Learning Nonlinearly Parametrized Decision Regions

Kim L. Blackmore* Robert C. Williamson*
Iven M. Y. Mareels*

June 21, 1995

Abstract

In this paper we present a deterministic analysis of an online scheme for learning very general classes of nonlinearly parametrized decision regions. The only input required is a sequence $((x_k, y_k))_{k \in \mathbb{Z}^+}$ of data samples, where $y_k = 1$ if x_k belongs to the decision region of interest, and $y_k = -1$ otherwise. Averaging results and Lyapunov theory are used to prove the stability of the scheme. In the course of this proof, conditions on both the parametrization and the sequence of input examples arise which are sufficient to guarantee convergence of the algorithm. A number of examples are presented, including the problem of learning an intersection of half spaces using only data samples.

Keywords: Online learning algorithm; nonlinear classifier; decision region; discriminant function; parametrization.

AMS Subject Classification: 68T05

1 Introduction

The problem of designing adaptive pattern classifiers has received a lot of attention recently, particularly in the neural networks literature. Whilst there are numerous examples of quite complex schemes that seem to work on some examples, there are few theoretical analyses of the convergence behaviour of these algorithms. Many of the algorithms that have been proposed (such as the "back-propagation" algorithm for neural networks) are gradient descent algorithms. To date there are still no theoretically compelling reasons for studying neural network parametrizations of decision regions over other schemes.

*Department of Engineering, Australian National University, Canberra ACT 0200, Australia

In this paper we provide a deterministic analysis of a gradient descent scheme for general classes of decision regions. The algorithm and corresponding analysis presented in this paper parallel the related problem of parameter estimation in nonlinear adaptive systems, though much additional complication is introduced by the binary nature of classification data. The algorithm we present (in section 5) is applicable to any class of decision boundaries which can be parametrized in a rather general nonlinear manner. The algorithm is a gradient descent based algorithm, chosen because it's simplicity makes analysis using dynamical systems theory possible. The analysis gives rise to conditions which guarantee that the algorithm will converge. These conditions impose constraints on the parametrization, and hence the decision regions, for which "learning" is possible with this algorithm.

A simple linear classifier (or perceptron) is one in which the two decision regions (in x) are given by $\operatorname{sgn}(w \cdot x - \theta)$, where $w, x \in \mathbb{R}^n$, and $\theta \in \mathbb{R}$. Gradient descent algorithms for such parametrized regions have been analyzed in [15, 20, 21]. Non-linear classifiers are more powerful (in a representational sense) but learning algorithms for them are rather harder to analyse. An old technique is to preprocess the inputs via a fixed nonlinearity (such as a power), and then perform linear classification [14, 17, 22]. However, this is still a *linearly parametrized* scheme.

More recently, Kuan and Hornik and White [10], Finnoff [5] and Leen and Moody [12] have performed analyses similar to that presented in this paper. The main difference is that we perform the analysis in a deterministic way (using averaging theory for ordinary differential equations), wheras they use stochastic methods due to Kushner [11] and others. On the other hand, Sontag and Sussmann [19] use ordinary differential equations to give a deterministic analysis of the back-propogation algorithm and Guo and Gelfand [7] provide a *quasi-linear* analysis of a certain class of nonlinearly parametrized classifiers.

In deterministic analysis of a gradient descent based algorithm for learning nonlinearly parametrized classifiers which is presented in this paper is new in several respects.

- It is for very general classes of nonlinear classifiers. The decision boundaries are defined in terms of $\operatorname{sgn} f(a, x)$, where a is a parameter vector, and $f(\cdot, \cdot)$ is a continuous function with certain properties;
- It gives conditions on the input examples (persistence of excitation) required for convergence to occur;
- It makes clear the value of a sigmoidal as opposed to a signum function in defining the classifier.
- It opens the way for a detailed noise and robustness analysis to be performed.

The rest of the paper is organised as follows: Section 3 contains definitions of online learning and approximate online learning, which formalise the problem of adaptive pattern classification. In section 4 the concept of a parametrization for a class of decision regions is introduced. This provides a very general setting in which to pose the problem of learning nonlinear classifiers. Section 5 introduces an algorithm which addresses the learning problem whose properties are analysed in section 6. Section 6 contains our main result (theorem 6.3), which is a proof that the algorithm under consideration is indeed an (approximate) online learning algorithm. The proof entails relating the algorithm to a nonlinear ordinary differential equation (ODE) and showing that (under the technical conditions) the solution of this ODE converges asymptotically to the parameters of the true decision region. Two existing mathematical techniques (averaging [18] and Lyapunov stability [16]) are used extensively in the proof. In section 7 the technical conditions of our main result are discussed in some detail. Section 8 contains details of how to learn intersections of half spaces. This problem, addressed by certain neural networks, is acknowledged to be a hard problem [1]. It is achieved here by parametrizing the (approximate) intersection of two half spaces in a smooth manner. Section 9 concludes.

2 Notation and Known Results

In this section we list a number of the notations and results used in the rest of the paper.

For any vector x, $\|x\|$ denotes the infinity norm, and for any matrix y, $\|y\|$ denotes the induced matrix infinity norm.

Let x and y be $m \times m$ matrices. x is less than or equal to y ($x \leq y$) if $y - x$ is positive semi–definite.

For any set U of $\mathbb{R}^n$, $\overset{o}{U}$ denotes the interior of U and ∂U denotes the boundary of U. The diameter of U is given by $\operatorname{diam} U := \sup_{x,y \in U} \|x - y\|$.

For any function $f(a, x) : A \times X \to \mathbb{R}$, where $A \subset \mathbb{R}^m$ and $X \subset \mathbb{R}^n$, $\frac{\partial f}{\partial a}$ denotes the gradient of f with respect to the first argument, and $\frac{\partial^2 f}{\partial a^2}$ denotes the Hessian matrix, of f with respect to the first argument.

Definition 2.1 *Let $X \subset \mathbb{R}^n$. A sequence $(x_k)_{k \in \mathbb{Z}^+}$ of elements of X is a* covering *of X if, for any measurable function $f : X \to \mathbb{R}$,*

$$\lim_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} f(x_k) = \frac{1}{\operatorname{vol} X} \int_X f(x) dx, \tag{1}$$

where $\operatorname{vol} X := \int_X dx$ is the volume of X.

A covering in a deterministic framework is equivalent to a uniform distribution in a stochastic framework.

Definition 2.2 *Given a set A, P is a* generic property *of A if the subset of A which exhibits property P is open and dense in A.*

Definition 2.3 *Let $U \subset \mathbb{R}^n$ be open and $V_\mu \subset \mathbb{R}^n$ be open for all $\mu > 0$. We say that the sets V_μ* converge to *U as $\mu \to 0$ ($\lim_{\mu\to 0} V_\mu = U$) if $\cap_{\mu\to 0} V_\mu = U$ and for some $\mu_0 > 0$, $V_{\mu_2} \subset V_{\mu_1}$ whenever $\mu_2 < \mu_1 \leq \mu_0$.*

Definition 2.4 *A function $h : \mathbb{R}^+ \to \mathbb{R}$ is called an* order function *if $h(\varepsilon)$ is continuous and sign definite in $(0, \varepsilon_0]$ for some $\varepsilon_0 > 0$, and if $\lim_{\varepsilon\downarrow 0} h(\varepsilon)$ exists.*

Definition 2.5 *Let $h(\varepsilon)$ and $l(\varepsilon)$ be order functions. Then the notations $O_\varepsilon(l(\varepsilon))$, $o_\varepsilon(l(\varepsilon))$ and $\Omega_\varepsilon(l(\varepsilon))$ are defined by*

1. *$h(\varepsilon) = O_\varepsilon(l(\varepsilon))$ if there exists a constant K such that $|h(\varepsilon)| \leq K l(\varepsilon)$ on some nonempty set $(0, \varepsilon_1]$, some $\varepsilon_1 > 0$.*
2. *$h(\varepsilon) = o_\varepsilon(l(\varepsilon))$ if $\lim_{\varepsilon\downarrow 0} \frac{h(\varepsilon)}{l(\varepsilon)} = 0$.*
3. *$h(\varepsilon) = \Omega_\varepsilon(l(\varepsilon))$ if there exists a constant K such that $|h(\varepsilon)| > K l(\varepsilon)$ on some nonempty set $(0, \varepsilon_1]$, some $\varepsilon_1 > 0$.*

Consider the initial value problem

$$\dot{a} = F(a(t), x(t)) \qquad ; \qquad a(0) = a_0 \tag{2}$$

for $t \geq 0$; $a, a_0 \in A \subset \mathbb{R}^m$; $x \in X \subset \mathbb{R}^n$. Suppose that $a \equiv a^*$ is a solution of the equation.

Definition 2.6 *The solution $a \equiv a^*$ of (2) is* uniformly exponentially stable *in $N \subset A$ if there exists constants $k \geq 1, \eta > 0$ such that for all $t_0 \geq 0$ and all $a(t_0) \in N$,*

$$\|a(t) - a^*\| \leq k\|a(t_0) - a^*\| e^{-\eta(t-t_0)} \qquad \forall t \geq t_0. \tag{3}$$

Definition 2.7 *The solution $a \equiv a^*$ of the initial value problem (2) is* uniformly asymptotically stable *in $N \subset A$ if:*

1. *it is* uniformly stable*:*
   for all $\varepsilon > 0$ there exists a $\delta > 0$ such that for all $t_0 \geq 0$,
   $$\|a(t_0) - a^*\| \leq \delta \Rightarrow \|a(t) - a^*\| < \varepsilon \qquad \forall t \geq t_0. \tag{4}$$
2. *it is* uniformly attractive *in N:*
   for all $\delta > 0$ and $\varepsilon > 0$, there exists $\sigma > 0$ such that for all $t_0 \geq 0$ and $a(t_0) \in N$,
   $$\|a(t_0) - a^*\| < \delta \Rightarrow \|a(t) - a^*\| < \varepsilon \qquad \forall t \geq t_0 + \sigma. \tag{5}$$

N is the *basin of attraction* of a^*. If $N = A$ in either of these definitions, the stability is *global*. Uniform exponential stability of a solution in N implies uniform asymptotic stability of that solution in N.

In section 6, we make use of a result by Kreisselmeier [9], and a result from averaging in dynamical systems [18], both of which we state here without proof:

Lemma 2.8 (Kreisselmeier) *Let $p(t) \in \mathbb{R}^n$ be governed by*

$$\dot{p}(t) = -G(t)z(t)z^\top(t)p(t), \tag{6}$$

where, for all $t \geq 0$, $G(t)$ is a symmetric, positive definite matrix and $z(t) \in \mathbb{R}^n$. If there exist constants α, β, and T such that

$$0 < \alpha I \leq \frac{1}{T}\int_{t_0}^{t_0+T} z(t)z^\top(t)dt \leq \beta I \qquad \forall t_0 \geq 0 \tag{7}$$

then the solution $p(t) \equiv 0$ of (6) is uniformly globally exponentially stable.

Theorem 2.9 (Eckhaus/Sanchez – Palencia) *Let $A \subset \mathbb{R}^m$, $X \subset \mathbb{R}^n$, and let $F(a,x)$ with $F : A \times X \to A$ be Lipschitz continuous in a on A, and continuous in a and x on $A \times X$. Assume $x(s) : \mathbb{R}^+ \to X$ is such that*

$$F^0(a) := \lim_{T\to\infty} \frac{1}{T}\int_0^T F(a, x(s))ds \tag{8}$$

exists, and

$$\delta(\mu) = \sup_{t_0} \sup_{T\in[0,\frac{1}{\mu})} \sup_{a\in A} \left| F^0(a) - \mu \int_{t_0}^{t_0+T} F(a, x(s))ds \right| \tag{9}$$

exists and is an $o_\mu(1)$ function. Consider the initial value problems

$$\dot{a} \quad = \quad \mu F(a(t), x(t)) \qquad ; \qquad a(0) = a_0, \tag{10}$$

$$\tag{11}$$

and

$$\dot{a}_{av} \quad = \quad \mu F^0(a_{av}(t)) \qquad ; \qquad a_{av}(0) = a_0 \tag{12}$$

Suppose $a = a^*$ is an asymptotically stable critical point in the linear approximation about a^* to (12) with domain of attraction $A^0 \subset A$ and F^0 is continuously differentiable in A. If $a_0 \in A^0$, then

$$\|a(t) - a_{av}(t)\| = o_\mu(1) \qquad 0 \leq t < \infty. \tag{13}$$

Note that the assumption 9 has been forgotten in [18].

3 Online Learning

In this section the online learning problem is discussed and a formal definition is given. On the basis of this definition it is shown (in section 6) that the algorithm we present in section 5 is indeed an (approximate) online learning algorithm (under certain conditions).

We confine our attention to two class classifiers of points in some *sample space* $X \subset \mathbb{R}^n$. It is assumed that there is an (unknown) subset $\Sigma \subset X$, called the *decision region*, and points in X are classified according to their inclusion (or otherwise) in X. The classification is described by a binary valued function $y : X \to \{-1, 1\}$ called the *discriminant function* for Σ. The discriminant function satisfies

$$y(x) := \begin{cases} +1 & \text{if} \quad x \in \Sigma \\ -1 & \text{otherwise} \end{cases} \tag{14}$$

The object of learning is eventual correct classification of all points in X, that is, identification of the correct discriminant function. To this end, the learner receives a sequence $((x_k, y_k))_{k \in \mathbb{Z}^+}$ of data samples, where $x_k \in X$ and $y_k = y(x_k)$. The learning is effected by choosing an estimate discriminant function, σ_k, which is updated if the received data samples are misclassified by the current estimate. The learning is said to be online if an estimate discriminant function is calculated as each new data sample is received using only the present information, i.e. the present data sample, (x_k, y_k), and the information stored in a state variable, a_k. Online learning algorithms require finite memory, since at any iteration the only information stored is the (fixed size) state variable.

Definition 3.1 *Let $\Sigma \subset X \subset \mathbb{R}^n$ and let $((x_k, y_k))_{k \in \mathbb{Z}^+}$, be a sequence of data samples, where (x_k) is a covering of X, $y_k = y(x_k)$, and y is defined by (14). An* online learning algorithm *for Σ is an algorithm for choosing functions $\sigma_k : X \to \{-1, 1\}$, $(k \in \mathbb{Z}^+)$ so that the following hold:*

1. *$\sigma_k(x) = \Psi(a_k, x)$, for some function Ψ, where a_k is a state variable satisfying $a_{k+1} = \Phi(x_k, y_k, a_k)$, for some function Φ.*
2. *$\lim_{k \to \infty} \sigma_k(\cdot) \equiv y(\cdot)$*

At each iteration

$$\Sigma_k := \{x \in X | \sigma_k(x) = 1\} = \{x \in X | \Psi(a_k, x) = 1\} \tag{15}$$

is an estimate of the decision region. The second condition says that the estimate decision regions converge to the true decision region, in which case all points are correctly classified. Many algorithms using a fixed stepsize do not exhibit this property. Frequently, points sufficiently far from the decision boundary are eventually correctly classified, but points on or near the decision boundary may not be. The estimate decision boundary gets

close to the true decision boundary, but then jiggles around indefinitely in a small neighbourhood of the true decision boundary. If the neighbourhood V_μ where correct classification is never guaranteed disappears as the stepsize goes to zero, we call the algorithm an approximate online learning algorithm.

Definition 3.2 *Let $\Sigma \subset X \subset \mathbb{R}^n$ and let $((x_k, y_k))_{k\in\mathbb{Z}^+}$, be a sequence of data samples, where where (x_k) is a covering of X, $y_k = y(x_k)$, and y is defined by 14. An* approximate online learning algorithm *for Σ, with stepsize $\mu \in \mathbb{R}^+\backslash\{0\}$, is an algorithm for choosing functions $\sigma_{k,\mu} : X \to \{-1, 1\}$, $(k \in \mathbb{Z}^+)$ so that the following hold:*

1. *$\sigma_{k,\mu}(x_k) = \Psi_\mu(a_{k,\mu}, x_k)$, for some function Ψ_μ, where $a_{k,\mu}$ is a state variable satisfying $a_{k+1,\mu} = \Phi_\mu(x_k, y_k, a_{k,\mu})$, for some function Φ_μ.*

2. *For each $\mu > 0$, there exists an integer $K_\mu \geq 0$ and a neighbourhood V_μ of $\partial\Sigma$, $V_\mu \subset X$ such that for all $k \geq K_\mu$,*

$$\sigma_{k,\mu}(x) = y(x) \qquad \forall x \in X\backslash V_\mu. \tag{16}$$

3. *The neighbourhoods V_μ are such that*

$$\lim_{\mu\to 0} V_\mu = \partial\Sigma. \tag{17}$$

The perceptron algorithm [15] is an example of an approximate online learning algorithm for decision regions which are half spaces (linear classifiers). In the following, we present an approximate online learning algorithm for more general classes of decision regions.

At any time k, the current value of the state variable determines the current estimate of the decision region, so it may appear more natural to focus on the choice of the state variables. However, online learning is described as choosing discriminant functions rather than state variables because there are some subtle but important points which can be missed when the emphasis is placed on the state variables. We are primarily interested in correct classification, and hence convergence of the discriminant function. There may arise situations where (1) convergence of the state variables does not imply convergence of the discriminant functions; (2) no value of the state variable gives the true discriminant function, in which case it is meaningless to talk of convergence of the state variables, though the algorithm may still be an online learning algorithm; or (3) many values of the state variable give the true discriminant function, in which case there are many possible points which the state variables are allowed to converge to. In the following, the first situation is excluded by our definition of a parametrization as a smooth and locally bounded function. The second is excluded for purposes of analysis, and the third is ignored in section 6.1 but discussed in section 6.2.

4 The Parametrization

According to the definition of learning given in the last section, some structure is imposed on the estimate decision regions by the choice of the function $\Psi(\cdot,\cdot)$. In order to ensure that convergence of the algorithm is possible, we now impose similar structure on the true decision region Σ. In particular, we assume that Σ belongs to a known class, C, of decision regions, and that there is a parameter space A and some epimorphism (surjective homomorphism) $A \to C,\ a \mapsto \Sigma(a)$. Any parameter $a \in A$ identifies a unique decision region $\Sigma(a) \in C$. Moreover, we assume that there exists a continuous, nonlinear, real-valued function f called a parametrization of C (defined below). The parametrization is defined on $A \times X$, and is positive for all points in the sample space which are inside the decision region $\Sigma(a)$, and negative at all other points. Then if we choose $\Psi(\cdot,\cdot) = \text{sgn}\,(f(\cdot,\cdot))$, the parameter values can be identified with the state variables in definitions 3.1 and 3.2, and the estimate decision regions will be $\Sigma_k = \Sigma(a_k)$.

In applying this to a practical learning problem, two problems are encountered. The first is in choosing C, which amounts to assuming some knowledge about the decision region to be learnt. We do not address this problem here. The second is in choosing f, the parametrization for C. This is also difficult, as there may be many ways of parametrizing a class of decision regions, and not all of them will satisfy the conditions for convergence which we derive in section 6. In section 7, we discuss a number of different parametrizations for a single class of decision regions (half spaces) in the light of the conditions for convergence.

Definition 4.1 *A* parametrization *of C is a function $A \times X \to \mathbb{R},\ (a,x) \mapsto f(a,x)$, which satisfies*

1. *For all $a \in A$*

$$f(a,x) \begin{cases} > 0 & \text{if} \quad x \in \overset{o}{\Sigma}(a) \\ = 0 & \text{if} \quad x \in \partial\Sigma(a) \\ < 0 & \text{if} \quad x \notin \Sigma(a) \end{cases} \tag{18}$$

2. *(smoothness) $f(a,x)$ is continuous in a and twice continuously differentiable with respect to a on $A \times X$.*

3. *(local boundedness) f, $\frac{\partial f}{\partial a}$ and $\frac{\partial^2 f}{\partial a^2}$ are bounded in a compact domain, and f is Lipschitz continuous in x on a compact domain. An upper bound of these functions exists:*

   For all $a \in A$, $\|a\| \le d$, and all $x \in X$, $\|x\|, \|y\| \le r$,

$$|f(a,x)| \le B_0(d,r) < \infty \tag{19}$$

$$\left\| \left. \frac{\partial f}{\partial a} \right|_{(a,x)} \right\| \le B_1(d,r) < \infty \tag{20}$$

$$\left\| \left. \frac{\partial^2 f}{\partial a^2} \right|_{(a,x)} \right\| \leq B_2(d,r) < \infty \tag{21}$$

$$|f(a,x) - f(a,y)| \leq L(d,r)\|x - y\|. \tag{22}$$

Example Let C be the class of all circles in $\mathbb{R}^2$. Elements in C can be identified by specifying two centre coordinates and a radius. Thus the parameter space for this C is $A := \mathbb{R}^2 \times (0, \infty)$. The function

$$f(a,x) = a(3) - (a(1) - x(1))^2 - (a(2) - x(2))^2 \tag{23}$$

is a parametrization for C: using (18) and (23) it can be seen that for $a = (a(1), a(2), a(3)) \in A$, $\operatorname{sgn}(f(a,x))$ defines a circular decision region in $\mathbb{R}^2$.

Remark 4.1. Note that in the example both A and X are unbounded. Compactness (and hence boundedness) of A and X is assumed in the following to prove convergence of algorithm 5.1. Boundedness of X is a natural property of practical applications, but often boundedness of A is not. Remark 6.2 discusses the consequences of requiring that A is bounded. ∎

Remark 4.2. We have defined an approximate online learning algorithm as one which eventually classifies all of the points in X correctly, (i.e. one whose discriminant function converges to the true discriminant function), except in some neighbourhood of the boundary of the true decision region, and that this neighbourhood converges to the true decision boundary in the limit $\mu \to 0$. This is what is desired in practice. For decision regions described by a known parametrization, there exists at least one parameter a^* which identifies the true decision region. The smoothness and local boundedness properties of the the parametrization mean that this convergence will be guaranteed if the estimate parameters asymptotically approach some neighbourhood of the parameters of the true decision region, and that, in the limit $\mu \to 0$, this neighbourhood contains only the parameters of the true decision region. In section 6 we use this in showing that our algorithm is a learning algorithm.

Note that the map $a \mapsto \Sigma(a)$ is only assumed to be an epimorphism, so there may be more than one point in A which maps to $\Sigma(a)$. That is why we refer to the "parameters of the true decision region", rather than the "true parameters". ∎

5 The Algorithm as a Perturbation Problem

In this section we present the algorithm we shall analyse. We discuss the heuristic behaviour of the algorithm, and show that it is a perturbation of

a gradient descent algorithm.

Let $((x_k, y_k))_{k \in \mathbb{Z}^+}$ be a sequence of data samples for some unknown decision region Σ. Let $\Sigma = \Sigma(a^*)$ be a member of a class C of decision regions with parametrization f and parameter space A. Define

$$g_\varepsilon(a, x) := \frac{2}{\pi} \arctan\left(\frac{f(a, x)}{\varepsilon}\right). \tag{24}$$

The algorithm we propose is as follows:

Algorithm 5.1

Step 0: *Choose the stepsize :* $\mu \in \mathbb{R}^+ \backslash \{0\}$.
Choose a boundary sensitivity parameter: $\varepsilon \in \mathbb{R}^+ \backslash \{0\}$.
Choose an initial parameter value: $a_{0,\mu} \in A$.

Step 1: *Commencing at* $k = 0$, *iterate the recursion below:*

$$a_{k+1,\mu} = a_{k,\mu} - \mu \left.\frac{\partial f}{\partial a}\right|_{(a_{k,\mu}, x_k)} (g_\varepsilon(a_{k,\mu}, x_k) - y_k)\,. \tag{25}$$

The function

$$\sigma_{k,\mu}(x) := \begin{cases} \operatorname{sgn}(f(a_{k,\mu}, x)) & \text{if } f(a_{k,\mu}, x) \neq 0 \\ 1 & \text{if } f(a_{k,\mu}, x) = 0 \end{cases} \tag{26}$$

is the estimate discriminant function and $\Sigma(a_{k,\mu})$ is the estimate decision region at time k. The notation $\hat{\Sigma}_{k,\mu} := \Sigma(a_{k,\mu})$ is used in the following.

The purpose of this paper is to prove the following proposition (stated only informally here).

Proposition 5.2 *Assume* A, X *are compact and* $f : A \times X \to \mathbb{R}$ *is a parametrization for a class* C *of decision regions. If the parametrization and the sequence* (x_k) *of sample points satisfy appropriate persistence of excitation and uniqueness of parametrization conditions,* μ *and* ε *are sufficiently small, and* a_0 *is appropriately chosen, then algorithm 5.1 is an approximate online learning algorithm for any decision region in* C.

This proposition is stated formally in theorems 6.3 and 6.4.

Remark 5.1. Observe that $\lim_{\varepsilon \downarrow 0} \frac{2}{\pi} \arctan\left(\frac{z}{\varepsilon}\right) = \operatorname{sgn}(z)$. Thus in the limit, the term $(g_\varepsilon(a_{k,\mu}, x_k) - y_k)$ in (25) is zero if x_k is correctly classified by the estimate discriminant function, but ± 2 otherwise. We call this the misclassification error. In the limit, the parameters update only if the misclassification error is nonzero. So, for sample points x_k not contained in $\partial\hat{\Sigma}_{k,\mu}$ the algorithm makes an update at the time step k only if x_k is misclassified by $\sigma_{k,\mu}$. If x_k is in $\Sigma \backslash \hat{\Sigma}_{k,\mu}$, the parametrization moves a distance 2μ times the magnitude of the gradient in the direction of steepest

ascent of f in parameter space, so that $\hat{\Sigma}_{k,\mu}$ "grows". If x_k is contained in $\hat{\Sigma}_{k,\mu} \backslash \Sigma$ the parametrization moves in the direction of steepest descent, so that $\hat{\Sigma}_{k,\mu}$ "shrinks". Test points in $\partial\hat{\Sigma}_{k,\mu}$ cause updates of half this size, with $\hat{\Sigma}_{k,\mu}$ growing if $x_k \in \Sigma$, and shrinking otherwise. ∎

Remark 5.2. For non–zero ε, and any z, let

$$\delta_\varepsilon(z) := \operatorname{sgn}(z) - \frac{2}{\pi} \arctan\left(\frac{z}{\varepsilon}\right). \tag{27}$$

Then there exist order functions $\alpha(\varepsilon) = o_\varepsilon(1)$, and $\beta(\varepsilon) = o_\varepsilon(1)$ such that, for each ε,

$$\begin{array}{rcll} |\delta_\varepsilon(z)| & \leq & \alpha(\varepsilon) & \forall |z| \geq \beta(\varepsilon) \\ |\delta_\varepsilon(z)| & < & 1 & \forall |z| < \beta(\varepsilon). \end{array}$$

Specifically, if $\varepsilon < \frac{1}{4}$ then

$$\begin{array}{rcll} |\delta_\varepsilon(z)| & \leq & \varepsilon^{\frac{1}{2}} & \forall |z| \geq \varepsilon^{\frac{1}{2}} \\ |\delta_\varepsilon(z)| & < & 1 & \forall |z| < \varepsilon^{\frac{1}{2}}. \end{array}$$

Because f is a Lipschitz continuous function of x, there exists an open neighbourhood, $U_{\varepsilon,a}$, of $\partial\Sigma_a$ for which

$$|f(a,x)| < \varepsilon^{\frac{1}{2}} \qquad \forall x \in U_{\varepsilon,a}. \tag{28}$$

Recalling that $f(a,x) = 0$ iff $x \in \partial\Sigma(a)$, it can be seen that $\lim_{\varepsilon \to 0} U_{\varepsilon,a} = \partial\Sigma(a)$.

At each iteration of equation 25 a neighbourhood $U_{\varepsilon,a_{k,\mu}}$ satisfying (28) can be found. For sample points outside this neighbourhood the algorithm behaves, to order ε, as described above. For points inside the neighbourhood, the algorithm makes updates in the same direction as above, but the update size is smaller. Thus test points close to the boundary of $\hat{\Sigma}_{k,\mu}$ region are given less weighting. This increases robustness of the algorithm in the presence of measurement noise in the sample points.

The function $\frac{2}{\pi}\arctan(\frac{z}{\varepsilon})$ is a sigmoidal squashing function. Other functions such as $\tanh(\frac{z}{\varepsilon})$ exhibit similar behaviour. We have chosen to use the arctan squashing function in this paper because its derivative is rational in z and ε, so the bounds on a compact domain are elegant. ∎

Remark 5.3. It has been assumed that the target decision region belongs to C. Thus there exists some $a^* \in A$ such that

$$\begin{array}{rcll} y_k & = & \operatorname{sgn}(f(a^*, x_k)) & (29) \\ & = & g_\varepsilon(a^*, x_k) + \delta_\varepsilon(f(a^*, x_k)) & (30) \end{array}$$

for all (x_k, y_k). Writing $\delta_\varepsilon(f(a^*, x_k)) =: \delta_{k,\varepsilon}$, the discrete time equation (25) can thus be written as a perturbation problem:

$$a_{k+1} = a_k - \mu \left.\frac{\partial f}{\partial a}\right|_{(a_k, x_k)} (g_\varepsilon(a_k, x_k) - g_\varepsilon(a^*, x_k)) + \mu \left.\frac{\partial f}{\partial a}\right|_{(a_k, x_k)} \delta_{k,\varepsilon} \quad (31)$$

The subscript μ on the estimated parameter value a_k has been dropped to streamline notation. The value of a_k is nonetheless dependent on μ.

If f is a parametrization of some class C of decision regions then for any $\varepsilon > 0$, the "squashed" function, g, is also a parametrization for C. Comparing equations 19 to 22 with (24), the following bounds arise:

For all $a \in A$, $\|a\| \leq d$, and $x \in \mathbb{R}^n$, $\|x\|, \|y\| \leq r$,

$$\begin{aligned}
|g_\varepsilon(a, x)| &< 1 && (32)\\
\left\| \left.\frac{\partial g_\varepsilon}{\partial a}\right|_{(a,x)} \right\| &\leq \frac{2}{\pi}\frac{B_1(d,r)}{\varepsilon} && (33)\\
\left\| \left.\frac{\partial^2 g_\varepsilon}{\partial a^2}\right|_{(a,x)} \right\| &\leq \frac{2}{\pi}\left(\frac{3^{\frac{3}{2}} B_1(d,r)^2}{2^3\varepsilon^2} + \frac{B_2(d,r)}{\varepsilon}\right) && (34)\\
|g_\varepsilon(a, x) - g_\varepsilon(a, y)| &\leq \frac{2}{\pi}\frac{L(d,r)}{\varepsilon}\|x - y\|. && (35)
\end{aligned}$$

Equation 34 uses the fact that $\frac{z}{(\varepsilon^2+z^2)^2} \leq \frac{3^{3/2}}{2^4\varepsilon^3}$.

As in remark 5.2, for any value of ε there exists a neighbourhood U_{ε,a^*} of $\partial\Sigma$, the boundary of the decision region to be learned. If $x_k \notin U_{\varepsilon,a^*}$, $\delta_{k,\varepsilon} = O_\varepsilon(\varepsilon^{\frac{1}{2}})$. If $x_k \in U_{\varepsilon,a^*}$, $\delta_{k,\varepsilon} = O_\varepsilon(1)$. However if the input sequence, (x_k), is a covering of X then the average (over k) of the final terms is $O_\varepsilon(\varepsilon^{\frac{1}{2}})$, as shown below.

Let $\mathbb{I}_Y : X \to \{0, 1\}$ be the indicator function for the set $Y \subset X$ (i.e. $\mathbb{I}_Y(x) = 1$ if $x \in Y$ and 0 otherwise). Then $\int_X \mathbb{I}_Y(x)dx$ is the volume of Y (when it is defined). Referring to equation 28, the volume of $U_{\varepsilon,a}$ is $O_\varepsilon(\varepsilon^{\frac{1}{2}})$, since f is Lipschitz continuous in x in a compact domain . Thus, if (x_k) is a covering of X,

$$\begin{aligned}
\lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1} |\delta_{k,\varepsilon}| &\leq \lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1} \mathbb{I}_{U_{\varepsilon,a^*}}(x_k) + \varepsilon^{\frac{1}{2}} \lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1} \mathbb{I}_{X\setminus U_{\varepsilon,a^*}}(x_k)\\
&\leq \frac{\operatorname{vol} U_{\varepsilon,a^*}}{\operatorname{vol} X} + \varepsilon^{\frac{1}{2}}\frac{\operatorname{vol} X\setminus U_{\varepsilon,a^*}}{\operatorname{vol} X}\\
&= O_\varepsilon(\varepsilon^{\frac{1}{2}}). \qquad (36)
\end{aligned}$$

Thus, if a and x are confined to compact subsets of A and X respectively, then the sum of the perturbation terms in (31) over all iterations is bounded above by $\mu B_1(d,r) O_\varepsilon(\varepsilon^{\frac{1}{2}})$. In the initial stages of the following analysis, the perturbation term in equation 31 is omitted. It is reintroduced in the final stage (A.28). ∎

Remark 5.4. Ignoring the perturbation, the misclassification error at time k is $(g_\varepsilon(a_k, x_k) - g_\varepsilon(a^*, x_k))$. In the course of the analysis, we relate the behaviour of algorithm 5.1 to the behaviour of

$$a'_{k+1} = a'_k - \mu \left.\frac{\partial f}{\partial a}\right|_{(a'_k, x_k)} (g_\varepsilon(a'_k, x_k) - g_\varepsilon(a^*, x_k)), \tag{37}$$

which is a stepwise gradient descent of the cost function

$$J_\varepsilon(a) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \left[f(a, x_k)(g_\varepsilon(a, x_k) - g_\varepsilon(a^*, x_k)) - \tfrac{\varepsilon}{\pi} \ln\left(1 + \tfrac{f(a,x_k)^2}{\varepsilon^2}\right)\right]. \tag{38}$$

Using bounds (19) and (32), it is clear that the cost function is bounded above and below on any compact domain: If $\|x_k\| \le r$ for all $k \in \mathbb{Z}^+$ then for all $a \in A$ such that $\|a\| \le d$,

$$2B_0(d, r) - \frac{\varepsilon}{\pi} \ln\left(1 + \frac{B_0(d, r)^2}{\varepsilon^2}\right) \le J_\varepsilon(a) \le 2B_0(d, r). \tag{39}$$

In the limit $\varepsilon \to 0$

$$J_0(a) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \left[|f(a, x_k)| - f(a, x_k)\mathrm{sgn}\,(f(a^*, x_k))\right], \tag{40}$$

which is nonnegative, since each term in the sum equals either $2|f(a, x_k)|$ or 0. If all points are correctly classified by the estimate decision region $\Sigma(a)$ (a is a true parameter value), then $J_0(a) = 0$, otherwise $J_0(a) > 0$. Thus J_0 attains its global minimum at the true parameter value and nowhere else. By continuity, this is also true for J_ε for sufficiently small ε. ∎

Remark 5.5. From remarks 5.3 and 5.2 it can be seen that, for non–zero μ and ε, the algorithm will never stop updating. Even if the estimate decision region equals the true decision region, i.e. $a_k = a^*$, $g_\varepsilon(a^*, x_k) \ne y_k$ for any value of x_k. The erroneous updates will rarely be large, since the value of $g_\varepsilon(a^*, x_k) - y(k)$ is only significant if $x_k \in U_{\varepsilon, a^*}$, and for small ε, U_{ε, a^*} is small. However for any values of ε and a_k, some small updates will always be made.

This erroneous updating can be avoided by using $\mathrm{sgn}\,(f(a_k, x_k))$ instead of $g_\varepsilon(a_k, x_k)$ in (25). This gives an algorithm which behaves similarly, though it is not robust to noise in the sample points around the decision boundary. However, the update term is then discontinuous in a_k, so it is more difficult to apply dynamical systems analysis to prove convergence of the algorithm. The nonlinearity cannot be regarded as a perturbation in the way that y_k was, because the neighbourhood U_{ε, a_k} where $\delta(f(a_k, x_k)) = O_\varepsilon(1)$ is not fixed with respect to k.

This particular problem does not appear in standard parameter estimation algorithms, where the data sequence is $(x_k, f(a^*, x_k))$ rather than (x_k, y_k). In that case introduction of the sigmoidal squashing function is unnecessary. ∎

6 Analysis of the Convergence Properties of the Algorithm

6.1 Unique True Parameter

Assume that a unique parameter $a^* \in A$ identifies the true decision region. An important case of this is when the mapping $a \mapsto \Sigma(a)$ is an isomorphism, so there is a unique parameter value identifying any decision region in C. This is the case for circles parametrized as in (23), and also for the half spaces we consider in section 7. However it is not the case for many interesting classes of decision regions, such as the intersections of halfspaces considered in section 8. In section 6.2 we relax this assumption.

In order to show convergence of algorithm 5.1 (in the sense of definition 3.2), we first investigate the stability properties of a related continuous time ordinary differential equation (ODE) using averaging and other techniques of dynamical systems analysis. We then show how this ODE relates to the difference equation 25, in order to derive conditions which guarantee that the estimate parameters derived by algorithm 5.1 asymptotically enter and remain in an $o_\mu(1)$ neighbourhood of the true parameters, and thus that algorithm 5.1 is an approximate online learning algorithm.

Consider the ODE

$$\dot{a}(t) = -\mu \left.\frac{\partial f}{\partial a}\right|_{(a(t),x(t))} (g(a(t), x(t)) - g(a^*, x(t))) \quad . \tag{41}$$

Simple inspection reveals that $a(t) \equiv a^*$ is a solution of (41) for any function g. The following theorem gives conditions which guarantee that this solution is globally uniformly asymptotically stable.

Theorem 6.1 *Let $A \subset \mathbb{R}^m$ and $X \subset \mathbb{R}^n$, X compact. Consider the initial value problem (41); $a(0) = a_0 \in A$, where $\mu \in \mathbb{R}^+\backslash\{0\}$, $t \in \mathbb{R}^+$, $x{:}\mathbb{R}^+ \to X$ is some known function, f is smooth and locally bounded and g is defined by (24). If $f(\cdot,\cdot)$ and $x(\cdot)$ are such that:*

A1. There exist positive constants α, β, and T such that

$$0 < \alpha I \leq \frac{1}{T}\int_{t_0}^{t_0+T} \left.\frac{\partial f}{\partial a}\right|_{(a^*,x(t))} \left(\left.\frac{\partial f}{\partial a}\right|_{(a^*,x(t))}\right)^{\mathsf{T}} dt \leq \beta I \quad \forall t_0 \geq 0; \tag{42}$$

A2. For all $a \in A$, the limit

$$\tilde{J}(a) := \lim_{T\to\infty} \frac{1}{T} \int_0^T \left[f(a,x(s))(g_\varepsilon(a,x(s)) - g_\varepsilon(a^*,x(s))) \right.$$
$$\left. -\frac{\varepsilon}{\pi} \ln\left(1 + \frac{f(a,x(s))^2}{\varepsilon^2}\right)\right] ds \tag{43}$$

exists;

A3. The bound

$$\sup_{t_0} \sup_{T\in[0,\frac{1}{\mu})} \sup_{a\in A} \left| \tilde{J}(a) - \mu \int_{t_0}^{t_0+T} \left[f(a,x(s))(g_\varepsilon(a,x(s)) - g_\varepsilon(a^*,x(s))) \right.\right.$$
$$\left.\left. -\frac{\varepsilon}{\pi} \ln\left(1 + \frac{f(a,x(s))^2}{\varepsilon^2}\right)\right] ds \right| \tag{44}$$

exists and is an $o_\mu(1)$ function;

A4.

$$\left.\frac{\partial \tilde{J}}{\partial a}\right|_a = 0 \qquad \text{only if} \quad a = a^*; \tag{45}$$

then for the given $x(\cdot)$, if μ is sufficiently small, the solution $a(t) = a^*$ of equation 41 is globally uniformly asymptotically stable.

Proof Using the Taylor series expansions of g and $\frac{\partial f}{\partial a}$ about a^*, (41) can be locally approximated by

$$\dot{a}_l = -\mu \left.\frac{\partial f}{\partial a}\right|_{(a^*,x_k)} \left(\left.\frac{\partial g_\varepsilon}{\partial a}\right|_{(a^*,x_k)}\right)^\top (a_l - a^*) \qquad ; \qquad a_l(0) = a_0. \tag{46}$$

The scalar quantity

$$\left.\frac{\partial g_\varepsilon}{\partial f}\right|_{(a^*,x_k)} = \frac{\varepsilon}{\varepsilon^2 + f(a^*,x_k)^2} \tag{47}$$

is always positive because f is locally bounded. Thus lemma 2.8 and assumption *A1* ensure that the solution $a_l \equiv a^*$ of (46) is uniformly globally exponentially stable for any value of $\mu \in \mathbb{R}^+\backslash\{0\}$.

The Poincaré-Lyapunov theorem [18] then indicates that the solution $a \equiv a^*$ of (41) is uniformly exponentially stable in some neighbourhood N of a^*. In the appendix (lemma A.1) we show that $\operatorname{diam} N = \Omega_\mu(1)$.

Equation 41 can be written

$$\dot{a} = -\frac{\mu}{2}\frac{\partial}{\partial a}\Bigg[f(a(t),x(t))(g_\varepsilon(a(t),x(t)) - g_\varepsilon(a^*,x(t))) - \frac{\varepsilon}{\pi}\ln\left(1+\frac{f(a(t),x(t))^2}{\varepsilon^2}\right)\Bigg] \quad ; \quad a(0)=a_0. \tag{48}$$

Averaging this equation over t yields

$$\dot{a}_{av} = -\frac{\mu}{2}\left.\frac{\partial \tilde{J}}{\partial a}\right|_{a_{av}} \quad ; \quad a_{av}(0)=a_0 \tag{49}$$

where $\tilde{J}(a)$ is defined by (43). Existence of $\tilde{J}$ is guaranteed by assumption *A2*.

Equation 49 is a gradient equation with unique critical point a^* (assumption *A4*). Thus a^* is the uniformly globally asymptotically stable solution of (49) [8]. Thus any trajectory that is a solution of (49) eventually becomes arbitrarily close to a^*.

Theorem 2.9 tells us that $a = a_{av} + o_\mu(1)$, so any trajectory which is a solution of (41) will come within $o_\mu(1)$ of a^*. But this means that, for μ sufficiently small, any trajectory eventually enters N. Once the trajectory enters N, it is exponentially attracted to a^*, so the solution $a \equiv a^*$ of (41) is uniformly globally asymptotically stable. ∎

Theorem 6.1 relies on the assumption that the average function $\tilde{J}$ has a unique critical point. Because a^* is always a critical point, this assumption implies that the surface described by $\tilde{J}$ has one global minimum, no non-global local minima, no saddle points or local maxima, and (49) has no attractors at infinity. If we choose $x(t) := x_k$ for all $t \in [k, k+1)$ then $\tilde{J}$ is the cost function J_ε defined in (38). So assumption *A4* refers to the topology of the level sets of the cost surface. This becomes assumption *B2* of our main result—theorem 6.3. Assumption *A1* is a persistence of excitation condition. It becomes assumption *B1* of theorem 6.3, and is discussed further in section 7. Assumption *A2* says that the average function $\tilde{J}$ exists, and *A3* is a requirement of smooth convergence to the average. Both *A2* and *A3* are satisfied under the assumptions of theorem 6.3.

In order to draw the connection between the difference equation (25) and the ODE (41), we first ignore the perturbation term in (31). So we consider the sequence a'_k generated by the difference equation 37. Linear interpolation of a'_k over unit time steps yields

$$\bar{a}(t) := a'_k + (a'_{k+1} - a'_k)(t-k) \qquad \forall t \in [k, k+1). \tag{50}$$

The function $\bar{a} : \mathbb{R}^+ \to \mathbb{R}^m$ is continuous and piecewise linear, and satisfies $\bar{a}(k) = a'_k$. Substituting from (37) and differentiating shows that $a(t)$ is

the solution of the ODE

$$\dot{\bar{a}}(t) = -\mu \left.\frac{\partial f}{\partial a}\right|_{(a'_k, x_k)} (g_\varepsilon(a'_k, x_k) - g_\varepsilon(a^*, x_k)) \qquad \forall t \in [k, k+1), \tag{51}$$

where $x(t) := x_k$ for all $t \in [k, k+1)$. This ODE resembles (41), but it is defined discontinuously in time. The solutions of (51) and (41) can be related according to the following theorem.

Theorem 6.2 *Let $A \subset \mathbb{R}^m$, $X \subset \mathbb{R}^n$. Assume $\Theta : A \times X \to \mathbb{R}^m$ is bounded and Lipschitz continuous in the first argument:*

$$\|\Theta(a, x)\| \leq B \qquad \forall a \in A,\ x \in X \tag{52}$$

$$\|\Theta(a, x) - \Theta(b, x)\| \leq L\|a - b\| \qquad \forall a, b \in A,\ x \in X. \tag{53}$$

Consider the initial value problems

$$\dot{\bar{a}}(t) = -\mu\Theta(\bar{a}(k), x_k) \quad \forall t \in [k, k+1) \quad ; \quad \bar{a}(0) = a_0 \tag{54}$$

$$\dot{a}(t) = -\mu\Theta(a(t), x_k) \quad \forall t \in [k, k+1) \quad ; \quad a(0) = a_0 \tag{55}$$

for some sequence (x_k) of points in X. If there exist constants $K \geq 1, \nu > 0$ such that

$$\|a(t) - a^*\| \leq Ke^{-\frac{\nu\mu}{2}t}\|a_0 - a^*\| \qquad \forall t \geq 0 \tag{56}$$

for some $a^* \in A$, then

$$\lim_{k\to\infty} \|\bar{a}(k) - a^*\| \leq \frac{4\mu K^{\frac{4L}{\nu}+1} LB}{\nu}. \tag{57}$$

Proof See appendix. ∎

Combining theorems 6.1 and 6.2, and reintroducing the perturbation term, we are able to state the main result of this paper.

Theorem 6.3 *Assume $A \subset \mathbb{R}^m$ and $X \subset \mathbb{R}^n$, are compact. Let $f : A \times X \to \mathbb{R}$ be a parametrization of a class $C = \{\Sigma(a) \subset X | a \in A\}$ of decision regions. If*

B1. There exist positive constants α and K such that

$$0 < \alpha I \leq \frac{1}{K} \sum_{k=k_0}^{k_0+K-1} \left.\frac{\partial f}{\partial a}\right|_{(a^*, x_k)} \left(\left.\frac{\partial f}{\partial a}\right|_{(a^*, x_k)} \right)^\top \qquad \forall k_0 \in \mathbb{Z}^+ \tag{58}$$

B2.

$$\left.\frac{\partial J_\varepsilon}{\partial a}\right|_a = 0 \qquad \text{only if} \quad a = a^*; \tag{59}$$

where J_ε is defined by (38) using (24)

and μ, ε are sufficiently small, then algorithm 5.1 is an approximate online learning algorithm for any decision region in C.

Proof See appendix. ∎

Remark 6.1. For an online learning algorithm it is sufficient to have convergence of the algorithm when the data points (x_k) cover X. In this case *B1* and *B2* impose restrictions on the nature of the parametrization alone. However more general $\{x_k\}$ can be considered, in which case *B1* and *B2* impose joint restrictions on the combination of data points and parametrization under which the algorithm will converge. These assumptions are discussed further in section 7. ∎

Remark 6.2. Theorem 6.3 assumes that A is compact. In practice, it often occurs that a parametrization that can be defined on a subset of $\mathbb{R}^m$ is most logically defined on a non-compact subset of $\mathbb{R}^m$, or on the whole of $\mathbb{R}^m$. Recall that the natural choice of A in the example of circles in $\mathbb{R}^2$ was $A = \mathbb{R}^2 \times (0, \infty)$. Even when C is restricted to circles in $\mathbb{R}^2$ which intersect some compact set $X \subset \mathbb{R}^2$, A must be unbounded in order to correctly parametrize all of the elements of C.

The assumption that A is compact is used in calculating bounds on the update size along the trajectories. This suggests that an alternative sufficient condition is that the solutions of (41) and (37) remain in some compact subset of A. Naturally this alternative condition is harder to test in general, but it follows naturally in theorem 6.3 from the assumptions that J_ε has a unique critical point and μ is "sufficiently" small.

The algorithm can be modified so that the estimated parameters are restricted to *any* convex compact set $A_c \subset A$, $a^* \in A_c$. This technique is well known to be compatible with gradient type algorithms, and is commonly used in adaptive control [6]. The restriction is performed by orthogonal projection of a_k to the boundary of A_c whenever it leaves A_c. Letting $\hat{a}_k$ denote the result of this projection, convexity of A_c implies that $\|\hat{a}_k - a^*\| \leq \|a_k - a^*\|$. Thus convergence of this modified algorithm follows easily from the above analysis.

∎

6.2 Multiple True Parameters

In this section the analysis of the previous section is generalized to include the possibility that multiple parameter values describe the true decision region. That is, the mapping $a \mapsto \Sigma(a)$ is an not an isomorphism but is still an epimorphism. However we still require that these "correct" parameter values be isolated from each other. Multiplicities of this type arise from

non-uniqueness in the parametrization, such as the obvious symmetry in the example of an intersection of half spaces which is given in the next section.

Assume there is some countable set of isolated points a^{*i}, i belonging to some index set I, for which identify the true decision region. We denote the true decision region by $\Sigma(a^*)$, where $a^* = a^{*i}$ for any $i \in I$. Now $y(\cdot) = \mathrm{sgn}\,(f(a^*, \cdot))$, so for any $x \in X$, $f(a^{*i}, x)$ has the same sign for all $i \in I$. We impose the stricter condition that $f(a^{*i}, x)$ has the same *magnitude* for all $i \in I$. Then $g_\varepsilon(a^{*i}, \cdot) \equiv g_\varepsilon(a^*, \cdot)$, so J_ε has a global miminum at a^{*i} for any $i \in I$, if ε is sufficiently small.

Now equation 49 is a gradient equation with isolated global minima, so each a^{*i} is a uniformly asymptotically stable solution of (49). Associated with each a^{*i}, is a basin of attraction, $A^{0i} \subset A$. If there are no non-global local minima of J_ε, and no attractors for (49) at the boundary of A, then $\cup_{i\in I} A^{0i}$ is open and dense in A. So for generic $a_0 \in A$, the solution of (49) is attracted to one of the a^{*i}. The non-generic case occurs when $a_0 \in \cup_{i\in I}\partial A^{0i}$, i.e. a_0 is on the boundary of one of the basins of attraction. In this case $a(t)$ remains in $\cup_{i\in I}\partial A^{0i}$ for all t. However the estimate parameters may leave the boundary, because the difference equation (25) is a perturbation of (49). Thus the estimate parameters may converge even though $a_0 \in \cup_{i\in I}\partial A^{0i}$.

If the persistence of excitation condition *A1* holds for each a^{*i}, then theorem 6.1 holds, so each a^{*i} is a uniformly asymptotically stable solution of (41). The averaging theorem (2.9) can be applied within each of the domains of attraction, provided the solution of (41) remains within the domain of attraction. Also, the linearization (51) can be effected at each of the a^{*i}, since $g_\varepsilon(a^{*i}, \cdot) \equiv g_\varepsilon(a^*, \cdot)$. From lemma A.1, it can be seen that the solution $a \equiv a^{*i}$ of (41) is uniformly exponentially stable within some neighbourhood N^i of a^{*i}, and $\mathrm{diam}\, N^i = \Omega_\mu(1)$. Thus if $a_0 \in A^{0i}$, the solution of (41) is either uniformly attracted to a^{*i} or it leaves A^{0i}. For any value of μ, neighbourhoods $\Delta^i_\mu \subset \partial A^{0i}$ can be constructed such that $\Delta^i_\mu \to A^{0i}$ as $\mu \to 0$ and a^{*i} is uniformly asymptotically stable in Δ^i_μ.

Theorem 6.2 can then be applied within each of the sets Δ^i_μ. This assumes that a'_k remains in Δ^i_μ for all k. For a given input sequence (x_k), we can construct new neighbourhoods $\Lambda^i_\mu \subset \Delta^i_\mu$ such that $\Lambda^i_\mu \to \Delta^i_\mu \to A^{0i}$ as $\mu \to 0$, and if $a_0 \in \Lambda^i_\mu$ then a'_k remains in Δ^i_μ for all $k \geq 0$. Now the difference between a_k and a'_k is bounded as $k \to \infty$ and is $o_\mu(1)$, so if the initial estimate a_0 is sufficiently far inside the basin of attraction of the true parameter, the estimate parameters asymptotically enter and remain in a neighbourhood of the parameters of the true decision region.

In theorem 6.3, there is only one true parameter, and it's basin of attraction is the whole parameter space. If there is a countable number of isolate global minima, there will be a countable number of (pairwise disjoint) basins of attraction within the parameter space. For generic $a_0 \in A$,

and any input sequence (x_k), there is a stepsize μ sufficiently small that the estimate parameters "converge" to one of the true parameters. This is summarised in the following theorem:

Theorem 6.4 *Assume $A \subset \mathbb{R}^m$ and $X \subset \mathbb{R}^n$, are compact. Let $f : A \times X \to \mathbb{R}$ be a parametrization of a class $C = \{\Sigma(a) \subset X | a \in A\}$ of decision regions. If*

C1. There is a countable set of isolated points a^{*i}, i in some index set I, for which $f(a^{*i}, \cdot) \equiv f(a^*, \cdot)$.

C2. For each i, there exist positive constants α^i and K^i such that

$$0 < \alpha^i I \leq \frac{1}{K^i} \sum_{k=k_0}^{k_0+K^i-1} \left.\frac{\partial f}{\partial a}\right|_{(a^{*i}, x_k)} \left(\left.\frac{\partial f}{\partial a}\right|_{(a^{*i}, x_k)} \right)^\top \quad \forall k_0 \in \mathbb{Z}^+ \tag{60}$$

then there exist basins of attraction $\Lambda^i_\mu \subset A$ such that

R1. Λ^i_μ are open for all $i \in I$

R2. Λ^i_μ are pairwise disjoint, i.e. for all $i, j \in I$, $i \neq j$, $\Lambda^i_\mu \cap \Lambda^j_\mu = \emptyset$

R3. $a^{*i} \in \Lambda^i_\mu$ for all $i \in I$

R4. If

$$\left.\frac{\partial J_\varepsilon}{\partial a}\right|_a = 0 \tag{61}$$

and $a \neq a^{*i}$ for any $i \in I$, where J_ε is defined by (38) using (24), then $a \notin \Lambda^i_\mu$ for all $i \in I$.

R5. If $a_0 \in \Lambda^i_\mu$ for some $i \in I$ and μ, ε are sufficiently small, then algorithm 5.1 is an approximate online learning algorithm for any decision region in C.

If, in addition,

C3. Local minima of J_ε occur only at the points a^{*i}.

C4. None of the solutions of (49) cross the boundary of A.

then

R6. $\lim_{\mu \to 0} \cup_{i \in I} \Lambda^i_\mu$ is dense in A.

Remark 6.3. Assumptions *C3* and *C4* are important, because without them there is no knowledge of the size of the set of suitable initial estimates. With these assumptions, we know that this set is (asymptotically as $\mu \to 0$) dense in the parameter space. So for almost any initial condition, the stepsize can be chosen small enough that the algorithm will converge (in the sense of definition 3.2). ∎

Remark 6.4. As mentioned in remark 6.2, it is often desirable to choose $A = \mathbb{R}^m$ for the parameter space. In this case *C4* says there are no attractors at infinity for equation 49. In fact assumptions *C3* and *C4* imply that for generic a_0 and sufficiently small μ, the solutions of (37) and (41) remain contained in a compact subset of A if a_0 is chosen from a compact subset of A. So again the assumption that A is compact can be ignored in practical applications. ∎

7 Assumptions of the Theory

In this section we illustrate the various assumptions of theorems 6.3 primarily by application to the class of linear classifiers. The learning algorithm we have developed can be applied to a much wider range of smoothly parametrized decision regions than is presented here. One such example is discussed in the following section. In this section, unless otherwise stated, we assume that $X \subset \mathbb{R}^2$, and C consists of half spaces which contain the origin, and whose intersection with X is not empty.

7.1 Relationship with Perceptron Algorithm

The algorithm we have developed can be applied successfully to the class of linear classifiers by letting $X \subset \mathbb{R}^n$, $A \subset \mathbb{R}^n$, and $f(a,x) = a^\top x + 1$. Letting $X = \mathbb{R}^n$ and $A = \mathbb{R}^n$, C is the set of all half-spaces in X which contain the origin, and the decision boundary for any element in C is a hyperplane with normal a and offset from the origin by $\frac{1}{\|a\|}$. The proof of convergence of the algorithm relies on the assumption that A is compact, so for any choice of A there is a non-zero minimum absolute offset of the decision boundary from the origin. If we wish to be able to learn half spaces whose boundaries pass through the origin, or half spaces which do not contain the origin, we must either use a different parametrization, or perform a translation of the coordinates in X, so that the origin becomes a regular point. This rather trivial application highlights the similarities between the algorithm we have presented and the classical perceptron learning procedure [15]. For any $a \in A$, $\left.\frac{\partial f}{\partial a}\right|_{(a,x)} = x$, so the algorithm update becomes

$$a_{k+1} = a_k - \mu x_k \left(\frac{2}{\pi} \arctan\left(\frac{a^\top x + 1}{\varepsilon} \right) - y_k \right). \tag{62}$$

In the limit $\varepsilon \to 0$, this reduces to the perceptron learning rule:

$$a_{k+1} = a_k + \begin{cases} 2\mu x_k y_k & \text{if } x_k \text{ is misclassified by } \Sigma(a_k) \\ 0 & \text{otherwise} \end{cases} \tag{63}$$

7.2 Assumption *C1*

Assumption *C1* deals only with the nature of the parametrization f. It prevents overparametrization, such as having a number of components in the parameter vector whose value doesn't affect the choice of Σ, or letting the parameter value corresponding to Σ be unique only up to multiplication by a scalar. For example, let $A \subset \mathbb{R}^3$, and parametrize the half spaces in $\mathbb{R}^2$ by either

$$f(a,x) \quad = \quad a(1)x(1) + a(2)x(2) + 1 \tag{64}$$

or

$$f(a,x) \quad = \quad a(1)x(1) + a(2)x(2) + a(3). \tag{65}$$

Both of these choices of f give smooth locally bounded parametrizations of C. In the first case, the third component of the parameter vector is ignored completely, and in the second case $f(a,x) \equiv f(ca,x)$ for any $c \in \mathbb{R}, c \neq 0$. For both there is a whole line in A for which $f(a,x) \equiv f(a^*,x)$. Simulations in both of these cases show that the algorithm still "learns" successfully, though there is a problem of noise accumulation once the parameters have converged to the correct line. Nevertheless, it is our belief that assumption *C1* is not necessary. In the future it would be desirable to ascertain how far this assumption can be generalised, by finding a corresponding necessary condition.

7.3 Assumptions *B1* and *C2*

Assumptions *B1* and *C2* are generalizations of the *persistence of excitation* condition that is commonly imposed in problems of adaptive control [6]. The sum

$$\sum_{k=k_0}^{k_0+K-1} \left.\frac{\partial f}{\partial a}\right|_{(a^*,x_k)} \left(\left.\frac{\partial f}{\partial a}\right|_{(a^*,x_k)}\right)^\top \tag{66}$$

parallels the *information matrix* in the adaptive control context. Assumption *B1* says that for some constants α, K,

$$\alpha\|a\|^2 \leq \frac{1}{K} \sum_{k=k_0}^{k_0+K-1} \left(a^\top \left.\frac{\partial f}{\partial a}\right|_{(a^*,x_k)}\right)^2 \tag{67}$$

for all $a \in \mathbb{R}^m$. This is satisfied if (and only if) for some K, any K successive vectors $\left.\frac{\partial f}{\partial a}\right|_{(a^*,x_k)}$, $k = k_0 \ldots k_0 + K - 1$, span X.

If a linear parametrization is used, *B1* is satisfied if and only if any K successive sample points span X. For example, let $A \subset \mathbb{R}^2$, and let f be defined by (64). If $x_k(1) = 0$ for all k then no updates of $a(1)$ will be made.

If a nonlinear parametrization is used, the condition is more complicated. Assumption *B1* may be violated due to an inappropriate choice of parametrization, such as having one component of the parameter vector appearing raised to an even power. For example let $A \subset \mathbb{R}^2$ and $f(a,x) = a(1)^2 x(1) + a(2)x(2) + 1$. Then $(a(1), a(2))$ and $(-a(1), a(2))$ both describe the same decision region. If we now choose $a^* = (0,1)^\top$ then the first component of $\frac{\partial f}{\partial a}\Big|_{(a^*, x_k)}$ is zero for any x_k.

If the persistence of excitation condition is not satisfied the algorithm will learn to correctly classify points in the subset of X which is spanned by the vectors $\frac{\partial f}{\partial a}\Big|_{(a^*, x_k)}$.

7.4 Assumptions *B2*, *C3* and *C4*

Assumptions *B2*, *C3* and *C4* deal with the topology of the level sets of the cost surface defined by J_ε. Assumption *B2* is violated if there exists some $a \in A$ such that

$$\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \frac{\partial f}{\partial a}\Big|_{(a, x_k)} (g_\varepsilon(a, x_k) - g_\varepsilon(a^*, x_k)) = 0 \tag{68}$$

and $a \neq a^*$. This can occur if all of the terms in the sum equal zero, or if only a finite number of the terms in the sum are nonzero.

For example, let $A \subset \mathbb{R}^2$, and let f be defined by (64). Choose the sample points so that $x_k(1) = \frac{1}{2} x_k(2)$ for all k. Then for any $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{R}$, $f((\alpha_1, \alpha_2), x) = f((\alpha_3, \alpha_2 + \frac{\alpha_1 - \alpha_3}{2}), x)$ if $x = x_k$ for some k, but *not* for all $x \in X$. Thus for any parameter satisfying $a = (\alpha, a^*(2) + \frac{a^*(1) - \alpha}{2})$, where $\alpha \in \mathbb{R}$, (68) holds, but $a \neq a^*$, so *B2* is violated. Again, the algorithm will learn to correctly classify points in the space spanned by the sample points, but not points in the rest of X. Whilst the "persistence of excitation" condition *B1* is not violated in this case, the sample points (x_k) still do not hold sufficient information that the algorithm can be guaranteed to learn the decision region correctly.

Assumption *B2* can also be violated when the terms in the sum (68) are nonzero but the sum is still zero. This may, for instance, occur when the sample points cycle through a finite set of j points chosen so that

$$\sum_{k=0}^{j} \frac{\partial f}{\partial a}\Big|_{(a, x_k)} (g_\varepsilon(a, x_k) - g_\varepsilon(a^*, x_k)) = 0. \tag{69}$$

We have as yet been unable to find such an example. Examples of this type are non-generic, however there is no reason to believe that such examples do not exist. Similar comments can be made about assumption *C3* and *C4* in the multiple solutions case. For a more thorough discussion of these assumptions see [3]. In particular, we show there that *B2* is satisfied for linear classifiers.

7.5 Smooth Locally Bounded Parametrization

Another basic assumption of this paper has been to assume that the parametrization to be used is smooth and locally bounded. However the algorithm has also been successfully applied in situations where this is not true. One such example is that of the (approximate) union of two circles. Choosing $X \subset \mathbb{R}^2$ and $A \subset \mathbb{R}^6$, the parametrization used was

$$\begin{aligned} f(a,x) &= \frac{a(3)}{(a(1)-x(1))^2+(a(2)-x(2))^2} \\ &\quad + \frac{a(6)}{(a(4)-x(1))^2+(a(5)-x(2))^2} - 1. \end{aligned} \tag{70}$$

This parametrization is unbounded at the points $x = (a(1), a(2))$ and $x = (a(5), a(6))$, and so is not locally bounded. Nevertheless, for uniformly distributed (x_k), the algorithm successfully learnt regions described by this parametrization in a number of experiments. This is because, for each a, the points where the parametrization is unbounded are isolated points in X, so the sample points will almost surely not coincide exactly with one of these isolated points. It appears that the assumption of smooth local boundedness can be relaxed somewhat, however we are unsure what would be the most useful (generally applicable) relaxation, or how to incorporate such a relaxation into the analysis.

8 Example—Intersections of Half Spaces

An interesting problem arising in the neural network literature is that of learning an intersection of half spaces. Whilst one can learn an intersection of half spaces using both examples (data samples) and queries [2], until recently no other online scheme had been developed which can solve this problem using only examples [4].

Let $X \subset \mathbb{R}^n$, $A \subset \mathbb{R}^{2n}$, and denote elements of A by $a = (n^1, n^2)$, where $n^1, n^2 \in \mathbb{R}^n$ are the normals to the boundaries of the two half spaces. Then define

$$f_p((n^1, n^2), x) = 1 - e^{-p(n^{1\top}x+1)} - e^{-p(n^{2\top}x+1)}. \tag{71}$$

For $p > 0$, the region $\Sigma((n_1, n_2))$ defined by $f_p((n^1, n^2), x) > 0$ is contained in the intersection of the half spaces $n^{1\top}x + 1 > 0$ and $n^{2\top}x + 1 > 0$ and as $p \to \infty$, $\partial\Sigma_{(n_1,n_2)}$ approaches the boundary of this intersection. This parametrization is constructed from the parametrization of a half space used in the previous section. It inherits the limitations of that parametrization, in that only those intersections of half spaces which contain the origin can be described by (71), and for a particular choice of A there is a nonzero minimum distance between the decision boundary and the origin in X. Figure 1 gives an example of the boundary $f_p(a, x) = 0$ approaching the boundary of the intersection of two half spaces as $p \to \infty$.

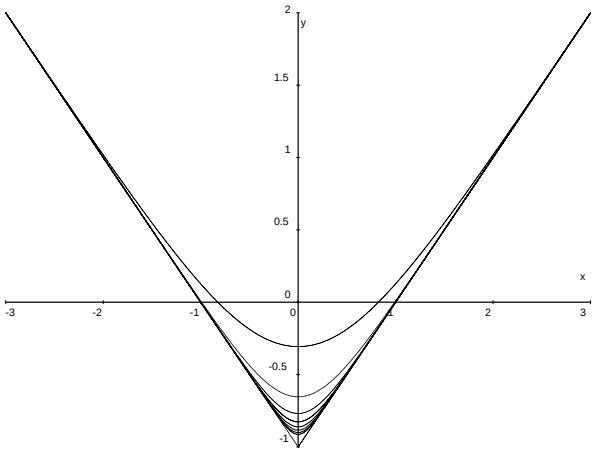

Figure 1: Example illustrating the difference between the intersection of two half planes and the solution of $f_p((n^1, n^2), x) = 0$. The normals to the half planes are $n^1 = (-1, 1)$ and $n^2 = (1, 1)$, and p ranges from 1 to 8 in steps of 1.

Presuming the true decision region is $\Sigma((n_1^*, n_2^*))$ for some $n_1^*, n_2^* \in \mathbb{R}^n$, assumption *C1* of theorem 6.3 is satisfied by the parametrization f_p. In particular, $f_p((n_1, n_2), \cdot) \equiv f_p((n_2, n_1), \cdot)$, so there are two critical points of the averaged equation. Assumption *C2* will be satisfied for generic input sequences (x_k). In [3] it is shown that *C4* is satisfied for this parametrization, and some further discussion of *C3* is given.

Figures 2 to 5 show the results of two different applications of the algorithm to (71) when $n = 2$. In both cases the final estimate is a good approximation of the true decision regin. In the first case the estimate parameters remain within one basin of attraction for all iterations. In the second case the estimate parameters jump from one basin to the other. Note that if the initial parameter estimate is chosen so that $n_1 = n_2$, both of the normals will update the same way, so the estimate parameters will not converge successfully but rather remain on the boundary between the two basins of attraction.

We have also successfully applied the algorithm to the obvious generalisation of the intersection on m half spaces in n dimensions. Again, the

algorithm performs well in practice. We have tried the following cases:

$$\begin{array}{ll} m = 2 & n = 4 \\ m = 3 & n = 2 \\ m = 3 & n = 3 \\ m = 5 & n = 3 \\ m = 5 & n = 5 \\ m = 10 & n = 10. \end{array}$$

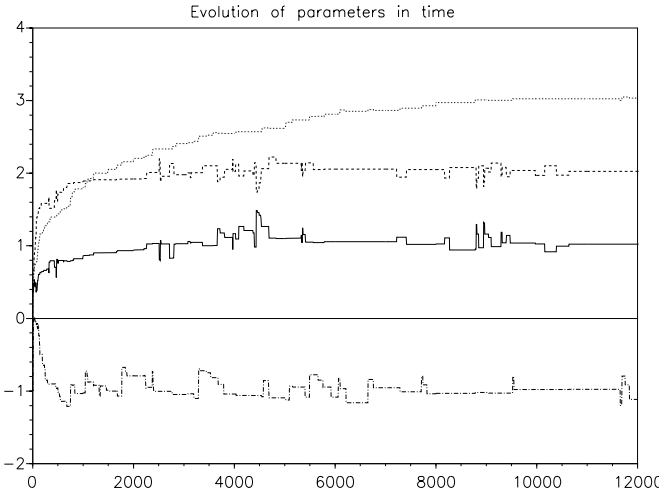

Figure 2: The evolution of the parameters when the algorithm was applied to (71). The parameters moved very quickly away from their initial values, then slowly converged toward the target. The true parameter vector was $a^* = (1, 2, 3, -1)$ and the initial and final estimates were $a_0 = (-1, 0, 0, -1)$ and $a_{12000} = (1.02, 2.03, 3.04, -1.12)$ respectively. The quantities μ, ε and p were 0.01, 0.00001 and 3 respectively. The sample points were independently uniformly distributed over the square $[-2, 2] \times [-2, 2]$.

A basic assumption of this paper is that the true decision region can be correctly parametrized by $f(a^*, \cdot)$ for some $a^* \in A$. In the case of intersecting half spaces, there is a unique decision region in C closest to the true decision region. The estimated parameters will asymptotically enter and remain in an open neighbourhood containing the parameters of the "best" decision region. However even in the limit $\mu \to 0$, and $\varepsilon \to 0$, the parameters will not converge exactly to the best parameter value since the estimate decision region can not exactly match the true decision region. Instead, the parameters will continue to move around in the neighbourhood

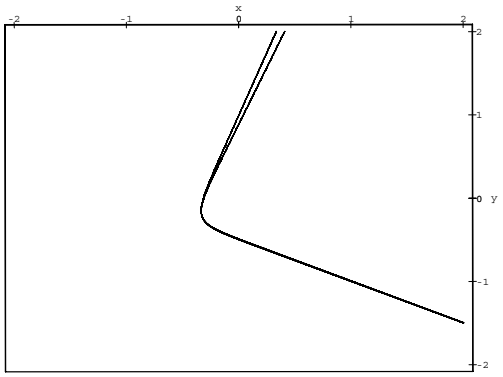

Figure 3: The target decision region and the final estimate for the problem described in figure 2.

of the best value. Nevertheless, p can be chosen so that the region in which the parameters jiggle is as small as required.

9 Conclusions

In this paper we have presented an algorithm for learning nonlinearly parametrized decision regions in an online fashion. The algorithm was defined in such a manner as to make an analysis of it's convergence properties possible. We have shown that, under certain conditions on the parametrization and the sequence of test points used in learning, the algorithm is an online learning algorithm. Standard techniques from averaging theory and Lyapunov stability were used to establish convergence of the algorithm. We have illustrated the power of the algorithm by applying it to the previously unsolved problem of learning an intersection of halfspaces using only examples.

A number of open questions arise from this work. Among them are:

- Are there conditions on the input sequence (x_k) which will force convergence of the estimate parameter values even when there are multiple critical points of the cost function J?
- It was mentioned in section 7 that the algorithm appears to be applicable even when the parametrization is not locally bounded. Possibly the smoothness condition can also be relaxed somewhat. It would be interesting to gain some theoretical insight into this relaxation.

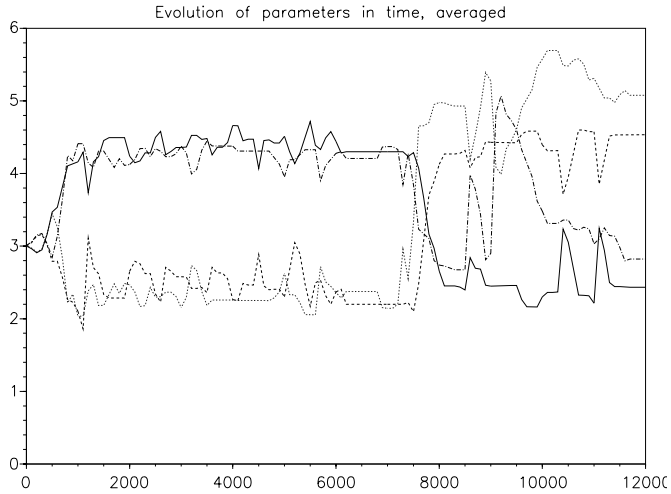

Figure 4: The evolution of the parameters when the algorithm was applied to (71). The parameters jumped between the two basins of attraction after about 7000 iterations. The true parameter vector was $a^* = (2, 4, 4, 2)$ and the initial and final estimates were $a_0 = (3, 3, 3.01, 3.01)$ and $a_{12000} = (2.43, 4.53, 5.08, 2.82)$ respectively. The quantities μ, ε and p were 0.025, 0.00001 and 3 respectively. The sample points were independently uniformly distributed over the square $[-2, 2] \times [-2, 2]$. For this value of μ the update size is large, so it is difficult to read a true plot of the evolution of the estimate parameters. For this reason, the average value over the previous 100 iterations has been plotted after each 100th iteration.

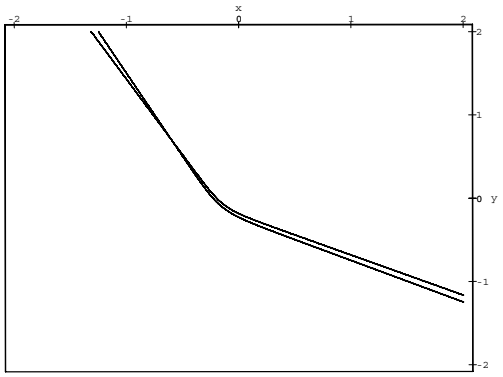

Figure 5: The target decision region and the final estimate for the problem described in figure 4.

- It was mentioned in section 8 that assumption *C1* appears not to be necessary. We would like to find a corresponding necessary condition.
- Various stochastic aspects of the problem can be investigated. For instance: "How much" does highly correlated (x_k) slow down convergence of the algorithm? "How robust" is the algorithm to model mismatch and classification error?
- By investigating the topological properties of the parametrizations, can an understanding of the types of decision regions which can be learned can be gained?

10 Acknowledgements

This work was supported by the Australian Research Council and the Australian Telecommunications and Electronics Research Board.

References

[1] E. Baum, "On Learning a Union of Half Spaces", *Journal of Complexity* **6**, 67–101 (1990).

[2] E. Baum, "Neural Net Algorithms That Learn in Polynomial Time from Examples and Queries" *IEEE Trans. Neural Networks* **2**, 5–19 (1991).

[3] K.L. Blackmore, R.C. Williamson, and I.M.Y. Mareels, "Local Minima and Attractors at Infinity in Gradient Descent Learning Algorithms", submitted to *J. Math. Systems, Estimation, and Control* (1993).

[4] J.A. Bucklew and W.A. Sethares, "The Covering Problem: Learning Decision Regions via Adaptive Algorithms", (submitted for publication).

[5] W. Finnoff, "Diffusion Approximations for the Constant Learning Rate Backpropogation Algorithm and Resistance to Local Minima" to appear in *Advances to Neural Information Processing* **5**, San Mateo: Morgan Kaufmann, 1993.

[6] G.C. Goodwin and K.S. Sin, *Adaptive Filtering Prediction and Control*, New York: Prentice-Hall, 1984.

[7] H. Guo and S.B. Gelfand, "Analysis of Gradient Descent Learning Algorithms for Multilayer Feedforward Neural Networks", *IEEE Trans. Circuits and Systems* **38**, 883–894 (1991).

[8] M.W. Hirsch and S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*, New York: Academic Press, 1974.

[9] G. Kreisselmeier, "Adaptive Observers with Exponential Rate of Convergence", *IEEE Trans. Auto. Control* **AC-22(1)**, 2–14 (1977).

[10] C.-M. Kuan and K. Hornik, " Convergence of Learning Algorithms with Constant Learning Rates", *IEEE Trans. Neural Networks* **2(5)**, 484–489 (1991).

[11] H.J. Kushner, *Approximation and Weak Convergence Methods for Random Processes*, Cambridge, MA:MIT Press, 1984.

[12] T.K. Leen and J. Moody, "Probability Densities and Equilibria in Stochastic Learning", to appear in *Advances in Neural Information Processing* **5**, San Mateo: Morgan Kaufmann, 1993.

[13] I.M.Y. Mareels and D.J. Hill, "Monotone Stability of Nonlinear Feedback Systems", *J. Math. Systems, Estimation, and Control* **2(3)**, 275–291 (1992).

[14] J.M. Mendel, "Synthesis of Quasi-Optimal Switching Surfaces by Means of Training Techniques", pp.163–194 in J.M. Mendel and K.S. Fu (Eds) *Adaptive Learning and Pattern Recognition Systems*, New York: Academic Press, 1970.

[15] M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*, Cambridge: MIT Press, 1969.

[16] N. Rouche, P. Habets and M. Laloy, *Stability Theory by Liapunov's Direct Method*, (Applied Mathematical Sciences; v. 22), New York: Springer-Verlag, 1985.

[17] N.J. Nilson, *Learning Machines*, New York: McGraw-Hill, 1965.

[18] J.A. Sanders and F. Verhulst, *Averaging Methods in Nonlinear Dynamical Systems*, (Applied Mathematical Sciences; v. 59), New York: Springer-Verlag, 1985.

[19] E.D. Sontag and H.J. Sussman, "Back Propagation Separates Where Perceptrons Do", *Neural Networks* **4**, 243–249, 1991.

[20] J.J. Shynk and S. Roy, "Convergence Properties and Stationary Points of a Perceptron Learning Algorithm", *Proceedings of the IEEE* **78**, 1599–1604 (1990).

[21] J.J. Shynk and N.J. Bershad, "Steady-State Analysis of a Single-Layer Perceptron Based on a System Identification Model with Bias Terms", *IEEE Trans. on Circuits and Systems* **38**, 1030–1042 (1991).

[22] J. Sklansky and G.N. Wassel, *Pattern Classifiers and Trainable Machines*, New York: Springer-Verlag, 1981.

Appendix

We make use of the Gronwall lemma [18], which we state here without proof:

Specific Gronwall Lemma *Suppose that for $t_0 \leq t \leq t_0 + T$ there are constants $\delta_1 > 0, \delta_2 \geq 0, \delta_3 \geq 0$ such that a continuous function $\phi(t)$ satisfies*

$$\phi(t) \leq \delta_2 (t - t_0) + \delta_1 \int_{t_0}^{t} \phi(s) ds + \delta_3 \tag{A.1}$$

and $\phi(t) \geq 0$ for $t_0 \leq t \leq t_0 + T$. Then

$$\phi(t) \leq \left(\frac{\delta_2}{\delta_1} + \delta_3 \right) e^{\delta_1 (t - t_0)} - \frac{\delta_2}{\delta_1} \tag{A.2}$$

for $t_0 \leq t \leq t_0 + T$.

Let

$$\begin{aligned} g^* &:= g(a^*, x(t)) \\ \frac{\partial f}{\partial a} &:= \left. \frac{\partial f}{\partial a} \right|_{(a(t), x(t))} \end{aligned}$$

$$\begin{aligned}\frac{\partial f^*}{\partial a} &:= \left.\frac{\partial f}{\partial a}\right|_{(a^*,x(t))} \\ \frac{\partial g^*}{\partial a} &:= \left.\frac{\partial g}{\partial a}\right|_{(a^*,x(t))}.\end{aligned}$$

Lemma A.1 *Let $A \subset \mathbb{R}^m$, $X \subset \mathbb{R}^n$, X bounded, let $f : A \times X \to (-1,1)$ be smooth and locally bounded, and let $g(\cdot,\cdot) = \frac{2}{\pi}\arctan\left(\frac{f(\cdot,\cdot)}{\varepsilon}\right)$ for some $\varepsilon > 0$. Consider the initial value problems*

$$\dot{a}(t) = -\mu\frac{\partial f}{\partial a}(g(a(t),x(t)) - g^*) \quad ; \quad a(t_0) = a_{t_0} \quad ; \quad t \geq t_0 \qquad \text{(A.3)}$$

$$\dot{b}(t) = -\mu\frac{\partial f^*}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (b(t) - a^*) \quad ; \quad b(t_0) = a_{t_0} \quad ; \quad t \geq t_0 \qquad \text{(A.4)}$$

for some $t_0 \geq 0$. If the solution $b \equiv a^*$ of (A.4) is uniformly globally asymptotically stable, then the solution $a \equiv a^*$ of (A.3) is uniformly exponentially stable in some neighbourhood N of a^*. Moreover, $\operatorname{diam} N = \Omega_\mu(1)$.

Proof Let the bounds on f be given by (19) to (22). Then the bounds on g are given by (32) to (35). Combining (A.3) and (A.4) we have

$$\dot{a}(t) - \dot{b}(t) = -\mu\frac{\partial f^*}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - b(t)) + h(a(t)), \qquad \text{(A.5)}$$

where

$$h(a(t)) := -\mu\frac{\partial f}{\partial a}(g(a(t),x(t)) - g^*) + \mu\frac{\partial f^*}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - a^*), \qquad \text{(A.6)}$$

It follows that

$$\begin{aligned}\|h(a(t))\| &= \Bigg\| -\mu\frac{\partial f}{\partial a}(g(a(t),x(t)) - g^*) + \mu\frac{\partial f}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - a^*) \\ &\qquad -\mu\frac{\partial f}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - a^*) + \mu\frac{\partial f^*}{\partial a}\left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - a^*)\Bigg\| \\ &\leq \mu \sup_{\substack{x\in X\\ a\in D}} \left\|\left.\frac{\partial f}{\partial a}\right|_{(a,x)}\right\| \left\| g(a(t),x(t)) - g^* - \left(\frac{\partial g^*}{\partial a}\right)^\top (a(t) - a^*)\right\| \\ &\qquad +\mu\left\|\frac{\partial f}{\partial a} - \frac{\partial f^*}{\partial a}\right\| \sup_{x\in X}\left\|\frac{\partial g^*}{\partial a}\right\| \|a(t) - a^*\| \qquad \forall a \in D.\end{aligned}$$

Thus, by the intermediate value theorem

$$\|h(a)\| \leq 2\mu C(d_1,R)\|a - a^*\|^2 \qquad \forall a \in D_1, \qquad \text{(A.7)}$$

where $R := \sup_{x \in X} \|x\|$, $D_1 := \{a \in A |\ \|a - a^*\| \leq d_1\}$, and

$$\begin{aligned} C(d,r) &= \sup_{\substack{\|a-a^*\|<d \\ \|x\|<r}} \left\| \left.\frac{\partial f}{\partial a}\right|_{(a,x)} \right\| \sup_{\substack{\|a-a^*\|<d \\ \|x\|<r}} \left\| \left.\frac{\partial^2 g}{\partial a^2}\right|_{(a,x)} \right\| \\ &\quad + \sup_{\substack{\|a-a^*\|<d \\ \|x\|<r}} \left\| \left.\frac{\partial g}{\partial a}\right|_{(a,x)} \right\| \sup_{\substack{\|a-a^*\|<d \\ \|x\|<r}} \left\| \left.\frac{\partial^2 f}{\partial a^2}\right|_{(a,x)} \right\| \\ &\leq \frac{2}{\pi}\left(\frac{3^{\frac{3}{2}} B_1(d+\|a^*\|,r)^3}{2^3 \varepsilon^2} + \frac{2B_1(d+\|a^*\|,r) B_2(d+\|a^*\|,r)}{\varepsilon} \right) \end{aligned} \tag{A.8}$$

according to equations 20, 21, 33 and 34.

Let Φ be the solution of the fundamental matrix equation

$$\dot{\Phi} = -\mu \frac{\partial f}{\partial a}^* \left(\frac{\partial g}{\partial a}^*\right)^\top \Phi \qquad ; \qquad \Phi(t_0) = I. \tag{A.9}$$

Global uniform asymptotic stability of a^* in (A.4) then implies that there exist constants $c \geq 1, \nu > 0$ such that

$$\|\Phi(t,t_0)\| \leq c e^{-\nu\mu(t-t_0)} \qquad \forall t \geq t_0 \tag{A.10}$$

and

$$\|b(t) - a^*\| \leq c\|a_{t_0} - a^*\| e^{-\nu\mu(t-t_0)} \qquad \forall t \geq t_0. \tag{A.11}$$

Variation of constants for (A.5) gives

$$\|a - b\| = \int_{t_0}^{t} \Phi(t,s) h(a(s)) ds. \tag{A.12}$$

Choose $d_1 > 0$. We now assume that

$$\|a_{t_0} - a^*\| \leq \frac{d_1}{2}. \tag{A.13}$$

Thus there exists a constant $t_1 > t_0$ such that

$$\|a(t) - a^*\| \leq d_1 \qquad \forall t \in [t_0, t_1]. \tag{A.14}$$

Combining (A.7), (A.10) and (A.12) gives

$$\begin{aligned} \|a(t) - b(t)\| &\leq 2\mu c C(d_1,R) \int_{t_0}^{t} \|a(s) - b(s)\|^2 e^{-\nu\mu(t-s)} ds \\ &\quad + 4\mu c C(d_1,R) \int_{t_0}^{t} \|a(s) - b(s)\| \|b(s) - a^*\| e^{-\nu\mu(t-s)} ds \\ &\quad + 2\mu c C(d_1,R) \int_{t_0}^{t} \|b(s) - a^*\|^2 e^{-\nu\mu(t-s)} ds \quad \forall t \in [t_0, t_1]. \end{aligned}$$

Equation A.11 then implies

$$\begin{aligned}\|a(t)-b(t)\| &\leq 2\mu cC(d_1,R)\int_{t_0}^{t}\|a(s)-b(s)\|(\|a(s)-b(s)\|e^{-\nu\mu(t-s)})ds \\ &\quad +4\mu c^2C(d_1,R)\|a_{t_0}-a^*\|e^{-\nu\mu(t-t_0)}\int_{t_0}^{t}\|a(s)-b(s)\|ds \\ &\quad +\frac{2c^3C(d_1,R)}{\nu}\|a_{t_0}-a^*\|^2e^{-\nu\mu(t-2t_0)}\int_{t_0}^{t}e^{-\nu\mu s}ds \quad \forall t\in[t_0,t_1].\end{aligned}$$

Because $a(t_0)=b(t_0)$, there exists a constant $t_2>t_0$ such that

$$\|a-b\|\leq\frac{\nu}{4cC(d_1,R)} \qquad \forall t\in[t_0,t_2]. \tag{A.15}$$

Let $\hat{t}=\min\{t_1,t_2\}$. Then

$$\begin{aligned}\|a(t)-b(t)\| &\leq \frac{\nu\mu}{2}\int_{t_0}^{t}\|a(s)-b(s)\|e^{-\nu\mu(t-s)}ds \\ &\quad +\nu\mu c\|a_{t_0}-a^*\|e^{-\nu\mu(t-t_0)}(t-t_0) \\ &\quad +\frac{2c^3C(d_1,R)}{\nu}\|a_{t_0}-a^*\|^2e^{-\nu\mu(t-t_0)} \qquad \forall t\in[t_0,\hat{t}].\end{aligned}$$

Applying Gronwall's lemma, this becomes

$$\begin{aligned}\|a(t)-b(t)\| &\leq \left(\tfrac{2c^3C(d_1,R)}{\nu}\|a_{t_0}-a^*\|^2+2c\|a_{t_0}-a^*\|\right)e^{-\frac{\nu\mu}{2}(t-t_0)} \\ &\quad -2c\|a_{t_0}-a^*\|e^{-\nu\mu(t-t_0)} \qquad \forall t\in[t_0,\hat{t}].\end{aligned} \tag{A.16}$$

Choosing $\|a_{t_0}-a^*\|\leq d_0:=\min\left\{\frac{d_1}{2},\frac{\nu}{10c^2C(d_1,R)}\right\}$, (A.16) implies that

$$\|a(t)-b(t)\|<\frac{\nu}{4cC(d_1,R)} \qquad \forall t\in[t_0,\hat{t}]. \tag{A.17}$$

It follows that $\hat{t}$ may be replaced by ∞ in (A.16).

Combining (A.11) with (A.16), we have

$$\begin{aligned}\|a(t)-a^*\| &\leq \left(\frac{2c^3C(d_1,R)}{\nu}\|a_{t_0}-a^*\|^2+2c\|a_{t_0}-a^*\|\right)e^{-\frac{\nu\mu}{2}(t-t_0)} \\ &\quad -c\|a_{t_0}-a^*\|e^{-\nu\mu(t-t_0)} \qquad \forall t\geq t_0.\end{aligned} \tag{A.18}$$

Recall that the only restriction on the choice of d_1 is that $a(t)$ is assumed to satisfy $\|a(t)-a^*\|\leq d_1$. Because the positive terms of (A.16) and (A.18) are the same, the upper bound (A.17) applies to $\|a(t)-a^*\|$. Thus d_1 can be replaced $\hat{d}_1$, the solution of

$$\hat{d}_1:=\frac{\nu}{4cC(\hat{d}_1,R)}. \tag{A.19}$$

This equation has a solution because ν and c are positive constants, and C is a positive, non-decreasing function of d_1. Now $\hat{d}_0 = \min\left\{\frac{\hat{d}_1}{2}, \frac{2\hat{d}_1}{5c}\right\} = \frac{2\hat{d}_1}{5c}$, since $c \geq 1$. Using this value of $\hat{d}_0$ to eliminate the squared $\|a_{t_0} - a^*\|$ term and ignoring the negative term in (A.18) gives

$$\|a(t) - a^*\| \leq \frac{11c}{5}\|a_{t_0} - a^*\| e^{-\frac{\nu\mu}{2}(t-t_0)} \qquad \forall t \geq t_0 \tag{A.20}$$

for any trajectory originating in $D_0 := \{a \in A | \, \|a - a^*\| \leq \hat{d}_0\}$. The region N of exponential attraction of $a(t)$ to a^* contains D_0, so $\operatorname{diam} N = \Omega_\mu(1)$. ∎

Proof of theorem 6.2 Let

$$V(a_0) := \sup_{t \geq 0}\left(\|a(t) - a^*\| e^{\frac{\nu\mu}{4}t}\right), \tag{A.21}$$

where $a(t)$ is the solution at time t of (55). Then $V: A \to \mathbb{R}$ is a Lyapunov function [16]. Furthermore, $V(a)$ satisfies

1. $\|a - a^*\| \leq V(a) \leq K\|a - a^*\| \qquad \forall a \in A$
2. $|V(a) - V(b)| \leq K^{\frac{4L}{\nu}+1}\|a - b\| \qquad \forall a, b \in A$
3. $\dot{V}_{(55)}(a) \leq -\frac{\nu\mu}{4}V(a) \qquad \forall a \in A,$

where the notation $\dot{V}_{(55)}(a)$ indicates the time derivative of $V(a)$ when a is a solution of (55). Derivation of these properties can be found in the appendix to Mareels and Hill [13].

The time derivative of V, for a a solution of (54), is

$$\begin{aligned}
\dot{V}_{(54)}(\bar{a}(t)) &= \limsup_{h \downarrow 0} \frac{V(\bar{a}(t) - \mu h \Theta(\bar{a}(k), x_k)) - V(\bar{a}(t))}{h} \\
&\text{where } k = \lfloor t \rfloor \\
&= \limsup_{h \downarrow 0} \frac{V(\bar{a}(t) - \mu h \Theta(\bar{a}(t), x_k)) - V(\bar{a}(t))}{h} \\
&\quad + \limsup_{h \downarrow 0} \frac{V(\bar{a}(t) - \mu h \Theta(\bar{a}(k), x_k)) - V(\bar{a}(t) - \mu h \Theta(\bar{a}(t), x_k))}{h} \\
&\leq -\frac{\nu\mu}{4}V(\bar{a}(t)) + \mu K^{\frac{4L}{\nu}+1}\|\Theta(\bar{a}(t), x_k) - \Theta(\bar{a}(k), x_k)\|
\end{aligned}$$

using properties 2 and 3 of $V(a)$. Thus

$$\begin{aligned}
\dot{V}_{(54)}(\bar{a}(t)) &\leq -\frac{\nu\mu}{4}V(\bar{a}(t)) + \mu K^{\frac{4L}{\nu}+1} L \|\bar{a}(t) - \bar{a}(k)\| \\
&\leq -\frac{\nu\mu}{4}V(\bar{a}(t)) + \mu^2 K^{\frac{4L}{\nu}+1} L B,
\end{aligned} \tag{A.22}$$

where we have used the bounds (52), (53) and the definition of $\bar{a}$. The second term in the bound (A.22) is independent of t. Property 1 of V and variation of constants on equation A.22 gives

$$\begin{aligned}\|\bar{a}(k+1) - a^*\| &\leq V(\bar{a}(k+1)) \\ &\leq V(a_0)e^{-\frac{\nu\mu}{4}(k+1)} + \mu^2 K^{\frac{4L}{\nu}+1} LB \int_0^{k+1} e^{-\frac{\nu\mu}{4}(k+1-t)} dt \\ &\leq K\|a_0 - a^*\| e^{-\frac{\nu\mu}{4}(k+1)} + \frac{4\mu}{\nu} K^{\frac{4L}{\nu}+1} LB(1 - e^{-\frac{\nu\mu}{4}(k+1)})\end{aligned}$$

Thus the long term behaviour of the solution to (54) is governed by (57). ∎

Proof of theorem 6.3 Using theorem 6.1, it is shown that the solution of (41) approaches a^*. We then use theorem 6.2 to show that the estimated parameters asymptotically enter and remain in a neighbourhood of a^*, and that the radius of this neighbourhood is $o_\mu(1)$. Following remark 4.2, this indicates that the algorithm is an approximate online learning algorithm.

Let $x(t) := x_k \ \forall t \in [k, k+1)$. Assumption *B1* gives the lower bound called for in assumption *A1* of theorem 6.1. The upper bound on this sum exists because both $\frac{\partial f}{\partial a}$ and $\frac{\partial g_\varepsilon}{\partial a}$ are bounded on $A \times X$. Also, $-1 < g < 1$ and f is bounded on $A \times X$ so assumption *A2* is satisfied, as is assumption *A3*. Assumption *B2* is identical to *A4*, so theorem 6.1 can be applied to equation 41. Thus the solution $a \equiv a^*$ of (41) is uniformly globally asymptotically stable.

For any $\varepsilon > 0$, setting

$$\Theta(a, x) := \left.\frac{\partial f}{\partial a}\right|_{(a,x)} (g_\varepsilon(a, x) - g_\varepsilon(a^*, x)) \tag{A.23}$$

allows application of theorem 6.2 to equations 51 and 41. Thus we have

$$\lim_{k\to\infty} \|a'(k) - a^*\| \leq \frac{4\mu LBK^{\frac{4L}{\nu}+1}}{\nu}, \tag{A.24}$$

where constants are determined as follows:

Let $d = \sup_A \|a\|$ and $R = \sup_X \|x\|$. Then equations 20, 21 and 32 to 35 show that $B = 2B_1(d, R)$ and $L = 2B_2(d, R) + \frac{2}{\pi}\frac{B_1(d,R)^2}{\varepsilon}$.

Existence of K and ν in (56) is guaranteed for sufficiently small μ by the uniform asymptotic stability of a^*. Expressions for these constants can be derived as follows from expressions in the proof of lemma A.1:

Let

$$\hat{d}_0 := \frac{\nu}{10c^2 C(\frac{5c\hat{d}_0}{2}, R)}, \tag{A.25}$$

where C is defined in (A.8). The constants c and ν depend on the matrix $\frac{\partial f}{\partial a}^* \left(\frac{\partial g_\varepsilon}{\partial a}^*\right)^\top$ only and are chosen as in equation A.10.

Choose T so that $\|b(T) - a^*\| \leq \hat{d}_0$ for all $a_0 \in A$. Existence of T is guaranteed because A is bounded. Using equation A.20 it can be seen that $\|b(t) - a^*\|$ satisfies the inequality

$$\|b(t) - a^*\| \leq K e^{-\frac{\nu\mu}{2}t} \|a_0 - a^*\| \qquad \forall t \geq 0, \tag{A.26}$$

where

$$K = \frac{11c}{5} e^{\frac{\nu\mu}{2}T} \sup_{a_0 \in A} \left\{ \frac{1}{\|a_0 - a^*\|} \sup_{0 \leq t \leq T} \|b(t) - a^*\| \right\}. \tag{A.27}$$

Comparing equations 31 and 37, we have that,

$$\begin{aligned} \limsup_{k \to \infty} \|a_k - a^*\| &\leq \|a'_k - a^*\| + \mu B_1(d, R) \lim_{k \to \infty} \sum_{j=0}^{k-1} |\delta_{k,\varepsilon}| \\ &\leq \frac{4\mu L B K^{\frac{4L}{\nu}+1}}{\nu} + \mu O_\varepsilon(\varepsilon^{\frac{1}{2}}), \end{aligned}$$

where (36) has been used.

Thus for each μ, the estimated parameter vector produced by the algorithm asymptotically approaches and remains in a neighbourhood of the parameters of the true decision region. Furthermore,

$$\limsup_{k \to \infty} \|a_k - a^*\| \leq o_\mu(1), \tag{A.28}$$

so the neighbourhood that the estimate parameters converge to converges to contain only the true parameter vector. By remark 4.2, the result follows. ∎