



**Universidade de São Paulo**  
B R A S I L

# Análise estatística multivariada aplicada a processos químicos

- **Escola Politécnica**
- Departamento de Engenharia Química
- Prof. Dr. Cláudio Augusto Oller do Nascimento
- Prof. Dr. Roberto Guardani

■ 2007

## Parte 7. Redes Neurais

### INTRODUÇÃO

As redes neurais representam uma técnica de inteligência artificial que permaneceu pouco estudada por muito tempo, sendo utilizada basicamente no reconhecimento de imagens. Um breve histórico de redes neurais é apresentado por Nelson e Illingworth (1991) mostrando algumas de suas facetas interessantes e pitorescas. Outra referência geral sobre redes neurais pode ser vista no livro de Dayhoff (1990). Na área de engenharia química em específico, sua utilização iniciou-se recentemente com trabalhos em detecção e diagnóstico de falhas (Hoskins *et al.*, 1988, 1990, Venkatasubramanian *et al.*, 1990, Watanabe *et al.*, 1989), assim como em modelagem e controle de processos químicos (Bhat *et al.*, 1990, Bhat e McAvoy, 1990, Ungar *et al.*, 1990, Psychogios e Ungar, 1991, Su e McAvoy, 1992).

Uma característica dos processos químicos é que a maioria deles apresenta fortes não-linearidades, as quais dificultam a obtenção de modelos fenomenológicos, a ponto de, em alguns casos, conseguir-se apenas um modelo extremamente simplificado, válido em condições muito específicas. É exatamente nessa área que as redes neurais surgem como uma técnica interessante de ser empregada, devido à sua característica de "aprender" o que ocorre no processo. Pode-se contornar, assim, em muitos casos, as dificuldades de obtenção de modelos fenomenológicos representativos do processo. Entretanto, o sucesso da modelagem e controle de processos via rede neural depende fortemente do conhecimento das principais variáveis do processo, além de ser necessária uma base de dados de boa qualidade, ou seja, que contenha todas as informações importantes do processo e o domínio desejado.

Embora existam muitos tipos de redes neurais, duas classes de rede são geralmente empregadas: a rede de múltiplas camadas, cuja informação segue uma única direção ("multilayer feedforward network"), e a rede recursiva ("recurrent network"). Estas duas classes de rede são mostradas respectivamente nas Figuras 1 e 2.

As etapas para o desenvolvimento de um modelo de rede neural podem ser descritas como: análise do processo e construção da base de dados, configuração de uma estrutura de rede neural e, por último, treinamento ou aprendizado da rede.

A fase de aprendizado ou treinamento é basicamente um problema de otimização e freqüentemente exige a apresentação da base de dados à rede por milhares de vezes. Desta forma, depara-se com as mesmas dificuldades associadas à otimização de processos, tais como: convergência, existência de mínimos locais, esforço computacional, entre outras. Não existe, até hoje, um algoritmo comprovadamente eficiente para assegurar um "treinamento ótimo".

A metodologia para a fase de treinamento de redes neurais foi basicamente consolidada por Rumelhart e McClelland (1986), conhecida como método de retropropagação ("backpropagation"). Tal metodologia foi obtida do método de otimização do gradiente descendente e tem sido utilizada na grande maioria dos trabalhos publicados na área de redes neurais, aplicadas a processos químicos.

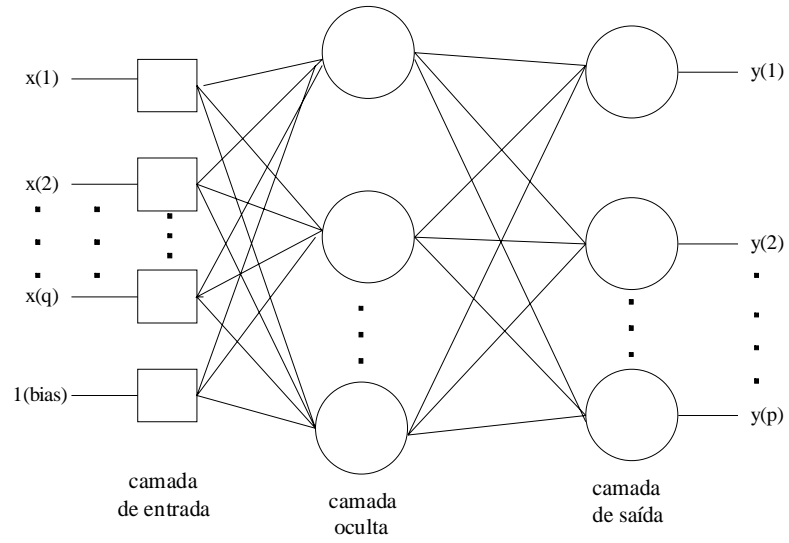


Figura 1 - Rede de múltiplas camadas com sentido único ("multilayer feedforward network").

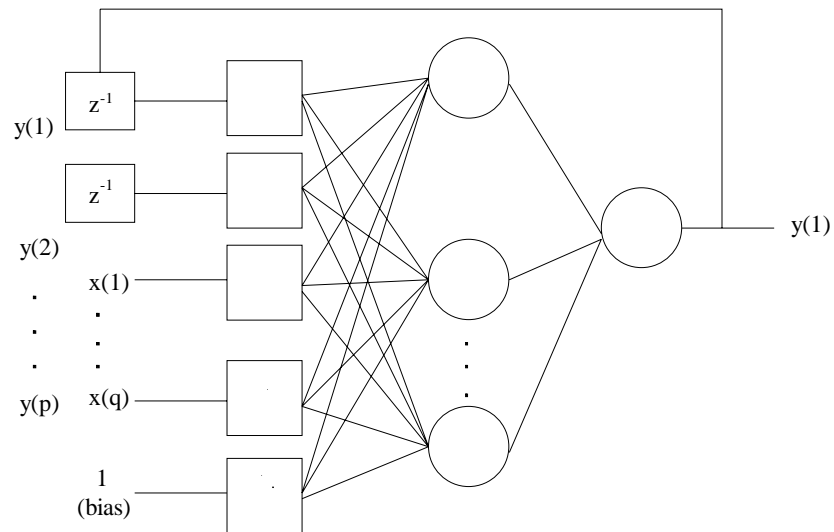


Figura 2 - Rede recursiva ("recurrent network").

A rede neural comumente empregada na modelagem e simulação de processos químicos é a "feedforward network" (abreviatura: FFNN). Nesta rede, os neurônios de uma camada são conectados a todos os neurônios da camada seguinte. Cada informação que sai de um neurônio de uma camada ( $i$ ) é ponderada por um dado peso ( $W_{i,j}$ ) e enviada a todos os neurônios da camada seguinte ( $j$ ). A convergência da rede neural está ligada à melhor escolha possível do conjunto de pesos  $\mathbf{W}$ . A rede escolhida possui três camadas: a camada de entrada de dados, a camada oculta ("hidden layer") e a camada de saída. A camada de entrada não efetua nenhum processamento. A fim de diminuir as dificuldades de convergência, todos os dados do conjunto de aprendizagem são normalizados entre zero e um. Além dos dados de entrada, alimenta-se também um valor de referência ("bias") à rede.

A Figura 3 apresenta um neurônio de uma camada  $j$  da estrutura da rede. Todas as variáveis de entrada  $X_i$  são ponderadas com um valor  $W_{i,j}$  e somadas, conforme a Equação (1):

$$S_j = \sum_{i=1}^n W_{i,j} X_i + W_{n+1,j} \quad (1)$$

A saída do neurônio,  $O_j$ , é, então, calculada a partir da expressão:

$$O_j = f(S_j) \quad (2)$$

A relação entre a entrada e a saída dos neurônios é dada por uma função ativadora sigmoidal, do tipo:

$$f(z) = \frac{1}{1 + e^{-z}} \quad (3)$$

Hunt *et al.* (1992) apresentam outras formas desta relação  $f(z)$ , utilizadas em modelagem e controle de processos químicos, bem como um apanhado de recomendações quanto à utilização de um ou outro tipo de rede neural.

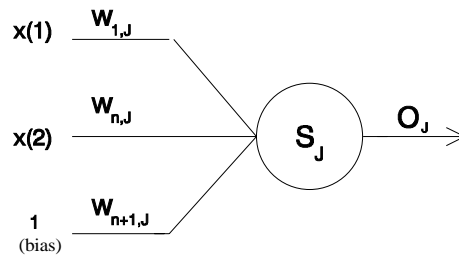


Figura 3: Representação de um neurônio  $j$  da estrutura da rede.

### ALGORITMO DE TREINAMENTO DE REDES NEURAIS

O treinamento de uma rede neural (ou aprendizado) consiste de uma adaptação dos pesos,  $W_{i,j}$ , aplicados sobre as informações entre as camadas de neurônios, de forma a minimizar o erro quadrático, expresso na forma da função  $E$ , abaixo. Esta função representa o valor acumulado para todos os dados experimentais, (de 1 a  $\gamma$ ), das somas dos erros quadráticos entre os valores experimentais e os calculados pela rede neural, para todas as saídas da rede (de 1 a  $p$ ) (Equação 4):

$$\min(E) = \min \left( \sum_{m=1}^{\gamma} \sum_{k=1}^p \left( Y_{k,calc}^{(m)} - Y_{k,exp}^{(m)} \right)^2 \right) \quad (4)$$

Outros critérios foram propostos recentemente na literatura, como os de Pollard *et al.* (1992), nos quais a função objetivo seria menos sensível a dados com erros grosseiros.

Para se obter o conjunto de pesos ( $W_{i,j}$ ), é preciso ter, em primeiro lugar, um conjunto de dados formado por subconjuntos de pares (entrada e saída) do processo, que constituem o conjunto de aprendizado ("learning set"). Este conjunto deve ser montado de tal forma que se contemplem todas as informações relevantes do processo. Tendo esse conjunto de dados, utiliza-se um algoritmo de otimização, a fim de se obter o conjunto de pesos que satisfaça um critério dado, por exemplo, pela Equação (4).

O algoritmo mais utilizado para determinar o conjunto de pesos ( $\mathbf{W}$ ) é o de retropropagação ("backpropagation"), que é uma derivação do gradiente descendente (Rumelhart e McClelland, 1986). Este é obtido a partir da minimização do resíduo quadrático ( $E$ ) com respeito a  $W_{i,j}$ . Desta forma, o gradiente  $\delta E / \delta W_{i,j}$  pode ser calculado e, conseqüentemente, a direção a ser seguida no processo de minimização do resíduo. Em cada passo da otimização,  $m$ , a atualização dos pesos pode ser dada pela equação:

$$W_{i,j}^{(m+1)} = W_{i,j}^{(m)} + \Delta W_{i,j}^{(m)} \quad (5)$$

Esta atualização pode ser feita, alternativamente, utilizando a técnica do “momento”, baseada na inclusão do termo de momento,  $\alpha$ , proporcional à variação dos pesos, tal que  $0 < \alpha < 1$ :

$$W_{i,j}^{(m+1)} = (1 - \alpha)W_{i,j}^{(m)} + \alpha \cdot \Delta W_{i,j}^{(m)} \quad (6)$$

A diferença entre o método do gradiente descendente e o "backpropagation" é que este último usa apenas um subconjunto de dados entrada/saída de cada vez, enquanto o método do gradiente tradicional emprega todos os dados do conjunto de aprendizado.

Para a camada de saída, a variação dos pesos referentes ao neurônio de saída  $k$ , incidentes sobre a saída do neurônio  $j$ , da camada oculta, a cada iteração  $m$ , é dada por:

$$\Delta W_{j,k}^{(m)} = \eta \cdot f'(S_k) (Y_k^{(m)} - O_k^{(m)}) O_j^{(m)} \quad (7)$$

Para a camada oculta, a variação dos pesos referentes ao neurônio oculto  $j$ , incidentes sobre a entrada  $X_i$ , a cada iteração  $m$ , é dada por:

$$\Delta W_{i,j}^{(m)} = \eta \cdot f'(S_j) \left\{ \left[ \sum_{k=1}^p f'(S_k) (Y_k^{(m)} - O_k^{(m)}) W_{j,k}^{(m-1)} \right] X_i^{(m)} \right\} \quad (8)$$

Nas Equações (7) e (8),  $f'$  é a derivada da função sigmoidal, dada por:

$$f'(z) = \frac{e^{-z}}{(1 + e^{-z})^2} \quad (9)$$

No algoritmo de retropropagação, em cada iteração, são calculadas inicialmente as saídas da camada oculta ( $O_j$ ) e da camada de saída ( $O_k$ ), com os pesos estimados na iteração anterior, pelas Eqs. (1) a (3), para os  $n$  neurônios da camada oculta e  $p$  neurônios da camada de saída; em seguida, as saídas da rede são utilizadas para o cálculo dos novos valores dos pesos, utilizando-se as Eqs. (9), (8) e (7) para as variações e a Eq. (5) ou (6) para os novos valores. Isto é feito para cada apresentação dos pontos experimentais. Assim, inicialmente os erros são propagados para trás, na rede, e utilizados para estimar os novos valores dos pesos nas camadas anteriores.

Uma vez determinado o conjunto de pesos ( $\mathbf{W}$ ) que minimiza o resíduo ( $E$ ), para simular o processo basta multiplicar um conjunto de dados de entrada qualquer pelos respectivos pesos, respeitando a estrutura da rede, obtendo-se o conjunto de resultados após a camada de saída da rede.

## BIBLIOGRAFIA

- BHAT, N. V. ; MCAVOY, T. Use of neural nets for dynamic modeling and control of chemical process systems , Computers & Chemical Engineering , v.14(4/5) , p. 573-583, 1990
- BHAT, N. V.; MINDERMAN JR., P. A.; MCAVOY, T.; WANG, N. S. Modeling chemical process systems via neural computation , IEEE Control Systems Magazine ; v. 4, p. 24-30, 1990.
- DAYHOFF, J.E., "Neural Network Architectures - An Introduction", Van Nostrand Reinhold, New York, 1990.
- HAYKIN, S. [18] "Neural Networks: A Comprehensive Foundation". 2nd. Ed. Prentice Hall, Upper Saddle River, 1999.
- HOSKINS, J. C.; HIMMELBLAU, D. M. Artificial neural network models of knowledge representation in chemical engineering , Computers & Chemical Engineering , v. 12(9/19), p. 881-890, 1988
- HOSKINS, J. C.; HIMMELBLAU, D. M. Fault detection and diagnosis using artificial neural networks , In Artificial Intelligence in Process Engineering , (M. L. Mavrovouniots, ed.) , N.Y., Academic Press, p 123-160 , 1990
- HUNT, J.K. ; SBARBARO, D. ; ZBIKOWSKI, R. ; Gawthrop, P.J. Neural networks for control systems - A survey, Automatica , Vol. 28 (6), pag. 1083-1112, 1992.
- JACOBS, R. A. Increased rates of convergence through learning rate adaptation , Neural Networks, v. 1, p. 295-307 , 1988
- JOSEPH, B.; WANG, F.H. E SHIEH, D.S.S. Exploratory data analysis: a comparison of statistical methods with artificial neural networks. Computers & Chemical Engineering , v. 16(4) , p. 413-423, 1992
- KOVÁCS, Z.L., "Redes Neurais Artificiais: Fundamentos e Aplicações". 3ª. Ed. Editora Livraria da Física, São Paulo, 2002.

- NELSON, M.M. E ILLINGWORTH, W.T. "A Practical Guide to Neural Nets", Addison-Wesley Publishing Company, 1991
- POLLARD, J. F. ; BROUSSARD, M. R. ; GARRISON, D. B. ; SAN , K. Y. Process identification using neural networks , Computers & Chemical Engineering , v. 16(4) , p. 253-270 , 1992
- PSICHOGIOS, D. C.; UNGAR, L. H. Direct and indirect model based control using artificial neural networks , Ind. Eng. Chem. Res. , v. 30 , p. 2564-2573 , 1991
- RUMELHART, D. ; MCCLELLAND, J. Parallel distributed processing explorations in the microstruture of cognition , v. 1, cap. 8 , MIT , Cambridge , Mass., 1986
- SU, H. T.; MCAVOY, T.; WERBOS, P. Long-term predictions of chemical processes using recurrent neural networks: A parallel training approach. , Ind. Eng. Chem. Res. , v. 31 , p. 1338-1352 , 1992
- UNGAR, L. H.; POWELL, B. A .; KAMENS, S. N. Adaptative networks for fault diagnosis and process control , Computers & Chemical Engineering , v. 14(4/5) , p. 651-572 , 1990
- VENKATASUBRAMANIAM, V.; VAIDYANATHAN, R.; YAMAMOTO, Y. Process fault detection and diagnosis using neural networks - I. Steady state process , Computers & Chemical Engineering , v. 14(7) , p. 699-712 , 1990
- WATANABE, K.; MATSURA, I.; KUBOTA, M.; HIMMELBLAU, D. M. Incipient fault diagnosis of chemical processes via artificial neural networks. , AIChE Journal , v. 35(11), p. 1803-1812 , 1989
-