CAP 4630 Assignment: Nearest Neighbor Classification for Digit Recognition

Objective

This project will guide you through implementing and evaluating a k-nearest neighbors (k-NN) classifier for digit classification using the `sklearn` digits dataset. By the end of this project, you will:

1. Load, explore, and visualize the dataset.
2. Train and evaluate a k-NN classifier using various values of k.
3. Analyze performance metrics quantitatively and qualitatively.

Guidelines

- Use **Jupyter Notebook** for this assignment.
- Organize your code into clear sections with descriptive titles, comments, and Markdown cells.
- Use Markdown cells to write observations and conclusions at each step.

Instructions

1. Load and Explore the Dataset

Objective: Load the dataset and examine its structure.
Tasks:

- Load the `digits` dataset from the `sklearn.datasets` library.
- Display the **shape of X** (feature data) and **y** (label data) to understand the dataset's dimensions.
- Randomly split the dataset into training and test sets, reserving **500 samples for the test set**.

- Display the shapes of `X_train`, `y_train`, `X_test`, and `y_test` to confirm the correct split.

Rubric (15 points)

- Correctly load and display shapes of `X` and `y`: **3 points**
- Implement a random split with 500 test samples: **8 points**
- Display and verify shapes of `X_train`, `y_train`, `X_test`, and `y_test`: **4 points**

2. Visualize Training Data

Objective: Become familiar with the images in the dataset.
Tasks:

- Select **10 random images** from `X_train` and their corresponding labels from `y_train`.
- Display these images in a grid using `matplotlib`, with each image labeled with its corresponding digit. Use grayscale to emphasize pixel intensity.

Rubric (10 points)

- Correctly select and display 10 random images from `X_train`: **5 points**
- Display labels accurately and clearly: **3 points**
- Proper use of grayscale and clean layout of images: **2 points**

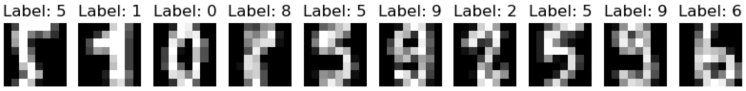

Figure 1: Training data with labels

3. Implement the k-Nearest Neighbor Classifier

Objective: Train multiple k-NN classifiers to predict digit labels.
Tasks:

- Use `sklearn`'s `KNeighborsClassifier` to create and train a k-NN classifier.
- Train five separate classifiers with $k = 1, 3, 5, 7$, and 9.
- For each classifier, fit the model using `X_train` and `y_train`.

Rubric (30 points)

- Correctly implement and train each classifier with specified values of k: **6 points per classifier (30 points total)**

4. Evaluate and Compare Classifiers

Objective: Compare classifier performance for different values of k using F1 scores.
Tasks:

- For each trained classifier (one per k value), predict the labels of `X_test` and calculate the **F1 score** using `y_test` as the ground truth.
- Use the `f1_score` function from `sklearn.metrics` with `average='weighted'` to account for class distribution.
- Display the F1 scores in a **table** (using Markdown or `pandas.DataFrame`) for each k value.
- Summarize findings in a Markdown cell, discussing any patterns or trends observed across different k values, including changes in F1 scores.

Rubric (25 points)

- Correctly calculate F1 scores for each classifier: **3 points per classifier (15 points total)**
- Create a clear, organized table for F1 scores: **5 points**
- Insightful discussion on the impact of different k values on performance: **5 points**

5. Visualize Qualitative Predictions

Objective: Evaluate classifier predictions qualitatively.
Tasks:

- For each classifier (with $k = 1, 3, 5, 7, 9$), select a few random images from `X_test` to display predictions.
- Display each image with its **predicted label** and **true label**, annotating each plot with the classifier's k value.
- In a concluding Markdown cell, reflect on the results, noting any patterns or observations, especially for cases where predictions were correct or incorrect.

Rubric (20 points)

- Correctly select random test samples and display images for each classifier: **10 points**
- Clear and accurate labeling of predictions and true labels: **10 points**

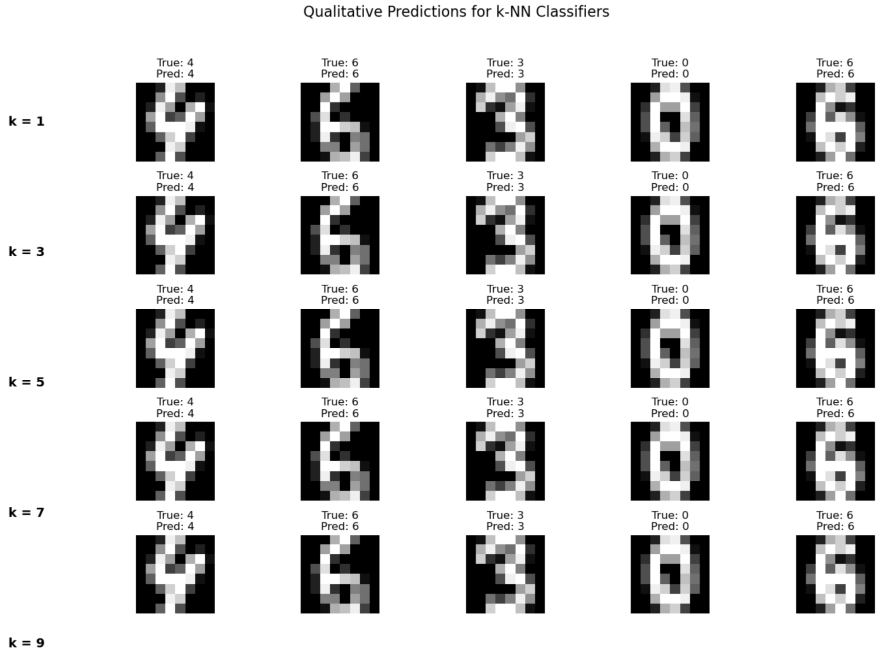

Figure 2: Qualitative Predictions for each classifier

Expected Output and Deliverables

- **Dataset Shapes:** Printout showing shapes of `X`, `y`, `X_train`, `y_train`, `X_test`, and `y_test`.
- **Visualizations:**
  - Display 10 random images from `X_train` with their labels.
  - Plot qualitative predictions for each classifier (5 plots total, one per k value).
- **Evaluation Metrics:** A table of F1 scores for each k value on the test dataset.
- **Analysis and Observations:** A summary of findings on how k impacts model performance, supported by both F1 scores and qualitative predictions.

Total Score: 100 points

This project will strengthen your understanding of k-NN classifiers and enhance your ability to assess model performance through both quantitative and qualitative metrics. Ensure that your notebook is well-organized, with each cell properly documented to support a clear and thorough analysis.