

SOM (SELF ORGANIZING MAPS)

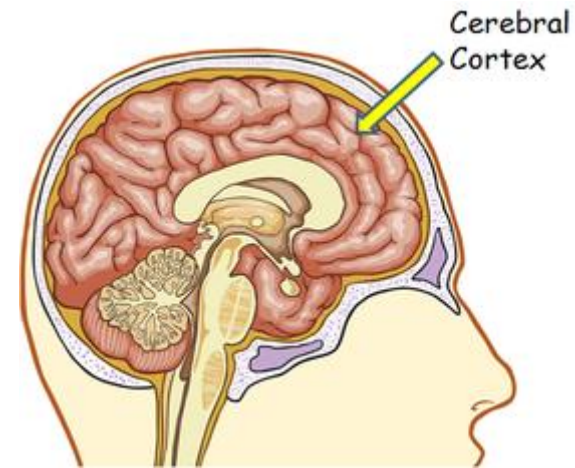
Prof. Valmir Macario

DC-UFRPE

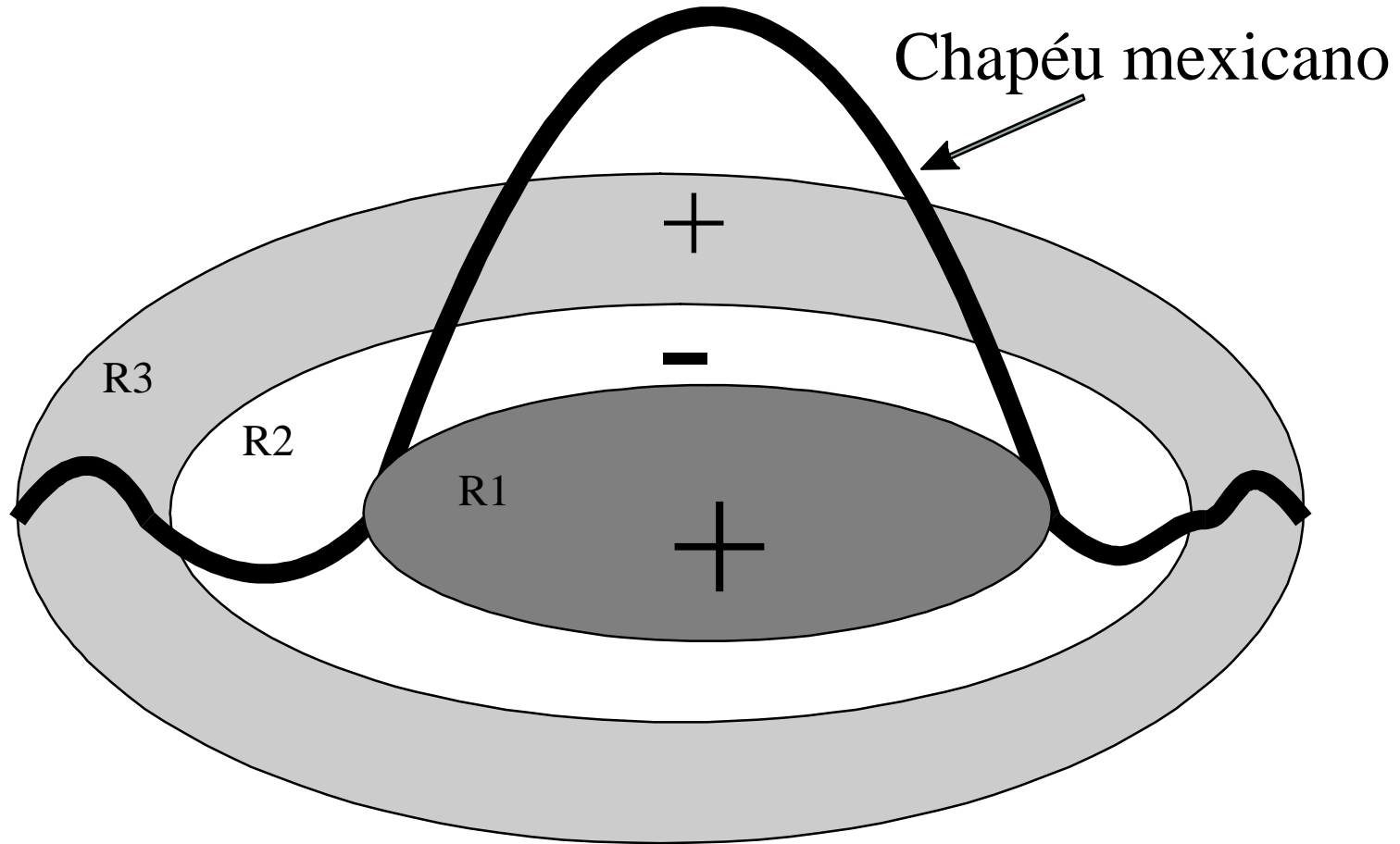


SOM (SELF-ORGANIZING MAPS)

- Foram desenvolvidas por Kohonen na década de 80.
- São baseadas no mapa topológico presente no córtex cerebral.
- Neurônios topologicamente próximos tendem a responder a padrões ou estímulos semelhantes.
- Sabe-se também que esta ordenação topológica é resultado de uso de feedback lateral entre as células do córtex cerebral.
- Esse feedback lateral é geralmente modelado em RNAs por uma função conhecida como chapéu mexicano.



CARACTERÍSTICAS BÁSICAS



SOM (SELF-ORGANIZING MAPS)

- Quando um neurônio é excitado, ao redor uma área entre 50 e 100 μm também sofre excitação
- Ao redor, uma área sofre inibição para impedir a propagação do sinal a áreas não relacionadas
- A função do chapéu mexicano influencia o estado de ativação de seus neurônios vizinhos de três formas possíveis:
 - Excitatória: se os vizinhos estão próximos a ele
 - Inibitória: se os vizinhos estão fora da área anterior mas dentro de uma segunda área
 - Levemente excitatória: se os vizinhos estão agora das áreas anteriores mas dentro de uma terceira área.



APRENDIZAGEM COMPETITIVA

- O objetivo da aprendizagem competitiva é criar uma regra de aprendizagem que possa ser aplicada a uma topologia de camada única e que atribua os PEs a áreas diferentes do espaço de entrada
- Aprendizagem competitiva é um paradigma de aprendizagem não-supervisionada, a qual extrai informação somente dos padrões de entrada sem a necessidade de uma resposta desejada



APRENDIZAGEM COMPETITIVA

- Neurônios de saída da RNA competem entre si para se tornar ativos
- Apenas um neurônio de saída está ativo em um determinado instante
- Três elementos básicos:
 1. Neurônios com mesma estrutura, diferente pelos *pesos*, de forma que tenham *respostas diferentes* a uma entrada
 2. Um *limite* imposto sobre a força de cada neurônio
 3. Mecanismo de competição entre neurônios, de forma que um neurônio é vencedor em um dado instante.



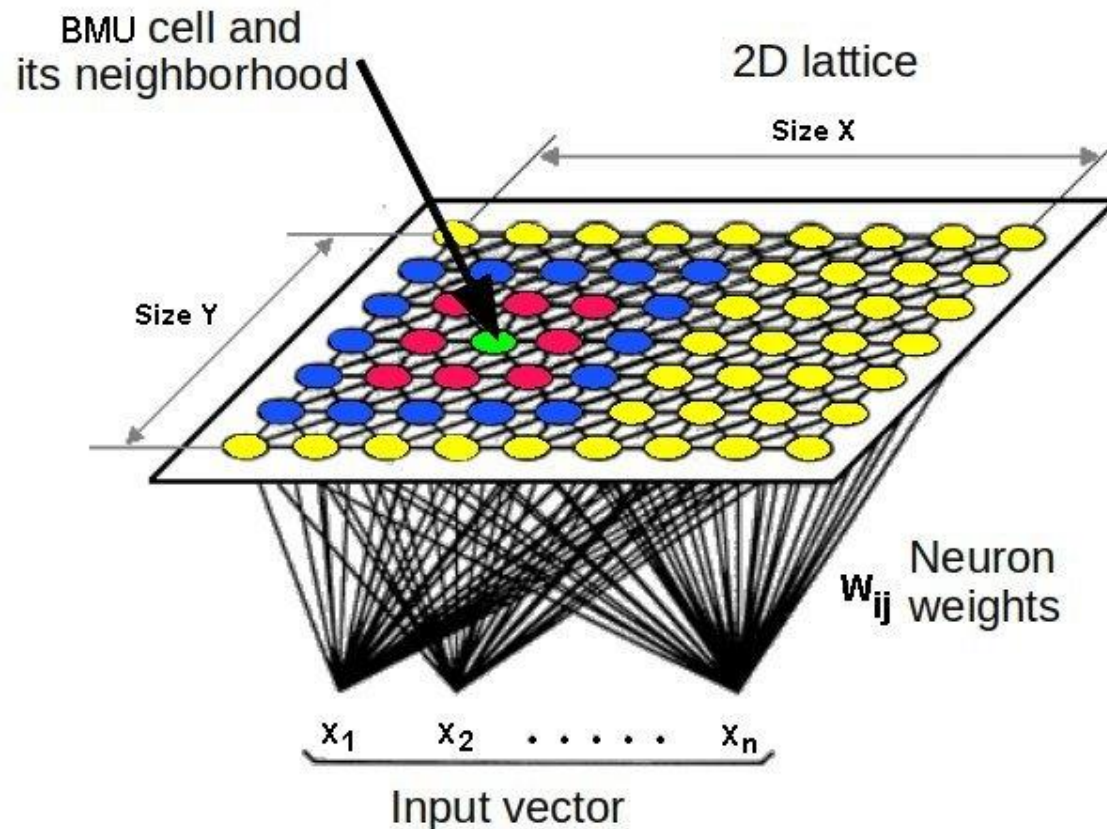
APRENDIZAGEM COMPETITIVA

- Em cada momento o neurônio vencedor:
 - aprende a se especializar em agrupamentos de padrões similares
 - tornam-se *detectores de características* para classes diferentes de padrões de entrada
- Dois tipos básicos:
 - *Hard*: somente um neurônio ganha os recursos
 - *Soft*: há um vencedor, mas seus vizinhos também recebem uma parte dos recursos.



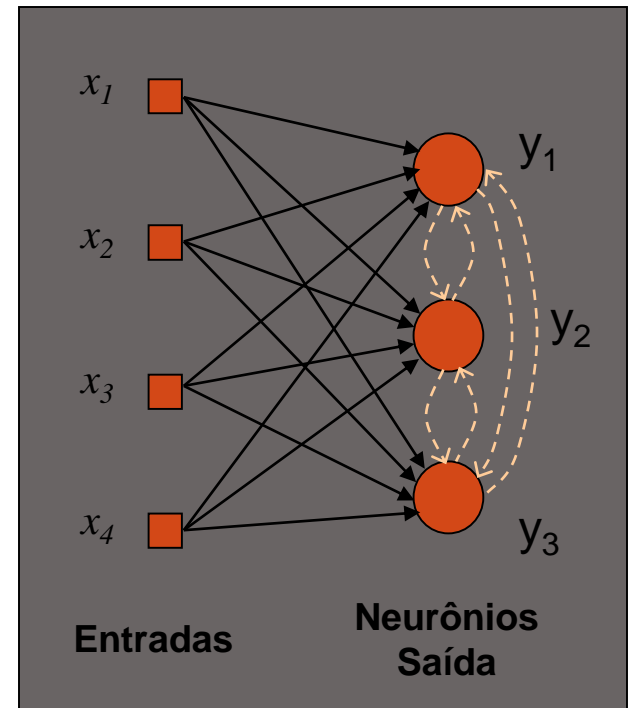
SOM (SELF-ORGANIZING MAPS)

- Topologia da rede tem a forma de um *matriz* de nodos.



SOM (SELF-ORGANIZING MAPS)

- O número de unidades de entrada define a dimensionalidade dos dados
- Exemplo de estrutura simples:
 - 1 camada de saída
 - Totalmente conectada à entrada
 - Pode incluir realimentação entre neurônios para inibição lateral
- Conexões excitadoras das entradas para os neurônios (setas fechadas com linhas contínuas)
- Conexões laterais inibitórias entre os neurônios (setas abertas em linhas tracejadas)



SOM – ALGORITMO BÁSICO

1. **Inicialização:** geralmente aleatória, pode ainda ser estimada por análise da representação dos dados
2. **Competição:** para cada padrão de entrada, calcula-se a resposta dos neurônios de saída (grade). O neurônio com a maior resposta é o vencedor da competição.
3. **Cooperação:** o neurônio vencedor define uma vizinhança topológica (em função da grade) de neurônios excitados
4. **Adaptação Sináptica:** aprendizado em relação ao padrão de entrada. Os pesos do neurônio vencedor, e de sua vizinhança, ficam mais próximos do padrão de entrada.

CARACTERÍSTICAS BÁSICAS

- Estados de ativação
 - $[0, N]$
- Função de ativação
 - $d_j = \sum (x_i - w_{ij})^2$
 - Baseada em distância Euclidiana
- Função de saída
 - Função identidade

CARACTERÍSTICAS BÁSICAS

■ Treinamento

■ Atualização dos pesos

- Atualiza neurônio vencedor e seus vizinhos dentro de um certo raio
- Raio e taxa de aprendizado são decrementados durante treinamento
- $w_{ij}(t+1) = w_{ij}(t) + \eta(t)(x_i(t) - w_{ij}(t))$
- (neurônio $j \in$ vizinhança do vencedor)
- Cria regiões que respondem a um grupo de entradas semelhantes



CRITÉRIO DE COMPETIÇÃO

- O critério do melhor casamento (*best matching*) é baseado na maximização do produto interno, como na aprendizagem competitiva
- Isto é matematicamente equivalente a minimizar a distância euclidiana entre **w** e **x**
- O neurônio vencedor, é escolhido por:

$$v = \arg \min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j = 1, 2, \dots, N$$

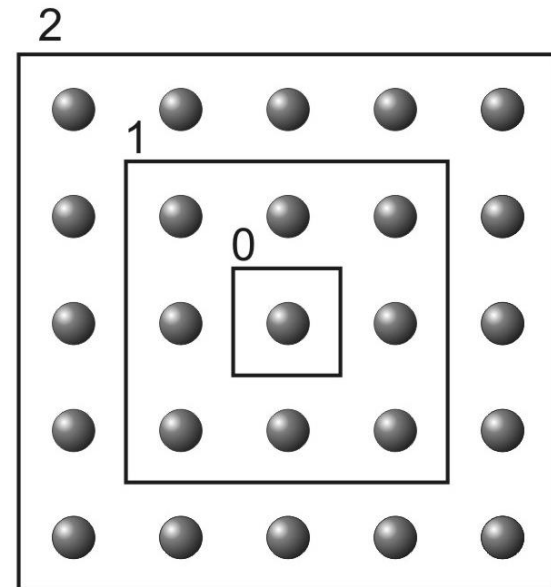
PROCESSO COOPERATIVO

- Então, a função de vizinhança torna-se:

$$h_{vj}(t) = \begin{cases} 1, & j \in N_v(t) \\ 0, & j \notin N_v(t) \end{cases}$$

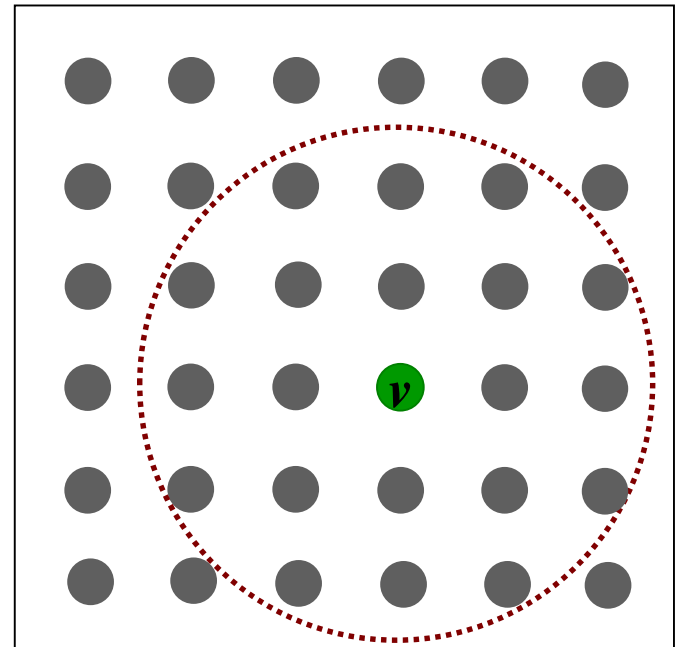
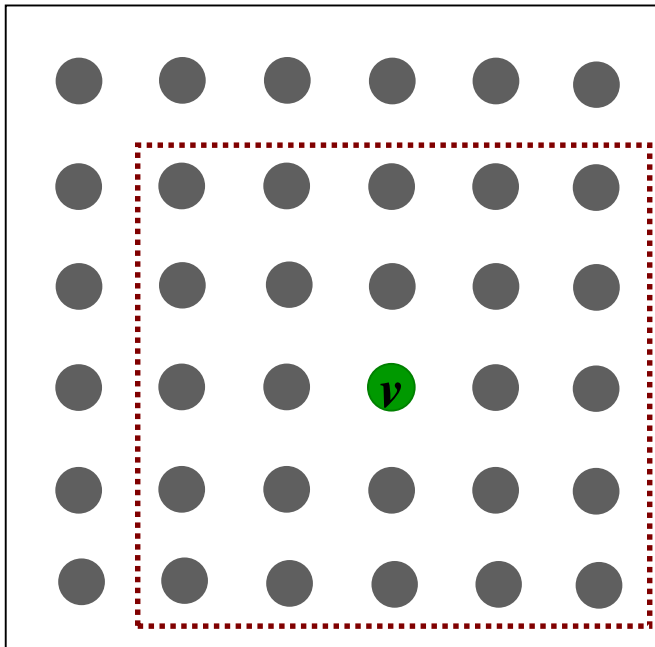
- $N_v(t)$ é função monotônica decrescente no tempo
- A função de vizinhança é mapeada no espaço de saída

Figura: Exemplo de níveis de vizinhança para uma grade retangular de neurônios



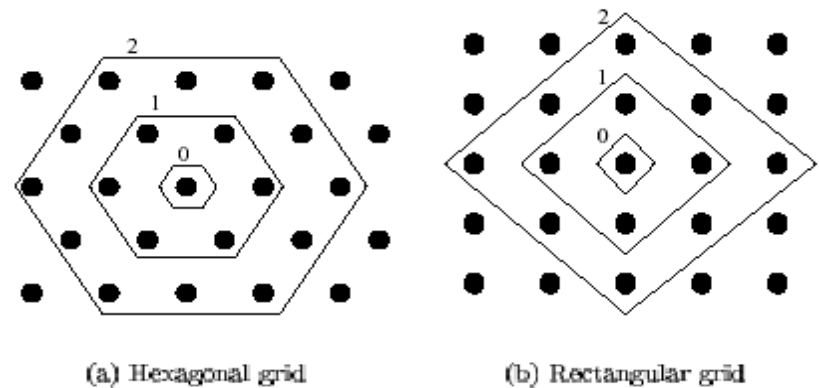
PROCESSO COOPERATIVO

- Exemplos de função de vizinhança
 - Tipo bolha (todos os neurônios da vizinhança serão atualizados igualmente)
 - Base Quadrada / Base Circular

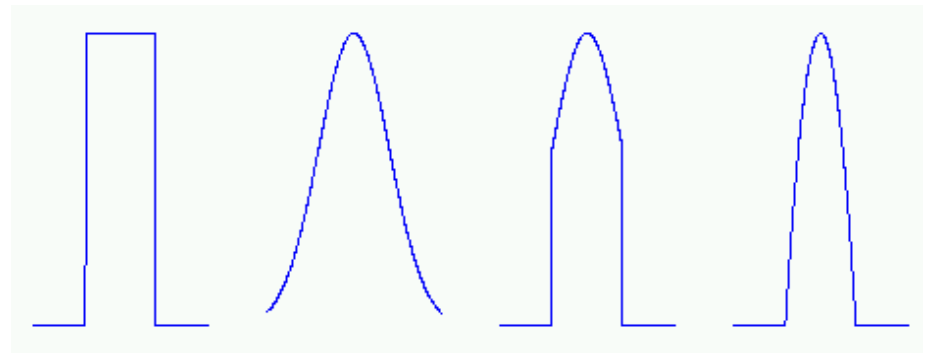


VIZINHANÇAS

- Pode ter diferentes formatos
 - Hexágono
 - Retangular



- Pode ter diferentes funções de vizinhança
 - Bubble
 - Gaussian
 - Cut gaussian
 - Epanechicov



PROCESSO COOPERATIVO

- Uma função muito interessante a ser usada como função de vizinhança é a Gaussiana, dada por:

$$h_{vj}(t) = \exp\left(-\frac{\|\mathbf{r}_v - \mathbf{r}_j\|}{2\sigma^2(t)}\right)$$

onde o termo $\|\mathbf{r}_v - \mathbf{r}_j\|$ é a distância entre o neurônio v vencedor e o neurônio j , que está sendo atualizado

PROCESSO COOPERATIVO

- O parâmetro σ define a largura da função e deve ser decrescente no tempo. Pode ser usada uma função linear, mas em geral é usada a exponencial:

$$\sigma(t) = \sigma(0) \exp\left(-\frac{t}{\tau_1}\right)$$

$\sigma(0)$ é um valor inicial para σ

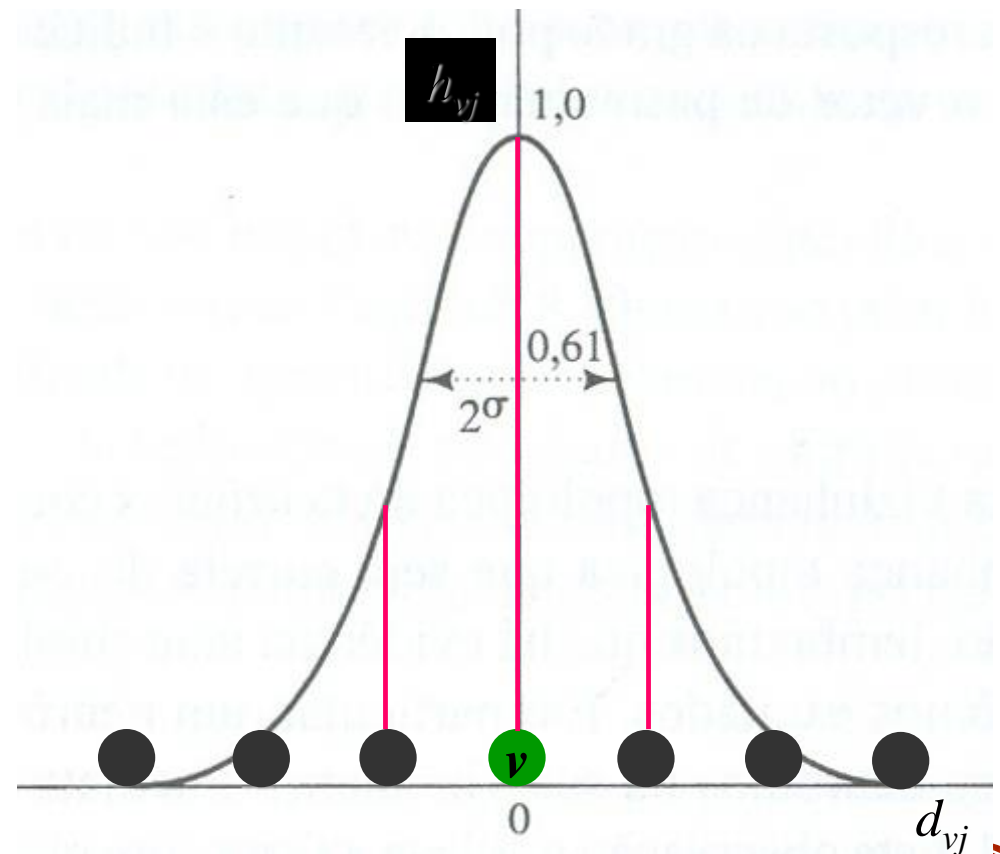
τ_1 é uma constante de tempo do SOM definida para que a taxa de aprendizagem nunca decaia para zero

$$\tau_1 = \frac{N_{iter}}{\log \sigma(0)}$$

PROCESSO COMPETITIVO

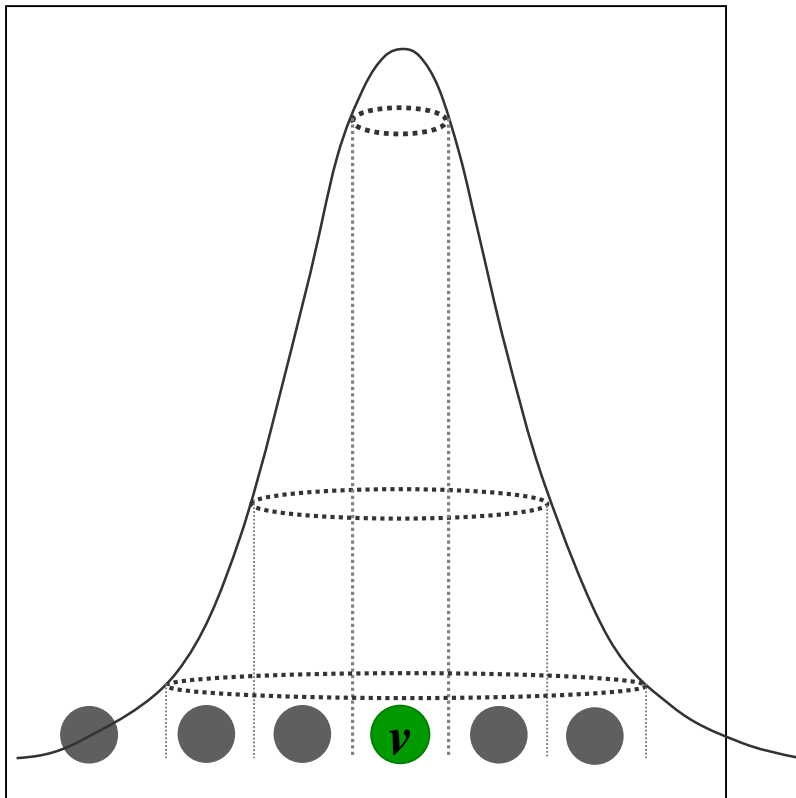
- Exemplo de função de vizinhança Gaussiana

- Os neurônios da vizinhança são atualizados de forma ponderada
- Quanto mais afastados, menor fica a taxa de aprendizado

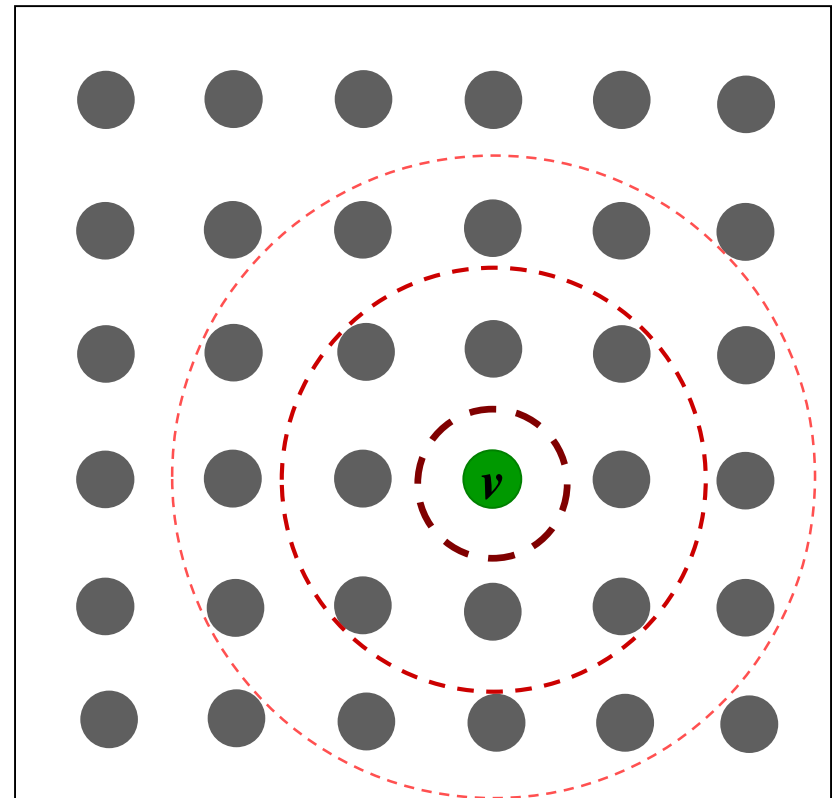


PROCESSO COOPERATIVO

- Exemplo de função de vizinhança Gaussiana



Visão Lateral



Visão Superior



PROCESSO COOPERATIVO

- Compreende a definição de uma função de vizinhança h_{vj} centrada no neurônio vencedor
- Define uma região de neurônios cooperativos, que **terão seus pesos ajustados juntamente com o neurônio vencedor**
- Há diversas formas de implementar a função de vizinhança
 - Mais simples é definir um conjunto $N_v(t)$ de níveis de vizinhança ao redor do neurônio vencedor

ADAPTAÇÃO SINÁPTICA

- Modificação dos pesos em relação à entrada, de forma iterativa
- A equação abaixo é aplicada a todos os neurônios da grade dentro da região de vizinhança h_{vj}

$$\mathbf{w}_j(t + 1) = \mathbf{w}_j(t) + \eta(t)h_{vj}(t)[\mathbf{x} - \mathbf{w}_j(t)]$$

Vetor peso atualizado

Vetor peso anterior

Taxa de aprendizagem

Vizinhança

Adaptação

ADAPTAÇÃO SINÁPTICA

- O parâmetro de aprendizagem $\eta(t)$, assim como a função de vizinhança deve decrescer com o tempo, para que as adaptações sejam cada vez mais “finas”
- Pode ser usada uma função linear decrescente, mas é recomendada uma função exponencial decrescente:

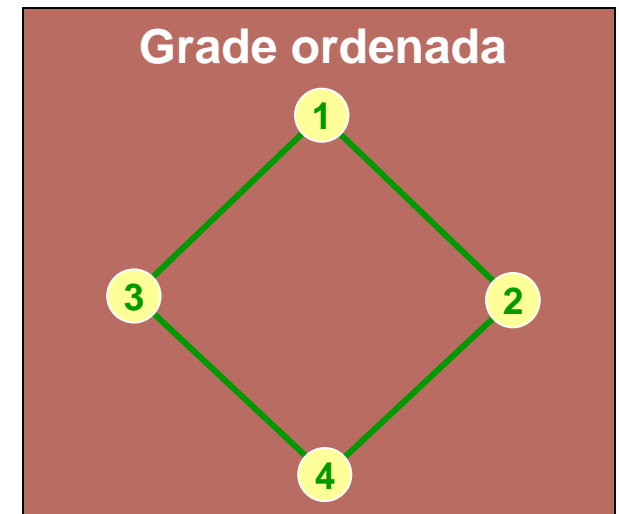
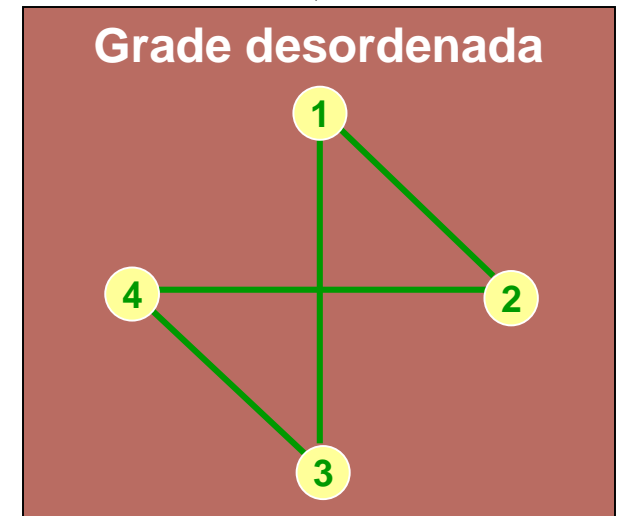
$$\eta(t) = \alpha_0 \exp\left(-\frac{t}{\tau_2}\right),$$

onde τ_2 é o número total de iterações



ADAPTAÇÃO SINÁPTICA (ORDENAÇÃO)

- Assumindo uma inicialização aleatória, é necessário duas fases de adaptação
 - Fase de Ordenação:** devido à inicialização aleatória, a grade está desordenada
 - A ordenação topológica dá-se pela movimentação da vizinhança, o que ocorre em geral nas primeiras 1000 iterações



ADAPTAÇÃO SINÁPTICA (ORDENAÇÃO)

- Para a **Fase de Ordenação**, a inicialização dos parâmetros: τ_2 , $\eta(0)$ e $\sigma(0)$ é importante e depende da aplicação

- Haykin (2000) sugere os seguintes valores iniciais:

$$\tau_2 = 1000$$

$$\eta(0) = 0,1$$

- O $\sigma(0)$, largura da função de vizinhança, depende do tamanho da rede. Para uma rede de 50 neurônios, por exemplo, uma possibilidade é:

$$\sigma(0) = 25$$



ADAPTAÇÃO SINÁPTICA (CONVERGÊNCIA)

- Após a fase de ordenação, inicia-se a adaptação aos padrões de entrada
 2. **Fase de Convergência:** ocorre uma “sintonia fina”, que pode durar muito mais iterações do que a fase de ordenação
 - Neste momento, a função de vizinhança deverá englobar apenas vizinhos próximos do neurônio vencedor
 - Pode ainda ser reduzida à apenas o neurônio vencedor
- Para a **Fase de Convergência**, a seguinte inicialização dos parâmetros: τ_2 , $\eta(0)$ e $\sigma(0)$ é recomendada por Haykin (2000):
$$\tau_2 = 500 \cdot N$$
$$\eta(0) = 0,01$$
onde N é o número de neurônios de saída,
é importante não permitir que η chegue a zero



ALGORITMO

1. Selecione a topologia da rede (defina os parâmetros)
2. Selecione a região de vizinhança inicial
3. Inicialize os pesos aleatoriamente e defina o n° de iterações
4. Para cada iteração:
 1. Selecione um padrão de entrada
 2. Encontre o neurônio vencedor (pela menor distância)
$$v(\mathbf{x}) = \arg \min_j \|\mathbf{x} - \mathbf{w}_j\|, j = 1, 2, \dots, N$$
 3. Atualize os pesos do neurônio vencedor e sua vizinhança
$$\mathbf{w}_j(t + 1) = \mathbf{w}_j(t) + \eta(t)h_{vj}(t)[\mathbf{x} - \mathbf{w}_j(t)]$$
 4. Decremente a região de vizinhança e a taxa de aprendizagem (caso desejado)
 5. Incremente o número da iteração
5. Fim

LIMITAÇÕES DA SOM

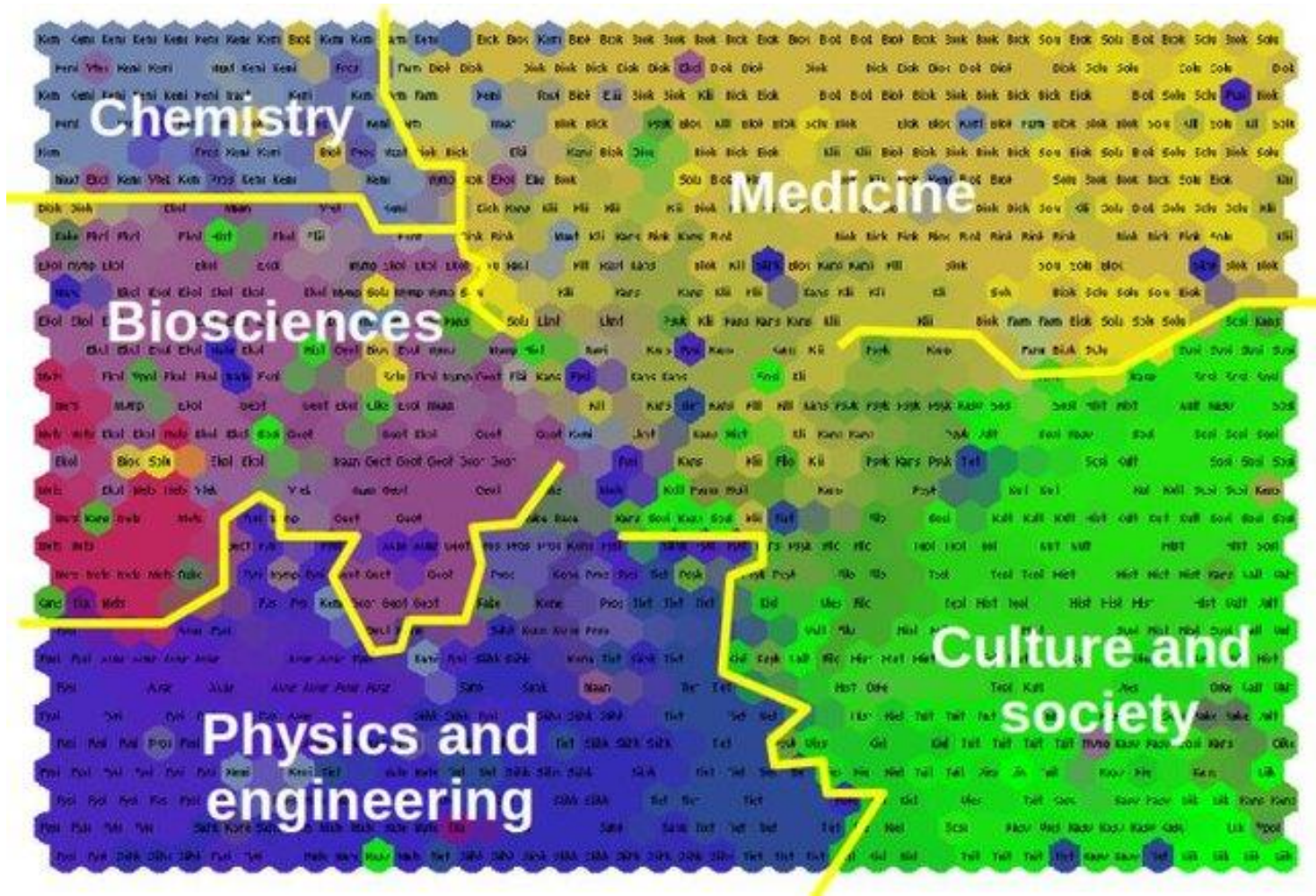
- Estrutura e dimensões decididas previamente, antes do aprendizado, sem conhecer a distribuição de probabilidade dos padrões de entrada.
- Estes parâmetros pré-definidos limitam o mapa resultante e a precisão da saída.
- Capacidade da rede é limitada pelo n° de nodos e parâmetros.
- Solução: Mapas auto-organizável sem topologia previamente definida.



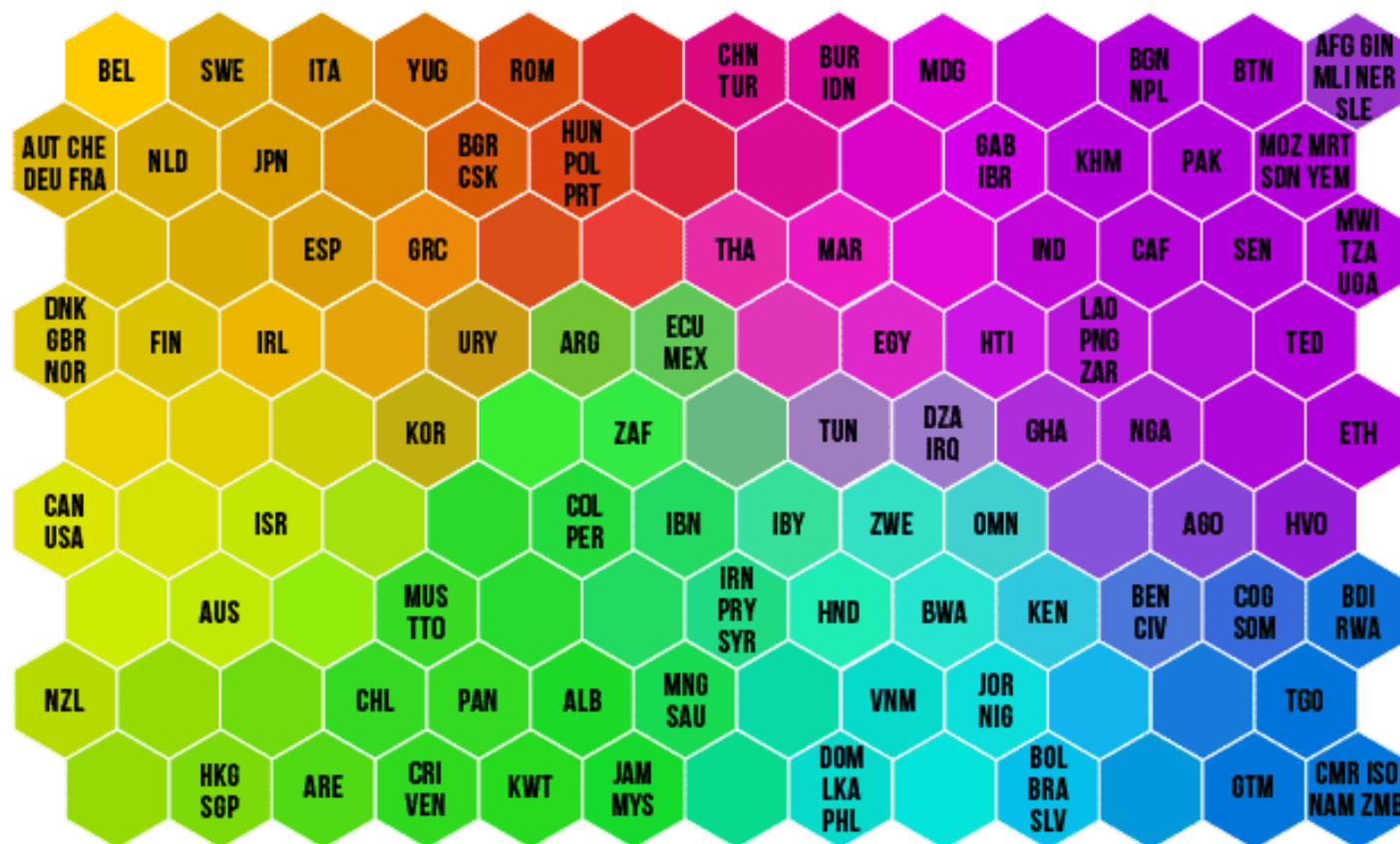
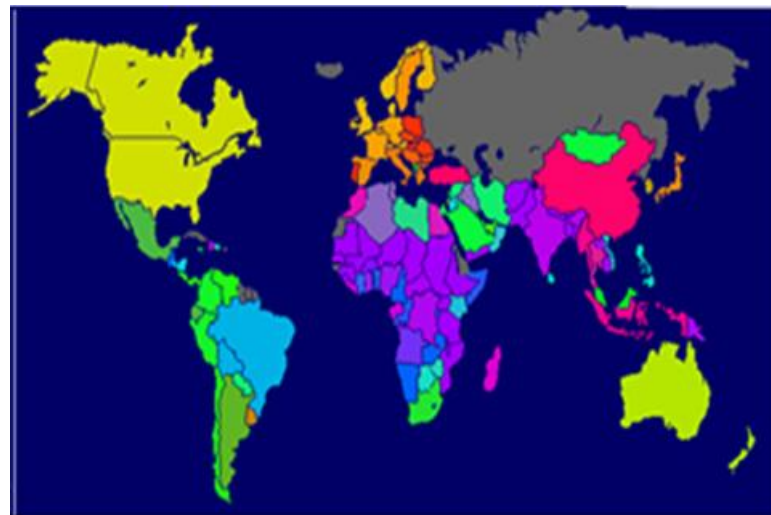
APLICAÇÕES

- Classificação de sinais de radar
- Controle de braços de robôs
- Segmentação de textura
- Modelamento do cérebro
- Tratamento de água
- Categorização automática de documentos

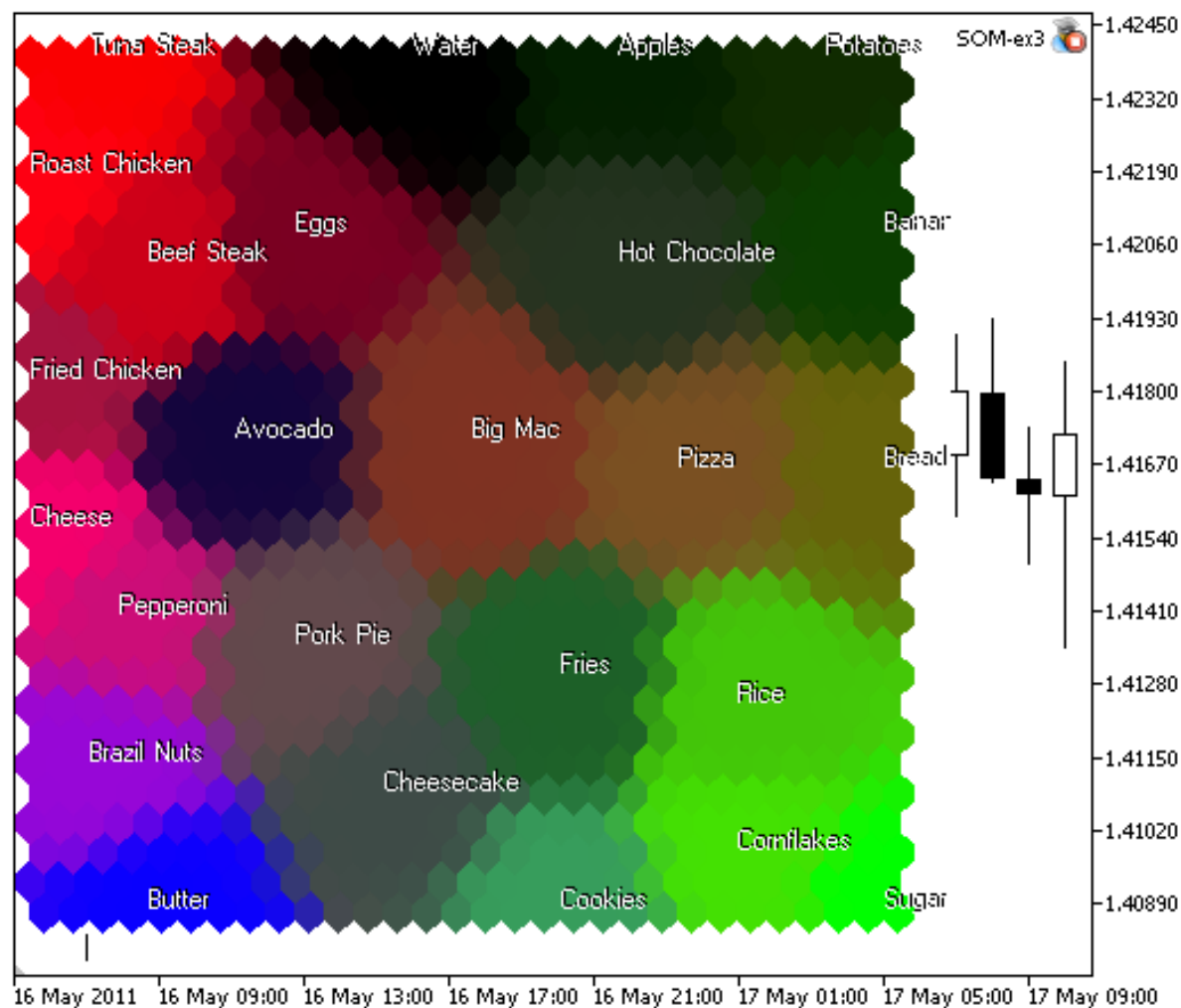
ARTIGOS



MAPA DA PROBREZA



ALIMENTOS



RECONHECIMENTO DE CARACTERES

