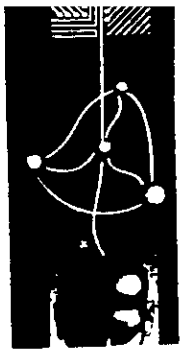

1° Congresso Brasileiro de Redes Neurais

Escola Federal de Engenharia de Itajuba
Itajuba, 24 a 27 de outubro de 1994

REDES NEURAIS COM DENSIDADE DE PROBABILIDADE UNIFORME NA ENTRADA-SAIDA

C.E.Pedreira and E.Parente
Dept. de Engenharia Elétrica
PUC-Rio C.P. 38063
22452-970 Rio de Janeiro
pedreira@ele.puc-rio.br

RESUMO: Neste artigo introduz-se um novo algorítimo de aprendizado que permite ter entradas as quais se atribui distribuições com densidade de probabilidade uniforme. A rede fornece na saida tambem uma distribuição com densidade de probabilidade uniforme. Este enfoque possui algumas vantagens relevantes com relação ao algorítimo de retropropagação original, sendo a principal destas a possibilidade de se operar com a falta eventual de alguma entrada sem a necessidade de um novo treinamento. Esta técnica é particularmente interessante em problemas com incertezas nas entradas e também quando se deseja estabelecer uma faixa de tolerância para a saída da rede. A função de custo proposta procura ao mesmo tempo incluir o intervalo de saída no intervalo alvo além de minimizar a distância entre seus centros. Simulações numéricas preliminares foram realizadas com o objetivo de ilustrar os resultados teóricos apresentados.

INTRODUÇÃO

O algoritimo de treinamento por retropropagação conforme originalmente proposto em [1] restringe sua aplicação a entrada com valores reais. Esta restrição leva a algumas limitações em uma classe importante de aplicações. O problema mais sério é o que se refere a ausência de parte dos atributos de entrada quando da fase de operação. Suponha-se por exemplo que após uma longa seção de treinamento em uma rede com n entradas, um destes dados não se encontra disponivel. Como proceder então? Retreinar uma nova arquitetura com n-1 entradas, normalmente não seria uma boa solução. Este problema de falta de atributos foi por algum tempo uma questão aberta do ponto vista teórico que possui grande relevância no que tange as aplicações. O uso de aritmética de intervalos [2] como ferramenta para solucionar esta questão foi originalmente proposto por Ishibuchi et al. em [3] dentro de um contexto de classificação de dois grupos, e de forma mais geral por Pedreira e Parente em [4]. O contexto exposto neste artigo propõe, no lugar de entradas compostas de números reais, entradas que assumem valores de intervalos reais. Deste modo, a rede estará sendo alimentada por intervalos reais fechados, ou equivalentemente, por distribuições com densidade de probabilidade uniforme. No caso exposto anteriormente, de não se ter acesso a algumas das entradas, se deverá alimentar a rede com o intervalo de máxima largura. Assim sendo, se estará informando à rede a máxima incerteza para o parâmetro em questão. Vale notar que qualquer número real pode ser representado como um intervalo real fechado e portanto, no caso de todas as entradas assumirem valores em $\Re$, o algoritimo aqui proposto se reduz ao esquema clássico de retropropagação. Diferentemente desta técnica tradicional e do algoritimo

proposto em [4], o alvo da rede é também um intervalo real fechado. Atraves desta modificação se torna possivel pré estabelecer a tolerância requerida pela aplicação em questão. Este enfoque parece ser bastante mais interessante do que, como é feito na tecnica tradicional, impor tolerância zero, o que muitas vezes não é realistico no que concerne a aplicações. Note-se que no enfoque aqui proposto é possivel controlar a precisão exigida em cada uma das variáveis de saída de modo independente. Questões ligadas a problemas com super treinamento e capacidade de generalização deste novo algoritimo permanecem em aberto e fogem ao escopo do presente artigo.

PRELIMINARES

Considere-se, por simplicidade de notação, uma rede do tipo *feedforward* com uma camada oculta e apenas uma unidade de saída. Assume-se que cada unidade da camada de entrada recebe, ao invés de um número real, um intervalo.

Notação

Considere-se a apresentação do padrão p. Sejam ε_{ip}^{L} e ε_{ip}^{U} , $i = 1,2,.....n$ os limites inferior e superior da i-ésima entrada respectivamente. o_{jp}^{L} e o_{jp}^{U} , $j = 1,2,....m$ denotam os limites inferior e superior da saída da j-ésima unidade da camada oculta, e O_p^L, O_p^U são os limites inferior e superior da unidade de saida da rede. Os n x m pesos de entrada são representados por w_{ji} , equanto que v_j são os m pesos que ligam a camada oculta à unidade de saida. A saída alvo, para cada padrão p tem como limites t_p^L e t_p^U. Seja $\overline{Y}$ o intervalo que tem como limites Y^L e Y^U.

Definições Básicas

Define-se a combinação linear dos intervalos de entrada, quando da apresentação do padrão p, como:

$$\overline{net}_{jp} \equiv \sum_{i=1}^{n} w_{ji}\bar{\varepsilon}_{ip} + \theta_j, \text{ para cada } j = 1,2,.....m \qquad (1)$$

enquanto que a combinação linear das saídas das unidades da camada oculta é dada por:

$$\overline{NET}_p \equiv \sum_{j=1}^{m} v_j\bar{o}_{jp} + \theta \qquad (2)$$

onde θ_j e θ são os termos polarizadores.
Nosso problema pode ser então estabelecido como: calcular os pesos da rede tal que a função de custo $E \equiv \sum_p e_p$ é minimizada. O custo devido ao padrão p é definido como:

$$e_p = \left(\frac{O_p^L+O_p^U}{2} - \frac{t_p^L+t_p^U}{2}\right)^2 + g(t_p^L - O_p^L) + g(O_p^U - t_p^U) \qquad (3)$$

onde,

$$g(x) = \begin{cases} 0, & x < 0 \\ x^2, & 0 \le x < \alpha \\ 2\alpha(x-\alpha)+\alpha^2, & x \ge \alpha \end{cases}$$

Esta função de custo reflete um problema de compromisso. Sua minimização leva à coincidência dos centros do intervalo de saída e do intervalo alvo ao mesmo tempo em que procura a inclusão do intervalo de saída no intervalo alvo. Este compromisso é balanceado pelo parâmetro α. O objetivo central é então definir um algoritimo tipo gradiente que modifique os pesos w_{ji} e v_j para i=1,2,...n , j=1,2,...m tal que a função de custo (3) seja minimizada.

Resultados Preliminares

Seja $I(\Re)$ o conjunto de todos os intervalos reais fechados. Note que qualquer número $x \in \Re$ pode ser considerado um elemento especial $[x,x]$ de $I(\Re)$. Sejam $\overline{X}, \overline{Y} \in I(\Re)$. As seguintes operações binárias podem ser calculadas como segue [2]:

$$\overline{X} + \overline{Y} = [X^L + Y^L, X^U + Y^U] \qquad (4)$$

$$\overline{X} - \overline{Y} = [X^L - Y^U, X^U - Y^L] \qquad (5)$$

$$\overline{X} \cdot \overline{Y} = [\min(X^L Y^L, X^L Y^U, X^U Y^L, X^U Y^U), \max(X^L Y^L, X^L Y^U, X^U Y^L, X^U Y^U)] \qquad (6)$$

$$\overline{X} \div \overline{Y} = [X^L, X^U] \cdot [1/Y^U, 1/Y^L] \qquad (7)$$

$$e^{\overline{X}} = [e^{X^L}, e^{X^U}] \qquad (8)$$

Assume-se que $0 \in \overline{Y}$ em caso de divisão (assumir $Y^U \neq 0$ e $Y^L \neq 0$ não e suficiente). Note-se que $I(\Re)$ é fechado sob as operações (4) - (8). Desta propriedade resulta que a aplicação do algoritimo proposto neste artigo implicará em que a saída da rede seja sempre um intervalo fechado. Além disso, pode-se provar [2] que se tem: (i) comutatividade e associatividade para adição e multiplicação; (ii) [0,0] e [1,1] são os únicos elementos neutros com relação a adição e multiplicação, respectivamente; e (iii) $I(\Re)$ não tem divisores zero.

Seja $\overline{Y} \in I(\Re)$ um intervalo pontual, i.e., $\overline{Y} = [y, y]$ onde $y \in \Re$. Tem-se então de (6) que:

$$\overline{X} \cdot \overline{Y} = [\min(Y^L y, X^U y), \max(Y^L y, X^U y)] \quad \forall \overline{X} \in I(\Re)$$

e portanto

$$\begin{aligned} y \cdot \overline{X} &= [yX^L, yX^U] \quad \text{para} \quad y \geq 0 \quad \forall \overline{X} \in I(\Re) \\ y \cdot \overline{X} &= [yX^U, yX^L] \quad \text{para} \quad y < 0 \quad \forall \overline{X} \in I(\Re) \end{aligned} \tag{9}$$

RESULTADOS PRINCIPAIS

Pode-se agora estabelecer um procedimento para calcular as mudanças nos pesos que minimizarão a contribuição do padrão p, e_p, na função de custo E. Vamos considerar a seguinte regra Delta:

$$\begin{aligned} \Delta_p v_j(s+1) &= \eta(-\partial e_p / \partial v_j) + \mu \Delta_p v_j(s) \\ \Delta_p w_{ji}(s+1) &= \eta(-\partial e_p / \partial w_{ji}) + \mu \Delta_p w_{ji}(s) \end{aligned} \tag{10}$$

onde η e μ representam os parâmetros de aprendizado e momento, respectivamente. A dificuldade restante está relacionada ao cálculo das derivadas parciais de e_p. De (1),(2) e (9) obtem-se:

$$\overline{net}_{jp} = \sum_{i=1}^{n} \left[w_{ji}\varepsilon_{ip}^{A}, w_{ji}\varepsilon_{ip}^{B} \right] + \theta_j \qquad \forall j = 1, 2, \ldots, m \ ,$$

e

$$\overline{NET}_p = \sum_{i=1}^{m} \left[v_j o_{jp}^{C}, v_j o_{jp}^{D} \right] - \theta$$

onde

A = L , B = U se $w_{ji} \geq 0$,
e A = U , B = L caso contrário,

C = L , D = U se $v_j \geq 0$,
e C = U , D = L caso contrário.

Com relação as unidades da camada oculta tem-se que:

$$\overline{o}_{jp} = f(\overline{net}_{jp})$$

e para a unidade de saída:

$$\overline{O}_p = f(\overline{NET}_p)$$

onde a função de ativação: $f: I(\Re) \to I(\Re)$ é uma logistica generalizada, i.e,

$$f(\overline{X}) = 1/(1 + \exp(-\overline{X})) \quad \forall \overline{X} \in I(\Re).$$

Note que, esta função logística generalizada pode ser definida por causa de (4),(7) e (8).

Define-se na sequência os seguintes parâmetros auxiliares:

$\Phi \equiv L$ se $w_{ji} \geq 0$, $\Phi \equiv U$ caso contrário

$\Psi \equiv U$ se $w_{ji} \geq 0$, $\Psi \equiv L$ caso contrário

$K \equiv 1$ se $v_j \geq 0$, $K \equiv 0$ caso contrário

Aplicando as operações básicas de aritimética de intervalos (4) - (7), e a regra da cadeia obtem-se:

(i) Para os pesos "entrada-oculta" :

$$\frac{\partial e_p}{\partial w_{ji}} =$$

$$\frac{\partial e_p}{\partial O_p^L}\frac{\partial O_p^L}{\partial NET_p^L}\left(\frac{\partial NET_p^L}{\partial o_{jp}^L}\frac{\partial o_{jp}^L}{\partial net_{jp}^L}\frac{\partial net_{jp}^L}{\partial w_{ji}} + \frac{\partial NET_p^L}{\partial o_{jp}^U}\frac{\partial o_{jp}^U}{\partial net_{jp}^U}\frac{\partial net_{jp}^U}{\partial w_{ji}}\right) +$$

$$\frac{\partial e_p}{\partial O_p^U}\frac{\partial O_p^U}{\partial NET_p^U}\left(\frac{\partial NET_p^U}{\partial o_{jp}^L}\frac{\partial o_{jp}^L}{\partial net_{jp}^L}\frac{\partial net_{jp}^L}{\partial w_{ji}} + \frac{\partial NET_p^U}{\partial o_{jp}^U}\frac{\partial o_{jp}^U}{\partial net_{jp}^U}\frac{\partial net_{jp}^U}{\partial w_{ji}}\right),$$

e então

$$\frac{\partial e_p}{\partial w_{ji}} = \left(\frac{\alpha}{2}\right)(2t_p - O_p^U - O_p^L))(\xi + \varpi) + \beta(\varpi - \xi) \tag{11}$$

onde

$$\xi \equiv K(1-O_p^L)O_p^L v_j o_{jp}^L (1-o_{jp}^L)\varepsilon_{ip}^{\Phi} +$$

$$+(1-K)(1-O_p^L)O_p^L v_j o_{jp}^U (1-o_{jp}^U)\varepsilon_{ip}^{\Psi}$$

$$\varpi \equiv (1-K)(1-O_p^U)O_p^U v_j o_{jp}^L (1-o_{jp}^L)\varepsilon_{ip}^{\Phi} +$$
$$+K(1-O_p^U)O_p^U v_j o_{jp}^U (1-o_{jp}^U)\varepsilon_{ip}^{\Psi})$$

(ii) Para os pesos "oculta - saída" :

$$\frac{\partial e_p}{\partial v_j} = \frac{\partial e_p}{\partial O_p^U}\frac{\partial O_p^U}{\partial NET_p^U}\frac{\partial NET_p^U}{\partial v_j} + \frac{\partial e_p}{\partial O_p^L}\frac{\partial O_p^L}{\partial NET_p^L}\frac{\partial NET_p^L}{\partial v_j}$$

$$= (\frac{\alpha}{\quad}(2t_p - O_p^U - O_p^L)(\pi+\sigma) + \beta(\sigma-\pi) \qquad (12)$$

onde

$$\pi \equiv o_{jp}^{\Psi} O_p^U (1-O_p^U) \quad \text{e} \quad \sigma \equiv o_{jp}^{\Phi} O_p^L (1-O_p^L)$$

Aplicando (11) e (12) em (10) pode-se treinar apropriadamente a Rede Neural de modo a minimizar a função de custo proposta.

RESULTADOS NUMÉRICOS PRELIMINARES

Os resultados numéricos aqui apresentados tem como objetivo ilustrar a técnica proposta, sem qualquer intenção de fazer uma análise detalhada do comportamento computacional desta metodologia ou de propor um novo sistema de diagnóstico inteligente. Simulou-se um sistema hipotético de apoio a decisão médica supondo-se que quatro exames são aplicados a cada um dos pacientes, sendo o primeiro destes mais relevante para a elaboração do diagnóstico do que os demais. A saída da rede indicará um índice que servirá de apoio a decisão médica. Assume-se que "0" e "1" indicam diagnóstico negativo e positivo respectivamente. Deste modo um indicador próximo a "1" apontará um paciente com boa saúde. Por outro lado, uma entrada igual a "1" reflete um exame cujo resultado foi positivo.

Em todos os exemplos que seguem a Rede Neural tem arquitetura (4:2:1). O seguinte conjunto de treinamento foi apresentado:

TABELA I - Padrões de treinamento

Apres.	Entrada				Alvo
	1	2	3	4	
1	1	1	1	1	1
2	1	1	1	0	1
3	1	1	0	0	1
4	1	0	0	0	0
5	1	1	0	1	1
6	1	0	1	0	1
7	1	0	1	1	1
8	1	0	0	1	1
9	0	0	0	0	0
10	0	1	0	0	0
11	0	1	1	0	0
12	0	1	1	1	1
13	0	0	1	1	0
14	0	0	0	1	0
15	0	1	0	1	0
16	0	0	1	0	0

Depois de uma sessão de treinamento de aproximadamente 4000 interações (Figura 1) obteve-se como saida correspondente às 16 apresentações de entrada os valores mostrados na tabela II.

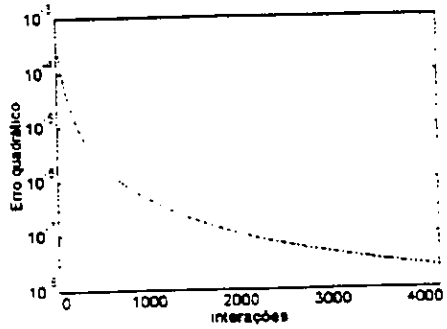

Figura 1

TABELA II - Saida da rede treinada

Apres.	Saida	Apres.	Saida
1	0,9998	9	0,0007
2	0,9996	10	0,0004
3	0,9876	11	0,0146
4	0,0190	12	0,9833
5	0,9996	13	0,0146
6	0,9996	14	0,0007
7	0,9876	15	0,0146
8	0,9876	16	0,0007

Note-se que foram encontradas somente saídas com valores reais, i.e. com limites superior e inferior dos intervalos iguais. Este resultado era esperado uma vez que todas as entradas apresentadas tambem foram pontuais. Em seguida o sistema foi colocado em operação para o caso do terceiro exame não ter sido realizado, i.e. entrada = (1, 1 , [0 , 1] , 0), e obteve-se como saída [0,9876 , 0,9996]. Este resultado parece coerente, uma vez que o exame não realizado não é o principal. Por outro lado, se omitirmos o primeiro exame, i.e. entrada = ([0 , 1] , 1 , 0 , 0), teria-se como saída [0,0007 , 0,9876], indicando, como esperado, uma saída indefinida. Na próxima ilustração usou-se o conjunto de treinamento exposto na tabela III, onde A = [0,8 , 1] e B = [0 , 0,2]. A rede foi treinada até atingir um somatório do erro médio quadrático de aproximadamente 10^{-7} . Esta rede foi operada posteriormente com entrada ([0 , 0,4] , 1 , 1 , 0) , obtendo-se como saída [0,9988 , 1,0000].

COMENTÁRIOS FINAIS

Neste artigo foram apresentados novos resultados teóricos que permitem treinar uma Rede Neural através de retropropagação do erro, com atributos que assumem valores de distribuições uniformes. Talvez a principal vantagem desta metodologia seja a possibilidade de se poder lidar com a eventual falta de alguns dos atributos de entrada sem a necessidade de um novo treinamento. Outro ponto que julgamos relevante é a formulação de modelo com tolerância pré determinada, para cada uma das saidas da rede. A intenção desta contribuição esta no plano teórico, embora visando fundamentalmente viabilizar uma classe relevante de aplicações, não se propõe a fazer estudo do comportamento numérico sistemático do algoritimo proposto, que fica assim para futuras investigações.

TABELA III - Apresentação de intervalos na entrada

Apres.	Entrada				Alvo
	1	2	3	4	
1	A	1	1	1	1
2	A	1	1	0	1
3	A	1	0	0	1
4	A	0	0	0	0
5	A	1	0	1	1
6	A	0	1	0	1
7	A	0	1	1	1
8	A	0	0	1	1
9	B	0	0	0	0
10	B	1	0	0	0
11	B	1	1	0	0
12	B	1	1	1	1
13	B	0	1	1	0
14	B	0	0	1	0
15	B	1	0	1	0
16	B	0	1	0	0

AGRADECIMENTOS

Os autores desejam agradecer a H.V. Carneiro e L.E. Sampaio pela cuidadosa leitura de revisão.

REFERÊNCIAS:

[1] D.E.Rumelhart, G.E.Hinton e R.J. Williams "Learning Internal Representation by Error Propagation" in Parallel Distributed

Processing, Vol 1, ed. Rumelhart e McClelland, 1986.

[2] G.Alefeld e J.Herzberger, "Introduction to Interval Computation" , Academic Press, N.Y.,1983.

[3] H.Ishibuchi, A.Miyazaki, K.Kwon e H.Tanaka, "Learning from Incomplete Training Data with Missing Values ansd Medical Application", Proc. of Int. Joint Conf. on Neural Networks, pp1871, Nagoya, Japan, Oct 1993.

[4] C.E.Pedreira e E.Parente, "An Interval Computation Approach To Backpropagation" Proc. of 1994 IEEE Workshop on Neural Networks For Signal Processing, Ermioni, Grécia, Set. de 1994.