The Improved SOM-Based Dimensionality Reducton Method for KNN Classifier Using Weighted Euclidean Metric

[1,*]Jiunn-Lin Wu and [2]I-Jing Li

Abstract

This paper presents an improved SOM-based dimensionality reduction method for k nearest neighbor classifier called weighted self-organizing feature map (WSOM). The self-organizing feature map is one of the appropriate dimensionality reduction techniques for k nearest neighbor classifier because it can highly maintain topological relationships among samples in a lower dimensional space. Finding the winner step of the SOM is usually based on the Euclidean metric. The Euclidean metric implies the equal importance of features, an impractical assumption in many applications. This paper proposes a feature weighting method in terms of the ratio of between class variance and within class variance in the training samples. Therefore, the larger weights are given to the more important features, and the lesser weights are offered to the less important ones. Both the synthetic datasets and the real-world datasets are used to compare the performance among SOM-based approaches. Experimental results show that the proposed method outperforms other SOM-based algorithms, and it is a useful feature reduction method for k nearest neighbor classifier.

Keywords: Weighted Euclidean metric, nearest neighbor classifier, dimensionality reduction, self organizing feature map

1. Introduction

The self-organizing feature map (SOM) is a data visualization tool used in many applications such as data mining and neural network [1]-[2]. The goal of the self-organizing feature map is to transform an input space into a one or two- dimensional discrete map for easily observing topological ordering from the feature map. The neurons placed on the maps are based on competitive learning, and the winner neuron wins the competition according to a winner-takes-all strategy. The method iteratively adjusts the weights of neurons, and considers the final neuron weights as prototypes of the input dataset. Therefore, the SOM algorithm is also the abstraction process that generates fewer prototypes from the original data space.

The great property of the self-organizing feature map is preserving topological mapping, and it often uses a 2D lattice of neurons for displaying the relationships among samples. This property is helpful for k nearest neighbor classifier [3] which has to choose k ordered patterns in the classification phase. The k nearest neighbor algorithm is a simple and nonparametric classifier. This algorithm has only one tuned parameter, and it does not need any class distribution assumptions widely used in many pattern recognition applications such as face recognition and context classification. The classification phase of k nearest neighbor algorithm assigns a query pattern class that is the most frequently occurring in k nearest neighbors. Since the SOM algorithm preserves the distance and proximity in the lower dimensional space, it is an appropriate feature reduction technique for the k nearest neighbor classifier [4].

One drawback of the k nearest neighbor rule is the dimensionality curse [5]-[6]. In general, large samples can maintain performance when the dimensionality is high. However, it usually does not obtain enough samples to maintain the performance in a high-dimensional space in the real world. A solution for this problem is to use a dimensionality reduction method for k nearest neighbor classifier [3]. Using the SOM algorithm for k nearest neighbor classifier is a logical way to reduce the number of features. The reasons are described as follows. First, topological mapping preservation may not significantly affect performance of the nearest neighbor classifier when there are not a large number of features. Secondly, using fewer features will speed up the classification time. Thirdly, using two dimensional mesh grids of neurons is easy for visualization and analysis.

*Corresponding Author: Jiunn-Lin Wu
(E-mail:jlwu@cs.nchu.edu.tw)
[1]Dept. of Computer Science and Engineering, National Chung Hsing University,250, KuoKuang Rd., 402 Taichung, Taiwan
[2]Dept. of Computer Science and Engineering, National Chung Hsing University,250, KuoKuang Rd., 402 Taichung, Taiwan

Although the SOM algorithm is a useful feature reduction method, the structures of class samples may not be apparent; and the shapes may often become distorted because of rectangular or hexagonal shaped maps, i.e., fixed size of maps. Yin proposed a visualization-induced self-organizing feature map (ViSOM) [7] to solve this shortcoming. This ViSOM approach adds a parameter that constrains the lateral feedback of inter neurons and controls the map resolution. The mapping results have shown that ViSOM preserves the data structures more faithfully and the neurons distributed smoothly. Obviously, the ViSOM is helpful for visualization and data clustering.

This paper focuses on another problem of SOM. Choosing the winner step of the self-organizing feature map algorithm is usually based on a typical Euclidean distance function. In fact, it is the nearest neighbor algorithm. The Euclidean metric treats all features equally; as a result, the nearest neighbor algorithm is easily over-fitted, especially in a high dimensional space. If the incorrect winner is chosen and used to adjust neuron weights on the feature map, the performance of SOM will be easily declined in the learning phase. To improve this situation, this study proposes a feature weighting method formed a weighted Euclidean distance function, namely weighted self organizing feature map (WSOM). The proposed method gives weights to features by the ratio of between class variance and within class variance from training samples. It is a similar concept of Linear Discriminant Analysis (LDA) [8]. A larger ratio value means it is a significant feature; in other words, it is beneficial for classification. On the contrary, a smaller value indicates it is an ambiguous feature that would lead misclassified. The weighted distance function makes the neighborhood of the input vector elongate the insignificant feature dimensions and shorten the more importance ones. Consequently, the proposed method chooses the more correct winner and adjusts the weights of neighbor neurons. For this reason, the performance of SOM will be improved.

In the first experiment, this study uses a noise dataset to verify the proposed feature weighting method. The results show that the proposed feature weighting method can discriminate between noisy features and normal features. Next, both the artificial datasets and the real datasets are used to compare the performance of different feature reduction methods for the k nearest neighbor classifier including principal component analysis (PCA), traditional SOM, the proposed WSOM and ViSOM. To demonstrate the performance with different feature weighting methods in WSOM algorithm, the paper uses a feature weighting approach based k -means clustering proposed by Huang [9]. The k -means clustering algorithm is popular and easy to implement. Experimental results show that using WSOM as a feature reduction method for k nearest neighbor obtains great performance.

The major contributions of this paper are as follows:

1). the proposed feature weighting method is effective and easily computed. The results show that it achieves better performance than k -means clustering algorithm.

2). the proposed WSOM method is superior to a traditional self-organizing feature map algorithm on account of the weighted distance in finding the winner step.

3). using WSOM as a feature reduction approach for k nearest neighbor classifier is appropriate and speeding up the classification time.

4). the classification accuracy of the nearest neighbor classifier is improved while the proposed WSOM is used to reduce the number of features.

This paper is organized in the following. We present related work about k nearest neighbor algorithm, dimensionality reduction methods and feature weighting methods in section 2. The details of WSOM algorithm is described in section 3. Some experimental results and comparisons with other methods are shown in section 4. Finally, conclusions are given in section 5.

2. Related Works

This section reviews briefly the k nearest neighbor algorithmand the dimensionality reduction methods including component analysis, traditional self- organizing feature maps, and the visualization-induced self-organizing feature map (ViSOM). This section also describes some feature weighting methods.

2.1 k Nearest Neighbor Classifier

The k nearest neighbor classifier is a simple and supervised classification widely used in many pattern recognition applications. The rule assigns a query pattern to the most frequently occurring in the nearest neighbors. The literature has proven the nearest neighbor rule to be robust with an asymptotic error rate [3]. The rule has only one tuned parameter and no prior knowledge about class distributions. Although the nearest neighbor has the above-mentioned advantages, it has some problems. The major issue is dimensionality curse [5]-[6], which means k nearest neighbor classifier is less effective in high dimensional feature space with a finite set of training samples. A major approach to solve this problem is to select an appropriate distance measure by weighting features [10]-[12]. These

methods provide small weights to insignificant features and large weights to influential ones. These approaches have proven these methods effective for a high-dimensional space; however, calculating the weights of features is time-consuming and complex.

To increase the performance of the k nearest neighbor algorithm, some papers addressed the prototype selection issues [13]-[18]. These methods accelerate classification time and diminish risk of the over-fitting problem. However, how to find the optimal number of prototypes is still an open problem. Moreover, prototypes are sensitive to noise and not robust. They can achieve excellent performance on some datasets, but poor performance on other datasets.

Using a feature reduction method offers another way to avoid dimensionality curse [3]. Principal component analysis (PCA) and the self-organizing feature map (SOM) [19] are common dimensionality reduction methods. If we adopt a feature reduction method appropriately, it does not affect the performance much. In addition, the classification time can speed up. The following briefly reviews these methods.

2.2 Principal Component Analysis

Principal component analysis is a well-known feature reduction method used in face recognition, handprint recognition, and data compression. The goal of principal component analysis is to find a subspace whose basis vectors correspond to the maximum-variance directions in the original space. This means that the newly transformed space retains the most intrinsic data information using fewer features. The transformation based on statistical properties of the vector is referred to as the most significant eigenvectors of the covariance matrix. The minimizing sum-square-error criterion for the optimization process is based on the following equation:

$$\min \sum_{x} \left[x - \sum_{j=1}^{m} (q_j^T x) q_j^T \right]^2 \quad (1)$$

where $x = [x_1, x_2, ..., x_d]^T$ is d -dimensional input vector, $q_j (j = 1,2,...,m \ m \le d)$ represents the first m principal eigenvectors of the covariance matrix of the original feature space. By selecting the dominant component of eigenvectors, the principal component analysis algorithm reduces the number of features and represents the data in a lower dimensional space. The disadvantage of PCA is that it cannot capture nonlinear relationships defined by higher than second-order statistics, so it is unable to maintain topological relationships in a lower dimensional space. Therefore, it is not a suitable feature reduction method for the k nearest neighbor classifier because k nearest neighbor classifier has to select ordered samples in the classification phase.

2.3 Self-organizing Feature Maps Algorithm

A proper dimension reduction method for k NN is Multidimensional Scaling (MDS) [3] which seeks to present data points in a lower dimensional space while preserving inter-distances between the data points as far as possible. However, these methods have high computational complexity, and the local minimum problems occur in the optimization process. In addition, they cannot provide an explicit mapping function to accommodate new data points [20]. An approximate algorithm corresponding to multidimensional scaling methods is called the self-organizing feature map or topological ordered maps [1]-[2].The following explicitly describes the self-organizing feature map algorithm.

The self-organizing feature map algorithm is an unsupervised learning algorithm that has used a finite number of neurons to map the input space into a lower feature space. The SOM method is based on competitive learning, and the neuron with the largest activation becomes the winner. The synaptic weights of neurons are iteratively adjusted at each learning cycle. When the SOM algorithm converges, the feature maps display the topological relationship among the data points. On the basis of this property, the study proposes a revised version of SOM as a dimensionality reduction technique for k nearest neighbor classifier.

One of the disadvantages of SOM is that the structures of the data are distorted and dissimilar to the original data distributions [21]-[22]. Yin proposed a novel feature extraction method called visualization induced self-organizing feature map (ViSOM) [7], [23] to overcome this defect. A parameter λ controls the lateral forces between neurons and yields better data structure on the feature map.

This paper deals with another problem for SOM. The shortcoming of the SOM algorithm is finding the winner $i(x)$ of the d -dimensional input vector x according to the formulation:

$$i(x) = \arg\min_j \left\| x - w_j \right\|, \ j = 1,...,n \quad (2)$$

where $w_j = [w_{j1}, ..., w_{jd}]^T$ is denoted by the synaptic weight of the neuron j and n is the number of neurons. The Eq. (2) usually is taken by a traditional Euclidean distance function that treats all features equally. It is unpractical in the real world, especially, when the number of features is high. We have developed a feature weighting method in terms of the ratio of between class variance and within class variance in the training samples. The formulation is adapted from the LDA algorithm [8]. Along a feature, larger variance of between classes means it is a more important feature. On the other hand, if the inner class variance is smaller, the class destructions are closed; and the larger weight is given to this feature. The next subsection reviews the literature of some feature weighting methods.

2.4 Feature Weighting Method

The typical Euclidean distance measure implies that the input space is isotropic and homogeneous. This assumption let it perform poor in a complex feature space. Many researches in recent years have focused on the feature weighting method in terms of margin distance [24]-[26]. The maximal margin is defined as the distance between the nearest same-labeled pattern to x and the nearest different-labeled data to x. The goal of these approaches is providing smaller weights to insignificant features and larger weights to influential ones. The advantages of feature selection or the feature weighting method are obvious. Firstly, fewer features reduce processing time. Next, using fewer features prevents over-fitting problems in a high-dimensional space. However, the above-mentioned methods are iteratively optimized and do not find the optimal solution easily. Veeman and Tax proposed a sparse classifier combining the feature weighting method and Nearest Mean Classifier (NMC), called LESS [27] The drawback about LESS is that it assumes each feature has equal variance, and is used for only two-class classification problems. Many simple weighting variable methods have been proposed such as Heterogeneous Euclidean-Overlap Metric (HEOM) [28], Value Different Metric (VDM) [28], [29] Heterogamous Value Difference Metric (HVDM) [28], [30], and Interpolated Value Difference Metric (IVDM) [30]. These methods are easily computed and applied to both categorical and numerical variables. Nevertheless, these feature weighting methods do not perform well in all cases. Furthermore, Huang et al proposed a feature weighting method based on k-means clustering algorithm [9]. The k-means is the most popular and well-known clustering algorithm in pattern recognition and data mining. The literature demonstrates mathematical proof and lists the weight results through the weighted k-means clustering algorithm. In the study, the artificial datasets and the real datasets are used to compare performance with the proposed method and the weighting feature method by weighted k-means clustering algorithm in the experiments.

3. Proposed Method

In this paper, we propose a revised version of SOM using weighted Euclidean metric. The feature weighting method is similar to LDA with some modifications. This section firstly introduces the proposed feature weighting method, and then describes weighted self-organizing feature algorithm.

3.1 Feature Weighting Method

Let $x_i^{w_j} = [x_{i1}^{w_j}, x_{i2}^{w_j}, ..., w_{id}^{w_j}]^T$, $(i = 1,..., N_j)$ and $(j \in \{1,..., N_c\})$, be the d-dimensional training sample of class w_j, where N_c is the number of classes and N_j is the number of the samples of the j th class. $N = \sum_{j=1}^{N_c} N_j$ is denoted by the total number of training samples. In order to measure the difference among features, the global mean of total training samples of feature v is calculated by $m_v = (1/N)\sum_{i=1}^{N_j}\sum_{j=1}^{N_c} x_{iv}^{w_j}$ $(v = 1,..., d)$, and local mean of class w_j of feature v is $u_v^{w_j} = (1/N_j)\sum_{i=1}^{N_j} x_{iv}^{w_j}$ $(v = 1,..., d)$.

The way to estimate the weights of features is adapted from LDA, the *between-class variance* S_v^B of the feature v which is defined by the following formulation:

$$S_v^B = \frac{1}{N_c}\sum_{j=1}^{N_c}(m_v - u_v^{w_j})(m_v - u_v^{w_j}),\ v = 1,..., d \qquad (3)$$

Similarly, the *within class variance* S_v^W of the feature v is computed by

$$S_v^W = \sum_{j=1}^{N_c} P_j S_{jv},\ v = 1,...d \qquad (4)$$

where $P_j = N_j / N$ is the prior probability of class w_j, and S_{jv} is the variance of class w_j of feature v. Finally, the weighting feature f_v is according to the ratio of the between-class variance S_v^B to the within-class variance S_v^W, that is

$$f_v = \frac{S_v^B}{S_v^W},\ v = 1,..., d \qquad (5)$$

It is apparently that this metric is based on LDA but with some modification. According to this rule, if the value of variance of between classes is large, it implies that the distributions of classes are separate, and this feature is helpful for classification. On the hand, if the value of S_v^W is small, it indicates that inter class samples are closed. Therefore, a larger weight should give to this feature. The advantages of the proposed feature weighting method are described as follows. In general, there exist correlations among features and noisy features in the data space. Our proposed method is easily computed and offers a means to discriminate between the noisy feature and the normal feature. Thus we used the feature weighting method to improve the performance of SOM.

3.2 Weighted Self-organizing Feature Map Algorithm

The WSOM algorithm uses a weighted Euclidean distance function in the learning phase. The modified formulation of finding the winning node $i(x)$ is defined as

$$i(x) = \arg\min_j \sum_{v=1}^{d} f_v (x_v - w_{jv})^2,\ j = 1,...,n \qquad (6)$$

where the feature weighting f_v is referredin Eq. (5). Since a self-organizing feature map uses a completive learning rule, the lateral feedback between neurons is often referred as the Gaussian function model. The neighborhood function around the winner neuron $i(x)$ at time t is given by

$$\Lambda(i, j, t) = \exp(-\frac{d_{i,j}^{\ 2}}{\sigma(t)^2}),\ j = 1,...,n \qquad (7)$$

where $d_{i,j} = \|i(x) - j\|$ is the Euclidean distance between the winning node $i(x)$ and the corresponding neuron j in the lattice, and the parameter $\sigma(t)$ defines an effective width on the feature maps around the winning node. Both $\sigma(t)$ and $\Lambda(i, j, t)$ are a monotonically decreasing time-varying function. Then the synaptic vector weights are adjusted by all neurons according to the formulation

$$w_j(t+1) = w(t) + \eta(t)\Lambda(i, j, t)[x(t) - w_j(t)],\quad j = 1,...,n \qquad (8)$$

where $\eta(t)$ represents the learning rate that also decreases as time goes by. Finally, the steps and the flowchart of the proposed method are illustrated below:

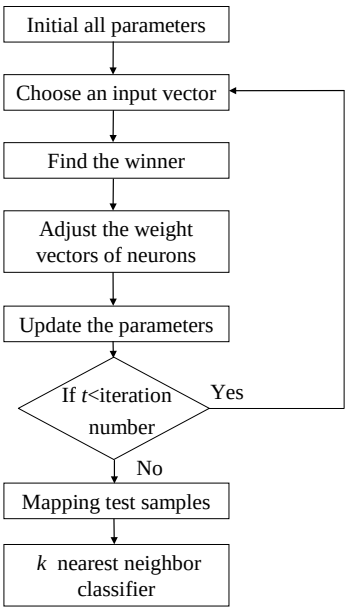


Figure 1: The flowchart of the proposed method.

Step 1 *Initialization*: initialize the weight vector, $w_j(0)$ which is a random value. Initialize the learning rate $\eta(0)$ and the neighborhood function $\Lambda(i, j, 0)$ ($\sigma(0)$).

Step 2 *Sampling*: randomly choose an input vector x from the feature original space.

Step 3 *Finding Winner*: select the best matching winning node $i(x)$ according to the minimum weighted Euclidean distance criterion defined by Eq.(6)

Step 4 *Learning*: adjust the weighted vector of all neurons using the formulation (8).

Step 5 *Updatin*g: reduce the learning rate $\eta(t)$ and the neighborhood function $\Lambda(i, j, t)$.

Step 6 *Continuations*: if the weights of the neurons do not change, exit the iteration. Otherwise, go to step 2.

Step 7 *Mapping*: draw all test samples to be projected to the learned feature maps.

Step 8 *Classification*: use the k nearest neighbor classifier in the two-dimensional mapping samples.

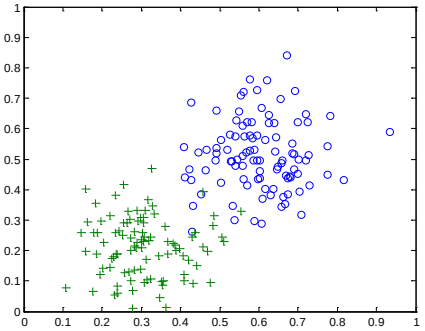
4. Experimental Results

In the experiments, we firstly use a noisy dataset to identify the normal feature and the noisy feature by the proposed feature weighting method. Afterward, both the simulated dataset and the real dataset are used to evaluate the performance with the proposed weighted self-organizing feature map algorithm and different feature reduction methods. In order to achieve an impartial result, we select the most two significant features of PCA for comparisons. All experiments were run on a 3.2 GHz Pentium IV machine with 4GB RAM. The size of the neuron-mapping grid was set to 10×10. The fixed parameters in the SOM algorithm were as follows: the width of the Gaussian-function $\sigma(t)$ was 5,and the minimum width of $\sigma(t)$ was 0.1. The minimum learning rate was set to 0.001. The weighting factor β is 2 in the k-means clustering algorithm in all datasets.

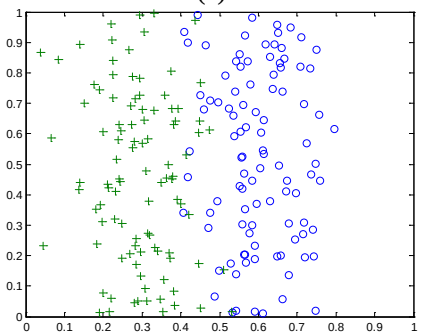
4.1 Experiment on a Noisy Dataset

The researchers often use a noisy dataset to identify noisy features through a feature weighting approach. In the first experiment, we generated 200 samples which are divided into two classes that contain three features. The noisy dataset has three features, f_1, f_2, f_3. The first two variables are normally distributed, and the third one is a uniformly distributed noisy variable. The centroids of the two

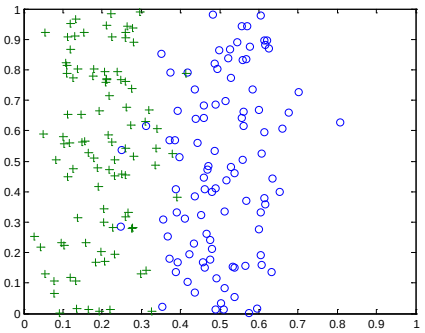
classes are (0.6, 0.5) and (0.3, 0.2). The Gaussian standard deviations are all set to 0.1. Figure 2 plots the 200 samples in different two-dimensional subspaces. It can be seen in Figure 2 (a) that f_1 and f_2 are significant features. The noisy feature can be observed in Figs 2 (b) and (c). The weighting results for features f_1, f_2, and f_3 through the proposed feature weighting method are 2.56, 2.09, and 0.056 respectively. It is evident from the results that the proposed feature weighting method can discriminate between the normal features and the noisy feature. In addition, comparing Figure 2 (b) and Figure 2 (c) shows f_1 is a better feature for classification which is ascribed to the distributions with somewhat being overlapped. The weighting results are consistent with the real feature data space. Thus the proposed weighting method is sufficient to display the influence of features.



(a)



(b)



(c)

Figure 2: Class distributions on a noisy dataset.

(a) Two classes in the subspace: f_1, f_2.

(b) Two classes in the subspace : f_1, f_3

(c) Two classes in the subspace : f_2, f_3.

4.2 Experiment on Synthetic Dataset

In this experiment, we use two typical synthetic datasets to evaluate the performance of the proposed method compared with other feature reduction methods. The synthetic datasets are all 8-dimensional Gaussian data. k nearest neighbor classifier is less effective, when the class distributions are not separable and overlapped.This study adopted the $I-4I$ dataset and the $I-\Lambda$ dataset [31] to compare performance among different feature extraction methods.

$I-4I$ dataset :

$$u_1 = u_2 = \mathbf{0},$$
$$\Sigma_1 = I_8, \Sigma_2 = 4I_8$$

$I-\Lambda$ dataset:

$$u_1 = \mathbf{0},\ u_2 = [3.86,3.10,0.84,0.84,1.64,1.08,2.06,0.01]^T,$$
$$\Sigma_1 = I_8, \Sigma_2 = \mathrm{diag}[8.41,12.06,0.12,0.22,1.49,1.77,0.35,2.73],$$

Where I_k and diag[·] denote the $k \times k$ identity and diagonal matrices. The $I-4I$ dataset contains two normally distributed classes with the same mean and different variance. The $I-\Lambda$ dataset contains two classes that distribute normally with different means and variance. This study generated 100 training samples in each class with 1000 samples in each class at the classification stage. Table 1 lists the parameters used in this experiment including the initial learning rate $\eta(0)$, the decreasing learning rate η_α, and the decreasing width rate σ_β. The parameter λ is set to 0.4 in the ViSOM algorithm in the synthetic datasets.

Table 1: The parameters of the proposed method used in the artificial datasets.

Dataset	*Parameters*
$I-4I$	$\eta(0)=0.1, \eta_\alpha=0.95, \sigma_\beta=0.975$
$I-\Lambda$	$\eta(0)=0.5, \eta_\alpha=0.975, \sigma_\beta=0.975$

Figure 3 shows the average accuracy with different k parameters of the four methods in the $I-4I$ dataset. It is obvious that k NN gets the wrong number of class labels from k candidate samples because the class distributions are overlapped. As a result, it achieves the worst performance. It can be shown that when using a feature reduction method for k NN, the performance is improved. PCA and SOM have close performance; however, the proposed WSOM obtains better classification accuracy than others. The classification performance with three SOM-based dimensionality reduction methods is displayed in Fig.5. It seems that the feature weighting method through k -means clustering algorithm does not get satisfied performance. It may suggest that k -means clustering algorithm perform poorly in the overlapped

dataset. However, the proposed method yields robust performance in the $I-4I$ dataset. Using a weighted distance function contributes to the performance of SOM. Table 2 lists the accuracy and classification time of six approaches on the $I-4I$ dataset. As can be seen, the proposed method achieves great performance than a traditional k nearest neighbor classifier, and it can accelerate the classification time.

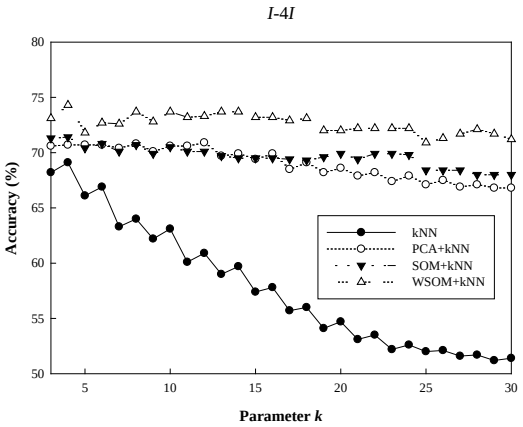


Figure 3: Average accuracy with varying parameter k among four methods.

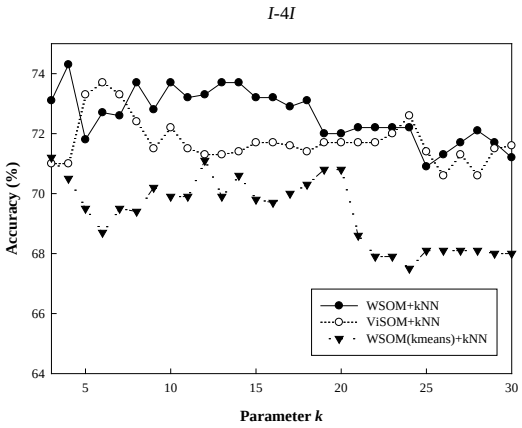


Figure 4: Average accuracy with varying parameter k among three SOM-based methods.

Table 2:Comparisons of six methods on the $I-4I$ dataset.

Method	*Accuracy (%)*	*Classification Time (ms)*
k NN	57.85	1985
PCA+ k NN	69.04	1725
SOM+ k NN	69.64	1606
WSOM(k-means)+ k NN	69.54	1615
ViSOM+ k NN	71.73	1622
WSOM+ k NN	**72.60**	1620

Similarly, Figure5 shows the accuracy rate as a function of k value of four methods in the $I-\Lambda$ dataset. The $I-\Lambda$ dataset is more complicated than the $I-4I$ dataset in terms of different means and variances in the feature space. In this case, the PCA obtains the worst performance because it cannot maintain the topological relationships in a lower dimensional space. Nerveless, both SOM and WSOM achieve good performance in the $I-\Lambda$ dataset on an account of topological preserving property. Figure6 shows the average classification accuracy with different k values among three SOM-based methods. It implies that the feature weighting method through k means algorithm and ViSOMis not useful to the complicated datasetwhile WSOM has robust performance with a wide range k of usage. Table 3 summarizes the classification accuracy and classification times of all approaches. The proposed method is efficient and obtains the best performance.

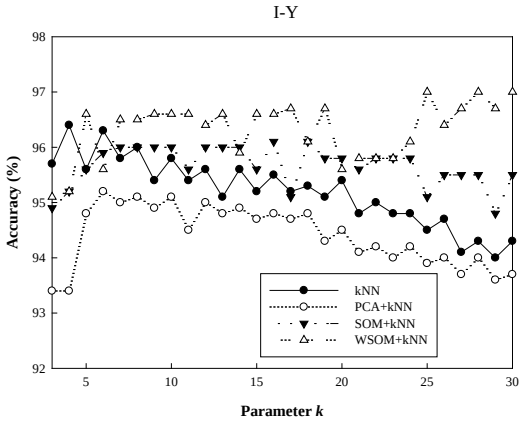


Figure 5: Average accuracy with varying parameter k among four methods.

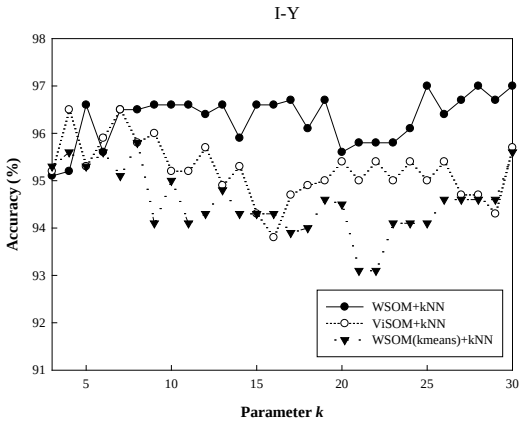


Figure 6: Average accuracy with varying parameter k among three SOM-based methods.

Table 3: Comparisons of six methods on the $I-\Lambda$ dataset.

Method	*Accuracy (%)*	*Classification Time (ms)*
k NN	95.20	2025
PCA+ k NN	92.40	1715
SOM+ k NN	95.66	1668
WSOM(k-means)+ k NN	94.55	1684
ViSOM+ k NN	95.22	1678
WSOM+ k NN	**96.31**	1681

4.3 Experiment on Real Dataset

Finally, two datasets are selected from UCI/STALOG machine learning corpora [32] for evaluation. The well-known Iris dataset contains 150 samples and 4 variables. The 150 samples include three classes and each class has fifty samples. This work applied the leave-one-out cross validation to measure classification performance since the training sample size in the Iris dataset is small. Another real dataset used in the study is Australian Credit Card dataset, which has 690 instances and 15 features. The Credit Card dataset includes two categories. Since some samples have a missing value, the study only considered 653 instances. We randomly chose 296 samples for learning, and 357 samples for classification. Table 4 shows the typical parameters of the SOM algorithm in these experiments. All features of real datasets were first normalized to the zero center mean and unit variance. The turning parameter λ is set to 0.075 in the Iris dataset and 0.6 in the Credit Card dataset in the ViSOM algorithm.

Table 4: The parameters of the proposed method used in the real datasets.

Dataset	*Parameters*
Iris	$\eta(0)=0.1, \eta_\alpha=0.96, \sigma_\beta=0.975$
Credit	$\eta(0)=0.3, \eta_\alpha=0.95, \sigma_\beta=0.95$

Figure 7 shows the average accuracy with different k parameters on the Iris dataset.The PCA method has the worst performances because it cannot preserve the topology using only two features. Figure 8 describes the performance of three SOM-based methods. It can be observe that the accuracy of WSOM method based on k means clustering algorithm is sensitive to different k values. However, WSOM and ViSOM get more stable performance. The classification time and average accuracy rate are shown in Table 5. It is apparent that all SOM-based methods achieve comparable performance The reason for the performance is that the number of features in the Iris dataset is small.

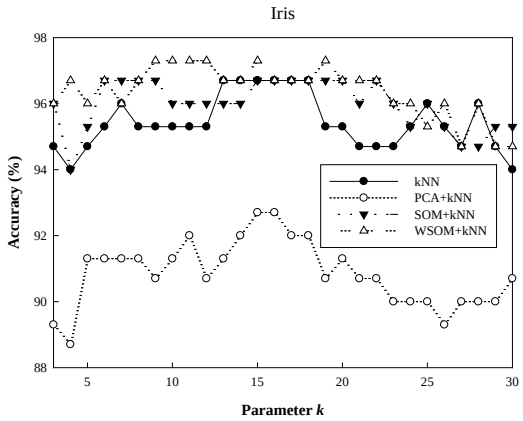


Figure 7: Average accuracy with varying parameter k among four methods.

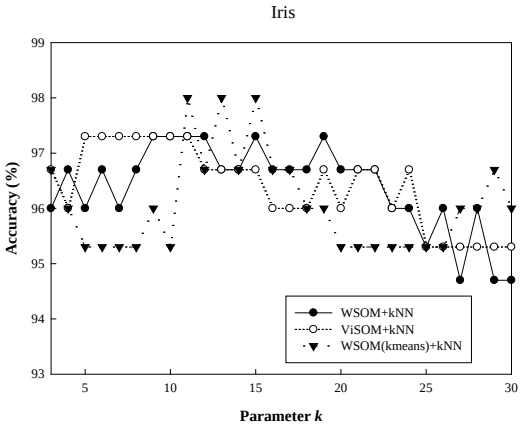


Figure 8: Average accuracy with varying parameter k among three SOM-based methods.

Table 5. The comparisons of three methods on the Iris dataset.

Method	Accuracy (%)	Classification Time (ms)
k NN	95.43	201
PCA+ k NN	90.86	189
SOM+ k NN	95.99	165
WSOM(k-means)+ k NN	96.05	171
ViSOM+ k NN	**96.40**	172
WSOM+ k NN	**96.39**	171

Similar to the previous cases, Figures 10 and 11 show the performance with different k parameters of all methods on the Credit Card dataset. The results show that the proposed WSOM method is superior to other feature reduction approaches such as PCA and SOM-based methods. According to the above experimental results, they show that the performance of k NN can be improved by a appropriate feature reduction method. The average accuracy rate and classification time of six methods are listed in Table 6. In theCredit Card dataset, the proposed WSOM gets the highest accuracy using only two features for classification. Thus the classification time of k NN is also reduced.

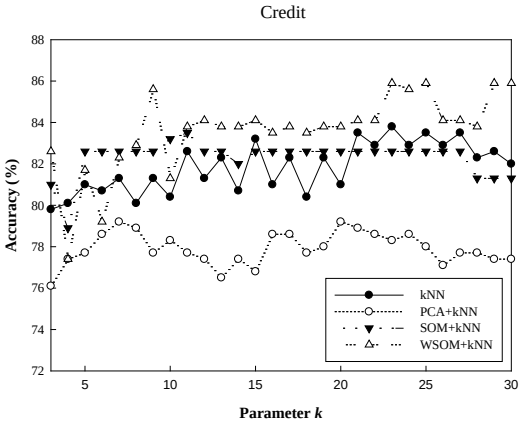


Figure 9: Average accuracy with varying parameter k among four methods.

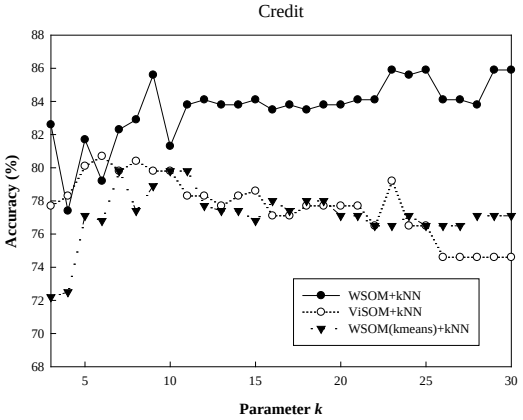


Figure 10: Average accuracy with varying parameter k among three SOM-based methods.

Table 6: The comparisons of three methods on the Credit dataset.

Method	Accuracy (%)	Classification Time (ms)
k NN	81.85	1330
PCA+ k NN	77.91	1021
SOM+ k NN	82.30	825
WSOM(k-means)+ k NN	77.15	832
ViSOM+ k NN	77.66	835
WSOM+ k NN	**83.59**	832

4.4 Discussions

In the experiments, we compared the performance among different feature reduction methods for k nearest neighbor classifier. The SOM algorithm plays the pivot in the study. SOM can preserve the topological relationships in a lower dimensional space, so it is a more suitable feature reduction method for k nearest neighbor classifier. In contrast, PCA gets poor performance using only two features. However, PCA can select more features in a high dimensional space. The proposed method may fail in the high dimensional space due to only two features for classification. Fortunately, feature subset selection or feature grouping (Liang *et al*. 2009) offers a solution for this problem. It is possible to develop a hierarchical feature reduction method based on WSOM for k NN in the high dimensional space for the future work.

The paper deal with the problem of SOM to find the winner steps often taken by Euclidean distance. The proposed feature weighting method is adapted from LDA algorithm. Therefore, the large weights are given to the more important features, and the lesser weights are offered to the smaller ones. In order to show the benefits of the proposed feature weighting method, we compare another method through k -means clustering algorithm. The evidence points out the weights of features affect the performance of SOM, and the proposed feature weighting method is more powerful.

To verify the effectiveness of the proposed method, we compared another revised version of SOM called ViSOM in the experiments. The ViSOM faithfully preservers the topology by adding a parameter to control the lateral forces between neurons. From the experimental results, it could be inferred that only adjusting the weights of neuron is not sufficient to improve the performance of SOM. Choosing the more accurate winners and then updating the weights of neurons are more useful to yield better results. In most cases, the proposed WSOM is an appropriate feature reduction method for k NN and speeds up the classification time.

5. Conclusion

In this paper, we present a revised self-organizing feature map using a weighted Euclidean distance metric as a dimensionality reduction method for k nearest neighbor classifier. We give weights to all features in terms of the ratio of between-class variance to within-class variance calculated from the training samples. It is similar to LDA approach. Thus the small weights are given to insignificant features, and large weights are offered to influential ones. Since SOM algorithm is topological preserving mapping, it is a suitable feature extraction method for k nearest neighbor classifier. This study compares the performance with other feature extraction methods such as PCA, SOM and ViSOM. The experimental results show that the weighted self-organizing feature map algorithm obtains the best performance. In addition, the detail comparisons of k -means clustering-based feature weighting methods are given. It is clear that the feature weighting method is more robust and effective in choosing winner steps. The future work is planned to develop a hierarchical feature reduction method based on weighted self-organizing feature algorithm in the a high dimensional space, and applied to face recognition, text classification, and image segmentation.

References

[1]. T. Kohonen, "The Self-Organizing Map," Proc. of the IEEE, Vol.78, No.9, pp.1464-1480, Sep 1990..

[2]. T. Kohonen., "The Self-Organizing Maps," 3rd edition, Berlin, Springer, 2001..

[3]. P. E. Hart., D. G. Stock., R. O. Duda, Pattern Classification, 2nd edition., Hoboken: NJ: Wiley, 2001.

[4]. J. L. Wu., and I. J. Li, "A SOM-based Dimensionality Reduction Method for KNN Classifiers," Intentional Conference System Science and Engineering, pp. 173-178., 2010.

[5]. A. Hinneburg, C. C. Aggarwal, D. A. Keim, "What is the nearest neighbor in high dimensional spaces? " Proc. of the 26th International Conference on Very Large Data Bases, pp. 506-515, 2000.

[6]. F. Korn, B. U. Pagel, C. Faloutsos, "On the 'Dimensionality Curse' and the 'Self-similarity Blessing'," IEEE Transations on Knowledge Data Engineering Vol. 13, No.1, 96-111, 2001.

[7]. H. Yin, "ViSOM-A Novel Method for Multivariate Data Projection and Structure Visualization," IEEE Transactions on Neural Networks, Vol. 13, No. 1, pp. 237-243, 2002.

[8]. S. Theodoridis, and K. Koutroumbas, Pattern Recognition, 3rd edition, Academic Press, 2006..

[9]. J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated Variable Weighting in k-means Type Clustering," IEEE Trans. Pattern Anal. Mach. Intel. ,Vol. 27, No. 5, pp. 657-668, 2005.

[10]. C. Domeniconi, P. Jing, and D. Gunopulos, "Locally Adaptive Metric Nearest-Neighbor Classification," IEEE Trans. Pattern Anal. Mach. Intel. , Vol. 24, No. 9, pp. 1281-1285, 2002.

[11]. T. Hastie, and R. Tibshirani, "Discriminant Adaptive Nearest Neighbor Classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 6, pp. 607-616, 1996.

[12]. R. Paredes and E. Vidal, "Leaning Weighted Metric to Minimize Nearest-Neighbor Classification Error," IEEE Transactions Pattern Analysis and Machine Intelligence, Vol. 28, No. 7, pp. 1100-1110, 2006.

[13]. Y. Wu, K. Ianakiev and V. Govindaraju, "Improved k-nearest neighbor classification," Pattern Recognit. , Vol. 3, No. 10, pp. 2311-2318, 2002.

[14]. W. Lam, C. K. Keung, and C. X. Ling, "Learning Good Prototypes for Classification Using Filtering and Abstraction of Instances," Pattern Recognition, Vol. 35, pp. 1491-1506, 2002.

[15]. C. J. Veenman, and M. J. T. Reinders, "The Nearest Subclass Classifier: A Compromise between the Nearest Mean and Nearest Neighbor Classifier," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 9, pp. 1417-1429, 2005.

[16]. F. Shena and O. Hasegawab, "A Fast Nearest Neighbor Classifier Based on Self-Organizing Incremental Neural Network," Neural Networks, Vol. 21, No. 10, pp. 1537-1547, 2008.

[17]. S. Garcia, J. Derrac, J. R. Cano, and F. Herrera, "Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical study," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 34, No.3, pp.417-134, 2012.

[18]. I. Triguero, J. Derrac, S. Garcıa, and F. Herrera, "A Taxonomy and Experimental Study on Prototype Generation for Nearest Neighbor Classification," IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Review, Vol.42, No.1, pp. 86-100, 2012.

[19]. A Bohlooli, K Jamshidi, "A GPS-free Method for Vehicle Future Movement Directions Prediction using SOM for VANET," Applied Intelligence, Vol.36, No.3, pp. 685-697, 2012 .

[20]. J. Mao and A. K Jain, "Artificial Neural Networks for Feature Extraction and Multivariate Data Projection," IEEE Transactions on Neural Networks, Vol. 6, No. 2, pp. 296-317, 1995,.

[21]. E. Berglund, "Improved PLSOM algorithm," Applied Intelligence, Vol.32, No. 1, pp.122-130, 2010.

[22]. R. Kamimura, "Structural enhanced information and its application to improved visualization of self-organizing maps,"Vol.34, No. 1, pp.102-115, 2011.

[23]. I. J. Li and J. L. Wu "A Fast Prototype Reduction Method based on Template Reduction and Visualization-Induced Self-organizing Map for Nearest Neighbor Algorithm," Applied Intelligence, Vol.39, No.3, pp. 564-582, 2013

[24]. Y. Sun, "Iterative RELIEF for Feature Weighting: Algorithms, Theories, and Applications," IEEE Transaction Pattern Analysis and Machine Intelligence, Vol. 29, No. 6, pp. 1-17, June 2007.

[25]. R. Gilad-Bachrach, A. Navot, and N. Tishby, "Margin Based Feature Selection—Theory and Algorithms," Proc. 21st Intentional Conference Machine Learning, pp. 43-50, 2004.

[26]. B. Chen and H. Liu, "Large Margin Feature Weighting Method via Linear Programming," IEEE Transaction on Knowledge and Data Engineering, Vol. 21, No. 10, pp. 1475-1488, October, 2009..
[27]. C. J. Veenman and D. M. J. Tax, "LESS: A Model-Based Classifier for Sparse subspaces," IEEE Transations. Pattern Analysis and Machine Intelligence, Vol. 27, No. 9, pp. 1496-1500, 2005.
[28]. D. R. Wilson and T.R. Martinez, "Improved Heterogeneous Distance Functions," Journal of Artificial Intelligence Research, Vol. 6, pp. 1-34, 1997.
[29]. C. Stanfill and D. Waltz, "Toward Memory-Based Reasoning," Communications of the ACM, Vol. 29, pp. 1213-1229, 1986.
[30]. H. Wang, "Nearest Neighbors by Neighborhood Counting", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, No. 6, 2006.
[31]. Y. Mitani and Y. Hamamoto, "A Local Mean-Based Nonparametric Classifier," Pattern Recognition Letters, Vol. 27, No. 10, pp. 1151-1159, 15 Jul 2006.
[32]. C. Blake, E. Keogh, and C. J. Merz, UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California, http://www.ics.uci.edu/~mlearn, 2009.
[33]. J. Liang S. Yang and Y. Wang, "An Optimal Feature Subset Selection Method based on Discriminate and Distribution Overlapping" Int. J. Pattern Recognit. Artif. Intell., Vol. 23, No. 8, pp. 1577-1597, 2009.

Jiunn-LinWu received the B.S., M.S. and Ph.D. degree in Electrical Engineering from National Cheng Kung University, Taiwan, in 1993, 1995, 2003 respectively. From 2002 to 2004, he was with the Graphic Division, Ulead systems, Inc., where he was a senior supervisory engineer. Since 2004 he has been with the Department of Computer Science and Engineering at National Chung Hsing University, Taichung, Taiwan, where he is currently an Associate Professor. His research interests include image processing, computational photography, pattern recognition and signal processing.

I-Jing Li received her Ph.D. in the Department of Computer Science and Engineering from National Chung Hsing University in 2013. Now, she is a senior engineer in Hermes Microvision Incorporation. Her researchinterests include image processing, pattern recognition and nonparametric classifier.