

## Software para Determinação de Arquiteturas de Redes Neurais Artificiais Otimizadas à Modelagens Matemáticas Diversas

**José Leonel L. Buzzo   Eric A. Krause   Vinícius de Souza   José C. Rocha**

Departamento de Ciências Biológicas, Engenharia Biotecnológica, FCL UNESP Assis  
19806-900, Assis, SP

E-mail: jcelso@assis.unesp.br

### RESUMO

São demandas crescentes as implementações de tecnologias matemáticas inovadoras, pela indústria de ponta atual, em simulações, modelagens e controle de processos. Neste ínterim, as Redes Neurais Artificiais (RNA) interpelam-se. Emergentes dentre promissoras técnicas de modelagem matemática computacional, a saber – o processamento paralelo e distribuído – possuem a peculiaridade de poderem ser treinadas ou condicionadas a se adaptarem a um conjunto prévio de dados de aprendizagem, generalizando os padrões deles extraídos para o reconhecimento e a classificação de outros dados ainda não vistos, extrapolando sua previsibilidade a eventuais situações extremas de operabilidade e, ainda assim, mantendo sua consistência.

Eis que, no contexto deste preâmbulo, propõe-se aqui um software, desenvolvido na plataforma *Matlab*, visando à solução do problema da determinação de uma arquitetura de RNA otimizada à modelagem de um problema dado. E, como exemplo de um problema de pesquisa para aplicação e análise da viabilidade do software, tomou-se a questão ecológica relativa à predição de riqueza e abundância de macroalgas em ambientes lóticos [1 e 2], por se tratar de um problema com grande número de variáveis de entrada e saídas com comportamento acentuadamente não lineares.

No entanto, para que se entenda o papel do software enquanto ferramenta de construção automático de RNAs ótimas, há de se esclarecer que o desempenho de uma RNA é diretamente influenciado pelas configurações de sua arquitetura e, por vezes, uma pequena alteração destas pode acarretar um decréscimo significativo em seu desempenho, o que confere um caráter heurístico ao paradigma de determinação de uma melhor arquitetura candidata à resolução do problema em questão. Exemplificando-se, os parâmetros ou configurações arquitetônicas dentre os mais relevantes são, admitidamente na literatura [3 e 4], o número de camadas de neurônios, o número de neurônios em cada camada e a função ou algoritmo de treinamento. Esta pluralidade de fatores possibilita inúmeras combinações, resultando em um sem número de arquiteturas, sendo que poucas delas apresentam desempenhos viáveis à requerida demanda.

Não obstante, o supracitado software alude a uma alternativa útil, por promover sucessivas iterações onde ocorre a variação exaustiva de todos os parâmetros arquitetônicos ditos relevantes dentro de limites convenientemente pré-determinados àquela modelagem, na busca de uma arquitetura resultante com desempenho proeminente. Pormenorizadamente, têm-se as seguintes etapas:

**1 – Estrutura de dados principal:** O algoritmo solicita ao usuário os dados relativos aos padrões de entrada do problema e seus respectivos dados de saída. O conjunto dos pares entrada-saída é referido aqui como padrões de aprendizagem. Segue-se à delimitação dos intervalos de variação de cada parâmetro arquitetônico, a saber, o número mínimo e máximo de neurônios desejados em cada camada (neste caso duas camadas); a lista de funções de transferência de cada camada para com a seguinte (*tansig*, *logsig* e *purelin*); os algoritmos ou funções de treinamento (*trainlm*, *trainrp* e *traingda*) e o parâmetro de análise dos desempenhos

das saídas (podendo ser 1, 2 ou 3), tratando-se de uma escolha entre: 1. correlação linear para dados de saída quantitativos; 2. correlação não paramétrica para dados qualitativos e 3. matriz de confusão para saídas binárias ordinais. De posse destes dados, monta-se uma estrutura que será referida como argumento à função combinatorial customizada a ser utilizada em seguida.

**2 – Função combinatorial:** Esta função promove as sucessivas iterações onde, a cada uma, varia-se um dos parâmetros arquitetônicos e, sob o arranjo arquitetônico resultante, confecciona-se uma RNA, criando, treinando, validando, testando e calculando seu desempenho segundo o parâmetro de análise pré-estabelecido. Tal processo culmina ao ser salvo um grande *array* unidimensional que, em cada posição, encerra uma arquitetura de rede.

**3 – Análise dos dados resultantes e apresentação ao usuário:** Uma segunda função percorre o *array* realizando, segundo até três critérios distintos (já mencionados), a busca de uma melhor arquitetura resultante que será apresentada ao usuário. O primeiro critério compara os desempenhos, para cada uma das saídas de duas redes, um a um, resguardando a arquitetura cuja rede apresentou desempenhos que superaram todos os das outras arquiteturas. O segundo critério, seleciona a arquitetura cuja rede apresentou a maior média de coeficientes de correlação para suas saídas para todo o conjunto de padrões de entrada utilizado. Já o terceiro critério, perfaz uma ANOVA não paramétrica de dois critérios (parâmetro arquitetônico *versus* desempenhos de cada saída), recursivamente entre grupos de arquiteturas formados, segundo uma hierarquia, resultando em apenas uma arquitetura discrepante.

Logo, tendo sido feito uso deste software aplicado ao problema ecológico supracitado, obteve-se, como melhor resultado, o primeiro critério de análise de desempenhos. Os resultados estão apresentados a seguir.

Relativamente à saída que prediz a riqueza de macroalgas, uma arquitetura de duas camadas, com doze neurônios na primeira camada e sete na segunda, sob a função de treinamento *trainrp*, apresentando, respectivamente para treinamento, validação e teste, os desempenhos 90%, 85% e 87%.

Analogamente, na saída para predição da abundância, foi indicada uma arquitetura de duas camadas, com dezenove neurônios na primeira camada e nove na segunda, sob a função de treinamento *trainlm*, apresentando, respectivamente para treinamento, validação e teste, os desempenhos 96%, 86% e 87%.

**Palavras-chave:** *Redes Neurais Artificiais, Software, Ecossistemas Lóticos.*

## Referências

- [1] J. D. Allan; M. M. Castillo, Stream Ecology: structure and function of running waters. Ed. 2. Springer, Netherlands.
- [2] P. L. M. Goethals; A. P. Dedeker; W. Gabriels; S. Lek; N. Pauw, Applications of artificial neural networks predicting macroinvertebrates in freshwaters. *Aquat Ecol*, v.41, p.491-508, 2007.
- [3] Haykin, S., “Redes Neurais: princípios e prática”, 2ª ed, Bookman, Porto Alegre, RS, 2008.
- [4] Walzack, S. e Cerpa, N. Heuristic principles for the design of artificial neural networks. *Information and Software Technology*, v.41, n.2, p.107-117, doi: 10.1016/S0950-5849(98)0019-5. 1999.