FINAL REPORT

Handwritten Digit Classification using Classical Machine Learning

1.Flow Diagram

Load MNIST CSV Dataset

|

Separate Features (X) and Labels (y)

|

Data Exploration(Class distribution, sample images)

|

Normalize Pixel Values (0–255 → 0–1)

|

Train-Test Split (80% / 20%)

|

Standardization (StandardScaler)

|

Apply PCA (Dimensionality Reduction)

|

Train Models

|----> KNN (PCA features)

|----> SVM (PCA features)

|----> Decision Tree (Original features)

|

Evaluate Models(Accuracy + Confusion Matrix)

|

Voting Ensemble (Majority Voting)

|

KNN from Scratch

|

Compare Results & Observations

1. Objective

The objective of this assignment is to classify handwritten digits (0–9) using a subset of the MNIST dataset provided in CSV format. Each image is represented as a flattened vector of 784 grayscale pixel values. The task focuses on implementing and evaluating **classical machine learning models**, including at least one algorithm built **from scratch**, and analyzing their performance.

2. Dataset Description

- **Dataset**: MNIST Digit Recognizer (CSV format)
- **Features**: 784 pixel values (28×28 image flattened)
- **Target**: Digit label (0–9)
- **Pixel range**: 0–255 (grayscale)

The dataset was divided into training and testing sets using an 80:20 split while preserving class distribution.

3. Data Preprocessing

The following preprocessing steps were applied:

- Normalization of pixel values to the range [0,1]
- Standardization using StandardScaler
- Dimensionality reduction using **PCA (50 components)** to improve computational efficiency, especially for KNN

4. Models Implemented

The following classical ML models were trained and evaluated:

1. **K-Nearest Neighbors (KNN)**

    - Trained on PCA-reduced features for faster execution
2. **Support Vector Machine (SVM)**
    - RBF kernel used for non-linear decision boundaries
3. **Decision Tree Classifier**
    - Used as a baseline tree-based model
4. **KNN from Scratch (Bonus)**
    - Implemented manually using Euclidean distance
    - Trained on a smaller subset to reduce computation time

5. Evaluation Metrics

Models were evaluated using:

- **Accuracy score**
- **Confusion matrix**
- Visualization of **misclassified samples**

6. Ensemble Learning (Bonus)

A **Voting Ensemble** was implemented by combining predictions from KNN, SVM, and Decision Tree using majority voting. The ensemble achieved higher accuracy than individual models, demonstrating improved generalization.

7. Observations and Conclusions

- PCA significantly reduced training time for KNN without major loss in accuracy
- SVM performed best among individual models
- Voting ensemble achieved the highest accuracy overall
- KNN from scratch validated understanding of distance-based learning
- Classical ML models perform competitively on MNIST with proper preprocessing

8. Conclusion

This assignment demonstrates that with effective preprocessing and dimensionality reduction, classical machine learning algorithms can achieve strong performance on

image classification tasks. Ensemble learning further improves robustness, and implementing algorithms from scratch deepens conceptual understanding. Misclassified images were visualized for KNN, SVM, and Decision Tree models. Most errors occurred between visually similar digits such as **3 vs 5**, **4 vs 9**, and **7 vs 1**.SVM produced fewer misclassifications due to its ability to learn non-linear decision boundaries.KNN errors were influenced by noisy neighbors, while Decision Tree showed signs of overfitting.