Redes Competitivas e de Kohonen

Redes Competitivas

- Até agora as redes estudadas possuíam muitas saídas ativas, no aprendizagem competitivo somente uma unidade de saída pode estar ativa a cada momento.
- As unidades de saída competem entre si para ser a que deve estar ativa como resposta a uma entrada determinada

Redes Competitivas

- Os neurônios de uma rede competitiva recebem idêntica informação na entrada, mais competem por ser o único a ficar ativo.
- Cada neurônio se especializa numa área diferente do espaço de entradas e suas saídas podem ser usadas para representar a estrutura do espaço de entradas.

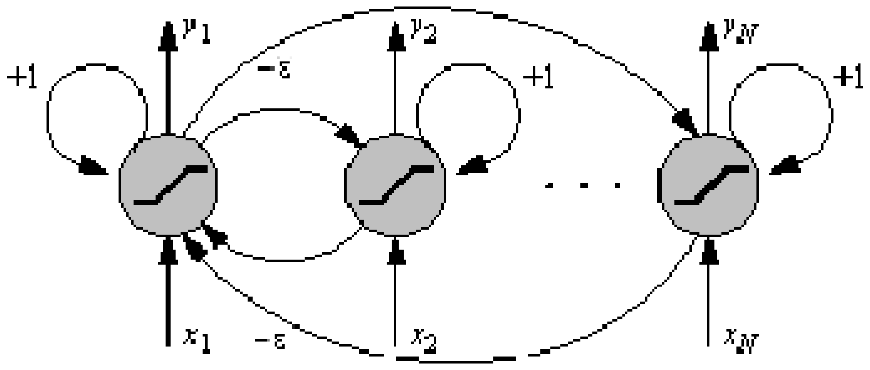
Historia

- Grossberg introduz a maior parte das idéias das redes competitivas. As suas redes trabalham em tempo continuo em termos matemáticos complexos
- Kohonen introduz uma serie de princípios facilmente implementáveis em sistemas digitais.

Competição

- A competição pode ser de dois tipos:
- Competição dura, solo um neurônio consegue ficar ativo;
- Competição branda, existe um vencedor, e seus visinhos compartem uma pequena percentagem desta ativação.

Redes Winner-take-all



$$y_k = \begin{cases} 1 & x_k \text{ es el mayor} \\ 0 & \text{en otro caso} \end{cases}$$

Redes winner-take-all

- Auto-alimentação com peso fixo
- conectada lateralmente a todas as demais por pesos negativos $0<e<1/n$
- Todas as entradas positivas
- Pendente dos neurônios igual a 1
- Condição inicial sem entrada, saída 0.

Redes Winner-take-all

- Ao apresentar-se uma entrada, a conexão lateral leva a todas os neurônios a 0. Segundo as mais pequenas se aproximam a 0, a mais grande será menos afetada. A sua vez a auto-alimentação eleva sua saída, o que reforça a tendência a 0 dos outros neurônios.

Regra de Aprendizagem

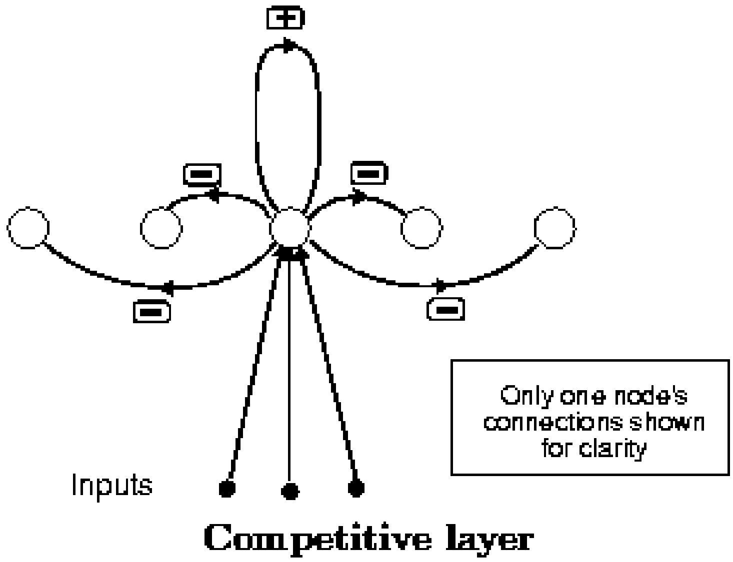
- A saída que se quer obter é

  y_i = {1, i = i* o neurônio vencedor; 0, para todos os outros}
- A regra de aprendizagem será

  $w_{i^*}(n+1) = w_{i^*}(n) + \eta(x(n) - w_{i^*}(n))$

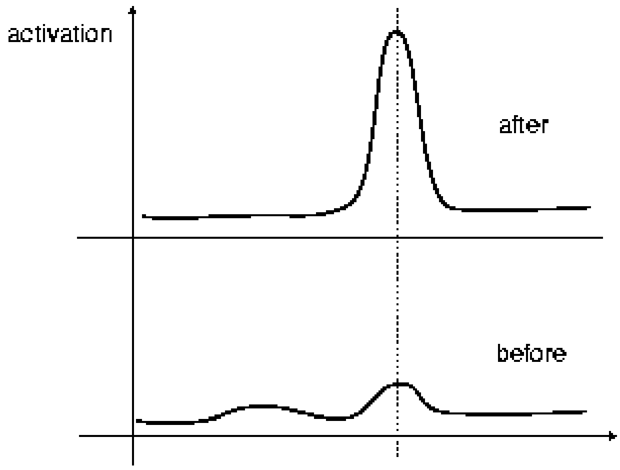
  i* é o neurônio ganhador, todos os outros pesos se mantém.

Redes de Kohonen



Aprendizado Competitivo

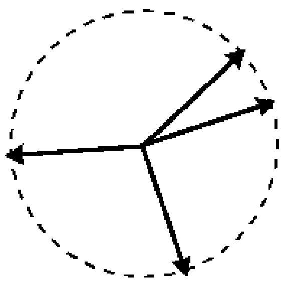
$$\frac{da}{dt} = \beta_s s + \beta_l l - \gamma a$$



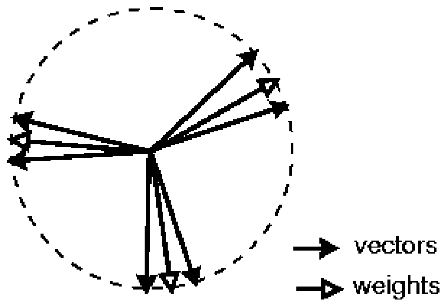
time evolution under competitive dynamics

$$\|\mathbf{x}\| = \sum_i x_i^2$$

$$\sum_i x_i = 1$$

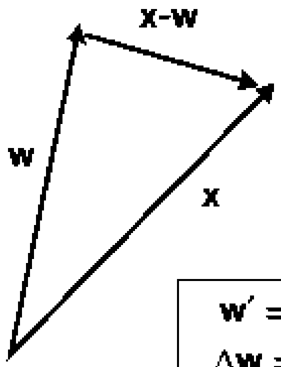


vectors on unit hypersphere



weights and vectors aligned

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



$$\mathbf{w}' = \mathbf{w} + \Delta\mathbf{w}$$
$$\Delta\mathbf{w} = \alpha(\mathbf{x} - \mathbf{w})$$

vector triangle - weights and inputs

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

Algoritmo de aprendizado

- Aplique o vetor de entradas a rede e avalie s para cada nó;
- Atualize a rede de acordo a (2) até chegar ao equilíbrio
- Treine todos os nós de acordo a (6)

Competição Branda

- A competição branda permite no somente ficar ativo neurônio ganhador, mais também os visinho
- Uma rede deste tipo pode ser criada usando alimentação lateral.
- Em este caso os pesos laterais variam segundo a distância ao neurônio vencedor.
- Os neurônios visinhos se excitam entre eles e inibem os distantes.

Regra de Aprendizado

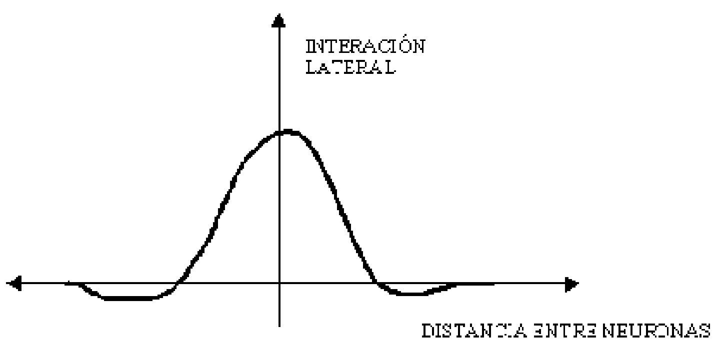
- Se atualizam os pesos para cada padrão de entrada que se apresenta
  - mais atenuando-se a atualização segundo a distância.
- Simula mapas topológicos cerebrais
- O modelo têm dois variantes:
  - LVQ (aprendizado de vetores)
  - TPM ou SOM (Mapas auto-organizáveis)

Arquitetura LVQ

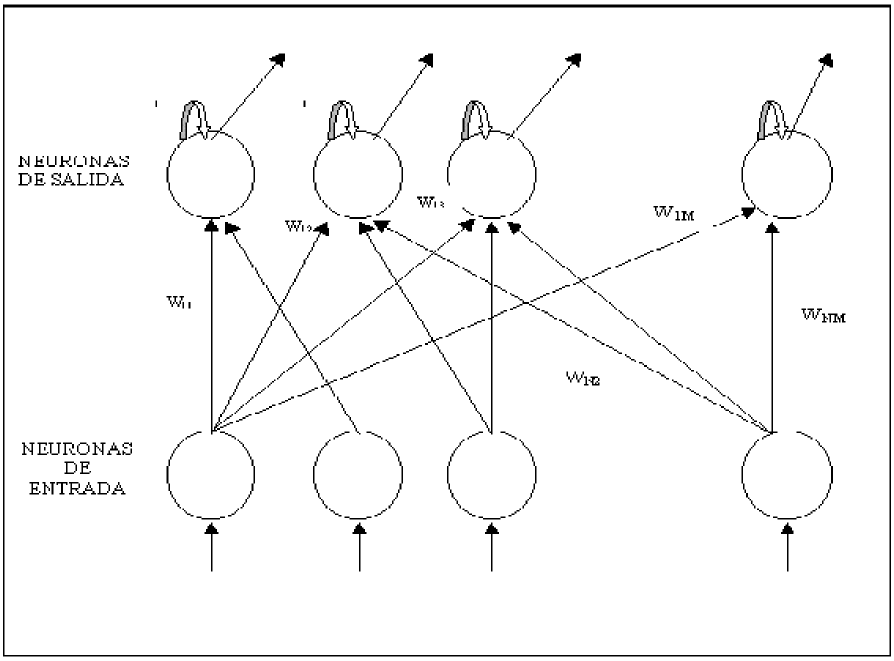
- É uma rede de duas camadas com N neurônios de entrada e M de saída.
- Cada um dos N neurônios de entrada se conecta aos M de saída por conexões para frente.
- Os neurônios de saída possuem conexões laterais
  - excitando os visinhos
  - Inibindo as mais distantes.

Influênças laterais

- São geralmente representadas por uma gaussiana



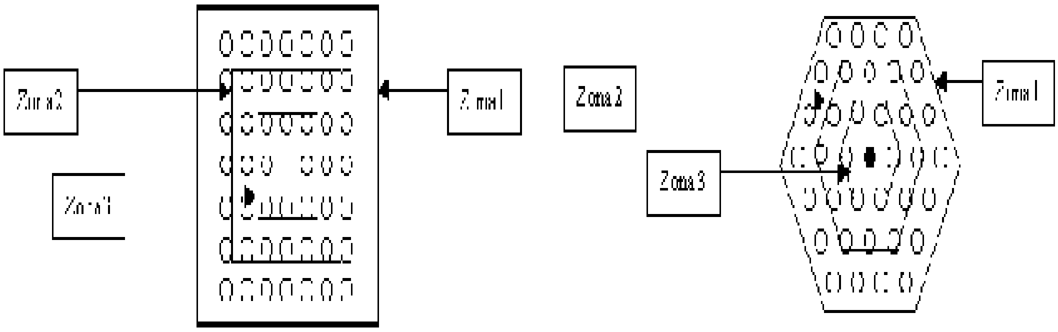
Arquitetura LVQ



Arquitetura SOM

- Trata de estabelecer uma correspondência entre os dados de entrada e um espaço bidimensional de saída.
- Dados com características comuns se ativem neurônios situados em zonas próximas da camada de saída.
- Cada neurônio da camada de entrada esta conectado a cada neurônio da camada de saída mediante pesos.
- As interações laterais seguem existindo

Arquitetura SOM



Funcionamento da rede

- Apresenta-se a la entrada uma informação em forma de vetor de N componentes (N neurônios);
- Os M neurônios que formam a camada de saída recebem a informação a través de conexões das entradas e das conexões laterais com o resto dos neurônios da camada de saída;
- A rede evoluciona até encontrar uma situação estável, na qual se ativa só um neurônio;
- Então os pesos são atualizados, para o ganhador e seus visinhos.

Atualização dos pesos

- Os pesos são atualizados segundo a distância ao neurônio ganhador

$$w_i(n+1) = w_i(n) + \Lambda_{i,i^*}(n)\,\eta(n)\,(x(n) - w_i(n))$$

$$\Lambda_{i,i^*}(n) = e^{\left(\frac{-d^2_{i,i^*}}{2\sigma^2(n)}\right)}$$

Donde d es la distancia entre las neuronas y σ disminuye con la iteración

Atualização dos pesos

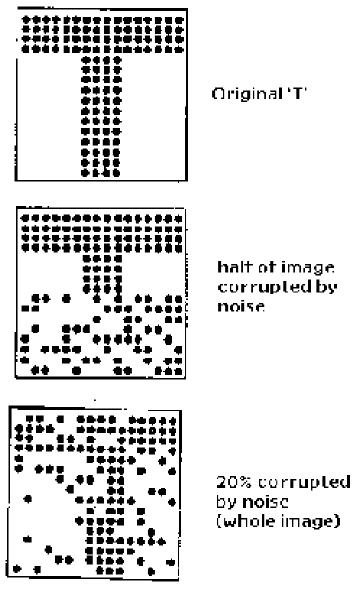
- A guassiana começa por cubrir casi todo o espaço e se reduz progressivamente até ser casi exclusivamente só o vencedor que se atualiza

Rede de Kohonen

- A rede traslada a estrutura do espaço de entradas ao espaço de saída
- Normalmente se distinguem 2 fases no aprendizado:
  – Organização topológica dos pesos, definição dos entornos
  – Convergência em que os neurônios se ajustam com precisão aos detalhes da distribuição das entradas.

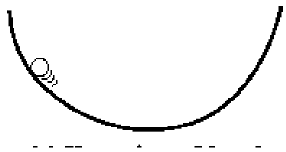
Redes Hopfield

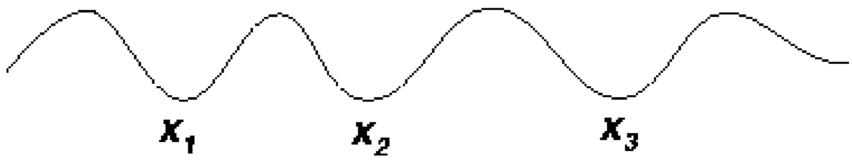
- Memórias associativas



slide of 'T's

Analogia física com memória





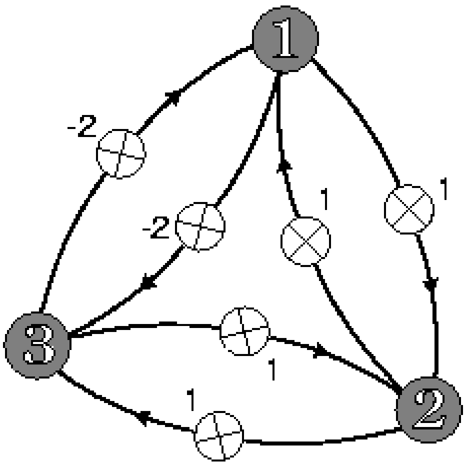
Memórias associativas

- Existem 2 formas de enfocar:
  - O sistema cai para o estado de minima energia
  - Guarda um conjunto de padrões e lembra aquele que esta mais perto do estado inicial.

Memórias Associativas

- Para construir uma rede com este comportamento
  - Descrita por um vetor de estado
  - Existe um conjunto de estados estáveis.
  - O sistema evolui no tempo de qualquer estado inicial para um dos estados estáveis

Redes de Hopfield



3 node Hopfield net

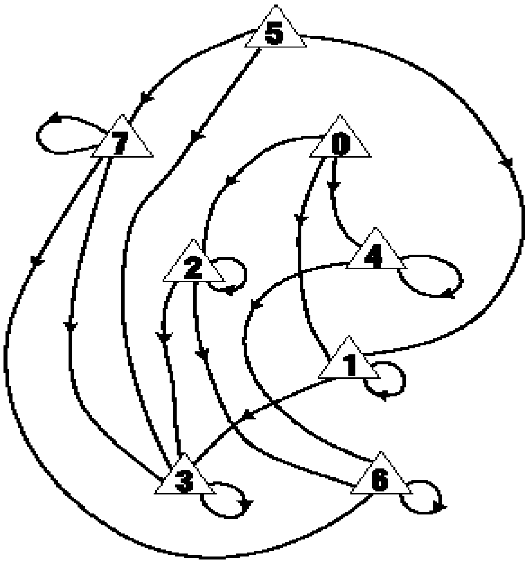
Redes de Hopfield

- Cada nó esta conectado a cada outro e as conexões são simetricas
- O estado da rede é dado pelo vetor de saída
- A rede encontra-se num estado inicial e aleatoriamente escolhemos um nó para atulizar
- Sua saída é 1 se sua ativação é maior ou igual a zero, depois escolhemos outro…

Tabela de transição de estados

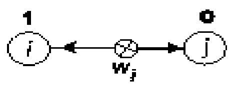
State				New state (after node has fired)		
Number	vector					
	x_1	x_2	x_3	Node 1	Node 2	Node 3
0	0	0	0	4	2	1
1	0	0	1	1	3	1
2	0	1	0	6	2	3
3	0	1	1	3	3	3
4	1	0	0	4	6	4
5	1	0	1	1	7	3
6	1	1	0	6	6	6
7	1	1	1	3	7	6

Gráfico de transição de estados



state transition diagram for 3 node net

Energia da rede



two nodes in conflict

$$\varepsilon_{ij} = -w_{ij} x_i x_j$$

x_i	x_j	ε_{ij}
0	0	0
0	1	0
1	0	0
1	1	$-w_{ij}$

Energia da Rede

$$E = \sum_{\text{pairs}} e_{ij} = -\sum_{\text{pairs}} w_{ij} x_i x_j$$

$$E = -\frac{1}{2}\sum_{i,j} w_{ij} x_i x_j$$

$$E = -\frac{1}{2}\sum_{\substack{i \neq k \\ j \neq k}} w_{ij} x_i x_j - \frac{1}{2}\sum_{i} w_{ki} x_k x_i - \frac{1}{2}\sum_{i} w_{ik} x_i x_k$$

Now, because $w_{ik} = w_{ki}$, the last two sums may be combined

$$E = -\frac{1}{2}\sum_{\substack{i \neq k \\ j \neq k}} w_{ij} x_i x_j - \sum_{i} w_{ki} x_k x_i$$

$$E = S - x_k \sum_{i} w_{ki} x_i \tag{6}$$

but the sum here is just the activation of the kth node so that

$$E = S - x_k a^k \tag{7}$$

Let the energy after k has updated be E' and the new output be x'_k. Then

$$E' = S - x'_k a^k \tag{8}$$

Denote the change in energy $E' - E$ by ΔE and the change in output $x'_k - x_k$ by Δx_k, then subtracting (7) from (8)

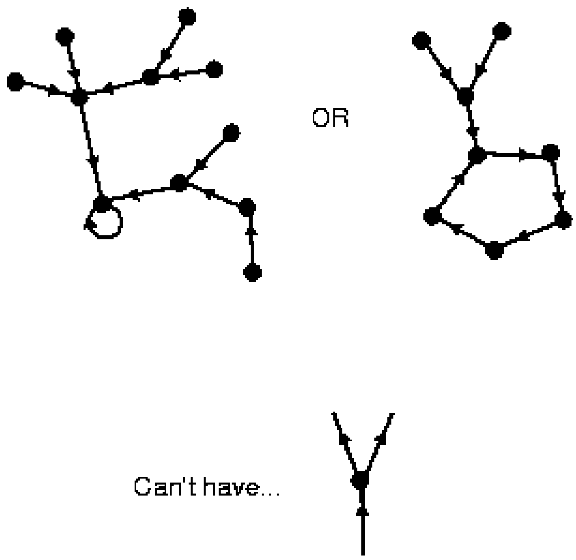
$$\Delta E = -\Delta x_k a^k \tag{9}$$

There are now two cases to consider

1. $a^k \geq 0$. Then the output goes from '0' to '1' or stays at '1'. In either case $\Delta x_k \geq 0$. Therefore $\Delta x_k a^k \geq 0$ and so, $\Delta E \leq 0$

2. $a^k < 0$. Then the output goes from '1' to '0' or stays at '0'. In either case $\Delta x_k \leq 0$. Therefore, once again $\Delta x_k a^k \geq 0$ and $\Delta E \leq 0$

Redes Síncronas



state diagrams for synchronous update

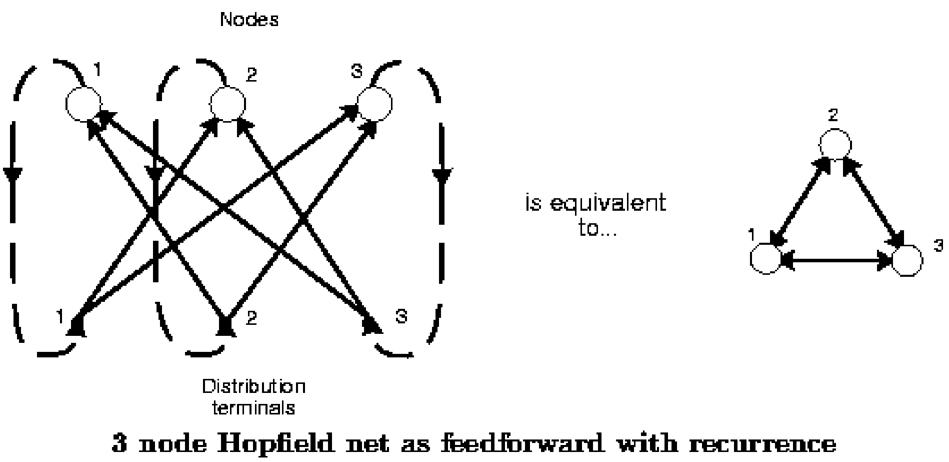
Pesos

- Obs: considere 2 nós
  - (0,0),(1,1) este comportamento estaria reforçado por pesos +
  - (0,1),(1,0) este comportamento estaria reforçado por pesos –
  - 0 - > -1; 1-> 1
- $W_{ij}= \sum_p v^p_i v^p_j$

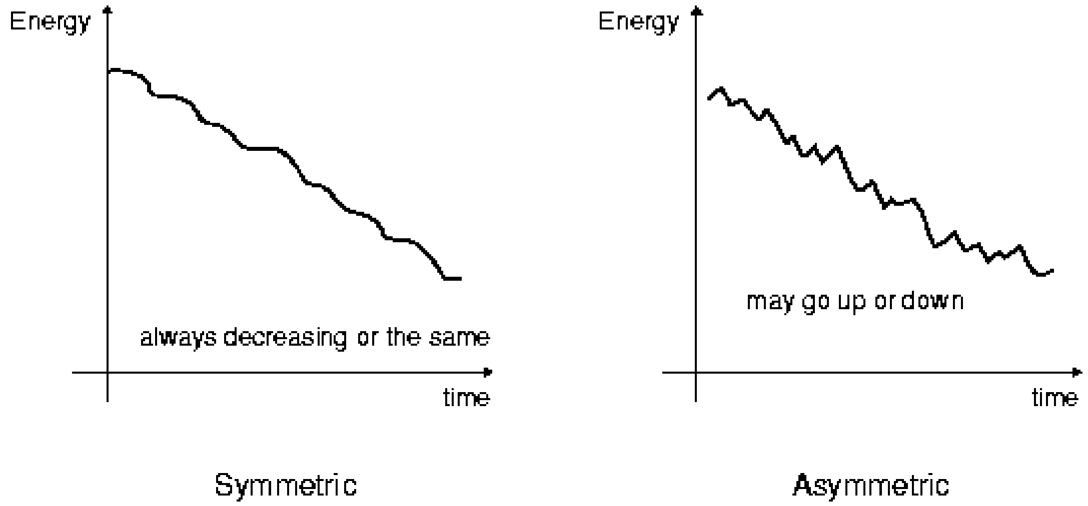
Regra Hebb

- Apresente um dos padrões;
- Se os nós possuem o mesmo valor, faça um incremento positivo senão um decremento aos pesos
- Repita o processo para todos os padrões
- $\Lambda W_{ij} = \alpha\, v^p_i v^p_j$

Usando a Regra Delta



3 node Hopfield net as feedforward with recurrence



energy v time for symmetric and asymmetric nets