**CS412 - Homework 1: k-NN and Decision Trees on MNIST Dataset**

Huseyin Samed Dagci

30996

İçindekiler

[**Introduction** 2](#_Toc191068329)

[Overview 2](#_Toc191068330)

[Importance of This Task 2](#_Toc191068331)

[Methodology 2](#_Toc191068332)

## Introduction

### Overview

The objective of this assignment is to classify handwritten digits from the **MNIST dataset** using two different machine learning models: **k-Nearest Neighbors (k-NN) and Decision Trees**. The MNIST dataset consists of **grayscale images of digits (0-9)**, each having a size of **28x28 pixels**. The classification task aims to predict the correct digit based on pixel intensity values.

### Importance of This Task

Handwritten digit recognition is a fundamental problem in computer vision and machine learning. The performance comparison between **k-NN and Decision Trees** provides insight into the trade-offs between model complexity, accuracy, and computational efficiency.

### Methodology

To complete this task, the following steps were taken:

1. **Data Preprocessing & Analysis**:
   * Normalization of image pixel values for uniform scaling.
   * Splitting the dataset into **training, validation, and test sets**.
   * Exploratory Data Analysis (EDA) to understand class distributions and dataset properties.
2. **Model Training & Hyperparameter Tuning**:
   * Training and optimizing **k-NN** by varying the number of neighbors k.
   * Training and optimizing **Decision Trees** by tuning depth and splitting criteria.
3. **Model Evaluation & Performance Analysis**:
   * Measuring accuracy, precision, recall, and F1-score.
   * Analyzing misclassifications and identifying patterns.
   * Visualizing confusion matrices to better understand classification errors.

By following this structured approach, we aim to build a strong understanding of these models and their performance on image-based classification tasks.

## Dataset & Preprocessing

### Dataset Overview

The **MNIST dataset** consists of 70,000 grayscale images, where each image represents a handwritten digit (0-9). The dataset is pre-split into:

* **Training set:** 60,000 images
* **Test set:** 10,000 images Each image is **28x28 pixels**, and pixel values range from **0 (black) to 255 (white)**.

### Data Preprocessing Steps

1. **Normalization**: Since pixel values range from 0 to 255, we normalize them to a range of [0,1] by dividing each pixel value by 255. This helps in stabilizing the learning process and improving convergence for distance-based models like k-NN.
   * Formula: X\_normalized = X / 255.0
2. **Splitting the Data**:
   * The training set was further split into **80% training and 20% validation**.
   * This ensures that hyperparameter tuning is done using a validation set before testing on unseen data.
3. **Exploratory Data Analysis (EDA)**:
   * **Class Distribution Check:** The dataset was analyzed to confirm that all digits (0-9) are evenly represented.
   * **Pixel Value Statistics:** The mean and standard deviation of pixel intensities were computed to understand overall brightness and contrast distribution.
   * **Sample Visualization:** A set of images from each class was displayed to confirm dataset correctness.

### Key Observations from EDA

* The dataset is **balanced**, meaning each digit has roughly the same number of samples, preventing any bias towards a particular class.
* Mean pixel intensity was approximately **0.1307**, with a standard deviation of **0.3082**, confirming that images are well-distributed in terms of brightness levels.
* Sample images showed **variability in handwriting styles**, which could influence model performance.

metin, yazı tipi, ekran görüntüsü, beyaz içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

## Model Training & Hyperparameter Tuning

### k-Nearest Neighbors (k-NN)

**Why k-NN?**

k-NN is a simple yet effective classification algorithm that predicts the class of a test instance based on the majority vote of its k-nearest neighbors. It is particularly useful for pattern recognition tasks like digit classification.

**Hyperparameter Tuning**

To optimize k-NN, different values of k were tested: (1, 3, 5, 7, 9). The model was trained on the training set and evaluated on the validation set.

metin, yazı tipi, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

metin, çizgi, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Best Hyperparameter**

* The best-performing value was **k = 1**, achieving **97.41% validation accuracy**.
* A lower k value results in **higher variance (overfitting)**, while higher k results in **smoother decision boundaries (better generalization)**.
* Since k=1 had the highest accuracy, it was chosen for final evaluation on the test set.

metin, ekran görüntüsü, menü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

### ****Decision Tree Classifier****

**Why Decision Trees?**

Decision Trees are widely used in classification tasks due to their **interpretable structure**. They work by recursively splitting the dataset based on feature values.

**Hyperparameter Tuning**

The following hyperparameters were tuned:

* **Max Depth:** {2, 5, 10}
* **Min Samples Split:** {2, 5}

**Best Hyperparameter**

* Best Parameters: max\_depth=10, min\_samples\_split=2 with 0.8582 accuracy
* Increasing depth improved accuracy but risked overfitting.

metin, ekran görüntüsü, menü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

## Model Evaluation & Performance Analysis

**Final Model Performance on Test Set**

### k-NN Performance

The optimized k-NN model was evaluated on the test set, achieving the following performance metrics:

* **Accuracy:** 97%
* **Precision, Recall, F1-score:** Consistently high for all digits (see classification report)
* **Confusion Matrix:** Shows that the model correctly classifies most digits, with minimal misclassification.

### Decision Tree Performance

The best Decision Tree model (max\_depth=10, min\_samples\_split=5) was also evaluated on the test set:

* **Accuracy:** 86%
* **Precision, Recall, F1-score:** Lower compared to k-NN, indicating more misclassifications.
* **Confusion Matrix:** Higher error rate for certain digits, suggesting the need for further tuning.

**Misclassification Analysis**

* k-NN made most errors between visually similar digits (e.g., 4 vs. 9, 3 vs. 5).
* Decision Tree had higher confusion in digits with complex patterns.

Overall, k-NN outperformed Decision Trees in this task, making it the preferred model for MNIST digit classification.

