

Redes Neurais Self-Organizing Maps

Marcos G. Quiles (quiles@unifesp.br)

Agenda

I. Mapas auto-organizáveis de Kohonen (SOM)



APRENDIZADO NÃO-SUPERVISIONADO

- Não utiliza informação do especialista
- O modelo deve extrair dos dados características ou padrões significativos
- Isto só é possível se houver redundância nos dados
- A ausência de redundância torna os dados semelhantes a ruídos aleatórios



APRENDIZADO NÃO-SUPERVISIONADO

- Existem Vários Modelos de RNAs Não-Supervisionados
- Os algoritmos de treinamento podem ser divididos em dois grupos principais:
 - Aprendizado Competitivo
 - **SOM, GNG, ART1, ART2, etc.**
 - Aprendizado Hebbiano
 - Hopfield, PCA, etc.

APRENDIZADO COMPETITIVO

- Os neurônios competem para se tornar ativos em função da apresentação de um dado padrão de entrada na rede. Em sua forma mais simples - Winner-Takes-All.



APRENDIZADO COMPETITIVO

Características Básicas

- Regra de Propagação $y_j = \mathbf{XW}_j$
- Função de ativação: degrau (para o neurônio vencedor)
- Topologia: uma única camada de processadores
- Algoritmo de aprendizado: não-supervisionado $\Delta w_{ij} = \alpha(x_i - w_{ij})$
- Valores de entrada: binário/continuo

OBS.: Regras de Propagação também pode ser

$$y_j = ||\mathbf{x}(n) - \mathbf{w}_j||, \quad j = 1, 2, \dots, l$$



APRENDIZADO COMPETITIVO

Funcionamento

- Os vetores X e W_j devem ser normalizados;
- Somente o neurônio vencedor é ativado

(neurônio com maior valor y)

$$s_j = 1 \text{ se } y_j > y_k, \forall j \neq k$$

$$s_j = 0 \text{ caso contrario}$$

- Somente os pesos do neurônio vencedor e seus vizinhos pre-definidos são atualizados.



APRENDIZADO COMPETITIVO

- Não existe conhecimento a respeito da classe a que o padrão pertence;
- Aprendizado depende das entradas e de suas densidades de probabilidade;
- Precisa de um grande conjunto de dados com redundância para adquirir conhecimento das propriedades estatísticas dos padrões;
- Dependência do histórico de apresentação dos padrões



MAPA DE KOHONEN

- Modelo neural que utiliza o aprendizado competitivo em seu treinamento
- Redes SOM (Self-Organizing Maps), SOFM, Mapas de Kohonen.
- Apresenta uma forte inspiração neurofisiológica. Baseada no mapa topológico presente no córtex cerebral.
 - Córtex Auditivo
 - Córtex Visual
 - Córtex Motor, etc.



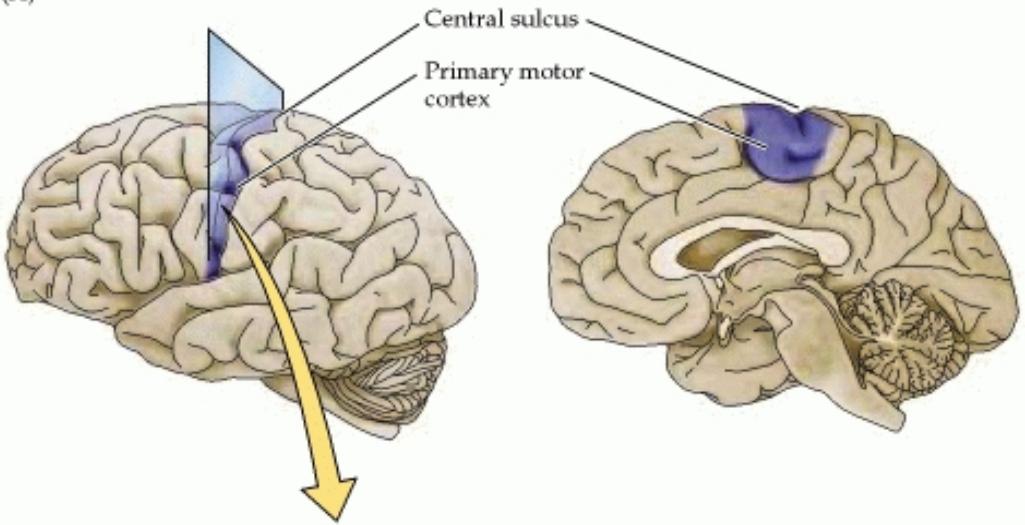
MAPA DE KOHONEN

- Observa-se que os neurônios dentro dessas regiões do cérebro são espacialmente ordenados. Desta forma, neurônios topologicamente próximos tendem a responder a padrões ou estímulos semelhantes

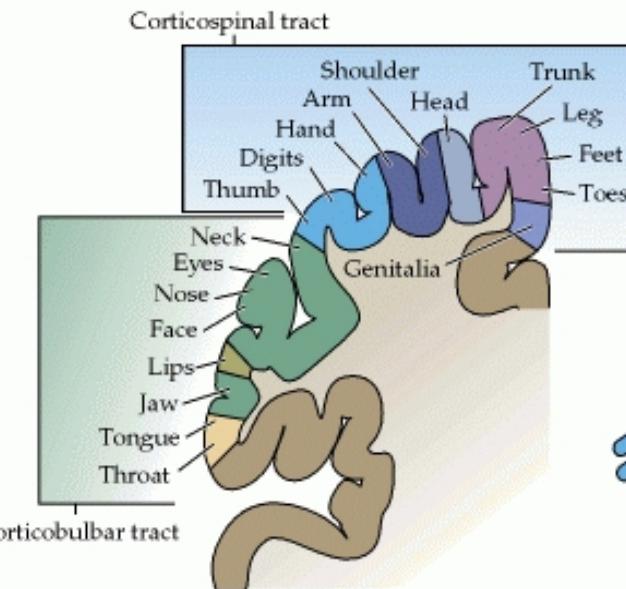


MAPA DE KOHONEN

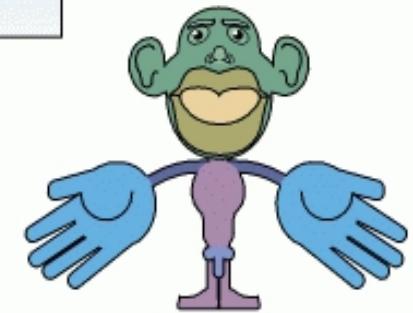
(A)



(B)

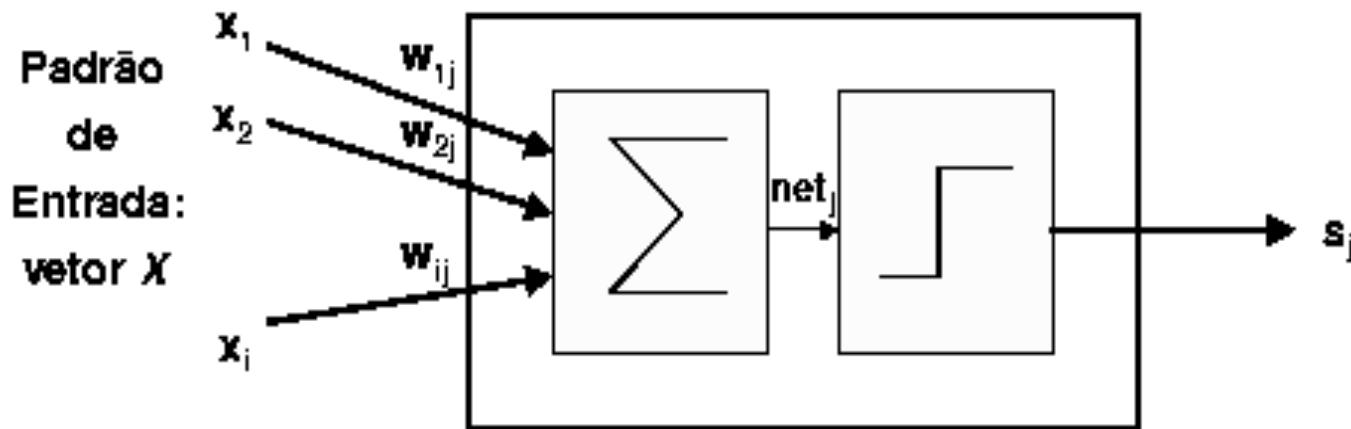


(C)



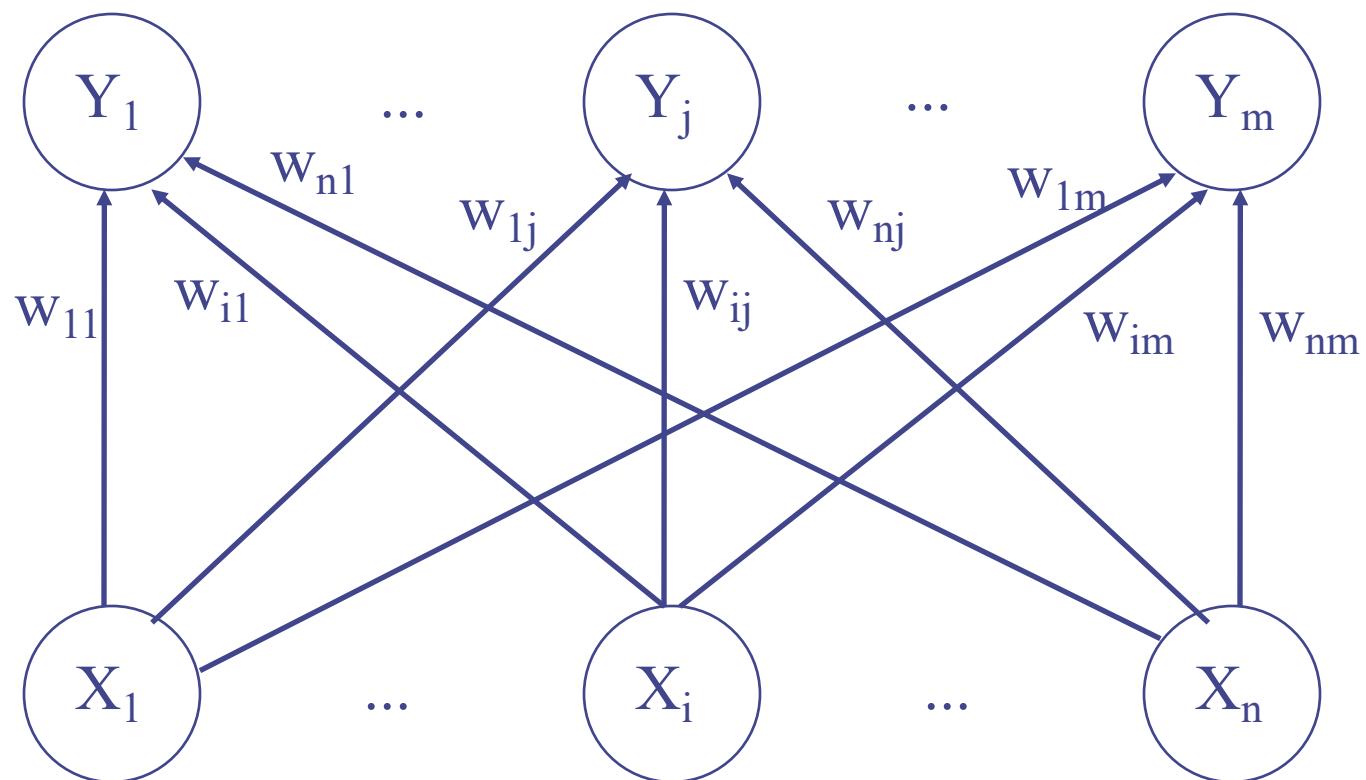
MAPA DE KOHONEN

- *Elemento Processador*

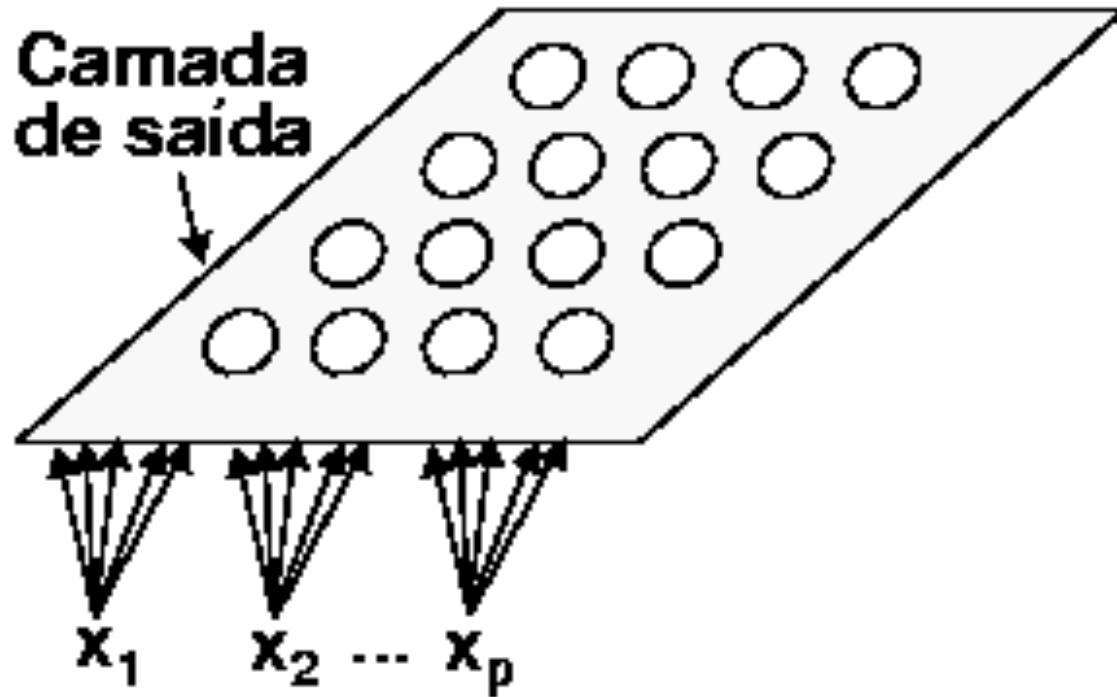


MAPA DE KOHONEN

Topologia



MAPA DE KOHONEN



MAPA DE KOHONEN

Dada uma amostra **X** do espaço de entrada representando um padrão de ativação aplicado à rede, três processos estarão envolvidos na formação do mapa auto-organizável:

Competição

Cooperação

Adaptação Sináptica



MAPA DE KOHONEN

Competição: Para cada padrão de entrada, os neurônios na rede computam seus respectivos valores de uma função discriminante. Esta função discriminante fornece a base para a competição entre os neurônios e aquele que melhor-equivale ao valor da função discriminante é declarado o *neurônio vencedor*.



MAPA DE KOHONEN

- O Processo Competitivo:

- Vetor de entrada:

$$\mathbf{x} = [x_1, x_2, \dots, x_m]^T$$

- Rede (pesos dos neurônios)

$$\mathbf{w}_j = [w_{j1}, w_{j2}, \dots, w_{jm}]^T, \quad j = 1, 2, \dots, l$$

- Neurônio vencedor

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j = 1, 2, \dots, l$$



MAPA DE KOHONEN

Cooperação: O neurônio vencedor determina a localização espacial de uma vizinhança topológica de neurônios excitados, fornecendo assim a base para a cooperação entre neurônios vizinhos.



MAPA DE KOHONEN

- O Processo Cooperativo:
 - O neurônio vencedor i , está no centro de uma região (vizinhança)
 - Neurônios vizinhos se beneficiam da ativação do neurônio vencedor
 - Definição da vizinhança topológica do neurônio i

$$h_{j,i(\mathbf{x})} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right)$$



MAPA DE KOHONEN

- O Processo Cooperativo:

- Definição da vizinhança topológica do neurônio i

$$h_{j,i(\mathbf{x})} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right)$$

- Decaimento da Vizinhança

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau}\right)$$



MAPA DE KOHONEN

Adaptação Sináptica: Este último mecanismo permite aos neurônios excitados incrementar seus valores individuais da função discriminante em relação ao padrão de entrada através de ajustes adequados aplicados à seus pesos sinápticos.



MAPA DE KOHONEN

Adaptação Sináptica (cont.): Este ajuste é feito de tal maneira que a resposta do neurônio vencedor à uma subsequente aplicação de um padrão similar é ampliada, e pode ser alcançado através da fórmula

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \eta(t)h_{j,i(\mathbf{x})}(t)(\mathbf{x} - \mathbf{w}_j(t))$$

onde $\eta(t)$ é um *parâmetro de taxa de aprendizado*, e $h_{j,i(x)}(t)$ uma *função de vizinhança* em torno de um neurônio

vencedor.

$$\eta(t) = \eta_0 \exp\left(-\frac{t}{\tau_2}\right)$$



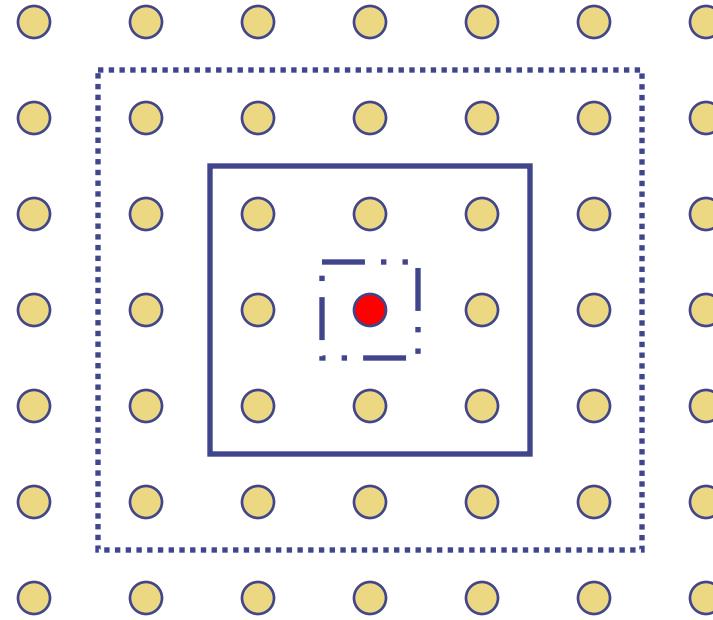
MAPA DE KOHONEN

Assim, após repetidas apresentações de dados de treinamento, os pesos sinápticos tendem a seguir a distribuição do vetor de entrada devido à esta atualização da vizinhança. O algoritmo portanto leva a uma ordenação topológica do mapa de característica em relação ao espaço de entrada, no sentido que os neurônios adjacentes terão vetores de pesos similares.



MAPA DE KOHONEN

Função de
Vizinhança



$R = 2$

$R = 1$ —

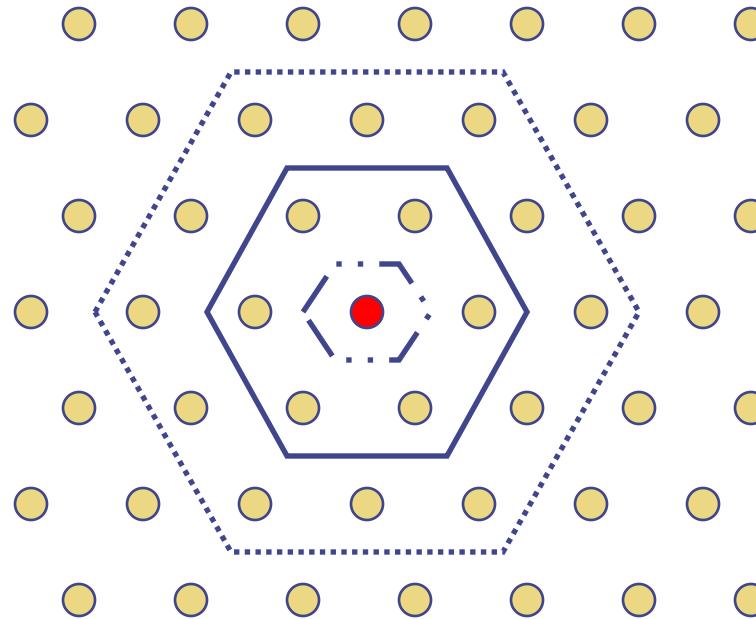
$R = 0$ - - -

Vizinhança de grade retangular



MAPA DE KOHONEN

Função de
Vizinhança



$R = 2$

$R = 1$ —

$R = 0$ - · -

Vizinhança de grade Hexagonal



MAPA DE KOHONEN

Analise qualitativa 1:

$$w_j(t+1) = w_j(t) + \eta(t)h_{j,i(x)}(t)(x(t) - w_j(t))$$

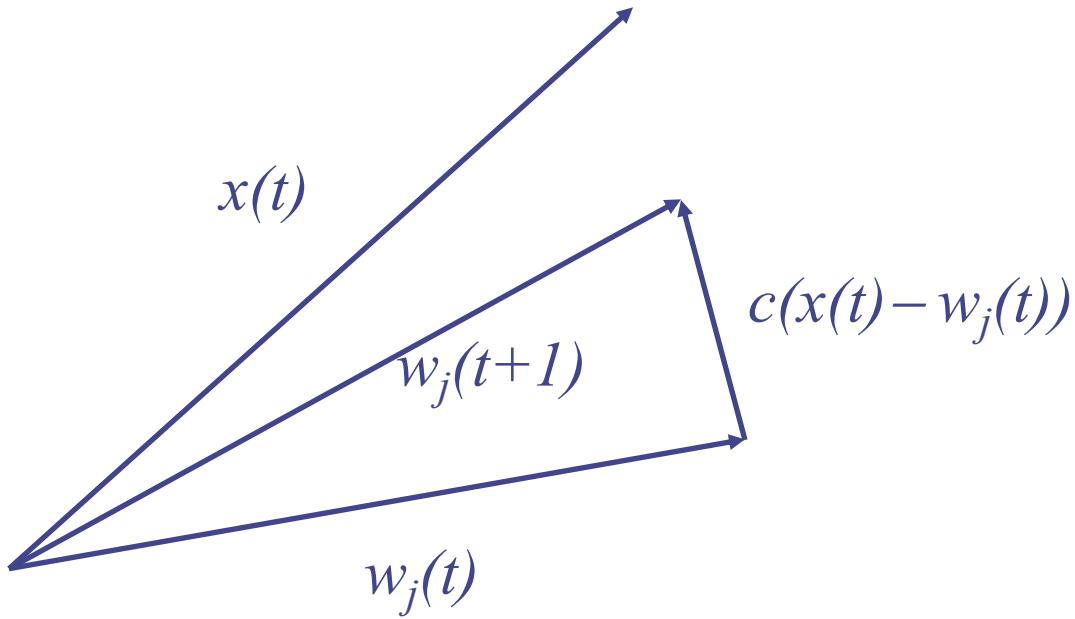
- Se $x_i(t) > w_{ij}(t)$, $w_{ij}(t+1) = w_j(t) + \text{valor positivo}$
Entao, valor do w_{ij} cresce.
- Se $x_i(t) < w_{ij}(t)$, $w_{ij}(t+1) = w_j(t) + \text{valor negativo}$
Entao, valor do w_{ij} diminue.
- Portanto, w_{ij} aproxima x_i



MAPA DE KOHONEN

Analise qualitativa 2:

$$w_j(t+1) = w_j(t) + \eta(t)h_{j,i(x)}(t)(x(t) - w_j(t))$$



MAPA DE KOHONEN - ALGORITMO

Passo 0: Inicialize os pesos w_{ij} , inicialize os parâmetros de vizinhança; inicialize os parâmetros de taxa de aprendizagem

Passo 1: Enquanto a condição de término é falsa, faça passos 2-8

Passo 2: Para cada vetor de entrada x , faça passos 3-5

Passo 3: Para cada j , calcula função discriminante;

Passo 4: Encontra a índice J tal que $D(J)$ é a mínima;

Passo 5: Para todas unidades j em uma vizinhança específica do J , e para todos i :

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + h_{jJ} \eta [x_i - w_{ij}(\text{old})]$$

Passo 6: Atualiza a taxa de aprendizagem;

Passo 7: Reduz região de vizinhança no tempo específico;

Passo 8: Testa a condição de término.



MAPA DE KOHONEN

Exemplo:

Usa o SOM para clusteriza 4 vetores:

$$(1, 1, 0, 0), (0, 0, 0, 1), (1, 0, 0, 0) \text{ e } (0, 0, 1, 1)$$

Suponhamos que o numero máximo de clusters que podem ser formados seja $m = 2$, a taxa de aprendizagem é definida como $\eta(0) = 0.6$

$$\eta(t+1) = 0.5 \eta(t)$$



MAPA DE KOHONEN

Passo 0: Raio inicial $R = 0$, taxa de aprendizagem inicial

$$\eta(0) = 0.6$$

Inicialize a matriz de pesos,

$$\begin{bmatrix} 0.2 & 0.8 \\ 0.6 & 0.4 \\ 0.5 & 0.7 \\ 0.9 & 0.3 \end{bmatrix}$$



MAPA DE KOHONEN

Passo 1: Começa o treinamento

Passo 2: Para o primeiro vetor $(1, 1, 0, 0)$, faça Passos 3-5

Passo 3: $D(1) = (0.2-1)^2 + (0.6-1)^2 + (0.5-0)^2 + (0.9-0)^2 = 1.86$

$$D(2) = (0.8-1)^2 + (0.4-1)^2 + (0.7-0)^2 + (0.3-0)^2 = 0.98$$

Passo 4: O vetor de entrada é mais próximo o nó 2.

Então, $J = 2$

Passo 5: Os pesos da unidade vencedor são atualizados:

$$w_{i2}(\text{new}) = w_{i2}(\text{old}) + 0.6[x_i - w_{i2}(\text{old})]$$

Isso resulta na matriz de pesos:

$$\begin{bmatrix} 0.2 & 0.92 \\ 0.6 & 0.76 \\ 0.5 & 0.28 \\ 0.9 & 0.12 \end{bmatrix}$$



MAPA DE KOHONEN

Passo 2: Para o segundo vetor $(0, 0, 0, 1)$, faça Passos 3-5

Passo 3: $D(1) = (0.2-0)^2 + (0.6-0)^2 + (0.5-0)^2 + (0.9-1)^2 = 0.66$

$$D(2) = (0.92-0)^2 + (0.76-0)^2 + (0.28-0)^2 + (0.12-1)^2 = 2.2768$$

Passo 4: O vetor de entrada é mais próximo a unidade 1.

Então, $J = 1$

Passo 5: Atualiza a primeira coluna da matriz de pesos:

$$\begin{bmatrix} 0.08 & 0.92 \\ 0.24 & 0.76 \\ 0.20 & 0.28 \\ 0.96 & 0.12 \end{bmatrix}$$



MAPA DE KOHONEN

Passo 2: Para o terceiro vetor $(1, 0, 0, 0)$, faça passos 3-5

Passo 3: $D(1) = (0.08-1)^2 + (0.24-0)^2 + (0.20-0)^2 + (0.96-0)^2 = 1.8656$

$$D(2) = (0.92-1)^2 + (0.76-0)^2 + (0.28-0)^2 + (0.12-0)^2 = 0.6768$$

Passo 4: O vetor de entrada é mais próximo a unidade 2.

Então, $J = 2$

Passo 5: Atualiza a segunda coluna da matriz de pesos:

$$\begin{bmatrix} 0.08 & 0.968 \\ 0.24 & 0.304 \\ 0.20 & 0.112 \\ 0.96 & 0.048 \end{bmatrix}$$



MAPA DE KOHONEN

Passo 2: Para o quarto vetor $(0, 0, 1, 1)$, faça Passos 3-5

Passo 3: $D(1) = (0.08-0)^2 + (0.24-0)^2 + (0.20-1)^2 + (0.96-1)^2 = 0.7056$

$$\begin{aligned} D(2) &= (0.968-0)^2 + (0.304-0)^2 + (0.112-1)^2 + (0.048-1)^2 \\ &= 2.724 \end{aligned}$$

Passo 4: O vetor de entrada é mais próximo a unidade 1.

Então, $J = 1$

Passo 5: Atualiza a primeira coluna da matriz de pesos:

$$\begin{bmatrix} 0.032 & 0.968 \\ 0.096 & 0.304 \\ 0.680 & 0.112 \\ 0.984 & 0.048 \end{bmatrix}$$



MAPA DE KOHONEN

Passo 6: Reduz a taxa de aprendizagem $\eta = 0.5 * (0.6) = 0.3$

Agora a equação para atualizar os pesos é

$$w_{ij} \text{ (new)} = w_{ij}(\text{old}) + 0.3[x_i - w_{ij}(\text{old})]$$

A matriz de pesos após a segunda época de treinamento é

$$\begin{bmatrix} 0.016 & 0.980 \\ 0.047 & 0.360 \\ 0.630 & 0.055 \\ 0.999 & 0.024 \end{bmatrix}$$

.....



MAPA DE KOHONEN

O resultado de 100 iterações (épocas) - a taxa de aprendizagem diminui de 0.6 para 0.01:

Iteração 0

$$\begin{bmatrix} 0.2 & 0.8 \\ 0.6 & 0.4 \\ 0.5 & 0.7 \\ 0.9 & 0.3 \end{bmatrix}$$

Iteração 1

$$\begin{bmatrix} 0.032 & 0.968 \\ 0.096 & 0.304 \\ 0.680 & 0.112 \\ 0.984 & 0.048 \end{bmatrix}$$

Iteração 2

$$\begin{bmatrix} 0.016 & 0.980 \\ 0.047 & 0.360 \\ 0.630 & 0.055 \\ 0.999 & 0.024 \end{bmatrix}$$



MAPA DE KOHONEN

Iteração 10

$$\begin{bmatrix} 1.5e-7 & 1.0000 \\ 4.6e-7 & 0.3700 \\ 0.6300 & 5.4e-7 \\ 1.0000 & 2.3e-7 \end{bmatrix}$$

Iteração 50

$$\begin{bmatrix} 1.9e-19 & 1.0000 \\ 5.7e-15 & 0.4700 \\ 0.5300 & 6.6e-15 \\ 1.0000 & 2.8e-15 \end{bmatrix}$$

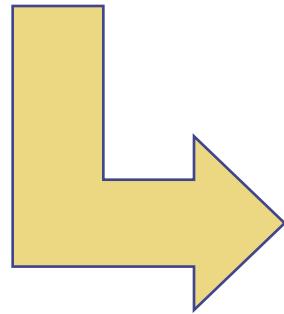
Iteração 100:

$$\begin{bmatrix} 6.7e-17 & 1.0000 \\ 2.0e-16 & 0.4900 \\ 0.5100 & 2.3e-16 \\ 1.0000 & 1.0e-16 \end{bmatrix}$$



MAPA DE KOHONEN

Iteração 100:

$$\begin{bmatrix} 6.7e-17 & 1.0000 \\ 2.0e-16 & 0.4900 \\ 0.5100 & 2.3e-16 \\ 1.0000 & 1.0e-16 \end{bmatrix}$$


$$\begin{bmatrix} 0 & 1.0 \\ 0 & 0.5 \\ 0.5 & 0 \\ 1.0 & 0 \end{bmatrix}$$

A primeira coluna representa a media de dois vetores que são colocados na cluster 1 e a segunda coluna representa a media de dois vetores que são colocados na cluster 2



MAPA DE KOHONEN

- De uma forma geral, as redes SOM podem ser vistas como uma ferramenta de visualização.
- Através deste modelo, um espaço n-dimensional pode ser representado (visualizado) em um reticulado de uma (1D) ou duas dimensões (2D).



MAPA DE KOHONEN – Medidas de Qualidade

▶ Quantization Error

$$QE(M) = \frac{1}{n} \sum_{i=1}^n \| \phi(x_i) - x_i \|$$

▶ Topographic Error

$$TE(M) = \frac{1}{n} \sum_{i=1}^n t(x_i)$$

$$t(x) = \begin{cases} 0 & \text{if } \mu(x) \text{ and } \mu'(x) \text{ are neighbors} \\ 1 & \text{otherwise} \end{cases}$$



MAPA DE KOHONEN – Medidas de Qualidade

► U-Matrix



MAPA DE KOHONEN – APLICAÇÃO - 1

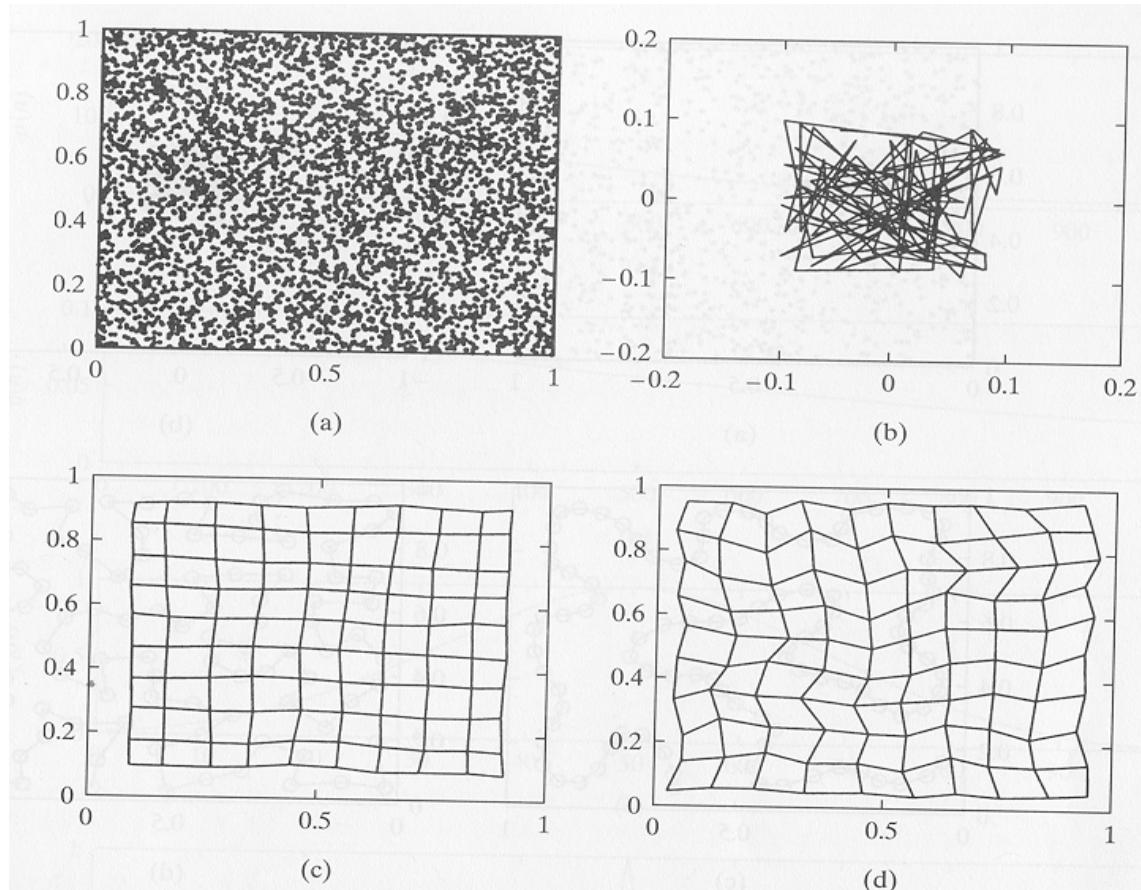


FIGURE 9.8 (a) Input data distribution. (b) Initial condition of the two-dimensional lattice. (c) Condition of the lattice at the end of the ordering phase. (d) Condition of the lattice at the end of the convergence phase.



MAPA DE KOHONEN – APLICAÇÃO - 1

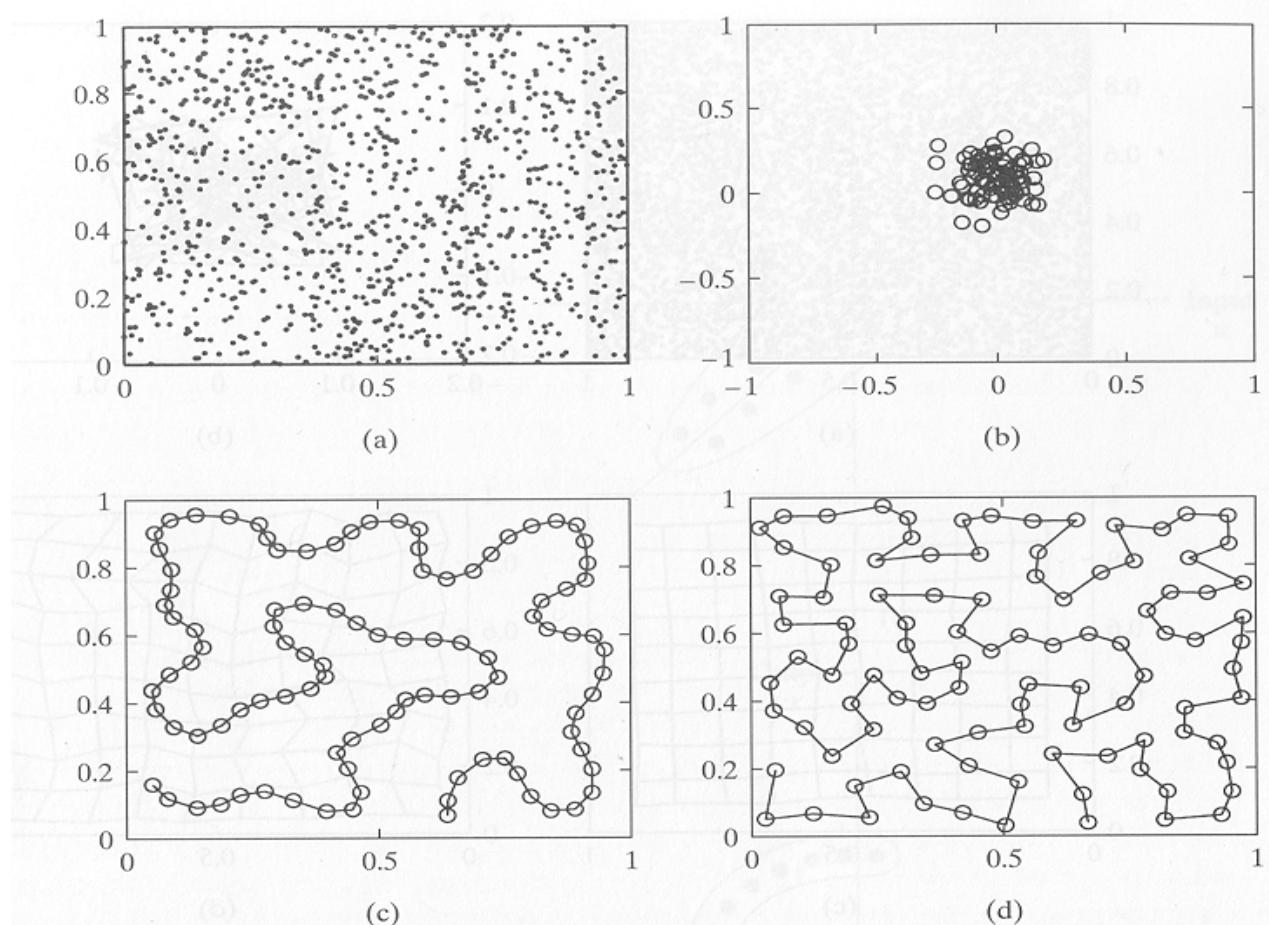


FIGURE 9.9 (a) Two-dimensional input data distribution. (b) Initial condition of the one-dimensional lattice. (c) Condition of the lattice at the end of the ordering phase. (d) Condition of the lattice at the end of the convergence phase.

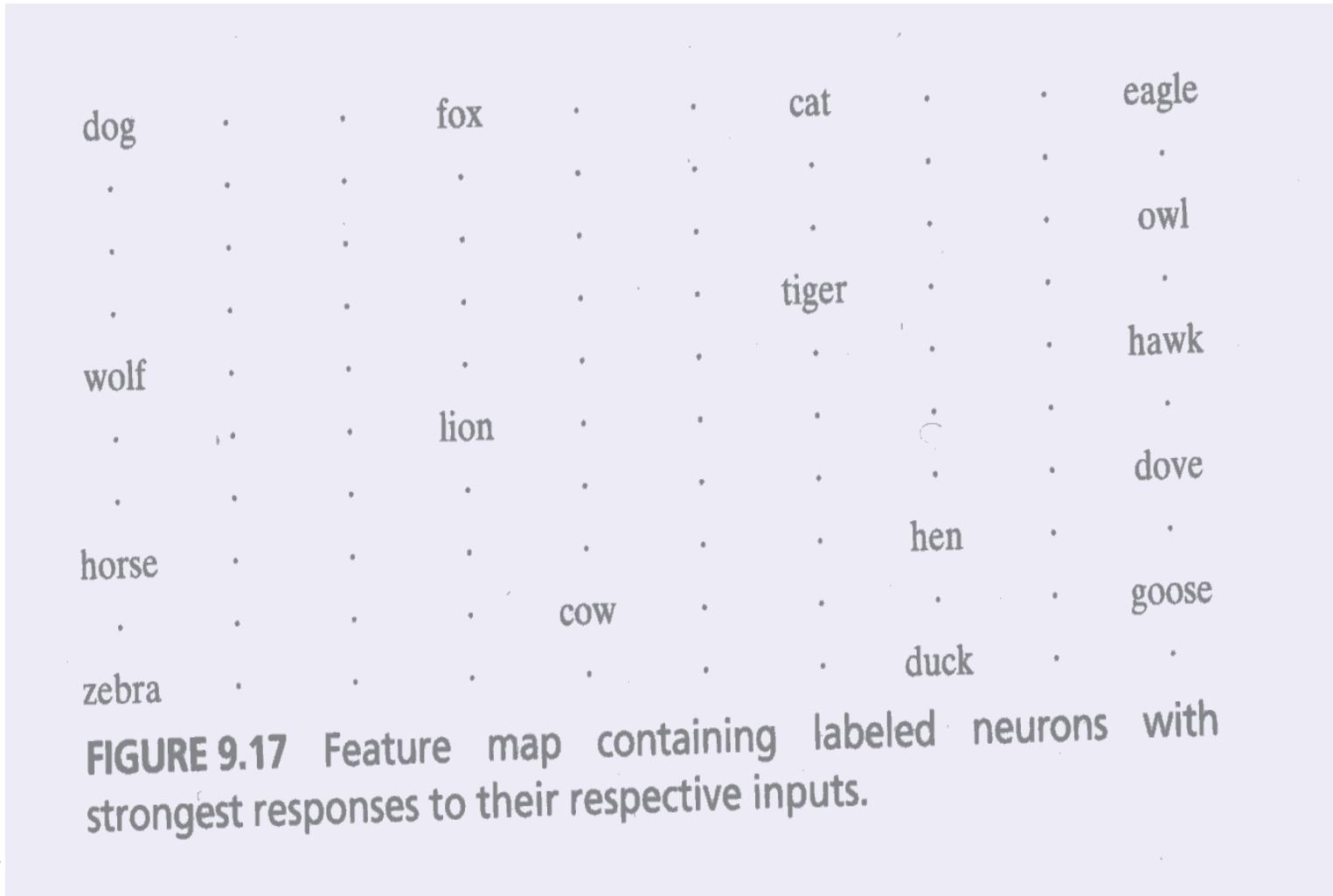
MAPA DE KOHONEN – APLICAÇÃO - 2

TABLE 9.3 Animal Names and Their Attributes

Animal	Dove	Hen	Duck	Goose	Owl	Hawk	Eagle	Fox	Dog	Wolf	Cat	Tiger	Lion	Horse	Zebra	Cow
is	small	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0
	medium	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
	big	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
has	2 legs	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
	4 legs	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
	hair	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
	hooves	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
	mane	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0
	feathers	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
likes to	hunt	0	0	0	0	1	1	1	0	1	1	1	1	0	0	0
	run	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0
	fly	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0
	swim	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0



MAPA DE KOHONEN – APLICAÇÃO - 2



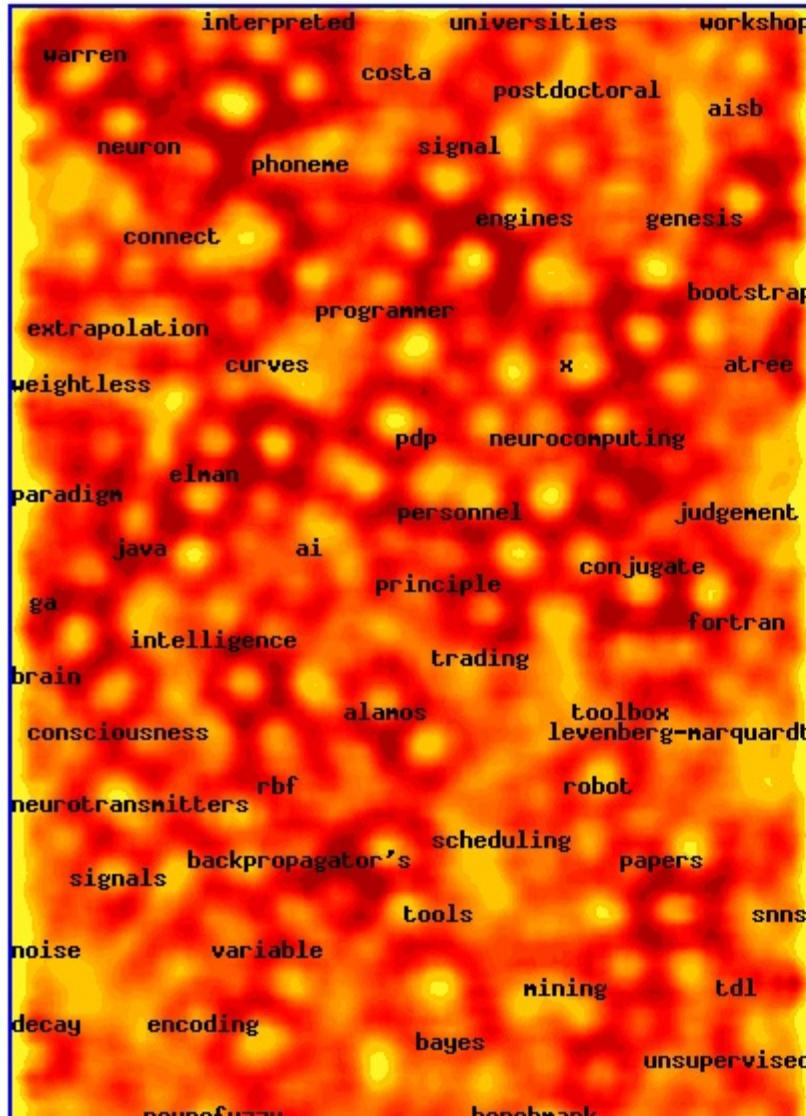
MAPA DE KOHONEN – APLICAÇÃO - 2

dog	dog	fox	fox	fox	cat	cat	cat	eagle	eagle
dog	dog	fox	fox	fox	cat	cat	cat	eagle	eagle
wolf	wolf	wolf	fox	cat	tiger	tiger	tiger	owl	owl
wolf	wolf	lion	lion	lion	tiger	tiger	tiger	hawk	hawk
wolf	wolf	lion	lion	lion	tiger	tiger	tiger	hawk	hawk
wolf	wolf	lion	lion	lion	owl	dove	hawk	dove	dove
horse	horse	lion	lion	lion	dove	hen	hen	dove	dove
horse	horse	zebra	cow	cow	cow	hen	hen	dove	dove
zebra	zebra	zebra	cow	cow	cow	hen	hen	duck	goose
zebra	zebra	zebra	cow	cow	cow	duck	duck	duck	goose

FIGURE 9.18 Semantic map obtained through the use of simulated electrode penetration mapping. The map is divided into three regions representing: birds, peaceful species, and hunters.

MAPA DE KOHONEN – APLICAÇÃO - 3

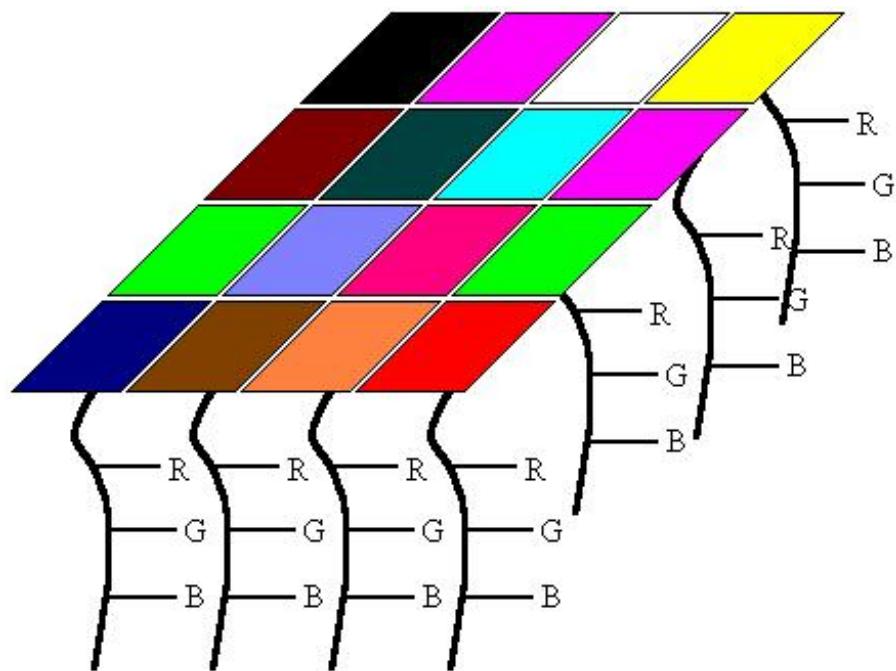
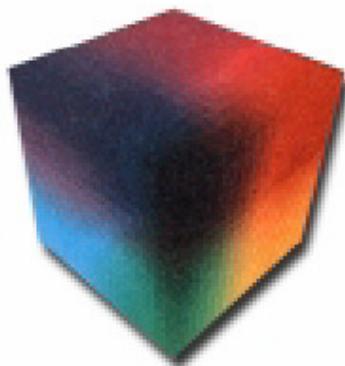
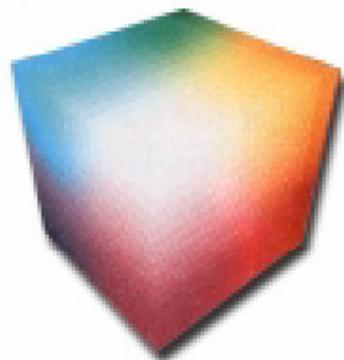
WEBSOM map - comp.ai.neural-nets



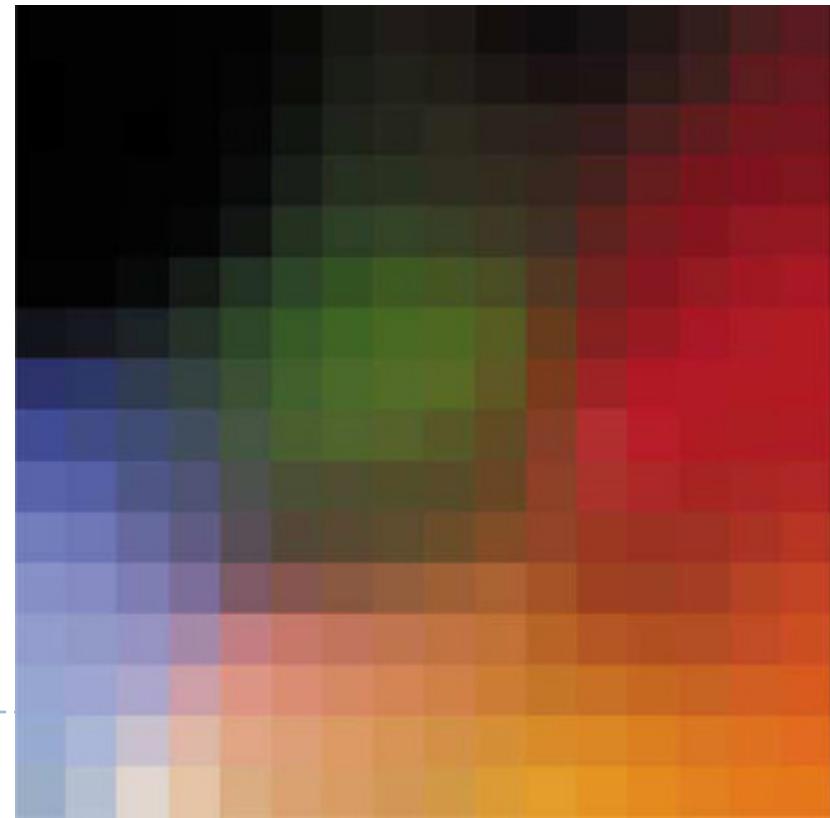
Instructions

Automatically generated labels and examples of titles in which the terms appear

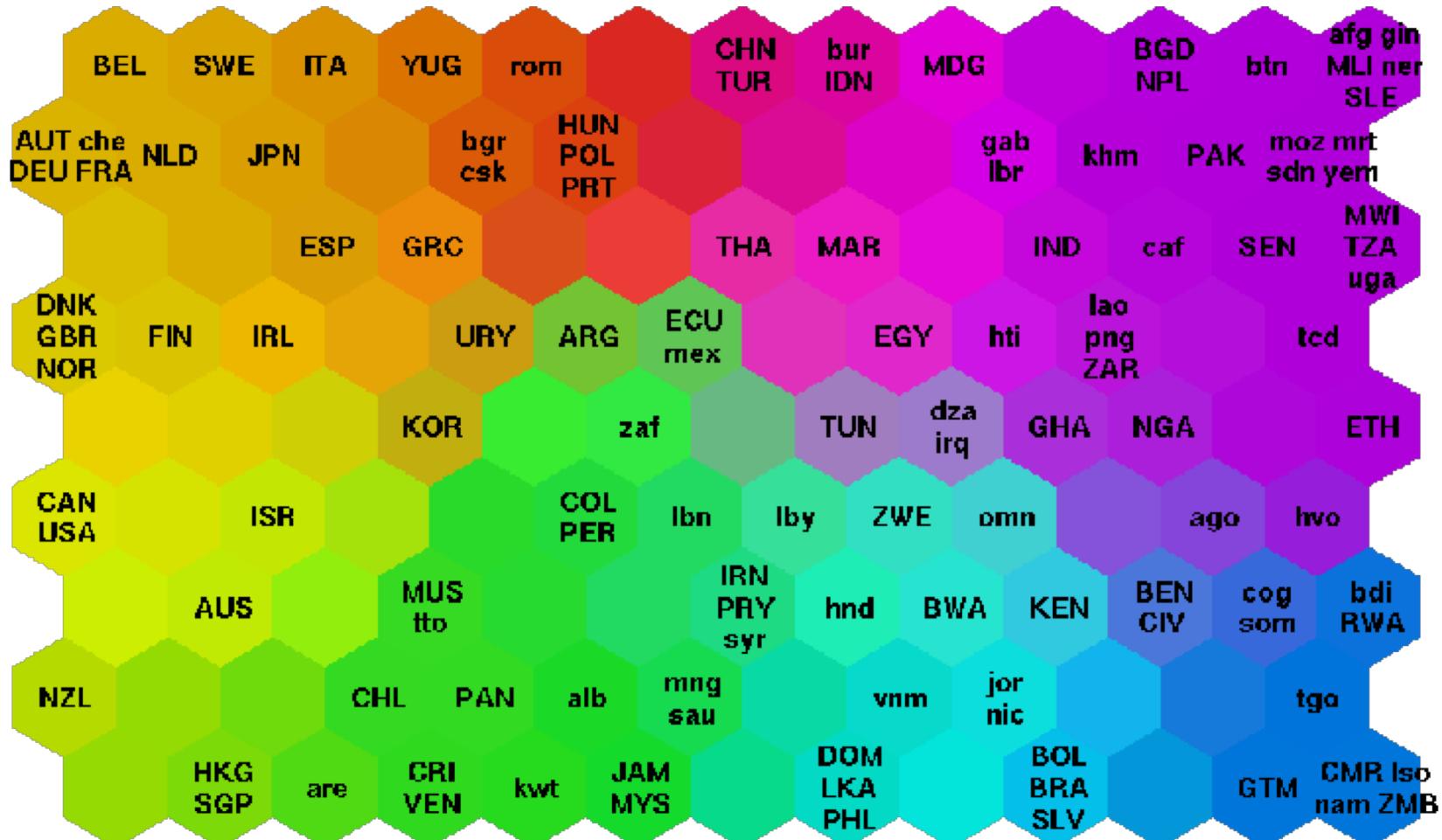
MAPA DE KOHONEN – APLICAÇÃO - 4



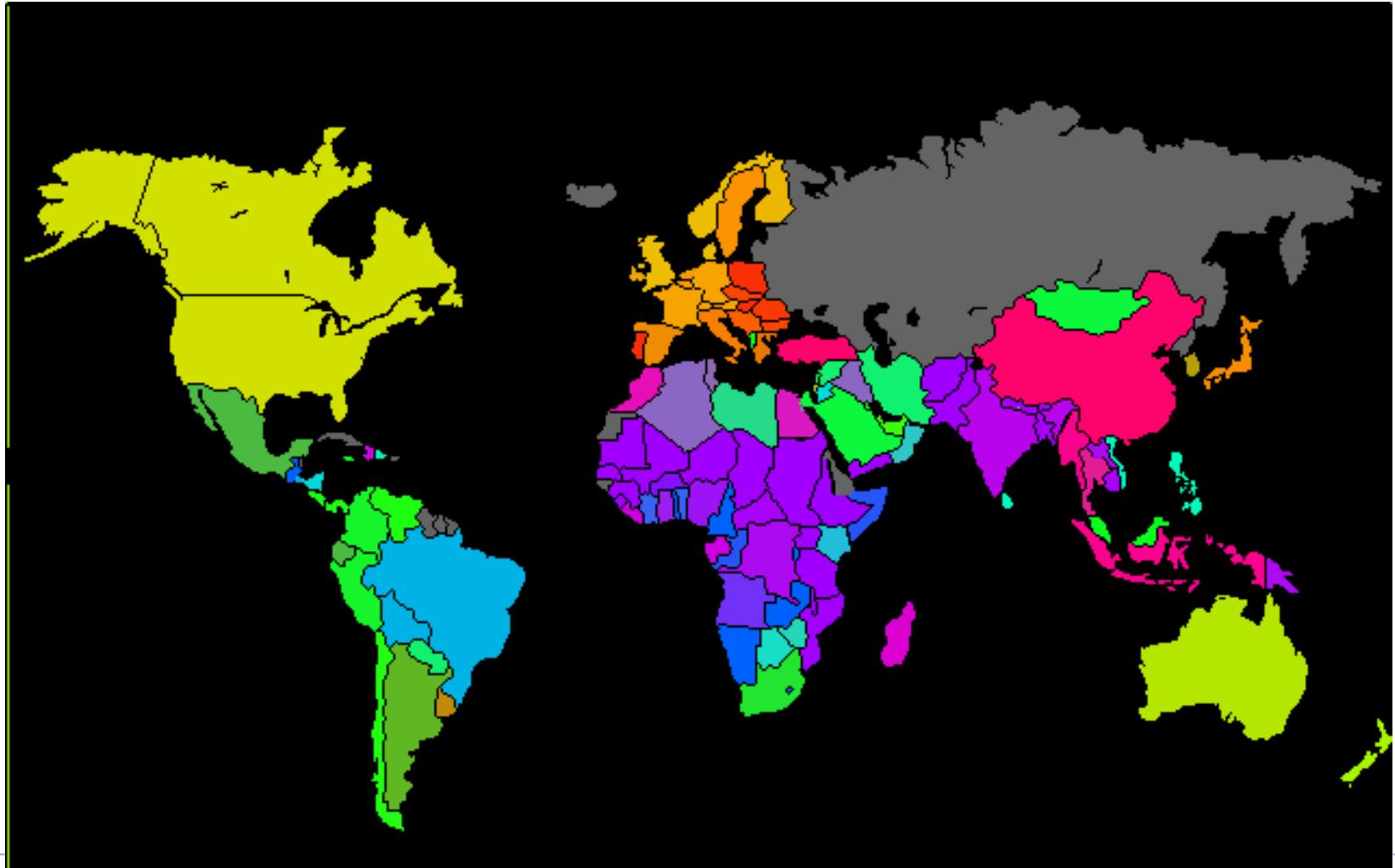
Samples



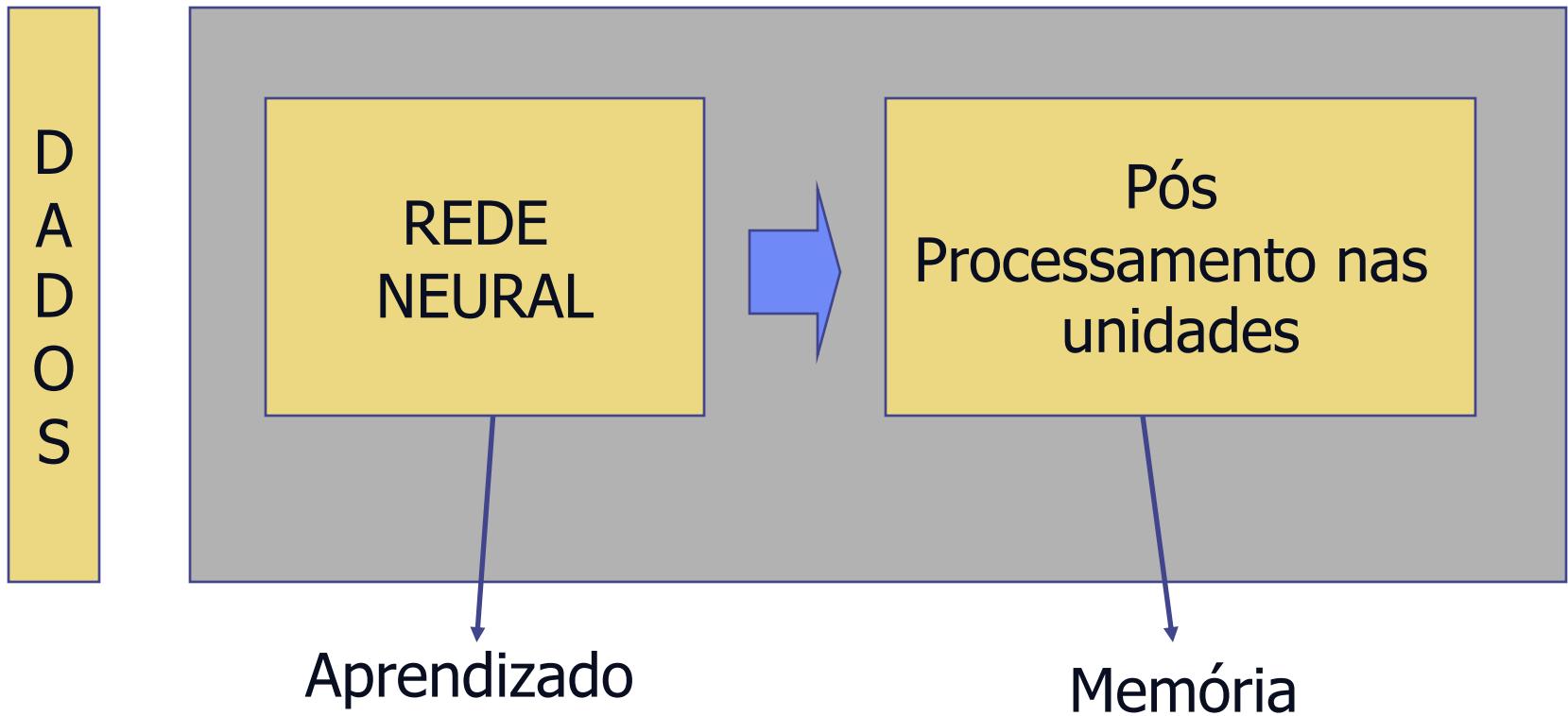
MAPA DE KOHONEN – APLICAÇÃO - 5



MAPA DE KOHONEN – APLICAÇÃO - 5

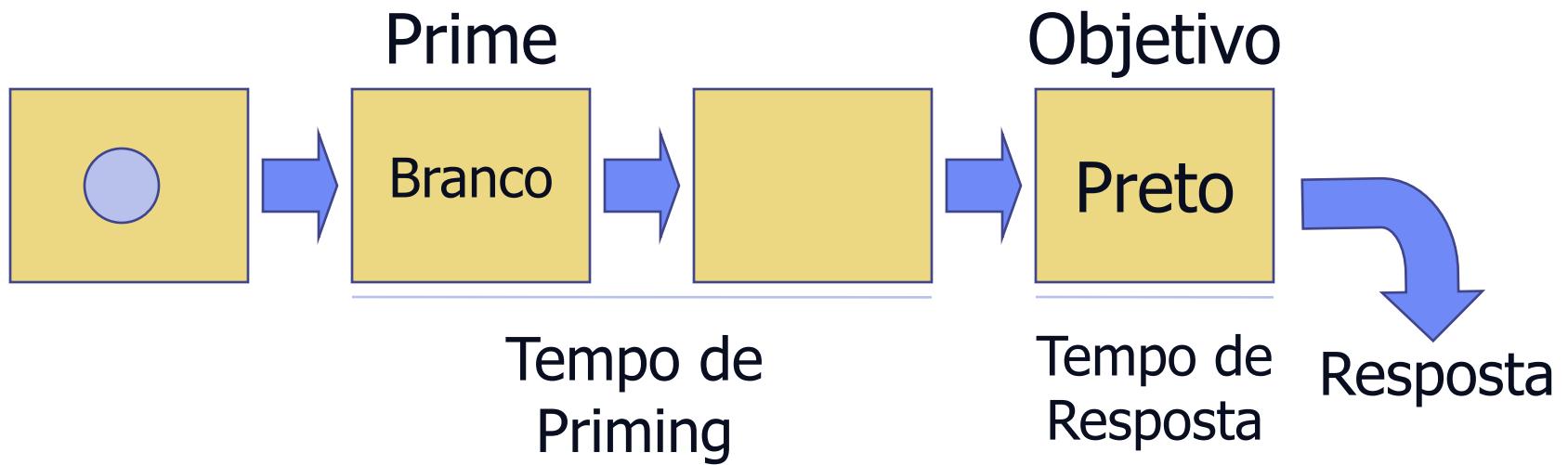


Redes Neurais Ampliadas



Técnica de Decisão Léxica

- ◆ Consiste em um experimento em que o sujeito decide se uma cadeia de caracteres é uma palavra ou não.

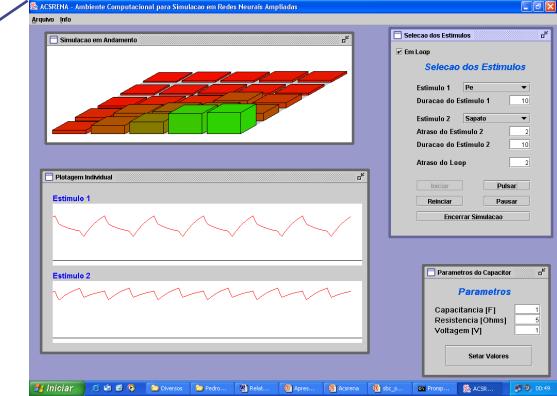
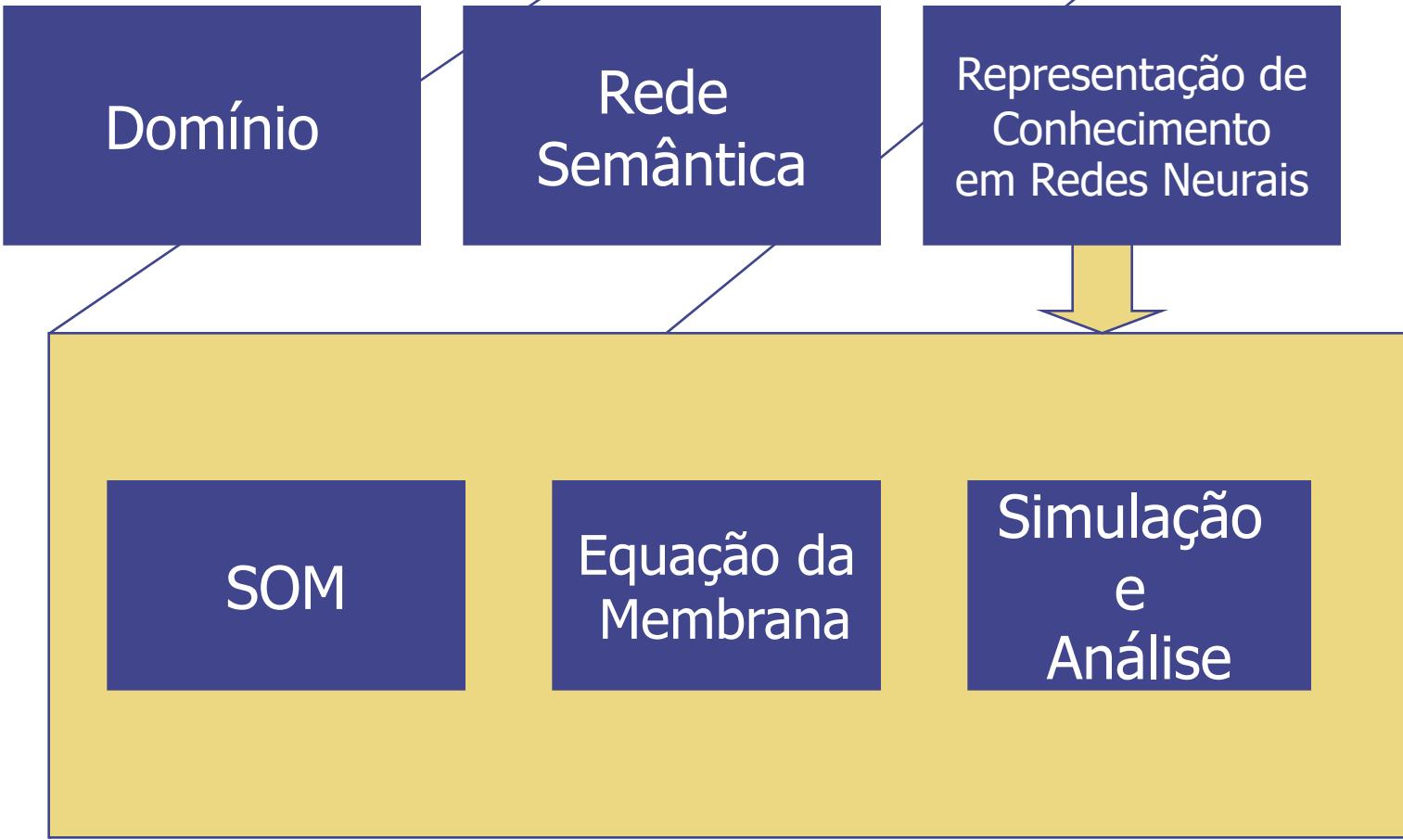


Priming Semântico

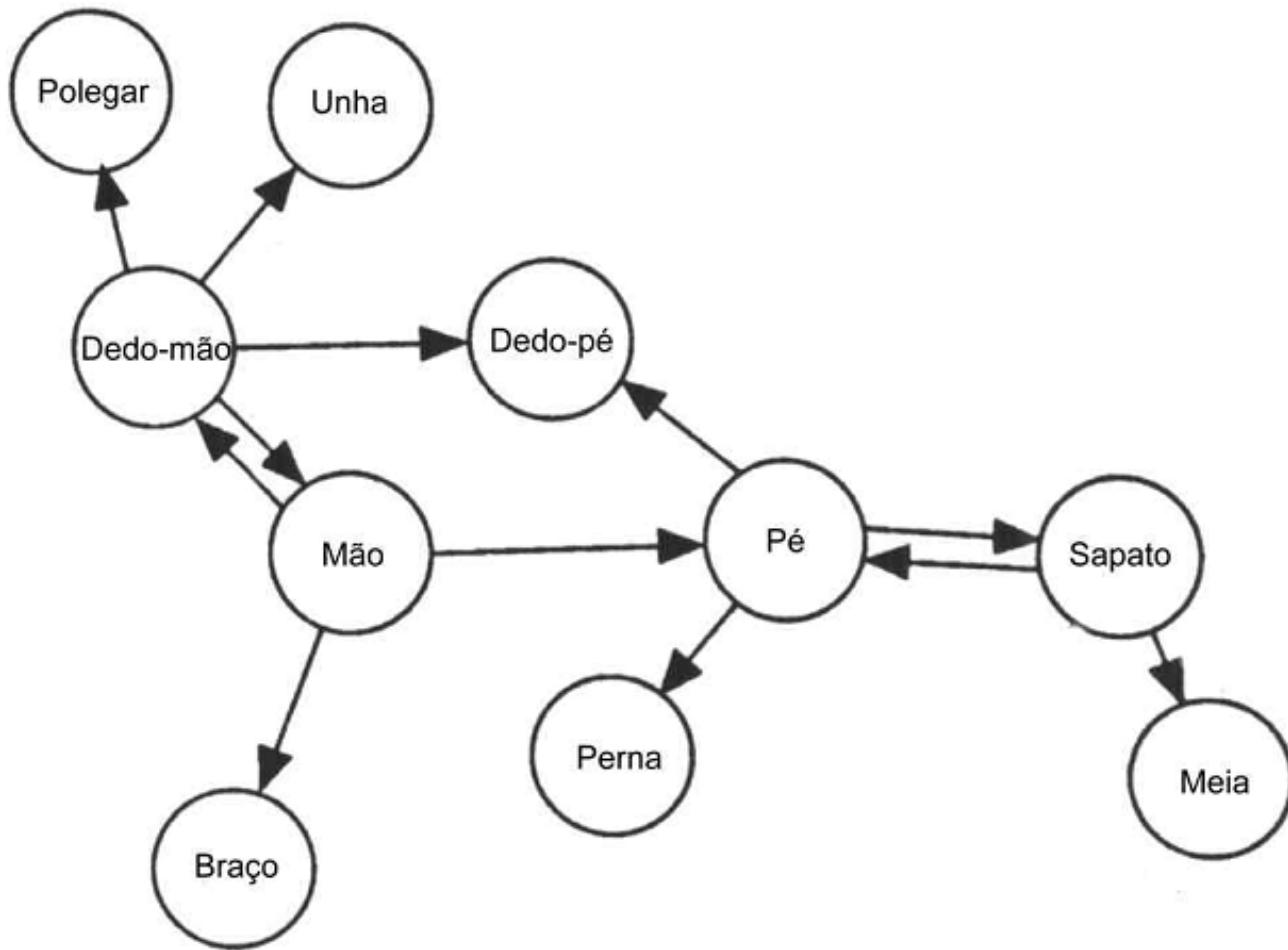
- ◆ O fenômeno de *priming* semântico é definido com segue: um sujeito reconhecerá uma palavra (objetivo) mais rapidamente se for precedida por uma palavra (*prime*) semanticamente relacionada do que por uma palavra que não apresente relação semântica com o objetivo.



Simulação



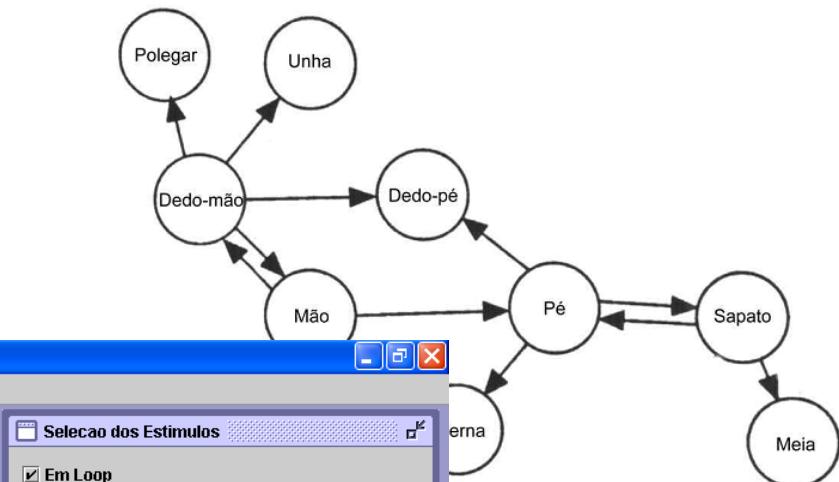
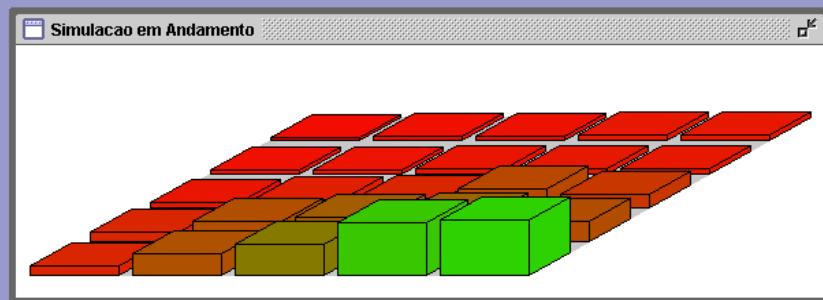
Simulação



Simulação

ACSRENA - Ambiente Computacional para Simulacao em Redes Neurais Ampliadas

Arquivo Info



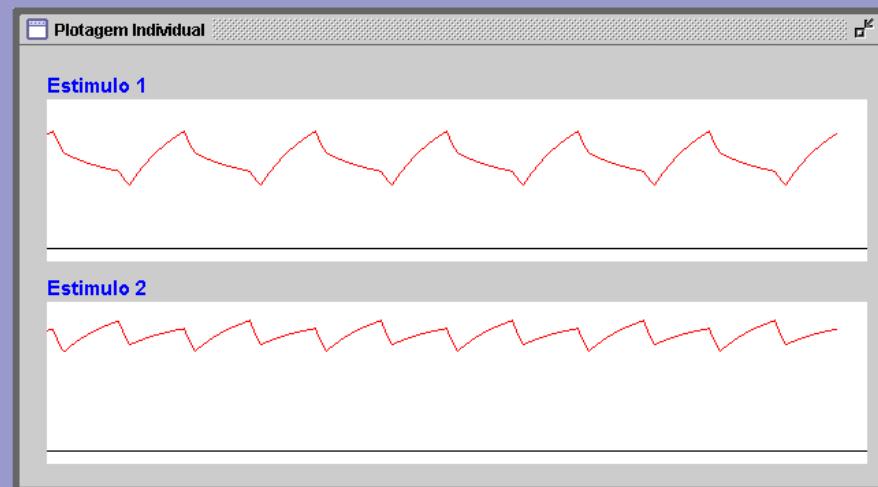
Selecao dos Estimulos

Em Loop

Selecao dos Estimulos

Estimulo 1	Pe	10
Duracao do Estimulo 1	10	
Estimulo 2	Sapato	2
Atraso do Estimulo 2	2	
Duracao do Estimulo 2	10	
Atraso do Loop	2	

Iniciar Pulsar
Reiniciar Pausar
Encerrar Simulacao



Parametros do Capacitor

Parametros

Capacitancia [F]	1
Resistencia [Ohms]	5
Voltagem [V]	1

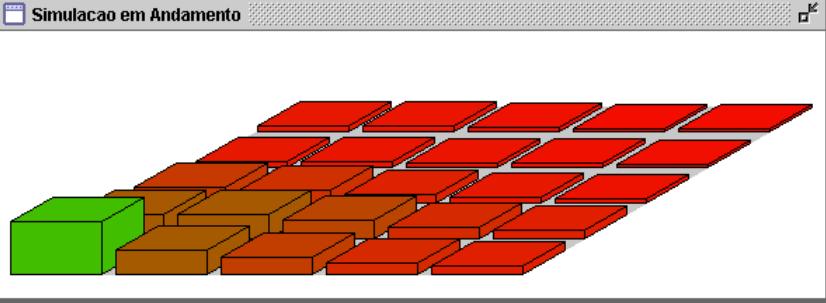
Setar Valores

Simulação

ACSRENA - Ambiente Computacional para Simulacão em Redes Neurais Ampliadas

Arquivo Info

Simulação em Andamento



Plotagem Individual

Estimulo 1



Estimulo 2



Seleção dos Estímulos

Em Loop

Seleção dos Estímulos

Estímulo 1	Sapato
Duração do Estímulo 1	5
Estímulo 2	Mao
Atraso do Estímulo 2	2
Duração do Estímulo 2	5
Atraso do Loop	5

Iniciar Pulsar

Reiniciar Pausar

Encerrar Simulação

Parâmetros do Capacitor

Parâmetros

Capacitância [F]	1
Resistência [Ohms]	15
Voltagem [V]	1

Setar Valores

Iniciar Diversos Roteiro1 - Micros... Apresentação Ba... Prompt de coman... ACSRENA - Ambie... 10:45

MAPA DE KOHONEN - Treinamento

Algumas Considerações

- O Treinamento nas redes SOM pode acontecer em duas fases:
 1. Fase de organização/ordenação topológica
 2. Fase de ajuste fino (convergência/especialização)
- Ajuste dinâmico da taxa de aprendizagem
- Ajuste dinâmico da função (raio) de vizinhança



Projeto 2

- ▶ Implementar(?) uma Rede SOM:
 - ▶ Realizar testes com 3 datasets selecionados do repositório da UCI (iris, etc...) para várias configuração da rede
 - ▶ Variar a função de vizinhança e decaimento
 - ▶ Variar a taxa de aprendizagem e decaimento
 - ▶ Gerar a U-Matrix



Bibliografia

1. Simon Haykin (2001) “Redes Neurais: Princípios e Prática”, 2 ed. Bookman
2. Kohonen, T. (2001). Self-Organizing Maps (3th Edition ed.). Information Science. Springer.

