Proceedings of the 6th WSEAS Int. Conf. on NEURAL NETWORKS, Lisbon, Portugal, June 16-18, 2005 (pp140-145)

Combination Methods for Ensembles of RBF Networks

CARLOS HERNÁNDEZ-ESPINOSA
JOAQUÍN TORRES-SOSPEDRA
MERCEDES FERNÁNDEZ-REDONDO
Dept. de Ingeniería y Ciencia de los Computadores
Universidad Jaume I
Avda. Vicente Sos Baynat s/n 12071 Castellón
SPAIN

Abstract: - Building an ensemble of classifiers is an useful way to improve the performance. In the case of neural networks the bibliography has centered on the use of Multilayer Feedforward (MF). However, there are other interesting networks like Radial Basis Functions (RBF) that can be used as elements of the ensemble. In a previous paper we presented results of different methods to build the ensemble of RBF. The results showed that the best method is in general the *Simple Ensemble*. The combination method used in that research was averaging. In this paper we present results of fourteen different combination methods for a simple ensemble of RBF. The best methods are Borda Count, Weighted Average and Majority Voting.

Key-Words: - Neural Networks, Ensembles, Radial Basis Functions, Combination methods.

1 Introduction

Probably the most important property of a neural network (NN) is the generalization capability. One method to increase this capability with respect to a single NN consist on training an ensemble of NNs, i.e., to train a set of NNs with different weight initialization or properties and combine the outputs in a suitable manner.

In the field of ensemble design, the two key factors to design an ensemble are how to train the individual networks and how to combine the different outputs.

It seems clear from the bibliography that this procedure generally increases the generalization capability in the case of the NN Multilayer Feedforward (MF) [1,2].

However, in the field of NNs there are other networks besides MF, and traditionally the use of ensembles of NNs has restricted to the use of MF.

Another useful network which is quite used in applications is Radial Basis Functions (RBF). This network can also be trained by gradient descent [3]. So with a fully supervised training, it can be also an element of an ensemble, and all methods of constructing the ensemble which are applicable to MF can now be also used with RBF networks.

In the paper [4] we obtain the first results on ensembles of RBF, we presented a comparison of different methods to build the ensemble and we concluded that the "Simple Ensemble" was the most appropriate. In that case, the combination method was one of the simplest: averaging the outputs.

In this paper we present results of different combination methods for the case of a "simple ensemble" of RBFs. The number of combination methods analyzed is fourteen. With these results we can have a hint to select the appropriate combination method and improve the performance of RBFs ensembles.

2 Theory

In this section, first we briefly review the basic concepts of RBFs networks and after that we review the different methods of combining the outputs of the ensemble.

2.1 RBF networks with gradient descent training

A RBF has two layer of networks. The first layer is composed of neurons with a Gaussian transfer function and the second layer has neurons with a linear transfer function. The output of a RBF network can be calculated with equation 1.

$$F_k(x) = \sum_{q=1}^{Q} w_q^k \cdot \exp\left(- \sum_{n=1}^{N} \frac{\left(C_{q,n}^k - X_n\right)^2}{\left(\sigma_q^k\right)^2} \right) \quad \textbf{(1)}$$

Where $C_{q,n}{}^k$ are the centers of the Gaussian units, $\sigma_q{}^k$ control the width of the Gaussian functions and $w_q{}^k$ are the weights among the Gaussian units and the output units.

The parameters which are changed during the training process [3] are $C_{q,n}{}^k$ and $w_q{}^k$, the width is the same for all Gaussian units and fixed before training. The equation for the adaptation of the weights is the following:

$$\Delta w_q^k = \eta \cdot \varepsilon_k \cdot \exp\left(-\sum_{n=1}^{N} \left(C_{q,n}^k - X_n\right)^2 \Big/ (\sigma)^2 \right) \quad \textbf{(2)}$$

Where η is the step size and ε_k is the difference between the target and the output.

And the equation for the adaptation of the centers is number 3.

$$\Delta C_q = \eta \cdot (X_k - C_q) \cdot \frac{2}{\sigma} \cdot \exp\left(-\sum_{n=1}^{N} \frac{\left(C_{q,n}^k - X_n\right)^2}{(\sigma)^2} \right) \cdot \sum_{k=1}^{n_o} \varepsilon_k \cdot w_q^k \quad \textbf{(3)}$$

2.2 Ensemble Combination Methods

Average: This approach simply averages the individual classifier outputs. The output yielding the maximum of the averaged values is chosen as the correct class.

Majority Vote: The correct class is the one most often chosen by different classifiers. If all the classifiers indicate different classes, then the one with the overall maximum output value is selected to indicate the correct class.

Winner Takes All (WTA): In this method, the class with overall maximum value in all the classifiers is selected.

Borda Count:. For any class q, the Borda count is the sum of the number of classes ranked below q by each classifier. If $B_j(q)$ is the number of classes ranked below the class q by the jth classifier, then the Borda count for class q is in the following equation.

$$B(q) = \sum_{j=1}^{L} B_j(q) \quad \textbf{(4)}$$

Bayesian Combination: This combination method was proposed in reference [5]. According to this reference a belief value that the pattern x belongs to class i can be approximated by the following equation.

$$Bel(i) = \frac{\prod_{k=1}^{L} P(x \in q_i \mid \lambda_k(x) = j_k)}{\sum_{i=1}^{Q} \prod_{k=1}^{L} P(x \in q_i \mid \lambda_k(x) = j_k)}, \quad 1 \le i \le Q \quad \textbf{(5)}$$

Where the conditional probability that sample x actually belongs to class i, given that classifier k assign it to class j ($\lambda_k(x)=j_k$) can be estimated from the values of the confusion matrix [6].

Weighted Average: This method introduces weights to the outputs of the different networks prior to averaging. The weights try to minimize the difference between the output of the ensemble and the "desired or true" output. The weights can be estimated from the error correlation matrix. A full description of the method can be found in reference [7].

Choquet Integral: This method is based in the fuzzy integral and the Choquet integral. The method is complex and a full description can be found in reference [6].

Combination by Fuzzy Integral with Data Dependent Densities (Int. DD): It is another method based on the fuzzy integral and the Choquet integral. But in this case, prior to the application of the method it is performed a partition of the input space in regions by k-means clustering or frequency sensitive learning. The full description can be found in reference [6].

Weighted Average with Data Dependent weights (W.Ave DD): This method is the weighted average described above. But in this case, a partition of the space is performed by using k-means clustering and the weights are calculated for each partition. We have a different combination scheme for the different partitions of the space. The number of partitions of the space is determined by cross-validation.

BADD Defuzzification Strategy: It is another combination method based on fuzzy logic concepts. The method is complex and the description can be found in [6].

Zimmermann's Compensatory Operator: This combination method is based in the Zimmermann's compensatory operator described in [8]. The method is complex and can be found in [6].

Dynamically Averaged Networks (DAN), version 1 and 2: It is proposed in reference [9]. In this method instead of choosing static weights derived from the network performance on a sample of the input space, we allow the weights to adjust to be proportional to the certainties of the respective network output. In the reference two different versions of the method are described and we have included both.

Nash Vote: In this method each voter assigns a number between zero and one for each candidate output. The product of the voter's values is compared for all candidates. The higher is the winner. The method is reviewed in reference [10].

3 Experimental Results

We have applied the combination methods to nine different classification problems. They are from the UCI repository of machine learning databases. Their names are Balance Scale (Bala), Cylinders Bands (Band), Liver Disorders (Bupa), Credit Approval (Credit), Glass Identification (Glass), Heart Disease (Heart), the Monk's Problems (Monk 1, Monk 2) and Voting Records (Vote). A full description can be found in the UCI repository (http://www.ics.uci.edu /~mlearn/ MLRepository.html).

We have constructed ensembles of 3 and 9 networks.

For the training of the ensembles, we repeated the process of training ten times for different partitions of data in training, cross-validation and test sets. With this procedure we can obtain a mean performance of the ensemble for each database (the mean of the ten ensembles) and an error in the performance calculated by standard error theory. The results of the performance are in table 1 for the case of ensembles of three networks and in table 2 for the case of nine.

As we can see in table 1 and 2 the improvements by the use of an ensemble are problem dependent. We get an improvement in problems Bupa, Credit, Glass, Heart, Monk 1 and Vote. This result was already known in the bibliography for the case of Multilayer Feedforward.

The results of tables 1 and 2 show that the improvement in performance of training nine networks (instead of three) is low. Taking into account the computational cost the best alternative might be an ensemble of three networks.

Comparing the results of the different combination methods of tables 1 and 2, we can see that the differences are low. The largest difference between simple average and other method is around 0.7% in the problem Heart.

However, we should point out that the computational cost of any combination method is very low in comparison with the computational cost of training the ensemble of networks. So the selection of an appropriate combination method allows an improvement in the performance without extra computational cost.

Table 1. Results for the ensemble of three networks.

	Bala	Band	Bupa
SingleNetwork	90.2 ± 0.5	74.0 ± 1.1	70.1 ± 1.1
Average	89.7 ± 0.7	73.8 ± 1.2	71.9 ± 1.1
Majority V.	89.9 ± 0.7	74.4 ± 1.2	72.0 ± 1.0
WTA	90.1 ± 0.8	72.9 ± 1.1	71.1 ± 1.2
Borda	89.8 ± 0.7	74.4 ± 1.2	72.0 ± 1.0
Bayesian	89.9 ± 0.7	74.4 ± 1.2	72.0 ± 1.0
W. Average	89.9 ± 0.7	72.9 ± 1.5	72.4 ± 1.2
Choquet	89.9 ± 0.7	73.1 ± 1.1	71.4 ± 1.0
Int. DD	89.9 ± 0.7	72.9 ± 1.2	71.9 ± 0.9
W. Ave DD	89.7 ± 0.7	74.2 ± 1.0	71.9 ± 1.1
BADD	89.7 ± 0.7	73.8 ± 1.2	71.9 ± 1.1
Zimmermann	65 ± 5	63 ± 5	62 ± 3
DAN	89.8 ± 0.8	73.5 ± 1.3	71.9 ± 1.0
DAN version 2	89.9 ± 0.8	73.1 ± 1.3	71.7 ± 1.0
Nash Vote	89.7 ± 0.7	74.0 ± 1.2	72.3 ± 1.1

Table 2. (continuation 1) Results for the ensemble of three networks.

	Credit	Glass	Heart
SingleNetwork	86.0 ± 0.8	93.0 ± 0.6	82.0 ± 1.0
Average	87.2 ± 0.5	93.2 ± 1.0	83.9 ± 1.6
Majority V.	87.1 ± 0.6	93.2 ± 1.0	84.6 ± 1.5
WTA	87.2 ± 0.6	93.2 ± 1.0	83.9 ± 1.6
Borda	87.1 ± 0.6	93.2 ± 1.0	84.6 ± 1.5
Bayesian	87.1 ± 0.6	93.2 ± 1.0	84.6 ± 1.5
W. Average	87.2 ± 0.5	93.0 ± 1.2	83.6 ± 1.6
Choquet	86.9 ± 0.6	93.2 ± 1.0	83.6 ± 1.6
Int. DD	86.9 ± 0.6	93.0 ± 0.9	83.6 ± 1.4
W. Ave DD	87.2 ± 0.5	93.2 ± 1.0	83.9 ± 1.6
BADD	87.2 ± 0.5	93.2 ± 1.0	83.9 ± 1.6
Zimmermann	70 ± 5	87.2 ± 1.5	75 ± 4
DAN	87.3 ± 0.5	93.6 ± 1.1	83.6 ± 1.3
DAN version2	87.2 ± 0.5	93.8 ± 1.1	83.4 ± 1.5
Nash Vote	87.2 ± 0.5	93.2 ± 1.0	84.1 ± 1.7

Table 3. (continuation 2) Results for the ensemble of three networks.

	Monk 1	Monk 2	Vote
SingleNetwork	98.5 ± 0.5	91.3 ± 0.7	95.4 ± 0.5
Average	99.6 ± 0.4	91.5 ± 1.2	96.3 ± 0.7
Majority V.	99.6 ± 0.4	90.9 ± 1.1	96.4 ± 0.6
WTA	99.6 ± 0.4	91.4 ± 1.3	96.3 ± 0.7
Borda	99.6 ± 0.4	90.9 ± 1.1	96.4 ± 0.6
Bayesian	99.4 ± 0.4	90.1 ± 1.1	96.4 ± 0.6
W. Average	99.8 ± 0.3	92.0 ± 1.2	96.3 ± 0.6
Choquet	99.6 ± 0.4	91.5 ± 1.2	96.3 ± 0.7
Int. DD	99.6 ± 0.4	91.1 ± 1.3	96.3 ± 0.7
W. Ave DD	99.6 ± 0.4	92.0 ± 1.2	96.3 ± 0.7
BADD	99.6 ± 0.4	91.5 ± 1.2	96.3 ± 0.7
Zimmermann	90 ± 5	82 ± 3	92 ± 3
DAN	99.5 ± 0.4	90.8 ± 1.3	96.0 ± 0.6
DAN version2	99.6 ± 0.4	90.6 ± 1.4	96.1 ± 0.6
Nash Vote	99.6 ± 0.4	91.6 ± 1.1	96.3 ± 0.7

Table 2. Results for the ensemble of nine networks.

	Bala	Band	Bupa
SingleNetwork	90.2 ± 0.5	74.0 ± 1.1	70.1 ± 1.1
Average	89.7 ± 0.7	73.3 ± 1.4	72.4 ± 1.2
Majority V.	89.7 ± 0.7	74.0 ± 1.5	72.1 ± 1.1
WTA	89.8 ± 0.8	73.6 ± 1.7	72.0 ± 1.3
Borda	89.6 ± 0.7	74.0 ± 1.5	72.1 ± 1.1
Bayesian	90.2 ± 0.7	74.2 ± 1.5	72.3 ± 1.1
W. Average	89.5 ± 0.7	73.1 ± 1.6	71.6 ± 1.3
Choquet	89.8 ± 0.8	74.0 ± 1.5	72.0 ± 1.4
Int. DD	89.8 ± 0.8	74.0 ± 1.5	72.1 ± 1.5
W. Ave DD	89.7 ± 0.7	73.3 ± 1.4	72.3 ± 1.2
BADD	89.7 ± 0.7	73.3 ± 1.4	72.4 ± 1.2
Zimmermann	69 ± 5	66 ± 3	62 ± 4
DAN	89.8 ± 0.8	72.7 ± 1.7	71.6 ± 1.3
DAN version2	89.8 ± 0.8	73.3 ± 1.7	71.4 ± 1.3
Nash Vote	89.6 ± 0.7	73.1 ± 1.4	72.6 ± 1.2

Table 2. (continuation 1) Results for the ensemble of nine networks.

	Credit	Glass	Heart
SingleNetwork	86.0 ± 0.8	93.0 ± 0.6	82.0 ± 1.0
Average	87.2 ± 0.5	93.0 ± 1.0	83.9 ± 1.5
Majority V.	87.1 ± 0.5	93.2 ± 1.0	84.6 ± 1.6
WTA	87.2 ± 0.5	93.4 ± 1.0	83.6 ± 1.7
Borda	87.1 ± 0.5	93.2 ± 1.0	84.6 ± 1.6
Bayesian	87.2 ± 0.5	92.6 ± 1.0	84.6 ± 1.6
W. Average	86.9 ± 0.5	92.8 ± 1.2	83.7 ± 1.4
Choquet	87.3 ± 0.5	93.4 ± 1.0	83.4 ± 1.6
Int. DD	87.2 ± 0.5	93.4 ± 1.0	83.4 ± 1.6
W. Ave DD	87.2 ± 0.5	93.0 ± 1.0	83.9 ± 1.5
BADD	87.2 ± 0.5	93.0 ± 1.0	83.9 ± 1.5
Zimmermann	75 ± 5	80 ± 3	73 ± 4
DAN	87.2 ± 0.5	92.8 ± 1.0	84.4 ± 1.7
DAN version2	87.3 ± 0.5	92.8 ± 1.0	84.1 ± 1.8
Nash Vote	87.2 ± 0.5	93.0 ± 1.0	83.9 ± 1.5

Table 2. (continuation 2) Results for the ensemble of nine networks.

	Monk 1	Monk 2	Vote
SingleNetwork	98.5 ± 0.5	91.3 ± 0.7	95.4 ± 0.5
Average	99.6 ± 0.4	91.4 ± 1.2	96.3 ± 0.7
Majority V.	99.6 ± 0.4	91.5 ± 1.2	96.4 ± 0.6
WTA	99.8 ± 0.3	90.8 ± 1.2	96.0 ± 0.6
Borda	99.6 ± 0.4	91.5 ± 1.2	96.4 ± 0.6
Bayesian	99.5 ± 0.3	90.9 ± 1.1	96.4 ± 0.6
W. Average	99.8 ± 0.3	91.8 ± 1.4	96.6 ± 0.7
Choquet	99.6 ± 0.4	90.6 ± 1.2	96.0 ± 0.6
Int. DD	99.6 ± 0.4	90.8 ± 1.2	96.1 ± 0.6
W. Ave DD	99.6 ± 0.4	91.3 ± 1.2	96.3 ± 0.7
BADD	99.6 ± 0.4	91.4 ± 1.2	96.3 ± 0.7
Zimmermann	92 ± 2	76 ± 5	81 ± 6
DAN	99.4 ± 0.4	88.9 ± 1.6	96.0 ± 0.6
DAN version2	99.5 ± 0.4	88.9 ± 1.5	96.0 ± 0.6
Nash Vote	99.6 ± 0.4	91.5 ± 1.1	96.3 ± 0.7

To appreciate the results more clearly, we have also calculated the percentage of error reduction of the ensemble with respect to a single network. We have used equation 6 for this calculation.

$$PorError_{reduction} = 100 \cdot \frac{PorError_{single\ network} - PorError_{ensemble}}{PorError_{single\ network}} \quad (6)$$

The value of the percentage of error reduction ranges from 0%, where there is no improvement by the use of a particular ensemble method with respect to a single network, to 100%. There can also be negative values, which means that the performance of the ensemble is worse than the performance of the single network.

We have the full results of percentage of error reduction for the ensemble of three networks in table 3.

Also, this new measurement is relative and we can calculate a mean value across all databases. The result is under the column header "MEAN" in table 3 for the case of three networks and in table 4 for the case of nine networks.

Table 3. Relative Performance with respect to a Single Network for the case of three networks in the ensemble.

	Bala	Band	Bupa
Average	-5,31	-0,69	5,89
Majority V.	-2,86	1,38	6,35
WTA	-1,22	-4,19	3,48
Borda	-3,67	1,38	6,35
Bayesian	-2,86	1,38	6,35
W. Average	-2,86	-4,19	7,79
Choquet	-2,86	-3,50	4,45
Int. DD	-2,86	-4,19	5,89
W. Ave DD	-5,31	0,69	5,89
BADD	-5,31	-0,69	5,89
Zimmermann	-259,18	-41,27	-28,06
DAN	-4,49	-2,12	5,89
DAN version2	-2,86	-3,50	5,38
Nash Vote	-5,31	0,00	7,32

Table 3. (continuation 1) Relative Performance with respect to a Single Network for the case of three networks in the ensemble.

	Credit	Glass	Heart
Average	8,21	2,86	10,56
Majority V.	7,71	2,86	14,33
WTA	8,21	2,86	10,56
Borda	7,71	2,86	14,33
Bayesian	7,71	2,86	14,33
W. Average	8,21	0,00	8,67
Choquet	6,57	2,86	8,67
Int. DD	6,57	0,00	8,67
W. Ave DD	8,21	2,86	10,56
BADD	8,21	2,86	10,56
Zimmermann	-114,86	-82,86	-41,22
DAN	9,36	8,57	8,67
DAN version2	8,79	11,43	7,72
Nash Vote	8,21	2,86	11,50

Table 3. (continuation 2) Relative Performance with respect to a Single Network for the case of three networks in the ensemble.

	Monk 1	**Monk 2**	**Vote**
Average	75,33	2,30	18,48
Majority V.	75,33	-4,83	21,30
WTA	75,33	0,92	18,48
Borda	75,33	-4,83	21,30
Bayesian	58,67	-4,83	21,30
W. Average	83,33	8,05	18,48
Choquet	75,33	2,30	18,48
Int. DD	75,33	-1,95	18,48
W. Ave DD	75,33	8,05	18,48
BADD	75,33	2,30	18,48
Zimmermann	-574,67	-105,40	-68,48
DAN	66,67	-6,32	13,04
DAN version2	75,33	-7,70	15,87
Nash Vote	75,33	3,79	18,48

Table 3. (continuation 3) Relative Performance with respect to a Single Network for the case of three networks in the ensemble.

	MEAN
Average	13,07
Majority V.	13,24
WTA	12,26
Borda	13,24
Bayesian	11,39
W. Average	13,89
Choquet	12,21
Int. DD	11,50
W. Ave DD	13,86
BADD	13,07
Zimmermann	-118,01
DAN	10,94
DAN version2	12,00
Nash Vote	13,58

Table 4. Relative Performance with respect to a Single Network for the case of nine networks in the ensemble.

	MEAN
Average	12,64
Majority V.	13,93
WTA	12,63
Borda	13,84
Bayesian	12,91
W. Average	13,62
Choquet	11,70
Int. DD	12,10
W. Ave DD	12,42
BADD	12,64
Zimmermann	-168,30
DAN	6,46
DAN version2	7,45
Nash Vote	12,68

According to the values of the mean performance of error reduction, the best performing methods are Majority Vote, Borda Count and Weighted Average.

If we consider the simplicity Majority Vote may be the best alternative.

4 Conclusion

In this paper we have presented experimental results of fourteen different methods to combine the outputs of an ensemble of RBF networks, using ten different databases. We trained ensembles of 3 and 9 networks in the ensemble. The results showed that in general the improvement by the use of the ensemble methods depends clearly on the database, in some databases there is an improvement but in other there is not improvement at all. Also the improvement in performance from three to nine networks in the ensemble is usually low. Taking into account the computational cost, an ensemble of three networks might be the best alternative for most of the methods. Also the differences among the different combination methods are low, but in this case the extra performance is obtained without additional computational cost, and it can be worthy to select the appropriate combination method. Finally, we have obtained the mean percentage of error reduction over all databases. According to the values of the mean performance of error reduction, the best performing methods are Majority Vote, Borda Count and Weighted Average. If we consider the simplicity Majority Vote may be the best alternative.

Acknowledgments

This research was supported by the project MAPACI TIC2002-02273 of CICYT in Spain.

References:

[1] Tumer, K., Ghosh, J., "Error correlation and error reduction in ensemble classifiers", *Connection Science*, Vol. 8, Nos. 3 & 4, 1996, pp. 385-404.

[2] Raviv, Y., Intrator, N., "Bootstrapping with Noise: An Effective Regularization Technique", *Connection Science*, Vol. 8, No. 3 & 4, 1996, pp. 355-372.

[3] Karayiannis, N.B., Randolph-Gips, M.M., "On the Construction and Training of Reformulated Radial Basis Function Neural Networks", *IEEE*

Trans. On Neural Networks, Vol.14, No. 4, 2003, pp. 835-846.

[4] Hernandez-Espinosa, Carlos, Fernandez-Redondo, Mercedes, Torres-Sospedra, Joaquin, “Ensambles of RBFs Trained by Gradient Descent”, *Internacional Symposium on Neural Networks*, *Lecture Notes in Computer Science*, Vol. 3174, 2004, pp. 223-228.

[5] Xu, L., Krzyzak, A., SUen, C.Y., “Methods of combining multiple classifiers and their applications to handwriting recognition”, *IEEE Trans. On Systems, Man and Cybernetics*, Vol. 22, No. 3, 1992, pp. 418-435.

[6] Verikas, A., Lipnickas, A., Malmqvist, K., Bacauskiene, M., Gelzinis, A., “Soft Combination of neural classifiers: A comparative study“, *Pattern Recognition Letters*, Vol. 20, 1999, pp 429-444.

[7] Krogh, A., Vedelsby, J., “Neural network ensembles, cross validation, and active learning”, *Advances in Neural Information Processing Systems 7*, 1995, pp. 231-238.

[8] Zymmermann, H.J., Zysno, P., “Decision and evaluations by hierarchical aggregation of information”, *Fuzzy Sets and Systems*, Vol. 10, No. 3, 1984, pp. 243-260.

[9] Jimenez, D., “Dynamically Weighted Ensemble Neural Network for Classification”, *IEEE World Congress on Computational Intelligence*, Vol. 1, 1998, pp. 753-756.

[10] Wanas, N.M., Kamel, M.S., “Decision Fusion in Neural Network Ensembles”, *International Joint Conference on Neural Networks*, Vol. 4, 2001, pp. 2952-2957.