CONF-970938--1

LA-UR- 97-516

Title: ENHANCED LOWER ENTROPY BOUNDS WITH APPLICATION TO CONSTRUCTIVE LEARNING

Author(s): Valeriu Beiu

Submitted to: Euromicro-Workshop on Computational Intelligence
Budapest, Hungary
September 3-4, 1997

DISCLAIMER

Los Alamos
NATIONAL LABORATORY

Form No. 836 R5
ST 2629 10/91

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Enhanced Lower Entropy Bounds with Application to Constructive Learning

Valeriu Beiu *

Los Alamos National Laboratory, Division NIS-1, Mail Stop D466
Los Alamos, New Mexico 87545, USA

Abstract — In this paper we shall prove two new lower bounds for the *number-of-bits* required by neural networks for classification problems defined by m examples from $\mathbb{R}^n$. Because they are obtained in a constructive way, they can be used for designing a constructive algorithm. These results rely on techniques used for determining tight upper bounds [6], which start by upper bounding the space with an n-dimensional ball. Very recently, a better upper bound has been detailed [9] by showing that the volume of the ball can always be replaced by the volume of the intersection of two balls. A first lower bound for the case of integer *weights* in the range $[-p, p]$ has been detailed in [14]: it is based on computing the logarithm of the quotient between the volume of the ball containing all the examples (rough approximation like in [6]) and the maximum volume of a polyhedron. A first improvement over that bound will come from a tighter upper bound of the maximum volume of the polyhedron by two n-dimensional cones (instead of a ball, as used in [14]). An even tighter bound will be obtained by upper bounding the space by the intersection of two balls (as has been done in [9] for obtaining a tight upper bound).

Keywords — neural networks, *size* complexity, entropy, classification problems, limited *weights*, constructive algorithms.

1. Introduction and Notations

Multilayer feedforward neural networks (NNs) have been experimentally shown to be quite effective in many different applications (see *Applications of Neural Networks* in [3], together with *Part F: Applications of Neural Computation* and *Part G: Neural Networks in Practice: Case Studies* from [15]), but cost effective solutions for large scale computational paradigms have to be hardware implementable — and NNs are by no means an exception. That is why a rigorous analysis of the mathematical properties that enable them to perform so well has generated two directions of research:

- one to find existence /constructive proofs for what is now known as the "*universal approximation problem*" (i.e., any continuous function can be approximated arbitrarily well by a NN);
- another one to find tight bounds on the number of neurons (*size*) needed by the approximation problem (or some particular cases).

The focus of this paper will be on the second aspect. Here we shall denote by *network* any acyclic graph having several input nodes (*inputs*) and some (at least one) output nodes (*outputs*). If with each edge a synaptic *weight* is associated and each node computes the weighted sum of its inputs to which a nonlinear activation function is then applied (*artificial neuron*):

* On leave of absence from the "Politehnica" University of Bucharest, Department of Computer Science, Spl. Independenței 313, RO–77206 Bucharest, Romania.

$$f(z) = f(z_1, \ldots, z_\Delta) = \sigma\left(\sum_{i=1}^{\Delta} w_i z_i + \theta\right), \tag{1}$$

the network is a NN ($w_i \in \mathbb{R}$ are the synaptic *weights*, $\theta \in \mathbb{R}$ is the *threshold*, Δ is the number of inputs of one neuron, and σ is a non-linear activation function).

A *classification problem* is defined by a set of m examples (i.e., data-set) belonging to k different classes. For simplicity we shall limit the number of classes to two ($k=2$), known as a *dichotomy*, but all our results are valid in general. Now:

$$m = m_+ + m_- \tag{2}$$

and $x_1, x_2, \ldots, x_{m_+}$ are the positive examples, while $y_1, y_2, \ldots, y_{m_-}$ are the negative examples; they are taken from an n-dimensional space $\mathbb{R}^n$ ($n \in \mathbb{N} \setminus \{1\}$):

$$x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n}) \in \mathbb{R}^n,\ i = 1, 2, \ldots, m_+,\ m_+ \in \mathbb{N} \text{ and}$$

$$y_j = (y_{j,1}, y_{j,2}, \ldots, y_{j,n}) \in \mathbb{R}^n,\ j = 1, 2, \ldots, m_-,\ m_- \in \mathbb{N}. \tag{3}$$

The *distance* between two vectors (examples) is the classical *Euclidean distance*:

$$dist_E(a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \ldots + (a_n - b_n)^2} = \left\{\sum_{i=1}^{n} (a_i - b_i)^2\right\}^{1/2}. \tag{4}$$

For characterising the data-set, we also define the *minimum* and the *maximum distance* between any positive and negative examples:

$$d = \min_{i=1,\ldots,m_+\ j=1,\ldots,m_-} [\, dist_E(x_i, y_j)\,] \quad \text{and} \quad D = \max_{i=1,\ldots,m_+\ j=1,\ldots,m_-} [\, dist_E(x_i, y_j)\,] \tag{5}$$

A *ball* of radius r ($r \in \mathbb{R}^+ \setminus \{0\}$) centered at $c \in \mathbb{R}^n$ will be denoted by $B_n[c, r]$:

$$B_n[c, r] = \{\, x \in \mathbb{R}^n :\ dist_E(c, x) \le r \,\}; \tag{6}$$

if $n=2$ this is a round disc; if $n=3$ it is a round ball. We shall denote by μ_n the *n-dimensional Lebesgue measure* in $\mathbb{R}^n$. If $A \subseteq \mathbb{R}^2$, $\mu_2(A)$ is the *'area'* of A; if $A \subseteq \mathbb{R}^3$, $\mu_3(A)$ is the *'volume'* of A. Finally, $\alpha(n) = \mu_n(B_n[0, 1])$ is the *volume of the unit ball* in $\mathbb{R}^n$. In particular we have $\alpha(2) = \pi$, $\alpha(3) = 4\pi/3$, while in general $\alpha(2n) = \pi^n / n!$ [12, 17, 20] and $\alpha(2n-1) = 2^n \cdot \pi^{n-1} / [1 \cdot 3 \cdot 5 \cdot \ldots \cdot (2n-1)]$, or in terms of the gamma function: $\alpha(n) = \pi^{n/2} / \Gamma(n/2 + 1)$.

This paper will prove two lower bounds — based on the entropy of the data-set — on the *number-of-bits* required for classification problems. In Section 2 we shall briefly go through some very recent results, while the proofs will be given in Section 3. They are based on computing the required *number-of-bits* for representing the data-set as the logarithm of the quotient between the volume of a ball containing all the examples (rough approximation like in [6]) and the maximum volume of a polyhedron [14]. The first improvement over the bound detailed in [14] is based on a tighter upper bound of the volume of the polyhedron by two n-dimensional cones instead of a ball (as used in [14]). An even tighter bound can be obtained because all the examples from one class always lie inside the intersection of two balls [9], thus the volume of a ball can be replaced by the volume of the intersection of two balls.

2. Previous Results

The problem to find the smallest *size* NN which can realise an arbitrary function given a set of m vectors (examples, or points) in n dimensions is not new. Many results have been obtained for NNs having a *threshold* activation function (see references in [7, 8]). Probably the first lower bound on the *size* of a threshold gate circuit for "almost all" n-ary Boolean functions (BFs) was given by Neciporuk in 1964: $size \geq 2\,(2^n/n)^{1/2}$ [23]. Later, Lupanov has proven a very tight upper bound: $size \leq 2\,(2^n/n)^{1/2} \times \{1 + \Omega\,[(2^n/n)^{1/2}]\}$ for the case when $depth = 4$ [22]. Similar existence exponential bounds can be found in [123], while in [27] a $\Omega\,(2^{n/3})$ existence lower bound for arbitrary BFs has been presented.

For classification problems, one of the first result was that a NN with only one hidden layer having $m-1$ nodes could compute an arbitrary dichotomy (sufficient condition). The main improvements have been as follows:

- Baum [4] presented a NN with one hidden layer having $\lceil m/n \rceil$ neurons[1] capable of realising an arbitrary dichotomy on a set of m points *in general position* in $\mathbb{R}^n$; if the points are on the corners of the n-dimensional hypercube (i.e., binary vectors), $m-1$ nodes are still needed;
- a slightly tighter bound was proven in [18]: only $\lceil 1 + (m-2)/n \rceil$ neurons are needed in the hidden layer for realising an arbitrary dichotomy on a set of m points which satisfy a more relaxed topological assumption (only the points from a sequence from some subsets are required to be *in general position*); also, the $m-1$ nodes condition was shown to be the least upper bound needed;
- Arai [2] showed that $m-1$ hidden neurons are necessary for arbitrary separability (any mapping between input and output for the case of binary-valued units), but improved the bound for the two-category classification problem to $m/3$ (without any condition on the inputs).

These results show that for binary inputs the *size* grows exponentially (as $m \leq 2^n$). Some existence lower bounds for the arbitrary dichotomy problem are (see [16]): (i) a *depth*-2 NN requires at least $m/[n \log(m/n)]$ hidden neurons (if $m \geq 3n$); (ii) a *depth*-3 NN requires at least $2\,(m/\log m)^{1/2}$ neurons in each of the two hidden layer (if $m \gg n^2$); this bound is identical to the one presented in [23] for $m = 2^n$; (iii) an arbitrarily interconnected NN without feedback needs $(2m/\log m)^{1/2}$ neurons (if $m \gg n^2$). Several other bounds for arbitrary BFs can be found in [25]. All of these are: (i) revealing a gap between the upper and the lower bounds, thus encouraging research efforts to reduce (or even close) these gaps; (ii) suggesting that NNs with more hidden layers might have a smaller *size*.

A different approach to classification problems has been presented in [6, 9, 14]; it is based on computing the entropy (see also [1] and [28]) of the data-set.

Proposition 1 (from [6]). The dichotomy of m examples from $\mathbb{R}^n$ *can be solved with:*

$$\#bits < m\,n \cdot \{\lceil \log (D/d) \rceil + 5/2\}.$$

Sketch of proof. Find the examples (from the different classes) which are the closest to one another: $x_d, y_{d'}$ (the distance between them is d). Translate the origin of the axes into

[1] $\lceil x \rceil$ is the ceiling of x, i.e., the smallest integer greater than or equal to x, and $\lfloor x \rfloor$ is the floor of x, i.e., the largest integer less than or equal to x, and all the logarithms are taken to base 2.

x_d and rotate the axes such as the origin (i.e., x_d) and y_d, represent the opposite corners of a hypercube of side length is $l = d/\sqrt{n}$. Quantize the whole space; as there are no examples situated at a distance closer than d, there will be no hypercube containing examples from the two different classes. Because the largest distance is D, there is a ball in $\mathbb{R}^n$ of radius D which contains all the m examples. The *number-of-bits* for *one example* can be computed as $\lceil \log (V_{ball}/v_{hc}) \rceil$ where the volume of the ball is $V_{ball}(D, n) = \alpha(n) \cdot D^n$, and the volume of the hypercube is $v_{hc}(d, n) = (d/\sqrt{n})^n$. By multiplying with m the result follows:

$$\#bits = \left\lceil \log \left\{ \frac{V_{ball}(D, n)}{v_{hc}(d, n)} \right\} \right\rceil = \left\lceil \log \left\{ \frac{\pi^{n/2} D^n}{\Gamma(n/2+1)} \cdot \frac{n^{n/2}}{d^n} \right\} \right\rceil < n \{ \lceil \log (D/d) \rceil + 5/2 \}. \quad \square$$

The exact value detailed is $\#bits_{example} < \lceil n \log (D/d) + 2.0471n - (\log n)/2 - 0.8257 \rceil$.

A non constructive bound has also been presented.

Proposition 2 (from [6]). The entropy of a dichotomy of m examples from $\mathbb{R}^n$ is bounded by $2^{m \lceil \log m \rceil}$.

A better bound has been obtained by replacing the volume V_{ball} with the volume of the intersection of two balls $V(D, n)$.

Proposition 3 (from [9]). The volume of the intersection of two balls in $\mathbb{R}^n$ of the same radius $r \in \mathbb{R}^+ \setminus \{0\}$, placed such that the center of each one is on the boundary of the other one, is $V(r, n) = 2\alpha(n-1) r^n \cdot a(n)$ with:

$$a(n) = \int_{\pi/6}^{\pi/2} (\cos\theta)^n \, d\theta = \frac{n-1}{n} \cdot a(n-2) - \frac{3^{(n-1)/2}}{n \cdot 2^n}.$$

Proposition 4 (from [9]). The dichotomy of $m = m_+ + m_-$ examples from $\mathbb{R}^n$ can always be solved with:

$$\#bits < m_{max} n \cdot \{ \lceil \log (D/d) \rceil + 2 \}$$

where $m_{max} = \max(m_+, m_-) > m/2$.

The exact value detailed is $\#bits_{example} < \lceil n \log (D/d) + 1.8396n - 1.0800 \rceil$.

These bounds are valid for NNs having integer $|weights| < 2^{\#bits_{example}/n}$ (see [10, 11]), but the bound on *weights* is a result of bounding the *number-of-bits*. The problem can be tackled the other way around, i.e. by taking the *weights* from $\{-p, -p+1, \ldots, p\}$ (see [19] and Fig.1a), and proving lower bounds on the *number-of-bits* [14].

Proposition 5 (from [14]). Using integer weights in the range $[-p, p]$, one can correctly classify any set of patterns for which the minimum distance between two patterns of opposite classes is $d_{min} = 1/p$.

Proposition 6 (from [14]). The number-of-bits necessary for the separation of the patterns in general position using weights in the set $\{-p, -p+1, \ldots, 0, \ldots, p\}$ is:

$$\#bits > m n \cdot \lceil \log (2pD) \rceil = m n \cdot \lceil \log (D/d) \rceil$$

3. Lower Entropy Bounds

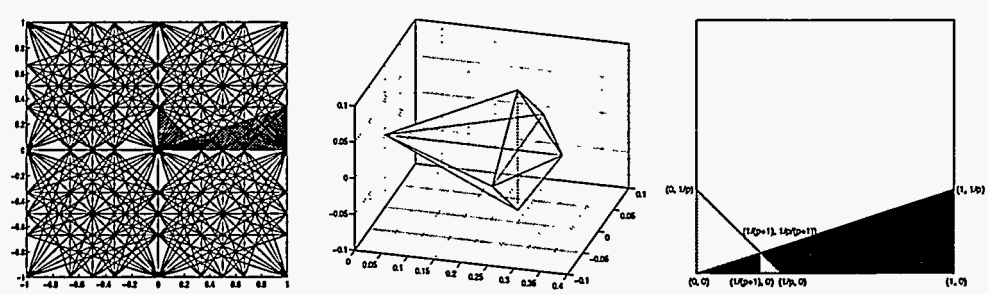
Proposition 7. The volume of an n-dimensional cone of height h and having as basis an $(n-1)$*-dimensional ball of radius* $r \in \mathbb{R}^+ \setminus \{0\}$ *is:*

$$V_{cone}(r, n, h) = \frac{h \cdot V_{ball}(r, n-1)}{n}.$$

Proof. The volume of an n-dimensional cone can be computed by summing the *'volume'* of very thin cylinders, or, at the limit, by summing the *'area'* of the thin discs. This gives:

$$\begin{aligned}
V_{cone}(r, n, h) &= \int_0^h \mu_{n-1}(B_{n-1}[(x, 0, \ldots, 0), r])\, dx \\
&= \int_0^h \alpha(n-1) \cdot \left(\frac{r\,x}{h}\right)^{n-1} dx \\
&= \alpha(n-1) \cdot \frac{r^{n-1}}{h^{n-1}} \cdot \int_0^h x^{n-1}\, dx \\
&= \alpha(n-1) \cdot \frac{r^{n-1}}{h^{n-1}} \cdot \left[\frac{x^n}{n}\right]_{x=0}^{x=h} \\
&= \alpha(n-1) \cdot \frac{r^{n-1}}{h^{n-1}} \cdot \frac{h^n}{n} \\
&= V_{ball}(r, n-1) \cdot \frac{h}{n}
\end{aligned}$$

and concludes the proof. □



(a) (b) (c)

Fig. 1. (a) The hyperplanes implemented with integer *weights* in [–3, 3] (adapted from [19]); (b) the largest resulting polythope in 3D (adapted from [14]); (c) the largest polythope in the plane (used for computing h_1, h_2 and r in *Proposition 8*).

We can now prove a tighter bound for the largest polyhedron (than that with a ball which has been used in [14] — see Fig.1b) .

Proposition 8. The volume of the two n-dimensional cones bounding the largest polyhedron obtained by using weights in the set $\{-p, -p+1, \dots, 0, \dots, p\}$ *is:*

$$V_{2\_cones}(p, n) = \frac{\alpha(n-1)}{n \cdot p^n \cdot (p+1)^{n-1}}.$$

Proof. We shall first determine the height of each of the two cones h_1 and h_2 and the radius of the $(n-1)$-dimensional ball r (see Fig.1c). From one triangle we have $r = h_1/p$, while from another triangle we have $r = h_2$; we also know that $h_1 + h_2 = 1/p$. By solving this system of three equations with three unknowns we get:

$$h_1 = \frac{1}{p+1}, \qquad h_2 = \frac{1}{p(p+1)} \qquad \text{and} \qquad r = \frac{1}{p(p+1)}.$$

The volume of the two n-dimensional cones bounding the largest polyhedron can now be easily computed using *Proposition 7*:

$$\begin{aligned} V_{2\_cones}(p, n) &= \alpha(n-1) \cdot r^{n-1} \cdot \frac{h_1 + h_2}{n} \\ &= \alpha(n-1) \cdot \frac{1}{p^{n-1}(p+1)^{n-1}} \cdot \frac{1/p}{n} \\ &= \frac{\alpha(n-1)}{n \cdot p^n \cdot (p+1)^{n-1}} \end{aligned}$$

which concludes the proof. □

Proposition 9. For solving a dichotomy of $m = m_+ + m_-$ *examples in general position in* $\mathbb{R}^n$*, more than:*

$$\#bits = m \cdot \lceil n\log(D/d) + 0.6515\,n + 0.6515 \rceil / 2$$

are needed.

Proof. The *number-of-bits* for *one example* can be computed as $\lceil \log(V_{ball}/V_{2\_cones}) \rceil$, where the volume of the ball of radius D is $V_{ball}(D, n) = \alpha(n) \cdot D^n$, and the volume of the largest polyhedron has been upper bounded by the volume of the two cones (given by *Proposition 8*):

$$\begin{aligned} \#bits_{example} &= \left\lceil \log\left\{\frac{V_{ball}(D, n)}{V_{2\_cones}(p, n)}\right\} \right\rceil = \left\lceil \log\left\{\frac{\alpha(n) \cdot D^n}{\alpha(n-1)/n/p^n/(p+1)^{n-1}}\right\} \right\rceil \\ &= \left\lceil \log\left\{\frac{\alpha(n)}{\alpha(n-1)}\right\} + n\log D + n\log p + (n-1)\log(p+1) + \log n \right\rceil \end{aligned}$$

We shall first compute a bound for:

$$\begin{aligned}
\log\left\{\frac{\alpha(n)}{\alpha(n-1)}\right\} &= \log\left\{\frac{\pi^{n/2}/\Gamma(n/2+1)}{\pi^{(n-1)/2}/\Gamma[(n-1)/2+1]}\right\} \\
&= \log\left\{\frac{\pi^{n/2}}{\pi^{n/2-1/2}}\cdot\frac{\Gamma(n/2-1/2+1)}{\Gamma(n/2+1)}\right\} \\
&= \log\left\{\pi^{1/2}\cdot\frac{\Gamma(n/2+1/2)}{\Gamma(n/2+1)}\right\} \\
&= \log\left\{\frac{\Gamma(1/2)\cdot\Gamma(n/2+1/2)}{\Gamma(n/2+1)}\right\} \\
&= \log\{\,B(n/2+1/2,\ 1/2)\,\}
\end{aligned}$$

where

$$B(m,n) = \int_0^1 x^{m-1}(1-x)^{n-1}\,dx = 2\cdot\int_0^{\pi/2}(\sin\theta)^{2m-1}(\cos\theta)^{2n-1}\,d\theta.$$

In our particular case $2m-1$ is in fact $2(n/2+1/2)-1=n$, while $2n-1$ becomes $2(1/2)-1=0$, and by substitution we obtain:

$$\log\left\{\frac{\alpha(n)}{\alpha(n-1)}\right\} = \log\left\{2\cdot\int_0^{\pi/2}(\sin\theta)^n\,d\theta\right\}.$$

Because θ belongs to $[0,\pi/2]$, we also have $\theta/(\pi/2)\le\sin\theta\le\theta$, which gives us the following bound (here we do need an upper bound as this term is negative):

$$\begin{aligned}
\log\left\{\frac{\alpha(n)}{\alpha(n-1)}\right\} &< \log\left(2\cdot\int_0^{\pi/2}\theta^n\,d\theta\right) \\
&= \log\left\{2\cdot\left[\frac{\theta^{n+1}}{n+1}\right]_{\theta=0}^{\theta=\pi/2}\right\} \\
&= \log\left\{2\cdot\frac{\pi^{n+1}}{2^{n+1}}\cdot\frac{1}{n+1}\right\} \\
&= 1 + (n+1)\log(\pi/2) - \log(n+1) \\
&= 0.6515\,n - \log(n+1) + 1.6515
\end{aligned}$$

For the interested reader we mention that a slightly tighter bound for $\log\{\alpha(n)/\alpha(n-1)\}$ could be obtained by using Stirling's formula, but we are anyhow going to prove a tighter bound in *Proposition 10.*

Using this result and taking $d=1/(2p)$, which is the minimum value, we have:

$$\#bits_{example} > \lceil 0.6515\,n - \log(n+1) + 1.6515 + n\log D + n\log p + (n-1)\log(p+1) + \log n\rceil$$

$$= \left\lceil 0.6515\, n + \log\left(\frac{n}{n+1}\right) + n\log D + n\log\left(\frac{1}{2d}\right) + (n-1)\log\left(1+\frac{1}{2d}\right) + 1.6515 \right\rceil$$

$$= \left\lceil 0.6515\, n + \log\left(\frac{n}{n+1}\right) + n\log(D/d) - n + (n-1)\log\left(\frac{2d+1}{2d}\right) + 1.6515 \right\rceil$$

In the worst case $\log\left(\frac{2d+1}{2d}\right)$ can be $\log 2 = 1$ (for $p = 1$, $d = 1/2$), thus:

$$\#bits_{example} > \left\lceil 0.6515\, n + n\log(D/d) - n + \log\left(\frac{n}{n+1}\right) + (n-1) + 1.6515 \right\rceil$$

$$> \lceil n\log(D/d) + 0.6515\, n + 0.6515 \rceil .$$

By multiplying with $\min(m_+, m_-) \le m/2$ the proof is concluded. □

A tighter lower bound can be obtained if instead of the volume V_{ball} used in *Proposition 9*, we use the volume of the intersection of two balls $V(D, n)$ as detailed in [9].

Proposition 10. For solving a dichotomy of $m = m_+ + m_-$ examples in general position in $\mathbb{R}^n$, more than:

$$\#bits = \max(m_+, m_-) \cdot \lceil n\log(D/d) - 0.2075\, n + \log n + 0.0665 \rceil$$

are needed.

Proof. The *number-of-bits* for *one example* can be computed as $\lceil \log(V(D, n)/V_{2\_cones}) \rceil$, where the volume of the intersection of two balls is $V(D, n) = 2 \cdot \alpha(n-1) \cdot D^n \cdot a(n)$ (see *Proposition 3* and *4* as well as [9]), and the volume of the largest polyhedron is upper bounded by the volume of the two cones (given by *Proposition 8*):

$$\#bits_{example} = \left\lceil \log\left\{\frac{V(D, n)}{V_{2\_cones}(p, n)}\right\} \right\rceil = \left\lceil \log\left\{\frac{2 \cdot \alpha(n-1) \cdot D^n \cdot a(n)}{\alpha(n-1)/n/p^n/(p+1)^{n-1}}\right\} \right\rceil$$

$$= \lceil 1 + n\log D + \log\{a(n)\} + \log n + n\log p + (n-1)\log(p+1) \rceil .$$

The bounds for $\log\{a(n)\}$ follow from the fact that $a(n) = \int_{\pi/2}^{\pi/6} (\cos\theta)^n\, d\theta$ (??); because $\theta \in [\pi/6, \pi/2]$, $(\sqrt{3}/2 + \pi/12) - \theta/2 \le \cos\theta \le \sqrt{3}/2 = \cos(\pi/6)$. Here again we have to compute an upper bound as $\log\{a(n)\}$ is negative for any n:

$$\log\{a(n)\} < \log\left\{\int_{\pi/6}^{\pi/2} (\sqrt{3}/2)^n\, d\theta\right\} = \log\left\{\left(\frac{\sqrt{3}}{2}\right)^n \cdot \frac{\pi}{3}\right\} = 0.0665 - 0.2075\, n$$

and by taking $d = 1/(2p)$, which is the minimum value, the result follows:

$$\#bits_{example} > \lceil 1 + n\log D + 0.0665 - 0.2075\, n + \log n + n\log p + (n-1)\log(p+1) \rceil$$

$$= \left\lceil n\log D - 0.2075\, n + \log n + n\log\left(\frac{1}{2d}\right) + (n-1)\log\left(1+\frac{1}{2d}\right) + 1.0665 \right\rceil$$

$$= \left\lceil n\log(D/d) - 0.2075\,n + \log n - n + (n-1)\log\left(\frac{2d+1}{2d}\right) + 1.0665 \right\rceil$$

$$> \lceil n\log(D/d) - 1.2075\,n + \log n + (n-1) + 1.0665 \rceil$$

$$= \lceil n\log(D/d) - 0.2075\,n + \log n + 0.0665 \rceil .$$

By multiplying with $\max(m_+, m_-) \geq m/2$, and the proof is concluded. □

4. Conclusions

Based on the entropy of the data-set, this paper has presented a new nonconstructive proof on the *number-of-bits* required for solving a dichotomy problem. The resulting lower bounds are tighter than the ones previously known. It seems very promising that the bound proved in *Proposition 10* is in fact lowering the $n\log(D/d)$ term!

Because the proofs for the *number-of-bits* are constructive, they can be used in conjunction with results like the ones presented in [19] for designing a constructive algorithm. There is still one problem: the shape of the bounding spaces does not lend itself easily to practical applications. Bounding the space with a ball, or the intersection of two balls — which, as we have seen, is theoretically possible — is computationally too difficult. For all practical cases, the simplest bounding space is a hypercube (or an intersection of hypercubes). Unfortunately, by using the intersection of two hypercubes we have to pay by a logarithmic factor of $\log n$ (for the *number-of-bits*). We are working on this particular aspect by trying to use other co-ordinates (e.g., polar co-ordinates instead of the rectangular ones).

References

1. Abu-Mostafa, Y.S.: Connectivity versus entropy, in D.Z. Anderson (ed.): *Neural information processing systems* (NIPS*87, Denver, CO), American Institute of Physics, NY (1988) 1–8.
2. Arai, M.: Bounds on the number of hidden units in binary-valued three-layer neural networks, *Neural Networks* **6**(6) (1993) 855–860.
3. Arbib, M.A. (ed.): *The handbook of brain theory and neural networks*, MIT Press, Cambridge, MA (1995).
4. Baum, E.B.: On the Capabilities of Multilayer Perceptrons, *J. of Complexity* **4** (1988) 193–215.
5. Baum, E.B.: When are *k*–nearest neighbour and back propagation accurate for feasible size sets of examples ?, in E. Amaldi and E. Mayoraz: *Mathematical foundations of artificial neural networks* (summer school lecture notes), Swiss Federal Institute of Technology (Lausanne) and Kurt Bösch Academic Institute (Sion), 1992. Also in S.J. Hanson, G.A. Drastal and R.L. Rivest (eds.): *Computational learning theory and natural learning systems—Vol. I: Constraints and prospects*, Bradford/MIT Press (1994).
6. Beiu, V.: Entropy bounds for classification algorithms, *Neural Network World* **6**(4) (1996) 497–505.
7. Beiu, V.: Digital integrated circuit implementations, *Chapter E1.4* in [15].
8. Beiu, V.: *VLSI complexity of discrete neural networks*, Gordon and Breach (1997).
9. Beiu, V., T. de Pauw: Tight bounds on the size of neural networks for classification problems, submitted to the *Intl. Work-Conf. on Artif. Neural Nets* (IWANN'97, Lanzarote, Spain), June 4–6, 1997.

10. Beiu, V., Taylor, J.G.: VLSI-optimal neural network learning algorithm, in D.W. Pearson, N.C. Steele and R.F. Albrecht (eds.): *Artificial neural nets and genetic algorithms* (ICANNGA'95, Alès, France), Springer-Verlag, Vienna (1995) 61–64.
11. Beiu, V., Taylor, J.G.: Direct synthesis of neural networks, *Proc. MicroNeuro'96* (Lausanne, Switzerland), IEEE CS Press, Los Alamitos, CA (1996) 257–264.
12. Bishop, C.M.: *Neural networks for pattern recognition*, Oxford University Press, Oxford, UK (1995).
13. Bruck, J., Smolensky, R.: Polynomial threshold functions, AC^0 functions and spectral norms, *Res. Rep. RJ 7410* (67387), 11/15/89, IBM Yorktown Heights, NY (1989). Also in *SIAM J. Computing* **21**(1) (1992) 33–42.
14. Draghici, S., Sethi, I.K.: On the possibilities of the limited precision weights neural networks in classification problems, submitted to the *Intl. Work-Conf. on Artif. Neural Nets* (IWANN'97, Lanzarote, Spain), June 4–6, 1997.
15. Fiesler, E., Beale, R. (eds.): *Handbook of neural computation*. Oxford University Press and the Institute of Physics Publishing, NY (1996).
16. Hassoun, M.H.: *Fundamentals of artificial neural networks*, MIT Press, Cambridge, MA (1995).
17. Hecht-Nielsen, R., *Neurocomputing*, Addison-Wesley, Reading, MA (1989).
18. Huang, S.-C., Huang, Y.-F.: Bounds on the number of hidden neurons of multilayer perceptrons in classification and recognition, *IEEE Trans. on Neural Networks* **2**(1) (1991) 47–55.
19. Khan, A.H., Hines, E.L.: Integer-weight neural networks, *Electronic Lett.*, **30**(15), (1994) 1237–1238 (see also Khan, A.H.: Feedforward neural networks with constrained weights, *PhD dissertation*, Intelligent Sys. Lab., Dept. of Eng., Univ. of Warwick, Coventry CV4 7AL, UK, August 1996, accessible at http://www.eng.warwick.ac.uk/~es2029).
20. Kohonen, T.: *Self-organizing and associative memory*, Springer-Verlag, Berlin (1988).
21. Krishnamoorthy, A.V., Paturi, R., Blume, M., Linden, G.D., Linden L.H., Esener, S.C.: Hardware tradeoffs for Boolean concept learning, *Proc. World Conf. on Neural Networks '94* (WCNN'94, San Diego), Lawrence Erlbaum & INNS Press, Hillsdale (1994) vol. 1, 551–559.
22. Lupanov, O.B.: The synthesis of circuits from threshold elements, *Problemy Kibernetiki* **20** (1973) 109–140.
23. Neciporuk, E.I.: The synthesis of networks from threshold elements, *Problemy Kibernetiki* **11** (1964) 49–62. English translation in *Automation Express* **7**(1) 35–39 and **7**(2) 27–32.
24. Parberry, I.: *Circuit complexity and neural networks*, MIT Press, Cambridge, MA (1994).
25. Paugam-Moisy, H.: Optimisation des réseaux des neurones artificiels. *PhD dissertation*, Laboratoire de l'Informatique du Parallélisme LIP-IMAG, École Normale Supérieure de Lyon, 46 Allée d'Italie, 69364 Lyon, France (1992).
26. Raghavan, P.: Learning in threshold networks: A computational model and applications. *Tech. Rep. RC 13859*, IBM Res. (1988). Also in *Proc. 1st Workshop on Computational Learning Theory* (Cambridge, MA), ACM Press, NY (1988) 19–27.
27. Siu, K.-Y., Roychowdhury, V., Kailath, T.: Depth-size tradeoffs for neural computations, *IEEE Trans. on Comp.* **40**(12) (1991) 1402–1412.
28. Williamson, R.C.: ε-entropy and the complexity of feedforward neural networks, in R.P. Lippmann, J.E. Moody and D.S. Touretzky (eds.): *Neural information processing systems* (NIPS*90, Denver, CO), Morgan Kaufmann, San Mateo, CA (1991) 946–952.