UNIVERSIDADE FEDERAL DE JUIZ DE FORA

Programa de Pós-Graduação em Engenharia Elétrica
Mestrado - Doutorado (Conceito 4 CAPES)

Técnicas Inteligentes

Redes Neurais Artificiais
Aula 02

Leonardo Willer de Oliveira

Juiz de Fora, 15 de Maio de 2017

Referências

- Haykin, S., *Neural Networks, A Comprehensive Foundation*, Prentice Hall, 1999.
- Haykin, S., *Redes Neurais, Teoria e Prática*, Bookman, 2001.
- Luiz P. Calôba, *Uma pequena introdução às Redes Neurais Artificiais e Aplicações*, Curso compacto – Informativo, Gramado, 2005.

Camada de Kohonen

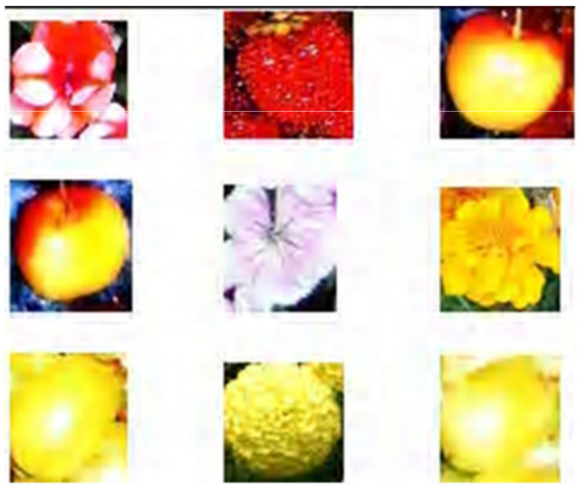
- Rede ART, Rede *Counterpropagation*
  - ✓ Aprendizado cego, não supervisionado, auto-organizáveis
  - ✓ Plasticidade
  - ✓ Classificadores por similaridade
  - ✓ Aplicações *on-line*
  - ✓ Biologicamente plausível (córtex auditivo)

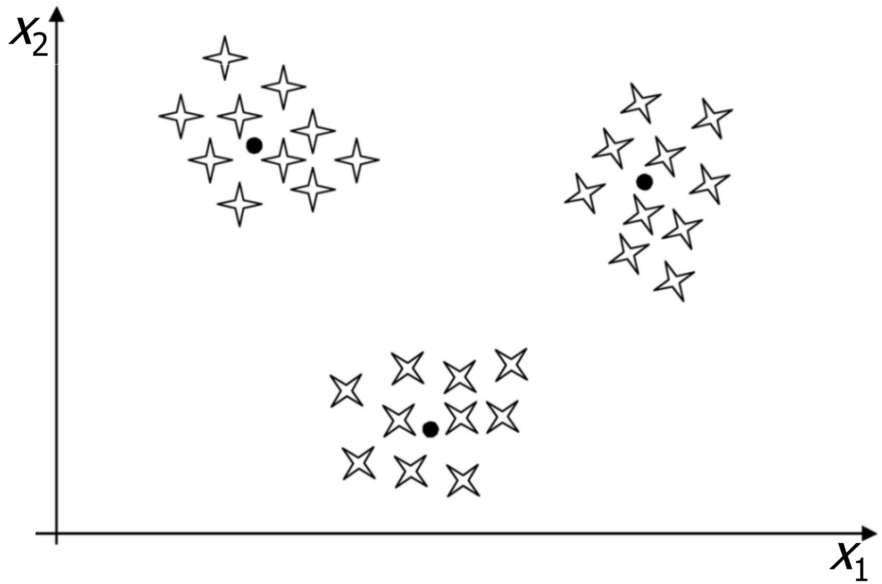
Camada de Kohonen

- Aplicações
  - ✓ Reconhecimento de padrões
  - ✓ Estatística de sinais
  - ✓ Memórias associativas
  - ✓ Filtragem não-linear
  - ✓ Compressão de dados
  - ✓ Mapas auto-organizáveis

Camada de Kohonen

- Classificação por similaridade

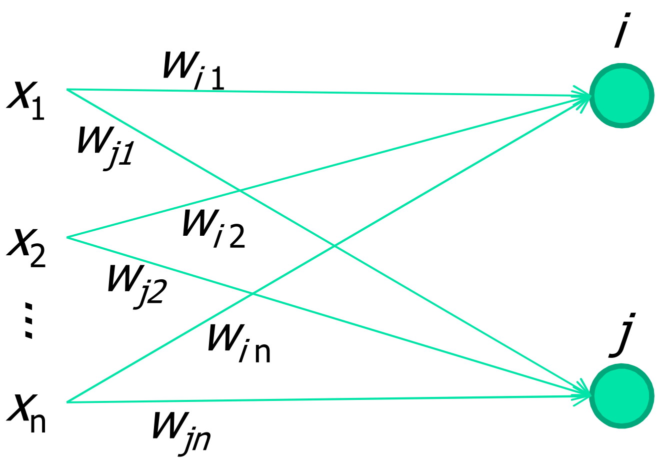




$$\underline{w}_i = \underset{\underline{x}j \in C_i}{E}\left[\underline{xj}\right] = \frac{1}{N_i} \cdot \sum_{\underline{x}j \in C_i} \underline{xj}$$

Camada de Kohonen

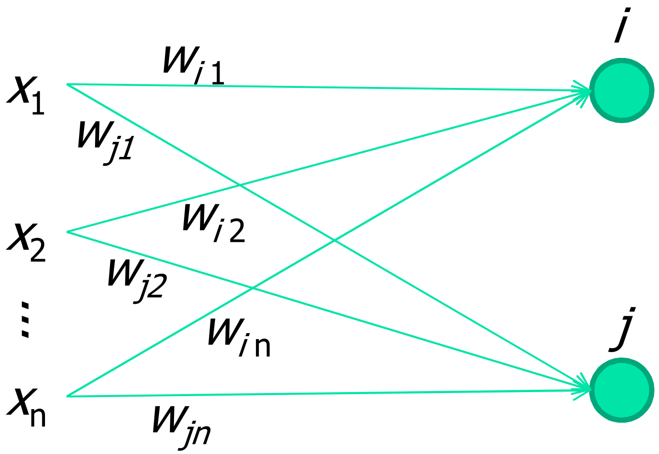
- Critério: Padrão mais similar à entrada



$$\text{Se } \left|\underline{x}-\underline{w}_i\right|^2 < \left|\underline{x}-\underline{w}_j\right|^2 \quad \forall j \neq i \quad \Longrightarrow \quad \underline{x} \in C_i$$

Camada de Kohonen

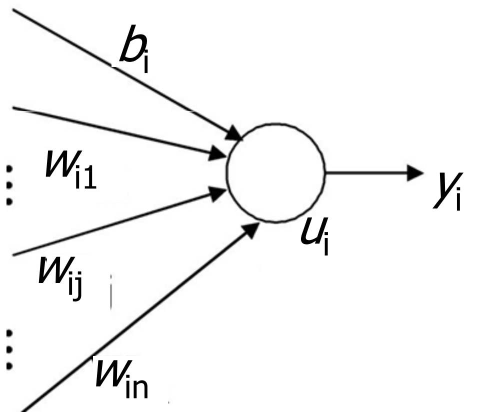
- Critério: Similaridade mínima

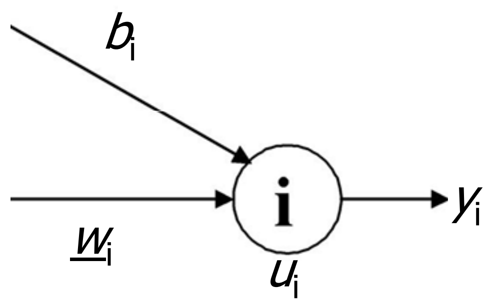


Se $\left|\underline{x} - \underline{w}_i\right|^2 < r_0^2 \Rightarrow \underline{x} \in C_i$ $\quad$ r_0 – raio de similaridade

Camada de Kohonen

- Neurônio medidor de similaridade





Considerando $|\underline{x}| = 1$

$|\underline{w}_i| = 1$

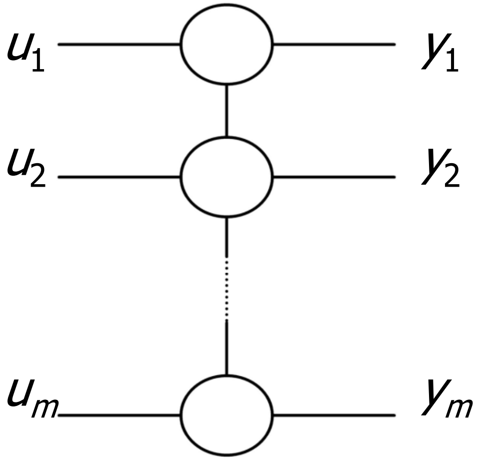
$b_i = 0$

$$u_i = 1 - \frac{d_i^2}{2} \quad \text{sendo} \quad d_i = |x - w_i|$$

u_i : similaridade entre $\underline{x}$ e $\underline{w}_i$ $u_i \in [-1,+1]$

Camada de Kohonen

- *Winner-takes-all*



$$y_i = 1 \text{ se } u_i > u_j \forall j \neq i$$

$$y_i = 0 \text{ caso contrário}$$

$$u_i = \underline{x}\,\underline{w}_i = 1 - \frac{d_i^2}{2}$$

$$d_i = \left| x - w_i \right|$$

Classe C_i

⇓

Padrão w_i

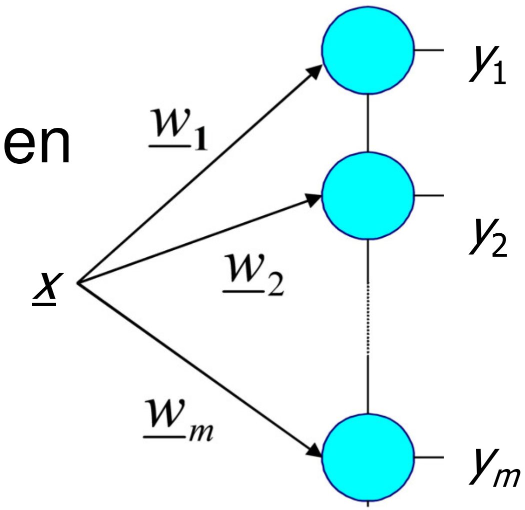
⇓

Indicador y_i

⇓

$\boxed{\text{Se } y_i = 1 \;\; \underline{x} \in C_i}$

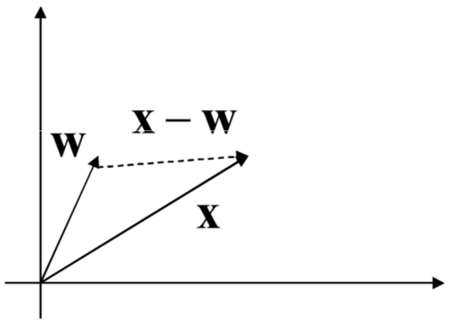
Camada de Kohonen



Camada de Kohonen

- Treinamento Supervisionado e Não-Supervisionado

n : iteração n



Se $\underline{x}(n) \in C_i$ ou $y_i = 1$

$$\underline{w_i}(n+1) = \underline{w_i}(n) + \alpha\left[\underline{x}(n) - \underline{w_i}(n)\right]$$

$$\underline{w_j}(n+1) = \underline{w_j}(n) \ \forall j \neq i$$

Convergência

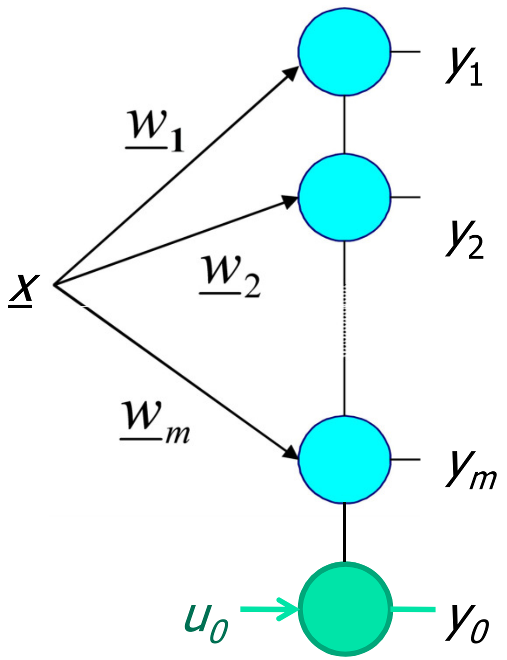
$$E\left[\Delta \underline{w}\right] = \underline{0}$$

Observação: α pode ser atualizado durante o processo iterativo

Exemplo

$$\alpha_n = 0{,}9 \cdot (1 - n/1000)$$

Camada de Kohonen Aumentada



Se $y_i = 1$

$\underline{x} \in C_i$ pelos critérios "padrão mais similar" e "similaridade mínima" (r_0)

Se $y_0 = 1$

$\underline{x}$ não atende a similaridade mínima para nenhuma classe

$$u_i = 1 - \frac{d_i^2}{2} \quad \text{sendo} \quad d_i = |x - w_i|$$

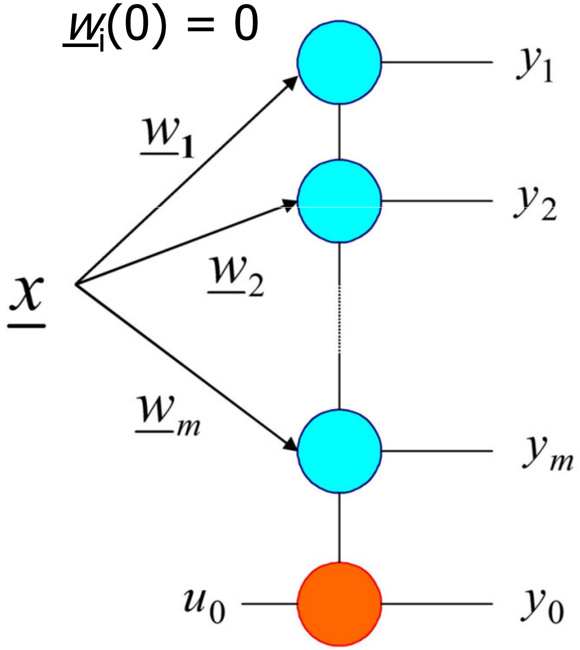
$$u_0 = 1 - \frac{r_0^2}{2}$$

ART

- Teoria de Ressonância Adaptativa (*Adaptive Ressonance Theory*)
  - ✓ Classificador
  - ✓ Aprendizado não-supervisionado
  - ✓ Semelhanças com o modelo de Kohonen no armazenamento de padrões

ART

$\underline{w}_i(0) = 0$



$$u_0 = 1 - \frac{r_0^2}{2}$$

- Similaridade mínima exigida

Se $y_0 = 1$ $\quad \underline{x} \notin C_i \quad \forall i$

Ativar um neurônio *i* desativado

$$\underline{w}_i(n+1) = \underline{x}(n)$$

$$\underline{w}_j(n+1) = \underline{w}_j(n) \quad \forall j \neq i$$

Se $y_i = 1$

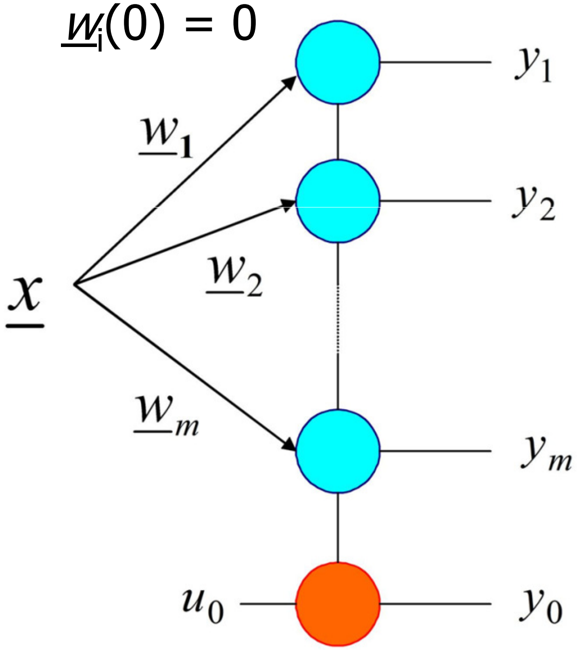
Treinar o neurônio *i* vencedor

$$\underline{w}_i(n+1) = \underline{w}_i(n) + \alpha\left[\underline{x}(n) - \underline{w}_i(n)\right]$$

$$\underline{w}_j(n+1) = \underline{w}_j(n) \quad \forall j \neq i$$

ART

$\underline{w}_i(0) = 0$



$$u_0 = 1 - \frac{r_0^2}{2}$$

- <u>Esquecimento</u>

  Se y_i = 0 por N entradas consecutivas:

✓ Guardar informação (lembrança) para o futuro

✓ Desativar neurônio $\underline{w}_i = 0$ $\underline{m}_k = \underline{w}_i$

- <u>Lembrança</u>

  Se y_j = 0:

✓ Ativar neurônio

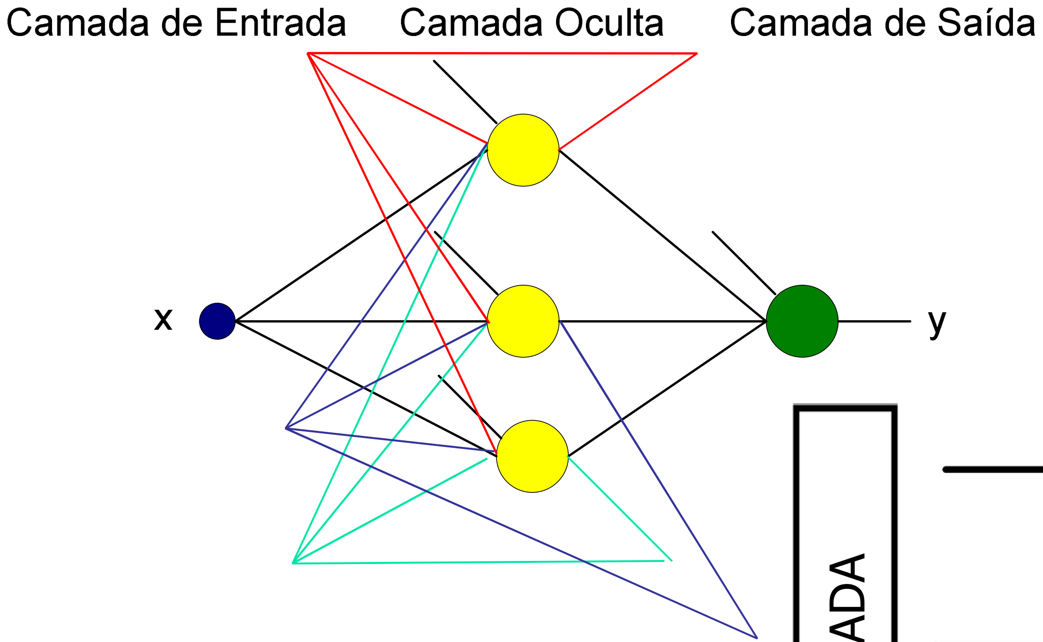
✓ Existe lembrança? $\underline{x}^t \underline{m}_k > u_0 \Rightarrow \underline{w}_j = m_k$

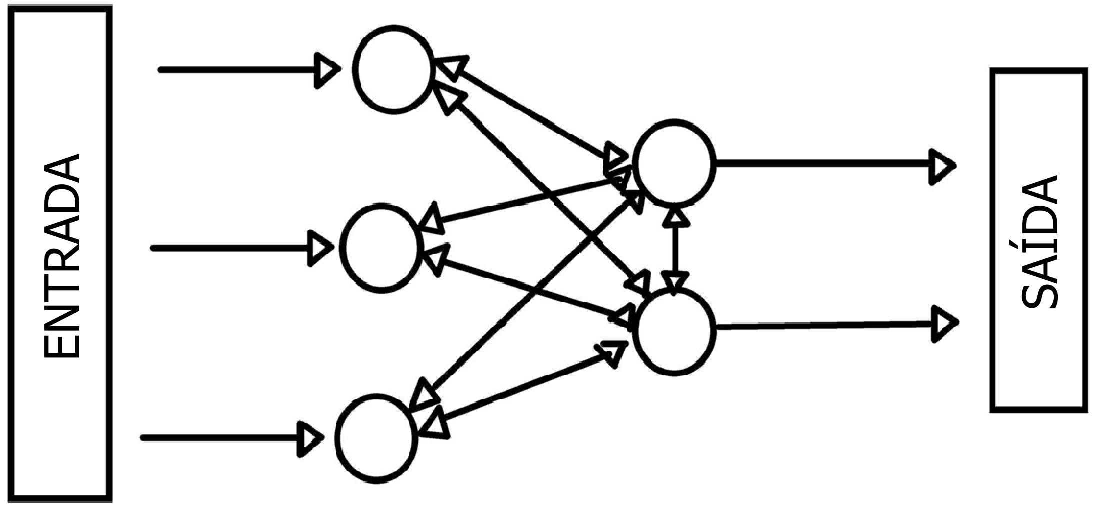
✓ Não há lembrança? $\underline{w}_j = \underline{x}$

Redes de Hopfield

- Redes realimentadas
  - ✓ Problemas de otimização combinatória
  - ✓ Binárias
  - ✓ Supervisionadas
  - ✓ Baseadas no conceito de energia da rede
  - ✓ Objetivo: minimizar a energia ou a oscilação entre estados na saída

Redes Retroalimentadas





Redes de Hopfield

- O estado y_i de um neurônio *i* no instante $t + 1$ é obtido pela seguinte função da soma ponderada das entradas:

$$y_i(t+1) = sgn\left(\sum_{j=1}^{n} w_{ij} y_j(t) - \theta_i\right)$$

n : número de associações

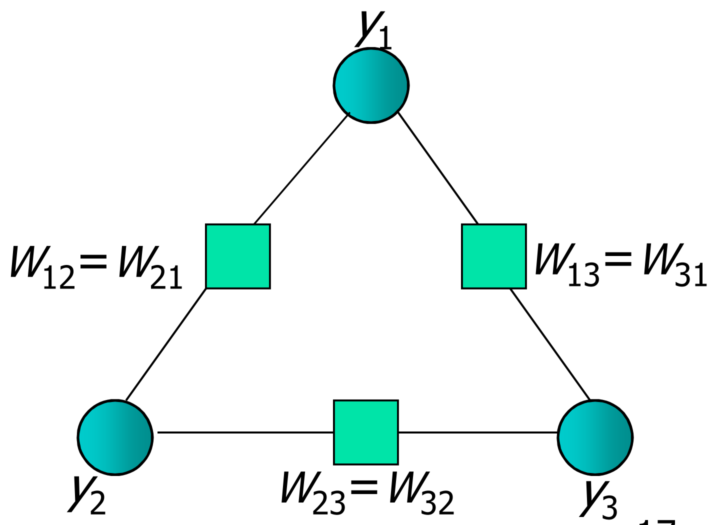
θ : limiar

A função *sgn* pode ser definida por:

$$sgn(x) = \begin{cases} +1 & se \quad x \geq 0 \\ -1 & se \quad x < 0 \end{cases}$$

$$\Downarrow$$

$$y \in \{-1, +1\}$$



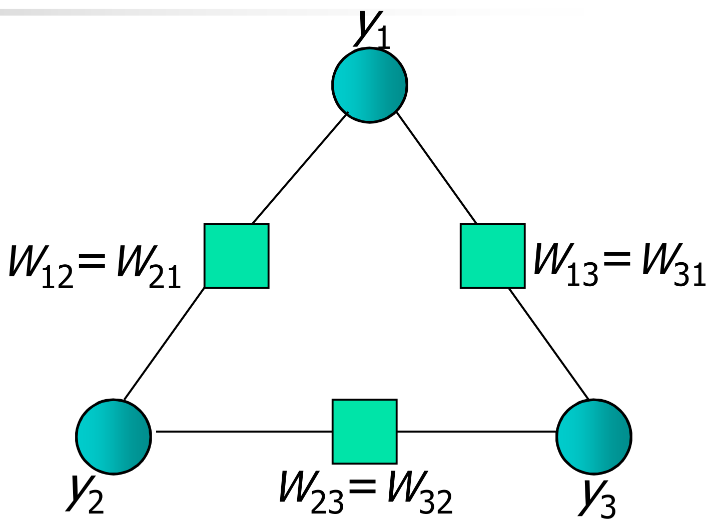
Redes de Hopfield

- Pesos

$$w_{ij} \quad \alpha \quad y_i y_j$$

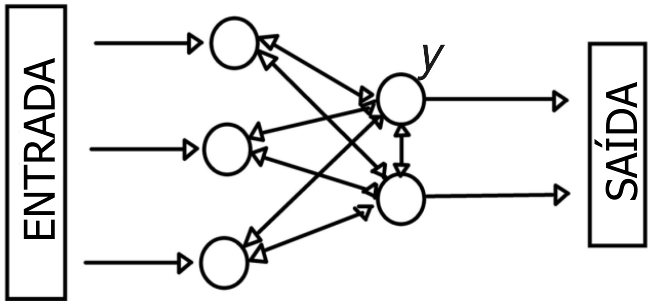
$$w_{ij} = \frac{1}{N} y_i y_j$$

N : número de neurônios da rede



Condição de estabilidade de um vetor **y** ($\theta = 0$)

$$y_i = sgn\left(\sum_j w_{ji} y_j\right) \quad \forall \quad i$$



Substituindo w_{ij} obtém-se

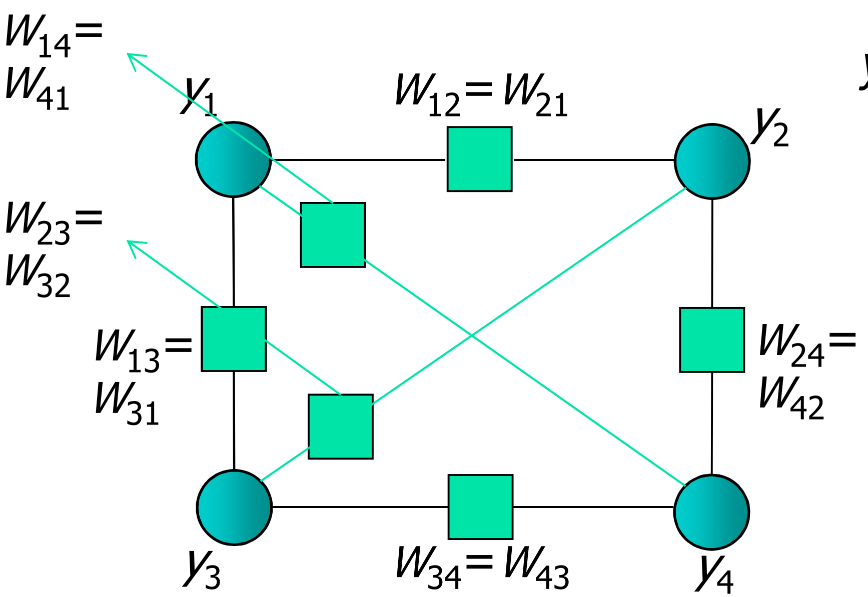
$$y_i = sgn\left(\sum_j w_{ji} y_j\right) = sgn\left(\frac{1}{N}\sum_j y_j y_i y_j\right) \quad \forall \quad i$$

Redes de Hopfield

- Se um y_j muda, mas a maioria se mantém, o somatório de

$$y_i = sgn\left(\sum_j w_{ji} y_j\right) = sgn\left(\frac{1}{N}\sum_j y_j y_i y_j\right) \quad \forall \quad i$$

tende a manter o sinal de y_i.



$$y_1 = sgn\left[\frac{1}{4}\cdot\left(w_{12}\cdot y_2 + w_{13}\cdot y_3 + w_{14}\cdot y_4\right)\right]$$

Se y_4 muda, y_1 mantém

Recuperação

Distância de Hamming

$$\sum_i \left[y_i \cdot \left(1 - y'_i\right) + \left(1 - y_i\right)\cdot y'_i \right]$$

Redes de Hopfield

- Exemplo: Para a matriz de pesos

$$\mathbf{W} = \begin{pmatrix} 0 & -0{,}333 & 0{,}333 \\ -0{,}333 & 0 & -0{,}333 \\ 0{,}333 & -0{,}333 & 0 \end{pmatrix}$$

Os vetores (-1 1 -1)^T e (1 -1 1)^T são estáveis

$$w_{ij} = \begin{cases} \dfrac{1}{N} \cdot \displaystyle\sum_{s=1}^{p} y_i^s \cdot y_j^s & \text{se } i \neq j \\ 0 & \text{se } i = j \end{cases}$$

Regra de Hebb

p : número de padrões

Convergência processo iterativo ⟹ $y_i(t+1) = y_i(t)$

Redes de Hopfield

- Algoritmo de Treinamento: sequencial

Para i = 1 até N

$y_i = y_i^0$

Fim_Para

Repita

Para i = 1 até N

$y_i^- = y_i$

$y_i = \text{sgn}(\Sigma wij^* y_j,\ j \neq i)$

Fim_Para

Até $y_i = y_i^-$ para todo $i \in N$

Fim

Redes de Hopfield

- A energia associada com o estado do sistema pode ser dada por

$$E = -\frac{1}{2}\left(\sum_i \sum_j w_{ij} \cdot y_i \cdot y_j\right)_{i \neq j}$$

Para uma rede de dimensão *N*, considera-se que o nó '*m*' mude o seu estado do valor y_m para y_m'

A variação de energia resultante pode ser formulada como

$$\Delta E = -\frac{1}{2}(\; w_{12}y_1y_2 + \ldots + w_{1N}y_1y_N + w_{21}y_2y_1 + \ldots + w_{2N}y_2y_N + \ldots$$

$$+w_{m1}y'_m y_1 + \ldots + w_{mN}y'_m y_N + \ldots + w_{N1}y_N y_1 + \ldots + w_{N(N-1)}y_N y_{N-1} \;) -$$

$$[-\frac{1}{2}(w_{12}y_1y_2 + \ldots + w_{1N}y_1y_N + w_{21}y_2y_1 + \ldots + w_{2k}y_2y_N + \ldots$$

$$+w_{m1}y_m y_1 + \ldots + w_{mN}y_m y_N + \ldots + w_{N1}y_N y_1 + \ldots + w_{N(N-1)}y_N y_{N-1}] \;)$$

Redes de Hopfield

- Manipulando-se a expressão anterior, obtém-se

$$\Delta E = -\frac{1}{2} \cdot y'_m \cdot \sum_i w_{mi} \cdot y_i + \frac{1}{2} \cdot y_m \cdot \sum_i w_{mi} \cdot y_i$$

Como $y_m = -y'_m$ tem-se: $\Delta E = y_m \cdot \sum_i w_{mi} \cdot y_i$

Sendo $y'_m = sgn\left(\sum_i w_{mi} \cdot y_i\right)$, para $\theta = 0$

conclui-se que $\sum_i w_{mi} \cdot y_i$ tem sinal contrário a y_m

Logo $\Delta E < 0$, ou seja, a função energia diminui quando $y_m \neq y'_m$.