PROCEEDINGS OF SCIENCE

A New Algorithm based on Bagging and NFS

Jianchun Lei[1]
School of Science, Minzu University of China
Beijing, 100081, China
E-mail: leijianchun@muc.edu.cn

Jinguo He
School of Science, Minzu University of China
Beijing, 100081, China
E-mail: hejinguo@pku.edu.cn

In order to improve the recognition rate of similar samples by NFS (neural-fuzzy system), bagging algorithm was proposed to improve the recognition rate. As the bagging algorithm needs a simple basic classifier, the basic classifier needs to be modified. In this paper, a new NFS was obtained when the traditional NFS input layer was removed. Then, with the combination of the bagging algorithm and a new NFS, a new model was set up. The experimental results showed that the recognition rate of the new model was not only 1.67% higher than that of a single NFS, but also the same as that of the decision tree, softmax and xgboost on Iris dataset. Based on the sensitivity and specificity analysis of the new model, the linear data can get better result of classification than the non-linear data. This new model as obtained by combining bagging with the new NFS features rapid prototyping, strong generalization ability and high recognition rate.

ISCC 2017
16-17 December, 2017
Guangzhou, China

[1]Speaker

PoS(ISCC 2017)008

1. Introduction

With the development of computer, there are more and more recognition algorithms. Some models make use of the theory of perceptron such as svm (Support Vector Machine)[1] and softmax. Some models are based on the information theory, for example, decision tree[2] and xgboost[3]. Some models take advantage of the principles of biological neurology such as BP neural network[4] and neural-fuzzy system (NFS)[5]. This paper will use NFS.

In 1996, Breiman proposed the bagging algorithm[6]. When he tested the real and simulated data sets, it showed that the bagging can give substantial gains in accuracy. At present, there are many different kinds of bagging because the basic classifier is different. For instance, Ma et al used likelihood-based belief decision trees as a basic classifier[7] and Pan et al treated BP network as a basic classifier[8]. This paper treats NFS as a basic classifier, and experimental result shows that the bagging can improve the recognition rate.

2. New Recognition Algorithm

2.1 The Theory of NFS in New Model

The membership was proposed by Professor Zadeh in 1965[9]. Then, it is not only the core of fuzzy mathematics but also the basic core of neural fuzzy system (NFS). The systemic study about the neural-fuzzy system is developed by Kosko[10]. Now there are two major trends: the fuzzy system based on the neural network (also known as the neural-fuzzy system) and the neural network by using fuzzy operation (also known as the narrow sense fuzzy neural network) [11]. As it uses the bagging algorithm to enhance the recognition ability, the new model needs several very simple classifiers. Therefore, the traditional NFS needs to be modified to form a new NFS (Figure 1).

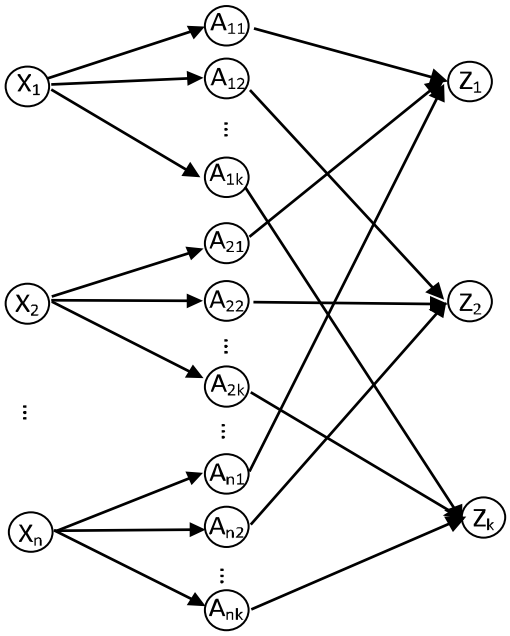

Figure 1 : The New Neural-fuzzy System

The final output of the new NFS is:

$$Z_j=\prod_{i=1}^{n} \mathrm{A}_{ij}(x^i)=\prod_{i=1}^{n} \exp(-\frac{(x^i-a_{ij})^2}{\sigma_{ij}^2});\ j=1,2,\ldots,k \quad (2.1)$$

For the convenience of calculation, the new model uses the improved model proposed by Li in his thesis [12]. Let $b_{ij}=1/\sigma_{ij}$, then

$$Z_j=\prod_{i=1}^{n} \exp(-(x^i-a_{ij})^2 b_{ij}^2) \quad (2.2)$$

The objective function is

$$\epsilon_t=\frac{1}{2\mathrm{kN}}\sum_{l=1}^{N}(1-h_t(x_l,y_l)+\sum_{y=1(y\neq y_l)}^{k} h_t(x_l,y)) \quad (2.3)$$

When all samples are classified correctly, the output of the objective function is 0. When all samples are classified incorrectly, the output is 1. It is generally known that the gradient descent method is a suitable method to find the extreme value, so the updated formula for s iterations is

$$a_{ij}(s+1)=a_{ij}(s)-\eta\frac{\partial\epsilon}{\partial a_{ij}};\forall i,j \quad (2.4)$$

$$b_{ij}(s+1)=b_{ij}(s)-\eta\frac{\partial\epsilon}{\partial b_{ij}};\forall i,j \quad (2.5)$$

where

$$\frac{\partial\epsilon}{\partial a_{ij}}=\frac{1}{kN}(\sum_{l=1(j\neq y_l)}^{N} z_{lj}b_{ij}^2(x_l^i-a_{ij})-\sum_{l=1(j=y_l)}^{N} z_{lj}b_{ij}^2(x_l^i-a_{ij})) \quad (2.6)$$

$$\frac{\partial\epsilon}{\partial b_{ij}}=\frac{1}{kN}(-\sum_{l=1(j\neq y_l)}^{N} z_{lj}b_{ij}(x_l^i-a_{ij})^2+\sum_{l=1(j=y_l)}^{N} z_{lj}b_{ij}(x_l^i-a_{ij})^2) \quad (2.7)$$

where x_l^i is i^{th} feature of l^{th} sample, and z_{lj} is j^{th} output of l^{th} sample.

2.2 Theory of Bagging in New Model

We can analyze the mean and variance of bagging algorithm to understand why the bagging can improve the recognition rate[6]. Let's assume that the mean of each weak classifier is μ and the variance of each weak classifier is σ^2 . Meanwhile, let's support that each classifier is independent of each other, so the mean and variance of bagging algorithm are

$$E(\frac{\sum_{i=1}^{T} X_i}{T})=E(X_i)=\mu \quad (2.8)$$

$$Var(\frac{\sum_{i=1}^{T} X_i}{T})=\frac{Var(X_i)}{T}=\frac{\sigma^2}{T} \quad (2.9)$$

According to the formulas above, the bagging can improve the recognition rate by reducing the variance of the model. Therefore, the key point of bagging is the creation of the basic classifier. The bagging gets different dataset by random sampling, so the basic classifiers on different dataset are different. As the NFS has the stability characteristics, if the new algorithm uses the random sampling mode, the bagging can't improve the recognition rate. In

order to adapt to the NFS model, the new algorithm uses a certain proportion of β to sampling without replacement.

2.3 The Process of Algorithm

Let's assume that there are N samples and a n-dimensional vector represents a sample. Then, $\boldsymbol{X}$ is $N\times n$ matrix, and $\boldsymbol{Y}$ is $n\times 1$ type vector. β is the sampling ratio.

Input: $\boldsymbol{X}$, $\boldsymbol{Y}$

Output: model

(1) given value β

(2) for $t=1$ to T

let $m=\beta N$

sampling m samples without replacement

① initialize parameters

let $p\epsilon_t=1$, $\epsilon_t=1$, $s=1$, $\eta=0.3$, $a_{ij}=E(x_l^i); y_l=j$,

$$b_{ij}=\frac{1}{SD(x_l^i)}; y_l=j$$

② forward calculate

calculate output $Z_{lj}=\prod_{i=1}^{n}\exp\left(-(x_l^i-a_{ij})^2 b_{ij}^2\right); l=1,2,\ldots,m,\ j=1,2,\ldots,k$

calculate error rate $\epsilon_t=\frac{1}{2\mathrm{km}}\sum_{i=1}^{m}\left(1-h_t(x_i,y_i)+\sum_{y=1(y\neq y_i)}^{k}h_t(x_i,y)\right)$

where $h_t(x_i,y)$ is value of z_{iy} .

③ for $s=2$ to 10000

if $p\epsilon_t<\epsilon_t$ and $\eta<0.3/1024$, then finish the loop; otherwise, go on

if $p\epsilon_t<\epsilon_t$, then $\eta=\eta/2$

let $p\epsilon_t=\epsilon_t$

update parameters $a_{ij}(s+1)$ and $b_{ij}(s+1)$ according to formula (2.4) and formula (2.5)

calculate output $Z_{lj}=\prod_{i=1}^{n}\exp\left(-(x_l^i-a_{ij})^2 b_{ij}^2\right); l=1,2\ldots m,\ j=1,2\ldots k$

calculate error $\epsilon_t=\frac{1}{2\mathrm{km}}\sum_{i=1}^{m}\left(1-h_t(x_i,y_i)+\sum_{y=1(y\neq y_i)}^{k}h_t(x_i,y)\right)$

where $h_t(x_i,y)$ is value of z_{iy} .

3. Dataset and Experiment

3.1 Experiment Data

The iris set was collected by Fisher in 1936. The data sets consist of three kinds of iris (setosa, versicolor and virginica), and each contains 50 samples. The vector on behalf of sample includes petal length, petal width, sepal length, sepal width and type. In iris set, the setosa is

linearly related to others, and the versicolor is non-linear to virginica. To maximize the performance of the new model, top 20 samples of each class were selected as the test set and others as a training set.

3.2 Experimental Platform and Method

We code the new model by using C++ on VS2010 platform according to the procedure above. The following is a figure about the relationship between the number of basic classifier and the error rate.

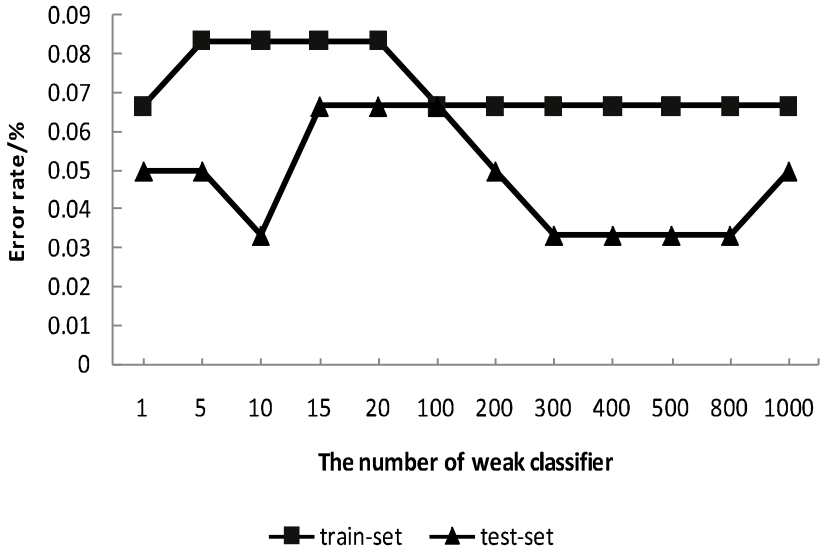

Figure 2 : The Error Rate of the Training Sample and Test Sample

As shown in Fig. 2, when the number of basic classifier is small, the error rate is unstable. When it is small, the randomness will have great influences on the new model. When the number of basic classifier is between 300 and 800, the error rate remains unchanged. When it is large, the randomness offsets each other. This phenomenon shows that the stability of the new model depends on the number of basic classifier. We can also know that the best number of basic classifier is 300. When the number is 1, we treat the new model as the single NFS.

Meanwhile, the new model is compared with svm, softmax, decision tree and xgboost. Those models use the existing methods in python. By appropriate tuning, the decision tree uses the gini coefficient and the svm penalty factor is 0.3. The results are shown in Table 1.

Methods	Setosa/%	Versicolor/%	Virginica/%	Total recognition rate/%
Svm	100	100	95	98.3
Softmax	100	100	90	96.7
Decision tree	100	100	90	96.7
XGBoost	100	100	90	96.7
Single NFS	100	95	90	95
New model	100	100	90	96.7

Table 1: The Recognition Rate of Iris

As shown in Table 1, svm is number one and the recognition rate of the new model is the same as that of softmax, decision tree and xgboost. The table also shows that the single NFS has the lowest recognition rate, which proves that bagging algorithm can improve the recognition rate.

Sensitiveness and specificity are used to describe the two-class classification. Because of the relationship in Iris, we can divide the iris data set into linear and non-linear parts from different angles. In the linear parts, we use setosa as positive sample and the others as negative samples. In non-linear parts, we treat versicolor as positive sample and virginica as negative samples. Table 2 shows sensitiveness and specificity about new model.

Types	Reality	Positive (prediction)	Negative (prediction)
Original	Positive	TP	FN
	Negative	FP	TN
Linear Dataset	Positive	20	0
	Negative	0	40
Non-linear Dataset	Positive	19	1
	Negative	2	18

Table 2: Sensitiveness and Specificity

The definition of sensitiveness is TPR=TP/(TP+FN) and the definition of specificity is TNR=TN/(FP+TN). For linear parts, the sensitiveness is TPR=20/(20+0)=100%, and the specificity is TNR=40/(0+40)=100%. It shows that the linear set that can be classified correctly by the new model. For the non-linear parts, the sensitiveness is TPR=19/(19+1)=95% and the specificity is TNR=18/(2+18)=90%. It shows that there are samples as classified incorrectly. In comparison of the two sets of data, the new model has greater classification ability on linear dataset than that on non-linear dataset.

4. Conclusion

The following items have their own advantages and disadvantages. Svm is good at generalization, but it is difficult to get proper parameters. The decision tree training is simple, but it is easy to overfit . Xgboost recognition is better, but its tuning is difficult. In this paper, the experiment shows that the new model has achieved good results. Even if there are larges samples, we can use parallel algorithm to speed up. The next step is to reconstruct more suitable basic classifiers, and we can try to use adaboost algorithm to enhance the recognition ability of the algorithm.

$\boldsymbol{X}$ is metric

$\boldsymbol{Y}$ is a vector

References

[1] C. J. C. Burges. *A tutorial on support vector machines for pattern recognition* [J]. Data Mining and Knowledge Discovery. 2(2), 121-167(1998)

[2] J.R. Quinlan. *Induction of decision trees* [J]. Machine Learning. 1(1), 81-106(1986)

[3] T.Q. Chen, C. Guestrin. *XGBoost: A scalable tree boosting system* [C]//In 22nd SIGKDD Conference on Knowledge Discovery and Data Mining. ACM, New York. 785-794(2016)

[4] J. Li, J.H. Chang, J.Y. Shi, F. Huang. *Brief introduction of back propagation(BP) neural network algorithm and its improvement*[C]//Advances in Conference on Computer Science and Information Engineering. Springer, Berlin. 553-558(2012)

[5] K. Cpałka, K. Łapa, A. Przybył, M. Zalasiński. *A new method for designing neuro-fuzzy systems for nonlinear modelling with interpretability aspects*[J]. Neurocomputing. 135(C), 203-217(2014)

[6] L. Breiman. *Bagging predictors* [J]. Machine Learning. 24(2), 123-140(1996)

[7] L.Y. Ma, B. Sun, Z.Y. Li. *Bagging likelihood-based belief decision trees*[C]//20th International Conference on Information Fusion. IEEE, Xi'an. 321-326(2017)

[8] J. Pan, J.J. Ding, X.X. He. *Research and application of PCA-BP-Bagging model in medical assistant diagnosis* [C]//Proceedings of 36th Chinese Control Conference. IEEE, Dalian. 4127-4130(2017)

[9] L. A. Zadeh. *Fuzzy sets* [J]. Information and Control. 8(3), 338–353(1965)

[10] J.J. Hopfield. *Neural networks and physical systems with emergent collective computational abilities* [J]. In Proceedings of National Academic Science. 79(8), 2554-2558(1982)

[11] K. Zhang, F. Qian, M. D. Liu. *A survey on fuzzy neural network technology* [J].Information and Control. 32(5), 431-435 (2003) (In Chinese)

[12] L. Li. *Learning algorithms and convergence analysis for fuzzy neural networks* [D]. Dalian: Dalian University of Technology(2010) (In Chinese)