Task 1

Implement the NOT Boolean logic gate using perceptron Neural Network. Inputs = x1, x2 and bias, weights should be fed into the perceptron with single Output = y. Display final weights and bias of each perceptron

In [112]:
```
import numpy as np
import tensorflow as tf
```

In [113]:
```
X,y = np.array([
    [0, 1],
    [1, 0]
])
w1 = 1
w2 = -2
b = 1
X,y
```

Out[113]:
```
(array([0, 1]), array([1, 0]))
```

In [114]:
```
def step(z):
    return 1 if z>=1 else 0
```

In [115]:
```
for i in range(X.shape[0]):
    z = w1*X[i] + w2*X[i] + b
    print(f"Input: {X[i]} → Output:", step(z))
```

```
Input: 0 → Output: 1
Input: 1 → Output: 0
```

Using the updating weights and bias approach:

In [116]:
```
w = np.random.rand(1)
b = np.random.rand(1)

epochs = 100
learning_rate = 0.1

for epoch in range(epochs):
    for i in range(X.shape[0]):
        z = np.dot(w, X[i]) + b
        y_pred = step(z)
        error = y[i] - y_pred
        w += learning_rate * error * X[i]
        b += learning_rate * error

print("Trained weights:", w)
print("Trained bias:", b)
```

```
Trained weights: [-0.10659023]
Trained bias: [1.04670623]
```

In [117]:
```
print("\nPredictions:")
for i in range(len(X)):
    z = np.dot(w, X[i]) + b
    print(f"Input: {X[i]} → Output:", step(z))
```

```
Predictions:
Input: 0 → Output: 1
Input: 1 → Output: 0
```

Task 2

1. Use the Iris Dataset
2. Create an Auto Encoder and fit it with our data using 3 neurons in the dense layer
3. Display new reduced dimension values
4. Plot loss for different encoders

In [118]:
```
import pandas as pd
```

In [119]:
```
from sklearn.datasets import load_iris
X = load_iris().data
```

In [120]:
```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

In [121]:
```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

In [122]:
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Input
```

In [123]:
```
autoencoder = Sequential([
    Input(shape = (X.shape[1], )),
    Dense(3, activation='relu'),
    Dense(X.shape[1], activation='sigmoid')
])
```

In [124]:
```
autoencoder.compile(optimizer='adam', loss='mse')
```

In [125]:
```
autoencoder.fit(X_scaled, X_scaled, epochs=50, batch_size=16, shuffle=True,
validation_split=0.2)
```

```
Epoch 1/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 13ms/step - loss: 1.5463 -
val_loss: 0.6187
Epoch 2/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.5356 -
val_loss: 0.6109
Epoch 3/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.5247 -
val_loss: 0.6034
Epoch 4/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.5138 -
val_loss: 0.5959
Epoch 5/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.5032 -
val_loss: 0.5887
Epoch 6/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4928 -
val_loss: 0.5814
Epoch 7/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4824 -
val_loss: 0.5745
Epoch 8/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4719 -
val_loss: 0.5671
Epoch 9/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4618 -
val_loss: 0.5606
Epoch 10/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4518 -
val_loss: 0.5546
Epoch 11/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4422 -
val_loss: 0.5479
Epoch 12/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4325 -
val_loss: 0.5416
Epoch 13/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4233 -
val_loss: 0.5358
Epoch 14/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.4141 -
val_loss: 0.5291
Epoch 15/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.4054 -
val_loss: 0.5232
Epoch 16/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3969 -
val_loss: 0.5171
Epoch 17/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3887 -
val_loss: 0.5118
Epoch 18/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3808 -
val_loss: 0.5073
Epoch 19/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3729 -
val_loss: 0.5028
Epoch 20/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3656 -
val_loss: 0.4982
Epoch 21/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3581 -
val_loss: 0.4940
Epoch 22/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3511 -
val_loss: 0.4896
```

```
Epoch 23/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3446 -
val_loss: 0.4861
Epoch 24/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.3380 -
val_loss: 0.4817
Epoch 25/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3316 -
val_loss: 0.4780
Epoch 26/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3255 -
val_loss: 0.4748
Epoch 27/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3195 -
val_loss: 0.4718
Epoch 28/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3138 -
val_loss: 0.4693
Epoch 29/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3080 -
val_loss: 0.4662
Epoch 30/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.3026 -
val_loss: 0.4634
Epoch 31/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2973 -
val_loss: 0.4608
Epoch 32/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2922 -
val_loss: 0.4581
Epoch 33/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2870 -
val_loss: 0.4555
Epoch 34/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2820 -
val_loss: 0.4532
Epoch 35/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2771 -
val_loss: 0.4510
Epoch 36/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2721 -
val_loss: 0.4489
Epoch 37/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.2675 -
val_loss: 0.4470
Epoch 38/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.2627 -
val_loss: 0.4455
Epoch 39/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.2581 -
val_loss: 0.4433
Epoch 40/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2533 -
val_loss: 0.4414
Epoch 41/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.2489 -
val_loss: 0.4401
Epoch 42/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 7ms/step - loss: 1.2443 -
val_loss: 0.4388
Epoch 43/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2397 -
val_loss: 0.4374
Epoch 44/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2351 -
val_loss: 0.4358
Epoch 45/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1s[0m 6ms/step - loss: 1.2305 -
val_loss: 0.4344
Epoch 46/50
```

```
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 6ms/step - loss: 1.2261 -
val_loss: 0.4332
Epoch 47/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 6ms/step - loss: 1.2215 -
val_loss: 0.4323
Epoch 48/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 6ms/step - loss: 1.2170 -
val_loss: 0.4314
Epoch 49/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 6ms/step - loss: 1.2125 -
val_loss: 0.4302
Epoch 50/50
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 6ms/step - loss: 1.2079 -
val_loss: 0.4295
```

Out[125]:
```
<keras.src.callbacks.history.History at 0x2667f0428a0>
```

In [126]:
```
encoder  =Sequential([autoencoder.layers[0]])
encoded_data = encoder.predict(X_scaled)
```

```
[1m5/5[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m1m0s[0m 5ms/step
```

The reduced dimention values are as follows:

In [127]:
```
encoded_data
```

Out[127]:
```
array([[0.9406513 , 0.        , 0.        ],
       [0.32772845, 0.        , 0.        ],
       [0.5817579 , 0.        , 0.        ],
       [0.40021795, 0.        , 0.        ],
       [1.0539247 , 0.        , 0.        ],
       [1.3686391 , 0.        , 0.        ],
       [0.7928662 , 0.        , 0.        ],
       [0.78659374, 0.        , 0.        ],
       [0.1745504 , 0.        , 0.        ],
       [0.41477722, 0.        , 0.        ],
       [1.1729696 , 0.        , 0.        ],
       [0.74580973, 0.        , 0.        ],
       [0.31568468, 0.        , 0.        ],
       [0.36487836, 0.        , 0.        ],
       [1.6417928 , 0.        , 0.        ],
       [2.0431776 , 0.        , 0.        ],
       [1.478569  , 0.        , 0.        ],
       [0.9460442 , 0.        , 0.        ],
       [1.2632741 , 0.        , 0.        ],
       [1.2783345 , 0.        , 0.        ],
       [0.7582318 , 0.        , 0.        ],
       [1.1638032 , 0.        , 0.        ],
       [1.1372516 , 0.        , 0.        ],
       [0.63453424, 0.        , 0.        ],
       [0.6633623 , 0.        , 0.        ],
       [0.27941424, 0.        , 0.        ],
       [0.7698972 , 0.        , 0.        ],
       [0.9198195 , 0.        , 0.        ],
       [0.82737774, 0.        , 0.        ],
       [0.4993105 , 0.        , 0.        ],
       [0.38603693, 0.        , 0.        ],
       [0.8239827 , 0.        , 0.        ],
       [1.633972  , 0.        , 0.        ],
       [1.8067242 , 0.        , 0.        ],
       [0.4201702 , 0.        , 0.        ],
       [0.6291927 , 0.        , 0.        ],
       [0.9947368 , 0.        , 0.        ],
       [1.041881  , 0.        , 0.        ],
       [0.3219571 , 0.        , 0.        ],
       [0.79324454, 0.        , 0.        ],
       [0.9668759 , 0.        , 0.        ],
       [0.        , 0.        , 0.48330545],
       [0.5618056 , 0.        , 0.        ],
       [0.90060747, 0.        , 0.        ],
       [1.1737976 , 0.        , 0.        ],
       [0.3264706 , 0.        , 0.        ],
       [1.2454591 , 0.        , 0.        ],
       [0.5476247 , 0.        , 0.        ],
       [1.1663188 , 0.        , 0.        ],
       [0.694152  , 0.        , 0.        ],
       [0.        , 1.006562  , 0.        ],
       [0.        , 0.76434267, 0.        ],
       [0.        , 1.4012833 , 0.        ],
       [0.        , 2.393271  , 1.0389706 ],
       [0.        , 1.9140996 , 0.16736107],
       [0.        , 1.334302  , 0.2978707 ],
       [0.        , 0.6249491 , 0.        ],
       [0.        , 1.3115089 , 0.84407806],
       [0.        , 1.4989582 , 0.        ],
       [0.        , 1.2518963 , 0.64380115],
       [0.        , 2.49902   , 1.38809   ],
       [0.        , 0.9663284 , 0.06718512],
       [0.        , 2.5914092 , 0.7899711 ],
       [0.        , 1.4137746 , 0.10196503],
       [0.        , 0.6836834 , 0.12815385],
       [0.        , 1.029037  , 0.        ],
       [0.        , 0.94827116, 0.19365673],
```

```
[0.        , 1.2063298 , 0.16099681],
[0.        , 3.3473697 , 1.1130521 ],
[0.        , 1.6733181 , 0.571141  ],
[0.        , 0.94652474, 0.04184525],
[0.        , 1.3206239 , 0.12217896],
[0.        , 2.7409878 , 0.68247384],
[0.        , 1.4889748 , 0.10002814],
[0.        , 1.298149  , 0.        ],
[0.        , 1.252321  , 0.        ],
[0.        , 2.0249991 , 0.00461288],
[0.        , 1.8104054 , 0.        ],
[0.        , 1.3904216 , 0.19452463],
[0.        , 1.2030888 , 0.29495907],
[0.        , 1.8588481 , 0.7404176 ],
[0.        , 1.7251654 , 0.6617637 ],
[0.        , 1.3226787 , 0.29005313],
[0.        , 2.2469902 , 0.5909833 ],
[0.        , 0.8607246 , 0.26384538],
[0.10412884, 0.15106341, 0.        ],
[0.        , 1.2382282 , 0.        ],
[0.        , 2.894474  , 0.78646755],
[0.        , 0.6053971 , 0.02222313],
[0.        , 1.8591564 , 0.75648034],
[0.        , 1.6471874 , 0.5718957 ],
[0.        , 1.1089631 , 0.        ],
[0.        , 1.6274903 , 0.43836117],
[0.        , 1.6223395 , 0.9502289 ],
[0.        , 1.4443234 , 0.45302153],
[0.        , 0.59099597, 0.        ],
[0.        , 0.95398194, 0.13543685],
[0.        , 1.2106025 , 0.        ],
[0.        , 1.1146638 , 0.6830466 ],
[0.        , 1.183285  , 0.2696191 ],
[0.        , 1.9791113 , 0.3461148 ],
[0.        , 2.4472294 , 0.875945  ],
[0.        , 2.709001  , 0.1956686 ],
[0.        , 2.2248235 , 0.38170654],
[0.        , 2.5045357 , 0.4707626 ],
[0.        , 3.1921468 , 0.06963722],
[0.        , 2.169002  , 1.2887248 ],
[0.        , 2.926836  , 0.08020346],
[0.        , 3.5436544 , 0.8204355 ],
[0.03260367, 1.609653  , 0.        ],
[0.        , 1.5142841 , 0.        ],
[0.        , 2.7853775 , 0.6795049 ],
[0.        , 2.4266644 , 0.2727    ],
[0.        , 2.9957452 , 1.2580577 ],
[0.        , 2.6598153 , 1.0926551 ],
[0.        , 1.8338052 , 0.25964332],
[0.        , 2.0075583 , 0.16320984],
[0.12463228, 1.2331438 , 0.        ],
[0.        , 4.609269  , 0.7638942 ],
[0.        , 3.448594  , 1.2185552 ],
[0.        , 2.2036884 , 0.11242323],
[0.        , 2.1130457 , 0.8623538 ],
[0.        , 3.7118602 , 0.25250506],
[0.        , 2.494659  , 0.61475664],
[0.        , 1.6572274 , 0.        ],
[0.        , 1.9686278 , 0.        ],
[0.        , 2.1460738 , 0.5015429 ],
[0.        , 1.6059402 , 0.26120985],
[0.        , 2.82344   , 0.7026306 ],
[0.        , 2.2353768 , 0.        ],
[0.        , 3.2580867 , 0.24381964],
[0.20959525, 1.0155708 , 0.        ],
[0.        , 2.9193687 , 0.7742214 ],
[0.        , 2.015324  , 0.27286416],
[0.        , 2.5547347 , 0.58926654],
[0.        , 3.239006  , 0.1424105 ],
[0.        , 1.4651085 , 0.10502701],
```

```
[0.        , 1.6967279 , 0.057059  ],
[0.        , 1.5244128 , 0.28924125],
[0.        , 2.165626  , 0.08929764],
[0.        , 2.4413738 , 0.38838518],
[0.        , 2.2442203 , 0.21129103],
[0.        , 2.4472294 , 0.875945  ],
[0.        , 2.2354236 , 0.16164334],
[0.        , 2.0409417 , 0.18454884],
[0.        , 2.4614854 , 0.42978776],
[0.        , 3.1624565 , 0.97590077],
[0.        , 2.086153  , 0.28520328],
[0.        , 1.2498982 , 0.05440451],
[0.        , 1.593902  , 0.34552425]], dtype=float32)
```

In [128]:
```
preds = []
loss = []
for i in range(1,5):
    autoencoder = Sequential([
        Input(shape = (X.shape[1], )),
        Dense(i, activation='relu'),
        Dense(X.shape[1], activation='sigmoid')
    ])
    autoencoder.compile(optimizer='adam', loss='mse')
    history = autoencoder.fit(X_scaled, X_scaled, epochs=20, batch_size=16,
shuffle=True, validation_split=0.2)
    preds.append(autoencoder.predict(X_scaled))
    loss.append(history.history['loss'])
    print(f'Encoding Dimension: {i}, Loss: {loss[-1]}')
```

```
Epoch 1/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 13ms/step - loss: 1.3208 -
val_loss: 0.5423
Epoch 2/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3146 -
val_loss: 0.5431
Epoch 3/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3084 -
val_loss: 0.5439
Epoch 4/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3022 -
val_loss: 0.5448
Epoch 5/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2958 -
val_loss: 0.5456
Epoch 6/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 7ms/step - loss: 1.2899 -
val_loss: 0.5465
Epoch 7/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2835 -
val_loss: 0.5474
Epoch 8/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2773 -
val_loss: 0.5482
Epoch 9/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2712 -
val_loss: 0.5490
Epoch 10/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2652 -
val_loss: 0.5497
Epoch 11/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2593 -
val_loss: 0.5505
Epoch 12/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2534 -
val_loss: 0.5513
Epoch 13/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2475 -
val_loss: 0.5520
Epoch 14/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2418 -
val_loss: 0.5528
Epoch 15/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 7ms/step - loss: 1.2359 -
val_loss: 0.5536
Epoch 16/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 7ms/step - loss: 1.2303 -
val_loss: 0.5543
Epoch 17/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2245 -
val_loss: 0.5550
Epoch 18/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2188 -
val_loss: 0.5557
```

```
Epoch 19/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2132 -
val_loss: 0.5563
Epoch 20/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.2077 -
val_loss: 0.5570
[1m5/5[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step
Encoding Dimension: 1, Loss: [1.3208034038543701, 1.3146188259124756,
1.3084051609039307, 1.3022257089614868, 1.2958046197891235, 1.289927363395691,
1.2834537029266357, 1.2772845029830933, 1.271241307258606, 1.2652369737625122,
1.2592722177505493, 1.2534306049346924, 1.2475420236587524, 1.2417625188827515,
1.235927700996399, 1.230284333229065, 1.2244750261306763, 1.2187851667404175,
1.213154673576355, 1.2076836824417114]
Epoch 1/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 13ms/step - loss: 1.5125 -
val_loss: 0.5473
Epoch 2/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.5033 -
val_loss: 0.5483
Epoch 3/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4939 -
val_loss: 0.5493
Epoch 4/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4848 -
val_loss: 0.5503
Epoch 5/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4755 -
val_loss: 0.5514
Epoch 6/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4662 -
val_loss: 0.5524
Epoch 7/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4569 -
val_loss: 0.5534
Epoch 8/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4480 -
val_loss: 0.5544
Epoch 9/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4391 -
val_loss: 0.5553
Epoch 10/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4295 -
val_loss: 0.5563
Epoch 11/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4205 -
val_loss: 0.5574
Epoch 12/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4113 -
val_loss: 0.5585
Epoch 13/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4018 -
val_loss: 0.5594
Epoch 14/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3923 -
val_loss: 0.5605
Epoch 15/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3831 -
val_loss: 0.5614
Epoch 16/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3737 -
val_loss: 0.5624
Epoch 17/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3642 -
val_loss: 0.5633
Epoch 18/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3553 -
val_loss: 0.5643
Epoch 19/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3459 -
val_loss: 0.5653
```

```
Epoch 20/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3370 -
val_loss: 0.5663
[1m5/5[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step
Encoding Dimension: 2, Loss: [1.5125142335891724, 1.5032681226730347,
1.4939273595809937, 1.4848178625106812, 1.475497841835022, 1.4661580324172974,
1.4569032192230225, 1.447950005531311, 1.439149022102356, 1.4295470714569092,
1.4205071926116943, 1.4112592935562134, 1.4017635583877563, 1.3923107385635376,
1.3830643892288208, 1.3737413883209229, 1.3641504049301147, 1.3552883863449097,
1.3459017276763916, 1.3369556665420532]
Epoch 1/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 13ms/step - loss: 1.4679 -
val_loss: 0.5715
Epoch 2/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4599 -
val_loss: 0.5693
Epoch 3/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4525 -
val_loss: 0.5674
Epoch 4/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4456 -
val_loss: 0.5654
Epoch 5/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4390 -
val_loss: 0.5630
Epoch 6/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4324 -
val_loss: 0.5609
Epoch 7/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4260 -
val_loss: 0.5585
Epoch 8/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4199 -
val_loss: 0.5563
Epoch 9/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4139 -
val_loss: 0.5543
Epoch 10/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4081 -
val_loss: 0.5519
Epoch 11/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4024 -
val_loss: 0.5494
Epoch 12/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3969 -
val_loss: 0.5473
Epoch 13/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3916 -
val_loss: 0.5452
Epoch 14/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3867 -
val_loss: 0.5436
Epoch 15/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3819 -
val_loss: 0.5419
Epoch 16/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3771 -
val_loss: 0.5399
Epoch 17/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3724 -
val_loss: 0.5381
Epoch 18/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3677 -
val_loss: 0.5362
Epoch 19/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3632 -
val_loss: 0.5348
Epoch 20/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 8ms/step - loss: 1.3585 -
val_loss: 0.5328
```

```
[1m5/5[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 5ms/step
Encoding Dimension: 3, Loss: [1.4678815603256226, 1.4599121809005737,
1.4524868726730347, 1.445599913597107, 1.4390289783477783, 1.4323586225509644,
1.4260315895080566, 1.4198707342147827, 1.4139213562011719, 1.408059000968933,
1.4023598432540894, 1.3969483375549316, 1.3915820121765137, 1.38671875,
1.3818602561950684, 1.377134084701538, 1.3724324703216553, 1.3677042722702026,
1.3631787300109863, 1.3585273027420044]
Epoch 1/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 13ms/step - loss: 1.5128 -
val_loss: 0.7940
Epoch 2/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.5025 -
val_loss: 0.7851
Epoch 3/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4933 -
val_loss: 0.7760
Epoch 4/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4839 -
val_loss: 0.7669
Epoch 5/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4753 -
val_loss: 0.7590
Epoch 6/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4665 -
val_loss: 0.7500
Epoch 7/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4581 -
val_loss: 0.7414
Epoch 8/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 7ms/step - loss: 1.4502 -
val_loss: 0.7334
Epoch 9/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4423 -
val_loss: 0.7244
Epoch 10/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 7ms/step - loss: 1.4347 -
val_loss: 0.7158
Epoch 11/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4269 -
val_loss: 0.7079
Epoch 12/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4194 -
val_loss: 0.7001
Epoch 13/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4121 -
val_loss: 0.6923
Epoch 14/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.4047 -
val_loss: 0.6849
Epoch 15/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3974 -
val_loss: 0.6771
Epoch 16/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3902 -
val_loss: 0.6696
Epoch 17/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3832 -
val_loss: 0.6627
Epoch 18/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3763 -
val_loss: 0.6550
Epoch 19/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3694 -
val_loss: 0.6480
Epoch 20/20
[1m8/8[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 6ms/step - loss: 1.3632 -
val_loss: 0.6415
[1m5/5[0m [32m━━━━━━━━━━━━━━━━━━━━[0m[37m[0m [1m0s[0m 5ms/step
Encoding Dimension: 4, Loss: [1.512825846672058, 1.5024619102478027,
1.4933172464370728, 1.48392653465271, 1.4753153324127197, 1.4664556980133057,
```

```
1.4581149816513062, 1.450237512588501, 1.4423035383224487, 1.4347180128097534,
1.4269241094589233, 1.419448733329773, 1.4121322631835938, 1.404748797416687,
1.3973814249038696, 1.3902099132537842, 1.3831521272659302, 1.3762688636779785,
1.3693735599517822, 1.3632349967956543]
```

In [130]:
```
import matplotlib.pyplot as plt

for i in range(4):
    plt.plot(loss[i], label=f'Encoding Dim: {i+1}')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
```

Out[130]: <matplotlib.legend.Legend at 0x2667f1313d0>