

Explanation:

1. **Data Loading and Preprocessing:** The MNIST dataset is loaded and normalized to a range of 0 to 1 by dividing by 255.
2. **Exploratory Data Analysis (EDA):** A simple visualization of the first 25 samples from the training set is shown using Matplotlib.
3. **CNN Model Architecture:**
 - **Convolutional Layers:** These layers apply a convolution operation to the input, passing the result to the next layer. Here, we use three convolutional layers.
 - **Max Pooling Layers:** These layers downsample the input, reducing the spatial dimensions.
 - **Flatten Layer:** This layer flattens the 2D matrix to a 1D vector.
 - **Dense Layers:** Fully connected layers are used to make predictions.
 - **Output Layer:** The final layer uses the Softmax activation function to output probabilities for each class (0-9 digits).
4. **Compilation:** The model is compiled with the Adam optimizer, sparse categorical crossentropy as the loss function, and accuracy as the metric.
5. **Training:** The model is trained for 5 epochs, with training and validation accuracy/loss monitored.
6. **Evaluation:** The model is evaluated on the test set, and the accuracy is printed.
7. **Visualization:** The training and validation accuracy/loss are plotted over epochs.