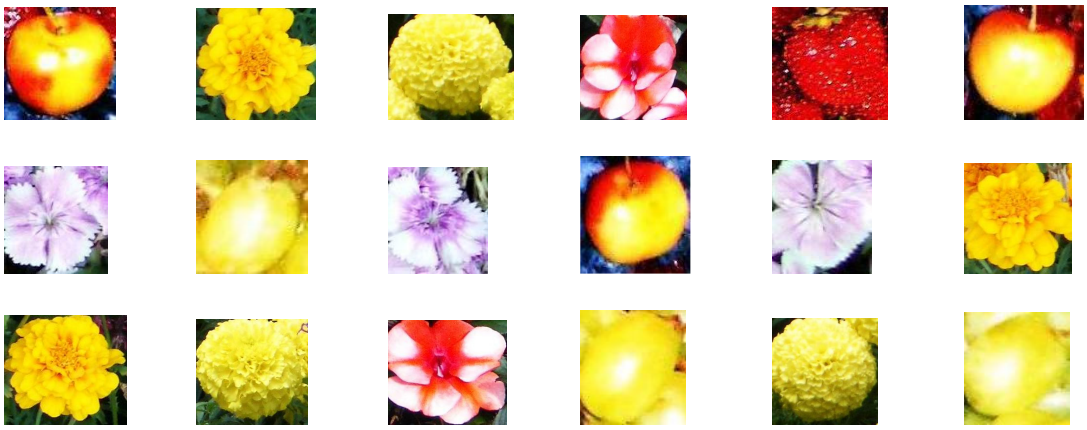
Camada de Kohonen

Classificação por Similaridade

Classes agrupam elementos “similares” entre si



versus classificadores arbitários

1 - Classificação por Similaridade – Critérios de pertinência

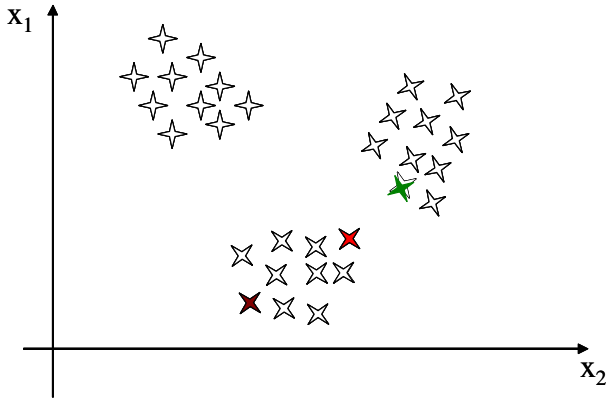
objetos físicos $\bigcirc_1$ $\bigcirc_2$ ⇒ objetos matemáticos $\underline{\mathbf{x}}_1$ $\underline{\mathbf{x}}_2$

similaridade física ⇒ similaridade matemática

$\bigcirc_1 \approx \bigcirc_2$ $\qquad$ $\underline{x}_1 \cong \underline{x}_2$ ou $\left|\underline{x}_1 - \underline{x}_2\right| <<$

Classificadores por similaridade

Critérios de Pertinência à uma Classe



Critério básico

Dois elementos pertencem à mesma classe se estão próximos entre si

Critério: k vizinhos mais próximos.

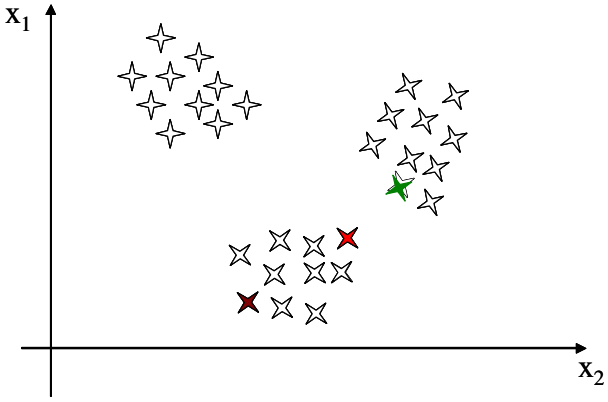
Considere os k elementos mais próximos da entrada que se pretende classificar. A entrada pertence à classe à qual pertencem a maioria dos k vizinhos.

Critério simplificado: vizinho mais próximo.

Se k=1 uma entrada pertence a uma classe se seu vizinho mais próximo é um elemento desta classe

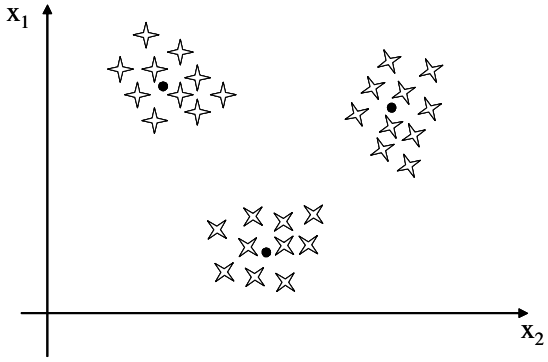
pouco prático

a descrição de uma classe exige a utilização de todos seus elementos (domínio da classe)



Padrão de classe:

Padrão $\underline{w}_i$ da classe C_i

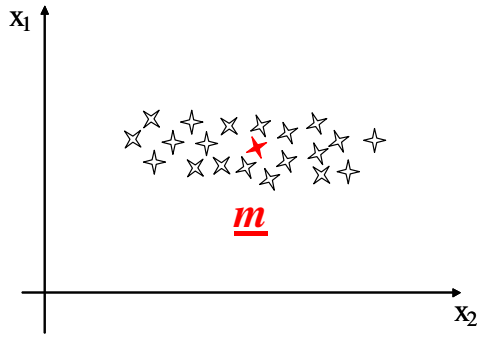


$$\underline{w}_i = \underset{\forall \underline{x}_j \in C_i}{E} \underline{x}_j = \frac{1}{N_i} \sum_{\forall \underline{x}_j \in C_i,\, j=1}^{N_i} \underline{x}_j$$

a **média** ou **baricentro** é escolhido como padrão da classe porque minimiza a **dissimilaridade** ou **erro de representação** da classe

Dispersão média intra classe ou erro (médio quadrático) de representação
Da classe C_j para o seu padrão $\vec{p}_j$

$$\sigma_j^2 = \underset{\forall x_i \in C_j}{E} \left\| x_i - p_j \right\|^2 = \frac{1}{n_j} \sum_{\forall x_i \in C_j} \left\| x_i - p_j \right\|^2$$



por componente k

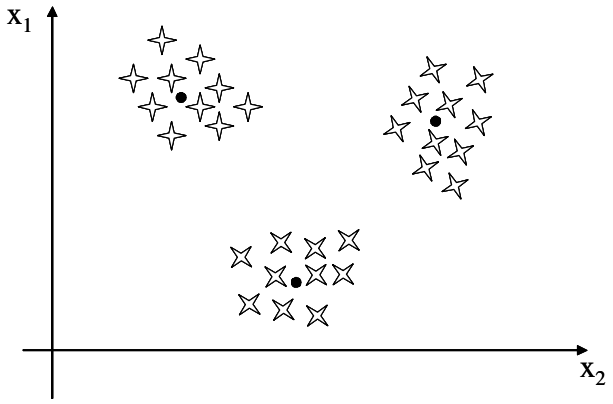
$$\sigma_{jk}^2 = \underset{\forall x_i \in C_j}{E} \left(x_{ik} - p_{jk} \right)^2 = \frac{1}{n_j} \sum_{\forall x_i \in C_j} \left(x_{ik} - p_{jk} \right)^2$$

após pequena álgebra $\quad \dfrac{\partial \sigma_{jk}^2}{\partial p_{jk}} = 0 \quad \Rightarrow \quad p_{jk} = m_{jk} \quad \Rightarrow \quad \vec{p}_j = \vec{m}_j$

o padrão que minimiza a dispersão intra classe (ou erro de representação) de uma classe é o seu baricentro

1.1 - Critério de pertinência 1 :

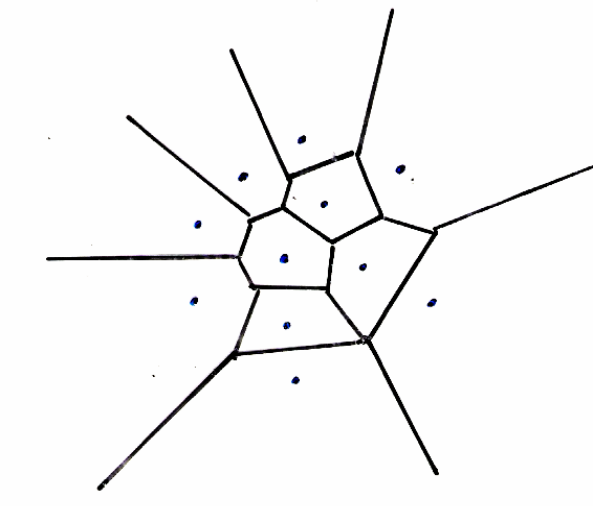
Padrão mais similar à entrada



$$\underline{x}_i \in C_i \quad \text{sse} \quad \left|\underline{x}-\underline{w}_i\right|^2 < \left|\underline{x}-\underline{w}_j\right|^2 \quad \forall\, j \neq i \qquad \text{eq (1)}$$

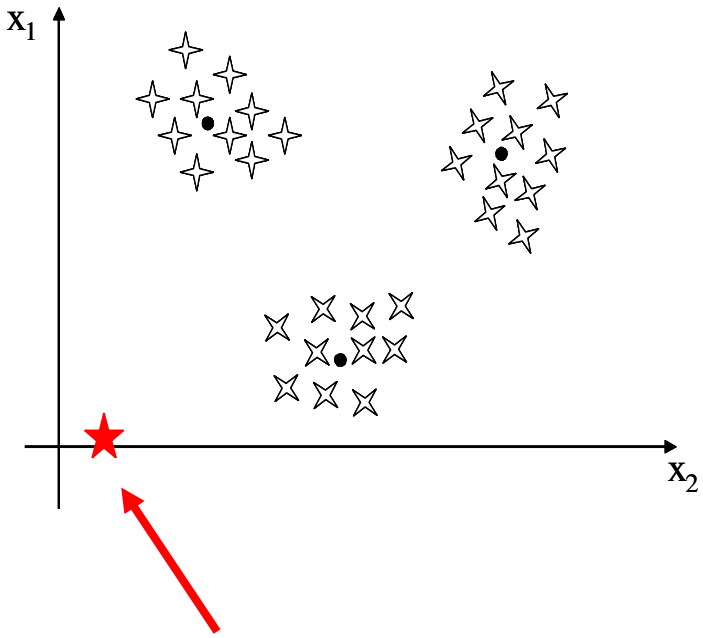
porque minimiza a dispersão intra-classes total

Separadores para o critério 1

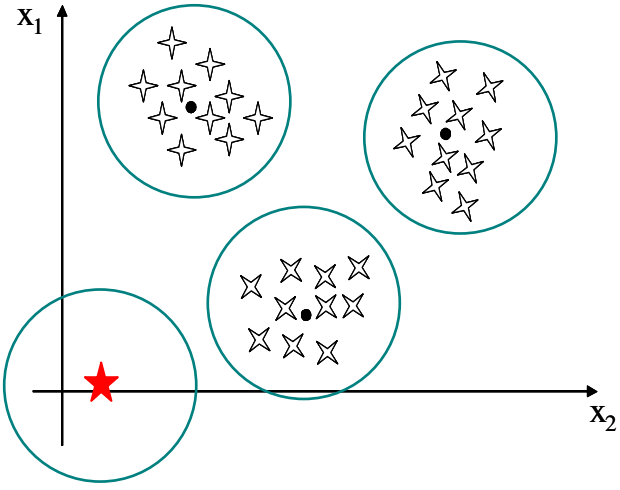


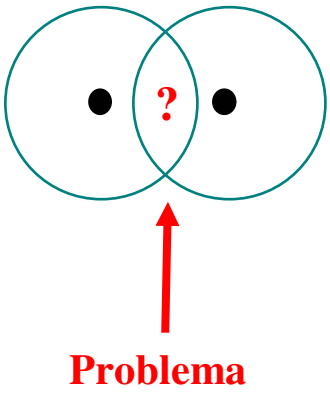
Tecelagem (diagrama) de Voronoi

Problema:



1.2 - Critério de pertinência 2 – Padrão que satisfaz uma similaridade minima



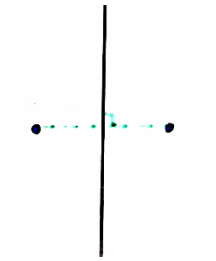


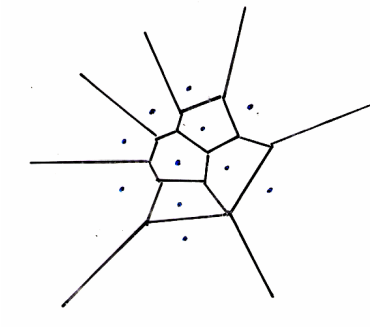
$\underline{x}_i \in C_i$ sse $\left|\underline{x} - \underline{w}_i\right|^2 < r_0^2$ eq (2)

r_0 – raio de similaridade

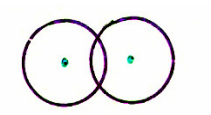
Separadores

Critério 1 - padrão mais similar (mais próximo) a entrada Eq(1)



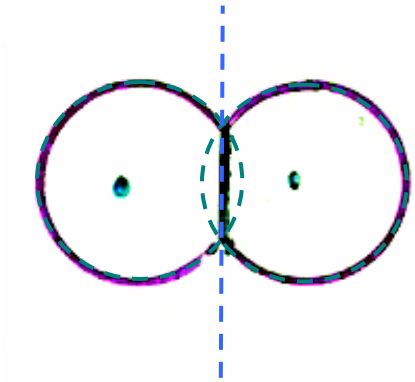


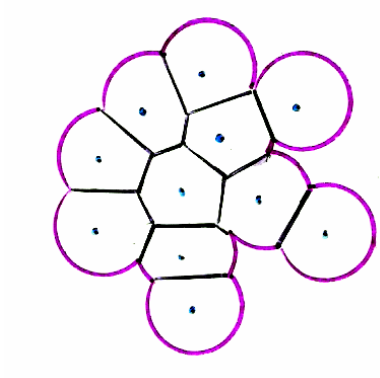
Critério 2 - mínima similaridade exigida Eq (2)



1.3 - Critérios 1 e 2, Eq(1) + Eq(2)

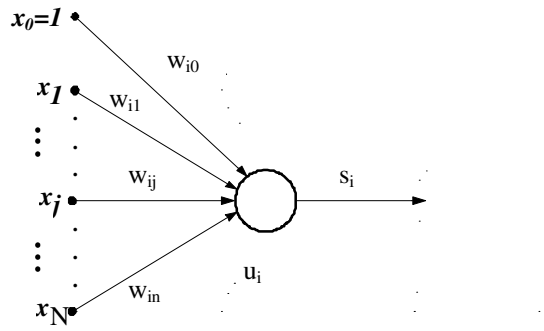
padrão mais similar & mínima similaridade atingida

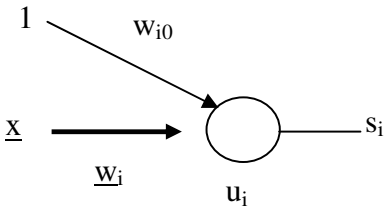




2 - Redes Neurais

Neurônio para redes feedforward





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

$$\tilde{y}_k = \varphi_k(u_k) = \begin{cases} u_k & \textit{neurônio linear} \\ tgh\,(u_k) & \textit{neurônio tgh} \end{cases}$$

Neurônio como comparador de similaridade

de uma entrada $\underline{x}$ com dois padrões $\underline{w}_i$ e $\underline{w}_j$

$$d_i^2 = |\underline{x} - \underline{w}_i|^2 = |\underline{x}|^2 + |\underline{w}|^2 - 2\underline{w}^t\,\underline{x}$$

$$u_i = \underline{w}^t\,\underline{x} + w_{i0}$$

$$\textit{fazendo} \quad w_{i0} = -\frac{1}{2}|\underline{w}_i|^2$$

$$u_i = \frac{1}{2}\left(|\underline{x}|^2 - d_i^2\right)$$

logo

$$u_i > u_j \;\Leftrightarrow\; d_i^2 < d_j^2$$

e u é um comparador de similaridade

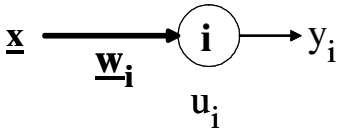
y_i – depende dos outros neurônios

Neurônio como medidor de similaridade

entre uma entrada x e um padrão w_i

uma outra definição

$$u_i = -\|x - w_i\|^2 = -d_i^2 \leq 0$$



u_i - medida de similaridade entre x e w_i

$u_i = 0$ distância nula = máxima similaridade

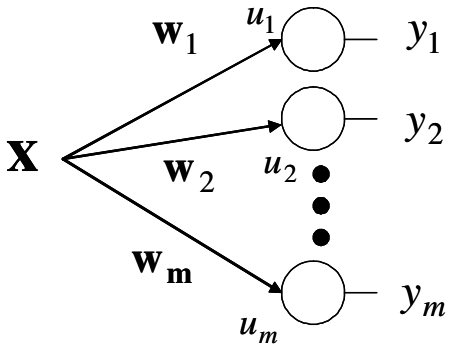
y_i – depende dos outros neurônios

3- Camada de Kohonen

Template Matching

$$u_i = -d_i^2$$

maior u_i =
menor distância =
maior similaridade

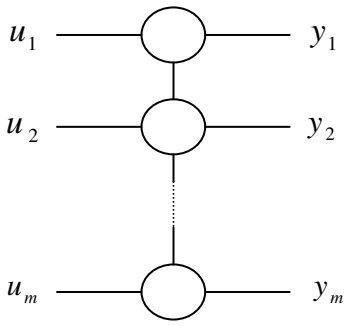


u_i é uma medida de similaridade entre x e w_i

Winner-takes-all

$$y_i = 1 \quad sse \quad u_i > u_j \quad \forall\, j \neq i$$

$$y_i = 0 \quad caso\ contrário$$



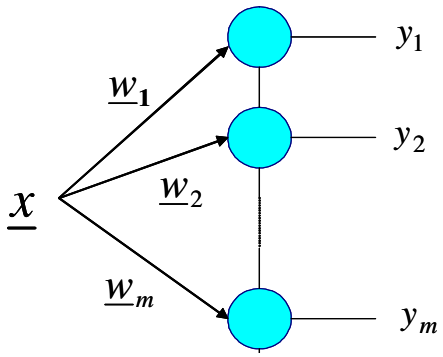
Camada de Kohonen

[versão unidimensional, simplificada (D=0)]

Classe C_i

Padrão $\underline{w}_i$

Indicador y_i

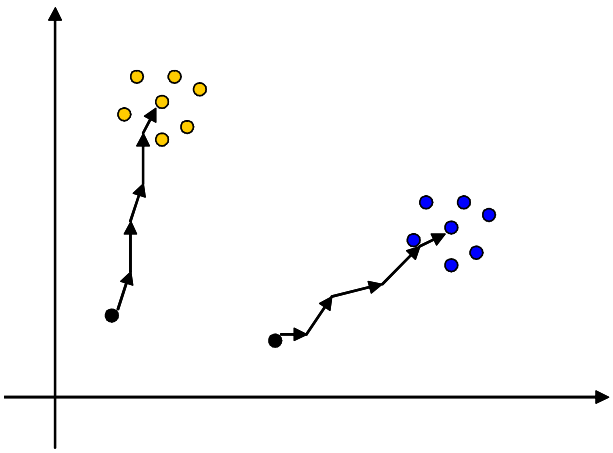


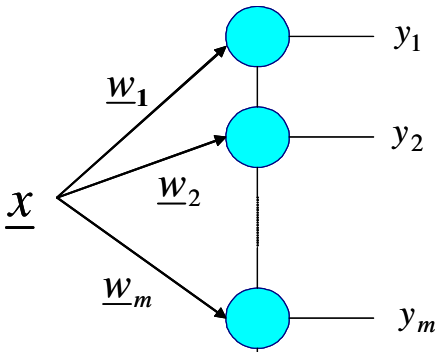
Se $y_i = 1$ então $\underline{x} \in C_i$

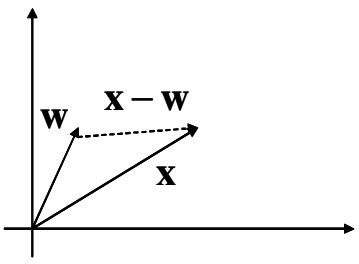
pelo critério 1 (padrão mais similar a entrada).

Camada de Kohonen

Treinamento Supervisionado

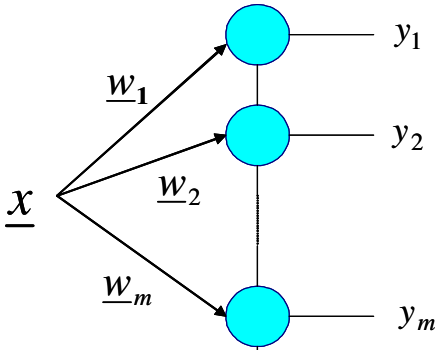






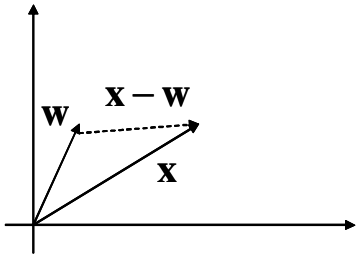
4 - Treinamento da Camada de Kohonen

$$\underline{x}(n) \in C_i \quad \Rightarrow \quad y_i = 1$$
$$y_j = 0 \;\forall\; j \neq i$$



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

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



Onde estabiliza, qual o valor final de $\underline{w}$?

Evolução do $\underline{w}$ de uma classe

Passo n: $\underline{x}(n)$ $\underline{w}(n)$

$x(n) \in C_i$

$$\begin{cases} \underline{w}_i(n+1) = (1-\alpha)\,\underline{w}_i(n) + \alpha\,\underline{x}(n) \\ \underline{w}_i(n+1) = \underline{w}_i(n) \qquad \forall j \neq i \end{cases}$$

$\underline{w}(0)$

$\underline{w}(1) = (1-\alpha)\,\underline{w}(0) + \alpha\,\underline{x}(1)$

$$\begin{aligned}\underline{w}(2) &= (1-\alpha)\,\underline{w}(1)+ \alpha\,\underline{x}(2)\\ &= (1-\alpha)[(1-\alpha)\,\underline{w}(0) + \alpha\,\underline{x}(1)] + \alpha\,\underline{x}(2)\\ &= (1-\alpha)^2\,\underline{w}(1)+ \alpha(1-\alpha)\,\underline{x}(1)+ \alpha\,\underline{x}(2)\end{aligned}$$

$$\begin{aligned}\underline{w}(3) &= (1-\alpha)\,\underline{w}(2) + \alpha\,\underline{x}(3)\\ &= (1-\alpha)[(1-\alpha)^2\,\underline{w}(0)+ \alpha(1-\alpha)\,\underline{x}(1)+ \alpha\,\underline{x}(2)] + \alpha\,\underline{x}(3)\\ &= (1-\alpha)^3\,\underline{w}(0)+ \alpha(1-\alpha)^2\,\underline{x}(1)+ \alpha(1-\alpha)\,\underline{x}(2)+ \alpha\,x(3)\end{aligned}$$

$\vdots$

$$\begin{aligned}w(n) &= (1-\alpha)^n\,\underline{w}(0)+ \alpha[(1-\alpha)^{n-1}\,\underline{x}(1)+ (1-\alpha)^{n-2}\,\underline{x}(2)+ \ldots + \underline{x}_n]\\ &= (1-\alpha)^n\,\underline{w}(0)+ \alpha\sum_{i=1}^{n}(1-\alpha)^{n-i}\,\underline{x}(i)\end{aligned}$$

algumas aproximações:

$0 < \alpha < 1 \qquad 0 < (1-\alpha) < 1 \qquad n >> 1 \qquad (1-\alpha)^n \to 0$

$$1+(1-\alpha)+(1-\alpha)^2+(1-\alpha)^3+\ldots = \sum_{i=0}^{\infty}(1-\alpha)^i = \frac{1}{1-(1-\alpha)} = \frac{1}{\alpha}$$

$n >> 1 \qquad \alpha \approx \dfrac{1}{\sum_{i=0}^{n-1}(1-\alpha)^i}$

$\underline{w}(n) = (1-\alpha)^n\,\underline{w}(0)+ \alpha\sum_{i=1}^{n}(1-\alpha)^{n-i}\,\underline{x}(i)$ (0 → $(1-\alpha)^n$; α → $\sum$)

$$\cong \frac{\sum_{i=1}^{n}(1-\alpha)^{n-i}\,\bar{x}(i)}{\sum_{i=0}^{n-1}(1-\alpha)^i} = \frac{\sum_{i=0}^{n-1}(1-\alpha)^i\,\bar{x}(n-i)}{\sum_{i=0}^{n-1}(1-\alpha)^i}$$

Média das entradas que pertencem a classe

ponderada geometricamente pelo tempo !

$$\underline{w}_n \cong \frac{\sum_{i=1}^{n}(1-\alpha)^{n-i}\vec{x}(i)}{\sum_{i=1}^{n}(1-\alpha)^{n-i}} = \left(\frac{1}{\sum_{i=0}^{n-1}(1-\alpha)^{i}}\right)\sum_{i=0}^{n-1}(1-\alpha)^{i}\vec{x}(n-i)$$

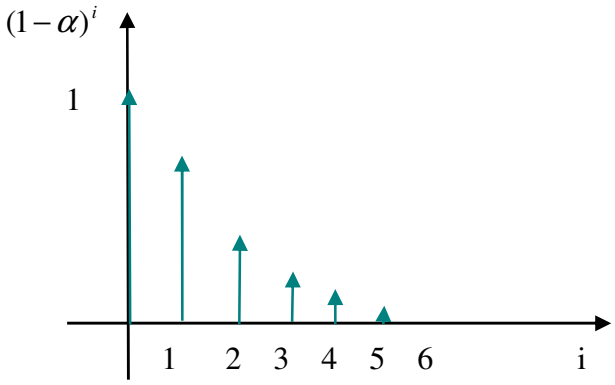
instante atual = n

x(n-í) = entrada atrasada

de i intervalos de tempo

$(1-\alpha)^i$ = ponderador

para a entrada x(n-í)



Note que se

$$\alpha \cong 1$$

e a estatística de $\vec{x}$ for invariante no tempo

$$\vec{w}_n \cong \left(\frac{1}{\sum_{i=0}^{n-1}(1-\alpha)^{i}}\right)\sum_{i=0}^{n-1}(1-\alpha)^{i}\vec{x}(n-i) \cong E[\vec{x}]$$

$$\vec{w}_n \cong E[\vec{x}]$$

4.1 – Tempo de medida

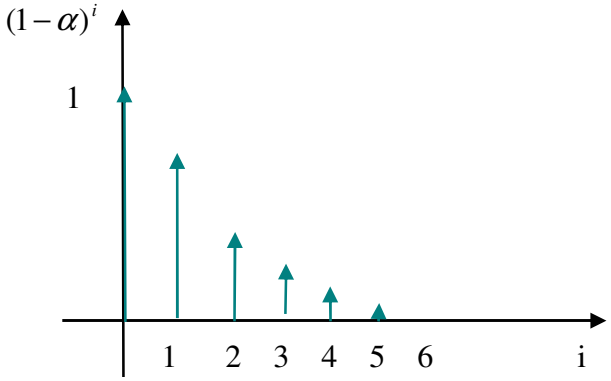
Fim (prático) da soma ponderada (tempo de medida)

Atraso zero **Ponderador = $(1-\alpha)^0 = 1$**

Último atraso à ser considerado: **Ponderador = $.02 = (1-\alpha)^i$**

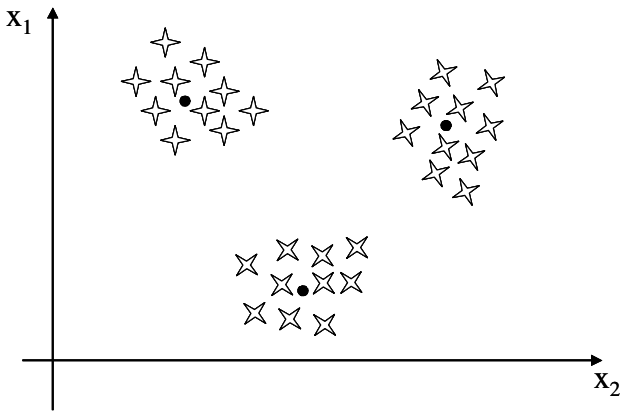
$$i = \frac{\ln .02}{\ln(1-\alpha)} \approx \frac{-4}{-\alpha}\Big|_{\alpha \to 0}$$

$$i \approx \frac{4}{\alpha}$$



4.2 - Erro na determinação do baricentro:

Se em uma classificação por similaridade cada elemento $\underline{x}$ de uma classe C_k pode ser representado pelo padrão $\underline{w}_k$ da classe adicionado de um vetor de ruído $\underline{r}$ com média nula.



$$\underline{x} = \underline{w}_k + \underline{r}$$

Cada componente x_j de $\underline{x}$ é então representada pela componente correspondente de $\underline{w}_k$, w_j, adicionada de um ruído r_j de média nula e variância σ_{xj}^2

$$x_j = w_j + r_j$$

Com que precisão componente w_j é calculada pela camada de Kohonen, qual a sua variância σ_{wj}^2 ? A componente j de $\underline{w}$, w_j, em um instante n >> 1 é dada por

$$w_j \cong \left(\frac{1}{\sum_{i=0}^{n} (1-\alpha)^i} \right) \sum_{i=0}^{n} (1-\alpha)^i x_j(n-i)$$

Sendo uma soma ponderada sua variância será dada por

$$\sigma_{wj}^2 \cong \left(\frac{1}{\sum_{i=0}^{\infty} (1-\alpha)^i} \right)^2 \sum_{i=0}^{\infty} (1-\alpha)^{2i} \sigma_{xj}^2 \qquad ou \qquad \sigma_{wj}^2 \cong \frac{\alpha}{2} \sigma_{xj}^2$$

4. 3 – Compromisso entre Erro vs. Tempo de Medida

Quanto menor α mais precisamente o baricentro será determinado. Mas, como esperado, maior o tempo (número de passos) necessário para calculá-lo

$$\sigma_{wj}^2 \cong \frac{\alpha}{2} \sigma_{xj}^2 \qquad \# passos = i \cong \frac{4}{\alpha}$$

Exemplo:

$\sigma_x = .05$ (5%)

$\sigma_w = .01$ (1%) requerido $\qquad \alpha = \frac{2\sigma_{wj}^2}{\sigma_{xj}^2} = .08 \qquad \# passos = \frac{4}{\alpha} = 50 \ passos$

Mas se:

$\sigma_w = .001$ (.1%) requerido $\qquad \alpha = \frac{2\sigma_{wj}^2}{\sigma_{xj}^2} = .0008 \qquad \# passos = \frac{4}{\alpha} = 5.000 \ passos$!!

4.4 Considerações:

4.4.1. Fim do treinamento ?

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

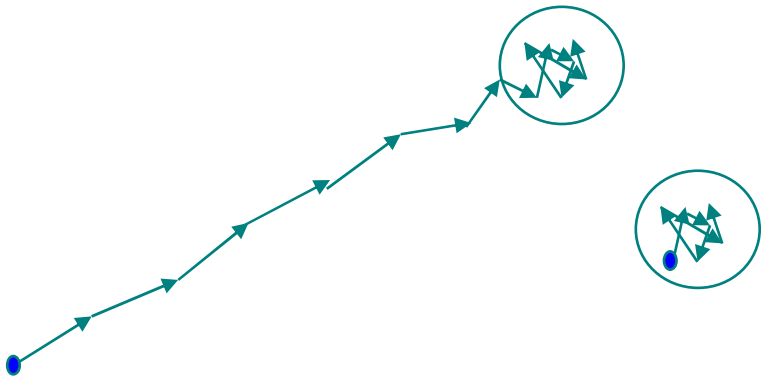
$$\Delta \vec{w} = \alpha(\vec{x} - \vec{w})$$

$$E(\Delta \vec{w}) = E[\alpha(\vec{x} - \vec{w})] = \alpha[E(\vec{x}) - \vec{w}] = 0$$

$$\vec{w} = E(\vec{x})$$

4.4.2. Valores iniciais das sinapses

Irrelevantes ! Mas escolher o valor inicial como o primeiro elemento da classe acelera o treinamento, porque a sinapse já começa dentro da classe.



4.4.3. Passo de treinamento

reduzir ao longo do tempo $\alpha(n) = \alpha_0 \, e^{-\frac{n}{N_0}}$

$\alpha(0) = \alpha_0 \qquad \alpha(n+1) = k\,\alpha(n) \qquad$ onde $\qquad k = 1 - \frac{1}{N_0}$

o processo acaba ($\alpha(n) << \alpha_0$) para $n > 4N_0$

e as sinapses pouco ativadas (classes pouco populosas) ?

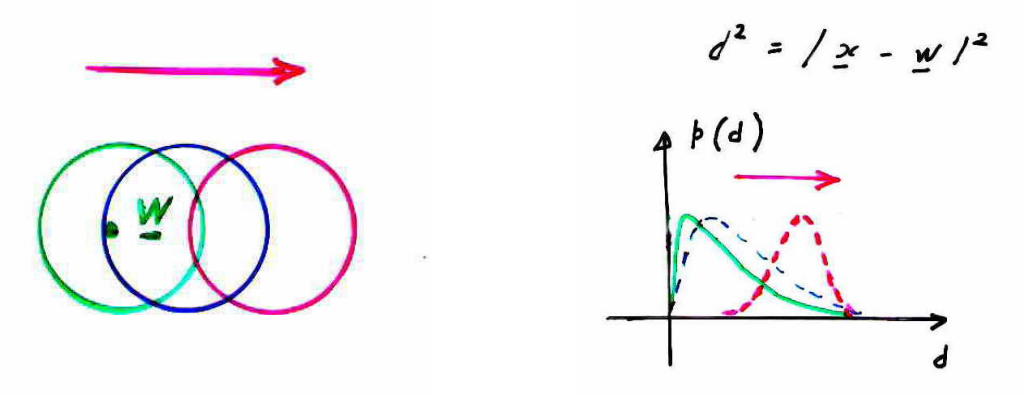
usar α diferenciado por sinapse

$\alpha_{w_j}(0) = \alpha_0 \qquad \alpha_{w_j}(n_j+1) = k\,\alpha_{w_j}(n_j)$

onde n_j é o número de vezes que a sinapse $\underline{w}_j$ foi treinada.

4.4.4 Treinamento dinâmico, adaptativo

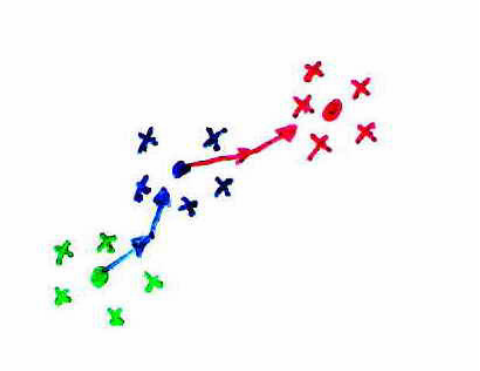
Uma classe varia de posição no tempo. Como saber ?

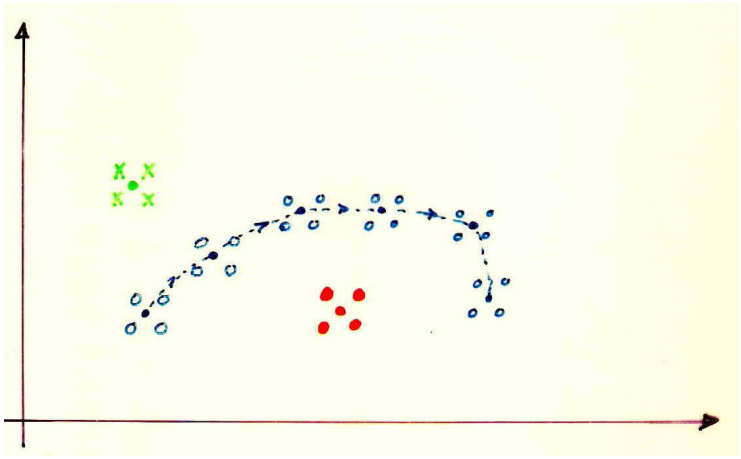


O valor médio de $u_i = -d^2_i$ diminui

(o valor da distância média (ou a da moda) aumenta)

Como corrigir ? Ligar o treinamento até que d^2_i volte a seus valores normais.





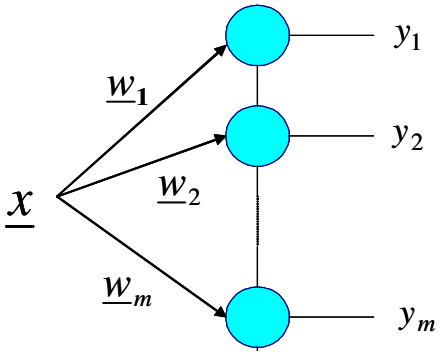
Se a variação do baricentro for lenta e o treinamento estiver ligado o padrão segue o baricentro

5 – LVQ – Learning Vector Quantization

$$\vec{x}(n) \in C_i$$

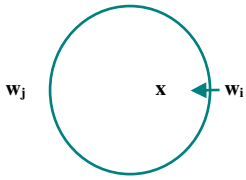
e

neurônio vencedor y_j



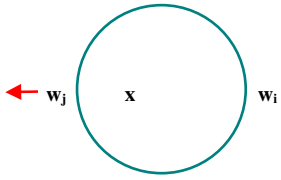
j = i ou j ≠i

$$se \quad j = i \qquad \vec{w}_j(n+1) = \vec{w}_j(n) + \alpha\left[\vec{x}(n) - \vec{w}_j(n)\right]$$



se a entrada pertence à classe vencedora, aproxima a sinapse vencedora da entrada e conseqüentemente do centro da classe

$$se \quad j \neq i \qquad \vec{w}_j(n+1) = \vec{w}_j(n) - \alpha\left[\vec{x}(n) - \vec{w}_j(n)\right]$$



se a entrada não pertence à classe vencedora, afasta a sinapse vencedora da entrada e, conseqüentemente, do centro $\underline{\mathbf{w}}_i$ da classe a que a entrada pertence. Isto aumenta a chance da classe certa vencer em uma próxima competição.

Note que se as sinapses tiverem sido inicializadas com um elemento da classe a probabilidade do segundo caso ($j \neq i$) ocorrer e é muito pequena. Isto também ocorre após a fase inicial do treinamento, quando as sinapdes já foram arrastadas para dentro de suas respectivas classes.

Obs:

1 - O passo de aprendizagem deve decrescer com o tempo

$$\alpha = \alpha(n) = \alpha_0 \exp\left(-\frac{n}{\tau_\alpha}\right) \qquad \alpha_0 \approx 0{,}1 \qquad \tau_\alpha \approx 1000$$

2 - O equilíbrio é atingido quando $E[\Delta\vec{w}_i] = \vec{0}$, o que ocorre quando

$$\vec{w}_i = \vec{m}_i = \underset{\forall \vec{x} \in C_i}{E}(\vec{x})$$

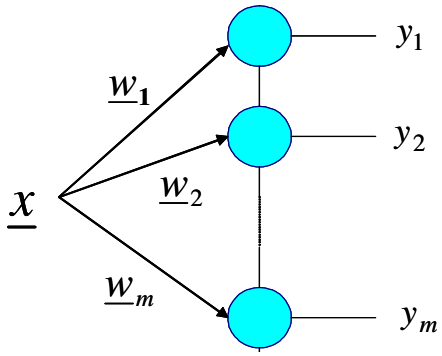
3 - Este processo corresponde a minimizar a função objetivo

$$F = \sum_{\forall C_i} F(\vec{w}_i) \qquad F(\vec{w}_i) = \sum_{\forall \vec{x} \in C_i} \left|\vec{x} - \vec{w}_i\right|^2$$

isto é, minimizar a dispersão intra-classe total

5a – Um método alternativo (???)

LVQ – Learning Vector Quantization



$\vec{x}(n) \in C_i$ aproxima o padrão vencedor de $\vec{x}(n)$ e afasta os demais padrões de $\vec{x}(n)$

$$\vec{x}(n) \in C_i \qquad \begin{cases} y_i = 1 \\ y_j = 0 \quad \forall\, j \neq i \end{cases}$$

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

$$\vec{w}_j(n+1) = \vec{w}_j(n) - \alpha\left[\vec{x}(n) - \vec{w}_j(n)\right] \quad \forall\, j \neq i$$

ou, usando uma fórmula única

$$\vec{x}(n) \in C_i \qquad \begin{cases} y_i = 1 \\ y_j = 0 \quad \forall\, j \neq i \end{cases}$$

$$\vec{w}_j(n+1) = \vec{w}_j(n) + (2y_i - 1)\,\alpha\left[\vec{x}(n) - \vec{w}_j(n)\right] \qquad \forall\, j$$

O passo de aprendizagem deve decrescer com o tempo

$$\alpha = \alpha(n) = \alpha_0 \exp\left(-\frac{n}{\tau_\alpha}\right) \qquad \alpha_0 = 0{,}1 \qquad \tau_\alpha = 1000$$

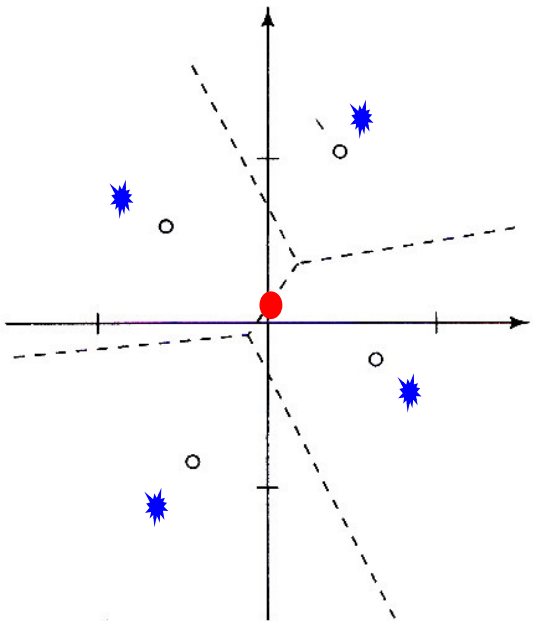
Este processo melhora as fronteiras de classificação (Haykin).

O equilíbrio é atingido quando $E[\Delta\vec{w}_i]=\vec{0}$, o que ocorre quando

$$\vec{w}_i = p_i \underset{\forall\vec{x}\in C_i}{E}(\vec{x}) - (1-p_i)\underset{\forall\vec{x}\notin C_i}{E}(\vec{x}) =$$

$$= 2p_i \underset{\forall\vec{x}\in C_i}{E}(\vec{x}) - \underset{\forall\vec{x}}{E}(\vec{x}) =$$

$$= 2p_i\vec{m}_i - \vec{m}$$



onde p_i é a probabilidade de uma entrada $\vec{x}$ pertencer à classe C_i , $\vec{m}_i$ é o baricentro de C_i e $\vec{m}$ é o baricentro de todas as entradas, normalmente $\vec{m}=\vec{0}$. O efeito é deslocar os padrões das classes (✸) para longe do centro de todas as entradas (●) mas também dos baricentros (O) das mesmas.

Este processo corresponde a otimizar a função objetivo

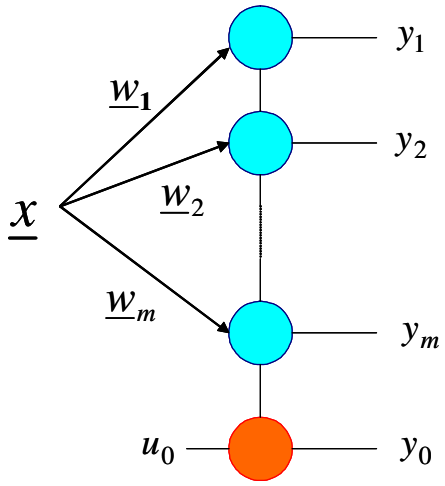
$$F(\vec{w}_i) = \sum_{\forall\vec{x}\in C_i}|\vec{x}-\vec{w}_i|^2 - \sum_{\forall\vec{x}\notin C_i}|\vec{x}-\vec{w}_i|^2$$

isto é, minimizar a dispersão intra classe de C_i e maximizar a distância de w_i aos elementos das outras classes.

6 - E o segundo critério (similaridade mínima) ?

Camada de Kohonen (aumentada)

$$u_0 = -\, r_0^2$$



Se $y_i = 1$ entao

$\underline{x} \in C_i$

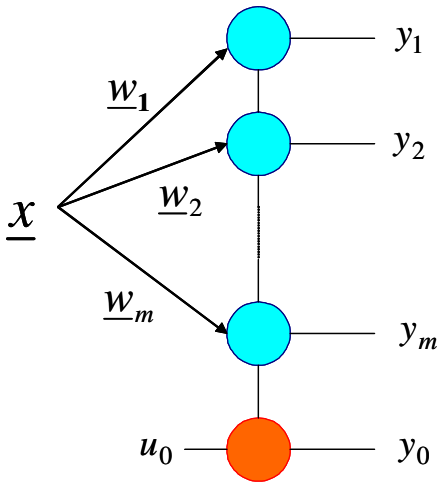
pelos critérios **1 (padrao mais similar a entrada) e**

2 (satisfaz a similaridade minima exigida).

Se $y_i = 1$ então

$\underline{x} \in C_i$

pelos critérios
1 (centro de classe mais similar à entrada) e
2 (satisfaz à similaridade mínima)



Se $y_0 = 1$ então

$\underline{x} \notin C_i \quad \forall\, i$

$\underline{x}$ não satisfaz ao critério 2
para nenhuma classe

7 – HVQ – Hierarquical Vector Quantization