



# 딥러닝 홀로서기#9

## 1. MLP Regression with Pytorch

- 방사형 데이터 분포를 가상으로 만든 후, 두 모델들이 비선형 Decision-Boundary를 형성할 수 있는지 테스트(기존 Multi-Label Classification과 동일)
- Data Generation
  - 기존 Multi-Label Classification과 동일하므로 코드는 생략

$$X_{train} \in \mathcal{R}^{8000 \times 2}, Y_{train} \in \mathcal{Z}^{8000}$$

$$X_{val} \in \mathcal{R}^{1000 \times 2}, Y_{val} \in \mathcal{Z}^{1000}$$

$$X_{test} \in \mathcal{R}^{1000 \times 2}, Y_{test} \in \mathcal{Z}^{1000}$$

- Hypothesis Define (Model Define)
  - Multilayer Perceptron Model
    - MLP를 파이토치로 구현할 때는 Linear Transformation을 한 후, Non-Linear Activation Function을 취해 준 것을 한개의 Hidden Layer로 정의, 이런 Hidden Layer가 다 수가 있을 때 Universal Approximation Theorem에 의하여 어떠한 Function도 근사할 수 있다는 개념을 활용하면 됨
    - Hypothesis :  $XW+b$ 를 non-linear activation(ReLU)를 취한 것을  $h$ 라고 하고 이를 다시  $h(x)$ 로 지정하여 softmax함수를 취함

$$\text{Let } \text{relu}(X) = \max(X, 0)$$

$$h = \text{relu}(XW_1 + b_1) \quad (W_1 \in \mathcal{R}^{2 \times 200}, b_1 \in \mathcal{R}^{200}, h \in \mathcal{R}^{N \times 200})$$

$$z = hW_2 + b_2 \quad (W_2 \in \mathcal{R}^{200 \times 3}, b_2 \in \mathcal{R}^3, z \in \mathcal{R}^{N \times 3})$$

$$H = \text{softmax}(z) \quad (H \in \mathcal{R}^{N \times 3})$$

- ① nn.Linear() 함수에 in\_feature값과(여기서는 Input이 2차원이기 때문에 2) out\_feature(여기서는 hidden layer unit 수 =200)를 입력
- ② 200차원의 hidden 벡터 h(X)를 pred\_y로 바꾸기 위해 nn.Linear()에 200과 1을 파라미터로 입력
- ③ non-linear activation function 함수를 정의한다. 여기서 ReLU를 사용하는 이유는 예전 방식인 Sigmoid 함수가 Vanishing Gradient 문제가 발생하기 때문임

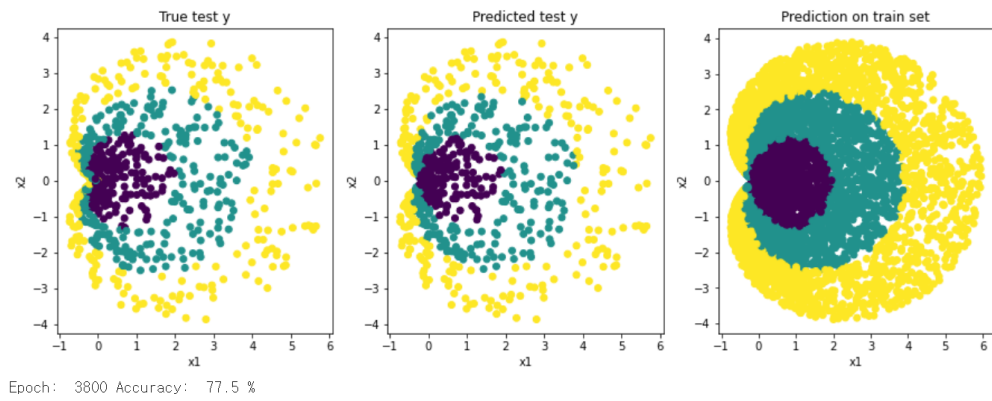
(ReLU : x가 0 이하일때는 y는 0, X가 0 이상일때는 Y=X)

```
class MLPModel(nn.Module):
    def __init__(self, in_dim, out_dim, hid_dim):
        super(MLPModel, self).__init__()
        self.linear1 = nn.Linear(in_dim, hid_dim)
        self.linear2 = nn.Linear(hid_dim, out_dim)
        self.relu = nn.ReLU()

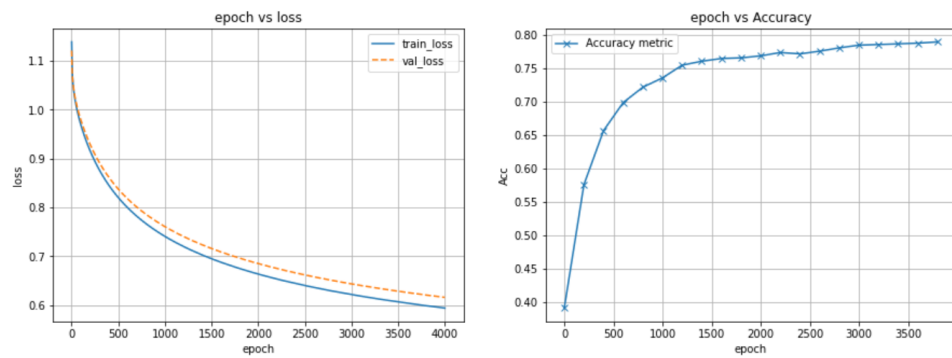
    def forward(self, x):
        x = self.linear1(x)
        x = self.relu(x)
        x = self.linear2(x)
        return x

model = MLPModel(2, 3, 200)
for i in model.parameters():
    print(i.size())
```

- Cost Function
  - 방식 : Cross Entropy 방식을 채택
  - 규칙
    - Multi-Label Classification과 동일하기에 생략
- Train & Evaluation
  - Accuracy를 활용함
  - 결과
    - Non-Linear 데이터를 정확하게 분류하는 것을 볼 수 있음



- Loss도 확실히 빠르게 감소하고 Accuracy도 80%가까이 되는 것을 볼 수 있다



## 2. Assignment#1 - MNIST MLP