

Modelagem de Redes Neurais Artificiais para predição da produção de H₂ por consórcios de microrganismos

Modelling of Artificial Neural Networks of the H₂ production by consortium of microorganisms

RESUMO

Hézila Raquel Dalla Costa
hezila@alunos.utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil

Elis Regina Duarte
erduarte@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil

Eduardo Sydney Bittencourt
eduardosydney@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil

O presente trabalho teve o objetivo de desenvolver um algoritmo no ambiente MATLAB a fim de obter uma Rede Neural Artificial (RNA) que modelasse a produção de H₂ pelo consórcio de microrganismos ESG. Foram testadas 12 configurações diferentes de RNA, variando a função de transferência, algoritmo de aprendizado e número de neurônios na camada oculta. Considerando o menor erro quadrático médio, a configuração escolhida foi a que usa a função de transferência logarítmica sigmoide, algoritmo de *backpropagation* de Levenberg Marquardt e 38 neurônios em uma única camada oculta. A configuração foi então repetida em 1.000.000 RNAs e dentre estas, a RNA com o melhor índice de correlação com dados experimentais foi escolhida. Dessa forma, o presente trabalho obteve sucesso na validação dos dois códigos desenvolvidos. Assim, foi possível encontrar uma RNA que modela com precisão o volume de H₂ produzido nas mais diversas condições de temperatura, pH, inóculo e razão carbono nitrogênio (C/N) pelo consórcio ESG.

PALAVRAS-CHAVE: Redes Neurais Artificiais. Produção de bio-hidrogênio. MATLAB.

ABSTRACT

This study had the objective of developing an algorithm in the MATLAB environment in order to obtain an Artificial Neural Network (ANN) that would model the production of H₂ by the consortium of microorganisms ESG. 12 different ANN configurations were tested, varying the transfer function, learning algorithm and number of neurons in the hidden layer. Considering the lowest mean square error, the configuration chosen was the one that used the sigmoid logarithmic transfer function, Levenberg Marquardt's backpropagation algorithm and 38 neurons in a single hidden layer. The configuration was then repeated on 1,000,000 ANNs and among these, the ANN with the best correlation index was chosen. Thus, this study was successful in validating the two codes developed. Therefore, it was possible to find an ANN that accurately models the volume of H₂ produced under the most diverse conditions of temperature, pH, inoculum and ratio carbon and nitrogen (C / N) by the ESG consortium.

KEYWORDS: Artificial Neural Networks. Production of bio-hydrogen. MATLAB.

Recebido: 19 ago. 2020.

Aprovado: 01 out. 2020.

Direito autorial: Este trabalho está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.



INTRODUÇÃO

O cérebro é o órgão mais complexo do corpo humano e é tido como o processador de informações mais potente conhecido pelo homem. Ele constrói um banco de dados que será usado nas decisões futuras, aperfeiçoando o processo, aprendendo sozinho (SILVA et al., 2011). O modelo desse órgão é tão fascinante que cientistas tentam replicá-lo de forma artificial, surgiram então as redes neurais artificiais (RNAs). Estas estruturas matematicamente complexas têm o mesmo princípio de funcionamento dos neurônios biológicos: as entradas são processadas e passam pelos neurônios através de impulsos elétricos, até que chegam à camada final onde ocorre a saída (decisão). Além disso, a RNA utiliza algoritmos de aprendizado para otimizar os resultados, assim como o cérebro humano.

Existem dois tipos de aprendizado: o supervisionado, onde buscamos prever uma variável dependente a partir das independentes, tendo um *target*. E o não-supervisionado que busca compreender padrões. Nos dois casos a RNA precisa de um conjunto de dados para usar como referência, os dados são divididos geralmente em 0,70; 0,15; 0,15. Cada uma das amostras é alocada aleatoriamente em um dos três grupos: treinamento, validação e testes, sendo 70% das amostras alocadas no primeiro e o restante dividido de maneira igualitária nos dois últimos.

Os dados do grupo treinamento serão usados para o ajuste inicial dos pesos da RNA. A rede irá percorrê-los até que se ajuste aos dados, no entanto há a possibilidade de *overfitting*, a RNA treinou tanto que é apenas capaz de prever corretamente os dados que já está “acostumada”, sendo ineficiente na predição de outras amostras. Entra então o conjunto de dados da validação, eles são usados para que a RNA possa ajustar os pesos mais uma vez, de forma “imparcial” já que as amostras desse grupo não lhe são familiares evitando o *overfitting*. Por último, há o conjunto de teste onde os dados são usados para avaliar a performance da RNA de maneira imparcial, já que as amostras desse grupo também nunca foram percorridas em nenhuma das etapas anteriores (BROWNLIE, 2017).

É necessário que haja um algoritmo que possibilite o aprendizado das RNAs, ele é definido de acordo com o comportamento do conjunto de dados estudado. Os quatro primeiros algoritmos combinam três parâmetros diferentes e segundo Brownlee (2017) os parâmetros são caracterizados da seguinte forma:

- Gradiente descendente: algoritmo de otimização iterativo que se ajusta à função de interesse pela taxa de aprendizado, que busca o menor erro;
- Aprendizado Adaptativo: método que adapta a taxa de aprendizado do gradiente descendente para que ele seja o maior possível a fim de chegar mais rápido à equação que se ajusta aos dados;
- Momentum: é responsável por supervisionar o gradiente descendente a fim de evitar que a estrutura dos dados “engane” o algoritmo, é um número entre 0 e 1 que faz com que o algoritmo consiga encontrar um conjunto de pesos melhor em menos períodos de treinamento.

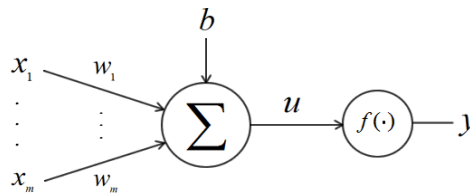
Os dois algoritmos restantes testados neste trabalho são o de Levenberg-Marquardt e o Bayesiano, onde o primeiro é indicado para treinamento supervisionado, e o segundo quando há um conjunto pequeno de dados e “[...]consiste na utilização do mecanismo de interferência bayesiana no cálculo dos parâmetros (pesos) da rede neural.” (HUARSAYA, 2009).

A estrutura de um neurônio artificial é mostrada na Figura 1, onde temos m entradas ($x_1...x_m$), cada uma delas será multiplicada por um peso sináptico ($w_1...w_m$), há também um deslocamento linear b (bias) e, finalmente, y a saída. “A

função $f(\cdot)$ é conhecida como função de ativação ou função de transferência. Ela é utilizada para adicionar uma não linearidade no sistema.” (PACHECO, 2015).

Todos os parâmetros supracitados são adaptáveis (pesos e bias) ou pré-definidos (função de transferência e algoritmo de aprendizado). Os algoritmos mostrados calculam o erro e o distribuem nos neurônios ajustando os pesos, o que faz com que a cada iteração a RNA se aproxime mais do resultado desejado.

Figura 1 – Estrutura de um neurônio



Fonte: PACHECO, 2015.

É importante ressaltar que a estrutura mostrada na Figura 1 é de apenas 1 único neurônio, as RNAs podem ter muitos deles, distribuídos em várias camadas, aumentando a complexidade do sistema. A primeira camada é chamada camada de entrada (*input*), é a receptora dos dados. A(s) camada(s) intermediárias, chamadas ocultas (CO), fazem os cálculos para que a RNA encontre o resultado que deseja. Por último, há a camada de saída, que é responsável por gerar o valor final encontrado pela rede. É devido à essa complexidade que as RNAs são usadas para previsões de dados, modelando sistemas complexos, como a vazão de rios, operações na bolsa de valores, ou ainda volume de produtos metabólicos gerados por microrganismos, este último é o caso do presente trabalho.

A produção de bio-hidrogênio vem sendo estudada devido ao seu apelo sustentável: a geração de energia limpa através de um processo orgânico. Entretanto, como afirma Sidney et al. (2019), devido à tamanha complexidade do processo, é difícil controlar, prever e monitorá-lo, o que afeta diretamente sua escala de produção. Há alguns modelos que tentam equacionar processos semelhantes, como o Modelo de Gompertz Modificado que modela a produção acumulada de biogás de biodigestores (SILVEIRA, 2018), todavia esse modelo tem limitações, principalmente quanto à quantidade de variáveis.

Há, portanto a necessidade de um modelo que admita vários parâmetros iniciais e bastante preciso, nesse aspecto, as RNAs proporcionam resultados satisfatórios (SIDNEY et al., 2019). Dessa forma, o presente trabalho objetiva desenvolver um método para identificar a melhor configuração de uma Rede Neural Artificial além de encontrar a RNA que mais se adequa à produção de bio-hidrogênio pelos microrganismos dispostos no consórcio ESG.

MATERIAL E MÉTODOS

Obteve-se um conjunto de dados a fim de validar os códigos produzidos, os dados utilizados são experimentais e produzidos nos laboratórios do Departamento Acadêmico de Engenharia de Bioprocessos e Biotecnologia (DAEBB). O consórcio denomina-se ESG, tem 27 amostras, variando quatro parâmetros: temperatura, pH, inóculo e razão carbono nitrogênio (C/N). Os dados

seguiram a divisão 0,70; 0,15; 0,15. Para decidir qual a configuração de RNA mais adequada aos dados, um script foi elaborado para que se pudesse comparar a performance de cada configuração. Foram analisadas RNAs geradas pelo MATLAB variando as funções de transferência, algoritmos de treinamento e o número de neurônios em uma única camada intermediária. Combinando os parâmetros analisados, chega-se a 12 casos possíveis que estão descritos no Quadro 1:

Quadro 1. Descrição de configurações.

Caso	Algoritmo	Função de Transferência
1	<i>Gradient descent backpropagation</i>	Logarítmica Sigmoide
2	<i>Gradient descent with adaptive learning rate backpropagation</i>	Logarítmica Sigmoide
3	<i>Gradient descent with momentum backpropagation</i>	Logarítmica Sigmoide
4	<i>Gradient descent with momentum and adaptive learning rate backpropagation</i>	Logarítmica Sigmoide
5	<i>Levenberg-Marquardt backpropagation</i>	Logarítmica Sigmoide
6	<i>Bayesian regularization backpropagation</i>	Logarítmica Sigmoide
7	<i>Bayesian regularization backpropagation</i>	Tangente hiperbólica sigmoide
8	<i>Gradient descent with adaptive learning rate backpropagation</i>	Tangente hiperbólica sigmoide
9	<i>Gradient descent with momentum backpropagation</i>	Tangente hiperbólica sigmoide
10	<i>Gradient descent with momentum and adaptive learning rate backpropagation</i>	Tangente hiperbólica sigmoide
11	<i>Levenberg-Marquardt backpropagation</i>	Tangente hiperbólica sigmoide
12	<i>Bayesian regularization backpropagation</i>	Tangente hiperbólica sigmoide

Fonte: Autoria própria (2020).

Para cada um dos doze casos o número de neurônios na CO variou de 1 até 50, totalizando 600 configurações, cada RNA teve seu erro quadrático médio (EQM) calculado conforme a Eq. (1), onde $\hat{\theta}$ é o valor estimado pela RNA e θ é o valor experimental (real), em n amostras. Além disso, cada uma das configurações foi repetida 120 vezes para que fosse possível extrair a média do erro.

$$EQM(\hat{\theta}) = \frac{1}{n} \sum_{i=1}^n (\hat{\theta}_i - \theta_i)^2 \quad (1)$$

Outro script foi então utilizado para encontrar os pesos da melhor RNA na configuração escolhida. Foram testadas 1.000.000 de RNAs e aquela com o maior coeficiente de correlação r de Person, Eq. (2), foi escolhida.

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \cdot \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (2)$$

Onde $x_1 \dots x_n$ sejam os resultados obtidos pela RNA e $y_1 \dots y_n$ sejam os resultados experimentais, ambos em n amostras e ainda que \bar{x} e \bar{y} são as médias aritméticas dos resultados obtidos pela RNA e de modo experimental, respectivamente.

RESULTADOS E DISCUSSÃO

O resultado da análise da melhor configuração está descrito na Tabela 1:

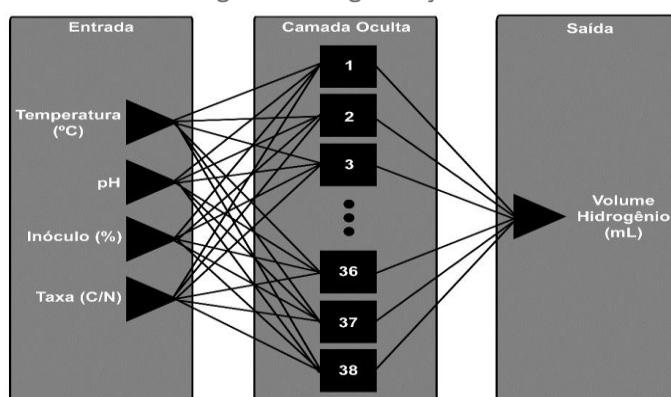
Tabela 1 – Resultado da análise de configurações

Número de Neurônios	Caso	EQM
38	5	0,003982077

Fonte: Autoria própria (2020).

A análise convergiu na configuração da RNA, sendo o caso 5 (Quadro 2) o mais adequado para o conjunto de dados testado, era esperado já que a MathWorks indica o algoritmo de Levenberg-Marquardt para RNAs com aprendizado supervisionado, assim como a deste estudo. O número de neurônios que obteve o menor EQM foi o de 38 neurônios, a configuração seguiu então para a segunda parte do trabalho. Sua estrutura é mostrada de forma esquemática na Figura 2.

Figura 2 – Organização da RNA



Fonte: Autoria própria (2020).

Na segunda fase foram testadas 1.000.000 de RNAs com a configuração da etapa anterior, destas, aquela que obteve o maior índice de correlação r comparando-se aos dados experimentais está descrita nas Tabelas 2 e 3.

Tabela 2 – Pesos das variáveis de Entrada

Neurônio	Peso Temperatura	Peso pH	Peso Inóculo	Peso (C/N)
1	2,049512992	1,773612752	2,133453603	0,012369453
2	0,589357175	2,936535931	-2,291567822	-0,089288069
3	0,397455634	-1,564977885	2,9855035	-0,580097293
4	2,243416403	-0,832982908	2,400934678	0,00919887
5	2,322885739	-0,111857225	2,316025888	-0,806505086
6	-1,577969278	-1,223582881	-1,511920951	2,369075349
7	0,408611788	-1,706631369	-0,038534783	2,989567082
8	0,924700678	1,399761648	-2,01748329	2,247095191
9	-1,469159493	-1,976399671	-2,266647636	-0,734628525
10	2,696247253	0,857851994	-1,901601671	1,286889798
11	0,772852304	1,956617142	0,164214373	2,777265699
12	-0,140633303	-1,10538373	-0,235403347	3,294196661
13	1,686939778	2,214746226	0,700363443	1,964923774
14	0,436703587	2,892182332	-1,188807713	-1,056240399
15	-0,552406955	0,117511209	2,51080877	-2,285485329
16	-0,601667397	2,074738725	-1,838046597	2,170738292
17	0,442783466	-0,884663033	2,364375921	-2,204269881
18	-0,676377562	-1,259288322	2,99610144	-0,925684247
19	0,747394395	-1,071271685	-1,90589931	-2,899264796

Neurônio	Peso Temperatura	Peso pH	Peso Inóculo	Peso (C/N)
20	-0,551066564	-2,787817694	-0,374276024	-1,924835287
21	-0,504988897	2,54619218	1,250760608	1,943738049
22	-2,80534612	-1,860364921	-0,6293886	-1,190381958
23	-1,871283579	-0,685514616	-1,00532297	2,716723976
24	-1,314593612	-2,170549526	0,736489882	2,223858308
25	0,644313165	-1,772933921	-1,599688464	2,610215816
26	-0,64768435	3,24076961	0,783620804	-0,467623148
27	2,859289919	0,874677346	0,389651352	1,045079979
28	0,406467603	0,856347043	2,706697819	1,85710194
29	0,200832456	-2,799903277	1,375650722	1,381365659
30	-1,540044078	2,056263993	0,541888242	2,349497192
31	1,814061018	0,094518859	0,922824761	-2,778161173
32	-0,651214463	0,234981309	-2,750870501	2,066083179
33	0,201009579	0,142082698	-2,285038462	-2,622617225
34	1,725241275	1,185072923	2,399715427	1,409902573
35	0,262235559	0,382370458	0,929898408	3,291651996
36	2,668252297	-0,85009413	-1,067457467	-1,869479735
37	0,737495067	1,299147677	-3,242512222	0,178303338
38	1,548829258	2,692684162	1,210773187	-0,594688726
38	1,548829258	2,692684162	1,210773187	-0,594688726

Fonte: Autoria própria (2020).

Tabela 3 – Valores das bias e pesos dos neurônios na camada intermediária

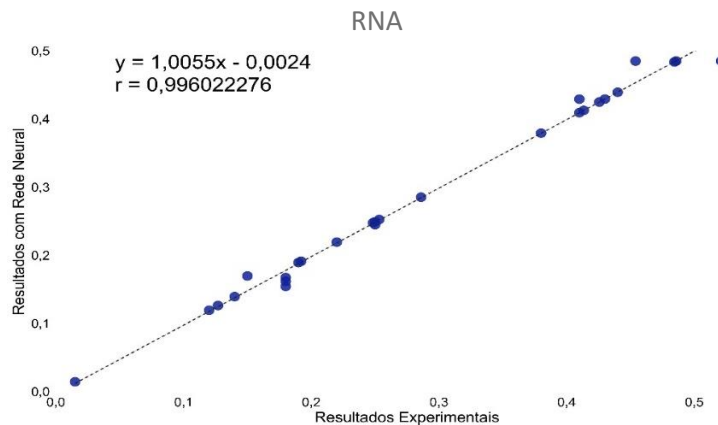
Neurônio	bias	Peso do Neurônio
1	-3,461500861	-0,099297299
2	-2,994474136	-1,35008987
3	-3,152725165	-0,422093674
4	-3,093841119	-0,58643103
5	-2,947375399	-0,64490418
6	2,569808901	-0,329310565
7	-2,393890093	-0,171081027
8	-2,114124303	0,29519073
9	2,071683693	0,329585662
10	-1,511002801	-1,056780872
11	-1,617588375	0,689358638
12	1,402478124	0,138589971
13	-1,272736235	-0,515666081
14	-0,690413171	-0,138979944
15	1,134048486	0,694743758
16	0,671211847	1,111646418
17	-0,487660977	0,238296897
18	0,298499088	-0,056025343
19	0,272122333	0,988916705
20	0,025894405	-0,264594858
21	-0,483379867	-0,0414811
22	-0,529416358	-0,941695923
23	-0,734227989	-0,917476995
24	-0,676631631	-0,078275484
25	0,992778028	0,516869352
26	-0,447172	0,472037172
27	1,712069998	-0,787307465

Neurônio	bias	Peso do Neurônio
28	1,762623985	0,645502055
29	1,735560124	0,185579496
30	-1,770548789	0,183415997
31	2,206108732	-0,702101004
32	-2,259613615	-0,0798652
33	2,520236802	0,078920365
34	2,716192383	0,034468918
35	2,996690033	0,387367368
36	3,088872971	0,459602093
37	3,207256521	-1,060058968
38	3,541138657	0,65342995
38	3,541138657	0,65342995

Fonte: Autoria própria (2020).

Por último, o valor encontrado na bia na camada de saída foi - 0,454372974. Para fins práticos, a RNA encontrada foi salva em forma de função no ambiente MATLAB. A Figura 3 mostra o gráfico que avalia a correlação entre os resultados experimentais comparados aos valores encontrados pela RNA. Percebe-se que o modelo matemático é consideravelmente preciso.

Figura 3 – Correlação entre os resultados experimentais e encontrados pela RNA



Fonte: Autoria própria (2020).

CONCLUSÕES

O presente trabalho obteve sucesso na validação dos dois códigos desenvolvidos. Assim, foi possível encontrar uma RNA que modela com precisão o volume de H₂ produzido nas mais diversas condições de temperatura, pH, inóculo e taxa (C/N) pelo consórcio ESG. Dessa forma, parte do problema na produção de H₂ por microrganismos é solucionada, já que prevendo o volume de gás produzido o processo pode ser melhor monitorado e controlado, melhorando o desempenho da operação.

AGRADECIMENTOS

À Fundação Araucária pela bolsa concedida.

REFERÊNCIAS

BROWNLEE, J. What is the Difference Between Test and Validation Datasets? **Blog Machine learning mastery**. Vermont Victoria, Austrália: Machine Learning Mastery Pty. Ltd., 2017. Disponível em: <https://machinelearningmastery.com/difference-test-validation-datasets/#:~:text=That%20the%20%E2%80%9Cvalidation%20dataset%E2%80%9D%20is,it%20to%20other%20final%20models.> Acesso em: 18 ago. 2020.

FONTANA, D. R. AND MARIM, L. R. Sistema de Autenticação/Identificação Pessoal Biométrica Através da Palma da Mão com o Auxílio de Redes Neurais Artificiais. **Anais do 14º Encontro de Iniciação Científica e Pós-Graduação do ITA**, São José dos Campos, p. 19-22, 2009. Disponível em: <http://www.bibl.ita.br/xvencita/FUND07.pdf> Acesso em: 24 jul. 2020.

HUARSAYA, E. A. **Aprendizado Bayesiano para Redes Neurais**. (Dissertação) – Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2009. Disponível em: <https://www.maxwell.vrac.puc-rio.br/colecao.php?strSecao=resultado&nrSeq=14538@1> Acesso em: 19 ago. 2020.

PACHECO, A. Introdução a Redes Neurais Artificiais. **Blog Computação Inteligente**, Vitória, 2015. Disponível em <http://computacaointeligente.com.br/artigos/redes-neurais-artificiais/> Acesso em: 19 ago. 2020.

SILVA, E. R. G.; OLIVEIRA, T. P. S; BEDIN, S. P. M.; ROVER, A J., Processamento cognitivo da informação. **Perspectivas em Gestão & Conhecimento**. Primeiro semestre, 2011, v. 1, n.1, p. 25-39. Disponível em: <https://periodicos.ufpb.br/ojs/index.php/pgc/article/view/9081> Acesso em: 18 ago. 2020.

SILVEIRA, S. C.; MUNIZ, J. A.; SOUSA, F. A.; CAMPOS, A. T. Modelos não lineares ajustados à produção acumulada de biogás provenientes de camas sobrepostas de suínos. **Revista Agrogeoambiental** Vol. 10. n.3, Lavras: Pouso Alegre, Setembro 2018.

SYDNEY, E. B.; DUARTE, E. R.; MARTINEZ, W. J. B; CARVALHO, J. C.; LARROCHE, C.; SOCCOL, C. R. Development of short chain fatty acid-based artificial neuron network tools applied to biohydrogen production. **Elsevier**. 2019, Vol. 45, pp. 5175-5181. Disponível em <https://doi.org/10.1016/j.ijhydene.2019.10.128> Acesso em: 19 ago. 2020.

MATLAB, **Users Guide** (R2016a). Natick, Massachusetts: The MathWorks Inc.; 2016.