Journal of Computers Vol. 30 No. 2, 2019, pp. 1-13
doi:10.3966/199115992019043002001

Improved BP Classifier via Distance for Sample Reduction

Jia Wen[1,2,3], Jia Deng[1,2,3*], Peng-Fei Liu[1,2,3], Hong-Jun Wang[1,2,3]

[1] The Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Yanshan University, Qinhuangdao, China

[2] School of Information Science and Engineering, Yanshan University, Qinhuangdao, China

[3] State Key Laboratory of Software Engineering of Hebei Province, Yanshan University, Qinhuangdao, China
{17483540, 1351270663, 2353865452, 2642264392}@qq.com

Received 26 July 2017; Revised 31 October 2017; Accepted 9 January 2018

Abstract. The BP neural network algorithm is a commonly used technique in the classification field of pattern recognition, which iterates training samples continuously to obtain the decision boundary, until the error reaches the convergence precision. However, due to the extensive redundancy of training samples, this carpet-like training method can easily lead to slow network convergence and low classification performance. In this paper, from the view of sample, a novel BP training method called improved BP Classifier algorithm using Distance for Sample Reduction (DRBP) is proposed. Through the sample weight to get the new training sample, which are the near of the classification interface like support vector, and then iterative training BP classifier to solve the classification issue. The main function of this method is to keep the original sample information, realize the sample reduction, and improve the generalization ability of the BP neural network. First, point sets is used to visualize the training process. And then use 10 sets of the University of California Irvine (UCI) to verify the effectiveness and superiority of our experiments. Finally through Cross-validation, the classification error rate of 10 sets of UCI datasets tested by DRBP training method is lower than that of standard BP neural network (STBP). So the result of the experiment showed that the method of the combined distance sample reduce with BP network can significantly enhance the generalization ability of BP network.

Keywords: BP neural network algorithm, classification interface, sample reduce, support vector

1 Introduction

The BP neural network has the advantages of simple structure, strong parallelism, low computation and self-learning function in the past research of artificial neural network model. It has been paid more attention by many researchers and research institutions [1-3]. However, as to a good pattern classification system, in addition to having a good classifier, good data set is also important. As we known, most of the original data sets contain many redundant samples, which not only meaningless to the training of classifiers, and their presence will consume a large amount of computing time, thereby reducing the whole convergence speed of classifier training. So through the data set, some methods about data reduction improving the classification performance has been studied by many researchers [4-6].

K. Nikolaidis and J. Y. Goulermas proposed a boundary protection algorithm named a class boundary preserving algorithm for data condensation (CBP) [7]. This method divided all data into boundary data and non-boundary data. The main idea of this method is to protect the boundary data and remove the non-boundary data. Although CBP has reduced the data set to a large extent, the complete removal of non-boundary samples can result data set information loss.

* Corresponding Author

In order to obtain a better classification effect, Lei Yu et al. proposed a sample method based on sample weight, named Stable Gene Selection from Microarray Data via Sample Weighting (SW) [8]. Actually, this idea of based on sample weight has proposed long before. Such as the condensed nearest neighbor rule (CNN) [9], the reduced nearest neighbor rule (RNN) [10], and an algorithm for the selective nearest neighbour decision rule (SNN) [11]. However, all these methods suffers from several drawbacks such as large computational complexity, low classification efficiency and poor ability of noise tolerance. In order to overcome these problems, many research programs have been proposed by researchers in recent years.

The more typical methods is the prototype selection of the nearest neighbor classification proposed by Garcia et al. [12], which cited a number of nearest neighbor data reduction classification algorithm articles. It is a general summary of the previous classification based on the proximity method. There are also some support vector machine (SVM) methods have been proposed, such as Tsai and Che-WeiChang proposed Support Vector Oriented Instance Selection for text classification (SVOIS) [13], and Chen et al. proposed Fast instance selection for speeding up support vector machines [14]. This SVM methods is based on the concept of classification boundary, and select the support vector from the original training samples. So that the division of the support vector is equivalent to the division of the whole sample data. But this method only consider the support vector, while ignoring most of the samples, and similar to the border protection law that mentioned above, it is also very easy to cause data sets information loss and cause over-fitting.

Last year, it was suggested that a new kd-tree approach to data reduction [15]. However, when the data set is too large, the structure of this method is too complicated, easily lead to over-fitting problems. The recently published method of dynamic data reduction (DDR) [16] is also an algorithm based on classification boundaries, but the method of random reduction data is used in this paper. Although the speed of convergence of the network will be improved to some extent, some samples that play an important role in classification can be removed at the same time, resulting in the lack of information on the data set and impacting on the classification performance.

In the past, the goal of training BP neural networks is to obtain the minimum value of the cost function by training a large amount data set. In this paper, a novel instance selection method for BP classifier, called improved BP Classifier via Distance for Sample Reduction (DRBP) is proposed. DRBP mainly borrows the idea is merged the data reduction and BP classifier training into a stage. Specifically, before the network training, through published the correct classified samples, and the same time, rewarded the misclassified samples, obtaining redundant samples from original samples, thus removed the redundant samples. And then the BP network is trained with reduced samples until the cost function converges to a predetermined minimum.

Unlike the previous of boundary protection algorithm, the DRBP attempts to obtain the boundary samples by dynamic update sample weight, and this classifier achieves the ability to improve the generalization of the network, on the basis of the sample reduction. Compared with the amount of calculation of this computation which the distance between all nuclear sample and all boundary sample [17] and a method Adaptive Pseudo Nearest Neighbor (APNN) [18], if the data set has N categories, the classifier can reduce the amount of computation by N times.

2 BP Classifier Based on Sample Reduction

2.1 The Basic Principle of BP Classifier

When the BP neural network is used to deal with the classification problem, the quality of the sample data directly affects the training rate and the network performance. In general, there are at least three criteria for measuring good data sets: (1) facilitating classifier training; (2) minimizing redundant data; (3) ensuring network training efficiency.

The standard three-layer BP neural network is composed of input layer, hidden layer and output layer. As shown in Fig. 1, first, perform a feedforward pass, computing the activations for each layers. Next, perform backpropagation, computing the error term for each layers. Finally, calculating the weight gradient, and then update the weight. Iterate above steps until the cost function values converge.

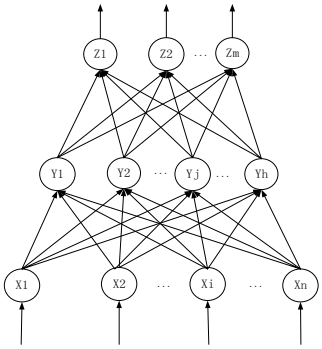

Fig. 1. Schematic diagram of a standard BP network [19]

Although the standard BP neural network self-learning ability is very strong, for the whole classification performance, because of the sample data may have some problems, the generalization ability of the network still needs to be improved.

As shown in Fig. 2(a), this is a simple classification data sets distribution. It is clear that most of the central cluster samples are not necessary as training samples for the next iteration, only a few samples near the classification interface are valuable. When the data sets are too large or too concentrated, that is, the data fluctuation is small. If these sample sets are directly used as training sets, not only the classification interface of the network is oriented towards the centers of the classes, so that the error rate would increase, and reduce the overall generalization ability, but also increased the training time of the network.

As shown in Fig. 2(b), this is the classification interface distribution of the data sets. The process of training the BP network is to adjust the process of training the classification interface in the yellow, green and blue regions of above the figure, aim at the three classes samples of black, blue and red are maximum distribution on both sides of the classification interface. Obviously, these nuclear samples that far away from the classification interface is easily classified into the correct classes, and these boundary samples that near the classification interface is easily classified into the error classes. In addition, we can also find that these nuclear samples accounts for the majority. However, these nuclear samples have little effect on the adjustment of the classification interface, and the sample which really affects the classification interface is boundary samples, which are easily misclassified and near the classification interface. When training the classifier, more boundary samples and less nuclear samples, the network training time and network generalization ability both improved significantly. In the early training network, the optimization of the sample sets is particularly important.

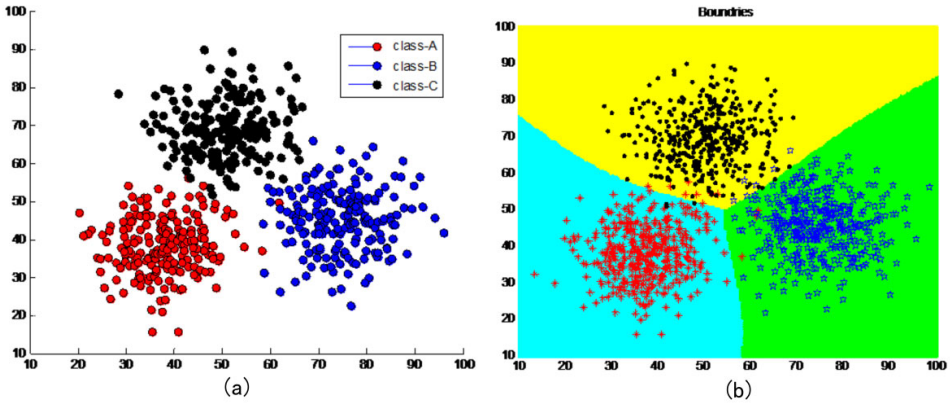

Fig. 2. Sample and classification interface distribution

2.2 The BP Classification Model Based on Sample Sets

Based on the above principles, this paper combines sample reduction with BP classifier. Filters samples before training the BP classifier, that is, removes the redundant samples to achieve the purpose of sample reduction. Fig. 3 is the Algorithm 1: DRBP algorithm.

Algorithm 1 :DRBP algorithm

1. Input: train data D, the maximum sample weight wt, the minimum sample weight wb, the sample weight increment value dw.

2. Initialize:the weight of training samples W: $W=W_{ini}=0.5$

3. Do for $i=1,\cdots,T$

3.1)use the train data to train BP network.

3.2) Calculate the mean error of the BP network.

$$e_i=\frac{1}{T}\sum_{j=1}^{T}(f(x_j)-y_j)^2, i=1,2...,T$$

3.3) Update the weight of the training samples**.**

$$W_i \leftarrow W_i - dw, f(x_i)=y_i$$

$$W_j \leftarrow W_j + dw, f(x_j)\neq y_j$$

3.4) The weights of the training samples normalized.

$$W=\max(W, wb)$$

$$W=\min(W, wt)$$

3.5) Run *algorithm 2*, perform sample reduction.

4. Output: a strong BP classifier.

Fig. 3. DRBP algorithm

The operation is as follows: First, give each original sample an equal initial weight, and according to the sample weight for subsequent new sample selection. Initialize the original sample into the new training sample. Next, training the BP network with the new training sample and then testing the trained network with the original sample, through the output of the incorrect classified samples and the correct classified samples, dynamically update the sample weight. After updating the sample weight, the sample weight is required to be normalized. Then, according to the size of the sample weight, all the original samples are divided into the boundary samples and the nuclear samples. In order to reduce the data fitting, except all the boundary samples are taken as new training samples for the next iteration, the nuclear samples near the boundary samples are also used as new training samples. Finally, the process is repeated until the output error rate reaches the convergence accuracy.

2.3 Sample Reduction Based on the Distance

The above section describes the entire BP Classification Model Based on Sample Sets. Next, the main question in this section is how to remove the redundant samples from the previous iteration sample to achieve the purpose of sample reduction. That is, this section is aims to describe the process of sample reduction in detail. In this section, the sample reduction is divided into two parts. One is how to obtain new training sample, which is the whole framework of the sample reduction. The other is nested under the framework of the sample reduction, how to obtain the nuclear sample near the boundary sample. That is, apart from all the boundary samples, find the remainder of the new training sample.

2.3.1 How to Get the New Training Samples X_{new}

The process of obtaining the new training sample is the process of reducing the sample. That is, the process of eliminating the redundant samples. At the beginning of the iteration, because the BP network is randomly generated, and the result of classification is not ideal. At this time, boundary sample occupies the majority, and then set a threshold b, when the number of the boundary samples X_{bad} is greater than this threshold, all the boundary samples directly act as the new training samples X_{new} of the next iteration.

$$X_{new}=X_{bad}, \left|X_{bad}\right|\geq b\ . \tag{1}$$

Because the boundary samples are the most valuable new training samples, so the boundary samples are the best new training samples when selecting new training samples. However, as the iteration increases, the boundary samples will be less and less, too less training samples can lead to over-fitting of

the network. Therefore, when the number of the boundary sample is less than the previously set the threshold *b*, if only use to the boundary samples as the new training samples of next iteration will cause the network turbulence. In order to prevent this phenomenon, at this point, on the basis of the use of all the boundary samples, must be according to the size of the European distance, select the appropriate amount of the nuclear samples near the boundary samples together as the new training sample of the next iteration.

$$X_{selects} \leftarrow dis\tan ce\_reduction(X_{bad}, X_{good}). \tag{2}$$

$$X_{new} = X_{bad} \cup X_{selects}, \left|X_{bad}\right| < b. \tag{3}$$

Of course, we also takes into account a situation that the distance threshold is dynamically expanded if the number of new training samples still does not meet the minimum.

$$D_{good \leftrightarrow bad} = D_{good \leftrightarrow bad} + dD. \tag{4}$$

Saving the distance from the boundary sample to the nuclear sample by classes, and then, through the distance, obtained the nuclear samples near the boundary sample as the new training samples. If the total number of new training samples is still very small, dynamically expand the distance threshold $D_{good \leftrightarrow bad}$, so that more nuclear samples near the boundary samples are selected, and to meet the requirements of more than the minimum training sample. Finally, there is a situation that is easy to overlook, when the classification interface can be completely classify each sample, immediately stop training, otherwise it will increase the unnecessary training time. Fig. 4 is the Algorithm 2.

Algorithm 2: sample reduction algorithm

1. Input: the boundary sample X_{bad} ,the boundary samples quantity threshold *b* ,the new samples quantity threshold *t*, the distance threshold $D_{good \leftrightarrow bad}$,the distance threshold increment *dD*.

2. Calculate the number of boundary samples， save as *n*.

3. Do for each X_{bad}

3.1) if n=0

then exit.

3.2)else *n*>*b*

then all boundary samples are used as new training samples.

$$X_{new} \leftarrow X_{bad}$$

3.3)else

run *algorithm 3*, to get the selected samples.

get the new training samples.

$$X_{new} \leftarrow X_{bad} \cup X_{selects}$$

4. Calculate the number of the new samples， save as *m*.

5. If m<t

then dynamically expand the distance threshold.

$$D_{good \leftrightarrow bad} \leftarrow D_{good \leftrightarrow bad} + dD$$

6. output: the new training samples X_{new}.

Fig. 4. Sample reduction algorithm

This algorithm is a process of obtaining the new training sample. For each training, if the boundary sample is empty, that is, there is no misclassified sample, indicating that the classifier has been able to classify the sample completely, so at this time, need to jump out of the cycle. If the number of boundary samples exceeds a relatively large threshold, all boundary samples will be used as the new training samples. As the training iterations continue, the boundary samples will gradually decrease. When the

boundary samples are smaller than the thresholds, the Algorithm 3 is run (Fig. 5). Get the nuclear samples near the boundary samples, and then all the boundary samples together with it formed the new training samples.

Algorithm 3 :Sample selection algorithm

1.Input: the boundary samples X_{bad} , the nuclear samples X_{good} , the distance of between nuclear and class centroid $D_{good\leftrightarrow ave}$, the distance of between nuclear and boundary $D_{good\leftrightarrow bad}$.

2.Calculate the centroid for each class.

$$ave_l = \frac{\sum_{i=1}^{m} 1\{S=l\}x^i}{\sum_{i=1}^{m} 1\{S=l\}}$$

3.If $\exists\{X_l \in X_{bad}\}$

3.1) Calculate the distance of all nuclear samples to the class centroid.

$$D(X_{good}, ave) = \sum_{i=1}^{m}\sum_{k=1}^{S}\left\|x^i - ave_l\right\|^2$$

3.2) Sort in ascending order of distance $D=\{d_1, d_2, ...d_n\}$.

3.3) Obtain nuclear samples which it`s distance smaller than the threshold $D_{good\leftrightarrow ave}$ as selected samples.

4.Else:

4.1)for each category:

4.1.1) Calculate the distance of all nuclear samples to the all boundary samples .

$$D(X_{good}, X_{bad}) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left\|x_{k^{(i)}} - b_{k^{(j)}}\right\|^2$$

4.1.2) Sort in ascending order of distance $D=\{d_1, d_2, ...d_n\}$.

4.1.3) Obtain nuclear samples which it`s distance smaller than the threshold $D_{good\leftrightarrow bad}$ as selected samples.

5.Output: the selected samples.

Fig. 5. Sample selection algorithm

2.3.2 How to Obtain the Selected Samples X_{select} in Nuclear Samples

The selected samples X_{select} is part of the new training sample, which is determined by the Euclidean distance between the boundary samples and the nuclear samples in the same class. First, calculate the centroid for each classes.

$$ave_l = \frac{\sum_{i=1}^{m} 1\{S=l\}x^i}{\sum_{i=1}^{m} 1\{S=l\}} . \quad \textbf{(5)}$$

Here the ave_l indicates the centroid of the class *l,S* represents the total number of classes, *m* is the number of samples, and x^i is the sample *i.*

And then select the appropriate selected samples by performing the following cycle: The specific selection process is divided into two cases: (1) When all the samples of some classes are boundary samples, or when the few of samples are nuclear samples while the vast majority of samples are boundary samples in some classes, calculate the Euclidean distance from the nuclear samples to each class centroid.

$$D(X_{good}, ave) = \sum_{i=1}^{m}\sum_{k=1}^{S}\left\|x^i - ave_l\right\|^2 . \quad \textbf{(6)}$$

Where x^i is the sample *i* in X_{good} , and *m* represents the number of nuclear samples. The results only retained the nearest Euclidean distance from the every nuclear samples to the centroid of each class. And

then, set a threshold $D_{good\leftrightarrow ave}$, take the X_{good} which it`s Euclidean distance less than the threshold $D_{good\leftrightarrow ave}$ as X_{select}. (2) The second case is the most common case, that is, after iterating many times, most of the samples are nuclear samples, and the boundary samples are basically distributed in the vicinity of the classification interface. At this time, directly calculated the Euclidean distance between all the nuclear samples and all the boundary samples.

$$D(X_{good}, X_{bad}) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left\| x_{k^{(i)}} - b_{k^{(j)}} \right\|^2 . \quad (7)$$

Where $x_{k^{(i)}}$ represents the nuclear sample *i* belongs to class *k*, $b_{k^{(j)}}$ represents the boundary sample *j* belongs to class *k, m* represents the number of nuclear samples, *n* represents the number of boundary samples, one nuclear sample X_{good} retains only the nearest distance to all boundary sample X_{bad}. And then set a threshold $D_{good\leftrightarrow bad}$, take X_{good} which it`s Euclidean distance less than the threshold $D_{good\leftrightarrow bad}$ as X_{select}. The following is the Algorithm 3.

The algorithm first calculates the centroid for each classes for later use. For each sample, if there are certain classes of all samples are belongs to the boundary samples, or most of the samples are boundary samples. In this case, calculate the distance of all nuclear samples to the class of centroids, and sort by ascending order of distance, obtain nuclear samples that smaller than the distance threshold $D_{good\leftrightarrow ave}$ as selected samples. If the boundary samples and the nuclear samples are evenly distributed in each classes, for each class: calculate the distance of all nuclear samples to the boundary samples, and each nuclear sample only retains a nearest distance. The distance of all retained nuclear samples are sorted by distance ascending order, Obtain nuclear samples that smaller than the distance threshold $D_{good\leftrightarrow bad}$ as selected samples.

Finally, using the selected samples and all the boundary samples to form a new training sample, and iterate step4 of the BP classification algorithm based on the datasets until we train a BP classifier with strong classification ability.

After trained BP classifier, test it by the test sets and get the classification results.

3 Experimental Analysis

In this section, we will explain the experiment in detail. By compared with the standard BP neural network, shows the superiority of our algorithm. This section is divided into four sections. First, in Section 3.1, we describe the preparation of the platform, datasets, language, etc. And then in Section 3.2, through the visualization, we prove the validity of our algorithm on the point sets. In section 3.3, we shows the validity of our algorithm on the official UCI classification datasets. Finally, in Section 3.4, we gives a concise description of our algorithm and a vision for our future development.

3.1 Experimental Environment and Data Set

Our experiment has two data sets. One is the point sets made by ourselves, which the role is to facilitate visualization. The other is UCI, the data sets can be downloaded directly from the Internet, the link address: Http://archive.ics.uci.edu/ml/datasets.html. Through this classification data sets, to prove the superiority of our algorithm.

In order to achieve the visualization of the experimental results, we set up the data sets Data_Point. The Data Point has three classes, each class has 400 samples, so the data sets has 1200 samples. Shown in Fig. 2, the nuclear samples of each category in the data set are relatively concentrated, and the boundary samples are relatively sparse. Because the UCI datasets has many types of data sets, so in our experiment, we did not use all UCI datasets, only randomly selected part of the data sets to test.

All the data sets in this experiment are trained on the shallow neural network. The experiment on the computer configuration requirements are not high. The use of the environment is Matlab and windows, operating conditions are easy to meet.

3.2 Data Point Datasets Experimental Analysis

The main idea of this experiment is to combine the sample reduction with the classifier into a whole. During each iteration, the training samples are dynamically updated, and the easily classified samples and the difficultly classified samples are treated differently. In the data reduction, the main idea is based on the Euclidean distance reduction, according to the size of the European distance, select the nuclear sample near the boundary sample. The selected nuclear samples were combined with all the boundary samples to form new training sample, and the BP network was trained with the new training samples to complete the classification problem.

The main parameters of the experiment are as follows: training samples *train_x* and test samples *test_x* are both 600, the network structure is 2-3-3, the number of network iterations is 500 times, the sample weight range $xw_i \in [0.2,1]$, The threshold distance $D_{good \leftrightarrow bad}$ between the nuclear sample and the boundary sample is 8, The threshold distance $D_{good \leftrightarrow ave}$ between the nuclear sample and mean of the boundary sample is 10, the threshold distance incremental coefficient dD=2.

In this experiment, through the sample weight, distinguish between the nuclear sample and the boundary sample. When the sample weight is equal or greater than 0.5, the sample is classified as a boundary sample, when the sample weight is less than 0.5, the sample will be classified as the nuclear sample. The entire sample weight is divided into four ranges: [0.2, 0.4], [0.4, 0.6], [0.6, 0.8], and [0.8, 1.0], the weight of the different ranges is represented by four colors of green, yellow, pink and red. Through several iterations training, training sample weight distribution map shown in Fig. 6.

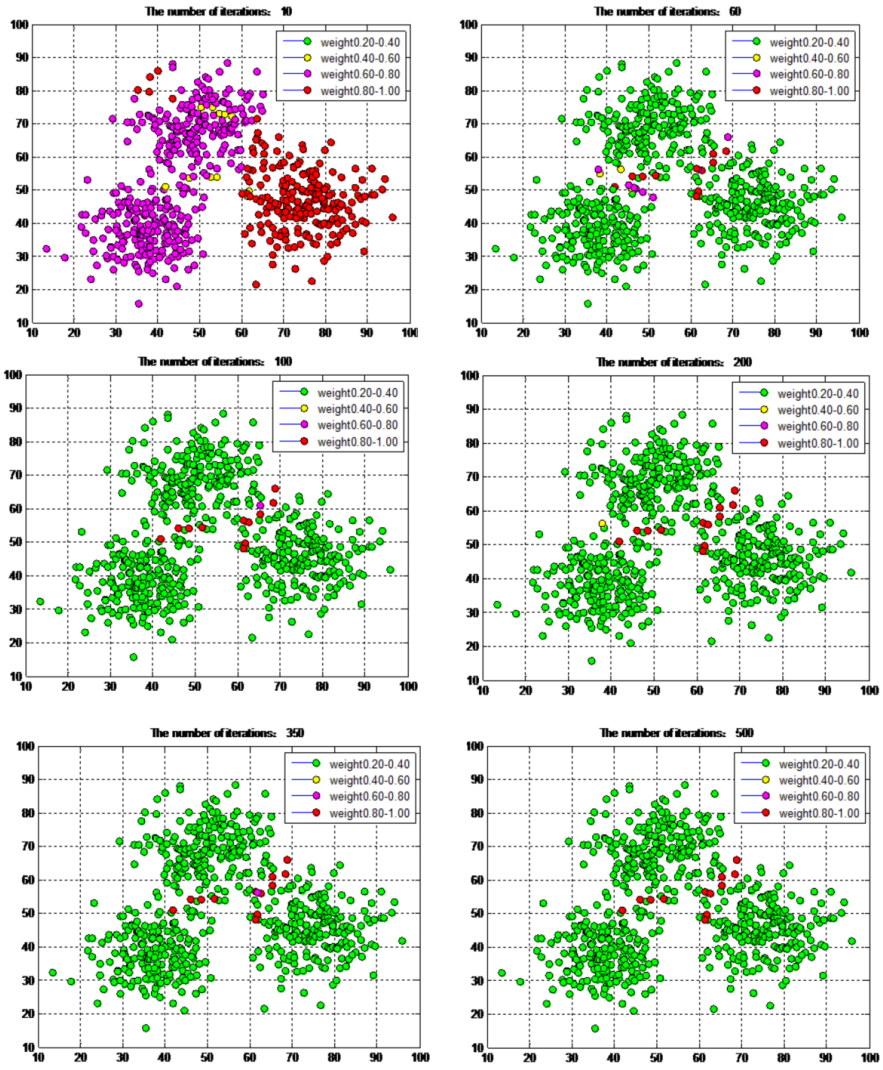

Fig. 6. Sample weight distribution map

As shown in Fig. 6, with the number of the iterations increases, the green samples are getting more and more, and most of them are distributed in the center of the class. The yellow and pink samples which have the middle size weight basically disappeared at the end, and the red samples which has the largest weight gradually decreased, besides, the distribution gradually moved closer to the middle boundary. This shows that as the number of iterations increases, the sample weights are gradually polarized, and more samples are classified, while the difficultly classified samples are getting less and less. What`s more,the difficultly classified samples are generally in the vicinity of the classification interface. So that, in each iteration training, each sample weight map corresponds to a sample selection map. Fig. 7 is the new sample selection map which is diagram consistent with Fig. 6. Among them, the red sample, the black sample and the blue sample represent different classes of samples, and the yellow sample represents the new training sample.

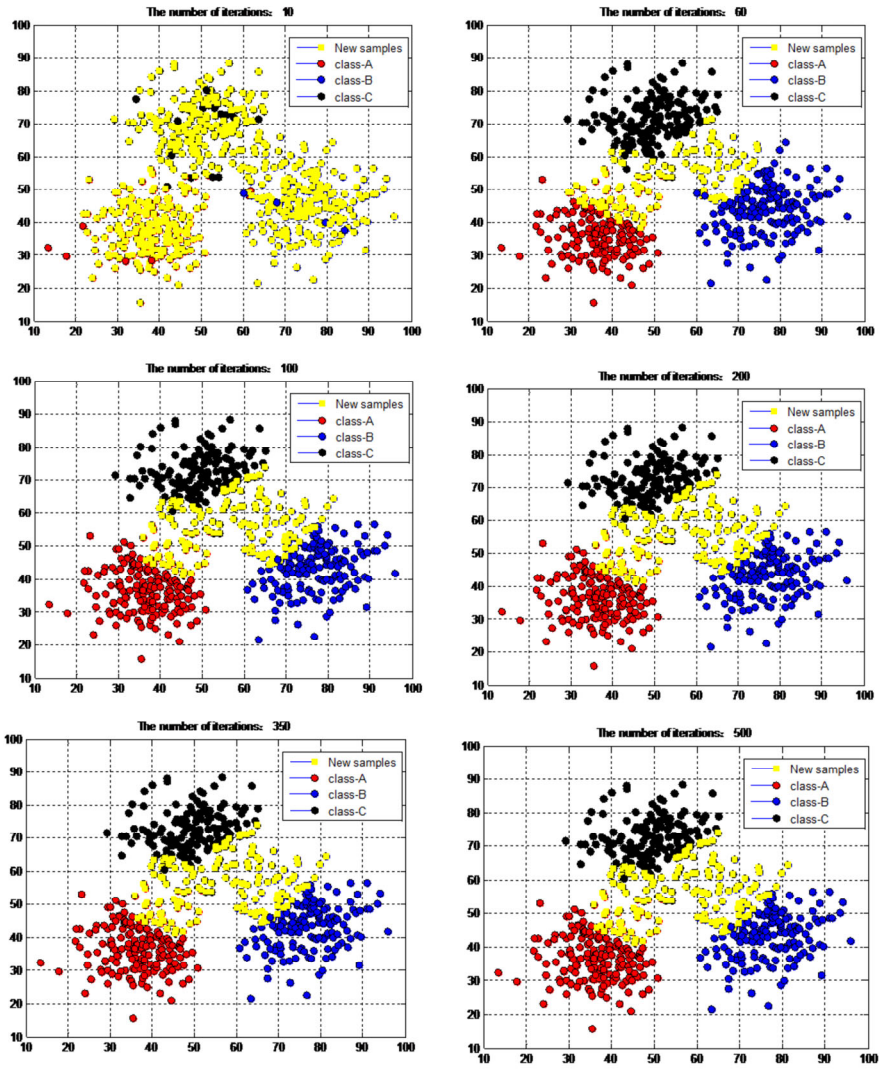

Fig. 7. New sample distribution map

As shown in Fig. 7, when iterated to the 10th time, the selected yellow samples are the majority, and then, with the number of iterations increases, the yellow sample gradually decreases and tends to stable. At the same time, the yellow samples have been moving closer to the middle boundary position, which is the place distributed by some sample with larger weighs in Fig. 6. This shows that in the early iteration, the classification of the network is very low, and there are many samples that are difficult to classify. Therefore, there are many samples that need to be newly trained. Later, with the number of the iterations increases, the classification of the network is improved and the number of samples that need to be selected as retraining is reduced.

With the number of the iterations increases, the difficultly classified samples will be decrease, in order to prevent the phenomenon of the network over-fitting, we must dynamically change the distance threshold to increase the number of selected samples, thereby, ensuring the stability of the network. Fig. 5 shows the whole iteration process, and comparison of the number of new samples and the number of classification difficult samples, in which the red curve shows the new training samples, and the blue thick curve shows the difficultly classified samples.

It can be seen from Fig. 8. On the one hand, at the beginning of the iteration, there are so many difficultly classified samples, so there are so many new training samples, however when iterated to 50th times or so, the number of these two samples tends to be stable. On the other hand, our data set as shown in Fig. 2(a), the entire data set is not linearly separable, that is, this data set is noisy. In general, this noisy data set in the iterative process is very easy to have local fluctuations, that is, there will be local turbulence. However, from Fig. 8, as the iteration continues, the number of new training samples and the number of classification difficult samples does not appeared in the case of periodic turbulence, indicating that this training method through DRBP not only converges faster but also reduces the periodicity fluctuations make the network more stable.

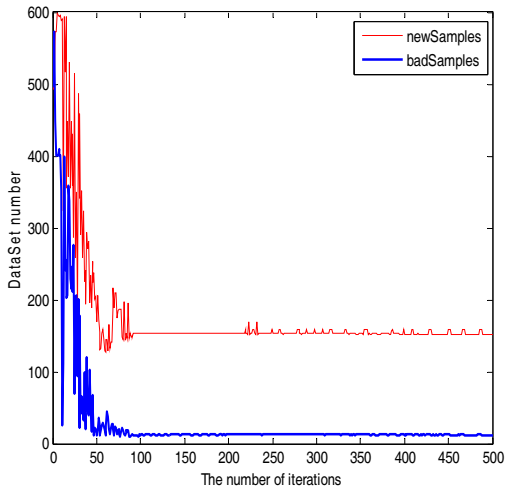

Fig. 8. Comparison of the number of new samples and the number of difficult samples

To evaluate the performance, we have trained the BP with three difference algorithms in point datasets, and drawn the curve of test error versus the number of iterations, corresponding to the average of the resultant classifiers, as shown in Fig. 9. These result suggest that DRBP can perform the better effectiveness than CBP and DDR. Or say, a classifier BP with DR processing have a higher detection accuracy and a faster convergence rate than ones with other data processing.

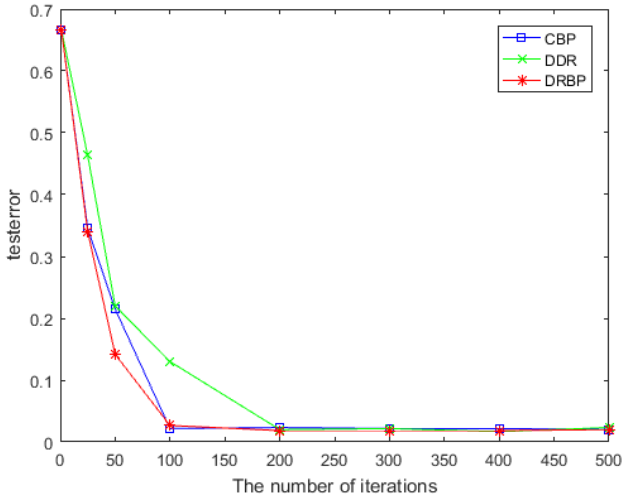

Fig. 9. Test error rate

3.3 UCI Datasets Experimental Analysis

Through the experimental analysis of the Data_Point datasets, we can visually see the whole process of the distance-based sample reduction algorithm. And now,verify the feasibility of our algorithm through the official UCI datasets. Selected 10 sets of UCI datasets to experiment, and the data used are shown in Table 1.

Table 1. Data set properties

Data Set	Number of train samples	Number of test samples	Feature dimension	Number of classes
Glass	105	105	7	3
Iris	50	50	4	3
Lir	10000	9993	16	26
Seeds	105	105	7	3
WF	2500	2499	21	3
Wine	89	88	13	3
Forest	260	263	27	4
IP	90	89	34	2
CMC	700	773	9	3
Diabet	600	600	19	2

In this experiment, we recorded the classification results of Standard BP (STBP) and BP Classifier via distance for sample reduction (DRBP) under the same experimental conditions. The classification results are reflected from four aspects, Mean square error (loss), Training set average error rate (train_avg), testing set average error rate (test_avg) and running time (time). Each group of experimental data is taken from the average of 10 groups of training results, the specific experimental results in Table 2.

Table 2. Classification of different training methods

Data Set	STBP				DRBP			
	Loss	Train_avg	Test_avg	Time	Loss	Train_avg	Test_avg	Time
Glass	0.0416	5.71	6.67	0.95	0.0368	2.86	4.76	1.69
Iris	0.0474	2.00	2.02	0.82	0.1030	0.00	0.03	1.37
Lir	0.1446	18.21	20.10	493.03	0.1426	15.38	17.29	527.48
Seeds	0.0317	2.86	5.71	2.67	0.0283	0.00	2.86	4.61
WF	0.0844	11.36	14.25	32.48	0.0845	11.32	13.85	51.44
Wine	0.0045	0.00	2.27	0.87	0.0193	0.00	0.01	0.39
Forest	0.0289	2.69	16.35	13.63	0.0438	2.69	12.55	20.66
IP	0.0503	5.56	33.71	1.89	0.0887	6.67	31.46	1.58
CMC	0.2343	35.57	44.89	13.36	0.2416	35.71	44.76	15.24
Diabet	0.3586	53.17	52.99	2.33	0.4995	46.83	47.01	2.88

From Table 2, using DRBP trained results is greater than using STBP trained results at almost the same circumstances. Because training classifier with DRBP, only the nuclear samples near the boundary samples and the boundary samples are trained in every iteration, and most of the redundant nuclear samples are not used. However, even in this case, when comparing the error rates of the two training methods, the test set average error rate use DRBP trained both lower than the test set average error rate use STBP trained. And except for IP data set, DRBP train set average error rate is also lower than the STBP train set error rate. Indicating that the DRBP training method is practicable. In other words, this training method, which focuses on the boundary samples and weakens the nuclear samples away from the classification interface is practicable. Experimental data show that this DRBP training method does improve the generalization ability of the network.

3.4 System Assessment and Future Outlook

The BP neural network classification algorithm based on distance sample reduction (DRBP), distinguishes between the boundary samples and the nuclear samples, through rewarding the sample weight of the classification wrong and punishing the samples weight of classification true. The samples

which have large weights as boundary samples and the samples which have small weights as nuclear samples. And then focus on training the sample near the classification interface, weakening the role of the nuclear sample which away from the classification interface. Using point sets visualization experiments and using UCI data sets shows that this DRBP training method is a kind of BP neural network training method with better classification and performance. However, in the selection of nuclear samples near the boundary samples, although the distance between all the nuclear samples and the boundary samples is calculated synchronously according to the different classes, the computational complexity has been reduced on the basis of guaranteed efficiency. But this is needed to be optimized, and follow-up studies can focus on finding a more concise approach to preserving all nuclear samples near the boundary sample.

4 Conclusion

In this paper, we mainly achieve distance based sample reduction. That is, focus on the protection of the sample near the classification interface. First, the nuclear samples near the boundary samples are selected by distance, and then using the selected nuclear samples combined with all the boundary samples to form new training samples. This method of screening new training samples can be done on the basis of protecting the vast majority of sample information, eliminating the vast majority of redundant nuclear samples. Through the point set visualization, we can visually clearly see the whole sample reduction process, besides, we compare DRBP with other data reduction method in point sets. Then we use the UCI data set to compare the experimental results of STBP with DRBP, to verify that the proposed method has the advantages of fast convergence and high precision.

In the future work, we plan to develop additional sample reduction algorithms under the general framework and study their effects on different sample selection methods. The main idea can be from two directions, one is combined with other classifiers. Since the sample reduction is not limited to the field of classification, but also can be applied to the fields of regression and clustering as well. In order to apply the method of data reduction to a wider area, these sample selections with different pattern recognition algorithm need to be extended. The other is find a better method of data reduction based on this framework. This question should be attributed to how to remove redundant samples from the original sample faster and more efficiently, and how to find redundant samples. Therefore, this general direction, subsequent studies can be expanded around the adaptive tuning of the reduction parameters in this framework, that is, simplify the parameters in the process of sample reduction of BP neural network, thus speeding up the convergence speed.

Acknowledgements

Fund Project: (020000532) Yanshan University Independent Youth Fund Project, Research on Fast Pedestrian Detection Based on Weight Deformation Part; (15210122) Hebei Science and Technology Research and Development Program Science and Technology Support Program, Research on Some Key Technologies of Activity Perception Based on Computer Vision.

References

[1] S. B. Kotsiantis, Supervised machine learning: a review of classification techniques, Informatica 31(2007) 249-268.

[2] A. Georgieva, I. Jordanov, Intelligent visual recognition and classification of cork tiles with neural networks, IEEE Transactions on Neural Networks 20(4)(2009) 675-685.

[3] X. Zhang, J. Huenteler, Classification recognition algorithm based on strongAssociation rule optimization of neural network, Telecommunication Computing Electronics and Control 14(2)(2016) 241-247.

[4] Y. Zhou, A. Zhu, X. Qian, A sample data selection method for neural network classifier, Journal of Huazhong University of Science and Technology 40(6)(2012) 39-43.

[5] K. Hara, K. Nakayama, A training method with small computation for classification, in: Proc. Ieee-Inns-Enns International Joint Conference on Neural Networks IEEE Computer Society, 2000.

[6] P. Jia, C. Zhan, Z. He, A new sampling approach for classification of imbalanced data sets with high density, in: Proc.

International Conference on Big Data and Smart Computing, 2014.

[7] K. nikolaidis, J.Y. Goulermas, Q.H. Wu, A class boundary preserving algorithm for data condensation, Pattern Recognition 44(3)(2011) 704 -715.

[8] L. Yu, Y. Han, M.E. Berens, Stable gene selection from microarray data via sample weighting, IEEE/ACM Transactions on Computational Biology and Bioinformatics 9(1)(2012) 262-272.

[9] B.P.E. HART, The condensed nearest neighbor rule, IEEE Transactions on Information Theory 14(5)(1968) 515-516.

[10] G.W. Gates, The reduced nearest neighbor rule, IEEE Transactions on Information Theory 18(3)(1972) 431-433.

[11] G. Ritter, H. Woodruff, S. Lowry, An algorithm for a selective nearest neighbor decision rule, IEEE Trans on Information Theory 21(6)(1975) 665-669.

[12] S. García, J. Derrac, J.R. Cano, Prototype selection for nearest neighbor classification: taxonomy and empirical study, IEEE Trans on Pattern Analysis and Machine Intelligence 34(3)(2012) 417-435.

[13] C.F. Tsai, C.W. Chang, SVOIS: support vector oriented instance selection for text classification, Information Systems 38(8)(2013) 1070-1083.

[14] J. Chen, C. Zhang, X. Xue, Fast instance selection for speeding up support vector machines, Knowledge-Based Systems 45(2013) 1-7.

[15] S. Xing, Y. He, H. Zhu, An approach to sample selection from big data for classification, in: Proc. IEEE International Conference on Systems Man and Cybernetics, 2017.

[16] W. Liu, S. Liu, R.C. Bai, Dynamic data reduction neural network training method, CAAI Transactions on Intelligent Systems 12(2)(2017) 258-265.

[17] J. Zhai, T. Li, X. Wang, A cross-selection instance algorithm, Journal of Intelligent & Fuzzy Systems 30(2)(2016) 717-728.

[18] Z. Yong, H. Shu, J. Hu, Adaptive pseudo nearest neighbor classification based on BP neural network, Journal of Electronics & Information Technology 38(11)(2016) 2774-2779.

[19] J. Sun, Modern Pattern Recognition, Higher Education Press, 2012.