```
Experiment:6-b
```

Implement a Simple Perceptron and a Multi-Layer Perceptron (MLP) using the MNIST dataset, and evaluate their classification performance using accuracy, precision, recall, and F1 score.

```
import numpy as np
import tensorflow as tf
from sklearn.linear_model import Perceptron
from sklearn.metrics import accuracy_score,precision_score,recall_score,f1_
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Flatten
from tensorflow.keras.utils import to_categorical
from sklearn.metrics import classification_report
```

```
np.random.seed(42)
tf.random.set_seed(42)
```

```
(x_train,y_train),(x_test,y_test)=tf.keras.datasets.mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.n
11490434/11490434 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
```

```
x_train.shape
```

```
(60000, 28, 28)
```

```
y_train.shape
```

```
(60000,)
```

```
x_train[0]
```

```
ndarray (28, 28) show data
```

```
x_train=x_train/255.0
x_test=x_test/255.0
```

```
x_train[0]
```

```
        0.        , 0.        , 0.15294118, 0.98039216, 0.89803922,
        0.99215686, 0.99215686, 0.99215686, 0.98039216, 0.71372549,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.09411765, 0.44705882, 0.86666667, 0.99215686, 0.99215686,
        0.99215686, 0.99215686, 0.78823529, 0.30588235, 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.09019608, 0.25882353,
        0.83529412, 0.99215686, 0.99215686, 0.99215686, 0.99215686,
        0.77647059, 0.31764706, 0.00784314, 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.07058824, 0.67058824, 0.85882353, 0.99215686,
        0.99215686, 0.99215686, 0.99215686, 0.76470588, 0.31372549,
        0.03529412, 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.21568627,
        0.6745098 , 0.88627451, 0.99215686, 0.99215686, 0.99215686,
        0.99215686, 0.95686275, 0.52156863, 0.04313725, 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.53333333,
        0.99215686, 0.99215686, 0.99215686, 0.83137255, 0.52941176,
        0.51764706, 0.0627451 , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        ]])
```

```
x_train_flat=x_train.reshape(-1,784)
x_test_flat=x_test.reshape(-1,784)
```

```
x_train_flat[0]
```

```
       0.97647059, 0.25098039, 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.18039216, 0.50980392,
       0.71764706, 0.99215686, 0.99215686, 0.81176471, 0.00784314,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.15294118,
       0.58039216, 0.89803922, 0.99215686, 0.99215686, 0.99215686,
       0.98039216, 0.71372549, 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.09411765, 0.44705882, 0.86666667, 0.99215686, 0.99215686,
       0.99215686, 0.99215686, 0.78823529, 0.30588235, 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.09019608, 0.25882353, 0.83529412, 0.99215686,
       0.99215686, 0.99215686, 0.99215686, 0.77647059, 0.31764706,
       0.00784314, 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.07058824, 0.67058824, 0.85882353,
       0.99215686, 0.99215686, 0.99215686, 0.99215686, 0.76470588,
       0.31372549, 0.03529412, 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.21568627, 0.6745098 ,
       0.88627451, 0.99215686, 0.99215686, 0.99215686, 0.99215686,
       0.95686275, 0.52156863, 0.04313725, 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.53333333, 0.99215686, 0.99215686, 0.99215686,
       0.83137255, 0.52941176, 0.51764706, 0.0627451 , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        ])
```

```
perceptron_model=Perceptron(max_iter=1000,tol=1e-3,random_state=42)
perceptron_model.fit(x_train_flat,y_train)
```

```
▾     Perceptron
Perceptron(random_state=42)
```

```
y_pred_perceptron=perceptron_model.predict(x_test_flat)
```

```
print("Accuracy:" ,accuracy_score(y_test,y_pred_perceptron))
print("Precision (macro):" ,precision_score(y_test,y_pred_perceptron,average='macro'))
print("Recall (macro):" ,recall_score(y_test,y_pred_perceptron,average='macro'))
print("F1 Score (macro):" ,f1_score(y_test,y_pred_perceptron,average='macro'))
```

```
Accuracy: 0.8633
Precision (macro): 0.8751002060577797
Recall (macro): 0.8631763828684068
F1 Score (macro): 0.8602286819132756
```

```
y_train_cat=to_categorical(y_train,10)
y_test_cat=to_categorical(y_test,10)
```

```
mlp_model=Sequential([
    Flatten(input_shape=(28,28)),
    Dense(128,activation='relu'),
    Dense(64,activation='relu'),
    Dense(10,activation='softmax')
])
```

```
/usr/local/lib/python3.12/dist-packages/keras/src/layers/reshaping/flatten.py:37: UserWar
  super().__init__(**kwargs)
```

```
mlp_model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
```

```
mlp_model.fit(x_train,y_train_cat,epochs=5,batch_size=128,verbose=1)
```

```
Epoch 1/5
469/469 ━━━━━━━━━━━━━━━━━━━━ 3s 4ms/step - accuracy: 0.8360 - loss: 0.5877
Epoch 2/5
469/469 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9579 - loss: 0.1468
Epoch 3/5
469/469 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9714 - loss: 0.0984
Epoch 4/5
469/469 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9792 - loss: 0.0725
Epoch 5/5
469/469 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9833 - loss: 0.0557
<keras.src.callbacks.history.History at 0x7e43f988d4c0>
```

```
y_pred_mlp=mlp_model.predict(x_test)
y_pred_mlp_labels=np.argmax(y_pred_mlp,axis=1)
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 0s 956us/step
```

```
print("Accuracy:" ,accuracy_score(y_test,y_pred_mlp_labels))
print("Precision (macro):" ,precision_score(y_test,y_pred_mlp_labels,average='macro'))
print("Recall (macro):" ,recall_score(y_test,y_pred_mlp_labels,average='macro'))
print("F1 Score (macro):" ,f1_score(y_test,y_pred_mlp_labels,average='macro'))
print(classification_report(y_test,y_pred_mlp_labels))
```

```
Accuracy: 0.9745
Precision (macro): 0.9742521011585508
Recall (macro): 0.9744194160763863
F1 Score (macro): 0.9742606060994415
              precision    recall  f1-score   support

           0       0.98      0.99      0.98       980
           1       0.99      0.99      0.99      1135
           2       0.98      0.98      0.98      1032
           3       0.97      0.98      0.97      1010
           4       0.98      0.98      0.98       982
           5       0.96      0.98      0.97       892
           6       0.97      0.98      0.98       958
           7       0.97      0.98      0.97      1028
           8       0.95      0.97      0.96       974
           9       0.99      0.94      0.96      1009

    accuracy                           0.97     10000
   macro avg       0.97      0.97      0.97     10000
weighted avg       0.97      0.97      0.97     10000
```