

-
-
-
-
-

Redes de Função de Base Radial

Aluizio Fausto Ribeiro Araújo
Universidade Federal de Pernambuco
Centro de Informática



Conteúdo

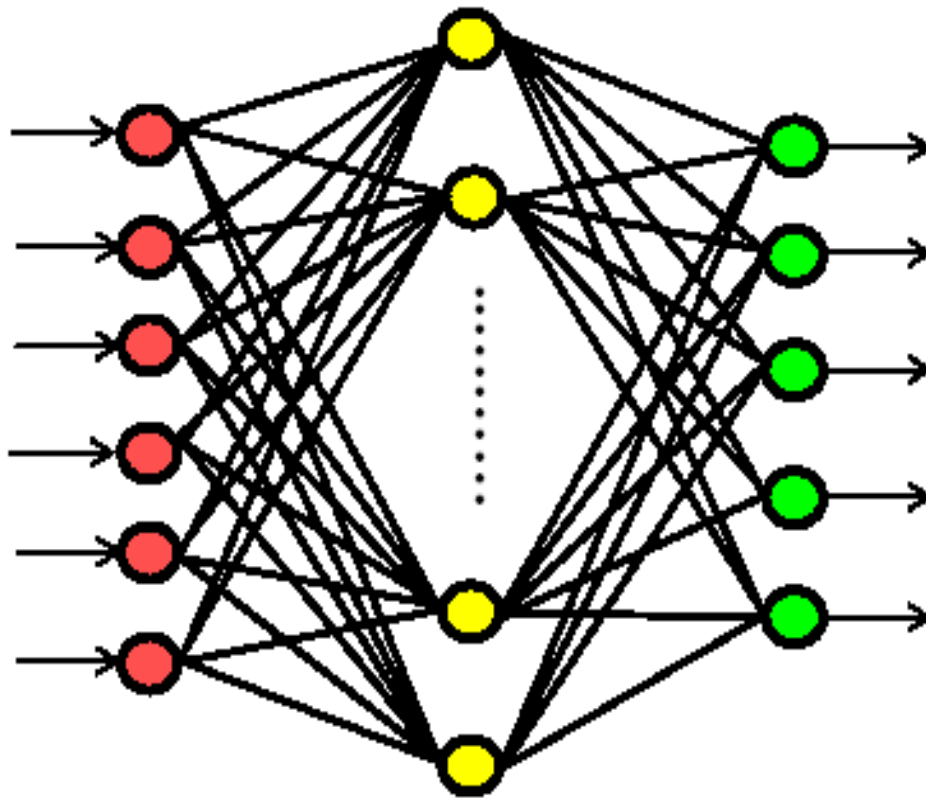
- Introdução
- Histórico
- Teoria que Fundamenta Redes RBF
- Redes de Função de Base Radial
- Aprendizagem em Redes RBF
- Desempenho das Redes RBF
- Comparando MLP-BP e RBF
- Exemplo

Introdução

- Este capítulo trata de redes neurais com múltiplas camadas que não são treinadas por retropropagação e que não têm unidades de processamento com função de ativação do tipo sigmoidal;
- Estas redes utilizam unidades com campos receptivos locais (*local receptive fields*) que recebem diretamente estímulos de entrada e são preparadas para responder apenas a parte destas entradas;
- Esta abordagem emprega, na maioria dos casos, treinamento supervisionado e não-supervisionado;
- As redes são muito usadas como interpoladores ou aproximadores e em tarefas de classificação:
 - A rede RBF tem sua origem na realização de interpolação de um conjunto de pontos de dados em um espaço multidimensional.

Introdução

- Arquitetura típica de uma rede RBF



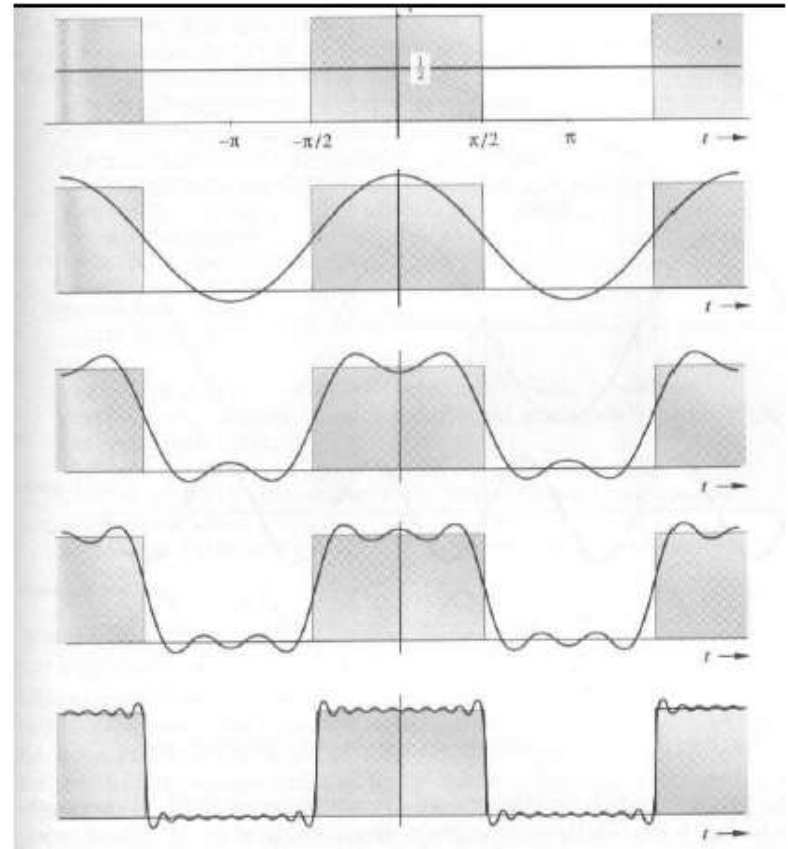
- Camada de entrada: Realiza propagação da entrada;
- Camada escondida: Realiza transformação não-linear, considerando campos receptivos locais;
- Camada de saída: Realiza combinação linear

Introdução

- Em uma rede MLP-BP, a regra de ativação de cada unidade da camada escondida é definida pelo produto interno do vetor de entrada pelo vetor de pesos;
- Em redes RBF, a regra de ativação das unidades da camada escondida é definida por uma função de base radial para a distância entre o vetor de entrada e um vetor de protótipo;
- Redes RBF podem ser caracterizadas por:
 - Definição associada com disciplinas tais como aproximação de funções, teoria da regularização, estimação de densidade e interpolação na presença de ruídos (Bishop, 1995);
 - Permitirem interpretação clara das representações internas da camada escondida;
 - Treinamento, em geral, muito mais rápido que as MLP-BP.

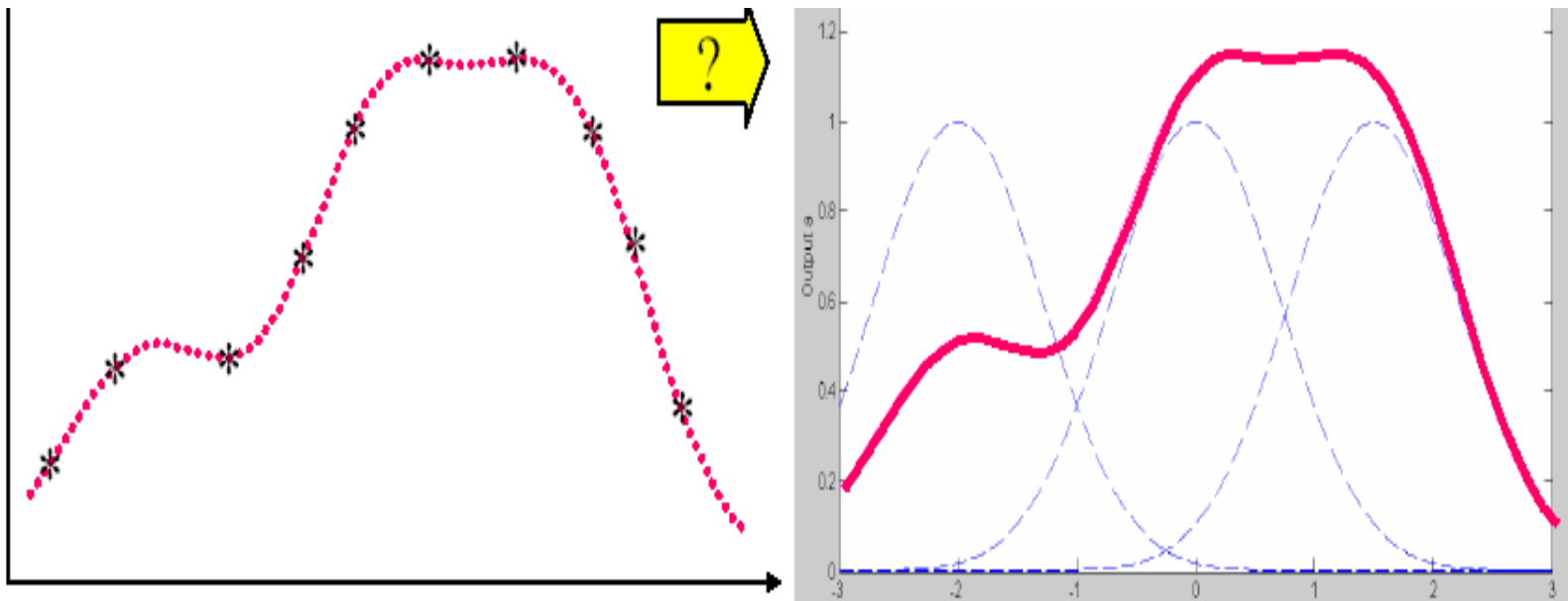
Introdução

- Construtores de funções complexas a partir de funções simples:
 - Série de Fourier
 - Transformada de Fourier
 - Transformada Wavelet
 - Redes RBF



Introdução

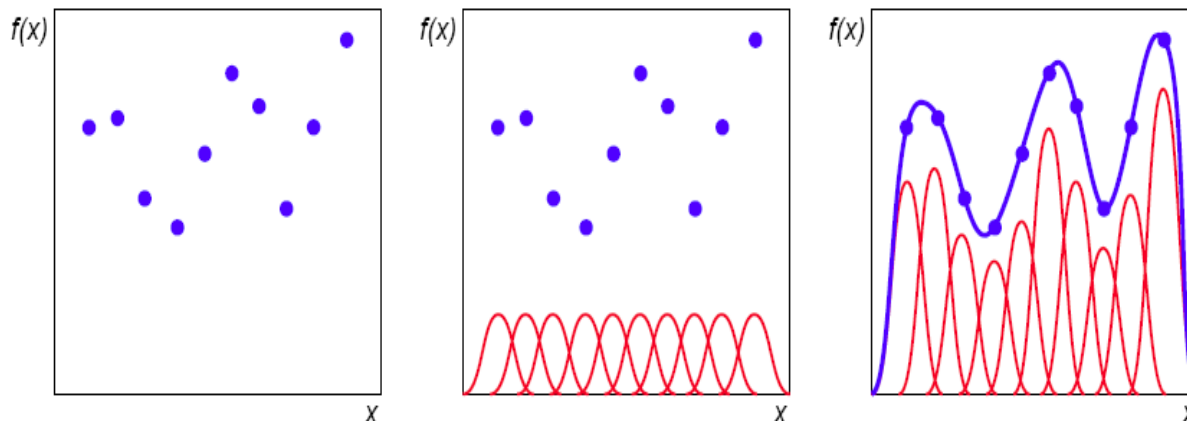
Redes RBF aproximam funções empregando funções de base radial como blocos construtores.



Introdução

- Redes RBF têm suas origens em técnicas para realizar interpolação exata de funções (Bishop, 1995).
- Abordagem RBF (Powell, 1987):
 - Empregue um conjunto de N funções de base, não-lineares, para calcular a função:

$$h(\mathbf{X}) = \sum_i w_i \Phi(\|\mathbf{X} - \mathbf{\Lambda}_i\|).$$



Histórico

- Esta abordagem é inspirada na propriedade de alguns neurônios biológicos chamada de resposta localmente sintonizada (*locally tuned response*). Tais células nervosas respondem seletivamente a um intervalo finito do espaço de sinais de entrada.
- Aprendizagem envolve encontrar uma superfície em um espaço de dimensão qualquer que produza o melhor ajuste (represente da melhor maneira) os dados de treinamento.

Histórico

- O primeiro trabalho lidando com funções de base radial foi introduzido por Medgassy (1961) cujos resultados foram posteriormente usados para interpolação (Micchelli, 1986; Powell, 1987), para estimação de densidade (Parzen, 1962; Duda e Hart, 1973; Specht, 1990) e para aproximação de funções multivariadas suaves (*smooth multivariate functions*) (Poggio and Girosi, 1989).

Histórico

- Atualmente, os modelos de RBF se diferenciam dos primeiros pois são de natureza adaptativa que permite a utilização, em muitas situações, de um número relativamente menor de unidades de processamento localmente sintonizadas.
- Redes RBF foram independentemente propostas por Broomhead e Lowe (1988), Lee e Kil (1988), Niranjan e Fallside (1988) e Moody e Darken (1989a, 1989b). Outros esquemas similares foram introduzidos por Hanson e Burr (1987), Lapedes e Faber (1987), Casdagli (1989), Poggio e Girosi (1990b), entre outros.

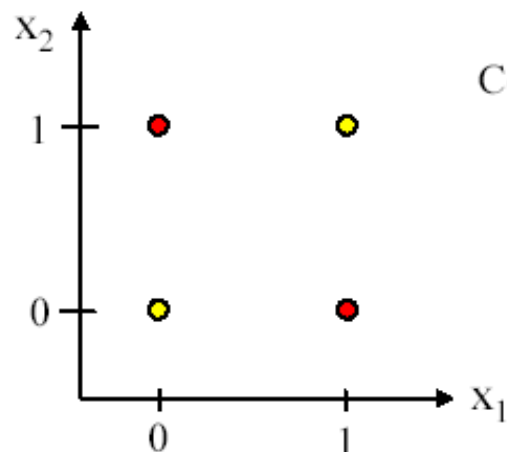
Teoria que Fundamenta as Redes RBF

- Teorema de Cover (1965) sobre a separabilidade de padrões: Um problema complexo de classificação de padrões tratado não-linearmente em um espaço de dimensão alta é mais provável de ser linearmente separável que em um espaço de dimensão baixa:
 - Problemas dicotômicos, com apenas duas classes (C_1 e C_2).
 - A dicotomia é separável se existir um vetor m -dimensional tal que: $\mathbf{W}_t \Phi(\mathbf{X}) > 0, \mathbf{X} \in C_1$ ou $\mathbf{W}_t \Phi(\mathbf{X}) < 0, \mathbf{X} \in C_2$
- Superfícies de separação: Hiperplanos, hiperesferas, ...

Teoria que Fundamenta as Redes RBF

- Problema de Interpolação com Rede RBF:
 - Considere uma rede com alimentação direta de 3 camadas (entrada, escondida, saída);
 - A rede realiza mapeamento não-linear do espaço de entrada para o espaço escondido seguido por mapeamento linear do espaço escondido para o de saída;
 - A fase de treinamento consiste na otimização de um procedimento de ajuste (*fitting*) para uma superfície contruída com base nas amostras de treinamento;
 - A fase de generalização corresponde à interpolação de novos pontos não empregados durante o treinamento.

Teoria que Fundamenta as Redes RBF

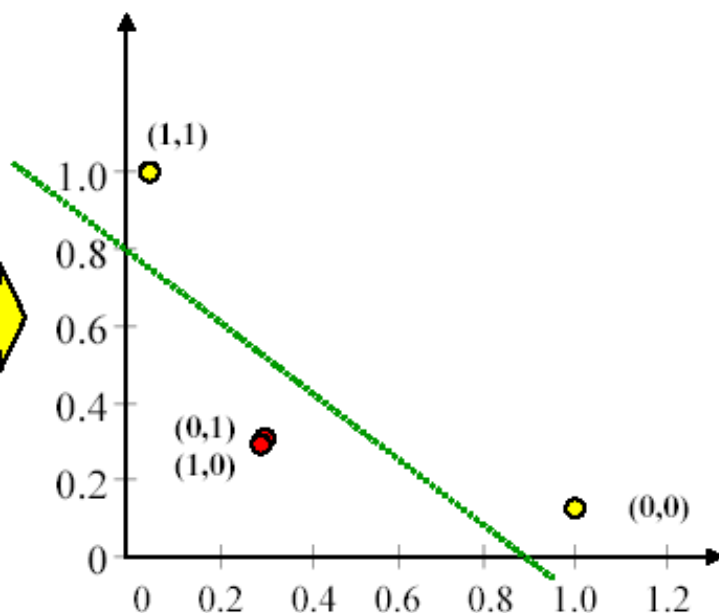


Consider the nonlinear functions to map the input vector \mathbf{x} to the φ_1 - φ_2 space

$$\mathbf{x} = [x_1 \ x_2] \quad \varphi_1(\mathbf{x}) = e^{-\|\mathbf{x} - \mathbf{t}_1\|^2} \quad \mathbf{t}_1 = [1 \ 1]^T$$

$$\varphi_2(\mathbf{x}) = e^{-\|\mathbf{x} - \mathbf{t}_2\|^2} \quad \mathbf{t}_2 = [0 \ 0]^T$$

Input \mathbf{x}	$\varphi_1(\mathbf{x})$	$\varphi_2(\mathbf{x})$
(1,1)	1	0.1353
(0,1)	0.3678	0.3678
(1,0)	0.3678	0.3678
(0,0)	0.1353	1

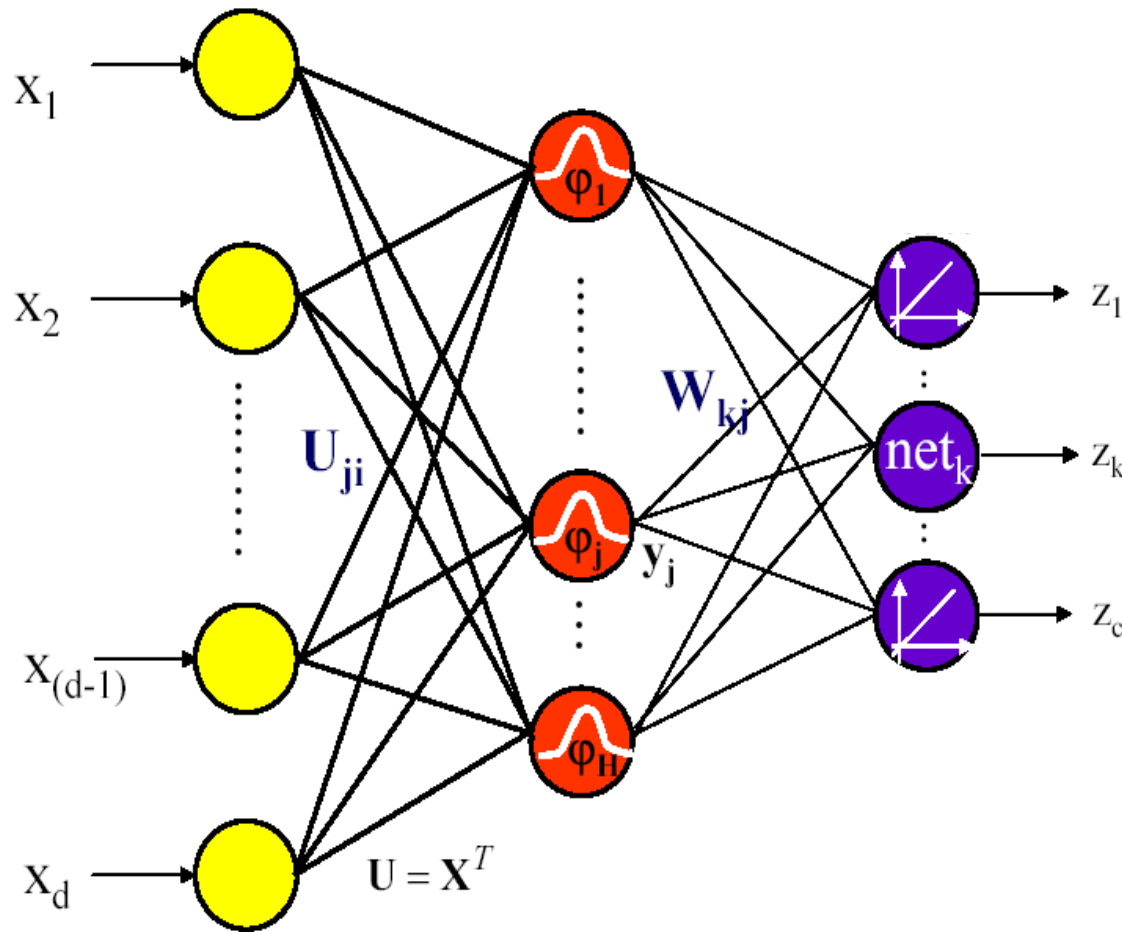


The nonlinear φ function transformed a nonlinearly separable problem into a linearly separable one !!!

Rede de Função de Base Radial - RBF

- As redes RBF são redes de alimentação direta (*feedforward*) consistindo tipicamente de três camadas: entrada, escondida e saída;
- Camada de entrada: propaga os estímulos;
- Camada escondida: unidades de processamento localmente sintonizáveis;
- Camada de saída: unidades de processamento lineares.

Redes RBF: Arquitetura e Funções



Redes RBF: Processamento

- Unidades escondidas recebem o vetor de entrada;
- Camada de entrada: propaga os estímulos;
- Camada escondida: unidades de processamento localmente sintonizáveis:

$$y_j(\mathbf{x}_p) = F_j(\mathbf{x}_p, \boldsymbol{\mu}_j, \sigma_j)$$

onde $F_j(\cdot)$ é uma função de base radial,

$\boldsymbol{\mu}_j$ é o j -ésimo centro e

σ_j é a largura do campo receptivo para o centro.

Redes RBF: Processamento

- Camada de saída: unidades de processamento lineares:

$$z_l(\mathbf{x}_p) = \sum_{j=1}^H w_{lj} y_j(\mathbf{x}_p)$$

onde w_{lj} é o peso entre a unidade escondida j e a de saída l .

- Redes RBF realizam aproximação de uma função por superposição de funções de base radial não-ortogonais que têm forma de sino,
 - O grau de precisão pode ser controlado por três parâmetros: o número de funções de base usadas, sua localização e sua largura do campo receptivo.

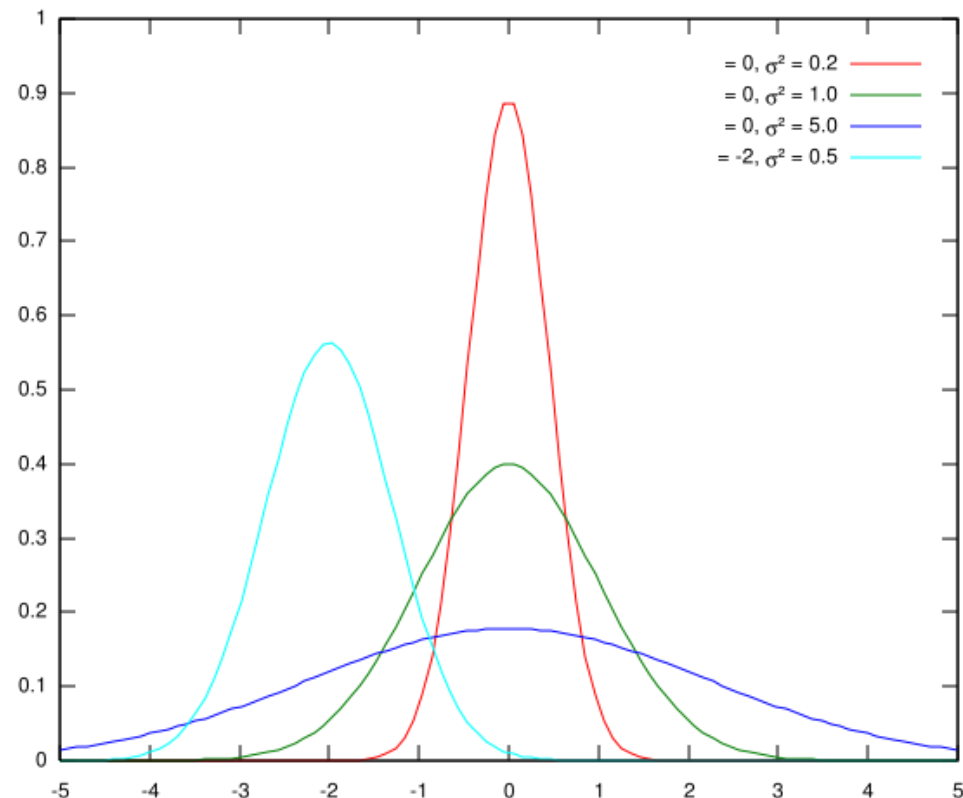
Redes RBF: Funções de Base

- Função Gaussiana

$$y_j(\mathbf{x}_p) = \exp\left(-\frac{(\mathbf{x}_p - \boldsymbol{\mu}_j)^2}{2\sigma_j^2}\right), \text{ onde}$$

$\boldsymbol{\mu}_j$ é o j -ésimo centro e

σ_j é a largura do campo receptivo.

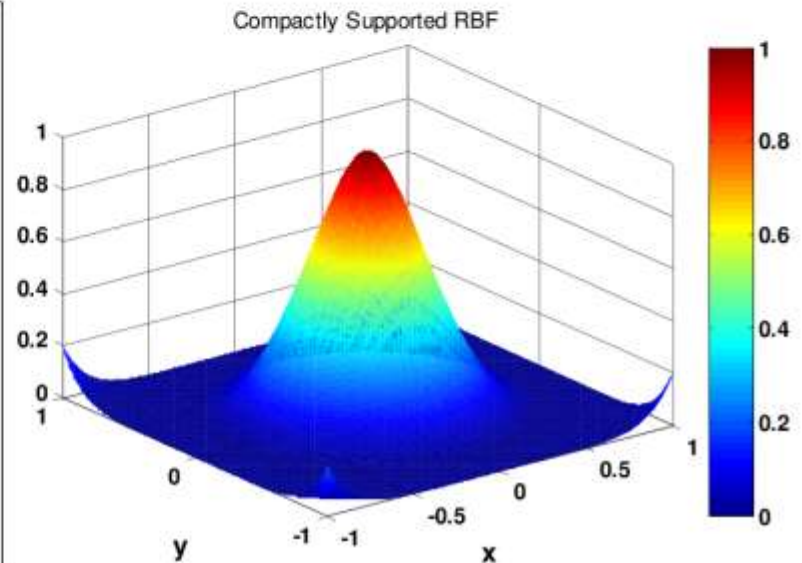
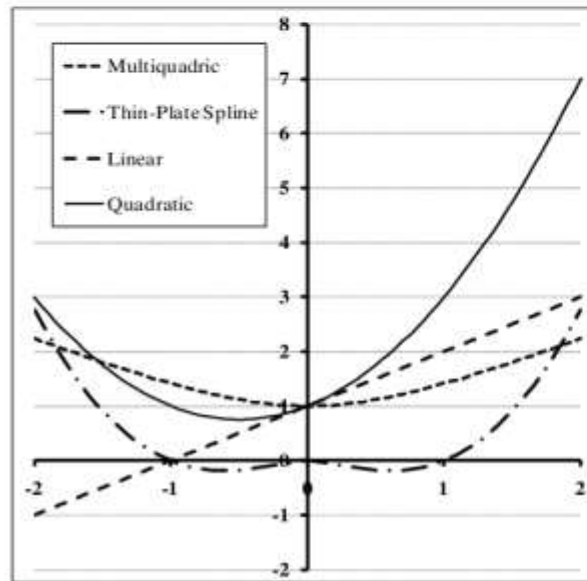


Redes RBF: Funções de Base

Table 1: Radial Basis Functions (RBF) chosen and their corresponding particular solution (Chen *et al.*, 1999, Li *et al.*, 2002, and Gumgum, 2010).

Abbreviation	Radial Basis Function Type	Corresponding Particular Solution
RBF 1	Linear (LR): $1 + r$	$\frac{r^3}{9} + \frac{r^2}{4}$
RBF 2	Thin-Plate Spline (TPS): $r^2 \ln r$	$\frac{r^4 \ln r}{16} - \frac{r^4}{32}$
RBF 3	Multiquadric (MQ): $\sqrt{r^2 + c^2}$	$-\frac{c^3}{3} \ln(c\sqrt{r^2 + c^2} + c^2) + \frac{1}{9}(r^2 + 4c^2\sqrt{r^2 + c^2})$
RBF 4	Compactly Supported (CS) $\begin{cases} \left(1 - \frac{r}{c}\right)^4 \left(4\left(\frac{r}{c}\right) + 1\right) & , 0 \leq r \leq c \\ 0 & , r > c \end{cases}$	$\begin{cases} \frac{4r^7}{49c^5} - \frac{5r^6}{12c^4} + \frac{4r^5}{5c^3} - \frac{5r^4}{8c^2} + \frac{r^2}{4} & , r \leq c \\ \frac{529c^2}{5880} + \frac{c^2}{14} \ln\left(\frac{r}{c}\right) & , r > c \end{cases}$
RBF 5	Quadratic (QC): $1 + r + r^2$	$\frac{r^2}{4} + \frac{r^3}{9} + \frac{r^4}{16}$

Tipos de função de base radial:



Redes RBF: Funções de Saída

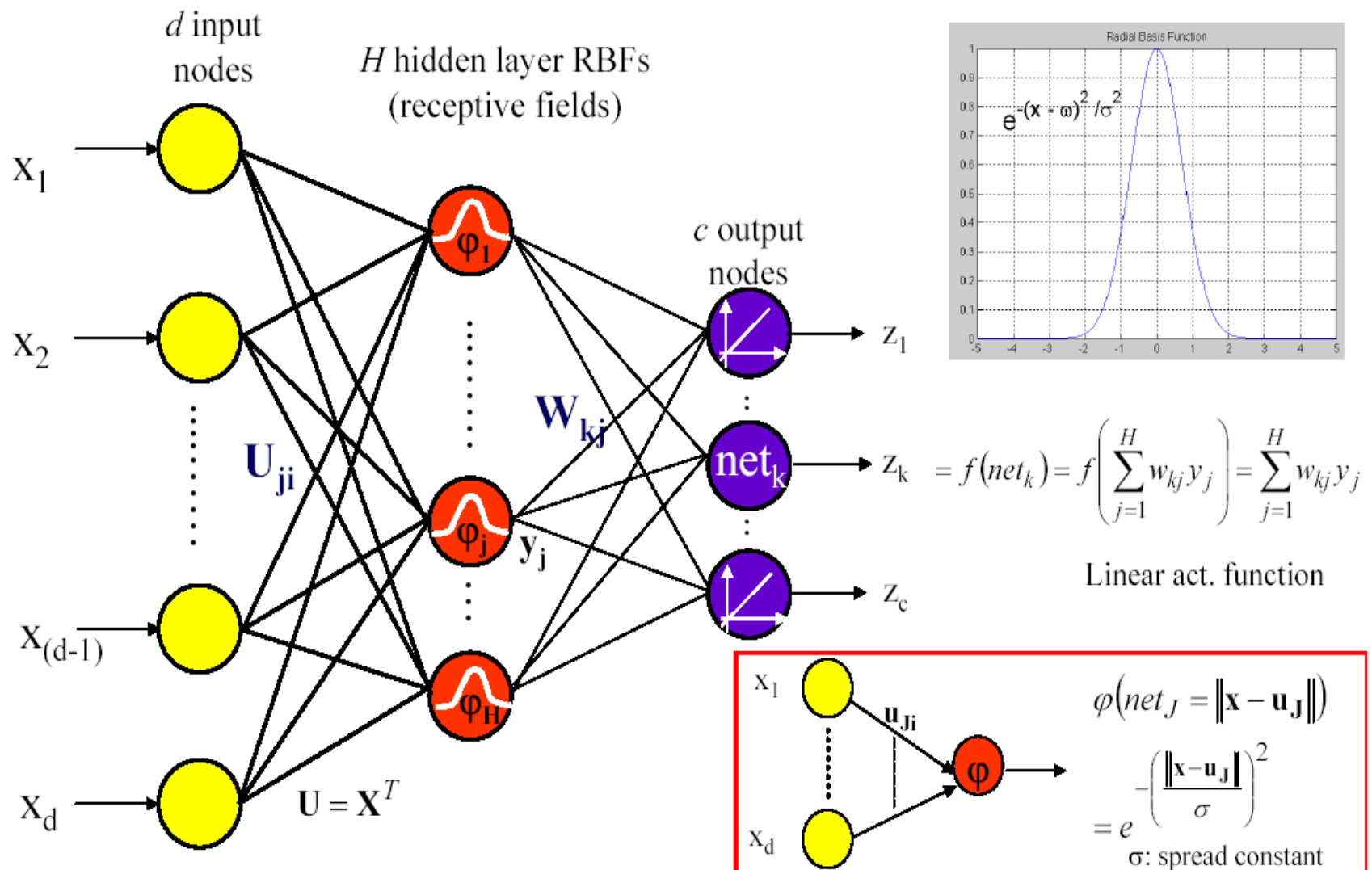
- Função Linear nas Unidades de Saída

$$z_k(\mathbf{x}_p) = f(net_k) = f\left(\sum_{j=1}^H w_{kj} y_j\right) = \sum_{j=1}^H w_{kj} y_j, \text{ onde}$$

H é o número de unidades escondidas

w_{kj} é o peso entre as unidades j e k .

Redes RBF: Arquitetura e Funções



Aprendizagem em Redes RBF

- Aprendizagens nas redes RBF (com N padrões de treinamento):
 - Aprendizagem exata: Tem N nodos na camada RBF, não usa processo iterativo, centros RBF são vetores de treinamento, dispersão é variância dos dados e pesos são solução de um conjunto de equações lineares;
 - Centros fixos escolhidos aleatoriamente: Número de nodos na camada RBF $H < N$, não usa processo iterativo, dispersão usa métricas euclidianas e pesos são solução de um conjunto de equações lineares;
 - Centros determinados por aprendizagem não-supervisionada (*clustering*) para a qual a dispersão é a variância dos grupos e pesos são obtidos através do algoritmo LMS;
 - Todas os parâmetros são determinados por aprendizado supervisionado.

Aprendizagem em Redes RBF

- Aprendizagem exata para RBF:
 - Garante a classificação correta de todas as instâncias de treinamento;
 - O cálculo dos pesos \mathbf{W} consiste na resolução de um conjunto de equações lineares, não executando treinamento iterativo;
 - Produz capacidade de generalização pobre e não suave;
- Definições na aprendizagem exata para RBF:
 - Os centros das RBFs são todas as instâncias de treinamento;
 - A dispersão é $\sigma = d_{max}/(2N)^{1/2}$ onde d_{max} é a distância euclidiana máxima entre quaisquer dois centros e N é o número de pontos de treinamento, onde $H=N$;
 - No treinamento deseja-se que a saída sejam os alvos desejados.

Aprendizagem em Redes RBF

- RBF com aprendizagem exata: Considere uma rede RBF com N unidades escondidas, na qual saída desejada $\mathbf{d} = [d_1 \ d_2 \ \dots \ d_N]^T$, pesos $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_N]^T$, e funções de base $\Phi = \{\varphi_{ij}, i, j = 1, \dots, N\}$, logo

$$z_k = \sum_{j=1}^N w_{kj} \cdot \underbrace{\varphi(\|\mathbf{x} - \mathbf{u}_j\|)}_y \quad \Rightarrow \quad \begin{bmatrix} \varphi_{11} & \varphi_{12} & \cdots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \varphi_{N1} & \varphi_{N2} & \cdots & \varphi_{NN} \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix}$$

- A interpolação exata (ou restrita) é definida como $\Phi \mathbf{w} = \mathbf{d} \Leftrightarrow \mathbf{w} = \Phi^{-1} \mathbf{d}$,
 - Ela ocorre se a matriz seja inversível;

Aprendizagem em Redes RBF

- Teorema de Michelli (1986): Seja $\{\mathbf{x}_i\}$, $i=1, \dots, N$ um conjunto de pontos distintos um espaço d -dimensional. Então a matriz de interpolação Φ , com dimensão $N \times N$, cujos elementos são obtidos de uma função de base radial, $\varphi_{ij} = \varphi(\|\mathbf{x}_j - \mathbf{x}_i\|)$ é não-singular, logo é inversível,
 - O teorema é válido para qualquer valor de N , tipo de função de base radial empregada e da natureza dos pontos;
- Devido ao ruído frequentemente presente nos dados, a interpolação exata dificilmente é útil;
- O procedimento de interpolação estrita não é uma boa estratégia de treinamento pois resulta em baixa capacidade de generalização;
- Aprendizagem é vista como um problema de reconstrução de hiperespaço, dado um conjunto de pontos que pode ser esparso.

Aprendizagem em Redes RBF

- O treinamento que combina aprendizagem não-supervisionada (ANS) com aprendizagem supervisionada (AS) é o mais comum pois não se sabe as saídas desejadas para a camada escondida:
 - ANS determina centros e campos receptivos da camada escondida;
 - AS determina os valores dos pesos entre as camadas escondida e de saída, considerando constantes os parâmetros já definidos.

Aprendizagem em Redes RBF

- Seleção não-supervisionada de centros:
 - Reduz número de centros comparada a escolhas por aprendizagem exata;
 - Três tipos comuns: Seleção aleatória de centros, seleção de centros por agrupamento e seleção de centros por estimativa de densidade;
- Seleção aleatória de centros:
 - Escolha aleatória de um conjunto de instâncias como centros;
 - Método muito simples e rápido;
 - Frequentemente vai gerar número alto de centros;

Aprendizagem em Redes RBF

- Seleção de centros por agrupamento:
 - Abordagem popular na qual os centros não coincidem com instâncias;
 - Emprega-se métodos de agrupamento para encontrar os centros;
 - Emprega-se métodos de agrupamento, por exemplo, o *K-means* ou mapas auto-organizáveis;
- Seleção de centros por estimativa de diversidade:
 - Modelagem da densidade do espaço de características;
 - Aplica-se modelo de misturas gaussianas empregando algoritmo de maximização de expectativa;

Aprendizagem em Redes RBF

- *K-means*:
 - Seja um conjunto de observações $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$, $\mathbf{x} \in \mathbb{R}^d$, o *k-means* visa particionar as n observações em k conjuntos ($k \leq n$) conjuntos $S = \{S_1, S_2, \dots, S_k\}$ com base na minimização da soma dos quadrados das distâncias dos elementos do grupo para seu centroide (variância):

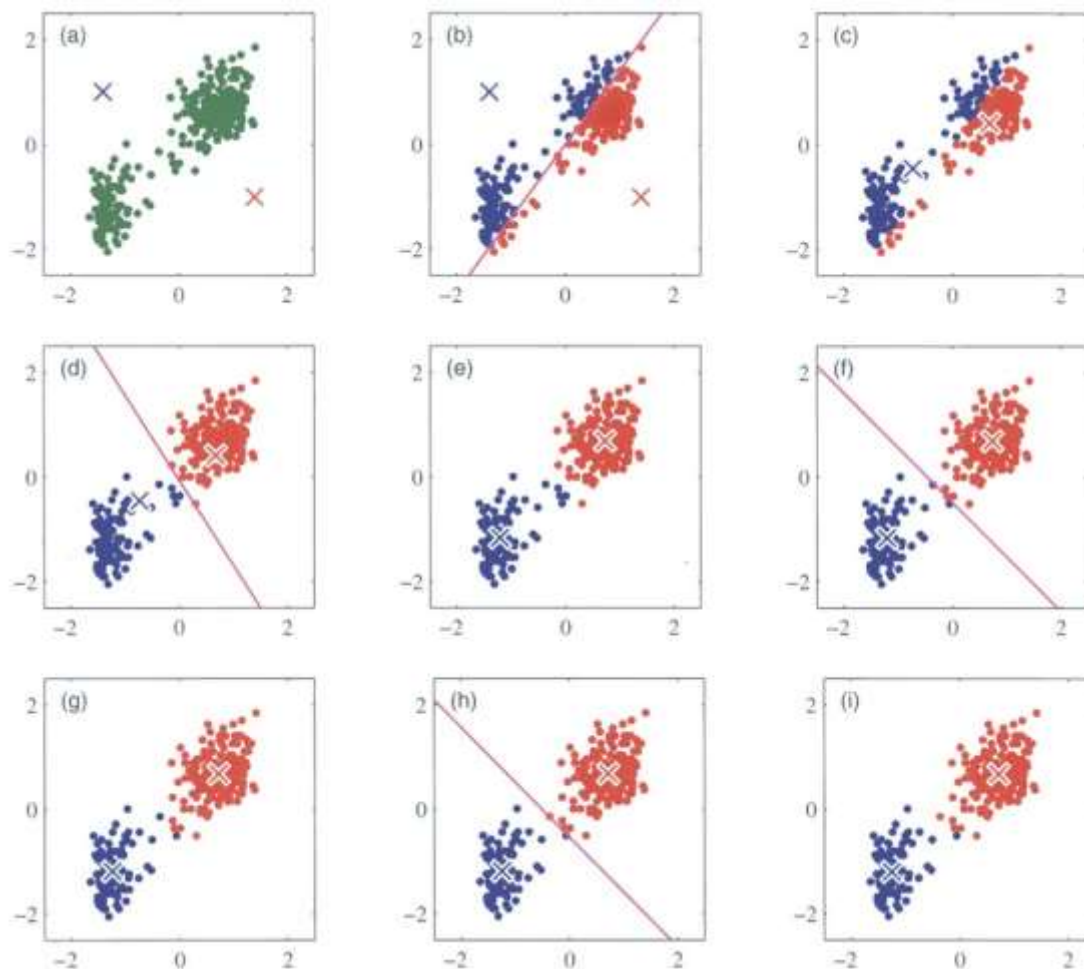
$$\arg \min_{\mathbf{S}} \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \arg \min_{\mathbf{S}} \sum_{i=1}^k |S_i| \text{Var } S_i$$

onde $\boldsymbol{\mu}_i$ é a média (centroide) de pontos em S_i :

$$\boldsymbol{\mu}_i = \frac{1}{|S_i|} \sum_{\mathbf{x} \in S_i} \mathbf{x},$$

Aprendizagem em Redes RBF

- Algoritmo *K-means*:
 - Sorteie k centros entre os pontos conhecidos;
 - Repita até que centros não mudem de uma iteração para outra;
 - Agrupe cada padrão para o centro mais próximo de si;
 - Recalcule o centro de cada agrupamento depois de reagrupar todos padrões;



Aprendizagem em Redes RBF

- Cálculo dos raios: Operação vem após a determinação dos centros:
 - Variância das instâncias de cada grupo (centros por agrupamento);
 - Matrizes de covariância dos componente gaussianos (centros por densidade);
 - Distância média entre centros (todos os casos);
- Determinação da largura do campo receptivo por distância euclidiana média entre centros:

$$\sigma = \frac{1}{ng} \sum_{j=1}^{ng} \|\mu_j - \mu_{j(mprox)}\|$$

onde $\mu_{j(mprox)}$ é o centro com menor distância euclideana para μ_j .

ng é o número de grupos que serão formados.

Aprendizagem em Redes RBF

- Determinação da largura do campo receptivo por distância euclidiana entre centro e vetor de entrada:

$$\sigma_j^2 = \frac{1}{nv} \sum_{\mathbf{x}_p \in \Psi_j} \|\boldsymbol{\mu}_j - \mathbf{x}_p\|^2$$

onde Ψ_j é o conjunto dos nv vetores mais próximos a $\boldsymbol{\mu}_j$.

- Determinação da largura do campo receptivo por distância euclidiana entre centros:

$$\sigma = \alpha \|\boldsymbol{\mu}_j - \boldsymbol{\mu}_{j(mprox)}\|$$

onde $1 \leq \alpha \leq 1,5$.

Aprendizagem em Redes RBF

- Desvantagens da seleção de centros não supervisionados:
 - A seleção não é guiada por uma medida de erro na saída;
 - Os centros obtidos pela densidade das características podem não capturar a estrutura que carrega a informação discriminante;
- Busca-se reduzir as desvantagens empregando-se algoritmos de aprendizagem supervisionada.

Aprendizagem em Redes RBF

Treinamento totalmente supervisionado

1. Linear weights (output layer)

$$\frac{\partial \mathcal{E}(n)}{\partial w_i(n)} = \sum_{j=1}^N e_j(n) G(\|\mathbf{x}_j - \mathbf{t}_i(n)\|_{C_i})$$

$$w_i(n+1) = w_i(n) - \eta_1 \frac{\partial \mathcal{E}(n)}{\partial w_i(n)}, \quad i = 1, 2, \dots, m_1$$

2. Positions of centers (hidden layer)

$$\frac{\partial \mathcal{E}(n)}{\partial \mathbf{t}_i(n)} = 2w_i(n) \sum_{j=1}^N e_j(n) G'(\|\mathbf{x}_j - \mathbf{t}_i(n)\|_{C_i}) \Sigma_i^{-1} [\mathbf{x}_j - \mathbf{t}_i(n)]$$

$$\mathbf{t}_i(n+1) = \mathbf{t}_i(n) - \eta_2 \frac{\partial \mathcal{E}(n)}{\partial \mathbf{t}_i(n)}, \quad i = 1, 2, \dots, m_1$$

3. Spreads of centers (hidden layer)

$$\frac{\partial \mathcal{E}(n)}{\partial \Sigma_i^{-1}(n)} = -w_i(n) \sum_{j=1}^N e_j(n) G'(\|\mathbf{x}_j - \mathbf{t}_i(n)\|_{C_i}) \mathbf{Q}_{ji}(n)$$

$$\mathbf{Q}_{ji}(n) = [\mathbf{x}_j - \mathbf{t}_i(n)][\mathbf{x}_j - \mathbf{t}_i(n)]^T$$

$$\Sigma_i^{-1}(n+1) = \Sigma_i^{-1}(n) - \eta_3 \frac{\partial \mathcal{E}(n)}{\partial \Sigma_i^{-1}(n)}$$

$$\mathcal{E} = \frac{1}{2} \sum_{j=1}^N e_j^2$$

$$e_j = d_j - \sum_{i=1}^M w_k \varphi(\|\mathbf{x}_j - \mathbf{t}_i\|)$$

$$G(\|\mathbf{x}_j - \mathbf{t}_i\|_C) = \varphi(\|\mathbf{x}_j - \mathbf{t}_i\|)$$

G' represents the first derivative of the function wrt its argument

Estratégias de Treinamento

- Busca-se o compromisso entre precisão e generalização. Para tal, duas estratégias de treinamento podem ser empregadas: *hold-out* e validação cruzada (*crossvalidation*).
 - *Hold-out*: O conjunto de padrões é dividido em três grupos: treinamento, validação e teste. Cada topologia com seus centros tem seu desempenho testado com respeito aos três conjuntos;
 - *Crossvalidation*: Todos os padrões (em geral poucos) são considerados para treinamento. Assim, divide-se os padrões em ng grupos, seleciona-se aleatoriamente $ng-1$ conjuntos para treinamento e testa-se a rede com o conjunto não selecionado. Repete-se o processo até todos os conjuntos serem usados para testes. A partir daí, calcula-se o erro (E).

$$E = \frac{1}{ng} \sum_{j=1}^{ng} E_i^2$$

Desempenho das Redes RBF

- Inicialmente, as redes RBF foram aplicadas com sucesso na aproximação de funções (Broomhead e Lowe, 1988; Lee e Kil, 1988; Casdagli, 1989; Moody e Darken, 1989a, 1989b) e em problemas de classificação (Nirajan e Fallside, 1988; Nowlan, 1990; Lee, 1991; Wettschereck e Dieterich, 1992; Vogt, 1993).
- Em tarefas difíceis de aproximação/interpolação (por exemplo, predição da série caótica de Mackey-Glass T instantes de tempo no futuro, $T > 50$), redes RBF que empregam a técnica de agrupamento no posicionamento dos campos receptivos podem alcançar desempenho comparável ao das redes MLP-BP, enquanto requerem tempo de treinamento algumas ordens de grandeza menor.

Comparando RBF e MLP-BP

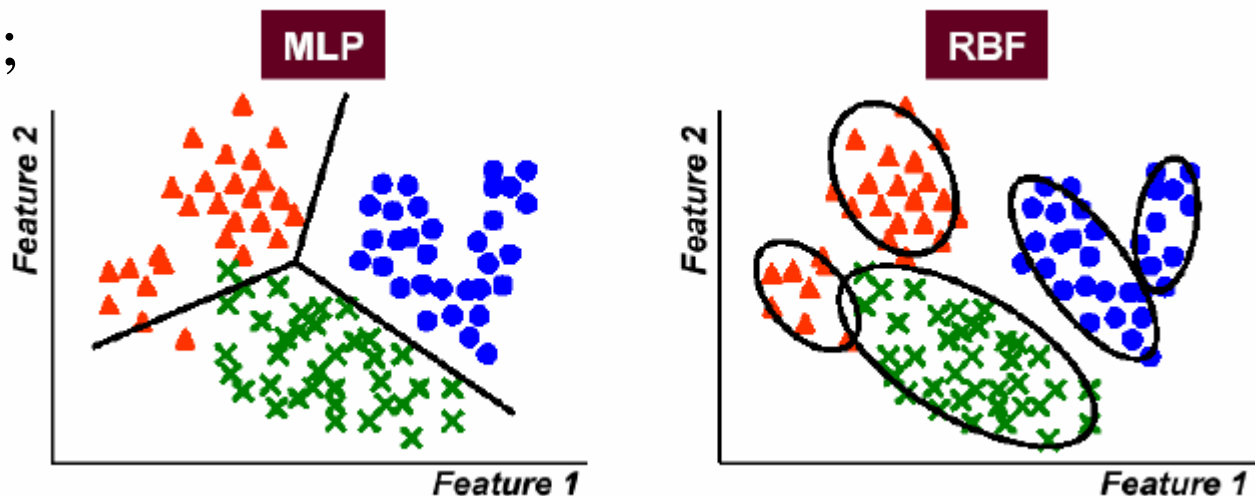
- RBF só tem uma camada escondida e a MLP-BP pode ter mais.
- RBF usualmente tem mais unidades escondidas que a MLP-BP.
- RBF tem ativação na camada escondida e de saída com função de base e linear e a MLP-BP tem funções sigmoidais de ativação.
- RBF é usualmente menos sensível a desaprendizagem, devido a dados novos, que a MLP-BP.
- RBF pode necessitar de maior número de parâmetros ajustáveis que a MLP-BP para realizar mapeamento entrada-saída não-linear.
- RBF pode ser afetada pela maldição da dimensionalidade.

Comparando RBF e MLP-BP

- Redes RBF geralmente precisam de ao menos dez vezes mais dados de treinamento para atingir a acuracidade das redes MLP-BP;
- Redes RBF com número adequado de padrões de treinamento e de unidades escondidas aplicadas a problemas difíceis de classificação podem atingir desempenho superior a redes MLP-BP;
- As redes RBF têm tempo de treinamento menor porque:
 - Apenas uma pequena fração de unidades escondidas responde a um dado padrão de entrada, pois são unidades localmente sintonizáveis, sensíveis apenas a padrões próximos de seus campos receptivos;
 - Todas as unidades de uma rede MLP-BP são avaliadas e têm seus pesos ajustados para cada vetor de entrada;
 - Empregam treinamento separado na camada escondida e na de saída.

Comparando MLP-BP e RBF

- RBF separa classes por hiperelipsoides e a MLP-BP por hiperplanos;



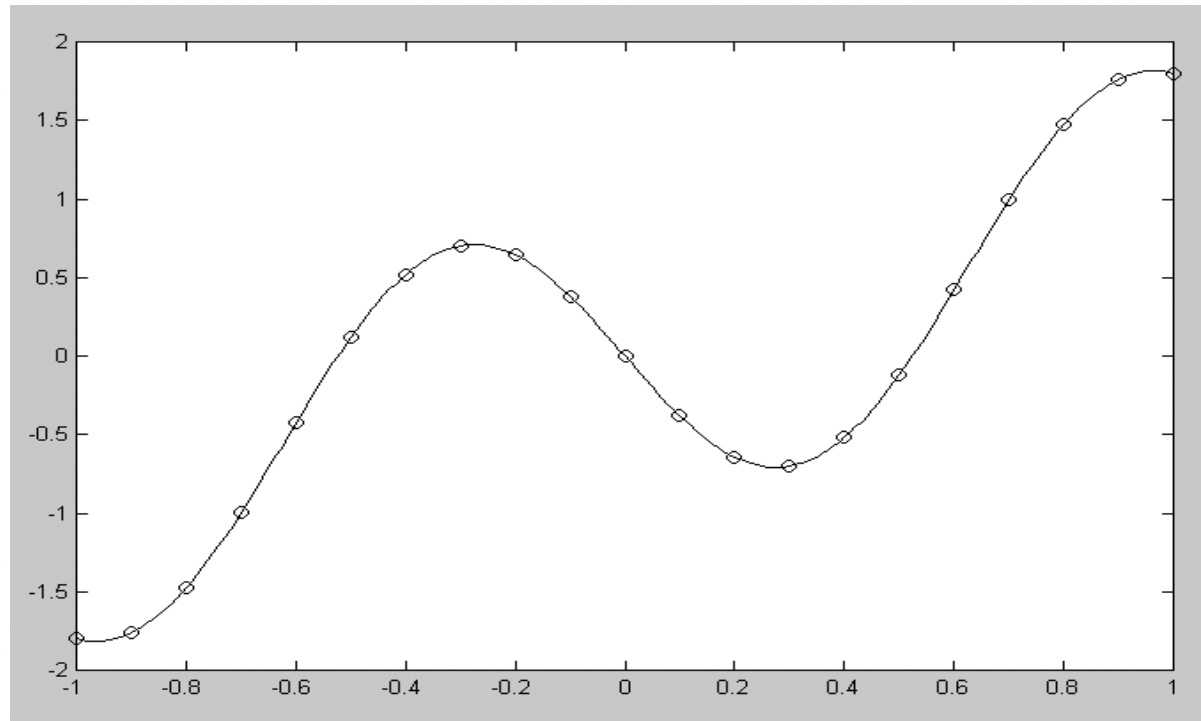
- RBF e MLP-BP são aproximadores universais local e global;
- MLP-BP gera regiões globais de decisão, logo ela tende a extrapolar melhor que a RBF. Cabe salientar que extrapolação pode significar lidar com *outliers*.

Comparando RBF e MLP-BP

- Quando utilizadas em aproximação de funções as redes MLP-BP tendem a obter maior capacidade de generalização (ou extrapolação) que as RBF pois ajustam globalmente os padrões de entrada enquanto estas últimas fazem um ajuste local;
- Pelo mesmo motivo, em problemas de classificação, redes RBF tendem a cometer menos erros de falsa classificação positiva do que as redes MLP-BP;
- Pode-se dizer que, em geral, é melhor o uso de redes MLP-BP quando os padrões de entrada são custosos (ou difíceis de se gerar) e/ou quando a velocidade de recuperação é crítica. No entanto, se os dados são baratos e abundantes, e se é necessário treinamento *on-line*, então as redes RBF são superiores.

Exemplo de Emprego de Redes RBF

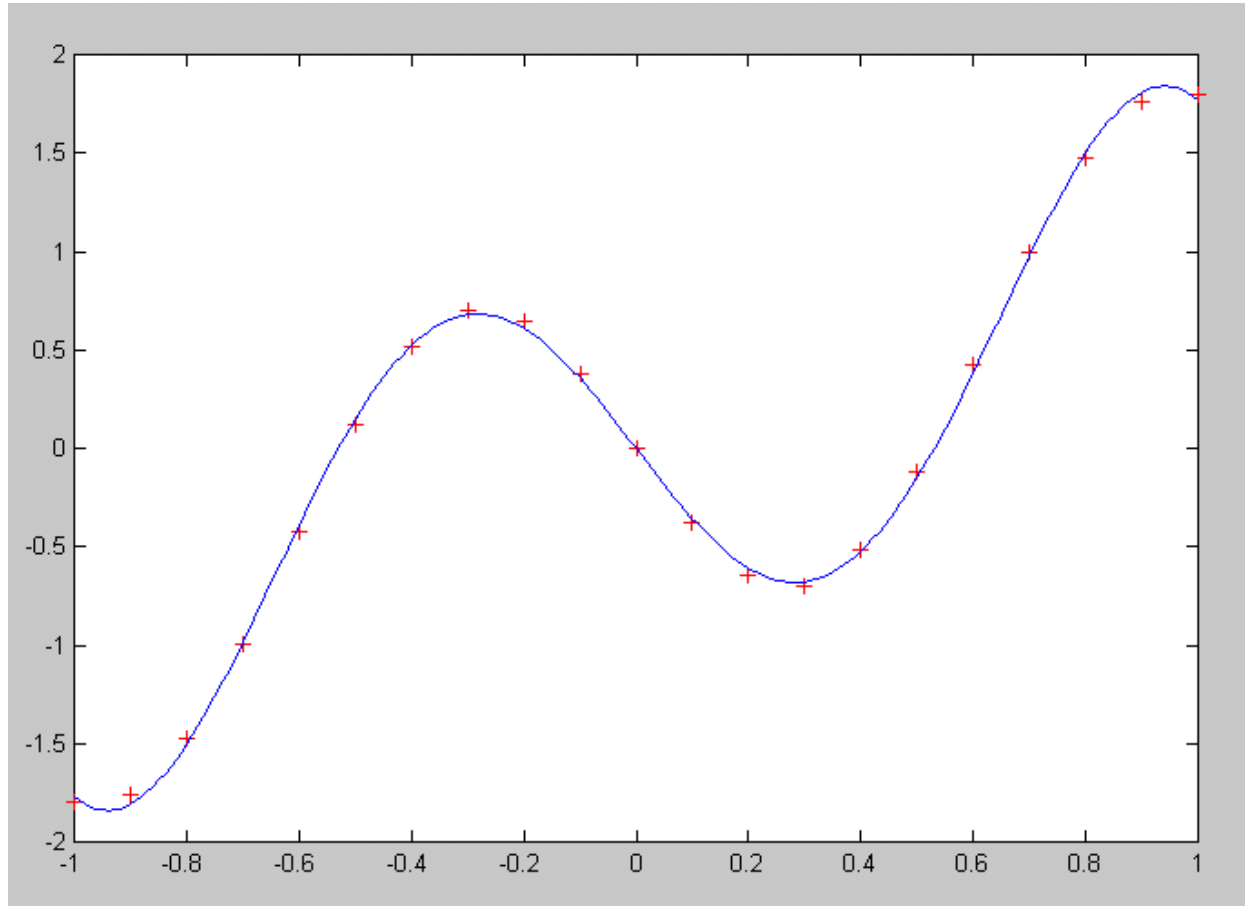
- Rede RBF para aproximar uma função e verificar papel dos parâmetros livres (os campos receptivos) afetam seu desempenho.
- A função: $y = f(x) = \sin(x) - \sin(5x)$ (Figura 1). Apresentam-se à rede 21 pares X-Y (círculos) onde X é a entrada e Y é sua saída desejada.



Exemplo de Emprego de Redes RBF

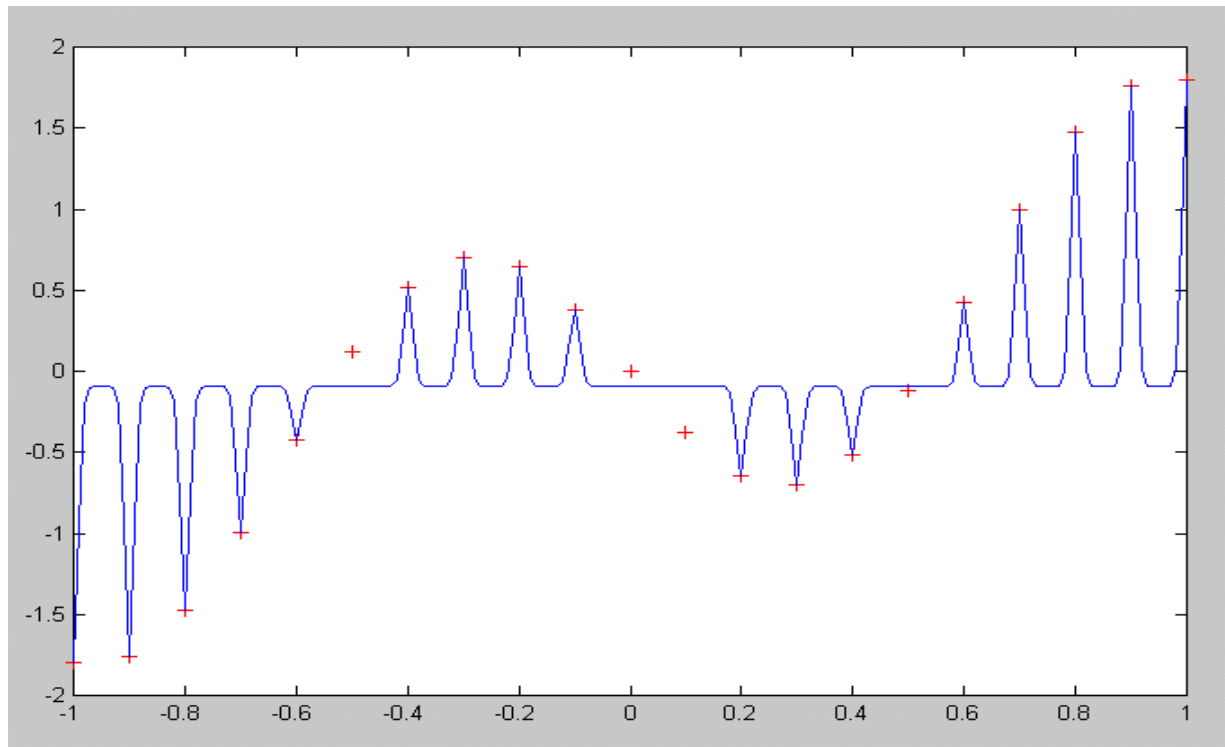
- A rede utilizada é composta por uma unidade de entrada, que recebe o valor de X , 21 unidades escondidas fixas (com centros localizados sobre os valores de entrada, que estão igualmente espaçados, ou seja, os centros estão fixados em uma “grade” regular) e uma unidade linear de saída, que produz o valor de Y . Os pesos da primeira camada são unitários e fixos e os pesos da segunda camada serão ajustados no treinamento. O erro será calculado pela função SSE.
- Depois de treinada, a rede aproxima a função dada com boa precisão e poderia ser utilizada para interpolação. O resultado está no gráfico da Figura 2, onde a linha cheia representa a saída da rede para 201 valores de entrada e as cruzes representam os 21 pontos apresentados à rede durante o treinamento.

Exemplo de Emprego de Redes RBF



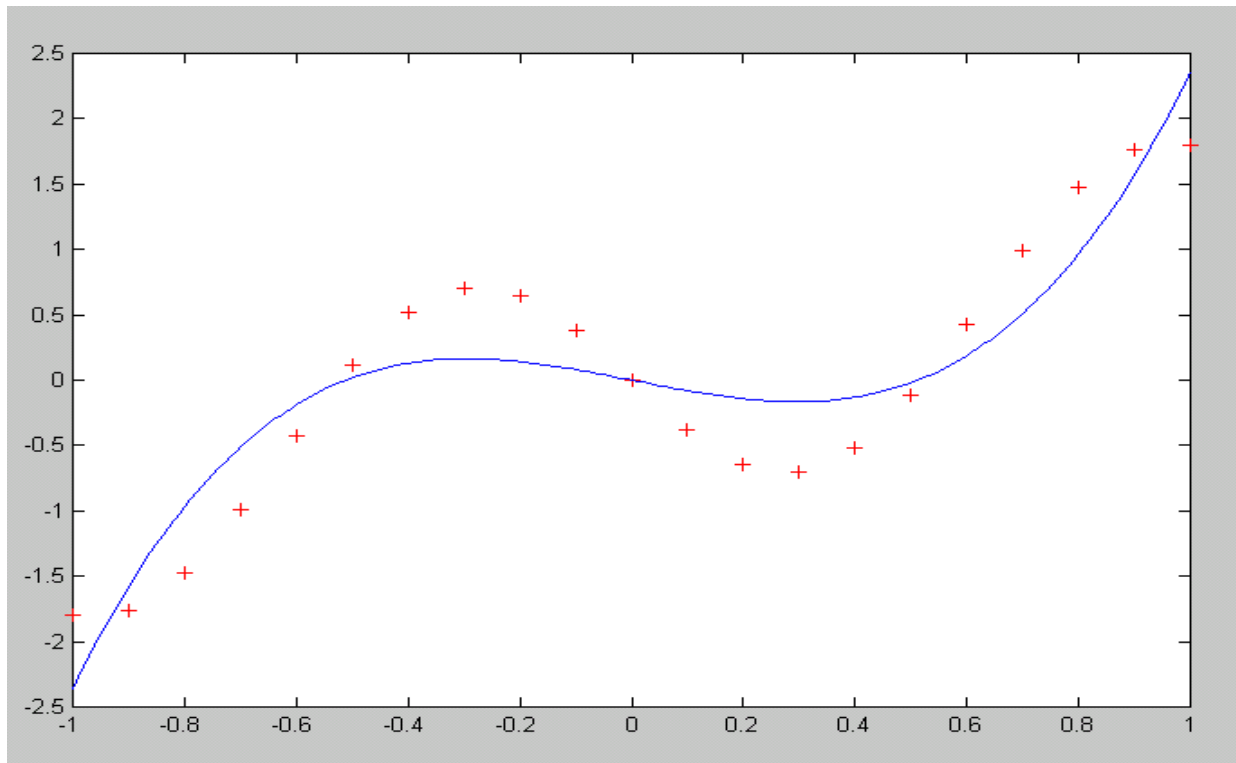
Exemplo de Emprego de Redes RBF

- Considere mesma rede anterior mas os campos receptivos são muito menores. Na Figura abaixo a linha cheia representa a saída da rede para 201 pontos de entrada e as cruzes representam os 21 pontos utilizados durante o treinamento da rede.



Exemplo de Emprego de Redes RBF

- Considere a mesma rede com campos receptivos muito grandes. O resultado visto na Figura abaixo onde linha cheia representa a saída da rede para 201 pontos de entrada e as cruzes representam os 21 pontos utilizados de treinamento.



Referências

- Du, K.-L. & Swamy M. N. S. (2019). *Neural Networks and Statistical Learning*. Springer, 2nd edition.
- Haykin, S. (2009). *Neural Networks and Learning Machines*. Third Edition. Pearson.
- Hassoun, M. H. (1995). *Fundamentals of Artificial Neural Networks*. Cambridge: The MIT Press.