Department of Electrical & Computer Engineering
Cullen College of Engineering
UNIVERSITY OF HOUSTON
Houston, TX 77204-4793

ECE 6313

NEURAL NETWORKS

H. ÖĞMEN

(713) 743 4428
ogmen@uh.edu

PURPOSE OF THE COURSE:

During the last two decades, there has been a considerable research interest in neural networks. Besides investigations concerning the basic research in this domain, a growing number of applications are under development. New societies devoted to neural network research have been formed and several conferences specializing in neural networks are held regularly. The interdisciplinary research efforts are directed towards the understanding of neural correlates of intelligent behavior as well as the application of this understanding to technological problems. Thus, this course is suitable for students in a wide range of disciplines. The course aims to provide students with a comprehensive overview of fundamental concepts and tools used in the domain and point out applications in various fields. It will serve as an introduction for students wishing to do research in neural networks and will let other students assess the applicability of the theory to their research field.

COURSE DESCRIPTION:

1. Review of philosophical and psychological underpinnings.
2. Review of basic neurophysiology.
3. Generalization of single cell models to networks.
4. Network dynamics, stability analysis, content addressable memory.
5. Qualitative and quantitative aspects of feed-forward additive and shunting equations.
6. Short-term memory and recurrent shunting network dynamics.
7. Long-term memory: Hebbian and non-Hebbian synaptic dynamics.
8. Supervised learning, function approximation theory, decision and game theory.
9. Associative learning: Classical and operant (reinforcement learning) conditioning.
10. Unsupervised pattern clustering: competitive learning, adaptive resonance theory.
11. Beyond empiricism and behaviorism: Future directions in neural network research.
12. Applications: pattern recognition and classification, adaptive sensorimotor mapping, adaptive control, perception, and related problems.

PREREQUISITES: Graduate standing or consent of instructor. (Familiarity with ordinary differential equations is expected.)

SUGGESTED BIBLIOGRAPHY:
Introduction to Neural and Cognitive Modeling by D.S. Levine (Lawrence Erlbaum Associates, Inc.), 2000 (Second Edition).

GRADING:
Your course grade will be based on a weighted average as follows:

1. Exam I: 30%
2. Exam II: 30%
3. Quizes: 25%
4. Homeworks: 15%

CHAPTER I: INTRODUCTION

1. A HISTORICAL OVERVIEW: PHILOSOPHICAL AND PSYCHOLOGICAL UNDERPINNINGS

- **Goal:** Understanding and emulation of intelligent behavior and its relationship to neural processes.

- **Assumption:** Intelligence is a natural *category*; i.e. a common set of principles underly all instances of intelligence.

- **Definition:** Intelligence is a highly complex phenomenon and no precise definition exists.

- **Studies of Intelligence–Knowledge:**

A. Philosophy

Most of the ideas present in major philosophical theories are also found in the pre-mediaval philosophy (e.g. Greek philosophers such as Plato, Aristotle). However, starting with Descartes, the philosophical formulations took more elaborate and precise form.

It is useful to categorize approaches in the continuum between two extremes given below, although no school adheres exactly to the extremes.

(a) A priorism (Descartes).

Also called *rationalism*: In its extreme form, it states that "knowledge is rooted in the *a priori* structures of the mind. Experience plays no role in the attainment of objective, "true" knowledge.

(b) Empiricism (Locke, Hume).

In its extreme form, it states that "all knowledge comes from experience". Uses the principle of *association*.

(c) Kantian synthesis (Kant).

Based on an analysis of necessary conditions for experience, it offers a synthesis of a priorism and empiricism. Contentless a priori structures such as time and space, called *categories*, make experience possible. Experience provides the contents of knowledge within the framework of the categories.

B. Psychology

(a) Birth of experimental psychology. (1875)

Wundt established the first experimental psychology laboratory. Experiments became the centerpiece of the studies and more objective methods, such as psychophysics, were used.

However, Wundt also acknowledged the limited ability of this approach in dealing with complex psychological phenomena such as personality and emotions.

In order to deal with such complex phenomena, psychological theories started using concepts such as consciousness and "internal mental states" which are not directly available to the observer. Freud's psychodynamic theory is an example.

(b) Behaviorism (Watson, Skinner, Hull). (ca. 1900-1960)

The use of such unobservable constructs led to a strong reaction among theorists who tried to use objective methods borrowed from other scientific disciplines such as physics. They defined the system using *only* observable variables, the inputs (S for stimulus) and the outputs (R for behavioral observable response). This approach is also known as S-R psychology in reference to stimulus and response. In its essence, it is similar to the engineering approach wherein a system is defined as a function of its inputs and outputs (c.f. transfer function). Note that eliminating internal constructs and reducing the system description to its inputs become equivalent to the empiristic explanation. Behaviorists used principles similar to the *association* principle of Hume. Inspired by Pavlov's experiments, Watson claimed *conditioning* (called *Classical or Pavlovian conditioning*) to be the basis of all knowledge. Skinner extended this by introducing the principle of *operant* (also called *instrumental*) conditioning. Hull felt the need for a rigorous mathematical language and developed a *mathematical* formulation of behavioristic postulates. As these theories were developed, shortcomings of a pure empiristic approach became evident once again (c.f. Kant). Among these, is the arbitrary way the inputs and outputs are defined. This is a "re-discovery" of Kant's critique. To deal with this problem, internal variables, called *intervening variables*, were introduced by Tolman and used by other theorists such as Hull.

C. Multi-disciplinary approaches

(a) Neural networks. (ca. 1960)

The rapid development of electrical technology fueled the growth of neuroscience. It became possible to reliably measure the electrical activity in the brain. The discoveries in neuroscience led to the birth of a field called *Neural Networks* which attempts to explain intelligent behavior based on neural principles. Around 1960s Frank Rosenblatt developed neural network models that exhibited learning behavior. Early neural network models were based on behavioristic postulates.

(b) Cognitive science – Artificial Intelligence. (ca. 1960–Present)

However around 1960s, due to its empiristic limitations, behaviorism lost its appeal in the psychological circles and psychologists moved to the opposite extreme, viz. an approach that minimized learning and emphasized the *a priori* structures of knowledge. The parallel development of digital computers provided psychologists with a model of information representation and processing. Psychologists, joined with computer scientists, developed a new approach called *cognitive science – artificial intelligence (AI)*. This research route prevailed over the neural network approach due to the demise of behaviorism and due to the fact that neuroscience was not fully developed to support complex neural theories.

(c) Neural networks (revisited). (ca. 1985–Present)

In 1960s, AI researchers were very optimistic and predicted that fundamental problems in their field would be solved within a decade or so. While, AI had limited success in certain areas, it was realized around 1980s that the AI approach was still far from reaching its promised goals. This prompted a return to the neural network approach, and the importance of learning was "re-discovered" once more. Unfortunately, this return has been drastic and most of the current neural network models are empiristic–behavioristic.

2. BASIC NEUROPHYSIOLOGY

Structural properties of neurons:

Nervous tissue consists of *neurons* and *satellite cells.* A neuron is a cell with the special electrical characteristics of excitability by which it can process and conduct information. They represent approximately 10% of the cells in the brain and are thought to be the main agents of information processing in the brain. Sattelite cells account for about 90% of the cells of the brain and their main roles are belived to be support cells.

Structural components of a typical neuron is shown in Figure 1. The cell is surrounded by a *membrane* that has specialized components such as ion channels. The *cell body (soma)* contains most of the metabolic apparatus of the cell. *Dendrites* are processes attached to the cell body. The cell receives inputs from other cells mainly at the dendrites. The *axon* is a stimulus conductor leading from the neuron and making connections with dendrites at the specialized sites, called *synapses.*

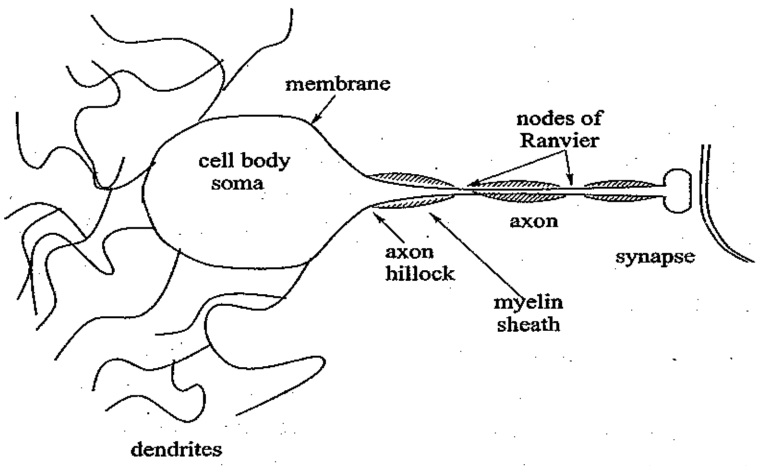


Figure 1: *Structural components of a neuron.*

Functional properties of neurons:

The outside and inside of the cell contain ionic solutions. While the membrane is mainly an isolating layer, it contains *ion channels* through which ions can travel. There exists ion channels that are *ion-selective*, in that they are *permeable* to specific ions. A channel that is permeable to Na^+ is called a sodium channel. For a given ion species, the equilibrium between the concentration gradient and voltage gradient is reached at a voltage, E, called the *Nernst potential*, given by the Nernst equation:

$$E = \frac{RT}{zF} ln \frac{C_1}{C_2} \ , \tag{1}$$

where R is the gas constant, F is Faraday's number, T is the absolute temperature, z is the valence of the ion, and C_1, C_2 are the concentrations of the ion inside and outside of the membrane. A membrane patch can be modeled by the equivalent electric circuit shown in Figure 2.

From this circuit we obtain:

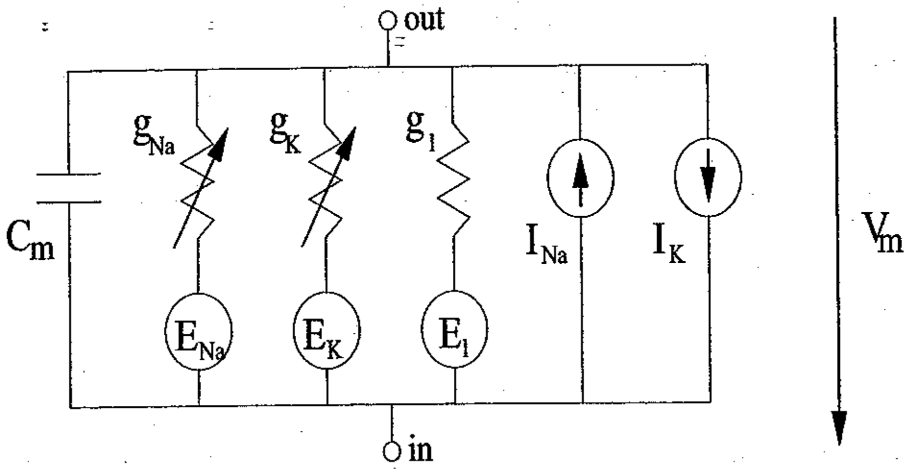


Figure 2: *Equivalent circuit of a membrane patch.*

$$C_m \frac{dV_m}{dt} = -(E_l + V_m)g_l + (E_{Na} - V_m)g_{Na} - (E_K + V_m)g_K \ , \tag{2}$$

where, V_m represents the *membrane potential*, C_m the membrane capacitance, g_{Na}, g_K, g_p, are membrane conductances for sodium, potassium, and (combined) passive ions, respectively. They represent the specific permeabilities of different ion channels. Similarly,E_{Na}, E_K, E_l represent the Nernst potentials for the different ion species. The two current sources represent the *ion-pump* which actively moves sodium and potassium ions in the opposite direction of their concentration gradient. The ion-pump is approximately electro-neutral.

The *resting membrane potential* is given by:

$$V_m = \frac{-E_l g_l + E_{Na} g_{Na} - E_K g_K}{g_l + g_{Na} + g_K} \ . \tag{3}$$

By substituting the following typical values: $E_l = 69\ mV$, $E_{Na} = 55\ mV$, $E_K = 75\ mV$, $g_l = 2.5 \times 10^{-6} MHO$, $g_{Na} = 0.5 \times 10^{-6} MHO$, and $g_K = 10 \times 10^{-6} MHO$, we find that the resting membrane potential is typically around -70 mV.

A change in the components that are involved in setting the membrane potential will cause a change in the membrane potential. A change that results in a net increase (decrease) in the membrane potential is called a *depolarization (hyperpolarization)*.

Comparison between neural and electronic devices:

Transmission of information:

graded potentials, action potentials (spikes), voltage-gated channels, frequency coding, myelinated fibers, nodes of Ranvier.

Synaptic function:

electrical vs chemical synapse, neurotransmitter, EPSP, IPSP, excitatory vs inhibitory synapse.

General structure of invertebrate, vertebrate, and primate nervous systems.

CHAPTER II
Quantitative Description of Neural Activity: From Single Cells to Networks

1 Towards a quantitative description of neural activity

1.1 Rashevsky's model (1933)

Based on Blair's theory of excitation (1932):

$$\frac{dc}{dt} = KI - k(c - c_0) \ , \tag{1}$$

where c is the average concentration of a positively charged ion, I is the current of that ion, c_0 is the resting concentration of the ion, K and k are positive constants. "Excitation occurs" when $c > c^*$, where c^* is a constant (threshold).

Rashevsky generalized this theory by introducing inhibitory effects described by the same type of equation used for excitatory effects. Accordingly, we have:

$$\begin{cases} \frac{d\epsilon}{dt} = & KI - k(\epsilon - \epsilon_0) \\ \frac{dj}{dt} = & MI - m(j - j_0) \ , \end{cases} \tag{2}$$

where ϵ and j are "excitatory" and "inhibitory" factors, respectively; and the rest of the symbols have the same meaning as in Equation (1). In this model, called the *two-factor model*, "excitation occurs" when $\epsilon \geq j$.

Important aspects: Analog model with a discontinuity generated by the threshold.

1.2 McCulloch-Pitts model (1943)

"We shall make the following physical assumptions for our calculus:

1. The activity of the neuron is an "all-or-none" process.

2. A certain fixed number of synapses must be excited within the period of latent addition in order to excite a neuron at any time, and this number is independent of previous activity and position on the neuron.

3. The only significant delay within the nervous system is synaptic delay.

4. The activity of any inhibitory synapse absolutely prevents excitation of the neuron at that time.

5. The structure of the net does not change with time."

(from W. S. McCulloch and W. H. Pitts (1943) "A logical calculus of the ideas immanent in nervous activity", *Bull. of Mathematical Biophysics*, vol. 5, pp. 115-133.)

Mathematically, these assumptions can be expressed as:

$$x_i(t+1) = f(\sum_j w_{ji} x_j(t) - \Gamma_i) \ , \tag{3}$$

where $x_i(t)$ is the activity of the ith neuron, w_{ji} is the synaptic weight between neuron j and neuron i, Γ_i is a constant threshold parameter. The function f is chosen as a binary function such as the unit step function or the sgn function (assumption 1). The synaptic weights and the threshold can be chosen to implement assumption 2. The use of a discrete-time process comes from assumption 3. Inhibitory synaptic weights can be chosen large enough to satisfy assumption 4. Finally, according to assumption 5 all parameter values are fixed.

Important aspects: Digital model with fixed network structure. Can be characterized using propositional logic.

1.3 Hodgkin-Huxley model (1952)

Based on the electric circuit equivalent of a membrane patch shown in Figure 1.

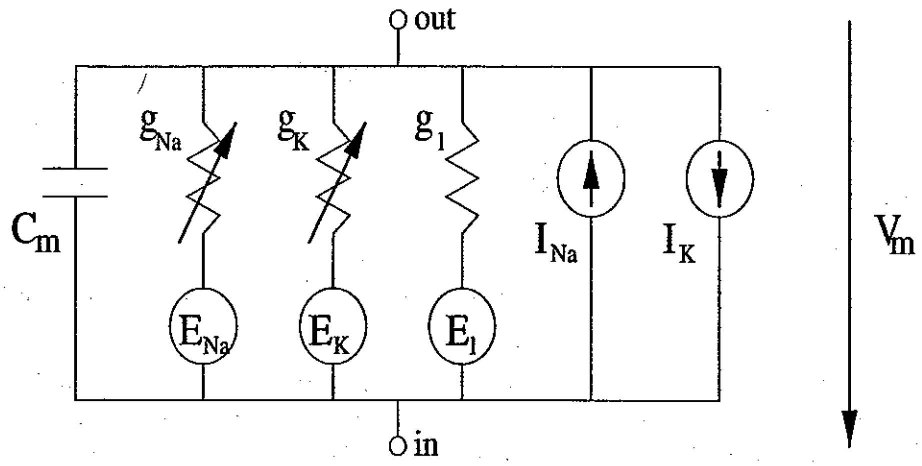


Figure 1: *Equivalent circuit for a membrane patch.*

this model can be written as:

$$C_m \frac{dV_m}{dt} = -(E_l + V_m)g_l + (E_{Na} - V_m)g_{Na} - (E_K + V_m)g_K \quad , \tag{4}$$

where C_m is the membrane capacitance, V_m is the membrane potential, E_{Na}, E_K, E_l are equilibrium (Nernst) potentials for Na, K, and leakage ions, respectively. The Na and K conductances, g_{Na} and g_K, respectively are (nonlinear) functions of V_m (voltage gated channels). The leakage conductance, g_l, is a constant.

Important aspects: This model gives a very accurate description of the membrane potential.

2 From cells to networks

2.1 McCulloch-Pitts (1943)

"Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic"
(from W. S. McCulloch and W. H. Pitts (1943) "A logical calculus of the ideas immanent in nervous activity", *Bull. of Mathematical Biophysics*, vol. 5, pp. 115-133.)

Some examples are shown in Figure 2.

2.2 Rashevsky (1961)

"The working of CNS is determined on one hand by the mechanism of transmission of excitation from one neuron to another at the synapse; on the other hand by the structural arrangements of the

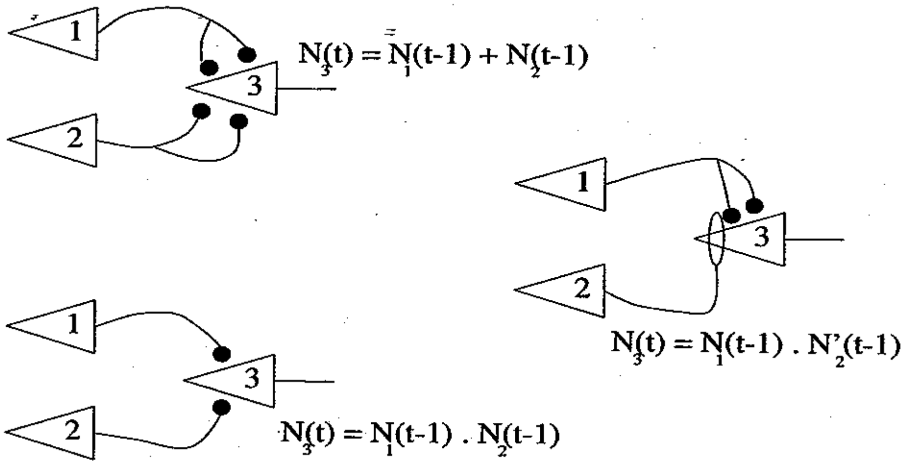

1
2
3
$N_3(t) = N_1(t-1) + N_2(t-1)$
1
2
3
$N_3(t) = N_1(t-1) . N'_2(t-1)$
1
2
3
$N_3(t) = N_1(t-1) . N_2(t-1)$

Figure 2: *Examples of Boolean functions computed by McCulloch-Pitts networks.*

numerous neurons. Both factors are still very little known. It is, however, known that the interaction at a synapse is a discontinuous process which obeys the "all-or-none" law. The intensity of an individual impulse in an axon is constant and independent of how the axon has been stimulated. This constant impulse when arriving at a synapse either evokes another constant impulse in the higher-order neuron or remains ineffective. Thus graded responses are impossible. W. McCulloch and W. Pitts have shown that a special branch of mathematics, the so-called Boolean algebra, is the appropriate mathematical tool for the description of those discontinuous phenomena.

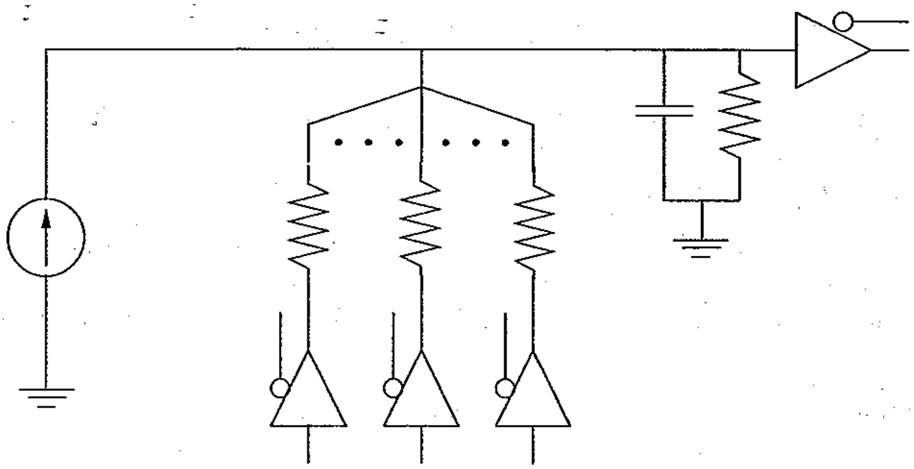
But in an animal we do mostly observe graded responses to graded stimuli. A muscular reaction may be weaker or stronger depending on the intensity of the stimulus which causes it. This apparent contradiction is readily explained by the fact that in *gross* responses not one or a few but a very large number of neurons, axons, and synapses are involved. The simple bending of a finger involves hundreds of individual efferent axons. In all such cases we deal with an average effect of a very large number of discontinuous phenomena, and this produces an apparently continuous result. The situation is not unlike that found in physics: the apparently continuous pressure which a gas exerts on the walls of a container is actually the result of a very large number of individual discontinuous impacts of molecules against the walls.

Groups of large numbers of individual axons or nerve fibers lead from different sense organs to the brain in anatomically discernible tracts. Similarly, groups of nerve fibers lead from the brain to different end organs. Such groups we shall call *pathways*, and we may speak of afferent and efferent pathways. Pathways may also lead from one part of the brain to another. Thus we can speak of first-order, second-order pathways, etc., just as we speak of neurons of different orders.

Whereas the process of excitation in a single nerve fiber is either absent or is constant, that process is graded in a pathway (...)."
(from N. Rashevsky (1961) *Mathematical Principles in Biology and their Applications*, Charles C Thomas Publ., Springfield, Ill.; pp. 20-21.)

2.3 Additive networks

Based on the *passive* properties of the membrane



$$\frac{dx_i}{dt} = -A_i x_i + \sum_j w_{ji} f_j(x_j) + I_i \ , \tag{5}$$

where x_i is the "membrane potential" of the ith neuron, $f_i(x_i)$ is its output, A_i is a constant decay rate, w_{ji} is the "synaptic weight" between neuron j and neuron i, and I_i is the external input to the $i$the neuron.

2.4 Shunting networks

Based on the Hodgkin-Huxley model:

$$\frac{dx_i}{dt} = -A_i x_i + (B_i - x_i)(\sum_j w_{ji}^{+} f_j(x_j) + I_i^{+}) - (D_i + x_i)(\sum_j w_{ji}^{-} g_j(x_j) + I_i^{-}) \ , \tag{6}$$

where the symbols have the same meanings as the additive model with the exception of the distinction that arises from the separation of excitatory and inhibitory inputs as they control different conductances.

CHAPTER III: MEMORY

1 Neural correlates for memory–learning in neural networks

1.1 McCulloch-Pitts (1943)

"The phenomena of learning, which are of a character persisting over most physiological changes in nervous activity, seem to require the possibility of permanent alterations in the structure of nets. The simplest such alteration is the formation of new synapses or equivalent local depressions of threshold (...)".

Theorem 1 [Theorem VII–McCulloch & Pitts (1943)] *Alterable synapses can be replaced by circles.*

(from W. S. McCulloch and W. H. Pitts (1943) "A logical calculus of the ideas immanent in nervous activity", *Bull. of Mathematical Biophysics*, vol. 5, pp. 115-133.)

1.2 Hebb's postulate (1949)

"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased."
(from D. Hebb (1949) *The Organization of Behavior: A Neuropsychological Theory*, John Wiley, New York, p. 62.)

1.3 Rosenblatt's Perceptrons (1962)

Combines ideas from Hebb and McCulloch-Pitts.

Definition 1 *The* perceptron *is a network of units S, A, and R with an interaction matrix V. The interaction matrix is* variable *and dependent on the past activities of the net. Both feed-forward* and feedback *are allowed.*

2 Distinction between Sensory Register (SR), Short-Term Memory (STM), and Long-Term Memory (LTM)

3 Content Addressable Memory (CAM)

3.1 Preliminaries

Let $\dot{x} = f(x)$. A point x_0 is called an *equilibirium point* if $f(x_0) = 0$.

3.1.1 Stability

Let $d(a, b)$ be a *distance function.*

Definition 2 *An equilibrium point x_0 is* stable *if for all $\epsilon > 0$, there exists $\alpha > 0$ and $t_0 > 0$ such that $d(x(0), x_0) < \alpha \implies d(x(t), x_0) < \epsilon$ for all $t > t_0$.*

Definition 3 *An equilibrium point is* asymptotically stable *if* $\lim_{t\to\infty} d(x(t), x_0) = 0$.

3.1.2 Local stability analysis vs global stability analysis, domain of attraction

3.1.3 Ljapunov's theory of stability

Theorem 2 [Ljapunov's second or direct method] *Let x_0 be an equilibrium point and $V(x)$ be a differentiable function defined on some neighborhood W of x_0 such that $V(x_0) = 0$ and $V(x) > 0$ for $x \neq x_0$. Let $\dot{V}(x) = \sum_j \frac{\partial V(x_j)}{\partial x_j} \frac{dx_j}{dt}$.*

(i) If $\dot{V}(x) \leq 0$ in $W - x_0$ then x_0 is stable.

(ii) If $\dot{V}(x) < 0$ in $W - x_0$ then x_0 is asymptotically stable.

Theorem 3 [La Salle's invariance principle] *Assume $V(x) \geq 0$ and $\dot{V}(x) \leq 0$, for all x. Let E be the locus $\dot{V}(x) = 0$ and let M be the union of all trajectories that remain in E for all t. Then the solutions that are bounded for $t > t_1$ for some t_1 approach M as $t \to \infty$.*

3.2 The concept of CAM in a dynamical system

memory ⟷ (asymptotically) stable equilibria
associated patterns ⟷ basin of attraction

3.3 Cohen-Grossberg Theorem

Theorem 4 [Cohen-Grossberg (1983)]: *In any system*

$$\frac{dx_i}{dt} = a_i(x_i)[b_i(x_i) - \sum_{k=1}^{n} c_{ki} d_k(x_k)] \tag{1}$$

such that

1. *the coefficient matrix $\| c_{ij} \|$ is symmetric and all $c_{ij} \geq 0$,*
2. *function $a_i(\xi)$ is continuous for $\xi \geq 0$ and function $b_i(\xi)$ is continuous for $\xi > 0$,*
3. *function $a_i(\xi) > 0$ for $\xi > 0$ and function $d_i(\xi) \geq 0$ for all ξ,*
4. *function $d_i(\xi)$ is differentiable and monotone non-decreasing for $\xi \geq 0$,*
5. *$lim_{\xi \to \infty} sup[b_i(\xi) - c_{ii} d_i(\xi)] < 0$ for all $i = 1, 2, ..., n$,*
6. *and either $lim_{\xi \to 0^+} b_i(\xi) = \infty$ or $lim_{\xi \to 0^+} b_i(\xi) < \infty$ and $\int_0^\epsilon \frac{d\xi}{a_i(\xi)} = \infty$ for some $\epsilon > 0$,*

all admissible trajectories approach the largest invariant set M contained in the set

$$E = \{y \in \Re^n : \frac{dV(y)}{dt} = 0, y \geq 0\} \quad , \tag{2}$$

where

$$\frac{dV(y)}{dt} = -\sum_{i=1}^{n} a_i d_i' [b_i - \sum_{k=1}^{n} c_{ki} d_k]^2 \tag{3}$$

Corollary 1 *If each function d_i is strictly increasing then the set E consists of equilibrium points of Eq. 1.*

Sketch of proof: Start with the Ljapuñov function

$$V = -\sum_{i=1}^{n} \int^{x_i} b_i(\xi_i) d_i'(\xi_i) d\xi_i + \frac{1}{2} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk} d_j(x_j) d_k(x_k) \tag{4}$$

Calculate $\frac{dV}{dt}$ along the trajectories of the system to obtain Eqn. (3).

3.4 Domain of applicability for the Cohen-Grossberg theorem

3.4.1 Additive networks

- Hopfield's theorem (1984)

- McCulloch-Pitts model (1943):

$$x_i(t+1) = sgn(\sum_j w_{ji} x_j(t) - \Gamma_i) \tag{5}$$

- Brain State in a Box (BSB) model (1977):

$$x_i(t+1) = S(x_i(t) + \sum_j w_{ji} x_j(t)) \tag{6}$$

- Hartline-Ratliff-Miller model (1963):

$$r_i(t) = e_i(t) - \sum_j k_{ji}[\frac{1}{\tau} \int_0^t e^{-\frac{(t-s)}{\tau}} r_j(s) ds - r_{ji}]^+ \tag{7}$$

- Amari-Arbib models (1977):

$$\tau \frac{\partial u(s,t)}{\partial t} = -u(s,t) + \int_y w(s-y) f[u(y,t)] dy + h + I(s,t) \tag{8}$$

3.4.2 Shunting networks

- Sperling-Sondhi model (1968):

$$\frac{dx_i(t)}{dt} = -x_i + x_{i-1} - x_i x_n \tag{9}$$

- Wilson-Cowan model (1972):

$$\Gamma_e \frac{dx_e(t)}{dt} = -x_e + (k_e - r_e x_e) f_e(c_1 x_e - c_2 x_i + I_e) \tag{10}$$

$$\Gamma_i \frac{dx_i(t)}{dt} = -x_i + (k_i - r_i x_i) f_i(c_3 x_e - c_4 x_i + I_i) \tag{11}$$

- Pinter's model (1984):

$$C_m \frac{de_i(t)}{dt} = -e_i(t) + L_i(t) - e_i(t) \sum_j f_j(e_j(t)) \tag{12}$$

- Grossberg's models

3.5 Interactive Activation Competition (IAC) model

3.6 Extensions: Optimization problems

CHAPTER IV: LTM DYNAMICS, LEARNING

1 Definition, classification

1.1 Non-associative learning:

(i) Habituation: Decrease of responsiveness in the presence of a repetitive or persistent stimulus.
(ii) Sensitization: Increase in responsiveness in the presence of a novel stimulus.

1.2 Associative learning:

(i) Classical/Pavlovian conditioning: A change in behavior as a result of a pairing of a stimulus (conditioned stimulus (CS)) with a reinforcing stimulus (unconditioned stimulus (UCS)) which is contingent on the CS. (Pavlov: physiological theory; Watson: psychological theory).
(ii) Operant/Instrumental conditioning: A change in behavior as a result of a reinforcing stimulus which is *contingent* on the performance of a behavior produced by the organism (Skinner: psychological theory).

1.3 Supervised vs unsupervised learning:

In the case of supervised learning, *both* input and desired output patterns are available to the organism. In the case of unsupervised learning, desired output patterns are not directly available.

1.4 The role of repetition in learning:

Learning can also occur without overt repetition.

2 Perceptron learning

2.1 Definitions

Definition: *A Sensory unit (S-unit)* is any transducer responding to physical energy by emitting a signal which is some function of the input energy. The input signal at time t to an S-unit s_i from the environment, W, is symbolized by $c_{wi}^*(t)$. The signal which is generated at time t is symbolized by $s_i^*(t)$.

Definition: *A Simple S-unit* is an S-unit with

$$s_i^* = u(c_{wi}^* - \theta_i) \ , \tag{1}$$

where s_i^* is the output of the unit, $u(.)$ is the unit step function, and θ_i is a threshold parameter.

Definition: *An Association unit (A-unit)* is a signal generating unit having input and output connections. An A-unit a_j responds to signals received by way of input connections c_{ij} by emitting a signal a_j^*.

Definition: *A Simple A-unit* is a logical decision element which generates an output signal according to

$$a_i^* = u(\alpha_i - \theta_i) \ , \tag{2}$$

where α_i is the algebraic sum of its inputs.

Definition: *A Response unit (R-unit)* is a signal generating unit having input connections and emitting a signal which is transmitted outside the network. The emitted signal from unit r_i is denoted by r_i^*.

Definition: *A Simple R-unit* is an R-unit obeying

$$r_i^* = sgn(\alpha_i) \ , \tag{3}$$

where $sgn(a) = 1$ if $a > 0$, $sgn(a) = -1$ if $a < 0$. When $a = 0$ $sgn(a)$ can be considered 0 or indeterminate.

Definition: *A Simple Perceptron* is a Perceptron with

(1) There is only one R-unit. All A-units are connected to this R-unit.

(2) Connections are *feed-forward.*

(3) Transmission delay is *fixed.*

(4) All signal generating functions for S-, A-, and R-units are of the following form:

$$u_i^* = f(\alpha_i) \ . \tag{4}$$

(5) Connections from S-units to A-units are *fixed.*

Definition: *An Elementary Perceptron* is a simple Perceptron with

$$a_i^*(t) = u(\sum_j v_{ji}^{(s)} s_j^*(t-\tau) - \theta) \tag{5}$$

and

$$r_i^*(t) = sgn(\sum_j v_{ji}^{(a)}(t) a_j^*(t-\tau)) \ , \tag{6}$$

where $v_{ji}^{(s)}$ and $v_{ji}^{(a)}(t)$ are the interaction coefficients (synaptic weights) between jth sensory and ith association unit and jth association and ith response unit, respectively.

Rosenblatt also distinguished between reinforcement controlled by the response only (spontaneous learning), by the stimulus only (forced learning), and by error. He also introduced the notion of *back-propagation of error*: "The procedure to be described here is called the 'back-propagating error correction procedure' since it takes its cue from the error of the R-units, propagating corrections back towards the sensory end of the network if it fails to make a satisfactory correction quickly at the response end" (Rosenblatt (1962) *Principles of Neurodynamics*, Spartan, Washington D.C., p. 292).

2.2 Learning algorithm

Consider an elementary perceptron. The response of the network is given by Eqn. (6). By dropping index i (since there is only one R-unit) and by neglecting the transmission delay, this can be written as

$$r^* = sgn(\sum_j v_j^{(a)} a_j^*) \tag{7}$$

Equivalently, this can be expressed as a dot product of two vectors:

$$r^* = sgn(v^{(a)} a^*) \ . \tag{8}$$

Therefore, for a *set* of inputs, training the network is equivalent to solving a *system of inequalities.*

For simplicity, since the S to A connections are fixed, we will assume that $a^* = y$, where y is the input vector. Now, assume that reinforcement is controlled by error. For definiteness, assume moreover that the desired output for all inputs is +1. A *cumulative error* for all inputs can be written as

$$E(v^{(a)}) = \sum_{y \in Y} -v^{(a)} y \ , \tag{9}$$

where Y is the set of inputs for which the output is in *error*. The global minimum for this error function is 0 and occurs when *all* inputs produced the desired outputs. A simple learning algorithm can be derived by a procedure which minimizes the error function. Using the steepest descent minimization technique, we obtain the following synaptic change (update) rule:

$$v^{(a)}(t+1) = v^{(a)}(t) + \epsilon \sum_{y \in Y} y \ , \tag{10}$$

where ϵ is a positive constant.

This algorithm is guaranteed to converge *provided that a solution exists, i.e. there exists weight values such that all inputs produce errorless (desired) outputs.*

This leads to the following question: When (or for what type of problems) do the solutions exist? We will call this *existence problem.* This issue was studied by M. Minsky and S. Papert in their book entitled *Perceptrons.*

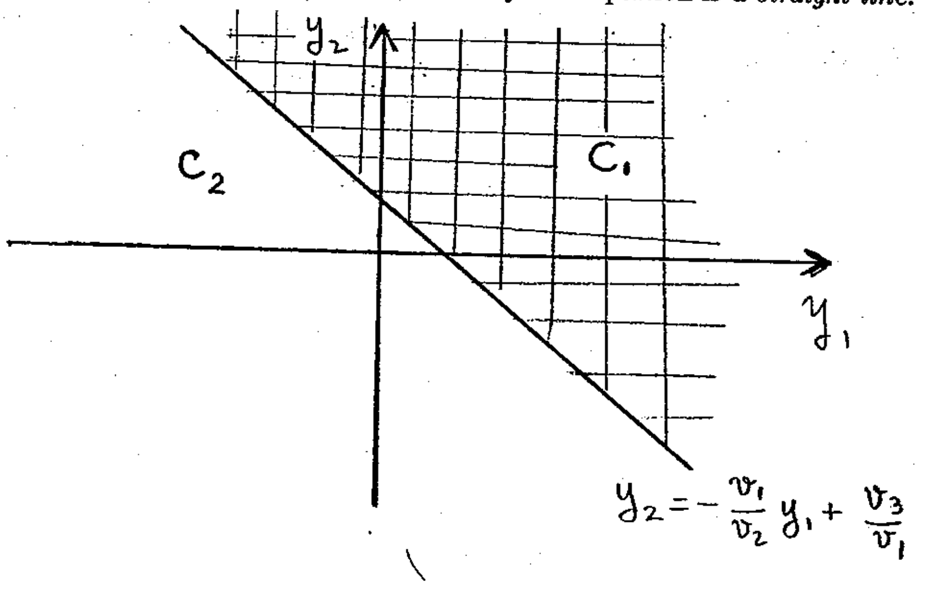
2.3 Geometrical analysis of Perceptrons: Linear separability

The solution weight vector can also be found by geometrical analysis instead of the iterative procedure above. For each input vector we can draw a perpendicular line. For that input to be correctly classified (i.e. +1 output) the weight vector should be on the same side as the input vector with respect to the perpendicular line. If we repeat this procedure for all training vectors and take the intersection of all admissible regions, we can determine the set of all solution vectors. Although this method can be used directly only for low-dimensional problems, it illustrates the geometrical nature of the existence problem.

Equivalently, we can also analyze the existence problem using the concept of *decision boundaries.* A decision boundary is a set of points that separate different classes in the feature space. For simplicity assume that $v^{(a)} = (v_1, v_2, v_3)$ and $y = (y_1, y_2, -1)$. The output of the network can be written as

$$r^* = sgn(v_1 y_1 + v_2 y_2 - v_3) \ . \tag{11}$$

The decision boundary is the loci of points for which the argument of the $sgn()$ equals zero; i.e. $v_1 y_1 + v_2 y_2 - v_3 = 0$. The decision boundary described by this equation is a *straight line.*



2.4 Linear approach: Widrow-Hoff, LMS procedure

The system of inequalities arising from the previous procedure can be transformed into a system of equalities by picking specific constants that satisfy the required sign constraint. Let

$$Y = \begin{bmatrix} y_1^{(1)} ... y_i^{(1)} ... y_n^{(1)} \\ .... \\ y_1^{(j)} ... y_i^{(j)} ... y_n^{(j)} \\ .... \\ y_1^{(m)} ... y_i^{(m)} ... y_n^{(m)} \end{bmatrix} \tag{13}$$

where the matrix Y is composed of m training vectors $y^{(j)}$ of dimension n. The system of inequalities can be then transformed into a system of equalities:

$$Yv^{(a)} = c \quad , \tag{14}$$

where c is a constant vector. If Y admits an inverse, then the solution can be calculated by matrix inversion as:

$$v^{(a)} = Y^{-1}c \quad . \tag{15}$$

However, in general the system is over-determined and Y does not admit an inverse. One can then calculate a solution by defining the objective function to minimize $J(v^{(a)})$ as the error:

$$J(v^{(a)}) = \left\| Yv^{(a)} - c \right\|^2 \quad . \tag{16}$$

Minimizing this error function by the steepest descent algorithm yields the following learning (synaptic update) rule:

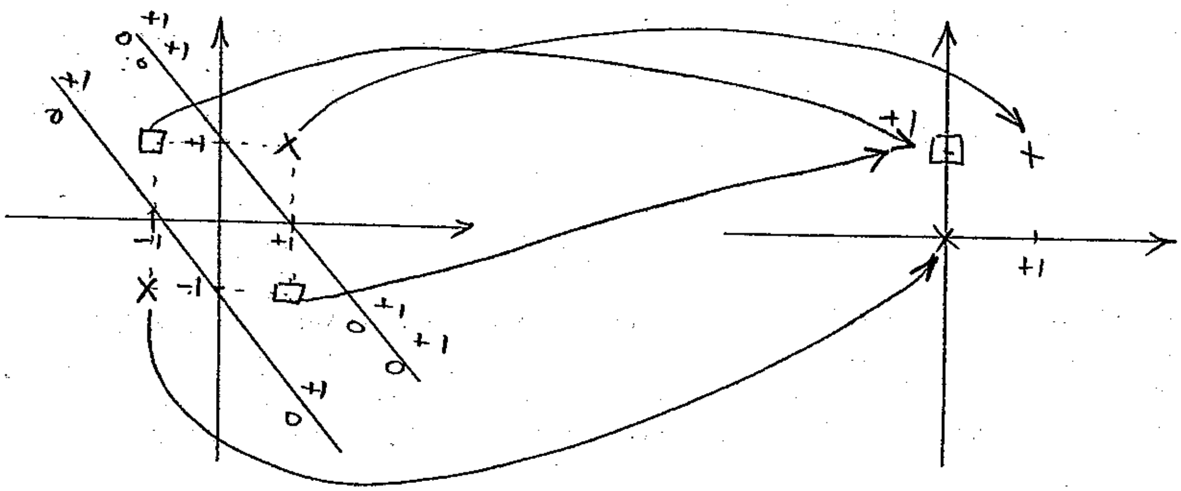
$$v^{(a)}(t+1) = v^{(a)}(t) + \epsilon \sum_i (d^{(i)} - o^{(i)}) y^{(i)} \quad , \tag{17}$$

where $d^{(i)}$ and $o^{(i)}$ are the desired and the actual outputs, respectively, for the ith training input $y^{(i)}$. Note that this equation is identical to the one derived for the elementary Perceptrons, with only one exception: the output of the network is no longer the nonlinear function $r^* = sgn(v^{(a)}y)$ but the *linear* function $o = v^{(a)}y$. This linear network was developed by Widrow and Hoff and was named ADALINE (ADaptive LINear Element). A network with several neurons was called MADALINE (Multiple ADALINEs).

When the solution of the problem does not exist, the elementary Perceptron will fail to converge, while the ADALINE network will converge to a least-mean-squares (LMS) solution. This difference in convergence behavior can be traced to the fact that the Perceptron is trying to obtain an *exact representation* (zero error criterion) while the ADALINE is trying to obtain a *minimum error representation.*

2.5 Beyond linear separability: Multi-layer networks, back-propagation algorithm

The Figure below shows how increasing the number of layers between the input and output can transform a linearly-inseparable problem into a linearly-separable problem. This answers the existence problem. It remains, however, to solve the *learning problem*, i.e. the development of a procedure whereby the solution is found. The gradient approach discussed above cannot be used here. This is because the gradient approach requires the computation of derivatives and the nonlinearities used in the *Elementary Perceptrons* are not differentiable in the classical sense (i.e. without using generalized functions, e.g. Dirac delta function $\delta(a)$).



This difficulty was by-passed by replacing the nonlinearities $u(.)$ and $sgn(.)$ with differentiable ones, such as $f(a) = (1 + e^{-a})^{-1}$. This function has the added advantage of having a simple derivative that can be written as: $f'(a) = f(a)(1 - f(a))$. The "back-propagation algorithm" derived by Werbos in 1974 (and rediscovered by others in the 80s) minimizes the sum of squared-errors ($\sum_{p \in P}(d^{(p)} - o^{(p)})^2$, where P is the set of all input patterns, $d^{(p)}$ and $o^{(p)}$ are the desired and actual outputs of the network for input pattern p) using the steepest-descent technique.

2.6 Critique

• *The existence problem:* Several recent theorems (e.g. Funahashi, 1989; Hornik et al. 1990; Park & Sandberg, 1991; Hornik, 1991) showed that multi-layer feed-forward nets are *universal approximators*, i.e. they can approximate arbitrary functions with arbitrary precision, with some mild conditions on the nonlinearities. The caveat here is that these theorems assume that there exits a "sufficient number" of hidden units without specifying what this sufficient number is for a specific problem.

In general, this type of neural networks can be viewed as a sub-class of the general mathematical theory of function approximation. Their difference, when compared to other methods such as Fourier series, Taylor series, is that they use *composition* (superposition) of basis functions rather than a linear combination.

A powerful theorem concerning the representation of functions by composition and addition was published by Kolmogorov in 1957:

Theorem (Kolmogorov): Any continuous function $f(x_1, x_2, ..., x_n)$ on I^n, where $I = [0,1]$, can be represented *exactly* in the form:

$$f(x_1, x_2, ..., x_n) = \sum_{j=1}^{2n+1} h_j(\sum_{i=1}^{n} g_{ij}(x_i)) \ , \tag{18}$$

where h_j and g_{ij}s are continuous functions of one variable, g_{ij}s are fixed, monotone increasing functions that are not dependent on $f(x_1, x_2, ..., x_n)$.

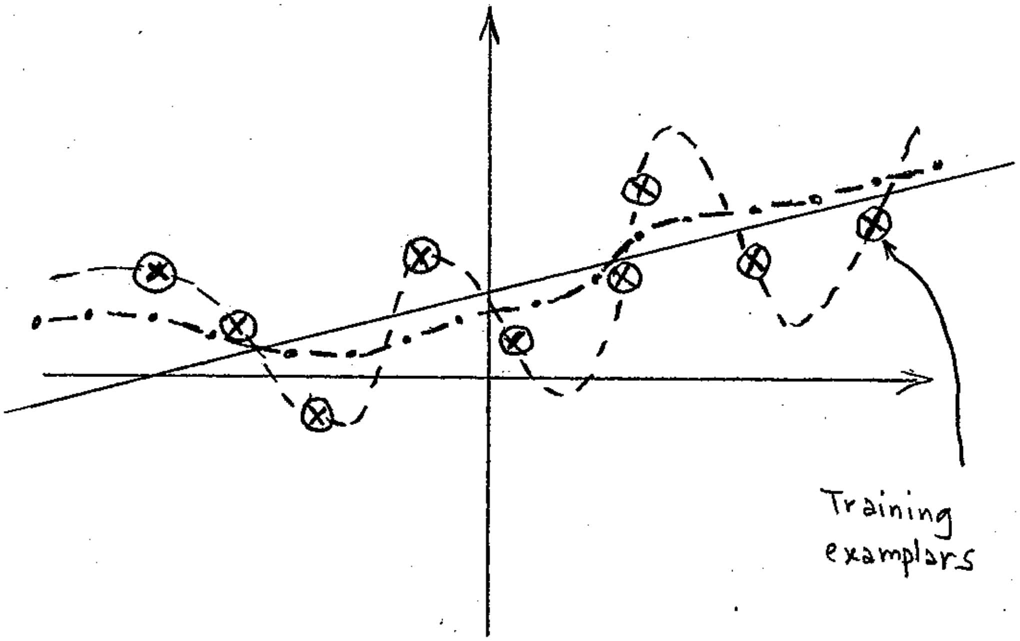
However, no general method is known that would generate this representation for a given function. Furthermore, the relevance of this theorem to neural networks is debated (see for example, Lin & Unbehauen, 1993).

Note that arguments on whether multi-layer networks overcome Minsky and Papert's criticism seem to be based on the use of *exact* versus *approximate* representation issue.

• *Comparison with other approaches, Decision-game theory:*

Definition: A problem in decision theory has the following elements: *(i)* a set $N = \{n_1, n_2, ...\}$ of *possible states of nature*; *(ii)* a set $A = \{a_1, a_2, ...\}$ of *possible actions*; *(iii)* a *loss function*, $L(n, a)$. A *game* is a triplet (N, A, L). A *decision* function is a function whose range is A. To choose a decision function, the goal could be set, for example, to minimize average loss or to minimize the maximum loss. If n is known, then a decision function can be generated by combining the goal with $L(n, a)$. In practice, n is not known directly but is available through a process of *observation*, $x = \Theta(n)$. If we can characterize the process of observation and if it admits an inverse, a decision function can be generated using $L(\Theta^{-1}(x), a)$. If we cannot characterize the process of observation, possible approaches include: 1) *Probabilistic approach:* Model x as a random variable whose distribution depends on n. Use the theory of probability and statistics (e.g. statistical pattern recognition). 2) *Deterministic approach:* Use the theory of functional approximation to select a decision function $a = d(x)$ (e.g. discriminant analysis, the neural net approach discussed above).

- *Generalization ability:* Here, we are dealing with an ill-posed problem (cf. bias-variance dilemma).



- *The learning problem:* Learning in these networks involves, in general, finding the global minimum of a non-convex function.
- *Real-time learning problem:* Learning is based on a global error measure, has difficulty in performing and learning simultaneously (cf. stability-plasticity dilemma).
- *Descriptive model:* Weak biological basis.
- *Success highly dependent on the "external designer":* Empiristic, behavioristic basis; requires "categories" (Kant, 1781), "intervening variables" (Tolman, 1922). Issue of representation.

Dfns:

Neuron.

Neural Networks

[Artificial Neural Networks]

Computational Neuroscience

Today ANN

Different varieties; currently most popular are learning

Learning: Process of acquisition of information into memory.

ANN: large focus. 2 types of learning problems

1. Unsupervised learning (Pattern clustering)
2. Supervised learning

↓

- Conditioning
- Labeled data / Examplar based learning

↓

Pattern classification problem.

Pattern classification problem

Training set $= \{(x^{(1)}, d^{(1)}), (x^{(2)}, d^{(2)}), \ldots (x^{(m)}, d^{(m)})\}$

input ↑ desired output ↑ m examples

Classification → output discrete - classes.

e.g. $d^{(i)} = \begin{cases} 0 \\ 1 \end{cases}$ 2 classes

Apples → Good
→ Bad

2 stages Training

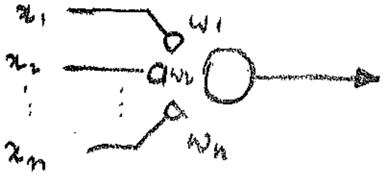
$\{(\ ), \ldots (\ )\}$ →

"Generalization"

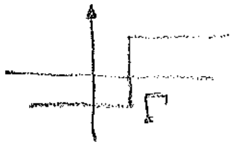
input not present in the training set.

A simple neural network approach

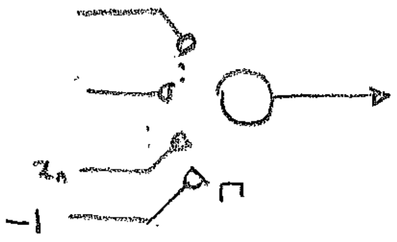
The simplest "net": 1 neuron.



$$y = \mathrm{sgn}\left(\sum_{i=}^{n} w_i x_i - T\right)$$



Equivalently



$$y = \mathrm{sgn}\left(\sum_{i=1}^{m+1} w_i x_i\right) = \mathrm{sgn}(w \cdot x)$$

Classification problem:

$$(1) \begin{cases} y^{(1)} = d^{(1)} \\ \vdots \\ y^{(m)} = d^{(m)} \end{cases}$$

find w_i to satisfy (1)

system of nonlinear equations

(2) $$\left\{ \; w x^{(i)} \gtrless 0 \quad \begin{matrix} > \text{if } d^{(i)} = +1 \\ < \text{if } d^{(i)} = -1 \end{matrix} \right.$$ system of linear inequalities.

Normalization

$$z^{(i)} \triangleq d^{(i)} \cdot x^{(i)} \qquad \begin{aligned} z^{(i)} &= x^{(i)} \text{ if } d^{(i)} = +1 \\ &= -x^{(i)} \quad\;\; = -1 \end{aligned}$$

(3) $$\left\{ \begin{matrix} \vdots \\ w z^{(i)} > 0 \\ \vdots \end{matrix} \right.$$

Instead of solving (3) directly, use a minimization approach

(i) Define an Error function such that $E(w) \geq 0$ and $E(w) = 0$ when (3) is satisfied.

(ii) Minimize $E(w)$

Perceptron error function:

$$E(w) = \sum_{z^{(i)} \in I} - w z^{(i)} \qquad I = \{ z^{(i)} \mid w . z^{(i)} < 0 \}$$

Minimization procedure:

Steepest Descent Algorithm:

Background:

THEOREM: Given a multivariate function $f(x)$, where $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ + the direction of the Gradient Vector, $\nabla f(x)$ is the direction of maximum increase for $f(x)$.

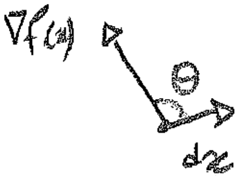
Proof: $\nabla f(x) = [\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots \frac{\partial f}{\partial x_n}]$

Total differential $df(x)$

$$df(x) = \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2 + \ldots \frac{\partial f}{\partial x_n} dx_n$$

$$= \nabla f(x) \cdot dx$$

$$= \| \nabla f(x) \| \; \| dx \| \cos \theta$$

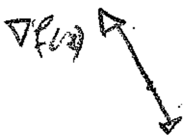


Choose θ so that $df(x)$ is maximum

$\theta = 0 \rightarrow$ maximum increase

Maximum decrease (minimization

$\theta = 180°$ $\cos\theta = -1$



Direction opposite to $\nabla f(x)$ maximum decrease

Steepest descent.

Steepest descent minimization algorithm:

Find x for which $f(x)$ is minimum.

(i) Start with a random value of x : $x(0)$

For $i=0,1,\ldots$

(ii) Calculate $\nabla f(x(i))$

(iii) Take a step in the direction opposite to the gradient

$$x(i+1) = x(i) - \varepsilon \nabla f(x(i))$$

At convergence (minimum of $f(x)$): $\nabla f = 0$

$$\Rightarrow x(i+1) = x(i)$$

Apply to Perceptron error function;

Find w to minimize $E(w) = \sum_{z^{(i)} \in I} -w z^{(i)}$

Find $\nabla E(w) = [\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots \frac{\partial E}{\partial w_n}]$

$$\frac{\partial E(w)}{\partial w_j} = \frac{\partial}{\partial w_j}\sum_{z^{(i)} \in I} -w z^{(i)} = \frac{\partial}{\partial w_j} -\sum_{z^{(i)} \in I} (w_1 z_1^{(i)} + w_2 z_2^{(i)} + \dots w_n z_n^{(i)},$$

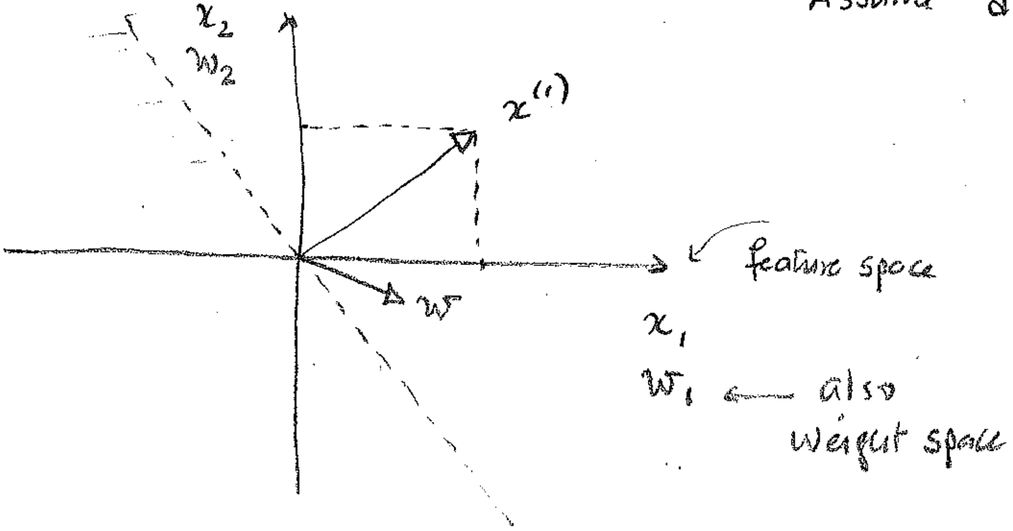
$$= -z_j^{(i)}$$

$$\nabla E(w) = -\sum_{z^{(i)} \in I} z^{(i)}$$

$$\boxed{w(i+1) = w(i) + \epsilon \sum_{z^{(i)} \in I} z^{(i)}}$$

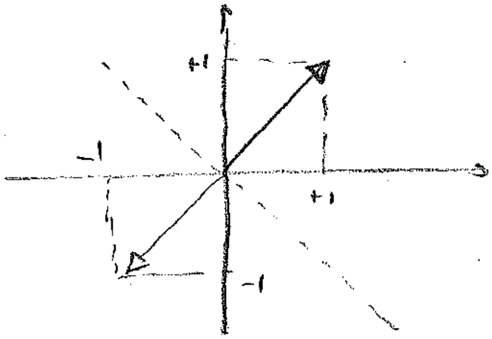
Geometric interpretation

Assume $d^{(i)} = +1$



Perceptron Convergence Theorem: If the solution exists, the P.L.A is guaranteed to find it.

When does the solution exist?

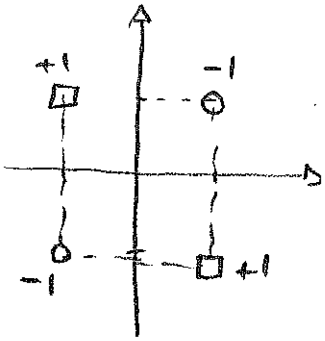


Solution space = ϕ

Minsky & Papert
XOR problem

0	0	0
0	1	1
1	0	1
1	1	0

-1	-1	-1
-1	+1	1
1	-1	1
1	1	-1



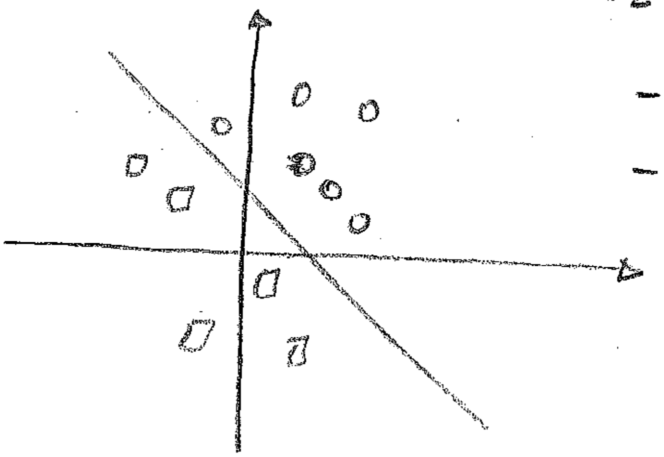
Alternative look at solution space:

$$\operatorname{sgn}[w_1 x_1 + w_2 x_2 - T] \begin{matrix} > 0 \\ < 0 \end{matrix}$$

Decision boundary

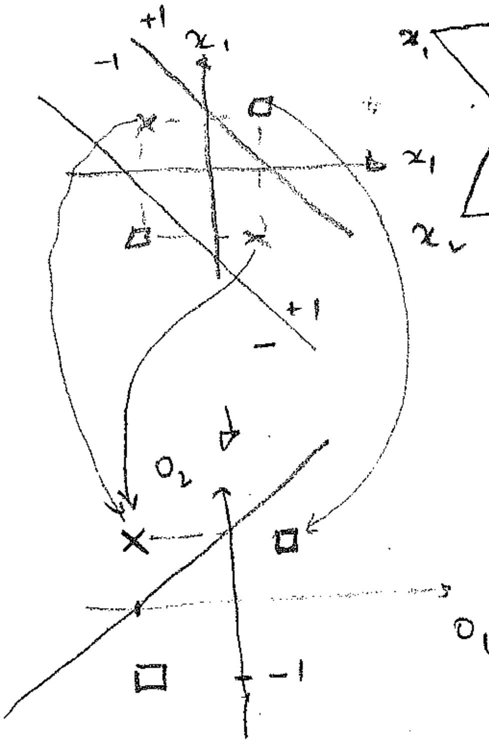
$$w_1 x_1 + w_2 x_2 - T = 0$$

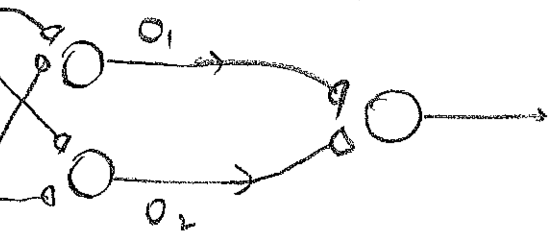
$$x_2 = -\frac{w_1}{w_2} x_1 + \frac{T}{w_2}$$



- Linear discriminant functions
- Linearly separable problems

Multilayer networks

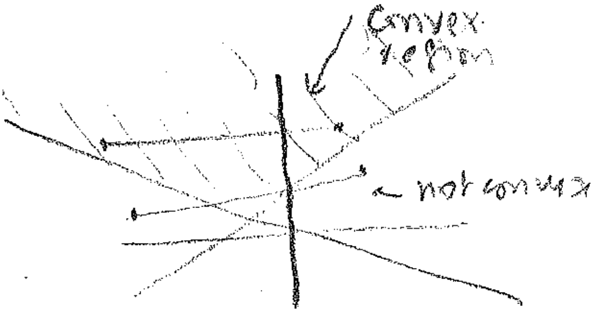




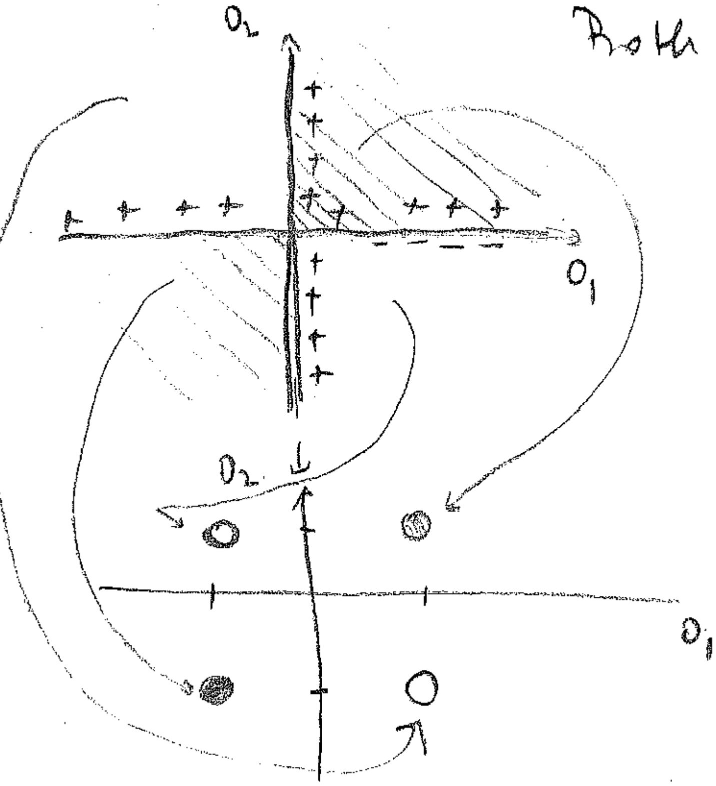
2 layer networks:

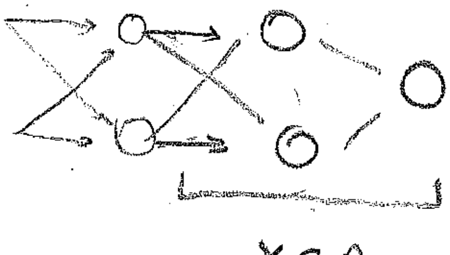
→ Problems with convex regions

Convex set (region): Given any 2 points in the region, all points on the line segment joining these two points belong to the region



Both regions non-convex





XOR

Existence Problem:

3Layer Perceptrons can solve arbitrarily complex problems, provided that they have sufficient number of neurons.

Training Problem:

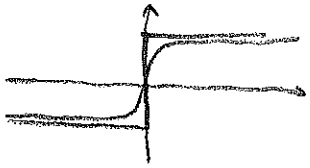
How to determine # of neurons and weights?

Discuss first weights in a fixed architecture

Gradient methods do not apply because sgn[ ] is not differentiable

Approximation approach:

Replace sgn[ ] by f()



Advantage:
- Differentiable
- Capture better the statistical nature of firing
- Produces analog output can be used with analog problems.

Universal approximation theorems:

3 layer networks with sigmoidal nonlinearities and with sufficient number of neurons can approximate arbitrary functions with arbitrary precision.

Note: Exact solution vs approximate solution

$\Downarrow$

$E = 0$ $\quad\quad$ $E \leq \varepsilon$ for a given ε.

BHAVIN'S CLASS

1/

Neural networks

Biological

Artificial

Learning - 1.1. non-associative learning
- habituation
- sensitization

1.2 Associative learning
- classical pavlovian cond.
- Operant instrumental cond.

1.3 Supervised vs unsupervised learning

⇨ Supervised learning
- training set

$\mathcal{T} = \{ (\text{input}, \text{output}), \ldots (\ , \ ) \}$

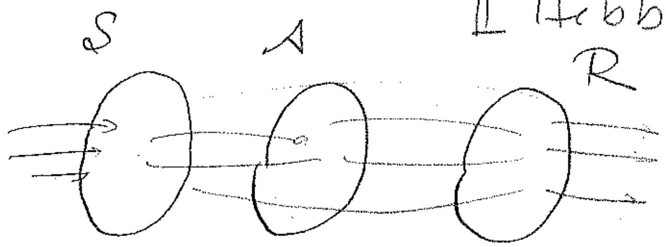
- train a network to produce outputs given inputs
"training problem"

- to produce correct outputs for inputs that were not in the training set
"generalization problem"

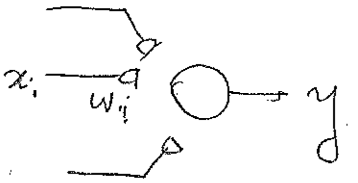
Rosenblatt

Perceptrons. [McCulloch-Pitts]

[Hebb]



$$y = \text{sgn}\left(\sum_i w_i x_i - T\right)$$



$$y = \text{sgn}\left(\sum_i w_i x_i\right)$$

2 class pattern classification problem.

$$\left\{ (x^0, d^0), \ldots (x^{(i)}, d^{(i)}), \ldots (x^{(m)}, d^{(m)}) \right\}$$

$$d^{(i)} = \begin{cases} +1 \\ -1 \end{cases}$$

$$\left\{ \begin{array}{l} \vdots \\ y^{(j)} = \text{sgn}\left(\sum_i w_i x_i^{(j)}\right) = d^{(j)} \\ \vdots \end{array} \right.$$

$$\sum_i w_i x_i^{(j)} = w \cdot x^{(j)}$$

System of nonlinear equations

$$\left\{ \begin{array}{l} \vdots \\ w \cdot x^{(j)} \begin{array}{ll} > 0 & d^{(j)} = +1 \\ < 0 & d^{(j)} = -1 \end{array} \\ \vdots \end{array} \right.$$

linear system of inequalities

define $y^{(j)} \triangleq d^{(j)} \cdot x^{(j)}$ "normalized input"

$$\begin{cases} \vdots \\ w \cdot y^{(j)} > 0 \\ \vdots \end{cases} \qquad (S)$$

Approach: Error function + minimization or error.

$$\Downarrow$$

$$E(w) = \sum_{y^{(j)} \text{ incorrectly classified}} - w \cdot y^{(j)} \qquad \begin{cases} E(w) > 0 \\ E(w) \text{ when } (S) \text{ is satisfied} \end{cases}$$

minimization:

Gradient approach

Theorem: Given a multivariate function $f(x)$, the direction of the gradient vector $\nabla f(x)$ is the direction of maximum increase for $f(x)$

Proof:

$$\nabla f(x) = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots \frac{\partial f}{\partial x_n} \right]$$

$$df(x) = \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2 + \dots \frac{\partial f}{\partial x_n} dx_n = \nabla f \cdot dx$$

$$= \|\nabla f\| \, \|dx\| \cos \alpha$$

minimum : opposite direction

Gradient descent algorithm for minimization

$E(w)$ find w such that $E(w)$ is minimum

$$\left[\, w(t+1) = w(t) - \underset{\substack{\uparrow \\ \text{step size}}}{\rho} \nabla E(w(t)) \,\right]$$

$$E(w) = -\sum_{y^{(j)} \in I} w \cdot y^{(j)}$$

$$\frac{\partial E(w)}{\partial w_i} = -\sum_{y^{(j)} \in I} y_i^{(j)} \Rightarrow \nabla E(w) = -\sum_{y^{(j)} \in I} y^{(j)}$$

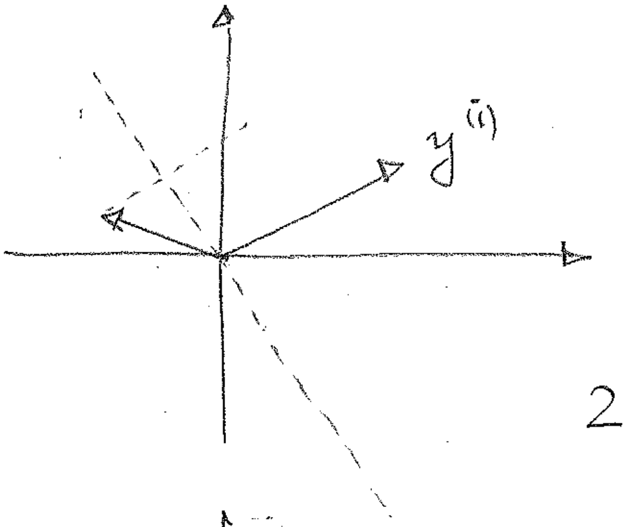
Perceptron learning algorithm:

$$\boxed{w(t+1) = w(t) + \rho \sum_{y^{(j)} \in I} y^{(j)}}$$

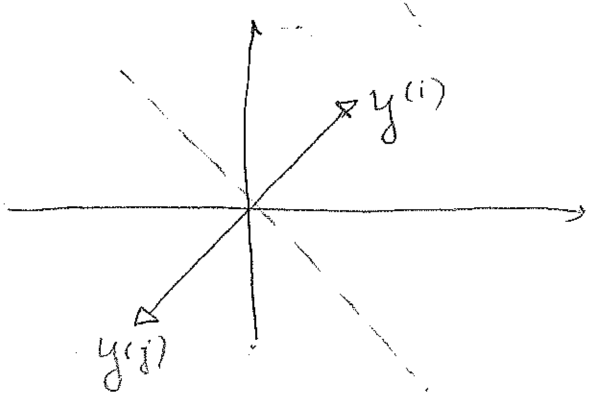
Perceptron convergence theorem:
If the soln exists, PLA is guaranteed to converge to the solution.

Geometric interpretation

$$\left\{ w y^{(i)} > 0 \right.$$



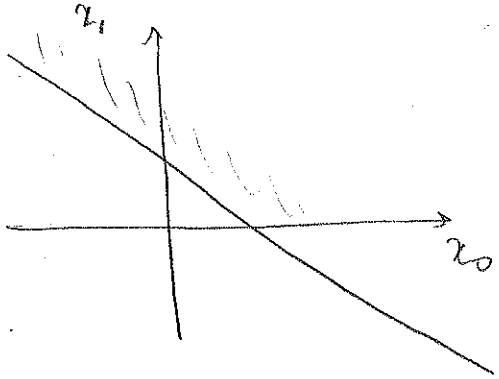
2 vectors + more



⟸ no solution

Geometric interpretation 2

$$\mathrm{sgn}\left[\underbrace{\sum w_i x_i - T} \right]$$

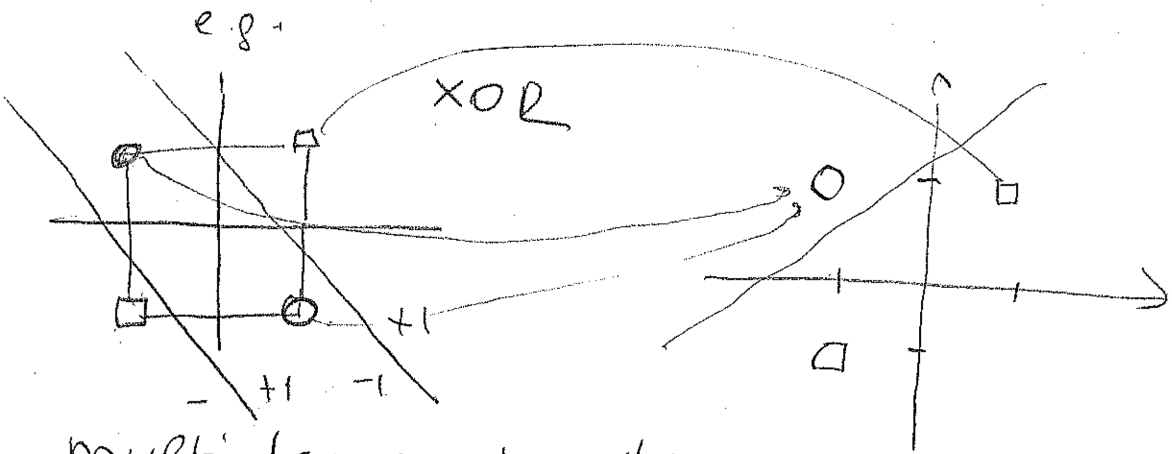


$$w_0 x_0 + w_1 x_1 - T \begin{array}{l} > \text{class 1} \\ = 0 \\ < \text{class 2} \end{array}$$

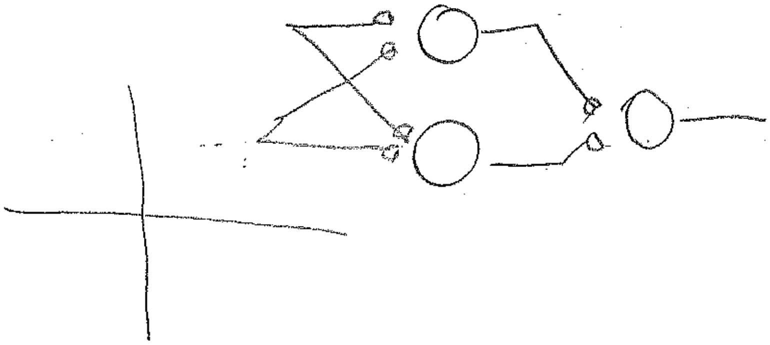
decision boundary

$$x_1 = -\frac{w_0}{w_1} x_0 + \frac{T}{w_0}$$

Linearly separable problems



multi-layer networks

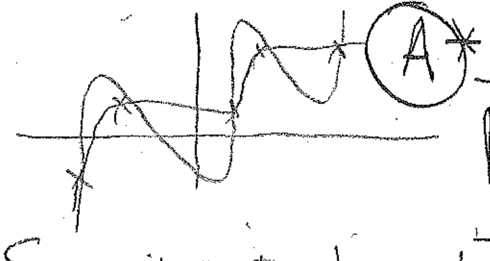


3 layer networks with mild restrictions are universal approximators
↓
EXISTENCE THEOREM

Catch: # of neurons assumed to be set "as needed"

Pratically how to determine it?

1. Assume noise free training examples

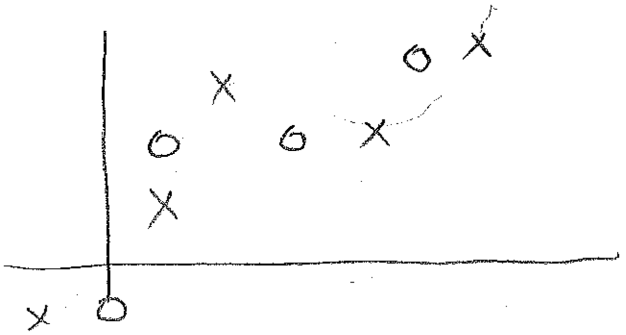


→ Generalization ill-posed problem

NO FREE LUNCH THEOREM

(B) Sensitivity to training set

Bias-variance dilemma



$$\begin{cases} f_1(x) = a_1 x + a_0 \\ f_2(\,) = a_2 x^2 + a_1 x + a_0 \\ f_n(x) = a_n x^n + \dots + a_0 \end{cases}$$

Equivalent
more neurons
more degrees
of freedom.

- Sensitivity to training set
- Sensitivity to noise