An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm

Kittipong Chomboon*, Pasapitch Chujai, Pongsakorn Teerarassamee, Kittisak Kerdprasop, Nittaya Kerdprasop

School of Computer Engineering, Institute of Engineering, Suranaree University of Technology, Nakhorn Ratchasima 3000, Thailand.

*Corresponding Author: chomboon.k@gmail.com

Abstract

This research aims at studying the performance of k-nearest neighbor classification when applying different distance measurements. In this work, we comparatively study 11 distance metrics including Euclidean, Standardized Euclidean, Mahalanobis, City block, Minkowski, Chebychev, Cosine, Correlation, Hamming, Jaccard, and Spearman. A series of experimentations has been performed on eight synthetic datasets with various kinds of distribution. The distance computations that provide highly accurate prediction consist of City block, Chebychev, Euclidean, Mahalanobis, Minkowski, and Standardize Euclidean techniques.

Keywords: Data Classification, Synthetic Data, Distance Metrics, k-Nearest Neighbors.

1. Introduction

Data mining is the extraction of knowledge hidden in the data. Data mining is often done with the large datasets. The knowledge from data mining has been used in various fields, such as prediction over future situation, assisting in medical diagnosis, forecasting relation of chronology.

Current data mining methodology has been classified into several tasks, such as classification, clustering, and association mining. Data mining each for task will have a different purpose. Classification task will be trying to classify data with high accuracy for classifying future example, such as trying to distinguish between patients with heart disease and those who are healthy. Clustering task will try to categorize groups of data such that data in the same group look similar, whereas they are dissimilar to others in different groups. Association mining task will try to find rules that represent relation between data with some support and confident values.

Classification task of data mining can be done with many algorithms such as k-nearest neighbor. Beyer[1] explained the significance and origin of the nearest neighbor. Cover[2] used k-nearest neighbor to classify data. Dudani[3] did research about weighting of distance matrix values with k-nearest neighbor. Fukunaga[4] developed techniques for running k-nearest neighbor faster. Keller[5] developed new algorithm named "Fuzzy K-Nearest Neighbor" based on k-nearest neighbor with the purpose to use it with fuzzy task. Köhn[6] used city-block distance matric to increase performance of k-nearest neighbor algorithm.

This research also studies classification technique with a specific interest in the k-nearest algorithm. We aim to analyze the performance of different distance metrics to finally choose a proper metric that makes a good classification performance. In this research use 8 synthetic datasets with different distribution, and a dataset for each distribution has 2 classes but has different amount of data in each class. This is to test the impact about amount in each class on the performance of classification.

The rest of this research is organized as follows: Section 2 gives details of the k-Nearest Neighbor and the computation of each distance metric. Section 3 gives details of our proposed method. The experimental results and analysis will be presented in Section 4. Finally, the research is concluded in Section 5.

DOI: 10.12792/iciae2015.051

2. Background

2.1 k-Nearest Neighbor

The k-nearest neighbor is a semi-supervised learning algorithm such that it requires training data and a predefined k value to find the k nearest data based on distance computation. If k data have different classes, the algorithm predicts class of the unknown data to be the same as the majority class. For example, to find the appropriate class of new datum using the k-nearest neighbor algorithm with a Euclidean distance metric, the concept can be shown in Fig. 1.

Fig. 1 shows the classification of iris data. The point to be classified is (5, 1.45), which is shown with "X". When applying k-nearest neighbor algorithm with k = 8 using Euclidean distance computation, the result is shown with a radius of dot line. It has two possible classes: virginica class with two instances and versicolor class with six instances. This algorithm will classify mark "X" to the class of versicolor because versicolor class is the majority of data within the radius.

2.2 Distance Metrics

Distance metrics are a method to find distance between a new data point and existing training dataset. In this research, we experiment with 11 distance metrics, which can be explained as follows.

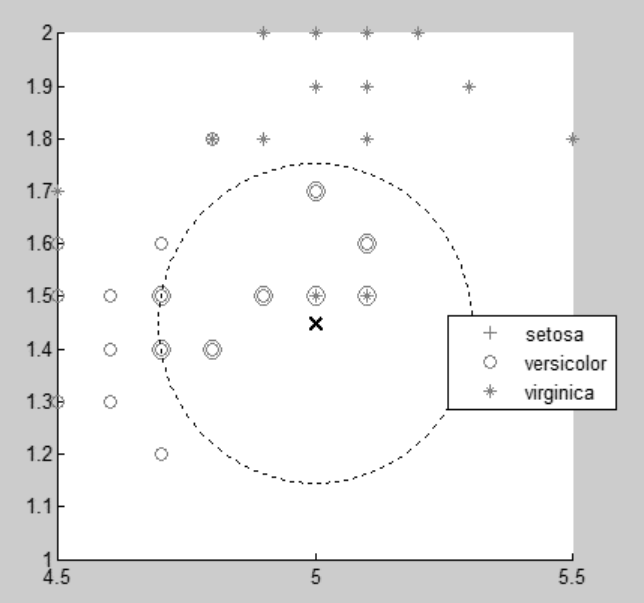


Fig. 1. The k-nearest neighbor prediction with k = 8.

Given an *mx*-by-*n* data matrix X, which is treated as *mx* (1-by-*n*) row vectors x_1, x_2, ..., xm_x, and *my*-by-*n* data matrix Y, which is treated as *my*(1-by-*n*) row vectors y_1, y_2,...,ym_y, the various distances between the vectors x_s and y_t are defined as follows:

1. Euclidean Distance

The Euclidean distance is a measure to find distance between two points, defined by Eq. (1)

$$d_{st}^2 = (x_s - y_t)(x_s - y_t)' \quad (1)$$

The Euclidean distance is a special case of the Minkowski metric, where p = 2.

2. Standardized Euclidean Distance

The standardized Euclidean distance is used to optimize the problem of finding the distance, defined by Eq. (2)

$$d_{st}^2 = (x_s - y_t)V^{-1}(x_s - y_t)' \quad (2)$$

where *V* is the *n*-by-*n* diagonal matrix whose *j*th diagonal element is $S(j)^2$, S is the vector containing the inverse weights.

3. Mahalanobis Distance

The Mahalanobis distance is a measure between a point and a distribution of data, defined by Eq. (3)

$$d_{st}^2 = (x_s - y_t)C^{-1}(x_s - y_t)' \quad (3)$$

where *C* is the covariance matrix.

4. City Block Distance

The city block distance between two points is the sum of the absolute difference of Cartesian coordinates, defined by Eq. (4)

$$d_{st} = \sum_{j=1}^{n} |x_{sj} - y_{tj}| \quad (4)$$

The city block distance is a special case of the Minkowski metric, where p = 1.

5. Minkowski Distance

The Minkowski distance is a method to find distance based on Euclidean space, defined by Eq. (5)

$$d_{st} = \sqrt[p]{\sum_{j=1}^{n} |x_{sj} - y_{tj}|^p} \quad (5)$$

For the special case of Minkowski distance

p = 1, the Minkowski metric gives the city block distance,

p = 2, the Minkowski metric gives the Euclidean distance, and

p = ∞, the Minkowski metric gives the Chebychev distance.

6. Chebychev Distance

The Chebychev distance is a measure to find distance between two vectors or points with standard coordinates, defined by Eq. (6)

$$d_{st} = max_j\{|x_{sj} - y_{tj}|\} \quad (6)$$

The Chebychev distance is a special case of the Minkowski metric, where p = ∞.

7. Cosine Distance

The Cosine distance is computed from one minus the cosine of the included angle between points, defined by Eq. (7)

$$d_{st} = \left(1 - \frac{x_s y'_t}{\sqrt{(x_s x'_s)(y_t y'_t)}}\right) \quad (7)$$

8. Correlation Distance

Distance based on correlation is a measure of statistical dependence between two vectors, defined by Eq. (8)

$$d_{st} = \left(1 - \frac{(x_s - \bar{x}_s)(y_t - \bar{y}_t)'}{\sqrt{(x_s - \bar{x}_s)(x_s - \bar{x}_s)'}\sqrt{(y_t - \bar{y}_t)(y_t - \bar{y}_t)'}}\right) \quad (8)$$

where

$$\bar{x}_s = \frac{1}{n}\sum_j x_{sj}$$

$$\bar{y}_t = \frac{1}{n}\sum_j y_{tj}$$

9. Hamming Distance

Hamming distance, which is the percentage of coordinates that differ, can be defined by Eq. (9)

$$d_{st} = \left(\frac{\#(x_{sj} \neq y_{tj})}{n}\right) \quad (9)$$

10. Jaccard Distance

Jaccard distance is computed from one minus the Jaccard coefficient, which is the percentage of nonzero coordinates that differ, defined by Eq. (10)

$$d_{st} = \left(\frac{\#\left[(x_{sj} \neq y_{tj}) \cap \left((x_{sj} \neq 0) \cup (y_{tj} \neq 0)\right)\right]}{\#\left[(x_{sj} \neq 0) \cup (y_{tj} \neq 0)\right]}\right) \quad (10)$$

11. Spearman Distance

Spearman distance is computed from one minus the sample Spearman's ranked correlation between observations, defined by Eq. (11)

$$d_{st} = 1 - \frac{(r_s - \bar{r}_s)(r_t - \bar{r}_t)'}{\sqrt{(r_s - \bar{r}_s)(r_s - \bar{r}_s)'}\sqrt{(r_t - \bar{r}_t)(r_t - \bar{r}_t)'}} \quad (11)$$

Where

r_{sj} is the rank of x_{sj} taken over x_1j, x_2j, ...xmx,j.

r_{tj} is the rank of y_{tj} taken over y_1j, y_2j, ...ymy,j.

r_s and r_t are the coordinate-wise rank vectors of x_s and y_t,

i.e., $r_s = (r_{s1}, r_{s2}, \ldots r_{sn})$ and $r_t = (r_{t1}, r_{t2}, \ldots r_{tn})$.

$$\bar{r}_s = \frac{1}{n}\sum_j r_{sj} = \frac{(n+1)}{2}$$

$$\bar{r}_t = \frac{1}{n}\sum_j r_{tj} = \frac{(n+1)}{2}$$

3. Empirical Study Methodology

In this section, we present our study framework using k-nearest neighbor algorithm with various distance metrics. The framework is shown in Fig. 2.

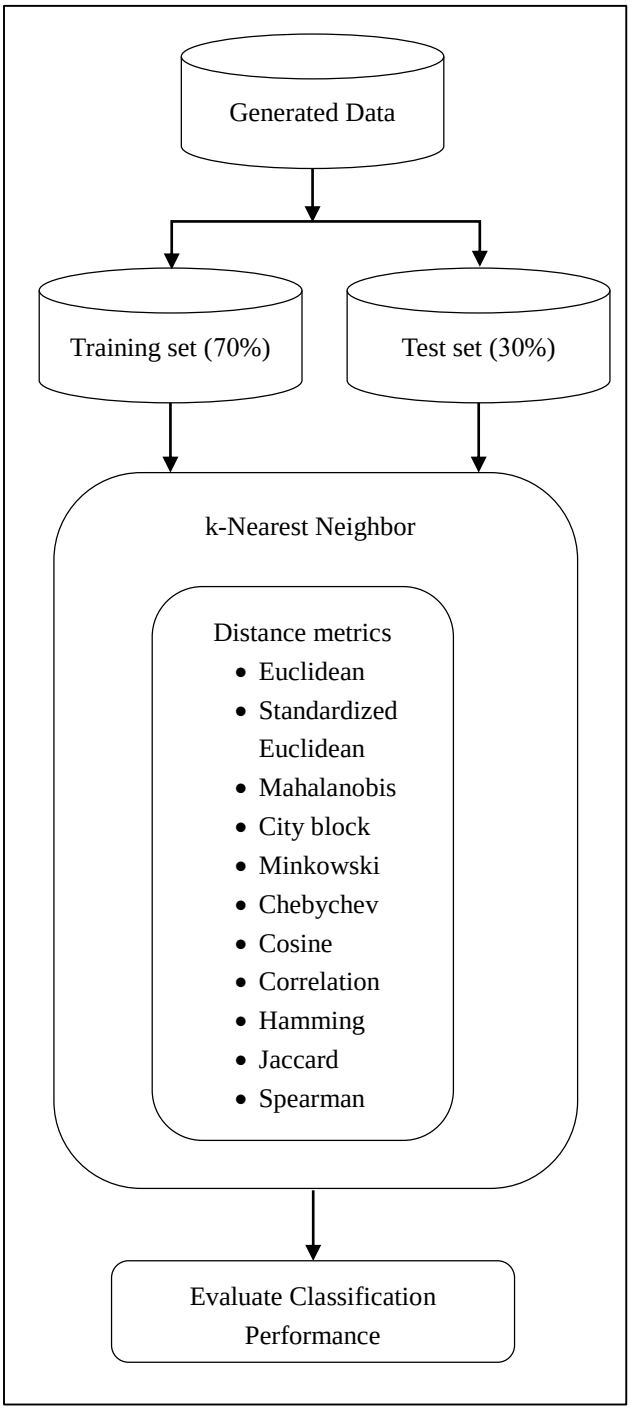


Fig. 2. The framework of our empirical study.

From Fig. 2 the detail of each step can be explained as follows:

Step 1: Generate binary data set with different distribution and different amount of data in each class. Then split data around 70% for training set and 30% for test set, which will be used for testing the performance of classification.

Step 2: Use data from step 1 for data classification by applying the k-nearest neighbor algorithm with various distance metrics to compute the k-nearest data points for making classification.

Step 3: Analyze the results and conclude about the performance of classification using various distance metrics.

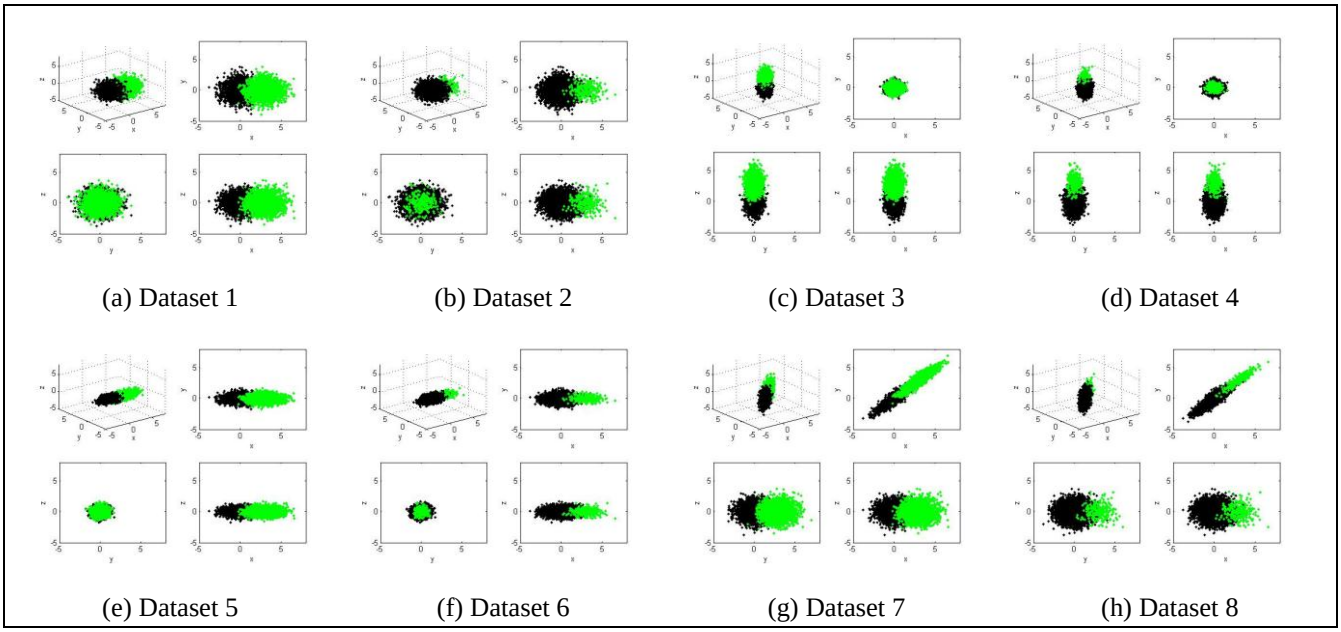
4. Experimental Results

4.1 Datasets

For our experiment, the proposed framework has been applied for classifying binary synthetic datasets. We generate eight synthetic datasets, each dataset has four different distributions, and each distribution has two of data in which class the amount of data in each class is varied. Each dataset has in total 5000 instances, and three features. We use MATLAB program to generate synthetic datasets. Details of the synthetic datasets are given in Table 1. Fig. 3 illustrates an overview of synthetic datasets.

Table 1. Details of synthetic datasets.

Dataset	Mean	SD	Class 1	Class 2	Total
1	[0 0 0; 3 0 0]	[1 0 0; 0 1 0; 0 0 1]	2500	2500	5000
2	[0 0 0; 3 0 0]	[1 0 0; 0 1 0; 0 0 1]	4750	250	5000
3	[0 0 0; 0 0 3]	[0.2 0 0; 0 0.2 0; 0 0 1]	2500	2500	5000
4	[0 0 0; 0 0 3]	[0.2 0 0; 0 0.2 0; 0 0 1]	4750	250	5000
5	[0 0 0; 3 0 0]	[1 0 0; 0 0.2 0; 0 0 0.2]	2500	2500	5000
6	[0 0 0; 3 0 0]	[1 0 0; 0 0.2 0; 0 0 0.2]	4750	250	5000
7	[0 0 0; 3 3 0]	[1 0.9 0; 0.9 1 0; 0 0 1]	2500	2500	5000
8	[0 0 0; 3 3 0]	[1 0.9 0; 0.9 1 0; 0 0 1]	4750	250	5000



(a) Dataset 1 (b) Dataset 2 (c) Dataset 3 (d) Dataset 4

(e) Dataset 5 (f) Dataset 6 (g) Dataset 7 (h) Dataset 8

Fig. 3. Distribution of the eight synthetic datasets, each one has four kinds of distribution.

4.2 Experimental Results

The results from the proposed study framework for eight synthetic datasets have been shown in Figs. 4 and 5. The data classification has been performed with the same algorithm (that is, k-Nearest Neighbor) and the same parameter setting. The only varied factor is a distance measurement. It turns out that the Hamming and Jaccard distance metrics perform badly on 4 out of 8 datasets.

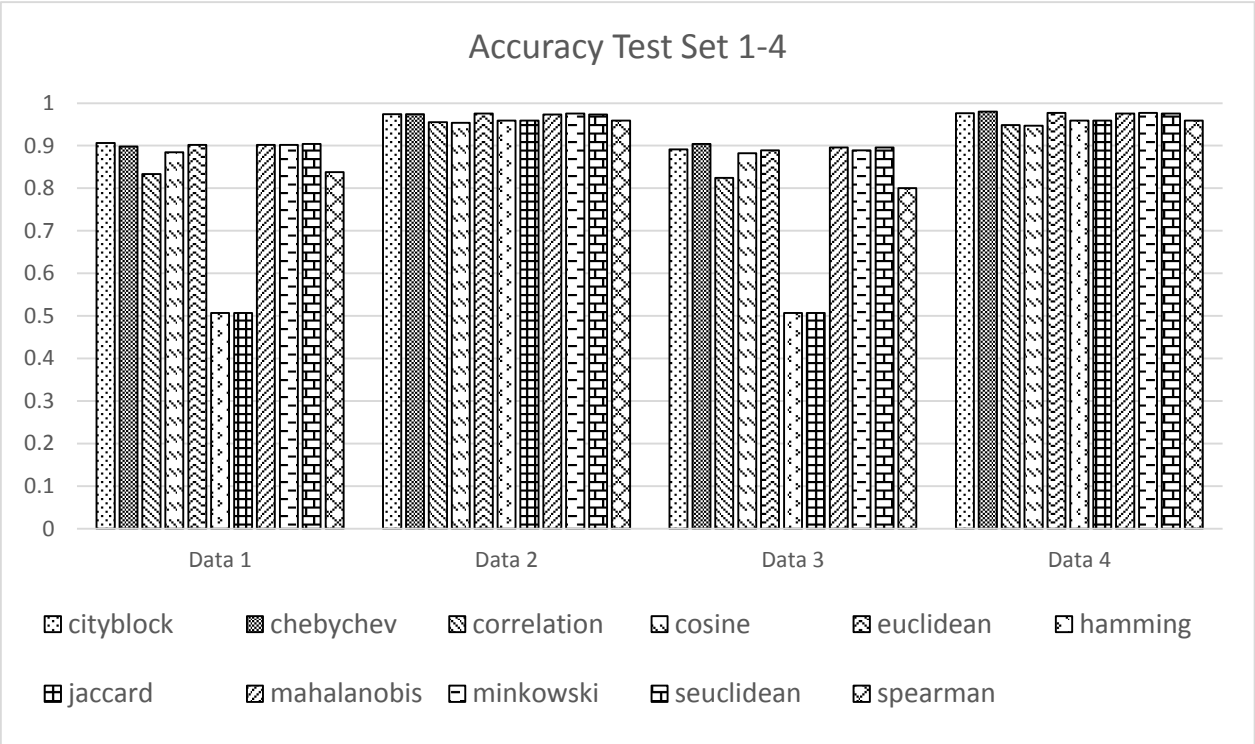


Fig. 4. Accuracy of synthetic datasets from no. 1 to no. 4.

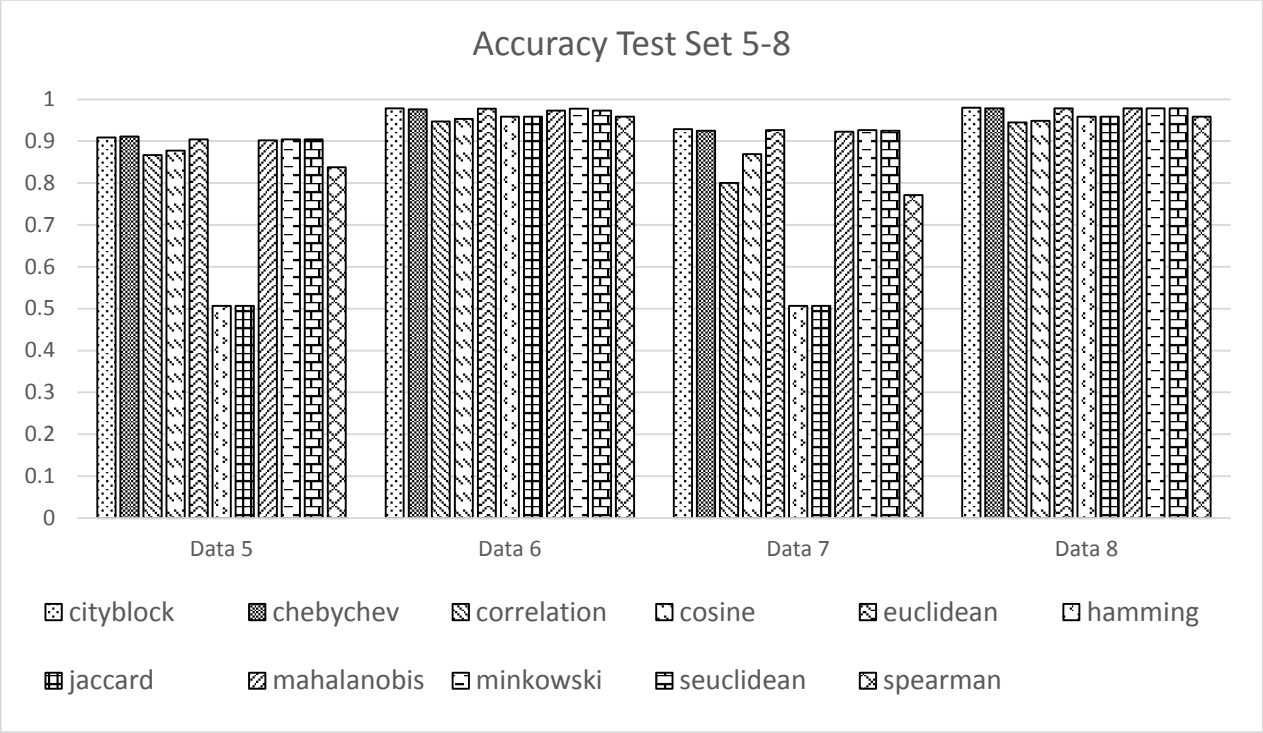


Fig. 5. Accuracy of synthetic datasets from no. 5 to no. 8.

5. Conclusions

The results of this research showed accuracy of k-nearest neighbor classification algorithm with different distance metrics. Experiments had been performed on eight synthetic datasets generated by MATLAB. The synthetic datasets have four distributions and have been split 70% to training set and 30% to test set. The results of classification over datasets in which amount of data in each class is equal showed that the Hamming and Jaccard techniques are low accuracy, while the other distance computation techniques have similar accuracy. The synthetic datasets in which amount of data in each class is different such as dataset 2, 4, 6 and 8 showed that the Hamming and Jaccard techniques are increasing in their classification accuracy. We can conclude that Hamming and Jaccard techniques are affected by the ratio of members in each class, while the other techniques are not affected by such phenomenon. The highest accuracy on classify data with k-Nearest Neighbor is obtained from the six distance metrics, that are City-block, Chebychev, Euclidean, Mahalanobis, Minkowski, and Standardized Euclidean techniques.

References

(1) Beyer Kevin, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft, When is “nearest neighbor” meaningful?, in Database Theory—ICDT’99. 1999, Springer. p. 217-235.aaaa

(2) Cover Thomas and Peter Hart: "Nearest neighbor pattern classification", Information Theory, IEEE Transactions on, Vol. 13, No. 1, pp. 21-27, 1967

(3) Dudani Sahibsingh A: "The distance-weighted k-nearest-neighbor rule", Systems, Man and Cybernetics, IEEE Transactions on, Vol. No. 4, pp. 325-327, 1976

(4) Fukunaga Keinosuke and Patrenahalli M Narendra: "A branch and bound algorithm for computing k-nearest neighbors", Computers, IEEE Transactions on, Vol. 100, No. 7, pp. 750-753, 1975

(5) Keller James M, Michael R Gray, and James A Givens: "A fuzzy k-nearest neighbor algorithm", Systems, Man and Cybernetics, IEEE Transactions on, Vol. No. 4, pp. 580-585, 1985

(6) Köhn Hans-Friedrich: "Combinatorial individual differences scaling within the city-block metric", Computational Statistics & Data Analysis, Vol. 51, No. 2, pp. 931-946, 2006