

## PREVISÃO DE VAZÃO DA BACIA DO RIBEIRÃO JOÃO LEITE UTILIZANDO REDES NEURAS COM TREINAMENTO LEVENBERG-MARQUARDT

JÚLIO CÉSAR FERREIRA\*, MÔNICA PAIS SAKURAY\*, GILBERTO ARANTES CARRIJO†, KEIJI YAMANAKA†

\*Rodovia Geraldo S. Nascimento, km 2.5  
IFGoiano - Campus Urutaí  
Urutaí - Goiás, Brasil

†Av. Engenheiro Diniz, 1178, Bairro Santa Mônica  
Universidade Federal de Uberlândia  
Uberlândia - Minas Gerais, Brasil

Emails: jc@ifgoiano.edu.br, monicaspais@urutai.ifgoiano.edu.br, gilberto@ufu.br, keiji@ufu.br

**Abstract**— This paper aims to provide a model that represents the behavior of the hydrological basin of the river Joao Leite, in Goiás, helping the decision-making by managers of water resources. The presented inflow forecast model and rainfall-runoff model use Artificial Neural Networks (ANN) with multiple layers and the Levenberg-Marquardt learning algorithm, and use as input the time series data of daily observed rainfall and daily observed inflows measured by stations along of the Joao Leite basin, in a period of six years, from 1991 to 1997. The data were pre-processed to correct cases of inconsistency or lack of information, and the computation of the average daily precipitation was performed by the Thiessen method. The choice of the Levenberg-Marquardt algorithm made the ANN training process much faster and efficient, and allowed to test different combinations of parameters and the choice of the best model. The results obtained by the presented model show the feasibility of using ANNs in the modeling of river basins and better results than the conceptual models ones.

**Keywords**— Precipitation, Inflow Forecast, Levenberg-Marquardt.

**Resumo**— Este trabalho busca fornecer um modelo que representa o comportamento hidrológico da bacia do ribeirão João Leite, em Goiás, facilitando a tomada de decisão dos gestores dos recursos hídricos. O modelo de previsão de vazão e o modelo de transformação precipitação-vazão apresentados utilizam Redes Neurais Artificiais (RNA) com múltiplas camadas e treinamento pelo algoritmo de Levenberg-Marquardt. A entrada da RNA é constituída das séries históricas dos dados de precipitação diária e vazão diária observadas nos postos pluviométricos e fluviométricos existentes ao longo da bacia em um período de seis anos, de 1991 a 1997. Os dados foram pré-processados para correção de casos de inconsistência ou inexistência e o cálculo da precipitação média diária foi realizado pelo método de Thiessen. A opção pelo uso do algoritmo de Levenberg-Marquardt tornou o processo de treinamento da RNA muito mais rápido e eficiente, e possibilitou que diversas combinações de parâmetros de configuração para escolha do melhor modelo fossem exploradas e apresentadas. Os resultados obtidos com o modelo apresentado mostram a viabilidade do uso de RNAs na modelagem de bacias hidrográficas, com melhores resultados que os modelos conceituais existentes.

**Keywords**— Precipitação, Previsão de Vazão, Levenberg-Marquardt.

### 1 Introdução

A água é um recurso natural que apresenta os mais variados usos e é indispensável ao desenvolvimento humano. A manutenção desse recurso finito em padrões de quantidade e qualidade representa um desafio para a sociedade e exige estudos sobre conservação, manutenção, escoamento e tratamento de água. Nesse sentido, estudos que proponham desenvolver modelos para representar o comportamento hidrológico de bacias são cada vez mais valorizados.

A bacia hidrográfica do ribeirão João Leite está localizada ao norte de Goiânia, capital de Goiás. Esta bacia é responsável por aproximadamente 52% do abastecimento urbano da cidade de Goiânia, além de suprir demandas de irrigação, piscicultura, etc. Nos últimos anos, a demanda por água na bacia hidrográfica do ribeirão João Leite tem superado sua disponibilidade hídrica, gerando con-

flitos entre os usuários.

Nesse contexto, o objetivo deste estudo é desenvolver um algoritmo computacional capaz de construir uma Rede Neural Artificial que preveja a vazão diária para um dia seguinte a dados fornecidos e comparar os resultados obtidos com trabalho semelhante realizado por Figueiredo et al. (2007).

### 2 Fundamentação Teórica

Nesta seção, conceituaremos a bacia do ribeirão João Leite, modelos de previsão diária de vazão e redes neurais artificiais - arquitetura e modelo de treinamento Levenberg-Marquardt.

### 2.1 Bacia do ribeirão João Leite

Este artigo foi baseado em séries históricas de precipitação e vazão da bacia hidrográfica do ribeirão João Leite, que drena uma área de 766,80 km<sup>2</sup>, com seção de controle nas estações pluviométricas e fluviométricas listadas na Tabela 1. O ribeirão João Leite tem sua nascente nas proximidades do município de Ouro Verde de Goiás e faz parte do complexo hidrográfico da bacia do rio Paraná. Além de Goiânia, outros municípios importantes de Goiás estão diretamente ligados à bacia em estudo, como é o caso do município de Anápolis, demonstrando a necessidade de um modelo de previsão de vazão.

### 2.2 Modelos de Previsão de Vazão

Modelos de precipitação-vazão é complexo devido a grande quantidade de variáveis envolvidas - precipitação, condensação, infiltração, interceptação, escoamento superficial e subterrâneo, percolação, evapotranspiração, etc. A outra dificuldade está associada ao fato da não linearidade entre essas variáveis e da sua modificação no tempo e no espaço. Os modelos atuais, denominados de modelos conceituais, procuram, com certo grau de precisão, simular os processos físicos da bacia hidrográfica. Após a calibração de tais modelos, são alterados os parâmetros e verificam-se os impactos de tais alterações sobre a bacia hidrológica. Apesar de fornecerem bons resultados, os modelos conceituais apresentam certas dificuldades, tais como: dificuldade na interpretação do algoritmo, incerteza fora do intervalo de ajuste, grande número de parâmetros a serem calibrados, necessidade de conhecimento aprofundado de hidrologia e levantamento de dados no campo. O modelo mais utilizado, denominado PREVIVAZH, consiste em utilizar apenas informações históricas de vazões observadas. Análise de desempenho disponibilizada para este modelo, ao longo dos anos, mostra que os índices Mape (Erros Médios Absolutos Percentuais) estão entre 10% a 52%, onde

$$Mape = \frac{1}{N} \sum_{t=1}^N \frac{|v_{previsto_t} - v_{observado_t}|}{v_{observado_t}} 100$$

Como visto, o sistema de precipitação-vazão é ruidoso, não linear e complexo. Uma alternativa aos modelos conceituais são os modelos baseados em Redes Neurais Artificiais, que procuram relacionar entrada e saída sem se fazer necessário simular todo o ciclo hidrológico. Alguns trabalhos que utilizaram Redes Neurais Artificiais obtiveram resultados significativamente superiores que modelos conceituais tradicionais.

### 2.3 Algoritmo Levenberg-Marquardt (LM)

Segundo Haykin (1999), Redes Neurais Artificiais (RNA) do tipo Multilayers Perceptron (MLP) têm sido utilizadas com sucesso na resolução de problemas diversos e difíceis, com o algoritmo de treinamento de retropropagação do erro (backpropagation). Este algoritmo é uma técnica de implementação de atualização dos pesos pelo gradiente descendente, onde os pesos movem-se direcionados pela negativa do gradiente da função objetivo. Existem algumas variações do algoritmo básico que são baseadas em técnicas de otimização, como o gradiente conjugado e os métodos de Newton. Estas técnicas de otimização operam em modo batch, e são baseadas em técnicas conhecidas de otimização numérica: algoritmo do gradiente conjugado, Quasi-Newton e Levenberg-Marquardt (LM).

O algoritmo LM foi projetado para se aproximar da velocidade dos algoritmos de treinamento de segunda-ordem, sem ter que calcular a matriz Hessiana. Quando a função objetivo tem a forma do somatório de potências:

$$\epsilon(n) = \frac{1}{N} \sum_{n=1}^N (d_j(n) - y_j(n))^2$$

Onde  $d_j(n)$  é a saída desejada para o neurônio  $j$  e  $y_j(n)$  é a saída calculada para o neurônio  $j$ . Então a matriz Hessiana pode ser aproximada como:

$$H = J^T J$$

e o gradiente pode ser calculado como

$$g = J^T e$$

onde  $\mathbf{J}$  é a matriz Jacobiana que contém as primeiras derivadas dos erros da rede em relação aos pesos e bias, e  $\mathbf{e}$  é um vetor de erros da rede. A matriz Jacobiana pode ser calculada pela técnica padrão de backpropagation que é uma maneira bem menos complexa que o cálculo da matriz Hessiana, de forma que  $\mathbf{J}$  é a matriz jacobiana que contém as primeiras derivadas dos erros da rede em relação aos pesos e bias da rede neural artificial, (Hagan and Menha, 1994).

A atualização dos pesos e bias é feita por  $x_{j+1} = x_k - [J^T J + \mu * I]^{-1} J^T e$ . Quando o escalar  $\mu$  é zero, o método torna-se o método de Newton usando a aproximação da matriz Hessiana. Quando  $\mu$  é grande, ele torna-se o método do gradiente descendente com passo de tamanho pequeno. O método de Newton é mais rápido e mais preciso quando perto de um erro mínimo, e desta forma o objetivo é ir em direção ao método de Newton o mais rápido possível. Então, o valor de  $\mu$  é decrementado após cada passo em que houver redução da função objetivo e é aumentado somente quando é verificado que se o passo for executado haverá um aumento da função objetivo. Desta forma, a função objetivo

é sempre reduzida a cada iteração do algoritmo. Para o treinamento com o algoritmo de LM são necessários os parâmetros:  $\mu$ ,  $\mu_{dec}$ ,  $\mu_{inc}$  e  $\mu_{max}$ . O parâmetro  $\mu$  é o valor inicial de  $\mu$ . Este valor é multiplicado por  $\mu_{dec}$  sempre que a função objetivo é reduzida em um dado passo. É multiplicada por  $\mu_{inc}$  sempre que é verificado se a execução de um passo implicará no aumento da função objetivo. Se  $\mu$  torna-se maior que  $\mu_{max}$ , o algoritmo para. A descrição original do algoritmo de LM está em Marquardt (1963), e uma aplicação do algoritmo no treinamento de redes neurais aparece em (Duda et al., 2001) e em (Hagan and Menha, 1994). Este algoritmo é eficiente e rápido para redes neurais MLP feedforward de tamanho moderado (até algumas centenas de pesos) e possui uma implementação eficiente no software MATLAB®, onde o cálculo da equação com matriz é uma função *built-in* do software. Apesar da grande eficiência do algoritmo LM, o custo computacional é grande, pois o tamanho da matriz  $J$  é determinado pelo número de padrões de treinamento e pesos.

### 3 Materiais e Métodos

Nesta seção discutiremos os procedimentos pragmaticamente inerente a este estudo.

#### 3.1 Séries Históricas

Os dados de precipitação e vazão diária foram disponibilizados pelo Sistema Nacional de Informações Hidrológicas coordenado pela Agência Nacional de Águas (ANA), através de seu site. Na Tabela 1, estão apresentadas as principais informações das estações pluviométricas e da estação fluviométrica empregadas nesse estudo. Os dados

Tabela 1: Informações das 3 estações Pluviométricas e da fluviométrica

CÓDIGO	NOME	MUNICÍPIO	LATITUDE	LONGITUDE
1649004	Goianápolis	Goianápolis	16°51'64"	49°02'03"
1649005	Goiânia - Agronomia	Goiânia	16°66'67"	49°26'67"
1649009	Ouro Verde de Goiás	Ouro Verde de Goiás	16°22'81"	49°14'36"
60642000	Captação João Leite	Goiânia	15°30'00"	49°14'34"

disponíveis para a construção do modelo de previsão de vazão dividem-se em dois tipos:

- Precipitação diária observada em três estações pluviométricas, em  $mm$ , de 1 janeiro de 1991 a 31 de julho de 1997.
- Vazão diária observada na estação de captação João Leite, em  $m^3s^{-1}$ , no mesmo período anterior.

Os dados disponíveis na Agência Nacional de Águas possuem inconsistência, falhas de dias e até

anos. Além disso, a estação pluviométrica Goiânia-Agronomia foi desativada em 1997. Neste contexto, a escolha dessa bacia foi definida pela nossa proximidade dela e pela existência de estudos recentes feito por Santos (2009).

#### 3.2 Pré-processamento de Dados

Para o cálculo da precipitação média diária utilizamos o método de polígonos de Thiessen. Segundo Pinto (1976), o método consiste em calcular a área influenciada por cada estação (polígono) e utilizá-la como pesos para o cálculo da média ponderada a cada dia, através da equação abaixo:

$$precipitacao\_media = \sum_{i=1}^n \frac{area_i * precipitacao_i}{soma\_areas}$$

Para encontrar esses polígonos, traçamos um segmento de reta entre duas estações pluviométricas específicas e traçamos sua mediatriz. Fizemos o mesmo para as outras estações, duas a duas. De modo que a área de cada polígono é o peso no cálculo da média ponderada.

Para os casos de inconsistência ou ausência de dados em uma estação ou mais, utilizamos o método de Thiessen para calcular a precipitação média diária, sem ser necessário completar este dado. Neste caso, foi feito um recálculo de novos pesos para as estações que possuíam dados e deixando o peso da estação ausente de fora através da seguinte equação

$$novo\_peso = \frac{peso\_antigo}{soma\_pesos\_postos - faltantes}$$

Como utilizamos apenas uma estação fluviométrica, não foi preciso cálculo de média para vazões.

Após uma análise minuciosa dos dados, eles foram divididos em dados de entrada e de saída. Os dados de entrada são as precipitações médias e as vazões observadas e os dados de saída são as vazões observadas. Os dados foram separados em dois conjuntos - um deles para treinamento e outro estritamente para teste da rede:

- treinamento - uma matriz de entrada com 2 linhas e 2192 colunas, onde a primeira linha são as precipitações e a segunda são as vazões e uma matriz de saída com 1 linha e 2192 colunas, onde as colunas são vazões observadas.
- teste - uma matriz de entrada com 2 linhas e 212 colunas, onde a primeira linha são as precipitações e a segunda são as vazões e uma matriz de saída com 1 linha e 212 colunas, onde as colunas são vazões observadas.

Devido ao fato de a maioria das funções de ativação utilizadas em redes neurais artificiais trabalhar com dados entre  $-1$  e  $1$  ou entre  $0$  e  $1$  e ao fato de precipitação e vazão estarem em escala diferentes, foi necessário normalizar os dados. Sajikumar and Thandaveswara (1999) utilizou seguinte equação de normalização:

$$x_{norm} = F_{min} + \frac{x - x_{min}}{x_{max} - x_{min}}(F_{max} - F_{min})$$

Onde  $x_{norm}$  e  $x$  é, respectivamente, o valor normalizado e o valor a ser normalizado.  $F_{min}$  e  $F_{max}$  são valores que devem ser bem definidos e  $x_{max}$  e  $x_{min}$  são o menor e maior valor dos dados, respectivamente. Por último, os dados foram agrupados em pares  $(X, y)$ , onde  $X$  é constituído de sete valores de precipitações, sete valores de vazões e um valor da última vazão e  $y$  é a vazão do oitavo dia em relação aos sete anteriores. Deste modo, passamos a ter 2185 pares no formato a seguir:  $(p_1 p_2 p_3 p_4 p_5 p_6 p_7 v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8)$ . Estes dados foram apresentados na entrada da rede no seguinte percentual: 60% para treinamento, 20% para validação e 20% para teste, ou seja, 1311, 437 e 437, respectivamente.

#### 4 Resultados

Realizamos 24 baterias de testes com 200 repetições em cada uma. Fixamos alguns parâmetros da configuração da RNA, tais como: 1 camada intermediária, função de ativação logarítmica sigmóide na camada intermediária e linear na camada de saída, percentagens 60 – 20 – 20 de apresentação dos dados e variamos o número de neurônios na camada oculta, normalização e número de entradas no treinamento. Baseado nessa configuração, armazenamos o Mape e o coeficiente de correlação  $R$  entre a saída observada e a prevista para duas situações: o momento de TREINO e o momento de TESTE. Os resultados estão exibidos na Tabela 2 e mostram que o melhor desempenho, considerando o menor Mape e a maior correlação linear  $R$ , foi alcançado pela configuração 9. A partir desse momento, definimos a linha 9 da Tabela 2 como a configuração utilizada em nosso modelo. Neste contexto, utilizando as 7 últimas precipitações, as 7 últimas vazões e a última vazão, todas observadas, foi prevista a vazão do dia subsequente para alguns exemplos. Na Tabela 3 é exibida a previsão para o dia 18 de abril de 1997 e para 8 de janeiro de 1997. É importante salientar que estes dias não fazem parte dos dados de treinamento. Em Figueiredo et al. (2007) foi feita previsões diárias para 1 dia entre as bacias UHE Foz do Areia, Reservatório Jordão e UHE Salto Osório usando Redes Neurais Backpropagation (Levenberg-Marquardt) com 11 neurônios na camada oculta e o Mape resultante foi igual a 10.46. Nosso melhor resultado para a bacia

Tabela 2: Variação de parâmetros de treinamento

#	PARÂMETROS				TREINO		TESTE			$\eta$
	Nr. de entradas treinamento	Neurônios na camada oculta	Normalização	Mape(%)	R	Mape(%)	R	Número de entradas no teste		
1	2185	17	[05 - 95]	6,26	0,97	11,61	0,82	212	rand	
2	2185	22	[05 - 95]	6,85	0,97	12,10	0,53	212	rand	
3	2185	30	[05 - 95]	6,65	0,97	24,80	0,34	212	rand	
4	2185	100	[05 - 95]	6,67	0,98	30,67	0,58	212	rand	
5	2185	17	[1 - 9]	5,30	0,96	12,18	0,77	212	rand	
6	2185	30	[1 - 9]	5,10	0,98	11,93	0,74	212	rand	
7	2185	100	[1 - 9]	4,82	0,97	25,36	0,31	212	rand	
8	2185	17	[3 - 7]	1,90	0,96	10,36	0,84	212	rand	
9	<b>2185</b>	<b>30</b>	<b>[3 - 7]</b>	<b>1,65</b>	<b>0,97</b>	<b>10,65</b>	<b>0,84</b>	<b>212</b>	<b>rand</b>	
10	2185	47	[3 - 7]	1,88	0,96	11,21	0,81	212	rand	
11	2185	30	[3 - 7]	1,88	0,96	11,28	0,71	212	.01	
12	2185	30	[3 - 7]	1,96	0,96	11,71	0,74	212	.001	
13	2185	30	[3 - 7]	1,88	0,96	13,05	0,66	212	.1	
14	1000	30	[3 - 7]	1,92	0,93	16,11	0,90	1397	rand	
15	2185	47	[2 - 5]	2,08	0,96	10,91	0,80	212	rand	
16	1700	30	[3 - 7]	1,94	0,95	15,23	0,90	704	rand	
17	1500	30	[3 - 7]	1,97	0,96	14,76	0,82	904	rand	
18	1000	30	[3 - 7]	1,94	0,95	16,88	0,90	1404	rand	
19	700	30	[3 - 7]	2,11	0,94	14,32	0,94	1704	rand	
20	300	30	[3 - 7]	2,13	0,96	13,90	0,92	1904	rand	
21	200	30	[3 - 7]	0,61	0,99	20,86	0,81	2304	rand	
22	100	30	[3 - 7]	0,44	0,99	97,94	0,05	2304	rand	
23	50	30	[3 - 7]	0,00	1	24,30	0,88	2354	rand	
24	10	30	[3 - 7]	0,00	-	626,79	0,64	2394	rand	

João Leite utilizando RNAs, com 30 neurônios foi Mape igual a 10.05.

Após obtenção e armazenamento dos melhores pesos para as opções de treinamento anteriormente definidas, fizemos diversas previsões de 1 dia para verificar a capacidade da rede em responder corretamente. Na Tabela 3 estão demonstradas duas das previsões geradas. Além do objetivo principal

Tabela 3: Duas previsões diárias nos meses de abril e janeiro de 1997.

Ano: 1997							
11/abr	12/abr	13/abr	14/abr	15/abr	16/abr	17/abr	18/abr
0,00	0,00	0,00	0,00	6,02	0,42	3,28	Vazão Observada: 9,49
7,97	7,81	7,67	7,81	7,81	7,81	9,32	Vazão Prevista: 9,04 Erro: 4,74%
1/jan	2/jan	3/jan	4/jan	5/jan	6/jan	7/jan	8/jan
0,00	15,70	0,00	0,00	10,02	0,68	3,84	Vazão Observada: 13,76
6,94	7,65	15,72	15,55	10,64	8,51	10,08	Vazão Prevista: 10,20 Erro: 25,87%

de fazer previsão diária para a bacia do ribeirão João Leite, criamos um modelo que pode ser utilizado por gestores de recursos hídricos na tomada de decisões. O modelo representa bem a transformação de precipitação em vazão sem o custo operacional exigido pelos modelos conceituais. Deste modo, armazenamos os melhores pesos para serem aplicados no conjunto total de dados e verificar sua capacidade de modelagem do sistema hidrológico. Na Figura 1, percebemos um ótimo ajuste entre a vazão observada e a vazão estimada pela rede neural. Observe que houve um bom ajuste do modelo previsto em relação ao modelo observado, inclusive a partir do 2192º dia, cujos dados não foram utilizados no treinamento da rede. Para esse modelo, obtivemos o Mape igual a 8,66 e correlação entre vazão observada e vazão prevista igual a 0.96.

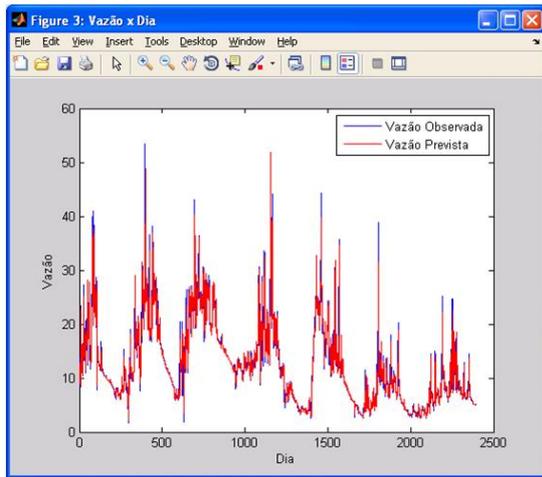


Figura 1: Modelo hidrológico de Vazão para o intervalo de 01/01/1991 a 31/07/1997

Além disso, verificamos a dificuldade de a rede responder a vazões extremas e a mudanças radicais no comportamento de vazão ou de precipitação. Na Figura 2 podemos verificar a regressão linear para o modelo computacional gerado pela RNA. Fundamentado na Figura 2, observamos que

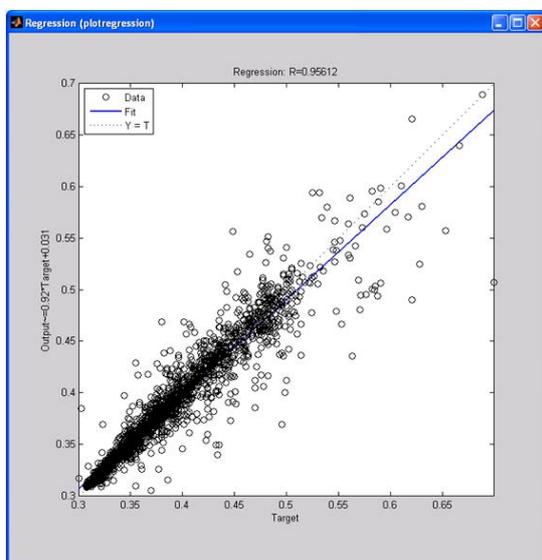


Figura 2: Regressão linear entre vazão prevista e vazão observada no intervalo de 01/01/1991 a 31/07/1997

o gráfico gerado pelo modelo não é estacionário, ou seja, possui leve tendência de diminuição de vazão com coeficiente de correlação de 0.96. Além disso, percebemos uma sazonalidade de 1 ano - o que representa os aumentos de vazão durante os meses de outubro a março, com picos em dezembro e janeiro e diminuição nos meses de julho e agosto. O mesmo conclui Santos (2009) em sua tese: uma redução de  $0.07m^3s^{-1}$  ao ano, com significância de 95%.

## 5 Conclusões

Com base na Figura 2 observa-se que o comportamento dos dados observados em relação aos dados gerados pela técnica de Redes Neurais com treinamento Levenberg-Marquardt apresentaram um melhor ajuste para vazões na faixa de 5 a  $10m^3s^{-1}$ , sendo que para vazões superiores a  $10m^3s^{-1}$  ocorre discrepância quanto ao ajuste. Para mudanças bruscas na vazão observou-se que a rede não permite um ajuste adequado dos dados observados em relação aos dados previstos para um primeiro momento. No entanto, após tais mudanças bruscas a função aproxima-se satisfatoriamente dos valores observados, podendo apresentar boa correlação linear entre vazão prevista e vazão observada para ajuste dos dados. Além disso, observamos uma boa representação do modelo hidrológico utilizando Redes Neurais Artificiais, o que é confirmado pelos coeficientes de correlação no intervalo de 0,75 a 0,90 durante a etapa de teste. Vale ressaltar que o modelo desenvolvido no presente trabalho obteve valores de correlação sensivelmente melhor que Santos (2009) utilizando modelos conceituais e índices Mape semelhantes a Figueiredo et al. (2007), embora não seja pertinente a comparação de Mape's para estações diferentes.

Isto mostra que RNAs, bem treinada e parametrizada, conseguem modelar bacias hidrográficas e executar previsões de vazões melhor que modelos conceituais. Como trabalhos futuros, sugerimos:

- melhorar o ajuste para dados discrepantes, através da abordagem do problema no domínio da frequência utilizando Transformada de Wavelet;
- aumentar o horizonte de previsão de vazão de 1 dia para valores maiores;
- utilizar mesma metodologia em uma bacia maior.

## Agradecimentos

O presente estudo contou com a colaboração do engenheiro agrícola DSc Marconi Batista Teixeira na interpretação dos resultados obtidos.

## Referências

- Duda, R. O., Hart, P. E. and Stork, D. G. (2001). *Pattern Classification*, Wiley, 2nd edition, New York.

- Figueiredo, K., Vellasco, M. M. B. R., Pacheco, M. A. C., da Cruz, A. V. A., Contreras, R. J. and Barbosa, C. (2007). *Neural Networks for Inflow Forecasting Using Precipitation Information*, 20th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, v.4570, Berlin.
- Hagan, M. T. and Menha, M. (1994). *Training feedforward networks with the Marquardt Algorithm*, IEEE Transactions on Neural Networks, v. 5, n. 6, p. 989-993.
- Haykin, S. (1999). *Neural Network - A Comprehensive Foundation*, Prentice-Hall, Inc. Pearson Education, New Jersey.
- Marquardt, D. (1963). *An algorithm for least squares estimation of non-linear parameters*, J. Soc. Ind. Appl. Math., pp. 431-441.
- Pinto, N. L. S. (1976). *Hidrologia Básica*, Editora Edgard Blucher, ISBN 85-212-0154-0.
- Sajikumar, N. and Thandaveswara, B. S. (1999). *A non-linear rainfall-runoff model using artificial neural network*, Elsevier, v.216, n. 1-2, p. 32-55, London.
- Santos, E. H. M. (2009). *Comportamento hídrico na bacia hidrográfica do ribeirão João Leite - Goiás*. Tese de doutorado em Agronomia: solo e água - 98 páginas - Universidade Federal de Goiás.