Experiment Task A: Handwritten Digit Recognition Using MLP (MNIST)

Description:

Use a three-layer fully connected neural network (MLP) to classify the MNIST handwritten digit dataset

Requirements:

1. Load the MNIST dataset using HuggingFace Datasets or PyTorch.
2. Flatten the input images (28 × 28 into a 784-dimensional vector).
3. Build an MLP model with at least two hidden layers, for example:
   784 → 256 → 128 → 10.
4. Use the following activation functions and perform comparison experiments:
   - ReLU
   - Sigmoid
   - Tanh
5. Use CrossEntropyLoss as the loss function.
6. Use the Adam optimizer for training.
7. Plot the Loss curve and Accuracy curve for both the training set and the validation set.

Experiment Task B: Fashion-MNIST Classification

Description:

Use an MLP model to classify the Fashion-MNIST dataset.

Requirements:

1. Load the Fashion-MNIST dataset using HuggingFace Datasets or PyTorch.
2. Flatten the input images (28 × 28 into a 784-dimensional vector).
3. Build an MLP model with at least two hidden layers, such as:
   784 → 256 → 128 → 10.
4. Use the following activation functions and conduct comparative experiments:
   - ReLU
   - Sigmoid
   - Tanh
5. Use CrossEntropyLoss as the loss function.

6. Use the Adam optimizer for training.
7. Plot the Loss and Accuracy curves for the training and validation sets.

Experiment Task C: Improving MLP Performance Using Batch Normalization

Description:

On the MNIST and Fashion-MNIST datasets, construct an MLP model that incorporates Batch Normalization and Dropout to improve performance and stability.

Requirements:

1. The model must contain linear layers, activation functions, Batch Normalization, and Dropout.
2. Compare the following two models:
   - Standard MLP
   - MLP with BatchNorm + Dropout
3. Compare the following metrics:
   - Whether training convergence becomes faster
   - Whether validation/test accuracy improves