



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

SCC-5809 - Capítulo 3

Aprendizado

João Luís Garcia Rosa¹

¹SCC-ICMC-USP - joaoluis@icmc.usp.br

2011

Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

Sumário

- 1 **Aprendizado**
 - **Sistemas**
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 **Supervisionado**
 - Com Professor
- 3 **Não Supervisionado**
 - Sem Professor
- 4 **Tarefas**
 - Associação
 - Reconhecimento
 - Aproximação de função

Sistemas de Aprendizado

- Define-se **aprendizado** no contexto de redes neurais como [1]:

Aprendizado é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida. O tipo de aprendizado é determinado pela forma como as mudanças nos parâmetros acontecem.

- Implica na seguinte sequência de eventos [1]:
 - 1 A RNA é estimulada pelo ambiente.
 - 2 A RNA sofre mudanças nos seus parâmetros livres como resultado desta estimulação.
 - 3 A RNA responde ao ambiente por causa das mudanças que ocorreram em sua estrutura interna.

Sistemas de Aprendizado

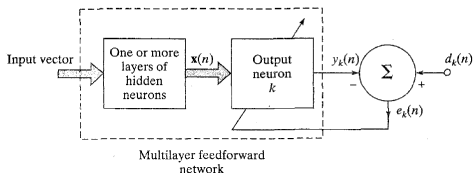
- Um conjunto de regras bem definidas para a solução de um problema de aprendizado é chamado de **algoritmo de aprendizado**.
- Há vários algoritmos de aprendizado.
- Cada um dos algoritmos oferece vantagens próprias.
- Basicamente, algoritmos de aprendizado diferem na **forma como os pesos sinápticos são ajustados**.
- Um outro fator é a maneira como a RNA se relaciona com seu ambiente.

Sumário

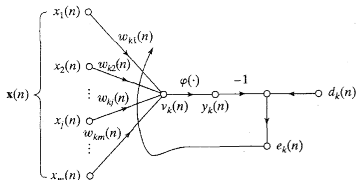
- 1 **Aprendizado**
 - Sistemas
 - **Correção erro**
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 **Supervisionado**
 - Com Professor
- 3 **Não Supervisionado**
 - Sem Professor
- 4 **Tarefas**
 - Associação
 - Reconhecimento
 - Aproximação de função

Regra de Aprendizado

- Considere uma RNA *feedforward* com um único neurônio k na camada de saída, como mostra a figura abaixo (parte a) [1]:



(a) Block diagram of a neural network, highlighting the only neuron in the output layer



(b) Signal-flow graph of output neuron

Regra de Aprendizado

- O neurônio k recebe um *vetor de sinal* $\mathbf{x}(n)$ produzido por uma ou mais camadas de neurônios escondidos, que recebem por sua vez, um vetor de entrada (estímulo) aplicado nos nós fonte (isto é, camada de entrada) da RNA.
- n denota o tempo discreto, ou seja, o passo de tempo de um processo iterativo envolvido no ajuste dos pesos sinápticos do neurônio k .
- O *sinal de saída* $y_k(n)$ do neurônio k representa a única saída da RNA e é comparado a uma *resposta desejada* $d_k(n)$.
- Como consequência, um **sinal de erro** $e_k(n)$ é produzido:

$$e_k(n) = d_k(n) - y_k(n) \quad (1)$$

Regra de Aprendizado

- O sinal de erro $e_k(n)$ atua em um *mecanismo de controle* para aplicar uma sequência de ajustes corretivos aos pesos sinápticos do neurônio k .
- Esses ajustes fazem o sinal de saída $y_k(n)$ se aproximar da saída desejada $d_k(n)$ passo a passo, através da minimização de uma *função de custo* $\xi(n)$, definida em termos do sinal de erro $e_k(n)$:

$$\xi(n) = \frac{1}{2} e_k^2(n) \quad (2)$$

- $\xi(n)$ é o *valor instantâneo da energia do erro*.
- O ajuste passo a passo dos pesos sinápticos continuam até que o sistema alcance um *estado estável*, ou seja, quando os pesos sinápticos estiverem estabilizados.
- Então o processo de aprendizado termina.

Regra delta

- Esse processo é chamado de **aprendizado por correção de erro**.
- Trata-se da **regra delta** ou *regra de Widrow-Hoff*.
- Seja $w_{kj}(n)$ o valor do peso sináptico do neurônio k excitado pelo elemento $x_j(n)$ do vetor de sinal $\mathbf{x}(n)$ no tempo n .
- De acordo com a regra delta, o ajuste $\Delta w_{kj}(n)$ aplicado ao peso sináptico w_{kj} no tempo n é definido por:

$$\Delta w_{kj}(n) = \eta e_k(n) x_j(n) \quad (3)$$

onde η é uma constante positiva que determina a *taxa de aprendizado* de um passo a outro no processo de aprendizado.

Regra delta

- O valor atualizado do peso sináptico w_{kj} :

$$w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n) \quad (4)$$

$$w_{kj}(n) = z^{-1}[w_{kj}(n+1)] \quad (5)$$

onde z^{-1} representa o *operador de atraso de unidade* ou *elemento de armazenamento*.

- A parte b da figura 2 mostra uma representação de grafo de fluxo de sinal do processo de aprendizado por correção de erro, focando na atividade ao redor do neurônio k .
- O sinal de entrada x_j e o campo local induzido v_k do neurônio k referem-se aos sinais *pré-sináptico* e *pós-sináptico* respectivamente.
- A escolha adequada do parâmetro η é muito importante para assegurar a **estabilidade da convergência** do processo de aprendizado iterativo.

Sumário

- 1 **Aprendizado**
 - Sistemas
 - Correção erro
 - **Hebbiano**
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

A Organização do Comportamento

- O **postulado do aprendizado de Hebb** é a mais antiga e a mais famosa regra de aprendizado [2]:

“Quando um axônio da célula A é próximo o suficiente de excitar uma célula B e repetidamente ou persistentemente toma parte em dispará-la, algum processo de crescimento ou mudança metabólica acontece em uma ou ambas as células tal que a eficiência de A, como uma das células que disparam B, é aumentada”.

- Hebb propôs esta mudança como base para o aprendizado associativo (no nível celular), que resultaria em uma modificação duradoura no padrão de atividade da “montagem de células nervosas” espacialmente distribuída.

A Organização do Comportamento

- Pode-se re-escrever a regra:
 - 1 Se os dois neurônios de uma sinapse (conexão) são ativados simultaneamente (isto é, sincronamente), então a força desta sinapse é aumentada.
 - 2 Se os dois neurônios de uma sinapse são ativados assincronamente, então a sinapse é enfraquecida ou eliminada.
- Tal sinapse é chamada de **sinapse Hebbiana**: usa mecanismo dependente de tempo, local e fortemente interativo para aumentar a eficiência sináptica como uma função da correlação entre as atividades pré-sináptica e pós-sináptica.

A Organização do Comportamento

- 4 mecanismos (propriedades) que caracterizam uma sinapse Hebbiana:
 - 1 *Dependente de tempo*: a modificação na sinapse depende do tempo exato de ocorrência dos sinais pré-sináptico e pós-sináptico.
 - 2 *Local*: a sinapse é o local de transmissão onde sinais de informação (representando atividade em andamento nas unidades pré e pós-sinápticas) estão em uma contiguidade espaço-temporal. Esta informação disponível localmente é usada por uma sinapse Hebbiana para produzir uma modificação sináptica local que é específica da entrada.
 - 3 *Interativo*: A mudança na sinapse depende dos sinais de ambos os lados da sinapse.
 - 4 *Conjuncional ou correlacional*: condição para uma mudança na eficiência da sinapse é a conjunção dos sinais pré e pós-sinápticos. A co-ocorrência de ambos os sinais dentro de um intervalo de tempo pequeno, é suficiente para produzir a modificação sináptica.

Fortalecimento e enfraquecimento sináptico

- Modificação sináptica:
 - 1 *Hebbiana*: aumenta sua força quando sinais pré e pós-sinápticos são correlacionados positivamente, e diminui sua força quando estes sinais ou não são correlacionados ou são correlacionados negativamente.
 - 2 *Anti-Hebbiana*: o oposto. Ainda Hebbiana, pois a modificação da eficiência sináptica fundamenta-se num mecanismo que é dependente de tempo, altamente local e fortemente interativo.
 - 3 *Não-Hebbiana*: não envolve o mecanismo Hebbiano.

Modelos matemáticos das modificações Hebbianas

- Considere um peso sináptico w_{kj} do neurônio k com sinais pré e pós-sinápticos x_j e y_k .
- O ajuste aplicado ao peso sináptico w_{kj} no passo de tempo n é:

$$\Delta w_{kj}(n) = F(y_k(n), x_j(n)) \quad (6)$$

onde F é uma função de ambos os sinais pós e pré-sinápticos.

- Os sinais $x_j(n)$ e $y_k(n)$ não têm dimensão.
- **Hipótese de Hebb:**

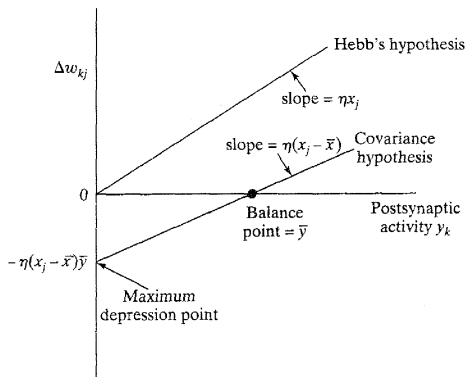
$$\Delta w_{kj}(n) = \eta y_k(n) x_j(n) \quad (7)$$

onde η é uma constante positiva que determina a **taxa de aprendizado**.

Modelos matemáticos das modificações Hebbianas

- A equação 7 enfatiza a natureza correlacional de uma sinapse Hebbiana.
- A figura 8 do próximo slide (parte superior) mostra uma representação gráfica da equação 7 com a mudança Δw_{kj} versus o sinal de saída y_k .
- Desta representação, vê-se que a aplicação repetida do sinal de entrada x_j leva a um aumento em y_k e portanto a um *crescimento exponencial* que finalmente leva a conexão sináptica à saturação.
- Neste momento, nenhuma informação será armazenada na sinapse e a seletividade estará perdida.

Modelos matemáticos das modificações Hebbianas [1]



Hipótese da Covariância

- Uma forma de superar a limitação da Hipótese de Hebb é usar a **hipótese da covariância**.
- Nesta hipótese, os sinais pré e pós-sinápticos da equação 7 são substituídos pela mudança dos sinais pré e pós-sinápticos dos seus valores médios respectivos sobre um certo intervalo de tempo.
- Seja \bar{x} e \bar{y} os *valores médios de tempo* de x_j e y_k .
- De acordo com a hipótese da covariância, o ajuste aplicado ao peso sináptico w_{kj} é definido como:

$$\Delta w_{kj} = \eta(x_j - \bar{x})(y_k - \bar{y}) \quad (8)$$

onde η é o parâmetro taxa de aprendizado.

Hipótese da Covariância

- Os valores médios \bar{x} e \bar{y} constituem os limiares pré e pós-sináptico, que determinam o sinal da modificação sináptica.
- A hipótese da covariância permite:
 - Convergência para um estado não trivial, que é alcançado quando $x_k = \bar{x}$ ou $y_j = \bar{y}$.
 - Previsão da **potenciação** e da **depressão** sinápticas.
- A figura 8 ilustra a diferença entre a hipótese de Hebb e a hipótese da covariância.
- Em ambos os casos, a dependência de Δw_{kj} em y_k é linear; entretanto a interceptação com o eixo y_k na hipótese de Hebb é na origem enquanto que na hipótese da covariância é em $y_k = \bar{y}$.

Hipótese da Covariância

- A partir da equação 8:
 - 1 O peso sináptico w_{kj} é **potencializado** se houver níveis suficientes de atividades pré e pós-sinápticas, ou seja, as condições $x_j > \bar{x}$ e $y_k > \bar{y}$ são ambas satisfeitas.
 - 2 O peso sináptico w_{kj} é **deprimido** se houver
 - uma ativação pré-sináptica (isto é, $x_j > \bar{x}$) na ausência de ativação pós-sináptica suficiente (isto é, $y_k < \bar{y}$) ou
 - uma ativação pós-sináptica (isto é, $y_k > \bar{y}$) na ausência de ativação pré-sináptica suficiente (isto é, $x_j < \bar{x}$).
- Este comportamento pode ser visto como uma forma de competição temporal entre os padrões de entrada.
- Há uma forte **evidência fisiológica** para o aprendizado Hebbiano no hipocampo (aprendizado e memória).

Sumário

- 1 **Aprendizado**
 - Sistemas
 - Correção erro
 - Hebbiano
 - **Competitivo**
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

Aprendizado Competitivo

- No **aprendizado competitivo** os neurônios de saída de uma RNA competem entre si para tornar-se ativos.
- Enquanto que na RNA baseada no aprendizado Hebbiano vários neurônios de saída podem estar simultaneamente ativos, no aprendizado competitivo apenas um único neurônio de saída está ativo em um determinado tempo.
- Isso torna o aprendizado competitivo útil para **descobrir traços ou características estatisticamente salientes** que podem ser usados para **classificar** um conjunto de padrões de entrada.

Aprendizado Competitivo

- Há três elementos básicos a uma regra do aprendizado competitivo:
 - 1 Um conjunto de neurônios, onde os elementos são todos os mesmos, exceto por alguns pesos sinápticos distribuídos aleatoriamente e que portanto *respondem diferentemente* a um dado conjunto de padrões de entrada.
 - 2 Um *limite* imposto na “força” de cada neurônio.
 - 3 Um mecanismo que permite aos neurônios *competirem* pelo direito de responder a um dado subconjunto de entradas, tal que apenas um neurônio de saída, ou apenas um neurônio por grupo, esteja ativo em um determinado instante. O neurônio que vence a competição é chamado de **neurônio vencedor-leva-tudo**.
- Os neurônios individuais da rede aprendem a especializar em conjuntos de padrões similares; assim tornam-se **detetores de características** para diferentes classes de padrões de entrada.

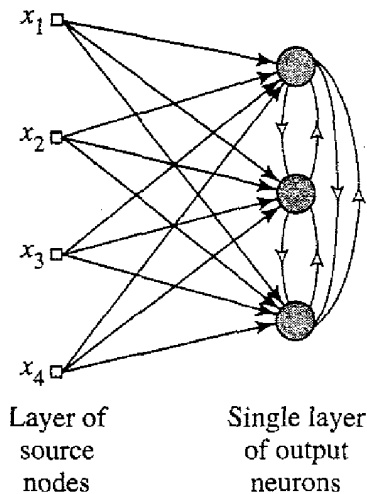
Aprendizado Competitivo

- Na forma mais simples do aprendizado competitivo, a RNA tem uma única camada de neurônios de saída.
- A rede pode incluir conexões de retro-alimentação entre os neurônios, como indicado na figura 5, realizando *inibição lateral*.
- As conexões sinápticas com alimentação a frente são todas *excitatórias*.
- Para um neurônio k ser o vencedor, seu campo local induzido v_k para um padrão de entrada \mathbf{x} deve ser o maior entre todos os neurônios da rede:

$$y_k = \begin{cases} 1, & \text{se } v_k > v_j \quad \text{para todo } j, j \neq k \\ 0, & \text{caso contrario} \end{cases}$$

onde v_k representa a ação combinada de todas as entradas *forward* e *feedback* ao neurônio k .

Aprendizado Competitivo [1]



Aprendizado Competitivo

- Seja w_{kj} o peso sináptico que conecta o nó de entrada j ao neurônio k .
- Suponha que a cada neurônio é alocada uma quantidade *fixa* de peso sináptico positivo, distribuído entre os nós de entrada:

$$\sum_j w_{kj} = 1, \text{ para todo } k \quad (9)$$

Aprendizado Competitivo

- Um neurônio aprende deslocando pesos sinápticos dos nós de entrada inativos para ativos.
- Se um neurônio ganha a competição, cada nó de entrada deste neurônio cede alguma proporção de seu peso sináptico e o peso cedido é então distribuído igualmente entre os nós de entrada ativos.

$$\Delta w_{kj} = \begin{cases} \eta(x_j - w_{kj}), & \text{se o neurônio } k \text{ vence a competição} \\ 0, & \text{se o neurônio } k \text{ perde a competição} \end{cases}$$

- Esta regra move o vetor de peso sináptico \mathbf{w}_k do neurônio vencedor k em direção ao padrão de entrada \mathbf{x} .

Sumário

- 1 **Aprendizado**
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - **Boltzmann**
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

Aprendizado de Boltzmann

- Homenagem a Ludwig Boltzmann: **algoritmo de aprendizado estocástico** derivado de ideias da mecânica estatística.
- Uma RNA projetada na base da regra de aprendizado de Boltzmann é chamada de *máquina de Boltzmann*.
- Na máquina de Boltzmann, os neurônios constituem uma rede recorrente e eles operam num modo binário (*on* = +1 e *off* = -1).
- Esta máquina é caracterizada por uma *função energia* E :

$$E = -\frac{1}{2} \sum_j \sum_k w_{kj} x_k x_j, j \neq k \quad (10)$$

Aprendizado de Boltzmann

- A máquina opera escolhendo um neurônio aleatoriamente, por exemplo neurônio k , em algum passo do aprendizado e move lentamente o estado do neurônio k do estado x_k para o estado $-x_k$ em alguma temperatura T com probabilidade:

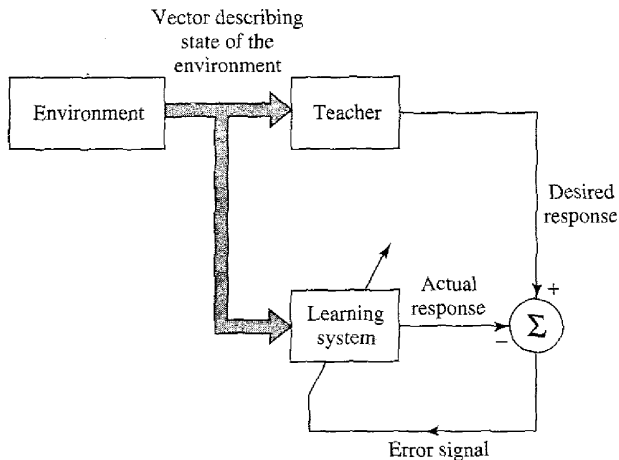
$$P(x_k \rightarrow -x_k) = \frac{1}{1 + \exp(-\Delta E_k / T)} \quad (11)$$

- Se a regra é aplicada repetidamente, a máquina atingirá o **equilíbrio térmico**.

Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 **Supervisionado**
 - **Com Professor**
- 3 Não Supervisionado
 - Sem Professor
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

Aprendizado Supervisionado [1]



Aprendizado Supervisionado

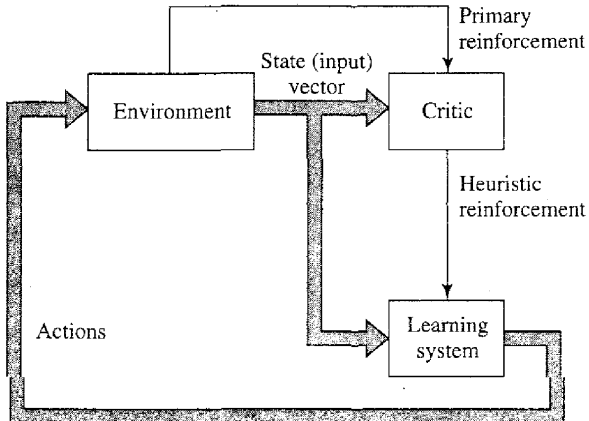
- O **professor** está apto a prover a resposta desejada à RNA para determinado vetor de treinamento.
- Os parâmetros da rede são **ajustados** sob a influência combinada do **vetor de treinamento** e do **signal de erro**.
- O **signal de erro** é definido como a diferença entre a resposta desejada e a resposta real da rede.
- O ajuste é feito iterativamente de uma forma passo a passo até que finalmente faz a RNA *emular* o professor.
- *Off-line*: uma vez que a resposta desejada é obtida, o projeto é congelado. A rede opera no modo *estático*.
- *On-line*: o aprendizado é executado em tempo real. A rede neural é *dinâmica* [3].

Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 **Não Supervisionado**
 - **Sem Professor**
- 4 Tarefas
 - Associação
 - Reconhecimento
 - Aproximação de função

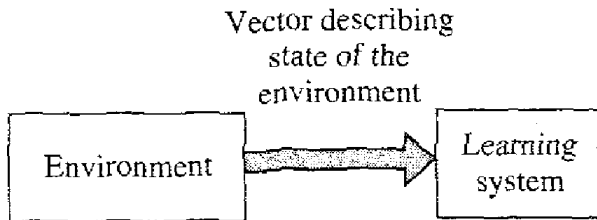
Aprendizado por Reforço

- No *aprendizado por reforço*, o aprendizado de um mapeamento entrada-saída se dá através de **interação contínua com o ambiente** [1].



Aprendizado não supervisionado

- Não há professor externo [1]:

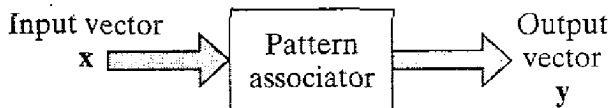


Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 **Tarefas**
 - **Associação**
 - Reconhecimento
 - Aproximação de função

Memória associativa

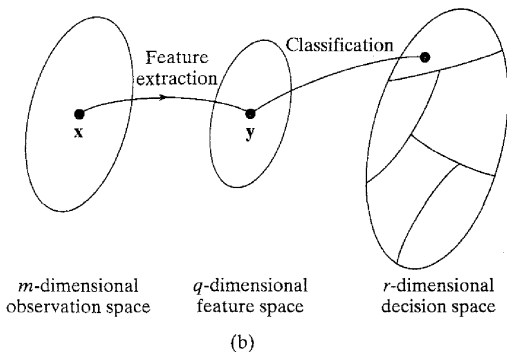
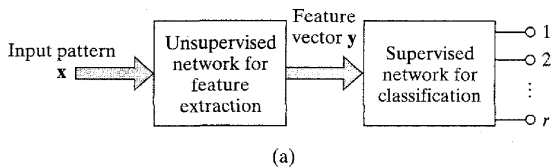
- Uma **memória associativa** é uma memória distribuída que aprende por *associação*:
 - *Autoassociação*: a RNA *armazena* um conjunto de padrões (vetores) apresentados repetidamente. A seguir, a rede recebe uma descrição parcial ou distorcida (ruído) e a tarefa é *recuperar* o padrão original: não-supervisionada.
 - *Heteroassociação*: um conjunto de padrões de entrada é *emparelhado* com um outro conjunto de padrões de saída: supervisionada [1].



Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 **Tarefas**
 - Associação
 - **Reconhecimento**
 - Aproximação de função

Reconhecimento de padrões [1]



Sumário

- 1 Aprendizado
 - Sistemas
 - Correção erro
 - Hebbiano
 - Competitivo
 - Boltzmann
- 2 Supervisionado
 - Com Professor
- 3 Não Supervisionado
 - Sem Professor
- 4 **Tarefas**
 - Associação
 - Reconhecimento
 - **Aproximação de função**

Aproximação de função

- Considere um mapeamento entrada-saída não linear

$$\mathbf{d} = \mathbf{f}(\mathbf{x}) \quad (12)$$

onde o vetor \mathbf{x} é a entrada e o vetor \mathbf{d} é a saída.

- A tarefa é projetar uma RNA que **aproxime** a função desconhecida \mathbf{f} tal que uma função \mathbf{F} , que descreve o mapeamento entrada-saída realmente realizado pela rede seja próximo de \mathbf{f} no sentido Euclidiano:

$$\|\mathbf{F}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\| < \epsilon, \text{ para todo } \mathbf{x} \quad (13)$$

onde ϵ é um número pequeno positivo.

Bibliografia I

- [1] S. Haykin
Neural networks - a comprehensive foundation.
2nd. edition. Prentice Hall, 1999.
- [2] D. O. Hebb
The Organization of Behavior: A Neuropsychological Theory.
Wiley, 1949.
- [3] R. A. F. Romero
SCC-5809 Redes Neurais.
Slides e listas de exercícios. Programa de Pós-Graduação em Ciência de Computação e Matemática Computacional. ICMC/USP, 2010.