

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

IAGO PATRICK MIORANZA FINGER

**APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA ANÁLISE DE DADOS
CINÉTICOS DE PROCESSOS DE GASEIFICAÇÃO**

**FRANCISCO BELTRÃO
2022**

IAGO PATRICK MIORANZA FINGER

**APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA ANÁLISE DE DADOS
CINÉTICOS DE PROCESSOS DE GASEIFICAÇÃO**

**Application of Artificial Neural Networks in the Analysis of Kinect Data of
Gasification Processes**

Trabalho de conclusão de curso de graduação
apresentada como requisito para obtenção do título
de Bacharel em Engenharia Química da Universidade
Tecnológica Federal do Paraná (UTFPR).

Orientador: Claiton Zanini Brusamarello.

FRANCISCO BELTRÃO

2022



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

IAGO PATRICK MIORANZA FINGER

**APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA ANÁLISE DE DADOS
CINÉTICOS DE PROCESSOS DE GASEIFICAÇÃO**

Trabalho de conclusão de curso de graduação
apresentada como requisito para obtenção do título
de Bacharel em Engenharia Química da Universidade
Tecnológica Federal do Paraná (UTFPR).

Orientador: Claiton Zanini Brusamarello.

Data de aprovação: 24/Junho/2022.

Prof. Claiton Zanini Brusamarello
Doutor em Engenharia Química
Universidade Tecnológica Federal do Paraná

Prof.^a Michele Di Domenico
Doutora em Engenharia Química
Universidade Tecnológica Federal do Paraná

Prof.^a Luciane Batistella
Doutora em Engenharia Química
Universidade Federal do Sul e Sudeste do Paraná

“A folha de aprovação assinada encontra-se na Coordenação do Curso”

FRANCISCO BELTRÃO

2022

Dedico este trabalho à minha família, que sempre me apoiou e me encorajou a estudar e me formar.

AGRADECIMENTOS

Agradeço ao Prof. Dr. Claiton Zanini Brusamarello pelos conselhos, pela orientação e pelo apoio. Além de me apresentar e ensinar o método das redes neurais artificiais.

Aos meus pais, Jacó e Silmara e a minha irmã, Thayane, que sempre me apoiaram desde sempre. Jamais estaria escrevendo isso se não fosse por vocês.

Agradeço aos colegas e grandes amigos que conquistei durante este curso, Aline Mafra, Alice Cristini, Bruna Barbieri, Eduarda Ghellere, Eduarda Leticia, Gustavo Persicotti, Ingrithy Vendruscolo, João Airoldi, Jonas Joacir, Matheus Mineli e Paulo Rubens.

E por fim agradeço a Deus, por ter me dado força para ir até o final e não desistir.

Educação não transforma o mundo. Educação muda as pessoas. Pessoas transformam o mundo. (FREIRE; PAULO, 1979).

RESUMO

O mundo está se tornando cada vez mais consciente sobre a necessidade de se utilizar fontes limpas de energia, principalmente na situação atual, onde a crise geopolítica fez com o suprimento de petróleo ficasse mais limitado no mundo. Uma dessas fontes alternativas de energia são os processos que fazem uso de matéria orgânica residual. Uma vez que a biomassa é uma matéria-prima renovável e pode ser aplicada em processos de geração de energia, como é o caso da gaseificação de biomassa, que está surgindo como uma fonte promissora de geração de energia. O grande problema desse processo é que muitas biomassas podem não ser adequadas para o mesmo e acabam tendo baixa eficiência. Para evitar isso são necessários estudos prévios da matéria-prima orgânica com o objetivo de dar uma previsão de sua efetividade para geração de energia em gaseificação. Esses estudos geralmente são demorados, e, para tentar contornar esse problema objetivou-se neste trabalho prever os parâmetros cinéticos de energia de ativação e fator pré-exponencial para a gaseificação com CO₂ de diferentes biomassas utilizando redes neurais artificiais (RNAs). Essas variáveis podem ajudar na análise de viabilidade de aproveitamento de biomassas em processos de gaseificação, sem a necessidade de estudos onerosos. Foi codificada uma rede neural artificial e foram realizados testes na mesma, para tentar obter uma RNA que conseguisse prever esses dados cinéticos sendo alimentada com dados mais facilmente obtidos. Foram utilizadas as estruturas de RNAs tipo *Feed-Forward*, *Cascade-Forward* e rede de Elman, alimentadas com dados de porcentagens dos principais elementos químicos presentes nas biomassas, porcentagens de cinzas, matéria volátil e carbono fixo e outros dados que se julgaram pertinentes para a realização dos testes. Quando os resultados iniciais não se demonstraram satisfatórios foram sendo feitas variações na metodologia para corrigir o problema. Embora essas variações não tenham corrigido completamente o problema, pois notou-se que os testes realizados apresentaram tendência de *overfitting*, foram dadas sugestões para melhorar a rede e aperfeiçoar a mesma. Os resultados desse trabalho e a base de dados obtida nele podem ser úteis em trabalhos posteriores para o desenvolvimento de um modelo viável de análise de gaseificação com RNAs.

Palavras-chave: biomassa, gaseificação, RNA, energia.

ABSTRACT

The world is becoming more and more aware of the need to use clean sources of energy, especially in the current conjuncture, where the geopolitical crisis as the supply of oil has become more limited in the world. One of the alternative sources of energy are the processes that make use of residual organic matter. Since biomass is a renewable raw material and can be applied in energy generation processes, such as biomass gasification, which has emerged as a promising source of energy generation. The big problem with the process is that many biomasses may not be suitable for it and end up with low efficiency. To avoid this, previous studies of organic raw materials are needed in order to provide a prediction of their effectiveness for the generation of energy in gasification. These studies are often delayed and, to try to circumvent this objective problem, we used this work to predict the kinetic parameters of activation energy and pre-exponential factor for CO₂ gasification of different biomasses using artificial neural networks (ANNs). These variations can help in the analysis of the feasibility of using biomass in gasification processes, without the need for costly studies. An artificial neural network was coded and tests were performed in the same way, to try to obtain an ANN that could predict this kinetic data being fed with more easily obtained data. The structures of Feed-Forward, Cascade-Forward and Elman network RNAs were used, fed with data of percentages of two main chemical elements present in the biomass, percentages of zinc, volatile matter and fixed carbon and other data that are considered relevant for a performing two tests. When the initial results were not satisfactory, there were several variations in the methodology to correct the problem. However, these variations did not completely correct the problem, so I noticed that the tests performed showed a tendency towards overfitting, so suggestions were given to improve the network and improve it. The results of the work based on the obtained data may not be useful in further work for the development of a viable model for the analysis of gasification with ANNs.

Keywords: biomass; gasification; ANN; energy.

LISTA DE FIGURAS

Figura 1 – Gaseificador de leito de arraste	16
Figura 2 – Gaseificador de leito fluidizado	17
Figura 3 – Gaseificador de leito fixo, fluxo ascendente	18
Figura 4 – Modelo de RNA.....	23
Figura 5 – Rede <i>Feed-Forward</i>	24
Figura 6 – Rede <i>Cascade-Forward</i>	25
Figura 7 – Rede de Elman	26
Figura 8 – Função <i>logsig</i>	28
Figura 9 – Função <i>tansig</i>	29
Figura 10 - Função <i>purelin</i>	29
Figura 11 – Função <i>satlin</i>	30
Figura 12 – RNA com 11 entradas e 2 saídas.....	32
Figura 13 – Imagens dos Rs, retiradas do MATLAB	35

LISTA DE TABELAS

Tabela 1 - Trabalhos com dados utilizados nos testes.....	31
Tabela 2 - Testes realizados com o código inicial	33
Tabela 3 - Diferentes proporções de treinamentos	34
Tabela 4 - Testes com código melhorado	34
Tabela 5 - Testes com o código inicial	36
Tabela 6 - Testes com maior proporção de treinamento	37
Tabela 7 - Testes com menor proporção de treinamento.....	39
Tabela 8 - Testes com código otimizado	40
Tabela 9 - Dados retirados da literatura	57
Tabela 10 - Testes com 1 saída	60
Tabela 11 - Testes com 2 saídas	61
Tabela 12 - Testes com proporções diferentes, maior proporção de treinamento	61
Tabela 13 - Testes com proporções diferentes, menor proporção de treinamento ...	67
Tabela 14 - Testes iniciais com código otimizado	69
Tabela 15 - Testes definitivos com primeiro conjunto de dados.....	70
Tabela 16 - Testes iniciais com segundo conjunto de dados	74

LISTA DE ABREVIATURAS E SIGLAS

ANN	Artificial Neural Network
MSE	Mean Squared Error (Erro quadrático médio)
R ²	Coefficiente de Determinação
RNA	Rede Neural Artificial
TGA	Termogravimetric Analysis (análise termogravimétrica)
X	Conjunto de dados de entrada, que teve alguns dados retirados para validação
X_VAL	Conjunto de dados de entrada que foi separado do original, para uso em validação da rede
X2	Conjunto de dados de entrada completo
Y	Conjunto de dados de saída, que teve alguns dados retirados para validação
Y_VAL	Conjunto de dados de saída que foi separado do original, para uso em validação da rede
Y2	Conjunto de dados de saída completo
FF	Rede <i>Feed-Forward</i>
CF	Rede <i>Cascade-Forward</i>
EL	Rede de Elman

LISTA DE SÍMBOLOS

E_a	Energia de ativação
k_0	Fator pré-exponencial
C	Carbono
H	Hidrogênio
N	Nitrogênio
O	Oxigênio
S	Enxofre
H ₂	Gás Hidrogênio
CO	Monóxido de Carbono
CO ₂	Gás Carbônico

SUMÁRIO

1	INTRODUÇÃO	13
2	OBJETIVOS	15
2.1	Objetivo geral.....	15
2.2	Objetivos específicos.....	15
3	FUNDAMENTAÇÃO TEÓRICA	16
3.1	Processo de gaseificação.....	16
3.2	Parâmetros cinéticos (E_a e k_0).....	19
3.3	RNA.....	20
3.3.1	Estruturas de uma RNA.....	22
3.3.1.1	Rede Feed-Forward	23
3.3.1.2	Rede Cascade-Forward	24
3.3.1.3	Rede de Elman.....	25
3.3.2	Overfeeding e underfitting em RNAs	26
3.3.3	Overtraining e overfitting em RNAs	27
3.3.4	Funções de ativação em RNAs	27
3.3.4.1	Função de ativação <i>logsig</i>	27
3.3.4.2	Função de ativação <i>tansig</i>	28
3.3.4.3	Função de ativação <i>purelin</i>	29
3.3.4.4	Função de ativação <i>satlin</i>	29
4	PROCEDIMENTOS METODOLÓGICOS.....	31
4.1	Testes iniciais	32
4.2	Testes com conjunto de dados totais	33
4.3	Testes variando proporção experimental	33
4.4	Testes com código otimizado	34
5	RESULTADOS E DISCUSSÃO	35
5.1	Resultados do conjunto de dados completo	35
6	CONCLUSÃO	44
	REFERÊNCIAS.....	46
	APÊNDICE A – CÓDIGO ORIGINAL	50
	APÊNDICE B – CÓDIGO OTIMIZADO.....	53
	APÊNDICE C – DADOS RETIRADOS DA LITERATURA	56
	APÊNDICE D – TABELAS COM TESTES COM GRUPO DE DADOS INICIAL	59
	APÊNDICE E – TESTES INICIAIS COM CONJUNTO DE DADOS COMPLETO ..	73

1 INTRODUÇÃO

Atualmente o petróleo está se tornando cada vez menos atrativo como uma fonte de energia, devido a sua escassez e geração de poluentes. Esses motivos desencadearam a busca por fontes alternativas para suprir as necessidades energéticas da população, sendo uma dessas fontes alternativas os combustíveis líquidos sintéticos, produzidos a partir de gás de síntese (*syngas*) que é basicamente composto de CO₂ e H₂ (Chojnacki *et al*, 2020).

Um dos principais processos de produção desse gás é a gaseificação de carvão ou de biomassa, com uso de vapor de água, sendo que a gasificação de biomassa é considerada melhor, do ponto de vista renovável. A gaseificação é muito usada na produção de gás de síntese com diversas matérias-primas com destaque para a produção a partir da biomassa lignocelulósica, que apresenta melhor custo benefício, em relação a métodos mais tradicionais de uso de biomassa (GAMERO, 2018).

É interessante conhecer as propriedades da matéria-prima que será gaseificada para aprimorar o processo e rendimento de gás de síntese a partir da gaseificação. Um jeito eficaz para verificar essas propriedades é a análise termogravimétrica (TGA). A análise de TGA monitora a variação de massa de uma amostra em função do tempo e/ou temperatura e fornece dados importantes das de suas propriedades, que podem servir, por exemplo, para aprimorar o processo de gaseificação da mesma.

Para análise da gaseificação alguns dos dados mais importantes que se pode obter de uma curva de TGA são o fator pré-exponencial e a energia de ativação, dados cinéticos que podem ser utilizados para conferir a qualidade do material a ser gaseificado e a eficiência do processo. Os parâmetros cinéticos são fundamentais para o projeto do equipamento de gaseificação.

Na atualidade, em acordo com os princípios da chamada indústria 4.0, é cada vez mais comum usar programas de computador para contornar dificuldades na indústria controlada pelas máquinas, pois graças às inovações tecnológicas recentes, é cada vez mais seguro e viável fazer com que um computador obtenha informações sobre todo um processo e as forneça de forma clara, para que um ser humano consiga tomar decisões. Se esse computador estiver bem programado a

chance de ele tomar uma decisão errada pode ser muitas vezes menor no gerenciamento do processo, ou seja, eliminando o erro humano, a chance de ocorrer algum problema com o processo diminui.

Um dos métodos utilizados para programar os computadores para que os mesmos possam tomar decisões é o método de redes neurais artificiais (RNAs). Esse método recebe esse nome pois simula o sistema nervoso biológico. Um dos diferenciais das RNAs é que elas podem reconhecer padrões de forma que a máquina aprenda com elas.

Outra vantagem das RNAs é que elas podem fazer a modelagem do comportamento do sistema com que se está trabalhando, mesmo que não exista uma fórmula matemática que descreva o sistema em questão; isso torna as RNAs atraentes para o uso em sistemas mais complexos, onde seria mais difícil encontrar parâmetros para postular as leis que são necessárias para a melhor faixa de trabalho do sistema. No caso das RNAs, não é preciso postulá-las, pois elas por si só são capazes de anexar essas leis durante seu processo de aprendizagem e desse modo, diminuem o trabalho e o tempo gasto antes de iniciar a produção, assim poupando recursos financeiros.

Pelo fato das RNAs possuírem a capacidade de modular problemas complexos, somente utilizando os dados experimentais, elas são amplamente utilizadas em problemas da área de engenharia. Um dos problemas desta área em que se utilizam as RNAs é o uso das mesmas para auxiliar na melhoria de processos de gaseificação, como é o caso dos trabalhos de OZONOH *et al.* (2020), de SHAHBAZ *et al.* (2019) e de GUO *et al.* (2001).

Logo, este trabalho tem por objetivo utilizar RNAs, mais especificamente os tipos: *Feed-Forward*, *Cascade-Forward* e *Elman*, para encontrar o fator pré-exponencial e a energia de ativação da gaseificação de biomassas, facilitando a aquisição de dados e o processo viável no caso de resíduos, mais eficiente e menos custoso.

2 OBJETIVOS

2.1 Objetivo geral

O objetivo geral desse estudo é desenvolver uma RNA que alimentada os dados de caracterização de biomassas e tempo de reação (porcentagens dos principais elementos químicos na biomassa, porcentagens de cinzas, matéria volátil e carbono fixo na mesma e outras variáveis que se julgaram pertinentes) consiga prever os parâmetros cinéticos de energia de ativação e fator pré-exponencial de reações de gaseificação com CO₂ de biomassas, de forma que a RNA possa, teoricamente, substituir a análise de TGA comumente aplicada para este tipo de processo.

2.2 Objetivos específicos

- Coletar na literatura os dados necessários ao desenvolvimento da rede: variáveis de entrada (dados de caracterização das biomassas e tempo de reação) e dados de saída (energia de ativação e fator pré-exponencial);
- Testar três diferentes arquiteturas de redes neurais, sendo elas a rede *Feed-Forward*, a rede *Cascade-Forward* e a rede de Elman;
- Comparar os resultados de cada modelo estatisticamente aos valores reais, com o uso do erro quadrático médio (MSE) e do R²;
- Encontrar o modelo que se enquadra melhor aos dados de energia de ativação e do fator pré-exponencial da gaseificação de biomassa.

3 FUNDAMENTAÇÃO TEÓRICA

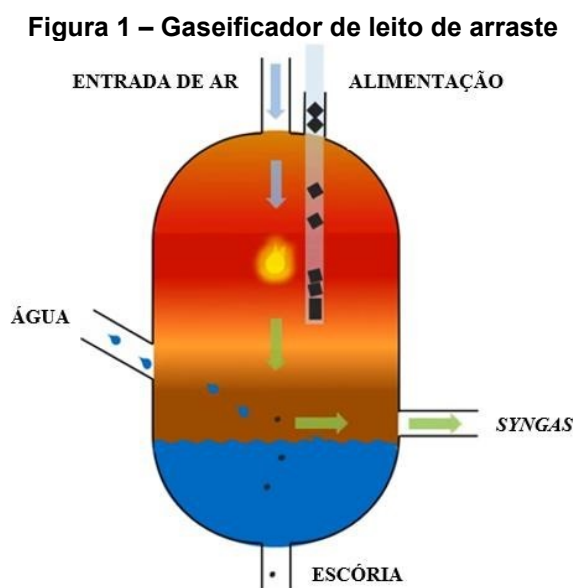
3.1 Processo de gaseificação

Quando se fala em gaseificação, normalmente se descreve esse processo como a reação entre combustíveis sólidos, como, por exemplo, carvão, biomassa e coque do petróleo, junto à oxidantes, como ar, oxigênio e vapor, em quantidades inferiores às estequiométricas, para converter os combustíveis sólidos em gases combustíveis ou gás de síntese (MINCHENER, 2005).

Os principais produtos primários da gaseificação são gás hidrogênio (H_2) e monóxido de carbono (CO), esses que, com o uso do vapor de água e oxigênio, são formados. Esse processo geralmente ocorre à pressão moderada e temperaturas que excedem $700^{\circ}C$, sendo um método limpo de geração de energia e obtenção de materiais químicos (DAI, HOADLEY, ZHANG, 2017).

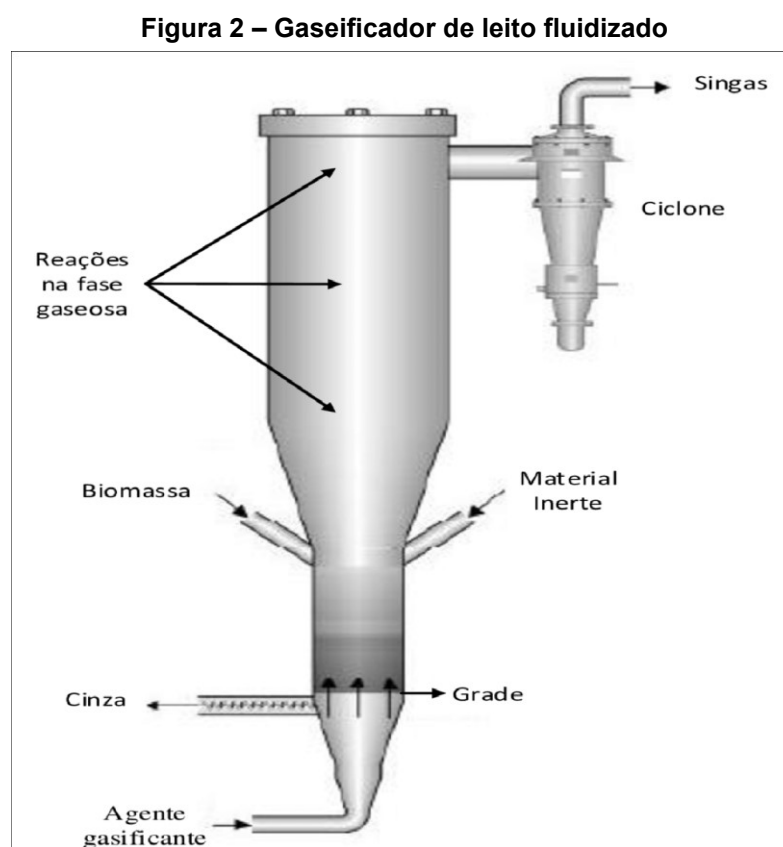
Existem vários tipos de tecnologias de gaseificação, e eles podem ser separados em três tipos principais, de acordo com a geometria de seus fluxos. Esses tipos são: gaseificadores de leito de arraste, gaseificadores de leito fluidizado e gaseificadores de leito fixo (COLLOT, 2006).

Nos gaseificadores de leito de arraste se tem combustível sólido pulverizado e fluxo de gases à alta velocidade. Esse fluxo de gases arrasta as partículas sólidas gerando o nome deste reator e. Um exemplo desse gaseificador está representado na Figura 1.



Fonte: Oliveira (2016)

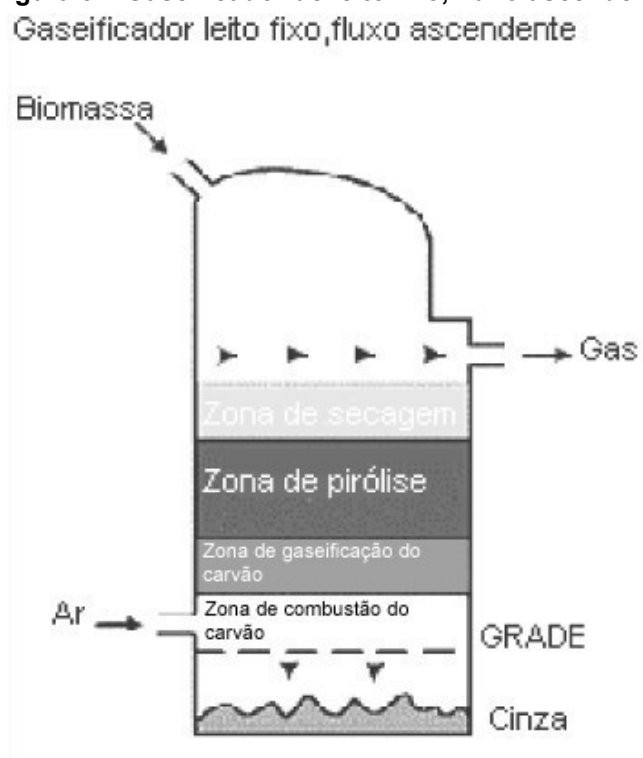
Nos gaseificadores de leito fluidizado, as partículas de sólido são suspensas num fluxo de gás, e, as partículas de entrada são misturadas com as partículas que já estavam sofrendo gaseificação. O nome desse tipo de gaseificador se deve ao fato de que os gases servem como o próprio leito nesses equipamentos. Um gaseificador de leito fluidizado está representado na Figura 2.



Fonte: Rey, Tuna e Silveira (2018)

Os gaseificadores de leito fixo, também podem ser chamados de leito móvel. Neles a corrente de gases flui lentamente para cima, através de um leito de combustível sólido. Existem tanto a forma concorrente, quanto a forma contracorrente, sendo esta última a mais comum. A diferença de nomenclatura se deve ao fato de que alguns autores consideram o sólido como leito e outros consideram o fluxo lento de gás como o leito (COLLOT, 2002). Um exemplo desse tipo de gaseificador está na Figura 3.

Figura 3 – Gaseificador de leito fixo, fluxo ascendente



Fonte: Kintol *et al.* (2002)

A demanda global de energia está crescendo mais rápido do que a disponibilidade da mesma, ao mesmo tempo que o mundo quer reduzir cada vez mais a emissão de carbono, para tentar diminuir o ritmo do aquecimento global. Essa informação torna ainda mais visível que as plantas de gaseificação são uma alternativa pouco explorada para produção de energia (KAMBLE *et al.*, 2019).

A maioria das plantas de gaseificação estão concentradas na Europa e nos Estados Unidos, demonstrando como essa é uma tecnologia que ainda não se espalhou totalmente pelo mundo, sendo mais comum nos países mais desenvolvidos. Esse também é o caso de várias tecnologias inovadoras, que demoram a ser adotadas por países menos desenvolvidos, muitas vezes por causa dos custos envolvidos na construção e adequação do sistema atual à nova fonte de energia.

Cerca de 19% dos processos de gaseificação tem função de produzir energia, cerca de 36% produzem combustíveis e cerca de 42% produzem elementos químicos que servem de matéria-prima para outros processos. Isso nos mostra que a gaseificação, apesar de ser um método de produção de energia mais limpa que as usinas termoelétricas, não é tão utilizada para esse objetivo. É claro que seu uso

para produção de substâncias químicas também é importante, mas não devemos deixar seu potencial energético de lado (EPA, 2016).

O fato de carvão ser o combustível sólido mais utilizado para gaseificação se deve à sua alta disponibilidade, dificultando a sua substituição nessa tecnologia. Outros fatores que podem ser mencionados incluem questões políticas, geográficas e, principalmente, econômicas. Para utilizar a biomassa na gaseificação, faz-se necessário, primeiramente, adaptar a tecnologia à essa alimentação (KAMBLE *et al*, 2019).

O uso quase que exclusivo do carvão como combustível prejudica bastante o processo de gaseificação e deve ser superado para que esse processo possa se desenvolver com todo seu potencial, não sendo em escala tão limitada, como é hoje. Uma alternativa crescente ao carvão como fonte de energia é o uso de biomassa, que é uma fonte renovável de energia.

3.2 Parâmetros cinéticos (E_a e k_0)

Energia de ativação e fator pré-exponencial são duas das principais variáveis cinéticas para análise de processos de gaseificação.

A E_a de uma reação química é uma variável cinética que indica a energia mínima que é necessária para que uma reação entre moléculas aconteça. Durante o momento do choque das mesmas, se a energia disponível for menor que a energia de ativação dessa reação, a reação não acontecerá e as moléculas se separarão, como se o choque entre elas não tivesse acontecido (OpenStax, 2016).

Pelo fato da energia de ativação ser a energia necessária para uma reação ocorrer, quanto maior ela for, mais difícil é de ocorrer a reação. Dessa forma uma energia de ativação muito alta pode até mesmo inviabilizar uma dada reação.

O k_0 , por sua vez, indica a frequência com que as moléculas colidem umas com as outras. Dessa forma um fator pré-exponencial maior aumenta a taxa de reação em um sistema (Kasseboehmer, 2015).

Conhecer a energia de ativação e o fator pré-exponencial de uma gaseificação serve para verificar sua viabilidade utilizando uma dada matéria-prima, além da necessidade de catalisadores para alguma das etapas do processo.

Geralmente, quando se quer obter as informações cinéticas de um processo de gaseificação é necessário realizar um experimento em analisador termogravimétrico (TGA). Esses dados, somados ao gráfico de degradação do combustível de gaseificação, são utilizados para otimização do processo de gaseificação (ÁVILA, CRNKOVIC, MILILI, 2007).

3.3 RNA

Uma rede neural artificial, RNA, é um modelo computacional ou matemático que tenta simular uma rede neural biológica (cérebro). Consiste em camadas interligadas de neurônios, onde a resposta de uma camada é a entrada da próxima, e ela usa essas conexões para tentar resolver algum problema proposto. Uma RNA aprende com exemplos e consegue achar padrões entre os dados de entrada e de saída, desse modo podendo resolver até problemas matemáticos muito complexos, sem precisar de uma fórmula matemática para isso (JI *et al*, 2011).

Um dos problemas das RNAs é que para elas terem uma boa precisão, elas precisam de uma quantidade de dados robustos para seu treinamento, ou seja, elas precisam de vários dados de entrada e seus respectivos dados de saída, todos encontrados por algum outro método. Esses dados experimentais juntos servem para que a RNA possa encontrar padrões. Quanto maior e melhor o número de dados para treinamento, maior a precisão da RNA (DAHMANI *et al*, 2014).

As RNAs são muito utilizadas para resolver problemas complexos, principalmente os que possuem não linearidades. Sua aplicação se estende em várias áreas da engenharia, principalmente onde métodos convencionais falham. Apesar de seu grande uso em engenharia, elas também têm sido utilizadas em várias outras aplicações, como por exemplo, para predição do tempo, controle de processos, medicina, otimização de processos e de manufatura, sistemas energéticos e processamento de sinais (TAGHAVIFAR, MARDANI, 2014).

Com a RNA é possível obter diretamente as informações de E_a e k_0 sem a necessidade de obtenção das perdas de massa durante o aquecimento da amostra em um TGA. Dessa forma, elimina-se o trabalho experimental demorado e oneroso, simplificando o trabalho por meio de uma RNA. A alimentação de variáveis de

entrada das amostras neste caso torna o processo de cálculo da rede muito mais ágil, facilitando a otimização da gaseificação.

O uso de RNAs em gaseificação já é feito há anos, como pode-se notar pelo trabalho de Bing Guo *et al.* (2001) que foi realizado há mais de 20 anos. Os autores conduziram a gaseificação de diversos tipos de biomassa em um reator de leito fluidizado e simultaneamente utilizaram uma RNA para encontrar o perfil de gaseificação de cada amostra. As taxas de produção de gás dos processos de gaseificação das amostras de biomassa previstas pela RNA foram consistentes com os resultados experimentais obtidos, demonstrando que as RNAs podem ser efetivas para esse tipo de processo.

Mesmo que tenha começado há anos, o estudo do uso de RNAs em processos de gaseificação é ainda escasso, porém, permanece em contínua expansão. Esta é uma área extensa que ainda possui muito a ser pesquisada, como pode-se avaliar pelo trabalho de Ozonoh *et al.* (2020), feito recentemente. Os autores utilizaram uma RNA para estudar um conjunto de dados contendo 315 experimentos de biomassa, carvão e misturas dos dois. Eles usaram 11 dados de entrada e 9 dados de saída, para tentar prever o rendimento da gaseificação.

Como dados de entrada, os autores utilizaram as porcentagens de carbono, hidrogênio, oxigênio, nitrogênio, enxofre, cinzas, massa volátil e umidade; temperatura de gaseificação; razão de ar/combustível; e poder calorífico inferior. Como dados de saída, eles utilizaram as porcentagens de produtos mais comuns da gaseificação; o poder calorífico inferior da mistura de gases; o rendimento de gás; a eficiência de gás frio; e, a eficiência de conversão de carbono. Com esta pesquisa, os autores encontraram os produtos da gaseificação, sem precisar realizar o processo em si, somente utilizando dados do combustível sólido. Desse modo, evitou-se a realização experimental do processo de gaseificação utilizando matérias-primas pouco viáveis, ou seja, que poderiam gerar quantidades suficientes do produto desejado.

Outro trabalho recente que usa RNA em gaseificação é o de Shahbaz *et al.* (2019). Nesse trabalho, os autores usam RNA para prever a composição de gás, o rendimento de gás, o poder calorífico inferior do gás e o poder calorífico superior do gás em um processo de gaseificação de resíduos de óleo de palmeira, usando cinzas e CaO como catalisadores. Eles usam dados de experimentos de um artigo

anteriormente publicado pelos próprios autores e os resultados da RNA se mostraram próximos aos resultados experimentais publicados, com um R^2 de 0,998 em quase todos os casos.

3.3.1 Estruturas de uma RNA

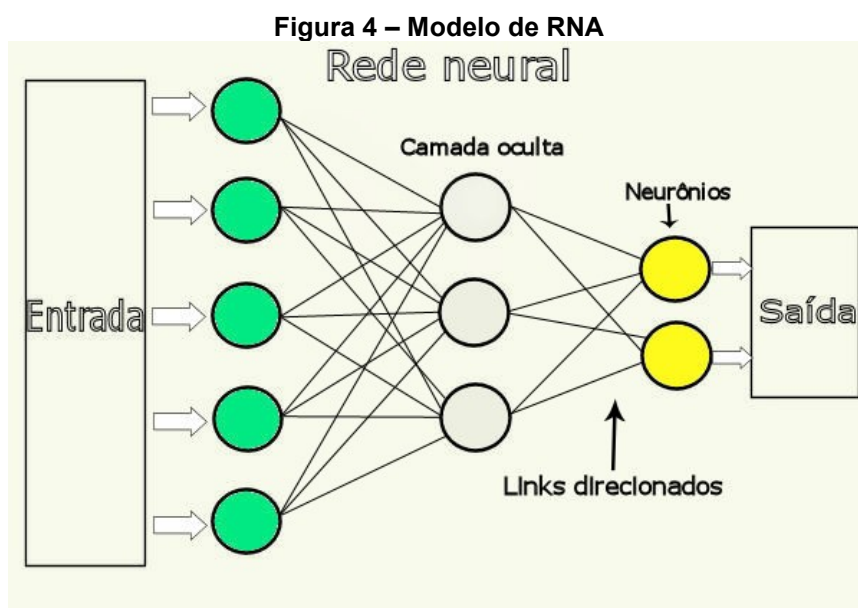
As RNAs são classificadas em três tipos principais, de acordo com seu número de camadas e se há ou não realimentação: camada simples, múltiplas camadas e recorrentes (HAYKIN, 2001).

No tipo de camada simples nós temos somente a camada de entrada e a camada de saída, sem ter nenhuma camada oculta. Isso torna a resposta muito rápida, mas ela tem menor grau de precisão, por isso esse tipo de RNA não é muito utilizado.

As RNAs de múltiplas camadas têm, além das camadas de entrada e saída, camadas ocultas. Nesse tipo de RNA a saída de uma camada é a entrada para a próxima camada, até chegar à saída da RNA. O número maior de camadas permite que esse tipo de RNA tenha uma precisão visivelmente superior à RNA de camada simples, mas esse tipo só pode conter unidades lineares, tornando-o assim limitado.

Por fim, as RNAs recorrentes também têm várias camadas, além disso, têm realimentação, ou seja, nelas a saída de uma camada além de ser entrada da próxima camada, também é utilizada como uma das entradas dessa mesma camada ou de camada anterior. Esse tipo de camada permite unidades não-lineares, aumentando assim a sua abrangência, porém as várias conexões entre camadas podem fazer com que tenha resposta mais lenta. (HAYKIN, 2001).

A Figura 4 representa um modelo de RNA de múltiplas camadas, tendo um total de três camadas, sendo uma delas oculta.

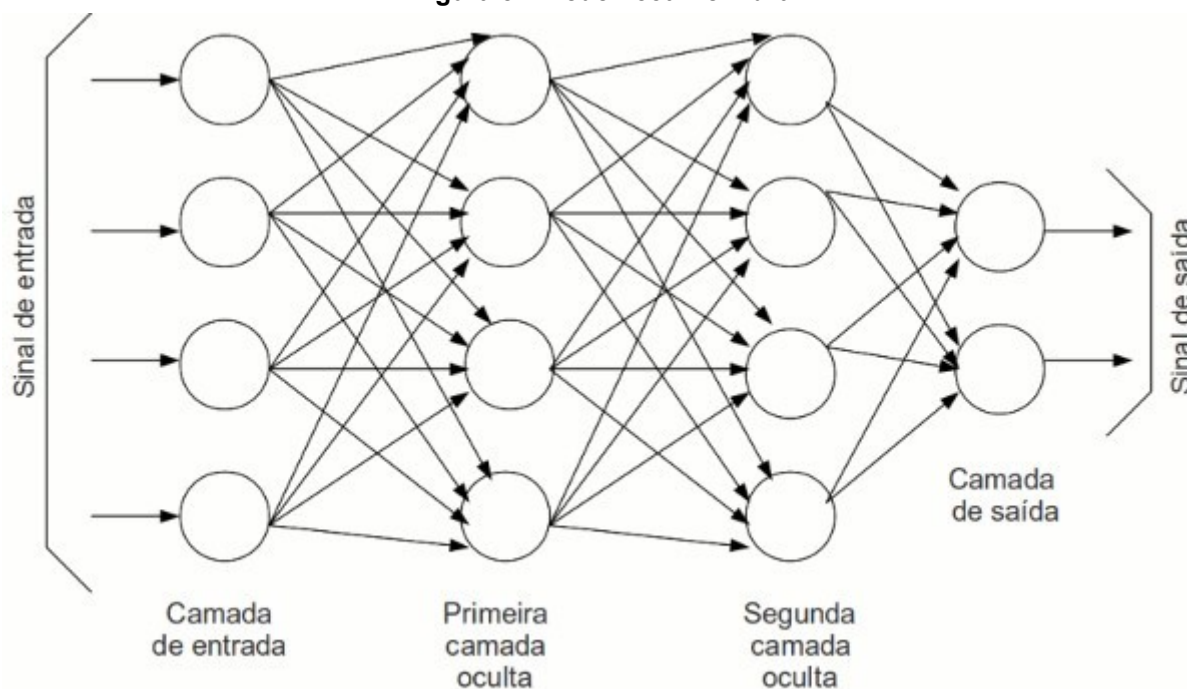


Fonte: Oliveira (2011)

Para cada situação, uma das classes de RNAs pode se adequar melhor, então é preferível fazer testes com várias classes em cada processo, para se encontrar a melhor para o processo em questão. Com testes é possível achar tanto a melhor classe quanto o melhor número de camadas e de neurônios para o processo com que se vai trabalhar e dessa forma ter-se a melhor RNA, levando a melhor precisão para resolução do problema, em menor tempo.

3.3.1.1 Rede Feed-Forward

A rede tipo *Feed-Forward* é um dos tipos mais básicos de RNA. Nessa rede a resposta de um neurônio em uma camada passa para a camada diretamente depois desta e a informação flui somente para frente, uma camada de cada vez. Pelo fato da informação de uma camada posterior não retornar para uma camada anterior, esse tipo de rede não consome muita memória do computador que está executando os cálculos. Por isso, essa é das redes com resposta mais rápida, mas pode ter precisão insuficiente para determinados problemas (CECCON, 2020). A Figura 5 ilustra uma rede do tipo *Feed-Forward*,

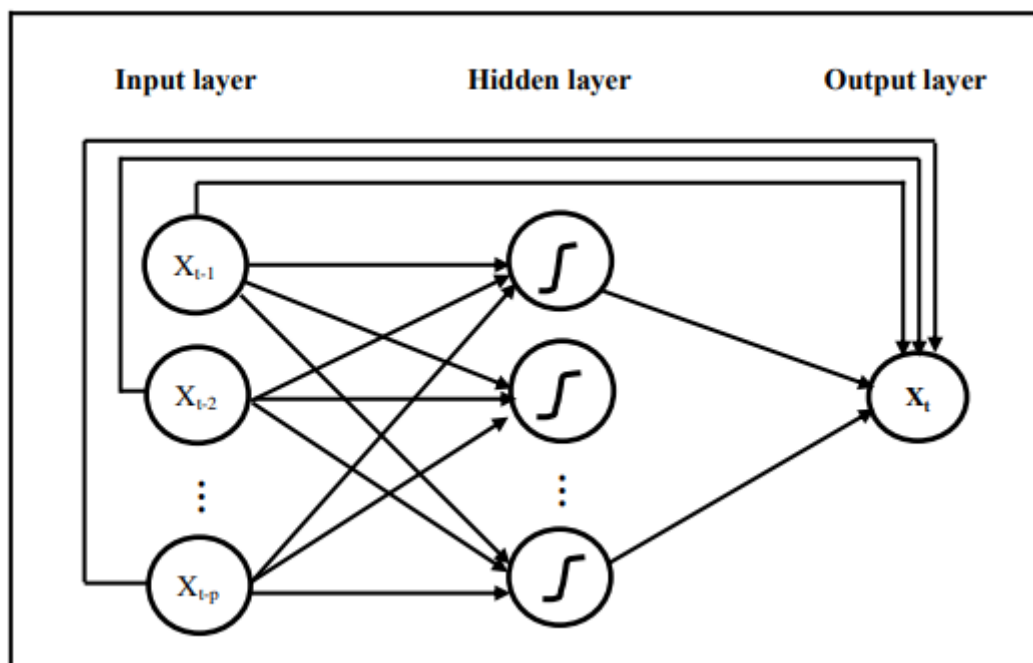
Figura 5 – Rede *Feed-Forward*

Fonte: Vinicius (2017)

3.3.1.2 Rede Cascade-Forward

A rede tipo *Cascade-Forward* é similar a *Feed-Forward*, no quesito que a informação flui somente para frente. A diferença está no fato de que enquanto a saída dos neurônios de uma camada da rede *Feed-Forward* é a entrada da camada diretamente posterior, na rede *Cascade-Forward* a saída de um neurônio de uma camada pode alimentar qualquer camada depois desta. Assim, um neurônio na primeira camada oculta pode alimentar um neurônio na última camada oculta, não importando quantas outras camadas há entre eles.

Por causa dessa forma de alimentação, os neurônios das últimas camadas podem receber diversas entradas, o que pode causar que a resposta desses neurônios necessite mais cálculos e por isso seja mais lenta. Pelo fato da saída de um neurônio não ser utilizada somente para a entrada da próxima camada, mas pode perdurar por toda a rede, isso faz com que as variáveis que tem influência maior no resultado possam demonstrar mais essa influência e por isso a rede *Cascade-Forward* pode gerar precisão maior nesses casos (WARSITO *et al*, 2018). A Figura 6 ilustra uma rede *Cascade-Forward*.

Figura 6 – Rede *Cascade-Forward*

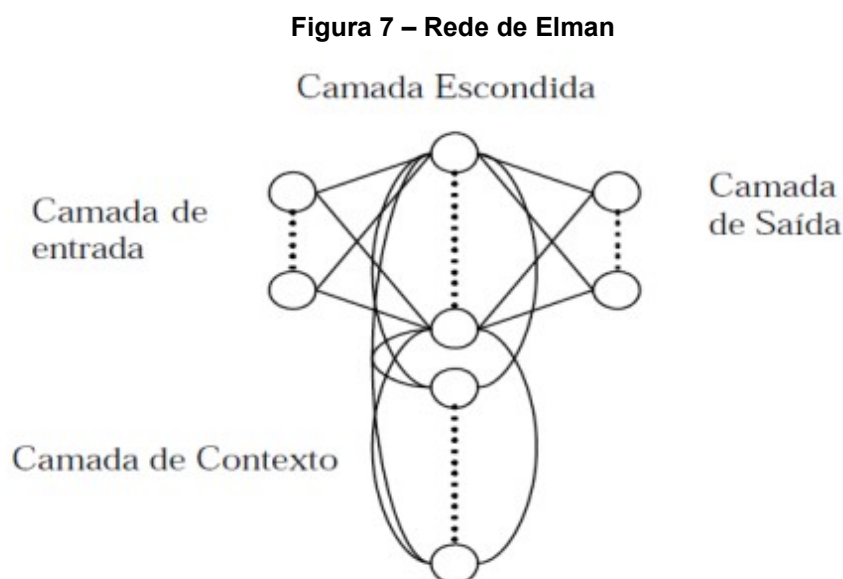
Fonte: Warsito et al. (2018)

3.3.1.3 Rede de Elman

A terceira e última estrutura de RNA utilizada nesse trabalho é a rede de Elman. Essa rede tem esse nome em homenagem a Jeffrey Elman, um psicolinguista de Universidade da Flórida, que foi o primeiro a usar esse tipo de rede. A estrutura de rede proposta por Elman é a única entre as três redes utilizadas que tem retroalimentação de dados, ou seja, em que a saída de uma camada pode ser usada como entrada de uma camada anterior a esta. A estrutura de Elman além de ter as camadas de entrada, saída e ocultas, tem também uma camada de contexto, que é sempre retroalimentada após cada camada oculta. Isso quer dizer que na rede de Elman os dados entram pela camada de entrada, são calculados na primeira camada oculta, são retroalimentados na camada de contexto, são alimentadas na segunda camada oculta, são retroalimentados na camada de contexto e assim em diante até chegar a camada de saída (FLORENCIO, VIEIRA, 2016).

A vantagem desse tipo de rede é que ela dispõe de retroalimentação, o que pode torná-la mais robusta que as estruturas anteriores e a sua desvantagem é que

ela fica muito dependente da camada de contexto e por isso pode acabar tendo precisão inferior aos outros tipos, já que qualquer variação na camada de contexto gera uma reação em cadeia na rede de Elman. A Figura 7 ilustra uma rede de Elman.



Fonte: Florencio e Vieira (2016)

3.3.2 Overfeeding e underfitting em RNAs

Para que as RNAs funcionem corretamente é necessário tanto ter camadas e neurônios suficientes para que a rede possa calcular as saídas, quanto um conjunto de dados robusto, para que a rede tenha exemplos suficientes em que se basear e desenvolver modelos matemáticos.

Quando se coloca poucas camadas ou poucos neurônios por camada em uma RNA, acontece o *underfitting*, onde a RNA não consegue realizar testes suficientes por não ter processadores suficientes e por isso ela tem problemas em fazer um modelo que se adeque aos dados de treinamento fornecidos. Por esse modelo não servir bem nem com os dados de treinamento, ele geralmente se comporta ainda pior com dados de validação (BRANCO, ano desconhecido).

Uma outra tendência onde a RNA não consegue gerar um modelo adequado aos dados é quando ocorre o *overfeeding*. Nesse caso a RNA é alimentada com

dados demais e não consegue encontrar nenhum modelo que se adeque de forma suficientemente boa com todos eles.

O *underfitting* e o *overtraining* podem parecer a mesma coisa, mas o primeiro ocorre por falta de processadores, enquanto o segundo ocorre por excesso de dados.

3.3.3 Overtraining e overfitting em RNAs

No outro extremo de problemas de RNA, temos o *overtraining* e o *overfitting*. Nesse caso a RNA desenvolve um modelo muito específico e focado para os dados de treinamento. Isso gera um modelo que se adequa muito bem aos dados de treinamento, mas que perde precisão quando utilizado para dados de validação (GOYAL, 2021).

O *overfitting* ocorre quando o tamanho da RNA excede o valor ótimo para o tamanho do grupo de dados e o *overtraining* ocorre quando a RNA treina por tempo demais em um único teste embora ambos levem a um modelo superespecializado para um grupo específico de dados o primeiro é muito mais comum que o segundo (TEKTO, LIVINGSTONE, LUIK, 1995)

3.3.4 Funções de ativação em RNAs

Para cada camada oculta de uma RNA é necessário se ter uma função de ativação. Essa função é utilizada em todos os neurônios da camada oculta na qual a mesma será relacionada.

A necessidade de funções de ativação é normalizar os dados dentro de uma faixa definida e dessa forma se ter entradas e saídas entre intervalos conhecidos, assim aumentando a eficiência de trabalho da RNA.

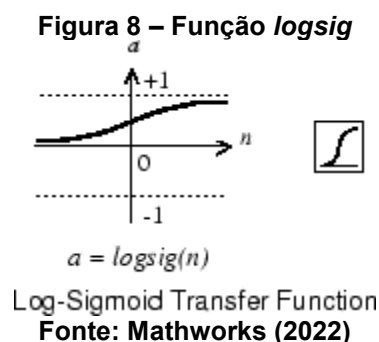
3.3.4.1 Função de ativação *logsig*

A função de ativação *logsig* gera uma saída de 0 a 1 e ela funciona de acordo com a Equação 1.

$$a = \text{logsig}(n) = 1 / (1 + \exp(-n)) \quad (1)$$

Onde “a” é a saída da função de ativação e “n” é a sua entrada.

O funcionamento dessa equação está ilustrado na Figura 8. Como pode-se perceber conforme a entrada da função aumenta, a sua saída tende a 1 e conforme a entrada diminui, a saída tende a zero (MathWorks, 2022).



3.3.4.2 Função de ativação *tansig*

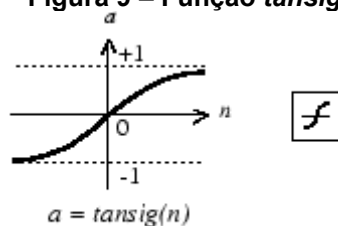
A função de ativação *tansig* gera uma saída entre -1 e 1 e ela funciona de acordo com a Equação 2.

$$a = 2 / (1 + \exp(-2*n)) - 1 \quad (2)$$

Onde “a” é a saída da função de ativação e “n” é a sua entrada.

Essa equação está ilustrada na Figura 9. Como pode-se perceber conforme a entrada da função aumenta, a sua saída tende a 1 e conforme a entrada diminui a saída tende a -1 (MathWorks, 2022).

Figura 9 – Função *tansig*



Tan-Sigmoid Transfer Function

Fonte: Mathworks (2022)

3.3.4.3 Função de ativação *purelin*

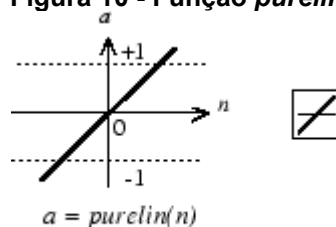
A função de ativação *purelin* gera uma saída entre -1 e 1 e ela funciona de acordo com a Equação 3.

$$a = n \quad (3)$$

Onde “a” é a saída da função de ativação e “n” é a sua entrada.

Essa equação está ilustrada na Figura 10. Como pode-se perceber a saída dessa função é igual à entrada. Por causa de limitações nas RNAs é melhor trabalhar com dados entre -1 e +1 e por isso, para se utilizar essa função os dados devem ser normalizados antes de serem utilizados como entrada da rede (MathWorks, 2022).

Figura 10 - Função *purelin*



Linear Transfer Function

Fonte: Mathworks (2022)

3.3.4.4 Função de ativação *satlin*

A função de ativação gera uma saída entre 0 e 1 e ela funciona de acordo com as Equações 4, 5 e 6.

$$a = 0, \text{ se } n < 0; \quad (4)$$

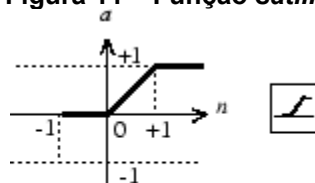
$$a = n, \text{ se } 0 \leq n \leq 1; \quad (5)$$

$$a = 1, \text{ se } n > 1. \quad (6)$$

Onde “a” é a saída da função de ativação e “n” é a sua entrada.

O resultado dessas equações trabalhando em conjunto está na Figura 11. Como pode-se perceber a saída da função *satlin* é zero se a entrada é negativa ou igual a zero, ela é igual à entrada se a mesma é entre zero e 1, e a saída é 1 se a entrada é maior que 1 (MATHWORKS, 2022).

Figura 11 – Função *satlin*



$$a = \text{satlin}(n)$$

Satlin Transfer Function

Fonte: Mathworks (2022)

4 PROCEDIMENTOS METODOLÓGICOS

Para dar início a esse estudo, foram selecionados dados na literatura sobre gaseificação com CO₂ em reator de leito fixo, aplicada em biomassas para alimentar o modelo da rede neural artificial. Para se utilizar os dados para este trabalho os mesmos deveriam estar em base seca, quando isso não era atendido foi feito tratamento sobre esses dados, para transformá-los em base seca.

As biomassas estudadas, o número de amostras de dados retirados de cada trabalho e os autores desses trabalhos estão apresentados na Tabela 1. Todos os dados utilizados estão disponíveis no Apêndice C.

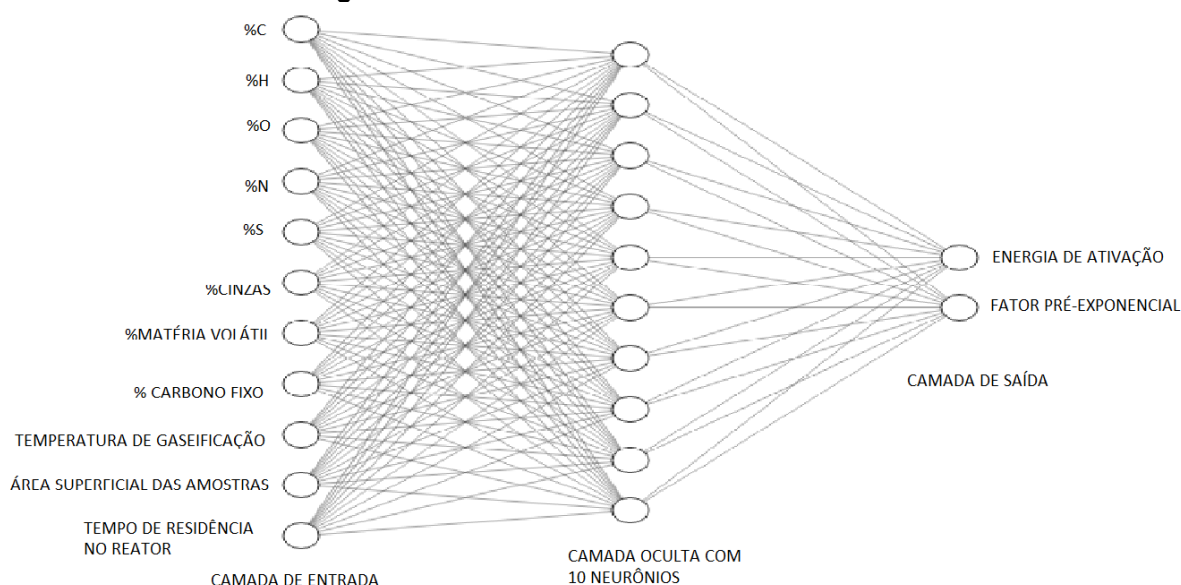
Tabela 1 - Trabalhos com dados utilizados nos testes

Amostras	Consuegra (2013)	Pacioni (2013)	Zuo, <i>et al.</i> (2015)	Seo, <i>et al.</i> (2010)	Tian <i>et al.</i> (2021)
Biomassa 1	Bagaço de laranja	Bagaço de maçã-pirólise lenta	Serragem de pinho	Pinheiro	<i>Miscanthus</i>
Biomassa 2	Serragem de madeira	Bagaço de maçã-pirólise rápida	Palha de trigo		
Biomassa 3		Borra de café	Espiga de milho		
Biomassa 4			Casca de amendoim		
Total de dados retirados do trabalho	6 conjuntos de dados	9 conjuntos de dados	16 conjuntos de dados	4 conjuntos de dados	9 conjuntos de dados

Fonte: Autoria própria (2022)

As concentrações de carbono (C), hidrogênio (H), nitrogênio (N), enxofre (S), oxigênio (O) cinzas (Cz), matéria volátil (MV) e carbono fixo (CF) (todas em base seca), tempo de residência no reator, temperatura de gaseificação e área superficial das amostras, foram utilizados para alimentar o modelo de RNA como parâmetros de entrada e a energia de ativação e fator pré-exponencial como variáveis de saída. Essa RNA está ilustrada na Figura 12.

Figura 12 – RNA com 11 entradas e 2 saídas



Fonte: Autoria própria (2022)

Os dados obtidos da literatura foram separados em dois conjuntos, um conjunto com 80% dos dados para um treinamento e validação inicial da rede e os outros 20% para validação secundária da mesma.

O primeiro conjunto, com 80% dos dados, foi separado em 70% dos dados para treinamento, 15% para testes e 15% para validação inicial. Resultando em uma RNA contendo 11 variáveis de entrada e duas variáveis de saída.

Para que os testes fossem realizados, construiu-se um código para o MATLAB®2021a, versão 2021a. O computador que se utilizou para essa codificação é da marca HP®, tem um processador Intel-Core i7-4700 MQ, 8GB de Memória RAM e sistema operacional de 64 bits.

Utilizando o *software* MATLAB®2021a, foram testados diferentes tipos de arquiteturas. As três arquiteturas de redes neurais testadas foram *Feed-Forward*, *Cascade-Forward* e de Elman. Também foram testados o número de neurônios presentes nas camadas e a quantidade de camadas presentes no modelo. Para esses testes foram utilizadas as funções tangente sigmoidal (*tansig*), logarítmica sigmoidal (*logsig*), função linear (*purelin*) e função de transferência de saturação (*satlin*).

4.1 Testes iniciais

Para os primeiros testes foi utilizado um conjunto de dados reduzido, somente com os dados dos 4 primeiros trabalhos (Consuegra (2013), Pacioni (2013), Zuo *et al* (2015) e SEO *et al* (2010)). Mas os mesmos apresentaram tendência de *overfitting*, então buscou-se na literatura mais dados e obteve-se os dados de Tian *et al* (2021) para complementar o conjunto de amostras.

4.2 Testes com conjunto de dados totais

Foram feitos diferentes testes para determinar a viabilidade da RNA. Primeiramente foram feitos 50 testes com o código inicial (Apêndice A), de acordo com a Tabela 2. Para que pudesse ser observado o desempenho da rede durante os testes, foi obtido o R^2 e o MSE da rede em cada teste.

Tabela 2 - Testes realizados com o código inicial

Número dos testes	Arquitetura da RNA	Funções de ativação da primeira camada oculta	Funções de ativação da segunda camada oculta	Número de Neurônios
1 - 5	<i>Feed-Forward</i>	<i>logsig</i>	<i>logsig</i>	10,20,30,40,50
6 - 10	<i>Feed-Forward</i>	<i>logsig</i>	<i>tansig</i>	10,20,30,40,50
11 - 15	<i>Feed-Forward</i>	<i>logsig</i>	<i>satlin</i>	10,20,30,40,50
16 - 20	<i>Feed-Forward</i>	<i>logsig</i>	<i>purelin</i>	10,20,30,40,50
21 - 25	<i>Feed-Forward</i>	<i>tansig</i>	<i>logsig</i>	10,20,30,40,50
26 - 30	<i>Feed-Forward</i>	<i>tansig</i>	<i>tansig</i>	10,20,30,40,50
31 - 35	<i>Feed-Forward</i>	<i>tansig</i>	<i>satlin</i>	10,20,30,40,50
36 - 40	<i>Feed-Forward</i>	<i>tansig</i>	<i>purelin</i>	10,20,30,40,50
41 - 45	<i>Feed-Forward</i>	<i>purelin</i>	<i>logsig</i>	10,20,30,40,50
46- 50	<i>Feed-Forward</i>	<i>purelin</i>	<i>tansig</i>	10,20,30,40,50

Fonte: Autoria própria (2022)

4.3 Testes variando proporção experimental

Após a realização dos primeiros testes da rede com o conjunto de dados completo, foram feitos novos testes semelhantes, somente variando-se a proporção de dados, para verificar se a proporção 75-15-15 era a mais adequada ou se podia ser melhorada. Na Tabela 3 estão indicadas as diferentes proporções durante cada conjunto de testes.

Tabela 3 - Diferentes proporções de treinamentos

Número do conjunto de testes	Treinamento	Validação	Testes
1	70%	15%	15%
2	75%	15%	10%
3	65%	20%	15%

Fonte: Autoria própria (2022)

4.4 Testes com código otimizado

Após a realização dos testes com proporções diferentes de dados, foram feitas algumas alterações no código, para melhorar o funcionamento da rede. O novo código (Apêndice B) tinha somente uma camada oculta e não fazia uso de dois conjuntos de dados separados, ou seja, todos os dados estavam num conjunto só. A diminuição no número de camadas ocultas diminuiu o treinamento que a rede fazia, assim amenizando a tendência de *overfitting* que já havia sido observada anteriormente e a junção de todos os dados em um conjunto fez com que a rede tivesse um maior conjunto de dados de treinamento, melhorando sua eficiência.

Com o novo código otimizado foram realizados testes com os três tipos de redes, como apresentado na Tabela 4.

Tabela 4 - Testes com código melhorado

Número dos testes	Arquitetura da RNA	Funções de ativação da camada oculta	Número de Neurônios
1 - 5	<i>Feed-Forward</i>	<i>logsig</i>	10,20,30,40,50
6 - 10	<i>Feed-Forward</i>	<i>tansig</i>	10,20,30,40,50
11 - 15	<i>Feed-Forward</i>	<i>satlin</i>	10,20,30,40,50
16 - 20	<i>Feed-Forward</i>	<i>purelin</i>	10,20,30,40,50
21 - 25	<i>Cascade-Forward</i>	<i>logsig</i>	10,20,30,40,50
26 - 30	<i>Cascade-Forward</i>	<i>tansig</i>	10,20,30,40,50
31 - 35	<i>Cascade-Forward</i>	<i>satlin</i>	10,20,30,40,50
36 - 40	<i>Cascade-Forward</i>	<i>purelin</i>	10,20,30,40,50
41 - 45	Elman	<i>logsig</i>	10,20,30,40,50
46 - 50	Elman	<i>tansig</i>	10,20,30,40,50
51 - 55	Elman	<i>satlin</i>	10,20,30,40,50
56 - 60	Elman	<i>purelin</i>	10,20,30,40,50

Fonte: Autoria própria (2022)

5 RESULTADOS E DISCUSSÃO

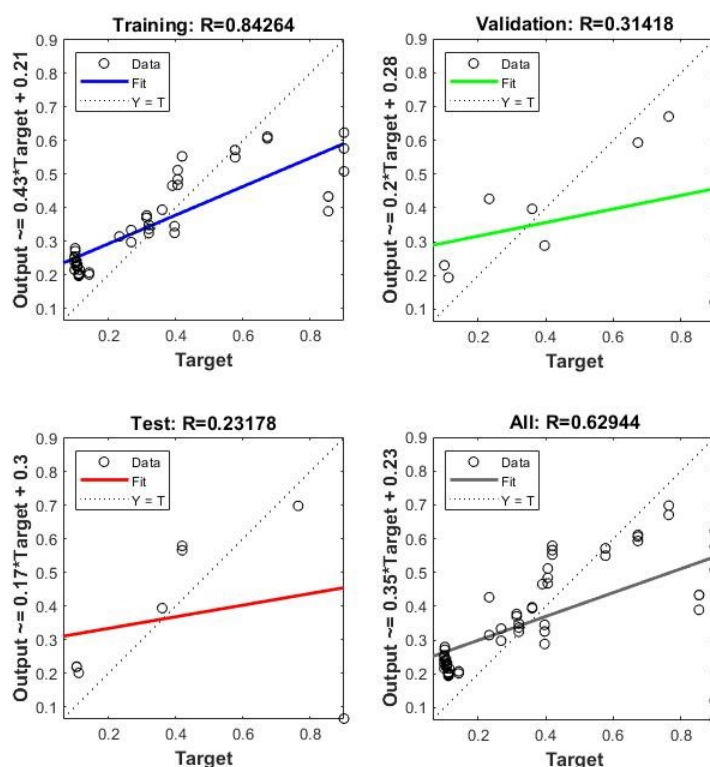
Os primeiros testes que foram realizados, com o grupo menor de dados, não apresentaram resultados conclusivos, mas, estão presentes no Apêndice D, para justificar a sequência de cálculos conduzida e complementar os dados desse trabalho.

5.1 Resultados do conjunto de dados completo

Primeiramente, utilizou-se o código com somente uma saída, para verificar se o mesmo era viável. Verificou-se que o mesmo era viável (como disposto no Apêndice E), logo, foi dado início aos testes reais.

Para se obter os dados de R^2 de cada rede, foi utilizada uma função inerente do MATLAB, que apresenta os Rs em forma de gráfico, foi anotado o valor do R geral e o mesmo foi elevado ao quadrado e também com valor numérico, como mostrado na Figura 13.

Figura 13 – Imagens dos Rs, retiradas do MATLAB



Fonte: Autoria própria (2022)

Os resultados dos primeiros testes estão sendo apresentados na Tabela 5 onde foram feitos testes somente com a estrutura de rede tipo *Feed-Forward* e com as 4 funções de ativação (*logsig*, *tansig*, *purelin*, *satlin*).

Tabela 5 - Testes com o código inicial

(continua)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	$7,50 \cdot 10^{-7}$	0,62944
2	FF	<i>logsig</i>	<i>logsig</i>	20	$1,33 \cdot 10^{-11}$	0,85678
3	FF	<i>logsig</i>	<i>logsig</i>	30	$1,41 \cdot 10^{-14}$	0,86876
4	FF	<i>logsig</i>	<i>logsig</i>	40	$7,87 \cdot 10^{-15}$	0,91376
5	FF	<i>logsig</i>	<i>logsig</i>	50	$2,69 \cdot 10^{-12}$	0,71714
6	FF	<i>logsig</i>	<i>tansig</i>	10	$1,91 \cdot 10^{-16}$	0,85929
7	FF	<i>logsig</i>	<i>tansig</i>	20	$2,07 \cdot 10^{-16}$	0,96687
8	FF	<i>logsig</i>	<i>tansig</i>	30	$3,50 \cdot 10^{-12}$	0,92890
9	FF	<i>logsig</i>	<i>tansig</i>	40	$3,85 \cdot 10^{-12}$	0,45389
10	FF	<i>logsig</i>	<i>tansig</i>	50	$8,36 \cdot 10^{-13}$	0,95950
11	FF	<i>logsig</i>	<i>purelin</i>	10	$1,39 \cdot 10^{-11}$	0,97610
12	FF	<i>logsig</i>	<i>purelin</i>	20	$8,86 \cdot 10^{-16}$	0,87332
13	FF	<i>logsig</i>	<i>purelin</i>	30	$6,52 \cdot 10^{-12}$	0,71706
14	FF	<i>logsig</i>	<i>purelin</i>	40	$6,19 \cdot 10^{-14}$	0,91165
15	FF	<i>logsig</i>	<i>purelin</i>	50	$4,24 \cdot 10^{-16}$	0,71625
16	FF	<i>logsig</i>	<i>satlin</i>	10	$4,38 \cdot 10^{-6}$	0,71811
17	FF	<i>logsig</i>	<i>satlin</i>	20	0,186	0,30778
18	FF	<i>logsig</i>	<i>satlin</i>	30	$5,60 \cdot 10^{-9}$	0,83229
19	FF	<i>logsig</i>	<i>satlin</i>	40	$6,51 \cdot 10^{-15}$	0,98803
20	FF	<i>logsig</i>	<i>satlin</i>	50	$8,52 \cdot 10^{-7}$	0,89861
21	FF	<i>tansig</i>	<i>logsig</i>	10	$1,29 \cdot 10^{-7}$	0,75487
22	FF	<i>tansig</i>	<i>logsig</i>	20	$1,66 \cdot 10^{-8}$	0,78011
23	FF	<i>tansig</i>	<i>logsig</i>	30	$4,97 \cdot 10^{-15}$	0,93698
24	FF	<i>tansig</i>	<i>logsig</i>	40	$4,32 \cdot 10^{-14}$	0,54391
25	FF	<i>tansig</i>	<i>logsig</i>	50	$1,83 \cdot 10^{-20}$	0,74306
26	FF	<i>tansig</i>	<i>tansig</i>	10	$7,29 \cdot 10^{-16}$	0,93750
27	FF	<i>tansig</i>	<i>tansig</i>	20	$1,04 \cdot 10^{-15}$	0,88498
28	FF	<i>tansig</i>	<i>tansig</i>	30	$1,07 \cdot 10^{-18}$	0,93492
29	FF	<i>tansig</i>	<i>tansig</i>	40	$8,04 \cdot 10^{-12}$	0,70001
30	FF	<i>tansig</i>	<i>tansig</i>	50	$5,26 \cdot 10^{-13}$	0,92942
31	FF	<i>tansig</i>	<i>purelin</i>	10	$9,95 \cdot 10^{-12}$	0,92202
32	FF	<i>tansig</i>	<i>purelin</i>	20	$7,28 \cdot 10^{-11}$	0,21987
33	FF	<i>tansig</i>	<i>purelin</i>	30	$4,52 \cdot 10^{-14}$	0,25325
34	FF	<i>tansig</i>	<i>purelin</i>	40	$7,96 \cdot 10^{-14}$	0,70612
35	FF	<i>tansig</i>	<i>purelin</i>	50	$6,10 \cdot 10^{-14}$	0,20443
36	FF	<i>tansig</i>	<i>satlin</i>	10	$7,48 \cdot 10^{-6}$	0,67652
37	FF	<i>tansig</i>	<i>satlin</i>	20	$6,30 \cdot 10^{-13}$	0,91319
38	FF	<i>tansig</i>	<i>satlin</i>	30	$1,05 \cdot 10^{-8}$	0,88028
39	FF	<i>tansig</i>	<i>satlin</i>	40	$1,21 \cdot 10^{-18}$	0,90046
40	FF	<i>tansig</i>	<i>satlin</i>	50	$1,36 \cdot 10^{-6}$	0,85425
41	FF	<i>purelin</i>	<i>logsig</i>	10	$2,23 \cdot 10^{-8}$	0,98834

Tabela 5 - Testes com o código inicial

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	(conclusão)
						R ² médio (rede)
42	FF	<i>purelin</i>	<i>logsig</i>	20	$9,82 \cdot 10^{-14}$	0,95881
43	FF	<i>purelin</i>	<i>logsig</i>	30	$1,60 \cdot 10^{-13}$	0,91350
44	FF	<i>purelin</i>	<i>logsig</i>	40	$5,94 \cdot 10^{-11}$	0,57099
45	FF	<i>purelin</i>	<i>logsig</i>	50	$6,02 \cdot 10^{-14}$	0,70322
46	FF	<i>purelin</i>	<i>tansig</i>	10	$9,49 \cdot 10^{-11}$	0,90337
47	FF	<i>purelin</i>	<i>tansig</i>	20	$9,75 \cdot 10^{-11}$	0,77369
48	FF	<i>purelin</i>	<i>tansig</i>	30	$1,49 \cdot 10^{-17}$	0,16907
49	FF	<i>purelin</i>	<i>tansig</i>	40	$2,99 \cdot 10^{-18}$	0,86412
50	FF	<i>purelin</i>	<i>tansig</i>	50	$5,58 \cdot 10^{-11}$	0,90292

Fonte: Autoria própria (2022)

Como as variáveis de saída tinham ordens de grandezas diferentes (E_a é na ordem de 10^2 e k_0 é na ordem de 10^5) o MSE com o X_VAL (conjunto separada de dados para validação) se apresentou tendendo somente para k_0 e por isso se apresentou inconclusivo e não foi indicado nessa tabela.

Pode-se perceber que com o aumento do número de neurônios o R² tem tendência de aumentar e depois diminuir, indicando que no começo um número maior de neurônios melhora a rede, mas quando ele fica muito grande a rede pode perder precisão.

Pode perceber também que na maioria dos testes o R² foi elevado, indicando que a rede consegue produzir um modelo adequado para os dados fornecidos. Para tentar contornar esse problema, foram trocadas as proporções do conjunto de dados. Até o momento estavam sendo utilizados 70% dos dados para treinamento, 15% para validação e 15% para teste.

Na Tabela 6 estão sendo apresentados os resultados onde foram utilizadas as proporções 75% para treinamento, 15% para validação e 10% para testes. Estes testes foram realizados para aumentar a proporção de treinamento e deixar a rede mais robusta, devido à maior quantidade de dados para o treinamento inicial.

Tabela 6 - Testes com maior proporção de treinamento

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	(continua)
						R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	$1,66 \cdot 10^{-6}$	0,62573
2	FF	<i>logsig</i>	<i>logsig</i>	20	$1,43 \cdot 10^{-13}$	0,88670

Tabela 6 - Testes com maior proporção de treinamento

(continuação)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
3	FF	<i>logsig</i>	<i>logsig</i>	30	1,40*10 ⁻¹⁶	0,87792
4	FF	<i>logsig</i>	<i>logsig</i>	40	8,21*10 ⁻¹³	0,91332
5	FF	<i>logsig</i>	<i>logsig</i>	50	1,58*10 ⁻⁹	0,73018
6	FF	<i>logsig</i>	<i>tansig</i>	10	1,16*10 ⁻⁶	0,86376
7	FF	<i>logsig</i>	<i>tansig</i>	20	4,68*10 ⁻¹⁶	0,96701
8	FF	<i>logsig</i>	<i>tansig</i>	30	3,86*10 ⁻¹³	0,93858
9	FF	<i>logsig</i>	<i>tansig</i>	40	1,37*10 ⁻¹²	0,66914
10	FF	<i>logsig</i>	<i>tansig</i>	50	2,76*10 ⁻¹²	0,92521
11	FF	<i>logsig</i>	<i>purelin</i>	10	1,05*10 ⁻¹¹	0,99957
12	FF	<i>logsig</i>	<i>purelin</i>	20	3,11*10 ⁻¹⁶	0,88911
13	FF	<i>logsig</i>	<i>purelin</i>	30	9,88*10 ⁻¹²	0,71736
14	FF	<i>logsig</i>	<i>purelin</i>	40	9,74*10 ⁻¹⁸	0,92436
15	FF	<i>logsig</i>	<i>purelin</i>	50	5,74*10 ⁻¹²	0,71514
16	FF	<i>logsig</i>	<i>satlin</i>	10	1,10*10 ⁻⁶	0,71369
17	FF	<i>logsig</i>	<i>satlin</i>	20	4,68*10 ⁻³	0,30252
18	FF	<i>logsig</i>	<i>satlin</i>	30	1,69*10 ⁻⁶	0,86458
19	FF	<i>logsig</i>	<i>satlin</i>	40	4,68*10 ⁻⁹	0,99286
20	FF	<i>logsig</i>	<i>satlin</i>	50	9,67*10 ⁻¹¹	0,95873
21	FF	<i>tansig</i>	<i>logsig</i>	10	8,31*10 ⁻¹¹	0,76124
22	FF	<i>tansig</i>	<i>logsig</i>	20	5,91*10 ⁻¹²	0,78731
23	FF	<i>tansig</i>	<i>logsig</i>	30	4,76*10 ⁻¹³	0,93502
24	FF	<i>tansig</i>	<i>logsig</i>	40	4,32*10 ⁻¹⁷	0,54412
25	FF	<i>tansig</i>	<i>logsig</i>	50	1,00*10 ⁻¹⁵	0,74026
26	FF	<i>tansig</i>	<i>tansig</i>	10	5,83*10 ⁻¹¹	0,94152
27	FF	<i>tansig</i>	<i>tansig</i>	20	2,09*10 ⁻¹⁴	0,88497
28	FF	<i>tansig</i>	<i>tansig</i>	30	1,42*10 ⁻¹⁴	0,93581
29	FF	<i>tansig</i>	<i>tansig</i>	40	1,66*10 ⁻¹⁶	0,69861
30	FF	<i>tansig</i>	<i>tansig</i>	50	3,75*10 ⁻¹⁷	0,92764
31	FF	<i>tansig</i>	<i>purelin</i>	10	6,55*10 ⁻¹⁰	0,88539
32	FF	<i>tansig</i>	<i>purelin</i>	20	1,14*10 ⁻¹⁶	0,21501
33	FF	<i>tansig</i>	<i>purelin</i>	30	1,08*10 ⁻¹⁸	0,29703
34	FF	<i>tansig</i>	<i>purelin</i>	40	1,66*10 ⁻¹¹	0,70203
35	FF	<i>tansig</i>	<i>purelin</i>	50	3,78*10 ⁻¹³	0,20503
36	FF	<i>tansig</i>	<i>satlin</i>	10	5,60*10 ⁻⁵	0,67994
37	FF	<i>tansig</i>	<i>satlin</i>	20	1,15*10 ⁻⁷	0,89340
38	FF	<i>tansig</i>	<i>satlin</i>	30	2,31*10 ⁻¹⁴	0,92803
39	FF	<i>tansig</i>	<i>satlin</i>	40	7,39*10 ⁻⁶	0,93174
40	FF	<i>tansig</i>	<i>satlin</i>	50	4,41*10 ⁻⁵	0,86907
41	FF	<i>purelin</i>	<i>logsig</i>	10	5,82*10 ⁻⁵	0,72343
42	FF	<i>purelin</i>	<i>logsig</i>	20	2,25*10 ⁻¹²	0,95702
43	FF	<i>purelin</i>	<i>logsig</i>	30	1,86*10 ⁻¹³	0,91488
44	FF	<i>purelin</i>	<i>logsig</i>	40	1,06*10 ⁻¹⁶	0,65927
45	FF	<i>purelin</i>	<i>logsig</i>	50	1,86*10 ⁻¹²	0,70190
46	FF	<i>purelin</i>	<i>tansig</i>	10	3,27*10 ⁻⁶	0,82909
47	FF	<i>purelin</i>	<i>tansig</i>	20	1,01*10 ⁻¹³	0,81199
48	FF	<i>purelin</i>	<i>tansig</i>	30	6,39*10 ⁻¹⁹	0,18567
49	FF	<i>purelin</i>	<i>tansig</i>	40	4,78*10 ⁻¹¹	0,86322

Tabela 6 - Testes com maior proporção de treinamento

(conclusão)						
Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
50	FF	<i>purelin</i>	<i>tansig</i>	50	$9,55 \cdot 10^{-12}$	0,87431

Fonte: Autoria própria (2022)

Já na Tabela 7 estão sendo apresentados os resultados das proporções de 65% para treinamento, 20% para validação e 15% para testes.

Tabela 7 - Testes com menor proporção de treinamento

(continua)						
Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	$1,81 \cdot 10^{-7}$	0,66610
2	FF	<i>logsig</i>	<i>logsig</i>	20	$2,90 \cdot 10^{-11}$	0,93062
3	FF	<i>logsig</i>	<i>logsig</i>	30	$3,20 \cdot 10^{-11}$	0,81509
4	FF	<i>logsig</i>	<i>logsig</i>	40	$7,66 \cdot 10^{-17}$	0,87687
5	FF	<i>logsig</i>	<i>logsig</i>	50	$5,30 \cdot 10^{-14}$	0,82918
6	FF	<i>logsig</i>	<i>tansig</i>	10	$7,40 \cdot 10^{-8}$	0,19832
7	FF	<i>logsig</i>	<i>tansig</i>	20	$1,30 \cdot 10^{-12}$	0,89069
8	FF	<i>logsig</i>	<i>tansig</i>	30	$7,42 \cdot 10^{-14}$	0,82304
9	FF	<i>logsig</i>	<i>tansig</i>	40	$9,61 \cdot 10^{-12}$	0,82994
10	FF	<i>logsig</i>	<i>tansig</i>	50	$1,06 \cdot 10^{-11}$	0,70493
11	FF	<i>logsig</i>	<i>purelin</i>	10	$9,46 \cdot 10^{-7}$	0,86653
12	FF	<i>logsig</i>	<i>purelin</i>	20	$2,43 \cdot 10^{-11}$	0,94432
13	FF	<i>logsig</i>	<i>purelin</i>	30	$2,88 \cdot 10^{-15}$	0,90641
14	FF	<i>logsig</i>	<i>purelin</i>	40	$7,08 \cdot 10^{-17}$	0,96932
15	FF	<i>logsig</i>	<i>purelin</i>	50	$7,20 \cdot 10^{-16}$	0,77753
16	FF	<i>logsig</i>	<i>satlin</i>	10	$1,59 \cdot 10^{-8}$	0,89813
17	FF	<i>logsig</i>	<i>satlin</i>	20	$9,97 \cdot 10^{-6}$	0,79197
18	FF	<i>logsig</i>	<i>satlin</i>	30	$2,08 \cdot 10^{-13}$	0,97060
19	FF	<i>logsig</i>	<i>satlin</i>	40	$4,05 \cdot 10^{-3}$	0,13333
20	FF	<i>logsig</i>	<i>satlin</i>	50	$3,05 \cdot 10^{-4}$	0,87255
21	FF	<i>tansig</i>	<i>logsig</i>	10	$6,43 \cdot 10^{-11}$	0,92280
22	FF	<i>tansig</i>	<i>logsig</i>	20	$5,26 \cdot 10^{-11}$	0,97092
23	FF	<i>tansig</i>	<i>logsig</i>	30	$4,36 \cdot 10^{-11}$	0,55528
24	FF	<i>tansig</i>	<i>logsig</i>	40	$6,35 \cdot 10^{-18}$	0,69499
25	FF	<i>tansig</i>	<i>logsig</i>	50	$4,87 \cdot 10^{-15}$	0,64709
26	FF	<i>tansig</i>	<i>tansig</i>	10	$2,65 \cdot 10^{-14}$	0,90173
27	FF	<i>tansig</i>	<i>tansig</i>	20	$1,13 \cdot 10^{-13}$	0,84819
28	FF	<i>tansig</i>	<i>tansig</i>	30	$7,73 \cdot 10^{-11}$	0,76531
29	FF	<i>tansig</i>	<i>tansig</i>	40	$8,60 \cdot 10^{-15}$	0,89024
30	FF	<i>tansig</i>	<i>tansig</i>	50	$1,18 \cdot 10^{-13}$	0,86348
31	FF	<i>tansig</i>	<i>purelin</i>	10	$8,47 \cdot 10^{-6}$	0,77318

Tabela 7 - Testes com menor proporção de treinamento

(conclusão)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
32	FF	<i>tansig</i>	<i>purelin</i>	20	3,04*10 ⁻¹⁸	0,72314
33	FF	<i>tansig</i>	<i>purelin</i>	30	1,63*10 ⁻¹⁵	0,69192
34	FF	<i>tansig</i>	<i>purelin</i>	40	5,83*10 ⁻¹³	0,45840
35	FF	<i>tansig</i>	<i>purelin</i>	50	4,60*10 ⁻¹⁵	0,49267
36	FF	<i>tansig</i>	<i>satlin</i>	10	1,86*10 ⁻³	0,91160
37	FF	<i>tansig</i>	<i>satlin</i>	20	1,49*10 ⁻¹⁸	0,70276
38	FF	<i>tansig</i>	<i>satlin</i>	30	2,39*10 ⁻¹⁹	0,86439
39	FF	<i>tansig</i>	<i>satlin</i>	40	2,94*10 ⁻¹⁷	0,87048
40	FF	<i>tansig</i>	<i>satlin</i>	50	2,18*10 ⁻¹¹	0,86199
41	FF	<i>purelin</i>	<i>logsig</i>	10	4,64*10 ⁻¹¹	0,98056
42	FF	<i>purelin</i>	<i>logsig</i>	20	1,58*10 ⁻¹¹	0,95462
43	FF	<i>purelin</i>	<i>logsig</i>	30	2,29*10 ⁻¹⁹	0,75856
44	FF	<i>purelin</i>	<i>logsig</i>	40	1,57*10 ⁻¹⁴	0,94783
45	FF	<i>purelin</i>	<i>logsig</i>	50	1,49*10 ⁻¹⁷	0,71502
46	FF	<i>purelin</i>	<i>tansig</i>	10	5,51*10 ⁻¹¹	0,80598
47	FF	<i>purelin</i>	<i>tansig</i>	20	6,44*10 ⁻¹⁸	0,74395
48	FF	<i>purelin</i>	<i>tansig</i>	30	5,03*10 ⁻¹⁴	0,83741
49	FF	<i>purelin</i>	<i>tansig</i>	40	3,27*10 ⁻¹⁵	0,75248
50	FF	<i>purelin</i>	<i>tansig</i>	50	2,96*10 ⁻¹⁵	0,42217

Fonte: Autoria própria (2022)

Após a realização dos testes com diferentes proporções, percebeu-se que os resultados da rede não mostraram melhora, e até mesmo demonstraram piora na maioria dos casos.

Foi utilizado o código otimizado no conjunto de dados completo, para analisar como o mesmo se comporta. Os resultados dos novos testes estão apresentados na Tabela 8.

Tabela 8 - Testes com código otimizado

(continua)

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	10	1,60*10 ⁻¹¹	1
2	FF	<i>logsig</i>	20	8,21*10 ⁻¹⁶	0,99949
3	FF	<i>logsig</i>	30	5,44*10 ⁻¹⁸	0,99967
4	FF	<i>logsig</i>	40	4,49*10 ⁻¹⁴	0,99729
5	FF	<i>logsig</i>	50	4,21*10 ⁻¹¹	0,99773
6	FF	<i>tansig</i>	10	2,28*10 ⁻¹⁴	1
7	FF	<i>tansig</i>	20	2,25*10 ⁻¹⁴	0,99970
8	FF	<i>tansig</i>	30	4,15*10 ⁻¹²	0,99745
9	FF	<i>tansig</i>	40	6,44*10 ⁻¹⁶	0,99719
10	FF	<i>tansig</i>	50	4,18*10 ⁻¹⁷	0,99485

Tabela 8 - Testes com código otimizado

(continuação)

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	R ² médio (rede)
11	FF	<i>purelin</i>	10	$1,20 \cdot 10^{-2}$	0,87071
12	FF	<i>purelin</i>	20	$1,20 \cdot 10^{-2}$	0,87214
13	FF	<i>purelin</i>	30	$1,20 \cdot 10^{-2}$	0,86918
14	FF	<i>purelin</i>	40	$1,20 \cdot 10^{-2}$	0,86982
15	FF	<i>purelin</i>	50	$1,20 \cdot 10^{-2}$	0,86982
16	FF	<i>satlin</i>	10	$2,32 \cdot 10^{-17}$	1
17	FF	<i>satlin</i>	20	$1,95 \cdot 10^{-12}$	1
18	FF	<i>satlin</i>	30	$4,08 \cdot 10^{-11}$	0,99999
19	FF	<i>satlin</i>	40	$7,53 \cdot 10^{-11}$	0,99988
20	FF	<i>satlin</i>	50	$1,15 \cdot 10^{-16}$	1
21	CF	<i>logsig</i>	10	$3,87 \cdot 10^{-11}$	0,99999
22	CF	<i>logsig</i>	20	$6,49 \cdot 10^{-16}$	0,99986
23	CF	<i>logsig</i>	30	$6,61 \cdot 10^{-16}$	0,99833
24	CF	<i>logsig</i>	40	$6,73 \cdot 10^{-11}$	0,99081
25	CF	<i>logsig</i>	50	$7,66 \cdot 10^{-11}$	0,99667
26	CF	<i>tansig</i>	10	$3,36 \cdot 10^{-11}$	1
27	CF	<i>tansig</i>	20	$2,00 \cdot 10^{-12}$	0,99932
28	CF	<i>tansig</i>	30	$9,53 \cdot 10^{-11}$	0,9924
29	CF	<i>tansig</i>	40	$8,75 \cdot 10^{-17}$	0,99818
30	CF	<i>tansig</i>	50	$1,33 \cdot 10^{-17}$	0,99502
31	CF	<i>purelin</i>	10	$1,20 \cdot 10^{-2}$	0,87022
32	CF	<i>purelin</i>	20	$1,20 \cdot 10^{-2}$	0,87119
33	CF	<i>purelin</i>	30	$1,20 \cdot 10^{-2}$	0,86863
34	CF	<i>purelin</i>	40	$1,20 \cdot 10^{-2}$	0,86512
35	CF	<i>purelin</i>	50	$1,20 \cdot 10^{-2}$	0,86512
36	CF	<i>satlin</i>	10	$1,30 \cdot 10^{-2}$	0,99999
37	CF	<i>satlin</i>	20	$1,02 \cdot 10^{-13}$	0,99999
38	CF	<i>satlin</i>	30	$1,35 \cdot 10^{-11}$	0,99987
39	CF	<i>satlin</i>	40	$1,75 \cdot 10^{-11}$	0,99886
40	CF	<i>satlin</i>	50	$1,89 \cdot 10^{-13}$	0,99991
41	EL	<i>logsig</i>	10	$1,90 \cdot 10^{-9}$	
42	EL	<i>logsig</i>	20	$9,00 \cdot 10^{-13}$	
43	EL	<i>logsig</i>	30	$2,40 \cdot 10^{-16}$	
44	EL	<i>logsig</i>	40	$7,04 \cdot 10^{-14}$	
45	EL	<i>logsig</i>	50	$3,76 \cdot 10^{-15}$	
46	EL	<i>tansig</i>	10	$6,48 \cdot 10^{-14}$	
47	EL	<i>tansig</i>	20	$2,37 \cdot 10^{-18}$	
48	EL	<i>tansig</i>	30	$3,96 \cdot 10^{-12}$	
49	EL	<i>tansig</i>	40	$1,87 \cdot 10^{-11}$	
50	EL	<i>tansig</i>	50	$1,29 \cdot 10^{-17}$	
51	EL	<i>purelin</i>	10	$1,11 \cdot 10^{-2}$	
52	EL	<i>purelin</i>	20	$1,32 \cdot 10^{-2}$	
53	EL	<i>purelin</i>	30	$1,32 \cdot 10^{-2}$	
54	EL	<i>purelin</i>	40	$1,43 \cdot 10^{-2}$	
55	EL	<i>purelin</i>	50	$1,28 \cdot 10^{-2}$	
56	EL	<i>satlin</i>	10	$8,54 \cdot 10^{-16}$	
57	EL	<i>satlin</i>	20	$2,53 \cdot 10^{-3}$	

Tabela 8 - Testes com código otimizado

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	R ² médio (rede)
58	EL	<i>satlin</i>	30	$9,62 \cdot 10^{-14}$	
59	EL	<i>satlin</i>	40	$1,19 \cdot 10^{-13}$	
60	EL	<i>satlin</i>	50	$1,08 \cdot 10^{-14}$	

(conclusão)

Fonte: Autoria própria (2022)

A rede de Elman não apresentou R², pois a existência da camada de contexto faz com que seja necessária uma alteração do modo como o MATLAB realiza testes na RNA e assim o *software* não consegue gerar os gráficos de R², como nas outras estruturas de rede.

Após analisar as tabelas, percebeu-se que o R² era alto e o MSE era baixo, mas isso só ocorria com os dados de treinamento. Para os dados de validação, o MSE era muito elevado, mostrando que a rede estava se adequando demais aos dados de treinamento e tendo problemas com os outros dados. Isso somado ao fato desse problema ser menos prevalente com 1 camada em vez de 2, demonstrou que essa rede estava ainda sofrendo de *overfitting*, mesmo após a adição dos dados de Tian *et al* (2021). A adição de mais dados amenizou esse problema, mas não o corrigiu.

Comparando o desempenho da RNA desse trabalho com outros presentes na literatura que abordam temas similares, esta rede não se demonstrou com resultados equivalentes. Muitos desses usaram quantidades menores de dados de entrada e obtiveram sucesso, como é o caso de Ozonoh *et al.* (2020) e Shahbaz *et al.* (2019). Isso dá indícios de que o alto número de variáveis tenha tornado o modelo muito complexo e isso foi um fator que levou ao *overfitting*.

O uso de uma base de dados mais robusta seria um fator que pode ser estudado em trabalhos futuros, assim como a grandeza do impacto do uso de 2 camadas ocultas em vez de 1. Como pode-se ver, os testes com 1 camada oculta geraram resultados bons e o uso de 2 camadas ocultas seria desnecessário, a menos que se utilizasse um conjunto muito grande de dados e a rede apresentasse tendência de *overfeeding* (TEKTO, LIVINGSTONE, LUIK, 1995).

As outras técnicas mais usuais para prevenir *overfitting* são a diminuição da complexidade do modelo e regulamentação de pesos, onde as variáveis com valores

de ordens maiores sofrem penalidades, para tentar equalizar o peso de todas as variáveis. Esse segundo método já é presente nesse trabalho, pois o código no mesmo normaliza as entradas antes de começar os testes da rede (GOYAL, 2021).

6 CONCLUSÃO

Os resultados dos testes finais apresentaram tendência de *overfitting* e essa tendência não foi completamente resolvida, logo, conclui-se que os resultados obtidos não foram totalmente satisfatórios.

Uma hipótese é que o conjunto de dados utilizado não foi robusto o suficiente para se conseguir resultados com a precisão desejada na rede e seria necessário um conjunto de dados maior para se obterem resultados mais próximos do desejado.

Outro fator que pode ter causado problemas é que não se utilizou todos os dados de entrada possíveis. Esse trabalho fez uso de 11 entradas pois havia receio que um número muito grande de entradas iria servir somente para confundir a rede e muitos dos trabalhos pesquisados como bases de dados não tinham todos os dados de entrada possíveis, então escolheu-se os 11 pois eles são os dados mais usuais de se encontrar na literatura.

Por fim, é possível que outros dados impactem de grande forma nas variáveis cinéticas, como por exemplo as porcentagens de elementos metálicos entre as cinzas, que podem servir de catalisadores para a gaseificação e muitas vezes os catalisadores são um dos fatores mais importantes na velocidade e rendimento de reações.

O estudo do impacto de cada variável de entrada e a necessidade do uso das mesmas pode ser feito em um trabalho futuro, que junto à esse pode ajudar a começar a desenvolver um modelo de análise de viabilidade de gaseificação de biomassa, sem a necessidade de estudos laboratoriais.

Para trabalhos futuros poderia ser utilizada essa base de dados e complementar a mesma, para acabar com a tendência de *overfitting*. Também pode ser feito um estudo para analisar o peso de cada variável no treinamento da rede e ver se realmente todas essas 11 variáveis são necessárias, ou se algumas delas podem ser descartadas, assim diminuindo a complexidade dos cálculos que a rede necessita realizar.

Embora o uso de RNA não tenha obtido sucesso nesse caso, essas redes podem ser uma ferramenta bastante útil nesta área e podem ajudar no estudo do fenômeno de gaseificação, diminuindo a complexidade do estudo desse processo,

assim estimulando até mesmo aqueles que não tem um conhecimento muito aprofundado do mesmo a iniciar estudos sobre gaseificação e assim trazendo mais relevância para setor.

REFERÊNCIAS

ÁVILA, I.; CRNKOVIC, P. M.; MILIOLI, F. E. **Determinação dos parâmetros de arrhenius da reação de sorção do dióxido de enxofre por calcário.** Química Nova, v. 30, n. 5, p. 1275–1281, 2007. SBQ.

BI, H. *et al.* **Thermodynamics, kinetics, gas emissions and artificial neural network modeling of co-pyrolysis of sewage sludge and peanut shell.** Fuel, v. 284, 2021. Elsevier Ltd.

BIOLOGY. **6.2 Potential, kinetic, free, and activation energy.** Openstax, 2016. Disponível em: <<https://openstax.org/books/biology/pages/6-2-potential-kinetic-free-and-activation-energy>>. Acesso em: 25 de Abril de 2022.

BRANCO, H. **Overfitting e underfitting em machine learning.** ABRACD, Ano Desconhecido. Disponível em: <<https://abracd.org/overfitting-e-underfitting-em-machine-learning/>>. Acesso em: 03 de Fevereiro de 2022.

CECCON, D. **Os tipos de redes neurais.** EXPERT ACADEMY, 2020. Disponível em: <<https://iaexpert.academy/2020/06/08/os-tipos-de-redes-neurais/>>. Acesso em: 12 de Março de 2022.

CHOJNACKI, J. *et al.* **Syngas composition: gasification of wood pellet with water steam through a reactor with continuous biomass feed system.** Energies 2020, 13(17), 4376, 2020.

COLLAZZO, G. C.. **Gaseificação de carvões nacionais com vapor d'água visando a produção de gás de síntese.** Florianópolis: UFSC, 2013.

COLLOT, A. G. **Matching gasification technologies to coal properties.** International Journal of Coal Geology, v. 65, n. 3–4, p. 191–212, 2006. Elsevier.

COLLOT, A. G. **Matching gasification technologies to coal properties.** Londres: IEA Clean, 2002.

CONSUEGRA, A. S. A. **Cinética de gaseificação do bagaço de laranja e serragem de madeira com CO₂.** Florianópolis: UFSC, 2013.

DAHMANI, K. *et al.* **Estimation of 5-min time-step data of tilted solar global irradiation using ANN (Artificial Neural Network) model.** Energy, v. 70, p. 374–381, 2014. Elsevier Ltd.

DAI, B.; HOADLEY, A.; ZHANG, L. **Characteristics of high temperature C-CO₂ gasification reactivity of Victorian brown coal char and its blends with high ash fusion temperature bituminous coal.** Fuel, v. 202, p. 352–365, 2017. Elsevier Ltd.

DOMENICO, M. D. **Gaseificação de carvão mineral brasileiro na presença de ortossilicato de lítio visando a produção aumentada de hidrogênio.** UFSC, 2013.

EPA. **Domestic and global usage of gasification technology.** Environmental Protection Agency, 2016. Disponível em: <<https://archive.epa.gov/epawaste/hazard/wastemin/web/html/gasdom.html>>. Acesso em: 23 de Maio de 2022.

FERRARA, F. *et al.* **Pyrolysis of coal, biomass and their blends: Performance assessment by thermogravimetric analysis.** Bioresource Technology, v. 171, p. 433–441, 2014. Elsevier Ltd.

FLORENCIO, P. H. B. VIEIRA, S. L. **Aplicação de redes neurais artificiais de elman na previsão de demanda de peças.** XXXVI ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO. João Pessoa, PB, 2016.

GOYAL, C. **Complete Guide to Prevent Overfitting in Neural Networks.** Analytics Vidhya, 2021. Disponível em: <<https://www.analyticsvidhya.com/blog/2021/06/complete-guide-to-prevent-overfitting-in-neural-networks-part-1/>>. Acesso em: 13 de Junho de 2022.

GUO, B. *et al.* **Simulation of biomass gasification with a hybrid neural network model.** Bioresource Technology, v. 76, n. 2, p. 77–83, 2001. Elsevier Science Ltd.

HAYKIN, S. **Redes neurais: princípios e práticas.** Tradução ENGEL, P.M. 2.ed. Porto Alegre: Bookman, 2001.

HELP CENTER. **logsig.** MathWorks, ano desconhecido. Disponível em: <<https://www.mathworks.com/help/deeplearning/ref/logsig.html>>. Acesso em: 13 de Março de 2022.

HELP CENTER. **purelin.** MathWorks, ano desconhecido. Disponível em: <<https://www.mathworks.com/help/deeplearning/ref/purelin.html>>. Acesso em: 13 de Março de 2022.

HELP CENTER. **satlin.** MathWorks, ano desconhecido. Disponível em: <<https://www.mathworks.com/help/deeplearning/ref/satlin.html>>. Acesso em: 13 de Março de 2022.

HELP CENTER. **tansig.** MathWorks, ano desconhecido. Disponível em: <<https://www.mathworks.com/help/deeplearning/ref/tansig.html>>. Acesso em: 13 de Março de 2022.

JI, G.; *et al.* **A comparative study on Arrhenius-type constitutive model and artificial neural network model to predict high-temperature deformation behaviour in Aermet100 steel.** Materials Science and Engineering A, v. 528, n. 13–14, p. 4774–4782, 2011. Elsevier.

KAMBLE, A. D. *et al.* **Co-gasification of coal and biomass an emerging clean energy technology: Status and prospects of development in Indian context.** International Journal of Mining Science and Technology, 2019. China University of Mining and Technology.

KASSABOEHRER, A. C. **SLC 0660 – química geral I.** USP, 2015. Disponível em: <https://edisciplinas.usp.br/pluginfile.php/4314494/mod_resource/content/1/Aula%2015-05.pdf>. Acesso em 22 de Abril de 2022.

KINTOL, O. T. *et al.* **Energia da gaseificação de biomassa como opção energética de desenvolvimento limpo.** SciELO, 2002. Disponível em: <http://www.proceedings.scielo.br/scielo.php?pid=MSC0000000022002000100061&script=sci_arttext>. Acesso em: 05 de Maio de 2021.

MATLAB M. **The language of technical computing.** The MathWorks, Inc, 2012. Disponível em: <<http://www.mathworks.com>>. Acesso em: 01 de Maio de 2021.

MINCHENER, A. J. **Coal gasification for advanced power generation. Fuel.** Anais. v. 84, p.2222–2235, 2005. Elsevier.

MOSESON, D. E. *et al.* **Application and limitations of thermogravimetric analysis to delineate the hot melt extrusion chemical stability processing window.** International Journal of Pharmaceutics, 2020. Elsevier B.V.

NAQVI, S. R. *et al.* **Pyrolysis of high-ash sewage sludge: Thermo-kinetic study using TGA and artificial neural networks.** Fuel, v. 233, p. 529–538, 2018. Elsevier Ltd.

OLIVEIRA, P. A. **Figura 12-Gaseificador-de-Fluxo-Arrastado.** Research Gate, 2016. Disponível em: <https://www.researchgate.net/figure/Figura-12-Gaseificador-de-Fluxo-Arrastado-Fonte-Adaptado-de-MCKENDRY-2001-BIOFUELS_fig9_305319084>. Acesso em: 05 de Maio de 2021.

OLIVEIRA, J. **Redes neurais artificiais.** PET News, UFGC, 2011. Disponível em: <<http://www.dsc.ufcg.edu.br/~pet/jornal/setembro2011/materias/informatica.html>>. Acesso em: 28 de Janeiro de 2022.

OZONOH, M. *et al.* **Performance evaluation of gasification system efficiency using artificial neural network.** Renewable Energy, 2020.

PACIONI, T. R. **Gaseificação de biochars de bagaço de maçã e de borra de café com CO₂: estudo cinético.** Florianópolis: UFSC, 2013.

REY, J. R. C.; TUNA, C. E.; SILVEIRA, J. L. **Figura 16-Gaseificadores-de-leito-fluidizado-borbulhante.** Research Gate, 2018. Disponível em: <https://www.researchgate.net/figure/Figura-16-Gaseificadores-de-leito-fluidizado-borbulhante_fig3_327755817>. Acesso em: 05 de Maio de 2021.

SHAHBAZ, M. *et al.* **Artificial neural network approach for the steam gasification of palm oil waste using bottom ash and CaO.** *Renewable Energy*, v. 132, p. 243–254, 2019. Elsevier Ltd.

SEO, D. K. *et al.* **Gasification reactivity of biomass chars with CO₂.** *Biomass and Bioenergy*, v.34, p. 1946-1953, 2010. Elsevier Ltd.

TAGHAVIFAR, H.; MARDANI, A. **Applying a supervised ANN (artificial neural network) approach to the prognostication of driven wheel energy efficiency indices.** *Energy*, v. 68, p. 651–657, 2014. Elsevier Ltd.

TIAN, H. *et al.* **Kinetic study on the CO₂ gasification of biochar derived from Miscanthus at different processing conditions.** *Energy*, v. 217, 119341, 2021. Elsevier Ltd.

TEKTO, I. V.; LIVINGSTONE, D. J.; LUIK, A. I. **Neural network studies. 1. comparison of overfitting and overtraining.** *J. Chem. Inf. Comput. Sci.*, v. 35, p. 826-833, 1995.

VINICIUS. **Redes neurais artificiais.** Monolito Nimbus, 2017. Disponível em: <<https://www.monolitonimbus.com.br/redes-neurais-artificiais/>>. Acesso em: 28 de Abril de 2021.

WARSITO, B. *et al.* **Cascade forward neural network for time series prediction.** *J. Phys.: Conf. Ser.* 1025 012097, 2018.

ZUO, H. *et al.* **Isothermal CO₂ gasification reactivity and kinetic models of biomass char/anthracite char.** *BioResources*, p. 5242-5255, 2015.

APÊNDICE A – CÓDIGO ORIGINAL

```

% ----- Garantindo que o programa está limpo antes de começar ---- %
clear;
clc;
% ----- Carregando os valores de entrada e saída ----- %

load X.mat;
load Y.mat;
load X_VAL.mat
load Y_VAL.mat

% ----- Normalizando os valores ----- %

[P,ps1] = mapminmax(X,0.1,0.9); %X
[T,ts1] = mapminmax(Y,0.1,0.9); %Y
[trainV,val,test] = dividevec(P,T,0.15,0.15);

% ----- A próxima parte do código deve ser escolhida de acordo com a
estrutura da rede desejada, se for rede Feed-Forward, Elman ou Cascade-
Forward, respectivamente----- %

net0 = newff(P,T,[10 10],{'logsig' 'logsig'},'trainlm','learngdm');
net0 = newelm(P,T,[10 10],{'logsig','logsig'},'trainlm');
net0 = newcfc(P,T,[10 10],{'logsig','logsig'},'trainlm')

% ----- E depois o código continua normalmente-----%

% ----- Treinando a rede-----%

net0.trainParam.epochs = 8000;
net0.trainParam.goal = 1e-10;
net0.trainParam.show = 10; % Tempos de visualização
net0.trainParam.max_fail = 6; % Maximum validation failures
net0.trainParam.mem_reduc = 1; % Factor to use for memory/speed tradeoff
net0.trainParam.min_grad = 1e-10; % Minimum performance gradient

```

```
net0.trainParam.mu = 0.001; % Initial mu
net0.trainParam.mu_dec = 0.1; % mu decrease factor
net0.trainParam.mu_inc = 10; % mu increase factor
net0.trainParam.mu_max = 1e10; % Maximum mu
net0.trainParam.show = 2; % Epochs between displays
net0.trainParam.showCommandLine = 0; % Generate command-line output
net0.trainParam.showWindow = 1; % Show training GUI
net0.trainParam.time = inf;
[net0,tr0] = train(net0,trainV.P,trainV.T,[],[],val,test);
```

APÊNDICE B – CÓDIGO OTIMIZADO

```
% ----- Garantindo que o programa está limpo antes de começar ---- %
```

```
clear;
```

```
clc;
```

```
% ----- Carregando os valores de entrada e saída ----- %
```

```
load X2.mat;
```

```
load Y2.mat;
```

```
% ----- Normalizando os valores ----- %
```

```
[P,ps1] = mapminmax(X2,0.1,0.9); %X
```

```
[T,ts1] = mapminmax(Y2,0.1,0.9); %Y
```

```
[trainV,val,test] = dividevec(P,T,0.15,0.15);
```

```
% ----- A próxima parte do código deve ser escolhida de acordo com a  
estrutura da rede desejada, se for rede Feed-Forward, Elman ou Cascade-  
Forward, respectivamente----- %
```

```
net0 = newff(P,T,[10],{'logsig'},'trainlm','learngdm');
```

```
net0 = newelm(P,T,[10],{'logsig'},'trainlm');
```

```
net0 = newcf(P,T,[10],{'logsig'},'trainlm')
```

```
% ----- E depois o código continua normalmente-----%
```

```
% ----- Treinando a rede-----%
```

```
net0.trainParam.epochs = 8000;
```

```
net0.trainParam.goal = 1e-10;
```

```
net0.trainParam.show = 10; % Tempos de visualização
```

```
net0.trainParam.max_fail = 6; % Maximum validation failures
```

```
net0.trainParam.mem_reduc = 1; % Factor to use for memory/speed tradeoff
```

```
net0.trainParam.min_grad = 1e-10; % Minimum performance gradient
```

```
net0.trainParam.mu = 0.001; % Initial mu
net0.trainParam.mu_dec = 0.1; % mu decrease factor
net0.trainParam.mu_inc = 10; % mu increase factor
net0.trainParam.mu_max = 1e10; % Maximum mu
net0.trainParam.show = 2; % Epochs between displays
net0.trainParam.showCommandLine = 0; % Generate command-line output
net0.trainParam.showWindow = 1; % Show training GUI
net0.trainParam.time = inf;
[net0,tr0] = train(net0,trainV.P,trainV.T,[],[],val,test);
```


APÊNDICE C – DADOS RETIRADOS DA LITERATURA

Tabela 9 - Dados retirados da literatura

(continua)

Dados de entrada	Consuegra (2013)		Pacioni (2013)			Zuo <i>et al.</i> (2015)				Seo <i>et al.</i> (2010)	Tian <i>et al.</i> (2021)		
	Bagaço de laranja	Serragem de madeira	Bagaço de maçã-PL	Bagaço de maçã-PR	Borra de café-PL	Serragem de pinho	Palha de trigo	Espiga de milho	Casca de amendoim	Pinheiro	Miscanthus	Miscanthus	Miscanthus
C(%)	46,1	53,65	52,2	52,2	55,68	50,95	47,69	47,75	47,97	49,35	49,14	49,14	49,14
H(%)	5,54	5,94	7,79	7,79	7,52	5,94	6,08	3,9	5,61	5,95	6,89	6,89	6,89
N(%)	2,13	0,74	1,71	1,71	2,37	0,39	0,31	0,48	0,49	0	0,26	0,26	0,26
S(%)	0	0,06	0	0	0,15	0,06	0,21	0,15	0,36	0	0,1	0,1	0,1
O (%)	40,14	38,72	36,51	36,51	30,61	42,18	40,34	45,63	41,38	44,2	40,1	40,1	40,1
CZ(%)	6,09	0,89	1,79	1,79	3,67	0,48	5,37	2,09	4,19	0,5	3,51	3,51	3,51
MV(%)	72,15	83,22	80,55	80,55	75,51	82,23	75,72	79,04	63,38	74,43	74,84	74,84	74,84
CF (%)	21,76	15,89	17,66	17,66	20,82	17,29	18,91	18,87	32,43	25,07	21,65	21,65	21,65
Tpir (C)	900	900	600	900	600	1100	1100	1100	1100	850	600	800	1000
tR (min)	15	15	60	18	60	60	60	60	60	17	120	120	120
Tgas (C)	730,780,825	850,900,925	760,810 e 855	760,810 e 855	760,810 e 855	850,900,950,1000	850,900,950,1000	850,900,950,1000	850,900,950,1000	850,900,1000,1050	800, 900, 1000	800, 900, 1000	800, 900, 1000

Tabela 9 - Dados retirados da literatura

(conclusão)

Consuegra (2013)		Pacioni (2013)			Zuo et al. (2015)			Seo et al. (2010)			Tian et al. (2021)		
Dados de saída	Bagaço de laranja	Serragem de madeira	Bagaço de maçã-PL	Bagaço de maçã-PR	Borra de café-PL	Serragem de pinho	Palha de trigo	Espiga de milho	Casca de amendoim	Pinheiro	Miscanthus	Miscanthus	Miscanthus
E_a (KJ/mol)	143,3	180,7	170,5	153,8	190,4	152,4	150,7	147,5	142,6	134	119,9	137,73	204,86
k_0 (min^{-1})	$3,366 \cdot 10^5$	$82,8 \cdot 10^5$	$12,0 \cdot 10^5$	$1,8 \cdot 10^5$	$223 \cdot 10^5$	$1,572 \cdot 10^5$	$4,182 \cdot 10^5$	$3,498 \cdot 10^5$	$2,352 \cdot 10^5$	$0,15 \cdot 10^5$	$0,12 \cdot 10^5$	$0,54 \cdot 10^5$	$210 \cdot 10^5$

Fonte: Autoria própria (2022)

APÊNDICE D – TABELAS COM TESTES COM GRUPO DE DADOS INICIAL

Tabela 10 - Testes com 1 saída

(continua)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)	MSE (validação)
1	FF	<i>logsig</i>	<i>logsig</i>	10	2,43*10 ⁻⁸	0,97066	338,605
2	FF	<i>logsig</i>	<i>logsig</i>	20	7,73*10 ⁻¹⁷	0,99785	397,352
3	FF	<i>logsig</i>	<i>logsig</i>	30	7,25*10 ⁻²⁰	0,98832	327,307
4	FF	<i>logsig</i>	<i>logsig</i>	40	3,60*10 ⁻²⁴	0,81211	286,537
5	FF	<i>logsig</i>	<i>logsig</i>	50	5,90*10 ⁻¹⁹	0,79963	206,935
6	FF	<i>logsig</i>	<i>tansig</i>	10	1,51*10 ⁻²¹	0,80964	226,417
7	FF	<i>logsig</i>	<i>tansig</i>	20	1,40*10 ⁻²⁴	0,8586	236,785
8	FF	<i>logsig</i>	<i>tansig</i>	30	7,61*10 ⁻¹⁷	0,98958	362,555
9	FF	<i>logsig</i>	<i>tansig</i>	40	1,03*10 ⁻²²	0,96456	333,157
10	FF	<i>logsig</i>	<i>tansig</i>	50	1,57*10 ⁻²⁴	0,76142	162,998
11	FF	<i>logsig</i>	<i>purelin</i>	10	3,57*10 ⁻²⁰	0,82494	25,565
12	FF	<i>logsig</i>	<i>purelin</i>	20	2,43*10 ⁻²⁰	0,98771	332,255
13	FF	<i>logsig</i>	<i>purelin</i>	30	1,69*10 ⁻¹⁸	0,8442	222,341
14	FF	<i>logsig</i>	<i>purelin</i>	40	5,56*10 ⁻²¹	0,35271	18,219
15	FF	<i>logsig</i>	<i>purelin</i>	50	3,36*10 ⁻¹⁸	0,70923	238,503
16	FF	<i>logsig</i>	<i>satlin</i>	10	2,43*10 ⁻⁵	0,77344	299,444
17	FF	<i>logsig</i>	<i>satlin</i>	20	1,26*10 ⁻⁷	0,95743	319,365
18	FF	<i>logsig</i>	<i>satlin</i>	30	1,00*10 ⁻²⁰	0,95769	313,809
19	FF	<i>logsig</i>	<i>satlin</i>	40	7,07*10 ⁻¹⁷	0,86573	266,442
20	FF	<i>logsig</i>	<i>satlin</i>	50	8,27*10 ⁻²¹	0,97979	313,658
21	FF	<i>tansig</i>	<i>logsig</i>	10	7,57*10 ⁻¹³	0,99241	382,894
22	FF	<i>tansig</i>	<i>logsig</i>	20	3,25*10 ⁻²⁶	0,98794	367,867
23	FF	<i>tansig</i>	<i>logsig</i>	30	1,35*10 ⁻¹⁸	0,99075	33,764
24	FF	<i>tansig</i>	<i>logsig</i>	40	4,78*10 ⁻¹⁶	0,94802	328,077
25	FF	<i>tansig</i>	<i>logsig</i>	50	2,49*10 ⁻¹⁸	0,78731	227,305
26	FF	<i>tansig</i>	<i>tansig</i>	10	1,18*10 ⁻²¹	0,97812	33,026
27	FF	<i>tansig</i>	<i>tansig</i>	20	6,24*10 ⁻²³	0,57168	22,578
28	FF	<i>tansig</i>	<i>tansig</i>	30	1,84*10 ⁻¹⁸	0,5873	203,789
29	FF	<i>tansig</i>	<i>tansig</i>	40	3,26*10 ⁻²¹	0,18604	145,662
30	FF	<i>tansig</i>	<i>tansig</i>	50	1,37*10 ⁻¹⁸	0,84345	242,767
31	FF	<i>tansig</i>	<i>purelin</i>	10	4,45*10 ⁻⁹	0,60366	208,647
32	FF	<i>tansig</i>	<i>purelin</i>	20	9,83*10 ⁻²⁰	0,91607	291,564
33	FF	<i>tansig</i>	<i>purelin</i>	30	2,21*10 ⁻¹⁸	0,04268	129,279
34	FF	<i>tansig</i>	<i>purelin</i>	40	9,85*10 ⁻²¹	0,38071	95,153
35	FF	<i>tansig</i>	<i>purelin</i>	50	1,53*10 ⁻²¹	0,54849	93,736

Tabela 10 - Testes com 1 saída

(conclusão)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)	MSE (validação)
36	FF	<i>tansig</i>	<i>satlin</i>	10	$8,64 \cdot 10^{-6}$	0,59834	269,321
37	FF	<i>tansig</i>	<i>satlin</i>	20	$1,46 \cdot 10^{-7}$	0,98948	387,546
38	FF	<i>tansig</i>	<i>satlin</i>	30	$7,32 \cdot 10^{-21}$	0,95813	309,819
39	FF	<i>tansig</i>	<i>satlin</i>	40	$3,73 \cdot 10^{-32}$	0,9323	283,032
40	FF	<i>tansig</i>	<i>satlin</i>	50	$4,13 \cdot 10^{-8}$	0,9958	413,295
41	FF	<i>purelin</i>	<i>logsig</i>	10	$1,02 \cdot 10^{-4}$	0,97836	332,516
42	FF	<i>purelin</i>	<i>logsig</i>	20	$5,50 \cdot 10^{-16}$	0,64135	211,026
43	FF	<i>purelin</i>	<i>logsig</i>	30	$1,58 \cdot 10^{-8}$	0,64327	209,021
44	FF	<i>purelin</i>	<i>logsig</i>	40	$4,29 \cdot 10^{-16}$	0,91194	238,543
45	FF	<i>purelin</i>	<i>logsig</i>	50	$6,20 \cdot 10^{-22}$	0,71954	222,585
46	FF	<i>purelin</i>	<i>tansig</i>	10	$3,87 \cdot 10^{-13}$	0,81668	244,544
47	FF	<i>purelin</i>	<i>tansig</i>	20	$1,99 \cdot 10^{-16}$	0,58842	204,786
48	FF	<i>purelin</i>	<i>tansig</i>	30	$5,97 \cdot 10^{-19}$	0,21368	140,887
49	FF	<i>purelin</i>	<i>tansig</i>	40	$1,18 \cdot 10^{-16}$	0,51174	104,532
50	FF	<i>purelin</i>	<i>tansig</i>	50	$2,67 \cdot 10^{-19}$	0,23555	151,768

Fonte: Autoria própria (2022)

Tabela 11 - Testes com 2 saídas

(continua)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	$4,00 \cdot 10^{-8}$	0,98865
2	FF	<i>logsig</i>	<i>logsig</i>	20	$1,03 \cdot 10^{-13}$	0,97899
3	FF	<i>logsig</i>	<i>logsig</i>	30	$3,96 \cdot 10^{-11}$	0,97139
4	FF	<i>logsig</i>	<i>logsig</i>	40	$3,72 \cdot 10^{-15}$	0,95033
5	FF	<i>logsig</i>	<i>logsig</i>	50	$7,51 \cdot 10^{-14}$	0,99092
6	FF	<i>logsig</i>	<i>tansig</i>	10	$6,00 \cdot 10^{-15}$	0,9358
7	FF	<i>logsig</i>	<i>tansig</i>	20	$5,16 \cdot 10^{-11}$	0,95641
8	FF	<i>logsig</i>	<i>tansig</i>	30	$2,40 \cdot 10^{-11}$	0,91578
9	FF	<i>logsig</i>	<i>tansig</i>	40	$1,53 \cdot 10^{-18}$	0,85839
10	FF	<i>logsig</i>	<i>tansig</i>	50	$3,65 \cdot 10^{-12}$	0,9704
11	FF	<i>logsig</i>	<i>purelin</i>	10	$4,97 \cdot 10^{-8}$	0,80948
12	FF	<i>logsig</i>	<i>purelin</i>	20	$4,89 \cdot 10^{-5}$	0,9592
13	FF	<i>logsig</i>	<i>purelin</i>	30	$9,27 \cdot 10^{-11}$	0,66142

Tabela 11 - Testes com 2 saídas

(continuação)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
14	FF	<i>logsig</i>	<i>purelin</i>	40	1,03*10 ⁻¹²	0,62545
15	FF	<i>logsig</i>	<i>purelin</i>	50	3,65*10 ⁻¹⁸	0,67166
16	FF	<i>logsig</i>	<i>satlin</i>	10	1,24*10 ⁻¹³	0,99725
17	FF	<i>logsig</i>	<i>satlin</i>	20	5,39*10 ⁻¹³	0,96689
18	FF	<i>logsig</i>	<i>satlin</i>	30	1,13*10 ⁻⁸	0,98273
19	FF	<i>logsig</i>	<i>satlin</i>	40	2,83*10 ⁻⁷	0,97718
20	FF	<i>logsig</i>	<i>satlin</i>	50	7,78*10 ⁻¹¹	0,97169
21	FF	<i>tansig</i>	<i>logsig</i>	10	1,06*10 ⁻¹³	0,99801
22	FF	<i>tansig</i>	<i>logsig</i>	20	7,97*10 ⁻¹⁷	0,99137
23	FF	<i>tansig</i>	<i>logsig</i>	30	4,40*10 ⁻¹³	0,99445
24	FF	<i>tansig</i>	<i>logsig</i>	40	1,89*10 ⁻¹⁵	0,99566
25	FF	<i>tansig</i>	<i>logsig</i>	50	4,72*10 ⁻¹⁴	0,90626
26	FF	<i>tansig</i>	<i>tansig</i>	10	5,90*10 ⁻⁷	0,87689
27	FF	<i>tansig</i>	<i>tansig</i>	20	1,34*10 ⁻¹⁸	0,92139
28	FF	<i>tansig</i>	<i>tansig</i>	30	2,91*10 ⁻¹²	0,94437
29	FF	<i>tansig</i>	<i>tansig</i>	40	5,19*10 ⁻¹¹	0,77096
30	FF	<i>tansig</i>	<i>tansig</i>	50	2,06*10 ⁻¹³	0,80296
31	FF	<i>tansig</i>	<i>purelin</i>	10	6,91*10 ⁻⁷	0,99628
32	FF	<i>tansig</i>	<i>purelin</i>	20	2,69*10 ⁻¹³	0,95478
33	FF	<i>tansig</i>	<i>purelin</i>	30	8,12*10 ⁻¹¹	0,80262
34	FF	<i>tansig</i>	<i>purelin</i>	40	1,01*10 ⁻¹²	0,76829
35	FF	<i>tansig</i>	<i>purelin</i>	50	1,81*10 ⁻¹⁴	0,22149
36	FF	<i>tansig</i>	<i>satlin</i>	10	6,73*10 ⁻¹⁴	0,97687
37	FF	<i>tansig</i>	<i>satlin</i>	20	2,12*10 ⁻¹²	0,93651
38	FF	<i>tansig</i>	<i>satlin</i>	30	8,74*10 ⁻⁸	0,9799
39	FF	<i>tansig</i>	<i>satlin</i>	40	8,71*10 ⁻¹¹	0,99
40	FF	<i>tansig</i>	<i>satlin</i>	50	1,72*10 ⁻¹¹	0,95725
41	FF	<i>purelin</i>	<i>logsig</i>	10	2,29*10 ⁻¹¹	0,98524
42	FF	<i>purelin</i>	<i>logsig</i>	20	1,23*10 ⁻¹¹	0,98025
43	FF	<i>purelin</i>	<i>logsig</i>	30	1,88*10 ⁻¹¹	0,77
44	FF	<i>purelin</i>	<i>logsig</i>	40	4,45*10 ⁻¹⁵	0,81383
45	FF	<i>purelin</i>	<i>logsig</i>	50	1,58*10 ⁻¹³	0,96498
46	FF	<i>purelin</i>	<i>tansig</i>	10	6,43*10 ⁻¹⁰	0,88976
47	FF	<i>purelin</i>	<i>tansig</i>	20	4,49*10 ⁻¹⁴	0,80026
48	FF	<i>purelin</i>	<i>tansig</i>	30	5,27*10 ⁻¹⁹	0,87836
49	FF	<i>purelin</i>	<i>tansig</i>	40	2,97*10 ⁻¹⁵	0,84319

Tabela 11 - Testes com 2 saídas

(continuação)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
50	FF	<i>purelin</i>	<i>tansig</i>	50	1,61*10 ⁻¹²	0,86064
51	EL	<i>logsig</i>	<i>logsig</i>	10	6,33*10 ⁻⁸	
52	EL	<i>logsig</i>	<i>logsig</i>	20	4,72*10 ⁻¹³	
53	EL	<i>logsig</i>	<i>logsig</i>	30	1,47*10 ⁻¹²	
54	EL	<i>logsig</i>	<i>logsig</i>	40	1,16*10 ⁻¹²	
55	EL	<i>logsig</i>	<i>logsig</i>	50	3,62*10 ⁻⁹	
56	EL	<i>logsig</i>	<i>tansig</i>	10	1,62*10 ⁻¹⁴	
57	EL	<i>logsig</i>	<i>tansig</i>	20	1,03*10 ⁻¹¹	
58	EL	<i>logsig</i>	<i>tansig</i>	30	3,80*10 ⁻¹¹	
59	EL	<i>logsig</i>	<i>tansig</i>	40	1,29*10 ⁻¹⁶	
60	EL	<i>logsig</i>	<i>tansig</i>	50	2,23*10 ⁻⁷	
61	EL	<i>logsig</i>	<i>purelin</i>	10	5,14*10 ⁻¹⁴	
62	EL	<i>logsig</i>	<i>purelin</i>	20	2,17*10 ⁻¹¹	
63	EL	<i>logsig</i>	<i>purelin</i>	30	5,56*10 ⁻¹⁴	
64	EL	<i>logsig</i>	<i>purelin</i>	40	2,06*10 ⁻¹⁵	
65	EL	<i>logsig</i>	<i>purelin</i>	50	9,97*10 ⁻¹²	
66	EL	<i>logsig</i>	<i>satlin</i>	10	4,48*10 ⁻⁶	
67	EL	<i>logsig</i>	<i>satlin</i>	20	4,86*10 ⁻¹¹	
68	EL	<i>logsig</i>	<i>satlin</i>	30	1,61*10 ⁻¹¹	
69	EL	<i>logsig</i>	<i>satlin</i>	40	3,68*10 ⁻⁹	
70	EL	<i>logsig</i>	<i>satlin</i>	50	1,94*10 ⁻¹¹	
71	EL	<i>tansig</i>	<i>logsig</i>	10	2,56*10 ⁻⁹	
72	EL	<i>tansig</i>	<i>logsig</i>	20	1,65*10 ⁻¹⁵	
73	EL	<i>tansig</i>	<i>logsig</i>	30	1,34*10 ⁻²⁰	
74	EL	<i>tansig</i>	<i>logsig</i>	40	1,59*10 ⁻⁹	
75	EL	<i>tansig</i>	<i>logsig</i>	50	1,31*10 ⁻¹⁴	
76	EL	<i>tansig</i>	<i>tansig</i>	10	6,09*10 ⁻¹¹	
77	EL	<i>tansig</i>	<i>tansig</i>	20	9,11*10 ⁻¹¹	
78	EL	<i>tansig</i>	<i>tansig</i>	30	8,32*10 ⁻¹³	
79	EL	<i>tansig</i>	<i>tansig</i>	40	1,39*10 ⁻¹⁶	
80	EL	<i>tansig</i>	<i>tansig</i>	50	1,73*10 ⁻¹¹	
81	EL	<i>tansig</i>	<i>purelin</i>	10	2,95*10 ⁻⁹	
82	EL	<i>tansig</i>	<i>purelin</i>	20	1,00*10 ⁻¹¹	
83	EL	<i>tansig</i>	<i>purelin</i>	30	6,50*10 ⁻¹²	
84	EL	<i>tansig</i>	<i>purelin</i>	40	1,64*10 ⁻¹³	
85	EL	<i>tansig</i>	<i>purelin</i>	50	2,04*10 ⁻¹³	

Tabela 11 - Testes com 2 saídas

(continuação)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
86	EL	<i>tansig</i>	<i>satlin</i>	10	1,76*10 ⁻¹⁹	
87	EL	<i>tansig</i>	<i>satlin</i>	20	7,48*10 ⁻⁹	
88	EL	<i>tansig</i>	<i>satlin</i>	30	2,32*10 ⁻¹³	
89	EL	<i>tansig</i>	<i>satlin</i>	40	4,77*10 ⁻¹²	
90	EL	<i>tansig</i>	<i>satlin</i>	50	1,23*10 ⁻¹⁵	
91	EL	<i>purelin</i>	<i>logsig</i>	10	1,77*10 ⁻¹²	
92	EL	<i>purelin</i>	<i>logsig</i>	20	1,70*10 ⁻¹⁴	
93	EL	<i>purelin</i>	<i>logsig</i>	30	3,72*10 ⁻¹⁶	
94	EL	<i>purelin</i>	<i>logsig</i>	40	2,09*10 ⁻¹⁶	
95	EL	<i>purelin</i>	<i>logsig</i>	50	3,00*10 ⁻¹²	
96	EL	<i>purelin</i>	<i>tansig</i>	10	4,62*10 ⁻¹⁵	
97	EL	<i>purelin</i>	<i>tansig</i>	20	5,30*10 ⁻¹⁸	
98	EL	<i>purelin</i>	<i>tansig</i>	30	3,08*10 ⁻¹⁴	
99	EL	<i>purelin</i>	<i>tansig</i>	40	2,63*10 ⁻¹⁷	
100	EL	<i>purelin</i>	<i>tansig</i>	50	4,66*10 ⁻¹⁷	
101	CF	<i>logsig</i>	<i>logsig</i>	10	4,98*10 ⁻¹³	0,87847
102	CF	<i>logsig</i>	<i>logsig</i>	20	3,28*10 ⁻¹²	0,97249
103	CF	<i>logsig</i>	<i>logsig</i>	30	8,46*10 ⁻¹²	0,78447
104	CF	<i>logsig</i>	<i>logsig</i>	40	3,55*10 ⁻¹⁵	0,2589
105	CF	<i>logsig</i>	<i>logsig</i>	50	2,39*10 ⁻¹⁶	0,73795
106	CF	<i>logsig</i>	<i>tansig</i>	10	1,22*10 ⁻¹¹	0,83061
107	CF	<i>logsig</i>	<i>tansig</i>	20	1,42*10 ⁻¹⁴	0,66319
108	CF	<i>logsig</i>	<i>tansig</i>	30	4,52*10 ⁻¹⁷	0,99386
109	CF	<i>logsig</i>	<i>tansig</i>	40	3,06*10 ⁻¹³	0,30667
110	CF	<i>logsig</i>	<i>tansig</i>	50	5,08*10 ⁻¹³	0,42668
111	CF	<i>logsig</i>	<i>purelin</i>	10	4,66*10 ⁻¹⁷	0,84953
112	CF	<i>logsig</i>	<i>purelin</i>	20	4,09*10 ⁻⁶	0,84386
113	CF	<i>logsig</i>	<i>purelin</i>	30	5,65*10 ⁻¹⁵	0,75568
114	CF	<i>logsig</i>	<i>purelin</i>	40	1,39*10 ⁻⁸	0,98414
115	CF	<i>logsig</i>	<i>purelin</i>	50	1,06*10 ⁻¹⁴	0,94105
116	CF	<i>logsig</i>	<i>satlin</i>	10	5,72*10 ⁻¹⁵	0,99135
117	CF	<i>logsig</i>	<i>satlin</i>	20	4,87*10 ⁻³	0,93539
118	CF	<i>logsig</i>	<i>satlin</i>	30	3,68*10 ⁻¹⁷	0,0834
119	CF	<i>logsig</i>	<i>satlin</i>	40	1,26*10 ⁻¹⁵	0,96918
120	CF	<i>logsig</i>	<i>satlin</i>	50	7,77*10 ⁻¹¹	0,10556
121	CF	<i>tansig</i>	<i>logsig</i>	10	1,76*10 ⁻¹⁶	0,73912

Tabela 11 - Testes com 2 saídas

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	(conclusão)
						R ² médio (rede)
122	CF	<i>tansig</i>	<i>logsig</i>	20	4,84*10 ⁻⁸	0,64294
123	CF	<i>tansig</i>	<i>logsig</i>	30	9,46*10 ⁻¹⁴	0,51402
124	CF	<i>tansig</i>	<i>logsig</i>	40	4,26*10 ⁻¹¹	0,65749
125	CF	<i>tansig</i>	<i>logsig</i>	50	3,41*10 ⁻¹⁵	0,80989
126	CF	<i>tansig</i>	<i>tansig</i>	10	1,65*10 ⁻¹²	-0,0117
127	CF	<i>tansig</i>	<i>tansig</i>	20	1,01*10 ⁻¹⁴	0,81001
128	CF	<i>tansig</i>	<i>tansig</i>	30	3,10*10 ⁻¹²	0,83946
129	CF	<i>tansig</i>	<i>tansig</i>	40	4,98*10 ⁻¹⁵	0,86077
130	CF	<i>tansig</i>	<i>tansig</i>	50	6,93*10 ⁻¹³	0,72957
131	CF	<i>tansig</i>	<i>purelin</i>	10	4,16*10 ⁻⁹	0,44864
132	CF	<i>tansig</i>	<i>purelin</i>	20	3,62*10 ⁻¹³	0,87329
133	CF	<i>tansig</i>	<i>purelin</i>	30	1,96*10 ⁻¹⁴	0,91874
134	CF	<i>tansig</i>	<i>purelin</i>	40	6,77*10 ⁻¹¹	0,75914
135	CF	<i>tansig</i>	<i>purelin</i>	50	5,79*10 ⁻¹³	-0,31404
136	CF	<i>tansig</i>	<i>satlin</i>	10	1,22*10 ⁻¹³	0,87318
137	CF	<i>tansig</i>	<i>satlin</i>	20	1,33*10 ⁻¹⁶	0,72883
138	CF	<i>tansig</i>	<i>satlin</i>	30	2,32*10 ⁻¹³	0,40876
139	CF	<i>tansig</i>	<i>satlin</i>	40	3,84*10 ⁻¹²	0,07603
140	CF	<i>tansig</i>	<i>satlin</i>	50	6,38*10 ⁻¹⁷	0,33829
141	CF	<i>purelin</i>	<i>logsig</i>	10	6,68*10 ⁻¹⁵	0,80508
142	CF	<i>purelin</i>	<i>logsig</i>	20	1,11*10 ⁻¹¹	0,51923
143	CF	<i>purelin</i>	<i>logsig</i>	30	2,59*10 ⁻¹⁴	0,93515
144	CF	<i>purelin</i>	<i>logsig</i>	40	1,08*10 ⁻¹⁴	-0,1983
145	CF	<i>purelin</i>	<i>logsig</i>	50	1,05*10 ⁻¹⁹	0,40273
146	CF	<i>purelin</i>	<i>tansig</i>	10	2,10*10 ⁻¹¹	0,31459
147	CF	<i>purelin</i>	<i>tansig</i>	20	7,43*10 ⁻¹³	0,97763
148	CF	<i>purelin</i>	<i>tansig</i>	30	2,87*10 ⁻¹⁶	-0,31214
149	CF	<i>purelin</i>	<i>tansig</i>	40	6,27*10 ⁻¹¹	0,67231
150	CF	<i>purelin</i>	<i>tansig</i>	50	1,58*10 ⁻¹⁹	0,71632

Fonte: Autoria própria (2022)

Tabela 12 - Testes com proporções diferentes, maior proporção de treinamento

(continua)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	2,14*10 ⁻¹¹	0,85972
2	FF	<i>logsig</i>	<i>logsig</i>	20	1,06*10 ⁻¹⁶	0,89335
3	FF	<i>logsig</i>	<i>logsig</i>	30	4,36*10 ⁻¹⁴	0,55943
4	FF	<i>logsig</i>	<i>logsig</i>	40	3,67*10 ⁻¹⁶	0,54142
5	FF	<i>logsig</i>	<i>logsig</i>	50	1,14*10 ⁻¹⁵	0,78889
6	FF	<i>logsig</i>	<i>tansig</i>	10	7,04*10 ⁻¹¹	0,82511
7	FF	<i>logsig</i>	<i>tansig</i>	20	7,95*10 ⁻¹⁶	0,93442
8	FF	<i>logsig</i>	<i>tansig</i>	30	1,50*10 ⁻¹²	0,5794
9	FF	<i>logsig</i>	<i>tansig</i>	40	1,10*10 ⁻¹⁴	0,0642
10	FF	<i>logsig</i>	<i>tansig</i>	50	2,97*10 ⁻¹²	0,55051
11	FF	<i>logsig</i>	<i>purelin</i>	10	1,00*10 ⁻¹⁰	0,68576
12	FF	<i>logsig</i>	<i>purelin</i>	20	2,08*10 ⁻¹¹	0,95444
13	FF	<i>logsig</i>	<i>purelin</i>	30	5,91*10 ⁻¹⁴	0,48567
14	FF	<i>logsig</i>	<i>purelin</i>	40	2,58*10 ⁻¹³	0,87495
15	FF	<i>logsig</i>	<i>purelin</i>	50	1,02*10 ⁻¹²	0,73675
16	FF	<i>logsig</i>	<i>satlin</i>	10	8,52*10 ⁻¹⁶	0,80378
17	FF	<i>logsig</i>	<i>satlin</i>	20	6,98*10 ⁻¹²	0,80936
18	FF	<i>logsig</i>	<i>satlin</i>	30	7,32*10 ⁻¹²	0,67573
19	FF	<i>logsig</i>	<i>satlin</i>	40	1,04*10 ⁻¹⁹	0,46952
20	FF	<i>logsig</i>	<i>satlin</i>	50	3,40*10 ⁻⁸	0,89275
21	FF	<i>tansig</i>	<i>logsig</i>	10	2,91*10 ⁻⁹	0,90171
22	FF	<i>tansig</i>	<i>logsig</i>	20	2,72*10 ⁻¹³	0,82621
23	FF	<i>tansig</i>	<i>logsig</i>	30	3,78*10 ⁻¹³	0,86393
24	FF	<i>tansig</i>	<i>logsig</i>	40	7,25*10 ⁻¹⁶	0,81275
25	FF	<i>tansig</i>	<i>logsig</i>	50	8,98*10 ⁻¹³	0,94169
26	FF	<i>tansig</i>	<i>tansig</i>	10	6,77*10 ⁻¹⁴	0,73651
27	FF	<i>tansig</i>	<i>tansig</i>	20	7,17*10 ⁻¹⁰	0,82419
28	FF	<i>tansig</i>	<i>tansig</i>	30	6,66*10 ⁻¹⁴	0,63134
29	FF	<i>tansig</i>	<i>tansig</i>	40	8,87*10 ⁻¹⁵	0,94814
30	FF	<i>tansig</i>	<i>tansig</i>	50	1,11*10 ⁻¹³	0,38566
31	FF	<i>tansig</i>	<i>purelin</i>	10	2,70*10 ⁻⁸	0,87453
32	FF	<i>tansig</i>	<i>purelin</i>	20	1,09*10 ⁻¹¹	0,70182
33	FF	<i>tansig</i>	<i>purelin</i>	30	1,93*10 ⁻¹⁶	-0,0076
34	FF	<i>tansig</i>	<i>purelin</i>	40	3,09*10 ⁻¹⁵	0,88504
35	FF	<i>tansig</i>	<i>purelin</i>	50	6,07*10 ⁻¹⁵	0,30084
36	FF	<i>tansig</i>	<i>satlin</i>	10	7,05*10 ⁻⁷	0,8898

Tabela 12 - Testes com proporções diferentes, maior proporção de treinamento

(conclusão)						
Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
37	FF	<i>tansig</i>	<i>satlin</i>	20	7,90*10 ⁻¹²	0,88132
38	FF	<i>tansig</i>	<i>satlin</i>	30	3,51*10 ⁻¹²	0,76698
39	FF	<i>tansig</i>	<i>satlin</i>	40	1,42*10 ⁻⁸	0,75934
40	FF	<i>tansig</i>	<i>satlin</i>	50	6,07*10 ⁻⁷	0,74717
41	FF	<i>purelin</i>	<i>logsig</i>	10	1,21*10 ⁻⁹	0,81212
42	FF	<i>purelin</i>	<i>logsig</i>	20	3,11*10 ⁻¹⁴	0,7977
43	FF	<i>purelin</i>	<i>logsig</i>	30	8,47*10 ⁻¹⁷	0,8045
44	FF	<i>purelin</i>	<i>logsig</i>	40	2,76*10 ⁻¹⁷	0,47699
45	FF	<i>purelin</i>	<i>logsig</i>	50	2,66*10 ⁻¹⁷	0,90185
46	FF	<i>purelin</i>	<i>tansig</i>	10	2,96*10 ⁻⁷	0,73345
47	FF	<i>purelin</i>	<i>tansig</i>	20	2,36*10 ⁻¹²	0,59709
48	FF	<i>purelin</i>	<i>tansig</i>	30	1,03*10 ⁻¹³	0,95765
49	FF	<i>purelin</i>	<i>tansig</i>	40	5,78*10 ⁻¹²	0,44781
50	FF	<i>purelin</i>	<i>tansig</i>	50	5,69*10 ⁻¹²	0,92141

Fonte: Autoria própria (2022)

Tabela 13 - Testes com proporções diferentes, menor proporção de treinamento

(continua)						
Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	<i>logsig</i>	10	8,80*10 ⁻¹¹	0,84241
2	FF	<i>logsig</i>	<i>logsig</i>	20	6,23*10 ⁻¹¹	0,89135
3	FF	<i>logsig</i>	<i>logsig</i>	30	2,74*10 ⁻¹¹	0,55707
4	FF	<i>logsig</i>	<i>logsig</i>	40	2,63*10 ⁻¹⁶	0,51927
5	FF	<i>logsig</i>	<i>logsig</i>	50	3,27*10 ⁻¹⁶	0,78862
6	FF	<i>logsig</i>	<i>tansig</i>	10	5,77*10 ⁻¹⁵	0,78715
7	FF	<i>logsig</i>	<i>tansig</i>	20	1,17*10 ⁻¹⁶	0,89907
8	FF	<i>logsig</i>	<i>tansig</i>	30	3,02*10 ⁻¹²	0,48632
9	FF	<i>logsig</i>	<i>tansig</i>	40	5,16*10 ⁻¹⁵	0,1507
10	FF	<i>logsig</i>	<i>tansig</i>	50	8,30*10 ⁻¹³	0,52883
11	FF	<i>logsig</i>	<i>purelin</i>	10	9,78*10 ⁻¹¹	0,64334
12	FF	<i>logsig</i>	<i>purelin</i>	20	3,51*10 ⁻¹³	0,95148
13	FF	<i>logsig</i>	<i>purelin</i>	30	2,67*10 ⁻¹⁴	0,43013
14	FF	<i>logsig</i>	<i>purelin</i>	40	1,37*10 ⁻¹⁵	0,76767
15	FF	<i>logsig</i>	<i>purelin</i>	50	1,34*10 ⁻¹²	0,72496
16	FF	<i>logsig</i>	<i>satlin</i>	10	8,55*10 ⁻¹²	0,81765

Tabela 13 - Testes com proporções diferentes, menor proporção de treinamento

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	(conclusão)
						R ² médio (rede)
17	FF	<i>logsig</i>	<i>satlin</i>	20	2,65*10 ⁻⁹	0,81022
18	FF	<i>logsig</i>	<i>satlin</i>	30	5,61*10 ⁻⁸	0,66883
19	FF	<i>logsig</i>	<i>satlin</i>	40	9,22*10 ⁻¹²	0,49935
20	FF	<i>logsig</i>	<i>satlin</i>	50	9,94*10 ⁻¹¹	0,89848
21	FF	<i>tansig</i>	<i>logsig</i>	10	6,85*10 ⁻¹⁵	0,80646
22	FF	<i>tansig</i>	<i>logsig</i>	20	3,66*10 ⁻¹³	0,89073
23	FF	<i>tansig</i>	<i>logsig</i>	30	4,81*10 ⁻¹⁶	0,64532
24	FF	<i>tansig</i>	<i>logsig</i>	40	1,67*10 ⁻¹¹	0,41634
25	FF	<i>tansig</i>	<i>logsig</i>	50	2,15*10 ⁻¹²	0,8305
26	FF	<i>tansig</i>	<i>tansig</i>	10	2,37*10 ⁻¹¹	0,78821
27	FF	<i>tansig</i>	<i>tansig</i>	20	8,77*10 ⁻¹³	0,91084
28	FF	<i>tansig</i>	<i>tansig</i>	30	5,65*10 ⁻¹⁷	0,53963
29	FF	<i>tansig</i>	<i>tansig</i>	40	5,46*10 ⁻¹²	0,24273
30	FF	<i>tansig</i>	<i>tansig</i>	50	1,16*10 ⁻¹⁶	0,60374
31	FF	<i>tansig</i>	<i>purelin</i>	10	5,14*10 ⁻¹¹	0,66781
32	FF	<i>tansig</i>	<i>purelin</i>	20	1,38*10 ⁻¹⁶	0,98111
33	FF	<i>tansig</i>	<i>purelin</i>	30	1,12*10 ⁻¹⁵	0,18654
34	FF	<i>tansig</i>	<i>purelin</i>	40	3,99*10 ⁻¹⁴	0,13377
35	FF	<i>tansig</i>	<i>purelin</i>	50	6,90*10 ⁻¹¹	0,61047
36	FF	<i>tansig</i>	<i>satlin</i>	10	1,52*10 ⁻⁶	0,71711
37	FF	<i>tansig</i>	<i>satlin</i>	20	4,25*10 ⁻¹¹	0,80169
38	FF	<i>tansig</i>	<i>satlin</i>	30	2,55*10 ⁻¹³	0,69383
39	FF	<i>tansig</i>	<i>satlin</i>	40	8,09*10 ⁻²²	0,86019
40	FF	<i>tansig</i>	<i>satlin</i>	50	3,91*10 ⁻¹⁰	0,90058
41	FF	<i>purelin</i>	<i>logsig</i>	10	1,34*10 ⁻⁸	0,78068
42	FF	<i>purelin</i>	<i>logsig</i>	20	9,11*10 ⁻¹⁰	0,59547
43	FF	<i>purelin</i>	<i>logsig</i>	30	1,58*10 ⁻¹¹	0,80321
44	FF	<i>purelin</i>	<i>logsig</i>	40	3,37*10 ⁻¹¹	0,50854
45	FF	<i>purelin</i>	<i>logsig</i>	50	4,29*10 ⁻¹⁴	0,86253
46	FF	<i>purelin</i>	<i>tansig</i>	10	2,23*10 ⁻⁸	0,98448
47	FF	<i>purelin</i>	<i>tansig</i>	20	1,10*10 ⁻¹³	0,65465
48	FF	<i>purelin</i>	<i>tansig</i>	30	6,28*10 ⁻¹¹	0,80485
49	FF	<i>purelin</i>	<i>tansig</i>	40	7,50*10 ⁻¹⁴	0,4026
50	FF	<i>purelin</i>	<i>tansig</i>	50	1,08*10 ⁻¹¹	0,14517

Fonte: Autoria própria (2022)

Tabela 14 - Testes iniciais com código otimizado

(continua)

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	R ² médio (rede)
1	FF	<i>logsig</i>	10	5,75*10 ⁻¹⁴	1
2	FF	<i>logsig</i>	20	5,14*10 ⁻¹⁷	0,99968
3	FF	<i>logsig</i>	30	1,17*10 ⁻¹¹	0,99958
4	FF	<i>logsig</i>	40	7,76*10 ⁻¹¹	0,99969
5	FF	<i>logsig</i>	50	5,79*10 ⁻¹¹	0,99697
6	FF	<i>tansig</i>	10	9,44*10 ⁻¹¹	1
7	FF	<i>tansig</i>	20	3,78*10 ⁻¹³	0,99989
8	FF	<i>tansig</i>	30	4,96*10 ⁻¹³	0,9999
9	FF	<i>tansig</i>	40	1,29*10 ⁻¹⁴	0,99962
10	FF	<i>tansig</i>	50	8,70*10 ⁻¹⁵	0,99619
11	FF	<i>purelin</i>	10	1,53*10 ⁻²	0,98203
12	FF	<i>purelin</i>	20	1,53*10 ⁻²	0,98189
13	FF	<i>purelin</i>	30	1,53*10 ⁻²	0,98188
14	FF	<i>purelin</i>	40	1,53*10 ⁻²	0,98188
15	FF	<i>purelin</i>	50	1,53*10 ⁻²	0,98188
16	FF	<i>satlin</i>	10	1,38*10 ⁻¹⁸	1
17	FF	<i>satlin</i>	20	1,76*10 ⁻¹⁵	0,99989
18	FF	<i>satlin</i>	30	2,15*10 ⁻¹⁴	0,99986
19	FF	<i>satlin</i>	40	4,01*10 ⁻¹²	0,99972
20	FF	<i>satlin</i>	50	6,21*10 ⁻¹¹	0,99795
21	CF	<i>logsig</i>	10	3,04*10 ⁻¹⁵	0,99999
22	CF	<i>logsig</i>	20	2,09*10 ⁻¹¹	0,99991
23	CF	<i>logsig</i>	30	3,86*10 ⁻¹³	0,9998
24	CF	<i>logsig</i>	40	3,40*10 ⁻¹²	0,99983
25	CF	<i>logsig</i>	50	4,68*10 ⁻¹²	0,99923
26	CF	<i>tansig</i>	10	2,82*10 ⁻¹⁰	0,99992
27	CF	<i>tansig</i>	20	1,04*10 ⁻¹¹	0,99698
28	CF	<i>tansig</i>	30	4,68*10 ⁻¹²	0,99592
29	CF	<i>tansig</i>	40	1,90*10 ⁻¹⁴	0,99911
30	CF	<i>tansig</i>	50	2,37*10 ⁻¹⁵	0,99782
31	CF	<i>purelin</i>	10	1,53*10 ⁻²	0,98204
32	CF	<i>purelin</i>	20	1,53*10 ⁻²	0,98188
33	CF	<i>purelin</i>	30	1,53*10 ⁻²	0,98187
34	CF	<i>purelin</i>	40	1,53*10 ⁻²	0,98187
35	CF	<i>purelin</i>	50	1,53*10 ⁻²	0,98187
36	CF	<i>satlin</i>	10	7,56*10 ⁻¹⁰	1

Tabela 14 - Testes iniciais com código otimizado

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	(conclusão)
					R ² médio (rede)
37	CF	<i>satlin</i>	20	3,84*10 ⁻¹⁴	0,99992
38	CF	<i>satlin</i>	30	5,09*10 ⁻¹⁷	0,99995
39	CF	<i>satlin</i>	40	2,07*10 ⁻¹⁴	0,99988
40	CF	<i>satlin</i>	50	8,96*10 ⁻¹⁸	0,9963
41	EL	<i>logsig</i>	10	8,33*10 ⁻¹⁵	
42	EL	<i>logsig</i>	20	2,96*10 ⁻¹³	
43	EL	<i>logsig</i>	30	1,57*10 ⁻¹¹	
44	EL	<i>logsig</i>	40	1,55*10 ⁻¹¹	
45	EL	<i>logsig</i>	50	1,52*10 ⁻¹¹	
46	EL	<i>tansig</i>	10	1,96*10 ⁻¹¹	
47	EL	<i>tansig</i>	20	4,95*10 ⁻¹¹	
48	EL	<i>tansig</i>	30	1,77*10 ⁻¹³	
49	EL	<i>tansig</i>	40	1,78*10 ⁻¹³	
50	EL	<i>tansig</i>	50	4,14*10 ⁻¹⁴	
51	EL	<i>purelin</i>	10	1,36*10 ⁻²	
52	EL	<i>purelin</i>	20	1,54*10 ⁻²	
53	EL	<i>purelin</i>	30	1,87*10 ⁻²	
54	EL	<i>purelin</i>	40	1,67*10 ⁻²	
55	EL	<i>purelin</i>	50	1,61*10 ⁻²	
56	EL	<i>satlin</i>	10	1,52*10 ⁻¹⁶	
57	EL	<i>satlin</i>	20	3,08*10 ⁻²¹	
58	EL	<i>satlin</i>	30	1,78*10 ⁻¹²	
59	EL	<i>satlin</i>	40	1,66*10 ⁻¹³	
60	EL	<i>satlin</i>	50	2,72*10 ⁻¹²	

Fonte: Autoria própria (2022)

Tabela 15 - Testes definitivos com primeiro conjunto de dados

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	(continua)
					R ² médio (rede)
1	FF	<i>logsig</i>	10	5,12*10 ⁻¹¹	1
2	FF	<i>logsig</i>	20	7,58*10 ⁻¹³	0,99994
3	FF	<i>logsig</i>	30	3,56*10 ⁻¹²	0,99978
4	FF	<i>logsig</i>	40	1,55*10 ⁻¹⁷	0,99944
5	FF	<i>logsig</i>	50	6,09*10 ⁻¹¹	0,99905
6	FF	<i>tansig</i>	10	2,73*10 ⁻¹¹	1

Tabela 15 - Testes definitivos com primeiro conjunto de dados

(continuação)

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	R ² médio (rede)
7	FF	<i>tansig</i>	20	3,34*10 ⁻¹³	0,99975
8	FF	<i>tansig</i>	30	4,30*10 ⁻¹⁶	0,99777
9	FF	<i>tansig</i>	40	4,95*10 ⁻¹²	0,99554
10	FF	<i>tansig</i>	50	1,03*10 ⁻¹⁷	0,99043
11	FF	<i>purelin</i>	10	1,20*10 ⁻³	0,9871
12	FF	<i>purelin</i>	20	1,20*10 ⁻³	0,98795
13	FF	<i>purelin</i>	30	1,20*10 ⁻³	0,98758
14	FF	<i>purelin</i>	40	1,20*10 ⁻³	0,98759
15	FF	<i>purelin</i>	50	1,20*10 ⁻³	0,98758
16	FF	<i>satlin</i>	10	1,84*10 ⁻¹³	1
17	FF	<i>satlin</i>	20	1,41*10 ⁻¹¹	0,99999
18	FF	<i>satlin</i>	30	4,70*10 ⁻¹¹	0,99964
19	FF	<i>satlin</i>	40	2,57*10 ⁻¹⁶	0,99952
20	FF	<i>satlin</i>	50	7,13*10 ⁻¹⁵	0,99899
21	CF	<i>logsig</i>	10	8,01*10 ⁻¹¹	0,99997
22	CF	<i>logsig</i>	20	6,22*10 ⁻⁹	0,99976
23	CF	<i>logsig</i>	30	8,23*10 ⁻¹²	0,99836
24	CF	<i>logsig</i>	40	3,86*10 ⁻¹⁶	0,99692
25	CF	<i>logsig</i>	50	1,46*10 ⁻¹¹	0,99765
26	CF	<i>tansig</i>	10	7,83*10 ⁻¹⁷	1
27	CF	<i>tansig</i>	20	1,28*10 ⁻¹⁷	0,9996
28	CF	<i>tansig</i>	30	1,50*10 ⁻¹⁶	0,99908
29	CF	<i>tansig</i>	40	2,94*10 ⁻¹²	0,99858
30	CF	<i>tansig</i>	50	1,33*10 ⁻¹⁸	0,99502
31	CF	<i>purelin</i>	10	1,20*10 ⁻³	0,98758
32	CF	<i>purelin</i>	20	1,20*10 ⁻³	0,98758
33	CF	<i>purelin</i>	30	1,20*10 ⁻³	0,98758
34	CF	<i>purelin</i>	40	1,20*10 ⁻³	0,98758
35	CF	<i>purelin</i>	50	1,20*10 ⁻³	0,98758
36	CF	<i>satlin</i>	10	4,08*10 ⁻¹⁵	1
37	CF	<i>satlin</i>	20	4,82*10 ⁻⁹	0,99986
38	CF	<i>satlin</i>	30	8,49*10 ⁻¹⁷	0,99961
39	CF	<i>satlin</i>	40	1,02*10 ⁻¹¹	0,99937
40	CF	<i>satlin</i>	50	6,91*10 ⁻¹³	0,99922
41	EL	<i>logsig</i>	10	5,28*10 ⁻¹¹	
42	EL	<i>logsig</i>	20	8,14*10 ⁻¹¹	

Tabela 15 - Testes definitivos com primeiro conjunto de dados

Teste	Tipo de rede	Função de ativação	Número de neurônios	MSE (rede)	(conclusão)
					R ² médio (rede)
43	EL	<i>logsig</i>	30	9,87*10 ⁻¹¹	
44	EL	<i>logsig</i>	40	1,23*10 ⁻¹³	
45	EL	<i>logsig</i>	50	2,30*10 ⁻¹⁵	
46	EL	<i>tansig</i>	10	4,07*10 ⁻¹⁶	
47	EL	<i>tansig</i>	20	3,06*10 ⁻¹³	
48	EL	<i>tansig</i>	30	2,27*10 ⁻¹¹	
49	EL	<i>tansig</i>	40	3,97*10 ⁻¹⁴	
50	EL	<i>tansig</i>	50	3,03*10 ⁻¹⁵	
51	EL	<i>purelin</i>	10	1,01*10 ⁻³	
52	EL	<i>purelin</i>	20	1,51*10 ⁻³	
53	EL	<i>purelin</i>	30	1,06*10 ⁻³	
54	EL	<i>purelin</i>	40	1,25*10 ⁻³	
55	EL	<i>purelin</i>	50	1,28*10 ⁻³	
56	EL	<i>satlin</i>	10	3,13*10 ⁻¹⁷	
57	EL	<i>satlin</i>	20	1,66*10 ⁻¹²	
58	EL	<i>satlin</i>	30	8,14*10 ⁻¹²	
59	EL	<i>satlin</i>	40	3,79*10 ⁻³	
60	EL	<i>satlin</i>	50	7,38*10 ⁻¹¹	

Fonte: Autoria própria (2022)

APÊNDICE E – TESTES INICIAIS COM CONJUNTO DE DADOS COMPLETO

Tabela 16 - Testes iniciais com segundo conjunto de dados

(continua)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)	MSE (validação)
1	FF	<i>logsig</i>	<i>logsig</i>	10	5,68*10 ⁻¹²	0,82343	209,329
2	FF	<i>logsig</i>	<i>logsig</i>	20	5,21*10 ⁻¹⁴	0,9492	11,613
3	FF	<i>logsig</i>	<i>logsig</i>	30	1,46*10 ⁻¹²	0,98046	111,536
4	FF	<i>logsig</i>	<i>logsig</i>	40	6,39*10 ⁻¹⁷	0,86859	185,382
5	FF	<i>logsig</i>	<i>logsig</i>	50	3,72*10 ⁻¹⁴	0,79616	192,521
6	FF	<i>logsig</i>	<i>tansig</i>	10	4,98*10 ⁻⁹	0,76633	269,427
7	FF	<i>logsig</i>	<i>tansig</i>	20	4,65*10 ⁻¹³	0,85278	225,196
8	FF	<i>logsig</i>	<i>tansig</i>	30	2,66*10 ⁻¹¹	0,74108	313,469
9	FF	<i>logsig</i>	<i>tansig</i>	40	4,96*10 ⁻¹³	0,61207	239,491
10	FF	<i>logsig</i>	<i>tansig</i>	50	7,04*10 ⁻¹⁴	0,71313	283,916
11	FF	<i>logsig</i>	<i>purelin</i>	10	5,87*10 ⁻¹⁴	0,6286	324,525
12	FF	<i>logsig</i>	<i>purelin</i>	20	3,57*10 ⁻¹⁶	0,99234	56,096
13	FF	<i>logsig</i>	<i>purelin</i>	30	3,99*10 ⁻¹⁶	0,87611	17,31
14	FF	<i>logsig</i>	<i>purelin</i>	40	1,29*10 ⁻¹³	0,85904	221,713
15	FF	<i>logsig</i>	<i>purelin</i>	50	9,14*10 ⁻¹³	0,30086	526,322
16	FF	<i>logsig</i>	<i>satlin</i>	10	8,39*10 ⁻⁸	0,99115	50,646
17	FF	<i>logsig</i>	<i>satlin</i>	20	3,00*10 ⁻¹⁷	0,93137	126,612
18	FF	<i>logsig</i>	<i>satlin</i>	30	1,51*10 ⁻¹⁰	0,86746	154,037
19	FF	<i>logsig</i>	<i>satlin</i>	40	1,20*10 ⁻¹⁷	0,98282	73,217
20	FF	<i>logsig</i>	<i>satlin</i>	50	7,22*10 ⁻⁵	0,94885	98,921
21	FF	<i>tansig</i>	<i>logsig</i>	10	7,09*10 ⁻¹¹	0,89247	149,769
22	FF	<i>tansig</i>	<i>logsig</i>	20	6,85*10 ⁻¹⁷	0,82214	129,776
23	FF	<i>tansig</i>	<i>logsig</i>	30	1,14*10 ⁻¹⁷	0,97489	77,081
24	FF	<i>tansig</i>	<i>logsig</i>	40	8,65*10 ⁻¹⁴	0,96943	116,031
25	FF	<i>tansig</i>	<i>logsig</i>	50	6,73*10 ⁻¹¹	0,94267	90,788
26	FF	<i>tansig</i>	<i>tansig</i>	10	1,60*10 ⁻¹⁷	0,99601	22,542
27	FF	<i>tansig</i>	<i>tansig</i>	20	3,42*10 ⁻¹³	0,98293	59,162
28	FF	<i>tansig</i>	<i>tansig</i>	30	8,62*10 ⁻¹⁹	0,91261	196,342
29	FF	<i>tansig</i>	<i>tansig</i>	40	2,37*10 ⁻¹⁶	0,80329	171,853
30	FF	<i>tansig</i>	<i>tansig</i>	50	9,32*10 ⁻¹⁷	0,91561	158,634
31	FF	<i>tansig</i>	<i>purelin</i>	10	4,45*10 ⁻¹⁴	0,69463	33,229
32	FF	<i>tansig</i>	<i>purelin</i>	20	9,88*10 ⁻¹⁸	0,91269	97,284
33	FF	<i>tansig</i>	<i>purelin</i>	30	2,29*10 ⁻¹³	0,91726	69,231
34	FF	<i>tansig</i>	<i>purelin</i>	40	2,11*10 ⁻¹⁸	-0,12348	161,186
35	FF	<i>tansig</i>	<i>purelin</i>	50	5,78*10 ⁻¹⁸	0,10073	979,203

Tabela 16 - Testes iniciais com segundo conjunto de dados

(conclusão)

Teste	Tipo de rede	Função de ativação (1)	Função de ativação (2)	Número de neurônios	MSE (rede)	R ² médio (rede)	MSE (validação)
36	FF	<i>tansig</i>	<i>satlin</i>	10	$7,86 \cdot 10^{-12}$	0,99512	37,457
37	FF	<i>tansig</i>	<i>satlin</i>	20	$5,17 \cdot 10^{-11}$	0,94472	104,731
38	FF	<i>tansig</i>	<i>satlin</i>	30	$4,56 \cdot 10^{-15}$	0,84029	216,719
39	FF	<i>tansig</i>	<i>satlin</i>	40	$5,68 \cdot 10^{-11}$	0,98551	94,172
40	FF	<i>tansig</i>	<i>satlin</i>	50	$4,30 \cdot 10^{-7}$	0,92352	204,836
41	FF	<i>purelin</i>	<i>logsig</i>	10	$7,19 \cdot 10^{-17}$	0,99586	42,236
42	FF	<i>purelin</i>	<i>logsig</i>	20	$1,11 \cdot 10^{-12}$	0,99859	17,295
43	FF	<i>purelin</i>	<i>logsig</i>	30	$2,18 \cdot 10^{-19}$	0,71226	186,974
44	FF	<i>purelin</i>	<i>logsig</i>	40	$7,50 \cdot 10^{-12}$	0,49467	358,133
45	FF	<i>purelin</i>	<i>logsig</i>	50	$1,47 \cdot 10^{-11}$	0,51755	346,206
46	FF	<i>purelin</i>	<i>tansig</i>	10	$2,12 \cdot 10^{-5}$	0,59567	173,339
47	FF	<i>purelin</i>	<i>tansig</i>	20	$9,83 \cdot 10^{-11}$	0,9631	137,947
48	FF	<i>purelin</i>	<i>tansig</i>	30	$1,18 \cdot 10^{-11}$	0,76233	16,412
49	FF	<i>purelin</i>	<i>tansig</i>	40	$1,34 \cdot 10^{-12}$	0,54462	504,647
50	FF	<i>purelin</i>	<i>tansig</i>	50	$1,02 \cdot 10^{-16}$	0,69835	141,486

Fonte: Autoria própria (2022)