Title: Image Classification Using K-Nearest Neighbors (KNN): A Comparison of Manhattan (L1) and Euclidean (L2) Distance Metrics with 5 Fold Cross Validation

1. Introduction

Image classification is a core problem in computer vision where the objective is to assign a correct label to an input image based on learned patterns. In this assignment, a distance based machine learning algorithm, K-Nearest Neighbors (KNN), is implemented for image classification. The main goal is to compare the performance of two distance metrics—Manhattan (L1) and Euclidean (L2)—using 5-fold cross-validation. The experiment is conducted on a grayscale image dataset consisting of three classes and 300 total images, and the effect of different values of K on classification accuracy is analyzed.

2. Dataset Description

The dataset contains three distinct image classes, with 100 images per class, resulting in a total of 300 images. The images are stored in separate directories corresponding to each class. The dataset is used only for training and validation through cross validation, as required by the assignment.Each image is preprocessed to ensure consistent input size and format before being passed to the classifier.

3. Image Preprocessing

To reduce computational complexity while preserving important visual features, the following preprocessing steps were applied:

1. Grayscale Conversion: All RGB images were converted to grayscale to remove color dependency and reduce dimensionality.
2. Image Resizing: All images were resized to 32 × 32 pixels using interpolation.
3. Flattening: Each 32 × 32 grayscale image was flattened into a one dimensional feature vector of length 1024.
4. Data Type Conversion: Pixel values were stored as floating point values to ensure numerical stability.

These preprocessing steps ensure that all images have uniform dimensions and are suitable as input to the KNN algorithm.

4. K-Nearest Neighbors (KNN) Algorithm

KNN is a supervised, non parametric, instance based learning algorithm. It does not build an explicit model during training. Instead, it stores all training data and makes predictions based on similarity measures at inference time.

Working Principle of KNN:

1. Store all feature vectors and corresponding class labels.
2. Compute the distance between a test sample and all training samples.
3. Select the K nearest samples based on the chosen distance metric.
4. Assign the class label based on majority voting among the K neighbors.

5. Distance Metrics

Two different distance metrics were implemented and compared in this assignment:

a. Manhattan (L1) Distance

Manhattan distance measures the absolute sum of differences between corresponding pixel values.

Formula:

$$D(x, y) = \Sigma |xi - yi|$$

This metric is less sensitive to outliers and emphasizes linear feature wise differences.

b. Euclidean (L2) Distance

Euclidean distance measures the straight line distance between two image vectors.

Formula:

$$D(x, y) = \sqrt{\Sigma} (x_i - y_i)^2$$

This metric is sensitive to large differences and captures overall geometric similarity.

6. 5 Fold Cross Validation

To evaluate the performance of the classifier reliably, 5 fold cross validation was applied as follows:

- The dataset was divided into 5 equal folds.
- In each iteration, 4 folds were used for training and 1 fold for validation.
- This process was repeated 5 times, ensuring each fold is used exactly once for validation.
- The final accuracy for each value of K was obtained by computing the mean accuracy across all 5 folds.

This method reduces bias and provides a more robust estimate of model performance.

7. Hyperparameter K Selection

The number of neighbors K is a critical hyperparameter in KNN. In this experiment, multiple values of K were tested. For each value of K:

- Cross validation was performed using Euclidean (L2) distance.
- Cross validation was also performed using Manhattan (L1) distance.
- The average accuracy over 5 folds was computed for both metrics.

The best value of K was selected based on the highest mean validation accuracy.

8. Results and Performance Analysis

The performance of the KNN classifier was evaluated for different values of K using both distance metrics. A graph was plotted showing:

- X axis: Values of K
- Y axis: Average classification accuracy
- Two curves representing:
  - Euclidean (L2) distance
  - Manhattan (L1) distance

The results show that accuracy varies with K and that Euclidean distance generally provides more stable and higher accuracy across different values of K.

9. Comparison Between L1 and L2 Distance

Criterion	**Manhattan (L1)**	**Euclidean (L2)**
Sensitivity to Noise	Lower	Higher
Geometric Interpretation	Linear	Geometric
Stability Across K	Moderate	High
Overall Accuracy	Good	Better

From experimental observation, Euclidean distance outperformed Manhattan distance for this grayscale image dataset.

10. Best K Selection

After evaluating multiple values of K, the best K for each distance metric was selected based on maximum average 5 fold accuracy:

- Best K for Euclidean (L2): Selected based on highest mean accuracy
- Best K for Manhattan (L1): Selected based on highest mean accuracy

These best K values were then used for final visualization using the top 5 predictions.

11. Top 5 Predictions

Using the best K values for both Euclidean and Manhattan distances, the model was applied to visualize the top 5 predictions:

- Five test images were selected.
- For each image, the following were displayed:
  - Original image
  - True class label
  - Predicted class label

These visual results provide a qualitative evaluation of the model's performance in addition to numerical accuracy.

12. Discussion

The results show that the KNN classifier performs well on grayscale image data with proper preprocessing. The Euclidean (L2) distance consistently achieved higher accuracy than the Manhattan (L1) distance, indicating better feature similarity measurement. The choice of K significantly affected performance: smaller K values led to overfitting, while larger K values caused underfitting. A moderate K value provided the best balance between bias and variance. The use of 5 fold cross validation ensured reliable model evaluation and accurate selection of the optimal K.

13. Limitations and Future Improvements

a. Limitations:

- KNN is computationally expensive for large datasets.
- Requires storing the entire training dataset in memory.
- Highly sensitive to feature scaling.

b. Future Improvements:

- Apply dimensionality reduction techniques such as PCA.
- Use optimized search structures like KD Trees or Ball Trees.
- Test the method on a larger and more complex dataset.
- Compare KNN performance with CNN based deep learning models.

14. Conclusion

This assignment successfully implemented a KNN based image classification system using grayscale images with 5 fold cross validation. A detailed comparison between Manhattan (L1) and Euclidean (L2) distance metrics was conducted. The results clearly show that Euclidean distance achieves better overall performance for this dataset. The experiment provided strong practical understanding of instance based learning, distance metrics, cross validation techniques, and hyperparameter tuning.