COURSEWORK SPECIFICATION
COMM113 – Deep Learning
Module Leader: **Gaojie (Jay) Jin**
Academic Year: 2025/26

Title: **Building MLP on FashionMNIST**

Submission deadline: ***17th November, 2025***

This assessment contributes bf 30% of the total module mark and assesses the following **intended learning outcomes**:

- Accurately explain a range of key concepts and advanced models of deep learning.
- Implement deep learning models to solve real-world problems.
- Formulate relevant real-world challenges as problems suitable for deep learning approaches.
- Critically evaluate the performance of different deep learning models and architectures and their application to a range of problems.
- Effectively communicate insights and evaluations drawn from research papers.
- Demonstrate independent study and research skills through conducting projects.

Plagiarism

This is an individual assessment.

Plagiarism is interpreted by the university as the act of presenting the work of others as one's own work, without acknowledgement. It is considered academically fraudulent and an offence against university discipline. Your attention is drawn to the university's regulations on plagiarism. Your work will be scrutinised for detection of plagiarism.

Generative AI

This assessment has been categorised as **AI-Prohibited**. You can find further information in the University's policies around using AI in assessed work.

Instructions

You will implement a fully connected multilayer perceptron (MLP) for image classification on the Fashion-MNIST dataset using two approaches (Python Libraries):

1. using PyTorch (high-level framework).
2. using NumPy only (from-scratch implementation).

Dataset: Fashion-MNIST (60,000 train / 10,000 test, 28×28 grayscale, 10 classes).

Please use the attached Jupyter notebook for coursework submission and do all your work in it. Make a separate cell for every few lines of code, and use separate cells for text. Save your file in the format COMM113_STUDENTNUMBER.ipynb and zip it. For example, if your student number is 12345678, save your coursework as COMM113_12345678.ipynb. Once you have done that, zip the file, producing a file called COMM113_12345678.zip. This is the file you will have to upload and submit to ELE.

Task 1 (30%): PyTorch MLP on Fashion-MNIST

Train a fully connected MLP for 10 epochs using PyTorch. Your implementation must include:

1. Loss: Appropriate multi-class loss (e.g., CrossEntropyLoss).
2. Activations: Suitable nonlinearity in hidden layers (e.g., ReLU).
3. Architecture: A reasonable choice of layer widths and depth (e.g., two or more hidden layers).
4. Optimisation: Gradient-descent-based training (e.g., SGD/Adam), appropriate mini-batches.

Output (required):

1. Print/log training loss and training accuracy at the end of each epoch (10 epochs).
2. (Optional) Also report test accuracy to demonstrate generalisation.

Task 2 (70%): NumPy-only MLP on Fashion-MNIST

Train a fully connected MLP for 10 epochs using NumPy only (no PyTorch/TensorFlow/JAX/Autograd libraries). Constraints:

1. You must build the MLP and the training algorithm from scratch with NumPy.
2. Implement forward and backward passes yourself (no automatic differentiation).

Your implementation must include:

1. Loss: Appropriate multi-class loss (e.g., CrossEntropyLoss).
2. Activations: Suitable nonlinearity in hidden layers (e.g., ReLU).
3. Architecture: A reasonable choice of layer widths and depth (e.g., two or more hidden layers).

4. Optimisation: Gradient descent with backpropagation (mini-batches). Include weight initialisation, learning rate, and (optionally) momentum.

Output (required):

1. Print/log training loss and training accuracy at the end of each epoch (10 epochs).
2. (Optional) Also report test accuracy to demonstrate generalisation.

Marking criteria

Task 1 (30%): PyTorch MLP on Fashion-MNIST			
Fail (0-49):	**Pass (50-59):**	**Merit (60-69):**	**Distinction (70+):**
Build a basic MLP with appropriate width and depth.	Basic implementation of MLP with appropriate multi-class loss and activation function.	Good implementation of MLP with appropriate multi-class loss, activation function, and optimisation setting.	Excellent implementation of MLP with all settings, the MLP can be trained, report training loss and accuracy for each epoch.
Task 2 (70%): NumPy-only MLP on Fashion-MNIST			
Fail (0-49):	**Pass (50-59):**	**Merit (60-69):**	**Distinction (70+):**
Build a basic MLP with appropriate width and depth.	Basic implementation of MLP with appropriate multi-class loss and activation function.	Good implementation of MLP with appropriate multi-class loss, activation function, and optimisation setting.	Excellent implementation of MLP with all settings, the MLP can be trained, report training loss and accuracy for each epoch.