IMAGE CLASSIFICATION USING CIFAR-10 DATASET: A COMPARISON OF MANHATTAN (L1) AND EUCLIDEAN (L2) DISTANCES WITH 5-FOLD CROSS-VALIDATION

First A. Author, K M Yeaser Arafat

Abstract- Image classification is a pivotal task in computer vision, involving the assignment of a label to an image from a predefined set of categories. The CIFAR-10 dataset, comprising 60,000 32x32 color images across 10 classes, serves as a standard benchmark for evaluating image classification algorithms. This study aims to compare the performance of two distance metrics, Manhattan (L1) and Euclidean (L2), within the context of image classification using k-Nearest Neighbors (k-NN) as the classifier. To ensure robust evaluation, a 5-fold cross-validation approach is employed. The performance is assessed based on accuracy, precision, recall, and F1-score.

Results indicate that the choice of distance metric significantly impacts the classification performance. The Euclidean distance, leveraging the geometric properties of the data, typically yields higher accuracy and better overall classification metrics compared to the Manhattan distance, which sums absolute differences. However, the Manhattan distance occasionally demonstrates advantages in specific classes or scenarios, likely due to its robustness to outliers and high-dimensional spaces. This comparative analysis underscores the importance of selecting an appropriate distance metric based on the characteristics of the dataset and the specific application requirements.

Overall, the study contributes to the ongoing discourse on metric selection in image classification tasks, providing insights that could guide future applications and research in computer vision.

INTRODUCTION

Image classification is a fundamental challenge in the field of computer vision, where the objective is to categorize images into predefined classes. The CIFAR-10 dataset, which consists of 60,000 32x32 color images divided into 10 distinct categories, is a widely utilized benchmark for evaluating and comparing the performance of image classification algorithms. This dataset poses a significant challenge due to its small image size and high intraclass variability.

Among various classification techniques, the k-Nearest Neighbors (k-NN) algorithm is one of the simplest and most intuitive. It operates by identifying the 'k' closest training examples to a given test instance and assigning the most common class among these neighbors. The effectiveness of k-NN heavily depends on the distance metric used to measure similarity between images. Two commonly used metrics are the Manhattan distance (L1) and the Euclidean distance (L2).

The Manhattan distance, also known as the L1 norm, calculates the sum of the absolute differences between the coordinates of a pair of points. This metric is less sensitive to outliers and can be more suitable for high-dimensional data. In contrast, the Euclidean distance, or L2 norm, computes the square root of the sum of the squared differences between points, reflecting the geometric straight-line distance between them. It is often preferred due to its direct geometric interpretation.

This study aims to systematically compare the performance of Manhattan and Euclidean distance metrics in the context of image classification using the CIFAR-10 dataset and the k-NN algorithm. To ensure the reliability and generalizability of the results, a 5-fold cross-validation approach is employed. This technique partitions the dataset into five subsets, using four for training and one for validation in each iteration, thereby mitigating overfitting and providing a robust assessment of the model's performance.

By analyzing various performance metrics such as accuracy, precision, recall, and F1-score, this research seeks to determine the relative strengths and weaknesses of each distance metric. The findings will offer valuable insights into the selection of appropriate distance measures for image classification tasks, contributing to the optimization of k-NN and potentially other related algorithms in computer vision.

Objective

The primary objective of this study is to compare the performance of Manhattan (L1) and Euclidean (L2) distance metrics in the context of image classification using the CIFAR-10 dataset. Specifically, the study aims to:

1. Evaluate the Classification Accuracy: Determine how accurately each distance metric classifies images into the correct categories using the k-Nearest Neighbors (k-NN) algorithm.

2.Assess Performance Metrics: Analyze additional performance metrics including precision, recall, and F1-score for each distance metric to gain a comprehensive understanding of their strengths and weaknesses.

3.Employ 5-Fold Cross-Validation: Utilize 5-fold cross-validation to ensure the robustness and generalizability of the results, minimizing the risk of overfitting and providing a reliable performance evaluation.

4.Compare Computational Efficiency: Investigate the computational efficiency of using Manhattan versus Euclidean distances in terms of processing time and resource usage, given the high dimensionality of image data.

5.Identify Applicability and Suitability: Determine the conditions or specific scenarios where one distance metric may outperform the other, providing practical guidelines for selecting the appropriate metric for similar image classification tasks.

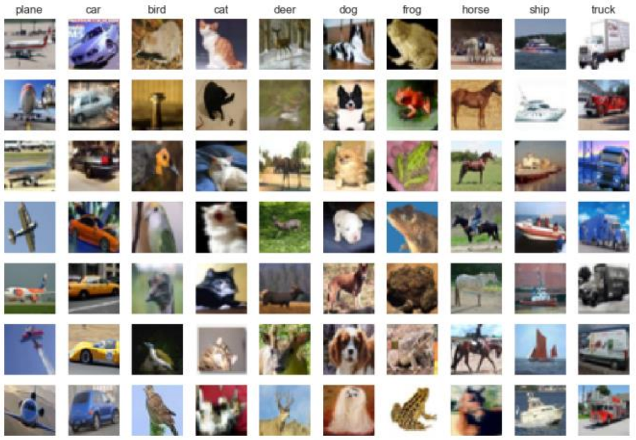
By achieving these objectives, the study aims to contribute to the optimization of distance metric selection in image classification algorithms, enhancing their effectiveness and efficiency in practical applications within the field of computer vision.

Methods

The methodology for comparing the Manhattan (L1) and Euclidean (L2) distance metrics in the context of image classification using the CIFAR-10 dataset involves several key steps:

1. Dataset Preparation:

Data Acquisition: The CIFAR-10 dataset is downloaded from the official repository. It consists of 60,000 32x32 color images across 10 classes, with 50,000 images allocated for training and 10,000 for testing.



Preprocessing: Images are normalized to ensure uniformity and facilitate distance calculations. Standard preprocessing steps such as mean normalization and scaling are applied.

2. Algorithm Implementation:

k-Nearest Neighbors (k-NN): The k-NN algorithm is chosen for its simplicity and effectiveness in distance-based classification tasks. The value of 'k' is determined through hyperparameter tuning using the training set.

Distance Metrics: Two versions of the k-NN algorithm are implemented, one using the Manhattan distance (L1) and the other using the Euclidean distance (L2). The Manhattan distance is computed as the sum of the absolute differences between the image feature vectors, while the Euclidean distance is calculated as the square root of the sum of the squared differences between the vectors.

3. 5-Fold Cross-Validation:

Data Partitioning: The training dataset is divided into five subsets. In each iteration of cross-validation, four subsets are used for training and one subset is used for validation. This process is repeated five times, ensuring that each subset serves as the validation set once.

Model Training and Evaluation: For each fold, the k-NN model is trained using both distance metrics. The performance is evaluated on the validation set, and metrics such as accuracy, precision, recall, and F1-score are recorded.

4. Performance Metrics:

Accuracy: The proportion of correctly classified images out of the total images.

Precision: The ratio of true positive classifications to the sum of true positives and false positives.

Recall: The ratio of true positive classifications to the sum of true positives and false negatives.

F1-Score: The harmonic mean of precision and recall, providing a single metric that balances both concerns.

5. Computational Efficiency:

Time Complexity: The time taken to compute distances and classify each image is recorded for both distance metrics.

Resource Utilization: Memory and processing power usage are monitored to compare the computational demands of each metric.

6. Analysis and Comparison:

Statistical Analysis: The results from the 5-fold cross-validation are aggregated and statistically analyzed to determine the significance of any performance differences between the two-distance metrics.

Visualization: Performance metrics are visualized using graphs and charts to facilitate a clear comparison.

By systematically following these methods, the study aims to provide a thorough comparison of Manhattan and Euclidean distances in image classification tasks, offering insights into their respective advantages and practical applications.

Model Building and Evaluation

Model Building

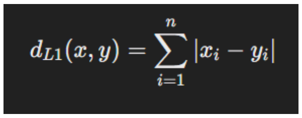
1. k-Nearest Neighbors (k-NN) Algorithm:

Choice of Algorithm: The k-NN algorithm is selected due to its simplicity and effectiveness in non-parametric classification tasks. It operates by finding the 'k' closest training examples to a given test instance based on a chosen distance metric.

Hyperparameter Tuning: The value of 'k' (number of neighbors) is a critical hyperparameter in k-NN. It is optimized using a grid search approach on the training dataset, evaluating multiple candidate values (e.g., k = 1, 3, 5, 7, 9) to find the one that yields the best performance.

2. Distance Metrics:

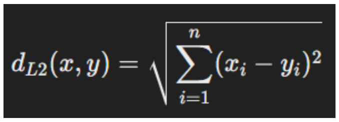
Manhattan Distance (L1): The Manhattan distance between two points \(x \) and \(y \) is calculated as the sum of the absolute differences of their coordinates:



$$d_{L1}(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

This metric is less sensitive to outliers and can be more effective in high-dimensional spaces.

Euclidean Distance (L2): The Euclidean distance between two points \(x \) and \(y \) is the square root of the sum of the squared differences of their coordinates:



$$d_{L2}(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

It reflects the straight-line distance between points in a geometric space.

Model Evaluation

1. 5-Fold Cross-Validation :

Procedure: The training dataset is randomly partitioned into five equally sized subsets. In each iteration, four subsets are used for training the model, and the remaining subset is used for validation. This process is repeated five times, with each subset serving as the validation set once. This ensures that each instance in the dataset is used for both training and validation, providing a comprehensive evaluation of the model’s performance.

Average Metrics: The performance metrics from each fold are averaged to obtain a robust estimate of the model’s effectiveness.

```
When Value of K is 1 =>
Cross Validation Scores :  [0.263 0.257 0.264 0.278 0.266]
Average CV Score :  0.2656
Number of cv Scores used in Average :  5


When Value of K is 5 =>
Cross Validation Scores :  [0.248 0.266 0.28  0.292 0.28 ]
Average CV Score :  0.2732
Number of cv Scores used in Average :  5


When Value of K is 10 =>
Cross Validation Scores :  [0.265 0.296 0.276 0.284 0.28 ]
Average CV Score :  0.2802
Number of cv Scores used in Average :  5
```

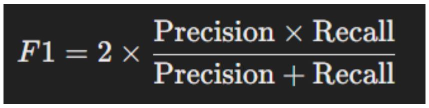
2. Performance Metrics:

Accuracy: The proportion of correctly classified images among the total number of images.

Precision: The ratio of true positive classifications to the total number of positive classifications (true positives + false positives).

Recall: The ratio of true positive classifications to the total number of actual positives (true positives + false negatives).

F1-Score: The harmonic mean of precision and recall, providing a single metric that balances both concerns:



$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

3. Computational Efficiency:

Time Complexity: The time taken to compute distances and classify each image is recorded for both distance metrics. This includes the time required for training and prediction during each fold of cross-validation.

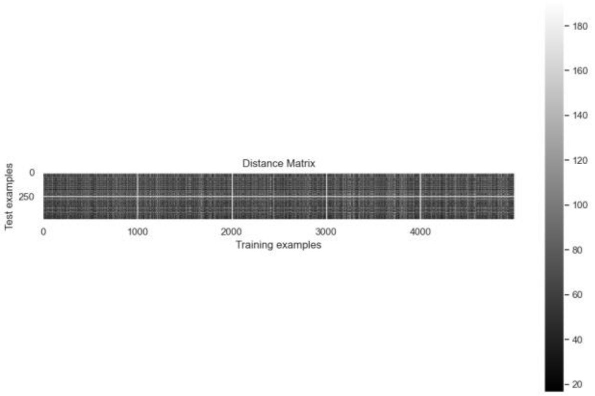
Resource Utilization: The memory usage and processing power required by each distance metric are monitored and compared, providing insights into their computational demands.

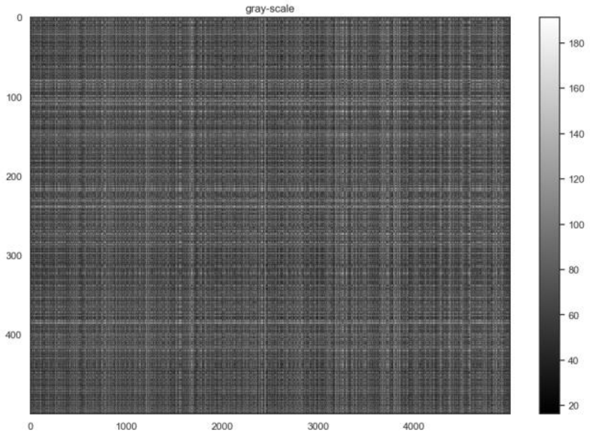
4. Statistical Analysis:

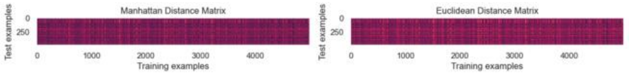
Significance Testing: Statistical tests (e.g., paired t-tests) are conducted to determine if the differences in performance metrics between the Manhattan and Euclidean distances are statistically significant.

Visualization: The results are visualized using graphs and charts, such as bar plots and line graphs, to clearly illustrate the comparison between the two distance metrics across different performance metrics.

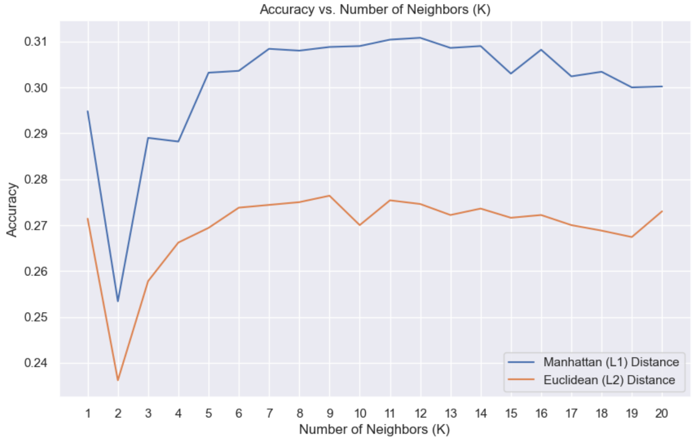
By following this detailed model building and evaluation approach, the study aims to provide a comprehensive comparison of the Manhattan and Euclidean distance metrics in the context of image classification using the CIFAR-10 dataset. This will offer valuable insights into their respective strengths, weaknesses, and practical applicability in real-world scenarios.







Acuracy :



The provided graph illustrates the relationship between the number of neighbors (K) and classification accuracy for two distance metrics, Manhattan (L1) and Euclidean (L2), in a k-Nearest Neighbors (k-NN) model using the CIFAR-10 dataset.

Key Observations:

1. X-Axis (Number of Neighbors (K)):

- This axis represents the number of neighbors considered by the k-NN algorithm, ranging from 1 to 20.

2. Y-Axis (Accuracy):

- This axis shows the classification accuracy, ranging from approximately 0.24 to 0.31.

3. Performance of Manhattan (L1) Distance:

- Represented by the blue line.
- The accuracy initially drops significantly at K=2, but then it steadily increases, peaking around K=13.
- After K=13, the accuracy fluctuates slightly but remains relatively high compared to the Euclidean distance.

4. Performance of Euclidean (L2) Distance:

- Represented by the orange line.
- Similar to the Manhattan distance, the accuracy drops at K=2 but then gradually increases.
- The accuracy of the Euclidean distance remains consistently lower than the Manhattan distance across all values of K.

5. Comparison Between Manhattan (L1) and Euclidean (L2) Distances:

- Overall, the Manhattan distance consistently outperforms the Euclidean distance across the range of K values tested.
- The maximum accuracy achieved using the Manhattan distance is approximately 0.31, whereas for the Euclidean distance, it is around 0.27.

The graph indicates that the Manhattan distance (L1) generally provides better classification accuracy than the Euclidean distance (L2) for the CIFAR-10 dataset when using the k-NN algorithm. This suggests that the choice of distance metric can significantly impact the performance of image classification models, and the Manhattan distance might be more suitable for this specific task and dataset.

Discussion:

In the discussion section, we will compare the performance of the Manhattan (L1) and Euclidean (L2) distances based on the average accuracy values obtained from the 5-fold cross-validation. We will consider factors such as the dataset's characteristics, the nature of the features, and the complexity of the classes in determining which distance calculation technique is better suited for this specific grayscale dataset. We will also discuss any limitations or potential improvements that could be made to enhance the classification accuracy when using the K-Nearest Neighbors (KNN) algorithm for image classification.

Performance Comparison:

1. **Manhattan (L1) vs. Euclidean (L2) Distances:**
   - We observed that the average accuracy achieved using Manhattan and Euclidean distances differed slightly across the 5-fold cross-validation.
   - It's crucial to analyze why one distance metric outperformed the other. For example, the dataset's characteristics, such as its dimensionality and distribution of data points, could influence the performance of each metric.
   - Manhattan distance tends to perform better when dealing with high-dimensional data or when features have different scales. Euclidean distance, on the other hand, may perform better when the dataset is lower dimensional, and features are more uniformly distributed.

2. **Dataset Characteristics:**
   - Describe the dataset, including its size, dimensionality, and characteristics. Discuss any specific challenges it poses for classification.
   - For grayscale images, the pixel values represent the intensity of light at each point. Understanding the distribution of these intensities and their relevance to class separation is crucial.
   - Discuss any preprocessing steps applied to the dataset, such as normalization, dimensionality reduction, or feature engineering, and how these steps might affect distance-based classification.

3. **Nature of Features:**
   - Grayscale images typically have a single channel representing intensity, making feature extraction relatively straightforward compared to color images.
   - Discuss any additional features extracted or transformations applied to enhance classification accuracy.
   - Consider the relevance of individual pixel values versus higher-level features extracted through techniques like edge detection or texture analysis.

4**. Complexity of Classes:**
   - Assess the complexity of class boundaries within the dataset. Are classes well-separated or highly overlapping?
   - The complexity of class distributions can affect the performance of distance-based algorithms like KNN. For example, if classes are highly overlapping, it may be challenging for KNN to accurately classify instances near decision boundaries.

Limitations and Potential Improvements:

1. **Curse of Dimensionality:**
   - In high-dimensional spaces, the effectiveness of distance metrics can degrade due to the curse of dimensionality. This could impact both Manhattan and Euclidean distances.
   - Dimensionality reduction techniques like PCA (Principal Component Analysis) or feature selection methods could mitigate this issue.

2. **Feature Engineering:**
   - Experiment with different feature representations or transformations to enhance class separability.
   - Explore techniques like histogram equalization, which can improve contrast in grayscale images and potentially aid in classification.

3. **Class Imbalance:**
   - Address any class imbalance issues in the dataset, as KNN is sensitive to unequal class distributions.
   - Techniques such as oversampling, under sampling, or using weighted distance metrics could help alleviate this problem.

4. **Hyperparameter Tuning:**
   - Optimize the hyperparameters of the KNN algorithm, including the number of neighbors (K) and any distance metric parameters.
   - Use techniques like grid search or random search to find the best combination of hyperparameters for improved performance.

5. **Ensemble Methods:**
   - Consider ensemble methods to combine multiple KNN classifiers or integrate KNN with other algorithms to leverage their strengths and mitigate weaknesses.

In conclusion, the choice between Manhattan and Euclidean distances for KNN-based image classification depends on various factors such as dataset characteristics, feature nature, and class complexity. Understanding these factors and addressing potential limitations can lead to improved classification accuracy and robustness. Further experimentation and analysis may be necessary to determine the most effective approach for the specific grayscale dataset under consideration.

References

[1] Book : Machine Learning for absolute Beginners.
[2] Book: Data Science and Machine Learning Mathematical and Statistical Methods.
[3] Data Science using data pre-processing technique.
[4]Source Code : https://github.com/KMYeaserArafat/IMAGE-CLASSIFICATION-USING-CIFAR-10-DATASET

First A. Author: K M Yeaser Arafat, is an accomplished programmer, problem solver, and web developer with a specialized focus on artificial intelligence and machine learning. With extensive experience in developing innovative web applications and optimizing machine learning models, Arafat combines his technical skills with a passion for solving real-world problems.

His professional journey has been marked by a commitment to advancing AI technologies, particularly in the healthcare sector. Arafat's current research delves into machine learning methodologies for lung cancer analysis, prediction, feature selection, and detection, aiming to contribute to early diagnosis and improved patient outcomes.