Adaptive Least Error Rate Algorithm for Neural Network Classifiers

S. Chen[†], B. Mulgrew[‡] and L. Hanzo[†]

[†] Department of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, U.K.
sqc@ecs.soton.ac.uk lh@ecs.soton.ac.uk

[‡] Department of Electronics and Electrical Engineering
University of Edinburgh, Edinburgh EH9 3JL, U.K.
Bernie.Mulgrew@ee.ed.ac.uk

Presented at IEEE Workshop NNSP 2001, September 10-12, 2001
Falmouth, Massachusetts, USA

Motivations

Equalization and multiuser detection applications $\rightarrow$ classification

- Real-time computational constraint

  Sample-by-sample adaptation or stochastic algorithms

- Minimize bit error rate

  Traditional mean square error based may not be right one

- System BER is very low

  "Adjusting classifier only when error occurs" strategy converges too slowly

Previous Works for Linear Case

- Difference approximation by perturbation to estimate stochastic gradient of one-sample error rate (Pados & Papantoni-Kazakos, Trans NN 1995; Psaromiligkos *et al*, Trans COM 1999)

  Readily applicable to nonlinear case. Effectively only adjusting when error occurs, complexity $O(N_p^2)$.

- AMBER or "modifying" sgn LMS so that algorithm continuously updates in a region around decision boundary even when error does not occur (Yeh & Barry, ICC'97; Yeh *et al*, Globecom'98)

  Not readily for nonlinear case. Very simple, complexity $O(N_p)$.

- LBER (Bulgrew and Chen, Symp. ASSPCC 2000; Chen *et al*, Trans SP 2001).

  Complexity $O(N_p)$, better performance $\rightarrow$ nonlinear case

Problem Formulation

Classifier

$$\hat{c}(k) = \text{sgn}(y(k)) \quad \text{with} \quad y(k) = f(\mathbf{r}(k); \mathbf{w})$$

$\mathbf{r}(k)$: M-dimensional pattern vector, $c(k) \in \{\pm 1\}$: class label
$\mathbf{w}$: parameters of classifier f, $\hat{c}(k)$: estimated class label for $\mathbf{r}(k)$.

$$\mathbf{r}(k) = \bar{\mathbf{r}}(k) + \mathbf{n}(k)$$

$\bar{\mathbf{r}}(k) \in \{\mathbf{r}_j,\ 1 \le j \le N_b\}$, and $\mathbf{n}(k)$ Gaussian with $E[\mathbf{n}(k)\mathbf{n}^T(k)] = \sigma_n^2\mathbf{I}$.
Each $\mathbf{r}_j$ has associated class label $c^{(j)} \in \{\pm 1\}$.

Let pdf of $y_s(k) = \text{sgn}(c(k))y(k)$ be $p_y(y_s)$

$$P_E(\mathbf{w}) = \text{Prob}\{\text{sgn}(c(k))y(k) < 0\} = \int_{-\infty}^{0} p_y(y_s)\, dy_s$$

Approximate Error Rate

Linearization around $\bar{\mathbf{r}}(k)$,

$$y(k) \approx f(\bar{\mathbf{r}}(k); \mathbf{w}) + e(k) = \bar{y}(k) + e(k)$$

$e(k)$: Gaussian with zero mean and variance $\rho^2 = \rho^2(\mathbf{w})$
$\bar{y}(k) \in \{y_j = f(\mathbf{r}_j; \mathbf{w}), \quad 1 \le j \le N_b\}$

$$p_y(y_s) \approx \frac{1}{N_b \sqrt{2\pi}\rho} \sum_{j=1}^{N_b} \exp\left(-\frac{(y_s - \mathrm{sgn}(c^{(j)}) y_j)^2}{2\rho^2} \right)$$

$$P_E(\mathbf{w}) \approx \frac{1}{N_b} \sum_{j=1}^{N_b} Q(g_j(\mathbf{w}))$$

$$g_j(\mathbf{w}) = \mathrm{sgn}(c^{(j)}) y_j / \rho = \mathrm{sgn}(c^{(j)}) f(\mathbf{r}_j; \mathbf{w}) / \rho$$

Approximate Minimum Error Rate Solution

Assume ρ^2 is fixed (to its optimal value $\rho^2(\mathbf{w}_{\text{opt}})$)

$$\nabla P_E(\mathbf{w}) \approx -\frac{1}{N_b\sqrt{2\pi}\rho}\sum_{j=1}^{N_b}\exp\left(-\frac{y_j^2}{2\rho^2}\right)\text{sgn}(c^{(j)})\frac{\partial f(\mathbf{r}_j;\mathbf{w})}{\partial \mathbf{w}}$$

Given $\mathbf{w}(0)$, at lth iteration:

$$\left.\begin{array}{l} y_j(l) = f(\mathbf{r}_j;\mathbf{w}(l-1)), \quad 1 \le j \le N_b \\ \nabla P_E(\mathbf{w}(l)) = -\frac{1}{N_b\sqrt{2\pi}\rho}\sum_{j=1}^{N_b}\exp\left(-\frac{y_j^2(l)}{2\rho^2}\right)\text{sgn}(c^{(j)})\frac{\partial f(\mathbf{r}_j;\mathbf{w}(l-1))}{\partial \mathbf{w}} \\ \mathbf{w}(l) = \mathbf{w}(l-1) - \mu\nabla P_E(\mathbf{w}(l)) \end{array}\right\}$$

- ρ^2, like adaptive gain μ, becomes a tunable algorithm parameter

Block-data Gradient Algorithm

Given training samples $\{\mathbf{r}(k), c(k)\}_{k=1}^{K}$, kernel density estimate of $p_y(y_s)$

$$\hat{p}_y(y_s) = \frac{1}{K\sqrt{2\pi}\rho} \sum_{k=1}^{K} \exp\left(-\frac{(y_s - \text{sgn}(c(k))y(k))^2}{2\rho^2} \right)$$

From estimated error probability

$$\hat{P}_E(\mathbf{w}) = \int_{-\infty}^{0} \hat{p}_y(y_s)\, dy_s$$

$$\nabla \hat{P}_E(\mathbf{w}) = -\frac{1}{K\sqrt{2\pi}\rho} \sum_{k=1}^{K} \exp\left(-\frac{y^2(k)}{2\rho^2} \right) \text{sgn}(c(k)) \frac{\partial f(\mathbf{r}(k); \mathbf{w})}{\partial \mathbf{w}}$$

$\Rightarrow$ block-data based gradient algorithm

Stochastic Gradient Algorithm

Using single-sample estimate of $p_y(y_s)$

$$\hat{p}_y(y_s,k) = \frac{1}{\sqrt{2\pi}\rho}\exp\left(-\frac{(y_s - \text{sgn}(c(k))y(k))^2}{2\rho^2}\right)$$

and instantaneous gradient $\nabla\hat{P}_E(k;\mathbf{w}) \Rightarrow$ LER algorithm

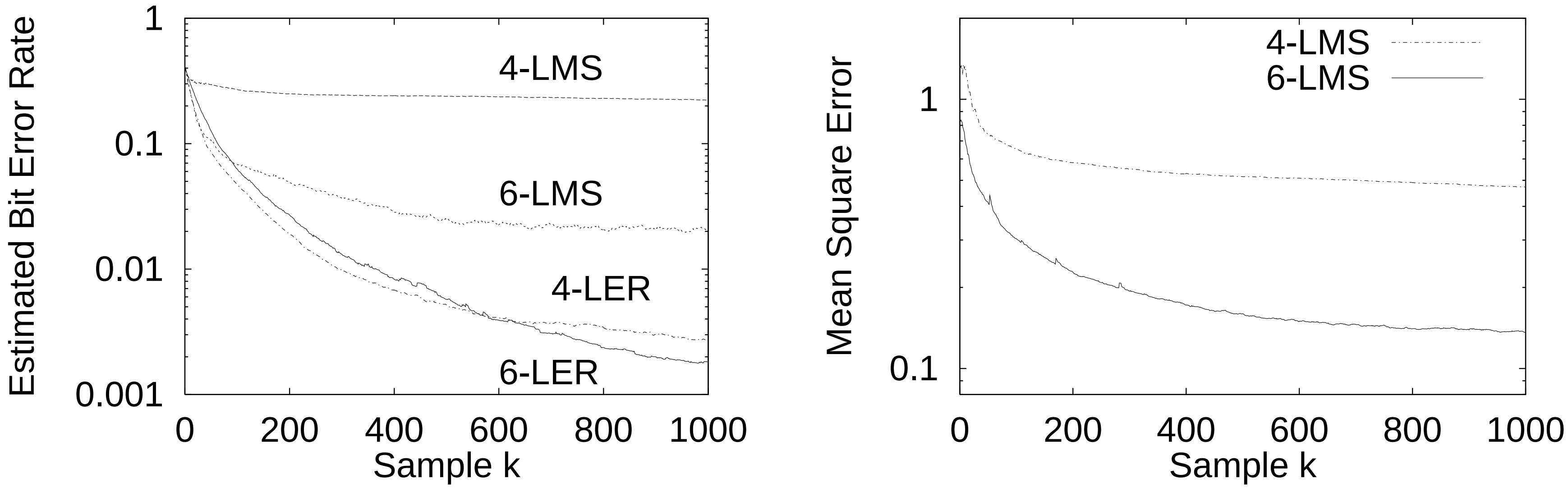
$$\left.\begin{aligned} y(k) &= f(\mathbf{r}(k);\mathbf{w}(k-1)) \\ \mathbf{w}(k) &= \mathbf{w}(k-1) + \frac{\mu}{\sqrt{2\pi}\rho}\exp\left(-\frac{y^2(k)}{2\rho^2}\right)\text{sgn}(c(k))\frac{\partial f(\mathbf{r}(k);\mathbf{w}(k-1))}{\partial \mathbf{w}} \end{aligned}\right\}$$

Two algorithm parameters: μ – adaptive gain, ρ – width

They need to be chosen appropriately

Equalization Example

Channel $A_0(z) = 0.5 + 1.0z^{-1}$, co-channel $A_1(z) = \lambda(1.0 + 0.5z^{-1})$ with λ set to give SIR$= 12$ dB, equalizer order $M = 2$ and decision delay $d = 1$, number of states $N_b = 64$. With SNR$= 20$ dB (SINR$= 11.36$ dB):



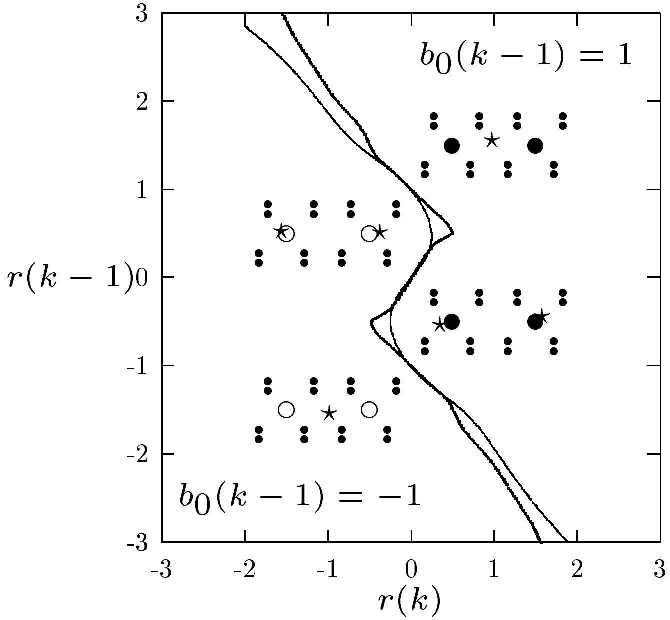
(a) (b)

Convergence rates in terms of (a) estimated BER for various adaptive RBF equalizers, and (b) MSE for LMS adaptive RBF equalizers. Results averaged over 100 runs.

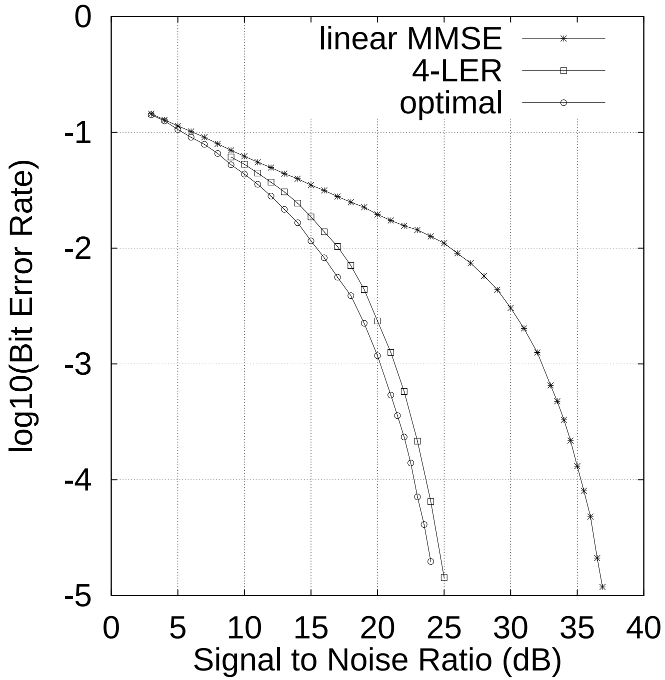
Equalization Example (continue)

(a) Comparison of optimal decision boundary (thick solid) with that of 6-center LER RBF equalizer (thin solid). SNR $= 20$ dB and SIR $= 12$ dB. Dots: noise-free states and stars: final centers. (b) Performance comparison of three equalizers in terms of BER versus SNR. SIR $= 12$ dB and adaptive LER RBF equalizer has 4 centers.

(a)

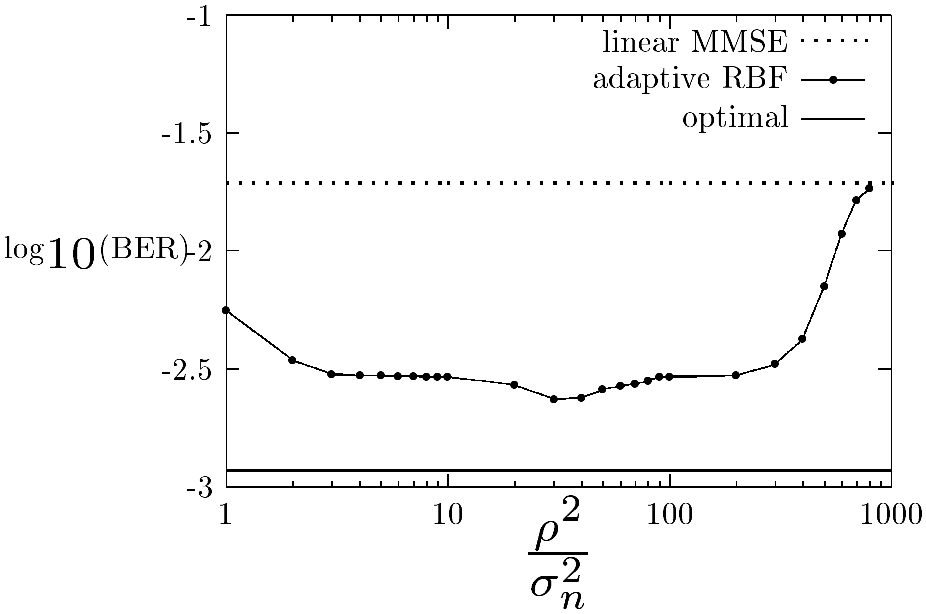


(b)



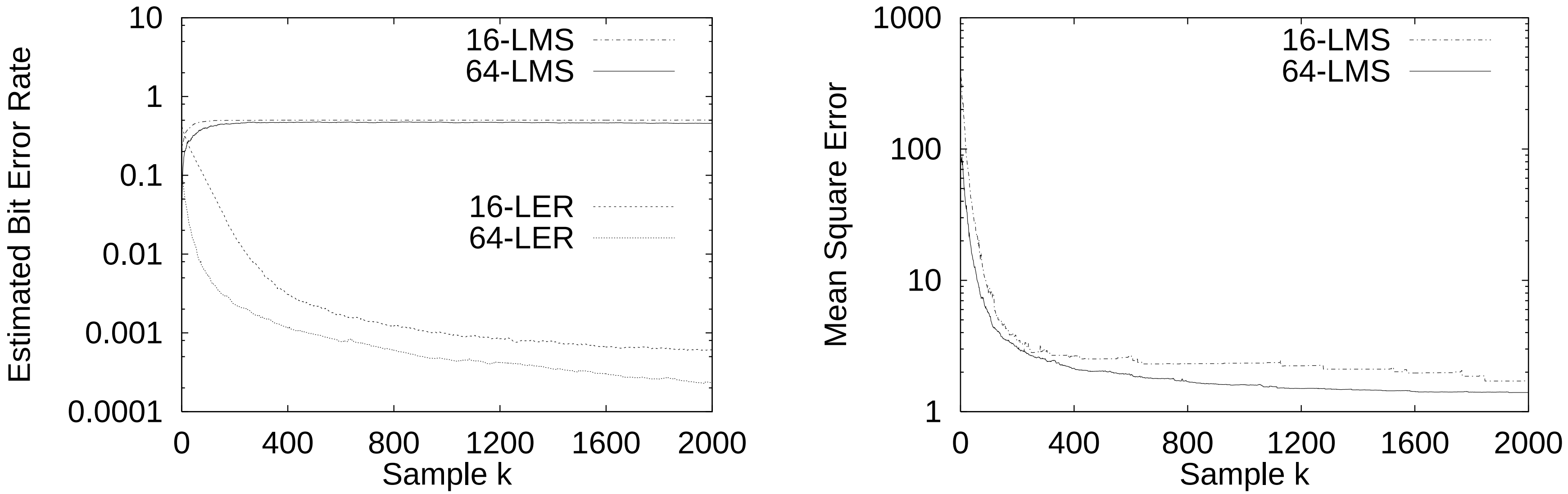
Equalization Example (continue)

Influence of ρ^2 to the performance of the LER algorithm. SIR $= 12$ dB and SNR $= 20$ dB. The adaptive RBF equalizer has 4 centers and the algorithm has a fixed μ_0.



CDMA Multiuser Detection Example

A three-equal-power-user system with eight chips per symbol. $M = 8$ and number of states $N_b = 64$. User 3 is considered. Given $\text{SNR}_3 = 15$ dB ($\text{SINR}_3 = -3.08$ dB):

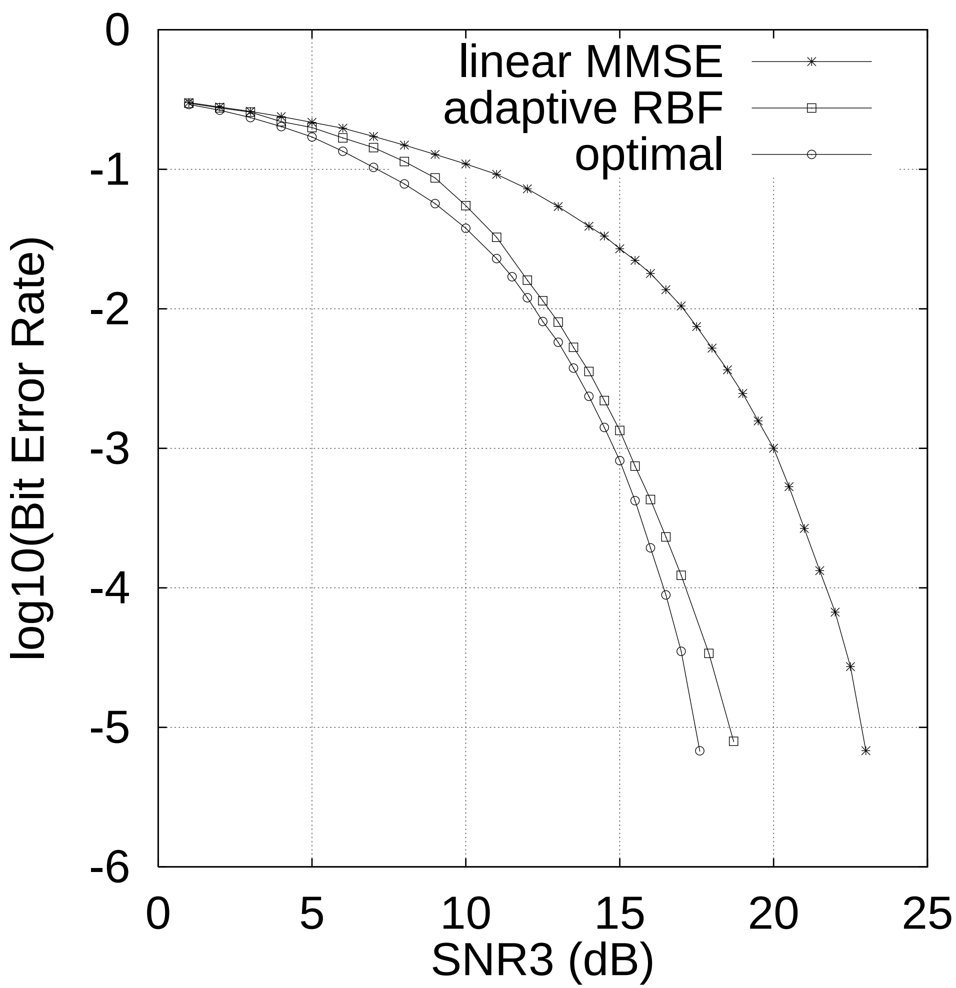


(a) (b)

Convergence rates in terms of (a) estimated BER for various user-3 adaptive RBF detectors and (b) MSE for user-3 LMS RBF detectors. Results averaged over 100 runs.

CDMA Multiuser Detection (continue)

Performance comparison of three detectors for user 3. SNR_i, $1 \leq i \leq 3$, identical. Adaptive RBF detector has 16 centers and trained by LER algorithm.



Conclusions

- LER: an adaptive stochastic gradient near minimum error rate training for nonlinear classifiers
  - ⋆ MSE criterion may not be relevant to problem and may lead to poor performance
  - ⋆ Approach based on kernel density estimation and stochastic approximation for sample-by-sample training
  - ⋆ Work well for low error rate or high performance situations
- Results verified in channel equalization and CDMA downlink multiuser detection