



UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE COMPUTAÇÃO  
Departamento de Ciências de Computação

# SCC-5809 - Capítulo 8

## Auto-Organização

João Luís Garcia Rosa<sup>1</sup>

<sup>1</sup>SCC-ICMC-USP - joaoluis@icmc.usp.br

2011

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

## Alguns princípios básicos

- Aprendizado auto-organizado (não-supervisionado) consiste em repetidamente modificar os pesos sinápticos de uma RNA em resposta a padrões de ativação e de acordo com regras prescritas, até que uma configuração final apareça.
- O importante é saber como uma configuração útil pode ser desenvolvida da auto-organização.
- A resposta está na observação de Alan Turing [7]:

*Ordem global pode surgir de interações locais.*

- Essa observação aplica-se ao cérebro e às RNA.
- Muitas interações locais originalmente aleatórias entre neurônios vizinhos de uma rede podem combinar em estados de ordem global e levar a um comportamento coerente na forma de padrões espaciais ou ritmos temporais: essência da auto-organização.

# Alguns princípios básicos

- A organização da rede acontece em dois níveis diferentes que se interagem na forma de um *loop* de retro-alimentação:
  - 1 *Atividade*: certos padrões de atividade são produzidos por uma dada rede em resposta aos sinais de entrada.
  - 2 *Conectividade*: forças de conexão (pesos sinápticos) são modificados em resposta aos sinais neuronais nos padrões de atividade, devido à plasticidade sináptica.
- A retro-alimentação entre mudanças nos pesos sinápticos e mudanças nos padrões de atividade deve ser *positiva* para alcançar a auto-organização (ao invés da estabilização) da rede.

# Alguns princípios básicos

- Princípios da auto-organização:
  - 1 Modificações nos pesos sinápticos tendem a auto-amplificação.
  - 2 Limitação de recursos leva à competição entre sinapses e portanto à seleção de sinapses que crescem mais vigorosamente (isto é, as mais adequadas) às custas das outras.
  - 3 Modificações nos pesos sinápticos tendem a cooperar.
  - 4 Ordem e estrutura nos padrões de ativação representam informação redundante que é adquirida pela rede neural na forma de conhecimento, que é um pré-requisito necessário ao aprendizado auto-organizado.

# Alguns princípios básicos

- **Princípio 1:** Modificações nos pesos sinápticos tendem a auto-amplificação.
  - O processo de auto-amplificação é restrito pela condição de que modificações sinápticas devem ser baseadas em sinais disponíveis localmente, pré e pós-sinápticos.
  - Uma sinapse forte leva a coincidência de sinais pré e pós-sinápticos.
  - A sinapse é aumentada em força por essa coincidência.
  - Trata-se do postulado do aprendizado de Hebb.
  - Para estabilizar o sistema, deve haver uma competição pelos recursos limitados (número de entradas, recursos de energia).
  - Ou seja, um aumento em algumas sinapses leva a uma diminuição em outras.

# Alguns princípios básicos

- **Princípio 2:** Limitação de recursos leva à competição entre sinapses e portanto à seleção de sinapses que crescem mais vigorosamente (isto é, as mais adequadas) às custas das outras.
  - Esse princípio também é possível pela plasticidade sináptica.
  - Uma sinapse, por si só, não pode produzir eficientemente eventos favoráveis.
  - Para fazer isso, precisa-se de cooperação entre um conjunto de sinapses convergentes para um determinado neurônio e carregando sinais coincidentes fortes o suficiente para ativar esse neurônio.



# Alguns princípios básicos

- **Princípio 3:** Modificações nos pesos sinápticos tendem a cooperar.
  - A presença de uma sinapse vigorosa pode enriquecer a adaptação de outras sinapses, a despeito da competição geral na rede.
  - Essa forma de cooperação pode surgir devido à plasticidade sináptica, ou devido à estimulação simultânea de neurônio pré-sinápticos.
- Os três princípios apresentados até agora estão relacionados apenas à própria rede neural.
- Entretanto, para que o aprendizado auto-organizado realize uma função de processamento de informação útil, deve haver *redundância* nos padrões de ativação supridos à rede pelo ambiente.

# Alguns princípios básicos

- **Princípio 4:** Ordem e estrutura nos padrões de ativação representam informação redundante que é adquirida pela rede neural na forma de conhecimento, que é um pré-requisito necessário ao aprendizado auto-organizado.
  - Parte desse conhecimento pode ser obtida pela observação de parâmetros estatísticos tal como média, variância e matriz de correlação dos dados de entrada.
- Os princípios de 1 a 4 provêm a base neurobiológica para os algoritmos adaptativos para a análise de componentes principais (PCA) e mapas auto-organizáveis de Kohonen.
- Esses princípios são também incorporados em muitos outros modelos de auto-organização que são motivados pelas considerações neurobiológicas: *modelo de Linsker* do sistema visual do mamífero [5].

# Sumário

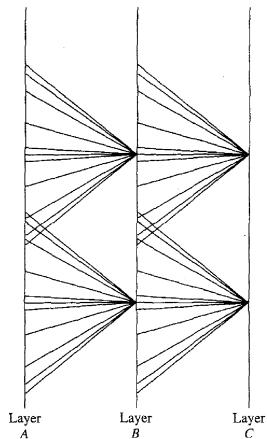
- 1 Auto-organização
  - Princípios
  - **Análise auto-organizada de características**
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

# Sistema visual

- O processamento de informação no sistema visual é realizado em estágios.
- Características simples como contraste ou orientação de borda são analisados nos primeiros estágios enquanto que características complexas mais elaboradas nos últimos.
- A figura 1 mostra uma estrutura de uma rede modular que lembra, grosso modo, o sistema visual.
- No modelo de Linsker, os neurônios da figura 1 são organizados em camadas de duas dimensões, com conexões locais *forward* de uma camada a outra.
- Cada neurônio recebe informação de um número limitado de neurônios localizados na região da camada anterior (*campo receptivo* desse neurônio).

# Sistema visual

Figure 1: *Layout da rede auto-adaptativa modular [1].*



# Sistema visual

- Os campos receptivos têm um papel fundamental no processo de desenvolvimento sináptico, porque permitem que os neurônios em uma camada respondam às *correlações espaciais* das atividades neuronais da camada anterior.
- Duas assunções de natureza estrutural são feitas:
  - 1 Uma vez escolhidas, as posições das conexões sinápticas são fixas para o processo de desenvolvimento neuronal inteiro.
  - 2 Cada neurônio age como um combinador linear.
- O modelo combina aspectos da modificação sináptica baseada em Hebb com aprendizado cooperativo e competitivo, de tal forma que as saídas da rede discriminem, de maneira ótima, um conjunto de entradas, com o aprendizado auto-organizado acontecendo numa base *camada-por-camada*.

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 **PCA**
  - **Análise de Componentes Principais**
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

# Projeção

- Seja  $\mathbf{X}$  um *vetor aleatório*  $m$ -dimensional que representa o ambiente de interesse.
- Assume-se que  $\mathbf{X}$  tem média zero:

$$E[\mathbf{X}] = \mathbf{0} \quad (1)$$

onde  $E$  é o operador de expectativa estatística.

- Se  $\mathbf{X}$  tem média não nula, subtrai-se essa média antes de analisar.
- Seja  $\mathbf{q}$  um *vetor unidade*, também de dimensão  $m$ , sobre o qual o vetor  $\mathbf{X}$  será *projetado*.
- Essa projeção é definida pelo produto interno dos vetores  $\mathbf{X}$  e  $\mathbf{q}$ :

$$A = \mathbf{X}^T \mathbf{q} = \mathbf{q}^T \mathbf{X} \quad (2)$$

sujeito à restrição

$$\|\mathbf{q}\| = (\mathbf{q}^T \mathbf{q})^{1/2} = 1 \quad (3)$$



# Matriz de correlação

- A projeção  $A$  é uma *variável aleatória* com média e variância relacionadas à estatística do vetor aleatório  $\mathbf{X}$ .
- Assumindo que  $\mathbf{X}$  tem média zero, segue que o valor médio da projeção  $A$  é zero também:

$$E[A] = \mathbf{q}^T E[\mathbf{X}] = 0 \quad (4)$$

- A variância de  $A$  é portanto a mesma do seu valor da média quadrática:

$$\sigma^2 = E[A^2] = E[(\mathbf{q}^T \mathbf{X})(\mathbf{X}^T \mathbf{q})] = \mathbf{q}^T E[\mathbf{X}\mathbf{X}^T] \mathbf{q} = \mathbf{q}^T \mathbf{R} \mathbf{q} \quad (5)$$

- A matriz  $m$ -por- $m$   $\mathbf{R}$  é a *matriz de correlação* do vetor aleatório  $\mathbf{X}$ , definido como a expectativa do produto externo do vetor  $\mathbf{X}$  com ele mesmo:

$$\mathbf{R} = E[\mathbf{X}\mathbf{X}^T] \quad (6)$$

# Sonda de variância

- Observe que a matriz de correlação é *simétrica*:

$$\mathbf{R}^T = \mathbf{R} \quad (7)$$

- Dessa propriedade segue que se  $\mathbf{a}$  e  $\mathbf{b}$  são vetores  $m$ -por-1 então

$$\mathbf{a}^T \mathbf{R} \mathbf{b} = \mathbf{b}^T \mathbf{R} \mathbf{a} \quad (8)$$

- Da equação 5 vê-se que a variância  $\sigma^2$  da projeção  $A$  é uma função do vetor unitário  $\mathbf{q}$ . Portanto

$$\psi(\mathbf{q}) = \sigma^2 = \mathbf{q}^T \mathbf{R} \mathbf{q} \quad (9)$$

$\psi(\mathbf{q})$  é uma *sonda de variância*.

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 **PCA**
  - Análise de Componentes Principais
  - **Autoestrutura**
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

# Autoestrutura do PCA

- Como achar os vetores unitários  $\mathbf{q}$  em que  $\psi(\mathbf{q})$  tem valores *extremos* ou *estacionários* (máximos ou mínimos locais), sujeito à restrição da norma euclidiana de  $\mathbf{q}$ .
- A solução a este problema está na autoestrutura da matriz de correlação  $\mathbf{R}$ .
- Se  $\mathbf{q}$  é um vetor unitário tal que a sonda de variância  $\psi(\mathbf{q})$  tem um valor extremo, então para qualquer perturbação pequena  $\delta\mathbf{q}$  do vetor unitário  $\mathbf{q}$ ,

$$\psi(\mathbf{q} + \delta\mathbf{q}) = \psi(\mathbf{q}) \quad (10)$$

- Da definição 9:

$$\psi(\mathbf{q} + \delta\mathbf{q}) = (\mathbf{q} + \delta\mathbf{q})^T \mathbf{R} (\mathbf{q} + \delta\mathbf{q}) = \mathbf{q}^T \mathbf{R} \mathbf{q} + 2(\delta\mathbf{q})^T \mathbf{R} \mathbf{q} + (\delta\mathbf{q})^T \mathbf{R} \delta\mathbf{q} \quad (11)$$

onde na segunda parte, usou-se a equação 8.

# Autoestrutura do PCA

- Ignorando o termo de segunda ordem  $(\delta\mathbf{q})^T \mathbf{R} \delta\mathbf{q}$  e usando a definição da equação 9:

$$\psi(\mathbf{q} + \delta\mathbf{q}) = \mathbf{q}^T \mathbf{R} \mathbf{q} + 2(\delta\mathbf{q})^T \mathbf{R} \mathbf{q} = \psi(\mathbf{q}) + 2(\delta\mathbf{q})^T \mathbf{R} \mathbf{q} \quad (12)$$

- Portanto, o uso da equação 10 em 12 implica que

$$(\delta\mathbf{q})^T \mathbf{R} \mathbf{q} = 0 \quad (13)$$

- Quaisquer perturbações  $\delta\mathbf{q}$  de  $\mathbf{q}$  não são admissíveis.
- Está-se restrito a usar apenas as perturbações para as quais a norma euclidiana do vetor perturbado  $\mathbf{q} + \delta\mathbf{q}$  permanece igual a unidade:

$$\| \mathbf{q} + \delta\mathbf{q} \|^2 = (\mathbf{q} + \delta\mathbf{q})^T (\mathbf{q} + \delta\mathbf{q}) = 1 \quad (14)$$

# Autoestrutura do PCA

- À luz da equação 3, é necessário que para uma primeira ordem em  $\delta\mathbf{q}$ ,

$$(\delta\mathbf{q})^T \mathbf{q} = 0 \quad (15)$$

Ou seja, as perturbações  $\delta\mathbf{q}$  devem ser ortogonais a  $\mathbf{q}$ , e portanto apenas uma mudança na direção de  $\mathbf{q}$  é permitida.

- Por convenção, os elementos do vetor unitário  $\mathbf{q}$  são adimensionais no sentido físico.
- Se se combinar as equações 13 e 15, deve-se introduzir um fator escalar  $\lambda$  na última equação com as mesmas dimensões das entradas da matriz de correlação  $\mathbf{R}$ :

$$(\delta\mathbf{q})^T \mathbf{R}\mathbf{q} - \lambda(\delta\mathbf{q})^T \mathbf{q} = (\delta\mathbf{q})^T (\mathbf{R}\mathbf{q} - \lambda\mathbf{q}) = 0 \quad (16)$$

# Autovalores e autovetores

- Para a condição da equação 16 valer, é necessário e suficiente ter

$$\mathbf{R}\mathbf{q} = \lambda\mathbf{q} \quad (17)$$

que é a equação que governa os vetores unitários  $\mathbf{q}$  para os quais a sonda variância  $\psi(\mathbf{q})$  tem valores extremos.

- A equação 17 é conhecida como o *problema do autovalor* da álgebra linear.
- O problema tem soluções não-triviais (isto é,  $\mathbf{q} \neq \mathbf{0}$ ) apenas para valores especiais de  $\lambda$ , chamados de *autovalores* da matriz de correlação  $\mathbf{R}$ .
- Os valores associados de  $\mathbf{q}$  são os *autovetores*.
- A matriz de correlação é caracterizada por autovalores reais não negativos.
- Os autovetores associados são únicos, assumindo que os autovalores são distintos.

## Exemplo: autovalores e autovetores [2]

- Seja a matriz  $\mathbf{A}$

$$\mathbf{A} = \begin{bmatrix} 2 & 5 \\ 3 & -4 \end{bmatrix} \quad (18)$$

- O seu polinômio característico é

$$p(\lambda) = \det(\mathbf{A} - \lambda \mathbf{I}) = \begin{vmatrix} 2 - \lambda & 5 \\ 3 & -4 - \lambda \end{vmatrix} = (2 - \lambda)(-4 - \lambda) - 15 = \lambda^2 + 2\lambda - 23 \quad (19)$$

cujas raízes são  $\lambda_1 = 3,8990$  e  $\lambda_2 = -5,8990$ .

- Para determinar os autovetores, utiliza-se a equação 17 para cada autovalor  $\lambda_i$ , na forma  $(\mathbf{A} - \lambda_i \mathbf{I})x_i = 0$ .
- Para o autovalor  $\lambda_2 = -5,8990$ , tem-se

$$(\mathbf{A} - -5,8990 \mathbf{I})x_2 = 0 \quad (20)$$

$$\begin{bmatrix} 7,8990 & 5 \\ 3 & 1,8990 \end{bmatrix} \begin{bmatrix} (x_2)_1 \\ (x_2)_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (21)$$

$$x_2 = \begin{bmatrix} k \\ -1,5798k \end{bmatrix}, \quad k \neq 0 \quad (22)$$



# Autoestrutura do PCA

- Sejam os autovalores da matriz  $m$ -por- $m$   $\mathbf{R}$  denotados por  $\lambda_1, \lambda_2, \dots, \lambda_m$  e os autovetores associados por  $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m$ , respectivamente.
- Pode-se escrever

$$\mathbf{R}\mathbf{q}_j = \lambda_j\mathbf{q}_j \quad j = 1, 2, \dots, m \quad (23)$$

- Sejam os autovalores correspondentes arranjados em ordem decrescente:

$$\lambda_1 > \lambda_2 > \dots > \lambda_j > \dots > \lambda_m \quad (24)$$

tal que  $\lambda_1 = \lambda_{max}$ .

- Sejam os autovetores associados usados para construir uma matriz  $m$ -por- $m$ :

$$\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_j, \dots, \mathbf{q}_m] \quad (25)$$

# Autoestrutura do PCA

- Pode-se combinar o conjunto de  $m$  equações representadas em 23 numa única equação

$$\mathbf{R}\mathbf{Q} = \mathbf{Q}\Lambda \quad (26)$$

onde  $\Lambda$  é uma matriz diagonal definida pelos autovalores da matriz  $\mathbf{R}$ :

$$\Lambda = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_j, \dots, \lambda_m] \quad (27)$$

- A matriz  $\mathbf{Q}$  é uma *matriz ortogonal (unitária)* no sentido de que seus vetores colunas (isto é os autovetores da matriz  $\mathbf{R}$ ) satisfazem as *condições de ortonormalidade*:

$$\mathbf{q}_i^T \mathbf{q}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \quad (28)$$

A equação 28 requer autovalores distintos.

# Autoestrutura do PCA

- Equivalentemente, pode-se escrever  $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$ , de onde deduz-se que a inversa da matriz  $\mathbf{Q}$  é a mesma que sua transposta:

$$\mathbf{Q}^T = \mathbf{Q}^{-1} \quad (29)$$

- Pode-se portanto re-escrever a equação 26 na forma *transformação de similaridade ortogonal*:

$$\mathbf{Q}^T \mathbf{R} \mathbf{Q} = \Lambda \quad (30)$$

ou na forma expandida

$$\mathbf{q}_j^T \mathbf{R} \mathbf{q}_k = \begin{cases} \lambda_j, & k = j \\ 0, & k \neq j \end{cases} \quad (31)$$

- A transformação da similaridade (unitária) ortogonal da equação 30 transforma a matriz de correlação  $\mathbf{R}$  em uma matriz diagonal de autovalores.

# Autoestrutura do PCA

- A matriz de correlação  $\mathbf{R}$  pode ser expressa em termos de seus autovalores e autovetores:

$$\mathbf{R} = \sum_{i=1}^m \lambda_i \mathbf{q}_i \mathbf{q}_i^T \quad (32)$$

referido como *teorema espectral*. O produto externo  $\mathbf{q}_i \mathbf{q}_i^T$  é de *rank* 1 para todo  $i$ .

- *Rank* de uma matriz é o número de linhas (ou colunas) linearmente independentes [8].
- Considere, por exemplo, a matriz  $\mathbf{A}$  4 por 4:

$$\mathbf{A} = \begin{bmatrix} 2 & 4 & 1 & 3 \\ -1 & -2 & 1 & 0 \\ 0 & 0 & 2 & 2 \\ 3 & 6 & 2 & 5 \end{bmatrix} \quad (33)$$

- Veja que a segunda coluna é duas vezes a primeira e que a quarta é igual a soma das primeira e terceira.
- A primeira e a terceira colunas são linearmente independentes, ou seja, o *rank* de  $\mathbf{A}$  é dois.

# Autoestrutura do PCA

- As equações 30 e 31 são duas representações equivalentes da *autodecomposição* de  $\mathbf{R}$ .
- PCA e autodecomposição da matriz  $\mathbf{R}$  são basicamente a mesma (o problema é visto de formas diferentes).
- Essa equivalência segue das equações 9 e 32 onde se vê que as sondas de variância e os autovalores são de fato iguais:

$$\psi(\mathbf{q}_j) = \lambda_j, \quad j = 1, 2, \dots, m \quad (34)$$

- Resumo da autoestrutura do PCA:
  - Os autovetores da matriz de correlação  $\mathbf{R}$  pertencentes ao vetor aleatório de média zero  $\mathbf{X}$  definem os vetores unitários  $\mathbf{q}_j$ , representando as direções principais ao longo das quais as sondas de variância  $\psi(\mathbf{q}_j)$  têm seus valores extremos.
  - Os autovalores associados definem os valores extremos das sondas de variância  $\psi(\mathbf{q}_j)$ .

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 **PCA**
  - Análise de Componentes Principais
  - Autoestrutura
  - **Dados**
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - Simulação

# Componentes Principais

- Seja o *vetor de dados*  $\mathbf{x}$  que denota a realização do vetor aleatório  $\mathbf{X}$ .
- Com  $m$  soluções possíveis para o vetor unitário  $\mathbf{q}$ , há  $m$  projeções possíveis do vetor de dados  $\mathbf{x}$ .
- Especificamente, da equação 2 nota-se que

$$a_j = \mathbf{q}_j^T \mathbf{x} = \mathbf{x}^T \mathbf{q}_j, \quad j = 1, 2, \dots, m \quad (35)$$

onde  $a_j$  são as projeções de  $\mathbf{x}$  nas direções principais representadas pelos vetores unitários  $\mathbf{q}_j$ .

- $a_j$  são chamados de *componentes principais* e têm as mesmas dimensões físicas do vetor de dados  $\mathbf{x}$ .
- A fórmula da equação 35 pode ser vista como uma *análise*.

# Representações de Dados

- Para reconstruir o vetor de dados original  $\mathbf{x}$  exatamente das projeções  $a_j$ :
  - Combine o conjunto de projeções  $\{a_j \mid j = 1, 2, \dots, m\}$  em um único vetor:

$$\mathbf{a} = [a_1, a_2, \dots, a_m]^T = [\mathbf{x}^T \mathbf{q}_1, \mathbf{x}^T \mathbf{q}_2, \dots, \mathbf{x}^T \mathbf{q}_m]^T = \mathbf{Q}^T \mathbf{x} \quad (36)$$

- Multiplique ambos os lados da equação 36 pela matriz  $\mathbf{Q}$  e então use a relação  $\mathbf{Q}^T = \mathbf{Q}^{-1}$ .
- O vetor de dados original  $\mathbf{x}$  pode ser reconstruído como

$$\mathbf{x} = \mathbf{Q}\mathbf{a} = \sum_{j=1}^m a_j \mathbf{q}_j \quad (37)$$

que pode ser vista como a fórmula para a *síntese*.

- Dessa forma, os vetores unitários  $\mathbf{q}_j$  representam uma *base* para o espaço de dados.
- A equação 37 é uma transformação de coordenadas, onde um ponto  $\mathbf{x}$  no espaço de dados é transformado em um ponto correspondente  $\mathbf{a}$  no espaço de características.



# Redução da dimensionalidade

- Pode-se reduzir o número de características necessárias para a representação de dados efetivos descartando as combinações lineares na equação 37 que têm pequenas variâncias, mantendo apenas os termos que têm grandes variâncias.
- Sejam  $\lambda_1, \lambda_2, \dots, \lambda_\ell$  os maiores autovalores da matriz de correlação  $\mathbf{R}$ .
- Pode-se aproximar o vetor de dados  $\mathbf{x}$  *truncando* a expansão da equação 37 depois de  $\ell$  termos:

$$\hat{\mathbf{x}} = \sum_{j=1}^{\ell} a_j \mathbf{q}_j = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_\ell] \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_\ell \end{bmatrix}, \quad \ell \leq m \quad (38)$$

# Redução da dimensionalidade

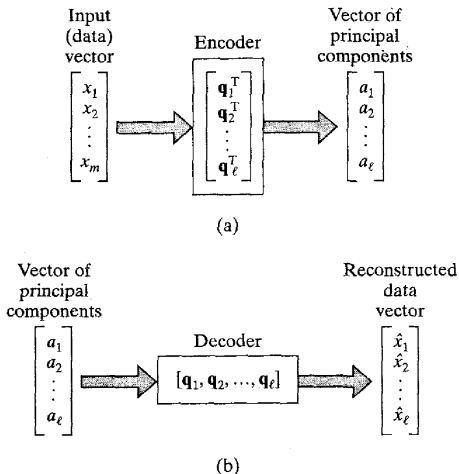
- Dado o vetor de dados original  $\mathbf{x}$ , pode-se usar a equação 35 para computar o conjunto de componentes principais mantidos na equação 38:

$$\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_\ell \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_\ell^T \end{bmatrix} \mathbf{x}, \quad \ell \leq m \quad (39)$$

- A projeção linear da equação 39 de  $\mathbb{R}^m$  para  $\mathbb{R}^\ell$  (isto é, o mapeamento do espaço de dados para o espaço de características) representa um *codificador* para a representação aproximada do vetor de dados  $\mathbf{x}$  como ilustrado na figura 2a.

# Redução da dimensionalidade

Figure 2: As duas fases do PCA: (a) codificação e (b) decodificação [1].



# Redução da dimensionalidade

- A projeção linear da equação 38 de  $\mathbb{R}^\ell$  para  $\mathbb{R}^m$  (isto é, o mapeamento do espaço de características para o espaço de dados) representa um *decodificador* para a reconstrução aproximada do vetor de dados original  $\mathbf{x}$  como ilustrado na figura 2b.
- Note que os autovalores *dominantes* (isto é, maiores)  $\lambda_1, \lambda_2, \dots, \lambda_\ell$  não participam da computação descrita nas equações 38 e 39; eles apenas determinam o número de componentes principais usados para codificação e decodificação, respectivamente.
- O *vetor de erro de aproximação*  $\mathbf{e}$  é igual a diferença entre o vetor de dados original  $\mathbf{x}$  e o vetor de dados de aproximação  $\hat{\mathbf{x}}$ :

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} \quad (40)$$

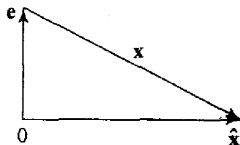
# Redução da dimensionalidade

- Substituindo as equações 37 e 38 na 40:

$$\mathbf{e} = \sum_{j=\ell+1}^m a_j \mathbf{q}_j \quad (41)$$

- O *vetor erro*  $\mathbf{e}$  é ortogonal ao vetor de dados de aproximação  $\hat{\mathbf{x}}$ , como ilustrado na figura 3.

**Figure 3:** Ilustração do relacionamento entre o vetor  $\mathbf{x}$ , seu vetor reconstruído  $\hat{\mathbf{x}}$  e o vetor de erro  $\mathbf{e}$  [1].



# Redução da dimensionalidade

- Em outras palavras, o produto interno dos vetores  $\hat{\mathbf{x}}$  e  $\mathbf{e}$  é zero.
- Essa propriedade é mostrada usando as equações 38 e 41 da seguinte forma:

$$\mathbf{e}^T \hat{\mathbf{x}} = \sum_{i=\ell+1}^m a_i \mathbf{q}_i^T \sum_{j=1}^{\ell} a_j \mathbf{q}_j = \sum_{i=\ell+1}^m \sum_{j=1}^{\ell} a_i a_j \mathbf{q}_i^T \mathbf{q}_j = 0 \quad (42)$$

- A equação 42 é conhecida como *princípio da ortogonalidade*.
- A variância total dos  $m$  componentes do vetor de dados  $\mathbf{x}$  é, via equação 9 e a primeira parte da equação 31,

$$\sum_{j=1}^{\ell} \sigma_j^2 = \sum_{j=1}^{\ell} \lambda_j \quad (43)$$

# Redução da dimensionalidade

- A variância total dos  $(\ell - m)$  elementos no vetor de erro de aproximação  $\mathbf{x} - \hat{\mathbf{x}}$  é

$$\sum_{j=\ell+1}^m \sigma_j^2 = \sum_{j=\ell+1}^m \lambda_j \quad (44)$$

- Os autovalores  $\lambda_{\ell+1}, \dots, \lambda_m$  são os *menores*  $(m - \ell)$  autovalores de  $\mathbf{R}$ : correspondem aos termos descartados da expansão da equação 38 usados para construir o vetor de aproximação  $\hat{\mathbf{x}}$ .
- Quanto mais próximos de zero esses autovalores, mais efetiva é a redução de dimensionalidade (resultante da aplicação da PCA ao vetor de dados  $\mathbf{x}$ ) na preservação do conteúdo da informação dos dados de entrada originais.

# Redução da dimensionalidade

- Ou seja, para realizar a redução de dimensionalidade nos dados de entrada (*decomposição de subespaço*):
  - 1 computa-se os autovalores e autovetores da matriz de correlação do vetor de dados de entrada e então
  - 2 projeta-se os dados ortogonalmente no subespaço criado pelos autovetores pertencentes aos autovalores dominantes.

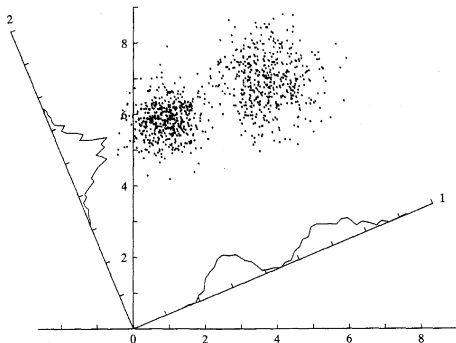


# Redução da dimensionalidade

- Exemplo: Conjunto de dados bi-dimensionais.
  - Assume-se (figura 4) que ambos os eixos de características são aproximadamente da mesma escala.
  - Os eixos horizontal e vertical representam as coordenadas naturais do conjunto de dados.
  - Os eixos rotacionados 1 e 2 resultam da aplicação da PCA a este conjunto de dados.
  - Vê-se que a projeção dos dados no eixo 1 captura a característica saliente dos dados, o fato de ser bimodal (há dois *clusters* nessa estrutura).
  - A variância das projeções dos pontos de dados no eixo 1 é maior que qualquer outro eixo de projeção.
  - Por outro lado, a natureza bimodal inerente do conjunto de dados é completamente obscurecida quando é projetada no eixo ortogonal 2.

# Redução da dimensionalidade

**Figure 4:** Uma nuvem de pontos de dados é mostrada em duas dimensões, e as densidades formadas pela projeção desta nuvem em cada um dos dois eixos 1 e 2 são indicadas. A projeção no eixo 1 tem máxima variância e claramente mostra o caráter bimodal ou agrupado dos dados [1].



# Sumário

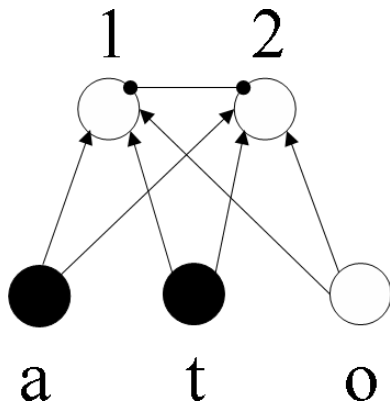
- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - **SOM**
  - Mapeamento
  - Formação do mapa
  - Simulação

# Introdução

- Considera-se agora uma classe especial de RNA: os mapas auto-organizáveis.
- São baseados no *aprendizado competitivo*: os neurônios de saída competem entre si para serem ativados: apenas *um* neurônio, ou um neurônio por grupo, é ativado por vez.
- O neurônio que ganha a competição é chamado de neurônio *vencedor-leva-tudo* (WTA).
- Uma forma de induzir a competição WTA é usar conexões inibitórias laterais (caminhos de retro-alimentação negativos) entre eles [6].

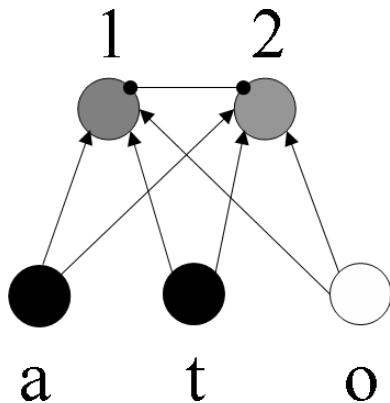
# Exemplo de aprendizado competitivo

- estímulo 'at' é apresentado:



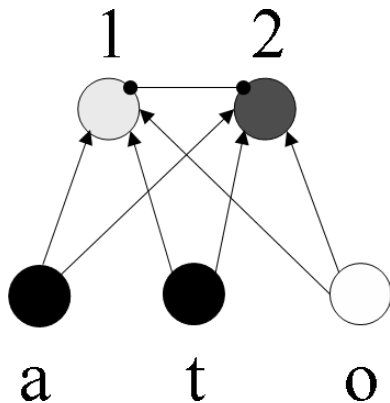
# Exemplo de aprendizado competitivo

- competição inicia no nível categoria:



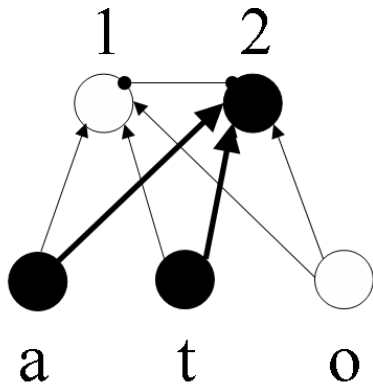
# Exemplo de aprendizado competitivo

- a competição resolve:



# Exemplo de aprendizado competitivo

- acontece o aprendizado Hebbiano:

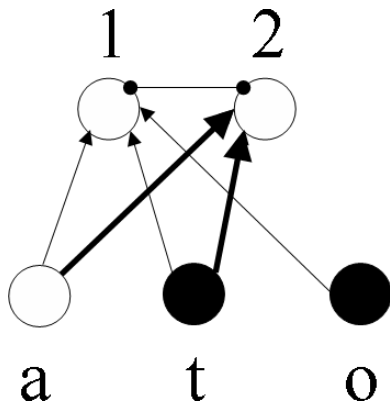


*Nó categoria 2 agora representa o 'at'*



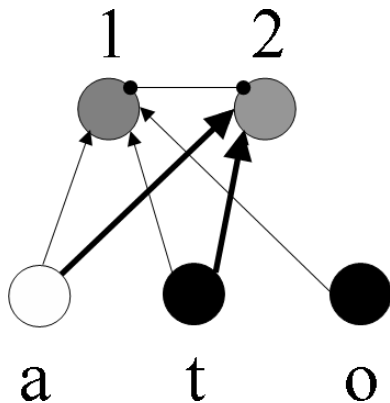
# Exemplo de aprendizado competitivo

- A apresentação do 'to' leva à ativação do nó categoria 1:



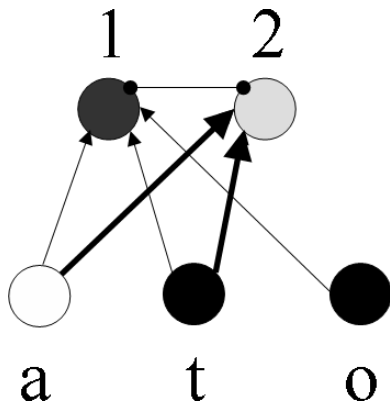
# Exemplo de aprendizado competitivo

- A apresentação do 'to' leva à ativação do nó categoria 1:



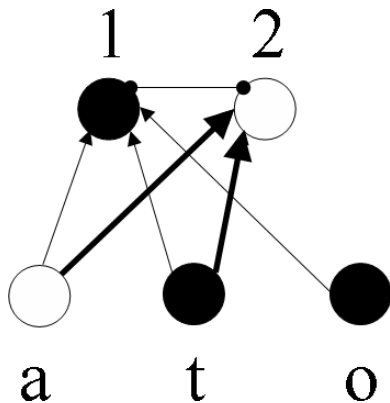
# Exemplo de aprendizado competitivo

- A apresentação do 'to' leva à ativação do nó categoria 1:



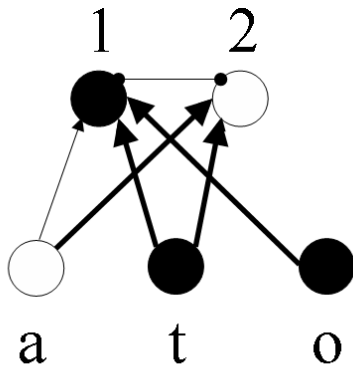
# Exemplo de aprendizado competitivo

- A apresentação do 'to' leva à ativação do nó categoria 1:



## Exemplo de aprendizado competitivo

- A categoria 1 é estabelecida através do aprendizado Hebbiano:



*Nó categoria 1 agora representa o 'to'*

# Mapa topográfico

- Em um *mapa auto-organizável* (SOM), os neurônios são colocados nos nós de um *retículo* de uma ou duas dimensões.
- Os neurônios tornam-se *sintonizados seletivamente* a vários padrões de entrada (estímulos) ou classes de padrões de entrada durante o aprendizado competitivo.
- As posições dos neurônios vencedores (sintonizados) tornam-se ordenadas entre si, de tal forma que um sistema de coordenadas significativas para diferentes *características* de entrada é criado no retículo.
- Um SOM é caracterizado pela formação de um *mapa topográfico* dos padrões de entrada, no qual as *posições espaciais* (isto é, *coordenadas*) dos neurônios no retículo são indicativas das características estatísticas contidas nos padrões de entrada.

# Níveis de adaptação

- Como um modelo neural, o SOM provê uma ponte entre dois níveis de adaptação:
  - Regras de adaptação formuladas no nível microscópico de um neurônio.
  - Formação de padrões de características experimentalmente melhores e fisicamente acessíveis seletivamente no nível microscópico de camadas neurais.
- Como um SOM é inerentemente não linear, pode ser visto como uma generalização não linear da PCA.
- O desenvolvimento de SOM como modelo neural é motivado pela característica distinta do cérebro humano: o cérebro é organizado de forma que entradas sensoriais diferentes são representadas por *mapas computacionais ordenados topologicamente*.
  - Por exemplo, entradas sensoriais como táteis, visuais e acústicas são mapeadas em áreas diferentes do córtex cerebral de uma maneira ordenada topologicamente.

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - **Mapeamento**
  - Formação do mapa
  - Simulação



# Mapeamento de características

- O córtex cerebral domina o cérebro, que é completamente envolvido, obscurecendo as outras partes.
- É impressionante a forma como entradas sensoriais diferentes (motora, somatosensorial, visual, auditiva, etc.) são *mapeadas* em áreas correspondentes do córtex cerebral de uma maneira *ordenada*.
- O uso de mapas computacionais oferece as seguintes propriedades:
  - A cada estágio de representação, cada parte de informação que chega é mantida no seu próprio contexto.
  - Neurônios, que tratam com partes de informação relacionadas, são aproximados para poderem interagir via conexões sinápticas curtas.

# Formação de mapas topográficos

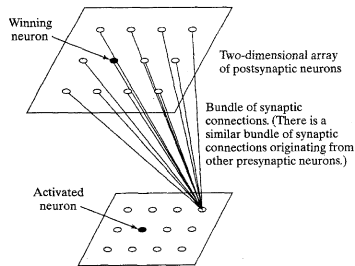
- O interesse está em construir mapas topográficos artificiais que aprendam através da auto-organização de uma maneira inspirada neurobiologicamente.
- Nesse contexto, o ponto importante que emerge da discussão de mapas computacionais no cérebro é o *princípio da formação de mapas topográficos*, estabelecida por Kohonen [3]:

*A posição espacial de um neurônio de saída em um mapa topográfico corresponde a um domínio particular ou à característica dos dados obtidos do espaço de entrada.*

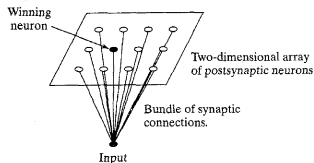
- Esse princípio provê a motivação neurobiológica para dois *modelos de mapeamento de características* basicamente diferentes, mostrados na figura 5.

# Formação de mapas topográficos

Figure 5: Dois mapas de características auto-organizados [1].



(a) Willshaw-von der Malsburg's model



(b) Kohonen model

# Retículos

- Em ambos os casos da figura 5, os neurônios de saída são arranjados em um retículo de duas dimensões, que assegura que cada neurônio tem um conjunto de vizinhos.
- Os modelos diferem entre si na maneira em que os padrões de entrada são especificados.
- O modelo da figura 5a foi proposto por Willshaw e von der Malsburg [9] em bases biológicas para explicar o problema do mapeamento da retina para o córtex visual:
  - Há dois retículos de neurônios de duas dimensões separados conectados juntos, com um projetando no outro.
  - Um retículo representa os neurônios pré-sinápticos (de entrada) e o outro representa os pós-sinápticos (de saída).
  - O retículo pós-sináptico usa um *mecanismo excitatório de curto alcance* e um *mecanismo inibitório de longo alcance*, locais e importantes para a auto-organização.
  - Os dois retículos são interconectados por sinapses modificáveis do tipo hebbiano.

# Modelo de Willshaw e von der Malsburg

- Modelo de Willshaw e von der Malsburg (figura 5a) (cont):
  - Os neurônios pós-sinápticos não são WTA: um limiar é usado para assegurar que apenas poucos neurônios pós-sinápticos sejam ativados de uma vez.
  - Para evitar que a rede fique instável, o peso total associado com cada neurônio pós-sináptico é limitado por uma condição de limite superior: para cada neurônio alguns pesos sinápticos aumentam enquanto outros diminuem.
  - A ideia básica é para que a proximidade geométrica de neurônios pré-sinápticos seja codificada na forma de correlações em sua atividade elétrica e sejam usadas essas correlações no retículo pós-sináptico para conectar neurônios pré-sinápticos vizinhos a neurônios pós-sinápticos vizinhos.
  - Um mapeamento ordenado topologicamente é portanto produzido pela auto-organização.
  - Entretanto, esse modelo é especializado em mapear onde a dimensão de entrada é a mesma da saída.

# Exemplo de Rede de Willshaw

- Aprendizado hetero-associativo nas *redes de Willshaw* é um exemplo de aprendizado supervisionado.
- Um peso tem o valor 0 ou 1.
- Um peso é colocado em 1 se a entrada e a saída são 1.
- Na recuperação a entrada da rede é dividida pelo número total de nós ativos no padrão de entrada.

# Exemplo de Rede de Willshaw

- Exemplo de uma memória hetero-associativa simples do tipo Willshaw:

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

# Exemplo de Rede de Willshaw

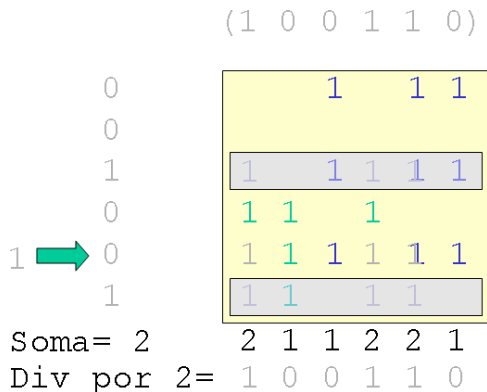
- Exemplo de recuperação de padrão:

		(1 0 0 1 1 0)
0		1 1 1
0		
1		1 1 1 1 1
0		1 1 1
1		1 1 1 1 1
1		1 1 1 1
Soma=	3	3 2 2 3 3 2
Div por	3=	1 0 0 1 1 0



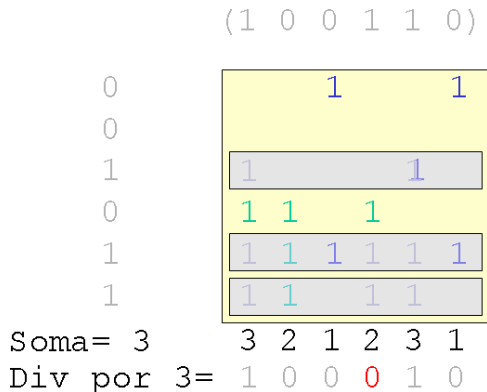
# Exemplo de Rede de Willshaw

- Exemplo: completar o padrão (de forma bem sucedida) usando um subpadrão:



# Exemplo de Rede de Willshaw

- Exemplo de degradação suave: pequenas lesões têm pequenos efeitos:



# Modelo de Kohonen

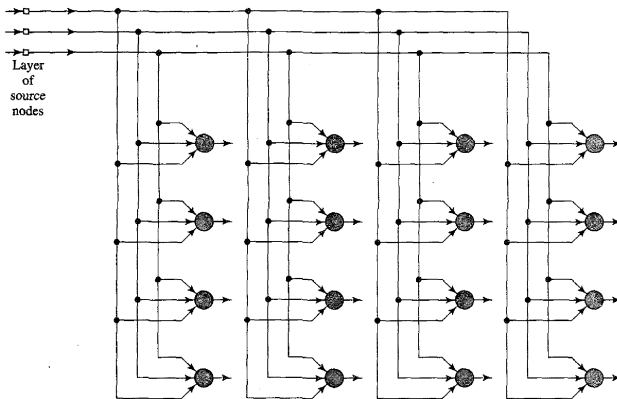
- O segundo modelo da figura 5b, introduzido por Kohonen [4], captura as características essenciais dos mapas computacionais no cérebro e ainda assim permanece computacionalmente tratável:
  - Parece que esse modelo é mais geral que o modelo de Willshaw e von der Malsburg no sentido de que é capaz de realizar compressão de dados (isto é, redução de dimensionalidade na entrada).
  - Pertence à classe de algoritmos de *codificação de vetor*.
  - Provê um mapeamento topológico que otimamente coloca um número fixo de vetores em um espaço de entrada de dimensão mais alta, facilitando a compressão de dados.
  - Na abordagem tradicional, pode-se usar as ideias básicas da auto-organização, motivadas pelas considerações neurobiológicas, para derivar o modelo.
  - Alternativamente, pode-se usar a abordagem de quantização de vetor que usa um modelo que envolve um codificador e um decodificador.

# Mapa auto-organizável

- O principal objetivo do mapa auto-organizável (SOM) é transformar um padrão de sinal de entrada, de dimensão arbitrária, em um mapa discreto de uma ou duas dimensões e realizar essa transformação adaptativamente de uma maneira ordenada topologicamente.
- A figura 6 mostra um retículo de neurônios de duas dimensões usado como mapa discreto.
- Cada neurônio é totalmente conectado a todos os nós fonte da camada de entrada.
- Essa rede representa um estrutura *feedforward* com uma única camada computacional consistindo de neurônios arranjados em linhas e colunas.
- Um retículo unidimensional é um caso especial dessa rede com apenas uma linha ou uma coluna.

# Mapa auto-organizável

Figure 6: Retículo de neurônios de duas dimensões [1].



# Mapa auto-organizável

- Cada padrão de entrada apresentado à rede consiste de uma região de atividade localizada.
- A posição e natureza de tal região varia de uma a outra realização do padrão de entrada.
- Todos os neurônios devem ser expostos a um número suficiente de diferentes realizações do padrão de entrada para assegurar que o processo de auto-organização tenha chance de amadurecer.
- Algoritmo responsável pela formação do SOM:
  - iniciação dos pesos sinápticos: atribui-se valores pequenos aleatórios, para não impor nenhuma ordem.
  - formação do mapa auto-organizável.

# Mapa auto-organizável

- Formação do mapa auto-organizável:
  - 1 *competição*: para cada padrão de entrada, os neurônios na rede computam seus valores respectivos de uma função discriminante, que provê a base para a competição. O neurônio com o maior valor é o vencedor.
  - 2 *cooperação*: o neurônio vencedor determina a posição espacial de uma vizinhança topológica de neurônios excitados, provendo a base para cooperação entre tais neurônios.
  - 3 *adaptação sináptica*: habilita os neurônios excitados para aumentar seus valores individuais da função discriminante em relação ao padrão de entrada através de ajustes adequados aplicados aos pesos sinápticos. Os ajustes são feitos para que a resposta do neurônio vencedor à aplicação subsequente de um padrão de entrada similar seja melhorada.

# Sumário

- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - **Formação do mapa**
  - Simulação



# Processo competitivo

- Seja  $m$  a dimensão do espaço (dados) de entrada.
- Seja um padrão (vetor) de entrada selecionado do espaço de entrada aleatoriamente denotado por

$$\mathbf{x} = [x_1, x_2, \dots, x_m]^T \quad (45)$$

- O vetor de pesos sinápticos de cada neurônio na rede tem a mesma dimensão do espaço de entrada.
- Seja o vetor de pesos sinápticos do neurônio  $j$ :

$$\mathbf{w}_j = [w_{j1}, w_{j2}, \dots, w_{jm}]^T, \quad j = 1, 2, \dots, \ell \quad (46)$$

onde  $\ell$  é o número total de neurônios na rede.

- Para achar a melhor combinação do vetor de entrada  $\mathbf{x}$  com os vetores de pesos sinápticos  $\mathbf{w}_j$ , compare os produtos internos  $\mathbf{w}_j^T \mathbf{x}$  para  $j = 1, 2, \dots, \ell$  e selecione o maior.

# Processo competitivo

- Assume-se que o mesmo limiar é aplicado a todos os neurônios.
- Portanto, ao selecionar o neurônio com o maior produto interno  $\mathbf{w}_j^T \mathbf{x}$ , está-se determinando a posição onde a vizinhança topológica de neurônios excitados será centralizada.
- O melhor critério de combinação, baseado em maximizar o produto interno  $\mathbf{w}_j^T \mathbf{x}$ , é matematicamente equivalente a minimizar a distância euclidiana entre os vetores  $\mathbf{x}$  e  $\mathbf{w}_j$ .
- Se se usar o índice  $i(\mathbf{x})$  para identificar o neurônio que melhor combina com o vetor de entrada  $\mathbf{x}$ , pode-se determinar  $i(\mathbf{x})$  aplicando a condição

$$i(\mathbf{x}) = \arg \min_j \| \mathbf{x} - \mathbf{w}_j \|^2, \quad j = 1, 2, \dots, \ell \quad (47)$$

que resume a essência do processo de competição entre os neurônios.

# Processo competitivo

- O neurônio  $i$  que satisfaz a condição da equação 47 ( $i(\mathbf{x})$ ) é chamado de *neurônio vencedor* para o vetor de entrada  $\mathbf{x}$ .
- A equação 47 leva à observação:

*Um espaço de entrada contínuo de padrões de ativação é mapeado em um espaço de saída discreto de neurônios por um processo de competição entre os neurônios na rede.*

- Dependendo da aplicação, a resposta da rede pode ser o índice do neurônio vencedor (isto é, sua posição no retículo), ou o vetor de pesos sinápticos que está mais próximo do vetor de entrada num sentido euclidiano.

# Processo cooperativo

- O neurônio vencedor localiza o centro de uma vizinhança topológica de neurônios cooperantes.
- Para se definir uma vizinhança topológica que seja correta neurobiologicamente:
  - há evidência de *interação lateral* entre neurônios excitados.
  - um neurônio que está ativo tende a excitar mais os neurônios em sua vizinhança.
  - a vizinhança topológica do neurônio vencedor  $i$  decai suavemente com a distância lateral.
- Seja  $h_{j,i}$  a *vizinhança topológica* centrada no neurônio vencedor  $i$ , abrangendo um conjunto de neurônios excitados (cooperantes), incluindo  $j$ .
- Seja  $d_{i,j}$  a distância *lateral* entre o neurônio vencedor  $i$  e o neurônio excitado  $j$ .
- A vizinhança topológica  $h_{j,i}$  é uma função unimodal da distância lateral  $d_{i,j}$ .

# Processo cooperativo

- Essa função satisfaz dois requisitos:
  - $h_{j,i}$  é simétrica sobre o ponto máximo definido por  $d_{i,j} = 0$ . Ou seja,  $h_{j,i}$  alcança seu valor máximo no neurônio vencedor  $i$  para o qual  $d_{j,i} = 0$ .
  - A amplitude de  $h_{j,i}$  decresce monotonicamente com o aumento da  $d_{j,i}$ , chegando a zero para  $d_{j,i} \rightarrow \infty$ : condição necessária para a convergência.
- Escolha típica de  $h_{j,i}$  que satisfaz esses requisitos é a função gaussiana:

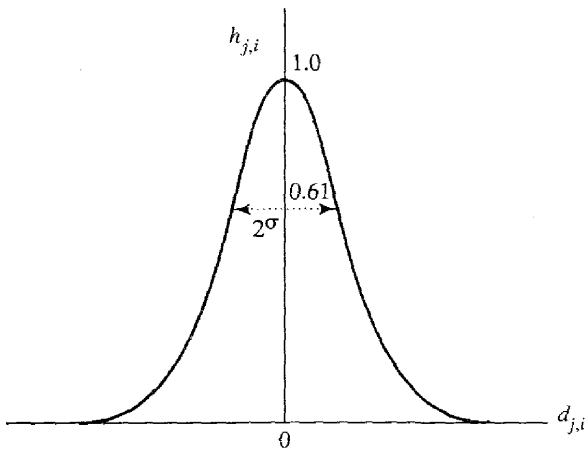
$$h_{j,i}(\mathbf{x}) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right) \quad (48)$$

que é *invariante de translação* (isto é, independente da posição do neurônio vencedor).

- O parâmetro  $\sigma$  é a “largura efetiva” da vizinhança topológica (figura 7).

# Função de vizinhança

Figure 7: Função de vizinhança gaussiana [1].



# Processo cooperativo

- $\sigma$  mede o grau em que os neurônios excitados na vizinhança do neurônio vencedor participam do processo de aprendizado.
- Num sentido qualitativo, a vizinhança topológica gaussiana da equação 48 é mais biologicamente apropriada que uma retangular: na gaussiana o algoritmo SOM converge mais rapidamente.
- No caso do retículo unidimensional,  $d_{j,i}$  é um inteiro igual a  $|j - i|$ .
- No caso do retículo bidimensional,

$$d_{j,i}^2 = \| \mathbf{r}_j - \mathbf{r}_i \|^2 \quad (49)$$

onde o vetor discreto  $\mathbf{r}_j$  define a posição do neurônio excitado  $j$  e  $\mathbf{r}_i$  define a posição discreta do neurônio vencedor  $i$ , ambos medidos no espaço de saída discreto.

# Processo cooperativo

- O tamanho da vizinhança topológica encolhe com o tempo no algoritmo SOM.
- Isso acontece fazendo a largura  $\sigma$  da função da vizinhança topológica  $h_{j,i}$  diminuir com o tempo.
- A dependência de  $\sigma$  no tempo discreto  $n$  é o decaimento exponencial

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) \quad n = 0, 1, 2, \dots, \quad (50)$$

onde  $\sigma_0$  é o valor de  $\sigma$  na iniciação do algoritmo SOM e  $\tau_1$  é uma *constante de tempo*.

- De forma correspondente, a vizinhança topológica assume uma forma variante no tempo (*função de vizinhança*):

$$h_{j,i(\mathbf{x})}(n) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2(n)}\right) \quad n = 0, 1, 2, \dots, \quad (51)$$



# Processo adaptativo

- O processo adaptativo sináptico é o último processo na formação auto-organizada de um mapa de características.
- Para que a rede seja auto-organizável, o vetor de pesos sinápticos  $\mathbf{w}_j$  do neurônio  $j$  na rede deve mudar em relação ao vetor de entrada  $\mathbf{x}$ .
- Pelo postulado de aprendizado de Hebb, um peso sináptico é aumentado com a ocorrência simultânea de atividades pré e pós-sinápticas.
- O uso de tal regra é adequada para o aprendizado associativo.
- Para o tipo de aprendizado não-supervisionado discutido aqui, entretanto, a hipótese de Hebb é insatisfatória pois mudanças nas conectividades ocorrem em apenas uma direção, levando todos os pesos sinápticos à saturação.

# Processo adaptativo

- Deve-se modificar a hipótese de Hebb incluindo um *termo de esquecimento*  $g(y_j)\mathbf{w}_j$ , onde  $\mathbf{w}_j$  é o vetor de pesos sinápticos do neurônio  $j$  e  $g(y_j)$  é uma função escalar positiva da resposta  $y_j$ .
- A única imposição à função  $g(y_j)$  é que o termo constante na expansão de  $g(y_j)$  da série de Taylor seja zero:

$$g(y_j) = 0 \quad \text{para } y_j = 0 \quad (52)$$

- Pode-se expressar a mudança ao vetor de pesos do neurônio  $j$  no retículo:

$$\Delta\mathbf{w}_j = \eta y_j \mathbf{x} - g(y_j)\mathbf{w}_j \quad (53)$$

- O primeiro termo do lado direito é o termo hebbiano e o segundo é o termo de esquecimento.

# Processo adaptativo

- Para satisfazer a equação 52, escolheu-se uma função linear para  $g(y_j)$ :

$$g(y_j) = \eta y_j \quad (54)$$

- Pode-se simplificar a equação 53, fazendo

$$y_j = h_{j,i}(\mathbf{x}) \quad (55)$$

- Usando as equações 54 e 55 em 53:

$$\Delta \mathbf{w}_j = \eta h_{j,i}(\mathbf{x})(\mathbf{x} - \mathbf{w}_j) \quad (56)$$

- Finalmente, usando formalismo de tempo discreto, o vetor de pesos atualizado no tempo  $n + 1$ :

$$\mathbf{w}_j(n + 1) = \mathbf{w}_j(n) + \eta(n) h_{j,i}(\mathbf{x})(n)(\mathbf{x} - \mathbf{w}_j(n)) \quad (57)$$

que é aplicado a todos os neurônios no retículo que se situam dentro da vizinhança topológica do neurônio vencedor  $i$ .

# Processo adaptativo

- A equação 57 tem o efeito de mover o vetor de pesos sinápticos  $\mathbf{w}_i$  do neurônio vencedor  $i$  em direção ao vetor de entrada  $\mathbf{x}$ .
- Após repetidas apresentações dos dados de treinamento, os vetores de pesos sinápticos tendem a seguir a distribuição dos vetores de entrada devido à atualização da vizinhança.
- O algoritmo leva a uma *ordenação topológica* do mapa de características no espaço de entrada: os neurônios que são adjacentes no retículo tenderão a ter vetores de pesos sinápticos similares.
- A equação 57 é usada para computar os pesos sinápticos do mapa de características.

# Processo adaptativo

- Além disso, precisa-se da heurística da equação 51 para selecionar  $h_{j,i(\mathbf{x})}(n)$  e uma outra heurística para selecionar  $\eta(n)$ , que deve variar (aproximação estocástica) na forma indicada na equação 57, começando por  $\eta_0$  e decrescendo gradualmente ao longo do tempo  $n$ .
- Isso pode ser satisfeito escolhendo um decaimento exponencial para  $\eta(n)$ :

$$\eta(n) = \eta_0 \exp\left(-\frac{n}{\tau_2}\right), \quad n = 0, 1, 2, \dots, \quad (58)$$

onde  $\tau_2$  é outra constante de tempo do algoritmo SOM.

- Apesar das fórmulas de decaimento descritas nas equações 50 e 58 para a largura da função de vizinhança e taxa de aprendizado não serem ótimas, são adequadas para a formação do mapa de características de uma maneira auto-organizada.

# As duas fases do processo adaptativo

- Começando de um estado inicial de completa desordem, o algoritmo SOM gradualmente leva a uma representação organizada dos padrões de ativação obtidos do espaço de entrada, desde que os parâmetros do algoritmo sejam selecionados apropriadamente.
- Pode-se decompor a adaptação dos pesos sinápticos na rede, computados de acordo com a equação 57, em duas fases:
  - 1 *Fase auto-organizável ou de ordenação.*
  - 2 *Fase de convergência.*

# As duas fases do processo adaptativo

- 1** *Fase auto-organizável* ou *de ordenação*: ordenação topológica dos vetores de pesos. 1000 iterações do algoritmo SOM. Considerações:
  - $\eta(n)$  deve começar com 0.1 e decrescer gradualmente até 0.01:  $\eta_0 = 0.1, \tau_2 = 1000$ .
  - $h_{j,i}(n)$  deve inicialmente incluir quase todos os neurônios na rede centrada no neurônio vencedor  $i$  e então encolher vagarosamente no tempo.

$\sigma_0$  igual ao “raio” do retículo.  $\tau_1 = 1000/\log \sigma_0$ .
- 2** *Fase de convergência*: ajuste fino do mapa de características para prover uma quantificação estatística acurada do espaço de entrada. O número de iterações deve ser 500 vezes o número de neurônios:
  - $\eta(n)$  deve se manter em 0.01.
  - $h_{j,i}(\mathbf{x})$  deve conter apenas os vizinhos mais próximos do neurônio vencedor.

# Sumário do algoritmo SOM

- A essência do algoritmo SOM de Kohonen é que ele substitui uma computação geométrica simples para as propriedades mais detalhadas da regra tipo Hebb e interações laterais.
- Os ingredientes ou parâmetros do algoritmo são:
  - Um espaço de entrada contínuo de padrões de ativação que são gerados de acordo com certa distribuição de probabilidade.
  - Uma topologia da rede na forma de um retículo de neurônios, que define um espaço de saída discreto.
  - Uma função de vizinhança variante no tempo  $h_{j,i(\mathbf{x})}(n)$  que é definida em volta de um neurônio vencedor  $i(\mathbf{x})$ .
  - Um parâmetro taxa de aprendizado  $\eta(n)$  que começa com um valor inicial  $\eta_0$  e então decresce gradualmente com o tempo  $n$ , mas nunca chega a zero.



# Sumário do algoritmo SOM

- Resumo do algoritmo:

- 1 *Iniciação*: escolha valores pequenos aleatórios para  $\mathbf{w}_j(0)$ , que tem de ser diferente para  $j = 1, 2, \dots, \ell$ , onde  $\ell$  é o número de neurônios no retículo. Ou selecione  $\{\mathbf{w}_j(0)\}_{j=1}^{\ell}$  do conjunto disponível  $\{\mathbf{x}_i\}_{i=1}^N$  de forma aleatória.
- 2 *Amostragem*: obtenha uma amostra  $\mathbf{x}$  do espaço de entrada com uma certa probabilidade.  $\mathbf{x}$  representa o padrão de ativação que é aplicado ao retículo. A dimensão de  $\mathbf{x}$  é  $m$ .
- 3 *Combinação de similaridade*: ache o neurônio que melhor combina (vencedor)  $i(\mathbf{x})$  no passo de tempo  $n$  usando a distância euclidiana mínima:  
$$i(\mathbf{x}) = \arg \min_j \| \mathbf{x}(n) - \mathbf{w}_j \|, \quad j = 1, 2, \dots, \ell.$$
- 4 *Atualização*: ajuste os vetores de pesos sinápticos de todos os neurônios:  
$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n)h_{j,i(\mathbf{x})}(n)(\mathbf{x}(n) - \mathbf{w}_j(n)).$$
- 5 *Continuação*: continue com o passo 2 até que nenhuma mudança no mapa de características seja observada.

# Sumário

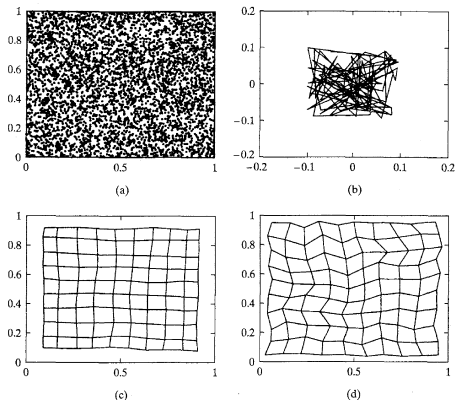
- 1 Auto-organização
  - Princípios
  - Análise auto-organizada de características
- 2 PCA
  - Análise de Componentes Principais
  - Autoestrutura
  - Dados
- 3 Mapas auto-organizáveis
  - SOM
  - Mapeamento
  - Formação do mapa
  - **Simulação**

# Retículo de duas dimensões

- Seja uma rede com 100 neurônios, arranjada na forma de um retículo de duas dimensões com 10 linhas e 10 colunas.
- A rede é treinada com um vetor de entrada bidimensional  $\mathbf{x}$ , cujos elementos  $x_1$  e  $x_2$  são uniformemente distribuídos na região  $\{(-1 < x_1 < +1); (-1 < x_2 < +1)\}$ .
- Para iniciar a rede os pesos sinápticos são escolhidos aleatoriamente.
- A figura 8 mostra três estágios do treinamento, quando a rede aprende a representar a distribuição de entrada.
- A figura 8a mostra a distribuição uniforme dos dados usados para treinar o mapa de características.
- A figura 8b mostra os valores iniciais dos pesos sinápticos.

# Retículo de duas dimensões

**Figure 8:** (a) Distribuição dos dados de entrada. (b) Condição inicial do retículo de duas dimensões. (c) Condição do retículo no final da fase de ordenação. (d) Condição do retículo no final da fase de convergência [1].



# Retículo de duas dimensões

- As figuras 8c e 8d apresentam os valores dos vetores de pesos sinápticos, plotados como pontos no espaço de entrada, depois de completar as fases de ordenação e convergência, respectivamente.
- As linhas da figura 8 conectam neurônios vizinhos (ao longo das linhas e colunas) na rede.
- Os resultados mostrados na figura 8 demonstram que as fases de ordenação e convergência que caracterizam o processo de aprendizado do algoritmo SOM.
- Durante a fase de ordenação, o mapa *desdobra-se* para formar um reticulado (figura 8c).
- Os neurônios são mapeados na ordem correta ao final dessa fase.

# Retículo de duas dimensões

- Durante a fase de convergência, o mapa estende-se para preencher o espaço de entrada.
- No final dessa fase (figura 8d) a distribuição estatística dos neurônios no mapa aproxima-se do vetor de entrada, exceto por efeitos de borda.
- Comparando as figuras 8d com 8a, vê-se que a sintonização do mapa durante a fase de convergência capturou as irregularidades locais que podem ser vistas na distribuição de entrada.
- A propriedade de ordenação topológica do algoritmo SOM é bem ilustrada na figura 8d.
- Observa-se que o algoritmo (depois da convergência) captura a topologia proveniente da distribuição uniforme na entrada.

# Bibliografia I

- [1] S. Haykin  
*Neural networks - a comprehensive foundation.*  
2nd. edition. Prentice Hall, 1999.
- [2] A. L. de Bortoli, C. Cardoso, M. P. G. Fachin, R. D. da Cunha  
*Introdução ao Cálculo Numérico - Capítulo 6. Autovalores e autovetores.*  
Instituto de Matemática - PPGMAp - UFRGS. Texto disponível em  
[www.mat.ufrgs.br/~dago/num/rcunha/cap6.pdf](http://www.mat.ufrgs.br/~dago/num/rcunha/cap6.pdf).
- [3] T. Kohonen  
*The self-organizing map.*  
*Proceedings of the Institute of Electrical and Electronics Engineers*, vol. 78, pp. 1464–1480, 1990.

## Bibliografia II

[4] T. Kohonen

*Self-organized formation of topologically correct feature maps.*

*Biological Cybernetics*, vol. 43, pp. 59–69, 1982.

[5] R. Linsker

*From basic network principles to neural architecture (series).*

*Proceedings of the National Academy of Sciences, USA*, vol. 83, pp. 7508–7512, 8390–8394, 8779–8783, 1986.

[6] F. Rosenblatt

*The Perceptron: A probabilistic model for information storage and organization in the brain.*

*Psychological Review*, vol. 65, pp. 386–408, 1958.



# Bibliografia III

[7] A. M. Turing

*The chemical basis of morphogenesis.*

*Philosophical Transactions of the Royal Society, B.* vol. 237,  
pp. 5–72, 1952.

[8] Wikipedia

[http://en.wikipedia.org/wiki/Rank\\_%28linear\\_algebra%29](http://en.wikipedia.org/wiki/Rank_%28linear_algebra%29).

[9] D. J. Willshaw and C. von der Malsburg

*How patterned neural connections can be set up by self-organization.*

*Proceedings of the Royal Society of London Series B.* vol. 194, pp. 431–445, 1976.