

PSI 2222 – Inteligência Computacional e Redes Neurais em Engenharia Elétrica

Prof. Emilio Del Moral Hernandez

**Escola Politécnica da Universidade de São Paulo
Departamento de Eng. De Sistemas Eletrônicos**



emilio_del_moral@ieee.org

www.lsi.usp.br/~emilio

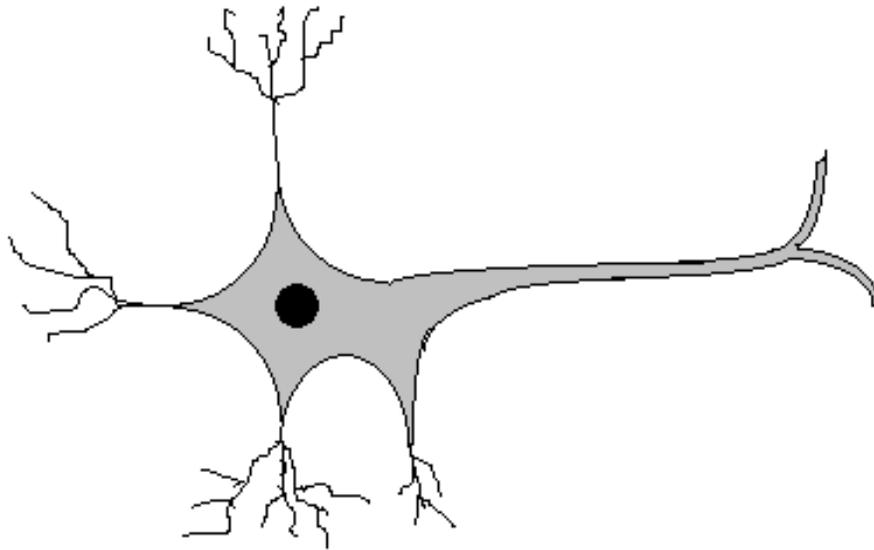
Inteligência Computacional e Neurocomputação

- **Neurocomputação**
- **Inteligência Artificial Simbólica**
- **Fuzzy Logic**
- **Algoritmos Genéticos**
- **Classificação Estatística**
- **Filtros Adaptativos**
- **Otimização**
- **Identificação de modelos paramétricos**
- **Aprendizado de máquina**
- **Sistemas para o apoio à decisão**
- **Processamento de informações não estruturadas**
- **Processamento / reconhecimento / classificação de sinais e imagens**
- **Interfaces homem-máquina naturais**
- **Modelagem de sistemas não lineares multivariáveis**
- **Bifurcação e dinâmica caótica no contexto de neurocomputação**

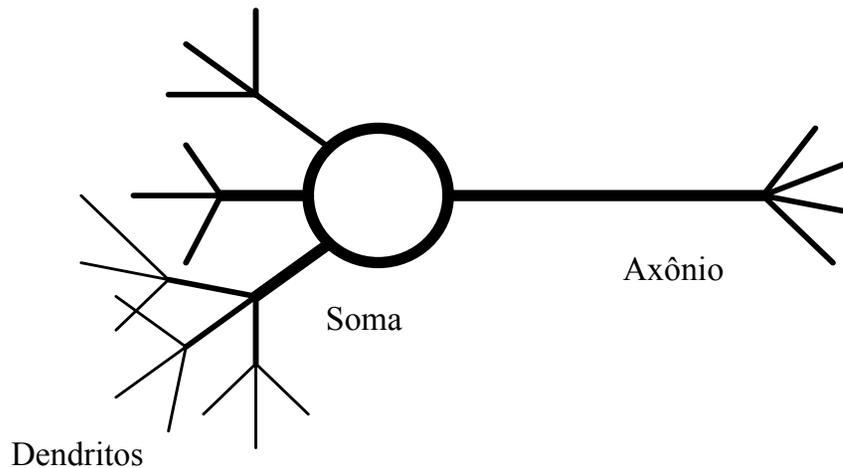
Redes Neurais Artificiais

- São sistemas computacionais, de implementação em hardware ou software, que *imitam* as habilidades computacionais do sistema nervoso biológico, usando um grande número de neurônios artificiais simples e interconectados entre si.
- Algumas aplicações típicas ...
 - Reconhecimento de caracteres
 - Reconhecimento e Síntese de Voz
 - Visão artificial
 - Riscos de inadimplência / detecção de padrões de risco
 - Previsão de vendas / previsão de séries temporais

Dos Neurônios Biológicos aos Artificiais



**Neurônio Biológico
e
Neurônio Artificial**



Redes Neurais Artificiais - Histórico

- **Década de 40: o neurônio McCulloch and Pitts**
- **Década de 50: Em 1956 na 1ª Conferência Internacional de Inteligência Artificial, foi apresentado um modelo de rede neural artificial pelo pesquisador Nathaniel Rochester, da IBM.**
- **Em 1959, foi demonstrada a rede Adaline**
- **Década 1960: a rede Perceptron de Rosenblatt**
- **Em 1969, Marvin Minsky e Seymour Papert publicaram um livro chamado “Perceptrons”**

Redes Neurais Artificiais - Histórico

- **Em 1982, John Hopfield, apresentou um sistema neural artificial que permitia memorizar imagens nas interconexões entre os neurônios.**
- **Ainda em 1982 a primeira conferência Estados Unidos - Japão sobre redes neurais realizou-se na cidade de Kioto.**
- **No Brasil, em 1994 ocorreu o primeiro congresso de Redes Neurais.**
- **Quadro atual ... SBRN / IJCNN / ICANN / ENNS / NNS etc**

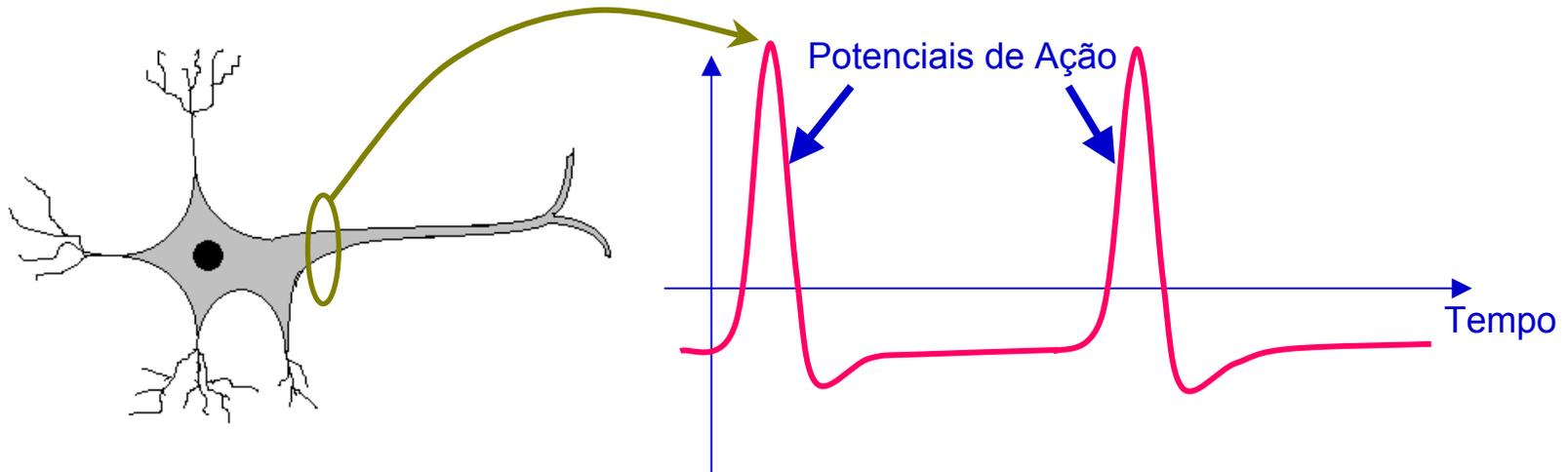
O Neurônio Biológico

- **Décadas de 50 e 60: neurônio biológico passou a ser visto como um dispositivo computacional elementar do sistema nervoso, com muitas entradas e uma saída.**

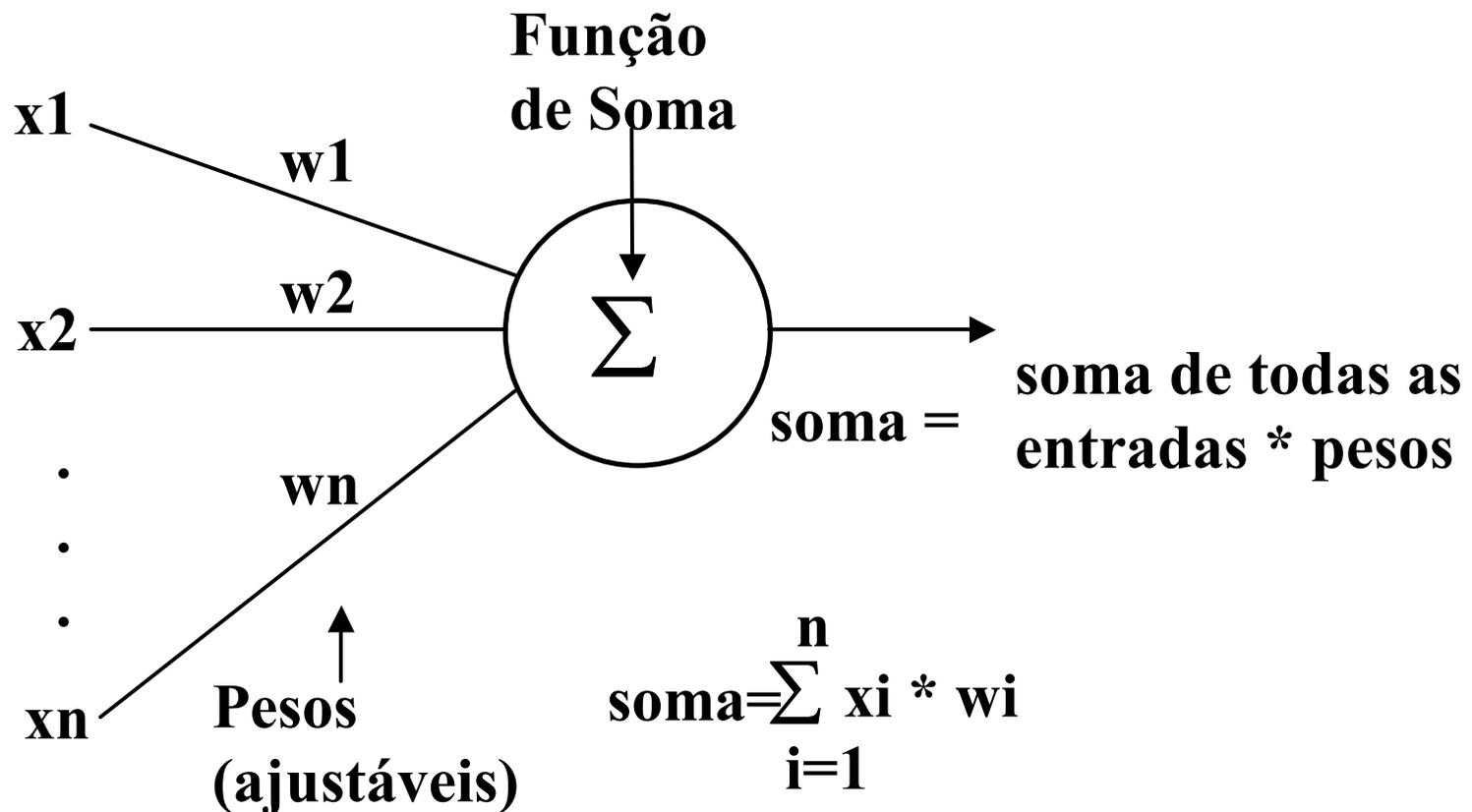
Entrada - Conexões Sinápticas

Informação - Impulsos Nervosos

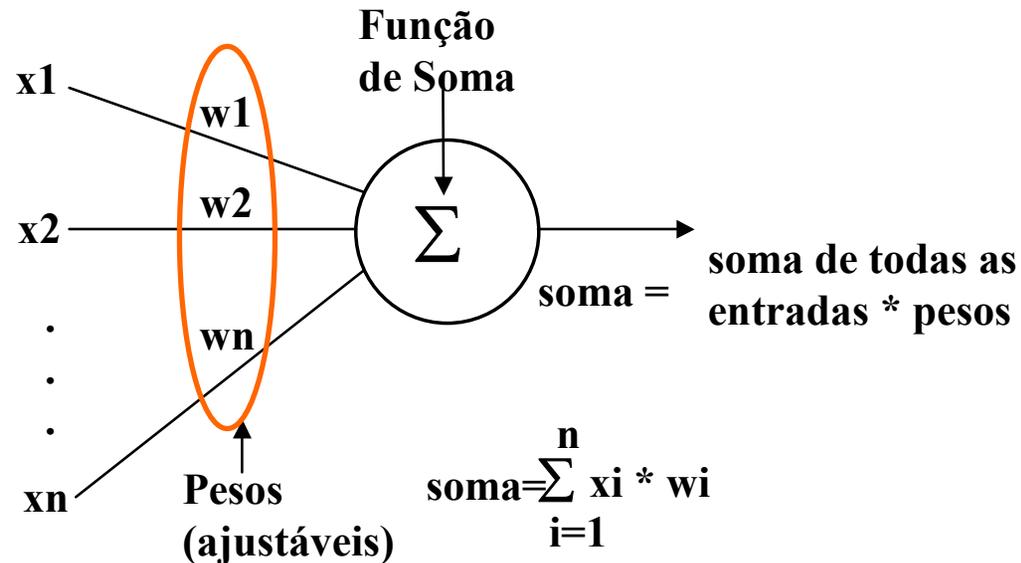
- **Impulso Nervoso: depolarização da membrana celular**



Adaline: Passos Iniciais ... um Modelo Linear



Adaline como Previsor Linear / Plasticidade sináptica

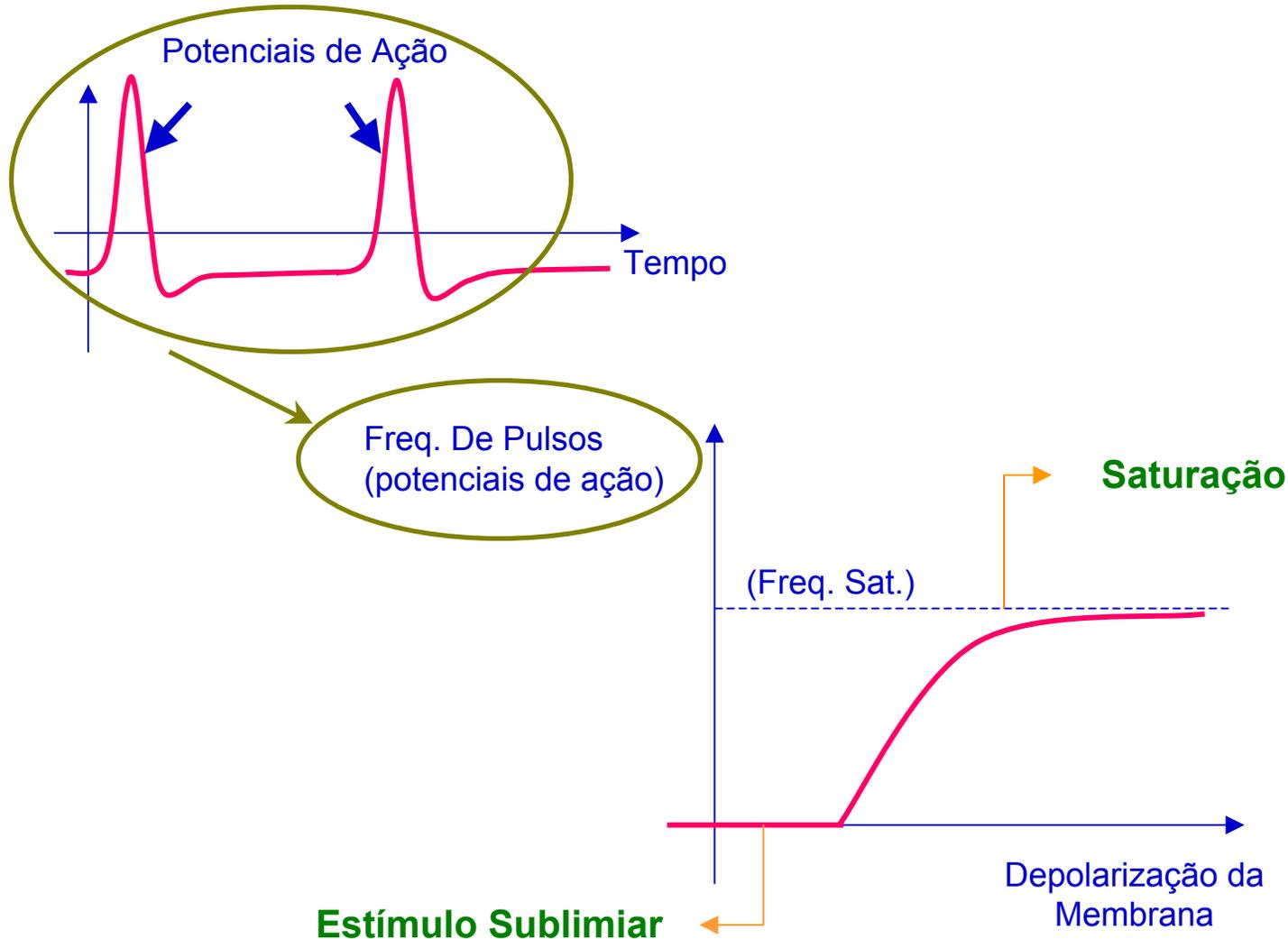


Filtros adaptativos / previsores lineares

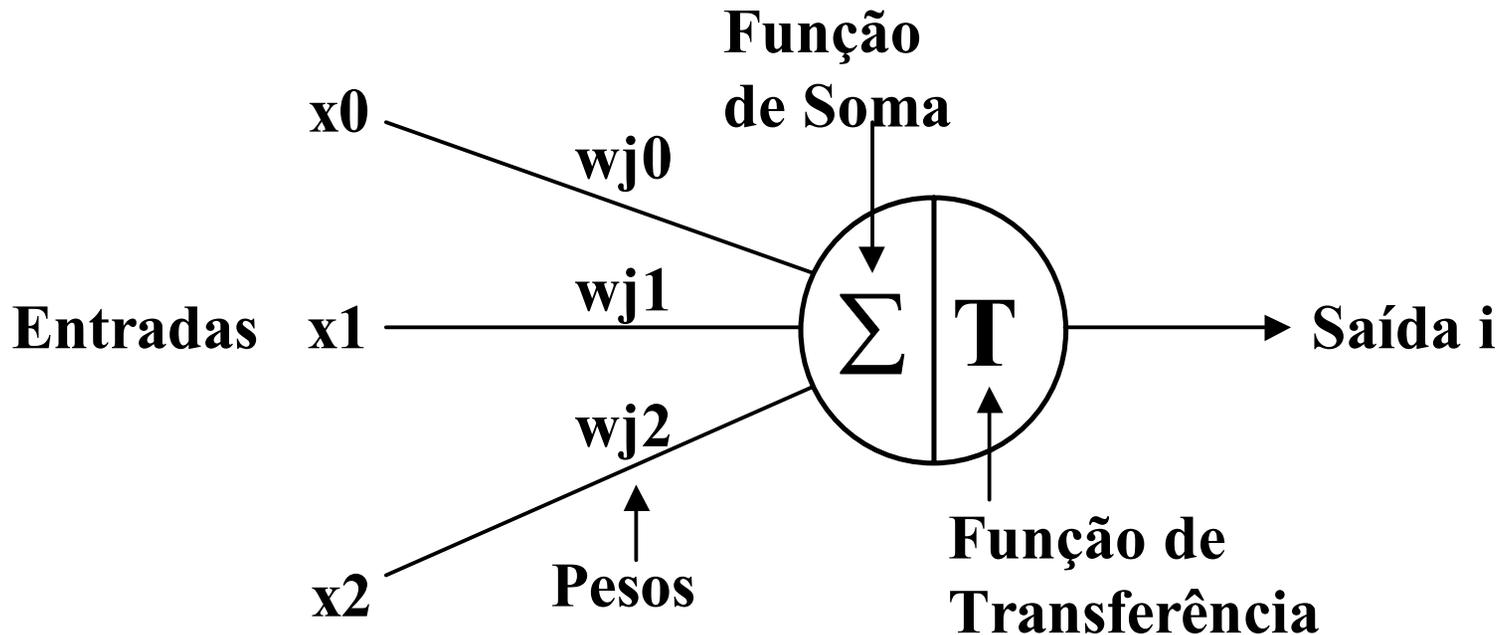
$$x(n+1) = a.x(n) + b.x(n-1) + c.x(n-2) + \dots$$

**a, b, c ... São adaptados para otimizar a previsão
Face a um conjunto de treinamento**

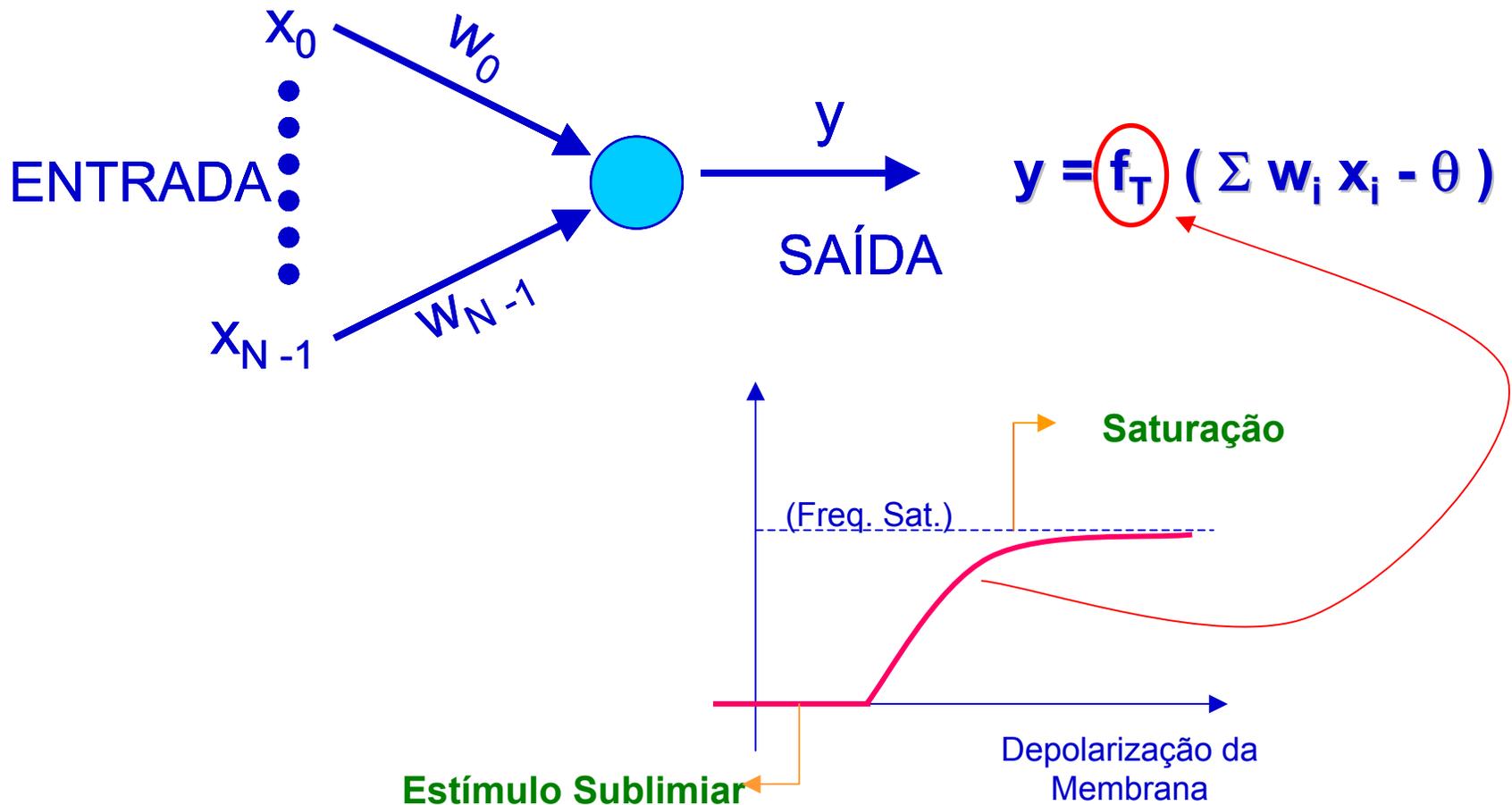
... Vendo com mais detalhe ... a relação entre estímulo e atividade na saída não é linear:



Modelando Melhor a Relação Entrada / Saída



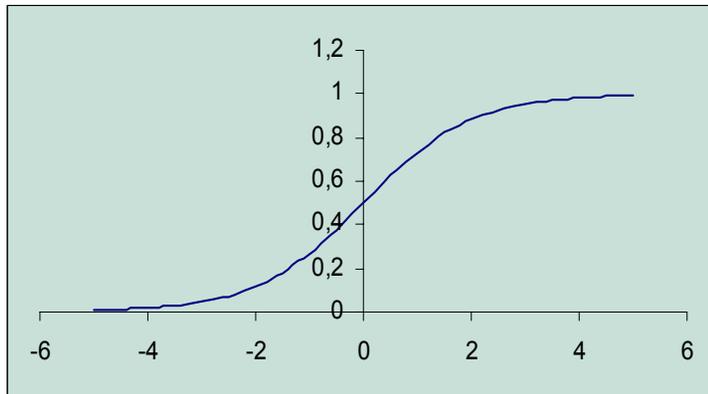
Resumo ... um nó (neural) realiza a seguinte computação analógica



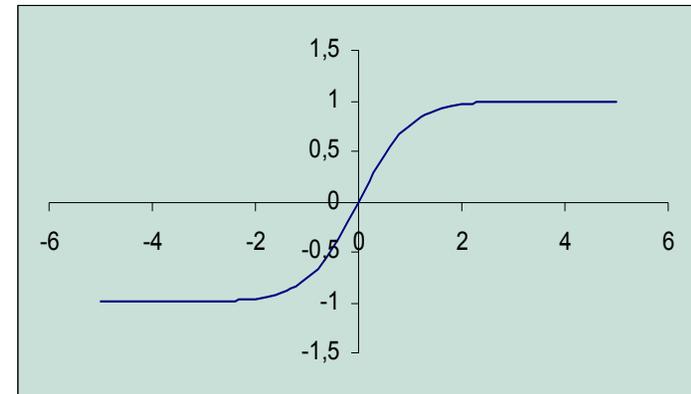
Função de transferência não linear do tipo sigmóide ou tangente hiperbólica

- Preserva - se o fenômeno da saturação
- Preserva - se a monotonicidade na faixa dinâmica
- Temos funções matematicamente amigáveis

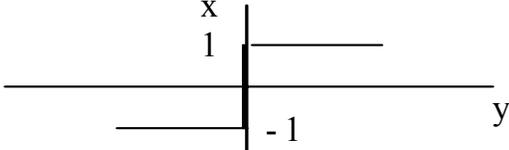
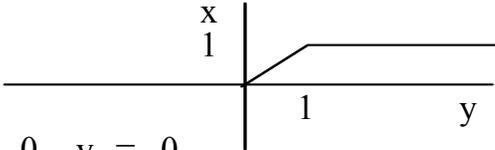
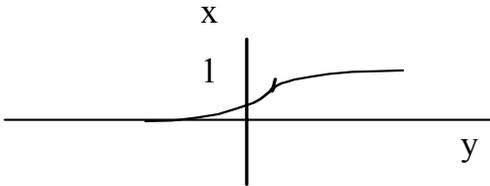
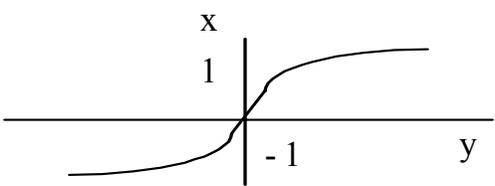
SIGMÓIDE



TANGENTE HIPERBÓLICA



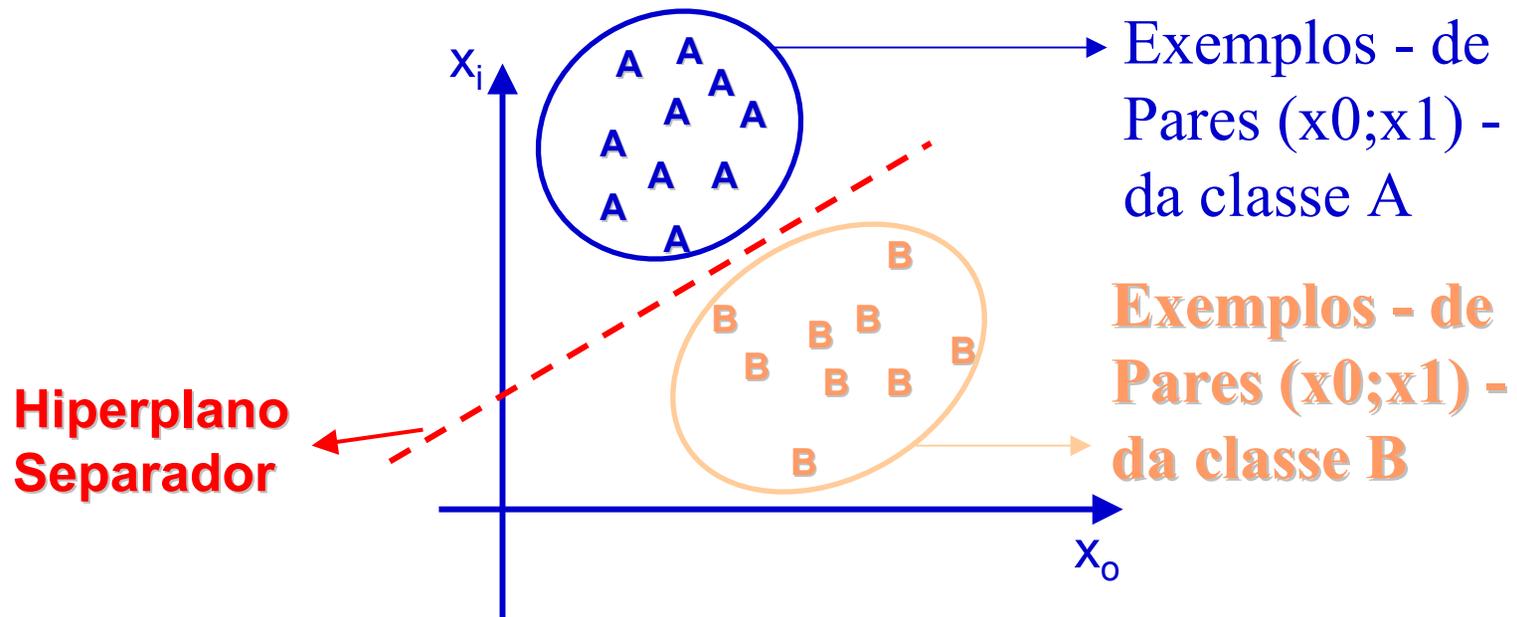
Outras funções de transferência não linear

<p>Hard Limiter (limite rápido)</p>  <p> $s < 0, y = -1$ $s > 0, y = 1$ </p>	<p>Ramping Function (função de rampa)</p>  <p> $s < 0, y = 0$ $0 \leq s \leq 1, y = s$ $s \geq 1, y = 1$ </p>
<p>Sigmoid Function (função sigmóide)</p>  <p> $y = 1 / (1 + e^{-s})$ </p>	<p>Sigmoid Function (função sigmóide)</p>  <p> $x \geq 0, y = 1 - 1 / (1 + s)$ $x < 0, y = -1 + 1 / (1 - s)$ </p>

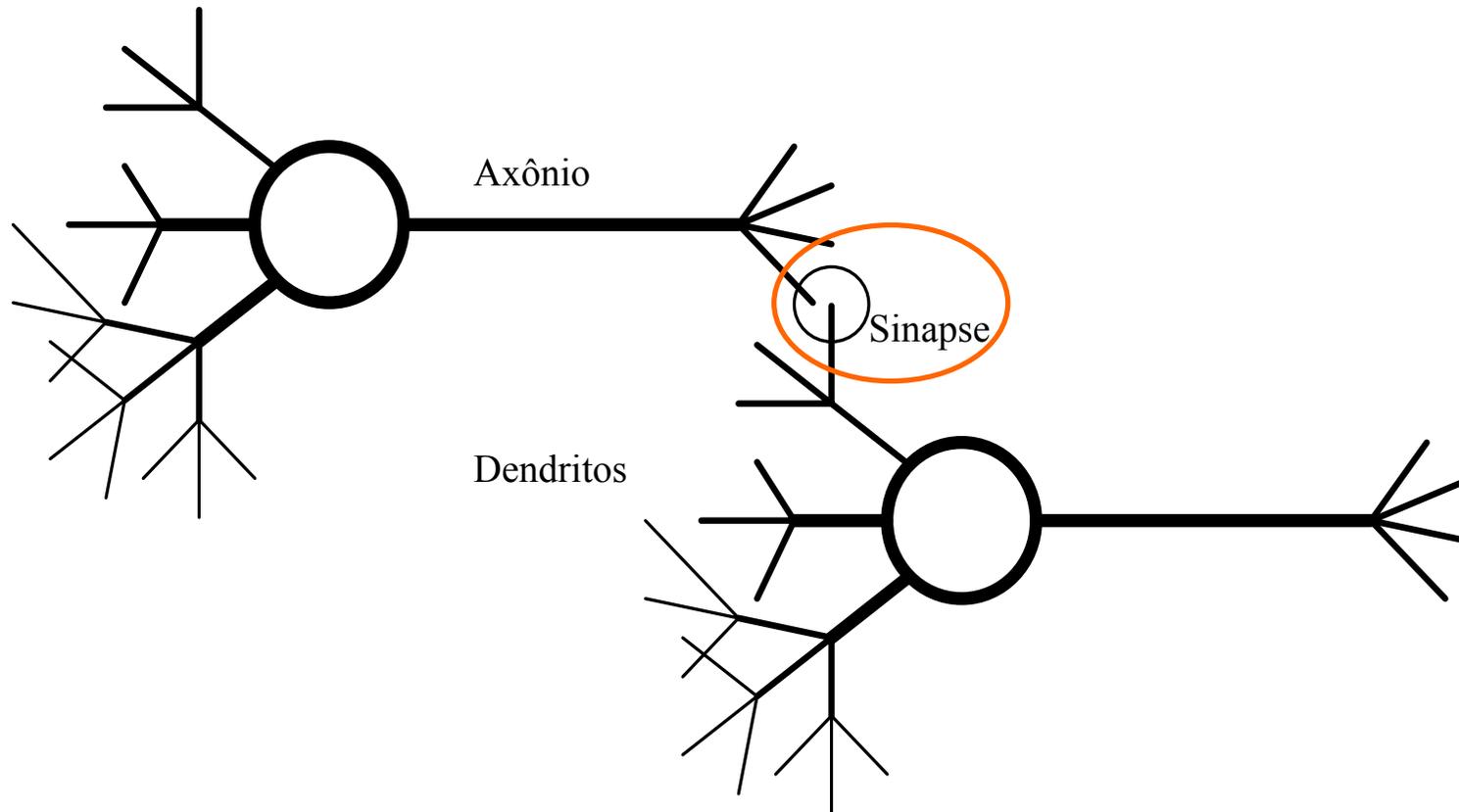
- Com escalamento do argumento, pode-se abarcar os universos digital e analógico / linear e não linear simultaneamente

O Perceptron: $y = \text{signal}(\sum w_i x_i - \theta)$ (função de transferência tipo “degrau”)

- Viabiliza a classificação de padrões com separabilidade linear
- O algoritmo de aprendizado adapta os W s de forma a encontrar o hiperplano de separação adequado
- Aprendizado por conjunto de treinamento



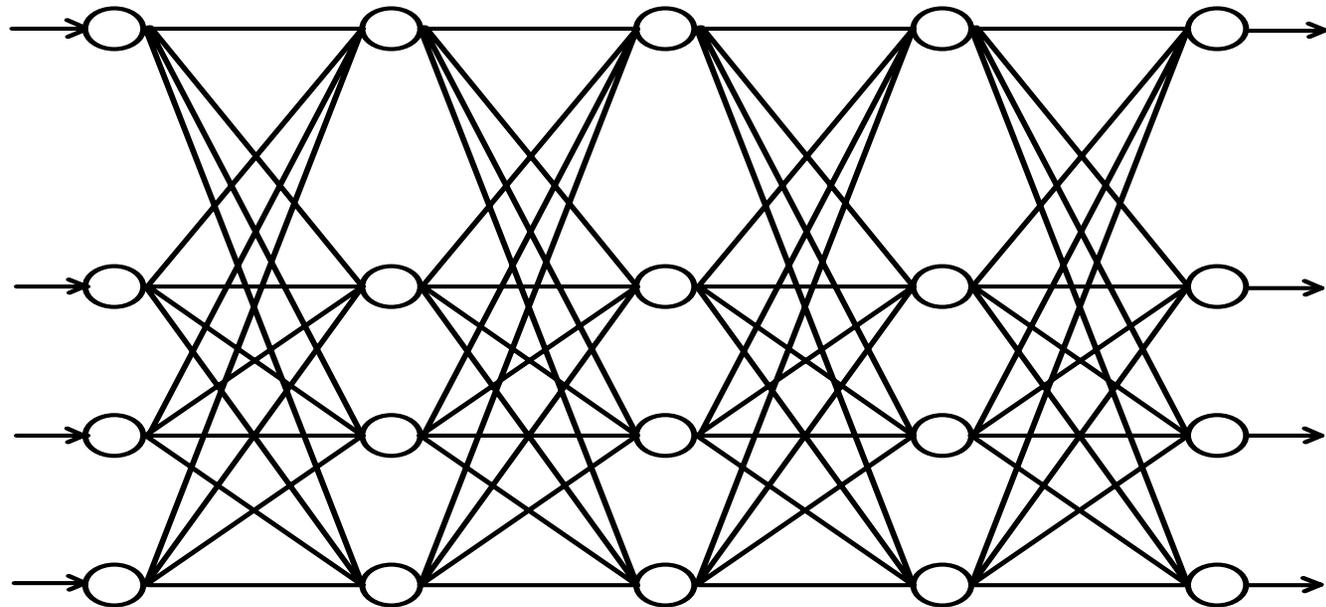
Cômputos mais complexos ... são realizados pelo encadeamento de vários nós



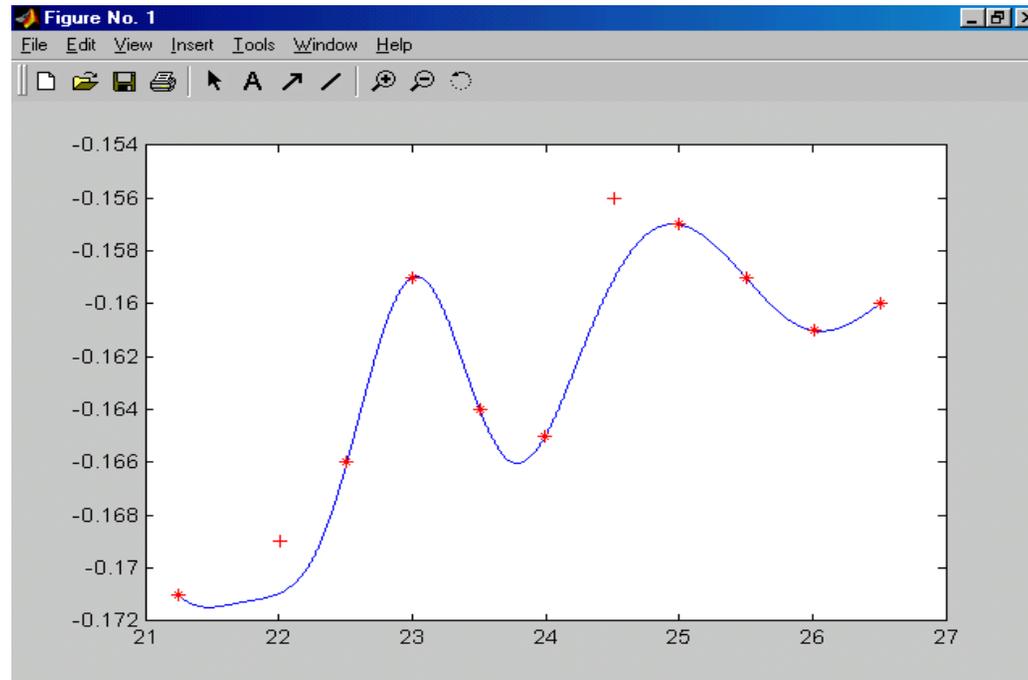
A conexão entre um axônio de um neurônio e um dendrito de outro é denominada **Sinapse**

O Multi Layer Perceptron (MLP)

- Múltiplas entradas / Múltiplas saídas
- Ambas podem ser analógicas ou digitais
- Não há mais a restrição de separabilidade linear



Aproximação de funções com o MLP / superposição de sigmóides deslocadas



- Ex: aplicação em metrologia e linearização de sensores - trabalho de mestrado de Itamar Magno Barbosa

Outras aplicações importantes do MLP

Além de aproximação de funções genéricas ...

- **Fusão não linear de grandezas analógicas multidimensionais**
- **Previsão de séries temporais não lineares**
- **Classificação de Padrões multidimensionais sem separabilidade linear**

- **Note que o aprendizado por exemplos do MLP permite que ele realize as funções acima sem a necessidade de um modelo matemático conhecido / confiável**

Fusão não linear de grandezas analógicas multidimensionais

- $y = F(x_1, x_2, x_3, x_4, \dots)$... F desconhecida

Possibilidade: regressão linear / regressão não linear

Problema: Com que forma funcional???

- Chute 1: $y = a.x_1 + b.x_2 + c.x_3 + d.x_4 + \text{etc} \dots$
- Chute 2: $y = a.x_1.x_1 + b.x_1.x_2 + c.x_2.x_2 + d.x_1.x_3 + \dots$
- Chute 3: $y = [a.x_1.x_2 + c.\log(x_3)] * d.x_4 + \text{etc} \dots$
- Chute 4: $y = \dots$

Fusão Não Linear – Aproximação Universal de Funções Quaisquer Multivariadas

- $y = F(x_1, x_2, x_3, x_4, \dots)$... F desconhecida

Teorema de Kolmogorov / Cybenko:

$F(x_1, x_2, x_3, x_4, \dots) \sim a \cdot f_1(x_1) + b \cdot f_2(x_2) + f_3(x_3) \dots$
sendo a aproximação arbitrariamente próxima

Cybenko >> adaptou Kolmogorov para $f_1, f_2 \dots$
sintetizadas por soma ponderada de sigmoidais
(ou seja, F arbitraria e´ implementavel em RNA !!)

Aproximação Universal / Outras Implicações

- **Classificação de Padroes:**

Pertinencia a Classe A = $F_A(x_1, x_2, x_3, x_4 \dots)$

... F_A , função digital desconhecida

Pertinencia a Classe B = $F_B(x_1, x_2, x_3, x_4 \dots)$

... idem para classe C, D, ... etc

- **Previsao de Series Temporais nao lineares:**

$x(n+1) = F(x(n), x(n-1), x(n-2), \dots)$

... F , função generica desconhecida

“Aprendizagem” das Redes Neurais

- O aprendizado por exemplos do MLP permite que ele realize diversas funções sem a necessidade de um modelo matemático conhecido / confiável
- *“Redes Neurais possuem capacidade de aprender por exemplos”*
- *“As informações são apresentadas a rede que extrai informações a partir dos padrões apresentados”*

Aprendizado Supervisionado

- As informações são apresentadas à rede sob forma de padrões de entrada e os resultados desejados são conhecidos previamente.
- O “supervisor” verifica a saída da rede e a compara com a saída esperada
- Minimização da diferença
- Os algoritmos mais conhecidos são :
 - Regra Delta
 - Backpropagation

O Ajuste sináptico durante o aprendizado

- Quando ocorre erro no reconhecimento / tratamento de uma entrada, um ajuste sináptico é necessário.
- O ajuste sináptico representa o aprendizado em cada neurônio do fato apresentado
- O ajuste sináptico procura corrigir os pesos de modo que se produza a saída desejada diante da respectiva entrada.
- Esse cálculo visa somar ao peso atual, um valor que corresponda a quantidade de erro gerada pela rede, e desta forma corrigir o valor do peso.
- O conhecimento dos neurônios reside nos pesos sinápticos.

Exemplo de ajuste sináptico



Algoritmo:

- .Aplicar entrada X
- .Comparar saída Y com a saída desejada
- .Se saída estiver errada então
 - .Calcular ajuste em pesos
- .Aplicar nova entrada

Exemplo de ajuste sináptico

A etapa “calcular erro” é obtida através da Regra Delta:

$$w_i(n+1) = w_i(n) + \Delta$$

Regra Delta “traduzida”...

Erro = saída esperada - saída obtida

*Peso novo = Peso antigo + (Erro * Entrada * Constante)*

Outras aplicações importantes do MLP

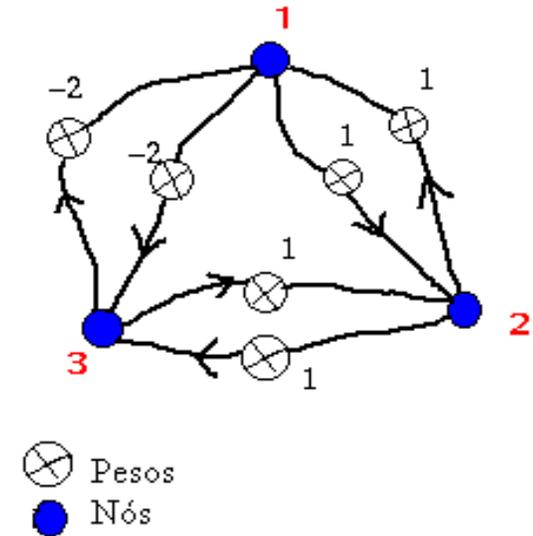
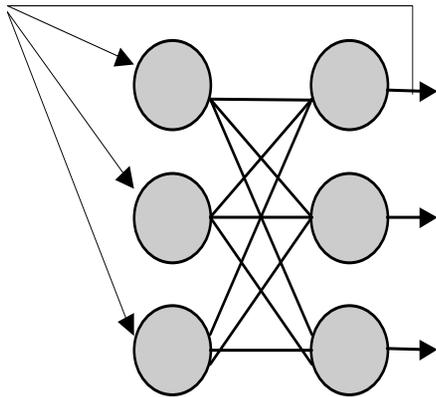
Além de aproximação de funções genéricas ...

- **Fusão não linear de grandezas analógicas multidimensionais**
- **Previsão de séries temporais não lineares**
- **Classificação de Padrões multidimensionais sem separabilidade linear**

- **Note que o aprendizado por exemplos do MLP permite que ele realize as funções acima sem a necessidade de um modelo matemático conhecido / confiável**

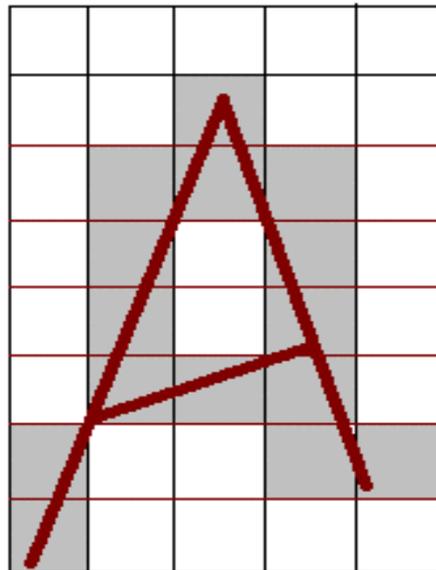
Redes neurais com realimentação

- **Conceito:** o estado presente de cada nó afeta os estados futuros
- **O sistema é dinâmico**
- **Uma arquitetura realimentada particular:** a rede de Hopfield
- **Pesos simétricos / nós digitais**



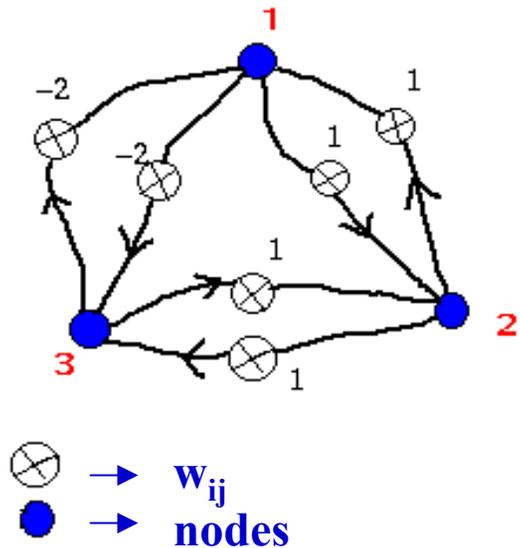
Aplicação de redes neurais de Hopfield

- Armazenamento de padrões (“imagens”) de interesse e recuperação dos mesmos a partir de versões distorcidas ou parciais – Memórias Autoassociativas / Endereçadas por Conteúdo



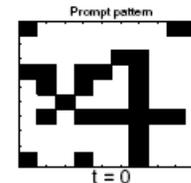
0	0	0	0	0
0	0	1	0	0
0	1	1	1	0
0	1	0	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	0	0	0

Memórias associativas de Hopfield ...

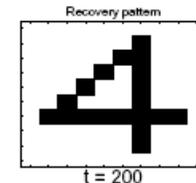


$$\mathbf{W} = \left(\sum_{\mu=1}^M \xi_{\mu} \xi_{\mu}^T \right) - M \mathbf{I D}$$

PROMPTING
PATTERN



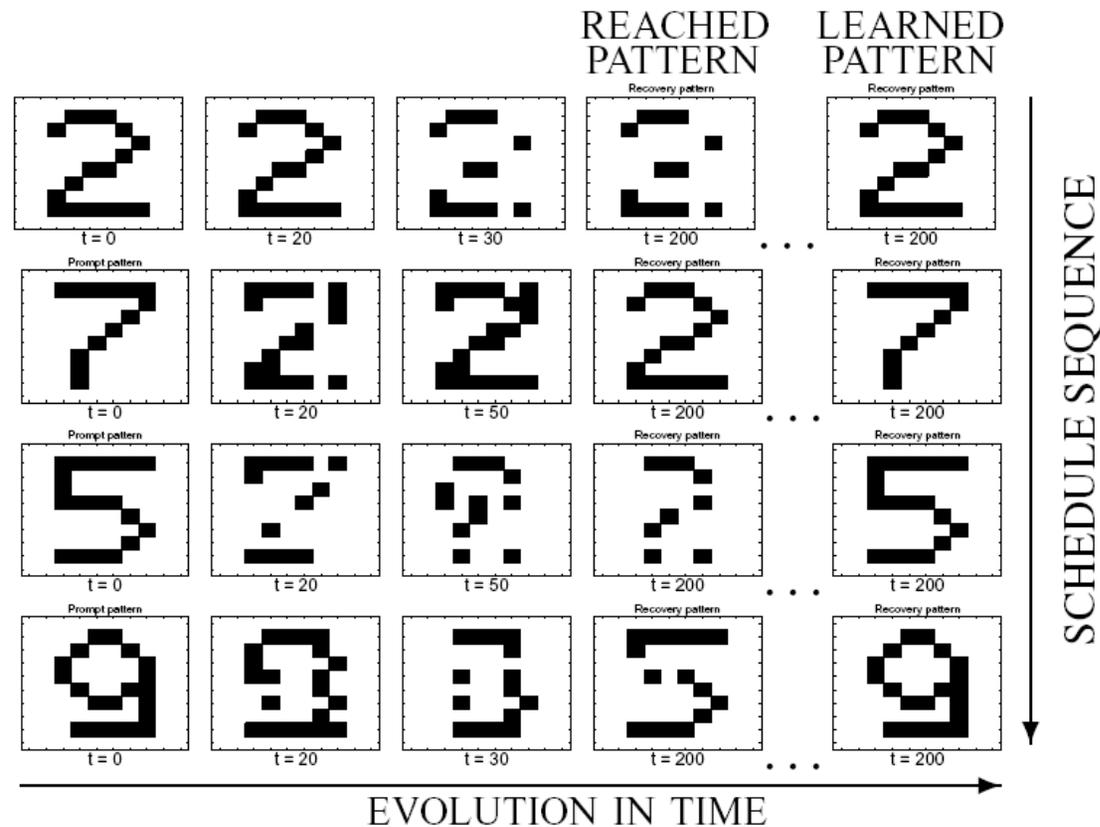
REACHED
PATTERN



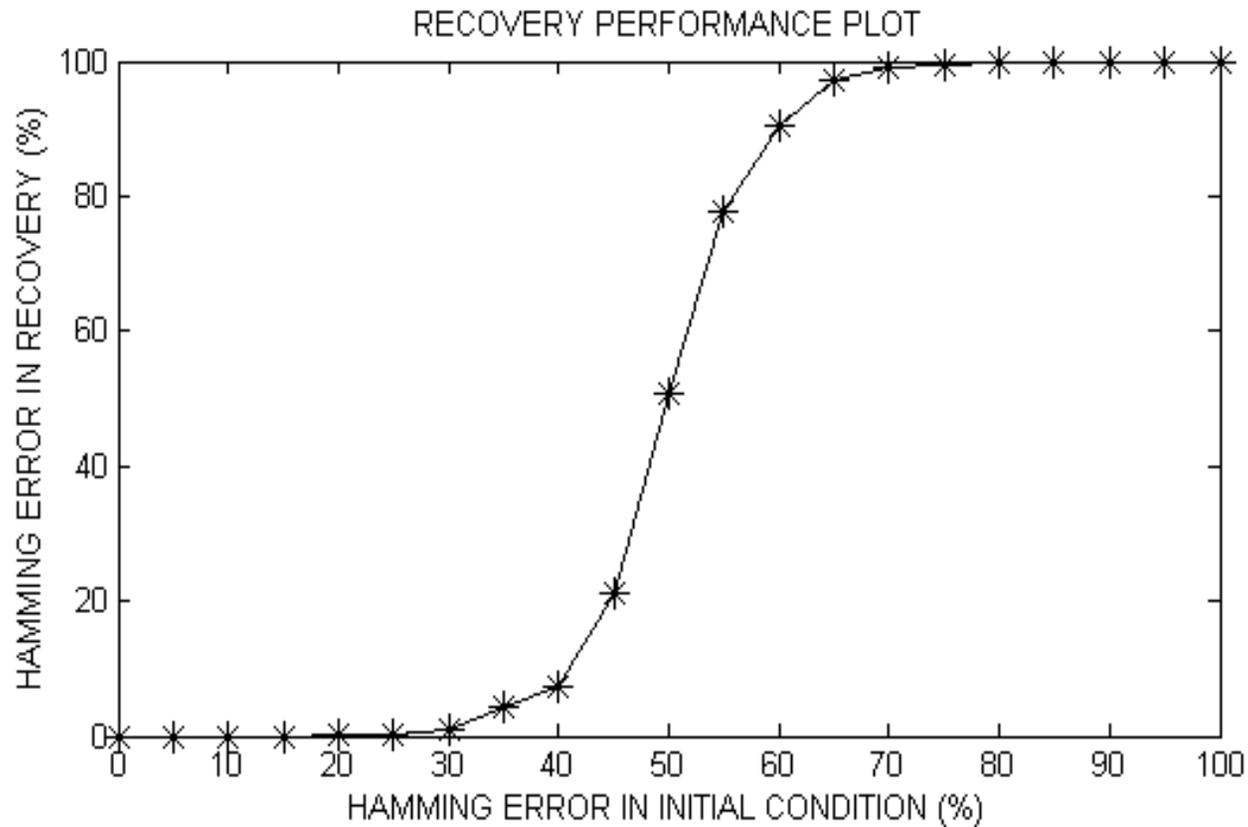
12% noise

Aplicações prototípicas de redes associativas operando em strings binários: recuperação de padrões a partir de versões distorcidas ou parciais dos mesmos

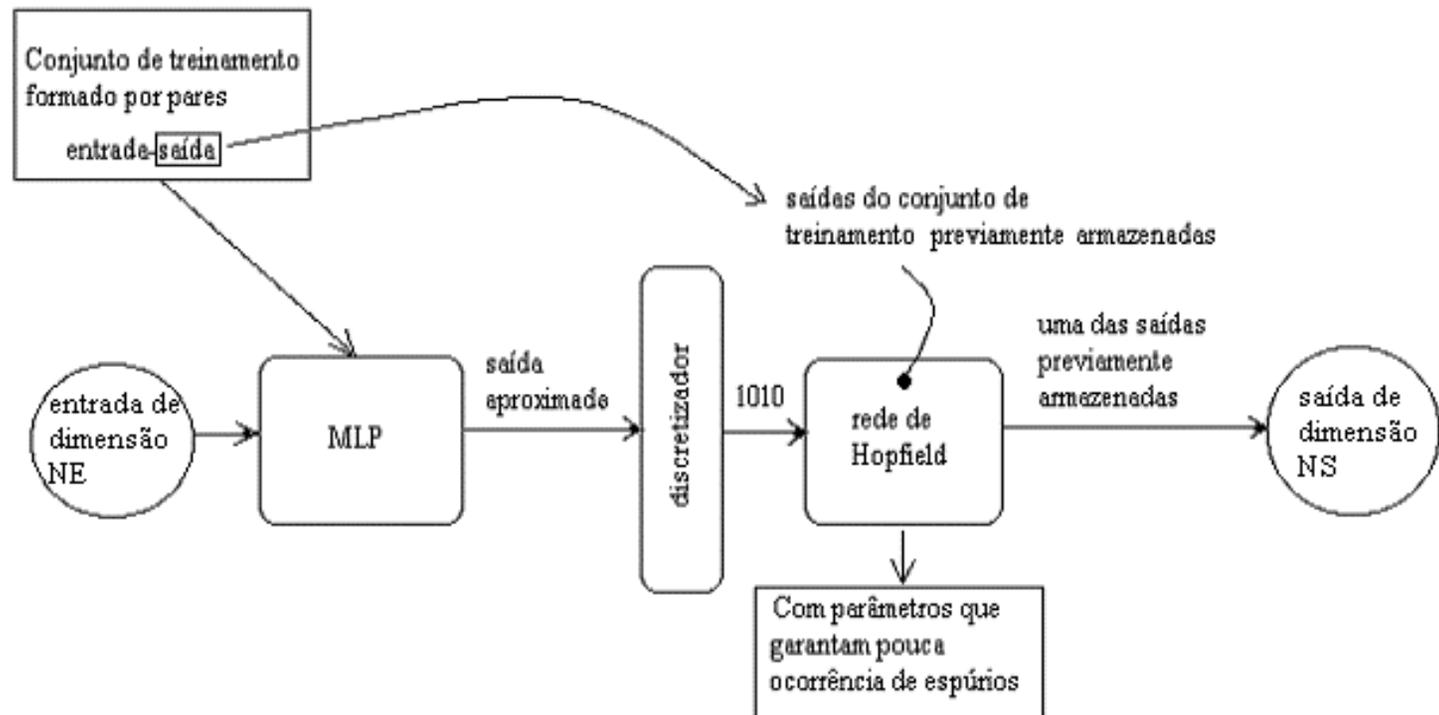
Detecção de novidade e aprendizado incremental em redes de Hopfield



Recuperação de bits típica em memórias autoassociativas de Hopfield



Redes Neurais Híbridas MLP+H

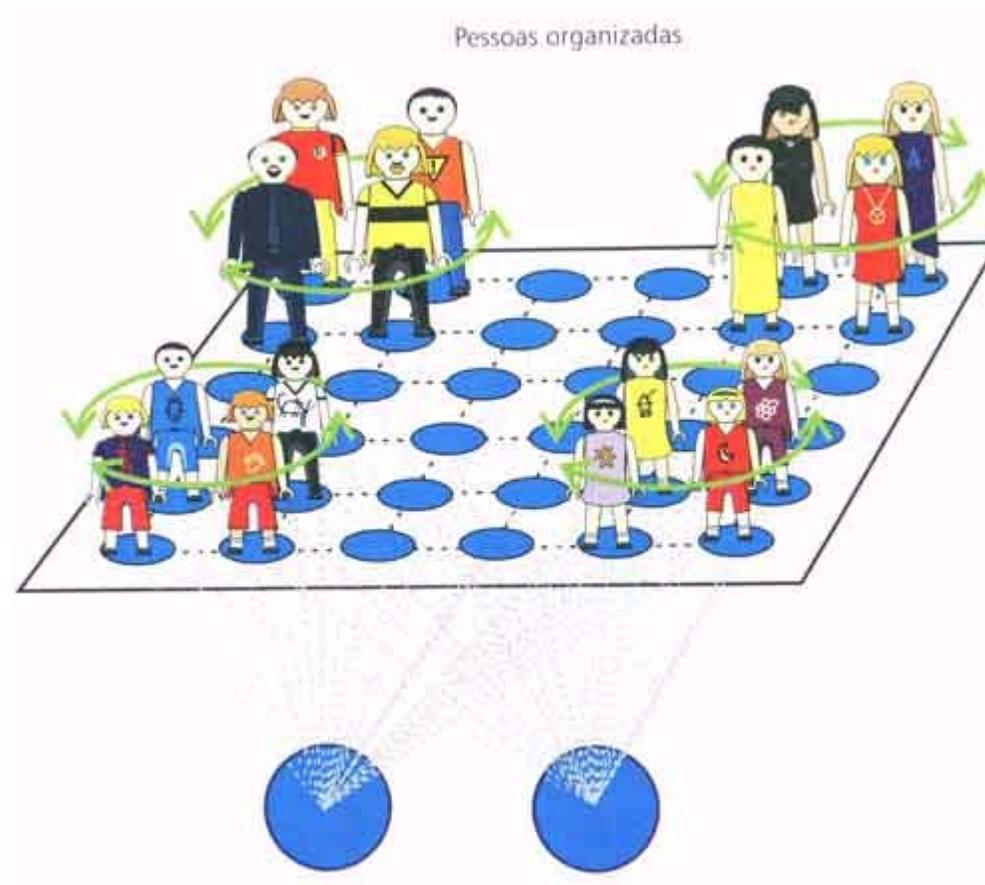


- Trabalho com Clayton Silva Oliveira

Arquiteturas de Kohonen (SOM) para clustering



Dados são agrupados em torno de protótipos organizados em espaço bidimensional



Redes SOM de Kohonen e o Aprendizado Não-Supervisionado

O aprendizado auto-supervisionado não requer saídas desejadas e por isso é conhecido pelo fato de não precisar de “professores” para o seu treinamento.

Para o treinamento da rede, são usados apenas os valores de entrada. A rede trabalha essas entradas e se organiza de modo que acabe classificando-as, usando para isso, os seus próprios critérios.

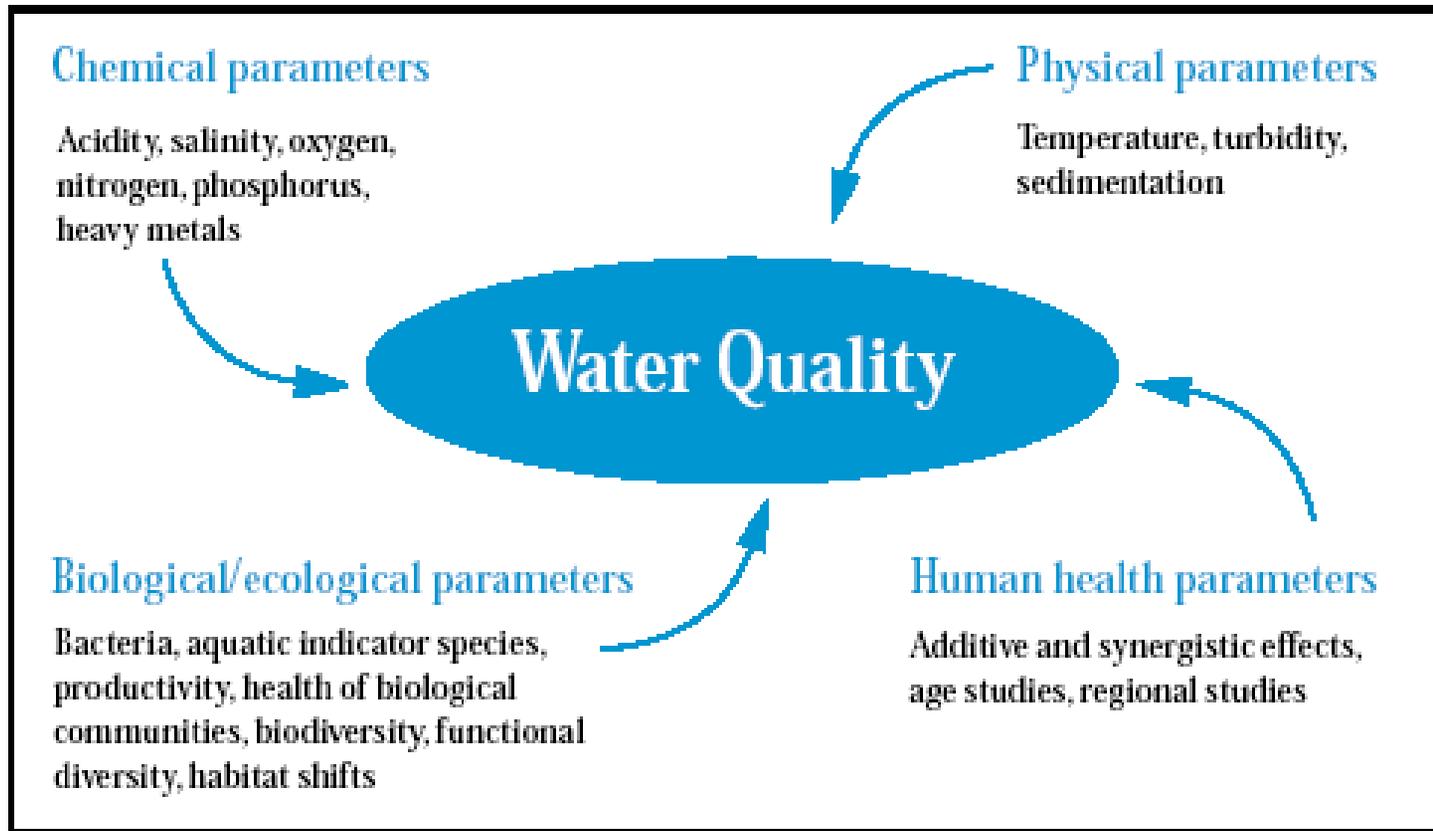
Algumas aplicações de mapas auto organizáveis de Kohonen (SOM)

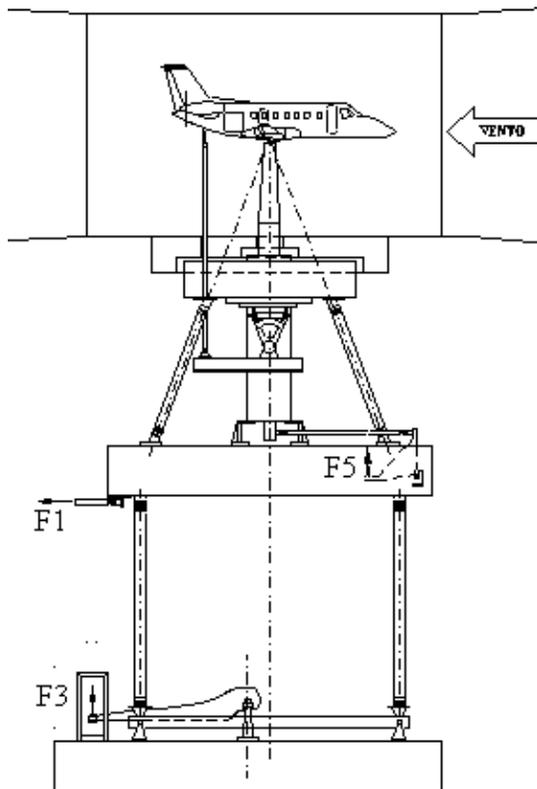
- **Divisão de universos de dados multidimensionais em sub-grupos de elementos similares**
(Congresso ABAR 2003 – segmentação de empresas do setor elétrico – trabalho de Virginia Parente et al.)
- **Data mining**
- **Pré processamento de dados multidimensionais para posterior classificação, de padrões dentro de sub universos mais específicos**

Projeto SIAM – CNPq (Análise Multissensorial)

- **Demonstradores de Sistemas Aplicáveis nas áreas ambiental e biomédica / análise multissensorial:**
 1. **Microssistemas e sensores com algum processamento local de dados +**
 2. **Rede de transmissão de dados +**
 3. **Central de processamento e armazenamento de informação**
- **Pesquisadores de três diferentes áreas do IPT e EPUSP**
- **Processamento de sinais / Reconhecimento de padrões / Inteligência computacional ... são particularmente importantes na implementação das componentes 1 e 3 acima**

Qualidade de águas / análise multissensores





- Aplicação das MLP's nas curvas de calibração da Balança Externa do túnel de vento TA-2.
- Balança Externa => Instrumento de medição de forças e momentos aerodinâmicos.

Sendo os símbolos:

w = vetor pesos sinápticos J = matriz Jacobiana

μ = taxa de aprendizado e = vetor erros na saída

u_j^2 = matriz de covariâncias I = matriz identidade

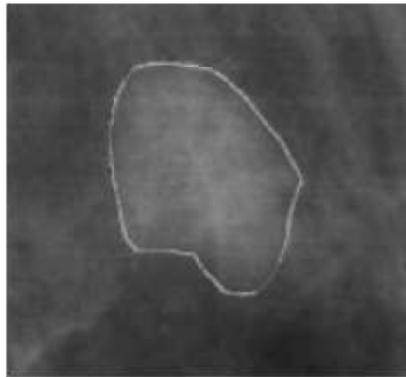
Ajuste clássico dos pesos sinápticos em MLP's:

$$w(n+1) = w(n) - (J^T(n) J(n) + \mu I)^{-1} J^T(n) e(n)$$

Proposta na dissertação de mestrado => Algoritmo de aprendizado para MLP's, levando em consideração as **incertezas** associadas às medições :

$$w(n+1) = w(n) - (J^T(n) (u_j^2)^{-1} J(n) + \mu I)^{-1} J^T(n) (u_j^2)^{-1} e(n)$$

Técnicas de Processamento de Imagens para extração de atributos de exames mamográficos



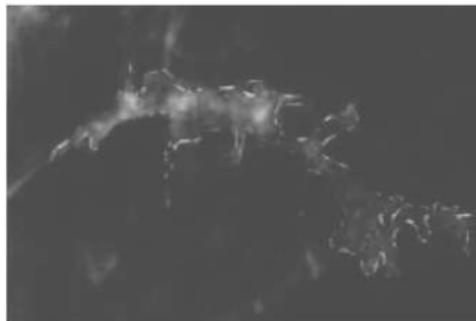
a)

nódulo benigno
com contorno
circunscrito



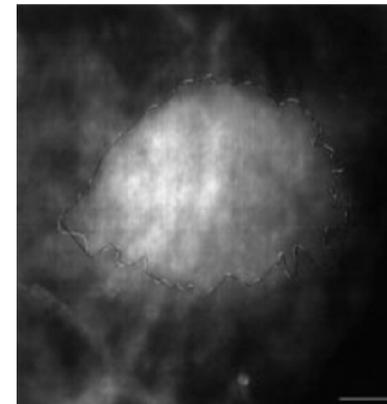
c)

nódulo benigno
com contorno
espiculado.



b)

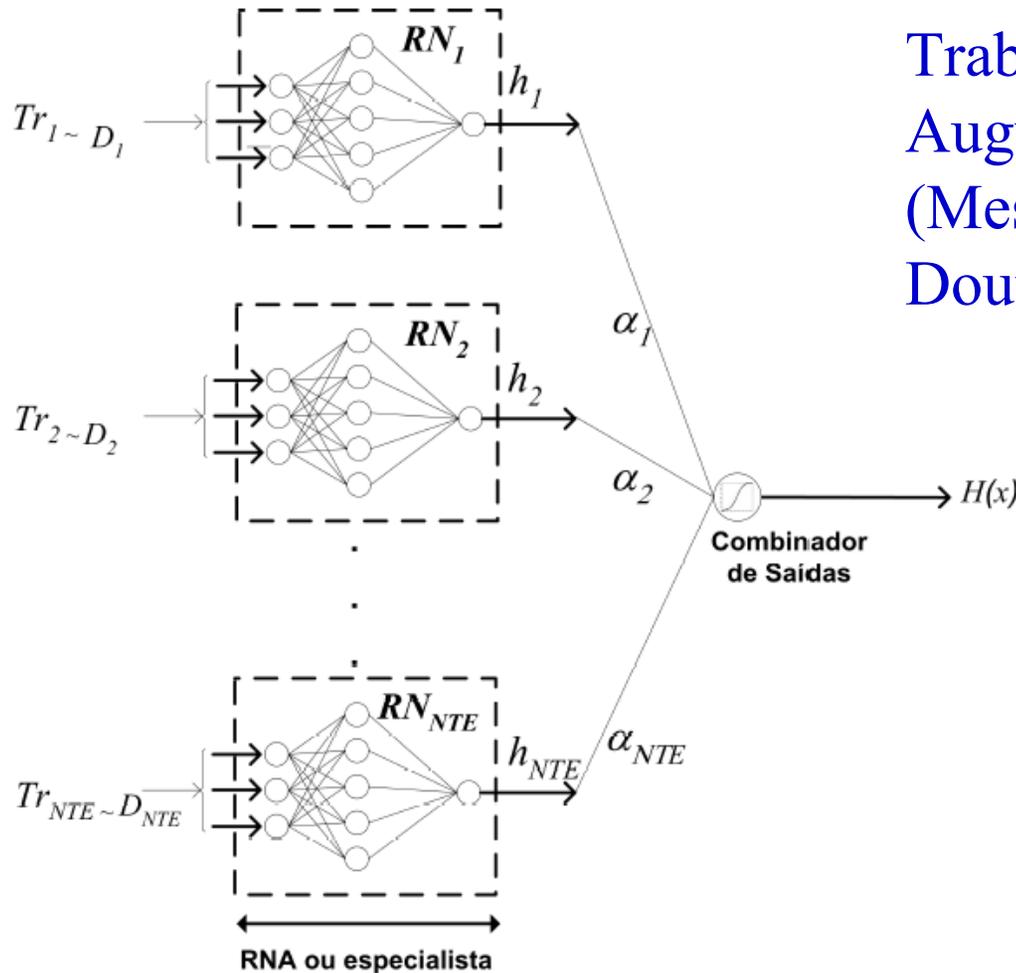
nódulo maligno
com contorno
espiculado.



d)

nódulo maligno
com contorno
circunscrito.

Comitê de Redes Neurais p/ classificação de detalhes do mamograma (ROI)



Trabalho com Leandro Augusto da Silva (Mestre em E.E., e Doutorando em E.E.)

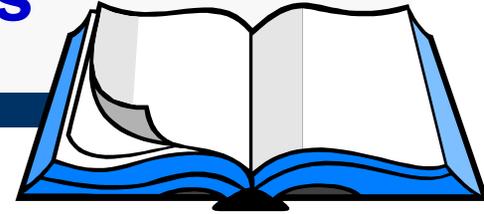
-1 → benigno
1 → maligno

$$H(x) = \text{sign} \left[\sum_{j=1}^t \alpha_j h_j(x) \right]$$

Destacando Trabalhos em Neurocomputação

- **Estudo de novos modelos de computação neural envolvendo fenômenos de bifurcação e dinâmica caótica**
- **Desenvolvimento de novas arquiteturas neurais a partir da fusão de mais de uma arquitetura clássica**
- **Clustering de dados multidimensionais**
- **Emprego de redes neurais em reconhecimento e classificação de padrões**
- **Fusão de informações em sistemas multissensores**
- **Data mining em bases de dados e na WWW**
- **Aproximação de funções e modelagem de histerese**

Alguma Bibliografia Básica + Links



- Kovács, Zsolt L. - *Redes Neurais Artificiais - Fundamentos e Aplicações*, Collegium Cognito, 1996
- Lippmann, Richard P. - *An Introduction to Computing with Neural Nets*, IEEE-Magazine April, 1987

- Referências de diversos níveis de profundidade através do site ww.lsi.usp.br/~emilio
- Simon Haykin – *Neural Networks: A Comprehensive Foundation*, 1999 – também traduzido

Minhas coordenadas

Prof. Emilio Del Moral Hernandez

**Escola Politécnica da Universidade de São Paulo
Departamento de Eng. De Sistemas Eletrônicos**



emilio_del_moral@ieee.org

www.lsi.usp.br/~emilio