MNIST Digit Recognizer

Project Title: MNIST Digit Recognizer
Author: Daniel (Minyoung) Park
Date: 26 May 2025
Outside Sources Used …

1. https://www.geeksforgeeks.org/mnist-dataset/
2. https://www.ibm.com/think/topics/machine-learning
3. https://lakefs.io/blog/data-preprocessing-in-machine-learning/
4. https://www.purestorage.com/knowledge/what-is-data-preprocessing.html
5. https://www.youtube.com/watch?v=ukzFI9rgwfU
6. https://www.geeksforgeeks.org/multi-layer-perceptron-learning-in-tensorflow/
7. https://www.geeksforgeeks.org/learning-model-building-scikit-learn-python-machine-learning-library/
8. https://stackoverflow.com/questions/43577665/deskew-mnist-images
9. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html
10. https://scikit-learn.org/stable/modules/calibration.html

Overview

This project implements an interactive MNIST digit recognition application with a graphical user interface built using Tkinter. It supports both loading existing models and training a new Multi-Layer Perceptron (MLP) classifier when none is found. Users can upload an image file of a handwritten digit, draw digits directly in the application, and visualize prediction results and confidence scores.

Features

- Preprocessing routines: deskewing, centering, thresholding, resizing, and normalization
- Automatic training of an MLP classifier when no pre-trained model file is present
- Model persistence using joblib
- GUI for uploading images and drawing digits by hand

- Real-time preview of preprocessed digit
- Display of prediction probabilities as a bar chart
- Confusion matrix visualization on validation data
- Interactive dialogs for handling border artifacts

Installation

1. Install PyCharm:
   a. MacOS: https://www.jetbrains.com/pycharm/download/?section=mac
   b. Windows: https://www.jetbrains.com/pycharm/download/?section=windows
   c. Linux: https://www.jetbrains.com/pycharm/download/?section=linux
2. Open the submitted project in PyCharm
   a. Launch PyCharm and select Open to browse to the submitted project via MyTA.
3. Install Scikit-Learn via PyCharm Terminal by executing one of the following commands
   a. pip install scikit-learn
   b. pip install sklearn
4. Install any additional required packages in addition to Scikit-Learn
   a. If additional packages are missing, PyCharm will show a red underline in the code.
   b. Click the line with the red underline.
   c. Click the red warning icon in the top-left corner of the editor.
   d. Select Install package 'package_name' from the dropdown menu.

Usage

Training the Model

If a saved model file (mnist_mlp.pkl) is not found, the application will automatically fetch the MNIST dataset from OpenML, train an MLP classifier with two hidden layers, and save the trained model and validation data to mnist_mlp.pkl.

Running the GUI

In the PyCharm project, press “Run”:A window titled **"MNIST Digit Recognizer"** will open.

Using the Application

- **Upload Digit Image**: Click this button to select an image file (.png, .jpg, etc.) containing a handwritten digit. The app will preprocess the image, display a preview, predict the digit, and show a probability bar chart.

- **Draw Digit**: Opens a canvas where you can draw a digit freehand. Use the brush size slider, **Clear** to reset, **Undo** for the last stroke, and **Done** to submit. The drawn digit will be preprocessed and classified.

- **View Confusion Matrix**: Displays the confusion matrix for validation data in a Matplotlib window, helping you assess model performance.

Configuration

- **MODEL_PATH**: Path to the saved model file (default: mnist_mlp.pkl).

- **Scaling & Preprocessing**: Modify the parameters in preprocess_opencv() for blur kernel size, thresholding, or canvas dimensions if needed.

- **MLP Parameters**: Adjust hyperparameters (hidden layers, activation, solver, learning rate, etc.) in the classifier instantiation.