# MISTURA DE ESPECIALISTAS EM REDES NEURAIS LOCAIS-GLOBAIS

MAYTE FARIÑAS, CARLOS E. PEDREIRA

DEE PUC-RIO
CP. 38063 Rio de Janeiro, CEP 22452-970
E-mails: {mayte ou pedreira}@ele.puc-rio.br

**Resumo**— Este artigo propõe uma nova arquitetura conexionista. O método proposto associa múltiplos especialistas em diferentes partes do domínio utilizando funções de pertinência. Os resultados numéricos obtidos até o momento apontam as potencialidades do esquema proposto.

**Abstract**— In this paper a new connectionist architecture is proposed. The proposed method associates multiple experts in different parts of the domain by using membership functions Numerical results are quite encouraging point to the potentiality of the proposed scheme.

**Keywords**— Mixture of Experts, Neural Networks

## 1 Introdução

Em Pedreira alii(2001 e 2001a) apresentou-se uma nova arquitetura conexionista para abordar o problema de aproximação funcional e interpolação. Propõe-se nesses artigos uma arquitetura conexionista para interpolação, uma forma de emular uma função em um intervalo do domínio onde apenas uma parte dos pontos é conhecida. O algoritmo proposto é capaz de reconstruir a função a partir de estimativas locais ao longo do domínio de interesse, por meio de uma arquitetura não usual. A função é aproximada, por um conjunto de funções de apoio muito simples, muitas vezes lineares e por funções de pertinência.

Neste artigo apresenta-se uma revisão dos resultados anteriores e introduz-se a idéia de misturas de especialistas associados a arquitetura proposta em Pedreira alii (2001 e 2001a).

## 2 Arquitetura Local-Global

Nesta seção apresenta-se uma breve revisão da estrutura de redes locais-globais Pedreira alii (2001 e 2001a). A idéia central é expressar o mapeamento entrada-saída através de uma função composta por partes. A estrutura básica é constituída pela combinação de pares compostos de funções de aproximação e funções de pertinência. As funções de pertinência definem em cada trecho do domínio a participação da função de ativação a ela associada. É possível a ocorrência de sobreposições parciais das funções de pertinência 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 um determinado trecho é dado pelo nível da função de pertinência. Por exemplo, em um trecho onde apenas uma das funções de pertinência assume valor alto, haverá uma dominância da função de aproximação associada à mesma.

Consideremos uma rede com m nós ou neurônios. Seja $\{x_i\}_1^n$ a partição dos dados usada para treinamento. Por simplicidade algébrica e de notação iremos considerar o caso onde $x \in \Re$ (o subscrito de x será omitido), a generalização para o caso onde $x \in \Re^n$ é algebricamente direta. Define-se, para cada ponto x, m funções de pertinência do seguinte modo:

$$B_j(x) = -C_j\left[\frac{1}{1+\exp(d_j(x-h_j^{(1)}))} - \frac{1}{1+\exp(d_j(x-h_j^{(2)}))}\right],$$

onde $C_j$, $d_j$, $h_j^{(1)}$ e $h_j^{(2)}$ j=1, ..., m, são parâmetros a serem ajustados. Note-se que o parâmetro $C_j$ reflete o nível da função de pertinência, enquanto $d_j$ está relacionado à declividade desta função. Os parâmetros $h_j^{(1)}$ e $h_j^{(2)}$ delimitam o setor do domínio no qual a função de aproximação associada a esta função de pertinência é mais ativa. (ver figura 1).

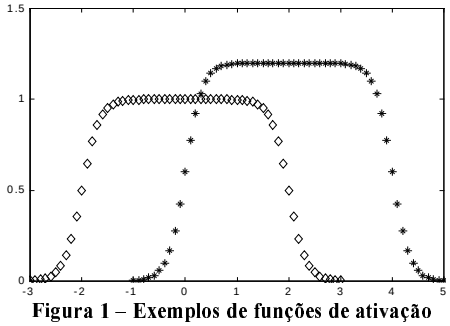



**Figura 1 – Exemplos de funções de ativação**

◊ C=1, d=6; $h_j^{(1)}$ = -2; $h_j^{(2)}$ =2

* C=1.2, d=6; $h_j^{(1)}$ = 0; $h_j^{(2)}$ =4

As funções de aproximação são tipicamente funções lineares ou quadráticas. Embora funções mais complexas possam ser usadas sem prejuízo da estrutura teórica proposta, não parecem trazer contribuição significativa ao modelo. Consideraremos então funções de aproximação lineares:

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

onde $a_j$ e $b_j$ são os parâmetros a serem estimados. Cada nó, ou neurônio, da rede é constituído de um par { função de pertinência ; função de aproximação} (ver figura 2). Então, para cada nó é necessário estimar 6 parâmetros (7 no caso de funções de aproximação quadráticas). Usualmente o número de nós indica a complexidade do modelo.

As entradas são conectadas ao nó onde é efetuado o produto da função de pertinência $B_j(x)$ 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 a saída da rede (veja figura 2). Deste modo a saída do j-ésimo nó é $B_j(x)\ \kappa_j(x)$, e a saída da rede é dada por:

$$g^m(x) = \sum_{j=1}^{m} B_j(x)\kappa_j(x) \qquad (2.1)$$

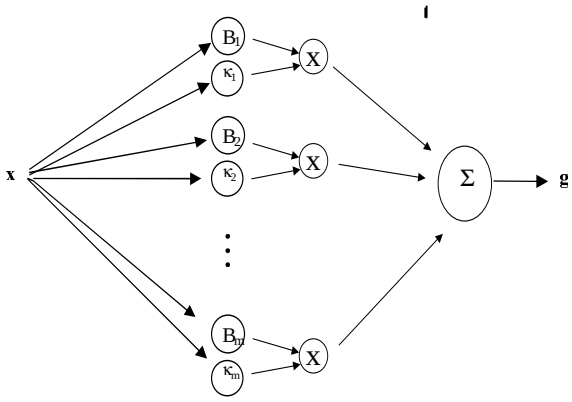


Figura 2 – A arquitetura proposta

Definindo-se $B_{vet}(x) \equiv (B_1(x), B_2(x), \ldots B_m(x))$ e $\kappa_{vet}(x) \equiv (\kappa_1(x), \kappa_2(x), \ldots \kappa_m(x))$ pode-se escrever a saída da rede em forma de produto interno, i.e. $g^m(x) = < B_{vet}(x), \kappa_{vet}(x) >$.

O objetivo central é projetar uma rede cuja saída aproxime uma função alvo, $f(x)$, da melhor forma possível. Define-se então, uma função de erro como uma combinação convexa do quadrado de duas medidas de erro $E_1$ e $E_2$:

$$E \equiv \alpha \sum_{i=1}^{k} E_1^2(x_i) + (1-\alpha)\ E_2^2 \qquad (2.2)$$

onde

$$E_1(x_i) \equiv g^m(x_i) - y(x_i) \text{ e } E_2 \equiv 1 - \sum_{j=1}^{m} C_j \qquad (2.3)$$

O termo $E_1$ está associado a qualidade da aproximação obtida enquanto $E_2$ é usado com a finalidade de manter as funções de pertinência limitadas. Na realidade se está penalizando soluções nas quais o somatório destas funções excede 1. A escolha da unidade como valor limitante não é necessária embora confira mais interpretabilidade aos resultados.

Define-se, para cada neurônio, um vetor de parâmetros $\Im^j \equiv (C_j, d_j, h_j^{(1)}, h_j^{(2)}, a_j, b_j)$ e o objetivo central será encontrar $\Im^j$ que minimize a função de erro E.

Em Pedreira alii (2001) e (2001a) enuncia-se e prova-se o teorema que garante que qualquer função $L^2$-integravél pode ser aproximada por funções da forma $g^m(x)$. Os valores iniciais dos parâmetros $h_1^{(1)}$ e $h_m^{(2)}$ podem refletir um conhecimento a priori do domínio da função; além disso, propõe-se uma heurística que pode ser utilizada para a escolha da solução inicial com a a finalidade de acelerar a convergência. Também em Pedreira alii (2001 e 2001a) apresenta-se uma comparação dos resultados obtidos com estruturas clássicas do tipo MLP e RBF.

## 3 Mistura de Especialistas

A idéia de utilizar uma mistura de especialistas para realizar mapeamento complexo de funções foi primeiramente discutido por Jacobs, Jordan, Nowlan e Hinton (1991). O desenvolvimento deste modelo encontra motivação na proposta descrita por Nowlan (1990) abordando a adaptação competitiva no aprendizado não-supervisionado como uma tentativa de ajustar uma mistura de distribuições de probabilidades simples (tais como Gaussianas) a um conjunto de pontos. É também influenciada pelas idéias desenvolvidas por Jacobs (1990), utilizando uma arquitetura similar mas com uma função de custo diferente.

Em Nowlan & Hinton (1991) compara-se a performance deste tipo de arquitetura, (a sugerida por Jacobs *alii* (1991)) frente a uma rede simples de backpropagation num problema de reconhecimento de voz. Ressalta-se os resultados interessantes em termos de decomposição de especialistas e as simulações mostram a superioridade em termos de propriedades de generalização frente a estrutura tradicional com treinamento por retro-propagação. Em termos gerais, o problema de mistura de especialistas, pode ser apresentado como (Jacobs *alii*, 1991)):

Supõe-se que, segundo o conhecimento do problema, o conjunto de casos de treinamento pode ser dividido de forma natural em subconjuntos que se correspondem a subtarefas diferentes. As interferências (erros de ajuste) podem ser reduzidas utilizando um sistema composto por vários especialistas e uma rede '*gatting'*, que decide qual especialista deve ser utilizado para cada caso.

Cada especialista pode ser uma rede neural e todos recebem as mesmas entradas e tem o mesmo número de saída. A rede '*gatting'* é também uma rede backpropagation e poderia receber outro tipo de entradas. A partir da saída desta rede obtém-se as probabilidades de selecionar cada especialista.

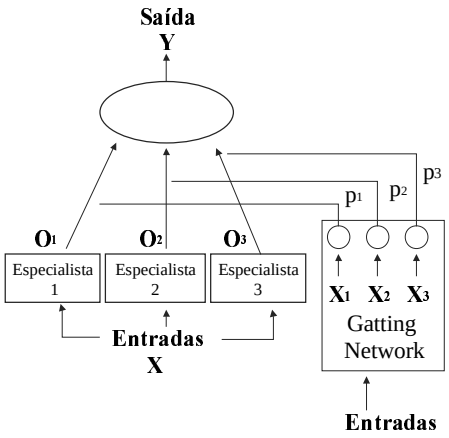



Figura 3 – Múltiplos Especialistas

Desde os primeiros artigos, ressalta-se o fato que numericamente uma destas probabilidades será muito próxima a 1 enquanto que as outras anulam-se.

A idéia proposta no presente artigo consiste em, ao invés de se trabalhar com as probabilidades P, que são a saída da rede 'gatting', utilizar funções de pertinência, como as usadas em Pedreira *alii* (2001) e (2001a) para propiciar uma transição suave de um especialista e obter uma "boa mistura".

Na seção seguinte são apresentados resultados numéricos preliminares. O intuito destes exemplos é avaliar o funcionamento do método, com exemplos simples e analisar como é feito numericamente a divisão dos especialistas.

## 3 Resultados Numéricos

Nesta seção serão apresentados três tipos de problemas: Problemas com 2 e 3 especialistas, onde considera-se que a resposta de cada especialistas se ajusta a um polinômios de grau 1 e 2 (Retas e Parábolas). Em seguida passa-se para um problema mais complexo onde as aproximações de grau 1 e 2 não são mais válidas e um dos especialistas passa a ser uma rede neural.

Os dados para o treinamento foram simulados e nenhum ruído foi adicionado aos dados. Nos exemplos das seções 4.1 e 4.3 (2 e 3 especialistas, polinômios de grau 1 e 2) a primeira figura (Figuras 1a e 2a) representa a situação hipotética que gerou os dados. A partir desta, geram-se os dados ($x$,f($x$)) utilizados no treinamento. Partindo de uma solução inicial razoável , treina-se a rede até obter a convergência, com relação à função de erro considerada (Eq. 2.2-2.3). O algoritmo de otimização utilizado foi o Levenberg Marquardt.

Os dados utilizados na etapa de generalização foram gerados através de uma distribuição uniforme no intervalo considerado, utilizando uma quantidade correspondente ao 40% do total de dados na etapa de treinamento. Em todas as tabelas apresenta-se como medidas de erros, o EQM (Erro Quadrático médio) e o MAPE(Mean Absolute Percentual Error).

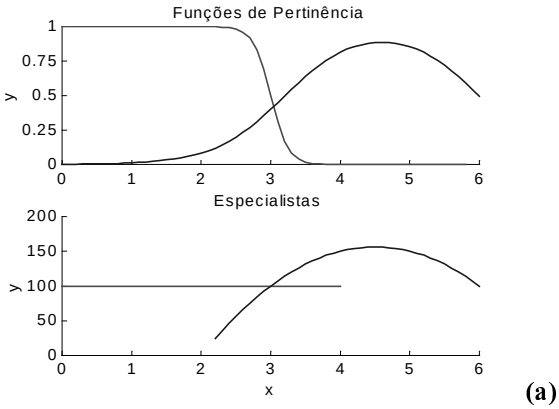
### *4.1 Exemplos com 2 Especialistas (Polinômios)*

Nestes dois exemplos, considera-se que a situação que gerou os dados é a representada nas figuras 4-a e 5-a . O conjunto de treinamento é formado a partir desta situação utilizando $x$=[0:.1:6] (61 pontos) no exemplo 1 e $x$=[0:.1:10] (101pontos) no exemplo 2.

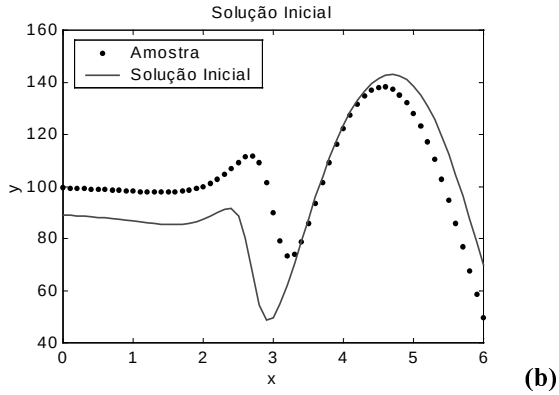
A tabela 1 resume os resultados obtidos no treinamento e generalização para estes dois exemplos.

Tabela 1. Resultados do ajuste. Mistura de dos especialistas (reta e parábola)

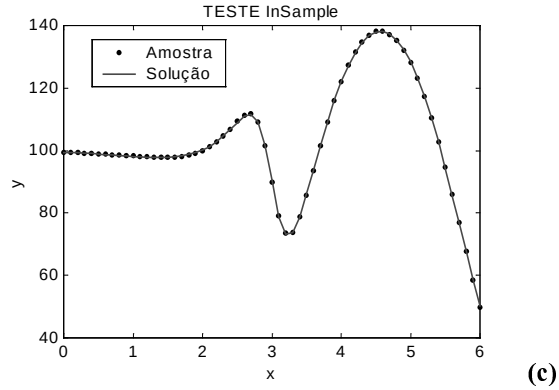
| Ex | No. de épocas | Treinamento | | | Generalização | |
|---|---|---|---|---|---|---|
| | | F.Erro | EQM | MAPE | EQM | MAPE |
| 1 | 136 | 0.0344 | 0.0406 | 0.1731 | 0.0566 | 0.2052 |
| 2 | 75 | 0.1088 | 0.0377 | 0.5198 | 0.0348 | 0.5327 |



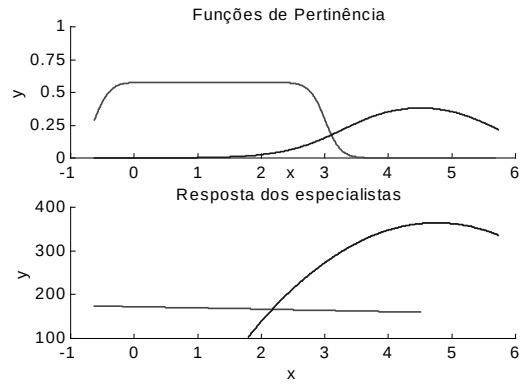

(a)




(b)

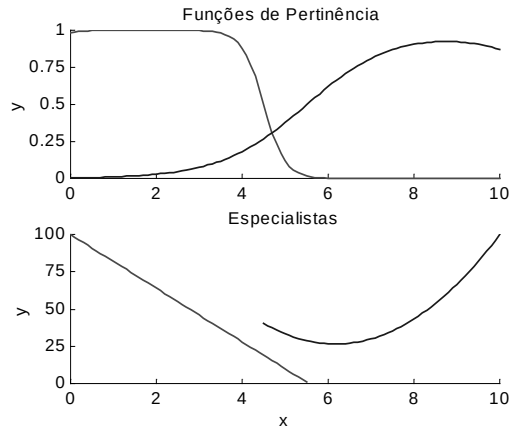



(c)

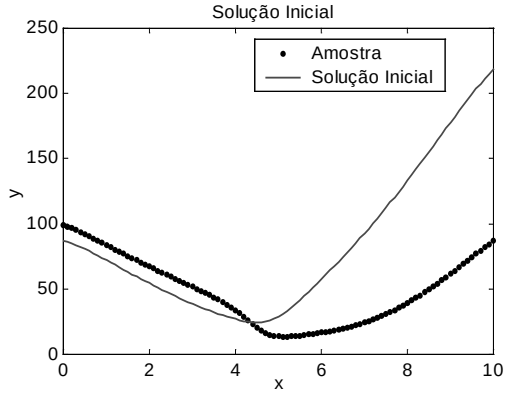



**(d)**

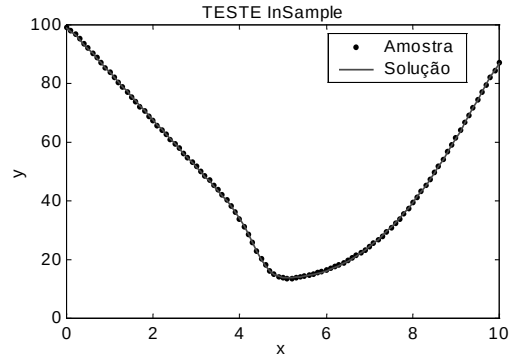
Figura 4 - Mistura de 2 Especialistas. Exemplo1.
a) Simulação dos Dados, partindo das funções de pertinência e Especialistas. b) Amostra e Solução inicial. c) Ajuste in Sample d) Solução obtida em termos de especialistas e funções de pertinência



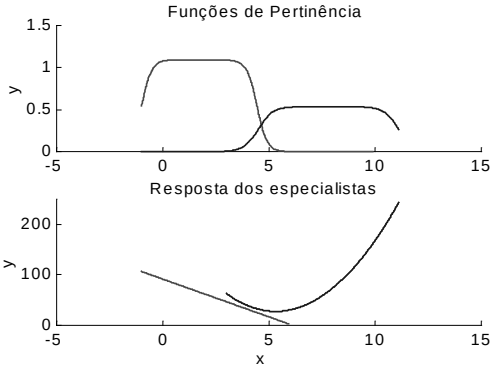

**(a)**




**(b)**




**(c)**




**(d)**

Figura 5 - Mistura de 2 Especialistas. Exemplo 2. a) Simulação dos Dados, partindo das funções de pertinência e Especialistas. b) Amostra e Solução inicial. c) Ajuste in Sample d) Solução obtida em termos de especialistas e funções de pertinência

Os resultados obtidos para dois especialistas, utilizando-se polinômios de grau 1 e 2 são ótimos. O algoritmo converge em relativamente poucas iterações (épocas) com excelentes valores no treinamento. Os valores obtidos na fase de generalização podem ser considerados bons.

### *4.2 Exemplos com 2 Especialistas (Polinômios e MLP)*

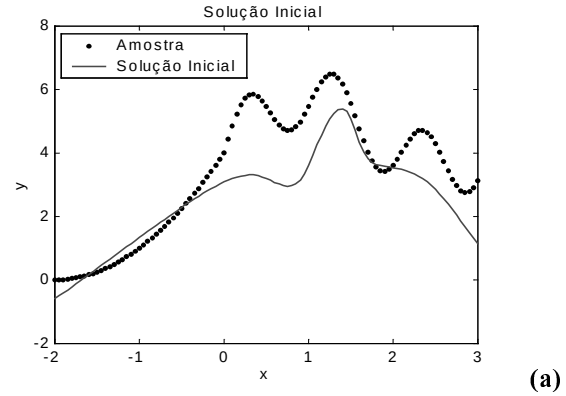
Problemas que envolvem representações funcionais mais complexas, necessitam de especialistas mais sofisticados. Por exemplo, se quisermos simular a função:

$$f(x) = \begin{cases} (x+2)^2 & x \in [-2,0] \\ sin(x) + sin(2x) + sin(6x) + 4 & x \in [0,3] \end{cases}$$

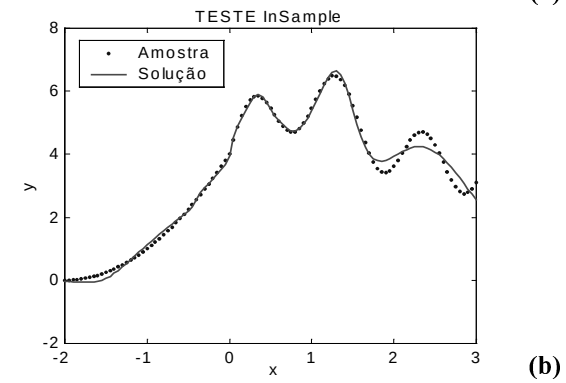
é razoável, utilizar dois especialistas, aquele que tenteacaptar o comportamento parabólico e outro como uma rede neural, para o intervalo onde a função tem uma forma funcional mais complexa. Utilizamos aqui uma rede neural com uma única camada oculta, com 5 e 6 neurônios. A solução inicial obtém-se ajustando os especialistas separadamente. A tabela 2 resume os resultados obtidos neste caso, onde foram utilizados para o treinamento conjuntos ($x$,f($x$)) com $x$=[-2:.05:3] (101 pontos).

Tabela 2. Resultados do ajuste. Mistura de dos especialistas [Pol(1) e MLP(n)]

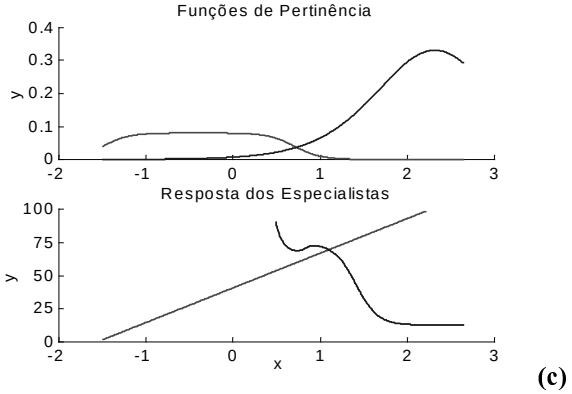
| No | No. de épocas | Treinamento | | | Generalização | |
|---|---|---|---|---|---|---|
| | | F.Erro | EQM | MAPE | EQM | MAPE |
| 1 | 500 | 0.0278 | 0.0348 | 0.0278 | 0.0266 | 4.9673 |
| 2 | 127 | 0.0022 | 0.0028 | 1.3765 | 0.0022 | 0.9236 |




(a)



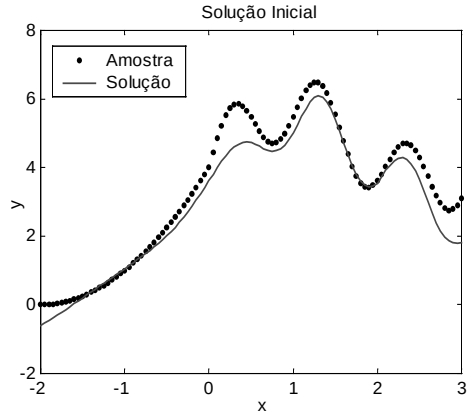

(b)



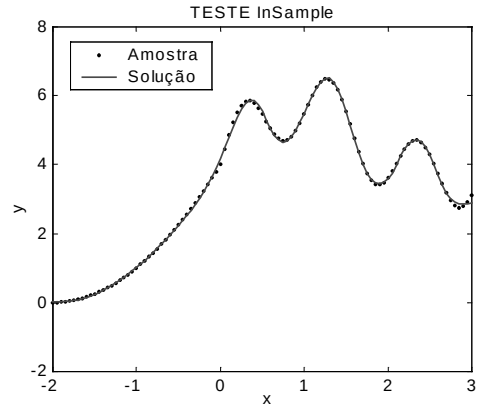

(c)

Figura 6 - Mistura de 2 Especialistas. Pol(1) + MLP(5). a)Amostra e Solução inicial. c) Ajuste in Sample d) Solução obtida em termos de especialistas e funções de pertinência.

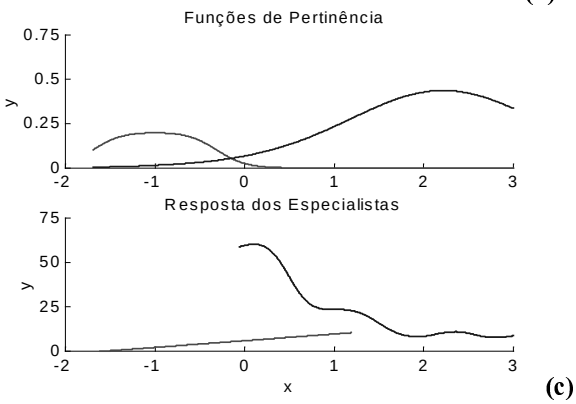
Nota-se que não é possível melhorar os resultados do treinamento sem perder capacidade de generalização. Com 5 neurônios perdemos um pouco no ajuste no final do intervalo (Figura 6c); isto indica a necessidade de alterar a arquitetura. Com 6 neurônios consegue-se um ajuste excelente após 127 iterações.




(a)




(b)




(c)

Figura 7 - Mistura de 2 Especialistas. Pol(1) + MLP(6). a)Amostra e Solução inicial. c)Ajuste in Sample d) Solução obtida em termos de especialistas e funções de pertinência

Vale a pena observar como o método consegue alocar cada especialista no seu intervalo, obtendo uma mistura na intercepção que suaviza a transição de um especialista a outro.

### *4.3 Exemplos com 3 Especialistas (Polinômios)*

Nesta seção apresentamos dois exemplos com 3 especialista, onde todos são polinômios. As figuras 8a e 9a mostram a situação hipotética que gerou os dados. O conjunto de treinamento é formado a partir desta situação utilizando $x$=[0:.1:30] (301 pontos).

Tabela 3. Resultados do ajuste. Mistura de dos especialistas [Reta, Parábola, Reta]

| | No. de | Treinamento | | | Generalização | |
|---|---|---|---|---|---|---|
| No | épocas | F.Erro | EQM | MAPE | EQM | MAPE |
| 1 | 133 | 0.0031 | 0.6586 | 0.0110 | 0.0265 | 1.2868 |
| 2 | 120 | 0.0408 | 0.0164 | 1.1157 | 0.0037 | 0.5468 |

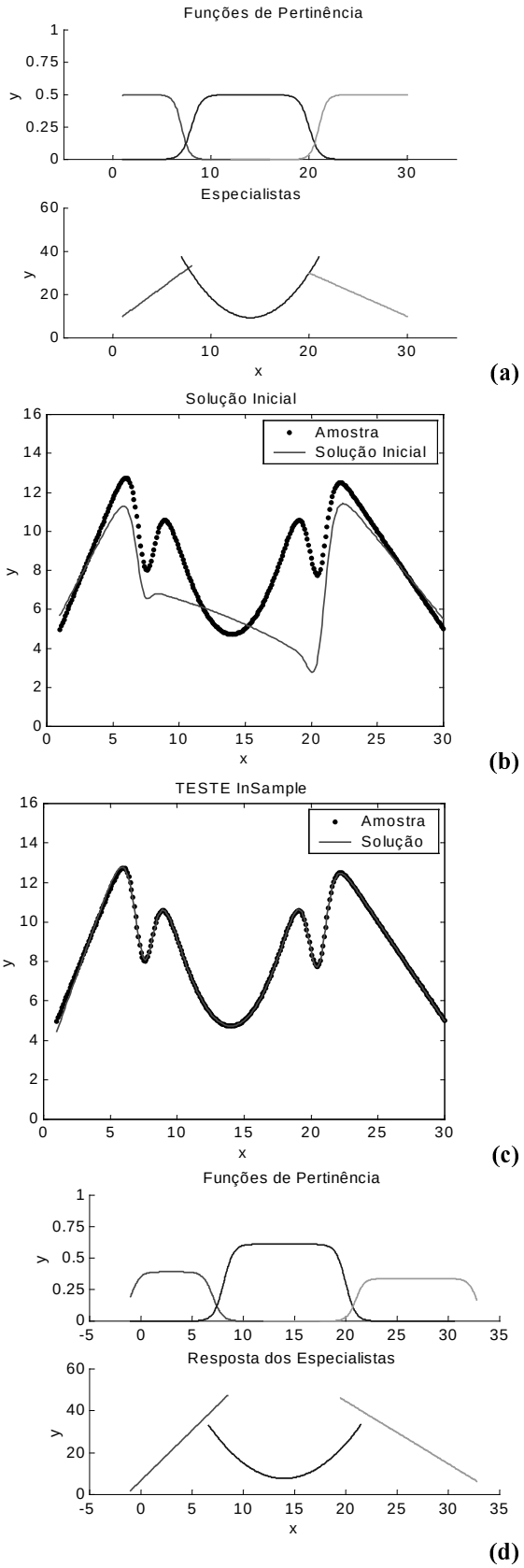



Figura 8 - Mistura de 3 Especialistas. Exemplo 1
a) Simulação dos Dados, partindo das funções de pertinência e os Especialistas. b) Amostra e Solução inicial. c) Ajuste in Sample. d) Solução obtida em termos de especialistas e funções de pertinência

## Conclussões

Neste artigo propõe-se uma solução de mistura de especialistas baseada nas idéias de Redes Globais Locais para reconstituição funcional. Resultados numéricos preliminares apresentam soluções bastante satisfatórias, enfatizando a potencialidade do método proposto. Este tipo de arquitetura abre também a possibilidade de interpretabilidade dos resultados uma vez que a localização das funções de pertinência 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 pertinência.

Encontra-se em investigação a utilização do método proposto em problemas mais complexos e com dados reais.

## Referências Bibliográficas


Haykin, S(1999) Neural Networks – A Comprehensive Foundation , 2nd. Edition, Prentice Hall, New Jersey.

Jacobs, R.A(1990). Task Decomposition Trough Computation in a Modular Connectionist Architecture. Ph.D. Thesis, University of Massachusetts, Massachusetts.

Jacobs, R.A., Jordan, M.I,. Nowlan, S.J and Hinton,G.E. (1991). Adaptative Misture of local Expert Neural Computation, vol. 3, pp. 79-87

Lawrence, S., Lee Giles, C., Tsoi, A.C.(1996) What size neural network gives optimal generalization ? Convergences properties of Backpropagation. Technical Report UMIACS-TR-96-22 and CS-TR-3617, University of Maryland, Maryland.

Nowlan, S.J.(1990). Maximum likelihood competitive learning Advances in Neural Information. *Processing Systems*, vol. 2, pp. 574-582, San Mateo, CA:Morgan Kaufmann.

Nowlan, S.J & Hinton (1991). Evaluation of Adaptative Mixture of Competing Experts. *Advances in Neural Information. Processing Systems*, vol. 3, pp. 774-780, San Mateo, CA: Morgan Kaufmann.

Pedreira, C. E., Pedroza, L.C. e Fariñas, M.(2001) Redes Neurais Locais-Globais – Uma Aplicação ao Problema de Dados Faltantes. *Proceedings V Congresso Brasileiro de Redes Neurais*, Rio de Janeiro, pp. 433–438,

Pedreira, C.E., Pedroza, L. C. and Fariñas, M.(2001a) Local-Global Neural Networks For Interpolation. *Proceeding of ICANNGA 2001*- Praga, pp.55-58

Pedroza L.C and Pedreira C.E. (1999). Multilayer Neural Networks and Function Reconstruction by Using a priori Knowledge. *International Journal of Neural Systems*, Vol 9, No. 3, pp 251-256.