Combining Neural-Based Regression Predictors Using an Unbiased and Normalized Linear Ensemble Model

Yunfeng Wu, Yachao Zhou, Sin-Chun Ng, and Yixin Zhong

Abstract— **In this paper, we combined a group of local regression predictors using a novel unbiased and normalized linear ensemble model (UNLEM) for the design of multiple predictor systems. In the UNLEM, the optimization of the ensemble weights is formulated equivalently to a constrained quadratic programming problem, which can be solved with the Lagrange multiplier. In our simulation experiments of data regression, the proposed multiple predictor system is composed of three different types of local regression predictors, and the effectiveness evaluation of the UNLEM was carried out on eight synthetic and four benchmark data sets. Results of the UNLEM's performance in terms of mean-squared error are significantly lower, in comparison with the popular simple average ensemble method. Moreover, the UNLEM is able to provide the regression predictions with a relatively higher normalized correlation coefficient than the results obtained with the simple average approach.**

I. Introduction

REGRESSION analysis is a statistical tool that helps develop a number of mathematical models in order to describe the causal effort of the dependent (or outcome) variable, upon the independent (or explanatory) variables. The regression analysis is very useful in various disciplines [1], such as data mining, pattern recognition, computational biology, and economics. So far, there have been rich investigations and applications about regression reported in the literature [2]–[6]. Despite that conventional relevant techniques can provide a good error-bar prediction in low dimensions, most of such approaches most of such approaches are not robust enough, when dealing with high-dimensional data points [7]. One promising solution is considered to design multiple predictor systems, in which a group of local regression predictors (LRPs) are combined to provide an overall prediction.

A multiple predictor system can be constructed by means of the ensemble methods [7], [8], which can rectify the warps of LRPs in real-world applications [9]–[15]. In other words, the ensemble techniques can help a multiple predictor system fuse the knowledge generated by its LRPs and make a consensus decision [16], which is expected to be more accurate than the one provided by an individual LRP. Nowadays, the advantages of the ensemble techniques for design of multiple predictor systems have been widely accepted by international professional communities [17], and the ensemble learning methods have been effectively utilized to solve complex regression problems [10], [18].

The pioneering ensemble algorithms in the literature are Boosting [19]–[21] and Bagging [7], [10], [22], [23]. Boosting works by repeatedly implementing a given weak-learning predictor on different training data sets with certain distributions, and then combining these predictors. The distribution of the training data set in the current iteration depends on the performance of prior predictors. The first version of the Boosting algorithm, proposed by Schapire [19], is Boosting-by-filtering, which involves a data filtering procedure with a weak-learning algorithm. Such approach, unfortunately, requires a large size of training data set, which limits its effectiveness in many practical applications. In order to overcome such a shortcoming, Freund and Schapire then proposed the AdaBoost [20] that tries to find a typical mapping function or hypothesis with a low error rate, in relation to a given probability distribution of training data. For regression, Freund and Schapire developed the AdaBoost.R [20] as an alternative solution. In spite of the effectiveness, the Boosting algorithms still result in some pitfalls [21], [24]. First, they have to project a regression data set into several classification sets, and the number of projected classification instances grows intensively larger after just a few boosted iterations. Second, the loss function varies from one iteration to another, and even changes between instances in the same iteration. In addition, the Boosting algorithms are very sensitive to outliers, and sometimes may cause over-fitting. The Bagging algorithm, on the other hand, introduces the bootstrap sampling procedure [25] into the construction of LRPs, with the suppose to generate enough independent variance among the LRPs [7]. The bias of the Bagging ensemble will converge through the bootstrap sampling and averaging procedures, whereas the variance gets much smaller than that obtained by any of its LRPs.

Recently, linear combination models, [18], [26]–[28], have been frequently applied in the Bagging, AdaBoost, and other popular ensemble methods. The simple average (SA) ensemble model is most popular and widely used, due to its simplicity. However, the SA approach treats all the LRPs equally, and is not able to make use of the knowledge generated by them [29]. The superiority of the weighted average approach, proposed by Fumera [26], cannot always be guaranteed in practical applications [30], because such algorithm suffers from estimating weights according to the

Mr. Y. F. Wu and Prof. Y. X. Zhong are with the School of Information Engineering, Beijing University of Posts and Telecommunications, No. 10 Xi Tu Cheng Road, Haidian District, Beijing 100876, China (e-mail: y.wu@ieee.org).

Ms. Yachao Zhou is with the Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China (e-mail: zhouyachao@gmail.com).

Dr. S. C. Ng is with the School of Science and Technology, The Open University of Hong Kong, 30 Good Shepherd Street, Homantin, Kowloon, Hong Kong (e-mail: scng@ouhk.edu.hk).

978-1-4244-1821-3/08/$25.00©2008 IEEE

possibility density function of the added errors from noisy data [27]. In this paper, we propose an unbiased and normalized linear ensemble model (UNLEM) to perform the ensemble in the multiple predictor system. The solution of optimum weights of the UNLEM can be obtained by solving a constrained quadratic programming problem with the Lagrange multiplier.

The rest parts of this paper are organized as follows. Section II describes the proposed UNLEM method, and derives an equivalent quadratic programming formulation, which provides the solution of optimum weights that linearly combine the LRPs. The regression experiments and simulation results on a number of synthetic and benchmark data sets are presented in Section III. Finally, Section IV concludes the paper with an emphasis of the merits of the UNLEM, and also comments on the future work.

II. The Unbiased and Normalized Linear Ensemble Model

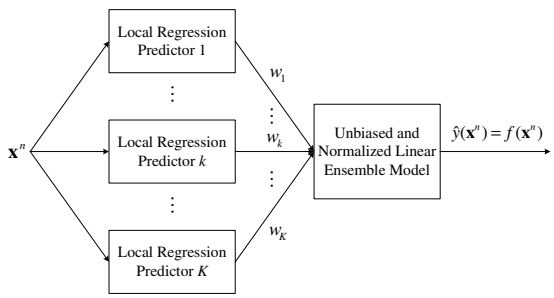

Fig. 1. Illustration of the multiple predictor system with the UNLEM.

An overview of the multiple regression predictor system with the proposed UNLEM is shown in Fig. 1. Assuming the system is composed of total K LRPs, the UNLEM combines the output from a group of LRPs, $r_k(\mathbf{x}^n)$, $k = 1, 2, \ldots, K$, and provide the ensemble prediction with regard to the nth input vector of instances, $\mathbf{x}^n$, $n = 1, 2, \ldots, N$. The UNLEM output, $f(\mathbf{x}^n)$, provides the estimated prediction, $\hat{y}(\mathbf{x}^n)$, with regard to the nth input instance. The corresponding mathematical expression of the UNLEM can be formulated as

$$f(\mathbf{x}^n) = \hat{y}(\mathbf{x}^n) = \sum_{k=1}^{K} w_k r_k(\mathbf{x}^n), \tag{1}$$

and the weights of the UNLEM satisfy the normalization condition given by

$$\sum_{k=1}^{K} w_k = 1. \tag{2}$$

The mean-squared error (MSE) between the prediction of the kth LRP and the desired values, $y(\mathbf{x}^n)$, over the total N instances, can be expressed as

$$E_k(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^{N} [y(\mathbf{x}^n) - r_k(\mathbf{x}^n)]^2. \tag{3}$$

Since the weights are normalized, as presented in (2), a desired prediction value $y(\mathbf{x}^n)$ can be split with a weight multiplier, such that the overall MSE of the UNLEM is then derived as follows:

$$\begin{aligned} E(\mathbf{x}) &= \frac{1}{N} \sum_{n=1}^{N} \left[y(\mathbf{x}^n) - \sum_{k=1}^{K} w_k r_k(\mathbf{x}^n) \right]^2 \\ &= \frac{1}{N} \sum_{n=1}^{N} \left[\sum_{k=1}^{K} w_k y(\mathbf{x}^n) - \sum_{k=1}^{K} w_k r_k(\mathbf{x}^n) \right]^2 \\ &= \sum_{k=1}^{K} w_k^2 \frac{1}{N} \sum_{n=1}^{N} [y(\mathbf{x}^n) - r_k(\mathbf{x}^n)]^2 \\ &= \sum_{k=1}^{K} w_k^2 E_k(\mathbf{x}). \end{aligned} \tag{4}$$

Because the MSE of the kth LRP, $E_k(\mathbf{x})$, can be estimated when the details of the LRP and the specific regression data are given, the optimum weights $\hat{w}_k$ of the UNLEM that minimize the overall MSE of the multiple predictor system, is equivalent to the solution of the following constrained quadratic programming (QP) problem as

$$\begin{cases} \text{minimize} & E(\mathbf{x}) = \sum\limits_{k=1}^{K} w_k^2 E_k(\mathbf{x}), \\ \text{subject to} & \sum\limits_{k=1}^{K} w_k = 1. \end{cases} \tag{5}$$

To solve this QP problem, we can define a Lagrange function of (5) as below

$$L(w_k, \lambda) = \sum_{k=1}^{K} w_k^2 E_k(\mathbf{x}^n) - 2\lambda \left(\sum_{k=1}^{K} w_k - 1 \right), \tag{6}$$

where λ is referred to the Lagrange multiplier. According to the normalization condition in (2), $\hat{w}_k$ are the estimated optimum weights of the UNLEM if and only if there exits $\lambda^* \in \Re$ as a solution to (5) such that ($\hat{w}_k$, λ^*) satisfies the following condition:

$$\begin{cases} \frac{\partial}{\partial \hat{w}_k} \left[\sum\limits_{k=1}^{K} \hat{w}_k^2 E_k(\mathbf{x}) - 2\lambda^* \left(\sum\limits_{k=1}^{K} \hat{w}_k - 1 \right) \right] = 0, \\ \sum\limits_{k=1}^{K} \hat{w}_k - 1 = 0. \end{cases} \tag{7}$$

By solving (7), the solution of λ^* can be derived as

$$\lambda^* = \hat{w}_k E_k(\mathbf{x}), \tag{8}$$

and then, the optimum weights of the UNLEM that minimize the MSE of the ensemble are derived as

$$\hat{w}_k = \frac{E_k^{-1}(\mathbf{x})}{\sum\limits_{k=1}^{K} E_k^{-1}(\mathbf{x})}, \tag{9}$$

from which we can infer that the LRP that provides the lowest MSE will be assigned the largest value of the weight, which is reasonable in accord with the engineering experience in practice.

TABLE I

DESCRIPTIONS OF SYNTHETIC REGRESSION DATA SETS

Name of data sets	Function expression	Distributions of Independent variables	Number of instances
Zigzag	$y = \sin x^2 \cos x^2 - 0.25x$	$x \sim \mathrm{U}[0, 3]$†	1500
Rhythm	$y = \left[\frac{\mathrm{mod}\ (x,11)-5}{8}\right]^3$	$x \sim \mathrm{U}[0, 20]$	1000
SinCos	$y = x \sin x \cos x$	$x \sim \mathrm{U}[0, 2\pi]$	2000
Gabor	$y = \frac{\pi}{2} \exp\left[-2(x_1^2 + x_2^2)\right] \cos\left[2\pi(x_1 + x_2)\right]$	$x_i \sim \mathrm{U}[0, 1], i = 1, 2$	2000
Friedman-1	$y = 10 \sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5$	$x_i \sim \mathrm{U}[0, 1], i = 1, 2, \ldots, 5$	5000
Friedman-2	$y = \sqrt{x_1^2 + \left[x_2 x_3 + \left(\frac{1}{x_2 x_4}\right)\right]^2}$	$x_1 \sim \mathrm{U}[0, 100]$ $x_2 \sim \mathrm{U}[40\pi, 560\pi]$ $x_3 \sim \mathrm{U}[0, 1]$ $x_4 \sim \mathrm{U}[1, 11]$	3000
Friedman-3	$y = \tan^{-1}\left[\frac{x_2 x_3 - \frac{1}{x_2 x_4}}{x_1}\right]$	$x_1 \sim \mathrm{U}[0, 100]$ $x_2 \sim \mathrm{U}[40\pi, 560\pi]$ $x_3 \sim \mathrm{U}[0, 1]$ $x_4 \sim \mathrm{U}[1, 11]$	3000
Polynomial	$y = 1 + 2x + 3x^2 + 4x^3 + 5x^4$	$x \sim \mathrm{U}[0, 1]$	1000

†U[a, b] means a uniform distribution over the interval from a to b.

III. EXPERIMENTS AND SIMULATION RESULTS

A. Description of regression data sets

To study the effectiveness of the proposed UNLEM method, we used total 12 regression data sets, including eight synthetic sets and four benchmark sets, to test the multiple regression predictor system. Details of the constraints with regard to the independent variables are listed in Table I. The Zigzag, Rhythm, and Polynomial data sets were used in [10] to test the Bagging-based least-mean-square (Bagging.LMS) fusion algorithm. The Gabor data set was used to test the unbiased linear neural-based fusion method presented in [18]. The Friedman-1, Friedman-2, and Friedman-3 data sets was ever used by Briedman [7] to test the regression performance of Bagging. The benchmark data sets, listed in Table II, were obtained from the UCI Machine Learning Repository [31] and the StatLib[1], respectively. The attributes indicated in Table II are referred to the independent variables in the regression experiments.

TABLE II

DESCRIPTIONS OF BENCHMARK REGRESSION DATA SETS

Name of data sets	Number of attributes	Number of instances	Data source
Abalone	7	4177	UCI
Boston Housing	13	506	UCI
Pollution	15	60	StatLib
Body Fat	14	252	StatLib

B. Experiments

In our experiments, we employed three different types of three-layer feedforward artificial neural networks (ANNs) as the LRPs to construct the multiple predictor system, because Cybenko [32] has justified that an ANN with a single hidden layer is able to perform function approximation with an arbitrary accuracy. The hidden layer of each ANN contains 10 hidden nodes, for the purpose of comparison, but the activation functions and training algorithms for the ANNs differ from one to another. The first LRP is a radial basis function (RBF) network with the nonlinear kernel function given by

$$\phi(\mathbf{x}^n, \mathbf{c}_j) = \exp\left(-\ln 2 \frac{\|\mathbf{x}^n - \mathbf{c}_j\|^2}{\sigma^2}\right), \qquad (10)$$

where $\mathbf{c}_j$, $j = 1, 2, \ldots, 10$, represents the center vector for the jth node in the hidden layer, and σ denotes the spread parameter ($\sigma = 2.0$ in our experiments) that determines the width of the area in the input space to which each hidden node responds. In the present study, we applied the orthogonal least-squares algorithm [33], a systematic method for center selection, which is able to significantly reduce the size of the RBF hidden nodes. The second LRP is also a

[1]Online available: http://lib.stat.cmu.edu/datasets/

RBF network, but with the thin plate spline function [34] as a kernel. The third LRP is a multi-layer perceptron (MLP), activated by the tan-sigmoid transfer function [35] in its hidden layer. The MLP was trained with the Levenberg-Marquardt algorithm [36]. Each regression data set was tested with the hold-out procedure [34], i.e., the data were randomly partitioned into two equal disjointed subsets, one for the training of ensemble weights and the other for performance testing. The MSEs of the LRPs, estimated by means of the training subset, were used to optimize the weights of the UNLEM. The ensemble performance was evaluated through the testing subset by the measures of MSE and normalized correlation coefficient (NCC) in percentage, the latter one is defined as

$$\mathrm{NCC}(\%) = 100 \times \frac{\sum_{n=1}^{N} y(\mathbf{x}^n) f(\mathbf{x}^n)}{\sqrt{\sum_{n=1}^{N} y(\mathbf{x}^n) \sum_{n=1}^{N} f(\mathbf{x}^n)}}. \qquad (11)$$

For statistical analysis, the experiment on each regression data set was repeated 10 times, recorded as 10 trials. We also implemented the widely used SA ensemble approach in all experiments, for the purpose of result comparison.

C. Results

Fig. 2 plots the data points obtained with the SA and the UNLEM methods in the first trial of four experiments, in relation to their expected one-dimensional regression curves. It is clear that the data points predicted by the UNLEM method are closer to the expected curves, in particular on the tails of the Zigzag and Rhythm curves, as shown in Fig. 2(a) and (b), respectively. In addition, the proposed UNLEM improves the prediction accuracy better than the SA in terms of MSE, as listed in Table III. Compared with the SA, the UNLEM can provide the remarkably lower MSEs, in particular MSE reduction of 1.97x10^{-3}, 5.7, 1.96x10^{-3}, 8.26x10^{4}, and 7.54x10^{-4}, for the SinCos, Friedman-2, Friedman-3, Pollution, and Body Fat data sets, respectively.

Concerning the NCC evaluation criterion, which is most frequently used to measure the association in time-series prediction [37]. It can be observed from Table IV that both of the SA and the UNLEM can characterize the nature of regression quite well (95% as significant). According to Table IV, the results of the UNLEM are slightly superior to those of the SA in most regression experiments, expect for the Gabor and Abalone data sets. The reason can be explained that the weights of the UNLEM optimized by the training set during the hold-out procedure do not make a good generalization on the testing set.

IV. Conclusions

The proposed UNLEM method is simple to implement, and the solution of the optimum weights can be derived by solving the equivalent constrained quadratic programming problem described in terms of the overall MSE. Our simulation experiments demonstrate that the UNLEM can

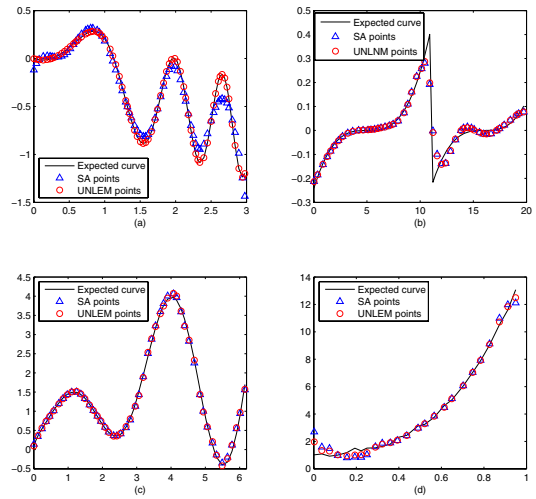

Fig. 2. Plots of the ensemble predictions obtained with the SA and the UNLEM methods on the one-dimensional regression data sets: (a) Zigzag, (b) Rhythm, (c) SinCos, (d) Polynomial.

TABLE III
MSE of each regression data set obtained with the SA and the UNLEM methods

Data sets	SA		UNLEM	
	Mean	SD‡	Mean	SD
Zigzag	0.98x10^{-2}	0.25x10^{-2}	0.45x10^{-2}	1.15x10^{-2}
Rhythm	0.98x10^{-2}	0.25x10^{-2}	0.45x10^{-2}	1.15x10^{-2}
SinCos	0.26x10^{-2}	0.24x10^{-2}	6.29x10^{-4}	0.19x10^{-2}
Gabor	0.09	0.30x10^{-2}	0.09	0.76x10^{-2}
Friedman-1	0.16x10^{-2}	8.21x10^{-4}	0.12x10^{-2}	0.35x10^{-2}
Friedman-2	6.17	4.68	0.47	0.44
Friedman-3	0.20x10^{-2}	5.81x10^{-4}	3.83x10^{-5}	4.33x10^{-5}
Polynomial	0.13	0.06	0.02	0.02
Abalone	4.78	0.17	4.70	0.51
Boston Housing	51.66	8.05	47.00	19.11
Pollution	8.81x10^{4}	1.06x10^{5}	5.53x10^{3}	2.23x10^{3}
Body Fat	8.53x10^{-4}	0.11x10^{-2}	9.86x10^{-5}	1.27x10^{-4}

‡SD: standard deviation.

TABLE IV

NCC (%) OF EACH REGRESSION DATA SET OBTAINED WITH THE SA AND THE UNLEM METHODS

Data sets	SA		UNLEM	
	Mean	SD	Mean	SD
Zigzag	98.37	0.43	99.19	2.11
Rhythm	98.37	0.43	99.18	2.11
SinCos	99.96	0.03	99.99	0.03
Gabor	89.75	0.37	89.54	0.98
Friedman-1	99.99	$1.65\text{x}10^{-4}$	99.99	$7.11\text{x}10^{-4}$
Friedman-2	96.42	2.85	99.99	$3.53\text{x}10^{-5}$
Friedman-3	99.95	0.01	99.99	0.01
Polynomial	99.83	0.08	99.97	0.02
Abalone	97.79	0.06	97.73	0.24
Boston Housing	95.66	0.72	95.96	1.78
Pollution	95.52	5.27	99.69	0.12
Body Fat	99.96	0.05	99.99	0.01

effectively combine the LRPs in a multiple predictor system to solve regression problems. The evaluation criteria of MSE and NCC measure the prediction accuracy and fidelity, and the results on the synthetic and benchmark regression data sets show that the UNLEM method can outperform the popular SA approach, leading a much lower MSE and relatively higher NCC. The future work could be directed toward a study of the UNLEM for design of multiple classifier systems.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation of China under Grant No. 60575034; the Doctoral Program Foundation of the Ministry of Education of China under Grant No. 20060013007; and the 2005 Innovation Research Funds from the Graduate School, Beijing University of Posts and Telecommunications. Mr. Y. F. Wu has been supported by the Croucher Foundation in Hong Kong in the form of a "Visitorship for Scholar in Mainland of China" in 2007.

REFERENCES

[1] A. Von Eye, *Regression Analysis for Social Sciences*. San Diego, CA: Academic Press, 1998.

[2] D. A. Belsley, E. Kuh, and R. Welsch, *Regression Diagnostics*. New York, NY: John Wiley and Sons, 1980.

[3] P. J. Huber, *Robust Statistics*. New York, NY: Wiley, 1981.

[4] R. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outliner Detection*. New York, NY: Wiley, 1987.

[5] G. A. F. Seber and C. J. Wild, *Nonlinear Regression*. New York, NY: John Wiley and Sons, 1989.

[6] L. Breiman, "Stacked regressions," *Machine Learning*, vol. 24, no. 1, pp. 49–64, 1996.

[7] ——, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[8] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matus, "On combining classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[9] Y. F. Wu and S. C. Ng, "Breast tissue classification based on unbiased linear fusion of neural networks with normalized weighted average algorithm," in *Proc. 2007 Int'l Joint Conf. Neural Networks (IJCNN'07)*, Orlando, FL, USA, 2007, pp. 2846–2850.

[10] Y. F. Wu, C. Wang, and S. C. Ng, "Bagging.LMS: A Baggging-based linear fusion with least-mean-square error update for regression," in *Proc. 2006 IEEE Region 10 Conf. (TENCON'06)*, Hong Kong, 2006, pp. 418–421.

[11] Y. F. Wu and S. C. Ng, "Combining neural learners with the naive Bayes fusion rule for breast tissue classification," in *Proc. 2nd IEEE Conf. Industrial Electronics and Applications (ICIEA'07)*, Harbin, China, 2007, pp. 709–713.

[12] Y. F. Wu, Y. Z. Ma, X. N. Liu, and C. Wang, "A bootstrap-based linear classifier fusion system for protein subcelluar location prediction," in *Proc. 28th Annu. Int'l Conf. IEEE Eng. Med. Biol. Soc. (EMBC'06)*, New York, NY, USA, 2006, pp. 4229–4232.

[13] Y. F. Wu and C. Wang, "Linear least-squares fusion of multilayer perceptrons for protein localization sites prediction," in *Proc. 32nd IEEE Annu. Northeast Bioeng. Conf. (NEBC'06)*, Easton, PA, USA, 2006, pp. 157–158.

[14] Y. F. Wu, J. M. Zhang, C. Wang, and S. C. Ng, "Linear decision fusions in multilayer perceptrons for breast cancer diagnosis," in *Proc. 17th IEEE Int'l Conf. Tools with Artificial Intelligence (ICTAI'05)*, Hong Kong, 2005, pp. 699–700.

[15] Y. F. Wu, J. J. He, Y. Man, and J. I. Arribas, "Neural network fusion strategies for identifying breast masses," in *Proc. 2004 IEEE Int'l Joint Conf. Neural Networks (IJCNN'04)*, Budapest, Hungary, 2004, pp. 2437–2442.

[16] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999.

[17] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Hoboken, NJ: Wiley, 2004.

[18] Y. F. Wu and S. C. Ng, "Unbiased linear neural-based fusion with normalized weighted average algorithm for regression," in *Proc. 4th Int'l Symp. Neural Networks (ISNN'07), LNCS 4493*, Nanjing, China, 2007, pp. 664–670.

[19] R. E. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, pp. 197–227, 1990.

[20] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[21] D. P. Solomatine and D. L. Shrestha, "AdaBoost.RT: A boosting algoirthm for regression probelms," in *Proc. 2004 IEEE Int'l Joint Conf. Neural Networks (IJCNN'04)*, Budapest, Hungary, 2004, pp. 1163–1168.

[22] D. Hernandez-Lobato, G. Martinez-Munoz, and A. Suarez, "Pruning in ordered regression bagging ensembles," in *Proc. 2006 IEEE Int'l Joint Conf. Neural Networks (IJCNN'06)*, Vancouver, BC, Canada, 2006, pp. 1266–1273.

[23] P. L. Braga, A. L. I. Oliveira, G. H. T. Ribeiro, and S. R. L. Meira, "Bagging predictors for estimation of software project effort," in *Proc. 2007 IEEE Int'l Joint Conf. Neural Networks (IJCNN'07)*, Orlando, FL, USA, 2007, pp. 1595–1600.

[24] G. Ridgeway, "The state of boosting," *Computing Science and Statistics*, vol. 31, pp. 172–181, 1999.

[25] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. New York, NY: Chapman and Hall, 1993.

[26] G. Fumera and F. Roli, "A theoretical and experimental analysis of linear combiners for multiple classifier systems," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 942–956, 2005.

[27] K. Tumer and J. Ghosh, "Analysis of decision boundaries in linearly combined neural classifiers," *Pattern Recognition*, vol. 29, no. 2, pp. 341–348, 1996.

[28] L. I. Kuncheva, "A theoretical study on six classifier fusion strategies," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 281–286, 2002.

[29] Y. F. Wu and J. I. Arribas, "Fusing output information in neural networks: Ensemble performs better," in *Proc. 25th IEEE EMBS Annu. Int'l Conf. (EMBC'03)*, Cancun, Mexico, 2003, pp. 2265–2268.

[30] Y. F. Wu, C. Wang, S. C. Ng, A. Madabhushi, and Y. X. Zhong, "Breast cancer diagnosis using neural-based linear fusion strategies," in *Proc. 13th Int'l Conf. Neural Information Processing (ICONIP'06), LNCS 4234*, Hong Kong, 2006, pp. 165–175.

[31] A. Asuncion and D. J. Newman, "UCI machine learning repository," University of California, Irvine, School of Information and Computer Sciences, 2007. [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html
[32] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303–314, 1989.
[33] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.
[34] S. Haykin, *Neural Networks: A Comprehensive Foundation, 2nd ed.* Englewood Cliffs, NJ: Prentice Hall PTR, 1998.
[35] T. M. Mitchell, *Machine Learning*. Columbus, OH: McGraw-Hill, 1997.
[36] M. T. Hagan and M. Menhaj, "Training feedforward networks with the marquardt algorithm," *IEEE Trans. Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.
[37] J. P. Marques de Sa, *Applied Statistics using SPSS, STATISTICA, and MATLAB*. Berlin, Germany: Springer-Verlag, 2003.