

## Uma Rede Neural Construtiva com Atualização Dinâmica dos Pesos

Mêuser Valença<sup>1</sup>, Teresa Ludermit<sup>2</sup>

<sup>1</sup>Companhia Hidrelétrica do São Francisco  
Rua Gaspar Peres 427/104, CEP 50670-350, CDU

<sup>2</sup>Universidade Federal de Pernambuco

Cx. Postal 7851 CEP 50732-970

E-mails: mjsv@di.ufpe.br, tbl@di.ufpe.br

### Abstract

*This paper is concerned with the applications of the GMDH to prediction models of river flow and exchange rates. Both applications are associated with the modeling of environmental, ecological, and social systems which have a great deal of uncertainty and unknown iterations between elements and parameters. An adaptive scheme of the prediction model is constructed by making use of the modified algorithm GMDH (multiplicative-additive algorithm) to improve the forecasting ability when a small number of data sequences are available for the model building. The GMDH algorithms have, of course, both strengths and weaknesses compared with other prediction models. Our self-organizing method, GMDH, has the specific features of algorithmic, short computation time, and applicability to a limited amount of data.*

### 1. Introdução

A Cibernética é a ciência de comunicação e controle em máquinas e seres vivos. A natureza emprega o melhor sistema cibernético que pode ser concebido, por exemplo, no sistema neurológico de seres vivos, no balanço ecológico envolvendo o feedback ambiental, no controle do movimento planetário e na regulação da temperatura do corpo humano. Todos estes sistemas cibernéticos são fascinantes, em função de sua acuracidade e eficiência. Eles são coesivos, auto-reguláveis, estáveis, além de terem uma extraordinária adaptabilidade e uma inerente capacidade para usar a experiência adquirida através de feedback no seu processo de aprendizagem.

O controle principal em cibernética está relacionado com o erro de atuação obtido por feedback e homeostasis. O número de variáveis envolvidas na análise de um sistema desse tipo são de uma maneira geral, bastante elevado. Estas variáveis têm, em geral, um comportamento estocástico, e interação entre si de forma bastante complexa. Exemplos de tais sistemas na natureza são os sistemas ambientais e meteorológicos, safra agrícola, vazões em um rio, poluição, etc. Os sistemas naturais são, portanto, chamados de sistemas

cibernéticos devido à sua complexidade, advinda do relacionamento entre as diversas variáveis que compõem estes sistemas na natureza.

As mudanças que ocorrem nestes sistemas são, em geral, lentas e regulares. Entretanto, algumas mudanças imprevistas também podem ocorrer, e são exatamente estas as de difícil previsão.

Os fundamentos teóricos de um sistema cibernético baseiam-se no conceito de “caixa preta” (que caracteriza um estudo da relação entre entrada e saída de um sistema), uma abordagem neural que utiliza o conceito de limiar lógico e conexionismo, uma abordagem indutiva que utiliza o conceito de mecanismo indutivo para manter o controle composto do sistema, uma abordagem probabilística que utiliza funções multiplicativas da teoria hierárquica de decisões estatísticas e uma abordagem lógica matemática de Gödel (teorema da incompletude) que utiliza o princípio de “complemento externo” como um critério de seleção [1].

Com o objetivo de representar a abordagem indutiva destes sistemas cibernéticos, vários métodos têm sido desenvolvidos. Esses métodos de aprendizagem indutiva são também chamados GMDH (Group Method of data Handling), Self-Organization, Sorting Out e métodos Heurísticos [2].

Os algoritmos de abordagem indutiva têm dois processos fundamentais à sua disposição: uma rede conexionista limitada, para geração de funções parciais, e uma função objetivo com limiar, para estabelecer o aprendizado competitivo. Por outro lado a abordagem dedutiva é baseada sobre a análise de causa e efeito.

Numa abordagem dedutiva, no processo de diálogo homem máquina o papel predominante é jogado pelo operador humano, tendo o computador o papel de uma grande máquina de calcular. Em contraste, nos algoritmos GMDH, o papel do operador humano é passivo. Ele não necessita ter um profundo conhecimento do sistema sob análise, pois precisa fornecer somente uma mínima quantidade de informação.

A vantagem destes algoritmos esta relacionada ao fato de que em cada etapa de geração de uma função parcial tem-se um sistema linear com apenas seis parâmetros a ser determinado. Este fato torna o

processo eficiente em termos computacionais. Por outro lado estes modelos têm como limitação não serem aproximadores universais de funções. Este trabalho tem por objetivo apresentar o uso da técnica recursiva dos mínimos quadrados, que é teoricamente elegante e computacionalmente eficiente, para atualização dinâmica dos pesos de uma rede neural construtiva que utiliza os princípios cibernético de auto-organização e abordagem indutiva [3].

Na seção 2 apresenta-se a metodologia da rede neural construtiva que utiliza-se neste trabalho e descreve-se a técnica recursiva dos mínimos quadrados, e como esta pode ser utilizada para atualização dinâmica dos pesos. Na seção 3 apresenta-se as aplicações realizadas. Finalmente seção 4 apresenta-se nossas conclusões e trabalhos futuros.

## 2. Redes Neurais Construtivas

O princípio de auto-organização pode ser formulado como se segue: quando a complexidade do modelo cresce gradualmente, certos critérios, que são chamados critérios de seleção ou funções objetivo (que têm a propriedade de complemento externo) passam por um mínimo global. A existência desse mínimo global é que indica a existência de um modelo de complexidade ótima (modelo mais parcimonioso).

A noção de que deve existir um modelo de complexidade ótima, determinado pelo princípio de auto-organização, é a base da abordagem indutiva. De acordo com o teorema da incompletude de Gödel, o modelo matemático de complexidade ótima para caracterizar um dado objeto complexo pode ser encontrado pelo mínimo de uma função objetivo.

### 2.1. Técnica básica

Os seguintes passos são a base dos modelos auto-organizáveis com abordagem indutiva:

- ◆ Selecionar um conjunto de dados com N observações para o sistema a ser estudado é selecionado, sendo esses dados divididos em pelo menos dois conjuntos – conjunto de treinamento (A) e conjunto de teste (B).
- ◆ Estabelecer uma função de referência que represente uma relação entre as variáveis de entrada e de saída.
- ◆ Classificar o tipo de problema e escolher uma função objetivo para critério de seleção que representará o complemento externo.
- ◆ Gerar um conjunto de funções parciais baseando-se na função de referência.
- ◆ Com o conjunto de treinamento, estimar os pesos de todas as funções parciais utilizando uma técnica de otimização eficiente.

- ◆ Calcular uma medida da qualidade de cada uma das funções parciais geradas de acordo com a função objetivo escolhida utilizando-se o conjunto de teste (B).
- ◆ Escolher o modelo parcial que apresenta a melhor medida como sendo o modelo ótimo; entretanto, se o valor encontrado não satisfaz, escolher os melhores modelos parciais, gerar novos modelos parciais e continuar o processo.

### 2.2. Modelo multiplicativo-aditivo

Existem vários algoritmos GMDH, entretanto a diferença básica entre eles está na forma de gerar as funções parciais. O algoritmo combinatorial é o principal algoritmo de uma única camada, enquanto o algoritmo multilayer é o principal algoritmo feedforward multicamadas.

Neste trabalho fizemos uso de um modelo Multiplicativo-Aditivo [4], com o objetivo de realizar uma busca polinomial no espaço de soluções mais genérico. Este modelo pode ser visto como uma generalização do algoritmo básico, onde as variáveis têm apenas potências inteiras.

Em primeiro lugar, nós devemos escolher alguns modelos multiplicativos de complexidade ótima, tendo como base algum critério externo de seleção.

Um modelo original é representado na forma de um produto de variáveis com potências desconhecidas, tais como:

$$y = a_0 x_1^{k_1} x_2^{k_2} x_3^{k_3} \dots x_m^{k_m} \quad (1)$$

Esta equação, após uma transformação logarítmica, pode ser escrita da seguinte forma:

$$\ln y = \ln a_0 + k_1 \ln x_1 + k_2 \ln x_2 + \dots + k_m \ln x_m \quad (2)$$

Os dados são separados em treinamento, teste e verificação, de tal forma que os melhores modelos parciais, baseando-se em um dado critério de seleção, serão escolhidos usando-se algum algoritmo de abordagem indutiva e algum critério de seleção. Neste trabalho nós utilizamos o algoritmo multilayer. Detalhes deste algoritmo podem ser encontrados em Valença e Ludermir, 1997 [5].

Posteriormente, para obtermos o modelo Multiplicativo-Aditivo generalizado, combina-se os modelos multiplicativos parciais selecionados em um simples polinômio completo, tal como:

$$z = b_0 + b_1 y_1 + b_2 y_2 + \dots + b_p y_p + \dots \quad (3)$$

onde z é a resposta desejada do processo,  $y_j$ ,  $j=1,2,\dots,p$ , ... são as saídas estimadas dos modelos multiplicativos selecionados,  $b_0$  é a tendência e  $b_j$  são os pesos (coeficientes).

### 2.3. Atualização dinâmica dos pesos

Um dos principais problemas relacionados ao esquema de predição com o uso de modelo adaptativo é o da estabilidade em termos da estrutura do modelo. É claro que, se a identificação da estrutura do modelo variar a cada nova informação, podemos ter problemas de overfitting ( haja vista que modificações na estrutura em tempo real pode levar a um ajuste polinomial de grau mais elevado desnecessariamente), além do que cada novo treinamento da rede neural consome tempo e necessita de análise por parte do usuário, que nem sempre é especialista em redes neurais.

Para evitar este inconveniente, nós propomos a seguinte metodologia seqüencial, que permite a atualização dos pesos da rede mantendo a estabilização da estrutura do modelo de predição. Primeiramente, estabelecemos a estrutura ótima da rede neural com os dados disponíveis. A seguir, sempre que um novo dado é obtido, os pesos dos polinômios parciais e de saída são atualizados de acordo com o processo apresentado a seguir (entretanto, a estrutura da rede estabelecida anteriormente fica fixa).

Algoritmo de atualização dos pesos: Consideremos a seguinte equação para cada variável intermediária:

$$A_p X = Y_p \quad (4)$$

Onde  $A_p$  é uma matriz  $p \times q$ ,  $X$  é o vetor de pesos  $q \times 1$ , e  $Y_p$  é o vetor de saída  $p \times 1$ .

A solução de mínimos quadrados para a equação 4 é dada por:

$$X^* = (A_p^t A_p)^{-1} A_p^t Y_p \quad (5)$$

Onde  $(A_p)^{-1}$  e  $A_p^t$  denotam a inversa e transposta da matriz  $A_p$ , respectivamente.

Vamos considerar que temos uma nova informação. Teremos então a seguinte equação adicional,

$$BX = Z_{p+1} \quad (6)$$

Onde  $B$  é um vetor  $q \times 1$  e  $Z_{p+1}$  é a última saída. Portanto, combinando as equações 4 e 6, nós temos:

$$A_{p+1} X = Y_{p+1} \quad (7)$$

Da equação 7, a solução de mínimos quadrados transforma-se em:

$$X_{p+1}^* = (A_{p+1}^t A_{p+1})^{-1} A_{p+1}^t Y_{p+1} \quad (8)$$

Logo, se nós definirmos

$$P_p = (A_p^t A_p)^{-1} \quad (9)$$

Da equação 7 obtemos

$$(P_{p+1})^{-1} = A_{p+1}^t A_{p+1} = A_p^t A_p + BB^t = (P_p)^{-1} + BB^t \quad (10)$$

Neste ponto, vamos estabelecer a fórmula recursiva baseada na equação 8 de modo a simplificar a solução dos mínimos quadrados quando da existência de uma nova informação. Se as matrizes  $P_{p+1}$ ,  $P_p$ ,  $H_{p+1}$  e  $R_{p+1}$  satisfazem a equação

$$(P_{p+1})^{-1} = H_{p+1}^t R_{p+1}^{-1} H_{p+1} + (P_p)^{-1} \quad (11)$$

então  $P_{p+1}$  é dado por

$$(P_{p+1})^{-1} = P_p - P_p H_{p+1}^t (H_{p+1} P_p H_{p+1}^t + R_{p+1})^{-1} H_{p+1} P_p \quad (12)$$

Onde  $(P_{p+1})^{-1}$ ,  $(P_p)^{-1}$  e  $R_{p+1}^{-1}$  são não singulares e  $H_{p+1}$  é de ordem máxima. Das equações 10, 11 e 12 nós temos  $H_{p+1}^t = B$  e  $R_{p+1} = I_q$  (matriz identidade  $q \times q$ ), o que nos fornece a seguinte equação

$$P_{p+1} = P_p - P_p B (B^t P_p B + 1)^{-1} B^t P_p \quad (13)$$

Finalmente, substituindo a equação 13 na solução de mínimos quadrados da equação 8, obtemos a seguinte fórmula recursiva

$$X_{p+1}^* = P_{p+1} (A_{p+1}^t Y_{p+1} + B Z_{p+1}) \quad (14)$$

$$X_{p+1}^* = X_p^* + P_p B (B^t P_p B + 1)^{-1} (Z_{p+1} - B^t X_p^*) \quad (15)$$

Portanto, desde que os valores iniciais de  $X_p$  e  $P_p$  sejam dados pelo modelo pré definido, precisamos calcular apenas a inversa de um valor escalar,  $B^t P_p B + 1$ , em vez do cálculo da matriz inversa da equação 8 de dimensão  $(p+1) \times (p+1)$ . Assim, para cada nova medida realizada os pesos são atualizados para cada polinômio com estrutura fixada para cada camada.

Neste trabalho nós estamos interessados na predição de fenômenos tipicamente não lineares quando poucos dados estão disponíveis. Portanto, para avaliar o desempenho da metodologia proposta duas aplicações serão realizadas. A primeira aplicação consiste na predição do processo chuva-vazão, que é um fenômeno bastante complexo, haja vista, o grande número de fatores intervenientes no processo. A segunda aplicação será feita com uma série temporal de taxa de câmbio, onde admitiremos o conhecimento de poucos dados passados no treinamento da rede GMDH.

## 3. Aplicações

### 3.1. Predição chuva-vazão

Para ilustrar a aplicação do nosso modelo, vamos primeiro descrever a bacia da barragem de

Guarapiranga. Esta bacia compreende uma área de drenagem de 631 Km<sup>2</sup> e está situada no estado de São Paulo. Os dados utilizados correspondem aos valores mensais de chuva e vazão no período de Janeiro de 1979 a março de 1983 (posto barragem).

Utilizou-se o período de 1979 a 1981 (sendo 12 valores para validação da rede neural) para treinamento da Rede Neural e o período de 1982 a 1983 para teste. A entrada da rede neural foi composta por 7 elementos de processamentos na camada de entrada - 4 para representar a sazonalidade, 3 representando os últimos 3 meses de chuva.

Com estes dados de entrada nós treinamos a rede GMDH e utilizamos para realizar as previsões futuras para 1 passo à frente (para este caso o erro médio quadrático foi de 51,7). Posteriormente nós partimos desta rede treinada, ou seja, com arquitetura fixa, e a cada novo dado nós atualizamos os pesos da rede GMDH conforme metodologia proposta (o erro médio quadrático neste caso é de 13,2). A figura 1 abaixo mostra os resultados obtidos para os dois casos.

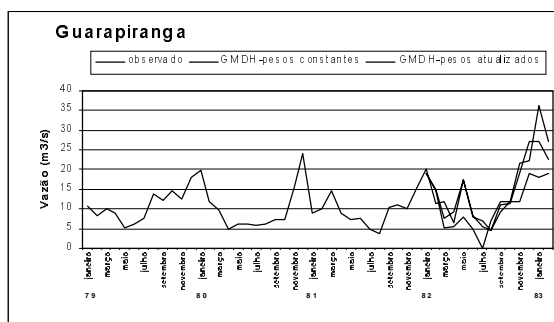


Figura 1: Resultados obtidos para Guarapiranga

### 3.2. Predição de taxa de câmbio

O valor da taxa de câmbio é um importante índice econômico no mercado monetário nacional e internacional. As previsões de taxa de câmbio são bastante úteis tanto para governo quanto para as empresas uma vez que esta reflete diretamente em decisões de investimento e nas transações comerciais.

Nesta aplicação vamos utilizar os dados de taxa de câmbio entre a libra e o marco (Financial Times), no período de 25/08/92 a 08/10/92. Como entrada da rede neural nós consideramos três valores passados da própria taxa de câmbio. O treinamento e validação da rede neural foi realizado com apenas 35 dias. Os dez 10 dias restantes do conjunto ficaram para avaliação da acuracidade das previsões a serem realizadas durante os próximos 10 dias, com e sem atualização dos pesos.

Da mesma forma que na aplicação anterior inicialmente treinamos e validação da rede neural com os 35 dias de dados admitidos como conhecidos e realizamos as previsões para os 10 dias seguintes, considerados como desconhecidos (obtemos um erro médio quadrático de 0,0021). Em seguida com esta

estrutura fixa nós atualizamos os pesos de acordo com a metodologia proposta à medida que uma nova informação estava disponível (o erro médio quadrático foi de 0,00065). A figura 2 abaixo resume os resultados desta aplicação.

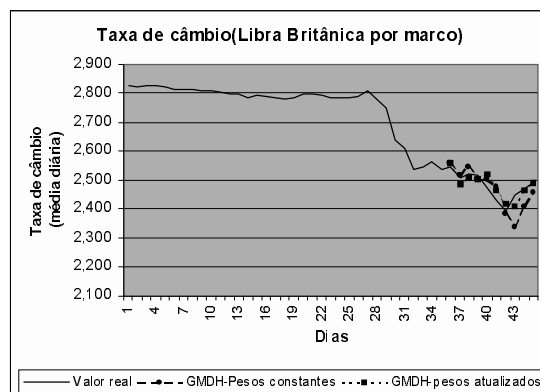


Figura 2 : Resultados obtidos para taxa de câmbio

## 4. Conclusões

Neste trabalho um modelo de predição adaptativa foi sugerido para um modelo modificado do tipo GMDH. O algoritmo proposto mostrou que é capaz de melhorar a qualidade da previsão de séries temporais, especialmente quando se dispõe de poucos dados para realização do treinamento inicial. As aplicações realizadas são simples mais extremamente ilustrativas pois têm como enfoque duas áreas de difícil previsão face a complexidade dos fatores que afetam estas variáveis.

Este algoritmo ainda tem como vantagem, além de sua aplicabilidade quando se dispõe de poucos dados, grande simplicidade (gera uma equação matemática polinomial) e treinamento rápido.

## Referências

- [1] A. G. Ivakhnenko. Polynomial theory of complex systems. IEEE Trans. Syst., Man, Cybern., vol 1, no. 4, páginas 364-378, oct.1971.
- [2] A. G. Ivakhnenko. Self-organisation of neuronet with active neurons for effects of nuclear test explosions forecasting. System Analysis Modeling Simulation (SAMS), vol.20, páginas 107-116, 1995.
- [3] M. J. S. Valença and T. B. Ludermir. Self-organizing modeling in forecasting daily river flows. Vth Brazilian Symposium on Neural Networks-Brazilian Computer Society, vol. 1, páginas 210-214, 1998.
- [4] M. J. S. Valença and T. B. Ludermir. Previsão de demanda máxima mensal utilizando um modelo auto-organizável. V Simpósio Brasileiro de Redes Neurais, vol. 2, páginas 311-314, 1998.
- [5] M. J. S. Valença and T. B. Ludermir. Uma nova rede neural polinomial com aplicação na previsão de vazões. V Simpósio Brasileiro de Redes Neurais, vol. 2, páginas 273-278, 1998.