Analysis of Data with Threshold Decision Lists

Martin Anthony

December 2002
CDAM Research Report LSE-CDAM-2002-12

Abstract

We apply techniques from probabilistic learning theory to analyse theoretically the accuracy of data classification techniques that are based on the use of *threshold decision lists*.

1 Introduction

Suppose that we have been given some data points in $\mathbb{R}^n$, each classified as either *positive* (with an attached label of 1) or *negative* (labelled 0). The data points, together with the positive/negative classifications will be denoted D. An *extension* of D is a Boolean function f such that f agrees with D; that is, if x is one of the data points given in D then $f(x) = 1$ if and only if x is classified as positive in D. The aim is to find an extension of f which will, in a sense to be made precise, be a good 'generalization' of the data. By this we mean that we should like it to be the case that for most points that are not in D, the extension f classifies y correctly. We might also consider *partial extensions*, by which we mean functions that agree with a large proportion–though not necessarily all–of the classifications of the points in D.

There are clearly very many extensions of a given data set. We shall analyse the performance of methods based on the use of *threshold decision lists*. In doing so, we employ a probabilistic framework that has been used extensively in the modelling of machine learning; see the books [26, 27, 5, 4], for example.

2 Threshold decision lists

2.1 Decision lists

We start by describing *decision lists*, introduced by Rivest [22]. Suppose that K be any set of Boolean functions on $\{0,1\}^n$, for some fixed n. We shall usually suppose (for the sake of simplicity) that K contains the identically-1 function $\mathbb{T}$. A Boolean function f with the same domain as K is said to be a *decision list* based on K if it can be evaluated as follows. Given an example y, we first evaluate $f_1(y)$ for some fixed $f_1 \in K$. If $f_1(y) = 1$, we assign a fixed value c_1 (either 0 or 1) to $f(y)$; if not, we evaluate $f_2(y)$ for a fixed $f_2 \in K$, and if $f_2(y) = 1$ we set $f(y) = c_2$, otherwise we evaluate $f_3(y)$, and so on. If y fails to satisfy any f_i then $f(y)$ is given the default value 0. The evaluation of a decision list f can therefore be thought of as a sequence of `'if then else'` commands.

We define $DL(K)$, the class of *decision lists based on* K, to be the set of finite sequences

$$f = (f_1, c_1),\ (f_2, c_2), \ldots,\ (f_r, c_r),$$

such that $f_i \in K$ and $c_i \in \{0,1\}$ for $1 \leq i \leq r$. The values of f are defined by $f(y) = c_j$ where $j = \min\{i \mid f_i(y) = 1\}$, or 0 if there are no j such that $f_j(y) = 1$. We call each f_j a *test* (or, following Krause [16], a *query*) and the pair (f_j, c_j) a *term* of the decision list.

2.2 Threshold functions and threshold decision lists

A function $t : \mathbb{R}^n \to \{0,1\}$ is a *threshold function* if there are $w \in \mathbb{R}^n$ and $\theta \in \mathbb{R}$ such that

$$t(x) = \begin{cases} 1 & \text{if } \langle w, x\rangle \geq \theta \\ 0 & \text{if } \langle w, x\rangle < \theta, \end{cases}$$

where $\langle w, x\rangle$ is the standard inner product of w and x. Thus, $t(x) = \mathrm{sgn}(\langle w, x\rangle)$, where Given such w and θ, we say that t is represented by $[w, \theta]$ and we write $t \leftarrow [\alpha, \theta]$. The vector w is known as the *weight-vector*, and θ is known as the *threshold*.

We now consider the class of decision lists, in which the tests are threshold functions, and in which the domain is $\mathbb{R}^n$ rather than $\{0,1\}^n$. We shall call such decision lists *threshold decision*

lists, but they have also been called *neural* decision lists [19] and *linear* decision lists [25]. Formally, a threshold decision list

$$f = (f_1, c_1), (f_2, c_2), \ldots, (f_r, c_r)$$

has each $f_i : \mathbb{R}^n \to \{0, 1\}$ of the form $f_i(x) = \mathrm{sgn}(\langle w, x \rangle)$, where $\mathrm{sgn}(x) = 1$ if $x \geq 0$ and $\mathrm{sgn}(x) = 0$ if $x < 0$. The value of f on $y \in \mathbb{R}^n$ is $f(y) = c_j$ if $j = \min\{i \mid f_i(y) = 1\}$ exists, or 0 otherwise (that is, if there are no j such that $f_j(y) = 1$).

It is instructive to give a geometrical motivation for the use of threshold decision lists. Suppose we are given some data points in $\mathbb{R}^n$, each one of which is labelled 0 or 1. Of course, since there are very few threshold functions, it is unlikely that the positive and negative points can be separated by a hyperplane. But we can certainly use a hyperplane to separate off a set of points all having the same classification (either all are positive points or all are negative points). These points can then be removed from consideration and the procedure iterated until no points remain. This procedure is similar in nature to one of Jeroslow [15], but at each stage in his procedure, only positive examples may be 'chopped off' (not positive *or* negative). We give one example for illustration.

Example: Suppose the data set D consists of all points of $\{0, 1\}^n$, labelled according to their parity, so the classification is 1 precisely when the point has an odd number of ones. We first find a hyperplane such that all points on one side of the plane are either positive or negative. It is clear that all we can do at this first stage is chop off one of the points since the nearest neighbours of any given point have the opposite classification. Let us suppose that we decide to chop off the origin. We may take as the first hyperplane the plane with equation $y_1 + y_2 + \cdots + y_n = 1/2$. We then ignore the origin and consider the remaining points. We can next chop off all neighbours of the origin, all the points which have precisely one entry equal to 1. All of these are positive points and the hyperplane $y_1 + y_2 + \cdots + y_n = 3/2$ will separate them from the other points. These points are then deleted from consideration. We may continue in this manner. The procedure iterates n times, and at stage i in the procedure we 'chop off' all data points having precisely $(i - 1)$ ones, by using the hyperplane $y_1 + y_2 + \cdots + y_n = i - 1/2$, for example. (These hyperplanes are in fact all parallel, but this is not in general necessary.)

We may regard the chopping procedure as a means of constructing a threshold decision list extension of the data set. If, at stage i of the procedure, the hyperplane with equation $\sum_{i=1}^n \alpha_i y_i = \theta$ chops off positive (negative) points, and these lie on the side of the hyperplane with equation $\sum_{i=1}^n \alpha_i y_i > \theta$, then we take as the ith term of the threshold decision list the pair $(f_i, 1)$ (resp., $(f_i, 0)$), where $f_i \leftarrow [\alpha, \theta]$; otherwise take the ith term to be $(g_i, 1)$ (resp., $(g_i, 0)$), where

$g_i \leftarrow [-\alpha, -\theta]$. (We may assume that no point lies on any of the defining hyperplanes.)

If this construction is applied to the sequence of hyperplanes resulting from the Jeroslow method, a restricted form of decision list results—one in which all terms are of the form $(f_i, 1)$. But such a decision list is quite simply the *disjunction* $f_1 \vee f_2 \vee \ldots$, where $\vee$ means 'or'. For Boolean functions, the problem of decomposing a function into the disjunction of threshold functions has been considered by Hammer *et al.* [14] and Zuev [29]. Hammer *et al.* defined the *threshold number* of a Boolean function to be the minimum s such that f is a disjunction of s threshold functions, and they showed that there is an increasing function with threshold number $\binom{n}{n/2}/n$. (A function is increasing if, when $f(x) = 1$ and $x_i = 0$, then $f(x + e_i) = 1$ too.) Zuev showed that almost all increasing functions have this order of threshold number, and that almost all Boolean functions have a threshold number that is $\Omega(2^n/2)$ and $O(2^n \ln n/n)$.

The decision lists arising from the chopping procedure are more general than disjunctions of threshold functions and may provide a more compact representation of the data. (That is, since fewer hyperplanes might be used, the decision list could be smaller.) Indeed, Jeroslow's method requires 2^{n-1} iterations in the parity-based Example given above, since at each stage it can only 'chop off' one positive point. Note that Jeroslow's method [15] (described above) requires 2^{n-1} iterations in this Example, since at each stage it can only 'chop off' one positive point.

The chopping procedure described above suggests that the use of threshold decision lists is fairly natural, if one is to take an iterative approach to data classification. There are other methods which similarly make use of such an iterative approach, by classifying some points of the data set, removing these from consideration, and proceeding. Magasarian's multisurface method [18] also has this character. At each stage, it finds two parallel hyperplanes (as close together as possible) such that the points not enclosed between the two planes all have the same classification. It then removes these points and repeats. We can see that the MSM method may be regarded as constructing a decision list, where the base functions K are the indicator functions of the regions which are the complements of the regions lying between two parallel hyperplanes.

The chopping procedure as we have described it is in some ways merely a device to help us see that threshold decision lists have a fairly natural geometric interpretation. But the practicalities have been investigated by Marchand *et al.* [19, 20], who derive a greedy heuristic for constructing a sequence of 'chops'. This relies on an incremental heuristic for the NP-hard problem of finding at each stage a hyperplane that chops off as many remaining points as possible. Reports on the experimental performance of their method can be found in the papers cited.

2.3 Multilevel threshold functions

We noted in the Example given above that the hyperplanes of the resulting threshold decision list were parallel. By demanding that the hyperplanes are parallel, we obtain a special subclass of threshold decision lists, known as the *multilevel threshold functions*. These have been considered in a number of papers, such as [10, 21, 24], for instance.

We define an *s-level threshold function* f to be one that is representable by a threshold decision list of length at most s with the test hyperplanes parallel to each other. Any such function is defined by s parallel hyperplanes, which divide $\mathbb{R}^n$ into $s+1$ regions. The function assigns points in the same region the same value, either 0 or 1. Equivalently (following Bohossion and Bruck [10]), f is an s-level threshold function if there is a weight-vector $w = (w_1, w_2, \ldots, w_n)$ such that $f(x) = F\left(\sum_{i=1}^{n} w_i x_i\right)$, where the function $F : \mathbb{R} \to \{0, 1\}$ is piecewise constant with at most $s+1$ pieces. Without any loss, we may suppose that the classifications assigned to points in neighbouring regions are different (for, otherwise, at least one of the planes is redundant); thus, the classifications alternate as we traverse the regions in the direction of the normal vector common to the hyperplanes.

This method of classification is reasonably powerful. For example, Bohossian and Bruck observed that any Boolean function is a 2^n-level threshold function, an appropriate weight-vector being $w = (2^{n-1}, 2^{n-2}, \ldots, 2, 1)$. (For that reason, they paid particular attention to the question of whether a function can be computed by a multilevel threshold function where the number of levels is polynomial.)

3 Generalisation from random data

Recall that an extension of a labelled data set D is a function f agreeing with the classifications of the points in D, and that a partial extension is one agreeing with at least some proportion of the classification in D. If a particularly simple type of extension (or a good partial extension) to a fairly large data set can be found we might expect, given the success of this simple function in explaining the large data set, that this extension will perform well on 'most' unseen data. (This is, in some senses, an instance of the 'Occam's razor' principle: we trust a simple explanation of the data.) Issues such as these have been well-studied in 'computational learning theory' and 'statistical learning theory'. (See [26, 4], for instance.) To formalise the ideas somewhat, we

assume that the types of extension which can be produced all belong to a particular class, H, of functions, known as the *hypothesis space*. The choice of hypothesis space might reflect either our belief about the mechanism by which the data points are labelled (for example, by some deterministic *target concept* of a particular type) or our intention only to accept simple types of explanation of the data.

We shall apply some probabilistic techniques to analyse the performance of threshold decision list classification of random data. These methods have been used in learning theory (see [5, 26, 9]) and originated in the work of Vapnik and Chervonenkis [28]. Following a form of the PAC model of computational learning theory, we assume that the labelled data points (x, b) (where $x \in \mathbb{R}^n$ and $b \in \{0, 1\}$) have been generated randomly (perhaps from some larger corpus of data) according to a fixed probability distribution P on $Z = \mathbb{R}^n \times \{0, 1\}$. (Note that this includes as a special case the situation in which x is drawn according to a fixed distribution μ on $\mathbb{R}^n$ and the label b is then given by $b = t(x)$ where t is some fixed function.) Thus, if there are m data points in D, we may regard the data set D as a vector in Z^m, drawn randomly according to the product probability distribution P^m. (This suggests that we must attach some ordering to the points, and clearly there is some ambiguity as to how to do this, but this will not turn out to be a problem for the analysis of this paper.) Given any function $f \in H$, we measure how well f extends the data set D through its *sample error* $\mathrm{er}_D(f) = |D|^{-1}|\{(x, b) \in D : f(x) \neq b\}|$ (which is the proportion of points of D incorrectly classified by f) and we measure how well f performs on further examples by means of its *error*

$$\mathrm{er}(f) = P\left(\{(x, b) \in Z : f(x) \neq b\}\right),$$

the probability that a further randomly drawn labelled data point would be incorrectly classified by f.

What we would wish for is some guarantee that the sample error $\mathrm{er}_D(f)$ is a good approximation to the error $\mathrm{er}(f)$ for all f, so that an f with small sample error will likely have small error and therefore be a good model of the data labels. The following result provides such a guarantee for threshold decision lists and multilevel threshold functions of at most a bounded length s. (Thus, the number of terms is no more than s.)

Theorem 3.1 *Suppose that s and n are fixed positive integers and that D is a data set of m labelled points (x, b) of $Z = \mathbb{R}^n \times \{0, 1\}$, each generated at random according to a fixed probability distribution P on Z. Let δ be any positive number less than one. Then the following hold with probability at least $1 - \delta$:*

1. *If f is a threshold decision list with at most s terms, then the error $\mathrm{er}(f)$ of f and its sample error on D, $\mathrm{er}_D(f)$ are such that*

$$\mathrm{er}(f) < \mathrm{er}_D(f) + \sqrt{\frac{8}{m}\left(2s\ln 2 + ns\ln\left(\frac{e(2m-1)}{n}\right) + \ln\left(\frac{8}{\delta}\right)\right)},$$

   for $m > n$.

2. *If f is an s-level threshold function, then*

$$\mathrm{er}(f) < \mathrm{er}_D(f) + \sqrt{\frac{8}{m}\left((n+s-1)\ln\left(\frac{2ems}{n+s-1}\right) + \ln\left(\frac{8}{\delta}\right)\right)},$$

   for $m \geq n+s$.

If there is f that is an extension of D, with no sample errors—in particular, if the labels correspond to a threshold decision list of length at most s, or to an s-level threshold function—then the following tighter bounds can be used.

Theorem 3.2 *Suppose that s and n are fixed positive integers and that D is a data set of m labelled points (x, b) of $Z = \mathbb{R}^n \times \{0, 1\}$, each generated at random according to a fixed probability distribution P on Z. Let δ be any positive number less than one. Then the following hold with probability at least $1 - \delta$:*

1. *If f is a threshold decision list with at most s terms and f is an extension of D (so that $\mathrm{er}_D(f) = 0$), then*

$$\mathrm{er}(f) < \frac{4}{m}\left(2s\ln 2 + ns\ln\left(\frac{e(2m-1)}{n}\right) + \ln\left(\frac{4}{\delta}\right)\right)$$

   for $m < n$.

2. *If f is an s-level threshold function and f is an extension of D, then*

$$\mathrm{er}(f) < \frac{4}{m}\left((n+s-1)\ln\left(\frac{2ems}{n+s-1}\right) + \ln\left(\frac{4}{\delta}\right)\right).$$

The following variations of these results, in which s is not prescribed in advance, are perhaps more useful, since one does not necessarily know *a priori* how many terms a suitable threshold decision list will have.

Theorem 3.3 *Suppose that n is a fixed positive integer and that D is a data set of m labelled points (x, b) of $Z = \mathbb{R}^n \times \{0, 1\}$, each generated at random according to a fixed probability distribution P on Z. Let δ be any positive number less than one. Then the following holds with probability at least $1 - \delta$:*

1. *If f is a threshold decision list, then*

$$\mathrm{er}(f) < \mathrm{er}_D(f) + \sqrt{\frac{8}{m}\left(2s\ln 2 + ns\ln\left(\frac{e(2m-1)}{n}\right) + \ln\left(\frac{14s^2}{\delta}\right)\right)},$$

   for $m \geq n + s$, where s is the number of terms of f.

2. *If f is a multilevel threshold function, then*

$$\mathrm{er}(f) < \mathrm{er}_D(f) + \sqrt{\frac{8}{m}\left((n+s-1)\ln\left(\frac{2ems}{n+s-1}\right) + \ln\left(\frac{14s^2}{\delta}\right)\right)},$$

   for $m \geq n + s$, where s is the number of levels (planes) of f.

3. *If f is a threshold decision list and f is an extension of D (so that $\mathrm{er}_D(f) = 0$), then*

$$\mathrm{er}(f) < \frac{4}{m}\left(2s\ln 2 + ns\ln\left(\frac{e(2m-1)}{n}\right) + \ln\left(\frac{7s^2}{\delta}\right)\right)$$

   for $m > n$, where s is the number of terms of f;

4. *If f is an multilevel threshold function and f is an extension of D, then*

$$\mathrm{er}(f) < \frac{4}{m}\left((n+s-1)\ln\left(\frac{2ems}{n+s-1}\right) + \ln\left(\frac{7s^2}{\delta}\right)\right),$$

   for $m \geq n + s$, where s is the number of terms of f.

4 Bounding error by bounding growth function

4.1 Bounding the error

To use results from statistical learning theory, we need to define the *growth function* of a set of functions H mapping from $X = \mathbb{R}^n$ to $\{0, 1\}$. Let $\Pi_H : \mathbb{N} \to \mathbb{N}$ be given by

$$\Pi_H(m) = \max\{|H|_S| : S \subseteq X, |S| = m\},$$

where $H|_S$ denotes H restricted to domain S. Note that $\Pi_H(m) \leq 2^m$ for all m. The function Π_H is known as the growth function of H, and it measures how expressive the hypothesis class H is. The key probability results we employ are the following bounds, due to Vapnik and Chervonenkis [28] and Vapnik [27] (see also [7, 4]): for any $\epsilon, \eta \in (0, 1)$,

$$P^m\left(\{D \in Z^m : \text{for all } f \in H,\ \mathrm{er}(f) < \mathrm{er}_D(f) + \epsilon\}\right) > 1 - 4\,\Pi_H(2m)\,e^{-m\epsilon^2/8},$$

and

$$P^m\left(\left\{D \in Z^m : \text{for all } f \in H,\ \frac{\mathrm{er}(f) - \mathrm{er}_D(f)}{\sqrt{\mathrm{er}(f)}} < \eta\right\}\right) > 1 - 4\,\Pi_H(2m)\,e^{-m\eta^2/4}.$$

Thus, we can obtain (probabilistic) bounds on the error $\mathrm{er}(f)$ of a (partial) extension from a class H when we know something about the growth function of H.

4.2 Growth function bounds

We start with general threshold decision lists. We consider the the set of threshold decision lists on $\mathbb{R}^n$ with at most some number s of terms. (So, the length of the list is no more than s.) We have the following bound.

Theorem 4.1 *Let H be the set of threshold decision lists on $\mathbb{R}^n$ with at most s terms, where $n, s \in \mathbb{N}$. Then*

$$\Pi_H(m) < 4^s \left(\frac{e(m-1)}{n}\right)^{ns},$$

for $m > n$.

Proof: Let S be any given set of m points in $\mathbb{R}^n$. Suppose we have two decision lists $f = (f_1, c_1), \ldots, (f_s, c_s)$, $g = (g_1, d_1), \ldots, (g_s, d_s)$ in H, where each f_i and g_j belong to K, the set of threshold functions on $\mathbb{R}^n$. (We can assume both are of length exactly s by padding with the term $(\mathbb{T}, 0)$ followed by any number of other terms, if necessary.) Certainly, if (i) $c_i = d_i$ for each i and (ii) $f_i(x) = g_i(x)$ for all $x \in S$, then f and g are equal on S. For fixed i, the condition in (ii) is an equivalence relation among functions in K, and the number of equivalence classes is $|K|_S|$ where K is the set of threshold functions. This is bounded by $\Pi_K(m)$, which, it is well-known [11, 9, 4], is bounded above as follows:

$$\Pi_K(m) = 2\sum_{i=0}^{n} \binom{m-1}{k} < 2\left(\frac{e(m-1)}{n}\right)^n.$$

We can therefore upper bound $|H|_S|$ as follows:

$$|H|_S| \leq 2^s \left(2\left(\frac{e(m-1)}{n}\right)^n\right)^s.$$

Here, the first 2^s factor corresponds to the number of possible sequences of c_i and the remaining factor bounds the number of ways of choosing an equivalence class (with respect to S) of threshold functions, for each i from 1 to s. The result follows. □

There is a useful connection between certain types of decision list and threshold functions. We say that a decision list defined on $\{0,1\}^n$ is a 1-*decision list* if the Boolean function in each test is given by a single literal. (So, for each i, there is some l_i such that *either* $f_i(y) = 1$ if and only if $y_{l_i} = 1$, *or* $f_i(y) = 1$ if and only if $y_{l_i} = 0$. Then, it is known [13] (see also [6, 2]) that any 1-decision list is a threshold function. In an easy analogue of this, any threshold decision list is a threshold function of threshold functions [3]. But a threshold function of threshold functions is nothing more than a two-layer threshold network, one of the simplest types of artificial neural network. (A similar observation was made by Marchand *et al.* [19, 20], who construct a 'cascade' network from a threshold decision list.) So another way of bounding the growth function of threshold decision lists is to use this fact in combination with some known bounds [8, 4] for the growth functions of linear threshold networks. This gives a similar, though slightly looser, upper bound.

To bound the growth function of the subclass consisting of s-level threshold functions, we use a result from [1], which shows that the number of ways in which a set S of m points can be partitioned by s parallel hyperplanes is at most $\sum_{i=0}^{n+s-1} \binom{sm}{i}$. (For fixed n and s, this bound is

tight to within a constant, as a function of m.) Noting that we may assume adjacent regions to have different labels, there corresponds to each such partition at most two s-level threshold functions (defined on the domain restricted to S) and we therefore have the following bound.

Theorem 4.2 *Let H be the set of s-level threshold functions on $\mathbb{R}^n$. Then*

$$\Pi_H(m) \leq 2 \sum_{i=0}^{n+s-1} \binom{sm}{i} < 2 \left(\frac{ems}{n+s-1} \right)^{n+s-1},$$

for $n \geq n+s$.

4.3 Proofs of the generalization bounds

From the bound

$$P^m \left(\{ D \in Z^m : \text{for all } f \in H,\ \mathrm{er}(f) < \mathrm{er}_D(f) + \epsilon \} \right) > 1 - 4\,\Pi_H(2m)\, e^{-m\epsilon^2/8},$$

it follows that with probability at least $1-\delta$, for all $f \in H$,

$$\mathrm{er}(f) < \mathrm{er}_D(f) + \sqrt{\frac{8}{m} \left(\ln\left(\Pi_H(2m)\right) + \ln\left(\frac{4}{\delta}\right) \right)}.$$

Theorem 3.1 now follows upon using Theorem 4.1 and Theorem 4.2, respectively.

From the bound

$$P^m \left(\left\{ D \in Z^m : \text{for all } f \in H,\ \frac{\mathrm{er}(f) - \mathrm{er}_D(f)}{\sqrt{\mathrm{er}(f)}} < \epsilon \right\} \right) > 1 - 4\,\Pi_H(2m)\, e^{-m\eta^2/4},$$

it follows that

$$P^m \left(\{ D \in Z^m : \text{there exists } f \in H,\ \text{such that } \mathrm{er}_D(f) = 0, \mathrm{er}(f) \geq \epsilon \} \right)$$

$$\leq P^m \left(\left\{ D \in Z^m : \text{there exists } f \in H,\ \frac{\mathrm{er}(f) - \mathrm{er}_D(f)}{\sqrt{\mathrm{er}(f)}} \geq \sqrt{\epsilon} \right\} \right)$$

$$< 4\,\Pi_H(2m)\,e^{-m(\sqrt{\epsilon})^2/4} = 4\,\Pi_H(2m)\,e^{-m\epsilon/4}.$$

So, with probability at least $1-\delta$, for any $f \in H$ with $\mathrm{er}_D(f) = 0$, we have

$$\mathrm{er}(f) < \frac{4}{\epsilon}\left(\ln(\Pi_H(2m)) + \ln\left(\frac{4}{\delta}\right)\right).$$

Theorem 3.2 now follows from Theorem 4.1 and Theorem 4.2.

To obtain Theorem 3.3 we use a well-known technique often found in discussions of 'structural risk minimization' and model selection (see [27, 23, 17, 12, 4], for instance.) We indicate how to obtain the first part of Theorem 3.3 (the proof of the other parts being very similar).

For $s \in \mathbb{N}$, let H_s denote the set of threshold decision lists of length at most s. We know, by Theorem 3.1 that, with probability at least $1-\delta$, any $f \in H_s$ satisfies $\mathrm{er}(f) < \mathrm{er}_D(f) + \epsilon_0(m, s, \delta)$, where

$$\epsilon_0(m, s, \delta) = \sqrt{\frac{8}{m}\left(2s\ln 2 + ns\ln\left(\frac{e(2m-1)}{n}\right) + \ln\left(\frac{8}{\delta}\right)\right)}.$$

Now let $(p_s)_{s=1}^{\infty}$ be any sequence of positive numbers such that $\sum_{s=1}^{\infty} p_s = 1$. Then, the probability that there is some $f \in H_s$ with $\mathrm{er}(f) \geq \mathrm{er}_D(f) + \epsilon_0(m, s, p_s\delta)$ is less than $p_s\delta$. Therefore,

$$P^m\left(\{D \in Z^m : \text{there is } s \in \mathbb{N} \text{ such that for some } f \in H_s, \mathrm{er}(f) \geq \mathrm{er}_D(f) + \epsilon_0(m, s, p_s\delta)\}\right)$$

$$\leq \sum_{s=1}^{\infty} p_s\delta = \delta.$$

The first part of Theorem 3.3 follows on taking $p_s = 6/(\pi^2 s^2)$. (It should be clear that other choices can be made, such as $p_s = 1/2^{s+1}$, for example. The type of sequence chosen can reflect a prior belief about the likelihood of there being a 'small' partial extension with low error, or can be thought of as a choice of penalty for having chosen a classifier involving a large number of hyperplanes.)

Acknowledgements

This work was initiated during a visit to RUTCOR and DIMACS, Rutgers University, in March 2002. I am grateful to Peter Hammer and colleagues for their hospitality and for stimulating discussion. I thank Kristin Bennett for drawing my attention to the multisurface method.

References

[1] M. Anthony. *Partitioning points by parallel planes*. RUTCOR Research Report RRR-39-2002, Rutgers Center for Operations Research. (Also, CDAM research report LSE-CDAM-2002-10, Centre for Discrete and Applicable Mathematics, London School of Economics.)

[2] M. Anthony. *Discrete Mathematics of Neural Networks: Selected Topics*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2001.

[3] M. Anthony. *Threshold functions, decision lists, and the representation of Boolean functions*. Neurocolt technical report NC-TR-96-028, 1996.

[4] M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

[5] M. Anthony and N. L. Biggs. *Computational Learning Theory: An Introduction*. Cambridge Tracts in Theoretical Computer Science, 30, 1992. Cambridge University Press, Cambridge, UK.

[6] M. Anthony, G. Brightwell and J. Shawe-Taylor. On specifying Boolean functions by labelled examples. *Discrete Applied Mathematics*, 61, 1995: 1–25.

[7] M. Anthony and J. Shawe-Taylor. A result of Vapnik with applications. *Discrete Applied Mathematics*, 47, 1994: 207–217.

[8] E. Baum and D. Haussler. What size net gives valid generalization? *Neural Computation*, 1(1):151–160, 1989.

[9] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth: Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM*, **36**(4), 1989: 929–965.

[10] V. Bohossian and J. Bruck. Multiple threshold neural logic. In *Advances in Neural Information Processing, Volume 10: NIPS'1997*, Michael Jordan, Michael Kearns, Sara Solla (eds), MIT Press, 1998.

[11] T. M. Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition, *IEEE Trans. Electronic Computers* 14, 1965: 326–334.

[12] L. Devroye, L. Györfi and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, 1996.

[13] A. Ehrenfeucht, D. Haussler, M. Kearns, and L. Valiant. A general lower bound on the number of examples needed for learning. *Information and Computation*, 82, 1989: 247–261.

[14] P. L Hammer, T. Ibaraki and U. N. Peled. Threshold numbers and threshold completions. *Annals of Discrete Mathematics* 11, 1981: 125–145.

[15] R.G. Jeroslow. On defining sets of vertices of the hypercube by linear inequalities. *Discrete Mathematics*, 11, 1975: 119–124.

[16] M. Krause. On the Computational Power of Boolean Decision Lists. In Proceedings of the 19th Annual Symposium of Theoretical Aspects of Computer Science (STACS), 2002.

[17] N. Linial, Y. Mansour and R.L. Rivest. Results on learnability and the Vapnik-Chervonenkis dimension. *Information and Computation* 90(1), 1991: 33–49.

[18] O.L. Mangasarian. Multisurface method of pattern separation. *IEEE Transactions on Information Theory* IT-14 (6), 1968: 801–807.

[19] M. Marchand and M. Golea. On learning simple neural concepts: from halfspace intersections to neural decision lists. *Network: Computation in Neural Systems*, 4, 1993: 67–85.

[20] M. Marchand, M. Golea and P. Ruján. A convergence theorem for sequential learning in two-layer perceptrons. *Europhys. Lett.* 11, 1990, 487.

[21] S. Olafsson and Y. S. Abu-Mostafa. The capacity of multilevel threshold functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10 (2), 1988: 277–281.

[22] R. Rivest. Learning Decision Lists. *Machine Learning* 2 (3), 1987: 229–246.

[23] J. Shawe-Taylor, P. Bartlett, R.C. Williamson and M. Anthony. Structural risk minimisation over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 44(5), 1998: 1926–1940.

[24] R. Takiyama. The separating capacity of a multi-threshold element. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7, 1985: 112–116.

[25] György Turán and Farrokh Vatan. Linear decision lists and partitioning algorithms for the construction of neural networks. Foundations of Computational Mathematics: selected papers of a conference held at Rio de Janeiro, Springer 1997, pp 414-423

[26] V. N. Vapnik: *Statistical Learning Theory*, Wiley, 1998.

[27] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, New York, 1982.

[28] V.N. Vapnik and A.Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, **16**(2), 1971: 264–280.

[29] A. Zuev and L. I. Lipkin. Estimating the efficiency of threshold representations of Boolean functions. *Cybernetics* 24, 1988: 713–723. (Translated from Kibernetika (Kiev), 6, 1988: 29–37.)