

Question 1: Convolutional Neural Network for MNIST Classification

You are required to create a Convolutional Neural Network (CNN) using Keras to classify the MNIST dataset. Follow the steps below to complete the task.

MNIST Dataset Overview:

The MNIST dataset consists of 70,000 grayscale images of handwritten digits (0-9). Each image is 28x28 pixels. The dataset is divided as follows:

- Training Set: 60,000 images
- Testing Set: 10,000 images

Each image is labeled with the corresponding digit.

Tasks:

1. Preprocessing Steps:

- Load the MNIST dataset using Keras.
- Normalize the pixel values of the images to the range [0, 1].
- Reshape the images to include a channel dimension (28, 28, 1) suitable for a CNN.
- Convert the labels to one-hot encoding using the `to_categorical` function.

2. Model Architecture:

- Create a Convolutional Neural Network with the following layers:
 1. A convolutional layer with at least 32 filters, a kernel size of 3x3, and ReLU activation.
 2. A max pooling layer with a pool size of 2x2.
 3. A second convolutional layer with at least 64 filters, a kernel size of 3x3, and ReLU activation.

4. A max pooling layer with a pool size of 2x2.
5. A third convolutional layer with at least 128 filters, a kernel size of 3x3, and ReLU activation.
6. A fourth convolutional layer with at least 256 filters, a kernel size of 3x3, and ReLU activation.
7. Flatten the output from the final pooling layer.
8. A fully connected layer with 128 units and ReLU activation.
9. A second fully connected layer with 50 units and ReLU activation.
10. A third fully connected layer with 50 units and ReLU activation.
11. An output layer with 10 units and softmax activation for classification.

3. Training and Evaluation:

- Train the model using the Adam optimizer and categorical cross-entropy loss.
- Evaluate the model on the test set and report the accuracy.

Submission Requirements:

- Submit your code as a .py or .ipynb file.
- Include a report or screenshot showing the model's performance metrics on the test set.