Approximate Maximum Margin Algorithms with Rules Controlled by the Number of Mistakes

Petroula Tsampouka[1] and John Shawe-Taylor[1,2]

[1] School of Electronics and Computer Science
University of Southampton, UK
[2] Department of Computer Science
University College London, UK
pt04r@ecs.soton.ac.uk J.Shawe-Taylor@cs.ucl.ac.uk

Abstract. We present a family of Perceptron-like algorithms with margin in which both the "effective" learning rate, defined as the ratio of the learning rate to the length of the weight vector, and the misclassification condition are independent of the length of the weight vector but, instead, are entirely controlled by rules involving (powers of) the number of mistakes. We examine the convergence of such algorithms in a finite number of steps and show that under some rather mild assumptions there exists a limit of the parameters involved in which convergence leads to classification with maximum margin. Very encouraging experimental results obtained using algorithms which belong to this family are also presented.

1 Introduction

Maximising the margin of the solution hyperplane, which plays an important role in the generalisation ability of a learning machine, is a central objective of Support Vector Machines (SVMs) [14, 1]. Their efficient implementation, however, is somewhat hindered by the fact that they require solving a quadratic programming problem.

The ambition to surpass the implementational difficulties associated with SVMs while retaining all the benefits of the large margin solutions led to a revival of the interest in alternative large margin classifiers which are able to operate directly on the primal maximal margin problem instead of its dual. Such algorithms include the standard Perceptron with margin [2, 7], the Maximal Margin Perceptron [6] and the related algorithm of [5], the aggressive ROMMA [8] and ALMA [4] algorithms. Among these algorithms the standard Perceptron with margin and ALMA may be considered as variants of the classical Perceptron algorithm [10]. Our purpose here is to address the maximal margin classification problem in the context of Perceptron-like algorithms which, however, differ from the above variants in that the "effective" learning rate [13] and the misclassification condition do not depend on the length of the weight vector at all but, instead, are entirely controlled by rules involving (powers of) the number of mistakes. This novel (class of) algorithm(s) will be called Mistake-Controlled Rule Algorithm(s) (MICRA). Under certain conditions MICRA converges in a finite number of steps to an approximation of the optimal solution which keeps improving as the parameters of the algorithm follow a specific limiting process.

An introductory discussion of Perceptron-like large margin classifiers leading to the construction of MICRA can be found in Sect. 2. MICRA is described in Sect. 3 together with an analysis regarding its convergence. Section 4 contains some experiments whereas Sect. 5 our conclusions.

2 Perceptron-Like Large Margin Classifiers

In what follows we make the assumption that we are given a training set which, even if not initially linearly separable can, by an appropriate feature mapping into a space of a higher dimension [14, 1], be classified into two categories by a linear classifier. This higher dimensional space in which the patterns are linearly separable will be the considered space. By adding one additional dimension and placing all patterns in the same position at a distance ρ in that dimension we construct an embedding of our data into the so-called augmented space [2]. The advantage of this embedding is that the linear hypothesis in the augmented space becomes homogeneous. Thus, only hyperplanes passing through the origin in the augmented space need to be considered even for tasks requiring bias. Throughout our discussion a reflection with respect to the origin in the augmented space of the negatively labelled patterns is assumed in order to allow for a uniform treatment of both categories of patterns. Also, we use the notation $R = \max_k \|\boldsymbol{y}_k\|$, where $\boldsymbol{y}_k$ is the k^{th} augmented pattern. Obviously, $R \geq \rho$.

The relation characterising optimally correct classification of the training patterns $\boldsymbol{y}_k$ by a weight vector $\boldsymbol{u}$ of unit norm in the augmented space is

$$\boldsymbol{u} \cdot \boldsymbol{y}_k \geq \gamma_{\text{d}} \equiv \max_{\boldsymbol{u'}: \|\boldsymbol{u'}\|=1} \min_i \{\boldsymbol{u'} \cdot \boldsymbol{y}_i\} \qquad \forall k \ . \tag{1}$$

The quantity γ_{d} will be referred to as the maximum directional margin. It coincides with the maximum margin in the augmented space with respect to hyperplanes passing through the origin if no reflection is assumed. Between γ_{d} and the maximum geometric margin γ in the original space the inequality

$$1 \leq \frac{\gamma}{\gamma_{\text{d}}} \leq \frac{R}{\rho} \tag{2}$$

holds. In the limit $\rho \to \infty$, $R/\rho \to 1$ and from (2) $\gamma_{\text{d}} \to \gamma$ [12].

We concentrate on algorithms that update the augmented weight vector $\boldsymbol{a}_t$ by adding a suitable positive amount in the direction of the misclassified (according to an appropriate condition) training pattern $\boldsymbol{y}_k$. The general form of such an update rule is

$$\boldsymbol{a}_{t+1} = (\boldsymbol{a}_t + \eta_t f_t \boldsymbol{y}_k)\, N_{t+1}^{-1} \ , \tag{3}$$

where η_t is the learning rate which could depend explicitly on the number t of updates that took place so far and f_t an implicit function of the current step (update) t, possibly involving the current weight vector $\boldsymbol{a}_t$ and/or the current misclassified pattern $\boldsymbol{y}_k$, which we require to be bounded by positive constants. We also allow for the possibility of normalising the newly produced weight vector $\boldsymbol{a}_{t+1}$ to a desirable length through a factor N_{t+1}. For the Perceptron $\eta_t = \eta$ is constant, $f_t = 1$ and $N_{t+1} = 1$. Each time the misclassification condition is satisfied by a training pattern, that is a mistake occurs, the algorithm proceeds to the update of $\boldsymbol{a}_t$. We adopt the convention of initialising t from 1.

A sufficiently general form of the misclassification condition is

$$\boldsymbol{u}_t \cdot \boldsymbol{y}_k \leq C(t) \quad ,$$

where $\boldsymbol{u}_t$ is the weight vector $\boldsymbol{a}_t$ normalised to unity and $C(t) > 0$ if we require that the algorithm achieves a positive margin. If $\boldsymbol{a}_1 = \boldsymbol{0}$ we treat the first pattern in the sequence as misclassified. In the case that $C(t)$ is bounded from above by a strictly decreasing function of t which tends to zero the minimum directional margin required by such a condition becomes lower than any fixed value provided t is large enough. Algorithms with such a condition have the advantage of achieving some fraction of the unknown existing margin provided they converge. Examples of such algorithms are the well-known standard Perceptron algorithm with margin [2, 7] with $C(t) = b/\|\boldsymbol{a}_t\|$ and the ALMA$_2$ algorithm [4] with $C(t) = b/(\|\boldsymbol{a}_t\| \sqrt{t})$. Here b is a positive constant. For the Perceptron the suppression of $C(t)$ with t increasing is due to the growth of $\|\boldsymbol{a}_t\|$ which is bounded from below by a positive linear function of t whereas for ALMA$_2$ $C(t)$ is partly suppressed due to the growth of $\|\boldsymbol{a}_t\|$ up to a fixed upper bound and partly due to the growth of $\sqrt{t}$.

Another important quantity characterising algorithms with the perceptron-like update rule (3) is the "effective" learning rate [13]

$$\eta_{\mathrm{eff}\,t} \equiv \frac{\eta_t R}{\|\boldsymbol{a}_t\|}$$

which controls the impact that an update has on the current weight vector. More specifically, $\eta_{\mathrm{eff}\,t}$ determines the update of the direction $\boldsymbol{u}_t$

$$\boldsymbol{u}_{t+1} = \frac{\boldsymbol{u}_t + \eta_{\mathrm{eff}\,t} f_t \boldsymbol{y}_k / R}{\|\boldsymbol{u}_t + \eta_{\mathrm{eff}\,t} f_t \boldsymbol{y}_k / R\|} \quad . \tag{4}$$

In the most well-known cases $\eta_{\mathrm{eff}\,t}$ is bounded from above by a strictly decreasing function of t which tends to zero. Examples are the standard Perceptron with margin in which $\eta_t = \eta$ remains constant and ALMA$_2$ in which η_t decreases as $1/\sqrt{t}$. In both cases the growth of $\|\boldsymbol{a}_t\|$ with t contributes to the suppression of $\eta_{\mathrm{eff}\,t}$. Moreover, both these algorithms have a t-independent ratio $\eta_{\mathrm{eff}\,t}/C(t)$.

From the above discussion it becomes obvious that a Perceptron-like algorithm with the additive update (3) is uniquely determined by the functions $C(t)$, $\eta_{\mathrm{eff}\,t}$ and f_t. In particular, it does not depend on $\|\boldsymbol{a}_t\|$ as long as the above functions are $\|\boldsymbol{a}_t\|$-independent. Our purpose here is to examine the sufficiently large subclass of such algorithms with $f_t = 1$ and $C(t)$, $\eta_{\mathrm{eff}\,t}$ inversely proportional to powers of the number of mistakes t and determine sufficient conditions under which algorithms in the above subclass converge asymptotically to the optimal solution hyperplane.

3 The Mistake-Controlled Rule Algorithm MICRA$^{\epsilon,\zeta}$

We consider algorithms with effective learning rate

$$\eta_{\mathrm{eff}\,t} = \frac{\eta}{t^{\zeta}} \tag{5}$$

and misclassification condition

$$\boldsymbol{u}_t \cdot \boldsymbol{y}_k \leq \frac{\beta}{t^{\epsilon}} \ . \tag{6}$$

Both (5) and (6) do not involve $\|\boldsymbol{a}_t\|$. Here η, ζ, β and ϵ are positive constants. The case $\zeta = 0$, corresponding to a constant effective learning rate, is treated in [13]. We assume that the initial value $\boldsymbol{u}_1$ of $\boldsymbol{u}_t$ is the unit vector in the direction of the first training pattern. Then,

$$\boldsymbol{u}_t \cdot \boldsymbol{u} > 0 \ . \tag{7}$$

This is true given that, on account of (4), $\boldsymbol{u}_t$ is a linear combination with positive coefficients of the training patterns $\boldsymbol{y}_k$ all of which have positive inner products with the optimal direction $\boldsymbol{u}$ because of (1). Additionally, we set $f_t = 1$. Since $\eta_{\mathrm{eff}\,t}$ of (5) and the misclassification condition of (6) do not depend on $\|\boldsymbol{a}_t\|$ and given that the update (4) of $\boldsymbol{u}_t$ with $f_t = 1$ depends on $\|\boldsymbol{a}_t\|$ only through $\eta_{\mathrm{eff}\,t}$ the algorithm is $\|\boldsymbol{a}_t\|$-independent.

The above (family of) algorithm(s) parametrised in terms of the exponents ϵ and ζ will be called the Mistake-Controlled Rule Algorithm MICRA$^{\epsilon,\zeta}$ and is summarised in Fig. 1.

Theorem 1. *The* MICRA$^{\epsilon,\zeta}$ *algorithm of Fig. 1 converges in a finite number of steps provided* $0 < \zeta \leq 1$. *Moreover, if* η *is given a dependence on* β *through the relation* $\eta = \eta_0 \left(\frac{\beta}{R}\right)^{-\delta}$ *the directional margin* γ'_{d} *that the algorithm achieves tends in the limit* $\frac{\beta}{R} \to \infty$ *to the maximum directional margin* γ_{d} *provided* $0 < \epsilon\delta + \zeta < 1$.

```
Require: A linearly separable augmented training set with reflection
assumed $S = (\boldsymbol{y}_1, \dots, \boldsymbol{y}_m)$
Define:
For $k = 1, \dots, m$
$R = \max_k \|\boldsymbol{y}_k\|$ ,  $\bar{\boldsymbol{y}}_k = \boldsymbol{y}_k / R$
Fix: $\eta$,  $\beta_1 \ (= \beta/R)$
Initialisation:
$t = 1$,  $\boldsymbol{u}_1 = \bar{\boldsymbol{y}}_1 / \|\bar{\boldsymbol{y}}_1\|$,  $\eta_{\mathrm{eff}\,1} = \eta$
repeat until no update made within the for loop
  for $k = 1$ to $m$ do
    if $\boldsymbol{u}_t \cdot \bar{\boldsymbol{y}}_k \leq \beta_t$ then
      $\boldsymbol{u}_{t+1} = \frac{\boldsymbol{u}_t + \eta_{\mathrm{eff}\,t} \bar{\boldsymbol{y}}_k}{\|\boldsymbol{u}_t + \eta_{\mathrm{eff}\,t} \bar{\boldsymbol{y}}_k\|}$
      $t = t + 1$
      $\beta_t = \beta_1 / t^{\epsilon}$,  $\eta_{\mathrm{eff}\,t} = \eta / t^{\zeta}$
```

Fig. 1. The mistake-controlled rule algorithm MICRA$^{\epsilon,\zeta}$.

Proof. Taking the inner product of (4) with the optimal direction $\boldsymbol{u}$, expanding $\|\boldsymbol{u}_t + \eta_{\mathrm{eff}} \boldsymbol{y}_k / R\|^{-1}$ and using the inequality $(1+x)^{-\frac{1}{2}} \geq 1 - \frac{x}{2}$ we have

$$\begin{aligned}\boldsymbol{u}_{t+1} \cdot \boldsymbol{u} &= (\boldsymbol{u}_t \cdot \boldsymbol{u} + \eta_{\mathrm{eff}\,t} \boldsymbol{y}_k \cdot \boldsymbol{u}/R) \left(1 + 2\eta_{\mathrm{eff}\,t} \boldsymbol{y}_k \cdot \boldsymbol{u}_t / R + \eta^2_{\mathrm{eff}\,t} \|\boldsymbol{y}_k\|^2 / R^2\right)^{-\frac{1}{2}} \\ &\geq (\boldsymbol{u}_t \cdot \boldsymbol{u} + \eta_{\mathrm{eff}\,t} \boldsymbol{y}_k \cdot \boldsymbol{u}/R) \left(1 - \eta_{\mathrm{eff}\,t} \boldsymbol{y}_k \cdot \boldsymbol{u}_t / R - \eta^2_{\mathrm{eff}\,t} \|\boldsymbol{y}_k\|^2 / 2R^2\right) \quad .\end{aligned}$$

Thus, we obtain for $\mathcal{D} \equiv \boldsymbol{u}_{t+1} \cdot \boldsymbol{u} - \boldsymbol{u}_t \cdot \boldsymbol{u}$

$$\begin{aligned}\frac{R}{\eta_{\mathrm{eff}\,t}} \mathcal{D} \geq \boldsymbol{y}_k \cdot \boldsymbol{u} - (\boldsymbol{u}_t \cdot \boldsymbol{u})(\boldsymbol{y}_k \cdot \boldsymbol{u}_t) - \frac{\eta_{\mathrm{eff}\,t}}{2R} \left(\|\boldsymbol{y}_k\|^2 \boldsymbol{u}_t \cdot \boldsymbol{u} + 2(\boldsymbol{y}_k \cdot \boldsymbol{u})(\boldsymbol{y}_k \cdot \boldsymbol{u}_t)\right) \\ - \frac{\eta^2_{\mathrm{eff}\,t}}{2R^2} \|\boldsymbol{y}_k\|^2 \boldsymbol{y}_k \cdot \boldsymbol{u} \quad .\end{aligned}$$

By employing (1), (6) and (7) we get a lower bound on $\mathcal{D}$

$$\mathcal{D} \geq \eta_{\mathrm{eff}\,t} \left(\frac{\gamma_{\mathrm{d}}}{R} - \frac{\eta_{\mathrm{eff}\,t}}{2} - \frac{\eta^2_{\mathrm{eff}\,t}}{2} \right) - \eta_{\mathrm{eff}\,t} \left(1 + \eta_{\mathrm{eff}\,t}\right) \frac{\beta}{R} t^{-\epsilon} \quad . \tag{8}$$

From the misclassification condition it is obvious that convergence of the algorithm is impossible unless $\beta / t^{\epsilon} < \gamma_{\mathrm{d}}$ i.e.

$$t > t_0 \equiv \left(\frac{\beta}{\gamma_{\mathrm{d}}} \right)^{\frac{1}{\epsilon}} \quad .$$

A repeated application of (8) $(t - [t_0])$ times, where $[t_0]$ denotes the integer part of t_0, yields

$$\begin{aligned}\boldsymbol{u}_{t+1} \cdot \boldsymbol{u} - \boldsymbol{u}_{[t_0]+1} \cdot \boldsymbol{u} \geq \eta \frac{\gamma_{\mathrm{d}}}{R} \sum_{m=[t_0]+1}^{t} m^{-\zeta} - \frac{\eta^2}{2} \sum_{m=[t_0]+1}^{t} m^{-2\zeta} - \frac{\eta^3}{2} \sum_{m=[t_0]+1}^{t} m^{-3\zeta} \\ - \eta \frac{\beta}{R} \sum_{m=[t_0]+1}^{t} m^{-(\zeta+\epsilon)} - \eta^2 \frac{\beta}{R} \sum_{m=[t_0]+1}^{t} m^{-(2\zeta+\epsilon)} \quad .\end{aligned}$$

By employing the inequalities

$$\frac{t^{1-\theta} - (t_0+1)^{1-\theta}}{1-\theta} = \int_{t_0+1}^{t} m^{-\theta} dm \leq \sum_{m=[t_0]+1}^{t} m^{-\theta}$$

and

$$\sum_{m=[t_0]+1}^{t} m^{-\theta} \leq \int_{t_0}^{t} m^{-\theta} dm + t_0^{-\theta} = \frac{t^{1-\theta} - t_0^{1-\theta}}{1-\theta} + t_0^{-\theta}$$

for $\theta > 0$ and taking into account (7) we finally obtain

$$1 \geq \eta \frac{\gamma_{\mathrm{d}}}{R} \left(\frac{t^{1-\zeta} - t_0^{1-\zeta}}{1-\zeta} \right) - \frac{\eta^2}{2} \left(\frac{t^{1-2\zeta} - t_0^{1-2\zeta}}{1-2\zeta} \right) - \frac{\eta^3}{2} \left(\frac{t^{1-3\zeta} - t_0^{1-3\zeta}}{1-3\zeta} \right)$$

$$-\eta\frac{\beta}{R}\left(\frac{t^{1-(\zeta+\epsilon)}-t_0^{1-(\zeta+\epsilon)}}{1-(\zeta+\epsilon)}\right)-\eta^2\frac{\beta}{R}\left(\frac{t^{1-(2\zeta+\epsilon)}-t_0^{1-(2\zeta+\epsilon)}}{1-(2\zeta+\epsilon)}\right)-\omega \ . \qquad (9)$$

Here

$$\omega \equiv \frac{\gamma_{\rm d}}{R}\eta t_0^{-\zeta}\left(2+\eta t_0^{-\zeta}\right)+\frac{1}{2}\eta^2 t_0^{-2\zeta}\left(1+\eta t_0^{-\zeta}\right) > 0 \ .$$

Let us define the new variable $\tau \geq 0$ through the relation

$$t = t_0\left(1+\tau\right) = \left(\frac{\beta}{\gamma_{\rm d}}\right)^{\frac{1}{\epsilon}}\left(1+\tau\right) \ . \qquad (10)$$

In terms of τ (9) becomes

$$\left(\eta t_0^{1-\zeta}\right)^{-1}\left(\frac{\gamma_{\rm d}}{R}\right)^{-1}(1+\omega) \geq \frac{(1+\tau)^{1-\zeta}-1}{1-\zeta}-\frac{(1+\tau)^{1-(\zeta+\epsilon)}-1}{1-(\zeta+\epsilon)}$$

$$-\frac{R}{2\gamma_{\rm d}}\eta t_0^{-\zeta}\frac{(1+\tau)^{1-2\zeta}-1}{1-2\zeta}-\frac{R}{2\gamma_{\rm d}}\eta^2 t_0^{-2\zeta}\frac{(1+\tau)^{1-3\zeta}-1}{1-3\zeta}$$

$$-\eta t_0^{-\zeta}\frac{(1+\tau)^{1-(2\zeta+\epsilon)}-1}{1-(2\zeta+\epsilon)} \ . \qquad (11)$$

Let $g(\tau)$ be the r.h.s. of the above inequality. Since $0<\zeta\leq 1$, $g(\tau)$ (with $\tau\geq 0$) is unbounded from above. Moreover, its derivative $g'(\tau)$ satisfies the relation

$$(1+\tau)^{\zeta}g'(\tau) = 1-(1+\tau)^{-\epsilon}-\frac{R}{2\gamma_{\rm d}}\eta t_0^{-\zeta}(1+\tau)^{-\zeta}-\frac{R}{2\gamma_{\rm d}}\eta^2 t_0^{-2\zeta}(1+\tau)^{-2\zeta}$$

$$-\eta t_0^{-\zeta}(1+\tau)^{-(\zeta+\epsilon)} \ .$$

The r.h.s. of the above equation is a monotonically increasing function of τ which is negative at $\tau=0$ and tends to 1 as $\tau\to\infty$. Therefore $g'(\tau)$ has a single root at $\tau=\tau_{\rm min}$ which corresponds to a minimum of $g(\tau)$ with $g(\tau_{\rm min})<0$. Moreover, the l.h.s. of (11) is positive. Thus, given that $g(0)=0$, there is a single value $\tau_{\rm b}$ of τ where (11) holds as an equality which provides an upper bound on τ

$$\tau \leq \tau_{\rm b} \qquad (12)$$

satisfying $\tau_{\rm b}>\tau_{\rm min}>0$. Combining (10) and (12) we obtain the bound on the number of updates

$$t \leq t_{\rm b} \equiv \left(\frac{\beta}{\gamma_{\rm d}}\right)^{\frac{1}{\epsilon}}\left(1+\tau_{\rm b}\right) \qquad (13)$$

proving that the algorithm converges in a finite number of steps. From (13) and taking into account the misclassification condition (6) we obtain a lower bound $\beta/t_{\rm b}^{\epsilon}$ on the margin $\gamma'_{\rm d}$ achieved. Thus, the fraction f of $\gamma_{\rm d}$ that the algorithm achieves satisfies

$$f \equiv \frac{\gamma'_{\rm d}}{\gamma_{\rm d}} \geq f_{\rm b} \equiv \frac{\beta/\gamma_{\rm d}}{t_{\rm b}^{\epsilon}} = \left(1+\tau_{\rm b}\right)^{-\epsilon} \ . \qquad (14)$$

Let us assume that $\frac{\beta}{R} \to \infty$ in which case from $\eta = \eta_0 \left(\frac{\beta}{R}\right)^{-\delta}$ and given that $0 < \epsilon\delta + \zeta < 1$ we have $\eta t_0^{1-\zeta} \sim \left(\frac{\beta}{R}\right)^{\frac{1-\zeta-\epsilon\delta}{\epsilon}} \to \infty$ whereas $\eta t_0^{-\zeta} \sim \left(\frac{\beta}{R}\right)^{-\frac{\zeta+\epsilon\delta}{\epsilon}} \to 0$. Consequently the l.h.s. of (11) vanishes in the limit $\frac{\beta}{R} \to \infty$ whereas its r.h.s. (i.e. $g(\tau)$) becomes a strictly increasing function for $\tau > 0$ (i.e. $\tau_{\min} \to 0$) since $(1+\tau)^{\zeta} g'(\tau) = 1 - (1+\tau)^{-\epsilon} > 0$. Obviously, (11) holds as an equality only for $\tau = 0$. Therefore,

$$\tau_{\mathrm{b}} \to \tau_{\min} \to 0 \quad \text{as} \quad \frac{\beta}{\mathrm{R}} \to \infty \ . \tag{15}$$

Combining (14) with (15) and taking into account that $f \leq 1$ by definition we conclude that

$$f \to 1 \quad \text{as} \quad \frac{\beta}{\mathrm{R}} \to \infty \ .$$

□

Remark 1. In the case that $\zeta + 2\epsilon = 1$ with $\frac{1}{2} < \zeta < 1$ it is possible to obtain explicitly an upper bound t_{b} on the number of updates and a lower bound f_{b} on the fraction f of the margin that the algorithm achieves. First we observe that since $1-2\zeta$, $1-3\zeta$ and $1-(2\zeta+\epsilon)$ are negative it is allowed to set the terms $(1+\tau)^{1-2\zeta}$, $(1+\tau)^{1-3\zeta}$ and $(1+\tau)^{1-(2\zeta+\epsilon)}$ to zero in the r.h.s. of (11). Then, the resulting less restrictive inequality with ζ expressed in terms of ϵ becomes

$$\frac{A^2}{2} \geq \frac{1}{2}(1+\tau)^{2\epsilon} - (1+\tau)^{\epsilon} + \frac{1}{2} \ , \tag{16}$$

where

$$\frac{A^2}{2} = \epsilon\eta^{-1}\left(\frac{\beta}{R}\right)^{-2}\frac{\gamma_{\mathrm{d}}}{R}(1+\omega) + \frac{\epsilon}{1-4\epsilon}\frac{\eta}{2}\left(\frac{\beta}{R}\right)^{2-\frac{1}{\epsilon}}\left(\frac{\gamma_{\mathrm{d}}}{R}\right)^{\frac{1}{\epsilon}-3}$$
$$+\frac{\epsilon}{1-3\epsilon}\frac{\eta^2}{4}\left(\frac{\beta}{R}\right)^{4-\frac{2}{\epsilon}}\left(\frac{\gamma_{\mathrm{d}}}{R}\right)^{\frac{2}{\epsilon}-5} + \frac{\epsilon}{1-3\epsilon}\eta\left(\frac{\beta}{R}\right)^{2-\frac{1}{\epsilon}}\left(\frac{\gamma_{\mathrm{d}}}{R}\right)^{\frac{1}{\epsilon}-2} \ .$$

Notice that $\epsilon < \frac{1}{4}$ if $\frac{1}{2} < \zeta < 1$. By solving the quadratic equation derived from (16) we obtain explicitly the bounds t_{b} and f_{b}. They are the ones of (13) and (14), respectively with

$$\tau_{\mathrm{b}} = (1+|A|)^{\frac{1}{\epsilon}} - 1 \ .$$

In the present case $0 < \epsilon\delta + \zeta < 1$ is equivalent to $2 - \frac{1}{\epsilon} < \delta < 2$. Then, with $\eta = \eta_0\left(\frac{\beta}{R}\right)^{-\delta}$ as $\frac{\beta}{R} \to \infty$ we get $A \to 0$ leading to $\tau_{\mathrm{b}} \to 0$. This demonstrates explicitly the statement of Theorem 1. It is worth emphasising, however, that $|A|$ may be suppressed even for small $\frac{\beta}{R}$ if $\frac{\gamma_{\mathrm{d}}}{R}$ is small.

Example: If $\epsilon = \zeta = \frac{1}{2}$ and moreover $\delta = 0$, i.e. η does not depend on β, $\epsilon\delta + \zeta = \frac{1}{2}$ and the condition of Theorem 1 is satisfied. Therefore such an algorithm attains asymptotically as $\frac{\beta}{R} \to \infty$ the maximum directional margin. The above algorithm is a version of a well-known approximate maximal margin classifier, namely ALMA$_2$ [4]. In this version the weight vector instead of being confined within a ball centered at the origin is normalised to a constant length which remains fixed during the asymptotic procedure. Thus, ALMA$_2$ can be thought of as belonging to the MICRA family. Then, the analysis of [4] confirms our conclusion regarding asymptotic convergence to the optimal solution hyperplane in this special case. In the case, instead, that $\epsilon = \zeta = \frac{1}{2}$ but $\delta = 1$, i.e. $\eta = \eta_0 \left(\frac{\beta}{R}\right)^{-1}$, $\epsilon\delta + \zeta = 1$ and the condition of Theorem 1 is violated. This case would correspond to a version of ALMA$_2$ with the parameter b entering the misclassification condition set to $b = \beta^2$ and the weight vector normalised to the constant length β which, however, does not remain fixed during the asymptotic procedure $\frac{\beta}{R} \rightarrow \infty$. Since the condition of Theorem 1 is violated we are unable to prove asymptotic convergence of such an algorithm to the maximal margin solution. The same conclusion is reached if the proof technique of [4] is employed which gives a lower bound

$$f_{\mathrm{b}} = \left(1 + \frac{1}{\eta_0} + \frac{3}{2}\eta_0 \frac{R^2}{\beta^2}\right)^{-1}$$

on the fraction of the maximum directional margin achieved by the algorithm. As $\frac{\beta}{R} \rightarrow \infty$ we get $f_{\mathrm{b}} \rightarrow \frac{\eta_0}{(1+\eta_0)} < 1$. We see that a “slight” modification of the asymptotic procedure is able to affect the ability of a Perceptron-like algorithm to attain the solution with maximum margin. We have reasons to believe that the inability in some cases of the Perceptron algorithm with margin, in contrast to ALMA$_2$, to approach the maximal margin solution is due to such “slight” differences between the two algorithms regarding the asymptotic procedure.

4 Experiments

In this section we present the results of experiments performed in order to verify our theoretical analysis and evaluate the performance of MICRA in comparison with other two well-known algorithms, namely the Perceptron with margin and aggressive ROMMA[1]. The Perceptron is chosen as a fast and simple algorithm close in spirit to MICRA. The choice of agg-ROMMA, instead, is justified by the fact that it is claimed in [8] to be faster than SMO [9]. For MICRA we employ a β-independent η ($\delta = 0$) and ϵ, ζ values for which, in most cases, the analysis of Remark 1 applies.

First we analyse the training data set of the sonar classification problem as originally selected for the aspect-angle dependent experiment. It consists of 104 instances each with 60 attributes obtainable from the UCI repository. Here the data are embedded in the augmented space at a distance $\rho = 1$ from the

[1] The parameter $\delta \in (0, 1]$ in agg-ROMMA controls the accuracy to which the maximum margin is approximated. It should not be confused with the parameter δ in Theorem 1.

Table 1. Results for the sonar data set. The directional margin $\gamma'_{\rm d}$ achieved and the number of updates (upds) are given for the Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.05, \zeta = 0.9$). For MICRA we choose $\eta = 50$.

Perceptron			agg-ROMMA			MICRA		
$\frac{b}{\eta R^2}$	$\gamma'_{\rm d}$	upds	δ	$\gamma'_{\rm d}$	upds	$100\frac{\beta}{R}$	$\gamma'_{\rm d}$	upds
0.7	0.00516	189,313	0.5	0.00506	210,228	0.240	0.00524	104,926
1.02	0.00585	251,534	0.4	0.00584	307,344	0.280	0.00590	140,634
1.76	0.00656	396,318	0.3	0.00656	466,874	0.320	0.00663	200,517
3.9	0.00727	820,261	0.2	0.00728	778,412	0.359	0.00729	327,469
30	0.00785	5,930,214	0.1	0.00785	1,546,595	0.404	0.00786	706,275
100	0.00791	19,599,882	0.05	0.00819	2,716,711	0.443	0.00819	1,932,166
500	0.00793	97,717,549	0.01	0.00837	14,079,715	0.495	0.00837	11,610,900

Table 2. The number of updates (upds) required to achieve $\gamma'_{\rm d} \simeq 0.00819$ in the sonar data set with MICRA and ALMA$_2$. For MICRA various ϵ, ζ values are considered and the η employed is given.

ϵ, ζ	0.005, 0.99	0.05, 0.9	0.1, 0.8	0.15, 0.7	0.2, 0.6	0.2, 0.5	0.5, 0.5	ALMA$_2$
η	190	60	17	4.4	1.2	0.28	0.35	
upds/10^6	1.53	1.86	2.32	2.89	3.57	3.74	7.54	53.4

origin in the additional dimension leading to $R \simeq 3.8121$ and $\gamma_{\rm d} \simeq 0.00841$. The results of our comparative study of the Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.05, \zeta = 0.9$) algorithms are presented in Table 1. We observe that MICRA is certainly the fastest. Moreover, the data suggest that the Perceptron algorithm is not able to approach the maximum margin arbitrarily close. We also present in Table 2 the number of updates required to achieve a margin $\gamma'_{\rm d} \simeq 0.00819$ using MICRA with several ϵ, ζ values and ALMA$_2$. For ALMA$_2$ the accuracy parameter α was set to $\alpha = 0.1527$ with the remaining parameters chosen to correspond to the ones of the Theorem in [4] if the data are normalised such that the longest pattern has unit length. From Table 2 it becomes clear that small ϵ's combined with relatively large ζ's lead to faster convergence.

We additionally analyse a linearly separable data set, which we call WBC$_{-11}$, consisting of 672 patterns each with 9 attributes. It is constructed from the Wisconsin Breast Cancer (WBC) data set obtainable from the UCI repository by first omitting the 16 patterns with missing features and subsequently removing from the data set containing the remaining 683 patterns the 11 patterns having the positions 2, 4, 191, 217, 227, 245, 252, 286, 307, 420 and 475. The value $\rho = 30$ is chosen for the parameter ρ of the augmented space leading to $R = \sqrt{1716}$ and $\gamma_{\rm d} \simeq 0.0243$. In Table 3 we present the results of a comparative study of the Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.1, \zeta = 0.8$) algorithms. The superiority of the performance of MICRA on this data set is remarkable.

Finally, we turn to an analysis of the linearly inseparable full WBC data set comprising 683 instances each with 9 attributes after ignoring the 16 instances

Table 3. Results for the WBC$_{-11}$ data set for the algorithms Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.1, \zeta = 0.8$). For MICRA the choice $\eta = 2.3$ is made.

Perceptron			agg-ROMMA			MICRA		
$\frac{b}{\eta R^2}$	γ'_{d}	upds	δ	γ'_{d}	upds	$100\frac{\beta}{R}$	γ'_{d}	upds
0.4	0.01650	1,401,984	0.5	0.01642	1,630,857	0.119	0.01658	55,319
0.65	0.01880	2,058,524	0.4	0.01848	2,382,480	0.144	0.01883	106,867
0.97	0.02033	2,894,811	0.3	0.02030	3,541,471	0.162	0.02035	154,077
1.8	0.02197	4,980,423	0.2	0.02195	5,784,868	0.185	0.02198	267,146
4.1	0.02321	10,761,773	0.1	0.02318	13,931,792	0.207	0.02324	467,370
8.5	0.02374	21,798,933	0.05	0.02373	31,156,487	0.222	0.02376	755,816
45	0.02415	113,406,210	0.01	0.02415	174,388,827	0.270	0.02415	4,533,156

Table 4. Results for the (extended) WBC data set (with $\mathit{\Delta} = 1$). The margin $\mathit{\Gamma}_{\mathit{\Delta}}$ and the number of updates (upds) are given for the Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.05, \zeta = 0.9$). For MICRA we choose $\eta = 20$.

Perceptron			agg-ROMMA			MICRA		
$\frac{b}{\eta R^2}$	$\mathit{\Gamma}_{\mathit{\Delta}}$	upds	δ	$\mathit{\Gamma}_{\mathit{\Delta}}$	upds	$100\frac{\beta}{R}$	$\mathit{\Gamma}_{\mathit{\Delta}}$	upds
0.22	0.07182	20,359	0.5	0.07259	13,983	0.380	0.07377	7,811
0.34	0.08338	27,706	0.4	0.08436	20,822	0.450	0.08540	11,584
0.64	0.09474	46,593	0.3	0.09591	33,728	0.520	0.09648	18,793
1.48	0.10815	95,245	0.2	0.10836	62,751	0.610	0.10869	40,825
3.5	0.11905	206,469	0.1	0.11916	169,588	0.702	0.11957	105,965
8.1	0.12462	457,334	0.05	0.12468	409,956	0.754	0.12470	183,644
700	0.12837	38,336,601	0.01	0.12928	1,554,492	0.840	0.12949	734,630

with missing attributes. Following [3] we make the data set linearly separable by extending the instance space by as many dimensions as the number of instances and placing each instance at a distance $\mathit{\Delta} = 1$ from the origin in the corresponding dimension. Then we attempt to obtain separating hyperplanes with as large a margin $\mathit{\Gamma}_{\mathit{\Delta}}$ as possible in the extended space relying on the observation that the hard margin optimisation task in the extended space is equivalent to the soft margin optimisation in the original space if the 2-norm of the slack variables is employed [11]. A more detailed analysis of this soft margin approach for Perceptron-like large margin classifiers is provided in [13]. Moreover, in order to incorporate some bias in the zero-threshold solution hyperplanes obtained we embed the data in an augmented space at a distance $\rho = 10$ from the origin in one additional dimension. This construction leads to $R = \sqrt{917}$ and to a maximum margin $\mathit{\Gamma}_{\mathit{\Delta}\mathrm{opt}} \simeq 0.13033$ with respect to zero-threshold hyperplanes in the extended (and augmented) space. In Table 4 we give the results of our comparative study of the Perceptron, agg-ROMMA and MICRA ($\epsilon = 0.05, \zeta = 0.9$) algorithms. We observe that the Perceptron shows again some difficulty in approaching the maximum margin and that once again MICRA is the fastest.

5 Conclusions

We presented MICRA, a family of Perceptron-like large margin classifiers completely independent of the length of the weight vector. Our theoretical approach proved sufficiently powerful in establishing asymptotic convergence to the optimal hyperplane for a whole class of such algorithms in which the misclassification condition and the effective learning rate are entirely controlled by rules involving arbitrary powers of the number of mistakes. Moreover, we provided experimental evidence in support of our theoretical analysis regarding convergence of our algorithms to the maximal margin hyperplane. The preliminary experimental results also suggest that algorithms belonging to the MICRA family with slow relaxation of the misclassification condition and relatively fast suppression of the effective learning rate with the number of mistakes are very powerful tools in the hands of a skillful practitioner. Of course, this does not diminish at all the value and usefulness of established fast and easy to use algorithms like agg-ROMMA or SMO which only need the choice of a single parameter determining the accuracy to which the optimal solution is approximated. It is remarkable, however, that simple extensions of the old Perceptron algorithm can be so competitive.

References

1. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines (2000) Cambridge, UK: Cambridge University Press
2. Duda, R.O., Hart, P.E.: Pattern Classsification and Scene Analysis (1973) Wiley
3. Freund, Y., Shapire, R. E.: Large margin classification using the perceptron algorithm. Machine Learning **37**(3) (1999) 277–296
4. Gentile, C.: A new approximate maximal margin classification algorithm. Journal of Machine Learning Research **2** (2001) 213–242
5. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: A fast iterative nearest point algorithm for support vector machine classifier design. Technical report, Indian Institute of Science, ISL-99-03 (1999)
6. Kowalczyk, A.: Maximal Margin Perceptron. Advances in Large Margin Classifiers (1999) MIT Press
7. Krauth, W., Mézard, M.: Learning algorithms with optimal stability in neural networks. Journal of Physics **A 20** (1987) L745–L752
8. Li, Y., Long, P.: The relaxed online maximum margin algorithm. Machine Learning **46** (2002) 361-387
9. Platt, J.C.: Sequential minimal optimization: A fast algorithm for training support vector machines. Technical Report MSR-TR-98-14, Microsoft Research (1998)
10. Rosenblatt, F.: The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review **65**(6) (1958) 386–408
11. Shawe-Taylor, J., Cristianini, N.: Further results on the margin distribution. In COLT'99 (1999) 278–285
12. Tsampouka, P., Shawe-Taylor, J.: Analysis of generic perceptron-like large margin classifiers. ECML 2005, LNAI **3720** (2005) 750–758, Springer-Verlag
13. Tsampouka, P., Shawe-Taylor, J.: Constant Rate Approximate Maximum Margin Algorithms. ECML 2006, LNAI **4212** (2006) 437–448, Springer-Verlag
14. Vapnik, V. N.: The Nature of Statistical Learning Theory (1995) Springer Verlag