Perceptrons

H.J.M. Peters

Department of Quantitative Economics, University of Limburg, Maastricht

1 Introduction

A *perceptron* is a neural network that is trained under supervision. This means that the perceptron's decisions during training are compared with the desired decisions; based on this comparison the internal weights of the network are adjusted until a satisfactory result is reached. Not only the (rate of) convergence of this learning process is important, but also the problem of representation: Which (practical) problems can be written in a form suited to apply the perceptron; that is, which problems can be written as a linear threshold function? What are the implications of such representations for the efficiency and the rate of convergence of the learning process, and the necessary storage capacity? The by now classical work of Minsky and Papert (1969,1988), *Perceptrons, An Introduction to Computational Geometry*, on which this paper is based, in particular provides a detailed study of representation problems in connection with the perceptron.

The organization of this paper is as follows. Section 2 gives a brief historical account of the development of perceptron theory. In section 3 definitions and a few preliminary results are presented. Section 4 develops some theoretical results which can be seen as exemplary for what perceptrons can and cannot do. Section 5 is on training and convergence; in particular, the basic perceptron convergence theorem is stated and proved. Section 6 contains a few concluding remarks.

2 Historical overview

Perceptrons were introduced by Rosenblatt (1959). In his *Principles of Neurodynamics* (1962) Rosenblatt writes:

> Perceptrons ... are simplified networks, designed to permit the study of lawful relationships between the organization of a nerve net, the organization of its environment, and the "psychological" performances of which it is capable. Perceptrons might actually correspond to parts of more extended networks and biological systems; in this case, the results obtained will be directly applicable. More likely they represent extreme simplifications of the central nervous system, in which some properties are exaggerated and others suppressed. In this case, successive perturbations and refinements of the system may yield a closer approximation.

Thus, the perceptron can be regarded as a highly simplified model of the human brain or at least part of it. Rosenblatt's book revived interest in neural

"connectionistic" networks, but was not the first work in this area. Neurological networks had been introduced and discussed earlier by McCulloch and Pitts in their articles *A Logical Calculus of the Ideas Immanent in Nervous Activity* (1943) and *How We Know Universals* (1947). In these articles network architectures were described which in principle were capable of recognizing spatial patterns in a way invariant under certain groups of geometric transformations. Further, Hebb's book *The Organization of Behavior* (1949) must be mentioned within this development.

Although, in the fifties, some further developments of neural networks occurred, things became quiet by the end of this decade. To a considerable extent, this was due to the success of the serial von Neumann computer. As an aside, note that neural networks were first developed in the forties, at a time when computers hardly existed, and programming languages above a minimal standard did not exist at all; in spite of this, neural networks are now often being offered as an alternative to "old fashioned" programming.

Rosenblatt's perceptron brought new life to an almost extinct area; this perceptron, in all its simplicity, appeared to be capable of "learning" certain things. On the other hand, it turned out that perceptrons were not able to learn certain other things, in spite of all the effort put into extending and refining the training process, and building bigger machines. Namely, most researchers in the field were looking for more general methods which should make the perceptron capable of handling a large(r) class of problems.

This is not true as far as Minsky en Papert in their book *Perceptrons: An Introduction to Computational Geometry* (1969) are concerned. Instead of looking for a method which would work in every possible situation, they provided a mathematical analysis and explanation of the fact that the particular method used by the perceptron performs well in some cases and badly in other cases. Consequently, the book reveals not only the possibilities but also the restrictions of the perceptron; for this reason, the limited interest in neural networks during the seventies has often been ascribed to the publication of this book. In the republication of the book in 1988, Minsky and Papert remark that research in the area of neural networks had come to a halt already at an earlier stage, due mainly to a lack of fundamental theories. Too much (vain) effort had been invested in the simple and somewhat *ad hoc* training process, at the expense of the more important problem of *representation of knowledge*. Indeed, during the seventies research in this last area has expanded enormously.

The present revival in the field of neural networks and, more generally, of parallellism (or "connectionism") is, among other things, perhaps due to the further development of multilayered perceptrons; these perceptrons will not be discussed in this paper (see, however, the contributions by Henseler and by Weijters and Hoppenbrouwers). It should be mentioned that Minsky and Papert have been rather sceptical concerning the possibilities of multilayered perceptrons, which makes the above reproach understandable. At this moment it is not yet clear whether multilayered perceptrons will lead to a breakthrough in parallel computing. The emphasis, however, that Minsky and Papert give to the importance

of fundamental theories of knowledge representation, remains justified.

3 Perceptrons and linear threshold functions

Perceptrons were introduced by Rosenblatt (1959, 1962). The present paper is based on Minsky and Papert (1969). The concept of a perceptron is illustrated by figure 1. Figure 1 shows the principle of parallel computation in general, and of the perceptron in particular. In this figure the general principle of parallel computation is applied to a problem of pattern recognition. The letter "X" is drawn in a plane, which is being scanned by local sensors. The information of these sensors is passed on to functions φ_i, which assign a certain value to it. For example, the plane R may be divided into small squares that are black or white depending on the pattern, in this case the letter "X". The function φ_i assigns a certain value depending on the configuration of white and black squares in its domain, which is the part of R covered by the corresponding sensor. In Ω the values of the φ_i's are combined, leading to a certain value of the function ψ; from this value it may be inferred, for example, that the pattern under consideration is the letter "X", or a cross and not a circle. An essential feature is that by parallel processing a *global* statement is obtained from *local* information.

In a perceptron, the function ψ is a predicate which is itself a linear combination of predicates φ_i. A *predicate* is a *function of subsets of* R that has

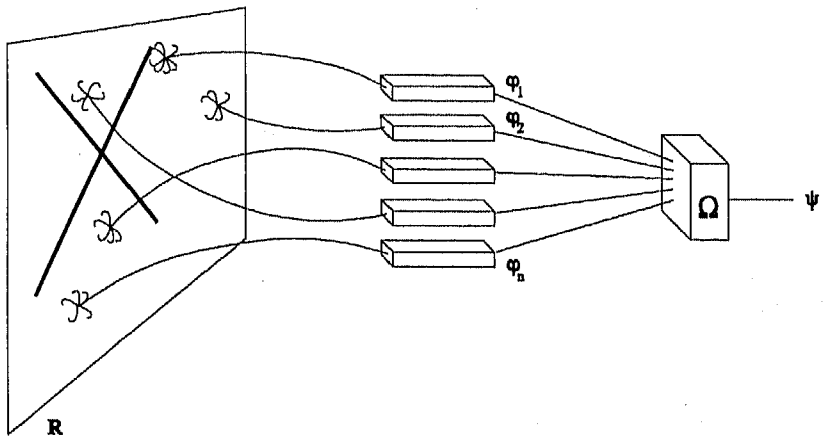

Fig. 1. Parallellism, and the perceptron.

two possible values. We think of these values as representing truth or falseness, and it is customary to associate 1 with "true" and 0 with "false". Let

$\Phi = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ be a set of predicates. The predicate ψ is called *linear* with respect to Φ if there exist a number θ and numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$ such that, for every $X \subset R$, $\psi(X) = 1$ if and only if $\alpha_1\varphi_1(X) + \ldots + \alpha_n\varphi_n(X) > \theta$. The number θ is the *threshold* and the α_i's are the coefficients or *weights*. The predicate ψ is called a *linear threshold function*. More compactly, ψ can be written as

$$\psi(X) = [\sum_{\varphi \in \Phi} \alpha_\varphi \varphi(X) > \theta].$$

Here, $[\ldots]$ is a predicate assigning to the expression between the brackets the value 0 if the expression is false, and 1 if the expression is true.

A perceptron is a device capable of computing all predicates which are linear with respect to some given set Φ of partial predicates. For instance, suppose the *retina* R is divided into a (finite) number of squares, and associate with each square i the predicate φ_i: "The square is black". Thus, the predicate φ_i has value 1 if and only if the corresponding square is black. By taking all weights α_i equal to 1, and θ equal to 25, the linear threshold function ψ assigns value 1 to a pattern X in R if and only if X occupies more than 25 squares. Furthermore, in a perceptron the weights are adjustable by a learning process; see section 5.

In what follows, a special kind of predicate called "mask" plays an important role. Suppose, as above, that the (bounded) retina R is divided into a finite number of squares. We identify these squares with points. With each point p of R the predicate φ_p is associated, defined by $\varphi_p(X) := [p \in X]$ for every $X \subset R$. More generally, with each subset A of R a predicate $\varphi_A : X \mapsto [A \subset X]$, the *mask* of A, is associated. These masks can be used in definitions of predicates. For instance:

$[X$ contains at least M points$] = [\sum_{p \in R} \varphi_p(X) > M - 1]$,

$[X$ contains more points than $Y] = [\sum_{p \in R} \varphi_p(X) - \varphi_p(Y) > 0]$,

$[X$ is for the larger part located in the right half of $R] =$
$[\sum_{p \in R_{\text{right}}} \varphi_p(X) - \sum_{p \in R_{\text{left}}} \varphi_p(X) > 0]$.

Suppose R contains n points. Then each predicate, being a function that assigns 0 or 1 to every subset of R, can be identified with a vector in 2^n-dimensional Euclidean space (with coordinates in $\{0, 1\}$). It is easy to verify that the 2^n masks form a basis of this space; consequently, each predicate can be written as a unique linear combination of masks. In particular, this implies the following theorem.

Theorem 1 *If the retina R is finite, then each predicate is a linear threshold function with respect to the set of all masks.*

Consequently, if R is finite, each predicate can be computed by a perceptron; therefore, problems that can be represented as a predicate can be computed by a perceptron. The performance of a perceptron mainly depends on the following two factors:

- How "local" are the predicates φ_i?

- How many predicates are needed, and what are the proportions of their weights?

Minsky and Papert (1969) distinguish between several measures of "localness" of a predicate. The most important of these are the maximal diameter of the area of R to which the predicate is restricted, and the maximal number of points determining the value of the predicate. We will confine our attention to the latter measure. In order to give a formal definition, let the *support* $S(\varphi)$ of an arbitrary predicate φ be the smallest subset S of R with $\varphi(X) = \varphi(X \cap S)$ for every subset X of R. It is not hard to show that, if the support exists, then it is unique. The cardinality $|S(\varphi)|$ of $S(\varphi)$ is called the *degree* of φ. Predicates with small supports are, generally speaking, too local to be interesting. We are, however, interested in predicates which have R as support, but can be expressed as a combination of predicates with small supports. The *order* of a predicate ψ is the smallest number k such that there is a collection of predicates $\Phi = \{\varphi\}$ with repect to which ψ is a linear threshold function and with

$$|S(\varphi)| \leq k \text{ for all } \varphi \in \Phi.$$

Observe that the order of a predicate ψ does not depend on its specific representation.

Masks have order 1, because for each subset A of R

$$\varphi_A(X) = [\sum_{x \in A} \varphi_x(X) > |A| - 1],$$

i.e., φ_A is a linear threshold function with respect to predicates of degree 1. Note, however, that the degree of φ_A is equal to $|A|$.

An example of a predicate of order 2 is the "counting predicate"

$$\psi^M(X) := [|X| = M],$$

where M is a nonnegative integer at most $|R|$. This can be seen as follows. Assume the points of R are numbered, with masks φ_i for points i and φ_{ij} for two-point sets consisting of points i and j $(i, j = 1, 2, \ldots)$. Then

$$\psi^M(X) = [(2M-1)\sum_{i \in R} \varphi_i(X) + (-2)\sum_{i<j} \varphi_{ij}(X) > M^2 - 1].$$

For the right hand side of this equality yields

$$[(2M-1)|X| - |X|(|X|-1) - M^2 > -1] = [(|X| - M)^2 < 1],$$

which has value 1 if and only if $|X| = M$. This shows that the order is at most 2. That the order is exactly 2 follows from corollary 3 in the next section. The counting predicate is an example of a predicate where a perceptron would perform quite well. The number of local predicates is relatively small, namely $|R| + \frac{1}{2}|R|(|R|-1)$, and the weights are not too large so that we can expect a reasonable rate of convergence during the training phase.

4 Easy and difficult predicates

Some predicates are "difficult" in the sense that they are of high order and that the weights in a representation are large. It is consequence of theorem 1 that if the order of a predicate is equal to k then the predicate can be written as a linear threshold function with respect to the set of masks of maximal degree k; namely, a predicate of order k can be written as a linear threshold function of predicates of degree at most k, and these can be written as linear threshold functions of masks of degree at most k. Consequently, to find the order of a predicate one only needs to consider representations in terms of masks.

It is not always clear at first sight whether the order of a predicate is small or big. For example, the interesting predicate which tells us whether a certain pattern in R is convex turns out to be of order at most 3, whereas the predicate which recognizes connected patterns is of order $|R|$. In this section, among other things, these statements will be proved together with a more general result, the *group invariance theorem*. This theorem applies to predicates which are invariant under certain permutations of the retina (for instance, the exact location of a pattern on the retina does not influence its convexity or connectedness), and states that the weights in a linear threshold function representation are independent of such permutations.

Throughout it is assumed that the retina R is a finite approximation of a (bounded) subset of the Euclidean plane—for instance, think of the page in front of you as being divided into a finite number of small squares. A subset X of R is *convex* if with each pair of points in X also all points on the connecting line segment are in X. (Of course, some caution is in order here because R is finite, but this caution is presumed from now on.) Figure 2 shows some examples of nonconvex sets. Observe that in a nonconvex set X there is always a pair of points a, b of which the midpoint is not in X. Based on this observation, we can define

$$\psi_{\text{CONVEX}}(X) = [\sum \varphi_{\{x_i,x_j,x_k\}}(X) - \varphi_{\{x_i,x_k\}}(X) > -1]$$

as the predicate recognizing convexity of X. Here, summation is over all triples x_i, x_j, x_k with x_j the midpoint of the other two points. Obviously, the order of ψ_{CONVEX} is at most three.

Before continuing with more "difficult" predicates we first formulate and prove the group invariance theorem. By way of an illustration, suppose we wish a predicate to recognize the letter "A" no matter where it is located on the retina. Then this predicate should not depend on certain permutations of the retina, e.g., certain translations. In order to formalize this notion, the concept of a group is important. Suppose G is a set with an operation under which that set is closed. Thus, denoting the operation by juxtaposition, we have $gh \in G$ for all $g, h \in G$. Now G is called a *group* if, additionally, the following conditions are satisfied:

(i) There is an element of unity, $e \in G$, with $eg = ge = g$ for all $g \in G$.
(ii) Each element $g \in G$ has an inverse element $g^{-1} \in G$ with $gg^{-1} = g^{-1}g = e$.
(iii) The group operation is associative: $(gh)i = g(hi)$ for all $g, h, i \in G$.

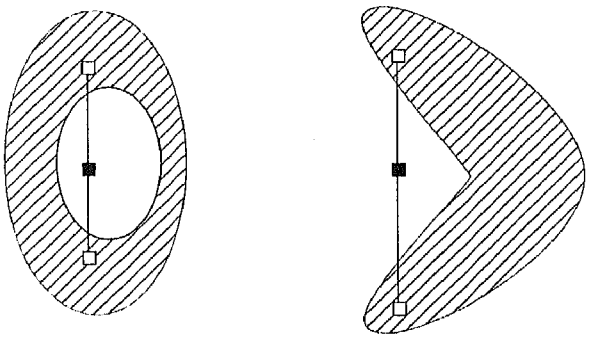

Fig. 2. Nonconvex sets.

In the present context the group of all permutations P of the finite retina R is of interest. For a subgroup G of P, we say that two *subsets* X and Y of R are *G-equivalent*, denoted by $X \equiv_G Y$, if there is an element g in G for which $X = g(Y)$. For instance, if G consists of all horizontal translations of R (think of R as transformed to a cylinder by gluing the left and right ends together, for instance), then the letter "A" somewhere on R is equivalent to the letter "A" shifted over any distance to the left or right. It is easy to verify that $\equiv_G$ is indeed an equivalence relation in the usual mathematical meaning of the word. We further say that two *predicates* φ and φ' are *G-equivalent*, denoted by $\varphi \equiv_G \varphi'$, if there is an element g in G for which $\varphi(g(X)) = \varphi'(X)$ for every $X \subset R$. Also this defines an equivalence relation in the usual sense. In the example above, the predicate recognizing an "A" located at certain fixed coordinates is equivalent under the group G of horizontal translations to predicates recognizing the "A" located at different horizontal coordinates. Finally, we say that the set of predicates Φ is closed under the group G if for every φ in Φ and g in G the predicate φg is also in Φ. For instance, the set of predicates such that each one recognizes the "A" at a different horizontal location is closed under the group of all horizontal translations. Now the *group invariance theorem* can be stated.

Theorem 2 *Let G be a subgroup of the group P of permutations of the finite retina R, and let Φ be a set of predicates closed under G. Let the predicate ψ be a linear threshold function with respect to Φ that is invariant under G. Then there exists a linear representation of ψ,*

$$\psi = [\sum_{\varphi \in \Phi} \beta(\varphi)\varphi > \theta]$$

for which the coefficients depend only on the equivalence classes of the predicates in Φ, that is, $\beta(\varphi) = \beta(\varphi')$ whenever $\varphi \equiv_G \varphi'$.

Proof Let ψ have a linear representation $[\sum_{\varphi \in \Phi} \alpha(\varphi)\varphi(X) > 0]$. (This is without loss of generality, for if the threshold is unequal to 0 we can always

normalize by adding the predicate $\varphi_\emptyset$ with constant value 1 to Φ. At the end of the proof we can drop this additional predicate again.) For any $g \in G$, the map $\varphi \mapsto \varphi g$ is a bijection on Φ, so that

$$\sum_{\varphi \in \Phi} \alpha(\varphi)\varphi(X) = \sum_{\varphi \in \Phi} \alpha(\varphi g)\varphi g(X)$$

for all X, because the same numbers are added in both sums. Let X be a subset of R with $\psi(X) = 1$. Then, for each $g \in G$, by G-invariance of ψ,

$$\sum_{\varphi \in \Phi} \alpha(\varphi g)\varphi g(g^{-1}(X)) > 0,$$

and therefore

$$\sum_{\varphi \in \Phi} \alpha(\varphi g)\varphi(X) > 0.$$

Summing over all g in G and interchanging summation signs, we obtain

$$\sum_{\varphi \in \Phi} (\sum_{g \in G} \alpha(\varphi g))\varphi(X) > 0$$

which can be written as

$$\sum_{\varphi \in \Phi} \beta(\varphi)\varphi(X) > 0,$$

with $\beta(\varphi) := \sum_{g \in G} \alpha(\varphi g)$ for all $\varphi \in \Phi$. The same argument for an X with $\psi(X) = 0$ will show that

$$\sum_{\varphi \in \Phi} \beta(\varphi)\varphi(X) \leq 0.$$

Combining the two inequalities yields

$$\psi(X) = [\sum_{\varphi \in \Phi} \beta(\varphi)\varphi(X) > 0].$$

Finally, suppose that $\varphi \equiv_G \varphi'$, and let $h \in G$ with $\varphi = \varphi' h$. Then

$$\beta(\varphi) = \sum_{g \in G} \alpha(\varphi g) = \sum_{g \in G} \alpha(\varphi' h g) = \sum_{g \in G} \alpha(\varphi' g) = \beta(\varphi')$$

where the third equality derives from the fact that the bijection $g \mapsto hg$ simply permutes the order of adding the same numbers. This concludes the proof. □

A first consequence of theorem 2 is the following corollary.

Corollary 3 *Let G be a group of permutations of R with the property that for any pair of points p, q of R there is a $g \in G$ with $g(p) = q$. Then the only first-order predicates invariant under G are $\psi(X) = [|X| > m]$, $\psi(X) = [|X| \geq m]$, $\psi(X) = [|X| < m]$, and $\psi(X) = [|X| \leq m]$, for some m.*

Proof Let $p, q \in R$, $X \subset R$, and let $g \in G$ with $g(p) = q$. Then $\varphi_p(X) = \varphi_q(g(X))$, so $\varphi_p \equiv_G \varphi_q$. Therefore, in view of theorems 1 and 2 we may assume

$$\psi(X) = [\sum_{p \in X} \alpha \varphi_p(X) > \theta],$$

for a first-order predicate ψ invariant under G. For $\alpha > 0$ this is equivalent to

$$\psi(X) = [|X| > \theta/\alpha].$$

The other predicates are obtained for $\alpha < 0$ or $\alpha = 0$ and by rewriting. □

Another consequence of the group invariance theorem concerns the following predicate

$$\psi_{\text{ODD}}(X) = [|X| \text{ is an odd number}].$$

We consider this predicate because it illustrates the mathematical methods used and the kind of questions they enable to discuss. It turns out that this predicate is of maximal order:

Theorem 4 *ψ_{ODD} is of order $|R|$.*

Proof Obviously, ψ_{ODD} is invariant under the group of all permutations. Suppose the order of ψ_{ODD} is equal to m. By theorems 1 and 2 it can be written as

$$[\sum_{j=0}^{m} \alpha_j (\sum_{\varphi \in \Phi_j} \varphi(X)) > 0]$$

where Φ_j contains all masks of degree j. (The threshold can be taken equal to 0 without loss of generality.) Observe that, for every j

$$\sum_{\varphi \in \Phi_j} \varphi(X) = \binom{|X|}{j} = \frac{|X|(|X|-1)\ldots(|X|-j+1)}{j!}$$

which is a polynomial of degree j in $|X|$. It follows that

$$\sum_{j=0}^{m} \alpha_j (\sum_{\varphi \in \Phi_j} \varphi(X))$$

is a polynomial of degree at most m in $|X|$, say $P(|X|)$.

Consider a sequence $X_0, X_1, \ldots, X_{|R|}$ of subsets of R with $|X_i| = i$. Since $P(|X|) > 0$ if and only if $|X|$ is odd,

$$P(|X_0|) \leq 0,\ P(|X_1|) > 0,\ P(|X_2|) \leq 0, \ldots$$

which is only possible if the degree of the polynomial $P(|X|)$ is at least $|R|$. But this implies $m \geq |R|$. □

The following theorem implies that the number of predicates needed in a representation of ψ_{ODD} is large.

Theorem 5 *Suppose ψ_{ODD} is represented as a linear threshold function with respect to a set of predicates Φ containing only masks. Then Φ contains all masks.*

Proof Suppose, to the contrary, that the mask φ_A ($A \subset R$) is not an element of Φ, and that $\psi_{\text{ODD}} = [\sum_{\varphi\in\Phi} \alpha(\varphi)\varphi > \theta]$. For any predicate ψ define ψ^A by $X \mapsto \psi(X \cap A)$. Then, for every $\varphi \in \Phi$, $\varphi^A = \varphi$ if $S(\varphi) \subset A$ and φ^A is identically zero otherwise. Let Φ^A be the set of masks in Φ whose supports are subsets of A. Then $\psi^A_{\text{ODD}} = [\sum_{\varphi\in\Phi^A} \alpha(\varphi)\varphi > \theta]$, and $|S(\varphi)| < |A|$ for all $\varphi \in \Phi^A$. This contradicts theorem 4 because it implies that the order of ψ^A_{ODD}, viewed as a predicate on A, is less than $|A|$. □

Summarizing, the predicate ψ_{ODD} has order equal to the cardinality of the retina, and in a linear threshold function representation with masks *all* masks are needed. Furthermore, by a combinatorial argument it can be shown that in such a representation the weights grow at least as fast as $2^{|S(\varphi)|-1}$ (see theorem 10.1 in Minsky and Papert). Such a representation is given by

$$\psi_{\text{ODD}}(X) = [-\sum(-2)^{|S(\varphi)|}\varphi(X) > 1]$$

where summation is over all masks. Consequently, a perceptron not only has to compute a large number of predicates, but also the weights of these predicates increase exponentially. For instance, for a relatively small retina of 5×5 squares the number of masks is 2^{25} and, in absolute value, the largest weight is 2^{25}; thus, the internal proportions of the weights grow exponentially large.

As a final example, the predicate $\psi_{\text{CONNECTED}}$ will be considered. Call two points of the finite retina R *adjacent* if they correspond to squares with a common edge. A subset X of R is *connected* if for any two points p, q in X there is a path of adjacent points in X through p and q. Connectedness is an important feature in pattern recognition. It will be shown that $\psi_{\text{CONNECTED}}$ has arbitrarily large orders as R grows in size. We first prove the following theorem.

Theorem 6 *Let $A_1, \ldots, A_m$ be disjoint subsets of R with equal cardinalities $4m^2$, and define the predicate*

$$\psi(X) = [|X \cap A_i| > 0 \text{ for every } i].$$

Then the order of ψ is at least m.

Proof Let G be the group of all permutations of R with $g(A_i) = A_i$ and $g(p) = p$ for every $g \in G$, $i = 1, \ldots, m$, and $p \in R \setminus \cup_j A_j$. Clearly, ψ is invariant with respect to G. Let Φ be the set of masks of degree k or less, where k is some number at most $|R|$. Note that, for $\varphi, \varphi' \in \Phi$, $\varphi \equiv_G \varphi'$ if and only if $|S(\varphi) \cap A_i| = |S(\varphi') \cap A_i|$ for every i. Let $\Phi_1, \Phi_2, \ldots$ denote the corresponding equivalence classes. For every equivalence class Φ_j and every subset X of R let $N_j(X) := |\{\varphi \in \Phi_j : S(\varphi) \subset X\}|$. By a simple combinatorial argument,

$$N_j(X) = \binom{|X \cap A_1|}{|S(\varphi) \cap A_1|}\binom{|X \cap A_2|}{|S(\varphi) \cap A_2|} \cdots \binom{|X \cap A_m|}{|S(\varphi) \cap A_m|},$$

where φ is an arbitrary element of Φ_j. This implies that $N_j(X)$ is a polynomial of the form $N_j(x_1, \ldots, x_m)$ of degree at most k by taking $x_i = |X \cap A_i|$. Suppose $[\sum \alpha_\varphi \varphi > 0]$ is a representation of ψ as a linear threshold function with respect to the set of masks of degree at most k; for what follows we can take the threshold equal to zero without loss of generality. By theorem 2, the group invariance theorem, we can write

$$\sum \alpha_\varphi \varphi(X) = \sum \beta_j (\sum_{\varphi \in \Phi_j} \varphi(X)) = \sum \beta_j N_j(X) = \sum \beta_j N_j(x_1, \ldots, x_m)$$

which is itself a polynomial of degree at most k. Thus, we can write

$$\psi(X) = [Q(x_1, \ldots, x_m) > 0]$$

where $Q := \sum \beta_j N_j$ is a polynomial of degree at most k. Consequently, by definition of ψ and x_i, $Q(x_1, \ldots, x_m) > 0$ if and only if $x_i > 0$ for all i. By making the substitution $x_i = (t - (2i - 1))^2$ in $Q(x_1, \ldots, x_m)$, Q becomes a polynomial of degree at most $2k$ in t. Let t take on the values $t = 0, 1, \ldots, 2m$. Then $0 \leq x_i \leq 4m^2$ for all x_i. Observe that one of the x_i's equals zero for t odd, and all x_i's are positive if t is even. So Q is positive for even t and nonpositive for odd t. By counting the number of sign changes we obtain $2k \geq 2m$, so $k \geq m$, which concludes the proof. □

Minsky and Papert call theorem 6 the "one-in-a-box" theorem since the predicate investigated in this theorem is true for those patterns which have a nonempty intersection with each member of a given collection of disjoint subsets of R. Theorem 6 will be used to prove the announced result concerning $\psi_{\text{CONNECTED}}$. Call a predicate, defined for differently sized retinas, *of finite order* if there is a number k such that the order of the predicate is at most k whatever the size of the (finite) retina.

Theorem 7 *The predicate $\psi_{\text{CONNECTED}}$ is not of finite order.*

Proof Suppose the order of $\psi_{\text{CONNECTED}}$ is uniformly bounded by k, and let $m > k$. Consider an array of $2m + 1$ rows each containing $4m^2$ squares, see figure 3. For each $i = 1, \ldots, m$ let A_i be the set of points (squares) of the $2i$th row. Let R be the union of the even rows, i.e., of the A_i, and $\bar{R}$ of the odd rows. Define the predicate ψ on R by $\psi(X) = 1$ if and only if $\psi_{\text{CONNECTED}}(X \cup \bar{R}) = 1$. Let $\psi_{\text{CONNECTED}}$ have a representation $[\sum \alpha(\varphi)\varphi > \theta]$ where the φ's are masks of degree at most k. Define, for a mask $\varphi = \varphi_A$ $(A \subset (R \cup \bar{R}))$, φ' by $X \mapsto \varphi_{A \cap R}(X)$ $(X \subset R)$. Then $\varphi'(X) = 1$ if and only if $\varphi(X \cup \bar{R}) = 1$; consequently, $[\sum \alpha(\varphi)\varphi' > \theta]$ is a representation for ψ, of order at most k. Because $k < m$, this contradicts theorem 6 applied to ψ. □

5 Learning and convergence

As is apparent from the preceding sections, the usefulness of perceptrons and of neural networks in general is intimately related to representation of knowledge.

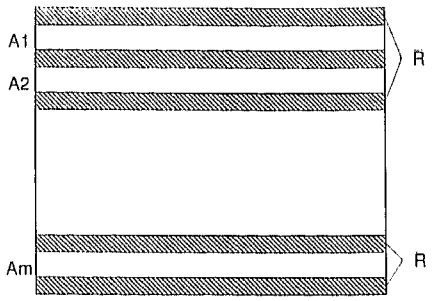

Fig. 3. Proof of theorem 7.

An essential feature of the perceptron is, however, that it can be trained. Because it is able to learn, one does not have to know the exact representation of a particular predicate in order to apply a perceptron.

Recall that a perceptron computes predicates of the form

$$\psi(X) = [\sum_{\varphi \in \Phi} \alpha_\varphi \varphi(X) > \theta].$$

This is an exact representation of the predicate. For many complex problems, however, we do not know this exact representation; in particular, we do not know the weights α_φ. A perceptron is programmed—parallel, or by simulation—in such a way that these weights can be adapted. The learning process starts with a more or less arbitrary set of weights. Next, the perceptron is "fed" some examples—for instance the complete set of objects to be classified, or a representative subset. For each example the output of the perceptron is compared with the desired output, and if necessary the weights are adapted. This process is repeated until a reasonable result is obtained. For an example, see the contribution by Weijters and Hoppenbrouwers in this book.

The algorithm to adapt the weights $\{\alpha_\varphi \mid \varphi \in \Phi\}$ may be as follows. Suppose we have a collection of patterns $F = F^+ \cup F^-$ we wish to classify and—for convenience—assume $\theta = 0$. We will denote a set of weights $\{\alpha_\varphi \mid \varphi \in \Phi\}$ as a vector A in $|\Phi|$-dimensional space. Further, for $X \in F$ the vector with the values $\varphi(X)$ as coordinates is denoted by $\Phi(X)$. The predicate ψ classifying the patterns in F can be written as

$$\psi(X) = [A \cdot \Phi(X) > 0]$$

for some weight vector A, where we assume that $A \cdot \Phi(X) > 0$ if $X \in F^+$ and $A \cdot \Phi(X) < 0$ if $X \in F^-$. Consider the following "learning algorithm":

Start Choose an arbitrary vector A.
Test Choose an $X \in F$.

If $X \in F^+$ and $A \cdot \Phi(X) > 0$: go to **Test**.
If $X \in F^+$ and $A \cdot \Phi(X) \leq 0$: go to **Add**.
If $X \in F^-$ and $A \cdot \Phi(X) < 0$: go to **Test**.
If $X \in F^-$ and $A \cdot \Phi(X) \geq 0$: go to **Subtract**.
Add Replace A by $A + \Phi(X)$. Go to **Test**.
Subtract Replace A by $A - \Phi(X)$. Go to **Test**.

Summarizing, if a pattern X is classified in the right way, then the next test pattern is chosen; if a pattern X is wrongly classified as belonging to F^-, then the corresponding Φ-vector is added to A; if a pattern X is wrongly classified as belonging to F^+, then the corresponding Φ-vector is subtracted from A. Surprisingly enough, it turns out that this simple algorithm works. We prove this result for a simpler formulation of the learning algorithm. Instead of distinguishing between vectors $\Phi(X)$ for patterns X, we will simply distinguish between vectors Φ in a collection F of zero-one vectors.

Consider the following program.

Start Set A to an arbitrary Φ of F.
Test Choose an arbitrary $\Phi \in F$.
If $A \cdot \Phi > 0$ go to **Test**;
otherwise go to **Add**. (P)
Add Replace A by $A + \Phi$. Go to **Test**.

Observe that this program can indeed replace the previous one by taking for F in (P) the set $\{\Phi(X) : X \in F^+\} \cup \{-\Phi(X) : X \in F^-\}$.

The following theorem is known as the *perceptron convergence theorem.*

Theorem 8 *Assume there exists a vector A^* for which $A^* \cdot \Phi > 0$ for all Φ in F, then program* (P) *will go to* **Add** *only a finite number of times.*

Proof Let $||\cdot||$ denote the Euclidean norm, and let m be the number of predicates φ, which is equal to the squared maximal length of a vector Φ in F. Since F is a finite set, there is a number $\delta > 0$ with $A^* \cdot \Phi > \delta$ for all Φ in F. Define the map $C : A \mapsto (A^* \cdot A)/||A||$. The Cauchy-Schwarz inequality, $|A^* \cdot A| \leq ||A^*||\,||A||$, implies $C(A) \leq ||A^*||$ for all vectors A. We consider the behavior of $C(A)$ on successive passes of the program through **Add**. Then

$$\begin{aligned} A^* \cdot A_{t+1} &= A^* \cdot (A_t + \Phi) \\ &= A^* \cdot A_t + A^* \cdot \Phi \\ &> A^* \cdot A_t + \delta \end{aligned}$$

so that, after the n th application of **Add** we obtain

$$A^* \cdot A_n > n\delta. \tag{1}$$

Because $A_t \cdot \Phi$ must be nonpositive (or the program would not have gone through **Add**), we further have

$$||A_{t+1}||^2 = A_{t+1} \cdot A_{t+1}$$

$$
\begin{aligned}
&= (A_t + \Phi) \cdot (A_t + \Phi) \\
&= ||A_t||^2 + 2A_t \cdot \Phi + ||\Phi||^2 \\
&\leq ||A_t||^2 + m
\end{aligned}
$$

so that, after the n th application of **Add** we obtain

$$||A_n||^2 \leq nm. \tag{2}$$

Combining equations refeq1 and 2 yields

$$C(A_n) = \frac{A^* \cdot A_n}{||A_n||} > \frac{n\delta}{\sqrt{nm}}.$$

Because $C(A) \leq ||A^*||$ the program can pass through **Add** only so long as $n \leq m||A^*||^2/\delta^2$. This completes the proof. □

Remark It is easy to verify that theorem 8 still holds if F is a compact set instead of a collection of zero-one vectors.

The algorithm in theorem 8 will after finitely many times result in a vector A^0 which has the property that $A^0 \cdot \Phi > 0$ for all Φ in F—the proof of the theorem actually gives an indication of the rate of convergence. In terms of the original problem, the predicate $\psi = [A^0 \cdot \Phi > 0]$ will have the following (desired) property:

$$X \in F^- \Rightarrow \psi(X) = 0, \; X \in F^+ \Rightarrow \psi(X) = 1.$$

This is often expressed as "the predicate ψ *separates* the sets F^+ and F^-." Of course, the vector A^0 does not have to be equal to A^*.

There exist some variations on this algorithm, for which variations of the perceptron convergence theorem hold. An important variation is classification in more than two classes. We conclude this section by formulating the corresponding algorithm. Let $F_1, F_2, \ldots$ be classes of patterns and assume that there exist a number $\delta > 0$ and vectors A_i^* for which, for all $j \neq i$

$$X \in F_i \Longrightarrow A_i^* \cdot \Phi(X) > A_j^* \cdot \Phi(X) + \delta.$$

The corresponding training program is as follows.

Start Choose $A_1, A_2, \ldots (\neq 0)$ arbitrary.
Test Choose i, j and $X \in F_i$.
If $A_i \cdot \Phi(X) > A_j \cdot \Phi(X)$, go to **Test**;
otherwise go to **Change**.
Change Replace A_i by $A_i + \Phi(X)$, A_j by $A_j - \Phi(X)$;
go to **Test**.

Under the mentioned conditions, this program will go to **Change** only a finite number of times.

6 Concluding remarks

The main conclusion of the preceding sections is that the usefulness of the simplest of neural networks, the perceptron, depends essentially on the representation of the problem to be handled. If many masks are needed or the internal proportions of the weights in an exact representation are large, then the perceptron might not perform very well. Examples were given in section 4. For instance, for the predicate ψ_{ODD} it can be shown that the number of "learning" examples must grow exponentially with the number of squares, i.e., the size of the problem. On the other hand, Minsky and Papert present some examples where the perceptron could perform well, such as recognition of convexity, of recognition of hollow or solid squares, and some others.

More complex problems may sometimes be solved by so-called multilayered perceptrons—an example is the *exclusive or* (XOR) predicate, see the contribution by Weijters and Hoppenbrouwers. These neural networks are also trained under supervision, according to the so called generalized Deltarule, which is an extension of the perceptron learning algorithm. Both are based on the "steepest descent" optimization principle. See the contribution by Henseler on Back Propagation.

References

D.O. Hebb (1949) *The Organization of Behavior.* Wiley, New York.

W.S. McCulloch and W. Pitts (1943) A logical calculus of the ideas immanent in neural nets. *Bulletin of Mathematical Biophysics*, 5, 115–137.

M.L. Minsky and S.A. Papert (1969, 1988) *Perceptrons: An Introduction to Computational Geometry.* The MIT Press, Cambridge, MA.

W. Pitts and W.S. McCulloch (1947) How we know universals. *Bulletin of Mathematical Biophysics*, 9, 127–147.

F. Rosenblatt (1959) Two theorems of statistical separability in the perceptron. *Proceedings of a Symposium on the Mechanization of Thought Processes*, Her Majesty's Stationary Office, London, 421–456.

F. Rosenblatt (1962) *Principles of Neurodynamics.* Spartan Books, New York.