Objective

The objective of this lab is to apply dynamic quantization to a logistic regression model in PyTorch, to understand the effects of quantization on model size, accuracy, and inference time. This report will present results from both the original and quantized models, comparing their performances.

Part 1: Setup and Data Preparation

1. **Environment Setup**
    - Imported necessary libraries including `torch`, `torchvision`, `sklearn`, and others required for data manipulation, training, and evaluation.
2. **Data Loading**
    - **Dataset**: Used the MNIST dataset from `torchvision.datasets`.
    - **Transformations**: Normalized the data to ensure consistency in training.
    - **Data Preparation**:
        - Extracted features (`X`) and labels (`y`) from the dataset.
        - Flattened the images (28x28 pixels) into vectors of length 784.
        - Split the dataset into training (80%) and test (20%) sets.

Part 2: Model Building

1. **Logistic Regression**
    - Implemented a logistic regression model using a neural network with a single linear layer.
    - Defined input dimensions (`28x28 = 784`) and output dimensions (10 classes for digits 0-9).

Part 3: Original Model Evaluation

1. **Training**
    - The model was trained for **10 epochs** using the cross-entropy loss function and the SGD optimizer.
    - **Loss** values were printed for each epoch to monitor the training progress.
2. **Evaluation**
    - **Test Accuracy**: Measured on the test set.

        - **Result**: `69.51%`
    - **Model Size**:
        - Saved and measured the size of the original model.
        - **Result**: `32.19 KB`
    - **Inference Time**:
        - Measured the time taken for the model to make predictions on the test set.
        - **Result**: `34.60 ms`

Part 4: Quantization

1. **Dynamic Quantization**
    - Defined a function `quantize_model` to scale the weights of the original model to an 8-bit integer representation.
    - Used a **scaling factor** of `2**7` to transform the weights.

Part 5: Inference Using the Quantized Model

1. **Inference**
    - Developed a function to perform inference using the quantized model to check its accuracy on the test set.
2. **Model Size & Inference Time**
    - Saved and measured the size of the quantized model:
        - **Quantized Model Size**: 27.04 KB
    - Measured inference time:
        - **Quantized Inference Time**: `14.93 ms`

Part 6: Comparison Results

Metric	Original Model	Quantized Model
Accuracy	69.51%	69.48%
Model Size	32.19 KB	27.04 KB
Inference Time	33.40 ms	14.93 ms

1. **Model Size Comparison**:

    - The quantized model is slightly smaller, achieving a reduction of approximately 16**%**.
2. **Inference Time**:
    - The quantized model has a very fast inference time i.e. 2.23x faster compared to the original model.
3. **Accuracy**:
    - There is a slight drop in accuracy, but the quantized model maintains good performance, only about **0.90%** lower than the original.

Conclusion

The results demonstrate the effectiveness of dynamic quantization in reducing model size and speeding up inference, with minimal loss in accuracy. Such techniques are beneficial for deploying machine learning models on resource-constrained devices, where storage and computation are limited. This lab provided valuable insights into model optimization, making models more efficient without significantly sacrificing performance.