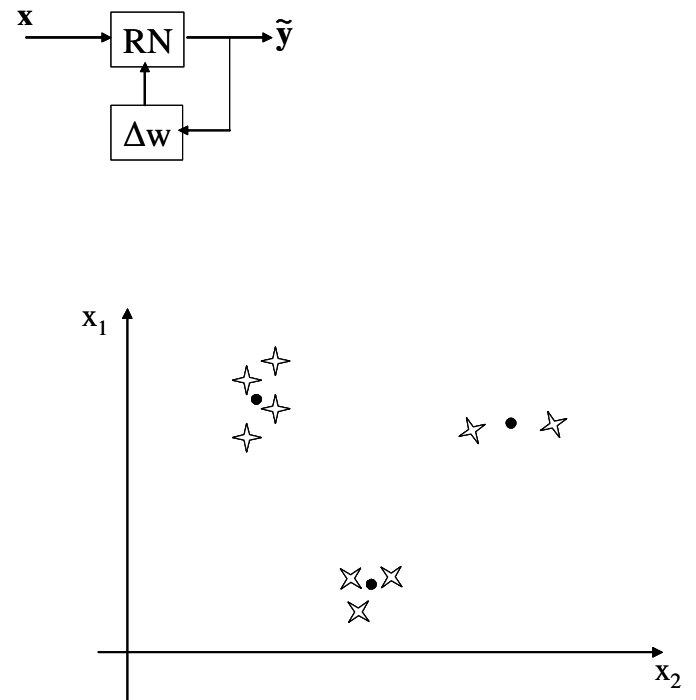
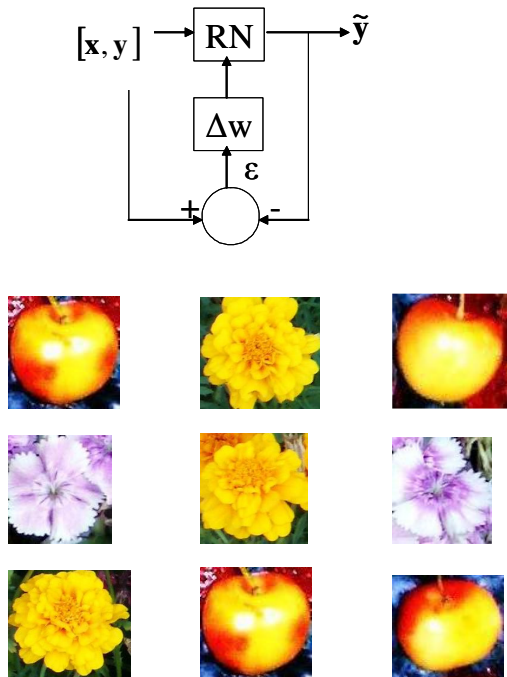


Camada de Kohonen – Treinamento cego

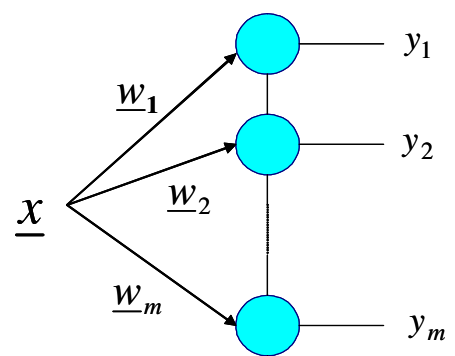
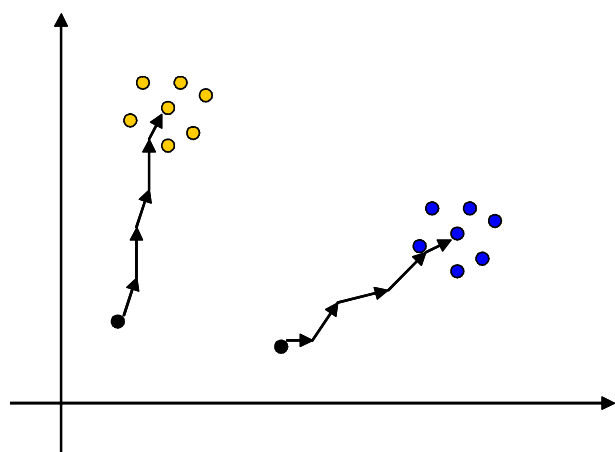
1 - Treinamento supervisionado e não-supervisionado



2 - Camada de Kohonen

Classificação por similaridade - Treinamento não supervisionado

-



Camada de Kohonen

Treinamento não supervisionado

$$\underline{x}(n) \gggg y_i = 1$$

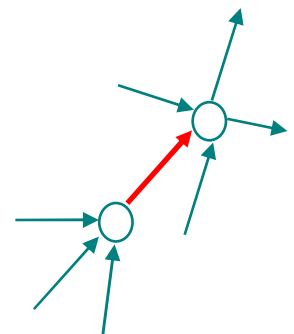
$$\begin{aligned}\underline{w}_i(n+1) &= \underline{w}_i(n) + \alpha [\underline{x}(n) - \underline{w}_i(n)] \\ &= (1 - \alpha) \underline{w}_i(n) + \alpha \underline{x}(n)\end{aligned}$$

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

O treinamento é dito **competitivo**, porque apenas o neurônio vencedor treina.

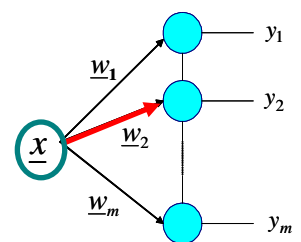
Obs: Este algoritmo pode ser visto como uma aplicação da Regra de Hebb, a mais antiga regra de aprendizado (1949), que pode ser escrita:

“Quando dois neurônios conectados por uma sinapse são ativados simultaneamente (em sincronismo) a sinapse entre eles tende a se fortalecer seletivamente.”



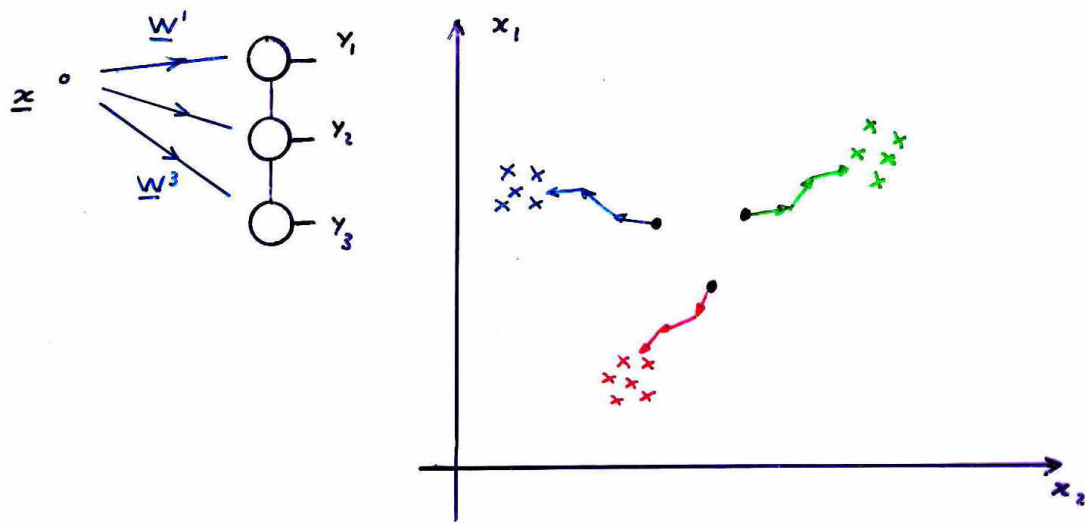
uma segunda parte foi posteriormente acrescentada a esta regra, e será utilizada futuramente para explicar o mecanismo de esquecimento:

“Quando dois neurônios conectados por uma sinapse não são ativados simultaneamente (são assíncronos) a sinapse entre eles tende a se enfraquecer seletivamente ou a ser eliminada.”



em nosso caso a entrada atua como um dos neurônios, e a sinapse entre os dois é o vetor sinapse.

Evolução do Treinamento



Fim do treinamento ?



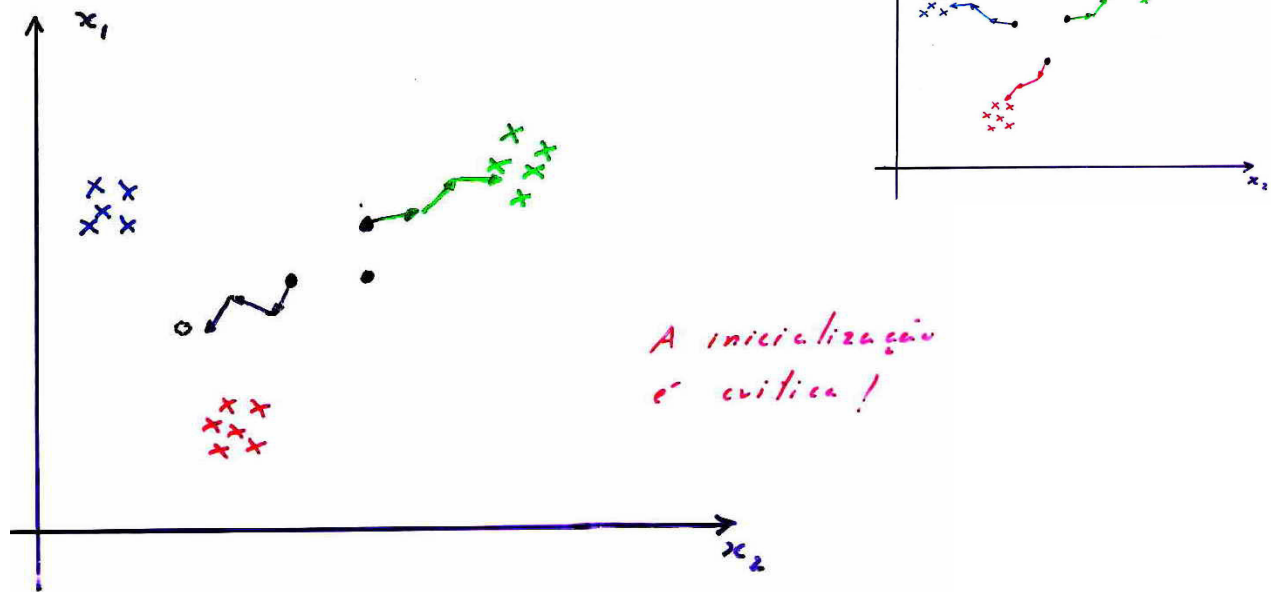
$$E[\Delta w] = \underline{0}$$



Usar um α por vetor sinapse (populações díspares por neurônio)

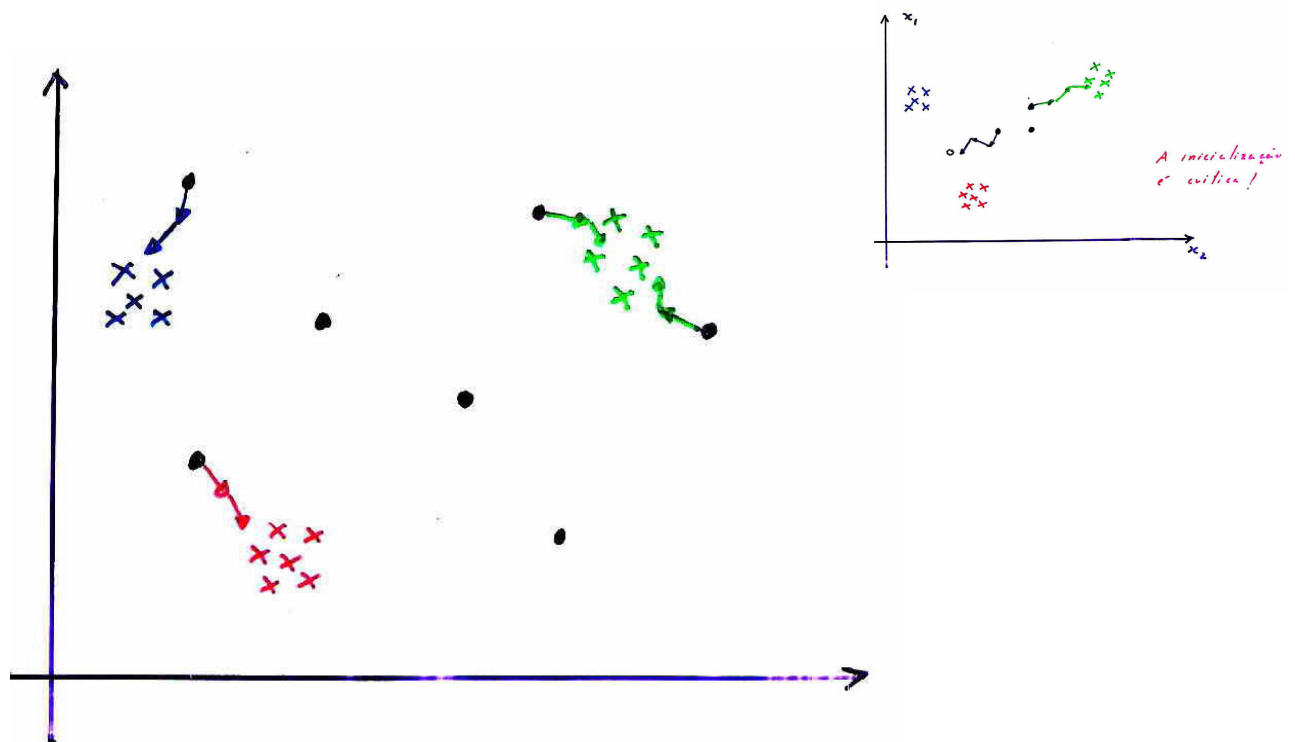
Realizar o treinamento com decaimento do passo por vetor sinapse

3 - Problemas no treinamento:



Aumentar o número de sinapses

→ aumentar soft ou hardware



uma classe é representada por mais de um padrão

3.1 - Inicialização

A inicialização é crítica $\underline{w}_i(0)$

- randômicos

- orientados pela população

primeiras entradas $\underline{w}_i(0) = \underline{x}(i)$

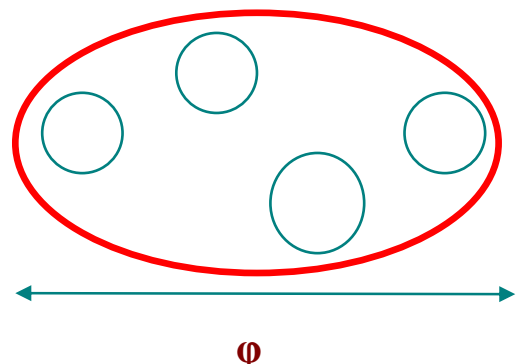
primeiras entradas não muito próximas $\underline{w}_i(0) = \underline{x}(i)$

O que significa “não muito próximas” ?

Domínio de todos os elementos

distância entre entradas

$$d_{ij} = \left| \vec{x}_i - \vec{x}_j \right|$$



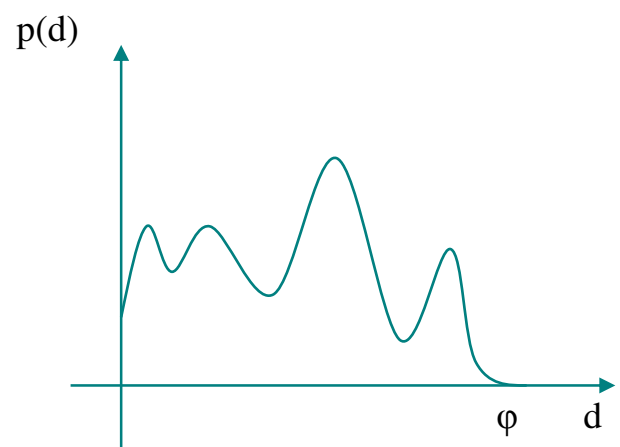
diâmetro da classe única

$$\varphi = \max (d_{ij})$$

N classes

Distância aceitável entre sinapses $\underline{w}$

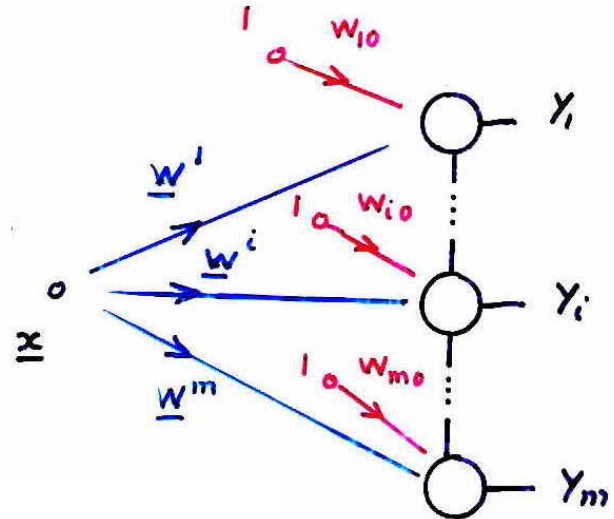
$$d_{\bar{w}} \approx \frac{\varphi}{N}$$



4 - Consciência

O neurônio que treinou muitas vezes abre mão do treinamento para o segundo ganhador.

É adicionado um “handicap”
no(s) neurônio(s) muito
treinados



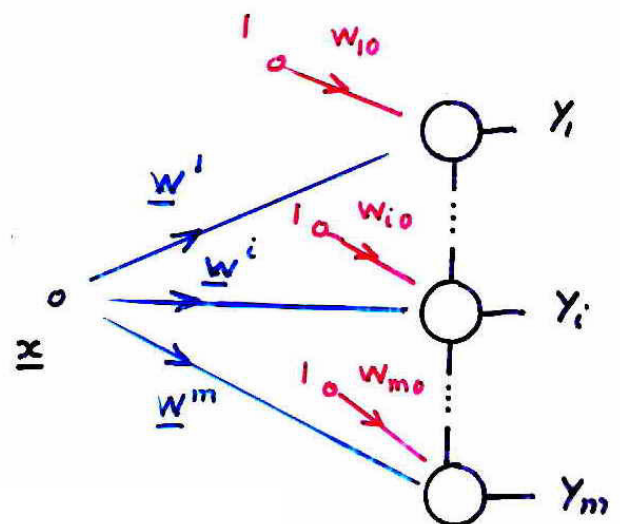
$$u_j = -d_j^2 + w_{j0} \quad w_{j0} < 0$$

$$u_k = -d_k^2 + w_{k0}$$

Para $u_j > u_k$

$$d_j^2 - w_{j0} < d_k^2 - w_{k0}$$

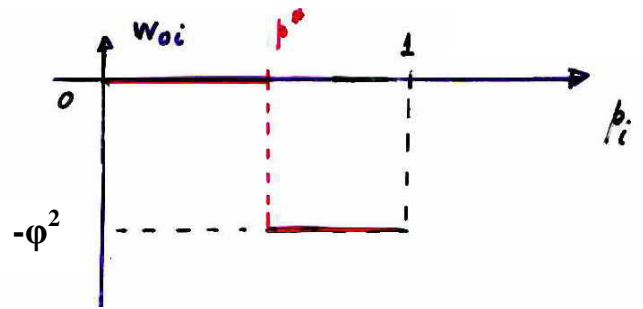
$$d_j^2 < d_k^2 - (w_{k0} - w_{j0})$$



Formas de consciência:

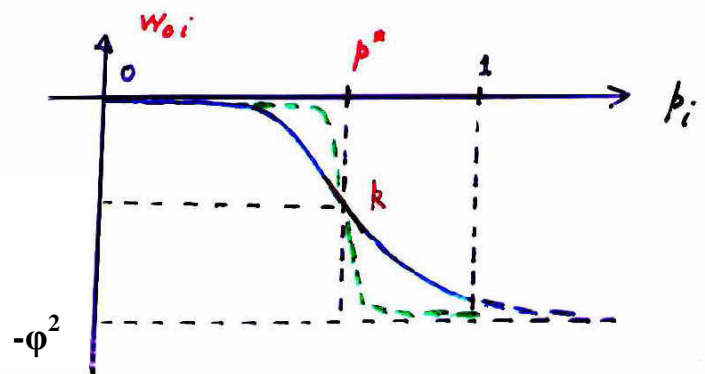
abrupta:

$$p_i \begin{cases} < p^* & w_{i0} = 0 \\ > p^* & w_{i0} = -\varphi^2 \end{cases}$$



suave

$$w_{i0} = -\frac{\varphi^2}{2} \{1 - \tanh[k(p_i - p^*)]\}$$



Obs:

a – Critério para φ : o neurônio i perde sempre

$$\varphi \mid \text{Max}(u_i) < \text{Min}(u_j) \quad \forall i \neq j \quad \forall \vec{x}$$

logo φ pode ser maior ou igual ao diâmetro da classe única que inclui todas as entradas

$$\varphi \geq \text{Max}|\vec{x}_k - \vec{x}_l| \quad \forall k, l$$

b – Critério para p_i^* : se as populações das classes tem a mesma ordem de grandeza

$$p_i^* \approx \frac{1}{\text{número_de_sin_apses}}$$

c - Obtenção de p_i

$$y_i \in \{0,1\}$$

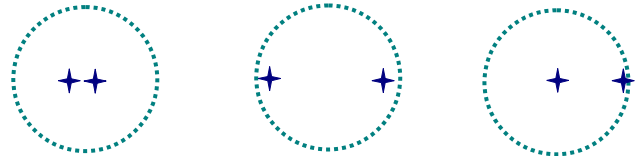
$$p_i(0) = 0$$

$$p_i(n+1) = (1-\alpha)p_i(n) + \alpha y_i(n)$$

$$p_i(n) \cong \frac{\sum_{j=1}^n (1-\alpha)^{n-j} y(j)}{\sum_{j=1}^n (1-\alpha)^{n-j}} \quad N \approx \frac{\alpha}{4}$$

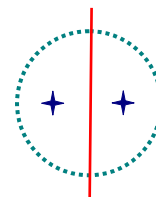
d – Evolução de padrões múltiplos por classe

Posições iniciais



Posição final

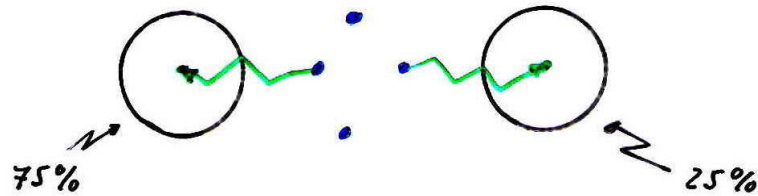
com ou sem consciência



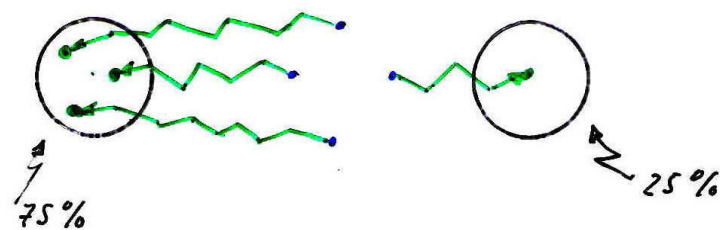
mas a evolução a partir da posição inicial assimétrica é muito mais lenta sem consciência.

5 - Geometria da Classificação:

Sem consciência por localização geométrica



Com consciência por população

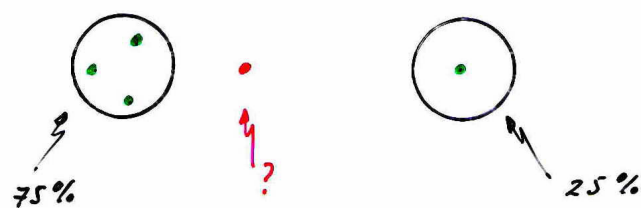


Representação da geometria das classes (sem consciência)

versus

Representação da população (com consciência)

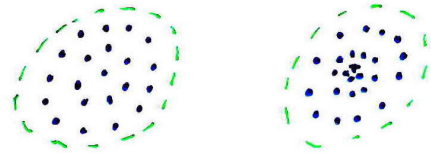
Com poucos neurônios pode ser imprecisa:



Definição de domínios

Caracterização da estatística de $\underline{x}$

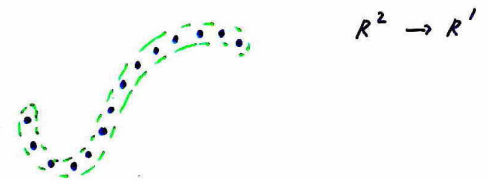
$$p(\underline{x}) = p(\underline{w})$$



Mapas Auto Organizáveis de Kohonen

SOM

Representação em dimensão reduzida



Classificadores



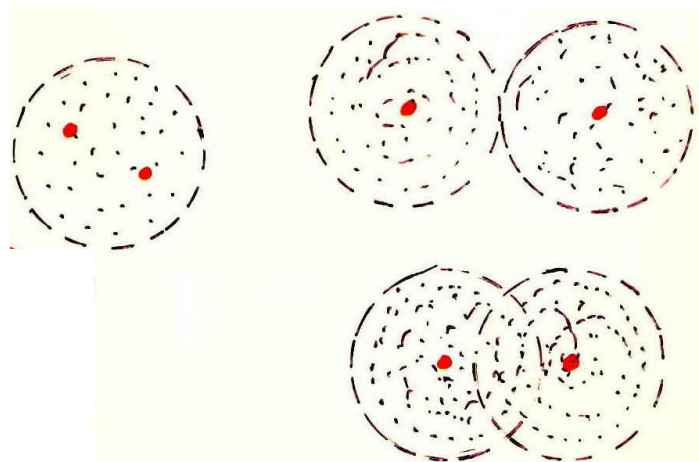
6 - Crítica pós treinamento (fundamental !)

a - Neurônios não (ou pouco) treinados - eliminar

b - Dois neurônios partilham a mesma classe

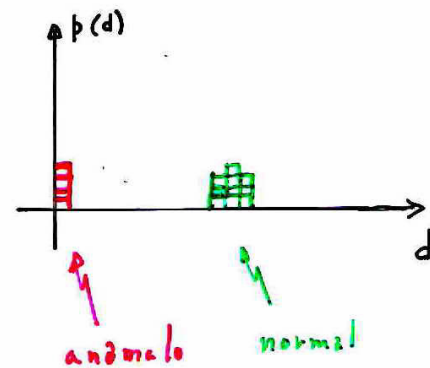
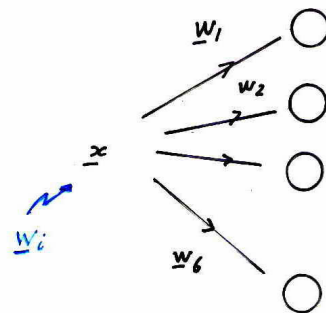
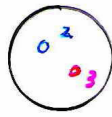
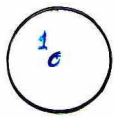
Deteccção:

$$|\vec{w}_i - \vec{w}_j| < 2r_0$$



Mais de um neurônio por classe mas r_0 é desconhecido ?

Deteção:



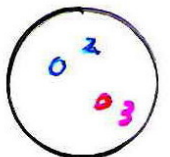
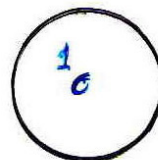
Correção

Retreinar ? não obrigatoriamente !

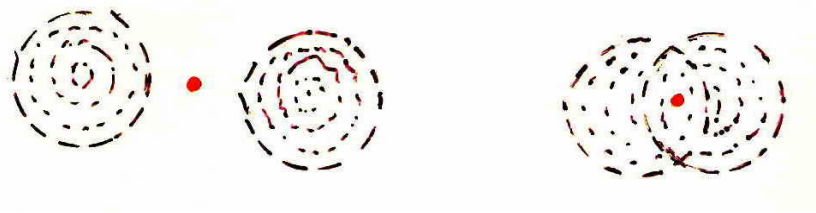
$$\vec{w}'_1 = \vec{w}_1$$

$$\vec{w}'_2 = \frac{1}{2}(\vec{w}_2 + \vec{w}_3)$$

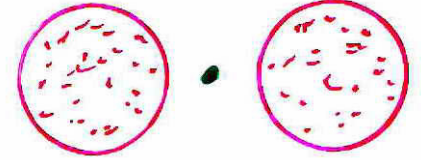
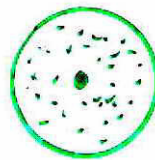
$$\vec{w}'_4 = \frac{1}{3}(\vec{w}_4 + \vec{w}_5 + \vec{w}_6)$$



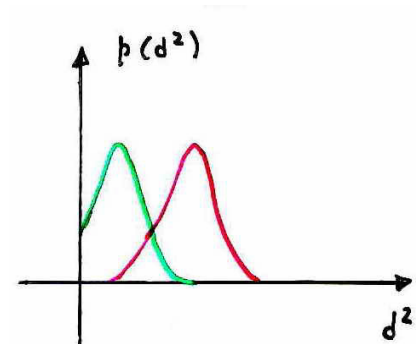
c – Um neurônio atende duas classes



r_0 desconhecido ?



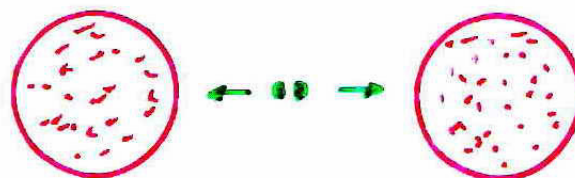
u_i do neurônio vencedor
é muito pequeno



Correção:

Duplicar w's anômalos

Retreinar os w's anômalos (duplicados)



7 – Resumo: O que usar ?

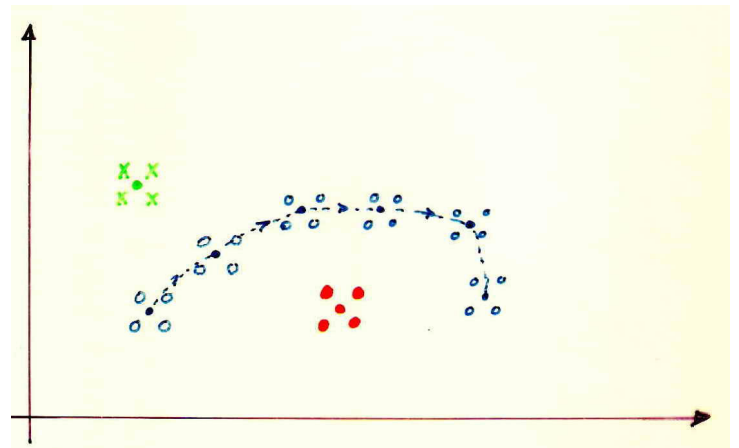
Número sinapses > número esperado de classes

Consciência

Crítica pós treinamento

8 - Sistemas Variantes no tempo:

Variações lentas dos
baricentros ? OK



Variações rápidas do baricentro ?

→ novas classes

reinicializar, ao menos parcialmente

