

## 1. 프로그램 4-4를 실행하여 결과를 정리하고, 코드를 설명하시오.

```

from sklearn.datasets import fetch_openml
from sklearn.neural_network import MLPClassifier
import matplotlib.pyplot as plt
import numpy as np

#mnist 데이터 가져옴
mnist=fetch_openml('mnist_784')
mnist.data=mnist.data/255.0 #[0,255]의 값을 가진 데이터를 255로 나눔으로써 [0,1]로
정규화한다.
x_train=mnist.data[:60000]; #학습데이터를 60000개로 설정한다.
x_test=mnist.data[60000:] #테스트 데이터는 10000개가 된다.
y_train=np.int16(mnist.target[:60000]); y_test=np.int16(mnist.target[60000:])
#np.int16을 적용해서 str형을 16비트 정수형으로 변환(혼동 행렬에서 계산하기 위해 필수로
수행해야하는 부분)

# MLP 분류기 모델을 학습
mlp=MLPClassifier(hidden_layer_sizes=(100),learning_rate_init=0.001,batch_size=512,max_iter=30
0,solver='adam',verbose=True) #은닉층 100, 배치크기는 512로 설정한다.
mlp.fit(x_train,y_train) #학습을 진행한다.

# 테스트 집합으로 예측
res=mlp.predict(x_test) #테스트 데이터로 예측한다.

# 혼동 행렬
conf=np.zeros((10,10),dtype=np.int16) #혼동 행렬 계산한다.
for i in range(len(res)):
    conf[res[i]][y_test[i]]+=1
print(conf)

# 정확률 계산
no_correct=0
for i in range(10):
    no_correct+=conf[i][i]
accuracy=no_correct/len(res) # 정확도를 계산한다.
print("테스트 집합에 대한 정확률은", accuracy*100, "%입니다.")

```

## 결과

```

Iteration 57, loss = 0.00250057
Iteration 58, loss = 0.00269600
Iteration 59, loss = 0.00246016
Iteration 60, loss = 0.00238001
Iteration 61, loss = 0.00232626
Iteration 62, loss = 0.00219560
Iteration 63, loss = 0.00195167
Iteration 64, loss = 0.00182147
Iteration 65, loss = 0.00173425
Iteration 66, loss = 0.00162174
Iteration 67, loss = 0.00153373
Iteration 68, loss = 0.00148125
Iteration 69, loss = 0.00141218
Iteration 70, loss = 0.00134812
Iteration 71, loss = 0.00126114
Iteration 72, loss = 0.00123354
Iteration 73, loss = 0.00114372
Iteration 74, loss = 0.00105909
Iteration 75, loss = 0.00105144
Iteration 76, loss = 0.00104671
Iteration 77, loss = 0.00109535
Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.
[[ 973   0   4   0   0   2   4   0   7   4]
 [   0 1123   1   0   0   0   1   5   1   2]
 [   1   3 1000   5   6   0   3   5   3   0]
 [   1   2   4  986   1   9   1   3   7   6]
 [   0   0   1   0  959   0   5   3   6   9]
 [   0   0   0   6   1  867   4   0   3   3]
 [   2   2   3   0   3   6  940   0   2   1]
 [   1   1   3   4   2   3   0 1006   4   3]
 [   1   4  14   5   2   5   0   3  937   2]
 [   1   0   2   4   8   0   0   3   4  979]]
테스트 집합에 대한 정확률은 97.7 %입니다.

In [2]:

```

## 결과 정리

- 77번 학습함.
- 최종 loss는 0.00109535
- 정확도 97.7%

2. btch\_size를 128로 하고, 은닉층 사이즈를 50으로한 수행결과를 비교하시오.

정확도 차이 97.7%에서 97.13%로 약간 감소하였으나, 큰 변화는 아니다.  
은닉층과 batch\_size가 줄었으나 횟수는 조금 더 많아 졌다.

따라서 batch\_size와 은닉층의 수는 정확도에 영향을 준다.

Console 1/A

Iteration 57, loss = 0.00230034

Iteration 58, loss = 0.00269600

Iteration 59, loss = 0.00246016

Iteration 60, loss = 0.00238001

Iteration 61, loss = 0.00232626

Iteration 62, loss = 0.00219560

Iteration 63, loss = 0.00195167

Iteration 64, loss = 0.00182147

Iteration 65, loss = 0.00173425

Iteration 66, loss = 0.00162174

Iteration 67, loss = 0.00153373

Iteration 68, loss = 0.00148125

Iteration 69, loss = 0.00141218

Iteration 70, loss = 0.00134812

Iteration 71, loss = 0.00126114

Iteration 72, loss = 0.00123354

Iteration 73, loss = 0.00114372

Iteration 74, loss = 0.00105909

Iteration 75, loss = 0.00105144

Iteration 76, loss = 0.00104671

Iteration 77, loss = 0.00109535

Training loss did not improve more than tol=0.000100 for 10 consecutive epochs.

[	973	0	4	0	0	2	4	0	7	4]
[	0	1123	1	0	0	0	1	5	1	2]
[	1	3	1000	5	6	0	3	5	3	0]
[	1	2	4	986	1	9	1	3	7	6]
[	0	0	1	0	959	0	5	3	6	9]
[	0	0	0	6	1	867	4	0	3	3]
[	2	2	3	0	3	6	940	0	2	1]
[	1	1	3	4	2	3	0	1006	4	3]
[	1	4	14	5	2	5	0	3	937	2]
[	1	0	2	4	8	0	0	3	4	979]]

테스트 집합에 대한 정확률은 97.7 %입니다.

In [2]:

Console 1/A

Iteration 61, loss = 0.00211330

Iteration 62, loss = 0.00926862

Iteration 63, loss = 0.00344475

Iteration 64, loss = 0.00234911

Iteration 65, loss = 0.00206944

Iteration 66, loss = 0.00198022

Iteration 67, loss = 0.00400893

Iteration 68, loss = 0.00741918

Iteration 69, loss = 0.00238668

Iteration 70, loss = 0.00184809

Iteration 71, loss = 0.00175394

Iteration 72, loss = 0.00169518

Iteration 73, loss = 0.00176099

Iteration 74, loss = 0.00794131

Iteration 75, loss = 0.00365270

Iteration 76, loss = 0.00188159

Iteration 77, loss = 0.00165468

Iteration 78, loss = 0.00162688

Iteration 79, loss = 0.00156371

Iteration 80, loss = 0.00157944

Iteration 81, loss = 0.00991569

Training loss did not improve more than tol=0.000100 for 10 consecutive epochs. Stopping.

[	972	0	7	2	1	3	7	1	7	2]
[	0	1114	5	0	1	0	1	4	0	2]
[	1	4	993	2	2	0	2	12	4	1]
[	2	1	9	989	1	14	0	2	6	5]
[	0	1	5	0	953	0	4	1	5	10]
[	0	1	1	3	1	861	4	0	7	6]
[	2	6	4	0	6	7	940	1	2	0]
[	1	2	3	5	2	1	0	984	3	2]
[	2	5	4	3	2	5	0	6	928	2]
[	0	1	1	6	13	1	0	17	12	979]]

테스트 집합에 대한 정확률은 97.13000000000001 %입니다.

In [3]:

batch\_size : 512  
은닉층 : 100

batch\_size : 128  
은닉층 : 50