In: *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, Seattle, WA, 2001, vol.2, pp.797-802

Genetic Algorithm based Selective Neural Network Ensemble

Zhi-Hua Zhou Jian-Xin Wu Yuan Jiang Shi-Fu Chen

National Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, P.R.China
zhou@nju.edu.cn {wujx, jy}@ai.nju.edu.cn chensf@nju.edu.cn

Abstract

Neural network ensemble is a learning paradigm where several neural networks are jointly used to solve a problem. In this paper, the relationship between the generalization ability of the neural network ensemble and the correlation of the individual neural networks is analyzed, which reveals that ensembling a selective subset of individual networks is superior to ensembling all the individual networks in some cases. Therefore an approach named GASEN is proposed, which trains several individual neural networks and then employs genetic algorithm to select an optimum subset of individual networks to constitute an ensemble. Experimental results show that, comparing with a popular ensemble approach, i.e. *averaging all*, and a theoretically optimum selective ensemble approach, i.e. *enumerating*, GASEN has preferable performance in generating ensembles with strong generalization ability in relatively small computational cost.

1 Introduction

Since neural computing has no rigorous theoretical framework until now, whether a neural network based application will be successful or not is almost fully determined by that who is the practitioner. In general, the more experiences the practitioner has on neural computing, the more chances the application will have in gaining success. However, in real-world applications, the users are often those with little knowledge on neural computing. Therefore the rewards that neural network techniques may return do not always appear.

In the beginning of the 1990's, Hansen and Salamon [1990] showed that the generalization ability of a neural network system can be significantly improved through ensembling neural networks, i.e. training several neural networks and combining their results in some way. Later, Sollich and Krogh [1996] defined neural network ensemble as a collection of a (finite) number of neural networks that are trained for the same task. Since it behaves remarkably well and is easy to use, neural network ensemble is regarded as a promising methodology that can profit not only experts in neural computing but also ordinary engineers in real-world applications. And neural network ensemble has already been used in many real domains such as handwritten digit recognition [Hansen *et al.*, 1992], scientific image analysis [Cherkauer, 1996], face recognition [Gutta and Wechsler, 1996; Huang *et al.*, 2000], OCR [Mao, 1998], seismic signals classification [Shimshoni and Intrator, 1998], etc. Many works have been done in investigating why and how neural network ensemble works. The classical one is Krogh and Vedelsby [1995] 's work, in which they derived a famous equation $E = \overline{E} - \overline{A}$ that clearly demonstrates that the generalization ability of the ensemble is determined by the average generalization ability and the average ambiguity of the individual neural networks that constitutes the ensemble.

In this paper, the relationship between the generalization ability of the neural network ensemble and the correlation of the individual neural networks is analyzed, which reveals that in some cases ensembling an appropriate subset of individual networks is superior to prevailing ensemble schemes, i.e. ensembling all the individual networks at hand. Based upon the recognition that the appropriate subset of individual networks is difficult to be found out directly, a genetic algorithm based approach named GASEN (Genetic Algorithm based Selective ENsemble) is proposed, which trains several individual neural networks and then employs genetic algorithm to select an optimum set of individual

networks to constitute an ensemble. Experiments show that GASEN is superior to a popular ensemble approach, i.e. *averaging all* that averages the outputs of all the individual networks at each output unit. Experiments also show that GASEN is superior to a selective ensemble approach that is theoretically optimum, i.e. *enumerating* that estimates the generalization ability of every possible subset of individual networks and then selects the best subset to make an ensemble. Moreover, most ensemble approaches require their individual networks be independently trained. But at present there is no method can guarantee the independence when there are many individual networks. Since GASEN can increase the ambiguity of the ensemble through selecting the appropriate subset of individual networks, it does not claim independent training, which makes it more easily to use than many other ensemble approaches.

The rest of this paper is organized as follows. In Section 2, the relationship between the generalization ability of the ensemble and the correlation of the individual neural networks is analyzed. In Section 3, GASEN is proposed to find out the optimum subset of individual networks. In Section 4, experiments on *averaging all*, *enumerating*, and GASEN are reported. In Section 5, related works are overviewed. Finally in Section 6, conclusions are drawn and several issues for future works are indicated.

2 Generalization and Correlation

Suppose the learning task is to use an ensemble that comprises N individual neural networks to approximate a function f: $\mathbf{R}^m \rightarrow \mathbf{R}^n$. The predictions of the individual networks are combined through *weighted averaging*, where a weight w_i (i = 1, 2, …, N) is assigned to the individual network f_i, and w_i satisfies equation (1) and (2):

$$0 < w_i < 1 \tag{1}$$

$$\sum_{i=1}^{N} w_i = 1 \tag{2}$$

Therefore the k-th output component of the ensemble is computed according to equation (3), where $f_{i,k}$ is the value of the k-th output component of the i-th individual network.

$$\overline{f}_k = \sum_{i=1}^{N} w_i f_{i,k} \tag{3}$$

For convenience of discussion, here we assume that each individual network has only one output component, i.e. the function to be approximated is f: $\mathbf{R}^m \rightarrow \mathbf{R}$. But note that following derivation can be easily generalized to situations where each individual network has multiple output components.

Suppose $x \in \mathbf{R}^m$ is randomly sampled according to a distribution $p(x)$. The expected output of x is $d(x)$. The actual output of the i-th individual neural network is $f_i(x)$. Then the output of the neural network ensemble is:

$$\overline{f}(x) = \sum_{i=1}^{N} w_i f_i(x) \tag{4}$$

The generalization error $E_i(x)$ of the i-th individual network on input x and the generalization error $E(x)$ of the ensemble on input x are respectively:

$$E_i(x) = (f_i(x) - d(x))^2 \tag{5}$$

$$E(x) = (\overline{f}(x) - d(x))^2 \tag{6}$$

Then the generalization error E_i of the i-th individual neural network on the distribution $p(x)$ and the generalization error E of the ensemble on the distribution $p(x)$ are respectively:

$$E_i = \int dx p(x) E_i(x) \tag{7}$$

$$E = \int dx p(x) E(x) \tag{8}$$

The average generalization error of the individual neural networks on input x is:

$$\overline{E}(x) = \sum_{i=1}^{N} w_i E_i(x) \tag{9}$$

Then the average generalization error of the individual neural networks on the distribution $p(x)$ is:

$$\overline{E} = \int dx p(x) \overline{E}(x) \tag{10}$$

Now we define the correlation between the i-th and the j-th individual neural networks as:

$$C_{ij} = \int dx p(x) (f_i(x) - d(x)) (f_j(x) - d(x)) \tag{11}$$

Note that C_{ij} satisfies equation (12) and (13):

$$C_{ii} = E_i \tag{12}$$

$$C_{ij} = C_{ji} \tag{13}$$

Considering equation (4) and (6) we get:

$$E(x) = \left(\sum_{i=1}^{N} w_i f_i(x) - d(x) \right) \left(\sum_{j=1}^{N} w_j f_j(x) - d(x) \right) \tag{14}$$

Then considering equation (14) and (11) we get:

$$E = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j C_{ij} \tag{15}$$

Different to Krogh and Vedelsby [1995] 's result $E = \overline{E} - \overline{A}$, equation (15) utilizes the correlation between the individual neural networks to represent the generalization error of the ensemble. Since the computation of C_{ij} only refers to f_i and f_j, equation (15) is easier to use

than $E = \overline{E} - \overline{A}$ in real-world applications.

Suppose that $w_i = 1/N$ ($i = 1, 2, \ldots, N$), i.e. the predictions of the individual neural networks are combined via *averaging*. Then equation (15) becomes:

$$E = \sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij} / N^2 \tag{16}$$

Let's assume that the k-th individual neural network is deleted from the ensemble. Then the generalization error of the new ensemble is:

$$E' = \sum_{\substack{i=1 \\ i\neq k}}^{N}\sum_{\substack{j=1 \\ j\neq k}}^{N} C_{ij} /(N-1)^2 \tag{17}$$

Considering equation (16) and (17) we get:

$$E - E' = \frac{2\sum_{\substack{i=1 \\ i\neq k}}^{N} C_{ik} + E_k - (2N-1)E}{(N-1)^2} \tag{18}$$

It is obvious that $E > E'$ when equation (19) is satisfied, which means that the new ensemble that omits f_k is more accurate than the original one that includes f_k.

$$E < \left(2\sum_{\substack{i=1 \\ i\neq k}}^{N} C_{ik} + E_k \right) /(2N-1) \tag{19}$$

Considering equation (19) and (17) we get the constraints on f_k:

$$(2N-1)\sum_{\substack{i=1 \\ i\neq k}}^{N}\sum_{\substack{j=1 \\ j\neq k}}^{N} C_{ij} < 2(N-1)^2 \sum_{\substack{i=1 \\ i\neq k}}^{N} C_{ik} + (N-1)^2 E_k \tag{20}$$

Now a conclusion is arrived that after the individual neural networks are trained, in some cases ensembling an appropriate subset of individual neural networks is superior to ensembling all the individual networks. The individual networks that should be omitted satisfy equation (20).

3 GASEN

Note that the individual neural networks to be omitted are hard to be found out directly by equation (20) due to the extensive computation required. Moreover, following observation is noteworthy that in real-world applications the generalization error of the individual neural networks and that of the ensemble are all unknown. However, we can employ cross-validation to get their generalization error on a validation set, which can be used to approximate the actual generalization error.

An approach named *enumerating* can be utilized to find out the appropriate subset of individual networks, which estimates the generalization error of all the possible subsets of $\{f_1, f_2, \ldots, f_N\}$ and then selects the best subset to make an ensemble. When N is a small number, *enumerating* can achieves optimum results. However, if N is a big number, such as $N > 30$, *enumerating* is nearly impossible to be realized due to its excessive computational cost (it will estimates the generalization error of $2^N - 1$ number of ensembles).

Here we present a practical routine to find out the appropriate subset of individual neural networks. Assume we can assign to each individual neural network an optimum weight that exhibits its importance in the ensemble. Then we can select the individual networks whose weight is bigger than a pre-set threshold λ to constitute the ensemble via *averaging*. Suppose the weight corresponding to the i-th individual neural network is w_i, which satisfies both equation (1) and (2). Then we have a weight vector $w = (w_1, w_2, \ldots, w_N)$. Since the optimum weight should minimize the generalization error of the ensemble, considering equation (15), the optimum weight vector w_{opt} can be expressed as:

$$w_{opt} = \arg\min\left(\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j C_{ij} \right) \tag{21}$$

$w_{opt.k}$, i.e. the k-th ($k = 1, 2, \ldots, N$) component of w_{opt}, can be solved by *lagrange* multiplier. $w_{opt.k}$ satisfies:

$$\frac{\partial\left(\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j C_{ij} - 2\lambda\left(\sum_{i=1}^{N} w_i - 1 \right) \right)}{\partial w_{opt.k}} = 0 \tag{22}$$

Equation (22) can be simplified as:

$$\sum_{\substack{j=1 \\ j\neq k}}^{N} w_{opt.k} C_{kj} = \lambda \tag{23}$$

Considering that $w_{opt.k}$ satisfies equation (2), we get:

$$w_{opt.k} = \frac{\sum_{j=1}^{N} C_{kj}^{-1}}{\sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij}^{-1}} \tag{24}$$

Although equation (24) is enough to solve w_{opt} in theory, it rarely works well in real-world applications. This is because in the ensemble of real-world applications there are often some individual neural networks that are quite similar in performance, which makes the correlation matrix $(C_{ij})_{N\times N}$ of the ensemble be an inreversible or ill-conditioned matrix so that equation (24) cannot be solved.

Since equation (21) can be viewed as an optimization problem, considering the success that has been obtained by genetic algorithms in optimization area [Goldberg, 1989], GASEN is proposed to find out the appropriate subset of the individual networks. After the individual networks being

trained, GASEN employs genetic algorithm to evolve the optimum weight vector w_{opt}. Then GASEN selects the individual networks whose corresponding optimum weight component is bigger than the pre-set threshold λ to constitute the ensemble. Note that if no individual network is washed out, i.e. every component of the evolved optimum weight vector is bigger than λ, all the individual networks are used to constitute the ensemble. We believe that this is corresponding to the situation that no individual networks satisfying equation (20). Following observation is noteworthy that the output of the ensemble is generated via *averaging*. In other words, the evolved optimum weight vector is only used in the selection of the individual networks. This is because we believe that using the weight vector both in the selection of the individual networks and the combination of the individual predictions is easy to cause overfitting.

Here GASEN is realized by utilizing the standard genetic algorithm [Goldberg, 1989] and a floating coding scheme that represents every component of the weight vector in 64 bits. Therefore each individual in the evolving population is coded in 8*N* bytes where *N* is the number of the individual networks. Note that since GASEN can be viewed as an abstract approach rather than a concrete algorithm, it can be easily realized by employing diversified kind of genetic algorithms and coding schemes.

Let *V* denotes the validation set. The estimated value of the correlation between the *i*-th and the *j*-th individual neural networks is:

$$C_{ij}^{V} = \frac{\sum_{x \in V} \left(f_i(x) - d(x)\right)\left(f_j(x) - d(x)\right)}{|V|} \qquad (25)$$

Considering equation (15), the estimated generalization error of the neural network ensemble corresponding to the individual *w* in the evolving population is:

$$E_w^V = \sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j C_{ij}^V = w^T C^V w \qquad (26)$$

It is obvious that E_W^V expresses the goodness of *w*. The smaller E_W^V is, the better *w* is. So we use $f(w) = 1/ E_W^V$ as the fitness function. Note that *w* may violate equation (2) during its evolving. Therefore it is necessary to do normalization on the evolved optimum *w* so that its components can be compared with λ. Here we use a simple normalization scheme:

$$w_{opt.i} = w_i / \sum_{i=1}^{N} w_i \qquad (27)$$

4 Experiments

We use four regression problems to compare the performance of three ensemble approaches, i.e. *averaging all*, *enumerating*, and GASEN.

The first problem is *Friedman#1* proposed by Friedman [1991]. There are 5 continuous attributes. The data set is generated according to equation (28) where the noise item ε satisfies normal distribution $N(0, 1)$ and x_i (i =1, 2, …, 5) satisfies uniform distribution $U[0, 1]$. In our experiments the size of the training set and the test set are respectively 200 and 1000.

$$t = 10\sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \varepsilon \qquad (28)$$

The second problem is *Boston Housing* from UCI machine learning repository [Blake *et al.*, 1998]. There are 11 continuous attributes and 1 categorical attribute. The data set comprises 506 examples among which 400 examples make up the training set and the rest 106 examples make up the test set in our experiments.

The third problem is *Ozone* proposed by Breiman and Friedman [1985]. There are 9 continuous attributes. The data set comprises 366 examples. Since the intention of the experiments is not to compare the ability of dealing with missing values, 1 attribute and 36 examples that has missing values are omitted as Briedman [1996] did on the data set. Therefore in our experiments there are 8 continuous attributes and 330 examples among which 250 examples make up the training set and the rest 80 examples make up the test set.

The fourth problem is *Servo* from UCI machine learning repository. There are 4 categorical attributes. The data set comprises 167 examples among which 130 examples make up the training set and the rest 37 examples make up the test set in our experiments. Note that some researchers [Quinlan, 1993] believe that this problem is very difficult because it involves some kind of extreme nonlinearity.

For each problem we use *Bagging* on the training set to generate 20 single-hidden-layered BP [Rumelhart *et al.*, 1986] networks. During the training process, the generalization error of each network is estimated in each epoch on a validation set generated via bootstrap sampling from the training set. If the error does not change in consecutive 5 epochs, the training of the network is terminated in order to avoid overfitting. Then we use *averaging all*, *enumerating*, and GASEN to ensemble those trained BP networks. In our experiments the genetic algorithm employed by GASEN is implemented by the GAOT toolbox developed by Houck *et al.* [1995]. The genetic operators, including select, crossover, and mutation, and the system parameters, including crossover probability, mutation probability, and stopping criterion, are all set to the default values of GAOT. The pre-set threshold λ used by GASEN is set to 0.05. The validation set *V* used by GASEN is also generated via bootstrap sampling from the training set. For every problem we perform 20 runs and record the average mean squared error and the standard deviation on

Table 1 Experimental results on *averaging all*, *enumerating*, and GASEN

Data set	*averaging all*		*enumerating*		GASEN	
	error	deviation	error	deviation	error	deviation
Friedman#1	1.33	0.26	0.47	0.12	0.5	0.14
Boston Housing	12.26	0.97	10.6	0.55	10.68	0.8
Ozone	22.29	2.00	19.85	1.72	19.99	1.63
Servo	0.21	0.026	0.21	0.045	0.226	0.058

the test set of the ensembles. The experimental results are tabulated in Table 1.

Pairwise one-tailed *t*-tests indicate that the generalization error of GASEN is significantly less than that of *averaging all* on *Friedman#1*, *Boston Housing*, and *Ozone*, while there is no significant difference between the performance of those two approaches on *Servo*. We believe that GASEN is superior to *averaging all* because ensembles generated by it are more accurate than that generated by *averaging all* in most cases. Pairwise one-tailed *t*-tests also indicate that there is no significant difference between the performance of GASEN and *enumerating* on all those data sets. Considering that *enumerating* can hardly work when there are lots of individual networks due to its extensive computational cost, we believe that GASEN is superior to *enumerating* because it can generate ensembles with comparable generalization ability in the cost of much smaller computation.

Following observation is interesting. Through analyzing the ensembles, we find that GASEN and *enumerating* averagely select a subset comprises 5 networks out of 20 individual networks to constitute the ensemble. And the subset selected by GASEN is the same as that selected by *enumerating* in 6 runs out of 20 runs on *Friedman#1*, 2 runs out of 20 runs on *Boston Housing*, 3 runs out of 20 runs on *Ozone*, and 7 runs out of 20 runs on *Servo*. It is obvious that the frequency of the appearance of same selected results is much higher than that should exhibit in random selection. Considering that *enumerating* is an optimum approach when the size of ensemble is small, we believe that this observation verifies the goodness of GASEN from another aspect. However, the reason for explaining the high frequency of the appearance of same selected results should be further explored.

5 Related Works

Neural network ensemble has become a very active area and there are a large number of research groups working on it. Besides the achievements cited in the brief review presented in Section 1, some significant developments in this area can be found in [Sharkey, 1999]. Moreover, there are some works very related to this paper.

Yao and Liu [1998] employed genetic algorithm to evolve a population of neural networks. Instead of choosing the best neural network in the last generalization as the final result, they regarded the entire population as a neural network ensemble and combining all the individuals in the last generalization in order to make best use of all the information contained in the population. Although genetic algorithm is used in both their and our works, there are many differences among which a conspicuous one is that they intended to utilize the information contained in the genetic population rather than performing selection on the individual networks.

Liu and Yao [1999] proposed an ensemble learning approach, i.e. negative correlation learning, where all the individual networks are trained simultaneously through the correlation penalty terms in their error functions. Rather than generating unbiased networks whose errors are uncorrelated, negative correlation learning can generate negatively correlated networks to encourage specialization and cooperation among the individual networks. The main reason that negative correlation learning can improve the generalization ability of an ensemble is that it increases the ambiguity item $\overline{A}$ in the famous equation $E = \overline{E} - \overline{A}$. In GASEN, when individual neural networks are selected according to the evolved optimum weight vector, the ambiguity of the ensemble is also increased. This can be derived from the observation that the generalization ability of the ensemble generated by GASEN is better than the ensemble that comprises same number of individual networks that rank toppest in generalization ability.

6 Conclusion

In this paper, the relationship between the generalization ability of the neural network ensemble and the correlation of individual networks is analyzed, which reveals that in some cases ensembling a selective subset of individual networks is superior to ensembling all the individual networks. Then a genetic algorithm based ensemble approach named GASEN is proposed. Experimental results show that GASEN is a promising ensemble approach that is superior to both *averaging all* and *enumerating*.

There are many works left to do in the near future. Firstly, at present GASEN has only been compared with *averaging all* and *enumerating* on several data sets. We plan to do comparisons with more ensemble approaches on more data sets. Secondly, the pre-set threshold λ is an important

parameter of GASEN, which determines the individual neural networks that constitute the ensemble. We hope to find out the relationship between λ and the generalization ability of the ensemble so that we can set λ to appropriate values in real-world applications. Thirdly, we want to explore why there is such a high frequency that GASEN and *enumerating* select the same subset of individual networks to make an ensemble.

Acknowledgements

The comments and suggestions from the anonymous reviewers greatly improve this paper. The National Natural Science Foundation of P.R.China and the Natural Science Foundation of Jiangsu Province, P.R.China, supported this research.

References

[Blake *et al.*, 1998] C. Blake, E. Keogh, and C. J. Merz. UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html], Department of Information and Computer Science, University of California, Irvine, California, 1998.

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

[Breiman and Friedman, 1985] L. Breiman and J. Friedman. Estimating optimal transformations in multiple regression and correlation (with discussion). *Journal of the American Statistical Association*, 80(391): 580-619, 1985.

[Cherkauer, 1996] K. J. Cherkauer. Human expert level performance on a scientific image analysis task by a system using combined artificial neural networks. In *Proceedings of the 13th AAAI Workshop on Integrating Multiple Learned Models for Improving and Scaling Machine Learning Algorithms*, pages 15-21, 1996. AAAI.

[Friedman, 1991] J. Friedman. Multivariate adaptive regression splines (with discussion). *Annals of Statistics*, 19(1): 1-141, 1991.

[Goldberg, 1989] D. E. Goldberg. *Genetic Algorithm in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, 1989.

[Gutta and Wechsler, 1996] S. Gutta and H. Wechsler. Face recognition using hybrid classifier systems. In *Proceedings of the IEEE International Conference on Neural Networks*, pages 1017-1022, 1996. IEEE Computer Society.

[Hansen and Salamon, 1990] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(10): 993-1001, 1990.

[Hansen *et al.*, 1992] L. K. Hansen, L. Liisberg L, and P. Salamon. Ensemble methods for handwritten digit recognition. In *Proceedings of the IEEE-SP Workshop on Neural Networks for Signal Processing*, pages 333-342, 1992. IEEE Computer Society.

[Houck *et al.*, 1995] C. R. Houck, J. A. Joines, and M. G. Kay. A genetic algorithm for function optimization: a Matlab implementation, NCSU-IE Technical Report 95-09, North Carolina State University, 1995.

[Huang *et al.*, 2000] F. J. Huang, Z.-H. Zhou, H.-J. Zhang, and T. Chen. Pose invariant face recognition. In *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 245-250, Grenoble, France, 2000. IEEE Computer Society.

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

[Liu and Yao, 1999] Y. Liu and X. Yao. Ensemble learning via negative correlation. *Neural Networks*, 12(10): 1399-1404, 1999.

[Mao, 1998] J. Mao. A case study on bagging, boosting and basic ensembles of neural networks for OCR. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, vol.3, pages 1828-1833, 1998. IEEE Computer Society.

[Quinlan, 1993] J. R. Quinlan. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, California, 1993.

[Rumelhart *et al.*, 1986] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by backpropagating errors. *Nature*, 323(9): 318-362, 1986.

[Sharkey, 1999] D. Sharkey, editor. *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems*, Springer-Verlag, London, 1999.

[Shimshoni and Intrator, 1998] Y. Shimshoni and N. Intrator. Classification of seismic signals by integrating ensembles of neural networks. *IEEE Trans. Signal Processing*, 46(5): 1194-1201, 1998.

[Sollich and Krogh, 1996] P. Sollich and A. Krogh. Learning with ensembles: How over-fitting can be useful. In *Advances in Neural Information Processing Systems 8*, pages 190-196, 1996. MIT.

[Yao and Liu, 1998] X. Yao and Y. Liu. Making use of population information in evolutionary artificial neural networks. *IEEE Trans. Systems, Man and Cybernetics – Part B: Cybernetics*, 28(3): 417-425, 1998.