

# Image Classification Project

## Objective

The goal of this project is to develop, improve, and optimize neural network models for image classification across two different datasets: CIFAR-10 and Fashion MNIST.

## Project Phases

### Phase 1: Base Model Evaluation

Evaluate a baseline Deep neural network (DNN) with intentionally limited complexity to demonstrate the importance of model design and optimization.

#### Base Model Specifications:

- A simple DNN architecture was used with:
  - One convolutional layer.
  - Two fully connected layers with minimal neurons.
  - Basic activation functions.
  - No regularization.
- The model has low initial accuracy to provide room for improvement.
- You should **evaluate** the base model and compare it to all improved ones.
- You **MUST** use the base model provided in the 'Starter Code Template' in Phase-1 section.

### Starter Code Template (Python with TensorFlow/Keras)

Please refer to the Jupyter notebook 'HW4ML project starter code template.ipynb'

### Phase 2: Model Enhancement Challenge

Students will be required to improve the base model's performance by implementing optimization techniques. The below techniques are examples where the student may use others as well:

1. Increase network depth
2. Add more convolutional layers
3. Implement dropout regularization
4. Use batch normalization
5. Experiment with different activation functions
6. Adjust learning rate

7. Get advantage of previous labs.

While improving the model, students must consider the below hardware efficiency metrics:

1.  $\frac{\text{Test accuracy}}{\text{number of model's parameters}}$ .
2.  $\frac{\text{Test accuracy}}{\text{training time}}$ .
3. Final inference time.
4. Total number of parameters

### Phase 3: Model Compression Techniques

This phase assumes that the model will be deployed on a constraint device. Implement and compare with one or more model compression techniques. You should at least use one of the below two techniques:

1. Pruning
2. Quantization

### Deliverables by 3 May 2025 11:59p.m.

1. **Script Submission:** Complete Python script with:
  - Base model implementation
  - Enhanced model implementation
  - Model compression techniques
2. **Report Submission:** Comprehensive report including:
  - Results table with the hardware efficiency metrics mentioned in phase 2 for:
    - Base Model
    - Enhanced Model
    - Compressed Model
  - Clear presentation of all used hyperparameters for each phase.
  - Comparative analysis of different approaches
  - Visualization of model performance and provide a proper results discussion.
  - Insights and observations
  - The report should contain at least ~1200-1500 words and at most ~2500-3000 words.

**You may have multiple submission attempts, however, only the last attempt will be graded.**

## Evaluation Criteria

- Test accuracy improvement from base model. Also, test accuracy per number of model's parameters is considered.
- Effectiveness of enhancement techniques.
- Quality of model compression
- Code quality and documentation
- Depth of analysis and main concepts understanding in the report.

## Submission Guidelines

- Submit a single ZIP file containing:
  1. Complete Python script
  2. Detailed report (PDF) as described in deliverable point 2.
  3. Any additional visualization or analysis files.
- The single ZIP file is named as: Name<studentFirst>\_<LastName>\_V<versionnumber>.rar. An example would be: Abdullah\_AIMohammed\_V2.rar

## Grading Rubric

- Base Model Evaluation: 10%.
- Model Enhancement: 30%
- Model Compression: 25%
- Documentation and Analysis: 35%