A New Distance-weighted k -nearest Neighbor Classifier

Article *in* Journal of Information and Computational Science · November 2011

4 authors, including:

Jianping Gou
125 PUBLICATIONS **3,745** CITATIONS

SEE PROFILE

Lan Du
Northeast University At Qinhuangdao Campus
78 PUBLICATIONS **1,213** CITATIONS

SEE PROFILE

Taisong Xiong
Chengdu University of Information Technology
18 PUBLICATIONS **488** CITATIONS

SEE PROFILE

Journal of Information & Computational Science 9: 6 (2012) 1429–1436
Available at http://www.joics.com

A New Distance-weighted *k*-nearest Neighbor Classifier

Jianping Gou[a,*], Lan Du[b], Yuhong Zhang[a], Taisong Xiong[a]

[a] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*

[b] *College of Engineering and Computer Science, the Australian National University Canberra 2601, Australia*

Abstract

In this paper, we develop a novel Distance-weighted k-nearest Neighbor rule (DWKNN), using the dual distance-weighted function. The proposed DWKNN is motivated by the sensitivity problem of the selection of the neighborhood size k that exists in k-nearest Neighbor rule (KNN), with the aim of improving classification performance. The experiment results on twelve real data sets demonstrate that our proposed classifier is robust to different choices of k to some degree, and yields good performance with a larger optimal k, compared to the other state-of-art KNN-based methods.

Keywords: K-nearest Neighbor Rule; Weighted Voting; Pattern Classification

1 Introduction

K-nearest Neighbor rule (KNN) has been one of the most well-known supervised learning algorithms in pattern classification, since it was first introduced [1]. This rule simply retains the entire training set during learning and assigns to each query a class represented by the majority label of its k-nearest neighbors in the training set. The Nearest Neighbor rule (NN) is the simplest form of KNN when $k = 1$. KNN has several main advantages: simplicity, effectiveness, intuitiveness and competitive classification performance in many domains. It has been found that the asymptotic error rate of KNN approaches the optimal Bayes error rate R^* when the number of the samples N and the number of neighbors k tend to infinity and $k/N \to 0$, and the error rate of NN is bounded above by twice the optimal Bayes error rate $2R^*$ [2, 3, 4]. Furthermore, the appeal of KNN stems from only a single integer parameter k, and the high classification performance with increasing the amount of training samples. As an improvement to KNN, Dudani introduced a distance-weighted KNN rule (WKNN) with the basic idea of weighting close neighbors more heavily, according to their distances to the query [5]. However, the number of available samples in real applications is usually too small to obtain a good asymptotic performance, which often leads to dramatic degradation of the classification accuracy, especially in the small sample size cases with the curse dimensionality [6] and existing outliers [7].

[*] Corresponding author.
Email address: cherish.gjp@gmail.com (Jianping Gou).

June 2012

Nowadays, one major challenging problem, yet to be resolved for KNN, is the selection of the neighborhood size k, which can have a significant impact on the performance of KNN-based classifiers [8]. It has been found that the classification performance of KNN intrinsically results in the estimate of the conditional class probabilities from training set in a local region of data space, which contains k nearest neighbors of the query [9]. The estimate is affected by the sensitivity of the selection of the neighborhood size k, because the radius of the local region is determined by the distance of the k-th nearest neighbor to the query and different k yields different conditional class probabilities. If k is very small, the local estimate tends to be very poor owing to the data sparseness and the noisy, ambiguous or mislabelled points. In order to further smooth the estimate, we can increase k and take into account a large region around the query. Unfortunately, a large value of k easily makes the estimate oversmoothing and the classification performance degrades with the introduction of the outliers from other classes. To deal with the problem, the related research works have been done to improve the classification performance of KNN [5, 10, 11, 12, 13]. Our work in this paper is inspired by the sensitivity issue of different choices of the neighborhood size k in KNN-based classifiers. We propose a new Distance-weighted k-nearest Neighbor rule (DWKNN) using the dual distance-weighted function, on basis of WKNN. In this new rule, we employ the dual distance-weights of k nearest neighbors to determine the class of the query by majority weighted voting. The experimental results suggest the superiority of our proposed DWKNN classifier in the practical situations.

The rest of this paper is organized as follows. In Section 2, we briefly review KNN and WKNN. In Section 3, we represent the sensitivity issue in KNN and describe our proposed DWKNN in detail. In Section 4, we give the experimental results on real data sets. Finally, we conclude our works in Section 5.

2 Outline of KNN and WKNN

2.1 KNN

In pattern classification, the Nearest Neighbor rule (NN) [2], is one of the oldest and simplest classifiers. The basic rationale for NN is defined as follows: given a set of the training samples and a query, find a point that is the closest to the query, and then assign its class label to the query. KNN is an extension of NN. In KNN, a query is labelled by a majority vote of its k-nearest neighbors in the training set. According to KNN or NN, we give a summary of the algorithmic procedure. Let $T = \{(x_i, y_i)\}_{i=1}^N$ denote the training set, where $x_i \in \Re^m$ is training vector in the m-dimensional feature space, and y_i is the corresponding class label. Given a query x', its unknown class y' is assigned by two steps.

Firstly, a set of k similar labelled target neighbors for the query x' is identified . Denote the set $T' = \{(x_i^{NN}, y_i^{NN})\}_{i=1}^k$, arranged in an increasing order in terms of Euclidean distance $d(x', x_i^{NN})$ between x' and x_i^{NN}

$$d(x', x_i^{NN}) = \sqrt{(x' - x_i^{NN})^T (x' - x_i^{NN})}. \tag{1}$$

Secondly, the class label of the query is predicted by the majority voting of its nearest neighbors:

$$y' = \arg\max_y \sum_{(x_i^{NN}, y_i^{NN}) \in T'} \delta(y = y_i^{NN}). \tag{2}$$

where y is a class label, y_i^{NN} is the class label for the i-th nearest neighbor among its k nearest neighbors. $\delta(y = y_i^{NN})$, the Dirac delta function, takes a value of one if $y = y_i^{NN}$ and zero otherwise.

2.2 WKNN

Dudani [5] first introduced a weighted voting method for KNN, called the distance-weighted k-nearest neighbor rule (WKNN). In WKNN, the closer neighbors are weighted more heavily than the farther ones, using the distance-weighted function. The weight w_i for i-th nearest neighbor of the query x' is defined as follow:

$$w'_i = \begin{cases} \frac{d(x',x_k^{NN})-d(x',x_i^{NN})}{d(x',x_k^{NN})-d(x',x_1^{NN})} & , \text{ if } d(x',x_k^{NN}) \neq d(x',x_1^{NN}), \\ 1 & , \text{ if } d(x',x_k^{NN}) = d(x',x_1^{NN}). \end{cases} \tag{3}$$

Then, the classification result of the query is made by the majority weighted voting:

$$y' = \arg\max_y \sum_{(x_i^{NN},y_i^{NN})\in T'} w'_i \times \delta(y = y_i^{NN}). \tag{4}$$

According to the Eq. (3), it can be see that a neighbor with smaller distance is weighted more heavily than one with greater distance: the nearest neighbor gets weight of 1, the furthest neighbor a weight of 0 and the other neighbors' weights are scaled linearly to the interval in between.

3 The Proposed DWKNN Classifier

In this section, we elaborate the sensitivity issue of the choice of the neighborhood size k in the KNN pattern classification setting, and then present our DWKNN classifier.

3.1 Problem Representation

How to select a suitable neighborhood size k is a key issue that largely affects the classification performance of KNN [8]. As for KNN, the small training sample size can greatly affect the selection of the optimal neighborhood size k and the degradation of the classification performance of KNN is easily produced by the sensitivity of the selection of k. Generally speaking, the classification results are very sensitive to two aspects: the data sparseness and the noisy, ambiguous or mislabelled points if k is too small, and many outliers within the neighborhood from other classes if k is too large. From a theoretical point of view, the classification performance of KNN is determined by the estimate of the conditional class probabilities of the query in a local region of the data space, which is determined by the distance of the k-th nearest neighbor to the query. So the classification performance is very sensitive to the selected value of k. Furthermore, the simplest majority voting of combining the class labels for KNN can be a problem if the nearest neighbors vary widely over their distances and the closer ones more reliably indicate the class of the query object. With the goal of addressing the sensitivity issue of different choices of the neighborhood size k, some weighted voting methods have been developed for KNN, for instance, in [5].

Despite WKNN is less sensitive to the choices of the neighborhood size k than KNN, the robustness to the change of k is still poor. WKNN is better than other weighted voting for KNN through their empirical comparisons in many cases [9]. Nevertheless, WKNN still suffers from the issue because of the existing outliers, particularly in the case of the small sample size [7]. Moreover, the irregular class distribution of a data set has a dramatic influence on both KNN and WKNN. On those data sets with the imbalanced class distribution, the number of training samples in some classes is much larger than that of the others. Under this condition, the estimate of each conditional class probability of each query is to be very unreliable because the query may be largely subject to the outliers, and the sum weighted vote of each class in k nearest neighbors is unsuitable to make decision. As a consequence, WKNN can also be affected by the sensitivity of different choices of k so as to degrade classification performance.

Motivated by the issue as mentioned above, we propose a new distance-weighted k-nearest neighbor rule (DWKNN). In our approach, the dual distance-weight substitutes the corresponding weight for each nearest neighbor in WKNN. The dual weight is determined through multiplying the original weight from Eq. (3) by another new weight. In contrast to WKNN, the new method reduces the weight of each nearest neighbor except the first closest and the k-th nearest neighbors. It can keep from giving too much weight to the outliers by reducing the weights of other neighbors in the set of k nearest neighbors for each query, and improve the classification performance. Hence, DWKNN can deal with the outliers in the local region of a data space, in order that the degree of the sensitivity of different choices of k can be degraded. The experimental results suggest that the new classifier is a promising algorithm in many practical situations.

3.2 The Proposed DWKNN Classifier

We design a simple and effective classifier, i.e. DWKNN, to reduce the influence of the sensitivity of the selection of the neighborhood size k to some degree and yield the good performance in pattern classification. Let $\bar{T} = \{(x_i^{NN}, y_i^{NN})\}_{i=1}^{k}$ denote the set of the k-nearest neighbors to the query $\bar{x}$ arranged in an increasing order according to the distance $d(\bar{x}, x_i^{NN})$ between $\bar{x}$ and x_i^{NN}, and $\bar{W} = \{\bar{w}_1, ..., \bar{w}_k\}$ be the set of the corresponding dual weights. DWKNN is based on WKNN: to give different weights to k nearest neighbors according to their distances, with closer neighbors having greater weights. Nevertheless, different from the weights in WKNN, we assign to the i-th nearest neighbor x_i^{NN} of the query $\bar{x}$ a dual weight $\bar{w}_i$, defined by the dual distance-weighted function as below:

$$\bar{w}_i = \begin{cases} \frac{d(\bar{x},x_k^{NN})-d(\bar{x},x_i^{NN})}{d(\bar{x},x_k^{NN})-d(\bar{x},x_1^{NN})} \times \frac{d(\bar{x},x_k^{NN})+d(\bar{x},x_1^{NN})}{d(\bar{x},x_k^{NN})+d(\bar{x},x_i^{NN})} & , \text{ if } \ d(\bar{x}, x_k^{NN}) \neq d(\bar{x}, x_1^{NN}), \\ 1 & , \text{ if } \ d(\bar{x}, x_k^{NN}) = d(\bar{x}, x_1^{NN}). \end{cases} \tag{5}$$

And then, we label the query $\bar{x}$ by the majority weighted vote of k nearest neighbors, the same as Eq. (4).

$$\bar{y} = \arg\max_{y} \sum_{(x_i^{NN}, y_i^{NN}) \in \bar{T}} \bar{w}_i \times \delta(y = y_i^{NN}). \tag{6}$$

It is important to note that the dual weight of each nearest neighbor consists of two parts: the first part is same as the weight in WKNN, the second one is a new distinct weight defined by ourself, and they both build on the basic idea of the distance-weighted scheme. With respect to Eq. (5), it is obvious that the dual weight $\bar{w}_i$ is smaller than the weight w_i computed by Eq. (3) in

WKNN, except the weights of the first and k-th nearest neighbors. As a result, the corresponding neighbor x_i^{NN} has less influence on the classification result of the query. The dual weight drops quickly from 1 at the distance of the first nearest neighbor to 0 at the distance of the furthest k-th nearest neighbor.

In summary, the algorithmic form of the proposed DWKNN is described in Algorithm 1. Note that the neighborhood size k, is optimally selected by k-fold cross validation.

Algorithm 1 The proposed DWKNN algorithm

```
Step 1: Compute the distances of nearest neighbors of the query $\bar{x}$.
for $i = 1$ to $N$ do
  $d(\bar{x}, x_i) = \sqrt{(\bar{x} - x_i)^T(\bar{x} - x_i)}$
end for
Step 2: Sort the distances in an ascending order.
$[sorted\_\ index, sorted\_\ dist] = sort(d(\bar{x}, x_i), ascend)$
Step 3: Search $k$-nearest neighbors of the query $\bar{x}$, $\bar{T} = \{(x_i^{NN}, y_i^{NN})\}_{i=1}^{k}$.
for $i = 1$ to $k$ do
  $x_i^{NN} = x_{sorted\_\ index(i)}$, $y_i^{NN} = y_{sorted\_\ index(i)}$
end for
Step 4: Calculate the dual weights of $k$ nearest neighbors, $\bar{W} = \{\bar{w}_1, ..., \bar{w}_k\}$.
for $i = 1$ to $k$ do
  if $d(\bar{x}, x_k^{NN}) \neq d(\bar{x}, x_1^{NN})$ then
    $\bar{w}_i = \frac{d(\bar{x},x_k^{NN})-d(\bar{x},x_i^{NN})}{d(\bar{x},x_k^{NN})-d(\bar{x},x_1^{NN})} \times \frac{d(\bar{x},x_k^{NN})+d(\bar{x},x_1^{NN})}{d(\bar{x},x_k^{NN})+d(\bar{x},x_i^{NN})}$
  else
    $\bar{w}_i = 1$
  end if
end for
Step 5: Assign a majority weighted voting class label $\bar{y}$ to the query $\bar{x}$.
  $\bar{y} = \arg\max_{y} \sum_{(x_i^{NN}, y_i^{NN}) \in \bar{T}} \bar{w}_i \times \delta(y = y_i^{NN})$
```

4 Experimental Results

In this section, we thoroughly explore the performance of the proposed DWKNN classifier, in comparison with NN, KNN and WKNN. To yield the reliable classification performance of all the classifiers, we conduct experiments by 20-fold cross validation with 95% confidence in term of the classification accuracy.

4.1 Experimental Data Sets

The twelve real data sets used in our experiments are available from the UCI machine learning repository [14]. The overall properties of the data sets are described in Table 1. For short, among these data sets, the abbreviated names for Optdigits, Ionosphere, Landsat Satellite, Image Segmentation, Cardiotocography and Parkinsons are Opt, Iono, Landsat, Image, Cardio and Parkin, respectively. In the experiments, each data set is randomly partitioned into the training and testing samples, shown in Table 1. We do experiments 20 trials and obtain 20 different training and testing sets on each data set for performance evaluation. Note that the values of the neighborhood size k in the experiments vary from 1 to 15 by Step 1. The optimal or sub-optimal value of k on each data set that obtains the highest accuracy rate, is chosen within the interval. The averaged classification accuracy rate is achieved as the final performance.

Table 1: The UCI data sets and the corresponding partitions

Data set	Features	Samples	Classes	Training samples	Testing samples
Glass	10	214	7	140	74
Wine	13	178	3	100	78
Sonar	60	208	2	120	88
Parkin	22	195	2	120	75
Iono	34	351	2	200	151
Musk	166	476	2	276	200
Vehicle	18	846	4	500	346
Image	19	2310	7	1310	1000
Cardio	21	2126	10	1126	1000
Opt	64	5620	10	3000	2620
Landsat	36	6435	7	3435	3000
Letter	16	20000	26	10000	10000

4.2 Experimental Comparisons

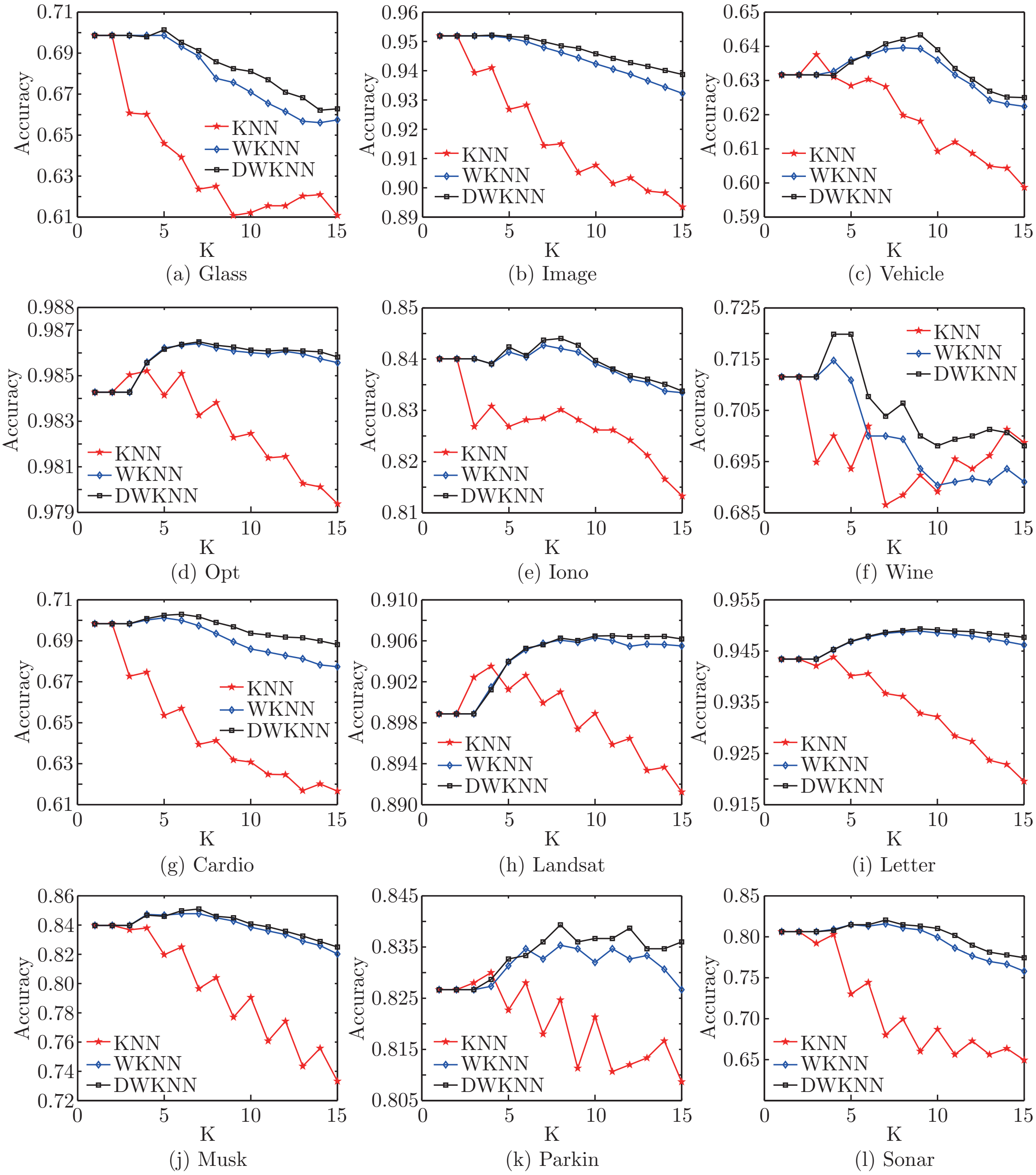
We compare DWKNN with NN, KNN and WKNN in terms of the best performance with the accuracy rates and the corresponding values of k and standard deviations (stds). The average best accuracy rates in % of each method are shown in Table 2. As shown in Table 2, we can clearly see that the best performance of KNN is almost similar to that of NN on each data set, WKNN and DWKNN are better than NN and KNN. More importantly, we can find that DWKNN is superior to NN, KNN and WKNN on all the data sets with larger optimal k than KNN. Hence, we can

Table 2: The best accuracy rates of each method with corresponding standard deviations (stds) and k in the parentheses (the accuracy rates in bold-face are the best performance among the methods)

Data set	NN	KNN	WKNN	DWKNN
Glass	69.86±1.35	69.86±1.35(1)	69.86±1.56 (1)	**70.14±1.35 (5)**
Wine	71.15±1.81	71.15±1.81(1)	71.47±1.54 (4)	**71.99±1.38 (4)**
Sonar	80.62±1.62	80.62±1.62 (1)	81.59±1.81 (4)	**82.05±1.81 (5)**
Parkin	82.67±2.05	83.00±1.80 (4)	83.53±1.65 (8)	**83.93±1.86 (8)**
Iono	84.01±1.36	84.01±1.36 (1)	84.27±1.24 (7)	**84.44±1.24 (8)**
Musk	83.98±0.94	83.98±1.06 (1)	84.77±0.87 (6)	**85.10±0.94 (7)**
Vehicle	63.16±0.79	63.76±1.24 (3)	63.96±1.02 (8)	**64.34±1.17 (9)**
Image	95.19±0.31	95.19±0.31 (1)	95.19±0.33 (1)	**95.21±0.31 (4)**
Cardio	69.84±0.49	69.84±0.49 (1)	70.12±0.48 (5)	**70.30±0.45 (6)**
Opt	98.43±0.12	98.52±0.12 (4)	98.64±0.13 (7)	**98.65±0.14 (7)**
Landsat	89.89±0.21	90.35±0.27 (4)	90.63±0.25 (10)	**90.65±0.26 (11)**
Letter	94.34±0.086	94.38±0.10 (4)	94.89±0.092 (9)	**94.93±0.088 (9)**

draw a sound conclusion that the preferable classification performance of DWKNN is attained by utilizing more nearest neighbors for each query.

To further explore the sensitivity of the DWKNN performance to the neighborhood size k, the average classification results of DWKNN, WKNN and KNN on each date set via k are shown in Fig. 1. It is clear from the figure that WKNN and DWKNN mostly outperform KNN with the different values of k. As shown in Fig. 1, the average accuracy rates of DWKNN are almost better than those of KNN and WKNN, particularly when k is large. Consequently, it suggests that the proposed DWKNN has the robustness to the sensitivity of different choices of the neighborhood



(a) Glass

(b) Image

(c) Vehicle

(d) Opt

(e) Iono

(f) Wine

(g) Cardio

(h) Landsat

(i) Letter

(j) Musk

(k) Parkin

(l) Sonar

Fig. 1: The average accuracy rates via the neighborhood size k

size k with good performance to some degree.

Across all our experiments above, it can be concluded that, using our dual distance-weighted function, the proposed DWKNN not only improves the classification performance, but also overcomes the sensitivity of the selection of the neighborhood size k in many practical situations.

5 Conclusion

In this paper, we present a new distance-weighted k-nearest neighbor rule, which is based on WKNN. In our approach, we concentrate on dealing with the sensitivity of different choices of k, with the goal of improving the classification performance. Our main contribution is to design a novel dual distance-weighted function for making classification decision. To systematically verify our proposed classifier, the experiments are conducted on twelve real data sets, compared to NN, KNN and WKNN. Through our experiments, the proposed method always outperforms the other classifiers among a large range of k and its effectiveness is demonstrated with good performance.

References

[1] E. Fix, J. L. Hodges, Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties, Technique Report No. 4, U. S. Air Force School of Aviation Medicine, Randolf Field Texas, 1951, 238-247

[2] T. M. Cover, P. E. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory, 13 (1), 1967, 21-27

[3] T. Wagner, Covergence of the nearest neighbor rule, IEEE Transactions on Information Theory, 17 (5), 1971, 566-571

[4] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statical Learning. Springer, New York, NY, USA, 2001

[5] S. A. Dudani, The distance-weighted k-nearest neighbor rule, IEEE Transactions on System, Man, and Cybernetics, 6 (1976), 325-327

[6] F. Fernández, P. Isasi, Local feature weighting in nearest prototype classification, IEEE Transactions on Neural Networks, 19(1), 2008, 40-53

[7] K. Fukunaga, Introduction to Statistical Pattern Recognition, second ed., Academic Press, 1990

[8] X. D. Wu, V. Kumar et al, Top 10 algorithems in data mining, Knowledge Information System, 14 (2008), 1-37

[9] J. Zavrel, An empirical re-examination of weighted voting for K-NN, Proceedings of the 7th Belgian-Dutch Conference on Machine Learning, Tilburg, 1997, 139-148

[10] Y. Mitani, Y. Hamamoto, A local mean-based nonparametric classifier, Pattern Recognition Letters, 27 (2006), 1151-1159

[11] Y. Zeng, Y. Yang, L. Zhao, Pseudo nearest neighbor rule for pattern classification, Expert Systems with Applications, 36 (2009), 3587-3595

[12] J. Wang, P. Neskovic, L. N. Cooper, Neighborhood size selection in the k-nearest-neighbor rule using statistical confidence, Pattern Recognition, 39 (2006), 417-423

[13] P. Kang, S. Cho, Locally linear reconstruction for instance-based learning, Pattern Recognition, 41 (2008), 3507-3518

[14] A. Frank, A. Asuncion, UCI Machine Learning Repository, http://www.archive.ics.uci.edu/ml, 2011