Dichotomization of Small Binary Sets using Feedforward Neural Networks

Yaakov Stein
Efrat Future Technology Ltd.
110 Yigal Alon St.
Tel Aviv 67891, Israel

PREPRINT

Abstract

Feedforward neural networks with n binary inputs, a single layer of threshold units, and a single binary output are capable of implementing all 2^{2^n} dichotomies of the set of all possible binary n-vectors. We concentrate on the *dichotomization capacity*, ie. the total number of dichotomies which specific architectures can realize. The number of hidden units required to produce all possible dichotomies is of particular interest. We consider these questions for both perceptrons and single hidden layer networks of biased or unbiased neurons, and continuous (real) valued or Ising (binary valued) couplings. We give some analytic results, and obtain exact values for small systems based on exhaustive enumeration. The latter is made feasible through the exploitation of symmetries in order to reduce the number of distinct networks in a given architecture.

1 Dichotomies of All Binary Vectors

In this paper we will consider vectors of n binary elements $\{S_i\}_{i=1,\ldots,n}$, $S_i = \pm 1$, which can be pictured as the corners of an n dimensional hypercube. We wish to partition all possible such binary vectors into one of two classes, which we label ± 1 as well. Such a boolean function is usually called a *dichotomy* in pattern recognition terminology, a *switching function* in the older logic design literature, and the set of positively classified vectors is a *concept* in the scope of modern learning theory. Since there are 2^n distinct vectors to be arbitrarily classified, the total number of dichotomies is 2^{2^n}. For example, for $n = 1$ there are two line interval vertices -1 and $+1$ and these can be dichotomized in $2^2 = 4$ ways, namely 1) both negatively classified (the empty concept), 2) $-1 \to -1$ and $+1 \to +1$, 3) $-1 \to +1$ and $+1 \to -1$, and 4) both positively classified. Similarly for $n = 2$ there are four corners to the square and these can be dichotomized in $2^4 = 16$ ways.

We will always consider the different vector elements to be distinguishable, although in logic design much attention is paid to the case of dichotomies constrained to be *symmetric* functions of the S_i — ie. functions of $S = \sum_{i=1}^{n} S_i$. Note that S changes in steps of two and that $-n \leq S \leq n$; thus it can take only $n+1$ distinct values. We conclude that only 2^{n+1} of the possible 2^{2^n} dichotomies are symmetric in this sense. The *parity* function, that returns $+1$ when the number of positive input units is odd, is symmetric in this sense.

We are more interested in the special case of dichotomies in which exactly one half of the 2^n vectors are positively classified and one half negatively classified. We call these *demichotomies* and there are $\binom{2^n}{2^{n-1}}$ possible. For $n = 1$, two out of the four aforementioned dichotomies are demichotomies; for $n = 2, 3$ there are 6 out of 16 and 70 out of 256 respectively. For large n, using Stirling's approximation, one can easily show that there are approximately $\sqrt{\frac{2}{\pi}} \left(2^{2^n - \frac{n}{2}}\right)$ demichotomies. There are special demichotomies for which if an input I is positively classified then $-I$ must be negatively classified. Both $n = 1$ demichotomies are of this sort, while for $n = 2$ four are and two (the exclusive or xor and its complement) are not. We call this type of demichotomy a *hemichotomy.* Since we can arbitrarily label half of the vector elements, the other half then being determined, there are exactly $2^{2^{n-1}}$ hemichotomies. For example, when $n = 3$, only 16 of the 70 demichotomies and 256 dichotomies are hemichotomies.

2 Feedforward Neural Networks

One device which explicitly computes a dichotomy is the layered feedforward network of hard limiting neurons. This device has n_I binary input units I_j, zero or more intermediate (hidden) layers of binary units and a single binary output unit O. In order to realize a dichotomy we select the proper input layer size $n_I = n$, and require that when one clamps the input units to any binary vector $I_j = S_j$, the output unit O must return the label appropriate to that vector.

We will restrict ourselves here to networks having no hidden layers (perceptrons) or with a single layer of $n_H \geq 2$ hidden units fully connected above and below and with no direct connections from input to output (two-layer network). The disallowed case of one hidden unit $n_H = 1$, is exactly equivalent to the perceptron. For the perceptron, the output is calculated by

$$O = \mathrm{sgn}(\sum_j J_{OI_j} I_j - \theta_O) \qquad (1)$$

while for the two-layer network

$$\begin{aligned} H_i &= \mathrm{sgn}(\sum_j J_{H_i I_j} I_j - \theta_{H_i}) \\ O &= \mathrm{sgn}(\sum_i J_{O H_i} H_i - \theta_O)\,. \end{aligned} \tag{2}$$

We will call the J values *couplings* although the biological term is *synaptic efficacies*. The θ terms will be denoted *biases*, although physicists usually refer to them as *thresholds*, the term 'bias' being reserved in the physics literature (eg. [2, 4]) for the relative excess of positives over negatives in the input pattern.

One may wish to consider unbiased networks, ie. ones with $\theta_{H_i} = \theta_O = 0$, or to allow the biases to take arbitrary values. For unbiased networks, the above equations imply

$$I_j \longrightarrow -I_j \qquad \Longrightarrow \qquad O \longrightarrow -O \tag{3}$$

and thus only hemichotomies can be realized. In the most general instance the couplings J_{OI_j}, $J_{H_iI_j}$, and J_{OH_i} are *continuous*, ie. can take arbitrary positive or negative real values; however the *Ising* case is of interest, wherein they are restricted to ±1. We have thus defined eight different special cases, namely :

1. the unbiased Ising perceptron (see section 8)
2. the biased Ising perceptron (see section 9)
3. the unbiased continuous perceptron (see section 7)
4. the biased continuous perceptron (see section 6)
5. the unbiased Ising two-layer network (see section 12)
6. the biased Ising two-layer network (see section 13)
7. the unbiased continuous two-layer network (see section 14)
8. the biased continuous two-layer network (see section 15).

In the course of our discussion we will also treat several other cases.

A dichotomy which can be realized by a perceptron is called *linearly separable* since in this case the positively labeled hypercube corners can be separated from the negatively labeled ones by a single $n-1$ dimensional hyperplane. In the old literature such a switching function was called a (*linear*) *threshold function*. For $n = 1$ all four dichotomies are trivially separated by a point; for $n = 2$ only 14 of the 16 dichotomies of the square are separable by a line, the two exceptions being, once again, the exclusive or and its complement. We will see later on that for $n = 3$ exactly 104 out of the 256 dichotomies are linearly separable. A hemichotomy which is linearly separable can always be separated by a hyperplane which passes through the origin. A dichotomy which is not linearly separable can always be implemented as the union of areas bounded by hyperplanes, ie. by a biased feedforward network with a single hidden layer. This is the rationale for our limiting ourselves to the discussion of single hidden layer networks.

3 Dichotomization capacity

Many studies of the performance of neural networks [2, 3, 4, 8, 13] have dealt with the question of *memory capacity*, that is the number of randomly chosen binary vectors which one can require to be positively classified. This approach is most appropriate when the network is to be used as an associative memory, as in the case of the Hopfield model [5]. In such an application only the recognition of stored memory patterns (with perhaps some degree of error correction) is required, while many spurious memories may exist. Feedforward networks are more commonly used as *pattern classifiers*, and in this application the cost of a 'false positive' reaction may be comparable to that of a 'rejection'.

We will thus address another question – given an architecture (by which we mean the constraints on the couplings and biases and the number of hidden units), how many distinct dichotomies can be realized? We call this number the *dichotomization capacity* of the architecture, and denote it by $D(n)$, using various sub- and superscripts to distinguish specific network architectures. A network which can realize all possible dichotomies, ie. one for which $D(n) = 2^{2^n}$, is called a *universal realizor*. In learning theory terminology one says that such a network 'shatters' the set of hypercube vertices. We will be particularly interested in the minimum number of hidden units n_H required for a two-layer network to be such a universal realizor.

The question of the number of realizable dichotomies is interesting for a somewhat different reason. In *real world* problems the dimension of the input space tends to be large, and specification of the required output for all possible input vectors is impractical. For such cases we specify the output for only a small number of representative input vectors (the *training set*) but desire that the network produce the required output for *all* possible inputs (*generalization*). In general, when the dichotomization capacity is large, the probability of proper generalization is small. Thus for large input dimension, one should strive to use the network with the smallest dichotomization capacity, which is still capable of realizing the training set.

4 Bounds and Exhaustive Enumeration

It may be useful to have upper bounds to the dichotomization capacity. The most general bound is the *enumeration bound* which results from noting that the number of dichotomies realized can not exceed the total number T of distinct networks for the given architecture. In general this bound will not be very tight since many different networks may produce the same dichotomy. We will see that this bound is always definable, even when the couplings take on continuous values, although at first sight the number of networks would then seem to be infinite. A more popular bound, generally applicable to networks with discrete couplings, is the so called *information theoretic bound* [3, 4] which results from requiring the amount of information (in bits) extracted from a network not to exceed the information put into it. Finally, since surfaces separating the classes are hyperplanes, one may be able to employ *hyperplane counting* arguments [1] to bound the number of attainable dichotomies, although these arguments are most useful for perceptrons.

In most cases we will find the bounds to be extremely loose, and will desire to find the dichotomization capacity exactly. For small enough systems, this can be done by *exhaustive enumeration*, that is, by explicitly producing all T distinct networks, inputing all 2^n possible patterns, and counting all the dichotomies thus formed. Since the number of networks and the number of input patterns are both exponential in n, this plan can only be carried out for extremely small systems, for example $n \leq 5$. Even substantial increase of computer power will not appreciably increase the pattern lengths amenable to this treatment. However, the proper exploitation of network symmetries, in order to reduce the number T of networks that need be checked, can significantly reduce computer time (in addition to sharpening the enumeration bound). This is perhaps the most significant contribution of this work.

5 Obtainable dichotomization capacities

Before dealing with the specific cases we ask whether all the values $1 \ldots 2^{2^n}$ are valid dichotomization capacities for *some* feedforward network architecture? We would perhaps think not, since an architecture which realizes a given pattern set will necessarily realize many others related to it by symmetry. The specific symmetry we have in mind is that of permuting the *input* neurons. For example, an architecture which can realize the dichotomy in which the only positive pattern has the first input neuron 1 and all the rest -1, can as easily implement a dichotomy for which any single input neuron is 1 in the positive pattern. Indeed, the number of positive inputs is obviously invariant under permutation, and thus the n pattern sets with a single positive input, the $\binom{n}{2} = \frac{n(n-1)}{2}$ with two positive inputs, etc. are all realizable together.

The characteristic of being necessarily realizable by the same architecture is an equivalence relation, which partitions the 2^{2^n} possible functions into mutually exclusive *permuta-*

tion sets. For a network with only a single input no permutations are possible, and thus all $2^{2^1} = 4$ functions are separate permutation sets. For more inputs there will be permutation sets with between 1 and $n!$ elements. The first few cases are :

$$\begin{array}{lrcl}
n = 1 & 2^{2^1} & = & 4 * 1 \\
n = 2 & 2^{2^2} & = & 8 * 1 + 4 * 2 \\
n = 3 & 2^{2^3} & = & 16 * 1 + 48 * 3 + 16 * 6 \\
n = 4 & 2^{2^4} & = & 32 * 1 + 32 * 3 + 224 * 4 + 224 * 6 + 1680 * 12 + 1792 * 24 \\
n = 5 & 2^{2^5} & = & 64 * 1 + 960 * 5 + 4032 * 10 + 96 * 12 + 3072 * 15 + 29,760 * 20 \\
 & & & +126,016 * 30 + 2,830,176 * 60 + 34,339,072 * 120\,.
\end{array} \tag{4}$$

To understand the coefficient of the first term, we consider once again the invariance of the number of positive inputs. Thus a dichotomy wherein exactly all patterns with a single positive input are labeled $+1$ is a singleton permutation set, as is that which consists of all patterns with exactly two positive inputs, or three, or any number up to n. In addition the dichotomies consisting of all single positive patterns and all double ones, or all singles and triples, etc. are singletons. More generally all singletons can be constructed from combinations of all the $0 \ldots n$ positive input patterns, and there are thus 2^{n+1} such.

The other terms are more difficult to understand, being more complex mixtures. For example, let us consider dichotomies with two positively labeled input patterns, one with a single positive input unit and the other with two positive inputs. There are

$$\binom{n}{1}\binom{n}{2} = \frac{n^2(n-1)}{2}$$

such dichotomies, but one must distinguish two cases. There may be a positive input unit in common (there are $2\binom{n}{2}$ such), or there may not, in which case the size of the permutation set is

$$\binom{n}{2}\binom{n-2}{1} = \frac{n(n-1)(n-2)}{2} + n(n-1)\,.$$

As a second example, let us take a three positive input and a two positive input patterns. Now there are three cases, with two, one or no positive inputs in common. The sizes of the permutation sets are determined from

$$\binom{n}{3}\binom{n}{2} = \binom{n}{3}\binom{3}{2} + \binom{n}{3}\binom{n-3}{2} + \binom{n}{3}\binom{3}{1}\binom{n-3}{1}\,.$$

We can proceed to generate permutation sets in this fashion, but the complexity increases exponentially in n. What one does discover from such calculations, as we indeed see in the above special cases, is that as n increases, more and more of the weight goes into the largest ($n!$) term.

We now return to the question with which we opened this section. An architecture realizes the union of permutation sets. For there to be disallowed dichotomization capacities, we must have a gap in the sums of the coefficients of the permutation set expansions, such as those of equation (4). For $n \leq 5$ such gaps are not possible, however we have seen that there are only 2^{n+1} singletons and that for large n, most of the other permutation sets are maximal. For large enough n, we have $n! > 2^{n+1}$, and thus one can not rule out such gaps.

6 The biased continuous perceptron

We start with this case since it has been the most seriously considered to date. In the terminology of the older literature we ask, how many of the 2^{2^n} switching functions are threshold functions? In more modern form we ask, how many dichotomies are linearly separable? We denote this number simply $D(n)$ (without subscripts).

Cover [1] produced a bound $C(n)$, based on hyperplane counting arguments first put forth by Steiner [15] in 1826 for two and three dimensions, and later extended to an arbitrary number of dimensions by Schläfli [12]. While the two dimensional case is trivial, the three dimensional one has all the elements of the general case, and thus the main result is often referred to as Steiner's theorem. Subsequently this same theorem and variations have been found many times. Cover's bound states that :

$$D(n) \leq C(n) = 2 \sum_{k=0}^{n} \binom{2^n - 1}{k} \tag{5}$$

This is, however, only an upper bound, holding exactly only when the points to be dichotomized are in *general position*, ie. when no subset of $n+1$ points lie on an $n-1$ or lower dimensional hyperplane. This is highly probable for points randomly chosen in space, however we are interested in the highly nonrandom case of all the vertices of the hypercube. For $n = 2$ there are indeed no three points of the square on a single line, and so the Cover bound is attained. For $n = 3$ however, 12 of the 70 four point subsets of the cube lie on a plane, and thus the cube vertices are *not* in general position. We indeed find that only 104 dichotomies are linearly separable, strictly fewer than the 128 of Cover's prediction. Similarly, out of the 4368 five point subsets of the tesseract's vertices, 1360 are on a three dimensional hyperplane; for $n = 5$ the 906,192 six point subsets divide up as follows : 8480 on a 3-plane, 341,520 on a 4-plane, and 556,192 do not fall on a lower dimensional plane. For large n, the probability of general position of $n+1$ randomly chosen hypercube points declines, and Cover's bound becomes extremely loose, as one can observe from the third column of table 1.

Cover's bound is not attained for points not in general position due to degeneracy of space partitions which the hyperplanes should have formed. An exact method of taking this degeneracy into account was developed by Winder [17], however the calculations involved in its evaluation (which we used in deriving the numbers of subsets in the previous paragraph) become quite involved for high dimension.

A lower bound has been derived based on the observation [14] that for any linearly separable dichotomy of n variables, we can create $(2^n + 1)$ distinct linearly separable dichotomies of $n + 1$ variables by a simple geometric procedure. The procedure consists of cloning the original n dimensional hypercube and considering the two hypercubes to be the $I_j = \pm 1$ projections of a $n + 1$ dimensional hypercube. On the clone one then places a hyperplane parallel to the separating hyperplane, and translates it (keeping it parallel to the first), forming a new dichotomy after crossing each of the 2^n vertices. We find that $D(n + 1) \geq (2^n + 1)D(n)$ and since $D(1) = 4$, we obtain

$$2^{\frac{1}{2}[n^2-n+4]} < 4 \prod_{k=1}^{n-1} (2^k + 1) \leq D(n)\,. \tag{6}$$

One can improve the constant in the leftmost exponent [18] by commencing the product at some higher n, and with little further work [10] for $n > 8$ can show $D(n) > 2^{\frac{1}{2}(n^2+n)+8}$.

Thus, for large enough n

$$2^{\frac{1}{2}n^2} < 2^{\frac{1}{2}[n^2+n]+8} \leq D(n) \leq 2 \sum_{k=0}^{n} \binom{2^n - 1}{k} < 2^{n^2}\,, \tag{7}$$

so that *asymptotically*

$$D(n) \sim 2^{\gamma n^2} \qquad \text{with} \qquad \frac{1}{2} < \gamma = \frac{\log_2 D(n)}{n^2} < 1\,. \tag{8}$$

Winder [16] and Muroga with co-workers [9, 10, 11] have determined $D(n)$ numerically for small n, (see table 1). In principle this involves generating all the possible dichotomies and checking for linear separability, eg. using linear programming. However, considerable ingenuity has been employed in order to reduce the number of checks which must actually be performed. An alternative approach relies upon Winder's previously mentioned theorem [17]. Although considerable effort was put into these compilations, the n values amenable to computer work are still too small to allow observation of asymptotic tendencies. For $n = 8$ the value derived for γ from equation (8) seems to be oscillating in the vicinity of $\gamma \approx \frac{2}{3}$, although some speculate that γ should approach one. Even were one to assume $\gamma = 1$, for large n equation (8) represents only a vanishingly small fraction of the total number of dichotomies 2^{2^n}.

n	$C(n)$	$D(n)$	$\frac{D(n)}{C(n)}$	$\gamma(n)$
1	4	4	1	2.0
2	14	14	1	0.952
3	128	104	0.8125	0.744
4	3882	1882	0.4848	0.680
5	412,736	94,572	0.2291	0.661
6	151,223,522	15,028,134	0.0994	0.662
7	189,581,406,208	8,378,070,864	0.0442	0.673
8	820,064,805,806,914	17,561,539,552,946	0.0214	0.687

Table 1: Known dichotomization capacities for continuous perceptrons $D(n)$ (from [9, 10, 11, 16]), with Cover's bound $C(n)$, the actual capcity to Cover bound ratio and the derived value of the asymptotic exponent $\gamma(n)$.

7 The unbiased continuous perceptron

The unbiased continuous perceptron, like all unbiased networks, can only realize hemichotomies. These perceptrons were called *self-dual threshold functions* in the old literature. The *dual* of a boolean function of n boolean variables $f(x_1, x_2, \ldots, x_n)$ is defined to be $-f(-x_1, -x_2, \ldots, -x_n)$, and thus the term 'self-dual' means $f(x_1, x_2, \ldots, x_n) = -f(-x_1, -x_2, \ldots, -x_n)$, which is exactly (3).

The unbiased case has lately been the subject of much interest. In the biologically inspired 'Hopfield model' [5] each neuron functions as an unbiased continuous perceptron receiving input from all the others. The memory capacity for this case has been studied extensively by Gardner [2], and that pioneering work triggered the interest of physicists in the question of neural network capacity.

We will denote the dichotomization capacity for unbiased cases by appending a superscripted asterisk, and so $D^*(n)$ stands for the capacity desired here. Unbiased networks, lacking the degrees of freedom of the biases, have severely reduced capacity as compared to similar but biased networks. In the present case, the bias is exactly equivalent to an input to output coupling, which leads to

$$D^*(n) = D(n-1) \sim 2^{\gamma(n-1)^2} \,. \tag{9}$$

The formal proof of the left hand equality proceeds along the following lines. Given any biased perceptron in $n-1$ variables, one can build an unbiased perceptron in n variables by substituting $J_{OI_n} I_n$ for θ_O with $I_n = -1$. Two such distinct biased perceptrons must differ in the classification of at least one input $(n-1)$-vector, and in this case the two corresponding unbiased perceptrons will also disagree on the corresponding augmented input vector; and we have thus shown that $D^*(n) \geq D(n-1)$. Conversely, we need to consider two distinct unbiased perceptrons, ie. perceptrons which differ regarding some n-vector. However, since these perceptrons only implement hemichotomies, we need only require different outputs for input vectors with $I_n = -1$. For this case the corresponding biased perceptrons will also differ, and thus $D(n-1) \geq D^*(n)$ as well. The asymptotic behavior now follows from equation (8).

8 The unbiased Ising perceptron

The unbiased Ising perceptron is in a sense the simplest of the eight cases we consider, in that it requires only n bits for its specification. The memory capacity has recently been studied for this case as well [6, 4].

There is a subtle problem in definitions (1) and (2), peculiar to unbiased Ising (or, more generally, discrete) cases, due to the sign function being undefined for a zero argument. In order to alleviate this problem we will require n_I (and likewise n_H for the two-layer case) to be odd.

We will denote the dichotomization capacity for Ising networks by appending a subscripted ±1, and thus this section deals with $D^*_{\pm 1}(n)$. From simple algebraic or geometric arguments we can show that

$$D^*_{\pm 1}(n) = 2^n \,. \tag{10}$$

The argument is that there are 2^{n_I} possible J_{OI_j}, each corresponding to a hypercube vertex and each realizing a distinct dichotomy. This follows algebraically from the unbiased version of equation (1) since the set of inputs giving positive output consists of all those vertices within Hamming distance $n_I/2$ of the chosen one. This set necessarily changes when the chosen vector is replaced. Geometrically, we see that the separating hyperplane contains the origin and is perpendicular to the vector from there to the chosen vertex. Choosing a new vertex rotates the plane through hypercube vertices and thus produces a distinct dichotomy.

Due to the fact that every distinct network produces a different dichotomy, the enumeration bound is attained. We see that although the unbiased Ising perceptron has extremely limited capacity, it functions optimally given its architectural constraints. This is in consonance with the finding [4], that the unbiased Ising perceptron is the most efficient from the memory capacity point of view.

9 The biased Ising perceptron

From comparison with equation (9) one might expect $D_{\pm 1}(n)$ to equal $D^*_{\pm 1}(n+1) = 2^{n+1} = 2 \cdot 2^n$. We may obtain higher capacity if the bias is allowed dynamic range greater than

±1. However, we have not considered the quantization of the bias term, and have implicitly taken it to be continuous valued. We will now show that there are a finite number of distinct bias values, even if *a priori* we make no discreteness assumptions. The sum in equation (1) can only take the $n_I + 1$ values between $-n_I$ and $+n_I$ in steps of two

$$\sum_j J_{OI_j} I_j = -n_I, -n_I + 2, -n_I + 4, \cdots, n_I - 4, n_I - 2, n_I \,. \tag{11}$$

Thus 'different' biases, which are between the same two values of the sum, result in the same partitioning of the input space, and are thus equivalent. Without limiting generality we can chose to use the $n_I + 2$ integral values $-n_I - 1, -n_I + 1, -n_I + 3, \cdots, n_I - 3, n_I - 1, n_I + 1$; moreover, the values $-n_I - 1$ and $n_I + 1$ give the same dichotomies for all J_{OI_j} values (the totally positive and totally negative ones respectively) , and thus should be enumerated separately. Thus we chose the n_I bias values

$$\theta_O = -n_I + 1, -n_I + 3, \cdots, n_I - 3, n_I - 1 \,. \tag{12}$$

Building on the arguments of the previous section, we find that

$$D_{\pm 1}(n) = 2^n n + 2 \,, \tag{13}$$

since for each of the 2^n unbiased perceptrons, we can supply n different biases, and then we must count the final two as remarked above. Note that once again the enumeration bound is attained, since each perceptron counted contributes a distinct dichotomy. It is also interesting to observe that the ratio between the capacity of equation (13) and that of equation (10) is asymptotically linear in n, and not exponential as it is for the continuous valued case. This means that bias is much less successful in enlarging the capacity in this case. This is due to the relatively few distinct values the bias can take.

10 Other constraints

Comparing the results of sections 7 and 8, we see a most interesting difference in behavior between the the Ising perceptron $D^*_{\pm 1}(n) = 2^n$ and the continuous valued perceptron $D^*(n) = 2^{\gamma n^2}$. For the memory capacity, the functional form is the same, namely $P = \alpha N$, with only the coefficients differing, $\alpha = 0.832$ for the Ising case [6, 4], and $\alpha = 2$ for the continuous case [2]. Gutfreund and Stein [4] have studied the transition between Ising and continuous cases by allowing successively more discrete values $J_{OI_j} = 0, \pm\frac{1}{L}, \pm\frac{2}{L}, \cdots, \pm 1$. For the simplest case, the *diluted Ising* perceptron, there are three discrete values $J_{OI_j} = 0, \pm 1$ and they found $\alpha = 1.174$; for the asymmetric constraint $J_{OI_j} = 0, 1$ the result is $\alpha = 0.59$.

This diluted Ising case can be studied here as well. The dichotomization capacity of the unbiased diluted Ising perceptron is given by

$$D^*_{0,\pm 1}(n) = \frac{1}{2}\left(3^n - (-1)^n\right) \tag{14}$$

and that of the biased version by

$$D_{0,\pm 1}(n) = 3^n n - 3^{n-1} + 2 = 2\left(3^{n-1} n + 1\right) . \tag{15}$$

(Once again the biased to unbiased ratio is linear in n.) We will derive the result for the unbiased case, leaving the extension to the biased case to the reader. We require the total number of *non-zero* couplings to be odd, and note that as for the Ising case the enumeration bound will be attained. This follows from the fact that inverting a non-zero coupling will change an Ising perceptron, while changing which couplings are zero, causes previously uninfluential inputs to effect the output. We thus conclude that

$$D^*_{0,\pm 1}(n) = \sum_{\substack{z=1 \\ z \text{ odd}}}^{n} \binom{n}{z} 2^z \tag{16}$$

Now, from the binomial theorem

$$\sum_{z=0}^{n} \binom{n}{z} (\pm 2)^z = (1 \pm 2)^n$$

subtracting the negative case from the positive we obtain (14).

We can endeavor to extrapolate these results further. With two possible coupling values the leading term is 2^n ($2^n n$ with threshold) and with three possible values we have $3^n = 2^{(\log_2 3)n}$ ($3^n n$ with threshold). With d possible values we must have *at most* $d^n = 2^{(\log_2 d)n}$ behavior ($d^n n$ with threshold). Thus, the couplings can not be essentially continuous until at least $d^n \sim 2^{(n^2)} = (2^n)^n$ ie. when $d \sim 2^n$, as far as the dichotomization capacity is concerned. This should be contrasted with the small number of values required for the memory capacity to resemble that of the continuous case [4].

Another constraint of importance is that of coupling non-negativity, which we will denote by a superscripted plus sign on $D(n)$. The simplest case is that of the binary perceptron [4] with $J_{OI_j} = 0, 1$. In order to derive its dichotomization capacity note that the trivial case of a perceptron with all couplings of strength one $J_{OI_j} = 1$ gives $D^*_1(n) = 1$ (for odd n) and $D_1(n) = n + 2$. The derivation is now similar to, but simpler than, that of $D_{0,\pm 1}(n)$. We find, whenever permitted,

$$D^*_{0,1}(n) = \sum_{\substack{z=1 \\ z \text{ odd}}}^{n} \binom{n}{z} = 2^{n-1} \tag{17}$$

and (for $n > 1$)

$$D_{0,1}(n) = \sum_{\substack{z=1 \\ z \text{ odd}}}^{n} \binom{n}{z} (z + 2) = 2^n + 2^{n-2} n \tag{18}$$

where in the latter case, we *do* allow negative biases. We note that $D^*_{0,1}(n) = \frac{1}{2}D^*_{\pm 1}(n)$, and $D_{0,1}(n) < D_{\pm 1}(n)$. The latter is always true even though it is possible to convert $J = 0, 1$ weights to $J = \pm 1$ ones via the transformation $J_{\pm 1} = 2J_{0,1} - 1$. This conversion does not, however, preserve dichotomies.

Finally, we mention the non-negative continuous valued case. For this case analytic results are not available, and once again we resort to exhaustive enumeration. We have performed this enumeration for small n and obtained table 2. These results will be useful in sections 14 and 15 below. For $n = 6$ we terminated the search after finding 240,000 pattern sets, having no particular need for the exact number, and as computer time was becoming excessive.

n	D^{+*}	D^{+}
1	1	3
2	2	6
3	4	20
4	12	150
5	81	3287
6	1684	$>$ 240,000

Table 2: Dichotomization capacity for non-negative continuous perceptrons.

11 Summary of perceptron capacities for small n

In table 3 we present the $n = 1 \ldots 8$ dichotomization capacity for for the most important perceptron architectures. The second column gives the total number of dichotomies 2^{2^n}, the upper bound on all the capacities. Note that each row is the square of the previous one. The next block details the biased cases in order of decreasing capacity, while the third block similarly contains the unbiased cases. The third and sixth columns contain the capacities for continuous perceptrons (see sections 6 and 7); the fourth and seventh columns represent the capacities for the diluted Ising case (see section 10); and the fifth and eighth columns are the Ising perceptron capacities (see sections 9 and 8). This latter column has entries only for odd n.

We observe that for large n, all of these perceptrons implement a vanishingly small fraction of the total number of functions. Even for the moderately small n values reported here, the differences in orders of magnitude between the different architectures are impressive.

n	2^{2^n}	$D(n)$	$D_{0,\pm1}(n)$	$D_{\pm1}(n)$	$D^*(n)$	$D^*_{0,\pm1}(n)$	$D^*_{\pm1}(n)$
1	4	4	4	4	2	2	2
2	16	14	14	10	4	4	—
3	256	104	56	26	14	14	8
4	65536	1882	218	66	104	40	—
5	$4.295 \cdot 10^9$	94,572	812	162	1882	122	32
6	$1.845 \cdot 10^{19}$	$1.503 \cdot 10^7$	2918	386	94572	364	—
7	$3.403 \cdot 10^{38}$	$8.378 \cdot 10^9$	10,208	898	$1.503 \cdot 10^7$	1094	128
8	$1.158 \cdot 10^{77}$	$1.756 \cdot 10^{13}$	34,994	2050	$8.378 \cdot 10^9$	3280	—

Table 3: The dichotomization capacity of various perceptron architectures for $n = 1 \ldots 8$ (see text).

12 The unbiased Ising two-layer network

In order to be able to implement more dichotomies we now turn to two-layer networks. We start with the two-layer unbiased Ising network, which naturally can implement only hemichotomies. For the same reasons given in section 8 we will only consider odd n_I and n_H.

With this architecture, we can at last realize the two nonlinearly separable hemichotomies in three dimensions, by using three hidden units. More generally, we shall now show that given a large enough number of hidden units all hemichotomies can be attained (the equivalent of universal realization for this case). Let us consider the matrix M of outputs of all possible unbiased Ising perceptrons for all possible input patterns. Since the set of all such perceptrons is in one-to-one correspondence with the set of all input patterns, this is a symmetric 2^n by 2^n matrix of ±1s. We can explicitly give the matrix elements as follows

$$M_{i,j}(n) \equiv \begin{cases} 1 & \mathcal{B}(i \,\mathrm{xor}\, j) > \frac{n}{2} \\ -1 & \mathcal{B}(i \,\mathrm{xor}\, j) \leq \frac{n}{2} \end{cases} \qquad i,j = 0 \ldots 2^n - 1 \tag{19}$$

where $\mathcal{B}(i)$ gives the number of ones in the binary representation of i, and xor is the bitwise exclusive or operator. We display the first few such matrices graphically in figure 1.

From inspection we see that $M(n)$ can be decomposed thus :

$$M(n) = \begin{pmatrix} A & B \\ B & A \end{pmatrix} .$$

For odd n, the A matrices are exactly the preceding matrix $M(n-1)$, and B corresponds to the negative of this same matrix after row or column reflection. The even n matrices have $B = M(n-1)$, while A is more complex. This A is composed of reflected 'positive-diagonal' matrices and of $M(\nu)$ of dimensions up to $\frac{n}{2}+1$. We will now show, by induction, that the even n matrices are full rank, while those with odd n are then obviously of half rank.

The proof goes as follows. The first even case is $n = 2$ (see figure b), and $M(2)$ is obviously full rank due to its diagonal structure. The first nontrivial even case, $n = 4$ (see figure d), is also easily shown to be full rank by the following argument. The upper half and lower halves must be of rank 8 due to the positive-diagonal blocks (as above). Thus if there is linear dependence it must involve both halves. Now lets concentrate on the upper right quadrant. Any linear combination of its rows must give a vector which is antisymmetric about its center $v_{9-i} = -v_i$. For a sum involving this type of vector and a sum of rows of the lower right quadrant to give zero, we must takes combinations in the lower right quadrant which build this antisymmetry as well. When doing this the lower left quadrant must sum to zero, and thus can not cancel the sum of upper left quadrant rows, which, being full rank, can not sum to zero.

Now let us demonstrate the induction step by looking at $M(6)$ (see figure e). Considering this matrix as being composed of sixteen submatrices, eight of these are exactly $M(4)$, the four on the minor diagonal are inverted negations of $M(4)$, and the four on the main diagonal are the reflected positive-diagonal matrices. It is obvious that no linear dependence can be totally composed of rows in any *strip* of quarter height, and so we consider mixing two such strips. Any pair of strips we chose to combine will involve mixing $M(4)$ (or its inverted negation) with the positive-diagonal submatrix. There are certainly sums mixing rows of these two submatrices which give zero. However, as is obvious from the figure, the same linear combination of rows would be required to give zero when these submatrices appear in opposite order. This possibility is ruled out by the fact that $M(4)$ is full rank. One can similarly see that mixing of three or even all four strips will still not lead to linearly dependent rows, due to the linear independence of the rows of $M(4)$. One can convince oneself that this same argument, that the linear dependence of rows of $M(n)$ would imply the dependence of rows of $M(n-2)$, $M(n-4)$, …, or $M(4)$ holds for all even n, thus completing the proof by induction.

Now we concentrate on the odd n cases, and chose a linearly independent set containing half of the perceptrons. Since there are exactly 2^{n-1} linearly independent perceptrons, this set spans all possible even functions, and thus *any* even function, in particular any even ± 1 function (hemichotomy), can be expanded in terms of these perceptrons. Thus a two layer network with 2^{n-1} hidden units each fed with ± 1 couplings, and a single output fed with possibly continuous couplings, is a universal realizor for hemichotomies. However, these continuous couplings can be taken to be integer valued in the range $-2^{n-1} \ldots 2^{n-1}$, since their explicit calculation involves projecting a vector of ± 1 of length 2^{n-1} on the basic Ising perceptron outputs. Furthermore, integer couplings can always be emulated by multiple identical Ising hidden units. We thus reach the conclusion that two layer unbiased networks with Ising couplings and enough hidden units, can realize arbitrary hemichotomies. There will be at most 2^{n-1} groups of 2^{n-1} hidden units, but in practice less units per group

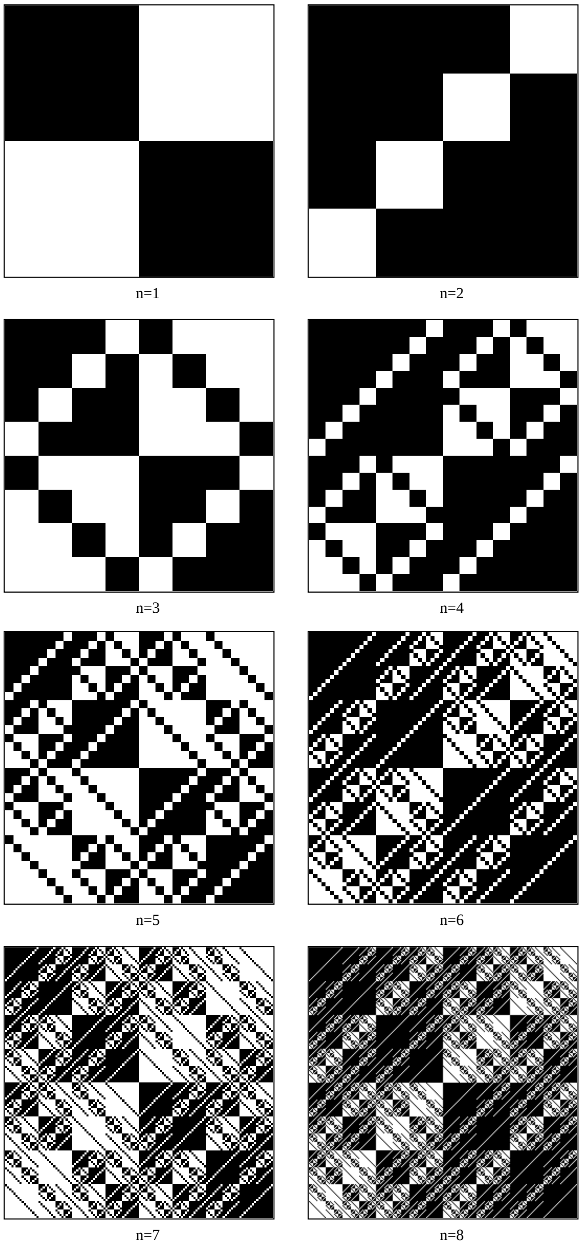

Figure 1: The first few M matrices for both even and odd n_I. Positive matrix elements are depicted as black squares, and negative elements as white ones. The recursive nature of the matrices is evident.

are required and common factors can be removed. In any case this construction, while necessarily producing a solution network, does not usually produce the smallest such.

Let us now turn to the dichotomization capacity $D^*_{\pm 1}(n_I, n_H)$ and to the n_H required for $D^*_{\pm 1}(n_I, n_H) = 2^{2^{n_I - 1}}$. Given n_H hidden units there are a priori $n_I n_H + n_H = n_H(n_I + 1)$ independent couplings, and thus $2^{n_H(n_I+1)}$ different networks. Before attempting to use this enumeration for bounding purposes, there are certain symmetries which can be exploited. First, if for a given set of patterns we find a solution network, there are automatically 2^{n_H} solutions, due to the degeneracy of all networks related by the gauge transformation

$$J_{H_k,I_j} \to -J_{H_k,I_j} \qquad \text{and} \qquad J_{O,H_k} \to -J_{O,H_k} \qquad \text{for any } j. \tag{20}$$

Thus, without limiting generality, we can *chose* all the hidden to output couplings to be +1. Since the output unit in such a network is +1 only when the majority of the hidden units are +1, this type of network is often called a *committee machine* [8]. The gauge symmetry implies that if there is a two-layer unbiased Ising solution, then there is a committee machine solution. Thus it is sufficient to consider $n_I n_H$ independent couplings.

Due to the gauge symmetry, we actually put only $n_I n_H$ bits into the network, and we wish to retrieve $2^{n_I - 1}$ bits (the information about whether a pattern or its negative gives +1). This leads to the following information theory bound on the number of hidden units of a universal hemichotomy realizor

$$n_H \geq \frac{2^{n_I - 1}}{n_I} \tag{21}$$

Thus, eg. for $n_I = 3$, 5, 7, 9, and 11 we require (taking the next odd number) $n_H \geq$ 3, 7, 11, 15, and 95 respectively.

There is yet another type of symmetry transformation which further reduces the number of distinct networks, permutation symmetry. If we find a solution network for which the couplings feeding hidden unit k are different from those feeding hidden unit k', then we can always obtain a new solution by interchanging all the couplings feeding these two hidden units. In general, since the order of the units in the hidden layer is arbitrary, we can perform any permutation of the hidden layer indices, however not every such permutation leads to a distinct network. Since the hidden units are indistinguishable and any number of them can have the same couplings, the hidden units are what physicists call *bosons*. We wish to determine the number $T^*_{\pm 1}(n_I, n_H)$ of different configurations of n_H units which can be in any of 2^{n_I} states (see equation 10). Using the basic formula of Bose-Einstein statistics, we find

$$T^*_{\pm 1}(n_I, n_H) = \binom{2^{n_I} + n_H - 1}{2^{n_I} - 1} = \binom{2^{n_I} + n_H - 1}{n_H} \tag{22}$$

Thus, for example for the $n_I = 5$, $n_H = 3$ architecture, instead of $2^{18} = 262{,}144$ different networks, we need only consider $\binom{32+3-1}{3} = 5984$. Even were each of these to realize a different set of patterns, we see that not all the $2^{2^4} = 65{,}536$ hemichotomies are attainable. In general, the enumeration bound tells us that with n_H hidden units we can implement no more than $T^*_{\pm 1}(n_I, n_H)$ hemichotomies, and in order to implement all possible hemichotomies, we must have

$$2^{2^{n-1}} < T^*_{\pm 1}(n_I, n_H) = \binom{2^{n_I} + n_H - 1}{n_H} . \tag{23}$$

For $n_I = 3$, 5, 7, 9 and 11 this means that n_H must be at least 3, 5, 15, 55 and 217 respectively. For large n this enumeration bound is stricter than the information theoretic one given above.

In order to directly determine the dichotomization capacity of unbiased two-layer Ising networks, we performed exhaustive search for small systems ie. n_I= 3, 5, 7. For each n_H tested, we constructed all $T^*_{\pm 1}(n_I, n_H)$ distinct networks, presented each with half of the possible patterns and determined which give +1 outputs, and then tested this pattern set against a list of pattern sets already found. In this way we found the results given in table 4.

	$n_I = 3$	$n_I = 5$
$n_H = 1$	8	32
$n_H = 3$	16	2112
$n_H = 5$		14999
$n_H = 7$		58412
$n_H = 9$		64916
$n_H = 11$		65536

Table 4: Dichotomization capacity for small unbiased Ising two-layer networks, as determined using exhaustive enumeration.

We see that for $n_I = 3$, three hidden units indeed suffice; however for $n_I = 5$, $n_H = 11$ and the enumeration and information theoretic bounds are evidently not very tight.

13 The biased Ising two-layer network

The biased Ising two-layer network with enough hidden units is a universal realizor. According to the usual proof, one realizes a given dichotomy by allocating hidden units for all 2^n possible input patterns. This result, as Minsky and Papert point out [7], is trivial, implying simply that every boolean function has a disjunctive normal form. Many researchers have assumed that 2^n hidden units are necessary, while actually, we can as easily construct any dichotomy with half as many hidden units — 2^{n-1} in all. This is due to the fact that patterns without hidden units will trivially give some output. Thus if less than half of the patterns in a dichotomy correspond to positive outputs, we need only allocate hidden units for them, and set the bias for all other patterns to give -1. If more than half do, we allocate hidden units only for the negative patterns and set the output bias such that unallocated patterns give positive output.

We need not, of course, restrict ourselves to odd n_I and n_H here. Three symmetries can be called into play – bias quantization (see equation 12), gauge invariance (as in subsection 12); and permutation symmetry (explained in the same subsection). Employing once again Bose-Einstein statistics we are lead to

$$T_{\pm 1}(n_I, n_H) = \binom{2^{n_I} + n_H - 1}{n_H} (n_I + 2)^{n_H} n_H + 2\,. \tag{24}$$

We can now determine the dichotomization capacity by exhaustive enumeration. We sequentially produce the $T_{\pm 1}(n_I, n_H)$ distinct networks, present all possible patterns to each such network, and determine the dichotomies produced. The results of such an experiment are displayed in table 5 with the biased Ising perceptron appearing as $n_H = 1$.

	$n_I = 2$	$n_I = 3$	$n_I = 4$
$n_H = 1$	10	26	66
$n_H = 2$	16	146	1298
$n_H = 3$		250	9858
$n_H = 4$		256	38068
$n_H = 8$			65536

Table 5: Dichotomization capacity for small biased Ising two-layer networks, as determined using exhaustive enumeration.

We see that only when $n_H = 2^{n_I - 1}$ can we actually realize *all* 2^{n_I} dichotomies. This is interesting since the parity function can always be realized with n_I hidden units (all with the equal couplings from input to hidden units, but with different hidden unit biases and alternating signs to the output unit). Thus the parity function is a comparably simple function for *biased* networks (due perhaps to its high symmetry).

14 The unbiased continuous two-layer network

An interesting fact about the unbiased continuous case is that adding a second hidden unit does not increase the dichotomization capacity over that of an unbiased perceptron

$$D^*(n_I, 2) = D^*(n_I, 1) = D^*(n_I) \tag{25}$$

a third hidden unit is required. The reason is that a two hidden unit network maps the input space onto the square, which must then be dichotomized by an unbiased perceptron. We recall from table 3 that there are only four such dichotomizations of the square, all of which can be realized by a perceptron with one coupling equal to zero. Thus any two hidden unit unbiased network is equivalent to one with one of the hidden units not effecting the output at all, which can not dichotomize better than a perceptron.

Now let us estimate the number of distinct networks for this case. The first symmetry for the continuous cases is that we can take all hidden to output couplings to be non-negative. Thus we employ the non-negative unbiased perceptrons enumerated in table 2. The input to hidden layer couplings are the continuous unbiased perceptrons, and we can exploit the same gauge symmetry as for the Ising cases (leading to Bose-Einstein statistics). Without further ado we present our estimate for the number of distinct networks,

$$T^*(n_I, n_H) = \binom{D^*(n_I) + n_H - 1}{n_H} D^{+*}(n_H) \, . \tag{26}$$

This is actually a gross overestimation, since not all hidden layer vectors are accessed.

The enumeration bound for n_H is

$$2^{2^{n_I - 1}} \leq T^*(n_I, n_H) \tag{27}$$

thus predicts for $n_I = 3$, 4, 5 and 6 that n_H must be at least 1, 2, 2, 2 and 3 respectively. However we know that for three inputs, one, and thus two, hidden units are insufficient (since only 14 of the 16 dichotomies are linearly separable).

For small systems, exhaustive enumeration can be performed here as well. We can produce all distinct two layer networks by employing the continuous unbiased perceptrons found in section 7. The results are shown in table 6. The two $n_I = 5$ pattern sets that can not be realized with three hidden units are the parity function and its inverse. We see that the parity function is very difficult to realize with *unbiased* threshold units. We could not afford to search through all combinations for four hidden units, but searches of likely combinations did not turn up solutions. We know that for the special case of Ising couplings eleven hidden units suffice, so that some n_H between four and eleven must be sufficient.

	$n_I = 2$	$n_I = 3$	$n_I = 4$	$n_I = 5$
$n_H = 1, 2$	4	14	104	1882
$n_H = 3$		16	256	65534

Table 6: Dichotomization capacity for small unbiased continuous two-layer networks, as determined using exhaustive enumeration.

15 The biased continuous two-layer network

This is certainly the most interesting case from the applications point of view. For the biased Ising two-layer network $n_H = 2^{n-1}$ is sufficient for universal realization. It is quite clear that fewer hidden units should be required here. The question is – how many?

The enumeration bound for this case gives (as should be obvious at this point)

$$2^{2^{n_I}} \leq T(n_I, n_H) = \binom{D(n_I) + n_H - 1}{n_H} D^+(n_H) \tag{28}$$

which merely gives $n_H \geq 1$ at least to $n = 5$. From exhaustive enumeration we find the values given in table 7. Comparing this with table 5 we see that the continuous couplings indeed increase the dichotomization capacity.

	$n_I = 2$	$n_I = 3$	$n_I = 4$
$n_H = 1$	14	104	1882
$n_H = 2$	16	254	41614
$n_H = 3$		256	65536

Table 7: Dichotomization capacity for small biased continuous two-layer networks, as determined via exhaustive enumeration.

16 Summary

We have studied the dichotomization of binary sets by various architectures of feedforward neural networks. We have reached several conclusions, notably explicit formulae for

- biased and unbiased Ising perceptrons,
- $0, 1$ perceptrons,
- $0, \pm 1$ perceptrons,

an existence proof that states that

- all hemichotomies are realizable by an unbiased one hidden layer Ising network,

and a limitation

- that unbiased continuous two-layer networks with two hidden units do not perform better than unbiased continuous perceptrons.

We have utilized symmetries, namely that the network is not changed by

- bias quantization,
- gauge transformations, and
- hidden unit permutation,

in order to reduce the number of distinct networks, for purposes of

- enumeration bounds,
- exhaustive enumeration for small systems.

However, we have left several interesting questions open. The major ones are

- Are all values $1 \cdots 2^{2^n}$ valid dichotomization capacities?
- What is the capacity of the continuous perceptron, in particular, what is the value of γ of equation (6)?
- How many hidden units are required for the continuous two layer network to universally realize?
- How many unbiased hidden units are required to realize the (five bit) parity function?

References

[1] Cover T.M. 1965. Geometric and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Trans. El. Comp.* **EC-14**, 326-334.

[2] Gardner E. 1988. The space of interactions in neural network models. *J. Phys. A: Math. Gen.* **A21**, 257-270.

[3] Gardner E., and Derrida B. 1988. Optimal storage properties of neural network models. *J. Phys. A: Math. Gen.* **A21**, 271-284.

[4] Gutfreund H., and Stein Y. 1990. *J. Phys. A: Math. Gen.* **23**, 2613-2630.

[5] Hopfield J.J. 1982. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79**, 2554-2558.

[6] Krauth W. and Mézard M. 1989. *J. Physique (Paris).* **50**, 3057-3062.

[7] Minsky M. and Papert S. 1969. *Perceptrons.* MIT Press, Cambridge, Mass.

[8] Mitchison G.J., and Durbin R.M. 1989. Bounds on the learning capacity of some multi-layer networks. *Biol. Cybern.* **60**, 345-356.

[9] Muroga S. 1965. Generation and Asymmetry of Self-Dual Threshold Functions. *IEEE Trans. El. Comp.* **EC-14**, 125-136; Lower bounds of the number of threshold functions and a maximum weight. *IEEE Trans. El. Comp.* **EC-14**, 136-148.

[10] Muroga S., and Toda I. 1966. Lower bound of the number of threshold functions. *IEEE Trans. El. Comp.* **EC-15**, 805-806.

[11] Muroga S., Toda I., and Takasu S. 1961. Theory of majority decision elements. *Jour. Franklin Inst.* **271**, 376-418.

[12] Schläfli L. 1950. *Gesammelte Mathematische Abhandlungen I.*, pp. 209-212. Verlag Birkhaäuser, Basel, Switzerland.

[13] Seung H.S., Sompolinsky H., and Tishby N. 1991. Statistical mechanics of learning from examples. *Phys. Rev.* **A 45**, 6056-6091.

[14] Smith D.R. 1966. Bounds on the number of threshold functions. *IEEE Trans. El. Comp.* **EC-15**, 368-369.

[15] Steiner J. 1826. *fur Math. (Crelle)* **1**, 349-364.

[16] Winder R.O. 1965. Enumeration of seven-argument threshold functions. *IEEE Trans. El. Comp.* **EC-14**, 315-325.

[17] Winder R.O. 1966. Partitions of N-space by hyperplanes. *J. SIAM Appl. Math.* **14**, 811-818.

[18] Yajima S., and Ibaraki T. 1965. A lower bound of the number of threshold functions. *IEEE Trans. El. Comp.* **EC-14**, 926-929.