KNN-Based Automation for Rapid and Accurate Exam Grading

Jean Baptiste Habyarimana

Content

Introduction

Objective

Automate grading of exam papers by employing a Naive Bayes and KNN-based approach for recognizing numerical handwritten digits answers.

Motivation

Time-consuming due to manual processing.
Unclear handwritten digits.
High disputed grades and appeals from students.

Scope

Implement both approaches (KNN and Naive Bayes Classifiers) to classify numerical digits from handwritten numeric answers. The scope is limited to numeric digit recognition, without addressing the recognition of handwritten text or letters.

Use Case Summary

- **Scenario**:

Teachers need to quickly and accurately grade exam papers that contain handwritten numeric answers.

- **Challenges:**

Variability in handwriting styles

Non-Uniform grading method across different exams and graders.

Slow processing time of exams from multiple classes or institutions

- **Solution:**

Minimizes the chance of grading errors.

Ensures fair grading regardless of the number written.

Dataset Overview

Dataset: MNIST Handwritten Digits Dataset.

Training Data:

Samples: 60000

Features: 784 pixel values (28x28 images)

Labels: Digits (0-9)

Testing Data:

Samples: 10,000

No empty values or duplicates

Process

1. Data Split

Split datasets (train and test) into features (x) and labels (y) subsets

2. Model implementation

KNN: Uses distance-based voting to classify numerical digits
Naive Bayes Classifier: Applies Gaussian distribution to model pixel values for each digit class.

3. Model Evaluation

1. **Implementation**: KNN and Naive Bayes Classifiers for numerical digits recognitions.
2. **Evaluation metrics**: classifiers using accuracy for assess classification performance.

Digit-wise Comparison: accuracy analysis on each digit (0-9) for both classifiers to identify which accurately classifies specific digits, with the goal of coupling them for enhanced performance

preprocessing and data split

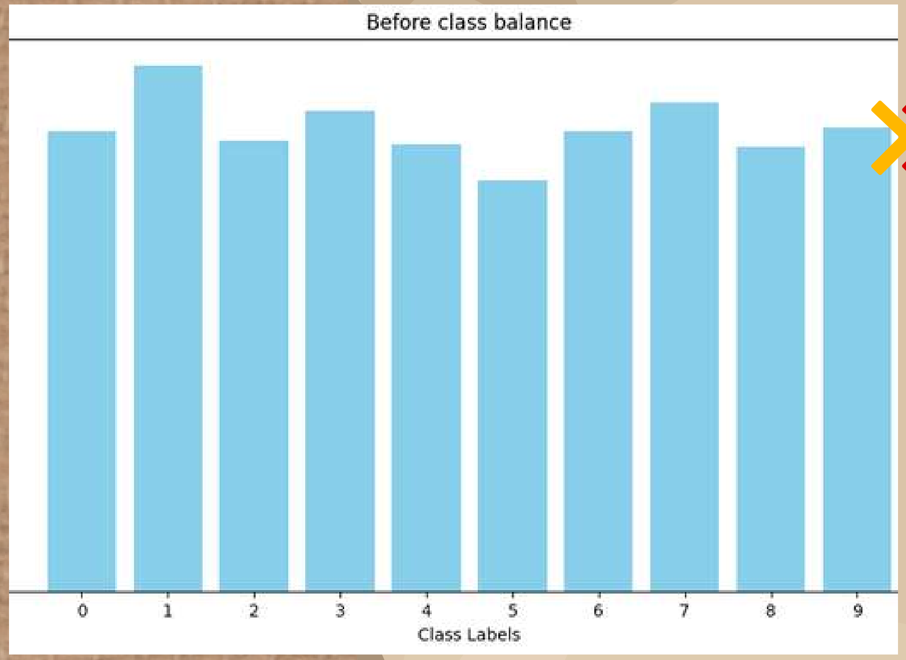
Data Split

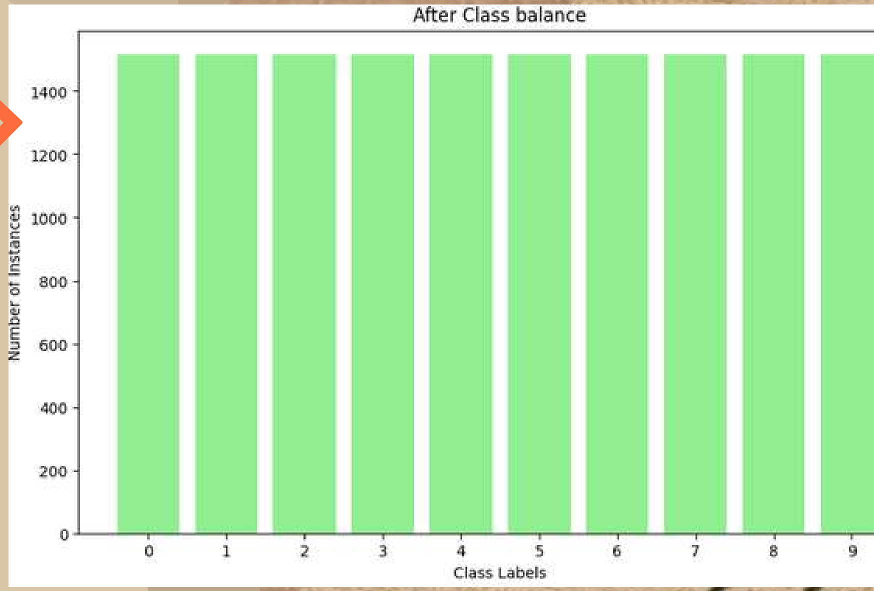
Train and test datasets were split into features (x) and class (y)

Normalization

pixel values were normalized from their original range of 0-255 to a range of 0-1 using Min_Max Normalization approach.

Class balance





Models Implementation

- **K-Nearest Neighbors (KNN)**: Classifies a digit based on the "K" closest training samples.

KNN Implementation Details:

K: number of neighbors considered (k=5)

prediction: Majority voting with weighted distances.

- **Naive Bayes**: Assumes features are normally distributed and independent.

Naive Bayes Implementation Details:

Training: For each digit class, mean and variance for each pixel were calculated.

Prediction: Compute log-probability for each class and predict the highest

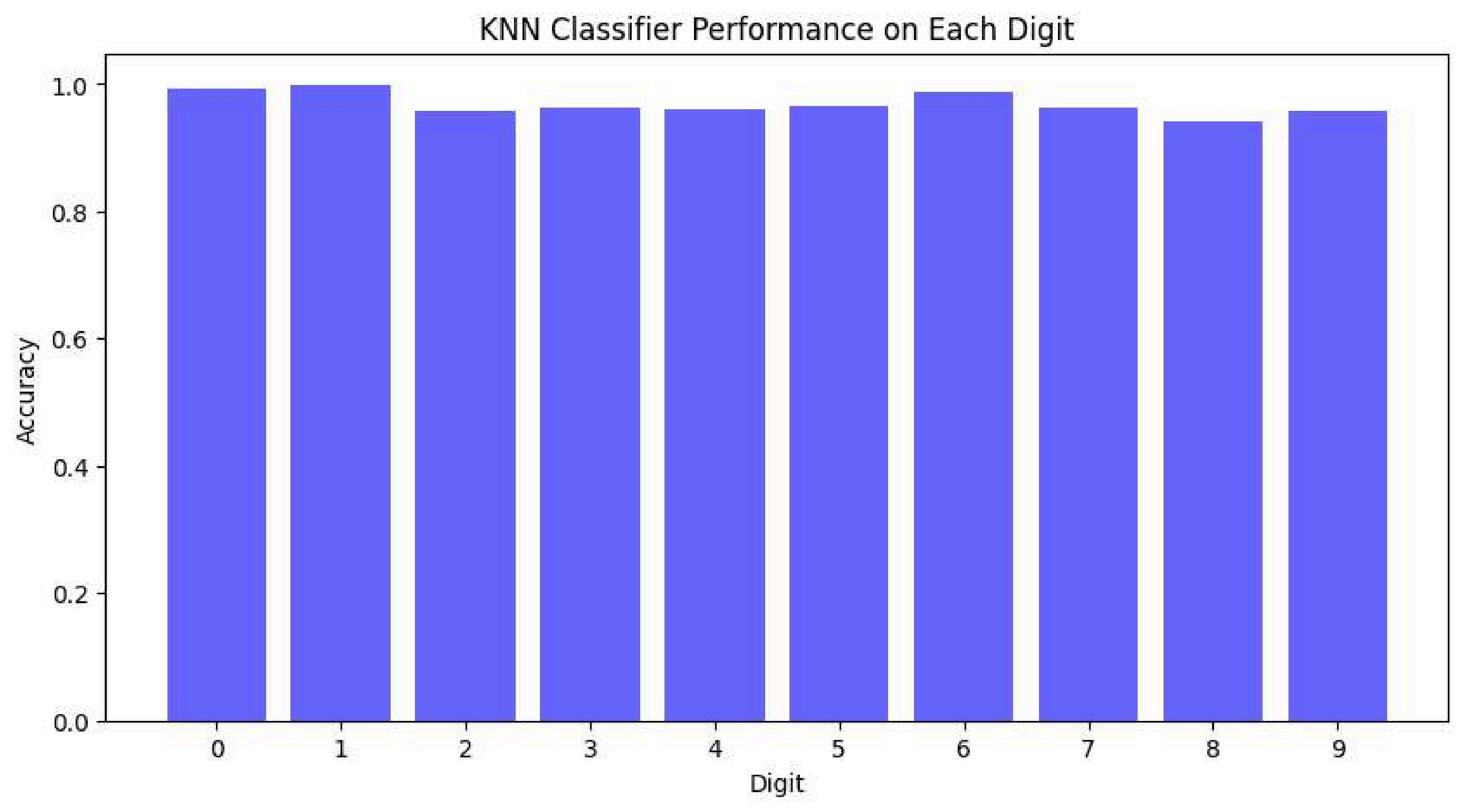
Performance evaluation

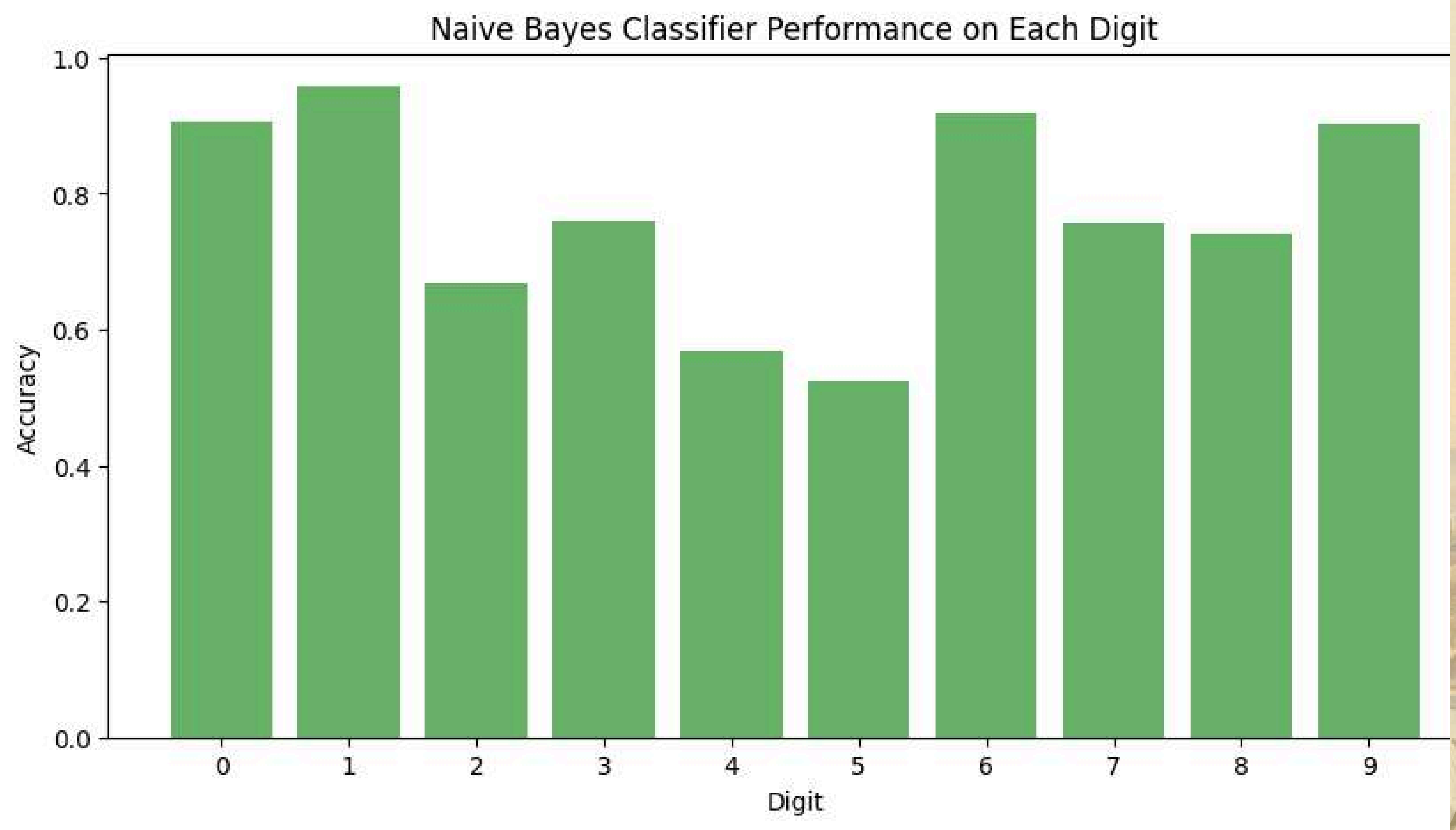
KNN Predictions

Strong and consistent performance across all digits ranged from 0.94 to 0.998. This consistency makes it well-suited for applications of exam grading as reliability across all numbers is crucial

Naive Bayes Predictions

Performance ranged from 0.52 to 0.96 with no consistent performance across all digits. This indicates that the classifier performs better on some digits than others





Conclusions

In this project, K-Nearest Neighbors (KNN) and Naive Bayes classifiers were evaluated for automating the grading of exam papers through recognition of numerical handwritten digits.

The KNN classifier achieved an overall accuracy of 96.91% which indicates its robustness and reliability in accurately recognizing handwritten numerical answers.

The Naive Bayes classifier achieved an overall accuracy of 77.46% with performance variability which affect its overall efficacy.

For the objective of automating exam grading based on recognizing numerical handwritten answers, KNN proves to be the more reliable approach.

Thank you for your attention!!