Multi class classification using KNN classifier on CIFAR10 dataset (Task 2)

[Group 4: 2101MC19, 2101MC41, 2101MC29]

August 17, 2024

1 Introduction

This report presents an analysis of the K-Nearest Neighbors (KNN) algorithm applied to CIFAR-10 dataset. We implement KNN using various distance metrics and evaluate their performance across different k values.

2 Methodology

2.1 Data Preparation

In this project, we utilized the CIFAR-10 dataset to develop a machine learning model. The dataset, consisting of 60,000 32x32 color images across 10 classes, was first partitioned into training (50,000 images) and testing (10,000 images) sets. The images were then flattened from their original 32x32x3 RGB format into 1D vectors of 3,072 features to prepare them for model input. This preprocessing step is crucial for enabling the model to effectively learn from the data.

2.2 KNN Implementation

We implemented KNN with five distance metrics:

- Euclidean Distance
- Manhattan Distance
- Minkowski Distance (with p=3)
- Cosine Similarity
- Hamming Distance

For each metric, we tested k values of 3, 4, 9, 20, and 47.

3 Results

3.1 Accuracy Comparison

Table 1 shows the accuracy for each distance metric and k value.

Table 1: Accuracy (%) for Different Distance Metrics and k Values

k Value	Euclidean	Manhattan	Minkowski	Cosine	Hamming
3	21.34	25.29	19.9	9.74	10.15
4	22.11	25.49	20.3	9.74	10.16
9	22.98	26.85	22.0	9.66	10.13
20	23.28	27.25	22.1	9.66	9.9
47	22.45	26.55	20.4	9.74	9.57

3.2 Graphical Representation

Figure 1 illustrates the relationship between k values and accuracy for each distance metric.

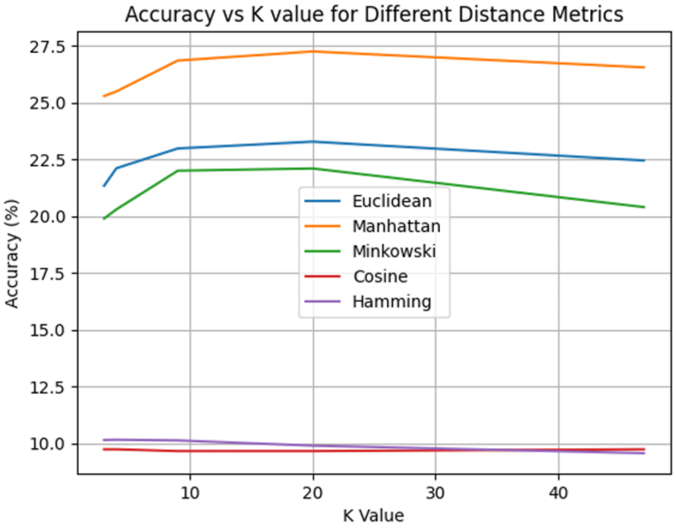

Figure 1: K Value vs Accuracy for Different Distance Metrics

3.3 Confusion Matrix and Performance Metrics

For the model using the Manhattan distance with $k = 20$, we computed the following confusion matrix:

Table 2: Confusion Matrix for Manhattan Distance, $k = 20$

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	560	5	123	14	71	3	16	7	197	4
1	349	137	138	54	103	14	65	11	99	30
2	228	7	371	42	230	20	56	7	37	2
3	195	10	207	156	192	48	118	27	41	6
4	139	9	318	43	383	6	56	21	25	0
5	133	21	231	138	195	108	107	23	35	9
6	200	5	291	81	175	17	204	10	16	1
7	184	21	204	77	242	17	43	144	49	19
8	196	32	78	28	85	10	32	5	526	8
9	289	63	132	118	60	22	38	13	129	136

Table 3: Precision and Recall for Each Class

Class	**Precision**	**Recall**
0	0.3545	0.5600
1	0.3808	0.1370
2	0.2846	0.3710
3	0.2337	0.1560
4	0.2445	0.3830
5	0.3023	0.1080
6	0.3018	0.2040
7	0.3818	0.1440
8	0.3472	0.5260
9	0.6495	0.1360

Performance metrics:

- Accuracy: 0.2725
- Average Precision: 0.3480
- Average Recall: 0.2723

4 Discussion

The results demonstrate that KNN performs poorly on the CIFAR-10 dataset across all distance metrics, with the highest accuracy being just 27.25% using Manhattan distance at $k = 20$. Even though Manhattan consistently outperforms the other metrics, none of the methods achieve acceptable accuracy for image classification tasks, indicating that KNN may not be well-suited for handling complex, high-dimensional image data like CIFAR-10.

Euclidean and Minkowski distances show similar performance to Manhattan but still fail to break 23.28% accuracy, highlighting the challenges of using distance-based metrics for such intricate datasets. The performance of Cosine similarity and Hamming distance is particularly poor, never exceeding 10.16%, suggesting that angular relationships between pixel vectors or binary comparisons offer little insight in distinguishing between complex image features.

This poor performance across all metrics and k values emphasizes that KNN, which relies heavily on distance measures, struggles with high-dimensional data like images, where feature representations are intricate and interdependent. For CIFAR-10, more sophisticated models, such as convolutional neural networks (CNNs), are typically required to capture the spatial hierarchies and textures that are essential for effective image classification.

5 Conclusion

The poor results of KNN on the CIFAR-10 dataset highlight the limitations of distance-based methods for image classification. Although Manhattan distance provides the best accuracy among the tested metrics, the overall performance remains far below what is expected for image recognition tasks. This indicates that KNN is not well-suited for handling the complexity of image data, where more advanced models like CNNs are typically required for meaningful classification accuracy.

Future work should focus on using more sophisticated techniques like deep learning architectures, which are better equipped to capture the spatial and hierarchical features present in image datasets like CIFAR-10. Exploring dimensionality reduction techniques or feature extraction methods specific to image data might also improve the performance of simpler models like KNN.