Selahaddin Batuhan AKBEN/ IU-JEEE Vol. 16(1), (2016), 2021-2026

A NEW METHOD FOR SELECTION OF NEIGHBORHOOD PARAMETER IN DISTANCE – WEIGHTED K-NEAREST NEIGHBORS CLASSIFFIER (DWKNN): CIRCULAR ATTRIBUTE NEIGHBORS

Selahaddin Batuhan AKBEN

Osmaniye Korkut Ata University, Bahce Vocational School, Bahce/Osmaniye, Turkey
batuhanakben@osmaniye.edu.tr

Abstract: *This study proposes a new neighborhood parameter selection method for distance-weighted k-Nearest Neighbors (DWKNN) classification. According to this method, individual circular neighborhood boundaries are formed as per class. Then, these circular boundaries are respectively positioned such that their centers become a test element. Membership of the test elements within classes is determined by the elements pertaining to the class which stays within the circle and constitute solely that circle. The proposed method is originally the state of circular attribute of the distance, or DWKNN. The proposed solution applies the circular attribute contribution approach in the issue of neighborhood boundary selection in the DWKNN method. Because the circles constituting the neighborhood boundary in the proposed method are determined of a given class structure for the first time, and since the circles were designated according to the classification nature, the goal is maximum performance of the proposed model. The method proposed in the study has been tested on real datasets, and these tests show that the proposed method has contributed approximately 2% to the success of the DWKNN method.*

Keywords: *K-Nearest Neighbor; Weighted Voting; Pattern Classification*

1. Introduction

Due to advantages such as simplicity and efficiency, the k-Nearest Neighbors algorithm (KNN) is a frequently used and well-known classification method (1, 2, 3, 4, 5). Moreover, it only requires one parameter, namely k, the neighborhood parameter (6). The KNN algorithm has subsequently been improved upon to create the distance-weighted KNN (DWKNN), which enhanced its performance (7). Nevertheless, one crucial problem in both the KNN and DWKNN methods is the selection of a suitable number of neighbors, as the identification of different classes might create different numbers of neighbor. If the chosen k parameter is too small, it leads to poor classification outcomes in discrete and noisy datasets. However, if the chosen k parameter is too big, the classification outcomes might also suffer due to the presence of outliers (8, 9, 10, 11).

In the literature, many methods have been developed for parameter selection (12, 13, 14, 15, 16). However, these methods are not often used in actual practice. As these methods entail sophisticated mathematical operations, they minimize the simplicity aspect of the KNN algorithm. For this reason, the k-fold cross validation technique is the best known method for parameter selection (17, 18, 19). However, the biggest disadvantage of this technique is its high processing time. Therefore, a new parameter selection method with short processing time is needed.

Current methods propose using the same neighborhood parameters for all classes. However, the structure and properties of each class might be different. For this reason, the neighborhood limit for each class should be distinctively designated and the class membership of the test elements should be evaluated according to this defined neighborhood limit. The neighborhood limit as per class should also be created so as to produce the best results, according to its particular structure. The membership of the test element as per class to that class should then be evaluated according to the distance of test elements that lie within the neighborhood boundary. Since the method proposed in the study based on circular features as per class, it is called Circular Attribute Neighbors (CAN).

In this study, the superiority of the proposed method over the current method is explained both experimentally and mathematically. A flowchart of this study can be seen in Figure 1.

Received on: 26.10.2015
Accepted on: 26.12.2015

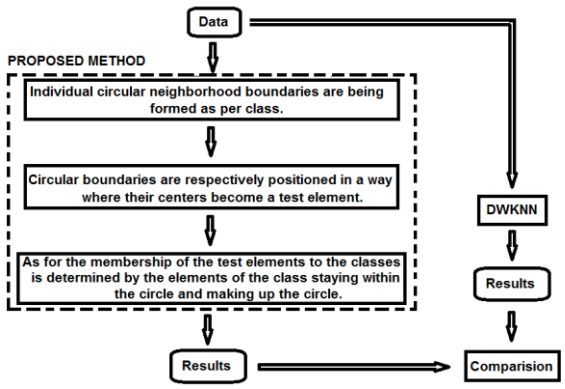

Figure 1. Study flowchart

2. Materials and Methods

In the study, four actual datasets (i.e., Wine, Seeds, Balance-Scale and Parkinson's Tele-monitoring datasets) were employed. These datasets were taken from the UCI machine learning repository (20). The properties of these datasets are shown in Table 1. Both CAN and DWKNN evaluate neighbors with the same logic, thus the performance of the proposed CAN method is compared with the results of each dataset when DWKNN is applied. To obtain a reliable classification performance, the classifiers were assessed with a 10-fold cross-validation technique. For every cycle, 90% of the dataset was used as training and the remaining 10% was used as the test element. The computer calculations were run with the MATLAB program.

Table 1. Features of UCI Datasets Used.

Dataset	Number of Instances	Number of Features	Classes	Mean Distances
Wine	178	13	3	4.55
Seeds	210	7	3	4.92
Balance scale	625	4	3	3.89
Parkinson's Tele-monitoring	5875	25	42	4.68

Calculation of mean distance can be seen in section 2.2

2.1. Technical background

A class element has a relationship with all other elements belonging to the same class. However, the degree of this relationship is not the same for all elements. For example, the relationship between a given class element and the elements closest to itself is strong, yet the relationship between this same class element and the element farthest from it is weak. Thus, remote elements within the same class of elements hold less meaning for a given element. Using this logic, meaningful neighbors of a class element should be designated according to their distance from that given class element.

In statistics, the term "mean value" is used to represent all values; in CAN, the relationship of an element to another element that belongs to the same class can also be calculated via mean value. Namely, the mean distance of an element from another element within the same class can be used significant neighbors within the same class. Thus elements that are closer to a given class element are the meaningful neighbors of this class element.

So if the mean inter-class element distance is calculated, the value found can be used to determine the meaningful neighborhood limit for all class elements. Namely, the meaningful elements are those elements that are closer to a test class element than the mean interclass element distance. Similarly, these elements closer to a test element than the mean inter-class elements distance are the meaningful elements (i.e., meaningful neighbors) of that class.

2.2. Mathematical Notation

Let us suppose that there is a class formed by elements n_1, n_2, n_3, and n_4 (Figure 2).

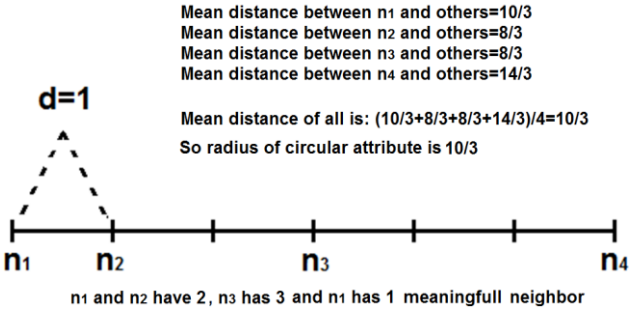

Figure 2. Determination of meaningful neighbors of a given class element, according to the mean inter-class elements distance

In this class, some terms can be shown in equations below:

If the data set N is in equation (1),

$$N = \left(n_1, n_{2,} \dots \dots \dots n_x\right) \tag{1}$$

The mean distance between each element is in equation (2)

$$D_x = \frac{1}{x-1} \sum_{m=1}^{x} |n_x - n_m| \tag{2}$$

Then, the mean distance of all (radius) is in equation (3).

$$A = \frac{1}{x} \sum_{m=1}^{x} D_m \tag{3}$$

Finally, the meaningful element can be determined in equation (4),

$$M_{x,m} = \begin{bmatrix} |n_x - n_m| \le A\,, & Meaningful \\ |n_x - n_m| > A\,, & Meaningless \end{bmatrix} \qquad (4)$$

While $M_{x,m}$ is the meaningfulness of n_m for n_x.

According to the equations above, the mean inter-class elements distance is 10/3. If we draw a circle 10/3 radius over a test element and centered the resulting circle over this test element, the number of other class members that remain inside the circle would represent the number of meaning neighbors for the given test element (i.e., the element at the center of the circle). The number of meaningful neighbors for the elements in Figure 2 is shown in Table 2.

Table 2. Meaningful Number of Neighbors For Each Element in the Sample Dataset Shown in Figure 2.

Class Elements	Meaningful Neighbor Number (number of neighbors closer than 10/3 distance)
n_1	2
n_2	3
n_3	3
n_4	1

The membership of a test element of a given class can be discovered in the same way. The meaningful neighbors are those elements whose distance from the test element is closer than mean inter-class elements distance value of 10/3. As seen in Table 2, whereas an element near the class center has many meaningful neighbors, the number of meaningful neighbors for the outermost elements is far fewer. According to the same computation method, whereas the neighbor element number of the test elements closer to the class center is higher, the neighbor element number of the test elements closer to the outermost element is lower. An illustration of the determination of the neighbors can be seen in Figure 3.

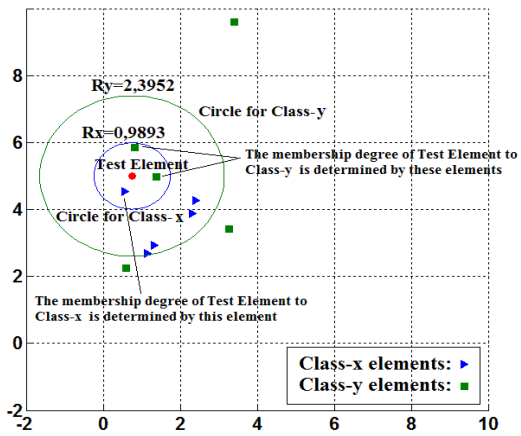

Figure 3. Visual representation of neighbor determination

2.3. Algorithm

The algorithm of the proposed method is as follows:
Let us assume that there are two classes

$A = \{A_1, A_{2,\cdots}, A_{n,}\}$ and $B = \{B_1, B_{2,\cdots}, B_{m,}\}$, and that X is a test element.

At the same time, rA, is the mean inter-elements distance of class A elements, and r_B is the mean inter-elements distance of class B elements.

Thus, if a Circle A is formed using radius r_A and a Circle B is formed using radius r_B:

1. The center of Circle A is positioned over X, as X is the test element.
2. The class A elements that remain inside Circle A constitute the neighbors of X that belong to Class – A.
3. The membership degree of X to Class A, according to the distance between X and X's neighbors within Class A, is fixed at M1.
4. The center of Circle B is positioned over X, as X is the test element.
5. The class B elements that remain inside Circle B constitute the neighbors of X that belong to Class B.
6. The membership degree of X to Class B, according to the distance between X and X's neighbors within Class – B and X, is fixed at M2.
7. If M1>M2 then X belongs to Class A, but if M1<M2 then X belongs to Class B.

Thus, a distinctive neighborhood boundary is generated in the method as per class and these limits are calculated according to the structure of the inter-class elements distance.

3. Results and Discussion

Both the CAN method proposed in this study and the DWKNN method both use the same neighbor assessment algorithms, as the neighbors in both methods are weighted according to distance. Thus the proposed CAN method will be compared to the DWKNN method by choosing the k – parameters that produce the best results for the DWKNN method. However, classic (individual) KNN also used as a classifier to datasets for preferable comparision. During these tests, the standard deviations formed due to application of cross validation method have been shown with the sign of "±". The success rates of the methods are as shown in Table – 3.

Table 3. Accuracy of the CAN and DWKNN Methods.

Data set	KNN	DWKNN	CAN
Wine	73.39±0.18 (1)	76.69±0.11 (1)	78.38±0.12
Seeds	87.73±0.23 (13)	90.76±0.56 (11)	92.85±0.59
Balance scale	84.75±0.31 (9)	87.95±0.47 (11)	89.48±0.81
Parkinson's Tele-monitoring	93.77±0.22 (1)	96.33±0.16 (1)	98.51±0.14

The values in the parentheses indicate the k values through which the best results were acquired. The standard deviations are shown with the ± values.

As seen in Table-3, success rates of the classic (individual) KNN method were very low to proposed method. KNN method has therefore not been taken into account in the detailed assessment (It has been excluded from detailed evaluation). Furthermore, classifiers were tested separately for each metric distances (Euclidean, Mahalanobis, City block, Minkowski, Chebyshev, Correlation, Hamming, Jaccard and Spearman). However, only the Euclidean Distance results are in Table 3, since the best results obtained in euclidean distance.

The accuracy rate of the CAN method is approximately 2% higher than the maximum performance of DWKNN (i.e., DWKNN results obtained by selection of best suitable k parameter; Table 3). Comparison of CAN and DWKNN methods for individual datasets seen in Figures 4 through 7.

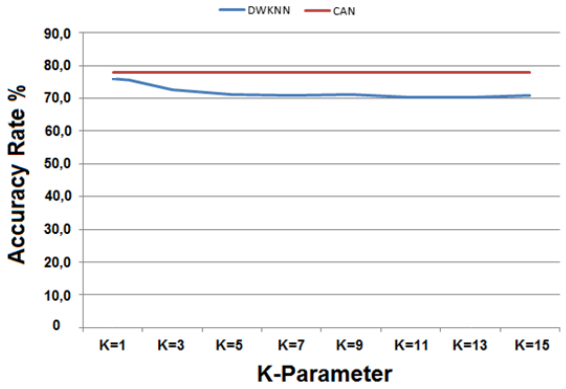

Figure 4. Success rates of the DWKNN and CAN methods for the Wine dataset.

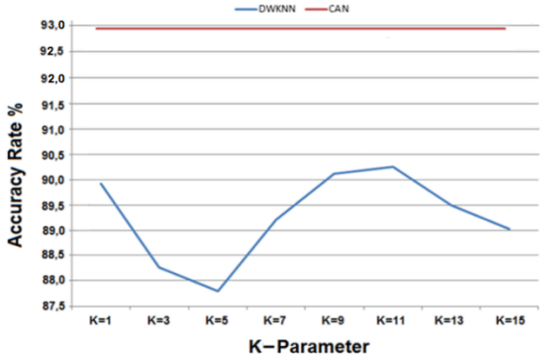

Figure. 5. Success rates of the DWKNN and CAN methods for the Seed dataset.

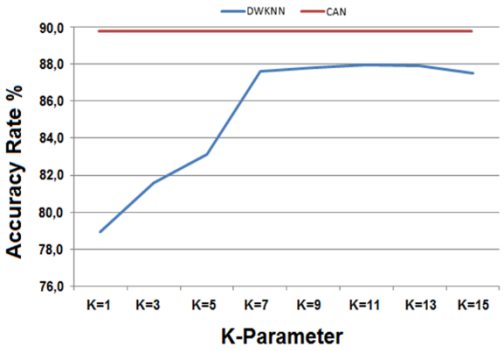

Figure 6. Success rates of the DWKNN and CAN methods for the Balance scale dataset.

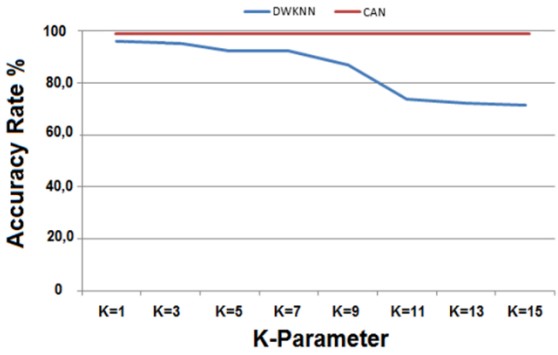

Figure 7. Success rates of the DWKNN and CAN methods for the Parkinson's Tele-monitoring dataset.

Also these figures can seen in Table 4.

Table 4. Accuracy Comparision of the CAN and DWKNN Methods for each k parameter.

	K Parameter	Wine	Seeds	Balance scale	Parkinson's Tele-monitoring
DWKNN	1	**76.69**	89.01	78.93	**96.33**
	3	73.32	88.32	81.81	95.21
	5	69.92	87.85	82.87	93.08
	7	69.87	89.23	87.26	93.03
	9	70.72	90.30	87.34	84.11
	11	70.02	**90.76**	**87.95**	73.74
	13	69.98	89.39	87.45	72.63
	15	70.02	89.12	87.08	71.34
CAN	Radius	78.38	92.85	89.48	98.51

Optimum K Values are Darkened

Figures 4 through 7 show that as different neighborhood parameters (i.e., k-parameters) for the DWKNN method are selected, the CAN method becomes 2% more successful than DWKNN method. Furthermore, since there is no parameter selection in the CAN method, the success remains fixed and

this rate is approximately 2% better than the highest performance obtained by the DWKNN method.

In addition, it can be seen that the success rates are inversely proportional to mean distances in Table 2. So,it can be said that the proposed method is more successful at the data sets having the low mean distance between the elements.

The method proposed was assessed by the following process. There are two stages of the DWKNN method, namely 1) parameter selection, and 2) making the test element a member. The difference between DWKNN and CAN methods lies in this initial parameter selection stage. In the CAN method, a separate neighborhood limit is created for each class, and it is when these limits are determined that the inter–class components distance is considered. Therefore, as the quantity and the number of classes increases, the CAN method's processing time also increases (i.e., in the parameter selection stage). In the DWKNN method, during first stage, either the parameter is estimated or the parameter selection algorithms are applied. If the parameter selection is determined through estimation, because it may take time to identify the accurate parameter, the parameter selection stage of the DWKNN method may be longer than that of the CAN method. Similarly, if the parameter selection algorithm is employed, because application of the parameter selection algorithm may also take time, the parameter selection algorithm of DWKNN method might again be longer than CAN method.

Moreover, if the suitable parameter is bigger, DWKNN's second stage (i.e., making the test element a member) will last longer, and DWKNN will need longer processing times for smaller datasets. Furthermore, it is unknown whether DKWKNN's parameter selection time will be faster or slow for bigger and multi-class datasets t. Processing times for the datasets used were also compared (Table 5).

Table 5. Processing Time for the DWKNN and CAN Methods.

Data set	DWKNN (As per parameter)	CAN
Wine	0,89	0,95
Seeds	0,86	0,91
Balance-scale	1,11	1,35
Parkinson's Tele-monitoring	3,2	5,8

Values are in seconds

The processing time of the CAN method is close to the processing time of the DWKNN method under ideal circumstances(Table 5). Namely, when the ideal parameter is selected for the DWKNN method, its processing time is roughly equivalent to the processing time of the CAN method, which means that unless lucky parameter estimations are done, the processing period of the DWKNN method will be longer than CAN method. If a suitable parameter selection shall be materialized by parameter selection algorithm, it is unavoidable that DWKNN method would be slower due to parameter selection algorithm. Thus, we conclude that the CAN method's processing time is superior to DWKNN. Finally,, it should be noted that the CAN method processing time for the Parkinson's Tele-monitoring dataset increased because that dataset had more the class numbers and element numbers than the other datasets.

4. Conclusions and Proposals

In this study, a new k-neighborhood parameter selection method, CAN, is proposed for the DWKNN method. The most important difference between the proposed method and the classical neighborhood parameter designation is the selection of ideal neighborhood limits by class individually. The ideal neighborhood limit associated with each class is determined according to dataset dissemination, thereby maximizing the performance of the method proposed and ultimately improving the performance of DWKNN. In addition, the proposed method does not require trial of parameter selection, and its performance is fixed so as to produce the best ideal performance.

Moreover, since the neighborhood limit is only designated once, the proposed method does not lengthen the processing period; rather, the processing time is often shortened as the neighborhood parameter is not subject to a time-consuming trial and error period.

The biggest disadvantage of the proposed method is the presence of local density zones in the classes. In such cases, some test elements might be affected by the local density of the classes. However, this disadvantage is not found in KNN methods.

Another disadvantage of the proposed method is that its processing time is a little bit long, as the neighborhood limit creation operation depends on the mean inter-elements distance. Nevertheless, when compared to the trial and error or parameter selection algorithms used in the DWKNN method during parameter selection, in general this proposed method is be expected to be faster than the DWKNN method.

In conclusion, the proposed method is suitable for use with actual datasets, as it augments the performance by selecting appropriate parameters that yield the highest performance without entailing parameter estimation.

5. References

[1] X. Wu, V. Kumar, JR. Quinlan, J. Ghosh, Q. Yang, H. Motoda, et al., "Top 10 algorithms in data mining", *Know. Inf. Syst.* Vol. *14*, pp. *1:37*, 2008.

2. M. Zbancios, SM. Feraru, "The Analysis of the FCM and WKNN Algorithms Performance for the Emotional Corpus SROL". *Advances in Electrical and Electronics Engineering*, vol. *12*, no. *3*, pp. *33-38*, 2012

3. X. Xu, C. Huang, C. Wu, Q. Wang, L. Zhao, "Graph Learning Based Speaker Independent Speech Emotion Recognition", *Advances in Electrical and Electronics Engineering*, vol. *14*, no. *2*, pp. *17-22*, 2014.

4. CT. Chang, JZC. Lai, MD. Jeng, "Codebook Generation Using Partition and Agglomerative Clustering", *Advances in Electrical and Electronics Engineering*, vol. *11*, no. *3*, pp. *91-98*, 2011.

5. V. Velican, R. Strungaru, O. Giorge, "Automatic Recognition of Improperly Pronounced Initial 'r' Consonant in Romanian",

Advances in Electrical and Electronics Engineering, vol. *12*. no. *3*, pp. *79-84*, 2012.
6. E. Fix, JL. Hodges, "Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties", *Randolf Field (TX): US Air Force School of Aviation Medicine*, Technique Report no. *4*, pp. *238-247*, 1951.
7. SA: Dudani, "The Distance-weighted k-Nearest Neighbor Rule", *IEEE Trans. Syst. Man Cybern*, vol. *6*, no. *4*. pp. *325-327*, 1976.
8. I. Guyon, A. Elisseeff, "An introduction to variable and feature selection", *Journal of Machine Learning Research*, vol. *3*, pp. *1157-1182*, 2003.
9. N. Morariu, S. Vlad, "Using Pattern Classification and Recognition Techniques for Diagnostic and Prediction". *Advances in Electrical and Electronics Engineering*, vol. *7*, no. *14*, pp. *63-67*, 2007.
10. G. Martinovic, D. Bajer, "Elitist Ant System with 2-opt Local Search for the Traveling Salesman Problem", *Advances in Electrical and Electronics Engineering*, vol. *12*, no. *1*, pp. *26-32*, 2012.
11. IV. Bornoiu, O. Grigore, "Kohonen Neural Network Stress Detection Using Only Electrodermal Activity Features", *Advances in Electrical and Electronics Engineering*, vol. 14. no. 3, pp. 71-78, 2014.
12. Z. Xie, W. Hsu, Z. Liu, M. Lee, "SNNB: A Selective Neighborhood Based Naive Bayes for Lazy Learning", Proceedings of the Sixth Pacific-Asia Conference on KDD, 2002.
13. E. Frank, M. Hall, B. Pfahringer, "Locally Weighted Naive Bayes", Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence, 2003, pp. 249-256.
14. ZB. Ozger, "KNN parameter selection via meta learning", 21 st. Signal Processing and Communications Applications Conference (SIU), 2013, pp.1-4.
15. BL. Pellom, R. Sarikaya, JHL. Hansen, "Fast likelihood computation techniques in nearest-neighbor based search for continuous speech recognition", *IEEE Signal Processing Letters*, vol. *8*, no. *8*, pp. *221-224*, 2001.
16. D. Stowell, MD. Plumbley, "Fast multidimensional entropy estimation by k-d partitioning", *IEEE Signal Processing Letters*, vol. *16*, pp. *537-540*, 2009.
17. I. Tsamardinos, A. Rakhshani, V. Lagani, "Performance-Estimation Properties of Cross-Validation-Based Protocols with Simultaneous Hyper-Parameter Optimization", *Artificial Intelligence: Methods and Applications Lecture Notes in Computer Science*, vol. *8445*, pp. *1-14*, 2014.
18. B. Dursun, F. Aydin, M. Zontul, S. Sener, "Modeling and Estimating of Load Demand of Electricity Generated from Hydroelectric Power Plants in Turkey using Machine Learning Methods", *Advances in Electrical and Electronics Engineering*, vol. *14*, no. *1*, pp. *121-132*, 2014.
19. M. Hacibeyoglu, A. Arslan, S. Kahramanli, "A Hybrid Method for Fast Finding the Reduct with the Best Classification Accuracy", *Advances in Electrical and Electronics Engineering*, vol. *13*, no. *14*, pp. *57-64*, 2013.
20. A. Frank, A. Asuncion, UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/

S. Batuhan AKBEN was born in Turkey in Istanbul. He is an Electrical and Electronics Engineer. He has an Phd. degree. He is an expert on biomedical signal processing, data mining nad artififcal intelligence. Now he is studying at Osmaniye Korkut Ata University, Bahce vocational School.