Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais, Vol. 1, No. 1, pp. 29-37, 2003

REDES NEURAIS LOCAIS-GLOBAIS – UMA APLICAÇÃO AO PROBLEMA DE DADOS FALTANTES

Carlos E. Pedreira[1], Mayte Fariñas[1] e Luiz Carlos Pedroza[2]

[1] DEE PUC-RIO CP. 38063 Rio de Janeiro, CEP 22452-970
[2] CEFET-RJ Av. Maracanã, 229 Rio de Janeiro, CEP 20271-110
E-mails: pedreira@ele.puc-rio.br, mayte@ele.puc-rio.br, pedroza@cefet-rj.br

Abstract: In this paper a new connectionist architecture is proposed. The architecture is trained by a scheme based on partition of the function domain, approximating the generator function by a set of very simple supporting functions. This method has an interesting ability concerning interpolation. A synthetic experiment and a real data missing data application are presented.

Palavras chaves: Neural network, Interpolation, Missing values, Function approximation

1. INTRODUÇÃO

Muitos problemas de interesse prático estão relacionados a interpolação de dados. Entre estes, de particular relevância está o problema de preenchimento de dados faltantes. Neste caso, o que se procura é uma forma de emular uma função em um intervalo do domínio onde apenas uma parte dos pontos é conhecida. A preocupação aqui não é a capacidade preditiva da rede, isto é, estimar pontos fora do domínio, mas sim de reproduzir a função da melhor forma possível dentro do domínio estabelecido. Neste artigo, propõe-se um algoritmo capaz de reconstruir a função a partir de estimativas locais ao longo do domínio de interesse.

A arquitetura conexionista proposta é não usual e utiliza uma metodologia de treinamento baseada no particionamento do domínio da função a ser emulada. Esta função é aproximada por um conjunto de funções de apoio muito simples, muitas vezes lineares. A idéia central é expressar o mapeamento entrada-saída através de uma função composta por partes. A estrutura básica é constituída pela combinação de vários pares de funções de aproximação e funções de nível de ativação. As funções de nível de ativação definem em cada trecho do domínio a participação da função de aproximação a esta associada. É possível a ocorrência de superposições parciais das funções de nível de ativação proporcionando uma maior riqueza do mapeamento pretendido. Desse modo o problema de aproximação de funções é enfocado especializando-se grupos de neurônios, formados pelos pares anteriormente descritos, que emulam a função geradora em cada setor do domínio. O grau de especialização em uma determinada região é dado pelo valor da função de nível de ativação. Por exemplo, em um trecho aonde apenas uma das funções de nível de ativação tem valor alto, haverá uma dominância da função de aproximação associada a esta.

2. A ARQUITETURA PROPOSTA

Como já foi comentado, a estrutura básica do RNGL é constituída pela combinação de vários pares de funções de aproximação e funções de nível de ativação. Definamos formalmente as funções de nível de ativação e as funções de aproximação

Definição 1. Funções de nível de ativação.

Seja $x \in \mathcal{R}$, a função de nível de nível de ativação define-se através da expressão:

$$B(x) = -\left[\frac{1}{1+\exp(d(x-h^{(1)}))} - \frac{1}{1+\exp(d(x-h^{(2)}))} \right],$$

onde d, $h^{(1)}$, $h^{(2)}$ são parâmetros reais. A Figura 1 ilustra o gráfico desta função para dois conjuntos diferentes de parâmetros. Nota-se que o parâmetro d está relacionado à declividade desta função entanto que os parâmetros $h^{(1)}$, $h^{(2)}$ delimitam a região do domínio em que a j-ésima função de aproximação é mais ativa (ver figura 1).

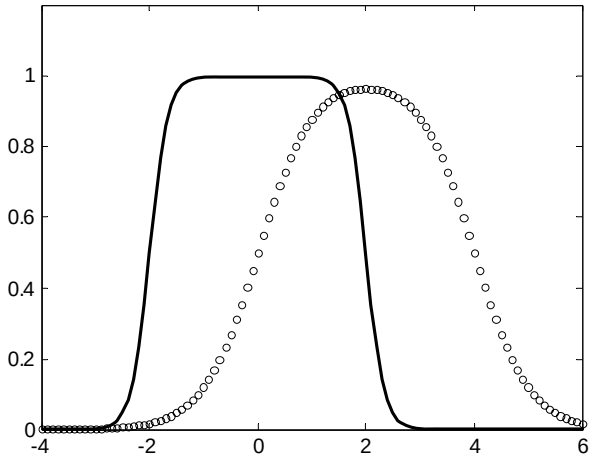

Figura 1– Exemplos de funções de ativação
— d=6; $h_j^{(1)}$ = -2; $h_j^{(2)}$ =2
o d=2; $h_j^{(1)}$ = 0; $h_j^{(2)}$ =4

Definição 2. Funções de aproximação.

As funções de aproximação são tipicamente funções lineares ou quadráticas. Embora funções mais complexas possam ser sem prejuízo da estrutura teórica proposta. Consideremos, por exemplo funções de aproximação lineares:

$\kappa_j(x) = a_j x + b_j$ j=1, . . . ,m

onde a_j e b_j são parâmetros a serem estimados.

A aproximação global de uma função pode ser abordada através de uma partição do domínio da função. Em cada região da partição, a função objetivo pode ser localmente aproximada por uma função de aproximação, de estrutura simples, como ilustra a Figura 2. O grau de especialização em uma determinada região é dado pelo valor da função de nível de ativação. Por exemplo, em um intervalo onde apenas uma das funções de nível de ativação tem valor alto, haverá uma dominância da função de aproximação associada a esta. É possível a ocorrência de superposições parciais das funções de nível de ativação proporcionando uma maior riqueza do mapeamento pretendido.

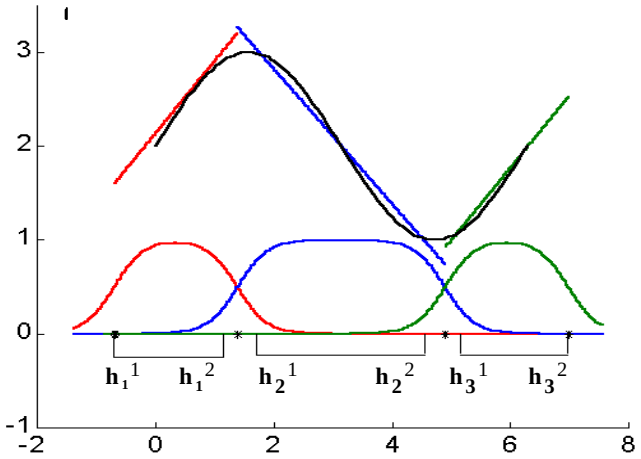

Figura 2– Idéia básica do modelo.

Assim, a função g(x) que aproxima à função objetivo pode ser expressada como:

$$g(x) = \sum_{j=1}^{m} B_j(x)\kappa_j(x) \qquad \textbf{(1)}$$

Na arquitetura RNGL, cada nó, ou neurônio da rede é constituído de um par {função de nível de ativação; função de aproximação} (ver Figura 3). As entradas são conectadas aos nós onde é efetuado o produto da função de nível de ativação $B_j(x)$ e da função de aproximação $\kappa_j(x)$. A saída da rede é um somatório da saída de cada um destes nós. Note que não há pesos ligando a saída dos nós-produto à saída da rede (ver Figura 3). Os parâmetros a estimar são os associados aos nós produto. Assim, para cada nó é necessário estimar 5 parâmetros (6 no caso de funções de aproximação quadráticas). De maneira usual, o número de nós indica a complexidade do modelo. Deste modo a saída do j-ésimo nó-produto é $B_j(x)\ \kappa_j(x)$, e a saída da rede é dada por (1)

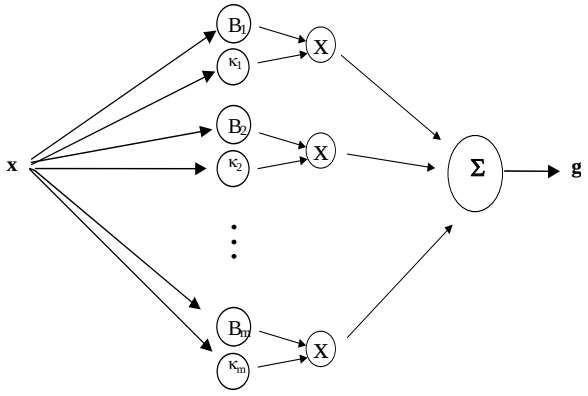

Figura 3 – A arquitetura proposta
(× representa a multiplicação dos pares de neurônios)

Definindo-se $\mathbf{B}(x)\equiv(B_1(x), B_2(x),...,B_m(x))$ e $\boldsymbol{\kappa}(x)\equiv(\kappa_1(x), \kappa_2(x),...,\kappa_m(x))$ pode-se escrever a saída da rede em forma de produto interno, i.e., $g_m(x)=<B(x), \boldsymbol{\kappa}(x)>$.

Para cada neurônio produto, $\Im^j = \left(d_j, h_j^1, h_j^2, a_j, b_j\right)$ denota o vetor de parâmetros a ele associado. Assim o conjunto de parâmetros a ser estimado resulta $\Im = (\Im^1,..., \Im^m)$. Como o objetivo é obter uma aproximação da função f(x), o vetor de parâmetros $\Im = (\Im^1,..., \Im^m)$ pode ser estimado como o argumento mínimo da função de erro considerada. Utiliza-se então, como função de erro, o erro quadrático médio definido por:

$$E = \frac{1}{n}\sum_{j=1}^{n}\left(g_m(x_j) - f(x_j)\right)^2 \qquad \textbf{(2)}$$

3. Um Resultado teórico

Um resultado interessante, apresentado no teorema T-1, dá consistência teórica à metodologia proposta. Neste teorema mostra-se que qualquer função L^2-integravél pode ser aproximada por funções da forma $g^m(x)$. A prova deste teorema faz uso de dois resultados auxiliares (RA) que são enunciados a seguir.

Resultado auxiliar RA-1: As funções simples são densas em L^2. (funções simples são funções da forma $S(x) = \sum_{j=1}^{m}\alpha_i X_{Ai}(x)$ onde α_i são constantes reais e $X_{Ai}(x)$ é a função indicadora do conjunto A_i; isto é, toma valor 1 se $x\in A_i$ e zero caso contrário).

Resultado auxiliar RA-2: As funções $\{g(x)\}$ definidas em (1) aproximam qualquer função simples. Isto é, para toda função simples S(x), existe uma seqüência de funções $g_n(x)\in\{g(x)\}$ tal que $g_n(x)\rightarrow S(x)$, esta convergência é referente à norma L^2.
O RA-1 é um conhecido resultado de teoria da medida é pode ser encontrado, por exemplo, em [3]. A seguir a prova do RA-2:

Prova de RA-2.

Seja $S(x) = X_{[0,\ 1]}(x)$. Consideremos a seqüência onde $\kappa_n(x)=C_n$, isto é: $g_n(x) = C_nB_n(x)$ com $h_1=0$ $h_2=1$.

$B_n(x) = -\left[\frac{1}{(1+e^{d_n x})} - \frac{1}{(1+e^{d_n (x-1)})}\right]$ e $C_n = \left(e^{d_n/2}+1\right)/\left(e^{d_n/2}-1\right)$ e $d_n \to \infty$ (o que garante que o máximo de $C_nB_n(x)$ em [0,1] seja igual a 1)

Devemos provar que $g_n(x)$ converge a S(x) em L^2. É fácil provar que, $g_n(x) \xrightarrow[n\to\infty]{} S(x)$ pontualmente, isto é:

$$C_n B_n(x) \to \begin{cases} 0 & x \notin [0,1] \\ 1 & x \in [0,1] \end{cases}$$

Para chegar a convergência em L^2, partindo-se da convergência pontual, pode-se utilizar o teorema da convergência dominada de Lebesgue: Se $|f_n(x)| \le g(x) \in L^2$, e $f_n(x) \to f(x)$ pontualmente então $f_n(x) \to f(x)$ em L^2 [3]. Como já se tem a convergência pontual, basta provar que $g_n(x)$ é dominada por uma função L^2-integrável. Seja g(x) definida como:

$$g(x) = \begin{cases} 1+e^{x} & x < 0 \\ 2 & x \in [0,1] \\ 1+e^{-x} & x > 1 \end{cases}$$

É fácil verificar que a função g(x) é L^2-integrável e limita a função $g_n(x)$ qualquer que seja n, como se queria provar. Para generalizar o resultado para funções do tipo $S_n(x)=X_A(x)$, $A=[\delta_1, \delta_2]$, basta considerar uma sequência de funções $g_n(x) = C_nB_n(x)$, como no caso anterior, com $h_1=\delta_1$ e $h_2=\delta_2$.

A extensão deste resultado para qualquer função simples S(x) segue do fato que as funções simples são combinações lineares finitas de funções do tipo X_{Ai}, isto é $S(x) = \sum_{j=1}^{m} \alpha_i X_{Ai}(x)$. Assim, considerando que a convergência em L^2 se mantém com as operações de soma e multiplicação por uma constante, a seqüência $g_n(x) = \sum \alpha_i C_n^{(i)} B_n^{(i)}(x)$ converge a S(x) em L^2.

Teorema T-1:

Seja $g^m(x) = \sum_{j=1}^{m} B_j(x)\kappa_j(x)$ onde,

$\kappa_j(x) = a_j x + b_j$

$$B_j(x) = -\left[\frac{1}{1+\exp(d_j(x-h_j^{(1)}))} - \frac{1}{1+\exp(d_j(x-h_j^{(2)}))}\right], \text{ para } j = 1,\ldots,m$$

Qualquer função L^2-integrável pode ser aproximada (na norma L^2) por funções da forma $g^m(x)$.

Prova:

O que se deseja provar é que o conjunto de funções $\{g^m(x)\}$ com norma $\|.\|_{L_2}$ é um conjunto denso em L^2. Deve-se provar que, para qualquer função em L^2, existe uma seqüência de funções $g_n(x) \in \{g^m(x)\}$ que converge a f(x) em L^2.

Para provar é que o conjunto de funções $\{g^m(x)\}$ com norma $\|.\|_{L_2}$ é um conjunto denso em L^2, basta provar que, para qualquer função em L^2, existe uma seqüência de funções $g_n(x) \in \{g^m(x)\}$ que converge a f(x) em L^2.

De RA-2 temos que as funções simples formam um conjunto denso em L^2;. Assim, existe uma seqüência de funções simples, $S_k(x)$, que converge para f(x) em L^2.

De RA-1, tem-se que as funções simples, por sua vez, podem ser aproximadas por funções do tipo $g^m(x)$. Para cada função simples $S_k(x)$, existe uma seqüência $g_{kn}(x)$ que converge para $S_k(x)$ em L^2:

$g_{11}(x)\ g_{12}(x) \ldots g_{1n}(x) \rightarrow S_1(x)$
$g_{21}(x)\ g_{22}(x) \ldots g_{2n}(x) \rightarrow S_2(x)$
: : : :
$g_{k1}(x)\ g_{k2}(x) \ldots g_{kn}(x) \rightarrow S_k(x)$

Para construir a $g_n(x)$ usa-se a idéia diagonal de Cantor [4]: Escolhendo $g_n(x)=g_{nn}(x)$, então $g_n(x)\rightarrow f(x)$ em L^2.

Note-se que na escolha de $g_n(x)$ só foram necessárias funções $\kappa_n(x)$ constantes. Por tanto, o teorema é válido inclusive nesse caso. A escolha da forma linear $\kappa(x)=ax+b$ (sub-índices omitidos) simplesmente melhora a aproximação e acelera a convergência.

4. Escolha inicial dos parâmetros

A relação entre a entrada e a saída da rede é aprendida através da estimação dos parâmetros que definem as funções de nível de ativação e de aproximação. As funções de nível de ativação podem se superpor em parte do domínio permitindo que um determinado ponto seja estimado através de uma combinação ponderada de mais de uma função de aproximação.

A escolha inicial dos parâmetros $h_1^{(1)}$ e $h_m^{(2)}$ pode refletir um conhecimento a priori do domínio da função. Pode-se ainda utilizar uma heurística de inicialização como a descrita abaixo com a finalidade de acelerar a convergência.

A idéia central da heurística que se segue é fazer uma divisão do domínio obtendo intervalos onde a função é aproximadamente monótona. Para tal, ajusta-se, sobre os dados, um polinômio com grau igual ao número de funções de aproximação que se pretende utilizar. Calculando-se os máximos e mínimos do polinômio, determina-se as regiões do domínio onde a função mantém-se monótona. Para definir os valores a e b associados à função de aproximação linear, para cada intervalo, ajusta-se, uma reta através de uma regressão linear.

Uma heurística automática para calcular a solução inicial:

Dados m, x_i, y_i

Passo 1: Ajustar aos dados o polinômio P(x) de grau m.

Passo2: Calcular os pontos máximos e mínimos de P(x): $m_1, \ldots m_m$

Se $Im(m_i)=0\ \forall i$, escolher $I_1=[\min(x_i)-\varepsilon, m_1-\varepsilon]$, $I_2=[m_1,m_2-\varepsilon],\ldots, I_m=[m_m, \max(x_i)+\varepsilon]$

Caso contrário, se algum m_i é complexo, dividir o intervalo em m intervalos uniformes disjuntos.

Passo 3: Para cada intervalo, ajustar regressão linear nos pares (x_i,y_i), $x_i \in I_i$ para obter a_i e b_i.

5. Resultados numéricos

Nesta seção apresentam-se os experimentos realizados com dados simulados para avaliar o comportamento da metodologia proposta na aproximação de funções.

O primeiro experimento numérico consiste em obter a aproximação da função sen(x)+2. São utilizados quatro conjuntos de dados, correspondendo a 100 pontos gerados através da função sen(x)+2 com diferentes níveis de ruído. O ruído foi gerado adicionando à função geradora sinais Gaussianos com média zero e desvio padrão igual a 0.1; 0.4 e 0.7 respetivamente. A Tabela 1 resume os resultados destes experimentos, reportando o MAPE (*Mean Absolute Percentual Error*). Como o objetivo aqui é medir a capacidade de aproximar a função f(x) utilizando dados com ruidos, a Tabela 1 inclui o MAPE do ruído[1] e além de indicar o MAPE do treinamento calcula-se o MAPE Relativo[2], que estima o MAPE do treinamento logo de eliminar a porção devida ao ruído. Em cada uma destas simulações foram utilizados 3 pares de funções de nível de

[1] MAPE do Ruído = $100 \times \frac{1}{n}\sum_{i=1}^{n}\left|(y_i - f(x))/y_i\right|$

[2] MAPE Relativo= (MAPE Treinamento – MAPE Ruído)/Mape Treinamento

ativação-aproximação. Em todos os experimentos, exceto no experimento 4, foi utilizada a heurística exposta na seção 4 para inicialização no algoritmo de estimação dos parâmetros.

No	Nível de ruído (σ2)	No. de épocas	Mape ruído	Mape Treinamento	Mape Relativo	Mape Generalização
1	0	111	0	0.14	1	0.157
2	0.1	137	4.79	4.64	0.03	1.04
3	0.4	78	16.13	16.80	0.04	4.91
4	0.7	37	30.88	52.36	0.410	7.87

Tabela 1 Resultados das simulações com a função senx+2 com vários níveis de ruído.

As Figuras 4a até 4d ilustram as soluções o ajuste obtido na etapa de treino e generalização para cada experimento.

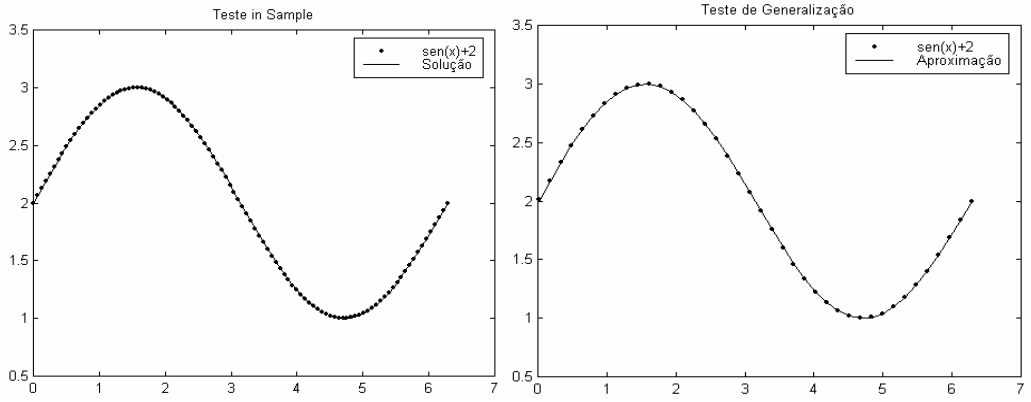

Figura 4a - Nível de ruído: 0

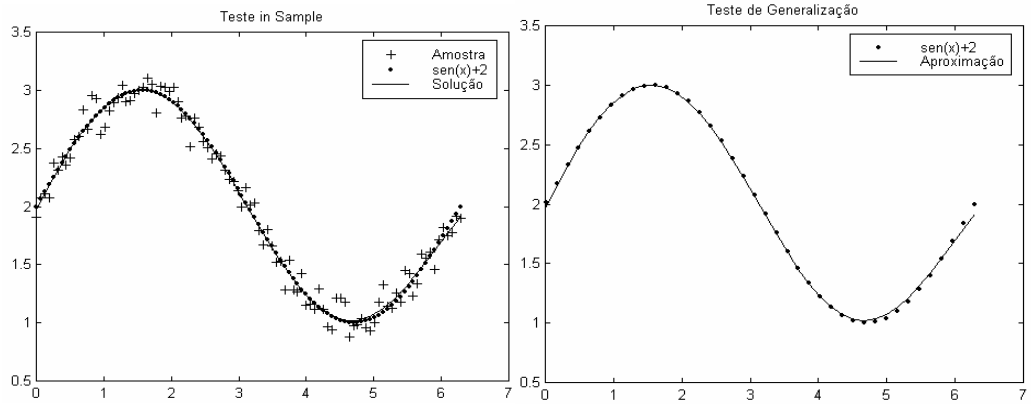

Figura 4b - Nível de ruído: 0.1

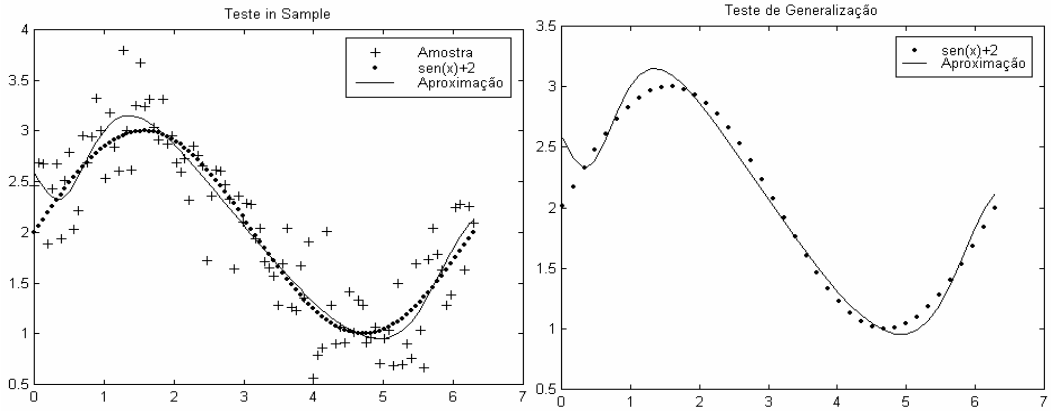

Figura 4c - Nível de ruído: 0.4

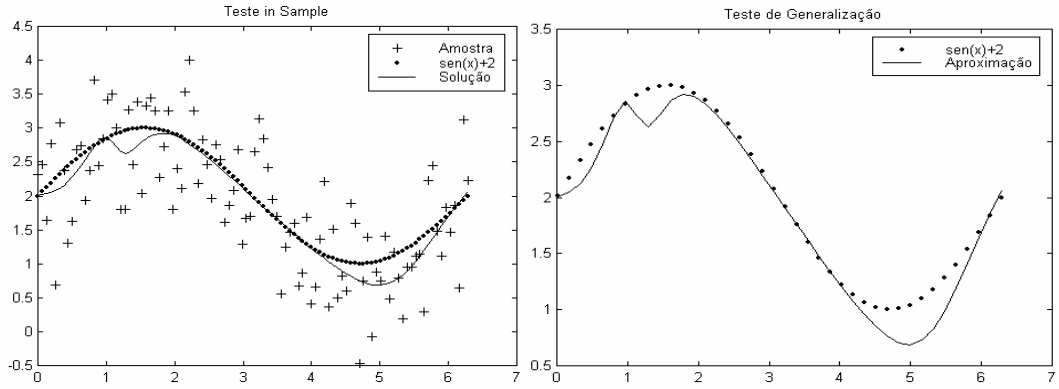

Figura 4d - Nível de ruído: 0.7

Utilizando os mesmos dados do experimento 4, tabela 2, inicializou-se o algoritmo deliberadamente com uma condição inicial diferente da solução esperada. Após 355 iterações obteve-se o Mape de 17 na etapa de treinamento e 3.2 na etapa de generalização. A evolução da convergência neste caso é apresentada na figura 5.

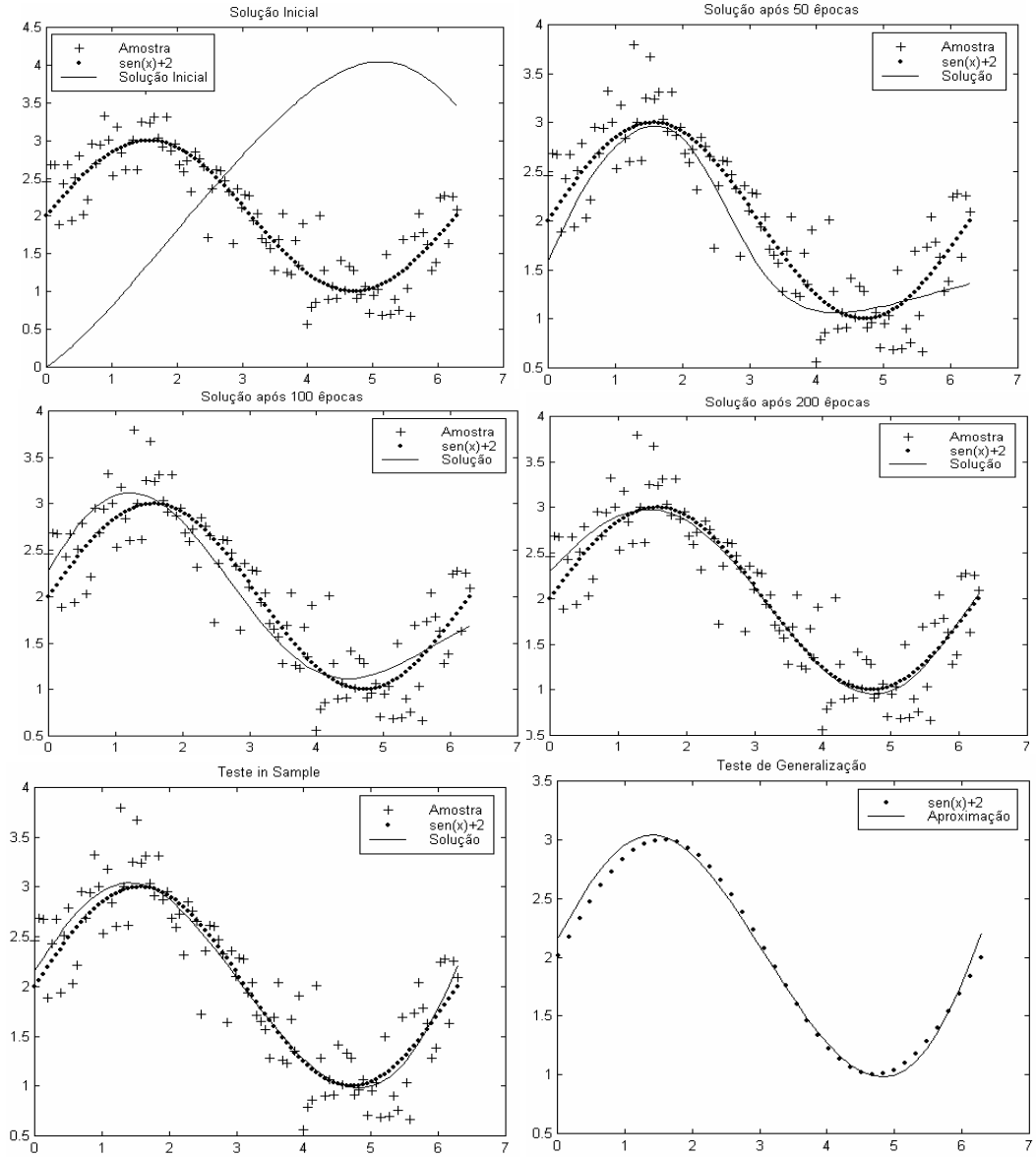

Figura 5 – Evolução da convergência com inicialização sem uso da heurística

6. Uma aplicação em um problema de dados faltantes

Nesta seção apresenta-se uma aplicação da metodologia proposta em um problema de dados faltantes. Foram usados aqui dados reais de uma série de carga elétrica. A série considerada refere-se aos dados de carga minuto a minuto para o dia 1 de Julho de 1999 da concessionária Light. Neste tipo de série o problema de dados faltantes é bastante comum.

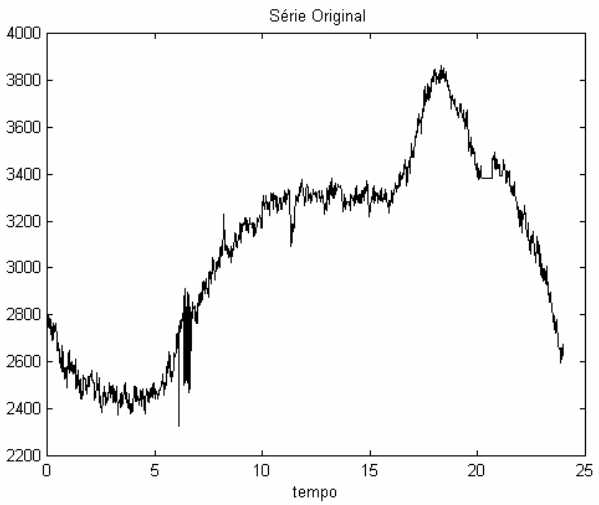

Figura 6 - Dados de carga minuto a minuto para o dia 1 de Julho de 1999

O experimento consiste em retirar da série original uma quantidade de pontos. A série é então recomposta pelo método proposto. Desta forma simula-se o problema de dados faltantes que é muito comum em várias aplicações e em particular no problema de previsão de carga. Em em cada um dos 5 experimentos realizados, foram retirados da série, aleatoriamente de modo uniforme, 5; 10; 20; 30 e 40 % de pontos respectivamente. A Tabela 2 resume o resultado encontrado.

Exp.	pontos retirados	Num. de épocas	Mape treinamento	Mape generalização
1	5%	75	0.99	1.21
2	10%	333	1.01	1.15
3	20%	106	1.06	1.116
4	30%	379	1.00	1.119
5	40%	105	1.07	1.117

Tabela 2 – Simulação de dados faltantes com dados reais de carga elétrica

Em seguida fez-se uma comparação deste método com uma interpolação através de spline cúbico suavizado.

Nota-se (ver Figura 7) que o spline não apresenta bons resultados quando se tem um número considerável de valores faltantes consecutivos. Este resultado era esperado uma vez que o spline fornece soluções com curvatura acentuada distanciando-se do padrão esperado de curva de carga diária.

Utilizou-se neste experimento comparativo o dia 1 de Julho omitindo-se 150 valores consecutivos. Após cerca de 400 épocas chegou-se ao resultado apresentado na figura 7. O MAPE obtido pelo método proposto foi de 1.02 na fase de treino e 2.19 no teste “out of sample”. Já a interpolação spline resultou em um MAPE de 9.31.

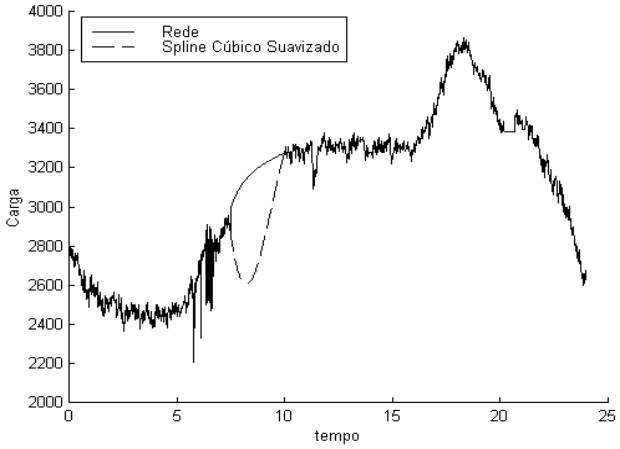

Figura 7 – Comparação do método proposto com uma interpolação através de Spline Cúbico Suavizado

7. Conclusões

Neste artigo foi proposta uma arquitetura conexionista que possui uma interessante capacidade interpolativa. Sua principal característica é a capacidade intrínseca de gerar soluções regulares. Este tipo de arquitetura abre também a possibilidade de interpretabilidade dos resultados uma vez que a localização das funções de nível de ativação pode indicar uma mudança no modelo. Mudanças na estrutura da função geradora dos dados devem se refletir em mudanças de posicionamento e de nível das funções de nível de ativação.

Os resultados simulados em ambientes ruidosos foram particularmente bons. Embora em alguns experimentos uma quantidade razoável de ruído tenha sido adicionada, o método mostrou-se robusto produzindo ótimos resultados, em especial na fase de generalização.

Do ponto de vista de aplicações reais o experimento para dados faltantes apresentou resultados muito animadores, mais uma vez demonstrando a capacidade do método de interpolar dados produzindo soluções regulares e consistentes.

No momento estamos pesquisando as propiedades estatísticas do modelo RNGL: existência e consistência dos estimadores, condições de identificabilidade do modelo. Nestes estudos estão sendo considerados uma formulação multivariada para o modelo e procedimentos numéricos de estimação estão sendo testados para acelerar a convergencia

Referências:

[1] Haykin S. “Neural Networks – A Comprehensive Foundation” , Prentice Hall, second edition, 1999.

[2] Pedroza L.C e Pedreira C.E. “Multilayer Neural Networks and Function Reconstruction by Using a priori Knowledge” International Journal of Neural Systems, Volume 9, number 3, pp 251-256, 1999.

[3] Bartle, R.G. Elements of integration. Wiley. New York, 1966.

[4] De Barra, G. (1974) Introduction to measure theory, New York, Van Nostrand Reinhold .