ms/step
Epoch 1/50
10/10             1s 3ms/step - loss:
1.1037
Epoch 2/50
10/10             0s 3ms/step - loss:
1.1091
```

```
Epoch 3/50
10/10             0s 3ms/step - loss:
1.0180
Epoch 4/50
10/10             0s 3ms/step - loss:
0.9229
Epoch 5/50
10/10             0s 3ms/step - loss:
0.9816
Epoch 6/50
10/10             0s 2ms/step - loss:
1.0011
Epoch 7/50
10/10             0s 2ms/step - loss:
0.9376
Epoch 8/50
10/10             0s 2ms/step - loss:
0.9339
Epoch 9/50
10/10             0s 2ms/step - loss:
0.8256
Epoch 10/50
10/10             0s 2ms/step - loss:
0.8554
Epoch 11/50
10/10             0s 2ms/step - loss:
0.8201
Epoch 12/50
10/10             0s 2ms/step - loss:
0.8510
Epoch 13/50
10/10             0s 2ms/step - loss:
0.8417
Epoch 14/50
10/10             0s 2ms/step - loss:
0.7874
Epoch 15/50
10/10             0s 2ms/step - loss:
0.7293
Epoch 16/50
10/10             0s 2ms/step - loss:
0.7833
Epoch 17/50
10/10             0s 2ms/step - loss:
0.7373
Epoch 18/50
10/10             0s 2ms/step - loss:
0.7357
```

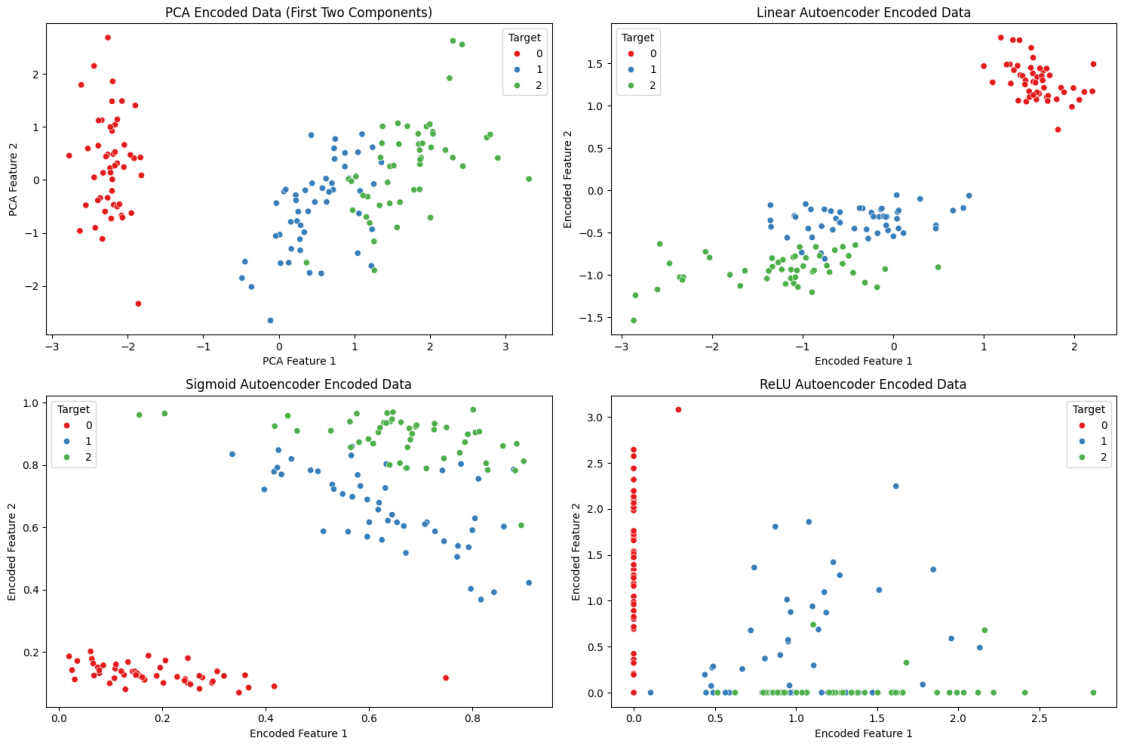
```
Epoch 19/50
10/10             0s 2ms/step - loss:
0.6885
Epoch 20/50
10/10             0s 3ms/step - loss:
0.6775
Epoch 21/50
10/10             0s 2ms/step - loss:
0.6395
Epoch 22/50
10/10             0s 2ms/step - loss:
0.6582
Epoch 23/50
10/10             0s 2ms/step - loss:
0.6728
Epoch 24/50
10/10             0s 2ms/step - loss:
0.5835
Epoch 25/50
10/10             0s 2ms/step - loss:
0.5819
Epoch 26/50
10/10             0s 2ms/step - loss:
0.5971
Epoch 27/50
10/10             0s 2ms/step - loss:
0.5796
Epoch 28/50
10/10             0s 2ms/step - loss:
0.4946
Epoch 29/50
10/10             0s 2ms/step - loss:
0.5212
Epoch 30/50
10/10             0s 2ms/step - loss:
0.4571
Epoch 31/50
10/10             0s 2ms/step - loss:
0.4249
Epoch 32/50
10/10             0s 2ms/step - loss:
0.4468
Epoch 33/50
10/10             0s 2ms/step - loss:
0.4785
Epoch 34/50
10/10             0s 2ms/step - loss:
0.5011
```

```
Epoch 35/50
10/10             0s 2ms/step - loss:
0.4946
Epoch 36/50
10/10             0s 2ms/step - loss:
0.3920
Epoch 37/50
10/10             0s 4ms/step - loss:
0.3943
Epoch 38/50
10/10             0s 2ms/step - loss:
0.4277
Epoch 39/50
10/10             0s 2ms/step - loss:
0.4415
Epoch 40/50
10/10             0s 2ms/step - loss:
0.4317
Epoch 41/50
10/10             0s 2ms/step - loss:
0.3893
Epoch 42/50
10/10             0s 2ms/step - loss:
0.3525
Epoch 43/50
10/10             0s 2ms/step - loss:
0.3236
Epoch 44/50
10/10             0s 2ms/step - loss:
0.3720
Epoch 45/50
10/10             0s 2ms/step - loss:
0.3494
Epoch 46/50
10/10             0s 2ms/step - loss:
0.3462
Epoch 47/50
10/10             0s 2ms/step - loss:
0.3661
Epoch 48/50
10/10             0s 2ms/step - loss:
0.3382
Epoch 49/50
10/10             0s 2ms/step - loss:
0.3552
Epoch 50/50
10/10             0s 2ms/step - loss:
0.2879
```

```
5/5                 0s 10ms/step
```

8 PCA Implementation

[26]:
```
# Visualization
plt.figure(figsize=(15, 10))

# PCA
plt.subplot(2, 2, 1)
sns.scatterplot(data=pd.DataFrame(X_encoded_pca, columns=['PCA Feature 1', 'PCA␣
↪Feature 2', 'PCA Feature 3']).assign(Target=y),
                x='PCA Feature 1', y='PCA Feature 2', hue='Target',␣
↪palette='Set1')
plt.title('PCA Encoded Data (First Two Components)')

# Linear Autoencoder
plt.subplot(2, 2, 2)
sns.scatterplot(data=pd.DataFrame(X_encoded_linear, columns=['Encoded Feature␣
↪1', 'Encoded Feature 2', 'Encoded Feature 3']).assign(Target=y),
                x='Encoded Feature 1', y='Encoded Feature 2', hue='Target',␣
↪palette='Set1')
plt.title('Linear Autoencoder Encoded Data')

# Sigmoid Autoencoder
plt.subplot(2, 2, 3)
sns.scatterplot(data=pd.DataFrame(X_encoded_sigmoid, columns=['Encoded Feature␣
↪1', 'Encoded Feature 2', 'Encoded Feature 3']).assign(Target=y),
                x='Encoded Feature 1', y='Encoded Feature 2', hue='Target',␣
↪palette='Set1')
plt.title('Sigmoid Autoencoder Encoded Data')

# ReLU Autoencoder
plt.subplot(2, 2, 4)
sns.scatterplot(data=pd.DataFrame(X_encoded_relu, columns=['Encoded Feature 1',␣
↪'Encoded Feature 2', 'Encoded Feature 3']).assign(Target=y),
                x='Encoded Feature 1', y='Encoded Feature 2', hue='Target',␣
↪palette='Set1')
plt.title('ReLU Autoencoder Encoded Data')

plt.tight_layout()
plt.show()
```



[27]:
```
import matplotlib.pyplot as plt

losses = {
    "PCA": [],
    "Linear Autoencoder": [],
    "Sigmoid Autoencoder": [],
    "ReLU Autoencoder": []
}

linear_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)
losses["Linear Autoencoder"] = linear_autoencoder.history.history['loss']

sigmoid_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)
losses["Sigmoid Autoencoder"] = sigmoid_autoencoder.history.history['loss']

relu_autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16)
losses["ReLU Autoencoder"] = relu_autoencoder.history.history['loss']

losses["PCA"] = [0] * len(losses["Linear Autoencoder"])

plt.figure(figsize=(10, 6))
for encoder, loss in losses.items():
    plt.plot(loss, label=encoder)
```

```
plt.title('Loss for Different Encoders')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
```

```
Epoch 1/50
10/10             0s 4ms/step - loss:
0.1284
Epoch 2/50
10/10             0s 3ms/step - loss:
0.1233
Epoch 3/50
10/10             0s 3ms/step - loss:
0.1241
Epoch 4/50
10/10             0s 3ms/step - loss:
0.1153
Epoch 5/50
10/10             0s 4ms/step - loss:
0.1191
Epoch 6/50
10/10             0s 3ms/step - loss:
0.1112
Epoch 7/50
10/10             0s 3ms/step - loss:
0.1150
Epoch 8/50
10/10             0s 3ms/step - loss:
0.1050
Epoch 9/50
10/10             0s 3ms/step - loss:
0.1020
Epoch 10/50
10/10             0s 3ms/step - loss:
0.1014
Epoch 11/50
10/10             0s 5ms/step - loss:
0.1072
Epoch 12/50
10/10             0s 5ms/step - loss:
0.0876
Epoch 13/50
10/10             0s 7ms/step - loss:
0.0906
Epoch 14/50
```

```
10/10             0s 4ms/step - loss:
0.1007
Epoch 15/50
10/10             0s 4ms/step - loss:
0.0971
Epoch 16/50
10/10             0s 3ms/step - loss:
0.0794
Epoch 17/50
10/10             0s 3ms/step - loss:
0.0857
Epoch 18/50
10/10             0s 3ms/step - loss:
0.0730
Epoch 19/50
10/10             0s 3ms/step - loss:
0.0855
Epoch 20/50
10/10             0s 3ms/step - loss:
0.0841
Epoch 21/50
10/10             0s 3ms/step - loss:
0.0730
Epoch 22/50
10/10             0s 3ms/step - loss:
0.0621
Epoch 23/50
10/10             0s 3ms/step - loss:
0.0696
Epoch 24/50
10/10             0s 3ms/step - loss:
0.0725
Epoch 25/50
10/10             0s 3ms/step - loss:
0.0736
Epoch 26/50
10/10             0s 3ms/step - loss:
0.0659
Epoch 27/50
10/10             0s 4ms/step - loss:
0.0748
Epoch 28/50
10/10             0s 2ms/step - loss:
0.0692
Epoch 29/50
10/10             0s 2ms/step - loss:
0.0655
Epoch 30/50
```

```
10/10             0s 2ms/step - loss:
0.0647
Epoch 31/50
10/10             0s 2ms/step - loss:
0.0599
Epoch 32/50
10/10             0s 2ms/step - loss:
0.0650
Epoch 33/50
10/10             0s 2ms/step - loss:
0.0554
Epoch 34/50
10/10             0s 3ms/step - loss:
0.0630
Epoch 35/50
10/10             0s 2ms/step - loss:
0.0632
Epoch 36/50
10/10             0s 2ms/step - loss:
0.0601
Epoch 37/50
10/10             0s 2ms/step - loss:
0.0635
Epoch 38/50
10/10             0s 2ms/step - loss:
0.0644
Epoch 39/50
10/10             0s 2ms/step - loss:
0.0592
Epoch 40/50
10/10             0s 2ms/step - loss:
0.0553
Epoch 41/50
10/10             0s 2ms/step - loss:
0.0605
Epoch 42/50
10/10             0s 3ms/step - loss:
0.0538
Epoch 43/50
10/10             0s 2ms/step - loss:
0.0508
Epoch 44/50
10/10             0s 2ms/step - loss:
0.0594
Epoch 45/50
10/10             0s 2ms/step - loss:
0.0546
Epoch 46/50
```

```
10/10             0s 2ms/step - loss:
0.0575
Epoch 47/50
10/10             0s 2ms/step - loss:
0.0574
Epoch 48/50
10/10             0s 2ms/step - loss:
0.0491
Epoch 49/50
10/10             0s 2ms/step - loss:
0.0484
Epoch 50/50
10/10             0s 2ms/step - loss:
0.0559
Epoch 1/50
10/10             0s 2ms/step - loss:
1.0063
Epoch 2/50
10/10             0s 2ms/step - loss:
0.9512
Epoch 3/50
10/10             0s 2ms/step - loss:
1.0700
Epoch 4/50
10/10             0s 2ms/step - loss:
0.9992
Epoch 5/50
10/10             0s 3ms/step - loss:
0.9045
Epoch 6/50
10/10             0s 2ms/step - loss:
1.0700
Epoch 7/50
10/10             0s 2ms/step - loss:
0.9887
Epoch 8/50
10/10             0s 2ms/step - loss:
0.9059
Epoch 9/50
10/10             0s 2ms/step - loss:
0.9681
Epoch 10/50
10/10             0s 2ms/step - loss:
1.0340
Epoch 11/50
10/10             0s 2ms/step - loss:
0.9953
Epoch 12/50
```

```
10/10             0s 3ms/step - loss:
0.8987
Epoch 13/50
10/10             0s 2ms/step - loss:
0.9718
Epoch 14/50
10/10             0s 2ms/step - loss:
1.0029
Epoch 15/50
10/10             0s 2ms/step - loss:
0.9040
Epoch 16/50
10/10             0s 2ms/step - loss:
0.9139
Epoch 17/50
10/10             0s 2ms/step - loss:
0.8853
Epoch 18/50
10/10             0s 2ms/step - loss:
0.8706
Epoch 19/50
10/10             0s 2ms/step - loss:
0.9792
Epoch 20/50
10/10             0s 2ms/step - loss:
0.9511
Epoch 21/50
10/10             0s 2ms/step - loss:
0.8829
Epoch 22/50
10/10             0s 2ms/step - loss:
0.9552
Epoch 23/50
10/10             0s 2ms/step - loss:
0.8769
Epoch 24/50
10/10             0s 2ms/step - loss:
0.9523
Epoch 25/50
10/10             0s 2ms/step - loss:
0.8935
Epoch 26/50
10/10             0s 2ms/step - loss:
0.9496
Epoch 27/50
10/10             0s 3ms/step - loss:
0.8485
Epoch 28/50
```

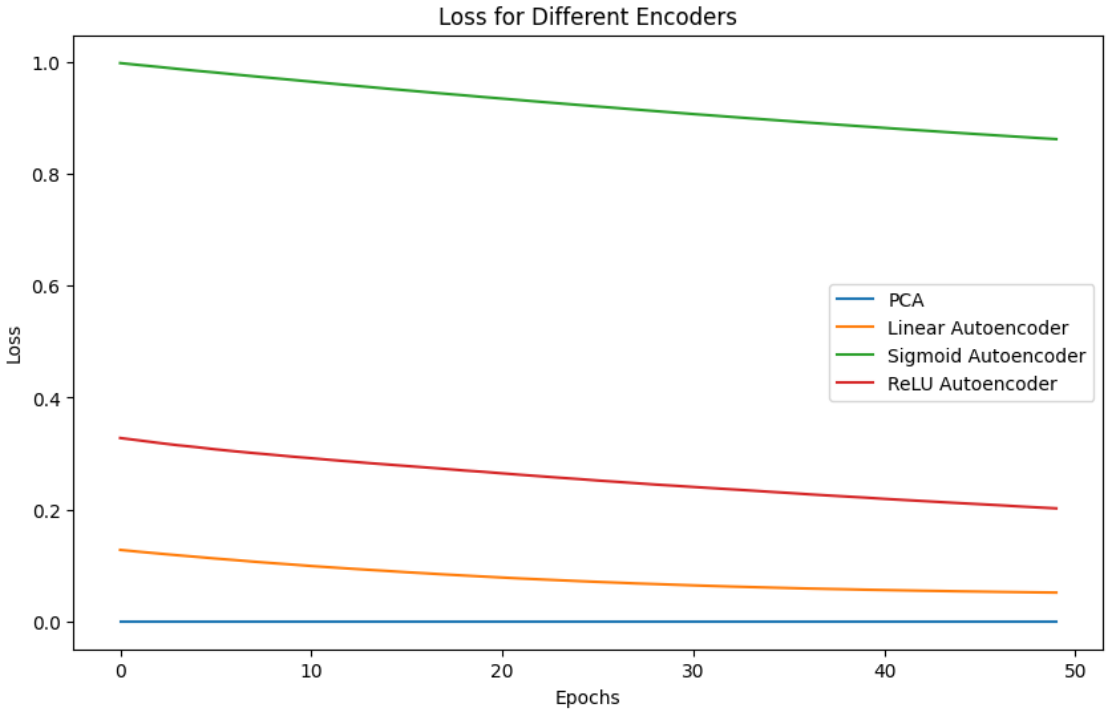
```
10/10             0s 2ms/step - loss:
1.0229
Epoch 29/50
10/10             0s 2ms/step - loss:
0.8583
Epoch 30/50
10/10             0s 2ms/step - loss:
0.8616
Epoch 31/50
10/10             0s 2ms/step - loss:
0.8343
Epoch 32/50
10/10             0s 2ms/step - loss:
0.9231
Epoch 33/50
10/10             0s 3ms/step - loss:
0.9357
Epoch 34/50
10/10             0s 2ms/step - loss:
0.8755
Epoch 35/50
10/10             0s 3ms/step - loss:
0.9271
Epoch 36/50
10/10             0s 2ms/step - loss:
0.9404
Epoch 37/50
10/10             0s 2ms/step - loss:
0.8352
Epoch 38/50
10/10             0s 2ms/step - loss:
0.8679
Epoch 39/50
10/10             0s 2ms/step - loss:
0.9442
Epoch 40/50
10/10             0s 2ms/step - loss:
0.8982
Epoch 41/50
10/10             0s 2ms/step - loss:
0.9067
Epoch 42/50
10/10             0s 2ms/step - loss:
0.8566
Epoch 43/50
10/10             0s 2ms/step - loss:
0.8628
Epoch 44/50
```

```
10/10             0s 2ms/step - loss:
0.8752
Epoch 45/50
10/10             0s 2ms/step - loss:
0.8722
Epoch 46/50
10/10             0s 2ms/step - loss:
0.9514
Epoch 47/50
10/10             0s 2ms/step - loss:
0.8571
Epoch 48/50
10/10             0s 2ms/step - loss:
0.8732
Epoch 49/50
10/10             0s 3ms/step - loss:
0.8262
Epoch 50/50
10/10             0s 2ms/step - loss:
0.9147
Epoch 1/50
10/10             0s 2ms/step - loss:
0.2872
Epoch 2/50
10/10             0s 2ms/step - loss:
0.3752
Epoch 3/50
10/10             0s 2ms/step - loss:
0.3089
Epoch 4/50
10/10             0s 2ms/step - loss:
0.3229
Epoch 5/50
10/10             0s 2ms/step - loss:
0.3621
Epoch 6/50
10/10             0s 2ms/step - loss:
0.3377
Epoch 7/50
10/10             0s 2ms/step - loss:
0.3156
Epoch 8/50
10/10             0s 2ms/step - loss:
0.2801
Epoch 9/50
10/10             0s 2ms/step - loss:
0.2798
Epoch 10/50
```

```
10/10             0s 2ms/step - loss:
0.3233
Epoch 11/50
10/10             0s 2ms/step - loss:
0.2780
Epoch 12/50
10/10             0s 3ms/step - loss:
0.2957
Epoch 13/50
10/10             0s 2ms/step - loss:
0.3312
Epoch 14/50
10/10             0s 2ms/step - loss:
0.2448
Epoch 15/50
10/10             0s 2ms/step - loss:
0.2689
Epoch 16/50
10/10             0s 2ms/step - loss:
0.2795
Epoch 17/50
10/10             0s 2ms/step - loss:
0.2877
Epoch 18/50
10/10             0s 2ms/step - loss:
0.2996
Epoch 19/50
10/10             0s 2ms/step - loss:
0.2099
Epoch 20/50
10/10             0s 2ms/step - loss:
0.3453
Epoch 21/50
10/10             0s 2ms/step - loss:
0.2880
Epoch 22/50
10/10             0s 2ms/step - loss:
0.2892
Epoch 23/50
10/10             0s 2ms/step - loss:
0.2610
Epoch 24/50
10/10             0s 2ms/step - loss:
0.2322
Epoch 25/50
10/10             0s 3ms/step - loss:
0.2938
Epoch 26/50
```

```
10/10             0s 3ms/step - loss:
0.2402
Epoch 27/50
10/10             0s 3ms/step - loss:
0.2647
Epoch 28/50
10/10             0s 2ms/step - loss:
0.2801
Epoch 29/50
10/10             0s 2ms/step - loss:
0.2426
Epoch 30/50
10/10             0s 2ms/step - loss:
0.2219
Epoch 31/50
10/10             0s 2ms/step - loss:
0.2423
Epoch 32/50
10/10             0s 2ms/step - loss:
0.2170
Epoch 33/50
10/10             0s 2ms/step - loss:
0.2249
Epoch 34/50
10/10             0s 2ms/step - loss:
0.2473
Epoch 35/50
10/10             0s 2ms/step - loss:
0.2191
Epoch 36/50
10/10             0s 3ms/step - loss:
0.1978
Epoch 37/50
10/10             0s 2ms/step - loss:
0.2021
Epoch 38/50
10/10             0s 3ms/step - loss:
0.2068
Epoch 39/50
10/10             0s 2ms/step - loss:
0.2308
Epoch 40/50
10/10             0s 2ms/step - loss:
0.2465
Epoch 41/50
10/10             0s 2ms/step - loss:
0.2697
Epoch 42/50
```

```
10/10             0s 2ms/step - loss:
0.2011
Epoch 43/50
10/10             0s 2ms/step - loss:
0.2227
Epoch 44/50
10/10             0s 2ms/step - loss:
0.2213
Epoch 45/50
10/10             0s 2ms/step - loss:
0.1785
Epoch 46/50
10/10             0s 2ms/step - loss:
0.2224
Epoch 47/50
10/10             0s 2ms/step - loss:
0.2523
Epoch 48/50
10/10             0s 2ms/step - loss:
0.2210
Epoch 49/50
10/10             0s 3ms/step - loss:
0.1844
Epoch 50/50
10/10             0s 2ms/step - loss:
0.2033
```



[ ]: