# CPE 721 - Redes Neurais *Feedforward*

Luiz **Calôba**

lcaloba@gmail.com caloba@ufrj.br

www.lps.ufrj.br/~caloba/CPE721

www.smt.ufrj.br/~luiz.caloba/CPE721

## 1 - Introdução

# Redes Neurais Artificiais Feedforward

**O que é isto ?**

**Para que serve ?**

**De onde veio ?**

# O que é isto ?

## Inteligência Artificial / Computacional

**Cognitiva, Simbólica**

**IA "tradicional"**

**Evolucionista**

**AG, Vida Artificial**

**Conexionista**

**Redes Neurais Artificiais**

## Redes Neurais Artificiais

### Redes Feedforward (CPE 721)

**Aproximadores e classificadores,**

**Redes "backpropagation"**

### Redes Não Supervisionadas (CPE 722)

**Classificadores**

## Cursos básicos em RNs no PEE da COPPE:

**CPE 721 – Redes Neurais Feedforward** - 2021/2
Pré requisitos CPE 721: apenas noções de álgebra linear, estatística e otimização contínua

**CPE 722 – Redes Neurais Não Supervisionadas e Agrupamento -** 2021/3 ?
Pré requisitos: apenas noções de álgebra linear, estatística e otimização contínua

**CPE 724 – Redes Neurais Feedforward - Aplicações –** ?
Pré requisito: CPE 721 - Redes Neurais Feedforward

## Outras disciplinas em IC (PEE, PESC, PEC)

*Disciplinas da Área Interdisciplinar de Engenharia e Ciência Computacional*

| Período | Programa | Código | Nome | Prof. |
|---|---|---|---|---|
| 2º | PEC | CPC881 | Métodos Comp. Inspirados na Natureza | Beatriz/Solange |
| | | COC891 | Deep Learning | Alexandre |
| | | CPC863 | Análise de Informação não Estruturada | Nelson/Rogerio |
| | | COC763 | Finite Elements in Fluids | Álvaro |
| | PEE | CPE721 | Redes Neurais Feedforward | Caloba |
| | | CPE884 | Aprendizado de Máquina Continuado | Seixas |
| | | CPE 720 | Otimização Natural | José Guilherme |
| | | | | |
| | PESC | CPS829 | Engenharia de Software Experimental II | Guilherme |
| | | CPS765 | Redes Complexas | Daniel Ratton |
| | | COS831 | Laboratório de Banco de Dados | Marta Mattoso |
| | PEQ | COQ897 | Otimização | Argimiro |
| | | COQ791 | Modelagem e Simulação de Processos | Priamo |
| | | COQ792 | Controle de Processos | Mauricio |

# CPE 721 - Redes Neurais Feedforward

## Programa:

- Introdução, motivação ao uso.
- Neurônios e redes neurais biológicas e artificiais. Estrutura feedforward.
- O treinamento  backpropagation.
- Métodos mais eficientes; resilient BP e métodos de segunda ordem.
- Pré-processamento: escolha das entradas, detecção de intrusos, escalamento, etc.
- Dimensionamento da rede, escolha dos parâmetros iniciais.
- Acompanhamento do treinamento: evolução do erro
- Pós-processamento: análise e correção do erro
- Aproximadores e Classificadores, capacidade de mapeamento. Operação.
- Outras redes: Redes de Base Radial, SVM, Deep Learning.
- Demonstrativos e exemplos de aplicações.

# Referências bibliográficas:

## Para iniciar:

*1 – Ivan Silva, I.; Spatti, D. e Flauzini, R. - "Redes Neurais Artificiais para Engenharia e Ciências Aplicadas", Artliber, 2010, cap 1-6.

## Para aprofundar:

*1 - Haykin, S., "Neural Networks and Learning Machines" Pearson – Prentice Hall, 2009, Cap 1-7. versão antiga: "Neural Networks, A Comprehensive Foundation", Prentice Hall, 1999. Ver também: Haykin, S., "Redes Neurais, Teoria e Prática", Bookman, 2001.

2 - Bishop, C. M. - "Pattern Recognition and Machine Learning", Springer, 2006, Cap 1 e 3-5.

*3 – Cichocki, A.; Unbehauen, R.- "Neural Networks for Optimization and Signal

## Software:

* Toolboxes: Matlab, Statistica

## Freeware:

* Python

WEKA www.cs.waikato.ac.nz/ml/weka/

## Base de dados

http://archive.ics.uci.edu/ml/

http://archive.ics.uci.edu/ml/datasets.html

**Avaliação:**

**Listas de exercícios** **(apenas aprendizado)**

**1ou 2 Provas** **(avaliação principal)**

## Para que serve ?

**Emular sistemas neuronais biológicos visando obter as propriedades destes sistemas:**

**Redes Neurais Artificiais**

**Aprendizado ?** **Generalização ?** **Robustez ?**

**Aplicações:**

**Cálculo ? Não !!!**

**Simulação de Sistemas não Lineares**
**Reconhecimento de Padrões**
**Otimização**

## De onde veio ?

## Breve Histórico

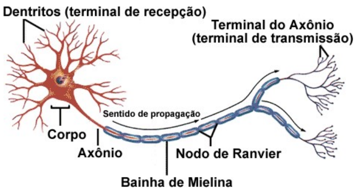
400 A.C. - Aristóteles (filósofo)

*"De memoria et reminiscentia"*

1900 - Biologia & Psicologia:

Santiago Cajal

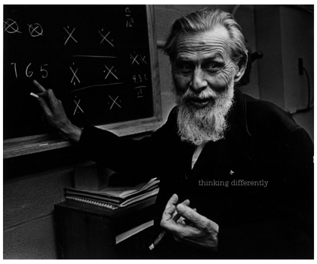
(médico, pai da neurociência moderna, Nobel 1906)



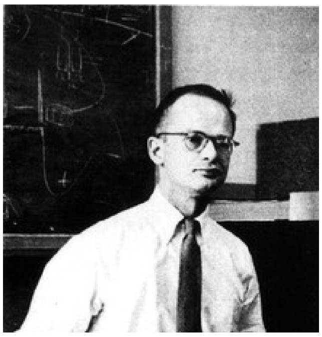

Frederick Skinner, etc.

(psicólogo, filósofo - aprendizagem)

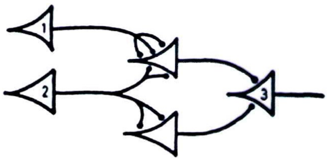
## 1943 – Warren McCulloch
(neuroanatomista, psiquiatra)



## Walter Pitts (autodidata)



*primeiro modelo matemático de neurônio*



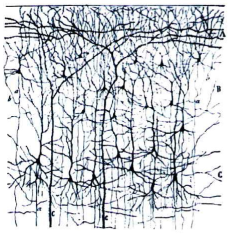



**Figure 4a** The following is the original caption. Kleine und mittelgrosse Pyramidenzellen der Sehrinde eines 20 tägigen Neugeborenen (Fissura calcarina). *A*, plexiforme Schicht; *B*, Schicht der kleinen Pyramiden; *C*, Schicht der mittelgrossen Pyramiden; *a*, absteigender Axencylinder; *b*, rückläufige Collateralen; *c*, Stiele von Riesenpyramiden.

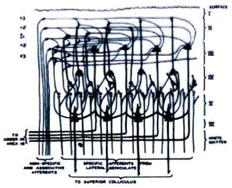
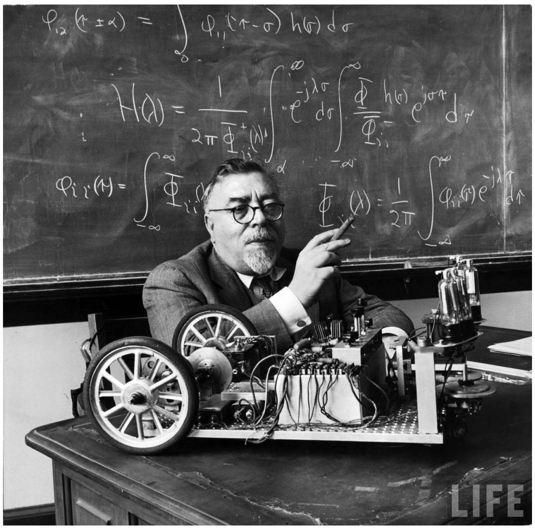



**Figure 3** Impulses relayed by the lateral geniculate from the eyes ascend in specific afferents to layer IV where they branch laterally, exciting small cells singly and larger cells only by summation. Large cells thus represent larger visual areas. From layer IV impulses impinge on higher layers where summation is required from nonspecific thalamic afferents or associative fibers. From there they converge on large cells of the third layer which relay impulses to the parastriate area 18 for addition. On their way down they contribute to summation on the large pyramids of layer V which relays them to the superior colliculus.

## 1947 - Norbert Wiener (matemático, PhD aos 18 anos, pai da cibernética)

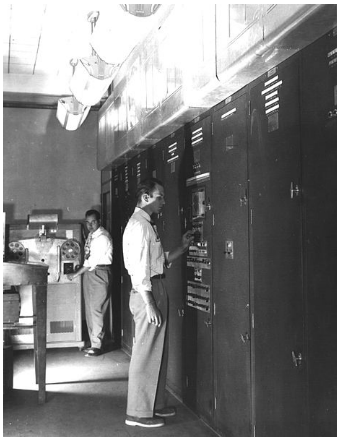
*"Cibernetics"*



Palomilla

1947 - 1951 – John Von Newman (matemático)

*EDVAC -*
*Electronic Discrete Variable*
*Automatic Computer*



6000 válvulas, 12.000 diodos,

56 kW, 45 m$^2$, 8 ton

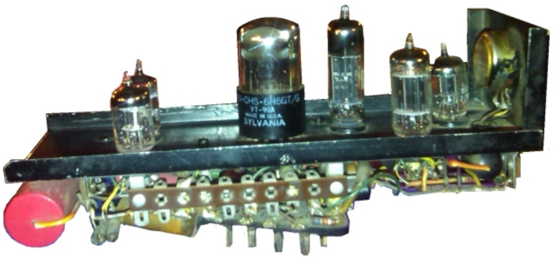
1949 - Donald Hebb (psicólogo)

*primeiro modelo de aprendizado não supervisionado*

1951 – Marvin Minsky (matemático, IA) e Dean Edmonds

*Snark - primeiro computador auto-adaptativo – "Hebbiano"*

"neurônio" do Snark



## 1957 - Frank Rosenblatt

(psicólogo, neurologista)

*Perceptrons - aprendizado em sistemas não lineares*

*Mark I Perceptron Neurocomputer*

*imagens 20x20 pixels*


*Fig. 1.4.* • Frank Rosenblatt (the inventor of the perceptron and designer of the I Perceptron neurocomputer) with the 400 pixel (20 × 20) Mark I Perceptron sensor. Photo courtesy of Arvin Calspan Advanced Technology Center.

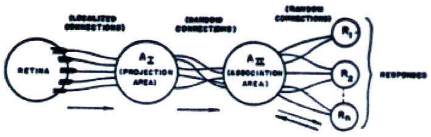
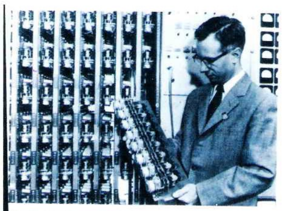



FIG. 1. Organization of a perceptron.



.5. • Charles Wightman holding a subrack of 8 motor/potentiometer pairs. Each /potentiometer pair functioned as a single adaptive weight value. The perceptron ng law was implemented in analog circuits that (when properly wired through the oard shown in Figure 1.6) would control the motor of each potentiometer (the nce of which functioned to implement one weight). Photo courtesy of Arvin Calspan ed Technology Center

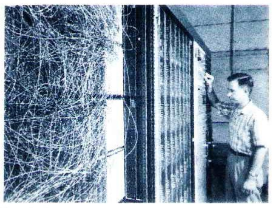
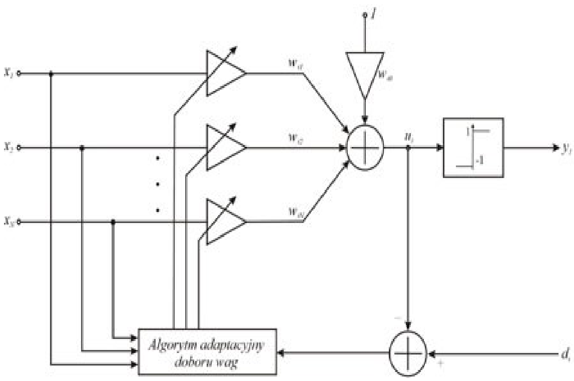


*Fig. 1.6.* • The Mark I Perceptron patchboard. The connection patterns were typically "random", so as to illustrate the ability of the perceptron to learn the desired pattern without need for precise wiring (in contrast to, unlike the precise wiring required in a programmed computer). Photo courtesy of Arvin Calspan Advanced Technology Center.

## 1960 – Bernard Widrow e Marcian (Ted) Hoff (engenheiros)

*Adaline - adaptive linear neuron*

***LMS***- *aprendizado em sistemas lineares*



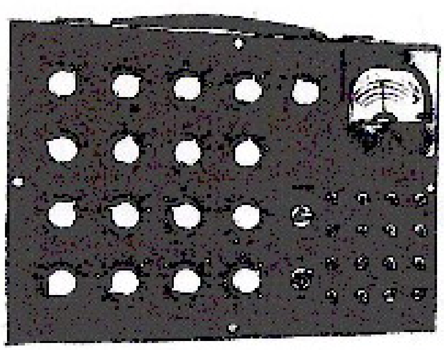



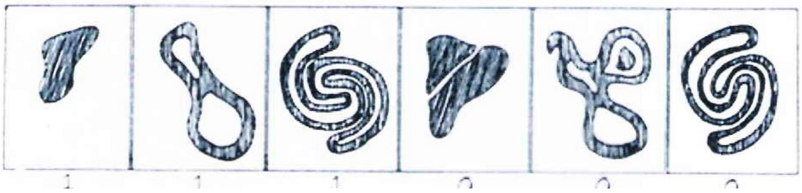
Figure 2 Adaline.

## 1969 – Marvin Minsky
(matemático)

## Seymour Papert
(matemático, educador)

*"Perceptrons"*

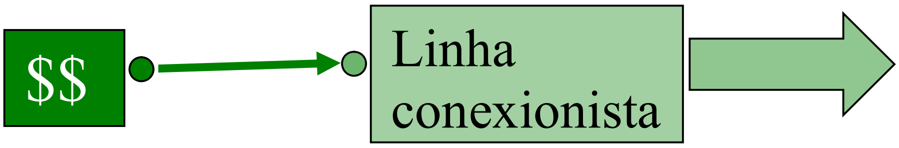
$$\psi_{\text{CONNECTED}}(X) = \begin{cases} 1 & \text{if } X \text{ is a connected figure,} \\ 0 & \text{otherwise.} \end{cases}$$



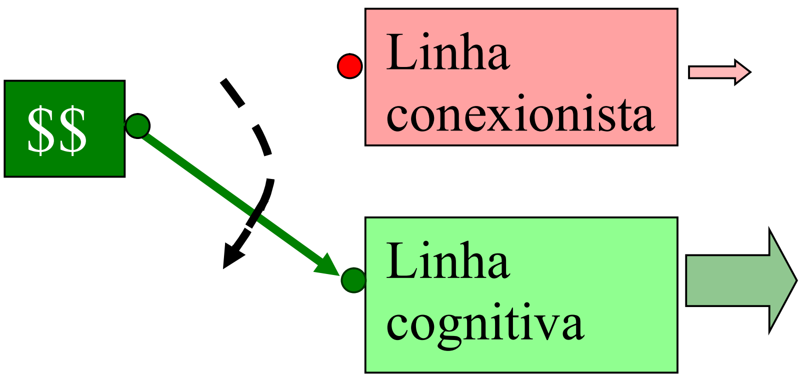
*Exclusive-OR gate*


| A | B | Output |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

*ANTES do "Perceptrons":*




*DEPOIS do "Perceptrons":*



## ”moh ching, moh meng” – sem dinheiro não há vida

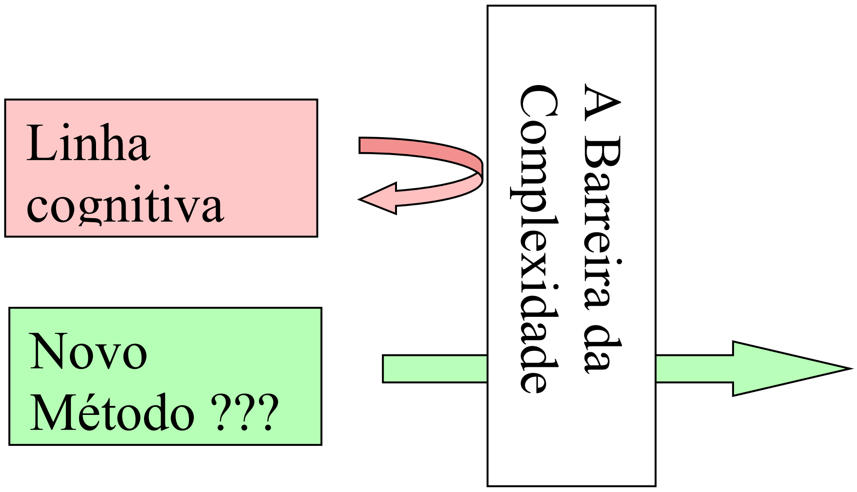
*James Clavel – “Casa Nobre”*

**1969 - 1982** **A época das trevas – O inverno da IA**

Aprendizado não supervisionado, auto-organização

Amari, Anderson, Fukushima, Grossberg, Kohonen

E a linha cognitiva ?



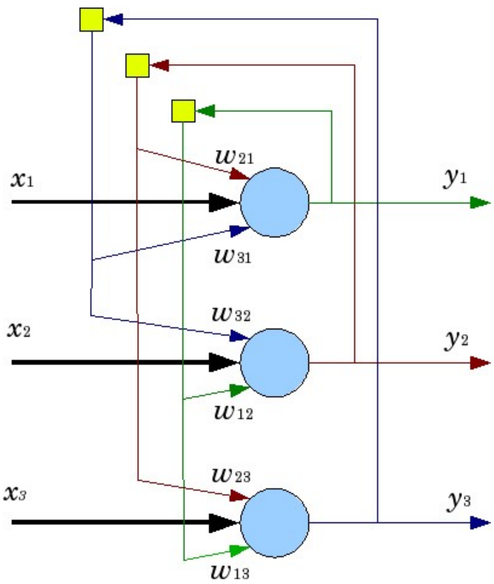
## 1974 - Paul Werbos (matemático)

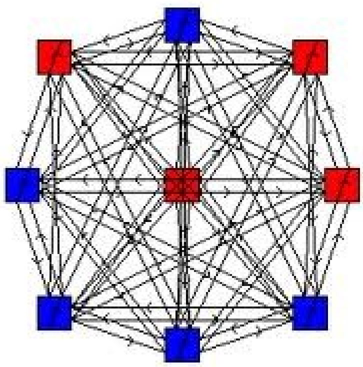
“Beyound Regression”

**CBRN / SBAI Florianópolis, 2007**

## 1982 John Hopfield (físico)

*Redes conexionistas em otimização*

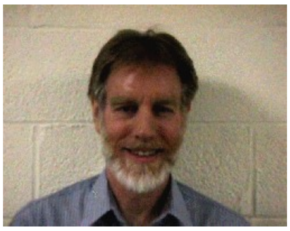


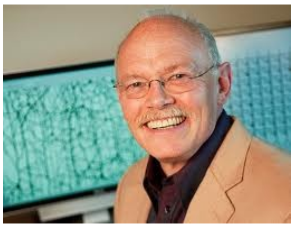



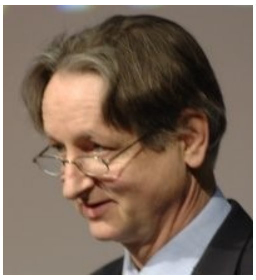
1986 – David Rumelhart (psicólogo)
Ronald Williams (computação)
Geoffrey Hinton (psicologia e IA)

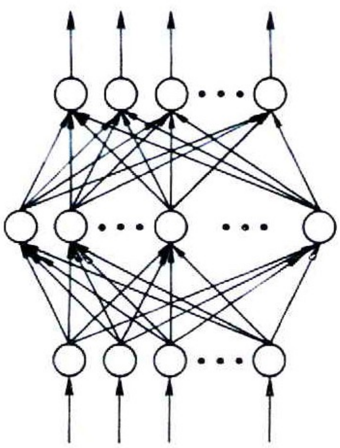
*"PDP (Parallel Distributed Processing) I e II"*

*a **re-**descoberta da Backpropagation.*










*1987 - 1º Congresso em Redes Neurais*
*(IEEE Int. Conf. on Neural Networks - 1700 participantes)*
*Fundação da International Neural Networks Society*

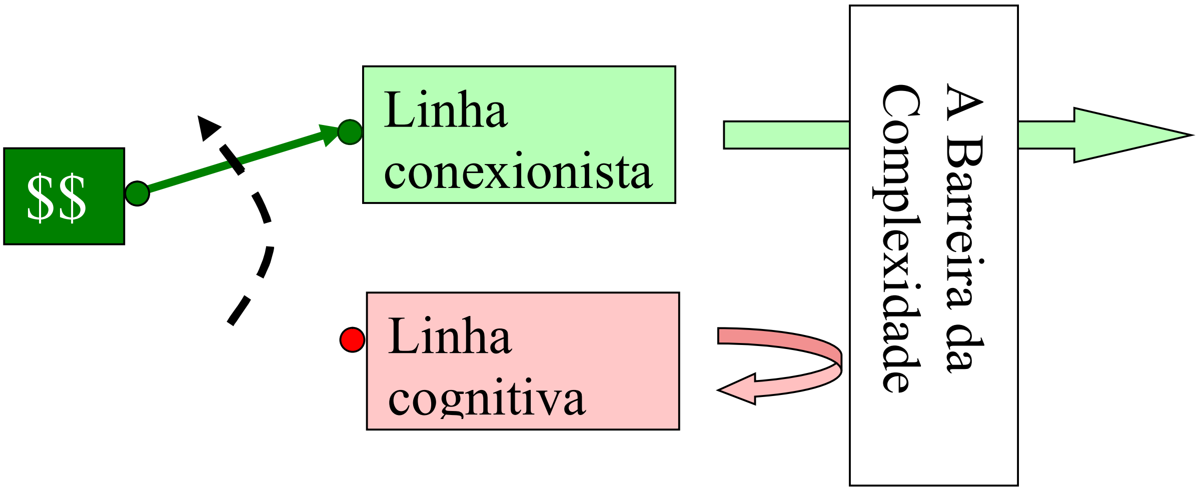
1988 - Broomhead e Low *Redes de Base Radial*

*1988 - Revista "Neural Networks"*

*1989 - IEEE Transactions on Neural Networks*

1992 - 98 – Vapnik *Máquinas de Vetor Suporte*

Geoffrey Hinton *Deep Learning*




**A situação atual:**

**Sociedades científicas:**

International Neural Networks Society, INNS

IEEE Computational Intelligence Society
(antiga IEEE Neural Networks Society)

Sociedade Brasileira de Inteligência Computacional, SBIC,
(antiga Sociedade Brasileira de Redes Neurais, SBRN), www.sbrn.org.br

**Revistas**

Neural Networks, INNS

IEEE Trans. on Neural networks, IEEE

Learning and Non-Linear Models, SBRN, www.sbrn.org.br

**Congressos:**

International Joint Conference on Neural Networks.
anual, da INNS e IEEE-NNS.

Congresso Brasileiro de Inteligência Computacional,
bi-anual, da SBIC.

## Quando usar redes neurais ?

Existe um algoritmo (modelo fenomenológico) satisfatório ?

**SIM - então use o algoritmo**

**NÃO - então pense em usar redes neurais**

## Redes Neurais

**não são a panacéia universal !!**

## Como implementar as Redes Neurais Artificiais ?

## Emulando os sistemas biológicos !

# Neurônio biológico

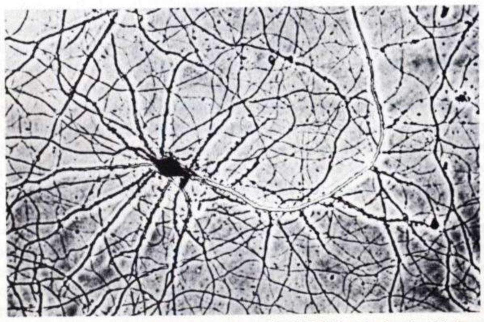
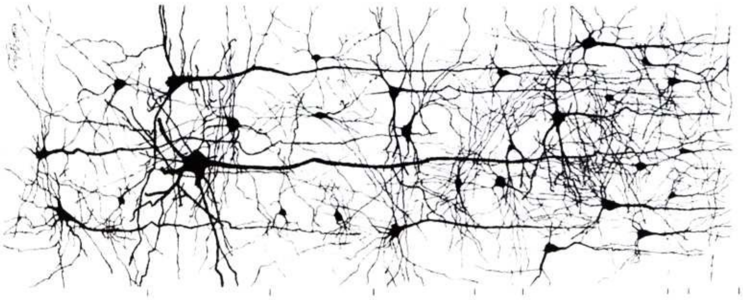
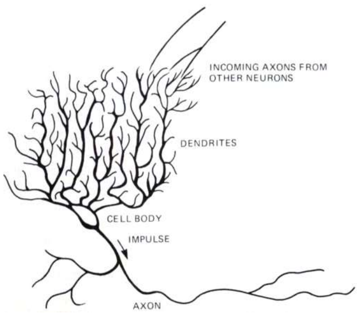
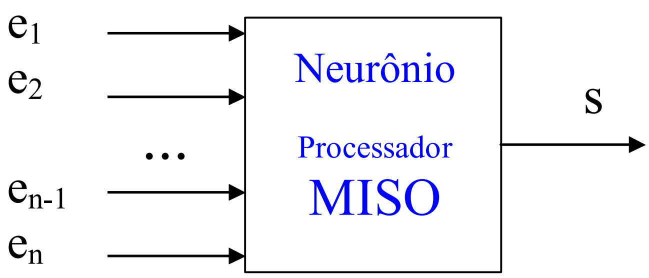


**Figure 1-2a.** A biological neuron magnified 400× with the dendritic tree in the foreground (courtesy of Gary Banker and Aaron Waxman, Univ. of Virginia).



# Neurônio biológico

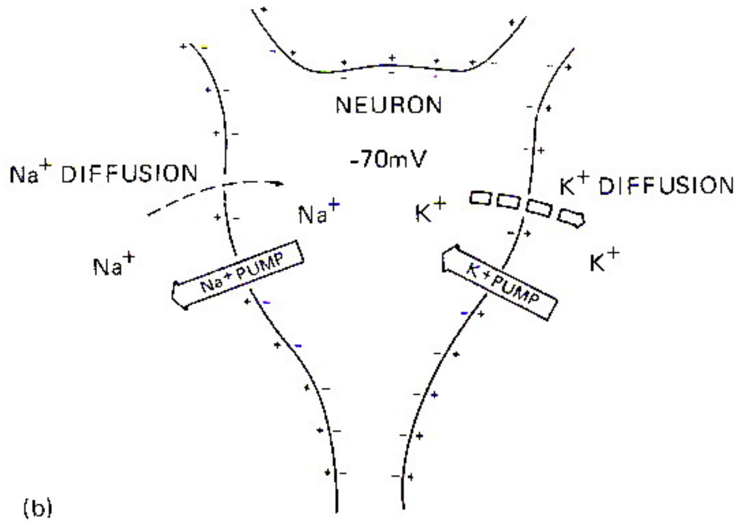


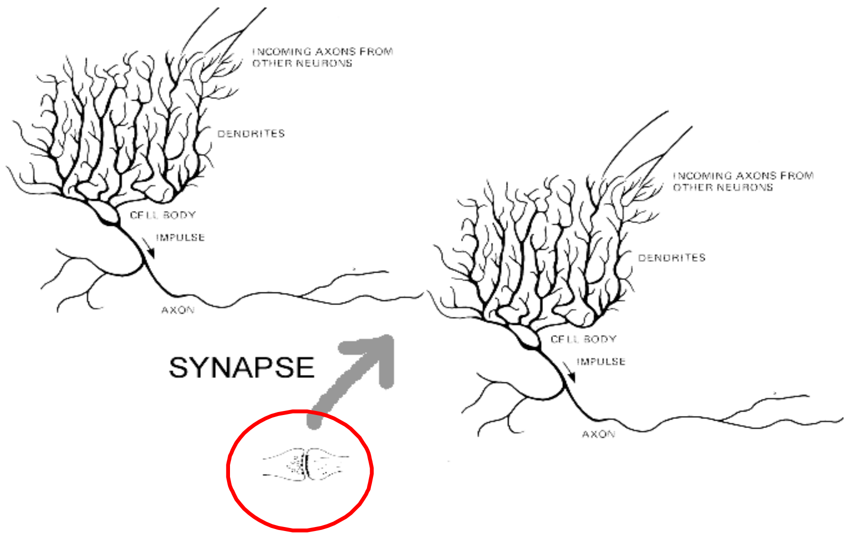

**- 70 mV < saída < + 50 mV**

$$\text{estado} \begin{cases} \text{ativo, excitado} & \text{saída} > s_0 \\ \text{inativo, inativo} & \text{saída} < s_0 \end{cases}$$

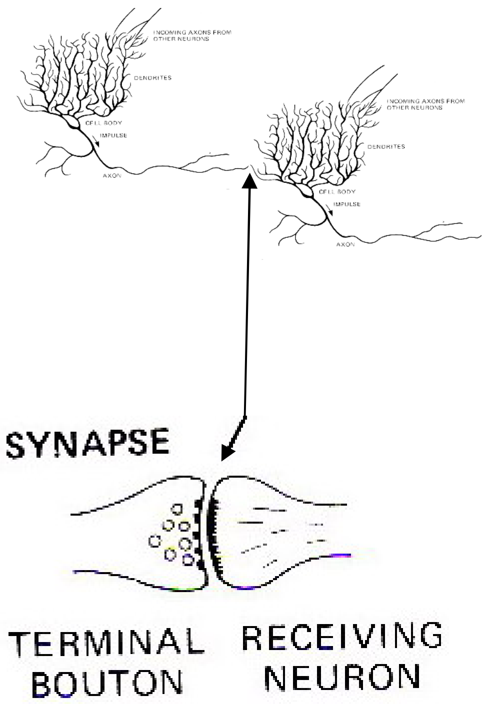
# Polarização dos Neurônios - Bombas de Íons




# Comunicação entre neurônios:



# Sinapse



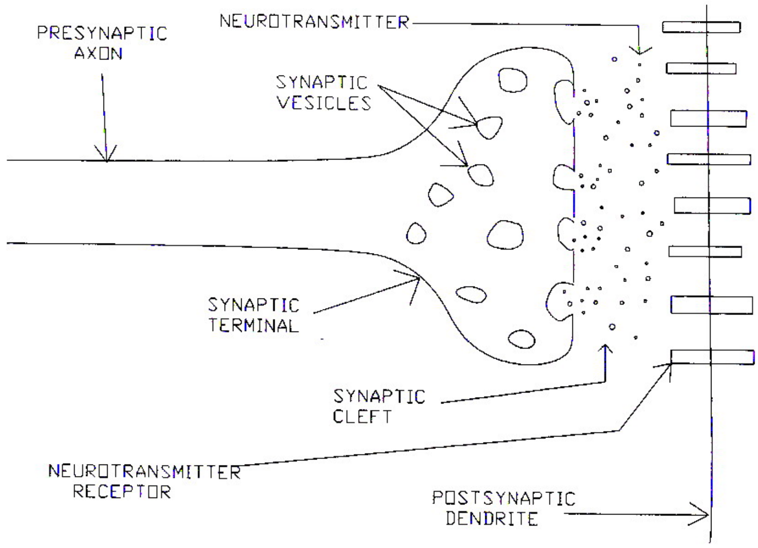
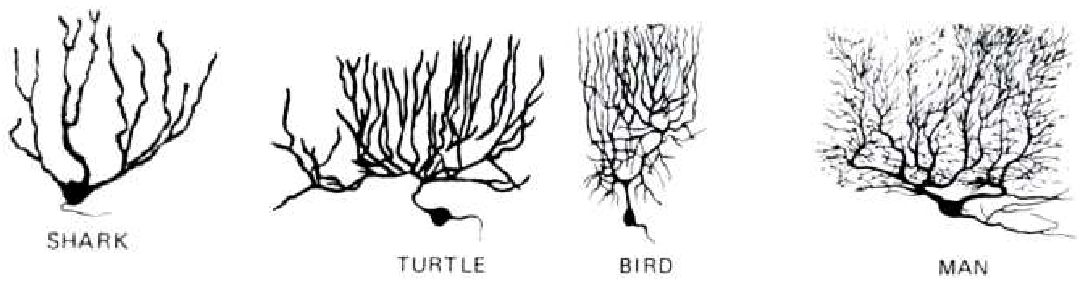




**Figure A-3.** Synapse

**sinapses são ponderadores**

# Memória:



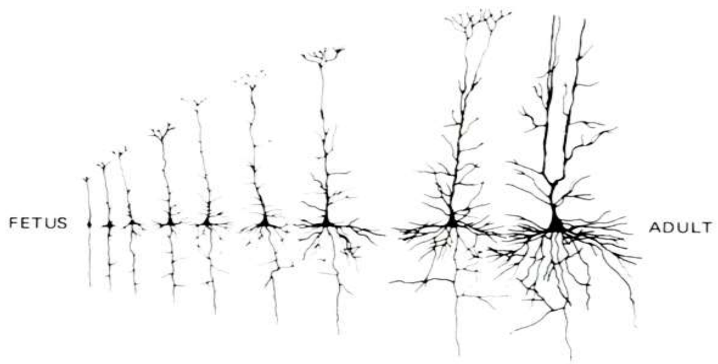
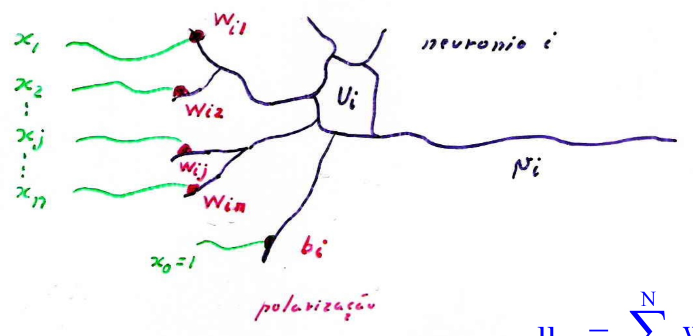




**Figure 7-13.** Growth of the dendritic trees and axon branches of cortical pyramidal cells in the human, from fetus to adult. (Courtesy of Sidman and Rakic 1982, and Poliakov in Sarkisov and Preobrazenskaya 1959.)

# Neurônio - elemento de processamento

Modelo do Neuronio

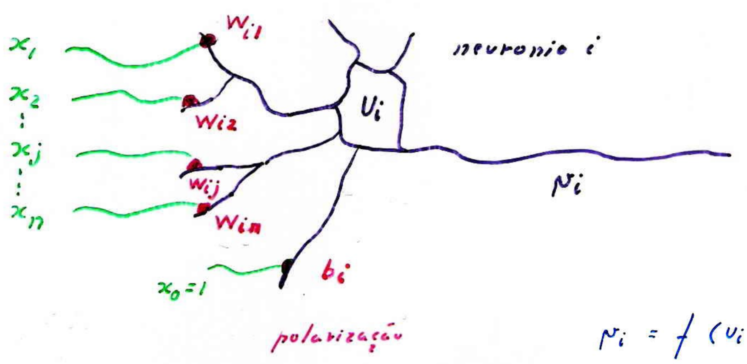



$$u_i = \sum_{j=1}^{N} w_{ij}\, x_j + w_{i0} = \underline{w}_i^t\, \underline{x} + w_{i0}$$

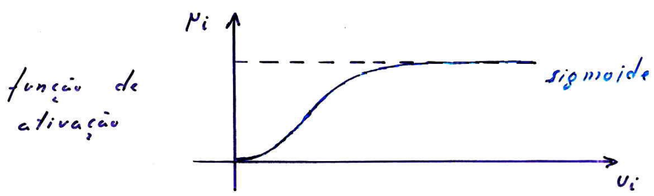
$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} \qquad \underline{w}_i = \begin{bmatrix} w_{i1} \\ w_{i2} \\ \dots \\ w_{in} \end{bmatrix}$$

# Neurônio - elemento de processamento

Modelo do Neuronio

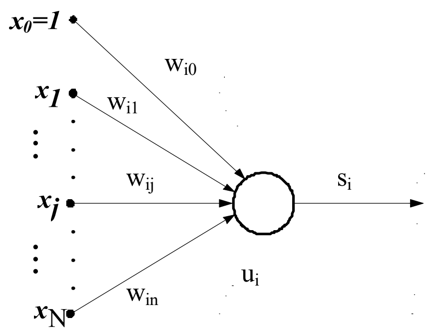


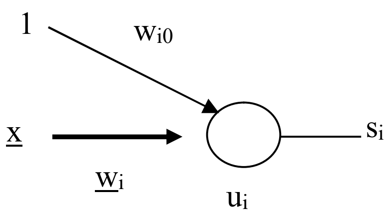

$\mu_i = f(u_i)$




$\frac{d\mu_i}{du_i} \geq 0$

# Neurônio - elemento de processamento

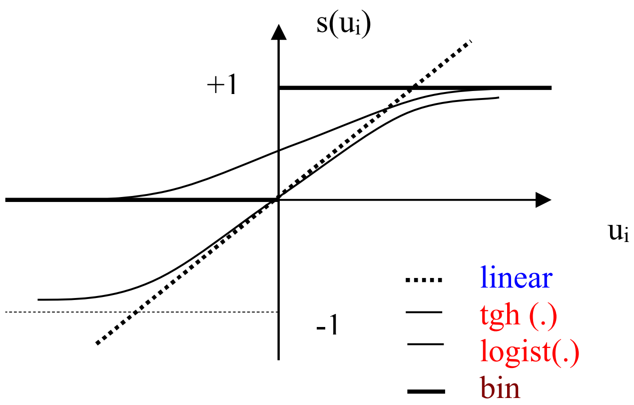



$$u_i = \sum_{j=1}^{N} w_{ij}\, x_j + w_{i0} = \underline{w}_i^t\, \underline{x} + w_{i0}$$

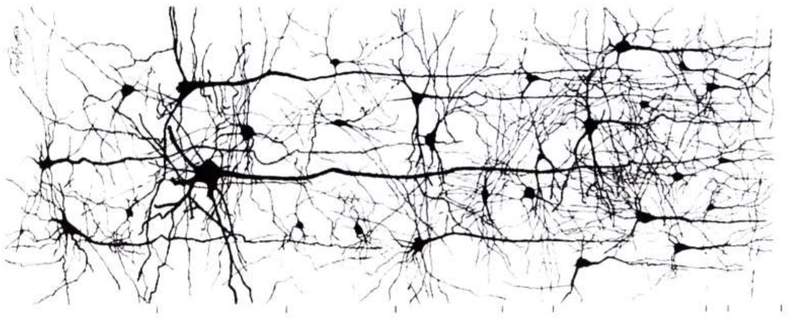
$$s_i = s(u_i)$$

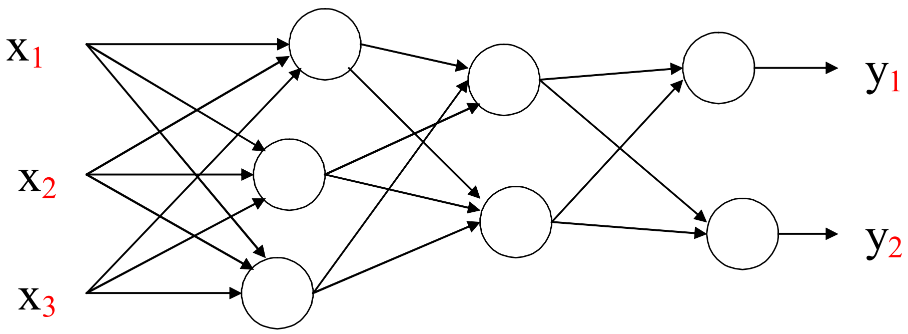
## Função de Ativação s(u)



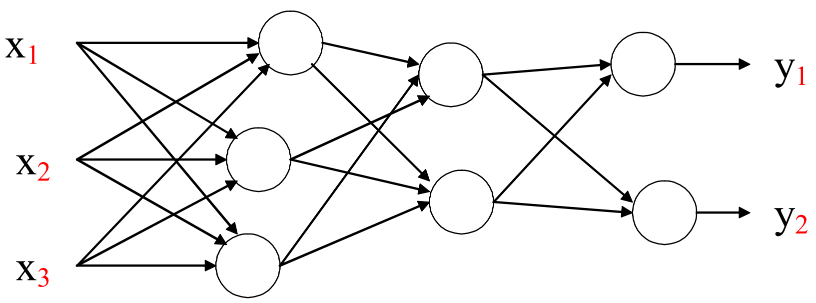

| Neurônio | Função de ativação $s_i(u_i)$ | Ganho linearizado $g_i(s_i) = ds_i / du_i$ |
|---|---|---|
| **linear** | $u_i$ | **1** |
| **Não linear, tipo tgh** | $tgh(u_i) = \dfrac{e^{u_i} - e^{-u_i}}{e^{u_i} + e^{-u_i}} = \dfrac{1 - e^{-2u_i}}{1 + e^{-2u_i}}$ | $1 - s_i^2$ |
| **Não linear, tipo logistica** | $logist(u_i) = \dfrac{1}{1 + e^{-u_i}}$ | $s_i(1 - s_i)$ |
| **Não linear, tipo binário** | $deg(u_i) = \begin{cases} 0 & \text{se } u_i < 0 \\ 1 & \text{se } u_i \geq 0 \end{cases}$ | - |

# Arquitetura da Rede






# Arquitetura da Rede

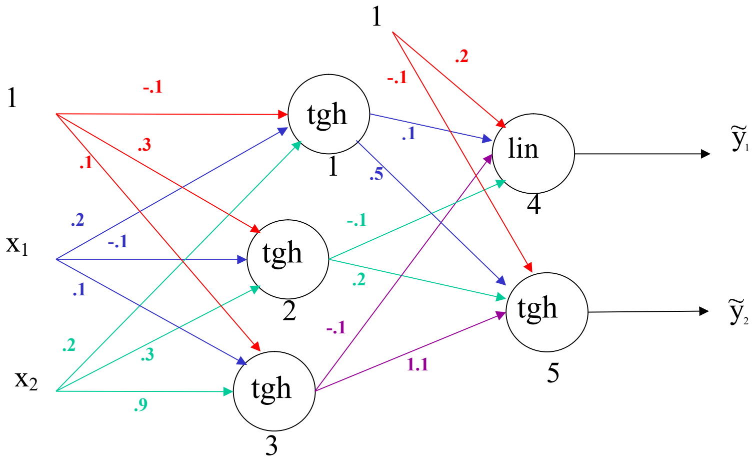



*Rede feedforward*

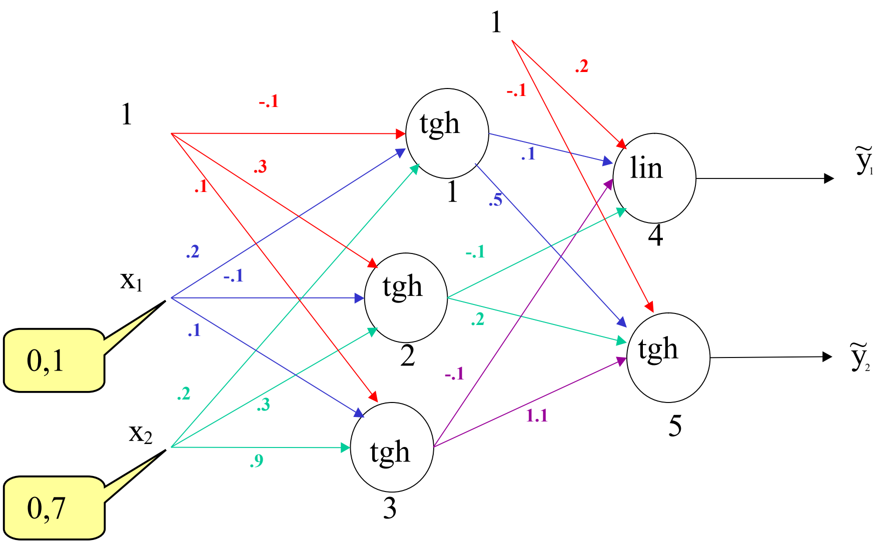
(sem realimentação)

- Estática
- Estruturalmente estável

# Exemplo de operação da rede:



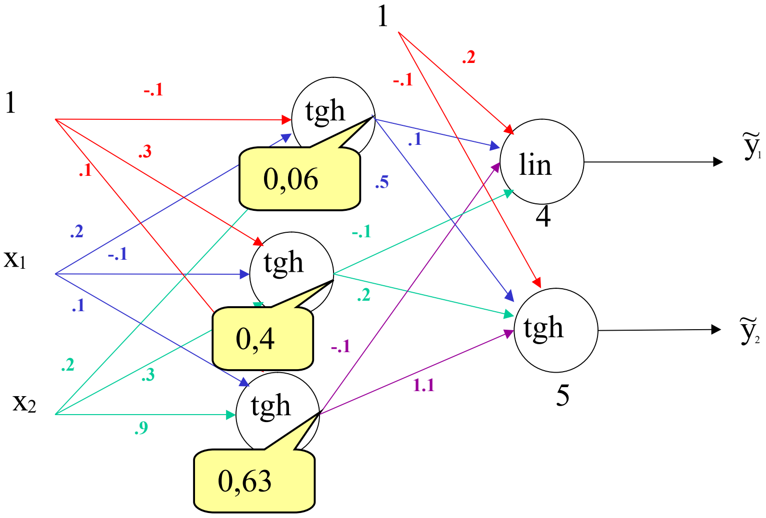

$$\underline{\mathbf{x}} = \begin{bmatrix} 0,1 \\ 0,7 \end{bmatrix} \qquad \tilde{\mathbf{y}} = ?$$




$$u_1 = -0{,}1 + (0{,}2)(0{,}1) + (0{,}2)(0{,}7) = 0{,}06$$

$$v_1 = \mathrm{tgh}(\,0{,}06\,) = 0{,}06$$

$v_2 = 0{,}46$ $\qquad$ $v_3 = 0{,}63$

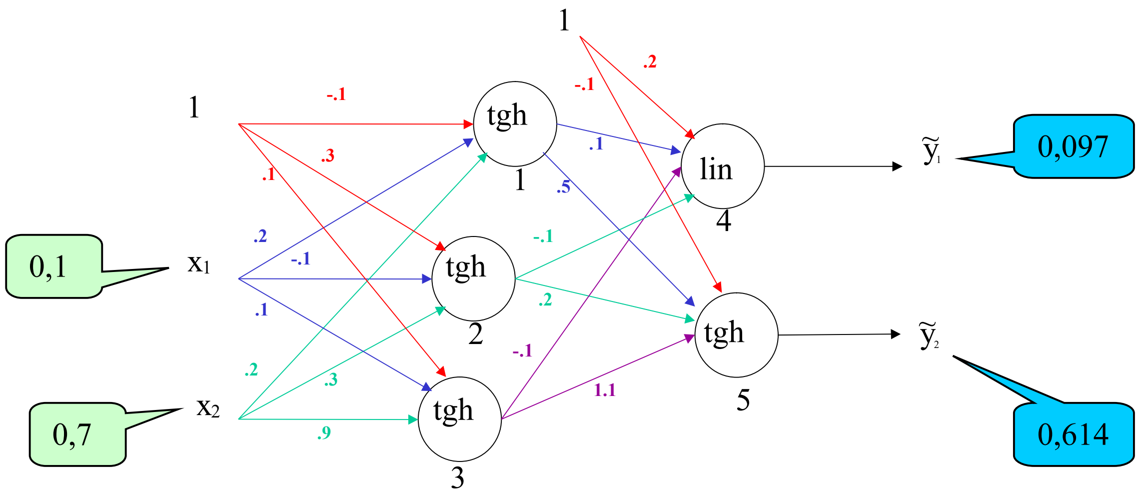



$$u_4 = 0{,}2 + (0{,}1)(0{,}06) + (-0{,}1)(0{,}46) + (-0{,}1)(0{,}63) = 0{,}097$$

$$v_4 = 0{,}097 \quad \text{(linear!)}$$

$$u_5 = -0{,}1 + (0{,}5)(0{,}06) + (0{,}2)(0{,}46) + (1{,}1)(0{,}63) = 0{,}715$$

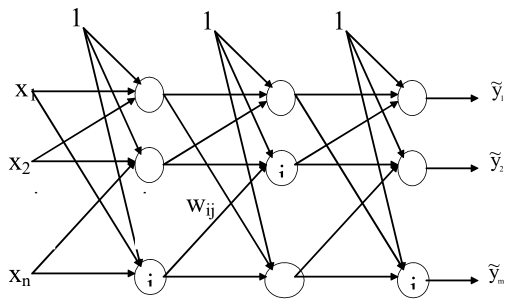
$$v_5 = \text{tgh}(0{,}715) = 0{,}614$$

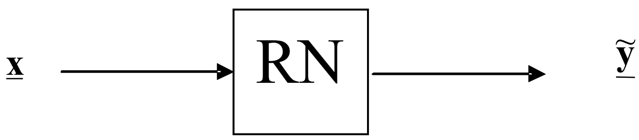



$$\underline{\mathbf{x}} = \begin{bmatrix} 0{,}1 \\ 0{,}7 \end{bmatrix} \qquad \tilde{\mathbf{y}} = \begin{bmatrix} 0{,}097 \\ 0{,}614 \end{bmatrix}$$

# Redes Neurais *FeedForward*

## Mapeador Não Linear




$$\underline{\tilde{y}} = \varphi\left(\underline{x}\right)$$

**Aproximador Universal !**