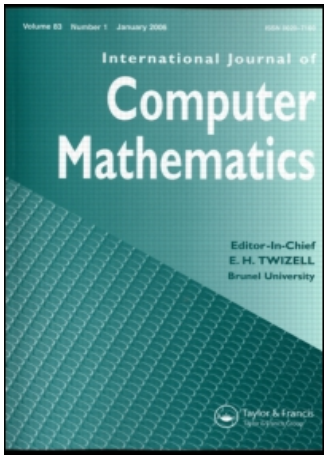


International Journal of Computer Mathematics
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713455451

LINEAR NEURAL NETWORK TRAINING ALGORITHMS FOR REAL-WORLD BENCHMARK PROBLEMS

K. Goulianas [a]; M. Adamopoulos; S. Katsavounis [b]; Ch. FRAGAKIS [b]; C. C. Tsouros [b]
[a] Department of Informatics, Technological Educational Institute of Thessaloniki, Greece.
[b] Faculty of Engineering, Aristotle University of Thessaloniki, Greece.

Online Publication Date: 01 January 2002
To cite this Article: Goulianas, K., Adamopoulos, M., Katsavounis, S., FRAGAKIS, Ch. and Tsouros, C. C. (2002) 'LINEAR NEURAL NETWORK TRAINING ALGORITHMS FOR REAL-WORLD BENCHMARK PROBLEMS', International Journal of Computer Mathematics, 79:11, 1149 - 1167
To link to this article: DOI: 10.1080/00207160213945
URL: http://dx.doi.org/10.1080/00207160213945

Intern. J. Computer Math., 2002, Vol. 79(11), pp. 1149–1167

LINEAR NEURAL NETWORK TRAINING ALGORITHMS FOR REAL-WORLD BENCHMARK PROBLEMS

K. GOULIANAS[a], M. ADAMOPOULOS[a,b], S. KATSAVOUNIS[c] Ch. FRAGAKIS[c] and C. C. TSOUROS[c]

[a]*Department of Informatics, Technological Educational Institute of Thessaloniki, Greece;*
[b]*Department of Informatics, University of Macedonia, Thessaloniki, Greece;*
[c]*Faculty of Engineering, Aristotle University of Thessaloniki, Greece*

(Received 22 June 2001; In final form 1 July 2001)

This paper describes the Adaptive Steepest Descent (ASD) and Optimal Fletcher-Reeves (OFR) algorithms for linear neural network training. The algorithms are applied to well-known pattern classification and function approximation problems, belonging to benchmark collection Proben1. The paper discusses the convergence behavior and performance of the ASD and OFR training algorithms by computer simulations and compares the results with those produced by linear-RPROP method.

Keywords: Neural nets; Training algorithms; Iterative methods

C.R. Categories: I.5.1, I.2.6, G.1.3

1 INTRODUCTION

Linear feedforward neural network architectures have been proved capable of solving systems of linear equations [2, 5–7, 14, 15], and pattern classification problems [8]. Most of these applications use LMS and Batch-LMS training [4, 17, 18], and require a selection of appropriate parameters by the user, executed with a trial-and-error process. In real world problems, it is essential to consider learning methods with a good average performance. This paper describes some methods that have been shown to accelerate the convergence of the learning phase, and that do not require the choice of critical parameters, like the learning rate or the momentum. A simple two-layer feedforward neural network with linear neuron functions is studied. Batch-LMS training is extended to implement steepest descent algorithms, like the Adaptive Steepest Descent (ASD) and the Optimal Fletcher-Reeves (OFR) algorithm. The performance of these algorithms is compared to linear-RPROP training [9], a technique for optimizing the backpropagation training [10, 11], which uses

ISSN 0020-7160 print; ISSN 1029-0265 online
DOI: 10.1080/0020716021000030684

a fixed update size not influenced by the magnitude of the gradient. Instead, only the sign of the derivative is used to find the proper update direction. Those three methods are applied to well-known benchmarks from the Proben1 collection. Proben1 contains data from the UCI repository of machine learning databases for 9 pattern classification problems and 3 function approximation problems. The 9 pattern classification problems are the following: *cancer* (a dataset for diagnosis of breast cancer, originally obtained from the University of Wisconsin Hospital, Madison, from Dr. William H. Wolberg), *card* (a dataset for approval or non-approval of a credit card to a customer), *diabetes* (a dataset for diagnosis of diabetes of Pima Indians), *gene* (a dataset for detection of intron/exon boundaries in nucleotide sequences), *glass* (a dataset for classification of glass types), *heart* (a dataset for heart disease prediction), *horse* (a dataset for prediction of the fate of a horse that has a colic), *soybean* (a dataset for recognition of 19 diseases of soybeans), and *thyroid* (a dataset for diagnosis of thyroid hyper- or hypofunction). The 3 function approximation problems are the following: *building* (a dataset for prediction of energy consumption in a building), *flare* (a dataset for prediction of sonar flares), and *hearta* (the analogue version of the heart disease diagnosis problem). All the datasets are partitioned into training, validation, and test set, while the size of the training, validation, and test set data files is 50%, 25%, and 25% respectively. Since results may vary for different partitionings, Proben1 contains three different permutations of each dataset. For instance, the problem *cancer* is available in three datasets *cancer1, cancer2*, and *cancer3*, which differ only in the ordering of the patterns. Validation set is used as a pseudo test set in order to evaluate the quality of the network during training, a method called cross-validation, which avoids overfitting, a problem created when many training examples are available causing the loss of much of the regularities needed for good generalization [13]. The method used for cross-validation is early stopping [1, 12, 16], where training proceeds until a minimum of the error on validation set (and not the training set) is reached.

The formulation of the above problems is as follows: Given a set of input patterns $x^i = [x_1^i, x_2^i, \ldots, x_n^i]^{\mathrm{T}}$ and the targets $d^i = [d_1^i, d_2^i, \ldots, d_p^i]^{\mathrm{T}}, i = 1, 2, \ldots, m$ the task is to find a set of weights $W \in \Re^{n \times p}$ that provides the best fit between the input/output pairs $(x^i, d^i), i = 1, 2, \ldots, m$. A simple linear architecture for the above mapping is a hetero-associative two-layer feedforward neural network, with n inputs and p output neurons shown in Figure 1. The m patterns are presented to the input layer in a cyclical fashion and an output

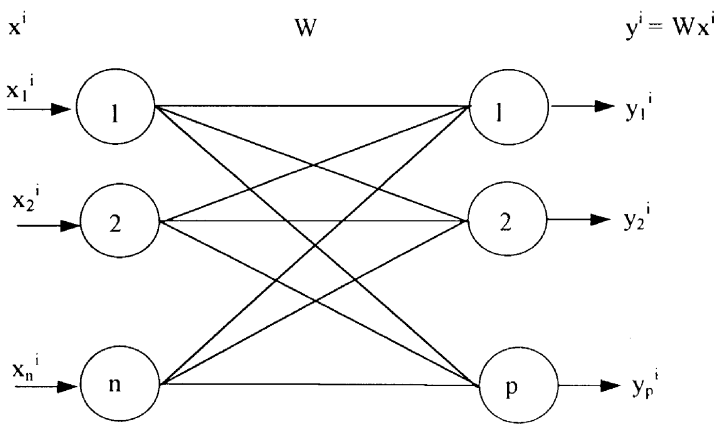


FIGURE 1 Linear neural network for classification and function approximation problems.

$y^i = [y_1^i, y_2^i, \ldots, y_p^i]^{\mathrm{T}}$ is generated. The goal is to minimize the mean square error, or the cost function

$$E(W) = \sum_{j=1}^{p} E(w^j) = \frac{1}{2}\sum_{j=1}^{p}\sum_{i=1}^{m}(d_j^i - y_j^i)^2 = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{p}\left(d_j^i - \sum_{k=1}^{n} x_k^i w_k^j\right)^2$$

$$= \frac{1}{2}\left\|X^{\mathrm{T}}W - D\right\|^2 \tag{1}$$

With $X = [x^1, x^2, \ldots, x^m]$, and $D = [d^1, d^2, \ldots, d^m]$, w^j the weight vector of the jth output neuron, and $E(w^j)$ its cost function. Using a general gradient approach, any minimum of the cost function in Eq. (1) must satisfy $\nabla E(W) = X(X^{\mathrm{T}}W - D) = 0$, which can be rewritten as

$$XX^{\mathrm{T}}W = XD \tag{2}$$

or the equivalent system $BW = C$ with $B = XX^{\mathrm{T}}$, $B \in \Re^{n\times n}$ and $C = XD$, $C \in \Re^n$. Equation (2) consists of p systems of normal equations, with $B = XX^{\mathrm{T}}$ positive definite and symmetric. The solution of systems (2) for a non-singular matrix X^{T} with $m \geq n$, gives the unique least mean square solution $W = X^+D$ with $X^+ = (XX^{\mathrm{T}})^{-1}X$ the Moore-Penrose generalized inverse [3].

The cost functions $E(w^j)$ defined in (1) are quadratic in the weights w^j and for $m \geq n$ they define convex hyper-paraboloidal surfaces with a single minimum, the global minimum, the solutions of Eq. (2), which are unique, since the Hessian matrix of $E(w^j)$ at w^j, $\nabla^2 E(w^j) = XX^{\mathrm{T}}$ is positive definite.

Since the mean squared error defined in (1) depends on the number of output coefficients, and on the range of the output values used, Prechelt [8] suggests the use of the squared error percentage, a normalization of these factors, as follows:

$$E(W) = \sum_{j=1}^{p} E\left(w^j\right) = 100 \cdot \frac{(o_{\max} - o_{\min})}{m \cdot p}\sum_{j=1}^{p}\sum_{i=1}^{m}\left(d_j^i - y_j^i\right)^2 \tag{3}$$

with $o_{\min}$ and $o_{\max}$ to be the minimum and maximum value of output coefficients, p the number of output nodes, and m the number of patterns in the dataset.

The material is organized as follows. In Section 2 we simulate the ASD and OFR algorithm with the above architecture and discuss convergence issues for obtaining estimates of the optimal solution. In Section 3, we compare the performance of the ASD and OFR training algorithm with linear-RPROP in the solution of the pattern classification and function approximation benchmarking problems. Finally, in Section 4, we draw some final conclusions.

2 ASD AND OFR METHODS FOR LINEAR NEURAL NETWORK TRAINING

With zero or random in $[-0.01, 0.01]$ initial weights w_k^j, $j = 1, 2, \ldots, p$, $k = 1, 2, \ldots, n$ the train set patterns $x^i = [x_1^i, x_2^i, \ldots, x_n^i]^{\mathrm{T}}$, $i = 1, 2, \ldots, m$ are presented to the network in a

cyclical fashion. Thus, with the presentation of the ith pattern, the outputs $y_j^{(t+1,i)}$, $j = 1, 2, \ldots, p$, will be

$$y_j^{(t+1,i)} = w^{(t,j)}x^i = \sum_{k=1}^{n} w_k^{(t,j)} x_k^i \tag{4}$$

with $(t+1, i)$ the step i of the training cycle $t+1$. Delta Rule is applied, and the discrepancy between desired and calculated output d_j^i and $y_j^{(t+1,i)}$, for every pattern i, $i = 1, 2, \ldots, m$ and output neuron j, $j = 1, 2, \ldots, p$ is

$$\varepsilon_j^{(t+1,i)} = d_j^i - y_j^{(t+1,i)} \tag{5}$$

We define the batch error $\delta_k^{(t+1)}$ for every input neuron k, $k = 1, 2, \ldots, n$ to be

$$\delta_k^{(t+1)} = \sum_{i=1}^{m} \varepsilon_j^{(t+1,i)} x_k^i = \sum_{i=1}^{m} \left(d_j^i - y_j^{(t+1,i)} \right) x_k^i \tag{6}$$

or in a matrix-vector form

$$\delta^{(t+1)} = \sum_{i=1}^{m} \varepsilon_j^{(t+1,i)} x^i = \sum_{i=1}^{m} \left(d_j^i - y_j^{(t+1,i)} \right) x^i \tag{7}$$

and the adaptive learning rate $\alpha_k^{(t+1)}$ as

$$\alpha_k^{(t+1)} = \frac{\delta_k^{(t+1)} \delta_k^{(t+1)}}{\delta_k^{(t+1)} X^{\mathrm{T}} X \delta_k^{(t+1)}} \tag{8}$$

2.1 The Adaptive Steepest Descent (ASD) Method

The connection update after the presentation of all the training set patterns x^i, $i = 1, 2, \ldots, m$ at training cycle $t+1$, could have the form

$$\begin{aligned} w_k^{(t+1,j)} &= w_k^{(t,j)} + \alpha_k^{(t+1)} \sum_{i=1}^{m} (d_j^i - y_j^{(t+1,i)}) x_k^i \\ &= w_k^{(t,j)} + \frac{\delta^{(t+1)} \delta^{(t+1)}}{\delta^{(t+1)} X^{\mathrm{T}} X \delta^{(t+1)}} \sum_{i=1}^{m} \left(d_j^i - \sum_{q=1}^{n} w_q^{(t,j)} x_q^i \right) x_k^i \\ &= w_k^{(t,j)} + \frac{\delta^{(t+1)} \delta^{(t+1)}}{\delta^{(t+1)} C \delta^{(t+1)}} \sum_{i=1}^{m} \left(c_k^i - \sum_{q=1}^{n} w_q^{(t,j)} b_q^i \right) \end{aligned} \tag{9}$$

with $k = 1, 2, \ldots, n$ and b_q^k, c_k^i the corresponding elements of B and C, as defined in (2).

The operation of the ANN in Figure 1 using ASD method with the adaptive learning rate $\alpha_k^{(t+1)}$ defined in (2) simulates the Adaptive Steepest Descent Method. Proof can be found in [2].

2.2 The Optimal Fletcher-Reeves (OFR) Method

We define $\beta^{(t+1)}$ to be

$$\beta^{(t+1)} = \frac{\delta^{(t+1)}\delta^{(t+1)}}{\delta^{(t)}\delta^{(t)}} \tag{10}$$

and the connection update after the presentation of all the training set patterns $x^i, i = 1, 2, \ldots, m$ at training cycle $t + 1$, has the form

$$\begin{aligned} w_k^{(t+1,j)} &= w_k^{(t,j)} + \alpha_k^{(t+1)} \sum_{i=1}^{m} \left(d_j^i - y_j^{(t+1,i)} \right) x_k^i + \beta^{(t+1)} \Delta w_k^{(t,j)} \\ &= w_k^{(t,j)} - \frac{\delta^{(t+1)}\delta^{(t+1)}}{\delta^{(t)} C \delta^{(t)}} \sum_{i=1}^{m} \left(d_j^i - \sum_{q=1}^{n} w_q^{(t,j)} x_q^i \right) x^i + \frac{\delta^{(t+1)}\delta^{(t+1)}}{\delta^{(t)}\delta^{(t)}} \Delta w_k^{(t,j)} \\ &= w_k^{(t,j)} + \frac{\delta^{(t+1)}\delta^{(t+1)}}{\delta^{(t+1)} C \delta^{(t+1)}} \sum_{i=1}^{m} \left(c_k^i - \sum_{q=1}^{n} w_q^{(t,j)} b_q^i \right) + \frac{\delta^{(t+1)}\delta^{(t+1)}}{\delta^{(t)}\delta^{(t)}} \Delta w_k^{(t,j)} \end{aligned} \tag{11}$$

with $k = 1, 2, \ldots, n$ and b_q^k, c_k^i the corresponding elements of B and C, as defined in (2). The operation of the ANN in Figure 1 using OFR method with the adaptive learning rate $\alpha_k^{(t+1)}$ defined in (2) simulates the Optimal Fletcher-Reeves Method.

3 EXPERIMENTAL STUDY

In order to check the performance and the convergence behavior of the proposed algorithm the benchmark problems used were taken from the Proben1 benchmark set, with the standard Proben1 benchmarking rules. The error measures reported include: the training set error (mean and standard deviation of minimum squared error percentage on training set, reached at any time during training), validation set error (mean and standard deviation of minimum squared error percentage on validation set, reached at any time during training), test set error (mean and standard deviation of minimum squared error percentage on test set, at point of minimum validation set error), and the test set classification error (*i.e.* the percentage of incorrectly classified examples). For the classification problems, winner-takes-all method was used to determine the classification, *i.e.* the output with the highest activation designates the class, while in approximation problems, a threshold of 0.3 was used in the output, and the network accepts an output as 0, if it is below 0.3, and as 1, if it is above 0.7. In order to measure the training time, we also report the number of epochs used, the number of relative epochs, *i.e.* the epochs needed to reach the minimum validation error, and the connection traversals. One stopping criterion is the loss of generality. The generalization loss at epoch t is defined as the relative increase of the validation squared error percentage over the minimum so far in percent

$$GL(t) = 100 \cdot \left(\frac{E_{va}(t)}{\min_{t' \le t} E_{va}(t')} - 1 \right) \tag{12}$$

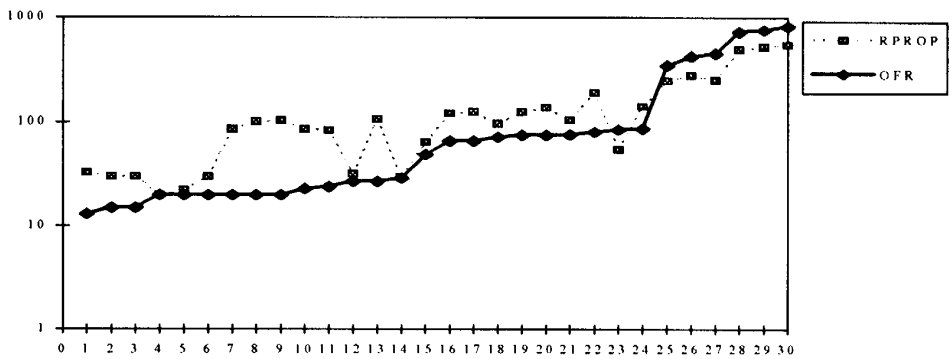
and the algorithm stops as the generalization loss exceeds a threshold $a = 5$. Another stopping criterion is the training progress, defined in terms of a training strip. A training strip of length k is a sequence of k epochs, and the training progress is how much is the average training error during the strip larger than the minimum error during the strip

$$P_k(t) = 1000 \cdot \left(\frac{\sum_{t' \in k-k+1 \cdots t} E_{tr}(t')}{k \cdot \min_{t' \in k-k+1 \cdots t} E_{tr}(t')} - 1 \right) \tag{13}$$

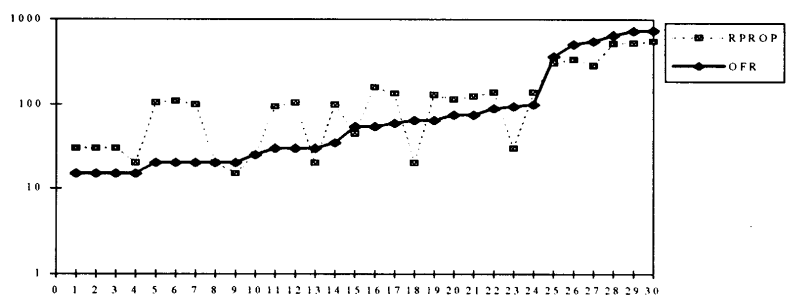
Since training involves some kind of random generalization, in order to make reliable statements about the performance of the three algorithms, we used 20 runs on each problem, for each one of the three datasets. The data reported include the mean and standard deviation of the above error and training measures for these 20 runs, along with generalization loss at end of training.

The results of ASD and OFR training of the linear network in Figure 1 with the classification and approximation problems, are compared to linear-RPROP, with the following parameters used by Prechelt [8]: $n^+ = 1.2, n^- = 0.5$, $\Delta_0 \in 0.005 \cdots 0.02$, $\Delta_{\max} = 50$, $\Delta_{\min} = 0$, and initial weights randomly chosen in $[-0.01, 0.01]$.

The first criterion checked is speed of convergence, *i.e.* the number of epochs required by each algorithm to reach one of the three stopping criteria. In most problems, the best results are achieved by OFR method, which outperforms the other two algorithms in 23 out of 30 datasets for the classification problems, and in 10 out of 12 datasets for the approximation problems. Graphs 3.1 and 3.2 show the mean value of the epochs required by each algorithm



GRAPH 3.1 Mean value of iterations needed for convergence of classification problems (30 datasets).



GRAPH 3.2 Best run iterations needed for convergence of classification problems (30 datasets).

TABLE III.1 Results for Classification Problems.

		Epochs		*Relevant epochs*		*Connection traversals*		*Training set error*		*Validation set error*		*Test set error*		*Test set classification*		*Generality loss*	
Problem	*Method*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*
Cancer1	RPROP	85	14	73	18	85	14	20.75	0.01	20.21	0.12	18.61	0.13	15.15	0.50	0.18	0.28
	ASD	40	0	23	1	40	0	20.74	0.00	20.20	0.00	18.63	0.00	16.09	0.00	0.16	0.01
	OFR	20	1	9	0	20	1	20.74	0.00	20.07	0.04	18.61	0.05	15.79	0.39	0.83	0.28
Cancer2	RPROP	84	23	63	19	84	23	18.71	0.18	20.56	0.10	22.03	0.19	19.03	0.45	1.12	2.68
	ASD	50	0	30	1	50	0	18.65	0.00	20.51	0.00	22.02	0.01	18.39	0.00	0.13	0.01
	OFR	24	4	11	0	24	4	18.64	0.00	20.47	0.02	22.14	0.03	18.94	0.13	0.37	0.08
Cancer3	RPROP	85	14	73	18	85	14	20.75	0.01	20.21	0.12	18.61	0.13	15.15	0.50	0.18	0.28
	ASD	48	2	21	0	48	2	20.43	0.00	18.70	0.00	20.34	0.01	17.66	0.25	0.46	0.03
	OFR	23	5	13	5	23	5	20.43	0.01	18.75	0.03	20.28	0.07	16.88	0.28	0.33	0.16
Card1	RPROP	64	9	27	5	64	9	9.83	0.01	8.93	0.14	10.67	0.20	13.71	0.74	4.19	1.23
	ASD	88	3	13	0	88	3	10.04	0.01	8.21	0.01	10.47	0.01	13.95	0.00	5.08	0.06
	OFR	49	13	13	0	49	13	10.05	0.072	8.20	0.01	10.46	0.01	13.95	0.00	5.90	0.86
Card2	RPROP	55	19	24	6	55	19	8.38	0.35	10.76	0.24	14.93	0.26	19.37	0.38	4.38	1.49
	ASD	231	4	21	1	231	4	8.32	0.00	9.71	0.01	13.67	0.03	19.77	0.00	5.02	0.04
	OFR	85	30	19	0	85	30	8.34	0.03	9.71	0.01	13.72	0.02	19.77	0.00	5.59	0.77
Card3	RPROP	104	11	45	13	104	11	9.47	0.00	8.39	0.10	12.60	0.21	14.60	0.65	1.75	1.19
	ASD	158	4	26	1	158	4	9.64	0.00	7.68	0.01	12.34	0.01	15.45	0.29	5.05	0.04
	OFR	76	35	20	2	76	35	9.65	0.04	7.67	0.01	12.34	0.03	15.51	0.42	5.69	0.81
Diabetes1	RPROP	103	15	85	19	103	15	20.34	0.01	22.58	0.03	24.06	0.09	38.65	0.60	0.19	0.16
	ASD	30	0	10	0	30	0	20.32	0.00	22.54	0.00	24.02	0.00	38.54	0.00	0.54	0.01
	OFR	20	1	8	0	20	1	20.31	0.00	22.51	0.00	23.89	0.01	37.50	0.00	0.88	0.04
Diabetes2	RPROP	100	16	99	16	100	16	21.05	0.01	20.87	0.06	24.29	0.08	37.31	0.78	0.02	0.03
	ASD	30	0	15	0	30	0	21.03	0.00	20.71	0.00	23.98	0.00	35.94	0.00	0.16	0.01
	OFR	20	0	11	1	20	0	21.02	0.00	20.72	0.00	24.16	0.09	36.38	0.72	0.22	0.02
Diabetes3	RPROP	106	11	35	8	106	11	20.40	0.01	23.03	0.14	22.64	0.17	37.06	1.24	2.55	0.57
	ASD	25	0	3	0	25	0	20.38	0.00	22.87	0.01	22.04	0.01	33.74	0.47	3.70	0.03
	OFR	27	8	3	0	27	8	20.39	0.01	22.89	0.01	22.03	0.01	33.28	0.53	3.65	0.22
Gene1	RPROP	30	0	18	6	30	0	21.48	0.00	25.55	0.05	25.03	0.04	39.06	0.60	0.24	0.20
	ASD	19	2	2	0	19	2	21.48	0.00	24.73	0.10	24.46	0.06	40.70	0.68	3.52	0.43
	OFR	20	2	2	0	20	2	21.48	0.00	24.74	0.08	24.53	0.07	40.66	0.66	3.46	0.34

TABLE III.1 *(continued)*

		Epochs		*Relevant epochs*		*Connection traversals*		*Training set error*		*Validation set error*		*Test set error*		*Test set classification*		*Generality loss*	
Problem	*Method*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*
Gene2	RPROP	30	0	19	5	30	0	21.62	0.00	25.20	0.03	24.99	0.04	39.74	0.50	0.15	0.09
	ASD	15	0	2	0	15	0	21.62	0.00	24.54	0.10	24.54	0.09	41.14	0.58	2.75	0.39
	OFR	15	0	2	0	15	0	21.62	0.00	24.54	0.11	24.57	0.11	41.05	0.77	2.75	0.45
Gene3	RPROP	30	0	18	5	30	0	21.88	0.00	24.33	0.05	25.37	0.07	41.93	0.64	0.21	0.20
	ASD	15	0	3	0	15	0	21.88	0.00	24.01	0.08	24.88	0.08	42.37	0.41	1.45	0.32
	OFR	15	0	3	0	15	0	21.88	0.00	24.00	0.06	24.86	0.06	42.50	0.34	1.50	0.26
Glass1	RPROP	124	14	24	5	124	14	8.84	0.01	9.71	0.07	10.13	0.12	47.67	3.05	3.75	0.69
	ASD	157	3	44	1	157	3	8.80	0.00	9.95	0.01	9.77	0.01	47.17	0.00	1.65	0.06
	OFR	75	14	29	8	75	14	8.79	0.02	9.86	0.08	9.80	0.11	43.30	2.07	3.68	1.95
Glass2	RPROP	33	9	16	3	33	9	8.75	0.16	10.27	0.16	10.30	0.14	55.91	2.39	6.19	1.09
	ASD	25	0	7	0	25	0	8.62	0.01	10.54	0.01	10.37	0.01	54.72	0.00	5.40	0.14
	OFR	13	3	5	0	13	3	8.70	0.11	10.59	0.02	10.48	0.01	52.83	0.00	6.16	0.97
Glass3	RPROP	124	22	26	12	124	22	8.72	0.02	9.36	0.05	11.18	0.20	60.77	4.13	2.03	0.55
	ASD	182	3	46	2	182	3	8.67	0.00	9.41	0.00	10.94	0.01	56.60	0.00	1.36	0.04
	OFR	66	22	28	7	66	22	8.68	0.06	9.41	0.03	11.01	0.07	56.31	1.26	4.07	4.40
Heart1	RPROP	138	15	49	9	138	15	11.19	0.01	13.25	0.06	14.31	0.05	21.08	0.41	1.22	0.55
	ASD	161	2	160	2	161	2	11.21	0.00	13.13	0.00	14.13	0.00	20.43	0.00	0.01	0.01
	OFR	75	17	69	15	75	17	11.19	0.02	13.11	0.02	14.08	0.02	20.48	0.13	0.06	0.06
Heart2	RPROP	189	24	168	26	189	24	11.67	0.02	12.21	0.02	13.53	0.03	16.50	0.22	0.15	0.11
	ASD	216	3	215	3	216	3	11.66	0.00	12.22	0.00	13.60	0.00	16.52	0.00	0.00	0.00
	OFR	81	23	72	23	81	23	11.64	0.02	12.22	0.01	13.63	0.04	16.54	0.10	0.07	0.08
Heart3	RPROP	141	13	85	48	141	13	11.11	0.00	10.73	0.06	16.40	0.11	23.18	0.90	0.42	0.48
	ASD	187	2	137	2	187	2	11.11	0.00	10.61	0.00	16.27	0.00	22.61	0.00	0.09	0.02
	OFR	87	21	57	10	87	21	11.09	0.01	10.56	0.02	16.25	0.04	22.68	0.32	0.58	0.32
Heartc1	RPROP	120	27	82	40	120	27	10.20	0.15	9.62	0.12	16.47	0.64	20.00	0.97	0.82	1.40
	ASD	155	0	155	0	155	0	10.18	0.00	9.61	0.00	16.00	0.00	20.00	0.00	0.00	0.00
	OFR	66	13	44	18	66	13	10.16	0.01	9.57	0.03	16.04	0.07	19.44	0.66	0.40	0.25
Heartc2	RPROP	96	61	26	16	96	61	11.78	0.93	16.58	0.43	6.66	0.98	3.79	1.90	5.38	2.25
	ASD	170	0	28	1	170	0	11.23	0.00	16.54	0.01	6.15	0.01	3.23	0.66	2.87	0.04
	OFR	72	19	22	2	72	19	11.22	0.01	16.53	0.04	6.22	0.10	2.46	0.49	3.41	0.41
Heartc3	RPROP	20	6	13	2	20	6	11.12	0.62	13.99	0.35	13.21	0.41	13.75	1.44	8.31	3.33

	ASD	24	2	5	0	24	2	10.44	0.06	13.12	0.03	12.08	0.02	15.44	0.66	6.06	0.53
	OFR	20	0	5	0	20	0	10.33	0.02	13.11	0.03	12.09	0.03	15.72	0.54	8.75	0.50
Horse1	RPROP	30	6	13	4	30	6	11.30	0.20	15.55	0.24	12.87	0.33	26.20	2.60	6.25	0.64
	ASD	32	2	5	0	32	2	11.25	0.05	15.21	0.04	12.76	0.05	26.49	1.23	5.59	0.36
	OFR	29	2	5	0	29	2	11.10	0.07	15.20	0.05	12.75	0.04	26.95	1.09	6.84	0.82
Horse2	RPROP	32	7	14	3	32	7	8.90	0.21	15.73	0.27	17.34	0.50	36.50	1.31	5.60	0.96
	ASD	60	0	15	1	60	0	8.36	0.01	15.68	0.02	16.62	0.06	35.80	0.54	5.22	0.13
	OFR	27	3	13	1	27	3	8.44	0.06	15.69	0.03	16.66	0.08	35.51	0.51	6.82	1.31
Horse3	RPROP	22	7	9	2	22	7	10.73	0.42	15.43	0.28	15.24	0.34	31.87	1.82	5.89	0.90
	ASD	25	1	7	1	25	1	10.26	0.03	15.33	0.04	15.13	0.05	33.20	0.67	5.30	0.29
	OFR	20	0	7	1	20	0	10.24	10.03	15.31	0.04	15.13	0.05	33.49	0.75	6.06	0.35
Soybean1	RPROP	543	15	420	51	543	15	0.65	0.00	0.98	0.00	1.16	0.00	9.57	0.32	0.26	0.11
	ASD	1198	20	1198	20	1198	20	0.67	0.00	0.96	0.00	1.15	0.00	9.35	0.18	0.00	0.01
	OFR	821	74	802	81	821	74	0.67	0.00	0.96	0.00	1.15	0.01	9.20	0.28	0.07	0.08
Soybean2	RPROP	491	18	474	28	491	18	0.80	0.00	0.81	0.00	1.05	0.00	4.12	0.00	0.05	0.04
	ASD	1013	15	1012	15	1013	15	0.82	0.00	0.82	0.00	1.07	0.00	4.12	0.00	0.00	0.01
	OFR	728	54	718	54	728	54	0.82	0.00	0.82	0.01	1.07	0.01	4.12	0.00	0.06	0.09
Soybean3	RPROP	518	20	498	34	518	20	0.78	0.00	0.96	0.00	1.04	0.00	7.09	0.13	0.04	0.03
	ASD	1128	25	1128	25	1128	25	0.80	0.00	0.96	0.00	1.02	0.00	6.53	0.18	0.00	0.01
	OFR	759	60	752	60	759	60	0.79	0.00	0.96	0.01	1.03	0.01	6.59	0.31	0.09	0.13
Thyroid1	RPROP	246	50	243	51	246	50	3.91	0.01	3.95	0.02	4.08	0.02	6.53	0.03	0.03	0.03
	ASD	810	11	810	11	810	11	4.03	0.00	4.17	0.00	4.22	0.00	6.56	0.00	0.01	0.01
	OFR	346	121	344	120	346	121	3.97	0.04	4.05	0.09	4.13	0.08	6.55	0.02	0.01	0.01
Thyroid2	RPROP	251	40	248	39	251	40	4.10	0.01	3.68	0.02	3.86	0.02	6.38	0.00	0.05	0.05
	ASD	914	10	913	10	914	10	4.23	0.00	3.81	0.00	3.99	0.00	6.38	0.00	0.01	0.01
	OFR	453	110	452	110	453	110	4.17	0.04	3.77	0.04	3.95	0.04	6.38	0.00	0.01	0.01
Thyroid3	RPROP	280	65	275	67	280	65	4.01	0.01	3.48	0.01	4.19	0.01	7.23	0.02	0.05	0.03
	ASD	904	7	903	7	904	7	4.16	0.00	3.65	0.00	4.33	0.00	7.17	0.00	0.02	0.01
	OFR	424	126	422	125	424	126	4.08	0.06	3.56	0.06	4.25	0.05	7.22	0.06	0.02	0.02

TABLE III.2 Results for Approximation Problems.

		Epochs		*Relevant epochs*		*Connection traversals*		*Training set error*		*Validation set error*		*Test set error*		*Test set classification*		*Generality loss*	
Problem	*Method*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*	*mean*	*stdev*
Building1	RPROP	348	35	344	35	348	35	0.34	0.00	0.37	0.00	0.35	0.00	0.29	0.00	0.03	0.03
	ASD	544	24	544	24	544	24	0.34	0.00	0.37	0.00	0.35	0.00	0.29	0.00	0.00	0.00
	OFR	343	130	342	129	343	130	0.33	0.00	0.37	0.00	0.34	0.00	0.29	0.00	0.00	0.01
Building2	RPROP	340	26	335	27	340	26	0.34	0.00	0.37	0.00	0.35	0.00	0.29	0.00	0.04	0.02
	ASD	525	31	525	31	525	31	0.34	0.00	0.37	0.00	0.35	0.00	0.29	0.00	0.00	0.00
	OFR	367	118	366	118	367	118	0.33	0.00	0.37	0.00	0.34	0.00	0.29	0.00	0.00	0.00
Building3	RPROP	351	23	341	22	351	23	0.35	0.00	0.35	0.00	0.34	0.00	0.29	0.00	0.02	0.01
	ASD	462	44	462	44	462	44	0.35	0.00	0.35	0.00	0.35	0.00	0.29	0.00	0.01	0.01
	OFR	304	59	298	59	304	59	0.35	0.00	0.35	0.00	0.35	0.00	0.25	0.05	0.04	0.07
Flare1	RPROP	24	20	8	6	24	20	0.39	0.02	0.34	0.01	0.55	0.03	3.84	0.45	6.94	4.82
	ASD	5	0	1	0	5	0	0.51	0.03	0.43	0.03	0.70	0.04	5.26	0.00	0.00	0.00
	OFR	5	0	1	0	5	0	0.52	0.03	0.43	0.02	0.70	0.03	5.26	0.00	0.00	0.00
Flare2	RPROP	19	20	7	6	19	20	0.46	0.03	0.48	0.02	0.31	0.01	2.56	0.24	6.29	4.67
	ASD	5	0	1	0	5	0	0.61	0.03	0.59	0.03	0.33	0.02	3.01	0.00	0.00	0.00
	OFR	5	0	1	0	5	0	0.61	0.03	0.59	0.03	0.33	0.02	3.01	0.00	0.00	0.00
Flare3	RPROP	34	19	11	7	34	19	0.41	0.03	0.47	0.02	0.36	0.02	3.01	0.12	4.63	6.01
	ASD	5	0	1	0	5	0	0.58	0.02	0.57	0.02	0.43	0.02	3.76	0.00	0.00	0.00

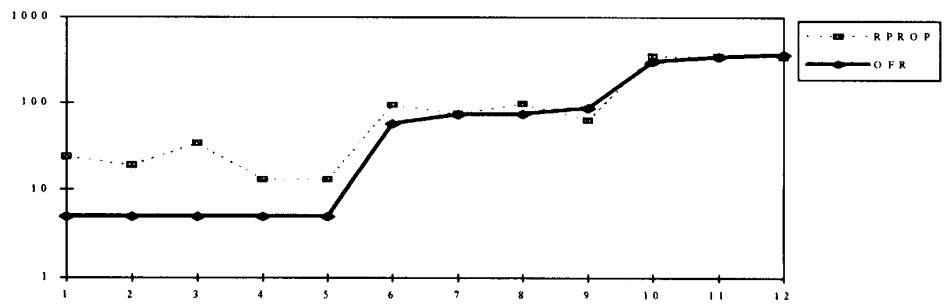
	OFR	5	0	1	0	5	0	0.58	0.03	0.58	0.03	0.44	0.03	3.76	0.00	0.00	0.00
Hearta1	RPROP	63	57	17	13	63	57	4.85	1.33	5.07	1.06	5.25	1.18	14.07	4.72	8.47	6.09
	ASD	225	4	8	1	225	4	3.84	0.00	4.36	0.01	4.65	0.02	10.92	0.44	2.59	0.26
	OFR	87	23	12	10	87	23	3.84	0.01	4.35	0.01	4.62	0.08	10.85	0.52	3.17	0.65
Hearta2	RPROP	98	12	90	11	98	12	4.17	0.01	4.27	0.03	4.19	0.01	10.69	0.57	0.10	0.12
	ASD	176	3	176	3	176	3	4.18	0.00	4.25	0.00	4.19	0.00	10.87	0.00	0.01	0.01
	OFR	75	19	73	19	75	19	4.17	0.01	4.25	0.01	4.18	0.01	10.59	0.21	0.01	0.02
Hearta3	RPROP	95	31	81	34	95	31	4.09	0.08	4.15	0.06	4.59	0.12	12.22	1.10	0.92	2.21
	ASD	143	3	143	3	143	3	4.08	0.00	4.12	0.00	4.58	0.00	12.11	0.16	0.02	0.02
	OFR	58	7	53	9	58	7	4.06	0.01	4.10	0.02	4.56	0.02	11.69	0.31	0.07	0.11
Heartac1	RPROP	75	41	69	41	75	41	4.10	0.09	4.71	0.04	2.73	0.14	5.05	0.82	2.74	4.35
	ASD	129	3	128	2	129	3	4.05	0.00	4.67	0.00	2.65	0.01	5.33	0.00	0.03	0.02
	OFR	74	24	71	25	74	24	4.05	0.01	4.66	0.02	2.68	0.03	4.91	0.62	0.04	0.05
Heartac2	RPROP	13	6	9	4	13	6	4.23	1.18	5.44	0.93	4.81	1.10	11.23	3.25	13.97	10.63
	ASD	5	0	2	0	5	0	3.89	0.02	4.67	0.04	4.07	0.04	9.47	0.41	15.25	0.59
	OFR	5	0	2	0	5	0	3.88	0.02	4.67	0.02	4.07	0.03	9.47	0.41	15.10	0.34
Heartac3	RPROP	13	10	8	5	13	10	4.04	1.28	5.89	0.79	5.99	0.82	16.42	3.77	12.44	6.71
	ASD	5	0	2	0	5	0	3.24	0.01	5.10	0.04	5.45	0.02	14.95	0.69	6.62	0.21
	OFR	5	0	2	0	5	0	3.24	0.01	5.08	0.04	5.45	0.03	15.09	0.97	6.81	0.25

for every dataset. OFR also requires the lowest number of relevant epochs, and connection traversals. These results, are shown in Tables III.1 and III.2.

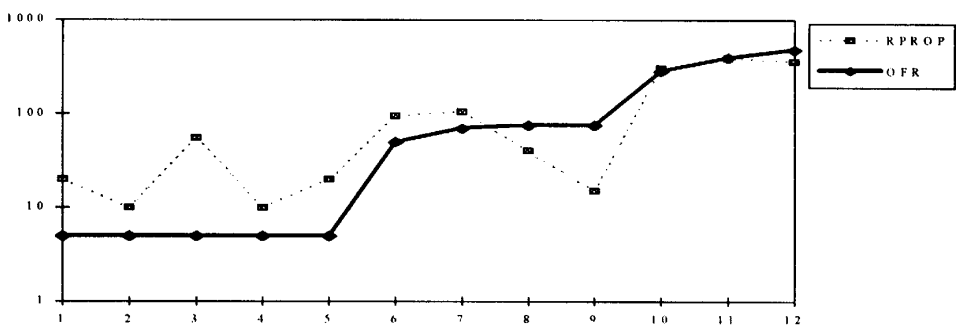
We also report some results of the above error and training measures of the 20 runs for each of the three datasets of each problem, as it concerns the "best run", *i.e.* the one with the lowest validation set error. These results of linear-RPROP, ASD and OFR training also show that OFR outperforms all other algorithms in 19 out of 30 datasets in classification problems, and in 9 out of 12 datasets in approximation problems, while OFR needs the lowest number of relevant epochs and connection traversals. Results, are shown in Graphs 3.3 and 3.4 and Tables III.3 and III.4.

The convergence behavior of the three algorithms, with respect to the test set classification error during time, applied on the "best run", is shown in Graphs 3.5–3.18. It can be seen that in most of the problems, the test set classification error for ASD and OFR methods decrease rapidly, and reaches its minimum value.

The results shown on Tables III.1–III.4, and Graphs 3.1–3.18, show that real world problems can be solved with linear neural networks, and training time for convergence can be accelerated by using adaptive steepest descent methods, like ASD and OFR. Some of the problems are very sensitive to overfitting, which suggests that using early stopping was very useful.



GRAPH 3.3 Mean value of iterations needed for convergence of approximation problems (12 datasets).



GRAPH 3.4 Best run iterations needed for convergence of approximation problems (12 datasets).

TABLE III.3 Best Run Results for Classification Problems.

Problem	*Method*	*Epochs*	*Relevant epochs*	*Connection traversals*	*Training set error*	*Validation set error*	*Test set error*	*Test set classification*	*Generality loss*
Cancer1	RPROP	105	79	105	20.75	20.03	18.45	14.37	0.38
	ASD	40	23	40	20.74	20.19	18.63	16.09	0.17
	OFR	20	9	20	20.75	20.01	18.54	16.67	1.30
Cancer2	RPROP	95	43	95	18.66	20.38	21.95	17.82	0.72
	ASD	50	31	50	18.65	20.51	22.02	18.39	0.13
	OFR	30	10	30	18.66	20.44	22.19	18.97	0.32
Cancer3	RPROP	100	49	100	20.44	18.48	20.39	16.09	1.23
	ASD	50	21	50	20.42	18.70	20.34	17.82	0.50
	OFR	35	11	35	20.42	18.71	20.29	17.24	0.54
Card1	RPROP	45	28	45	9.86	8.64	10.67	13.95	6.24
	ASD	80	13	80	10.06	8.19	10.46	13.95	5.03
	OFR	55	13	55	10.05	8.19	10.44	13.95	5.21
Card2	RPROP	20	15	20	8.48	10.07	14.85	19.77	6.56
	ASD	225	21	225	8.32	9.70	13.65	19.77	5.01
	OFR	65	19	65	8.35	9.69	13.70	19.77	5.10
Card3	RPROP	115	23	115	9.48	8.16	12.23	13.95	4.45
	ASD	155	27	155	9.64	7.67	12.35	15.70	5.00
	OFR	75	24	75	9.68	7.68	12.27	16.28	5.06
Diabetes1	RPROP	110	87	110	20.33	22.51	23.96	38.54	0.27
	ASD	30	10	30	20.32	22.54	24.01	38.54	0.55
	OFR	20	8	20	20.31	22.50	23.87	37.50	0.91
Diabetes2	RPROP	100	97	100	21.03	20.75	24.34	38.02	0.03
	ASD	30	15	30	21.03	20.71	23.99	35.94	0.17
	OFR	20	10	20	21.02	20.71	24.20	35.94	0.24
Diabetes3	RPROP	105	33	105	20.40	22.83	22.50	37.50	3.22
	ASD	25	3	25	20.38	22.86	22.05	34.38	3.73
	OFR	30	3	30	20.39	22.88	22.03	33.33	3.70
Gene1	RPROP	30	13	30	21.48	25.45	25.01	39.34	0.65
	ASD	20	2	20	21.48	24.52	24.36	41.74	4.38
	OFR	15	2	15	21.48	24.63	24.48	40.48	3.82
Gene2	RPROP	30	16	30	21.62	25.15	24.91	38.59	0.32
	ASD	15	2	15	21.62	24.34	24.50	40.98	3.58
	OFR	15	2	15	21.62	24.38	24.49	42.12	3.38

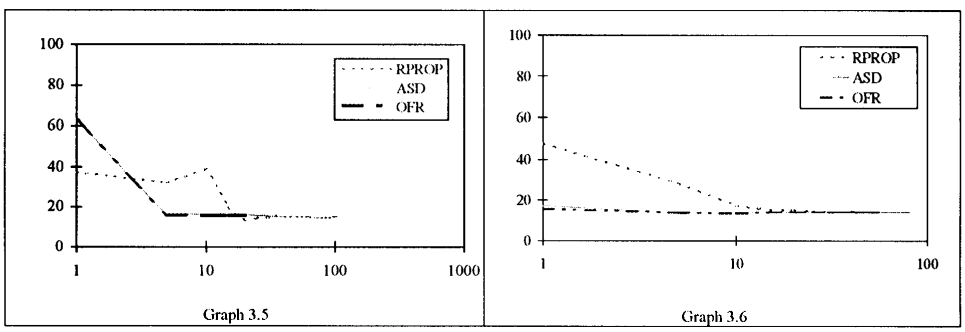
TABLE III.3 *(continued)*

Problem	*Method*	*Epochs*	*Relevant epochs*	*Connection traversals*	*Training set error*	*Validation set error*	*Test set error*	*Test set classification*	*Generality loss*
Gene3	RPROP	30	9	30	21.88	24.20	25.28	40.23	0.77
	ASD	15	3	15	21.88	23.91	24.85	42.62	1.89
	OFR	15	3	15	21.88	23.90	24.75	42.62	1.96
Glass1	RPROP	140	18	140	8.84	9.59	10.32	49.06	4.87
	ASD	160	45	160	8.80	9.94	9.75	47.17	1.79
	OFR	90	34	90	8.78	9.73	9.63	43.40	3.88
Glass2	RPROP	20	14	20	8.95	9.93	10.33	54.72	5.45
	ASD	25	7	25	8.62	10.51	10.36	54.72	5.61
	OFR	15	5	15	8.58	10.52	10.46	52.83	8.71
Glass3	RPROP	140	18	140	8.71	9.24	11.02	60.38	3.33
	ASD	185	43	185	8.67	9.40	10.95	56.60	1.44
	OFR	100	28	100	8.61	9.36	10.95	56.60	3.38
Heart1	RPROP	135	39	135	11.18	13.15	14.36	22.17	1.82
	ASD	160	159	160	11.21	13.12	14.13	20.43	0.02
	OFR	60	52	60	11.18	13.07	14.06	20.43	0.11
Heart2	RPROP	160	137	160	11.67	12.18	13.51	16.52	0.17
	ASD	220	220	220	11.66	12.22	13.60	16.52	0.00
	OFR	55	44	55	11.64	12.20	13.59	16.52	0.33
Heart3	RPROP	130	34	130	11.10	10.59	16.56	23.48	1.42
	ASD	190	135	190	11.11	10.61	16.27	22.61	0.12
	OFR	65	52	65	11.09	10.53	16.21	22.17	0.70
Heartc1	RPROP	125	17	125	10.16	9.44	17.45	22.67	2.19
	ASD	155	155	155	10.18	9.60	16.00	20.00	0.00
	OFR	75	24	75	10.14	9.51	16.04	18.67	0.74
Heartc2	RPROP	30	15	30	11.87	15.80	6.08	2.67	5.31
	ASD	170	27	170	11.23	16.52	6.14	4.00	2.99
	OFR	95	26	95	11.23	16.39	6.27	1.33	4.01
Heartc3	RPROP	20	11	20	10.66	13.24	12.64	16.00	12.72
	ASD	20	5	20	10.55	13.06	12.08	16.00	5.19
	OFR	20	5	20	10.32	13.05	12.06	16.00	9.17
Horse1	RPROP	25	14	25	11.43	15.18	12.91	27.47	5.60
	ASD	30	5	30	11.29	15.13	12.70	26.37	5.45
	OFR	25	5	25	11.24	15.10	12.73	26.37	5.45

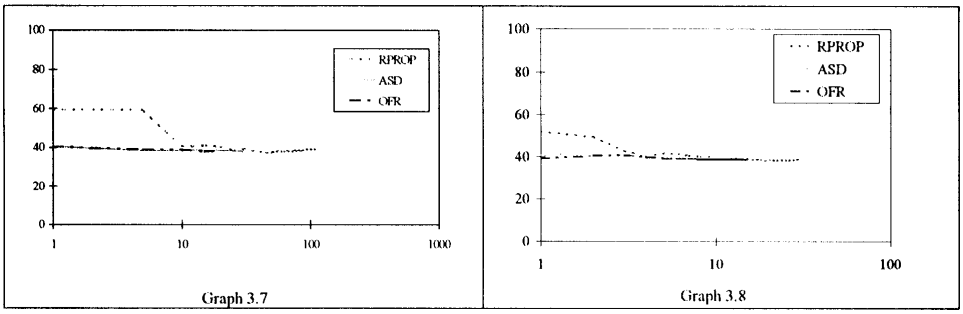
Horse2	RPROP	20	12	20	9.36	15.18	17.48	35.16	9.65
	ASD	60	15	60	8.38	15.63	16.61	35.16	5.14
	OFR	30	13	30	8.41	15.65	16.68	35.16	6.45
Horse3	RPROP	15	9	15	11.01	14.71	15.33	30.77	9.08
	ASD	25	5	25	10.24	15.22	15.00	34.07	5.88
	OFR	20	7	20	10.23	15.25	15.12	32.97	6.37
Soybean1	RPROP	525	486	525	0.65	0.98	1.16	8.82	0.22
	ASD	1170	1170	1170	0.67	0.96	1.15	9.41	0.00
	OFR	735	728	735	0.67	0.96	1.14	9.41	0.01
Soybean2	RPROP	520	484	520	0.80	0.81	1.05	4.12	0.11
	ASD	1035	1035	1035	0.82	0.82	1.07	4.12	0.00
	OFR	655	630	655	0.82	0.81	1.05	4.12	0.02
Soybean3	RPROP	555	548	555	0.78	0.96	1.03	7.06	0.04
	ASD	1100	1100	1100	0.80	0.96	1.02	6.47	0.00
	OFR	745	739	745	0.79	0.94	1.02	6.47	0.00
Thyroid1	RPROP	290	287	290	3.91	3.96	4.08	6.56	0.01
	ASD	815	814	815	4.03	4.17	4.22	6.56	0.02
	OFR	555	548	555	3.88	3.87	3.99	6.56	0.03
Thyroid2	RPROP	310	297	310	4.08	3.65	3.82	6.38	0.14
	ASD	930	929	930	4.22	3.81	3.99	6.38	0.02
	OFR	370	370	370	4.10	3.67	3.83	6.38	0.00
Thyroid3	RPROP	340	337	340	4.02	3.48	4.19	7.22	0.05
	ASD	915	915	915	4.16	3.64	4.32	7.17	0.00
	OFR	510	503	510	3.98	3.46	4.16	7.28	0.07

TABLE III.4 Best Run Results for Approximation Problems.

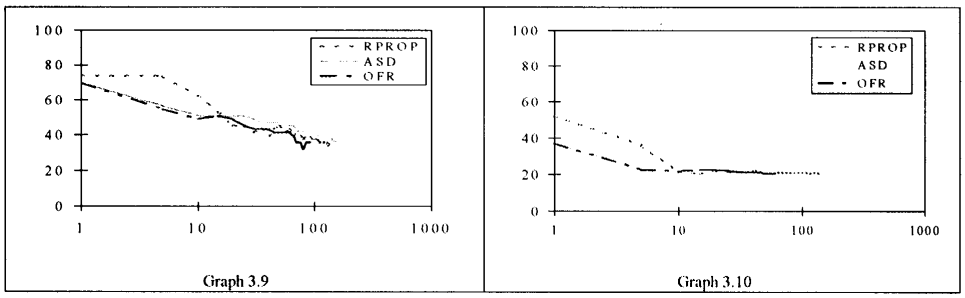
Problem	*Method*	*Epochs*	*Relevant epochs*	*Connection traversals*	*Training set error*	*Validation set error*	*Test set error*	*Test set classification*	*Generality loss*
Building1	RPROP	390	390	390	0.33	0.37	0.35	0.29	0.00
	ASD	470	470	470	0.34	0.37	0.35	0.29	0.00
	OFR	395	394	395	0.33	0.37	0.34	0.29	0.00
Building2	RPROP	360	357	360	0.33	0.37	0.35	0.29	0.06
	ASD	475	475	475	0.34	0.37	0.35	0.29	0.00
	OFR	480	475	480	0.33	0.37	0.34	0.29	0.00
Building3	RPROP	305	295	305	0.35	0.34	0.34	0.29	0.03
	ASD	525	525	525	0.35	0.34	0.34	0.29	0.00
	OFR	285	278	285	0.35	0.34	0.34	0.29	0.03
Flare1	RPROP	20	6	20	0.37	0.32	0.54	3.38	5.38
	ASD	5	1	5	0.47	0.39	0.65	5.26	0.00
	OFR	5	1	5	0.46	0.40	0.64	5.26	0.00
Flare2	RPROP	10	8	10	0.44	0.45	0.31	2.26	5.34
	ASD	5	1	5	0.57	0.54	0.30	3.01	0.00
	OFR	5	1	5	0.57	0.54	0.29	3.01	0.00
Flare3	RPROP	55	8	55	0.39	0.45	0.36	3.38	3.09
	ASD	5	1	5	0.54	0.55	0.40	3.76	0.00
	OFR	5	1	5	0.53	0.54	0.40	3.76	0.00
Hearta1	RPROP	40	10	40	3.94	4.26	4.60	12.17	8.67
	ASD	220	6	220	3.84	4.34	4.69	11.30	2.96
	OFR	75	6	75	3.82	4.34	4.68	11.30	3.67
Hearta2	RPROP	105	86	105	4.16	4.20	4.19	10.87	0.34
	ASD	175	175	175	4.18	4.25	4.18	10.87	0.00
	OFR	70	64	70	4.16	4.23	4.18	10.43	0.06
Hearta3	RPROP	95	57	95	4.06	4.07	4.56	11.30	1.06
	ASD	140	139	140	4.08	4.11	4.58	12.17	0.04
	OFR	50	48	50	4.06	4.06	4.52	11.30	0.02
Heartac1	RPROP	15	9	15	4.19	4.63	3.14	6.67	7.85
	ASD	125	125	125	4.05	4.67	2.66	5.33	0.00
	OFR	75	75	75	4.04	4.62	2.72	5.33	0.00
Heartac2	RPROP	10	7	10	3.74	4.49	4.28	6.67	11.97
	ASD	5	2	5	3.90	4.59	4.11	9.33	16.32
	OFR	5	2	5	3.89	4.64	4.09	9.33	15.91
Heartac3	RPROP	20	9	20	2.89	5.12	5.07	12.00	7.43
	ASD	5	2	5	3.23	5.00	5.43	16.00	7.04
	OFR	5	2	5	3.24	5.00	5.44	16.00	6.84



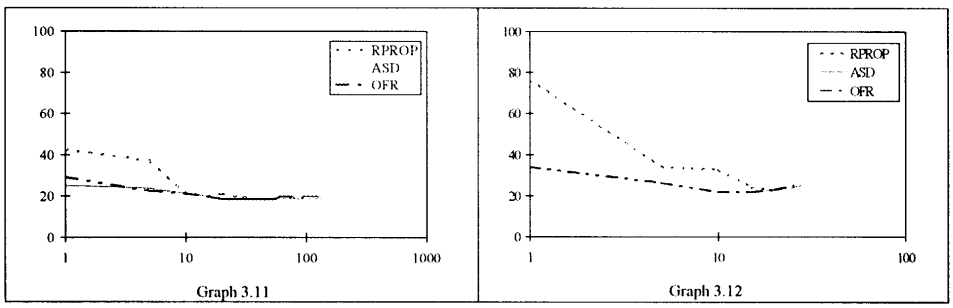
GRAPHS 3.5, 3.6 Test set classification error for the best run of problems cancer1, card1.



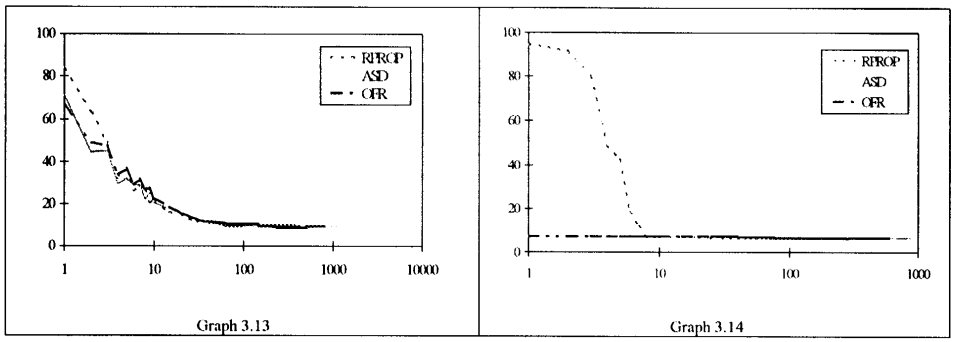
GRAPHS 3.7, 3.8 Test set classification error for the best run of problems diabetes1, gene1.



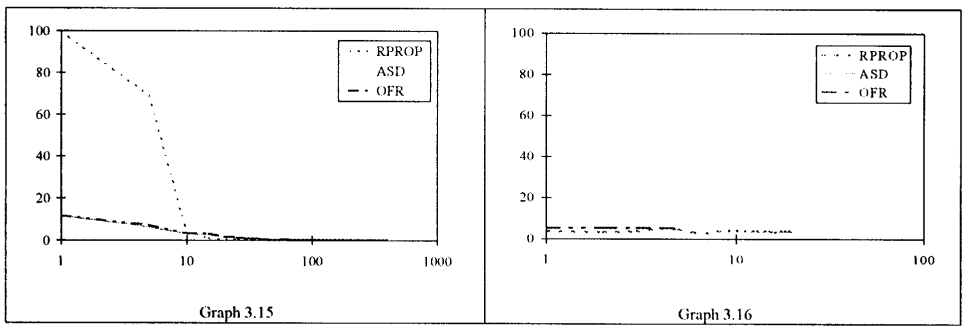
GRAPHS 3.9, 3.10 Test set classification error for the best run of problems glass1, heart1.



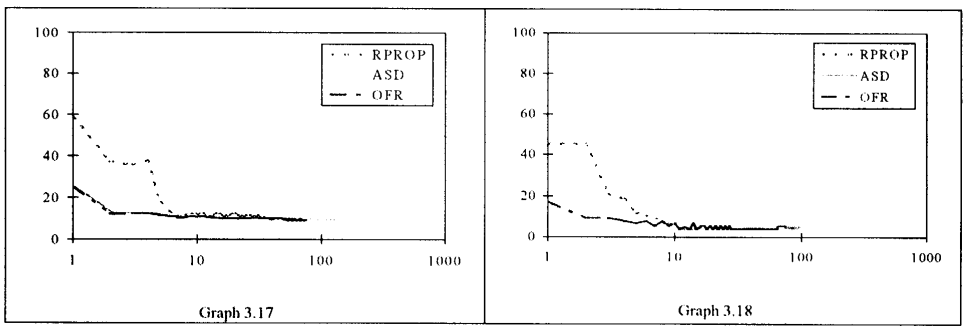
GRAPHS 3.11, 3.12 Test set classification error for the best run of problems heartac1, horse1.



GRAPHS 3.13, 3.14 Test set classification error for the best run of problems soybean1, thyroid1.



GRAPHS 3.15, 3.16 Test set classification error for the best run of problems building1, flare1.



GRAPHS 3.17, 3.18 Test set classification error for the best run of problems hearta1, heartac1.

4 CONCLUSIONS

In this paper we discussed a linear neural network design and implementation for solving pattern classification and function approximation problems, taken by the Proben1 benchmark collection. Batch-LMS neural network training rule is modified in order to lead to Adaptive

Steepest Descent (ASD) and Optimal Fletcher-Reeves (OFR) methods. These methods have been shown to accelerate the convergence of the learning phase, and they do not require the choice of critical parameters, like the learning rate or the momentum. When these methods are applied to real-world benchmarking problems, they produce adequate solutions, although they use a linear neural network architecture. They guarantee fast network convergence, generating a Least Square Solution, and they have good convergence properties. An extension of the presented architecture could be used, introducing multilayer networks with sigmoidal hidden nodes. Such an architecture has been tested by Prechelt [8], who used one-hidden and two-hidden layer networks, with various numbers of hidden nodes, and linear-RPROP training. The results of these architectures for some problems were worse than those obtained using linear networks, and the tendency to overfit was much higher than for linear networks, which suggests that introducing non-linearity is not an improvement.

References

[1] Amari, S., Murata, N., Muller, K. R., Finke, M. and Yang, H. (1995) Asymptotic statistical theory of overtraining and cross-validation, METR 95-06; Department of Mathematical Engineering and Information Physics, University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo 113, Japan.

[2] Goulianas, K., Adamopoulos, M. and Margaritis, K. G. (1997) Structured artificial neural networks for fast batch LMS algorithms, *Neural, Parallel and Scientific Computations* **5**(4), 549–562.

[3] Golub, G. H. and Van Loan, C. F. (1983) *Matrix Computations* (The Johns Hopkins University Press, Baltimore, MD).

[4] Hassoun, M. H. (1995) *Fundamentals of Artificial Neural Networks* (The MIT Press, Cambridge, MA).

[5] Luo Zhi-Quan (1991) On the convergence of the LMS algorithm with adaptive learning rate for linear feedforward networks, *Neural Computation* **3**, 226–245.

[6] Margaritis, K. G., Adamopoulos, M., Goulianas, K. and Evans, D. J. (1994) Artificial neural networks and iterative linear algebra methods, *Parallel Algorithms and Applications* **3**, 31–44.

[7] Polycarpou, M. and Ioannou, P. (1992) Learning and convergence analysis of neural-type structured networks, *IEEE Transactions on Neural Networks* **3**(1), 39–50.

[8] Prechelt Lutz (1994) *Proben1 – A Set of Neural Network Benchmark Problems and Benchmarking Rules*, Technical Report, Universitat Karlsruhe 21/94.

[9] Riedmiller, M. and Braun, H. (1993) A direct adaptive method for faster backpropagation learning: the RPROP algorithm, *Proceedings of the IEEE International Conference on Neural Networks*, San Fransisco, CA, April 1993, IEEE.

[10] Rumelhart, D. E., Hinton, G. E. and Williams, R. J. (1986) Learning internal representation by error propagation. In: Rumelhart, D. E. and McClelland, J. (eds.), *Parallel Distributed Processing I* (MIT Press, Cambridge, MA).

[11] Rumelhart, D. E. and McClelland, J. (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (MIT Press, Cambridge, MA).

[12] Sarle, W. S. (1995) Stopped training and other remedies for overfitting, *Proceedings of the 27th Symposium on the Interface of Computing Science and Statistics*, pp. 352–360.

[13] Stuart, G., Bienenstock, E. and Doursat, R. (1992) Neural networks and the bias/variance dilemma, *Neural Computation* **4**, 1–58.

[14] Wang, L. X. and Mendel, J. M. (1991) Three-dimensional structured networks for matrix equation solving, *IEEE Transactions on Computers* **40**(12), 1337–1346.

[15] Wang, L. X. and Mendel, J. M. (1992) Parallel structured networks for solving a wide variety of matrix algebra problems, *Journal of Parallel and Distributed Computing* **14**, 236–247.

[16] Wang, C., Venkatesh, S. S. and Judd, J. S. (1994) Optimal stopping and effective machine complexity in learning, *NIPS6*, 303–310.

[17] Widrow, B. and Hoff, M. E. Jr. (1960) Adaptive switching circuits, *IRE Western Electric Show and Convention Record*, Part 4, 96–104.

[18] Widrow, B. and Lehr, M. (1990) 30 years of adaptive neural networks: perceptron, madaline and backpropagation, *Proceedings of the IEEE* **78**(9), 1415–1442.