

# A TÉCNICA DE REDE NEURAL ARTIFICIAL APLICADA NA PREVISÃO DE VAZÕES

Rafael de Souza Favoreto<sup>1</sup>, Márian da Costa Rohn<sup>2</sup> e Míriam Rita Moro Mine<sup>3</sup>

**Resumo** – A Técnica de Rede Neural Artificial (RNA), neste início de século, ainda não tem sólido e pleno desenvolvimento de suas aplicações no campo da Hidrologia. Este artigo procura ser um passo adiante no conhecimento da técnica e suas aplicações na previsão de variáveis hidrológicas. A revisão bibliográfica realizada, e aqui demonstrada, destina-se a definir alguns limites e servir como balizamento para futuros estudos na área de previsão de vazões de pequeno, médio e longo prazos em diversos rios do estado do Paraná.

**Abstract** – The Artificial Neural Network Technique (ANN), in beginning of this century, still does not have solid and full development of its applications in the field of the Hydrology. This article intends to be a new step in the knowledge of the technique and its applications in hydrological forecast. The literature review done, and demonstrate in this paper, destines to define some limits and to serve as guide for future studies in the area of forecast of small, medium and long run outflows in diverse rivers of the Paraná state.

**Palavras-Chave:** Redes Neurais Artificiais, Previsão Hidrológica

## INTRODUÇÃO

A técnica de Rede Neural Artificial (RNA) tenta simular o processo de funcionamento do cérebro humano. Seu funcionamento é base primordial para a busca e modelagem de Inteligência

---

<sup>1</sup> Pesquisador do CEHPAR/LACTEC, Centro Politécnico - Caixa Postal 1309, CEP 80011-970, Curitiba, PR. Telefone: (41) 361-6312, Fax: (41) 266-3582, e-mail: favoreto@lactec.org.br.

<sup>2</sup> Mestranda do PPGEH da UFPR - Caixa Postal 1309, CEP 80011-970, Curitiba, PR. Telefone: (41) 361-6313, Fax: (41) 366-3464, e-mail: marianrohn@hotmail.com.

<sup>3</sup> Professora doutora da Universidade Federal do Paraná - Caixa Postal 1309, CEP 80011-970, Curitiba, PR. Telefone: (41) 361-6310, Fax: (41) 366-3464, e-mail: mrmine@cce.ufpr.br.

Artificial. Este processo é muito útil para tarefas subjetivas que o homem executa com extrema facilidade, mas que para os tradicionais processos algorítmicos sequenciais é impossível.

Apesar de em algumas áreas a técnica de Rede Neural Artificial ser a única possível, ela pode competir e muitas vezes ser compatível em outras áreas, onde a técnica sequencial produz resultados satisfatórios.

Uma das áreas promissoras para grande aplicabilidade das RNA é a previsão de dados hidrológicos, vazão e precipitação, nos seus mais diferentes usos e o seu emprego, neste contexto, é ainda recente. O presente artigo procura levantar algumas dessas aplicações, publicadas em artigos internacionais e iniciar o entendimento dessa técnica, sanando algumas dúvidas quanto a sua aplicação e gerando novos questionamentos. É, portanto, um artigo resultante de pesquisa bibliográfica feita com o intuito de aprimorar-se o conhecimento na previsão hidrológica. Foram levantadas diversas aplicações em hidrologia, procurando delimitar algumas informações e possam balizar os estudos futuros com o uso da técnica de Rede Neural Artificial, sobretudo com interesse voltado à previsão de chuva e vazão de curto e médio termos, para fins operacionais de reservatórios de usinas hidrelétricas e controle de cheias.

As informações genéricas sobre os modelos de redes neurais afirmam que sua modelagem se adapta a qualquer problema onde seja necessário o reconhecimento de padrões, sem haver a necessidade de conhecimento da origem dos fenômenos, seus limites e a própria relação entre as variáveis. Nessa análise, será visto o quanto isso é verdade e quanto é mito, se realmente é possível a negligência da relação entre as variáveis que levam a um fenômeno hidrológico e se o resultado obtido é satisfatório.

## **REDES NEURAIS ARTIFICIAIS**

Um computador pode operar de duas maneiras distintas. A primeira concepção é chamada simbolista, e vê a solução de problemas como um processo essencialmente algorítmico, ou seja, seguindo uma sequência de passos lógicos ou aritméticos.

A segunda abordagem, chamada conexionista, afirma que certas tarefas que a mente humana faz com facilidade e rapidez são virtualmente impossíveis de serem transformadas em algoritmos. São tarefas de associação, categorização e percepção de traços marcantes naturalmente realizadas pelo homem. É a capacidade que o cérebro humano tem de se auto-organizar.

O desafio desta segunda concepção é, portanto, desenvolver e implementar, em um computador, modelos que reproduzam as capacidades naturais do cérebro humano. Para que isto seja possível, a informação inteligente deve ser processada da mesma maneira como é realizada no cérebro humano, ou seja, através de um conjunto de elementos computacionais simples, chamados

neurônios, tais como às células do sistema nervoso. Os neurônios artificiais atuam em várias vias paralelas. Estes modelos conexionistas ou cognitivos são chamados de redes neurais artificiais (RNA, ou ANN, do inglês *artificial neural networks*).

É conveniente destacar, contudo, que o desenvolvimento de modelos conexionistas não significa uma rejeição total à abordagem simbolista, mas sim uma tentativa de resolver, com recursos adicionais que a inteligência artificial proporciona, problemas não redutíveis a algoritmos.

A primeira tentativa de reproduzir o funcionamento de neurônios em um modelo cognitivo foi realizada em 1943. Warren McCullock e Walter Pitts criaram os “neurônios lógicos”, que podiam ser conectados, formando diversos tipos de RNA. Estes executavam tarefas lógicas elementares e reconheciam padrões visuais e auditivos simples.

Em 1958, Frank Rosenblatt cria o “Perceptron”, demonstrando que se fossem acrescentadas de sinapses ajustáveis, as RNA’s de McCullock e Pitts poderiam ser “treinadas” para classificar certos tipos de padrões.

Ao longo dos anos 60, os simbolistas passam a frente, e em 1969, Marvin Minsky e Seymour Papert publicam “Perceptrons”, encerrando aparentemente a disputa entre simbolistas e conexionistas. Tal publicação contribuiu significativamente para que a abordagem conexionista fosse esquecida na década seguinte. Nos anos 80, contudo, as RNA’s ressurgiram, devido, sobretudo, ao avanço da microeletrônica e ao fracasso dos modelos simbolistas na realização de tarefas banais, como distinguir um gato de um cachorro. Estes fatos foram decisivos na retomada dos modelos conexionistas.

Uma rede neural artificial é composta por várias unidades de processamento. Essas unidades geralmente são conectadas por canais de comunicação que estão associados a um determinado peso. O comportamento inteligente de uma rede neural artificial vem das interações entre as unidades de processamento da rede.

A operação de uma unidade de processamento foi proposta por McCullock e Pitts em 1943. Esta operação pode ser resumida em várias etapas, conforme esquematizado na figura 1. Em uma primeira etapa, sinais são apresentados à entrada, e cada um destes sinais é multiplicado por um fator (peso) que indica a influência deste sinal na saída da unidade. É efetuada, então, a soma ponderada dos sinais, que produz um nível de atividade. Se este nível for superior a um determinado limite, a unidade irá produzir uma determinada resposta de saída.

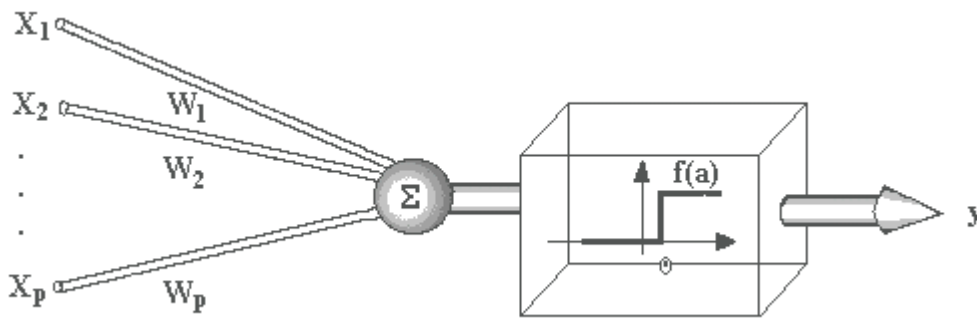


Figura 1 - Esquema de unidade McCullock – Pitts (fonte: Mendes (1998))

A partir do esquema acima, supõe-se que existam  $p$  sinais de entrada  $X_1, X_2, \dots, X_p$  e pesos  $w_1, w_2, \dots, w_p$  e limitador  $t$ ; com sinais assumindo valores *booleanos* (0 ou 1) e os pesos, valores reais.

Neste modelo, o nível de atividade  $a$  é dado por pela equação 1:

$$a = w_1X_1 + w_2X_2 + \dots + w_pX_p \quad (1)$$

A saída  $y$  é dada por pelo conjunto de equações (2):

$$\begin{aligned} y &= 1, \text{ se } a \geq t \text{ ou} \\ y &= 0, \text{ se } a < t. \end{aligned} \quad (2)$$

Em modelos simples de redes neurais, a função de ativação  $f(a)$  pode ser a própria função de soma das entradas ponderadas do neurônio. Em modelos mais complexos, a função de ativação possui um processamento atribuído. Esse processamento pode usar, por exemplo, o valor prévio de saída como uma entrada para o próprio neurônio.

A função de transferência pode ter muitas formas e métodos, podendo ser simples ou complexa. A função de transferência também é conhecida como limiar lógico (*threshold*). Essa função é quem define e quem envia para fora do neurônio o valor passado pela função de ativação.

Nenhum tipo de rede pode garantir resultados absolutamente corretos, especialmente se os padrões são de alguma forma incompletos ou conflitantes. Os resultados devem ser avaliados em termos da percentagem de respostas corretas que resultaram do modelo. A esse respeito, a tecnologia é similar ao funcionamento do neurônio biológico depois deste ter sido "projetado", e difere significativamente de todos outros programas convencionais de computador.

As redes neurais podem não ser adequadas para algumas aplicações. Alguns problemas são bem apropriados para a capacidade de reconhecimento de padrões de uma rede neural, ao passo que outros são melhor resolvidos a partir de métodos tradicionais. As redes neurais são largamente

indicadas, por exemplo, em casos nos quais há pouco conhecimento das relações entre as variáveis de entrada e saída.

A maioria dos modelos de redes neurais possui alguma regra de treinamento, onde os pesos de suas conexões são ajustados de acordo com os padrões apresentados. Em outras palavras, elas aprendem através de exemplos.

As redes neurais são tipicamente organizadas em camadas, com unidades que podem estar conectadas às unidades da camada posterior. As camadas são classificadas em três grupos: camada de entrada (na qual os padrões são apresentados à rede), camadas intermediárias, também denominadas ocultas ou escondidas (nas quais é feita a maior parte do processamento, através das conexões ponderadas) e camada de saída (na qual o resultado final é apresentado).

Uma rede neural é especificada, principalmente pela sua topologia, pelas características dos nós (neurônio) e pelas regras de treinamento (ver figura 2).

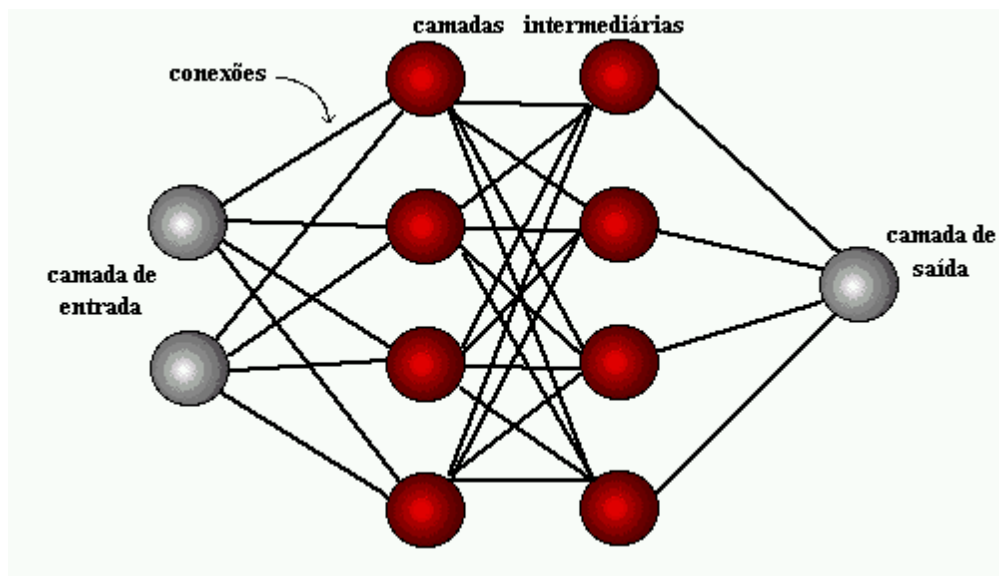


Figura 2 - Camadas de uma rede (fonte: (Mendes (1998)))

## CONEXÕES E NEURÔNIOS

Uma rede neural é formada por diversos neurônios. O processamento em cada neurônio é relativamente simples, mas a complexidade reside na interação entre diversos neurônios. Conforme já citado anteriormente, os neurônios são organizados em camadas. Cada neurônio de uma camada é conectado com todos os neurônios das camadas anteriores e posteriores e a cada uma dessas ligações é arbitrado um peso, além de haver um valor pré-arbitrado (*neuron threshold value*) para

cada neurônio. Ao somatório dos pesos multiplicados pelo valores de entrada e do valor pré-arbitrado é associada uma equação de transferência, usualmente não linear.

Thirumalaiah (1998), em sua previsão contínua de níveis de água em estações fluviométricas, utilizou uma RNA que segue a configuração tradicional, com os nós das camadas escondidas e da camada de saída sendo alimentada por todos os nós das camadas anteriores. Cada uma dessas entradas é multiplicada por um peso diferente, acrescentando uma propensão (bias) e multiplicados por uma função não linear (função sigmóide de transferência). A representação matemática é dada pela equação (3).

$$y_i = f_j \left( b_j + \sum_i w_{ij} x_i \right) \quad (3)$$

onde:

$y_i$  = saída transformada do nó  $j$ ;

$f_j$  = função sigmóide de transferência;

$b_j$  = propensão do nó  $j$ ;

$w_{ij}$  = peso da alimentação do nó  $i$  para  $j$ ;

$x_i$  = alimentação do nó  $i$ .

Na rede utilizada por Zealand et al. (1999), para prever vazões a curto prazo, cada nó é conectado com todos os nós das camadas anteriores e posteriores e a cada uma dessas ligações é arbitrado um peso ou força de conexão, além de haver um valor pré-arbitrado (*bias*) para cada nó. Ao somatório dos pesos multiplicados pelo valores de entrada e do valor pré-arbitrado é associada uma função não-linear. Em geral, essa função é uma função sigmóide que restringe a saída entre 0 e 1. Por exemplo:

$$OUT_j = 1 / (1 + \exp(-NET_j)) \quad (4)$$

onde  $NET_j$  é o somatório dos valores de entrada e seus pesos e ainda do valor pré-arbitrado.

Sajikumar e Thandaveswara (1999) utilizaram em seu modelo chuva-vazão uma rede neural do tipo temporal, ou seja, dependente do tempo. Isso implica em que a conexão entre cada neurônio não é representada apenas por um peso, e sim por um arranjo de vários pesos, chamados de Resposta de Impulso Finito (FIR). A “FIR” tem um valor para cada tempo distinto, pois a rede neural é recorrente, ou seja, ela é realimentada por seus resultados. A cada neurônio está associada uma equação sigmóide (ou logística) que limita a saída de cada neurônio entre [0,1].

Os dados, tanto de entrada como de saída, devem ter seus valores transformados em uma nova escala no intervalo de 0 a 1. Não se deve aproximar os valores máximo e mínimo dos extremos por dois motivos: i) deve-se ter uma folga para valores previstos maiores e menores que os extremos do aprendizado; ii) a função sigmóide tem uma convergência lenta para valores próximos a 0 e 1, criando-se uma região de ineficiência. Assim sendo, mesmo para valores mínimos iguais e nunca menores que zero, deve-se usar um valor um pouco acima disso, por exemplo 0,05.

Segundo Luk et al. (2000), que usaram a técnica das RNA para prever precipitações, existe um grande número de RNA alternativas que podem ser utilizadas na previsão de chuvas. Estudos anteriores citados pelos autores mostraram que a *Multi-layer Feedforward Neural Network* (MLFN), que é a configuração básica de uma RNA, se corretamente desenvolvida, pode alcançar desempenhos comparáveis a redes mais elaboradas na previsão de chuvas. Neste estudo, portanto, esta rede foi empregada devido à sua simplicidade e eficiência computacional. Foi utilizada uma função de ativação sigmóide nas camadas escondidas, e uma função de ativação linear na camada de saída. O uso de uma função sigmóide forneceu uma característica não-linear à rede. Esta função, contudo, não é adequada para a camada de saída porque força uma saída limitada entre 0 e 1. Isto quer dizer que o valor máximo de chuva corresponde a 1 e é conhecido. Tal fato é indesejável na previsão de chuvas, já que o seu valor máximo não é conhecido realmente. A fim de contornar esta situação, os autores utilizaram uma função linear.

Carriere et al. (1996) utilizaram uma rede neural tempo-dependente ou rede neural espaço-temporal para prever hidrogramas. Os neurônios (ou elementos de processamento – PE) funcionam com vários sinais de entrada e um de saída. Matematicamente as entradas (I) e os pesos dessas entradas (W) são considerados vetores. O sinal de entrada é o produto de entrada destes dois vetores, com resultado escalar. Os resultados podem ser positivo (excitatórios) ou negativo (inibitório). Os resultados positivos ativam o elemento de processamento e os negativos evitam a ativação. A retropropagação recorrente foi utilizada porque a retropropagação comum não poderia ser utilizada por ser um processo altamente dependente do tempo. A retropropagação comum é um processo estático e não produz bons resultados em processos cuja dependência do tempo é fator primordial. Em processos tempo-dependentes todo ponto do processo é fator de pontos anteriores a ele, tanto quanto outros fatores no processo. Num algoritmo comum de retropropagação todo formato (*pattern*) do vetor de entrada é visitado, independentemente de outros formatos. Muitas vezes uma visitação aleatória dos formatos é sugerida apesar de em um processo tempo-dependente a ordem de entrada dos formatos é de grande importância. Numa rede neural espaço-temporal o processo de retroalimentação é fundamental para capturar a essência do processo. A determinação da(s) camada(s) de retroalimentação é uma questão dependente do problema. Neste caso, os autores definiram como camada de retroalimentação apenas a camada de entrada.

## NÚMERO DE NEURÔNIOS DE ENTRADA

A escolha do número de neurônios na camada de entrada é um fator que exige experiência e sensibilidade da parte de quem está definindo a topologia da rede. É necessário que se analise com bom-senso as variáveis que possivelmente irão influenciar, ou que são relevantes na determinação dos dados de saída que se deseja obter. É interessante, por isso, realizar vários testes com diferentes configurações de redes já que é impossível saber, somente com uma tentativa, se a rede está extraindo os melhores resultados possíveis.

Sajikumar e Thandaveswara (1999), em seu modelo chuva-vazão, utilizaram, na camada de entrada, apenas um valor de entrada de chuva. Este estudo, contudo, não testou sequer a possibilidade de utilizar mais dados de entrada.

Shamseldin (1997), que também aplicou as RNA's em uma modelagem chuva-vazão, definiu como dado de entrada um índice que tenta interpretar os valores de precipitação, gerando, desta maneira, um erro logo na entrada. Não há restrição nenhuma para que os dados de entrada sejam diretamente as várias precipitações, mas é interessante aproveitar ao máximo a ferramenta que está sendo utilizada, procurando dados de entrada alternativos, e não somente àqueles já empregados em modelos convencionais.

Luk et al (2000) testaram diversas configurações de redes, variando os dados de entrada tanto espacialmente quanto temporalmente, e posteriormente verificaram a configuração mais adequada para realizar a previsão. Nesse estudo, cujo objetivo é a previsão de precipitações, a camada de entrada é formada pelas chuvas atual e passada, isto é,  $x(t)$ , ...,  $x(t-k+1)$ , onde  $x(t)$  representa o vetor de chuvas em  $M$  postos no tempo  $t$ . O valor de  $k$  também varia. O número de variáveis de entrada foi tratado pelos autores como uma variável, que é testada por sua acurácia na previsão das chuvas.

Zealand et al.(1999), em sua previsão de vazões de curto prazo, utilizaram os seguintes dados de entrada: precipitação total semanal, temperatura média semanal e vazão média mensal. Para previsão de uma semana, os dados utilizados foram as precipitações médias das sete semanas anteriores, as sete últimas médias de temperatura, as duas últimas vazões afluentes, a precipitação média da semana seguinte e a temperatura média da semana seguinte, totalizando 18 dados de entrada. Para 2, 3 e 4 semanas foram incluídos, ainda, os valores médios da precipitação e temperatura da semana, além do valor previsto para a semana anterior, totalizando 21, 24 e 27 dados de entrada, respectivamente. Num análise de sensibilidade notaram que a quantidade de dados do experimento 1 era excessivo, reduzindo-se para 3 valores de precipitação, um de temperatura e quatro de vazão, para os quatro períodos pesquisados, totalizando 10 dados de entrada, todos eles de valores medidos, sem médias históricas. Pelos parâmetros de entrada,



mostrou-se importante um bom conhecimento local e das corretas variáveis hidrológicas que podem influenciar o resultado final.

Uma conclusão interessante é que uma maior quantidade de dados de entrada não produz, necessariamente, melhores resultados e que a inclusão de dados irrelevantes ao processo pode prejudicar o desempenho da rede.

## **NÚMERO DE CAMADAS INTERMEDIÁRIAS**

A flexibilidade das RNA vem do fato que o número de camadas escondidas ou até mesmo o número de nós em cada camada pode ser facilmente mudado. Esta flexibilidade permite a modelagem de sistemas complexos até mesmo quando há um conhecimento escasso sobre as relações entre as variáveis.

As RNA são formadas por várias camadas distintas mas, de acordo com Zealand et al (1999), em geral são suficientes somente três, uma de entrada, uma de saída e uma intermediária (que por não ter contato direto com o exterior da rede é também chamada de escondida). Em seu trabalho, no qual foi realizada a previsão de vazões de curto prazo, foi utilizada uma única camada escondida, devido ao fato que testes iniciais demonstraram que o uso de duas camadas trariam um ganho de desempenho da rede irrelevante e o tempo de processamento passaria, contudo, de 10 minutos para duas horas.

Em geral, o principal foco de um estudo fica nas camadas intermediárias. A maior parte dos estudos relacionados ao tema indica a predominância da utilização de uma camada intermediária. Segundo Shamseldin (1997), o uso de mais de uma camada intermediária aumenta substancialmente o número de parâmetros a serem estimados. Consequentemente o processo de calibração fica mais lento, embora não se obtenha um aumento significativo na eficiência do modelo. Assim sendo, a adoção de apenas uma camada escondida é a prática mais comum. Apesar das evidências, o emprego de mais camadas intermediárias não pode ser descartado antes de um análise prévia da situação que se deseja prever.

Sajikumar e Thandaveswara (1999) também realizaram testes com 1, 2 e 3 camadas escondidas e chegaram às mesmas conclusões, pois o tempo de processamento aumentou drasticamente e a performance não teve aumento significativo.

A determinação da topologia da rede a ser utilizada – o número de camadas, o número de unidades em cada camada, etc. – é normalmente realizada de forma empírica. A definição da configuração de redes neurais é ainda considerada uma arte, que requer grande experiência dos projetistas.

## NÚMERO DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA

Como visto no item anterior, em geral, nos artigos estudados, é usada apenas uma camada intermediária. O passo seguinte é analisar o número de neurônios utilizados nessa única camada intermediária.

Carriere et. al. (1996) optaram em seu modelo por utilizar 35 neurônios na camada escondida, numa arquitetura 7-35-1 (7 neurônios de entrada, 35 na camada intermediária e 1 neurônio de saída). Os autores entenderam que se uma rede com menos neurônios na camada escondida resultasse num modelo mais preciso, a própria rede se encarregaria de anular (peso zero) esses neurônios. Esse número foi escolhido através de uma rápida análise preliminar, sem maiores detalhes.

Shamseldin (1997) destaca que a definição do número de neurônios na camada intermediária é um importante passo no processo e um número de neurônios subestimados leva a uma performance deteriorada, e no caso da subestimação os dados de calibração podem se tornar insuficientes para uma boa modelagem, além de elevar o tempo de processamento. Justificando análises preliminares, o autor impôs um número de 2 neurônios, com os dados de entrada variando de um a cinco. Pelos resultados apresentados, esse parece ser um número baixo de neurônios, pois os resultados indicam que a rede se moldou bem aos dados de calibragem, mas o resultado de previsão não foi satisfatório.

A opção por apenas dois neurônios na camada escondida também foi feita por Thirumalaiah e Deo (1998) na aplicação em um método de previsão contínua de níveis de água em estações fluviométricas. Nesse caso, rede neural utilizada possuía apenas um dado de entrada e dois de saída. Ao autores fizeram uma comparação com três neurônios na camada intermediária, com resultados compatíveis. O tempo de processamento foi mais influenciado pelo algoritmo de treinamento que pelo número de neurônios.

A rede neural utilizada por Zealand et. al. (1999) é a que teve uma análise mais detalhada do número de neurônios na camada intermediária. O emprego da RNA foi feito com auxílio de um *software* próprio para redes neurais com maior flexibilidade. Foram estudadas diversas possibilidades de número de neurônios, com os mesmos dados de entrada na fase de treinamento, escolhendo a que obteve-se a melhor performance. As redes escolhidas foram 18-17-1, 21-9-1, 24-24-1, 27-18-1, 10-17-1, 10-5-1, 10-7-1 e 10-16-1. O que se pode notar é que não há um parâmetro ou regra aparente nos resultados. Há bons resultados com redes, com menos da metade do número de neurônios na camada intermediária, em relação aos dados de entrada, bem como com mais de 70% a mais de neurônios na camada intermediária, novamente em relação aos dados de entrada. Ou seja, o melhor resultado em cada caso, depende de quais são os dados de entrada, não sendo dependente de quantos ou quais são.

Sajikumar e Thandaveswara (1999) ajustaram a um modelo chuva-vazão uma rede com um dado de entrada (chuva) e um de saída (vazão). Testando-se várias arquiteturas, na camada intermediária, concluíram que o número ideal é de 5 a 7 neurônios, o que permitiu obter resultados semelhantes. Ao autores optaram por uma arquitetura 1-5-1 devido ao tempo de processamento.

De um modo geral, pelos artigos estudados, percebe-se que não há um consenso e definição exata quanto ao número de neurônios na camada intermediária. Em alguns estudos, o número é bastante variável, refletindo a dependência aos dados de entrada. O principal cuidado a se tomar é evitar a pré-definição deste valor, pois isso pode gerar um erro considerável. O ideal é que sejam usadas várias arquiteturas e defina-se qual se molda melhor ao estudo em questão. Mesmo assim, de uma forma geral, o número de neurônios na camada intermediária tem a tendência de fornecer melhores resultados com um número equivalente de dados de entrada e de saída, havendo numerosas exceções. O tempo de processamento tem uma de suas dependências na definição de número de neurônios na camada intermediária e em alguns casos esse tempo pode ser decisivo.

## **TREINAMENTO**

Uma fase importantíssima no uso da técnica de redes neurais artificiais é o treinamento da rede. Nessa fase os dados de treinamento são mostrados a rede e conforme o erro apresentado, os pesos são corrigidos na intenção de minimizar esse erro. Basicamente o objetivo do treinamento é a diminuição do erro global.

A forma e o critério de correção dessa fase de minimização do erro é um dos fatores decisivos à correta modelagem.

Os dados que serão mostrados a rede devem ser criteriosamente escolhidos, pois a rede deve ter os dados a serem moldados exibidos a ela. Uma rede neural é capaz de generalizações, não de deduções ou adivinhações.

A retropropagação recorrente foi utilizada por Carriere et. al. (1996) com a justificativa de que a retropropagação comum não poderia ser utilizada pelo fato do objeto de estudo ser um processo altamente dependente do tempo. A retropropagação comum é um processo estático e não produz bons resultados em processos cuja dependência do tempo é fator primordial. Em processos tempo-dependentes, todo ponto do processo é fator de pontos anteriores a ele tanto quanto outros fatores no processo. Num algoritmo comum de retropropagação todo formato do vetor de entrada é visitado, independentemente de outros formatos. Muitas vezes, uma visita aleatória dos formatos é sugerida, apesar de, em um processo tempo-dependente, a ordem de entrada dos formatos ser de grande importância. Ainda segundo Carriere et. al. (1996), numa rede neural espaço-temporal o

processo de retroalimentação é fundamental para capturar a essência do processo e a determinação da(s) camada(s) de retroalimentação é um assunto dependente do problema.

No treinamento (ou calibração) da Rede Neural de Shamseldin (1997), foi utilizado o algoritmo de Retropropagação. A função otimizada foi a soma dos erros quadráticos da saída estimada da rede. O gradiente descendente foi estimado através da derivada de primeira ordem na função. O algoritmo utilizado foi o de gradiente conjugado, em que uma parte dos parâmetros são fixados e a busca da maximização é nos restantes. O autor usa, em média, 72 meses de dados numa rede com a estrutura 1-2-1.

Thirumalaiah e Deo (1998) utilizaram três diferentes algoritmos: retropropagação do erro, gradiente conjugado e correlação em cascata.

No Algoritmo de Retropropagação do Erro de Thirumalaiah e Deo (1998) cada peso é inicialmente estipulado entre  $-0,5$  e  $0,5$ . A diferença entre o valor esperado e o calculado em cada nó é o gradiente de erro. Os pesos são corrigidos pela maximização da equação  $X_{k+1} = X_k - n \cdot g$  ( $X_{k+1} - 1$ ), onde  $X$  é o Vetor dos pesos,  $n$  é o passo de correção e  $g$  o vetor gradiente, função do erro médio global (de todos valores de treinamento). Esta etapa é repetida em interações até que se atinja um mínimo de erro médio quadrático da equação.

O segundo algoritmo utilizado por Thirumalaiah e Deo (1998) é o Algoritmo de Gradiente Conjugado. Ele difere da retropropagação do erro pelo cálculo dos gradientes e conseqüente correção dos pesos e propensões. Inicialmente são definidos aleatoriamente os pesos de cada ligação e então calculado o vetor gradiente de erro ( $g_0$ ) que define a direção de cálculo ( $d_0 = -g_0$ ). Para cada interação é determinada uma constante  $\alpha_k$  que minimiza a função  $f(X_k + \alpha_k \cdot d_k)$ . Assim procedido, tem-se  $X_{k+1} = X_k + \alpha_k \cdot d_k$ . O novo erro é calculado, sendo aceitável param-se as integrações, caso contrário uma nova direção de cálculo é definida através de novo gradiente de erro.

O terceiro e último algoritmo usado por Thirumalaiah e Deo (1998) é o Algoritmo de Correlação em Cascata. Enquanto no esquema de retropropagação todos os pesos são mudados simultaneamente para cada padrão de treino, na correlação em cascata apenas uma camada é modificada. A forma da rede neural não é predefinida, iniciando-se com zero nós na camada escondida, incluindo-se um a um até que seja atingido o critério. Não são envolvidos gradientes de erro e sim maximização do efeito (correlação) no erro residual. O erro residual é dado pela soma do módulo de todos os nós de saída, resultante da soma de todas os conjuntos de treinamento. Essa soma é feita pela diferença entre o valor de cada nó escondido e a média de todos os erros, multiplicado pelo erro residual observado em cada nó durante o treinamento e a média de todos os erros residuais.

Thirumalaiah e Deo (1998) perceberam que o Algoritmo de Retropropagação do Erro levava cerca de 300 vezes mais tempo para atingir um resultado satisfatório que os outros dois algoritmos. O Algoritmo de Correlação em Cascata definia a forma da rede e ainda sim foi o mais rápido dos três, apesar de levar a uma arquitetura mais simples. Ainda assim os autores concluíram que um maior tempo de processamento não significa melhores resultados. Os autores usaram 800 dados diários numa rede com arquitetura 1-3-2 e 1-2-2. Ainda assim, os próprios autores reconhecem que os dados para vazões altas não são satisfatórios, principalmente porque os dados fornecidos (como em grande parte dos rios) concentram-se nos valores de pequena vazão.

A rede neural utilizada por Zealand et. al. (1999) foi treinada (calibrada) para produzir as saídas desejadas através das entradas conhecidas. Os dados de treinamento devem representar com precisão o domínio do problema. Em geral, 80% das RNA utilizam-se de algoritmos de retropropagação para seu treinamento. Mais detalhes deste algoritmo são encontrados em Rumelhart et al. (1987) (citado por Zealand et. al. - 1999); seu funcionamento básico é feito através da comparação dos pesos existentes e dos desejáveis, fazendo um ajuste através do erro médio quadrático. Nesta pesquisa foi utilizado um *software* comercial para implementação da RNA, o *Braincel*, da *Promised Land Technologies*, da 1993.

O estudo de Sajikumar e Thandaveswara (1999) investiga a interferência que uma escassez de dados pode ter numa previsão. A rede utilizada é de arquitetura 1-5-1 e conclui que o resultado do treinamento com menos de 48 dados não é satisfatório, gerando bastante erros. Outra conclusão é que o resultado de 48 dados não difere muito daquele com três vezes mais dados.

Luk et. al.(2000) utilizaram o usual algoritmo de retropropagação. No entanto, eles sofisticaram um pouco o método introduzindo um *early stopping technique* - (técnica de parada antecipada). Essencialmente está técnica monitora o progresso do treinamento da rede utilizando um segundo conjunto de dados, chamados de dados de monitoramento. Eles são utilizados para checar o progresso da rede para definir a época na qual o treinamento deverá ser interrompido. O treinamento é interrompido quando o erro nos dados do monitoramento alcança um valor mínimo. Neste ponto, a rede alcançou a melhor generalização. Se o treinamento ultrapassa este ponto, a rede ativa mais pesos do que o necessário e passa a acrescentar ruídos aos dados de treinamento. Consequentemente, ocorre uma diminuição do erro nos dados de treinamento e um aumento do erro nos dados de monitoramento. Além disso, o tempo de processamento pode ser diminuído. Os autores mostraram 34 eventos diferentes de chuva, com dados de 15 em 15 minutos. A intenção era previsão de chuvas médias.

Como citado por Zealand et. al. (1999) 80% das RNA utilizam-se de algoritmos de retropropagação para seu treinamento. Pelos exemplos pesquisados, esse algoritmo pode onerar o processamento dos dados. Outras técnicas podem ser empregadas para substituir esse algoritmo ou

mesmo para incrementá-lo. Com a atual oferta de velocidade e modos de processamento, a definição de uma técnica de processamento pode ainda ser decisiva no uso de um sistema de redes neurais.

No tocante à quantidade de dados, o ideal é que sejam mostrados à rede os eventos com exemplos de todos os valores que se pretende prever ou modelar. A precisão dos resultados pode ser visualizada durante a própria calibragem, mas deve-se tomar cuidado para que a gama de opções seja tal, que os pesos possam ser gerais, não apenas moldados aos dados apresentados. Como recomendação geral, os dados não devem ser desprezados ou desperdiçados.

## **CONCLUSÃO GERAL E NOVAS INVESTIGAÇÕES**

De modo geral as redes neurais têm permitido obter resultados satisfatórios e melhores que os modelos estatísticos tradicionais. Ainda assim, essa é uma técnica relativamente pouco empregada no contexto a que se destina esta pesquisa bibliográfica e há muito a ser pesquisado.

Nos artigos analisados, notou-se que o conhecimento do problema não é de forma alguma descartado, sendo fundamental em alguns casos, para o correto equacionamento, o levantamento e uma boa definição de dados.

Apesar disso, a relação e o dimensionamento dos dados podem, muito bem, ser determinados pela própria rede, sem a necessidade de definir-se a priori quais e quantas exatamente são as variáveis de entrada e como elas devem ser modificadas ao longo do tempo e espaço.

Observa-se que, um caso difere de outro, e não há uma referência segura. Nada garante que, uma rede que funciona para um local, será eficiente para outro com características semelhantes.

A pesquisa bibliográfica aqui realizada servirá de embasamento para as aplicações e estudos de caso, no contexto de previsão de curto e médio prazos para diversas aplicações hidrológicas, que estão sendo desenvolvidas no âmbito da Universidade Federal do Paraná.

## **AGRADECIMENTOS**

Os autores deste artigo agradecem ao Centro de Hidráulica e Hidrologia Professor Parigot de Souza - CEHPAR e ao Instituto de Tecnologia para o Desenvolvimento - LACTEC pelo apoio recebido no desenvolvimento da pesquisa.

## REFERÊNCIAS BIBLIOGRÁFICAS

- CARRIERE, Patrick; MOHAGHEGH, Shab; GASKARI, Razi. *Performance of a virtual runoff hydrograph*. Journal of Water Resources Planning and Management, vol. 122 (1996), no. 6.
- LUK, K.C.; BALL, J.E.; SHARMA, A. *A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting*. Journal of Hidrology, Amsterdam: Elsevier, v.227. n.1-4, p.56-65, 2000.
- MENDES FILHO, Elson F., **Redes Neurais Artificiais** 1998. Disponível: site URL: <http://www.icmsec.sc.usp.br/~prico/neural1.html#topicos>.
- SAJIKUMAR, N.; THANDAVESWARA, B. S. *A non-linear rainfall-runoff model using an artificial neural network*. Journal of Hydrology 216 (1999).
- SHAMSELDIN; Asaad Y. *Application of a neural network technique to rainfall-runoff modeling*. Journal of Hydrology 199 (1997).
- THIRUMALAIAH, Konda; DEO, M. C. *River stage forecasting using artificial neural networks*. Journal of Hydrologic Engineering, vol. 3, no. 1 (1998).
- ZEALAND, Cameron M.; BURN, Donald H.; SIMONOVIC, Slobodan P. *Short term streamflow forecasting using artificial neural networks*. Journal of Hydrology 214 (1999).