



Universidade Federal
do Rio de Janeiro
Escola Politécnica

APLICAÇÃO DE REDES NEURAIS EM PERMUTADORES DE CALOR
TIPO CIRCUITO IMPRESSO

Lucas Rangel Freire

Projeto de Graduação apresentado ao Curso
de Engenharia Mecânica da Escola
Politécnica, Universidade Federal do Rio de
Janeiro, como parte dos requisitos
necessários à obtenção do título de
Engenheiro.

Orientador:

Prof. Carolina Palma Naveira Cotta

Rio de Janeiro, junho de 2021



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Departamento de Engenharia Mecânica
DEM / POLI/ UFRJ



Universidade Federal
do Rio de Janeiro
Escola Politécnica

APLICAÇÃO DE REDES NEURAIS EM PERMUTADORES DE CALOR
TIPO CIRCUITO IMPRESSO

Lucas Rangel Freire

PROJETO FINAL SUBMETIDO AO CORPO DOCENTE DO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA ESCOLA POLITÉCNICA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO MECÂNICO.

Aprovada por:

Prof. ^a Carolina Palma Naveira Cotta, D.Sc.

Prof. Su Jian, D.Sc

Prof. Helcio Rangel Barreto Orlande, D.Sc

Prof. Cláudio Márcio do Nascimento Abreu Pereira, D.Sc

Rio de Janeiro, junho de 2021.

Rangel Freire, Lucas

APLICAÇÃO DE REDES NEURAIS EM
TROCADORES DE CALOR TIPO PCHE / Lucas Rangel
Freire. – Rio de Janeiro: UFRJ/ Escola Politécnica, 2021.

IX, 79 p.:il.;29,7cm

Orientador: Carolina Palma Naveira Cotta

Projeto de Graduação – UFRJ/ Escola Politécnica /
Curso de Engenharia Mecânica, 2021.

Referência Bibliográficas: p. 112-113.

1. PCHE. 2. Inteligência Artificial 3. Redes Neurais
4. Correlações Nusselt 5. Correlações Fator de Atrito

I. Palma Naveira Cotta, Carolina. II. Universidade
Federal do Rio de Janeiro, UFRJ, Curso de Engenharia
Mecânica.

III. APLICAÇÃO DE REDES NEURAIS EM
TROCADORES DE CALOR TIPO PCHE

Dedico este trabalho aos meus pais Edina e Neviton.

AGRADECIMENTOS

Aos meus pais Edina e Neviton que sempre estiveram ao meu lado me apoiando ao longo de toda a minha trajetória.

Aos meus familiares que estiveram ao meu lado em todos os momentos.

A todos os meus professores do curso de Engenharia Mecânica da Universidade Federal do Rio de Janeiro pela excelência da qualidade técnica de cada um.

A minha professora orientadora Carolina pelas valiosas contribuições dadas durante todo o processo.

Resumo do Projeto de Graduação apresentado à Escola Politécnica/UFRJ como parte dos requisitos necessários para a obtenção do grau de Engenheiro Mecânico

APLICAÇÃO DE REDES NEURAIS EM TROCADORES DE CALOR TIPO PCHE

Lucas Rangel Freire

Junho de 2021

Orientador: Carolina Palma Naveira Cotta

Programa: Engenharia Mecânica

Este trabalho envolve o estudo de trocadores de calor compactos do tipo Circuito Impresso (PCHE - Printed Circuit Heat Exchangers). Os PCHEs apresentam vantagens em diversas aplicações industriais quando comparado com outros trocadores de calor tradicionais do mercado, dentre elas se destacam: tamanho compacto, peso reduzido e a capacidade de operar à altas pressões e altas temperaturas. O objetivo deste trabalho foi aplicar uma abordagem via redes neurais para prever o impacto na performance termohidráulica de um trocador de calor PCHE devido a mudanças na geometria interna dos canais ou mudanças nas condições de operação. Simulações de elevado custo computacional envolvendo a solução de equações diferenciais parciais relacionadas aos princípios de conservação de massa, quantidade de movimento e energia foram empregadas no treinamento da rede neural. Os resultados para o número de Nusselt e fator de atrito de Fanning obtidos por meio da abordagem via Inteligência artificial foram comparados com as previsões fornecidas por correlações da literatura.

Palavras-chave: Trocador de calor de circuito impresso, Rede neural artificial, PCHE, Inteligência Artificial.

Abstract of Undergraduate Project presente to POLI/UFRJ as a partial fulfillment of the requirements for the degree of Mechanical Engineer.

APPLICATIONS OF NEURAL NETWORKS INTO HEAT EXCHANGERS TYPE PCHE.

Lucas Rangel Freire

Jun / 2021

Advisor: Carolina Palma Naveira Cotta

Department: Mechanical Engineering

This work involves the study of compact printed circuit heat exchangers. The PCHEs have advantages in several industrial applications when compared to other traditional heat exchangers on the market, features stand out are: compact size, reduced weight and the ability to operate at high pressures and high temperatures. The objective of this work was to apply the neural networks method to predict the impact on the thermo-hydraulic performance of a heat exchanger, due to changes in the internal geometry of the channels or changes in operating conditions. High computational cost simulations involving the solution of partial differential equations related to the principles of conservation of mass, momentum and energy were employed in the training of the neural network. The results for the Nusselt number and Fanning friction factor were obtained through the artificial intelligence method and compared with the predictions provided by correlations in the literature.

Key words: Printed circuit heat exchanger, Artificial neural network, PCHE, Artificial Intelligence.

Sumário

Lista de Figuras.....	10
Lista de Tabelas	12
Lista de Símbolos.....	15
1. Introdução	18
1.1 Motivação e contextualização.....	18
1.2 Objetivos	22
1.3 Organização do trabalho	22
2. Revisão Bibliográfica	23
2.1 Trocador de calor tipo circuito impresso (PCHE)	23
2.2 Redes Neurais.....	28
3. Fundamentação Teórica.....	30
3.1 Aplicações de IA	31
3.2 Aprendizado de Máquina	32
3.3 Fundamentos Redes Neurais Artificiais.....	34
3.3.1 Redes Neurais Biológicas e Redes Neurais Artificiais.....	34
3.3.2 Treinamento e validação da Rede.....	36
3.3.3 Regularização dos dados	37
3.3.4 Número de camadas intermediárias e Neurônios	39
3.3.5 Algoritmos para treinamento.....	39
3.3.6 Função Transferência	40
3.3.7 Pesos.....	42
3.3.8 Critério do Erro	42
3.3.9 Critério de parada e <i>Overfitting</i>	42
3.3.10 Método feed-forward e back-propagation	43
3.3.11 Limitações ANN	49
4. Resultados	50
4.1 Análise Correlação.....	50
4.1.1 Correlação matemática	50
4.1.2 Conjunto de dados	51
4.1.3 Performance da ANN.....	53
4.2 Análise Correlação com ruído branco	73

4.2.1 Caso 1	73
4.2.2 Caso 2	79
4.2.3 Análise da influência do ruído de entrada na predição de Nusselt e Fator de Atrito.....	84
4.3 Análise simulação CFD 3D	90
4.3.1 Análise do tamanho dos dados de treinamento.....	90
4.3.2 Estrutura da ANN	94
4.3.3 Imprevisibilidade devido a divisão dos dados.....	96
4.3.4 Performance da ANN.....	97
4.3.5 Gráfico 3D da rede neural	102
4.4 Análise do treinamento da rede neural com dados da simulação CFD 3D incluindo ruído na saída	105
5 Considerações finais.....	111
6. Referências Bibliográficas	112
Anexos.....	114
A.1 Conjuntos de dados.....	114
A.2 Amostra de 40 dados do Conjunto 1.....	117
A.3 Rol exemplificativo dos dados da simulação CFD 3D com seus respectivos desvios padrão ...	118
A.4 Resumo de correlações da literatura	119

Lista de Figuras

Figura 1.1 - PCHE numa planta de mistura de Gás-Água.....	18
Figura 1.2- Trocador de Calor Casco-Tubo.....	19
Figura 1.3- Trocador de Calor a Placas	19
Figura 1.4- Diagrama esquemático de um PCHE.....	19
Figura 1.5- Capacidades de temperatura e pressão dos trocados de calor.....	20
Figura 2.1 - Zigue-zague de borda aguda e borda arredondada.	23
Figura 2.2- Trocador de calor tipo circuito impresso (PCHE).	24
Figura 2.3- Exemplo de Correlações propostas para diferentes tipos de canais usados em PCHEs.	25
Figura 2.4- Variação do número de Nusselt e fator de atrito locais do canal quente com o número de Reynolds para cada ângulo.	27
Figura 2.5- Valor obtido do RMSE com uma rede neural com diferente número de neurônios na camada intermediária.....	29
Figura 3.1- Fluxograma IA.....	30
Figura 3.2- Perspectiva histórica da Inteligência Artificial.....	32
Figura 3.3- Relação entre IA e ML.....	33
Figura 3.4- Neurônio Biológico vs artificial	35
Figura 3.5- Comparação entre as conexões em um sistema biológico e em um sistema artificial.	36
Figura 3.6- Funções transferência da rede neural.....	41
Figura 3.7- Cálculo de um neurônio.....	45
Figura 3.8- Diagrama algoritmo Neural network.	46
Figura 3.9- Diagrama de um código para ANN	48
Figura 4.1- Geração de 1 unidade do conjunto dados	52
Figura 4.2- Erro das redes neurais baseado na quantidade de camadas intermediárias. (a) Camada intermediária está no eixo da abscissa. (b) Neurônios está na abscissa.	54
Figura 4.3-Performance Função de otimização para cada conjunto.....	57
Figura 4.4- Influência do <i>batch</i> no erro. (a) Figura com todos os <i>batches</i> . (b) Figura com 3 tipos de <i>batch</i>	58

Figura 4.5- Efeito do Número de Épocas no erro.....	60
Figura 4.6- Ilustração da rede neural, destaque para valores de input e output.....	63
Figura 4.7- BoxPlot associado a um histograma.	66
Figura 4.8- Teste massivo da rede neural.....	68
Figura 4.9- Redução do MSE por época. Curva azul treinamento, curva laranja teste.....	69
Figura 4.10- Ajuste curva ANN X Correlação	70
Figura 4.11- Ajuste do modelo a variação do ângulo α	71
Figura 4.12- Ajuste do modelo ao se variar o ângulo LDh.....	72
Figura 4.13 – Comparação do erro relativo na entrada durante o treinamento e o erro relativo da saída.....	85
Figura 4.14 - Comparação do erro relativo na entrada durante o treinamento e o erro relativo da saída.....	87
Figura 4.15 - Comparação do erro relativo na entrada durante o treinamento e o erro relativo da saída.....	88
Figura 4.16 - Compilado dos coeficientes de Person calculados entre os ruídos de entrada e os erros.....	89
Figura 4.17- BoxPlot	97
Figura 4.18- Comparação do ajuste da ANN e Correlação para α e LDh pequenos	100
Figura 4.19- Comparação do ajuste da ANN e Correlação para α grande.	101
Figura 4.20- Gráfico 3D de Nusselt para Reynolds de 1000. As figuras (a), (b), (c) e (d) representam uma rotação em torno do eixo Z.	102
Figura 4.21- Análise de gráfico 3D para quatro Reynolds diferentes.	103
Figura 4.22- Gráfico 3D do Fator de atrito para Reynolds de 1000. As figuras (a), (b), (c) e (d) representam uma rotação em torno do eixo Z.	104
Figura 4.23 – Gráfico do MSE por época.....	106
Figura 4.24 – Ajuste da rede neural para condição geométrica específica.....	109
Figura 4.25 – Ajuste da rede neural para condição geométrica específica.....	110

Lista de Tabelas

Tabela 2.1- Correlações numéricas para PCHE	26
Tabela 3.1– Rol exemplificativo de Técnicas e Modelos de ML	34
Tabela 3.2– Funções de ativação usualmente utilizada	40
Tabela 4.1- Conjunto 1 (252 combinações).....	51
Tabela 4.2- Número de dados por conjunto	52
Tabela 4.3- Parâmetros nas simulações para camada intermediária.	53
Tabela 4.4- Valor mínimo de MSE para cada N° de camada intermediária.....	54
Tabela 4.5- Parâmetros de simulações do estudo da influência da quantidade de dados de treinamento da ANN.....	55
Tabela 4.6- Comparações do MSE em relação a quantidade de dados usados no treinamento.	55
Tabela 4.7- Parâmetros de simulações de estudo da função de otimização.	56
Tabela 4.8- Parâmetros de simulações de estudo do <i>Batch_size</i>	57
Tabela 4.9- Dez <i>batchs</i> com menores valores do MSE.....	58
Tabela 4.10- Parâmetros de simulações de estudo da época	59
Tabela 4.11- Tabela em ordem decrescente de erro, 10 primeiras configurações.....	60
Tabela 4.12- Parâmetros para testes de diferentes funções de transferência.....	60
Tabela 4.13- Diferentes configurações de funções transferência por camada.	61
Tabela 4.14- MSE para diferentes composições de função transferência e com diferentes números de neurônios.....	62
Tabela 4.15- Propriedades da rede neural	64
Tabela 4.16- Parâmetros para cálculo do Intervalo de confiança de 95%.....	67
Tabela 4.17- Propagação de erro	73
Tabela 4.18- Tabela de desvio padrão do input (ruído branco).....	74
Tabela 4.19- Características das simulações	75
Tabela 4.20- Melhores estruturas de rede para 2 camadas	76
Tabela 4.21- Impacto no ajusto devido ao ruído branco (2 camadas intermediárias).....	76
Tabela 4.22- Características das simulações com 3 camadas intermediárias.....	77
Tabela 4.23- Melhores estruturas de rede para 3 camadas.	78

Tabela 4.24- Impacto no ajusto devido ao ruído branco (3 camadas intermediárias)	79
Tabela 4.25- Ruído input para o caso 2	79
Tabela 4.26- Características das simulações	80
Tabela 4.27- Melhores estruturas de rede para 2 camadas	81
Tabela 4.28- Impacto no ajusto devido ao ruído branco (2 camadas intermediárias)	81
Tabela 4.29- Características das simulações com 3 camadas intermediárias.....	82
Tabela 4.30- Melhores estruturas de rede para 3 camadas.	83
Tabela 4.31- Impacto no ajuste devido ao ruído branco (3 camadas intermediárias)	83
Tabela 4.32 – Configuração da rede neural nesta seção de testes	84
Tabela 4.33 – MSE médio da rede neural	85
Tabela 4.34 – Pearson entre ruído e MSE	86
Tabela 4.35 – MSE médio da rede neural	86
Tabela 4.36 – Pearson entre ruído e MSE	87
Tabela 4.37 – MSE médio da rede neural	88
Tabela 4.38 – Pearson entre ruído na entrada e MSE.....	89
Tabela 4.39- Parâmetros das simulações.....	90
Tabela 4.40- Melhor estrutura para cada split size.....	91
Tabela 4.41- Percentual dos dados usados no treinamento da rede.....	92
Tabela 4.42- Percentual dos dados usados no treinamento da rede.....	92
Tabela 4.43- Percentual dos dados usados no treinamento da rede.....	93
Tabela 4.44- Simulações realizadas para os dados CFD	94
Tabela 4.45- 10 menores MSE para a Transformada de potência.....	95
Tabela 4.46- 10 menores MSE para a MinMax.....	95
Tabela 4.47- Parâmetros para cálculo do Intervalo de confiança de 95%.....	96
Tabela 4.48 – Configuração da ANN avaliada.....	97
Tabela 4.49- Comparaçao dos resultados para Nusselt entre ANN e Correlação	98
Tabela 4.50- Comparaçao dos resultados para Fat entre ANN e Correlação	99
Tabela 4.51 – Estrutura da rede neural	105
Tabela 4.52 – Comparaçao entre os erros relativos da ANN e correlação para o Nusselt....	107
Tabela 4.53 - Comparaçao entre os erros relativos da ANN e correlação para o Fator de atrito.	
	107

Lista de Símbolos

Símbolos e Siglas latinas:

b	Constante de soma associado ao neurônio
D_h	Diâmetro hidráulico
$IC_{95\%}$	Intervalo de confiança de 95%
E	Erro da rede neural
f	Fator de atrito de Fanning
h	Coeficiente de transferência de calor
k	Condutividade térmica do fluido
L	Comprimento segmento do canal
n	Número de elementos da amostra
Nu	Número de Nusselt
P	Pressão
Pr	Número de Prandtl
Re	Número de Reynolds
s	Desvio padrão amostral
V	Velocidade do escoamento
w	Peso do neurônio
\bar{x}	Média amostral

$z_{95\%}$ Escore padrão

Símbolos e Siglas gregas:

α Ângulo de canal

σ Desvio padrão

μ Média

η Taxa de aprendizagem

ΔP Diferença de pressão

X_{scaled} Dado regularizado

X_i Dados de entrada

X_{std} Dado transformado

X_{max} Dado de entrada com maior valor

X_{min} Dado de entrada de menor valor

Y Valor calculado pela rede neural

\hat{Y}_i Valor de referência de treinamento

ΔT Diferença entre as temperaturas

Abreviações:

ANN *Artificial Neural Network*

RNN *Recurrent Neural Network*

CFD	<i>Computational Fluid Dynamics</i>
IA	Inteligência Artificial
ML	<i>Machine Learning</i>
MSE	<i>Mean Squared Error</i>
PCHE	<i>Printed-Circuit Heat Exchanger</i>
RMSE	<i>Root Mean Squared Error</i>

1. Introdução

1.1 Motivação e contextualização

Este trabalho envolve o estudo de Trocadores de Calor de Circuito Impresso (PCHE). Os PCHE possuem vantagens consideráveis em aplicações industriais quando comparado com outros trocadores de calor tradicionais do mercado, dentre elas se destacam: tamanho compacto, peso reduzido, resistência à altas pressões e temperaturas. Esses trocadores são de grande valor para indústria petroquímica, principalmente para cenários offshore, onde existe grande restrição de peso e espaço. O PCHE possui um amplo potencial de aplicação na indústria de energia nuclear, energia solar concentrada, energia de hidrogênio e geração de energia a partir do carvão.

Várias empresas do setor de óleo e gás utilizam os PCHE para o serviço de resfriamento de gás metano, utilizando água como fluido frio, entre os estágios de compressão. Na análise e projeto de um PCHE, existem vários parâmetros geométricos dos canais que exercem um papel essenciais para que este atinja o desempenho desejado. A Figura 1.1 é um exemplo de utilização da PCHE para resfriamento de Gás em um ciclo fechado, enquanto a água é aquecida em um ciclo aberto.

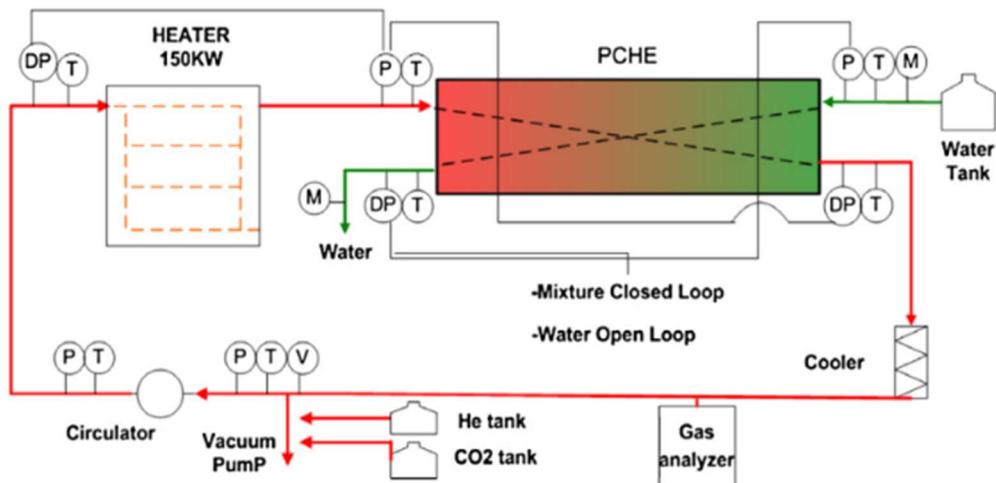


Figura 1.1 - PCHE numa planta de mistura de Gás-Água

Fonte: [24]

Existe uma ampla gama de tipos de trocadores de calor, pode-se citar como exemplo o tipo casco tubo (Figura 1.2) e o de placas paralelas (Figura 1.3).



Figura 1.2- Trocador de Calor Casco-Tubo

Fonte: (INCASE, 2020)



Figura 1.3- Trocador de Calor a Placas

Fonte: (APV Heat Exchanger Product Group, 2020)

O trocador de calor tipo impresso (Figura 1.4) é geralmente fabricado através de um processo de gravação fotoquímica. Comumente os canais nele gravados são da ordem de 0,5 ~ 2mm. O pequeno diâmetro hidráulico permite que o PCHE tenha as características de boa compactação e alta eficiência de transferência de calor. A densidade da área de transferência de calor do PCHE pode atingir o valor de 2500 m²/m³. Sob de mesmas condições de carga térmica,

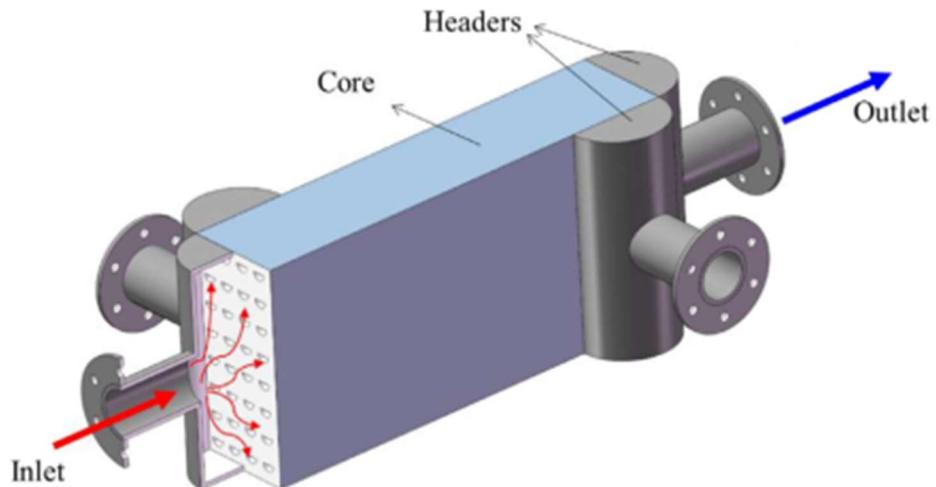


Figura 1.4- Diagrama esquemático de um PCHE

Fonte: [11]

o volume do PCHE é cerca de um quinto menor do que um trocador de calor tipo casco-tubo. Por causa dessas características o trocador de calor compacto alcança pequenas diferenças de temperaturas e reduz as irreversibilidades térmicas - T. Ma [1]. Na Figura 1.5 pode-se ver uma comparação entre as faixas de pressão e temperatura de operação possíveis para o PCHE e para os outros tipos de trocadores de calor padrões.

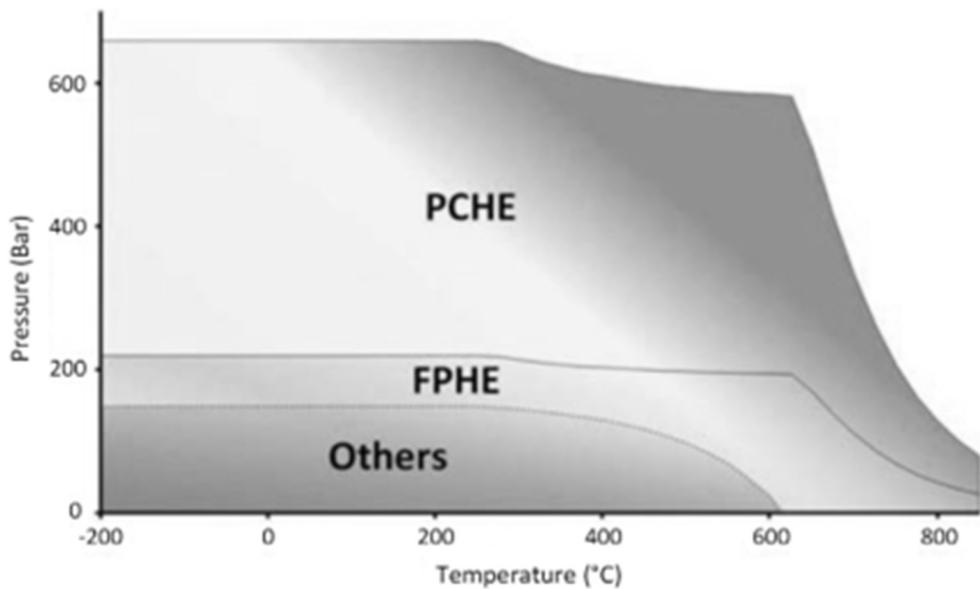


Figura 1.5- Capacidades de temperatura e pressão dos trocados de calor.

Fonte: [7]

O desenvolvimento de algoritmos computacionais para problemas de transferência de calor tem surgido como uma ferramenta de sucesso no campo de engenharia térmica. Os algoritmos de Inteligência artificial usados na área são modelos simplificados da inteligência humana e a evolução a partir da experiência. O aprimoramento na resolução de questões de troca térmica tem atraído a atenção de pesquisadores. Os métodos em geral usados para tratar desta questão incluem a utilização das equações fundamentais, empregando correlações ou desenvolvimento de modelos únicos oriundos de dados experimentais de tentativa e erro. Os problemas envolvendo transferência de calor também têm se tornado mais complexos e uma modelagem requer lidar com a performance, dinâmica do sistema, otimização e controle. Para superar essas dificuldades, a rede neural artificial tem sido implementada em várias pesquisas de transferência de calor baseado em dados experimentais disponíveis. ANN (Artificial Neural

Network) tem sido empregada com sucesso em diversas aplicações, como: energia solar; projeto de usinas de termoelétricas; estimava de carga térmica de aquecimento de prédios; recuperação de calor residual em trocadores de calor; Fluidização de Gás-Sólido. A vantagem de usar *ANNs* para simular processos térmicos é, uma vez treinado, o modelo representa uma forma rápida e confiável para prever a performance. A modelagem inclui numerosas vantagens como uma precisa aproximação para problemas complexos, maior eficiência do que modelos fenomenológicos, mesmo para múltiplos cálculos de resposta e maior eficácia até com dados de entrada incompletos e ruidosos. ANN não precisa de correlações ou método iterativo, ela precisa apenas de amostras de entrada/saída para treinar a rede neural. Os primeiros modelos de rede neural usam o princípio simples no qual o processo de aprendizagem do ser humano é indutivo, e que em cada passo de aprendizagem os erros são reduzidos e a precisão melhorada. Os modelos empíricos e correlações desenvolvidas por métodos convencionais são complexos, por natureza, difíceis de prever relações não lineares e exigem muitos recursos computacionais. A ANN fornece uma plataforma para resolver esses problemas térmicos, prevendo de maneira rápida e confiável a performance. As mudanças no sistema podem ser atualizadas continuamente e facilmente. Dentre as características mais relevantes das redes neurais podemos citar a capacidade de aprendizado por intermédio da aplicação de um método de treinamento, onde a rede consegue extrair o relacionamento existente entre as diversas variáveis que compõe a aplicação, e em seguida, tem-se a possibilidade de generalização deste conhecimento adquirido no treinamento, o que possibilita predizer comportamentos e soluções que eram, até então, desconhecidos.

Vale aqui comentar que existe uma demanda grande por projetos e análises dedicadas de PCHE, que podem ter a sua configuração geométrica otimizada de acordo com a aplicação e as condições operacionais. Justificando assim, uma abordagem via simulação computacional do projeto ótimo destes dispositivos. Todavia, as simulações computacionais com modelos completos (3D, transiente, em regime turbulento) pode ser por vezes cara computacionalmente, e que a depender da aplicação, tipo de fluidos, e faixas de operação não se tem disponível na literatura correlações para uso mais imediato e facilitado. Deste modo, tem-se aqui configurada uma possibilidade de emprego das redes neurais na previsão do comportamento destes sistemas. Vale ressaltar, que a rede neural também pode ser usada para encontrar um melhor ajuste de dados, aproximação, reconhecimento de padrão, agrupamento de dados, reconhecimento de

imagem, classificação de imagem, extração de características das imagens, redução de ruído, extrapolação, modelagem dinâmica e previsão.

1.2 Objetivos

Este trabalho tem como objetivo aplicar uma abordagem via inteligência artificial para resolver problemas de engenharia envolvendo um tipo específico de trocador de calor: trocador de calor de Circuito Impresso. A Inteligência Artificial será utilizada para prever os impactos na performance do trocador devido a mudanças na geometria do trocador ou mudanças nas condições de operação. Neste trabalho será avaliado a efetividade das soluções obtidas por meio de Inteligência artificial e seus resultados serão comparados com correlações. Todo o código foi desenvolvido em nuvem pelo Google Colab, através de uma GPU do Google, na linguagem de programação Python, utilizando as bibliotecas TensorFlow e Pandas.

1.3 Organização do trabalho

No capítulo 2 é feita uma revisão bibliográfica, onde foram citados os principais trabalhos que contribuíram para a realização deste projeto e suas contribuições destacadas.

Em seguida, no capítulo 3 tem por finalidade explicar os conceitos utilizados em Inteligência Artificial e apresentar os fundamentos teóricos que descrevem o comportamento de trocadores de calor.

Ainda, no capítulo 4, são expostos os resultados e discussões oriundas das simulações das redes neurais artificiais.

Por fim, no Capítulo 5 são discutidas as conclusões sobre o projeto como um todo e sugestões para trabalhos futuros.

2. Revisão Bibliográfica

Neste capítulo será apresentada uma revisão bibliográfica reunindo os principais trabalhos que contribuíram para o presente estudo, abordando os temas de interesse. O capítulo é dividido em 2 partes, sendo a primeira destinado a explicar sobre os trocadores de calor de circuito impresso (PCHE) e a segunda reúne as informações acerca da utilização de inteligência artificial em trocadores de calor, especificamente redes neurais.

2.1 Trocador de calor tipo circuito impresso (PCHE)

Para analisar do trocador de calor com a finalidade de propor novas correlações é preciso fazer experimentos ou simulações. T. Ma e colaboradores [1] montou um esquema, representado na Figura 2.1, o qual mostra uma arquitetura do trocador de calor: Seguindo o esquema proposto por T. Ma [1], as camadas podem ser montadas de forma que o canal quente fique exatamente encima do frio (a); pode-se montar de forma a alterar o formato e dimensões dos canais do fluido quente e frio (c); comprimento do canal (b); orientação do fluxo dos fluidos, determinar os ângulos de zigue-zague e passo do canal (Figura 2.1). Nesse trabalho não serão vistos os canais reproduzidos na Figura 2.2, contudo existem estudos para determinar correlações para os mais variados tipos de formatos e geometrias de canais como aerofólio e formato em S.

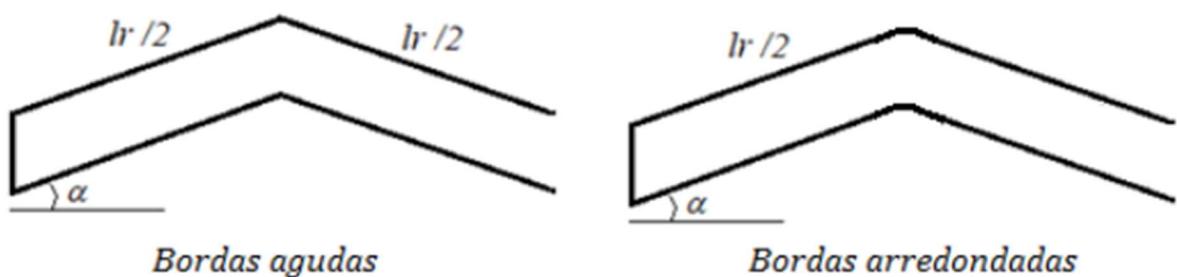


Figura 2.1 - Zigue-zague de borda aguda e borda arredondada.

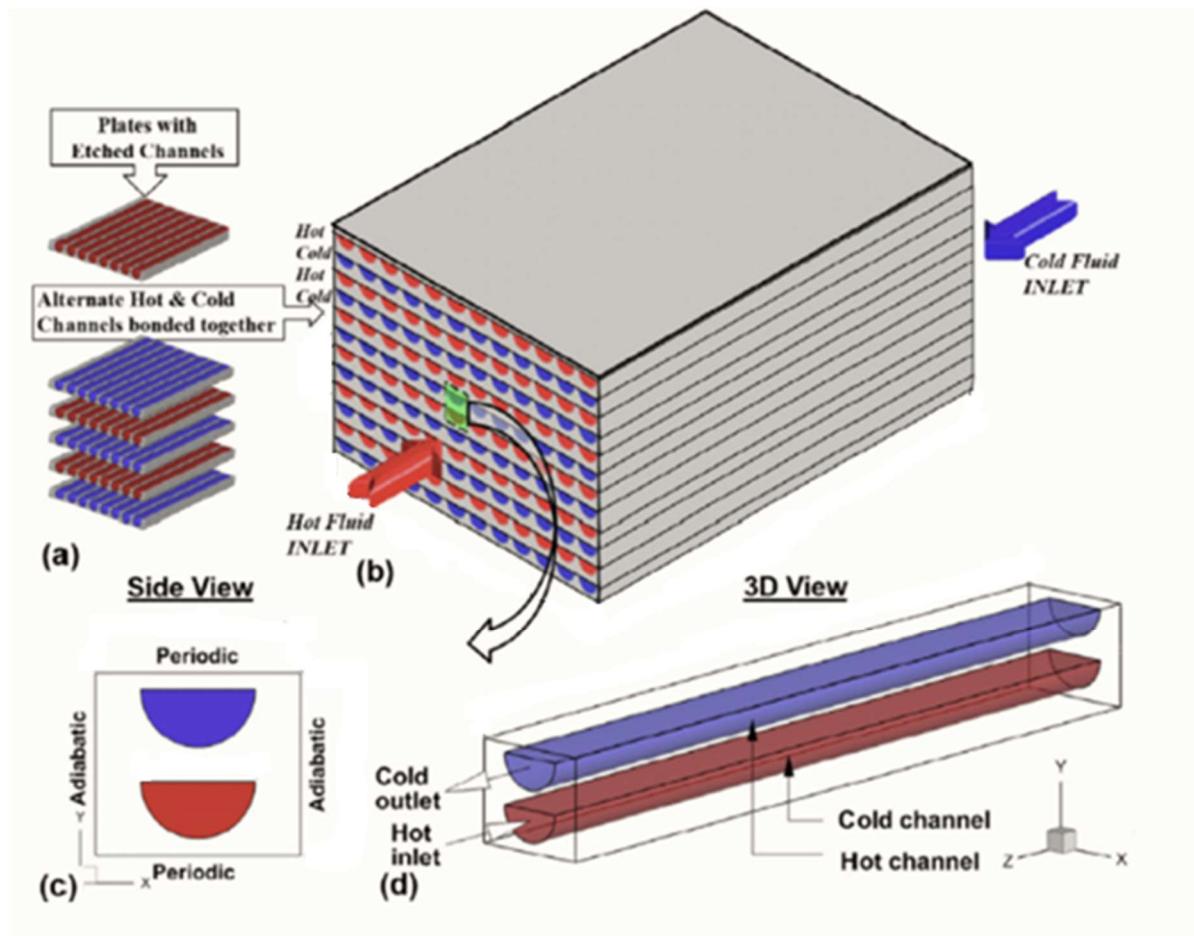


Figura 2.2- Trocador de calor tipo circuito impresso (PCHE).

Fonte: [1] (Adaptado)

Na literatura se encontra uma ampla quantidade de correlações de Nusselt e Fator de atrito de Fanning para várias condições de operação. Esses parâmetros de operação incluem a escolha do tipo de fluido frio e quente, incluir aspectos geométricos do trocador de calor, tipo de escoamento no canal etc.

Nas PCHEs com canal reto são aplicadas as correlações clássicas da literatura, mesmo com diferentes seções transversais. Contudo, para diferentes tipos de canais, é comum que se desenvolva novas correlações que sejam mais adequadas para cada caso.

A Figura 2.3 reúne uma série de correlações de diferentes autores em condições de operação distintas. De imediato é possível notar a similaridade da forma das correlações, apesar dos coeficientes serem diferentes. Ainda que as correlações de Nusselt sejam ótimas ferramentas na caracterização de trocador de calor, essas possuem limitações ao seu uso.

Ao trabalhar com trocadores de calor, por exemplo a alteração da geometria para atender demandas do projeto, e que seja necessário conhecer as correlações, duas opções podem ser executadas. A primeira, por simplicidade, pode-se basear em alguma correlação válida. Cabe ao responsável ponderar sobre os limites de validade e se outros parâmetros podem, ou não, serem desconsiderados. Se o caso em análise for muito diferente daquele que a correlação foi construída, uma proposição pode ser necessária, utilizando dados experimentais ou de simulação.

Channel Configuration	Description	Heat transfer and pressure drop correlations
Zigzag channel (40^0) Ishizuka et al. [1]		$h = 0.2104 Re + 44.16$ $f = -2 \times 10^{-6} Re + 0.1023$ $(5000 \leq Re \leq 13000)$ $Nu = 0.0188 Re^{0.8742}$ $f = 0.2881 Re^{-0.1322}$ $(2000 \leq Re \leq 55000)$ $(0.7 \leq Pr \leq 1.0)$ $Nu = 0.1696 Re^{0.629} Pr^{0.371}$ $f = 0.1924 Re^{-0.091}$ $(3500 \leq Re \leq 22000)$ $(0.7 \leq Pr \leq 2.2)$
Zigzag channel (40^0) Kim et al. [10]		
Zigzag channel (52^0) Ngo et al. [19]		
S-shape channel Ngo et al. [19]		$Nu = 0.174 Re^{0.539} Pr^{0.43}$ $f = 0.4545 Re^{-0.34}$ $(3500 \leq Re \leq 23000)$ $(0.7 \leq Pr \leq 2.2)$
Airfoil channel Yoon et al. [20]		$Nu = 0.027 Re_{min}^{0.78} Pr^{0.4}$ $f Re_{min} = 0.931 + 0.028 Re_{min}^{0.86}$ $(3000 \leq Re \leq 1.5 \times 10^5)$ $(0.6 \leq Pr \leq 0.8)$

Figura 2.3- Exemplo de Correlações propostas para diferentes tipos de canais usados em PCHEs.

Fonte: [12]

No trabalho A. C. P. d. Oliveira e colaboradores [2] é possível encontrar a Tabela 2.1, a qual contem correlações para PCHE com seções semicirculares; com diferentes ângulos de ziguezague; com diferentes fluidos. Nota-se que cada correlação tem uma faixa de válida de específica.

No trabalho A. MESHRA, et al [3] podemos ver a aplicação de correlações para um trocado de calor com fluido frio CO₂ e quente CO₂, com um ângulo fixo de zigue-zague de 36°. As Equações 2.1 e 2.2, são respectivamente correlação para o Nusselt e para o Fator de atrito.

A. MESHRA [3] também disponibiliza um gráfico no qual é possível visualizar a influência do ângulo de zigue-zague no valor de Nusselt Figura 2.4(a) e do Fator de atrito Figura 2.4(b).

$$Nu_h = 0,0174Re^{0,893}Pr^{0,7} \quad (2.1)$$

$$f = 0,867Re^{-0,522} + 0,040 \quad (2.2)$$

$$470K < T_b < 630K ; 5.000 \leq Re \leq 32.000;$$

Tabela 2.1- Correlações numéricas para PCHE.

Autor	Correlação Nu e f	Faixas de validade	Fluido	Ângulo	Seção transversal
Berbish <i>et al.</i> [30]	$Nu = 0,0228Re^{0,8}$ $f = 0,487Re^{-0,26}$	$8242 \leq Re \leq 57794$	Ar	0° (reto)	Semicirc.
Chen <i>et al.</i> [37]	$Nu = (0,01352)Re^{0,080058}$ $Nu = (3,6361 \times 10^{-4})Re^{1,2804}$	$1200 \leq Re \leq 1850$ $1850 < Re \leq 2900$	He-He	15°	Semicirc.
Chen <i>et al.</i> [37]	$Nu = (0,047516)Re^{0,633151}$ $Nu = (3,680123 \times 10^{-4})Re^{1,282182}$	$1200 \leq Re \leq 1850$ $1850 < Re \leq 2900$	He-He	15°	Semicirc.
Chen <i>et al.</i> [36]	$Nu = (0,05516)Re^{0,69195}$ $f = 17,639Re^{-0,8861}$ $Nu = (0,09221)Re^{0,62507}$ $f = 17,639Re^{-0,8861}$	$1400 < Re \leq 2200$ $2200 < Re \leq 3558$	He-He	15°	Semicirc.
I. Kim <i>et al.</i> [29]	$Nu = 3,255 + 0,00729(Re - 350)$ $f \cdot Re = 16,51 + 0,01627Re$	$350 < Re < 800 \quad Pr = 0,66$ $350 < Re < 1200$	He-He	15°	Semicirc.
I. Kim <i>et al.</i> [34]	$Nu = 4,089 + 0,00365RePr^{0,58}$ $f \cdot Re = 15,78 + 0,0487Re^{0,84}$ $Nu = 4,089 + 0,00497Re^{0,95}Pr^{0,55}$ $f \cdot Re = 15,78 + 0,0557Re^{0,82}$	$0 < Re < 2500$ $0,66 < Pr < 13,41$ $0 < Re < 3000$ $0,66 < Pr < 13,41$	He-H ₂ O He-CO ₂	15°	Semicirc.

Fonte: [2]

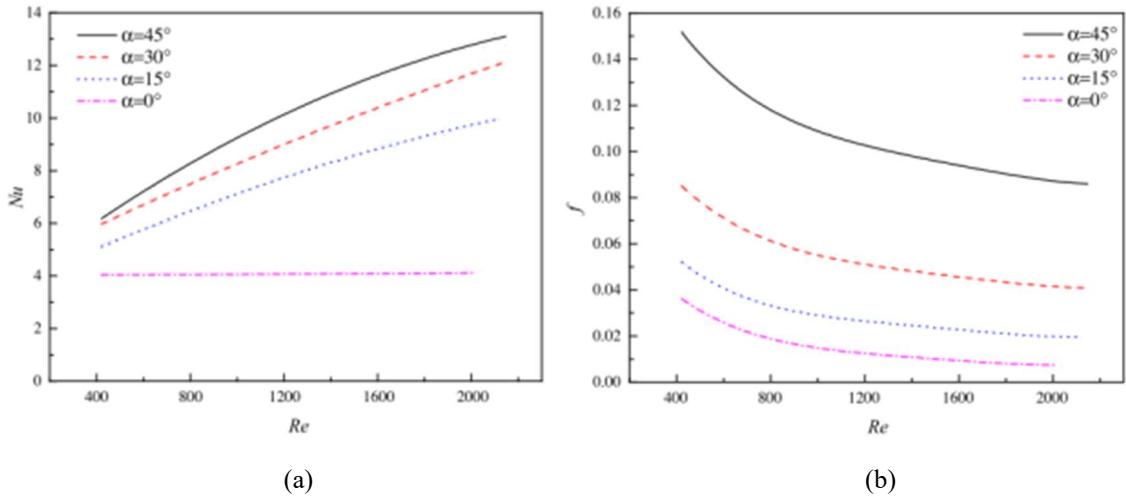


Figura 2.4- Variação do número de Nusselt e fator de atrito locais do canal quente com o número de Reynolds para cada ângulo.

Fonte: [20]

Os aspectos geométricos não estão explícitos como variáveis nas correlações, mas são descritos na obtenção delas. Portanto utilizar a equação acima para situação distintas pode acarretar resultados insatisfatórios.

O trabalho de A. T. d. Silva [4] descreve como são determinadas as formas das correlações e mostra o método para calcular o coeficiente. O autor explica que a obtenção dessas expressões é feita a partir de dados gerados experimentalmente ou por simulação e que, por causa da distribuição de dados, a maioria dos autores escolhem uma forma exponencial que no geral tem a seguinte forma:

$$Nu = \alpha Re^\beta Pr^\gamma \quad (2.3)$$

As letras gregas α , β e γ são coeficientes que se ajustam. Vale ressaltar que as equações não se limitam a está forma, pode haver modificações a fim de agregar mais informações às suas expressões. Uma vantagem das redes neurais é que elas podem ter milhares de coeficientes e não é preciso definir um formato para a equações, durante o treinamento da rede neural essas relações não lineares vão se otimizando para encontrar a melhor “formato” entre os parâmetros.

2.2 Redes Neurais

No trabalho de M. Venturini [5] é esclarecido que rede neurais com múltiplos *outputs* possui uma performance melhor do que se fosse construída para ANN específica para cada tipo de *output*, portanto é indicado a construção de um modelo que inclua treinar simultaneamente o fato de atrito e o nusselt.

O trabalho de H. Asgari [6] compara o controle de sistema de uma turbina de gás entre PID e ANN e apresenta um modelo para iterativo para selecionar a rede neural com o menor erro.

O trabalho de S.-J. Yoon, et al [7] mostra que para projetar e avaliar o desempenho termo-hidráulico dos trocadores de calor PCHE é necessário incluir os fatores geométricos no desenvolvimento das correlações para o Fator de atrito e para o número de Nusselt. Na pesquisa os autores concluíram que o fator de atrito do canal zigue-zague é influenciado principalmente pela geometria do canal zigue-zague, enquanto o número de Nusselt é influenciado pelo design geral do trocador de calor.

A pesquisa de T. Ma [1] mostrou um estudo, através de CFD, sobre as características dinâmicas de um PCHE localizado numa planta de carvão de 1000 MW S-CO₂, incluindo: As temperaturas de saída do trocador de calor; fluxo de calor total e coeficiente de calor das superfícies dos canais onde passaram os fluídos. Por fim, o autor utilizou uma rede neural para prever a performance do trocador de calor.

O trabalho de E. Reynoso-Jardón [8] fez um estudo sobre trocadores de calor e aplicou rede neural para prever a performance, queda de pressão e dinâmica de controle térmica do sistema. O trabalho utilizou simulações de CFD para gerar os dados de treinamento da rede neural e inclui nas variáveis o tipo de material dos trocadores de calor. Foram utilizados 4 tipos de materiais: aço inoxidável, cobre, alumínio e titânio. As seções transversais e longitudinais foram modificadas, gerando 143 geometrias para análise. Uma importante informação no trabalho de E. Reynoso-Jardón [8] foi que duas camadas intermediárias com uma quantidade adequada de neurônios são capazes de estimar aproximadamente qualquer tipo de função não linear. O máximo desvio entre o valor previsto e o valor de referência do coeficiente calor global foi abaixo de 0,4; 0,1; 0,2% para o cobre, aço inoxidável e titânio, respectivamente; enquanto para queda de pressão, o maior desvio foi abaixo de 0,1; 0,6; 0,8% para os mesmos materiais.

Por fim o autor concluiu que a abordagem por ANN contém capacidades notáveis para generalizar a complexa dependência entre as varáveis.

O trabalho de F. Diez [9] faz uma análise detalhada de um coletor solar de placa plana com três fluidos diferentes e utilizou as redes neurais para modelar o problema. A *ANNs* foi construída para determinar a temperatura de saída e teve como input vários fatores, como: medidas da irradiação solar; temperatura ambiente; temperatura de entrada e fluido de trabalho. Após vários testes com diferentes arquiteturas de rede, a inteligência artificial apresentou bons resultados. A Figura 2.5 mostra os testes realizados pelo F. Diez [9] com diferentes números de neurônios para a camada intermediária. A quantidade de neurônios variou de 1 a 50, e verificou-se que o RMSE não diminuiu à medida que a quantidade de neurônios aumenta. Essa verificação no qual o erro não necessariamente diminui à medida que a quantidade de neurônio cresce também será observada nos casos envolvendo a construção do modelo de previsão da PCHE.

Neuronal network architecture solar collector model	
Neurons input-hidden-output	RMSE (°C) Outlet temperature $T_{out}(t + 1)$
4-1-1	0.23
4-2-1	0.23
4-3-1	0.20
4-4-1	0.23
4-5-1	0.20
4-6-1	0.26
4-7-1	0.23
4-8-1	0.27
4-9-1	0.23
4-10-1	0.24
4-20-1	0.22
4-30-1	0.23
4-40-1	0.24
4-50-1	0.25

Figura 2.5- Valor obtido do RMSE com uma rede neural com diferente número de neurônios na camada intermediária.

Fonte: [9]

3. Fundamentação Teórica

A inteligência artificial (IA) está remodelando economias, prometendo gerar ganhos de produtividade, melhorar a eficiência e reduzir custos. Contribuir para uma vida melhor e ajudar as pessoas a fazer melhores previsões e a tomar decisões mais informadas. Essas tecnologias, no entanto, ainda estão na infância e as promessas precisam enfrentar os desafios globais e promover a inovação e o crescimento. Como os impactos da IA permeiam nossas sociedades, seu poder de transformação deve ser colocado a serviço das pessoas e do planeta.

O panorama técnico da inteligência artificial (IA) evoluiu significativamente desde 1950, quando Alan Turing levantou a questão de saber se as máquinas podem pensar. Cunhado como um termo em 1956, a IA evoluiu da IA simbólica, onde os humanos construíam sistemas baseados em lógica, durante o “inverno” da IA dos anos 1970 até o computador jogador de xadrez Deep Blue nos anos 90. Desde 2011, avanços em aprendizado de máquina ou *machine learning* (ML), um subconjunto de IA que usa uma abordagem estatística, têm aprimorado a capacidade das máquinas de fazer previsões a partir de dados históricos. Uma técnica de modelagem de ML bem consolidada atualmente se chama redes neurais, reúne grandes conjuntos de dados e poder de computação, e está por trás da expansão no desenvolvimento de IA. As fases do ciclo de vida do sistema de IA são: i) planejamento e design, coleta e processamento de dados, e construção e interpretação de modelos; ii) verificação e validação; iii) implementação; e iv) operação e monitoramento (Figura 3.1).

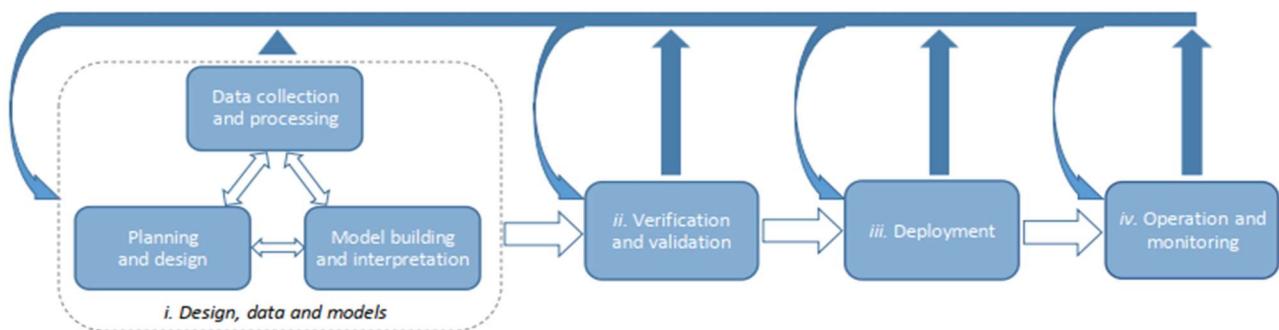


Figura 3.1- Fluxograma IA

Fonte: (OECD, 2019)

3.1 Aplicações de IA

Aplicações de IA estão experimentando rápida aceitação em uma série de setores onde é possível detectar padrões em grandes volumes de dados e modelar sistemas complexos e interdependentes para melhorar a tomada de decisão e economizar custos.

- No setor de transportes, veículos autônomos com sistemas de motoristas virtuais, mapas de alta definição e rotas de tráfego otimizadas, todos prometem custo, segurança e qualidade de vida e benefícios ambientais.
- A pesquisa científica usa IA para coletar e processar dados em grande escala, para ajudar a reproduzir experimentos e reduzir seu custo, e para acelerar a descoberta científica.
- Na saúde, os sistemas de IA ajudam a diagnosticar e prevenir doenças e surtos precocemente, descobrir tratamentos e drogas, propor intervenções sob medida e ferramentas de automonitoramento de energia.
- Na justiça criminal, a IA é usada para policiamento preditivo e avaliação do risco de reincidência.
- Os aplicativos de segurança digital usam sistemas de IA para ajudar a automatizar a detecção de e resposta a ameaças, cada vez mais em tempo real.
- Aplicações de IA na agricultura incluem monitoramento e previsão da saúde do solo e da cultura o impacto dos fatores ambientais no rendimento das culturas.
- Serviços financeiros alavancam IA para detectar fraude, avaliar a capacidade de crédito, reduzir custos de atendimento ao cliente, automatizar a negociação e apoiar a conformidade legal.
- Em marketing e publicidade, a IA extrai dados sobre o comportamento do consumidor para direcionar e personalizar conteúdo, publicidade, bens e serviços, recomendações e preços

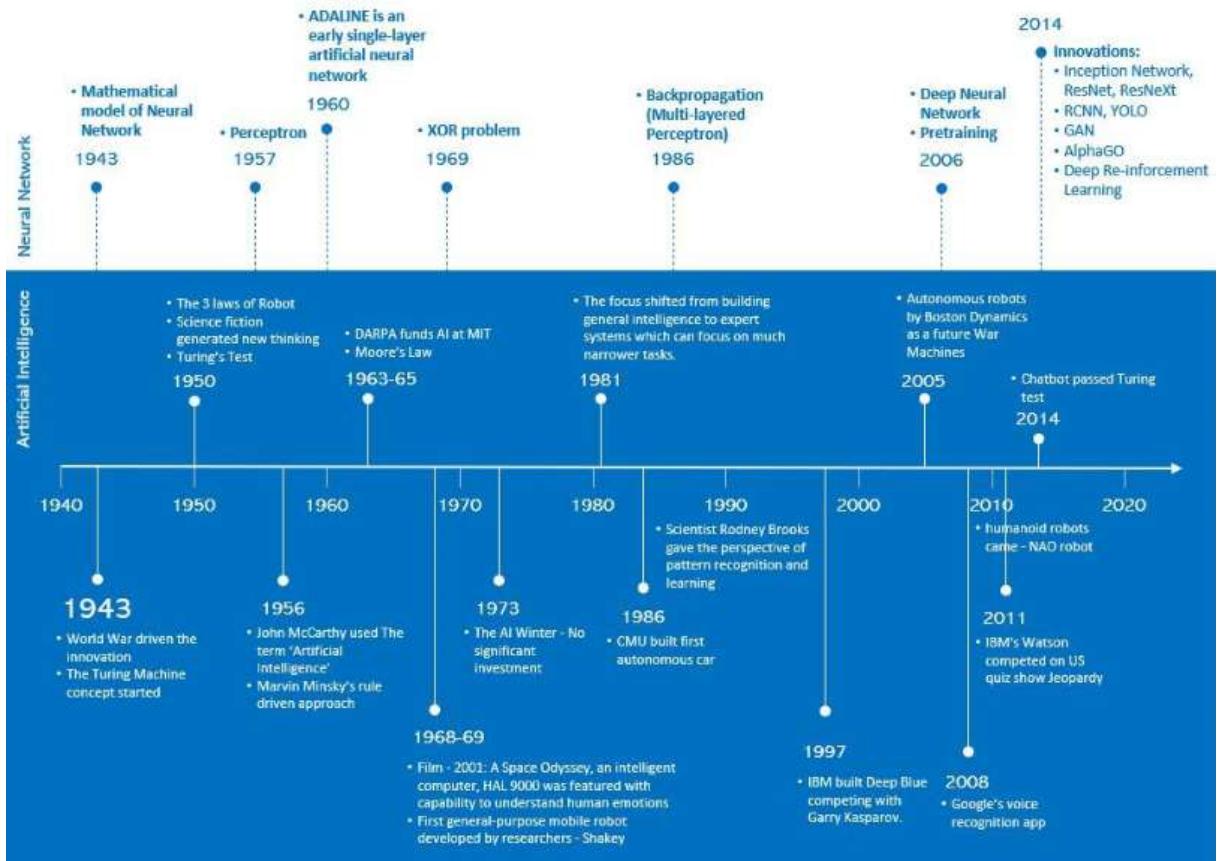


Figura 3.2- Perspectiva histórica da Inteligência Artificial

Fonte: Em nota de rodapé¹. Acesso em maio de 2021

3.2 Aprendizado de Máquina

Os sistemas de IA usam cada vez mais o ML. Este é um conjunto de técnicas que permitem que as máquinas aprendam de uma maneira automatizada e por meio de padrões e inferências, em vez de instruções explícitas de um humano. As abordagens de ML muitas vezes ensinam as máquinas a alcançar um resultado, mostrando-lhes muitos exemplos de resultados corretos. No entanto, eles também podem definir um conjunto de regras e permitir a máquina aprender por tentativa e erro. ML geralmente é usado na construção ou ajuste de um modelo,

¹ <https://medium.com/@amritenduroy/a-perspective-on-the-history-of-artificial-intelligence-ai-13fbb5a53ec7>

mas também pode ser usado para interpretar os resultados de um modelo (Figura 3.3). ML contém várias técnicas que têm sido usados por economistas, pesquisadores e tecnólogos há décadas. Estes variam de regressões lineares e logísticas, árvores de decisão e análise de componentes principais para redes neurais profundas.

A verdadeira tecnologia por trás da onda atual de aplicações de ML é uma sofisticada técnica de estatística de modelagem denominada redes neurais. Esta técnica é acompanhada por crescimento poder computacional e a disponibilidade de conjuntos de dados massivos (*big data*). Redes neurais envolvem interconectar repetidamente milhares ou milhões de transformações simples em uma máquina estatística maior, que pode aprender relações sofisticadas entre os dados.

Em outras palavras, as redes neurais modificam seu próprio código para encontrar e otimizar conexões entre as entradas e saídas. Por fim, a aprendizagem profunda (*Deep learning*) é uma frase que se refere a particularmente grandes redes neurais; não há um limite definido de quando uma rede neural se torna "profunda".

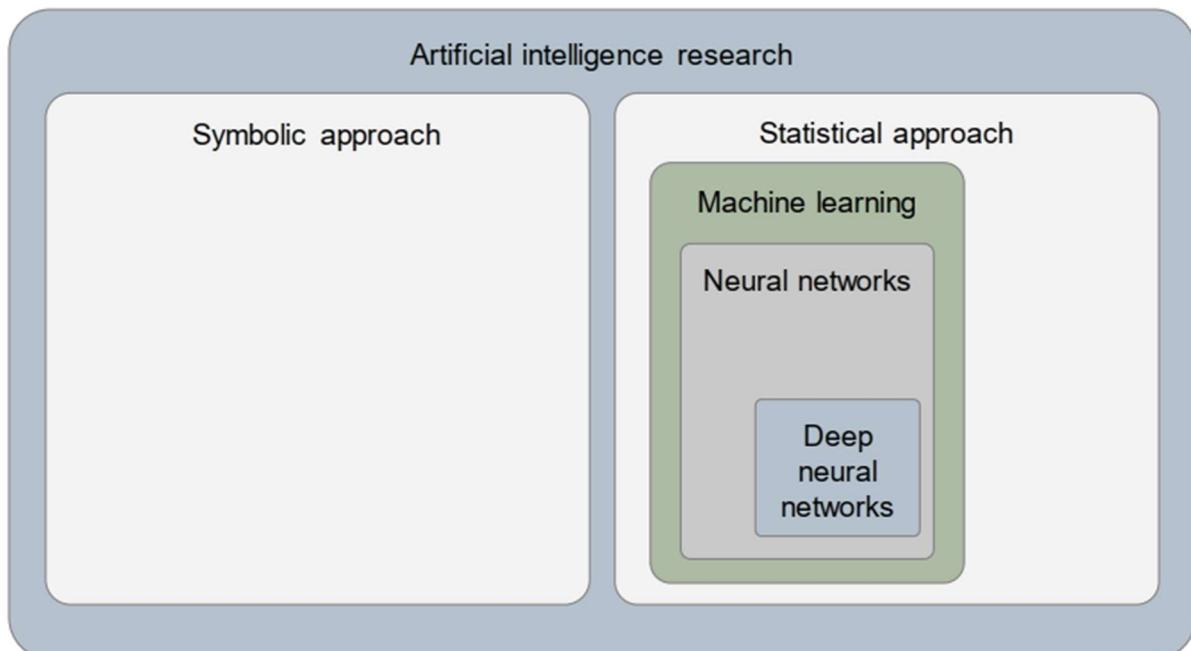


Figura 3.3- Relação entre IA e ML

Fonte: Massachusetts Institute of Technology (MIT)'s Internet Policy Research Initiative (IPRI).

Esta dinâmica em evolução na pesquisa de IA está associada a avanços contínuos em habilidades de computação, disponibilidade de dados e projeto de rede neural. Juntos, eles significam que a abordagem estatística da IA provavelmente continuará como uma parte importante da pesquisa no curto prazo.

Na Tabela 3.1 consta um rol exemplificativo de técnicas e modelos de Aprendizagem de Máquina bastante estudado e aplicado.

Tabela 3.1 – Rol exemplificativo de Técnicas e Modelos de ML

MACHINE LEARNING	Técnicas	Deep learning
		Simulation-based learning
		Crowdsourcing and human computation
		Evolutionary computing
		Techniques beyond neural networks
Modelos	Supervised learning	
	Reinforcement learning	
	Generative models/generative adversarial networks	

Fonte: (MIT's IPRI).

3.3 Fundamentos Redes Neurais Artificiais

3.3.1 Redes Neurais Biológicas e Redes Neurais Artificiais

Redes Neurais Artificiais são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência. Uma grande rede neural artificial pode ter centenas ou milhares de unidades de processamento; já o cérebro de um mamífero pode ter muitos bilhões de neurônios.

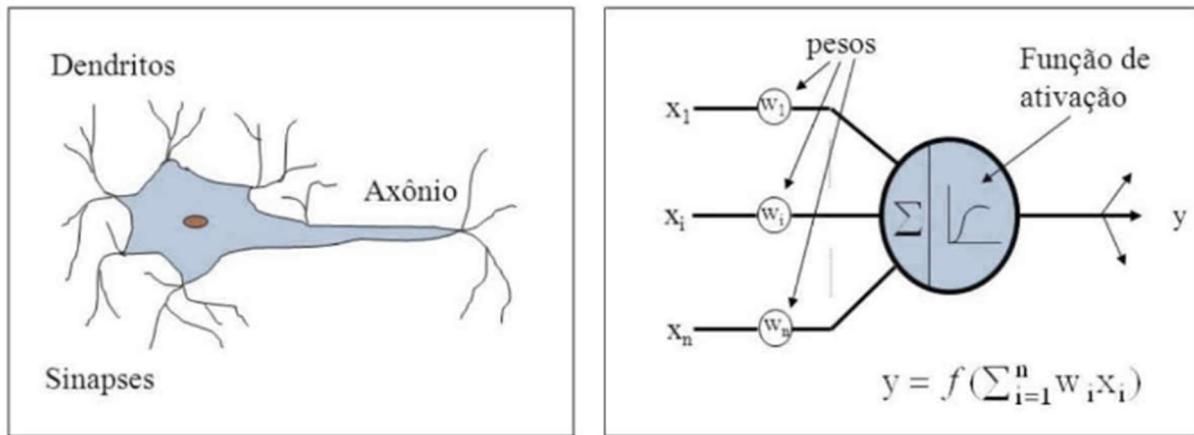


Figura 3.4- Neurônio Biológico vs artificial

Fonte: Em nota de rodapé². Acesso em mar. de 2021

O sistema nervoso é formado por um conjunto extremamente complexo de células, os neurônios. Eles têm um papel essencial na determinação do funcionamento e comportamento do corpo humano e do raciocínio. Os neurônios são formados pelos dendritos, que são um conjunto de terminais de entrada, pelo corpo central, e pelos axônios que são longos terminais de saída.

Os neurônios se comunicam através de sinapses. Sinapse é a região onde dois neurônios entram em contato e através da qual os impulsos nervosos são transmitidos entre eles. Os impulsos recebidos por um neurônio A, em um determinado momento, são processados, e atingindo um dado limiar de ação, o neurônio A dispara, produzindo uma substância neurotransmissora que flui do corpo celular para o axônio, que pode estar conectado a um dendrito de um outro neurônio B. O neurotransmissor pode diminuir ou aumentar a polaridade da membrana pós-sináptica, inibindo ou excitando a geração dos pulsos no neurônio B. Este processo depende de vários fatores, como a geometria da sinapse e o tipo de neurotransmissor.

² <https://www.infoq.com.br/presentations/classificacao-de-padroes-uma-abordagem-pratica-com-redes-neurais-artificiais/>

Em média, cada neurônio forma entre mil e dez mil sinapses. O cérebro humano possui cerca de 10 E11 neurônios, e o número de sinapses é de mais de 10 E14, possibilitando a formação de redes muito complexas.

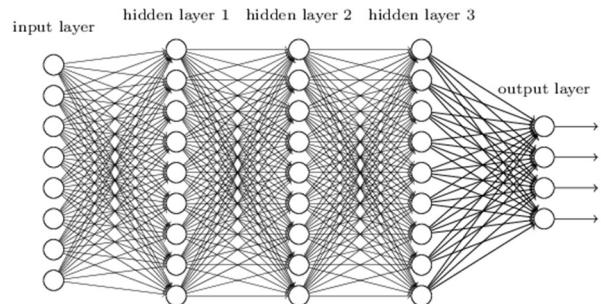
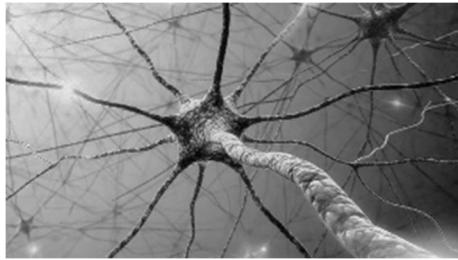


Figura 3.5- Comparaçāo entre as conexōes em um sistema biológico e em um sistema artificial.

Fonte: Em nota de rodapé:³ Acesso em mar. 2021

Fonte: Em nota de rodapé:⁴ Acesso em mar. 2021

Uma rede neural artificial é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Essas unidades, geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede.

3.3.2 Treinamento e validação da Rede

Treinamento ou aprendizagem para as redes neurais podem ser classificadas em aprendizado supervisionado e não supervisionado. No aprendizado supervisionado, *inputs* e *outputs* são conhecidos e o modelo ANN é treinado de maneira a mapear esses *inputs* e *outputs*. Aprendizado supervisionado é empregado para propósitos de regressão e classificação. Entretanto, o aprendizado não supervisionado, os *outputs* são desconhecidos e as relações

³ <https://olhardigital.com.br/noticia/redes-neurais-estao-apenas-no-comeco-diz-microsoft/54378>

⁴ <http://deeplearningbook.com.br/introducao-as-redes-neurais-convolucionais/>

ocultas no conjunto de dados é descoberto pelo ANN usando o método de agrupamento de dados. O aprendizado não supervisionado é usado para filtragem e agrupamento de dados.

O treinamento (otimização) pode ser feito por diferentes métodos, dentre eles pode-se citar: métodos baseados em gradiente descendente; aprendizado competitivo (redes auto-organizáveis); algoritmo genético e concorrentes etc.

É possível, ainda, dividir o treinamento em: incremental e batch (lote). No treinamento incremental, os pesos e vieses da rede são atualizados a cada vez que um input está presente na rede. No treinamento por batch, os pesos e vieses somente são atualizados após todos os *inputs* do batch passarem pela rede.

O treinamento da ANN envolve a variação de vários parâmetros. Por exemplo, é preciso mudar o número de neurônios das camadas intermediárias com objetivo de atingir a melhor convergência. O número de neurônios indica a complexidade do problema envolvido. Sempre é de interesse selecionar a estrutura da rede neural mais simples, com o menor número de parâmetros de *input*. O procedimento da rede neural artificial é em geral baseado numa aproximação iterativa, na qual os parâmetros são sucessivamente atualizados. Em cada etapa, o resultado é comparado com o valor de referência, e certos parâmetros são alterados para diminuição do erro.

Antes do treinamento começar, os dados podem ser divididos em 3 subconjuntos, conjunto de treino, teste e validação. O conjunto de treino, como o nome diz ele é utilizado para treinar a rede; o segundo, é usado para o cálculo do erro em cada época; o terceiro, é utilizado quando a rede neural já está finalizada e serve para o teste final.

3.3.3 Regularização dos dados

Em geral, os algoritmos de aprendizagem se beneficiam da padronização do conjunto de dados. Se alguns pontos fora de série (*outlier*) estiverem presente no conjunto, os métodos de regularização podem ser usados. A depender do tipo de conjunto de dados, pode-se escolher métodos que alteram a escala dos dados; métodos que transformam os dados ou até normalizar o conjunto.

A funções de MinMax é um tipo de função que altera a escala dos dados, baseado no limite superior e inferior, e equação 4 e 5 mostram como são feitos os cálculos. Existem outras equações de escala, entretanto elas possuem o mesmo princípio.

$$X_{std} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

$$X_{scaled} = X_{std}(X_{max} - X_{min}) + X_{min} \quad (3.2)$$

A função *Standardization*, alterar os dados de forma que a média seja centrada no zero e a variância seja unitária. A equação 6 mostra o cálculo.

$$X_{scaled} = \frac{X_i - \mu}{\sigma} \quad (3.3)$$

Se o conjunto de dados possuir muitos *outliers* a função de *standardization*, talvez, não funcione muito bem. A função *RobustScaler* é mais indicada.

Além destes, existem transformações não lineares, podem ser citadas a transformação por quantil e a *power*. Ambas são baseadas em transformações monotônicas dos recursos e, portanto, preservam a classificação dos valores ao longo de cada recurso.

As transformadas de potência *powertransform* são uma família de transformações paramétricas monotônicas que visam mapear dados de qualquer distribuição almejando uma distribuição gaussiana, a fim de estabilizar a variância e minimizar a assimetria.

Atualmente as transformados de potência fornecem duas transformações, “Yeo-Johnson” e a “Box-Cox”.

“Yeo-Johnson” é dada por:

$$x_i^{(\lambda)} = \begin{cases} \frac{(x_i + 1)^\lambda - 1}{\lambda} & \text{se } \lambda \neq 0, x_i \geq 0, \\ \ln(x_i + 1) & \text{se } \lambda = 0, x_i \geq 0 \\ \frac{(-x_i + 1)^{2-\lambda} - 1}{\lambda} & \text{se } \lambda \neq 2, x_i < 0, \\ -\ln(-x_i + 1) & \text{se } \lambda = 2, x_i < 0 \end{cases} \quad (3.4)$$

Transformada de “Box-Cox” é dada por:

$$x_i^{(\lambda)} = \begin{cases} \frac{x_i^\lambda + 1}{\lambda} & \text{se } \lambda \neq 0, x_i \geq 0, \\ \ln(x_i) & \text{se } \lambda = 0, x_i \geq 0 \end{cases} \quad (3.5)$$

“Box-Cox” somente pode ser aplicada para dados estritamente positivos. Em ambos os métodos, a transformação é parametrizada pelo λ , que é determinado por meio da máxima verossimilhança *Maximum likelihood estimation*.

3.3.4 Número de camadas intermediárias e Neurônios

Escolher o valor certo para o número de camadas intermediárias e a quantidade de neurônios disponível por camada é vital para o treinamento da ANN. Já é demonstrado que mapeamento multidimensional não linear de qualquer função contínua pode ser resolvido por uma rede com duas camadas intermediárias com o número adequado de neurônios. Portanto, a tarefa principal é determinar a quantidade de neurônios nas camadas. Embora aumentar a quantidade de neurônios seja necessário para capturar as não linearidades dinâmicas do sistema, isto não significa que sempre esse aumento irá melhorar a acurácia e a capacidade de generalização.

3.3.5 Algoritmos para treinamento

Diferentes algoritmos de treinamento são utilizados para treinar as *ANNS*. Alguns dos algoritmos de treinamento disponíveis no Tensorflow são:

- SGD
- Adam

A otimização de Adam é um método de gradiente descendente estocástico que se baseia na estimativa adaptativa de momentos de primeira e segunda ordem

- Adamax

É uma variante de Adam baseada na norma do infinito. Adamax às vezes é superior ao adam, especialmente em modelos com *embeddings*.

- Adagrad

É um otimizador com específicas taxas de aprendizagem, que são adaptadas em relação à frequência com que um parâmetro é atualizado durante o treinamento.

Quanto mais atualizações um parâmetro recebe, menores são as atualizações.

- RMSprop

Em resumo, é uma extensão do Adagrad que lida com taxas de aprendizagem radicalmente decrescentes.

- Nadam

Assim como Adam é essencialmente RMSprop com momento, Nadam é Adam com momento de Nesterov.

Além disso, a taxa de aprendizagem das redes neurais deste trabalho foi obtida de forma automática, não foi necessário realizar incrementos ou alterações nela.

3.3.6 Função Transferência

Funções transferência de ativação transformam o nível do neurônio em um *output*. Existem vários tipos dessas funções sendo utilizadas. As funções mais comuns de serem utilizadas na função transferência são a sigmoid, a Tanh e a ReLu. Essas funções são diferenciáveis e reproduzem a não linearidade dos sistemas. A Figura 3.6 mostra diferentes funções transferência que podem ser utilizadas para treinar uma rede neural.

Ao todo, nesse trabalho, dez tipos de função transferência foram utilizadas para teste: relu; sigmoid; tanh; swish; elu; gelu; selu; softplus; softsign e softmax. As funções mais comuns de serem usadas na função transferência são: tangente hiperbólica (Tanh), Sigmoid e Relu. Se os valores de *input* ou de *output* alternam entre valores negativos e positivos, a função Tanh é recomendável. O gráfico das funções de transferência pode ser observado na Tabela 3.2.

Tabela 3.2– Funções de ativação usualmente utilizada

TIPO DE FUNÇÃO	EQUAÇÃO
Sigmoid	$f(x) = \frac{1}{1 + e^{-x}}$
Swish	$f(x) = x * sigmoid(x)$
Relu	$f(x) = x^+ = max(0, x)$
Elu	$f(x) = \begin{cases} x & x > 0 \\ alpha * (e^x - 1) & x < 0 \end{cases}$
Selu	$f(x) = \begin{cases} x & x > 0 \\ scale * alpha * (e^x - 1) & x < 0 \end{cases}$
Tanh	$f(x) = \frac{e^x - e^{-x}}{e^{-x} + e^x}$
Softplus	$f(x) = \ln(e^x + 1)$
Softsign	$f(x) = \frac{x}{abs(x) + 1}$

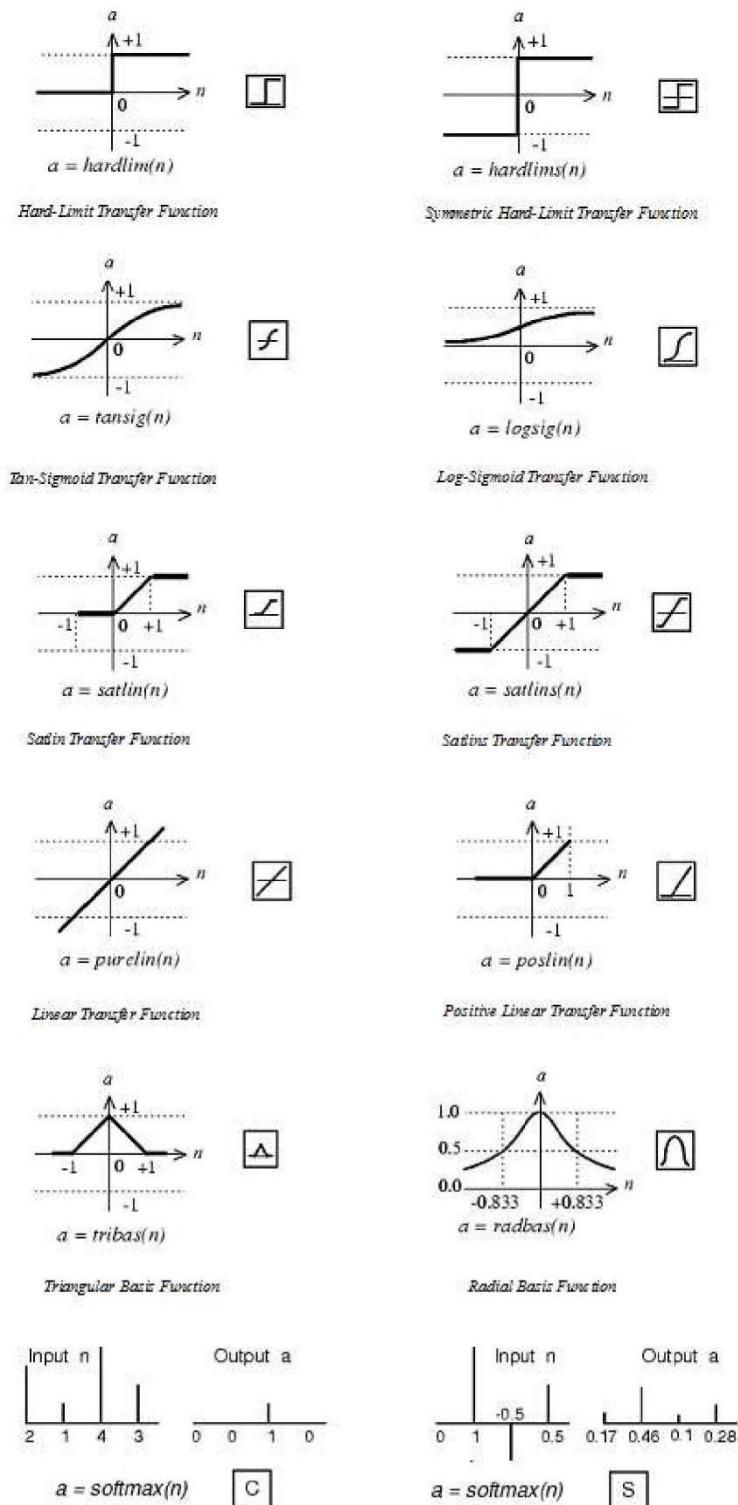


Figura 3.6- Funções transferência da rede neural.

Fonte: [6]

3.3.7 Pesos

Antes do treinar a ANN, os pesos e vieses devem ser determinados. A inicialização dos pesos e vieses são feitos automaticamente pela biblioteca do Python, Tensorflow.

3.3.8 Critério do Erro

O objetivo no treinamento da rede neural é a redução do erro tanto quanto possível. A redução do erro simplesmente significa melhorar a performance do treinamento e alcançar um modelo com mais acurácia. Diferentes definições e tipos de erros podem ser considerados durante o treinamento da rede neural. Por exemplo, o erro absoluto é definido como a diferença entre o valor medido (*output*) e o valor de referência (valor almejado). Contudo, é mais comum usam o erro médio quadrático, muito conhecido pela sigla MSE (*Mean Squared Error*) ou pela raiz do erro médio quadrático, também conhecido pela sigla RMSE (*Root Mean Squared Error*). O MSE e RMSE são definidos de acordo com a Equação 3.6 e Equação 3.7 respectivamente, no qual \hat{Y}_i é o valor previsto pelo modelo, Y_i é o valor de referência, n_d é a quantidade de dados. A redução do erro pode ser alcançada através de modificações nos pesos e/ou por algoritmos de treinamento.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (3.6)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (3.7)$$

3.3.9 Critério de parada e *Overfitting*

No treinamento das redes neurais, o critério de parada é determinado para evitar o chamado *overtraining* ou *overfitting*. ANN possui a potencial tendência de um sobreajuste durante o processo de treinamento. *Overfitting* pode ocorrer durante o processo de treinamento quando a ANN se especializa especificamente nos dados de treinamento e perde a capacidade de generalização. Esse sobreajuste é visto quando o erro de validação aumenta, enquanto o erro

de treinamento diminui. Uma performance é devido ao subajuste é um dos problemas mais comuns no treinamento de *ANNs*. Isto pode ser superado usando o método de validação-cruzada, diminuindo o número de neurônios da camada intermediária, ou adicionando obstáculos na função objetivo para arquiteturas com grande quantidade de pesos. Através do método da validação cruzada, a performance da rede é medida durante o processo de treinamento e se algum comando for dado, o treinamento é parado antes do número máximo de época ser alcançado. Época é um termo da rede neural para iteração no aprendizado. Ele representa a quantidade de vezes que todos os padrões são apresentados à rede neural. Valores maiores significa mais tempo de treinamento. Em cada época da ANN, todos os pesos dos neurônios são atualizados.

3.3.10 Método feed-forward e back-propagation

As principais características do *ANNs* são a capacidade adaptáveis de aprendizado, generalização, massivo paralelismo, robustez, associação de dados e análise espacial e temporal dos dados. O *feed-forward* com *back-propagation* é um dos métodos mais conhecidos. O modelo é formado por uma camada de entrada, seguido por uma ou mais camadas intermediárias (*hidden layers*), por último a última camada com os resultados (*output layer*). Cada camada possui neurônios artificiais (nós), associado a eles estão:

- a) Os pesos, de cada neurônio da camada anterior. Os pesos são representados pela letra w;
- b) O viés (b), que é uma constante associada exclusivamente a cada neurônio. Portanto, cada neurônio tem uma constante associada a ele chamada de viés, denominada pela letra b. Durante o treinamento da rede esse viés se modifica, assim como os pesos (w), com a finalidade de diminuir os erros.

A primeira camada de neurônios- que recebe os dados externos- é chamada de input. A última camada e que fornece o resultado da rede para o usuário é chamada output, as camadas internas são chamadas camadas intermediárias (*hidden layers*). Múltiplas camadas de neurônios lineares permitem que a rede aprenda relações lineares e não lineares entre os parâmetros de entrada (*input*) e saída (*output*). Em *feed-forward networks*, cada elemento de entrada é multiplicado pelos seus respectivos pesos (w) e somados com viés (b) do neurônio da camada seguinte, de acordo com a seguinte formula:

$$S_j = \left(\sum_{i=1}^q w_{ij}x_i \right) + b_j \quad (3.8)$$

i é um índice para se referir a um neurônio da camada anterior, j é um índice para se referir a um neurônio da camada atual, X_i é o valor neurônio da camada anterior e W_{ij} é o peso do neurônio da camada anterior no cálculo do neurônio da camada seguinte. Um neurônio afetará com pesos distintos os neurônios da camada seguinte. S é soma de todos os neurônios da camada anterior com seus respectivos pesos somado com uma constante específica denominada viés, representada por b_j . S_j será um input para uma função, conhecida como função transferência (f), essa função será responsável pela não linearidade das redes neurais. O output da função transferência será o valor do neurônio cujo índice é j . O resultado $f(x)$ é atribuído ao respectivo neurônio, e é também um elemento de entrada para os nós da camada seguinte. Todo esse processo descrito nos parágrafos anteriores é repetido de camada para camada até o resultado.

$$Y_j = f\left(\sum_{i=1}^q w_{ij}x_i + b_j\right) \quad (3.9)$$

A equação 3.9 determina o valor de cada neurônio, a Figura 3.7 ilustra essa equação. A camada final é mais simples, pois não é necessário aplicar nenhum tipo de função transferência. Após obter o resultado, é possível determinar o erro entre o modelo e os dados de referência. Existem várias formas de calcular o erro, e isto varia para cada aplicação de rede neural. O erro quadrático médio é bastante utilizado para estudo envolvendo regressão, e é dado por:

O algoritmo de *Back Propagation* (BP) atualiza os pesos (w) e viés (b) da rede neural, a fim de reduzir o MSE. O algoritmo BP atualiza a rede neural a partir da última camada, até chegar na camada de entrada da rede. A Figura 3.8 representa visualmente esse método. O método ajustando os pesos e viés de forma que o erro seja reduzido a cada iteração. Esse processo é chamado de aprendizado é iterativo, e termina quando o resultado previsto pelo

modelo esteja dentro de uma tolerância aceitável. Durante o processo de treinamento da rede, os pesos (w) e viés (b) são iterativamente ajustados de acordo com o erro entre o resultado obtido e o valor esperado.

A fórmula geral de atualização da rede neural:

$$w \leftarrow \left(w - \eta \frac{\partial E}{\partial w} \right) \quad (3.10)$$

$$b \leftarrow \left(b - \eta \frac{\partial E}{\partial b} \right) \quad (3.11)$$

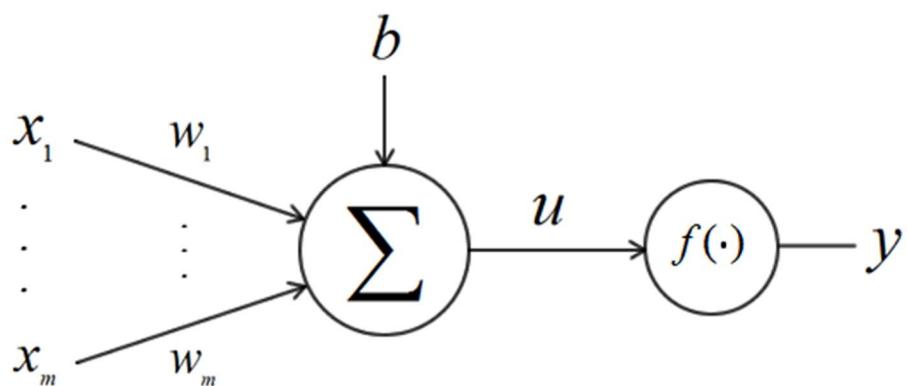


Figura 3.7- Cálculo de um neurônio.

Fonte: Em nota de rodapé:⁵. Acesso em mar. de 2021

Fica claro pelas fórmulas 3.10 e 3.11:

- Na tentativa de minimizar o valor da função de erro, calculam-se os valores dos gradientes para cada peso da rede. Do Cálculo, sabemos que o vetor gradiente fornece a direção de maior crescimento de uma função; aqui, como queremos caminhar com os pesos na direção de maior decréscimo da função de erro, basta tomarmos o sentido contrário ao do gradiente
- É importante que a função erro seja diferenciável.
- Uma vez que temos o vetor gradiente calculado, atualizamos cada peso de modo iterativo, sempre recalculando os gradientes em cada passo de iteração, até o erro

⁵ <http://computacao-inteligente.com.br/artigos/redes-neurais-artificiais/>

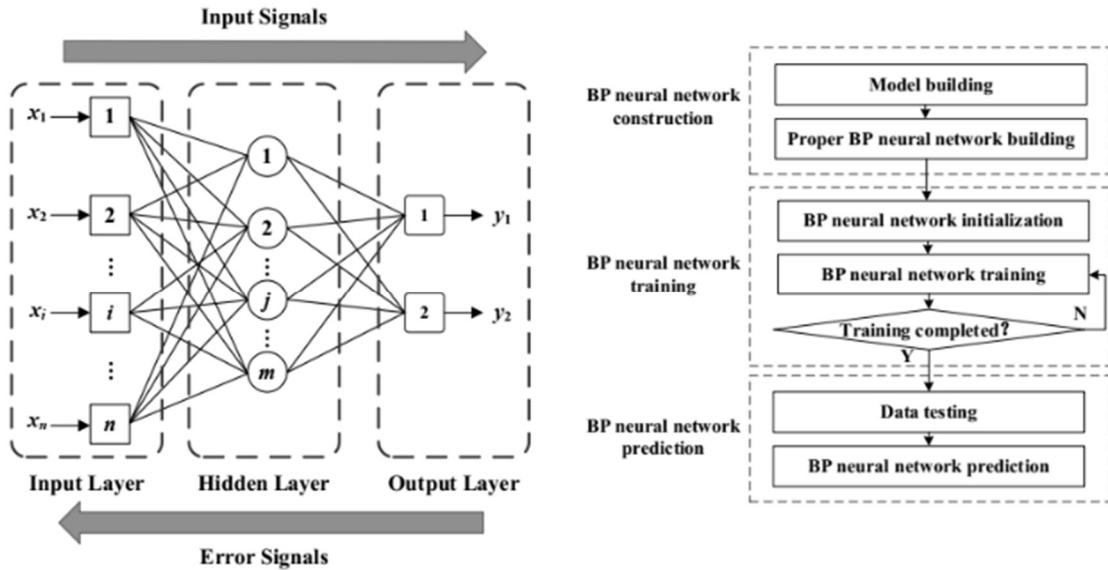


Figura 3.8- Diagrama algoritmo Neural network.

Fonte: [1]

diminuir e chegar abaixo de algum limiar preestabelecido, ou o número de iterações atingir um valor máximo, quando enfim o algoritmo termina e a rede está treinada.

- η é conhecido como a taxa de aprendizagem.

O cálculo da derivada parcial pode ser realizado através da regra da cadeia em relação ao w ou b. Segue um exemplo para o caso de w.

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial \hat{Y}} \frac{\partial \hat{Y}}{\partial w_{ij}} = -\frac{2}{n} (Y_m - \hat{Y}_m) \frac{\partial}{\partial w_{ij}} \left(f \left(\sum_{i=1}^q w_{ij} x_i + b_j \right) \right) \quad (3.12)$$

A função transferência é muito importante nessa etapa, porque dependendo da quantidade de camadas intermediária pode ocorrer o fenômeno conhecido como problema da Dissipação do Gradiente (The Vanishing Problem). A função transferência Relu possui a característica de não gerar esse tipo de problema, isso ocorre por causa da sua derivada nem sempre ser menor que 1.

Um dos principais erros associados ao treinamento da ANN é o *over-training*, nesse problema o modelo só produz um bom resultado para os dados de treinamento e perde capacidade de generalização. Uma maneira de evitar o *over-training/over-fitting* é dividir os dados em 2 sub-conjuntos: treinamento e teste. O subconjunto treinamento é utilizado para treinar o modelo (ajustando os pesos e viés) e verificando a acurácia e eficiência durante o treinamento. Uma vez que o processo de treinamento seja finalizado o subconjunto teste é usado para examinar a performance final do modelo.

A Figura 3.9 mostra um exemplo de diagrama de fluxo para *ANNs*. Observa-se que após a alimentação e normalização do conjunto de dados, os dados são divididos em 70% para treinamento e 30% para teste. Na próxima etapa, a estrutura da rede é especificada. Após a determinação do número de neurônios na camada intermediária, função de otimização, função de transferência começa o treinamento. O processo é repetido em 4 loops de diferentes números de neurônios, diferentes tipos de função de otimização, assim como diferentes combinações de função transferência para camada intermediária. O resultado de todas as performances é guardado em uma matriz de erro (MSE). No final, todos os resultados são ordenados e o modelo com melhor acurácia é escolhido.

Algoritmo 1. Estrutura geral do algoritmo de optimização

Requere: Função Objetivo f

```

 $x^{(0)} \leftarrow$  ponto aleatório no domínio de  $f$ 
For i = 1, 2, ... do
     $\Delta x \leftarrow \emptyset(f, \{x^{(0)}, \dots, x^{(i-1)}\})$ 
    if parar quando encontrar a condição then
        return  $x^{(-1)}$ 
    end if
     $x^{(i)} \leftarrow x^{(-i)} + \Delta x$ 
enf for

```

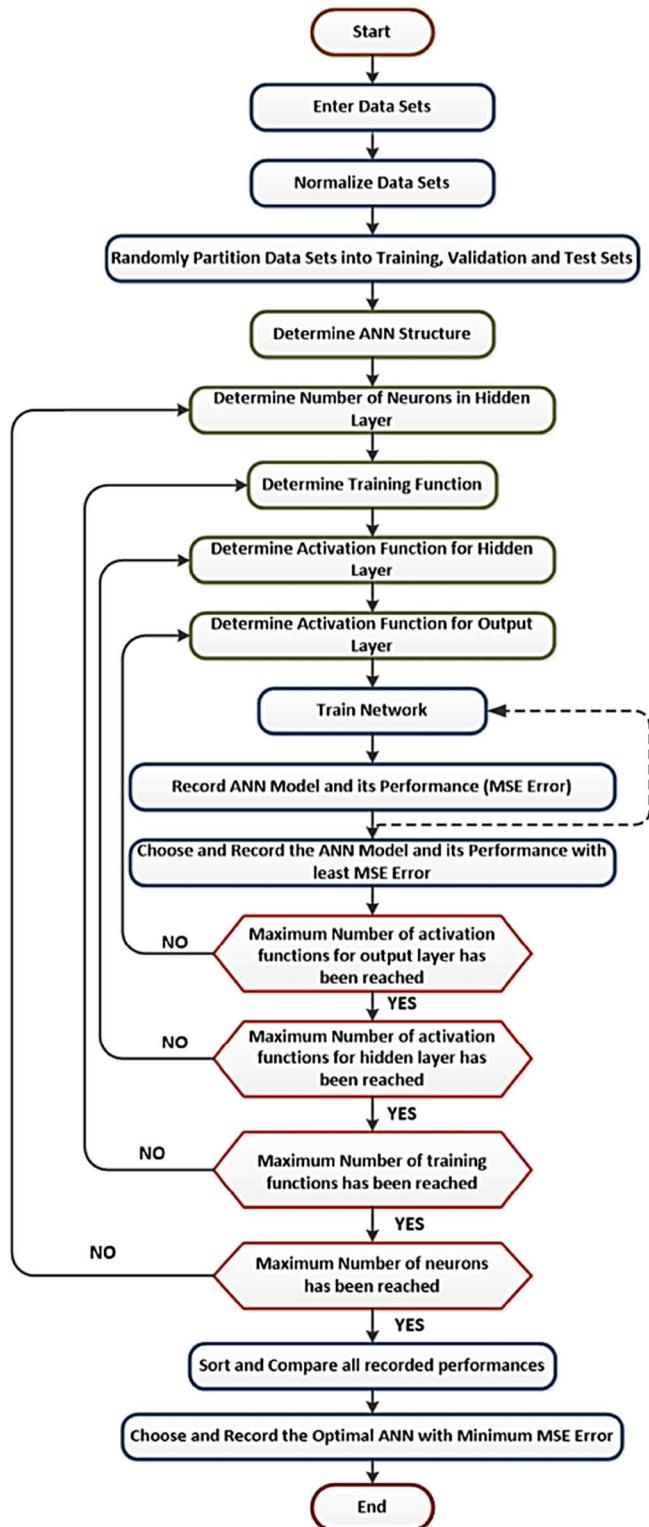


Figura 3.9- Diagrama de um código para ANN

Fonte: [6]

3.3.11 Limitações ANN

Como uma técnica de modelagem, as redes neurais artificiais possuem suas próprias limitações baseado na aplicação e metodologia particulares levados em consideração. Os desafios básicos incluem tempo de treinamento, atualização das redes neurais, seleção do conjunto de treinamento. Apesar de todas as investigações realizadas, o porquê que a ANN funciona permanece incerto sem respostas. Existe um pouco de intuição do que realmente acontece dentro da rede neural durante o processo de aprendizagem, e existem muitos problemas, em termos de metodologia, que precisam de resolvidos.

Para treinar, validar e testar uma ANN, geralmente é necessária uma grande quantidade de conjuntos de dados. Não há número fixo de conjuntos de dados para um processo de treinamento ideal. No entanto, a quantidade de dados obtidos deve ser grande o suficiente para treinar a rede com a maior precisão possível e para fornecer compreensão suficiente da dinâmica do sistema. Os conjuntos de dados podem ser dados operacionais ou dados simulados por um modelo previamente confirmado. A aquisição de dados, especialmente em regiões operacionais, pode ser difícil e o processo demorado. Novos conjuntos de dados não podem alimentar uma ANN já treinada; para melhorar seu desempenho é necessário treinar a rede novamente com os novos conjuntos de dados disponíveis. Um modelo bem treinado para um problema específico, não significado que servirá para outro problema. A ANN depende do desenvolvimento empírico. É quase nova técnica e ainda precisa ser desenvolvida com base nas implementações práticas e experimentos adquirido por pesquisadores H. Asgari [6]. Outrossim, a extrapolação do domínio de entrada da rede neural não é indicada, dessa maneira, a predição dos valores devem respeitar os limites dos domínios usados para treinar a rede.

4. Resultados

Este capítulo se destina a explicar as bases dos métodos utilizados no desenvolvimento do projeto. Serão abordados os conceitos por trás da inteligência artificial e principalmente das redes neurais. Os erros utilizados para análises nesta seção se referem ao conjunto de teste, não o de validação. Além disso, o tempo de simulação no treinamento da rede neural é dependente do processador e placa gráfica.

4.1 Análise Correlação

4.1.1 Correlação matemática

Os dados para Nusselt e fator de atrito foram gerados a partir das correlações do estudo de caso feito no LabMEMS C. P. N. Cotta e colaboradores [10]. As correlações podem ser vistas nas equações 4.1 e 4.2.

$$\frac{\overline{Nu}}{Nu_{\alpha=0}} = 1 + 0,099 \left(\frac{L}{D_h} \right)^{-0,852} \alpha^{1,055} \quad (4.1)$$

$$\frac{\bar{f}}{f_{\alpha=0}} = 1 + 0,16 \left(\frac{L}{D_h} \right)^{-1,318} \alpha^{2,526} \quad (4.2)$$

As correlações acimas se aplicam a:

$$1mm \leq D \leq 5mm; 40.00 \leq Re \leq 100.000; 2,8 \leq \frac{L}{D_h} \leq 19,3; 15^\circ \leq \alpha \leq 45^\circ$$

Onde $Nu_{\alpha=0}$ (Dittus-Boelter) válido para $0,6 \leq Pr \leq 160; Re \geq 10^5; \frac{L}{D_h} > 10$ é:

$$Nu_{\alpha=0} = 0,023 Re^{0,8} Pr^{0,3} \quad (4.3)$$

e $f_{\alpha=0}$ (Blasius) válido para $4000 < Re < 10^5$ é:

$$f = \frac{0,316}{Re^{0,25}} \quad (4.4)$$

Como temos uma equação para gerar dados, a quantidade de dados possíveis de serem criadas é ilimitada. Contudo para se aproximar de casos em que não há correlações disponíveis, foram feitos testes com a finalidade de reduzir o máximo possível a essa quantidade de dados, visto que gerar milhares de dados a partir de simulações e experimentos é inviável.

4.1.2 Conjunto de dados

Foram criado 6 conjuntos de dados, distintos entre si, com o intuito de analisar a influência dos dados no treinamento da rede neural. A tabela 4.1, como exemplo, apresente o primeiro deles, o conjunto 1. O Conjunto 1 possui: 7 valores de Reynolds, 6 valores de α e 6 valores de $\frac{L}{D_h}$. Logo, ao se multiplicar $7 \times 6 \times 6$, é possível formar 252 dados. Vale lembrar que para cada valor de input, existe o valor correspondente de Nusselt e Fator de Atrito. Estes são reservados para o *output*, enquanto aqueles são reservados para o “input”. Os outros conjuntos podem ser conferidos no Anexo A.

Tabela 4.1- Conjunto 1 (252 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000; 50.000; 60.000; 70.000; 80.000; 90.000; 100.000
α	15° 21° 27° 33° 39° 45°
$\frac{L}{D_h}$	2,8 6,1 9,4 12,7 16 19,3

Fonte: Própria

A Figura 4.1 mostra como é a geração de 1 dos 252 dados presentes no Conjunto 1. Os outros conjuntos possuem o mesmo procedimento de geração de dados, contudo eles apresentam mais dados e dados diferentes. A Tabela 4.2 apresenta a quantidade de dados em cada conjunto.

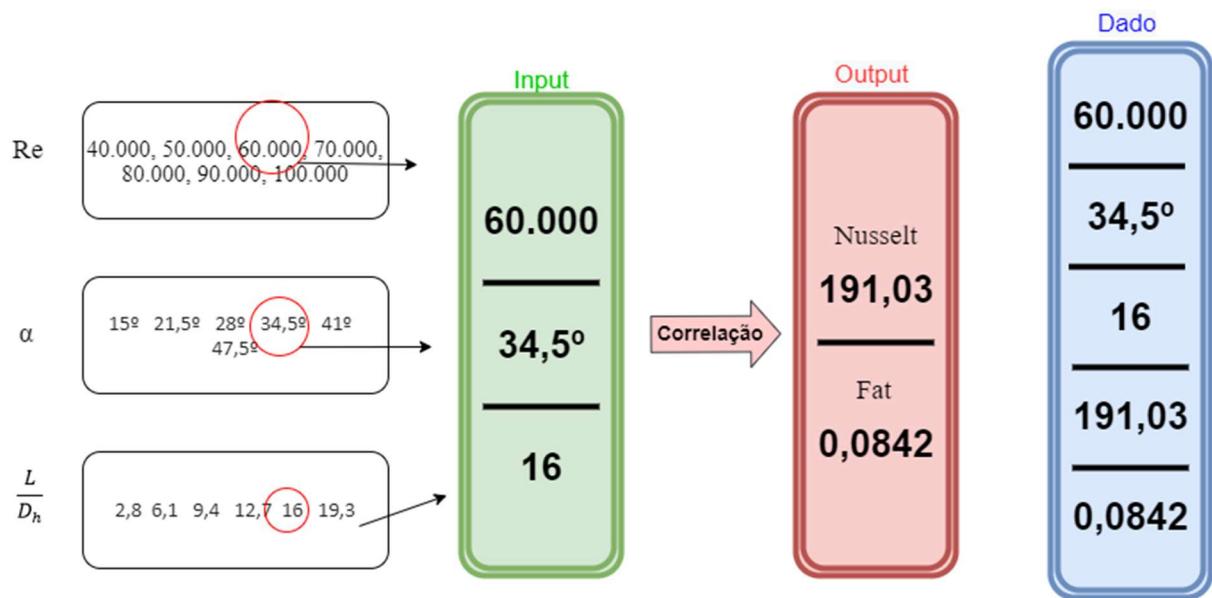


Figura 4.1- Geração de 1 unidade do conjunto dados

Fonte: Própria

Tabela 4.2- Número de dados por conjunto

CONJUNTO	QUANTIDADE DE DADOS
1	252
2	343
3	847
4	2197
5	5733
6	16337

Fonte: Própria

Vale relembrar que da quantidade de dados descrito para cada conjunto 70% é destinado para treinar a rede e 30% para o teste final. Por exemplo, no conjunto 1, 70% dos 252 dados, isto é, 176 são destinados para treinamento e 76 para o teste.

4.1.3 Performance da ANN

A) Estrutura Rede Neural

Os grandes valores para o número de neurônios em alguns casos nessa seção foram utilizados com intuito de estudo, pois entende-se que esses grandes valores podem acarretar a especialização ou *over-fitting da rede neural*.

i) Influência da Camada intermediária

Nesta seção, apesar de existirem vários trabalhos recomendando o uso de 2 camadas intermediárias, mostra o comportamento do erro em função do número de camadas intermediárias.

Foram realizadas 160 simulações, as configurações dos parâmetros das simulações seguem a tabela 4.3.

Tabela 4.3- Parâmetros nas simulações para camada intermediária.

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 3
Camadas intermediárias	Variável	1 ou 2 ou 4 ou 6 ou 8 ou 10 ou 12 ou 16
Neurônios	Variável	4 ou 64 ou 128 ou 512 ou 1024
Batch_size	Variável	Off (= 0) ou 32
Nº épocas	Fixo	50
Função transferência	Fixo	ReLU
Função otimização	Variável	Adam ou Adamax
Função erro	Fixo	MSE

Fonte: Própria

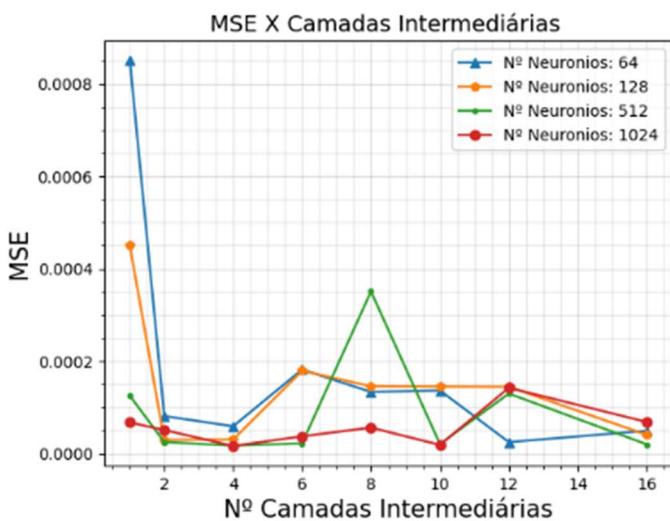
Conforme o esperado, a tabela o confirma que o menor erro MSE está associado a rede neural de 2 camadas intermediárias. A figura 26 mostra como se comporta o erro diante dessas múltiplas camadas. As duas figuras (a) e (b) são duas formas diferentes de observar o mesmo dado. É possível visualizar a partir da Figura 4.2 (a) a instabilidade das redes neurais, quando possui mais que 2 camadas intermediárias, não existindo nenhuma previsibilidade e garantia de

redução de erro nessas estruturas, ao contrário da rede neural de 2 camadas que é baixa para todos os dados. Pode-se notar uma convergência de queda em torno de 2 camada intermediárias, independentemente do número de neurônios. Pela Figura 4.2 (b) é possível observar a redução do erro à medida que o número de neurônios cresce

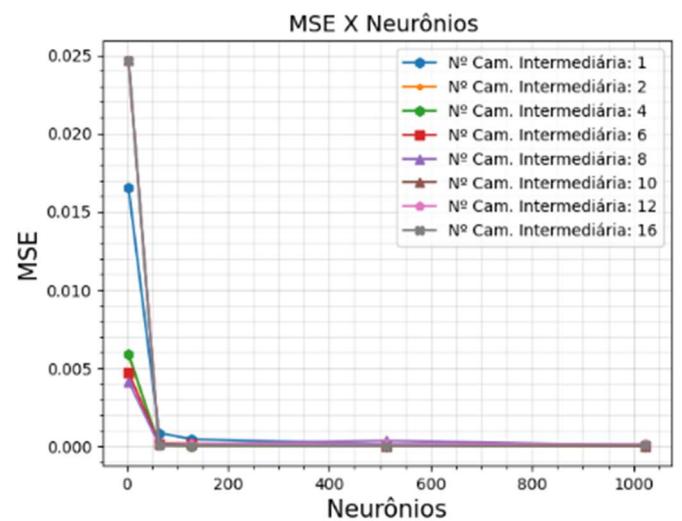
Tabela 4.4- Valor mínimo de MSE para cada Nº de camada intermediária

Dados	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
847	1	1024	Adam	0	50	0min4s	0.997823	0.000053
847	2	1024	Adam	0	50	0min4s	0.999512	0.000012
847	4	1024	Adam	32	50	0min4s	0.999331	0.000016
847	6	512	Adam	0	50	0min4s	0.999418	0.000014
847	8	1024	Adamax	32	50	0min5s	0.999103	0.000022
847	10	1024	Adam	32	50	0min5s	0.999240	0.000019
847	12	512	Adam	0	50	0min5s	0.999205	0.000020
847	16	512	Adam	32	50	0min5s	0.999187	0.000020

Fonte: Própria



(a)



(b)

Figura 4.2- Erro das redes neurais baseado na quantidade de camadas intermediárias. (a) Camada intermediária está no eixo da abscissa. (b) Neurônios está na abscissa.

Fonte: Própria

ii) Análise da influência da quantidade de dados na rede neural

A Tabela 4.5 representa as circunstâncias nos quais foram realizadas as simulações. A quantidade de dados é sem dúvida um dos elementos mais importantes no treinamento de redes neurais. As redes neurais treinadas com 252 dados, melhor que fosse a configuração da rede, alcançaram no geral o desvio padrão na ordem de 10^{-5} , enquanto as redes neurais treinadas com 16337 dados, alcançaram facilmente o MSE na ordem de 10^{-7} , duas ordens de grandeza a menos, isto pode ser conferido na Tabela 4.6.

Tabela 4.5- Parâmetros de simulações do estudo da influência da quantidade de dados de treinamento da ANN.

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Variável	Conjunto 1 ou 2 ou 3 ou 4 ou 5 ou 6
Camadas intermediárias	Fixo	2
Neurônios	Variável	4, 32 64, 128, 256 e 512
Batch_size	Variável	32
Nº épocas	Fixo	50
Função transferência	Fixo	ReLU
Função otimização	Variável	Adam ou Adamax
Função erro	Fixo	MSE

Tabela 4.6- Comparações do MSE em relação a quantidade de dados usados no treinamento.

Dados	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
252	512	Adam	32	50	0min2s	0.996624	4.813407e-05
343	512	Adam	32	50	0min2s	0.997601	4.824005e-05
847	254	Adam	32	50	0min3s	0.999358	1.690280e-05
2197	128	Adam	32	50	0min6s	0.999771	4.699408e-06
5733	254	Adam	32	50	0min13s	0.999801	3.703753e-06
16337	254	Adam	32	50	0min34s	0.999943	8.710346e-07

Fonte: Própria

iii) Análise da influência do tipo de função otimizadora na rede neural

Tabela 4.7- Parâmetros de simulações de estudo da função de otimização.

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Variável	Conjunto 1 ou 2 ou 3 ou 4 ou 5 ou 6
Camadas intermediárias	Fixo	2
Neurônios	Variável	1000 ou 2000 ou 4000 ou 6000 ou 8000 ou 10000 ou 12000 ou 16000
Batch_size	Variável	32
Nº épocas	Fixo	50
Função transferência	Fixo	ReLU
Função otimização	Variável	Adam ou Adamax
Função erro	Fixo	MSE

Fonte: Própria

Tempo de simulação de todas as hipóteses foi de 144 minutos. A função Adam se saiu melhor para os conjuntos com menor quantidade de dados (conjunto 1, 2), na Figura 4.3 pode visualizar essa diferença. Para os conjuntos maiores a função Adamax apresentou melhores resultados, mas ainda sim a função Adam obteve bons resultados e em alguns momentos melhores que Adamax.

iv) Análise da influência do *batch* no treinamento de redes com diferentes valores de neurônios nas camadas intermediárias.

O lote de treinamento ou *batch* se refere a quantidade de dados que irão passar pela rede neural até ela atualizar os pesos. O lote ou *batch* de tamanho 0 significa que esse tipo de recurso não está sendo utilizado e que os pesos e vieses estão se atualizando a cada vez que um dado é utilizado para treinar a rede. A Tabela 4.8 mostra os parâmetros utilizados para as simulações, somente o número de neurônios e o *batch* estão variando, este com 5 opções e aquele com 8. Ao se multiplicar 5×8 temos 40 diferentes redes neurais

Tabela 4.8- Parâmetros de simulações de estudo do *Batch_size*

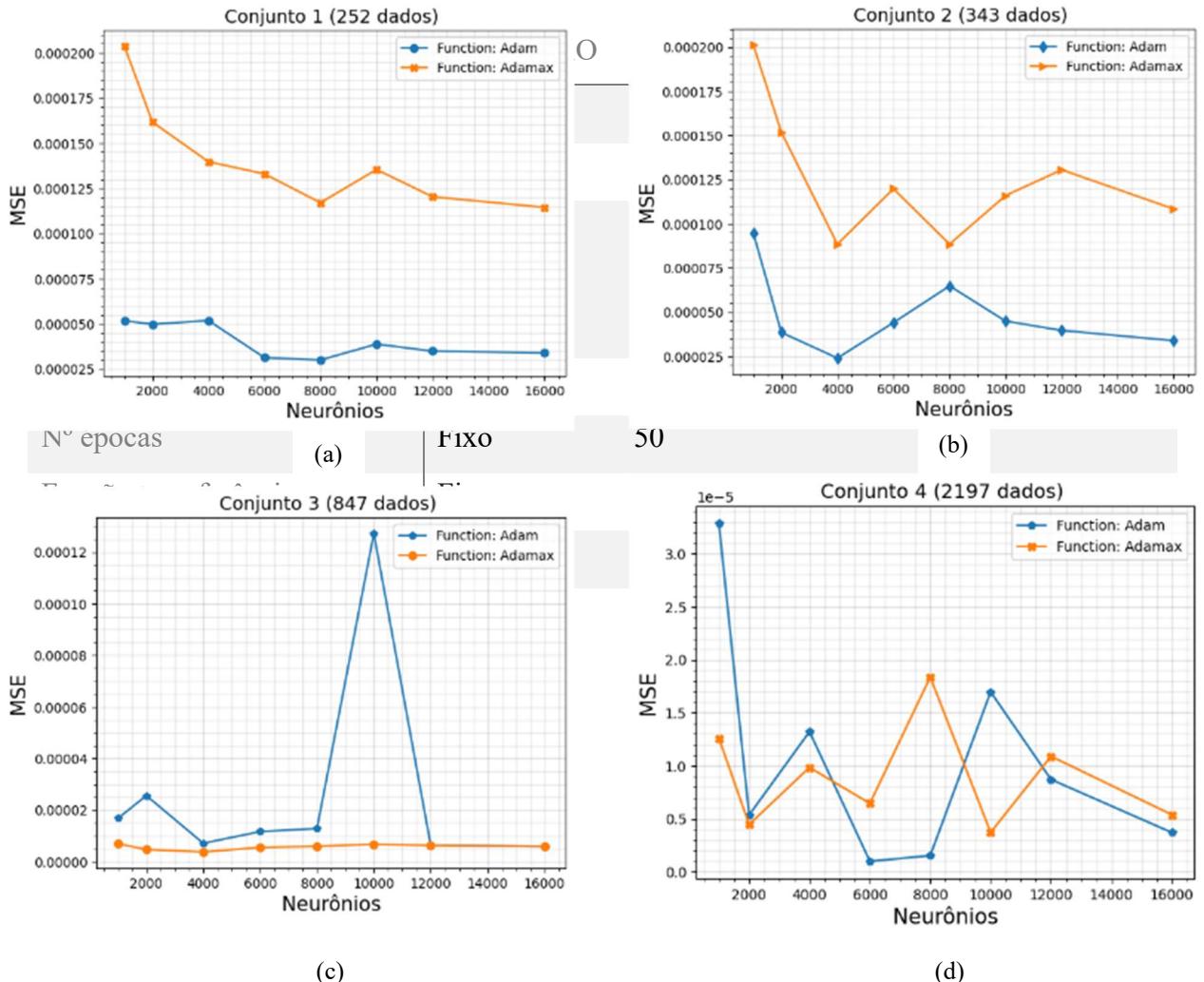


Figura 4.3-Performance Função de otimização para cada conjunto.

Fonte: Própria

Os resultados das simulações podem ser vistos na Figura 4.4 e Tabela 4.9. A Figura 4.4 (a) mostra o erro associado a cada tipo de *batch* à medida que o número de neurônios na camada intermediária aumenta. É possível notar que para grandes *batchs* aumenta bastante, não sendo de muito útil aplicados deles. A Figura 4.4 (b) mostra somente as *ANNs* com valores “batchs” menores. Embora não seja possível ter uma conclusão sólida, o valor *batch* de 32 apresentou bons resultados para quantidade de neurônios abaixo de 8000. O pico representado na figura

não deve ser considerado, pois tem origem de algum erro durante o treinamento e pode ou não aparecer nas simulações. Por isso é importante a análise dos dados, para selecionar os resultados válidos e descartar os resultados contaminados.

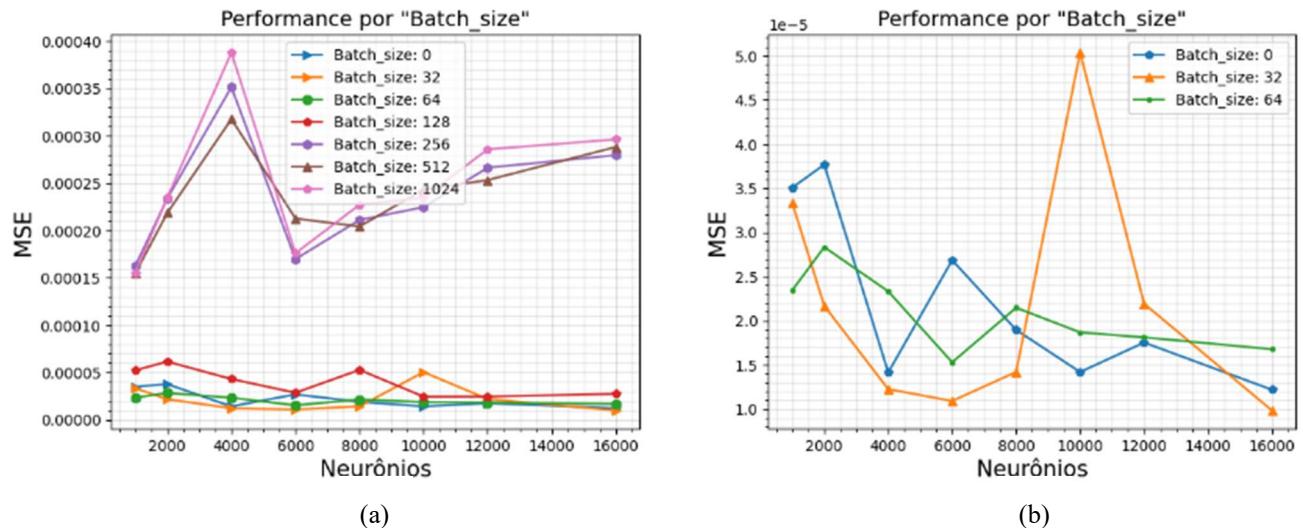


Figura 4.4- Influência do *batch* no erro. (a) Figura com todos os *batches*. (b) Figura com 3 tipos de *batch*.

Fonte: Própria

Tabela 4.9- Dez *batches* com menores valores do MSE

Dados	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE	
0	252	2	16000	Adam	32	50	0min18s	0.999594	0.000010
1	252	2	6000	Adam	32	50	0min4s	0.999550	0.000011
2	252	2	16000	Adam	0	50	0min18s	0.999497	0.000012
3	252	2	4000	Adam	32	50	0min3s	0.999495	0.000012
4	252	2	4000	Adam	0	50	0min3s	0.999417	0.000014
5	252	2	10000	Adam	0	50	0min8s	0.999415	0.000014
6	252	2	8000	Adam	32	50	0min6s	0.999414	0.000014
7	252	2	6000	Adam	64	50	0min3s	0.999369	0.000015
8	252	2	16000	Adam	64	50	0min11s	0.999308	0.000017
9	252	2	12000	Adam	0	50	0min11s	0.999277	0.000018

Fonte: Própria

v) Análise da influência da época na rede neural

Tabela 4.10- Parâmetros de simulações de estudo da época

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1
Camadas intermediárias	Fixo	2
Neurônios	Variável	1000 ou 2000 ou 4000 ou 6000 ou 8000 ou 10000 ou 12000 ou 16000
Batch_size	Fixo	32
Nº épocas	Variável	50, 500, 1000, 2000, 4000, 6000, 8000
Função transferência	Fixo	ReLU
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

O valor de época, como já descrito antes, se remete a quantidade de vezes que o conjunto de dados treina a rede neural. Ele está intimamente ligado ao custo computacional da rede neural, podendo elevar o tempo de aprendizagem, de uma única rede neural, de segundos para mais de 30 minutos. Já é de se esperar que o aumento do número de iterações – época- cause uma diminuição do MSE, todavia é necessário ter cautela, porque associado a essa diminuição do erro pode estar o *overfitting*.

O tempo total de simulações dos 56 modelos de *ANNs* durou 3 horas e 45 minutos- no GPU do Google. A Figura 4.5 mostra o comportamento do erro de 8 curvas à medida que o número de época cresce. A tendência geral é sempre de diminuição do erro. A Tabela 4.11 mostra as dez configurações de rede neural com o menor erro, pode-se observar que todas possuem o erro com a mesma ordem de grandeza. A critério para escolha da melhor configuração foi o menor número de Épocas e consequentemente menor tempo de treinamento da rede neural. Dessa forma, um baixo número de época é desejável, pois, reduzem a chance de

ocorrer o *overfitting*. A configuração de índice 4, com a combinação de 4000 neurônios e 500 Épocas se sagrou a melhor opção.

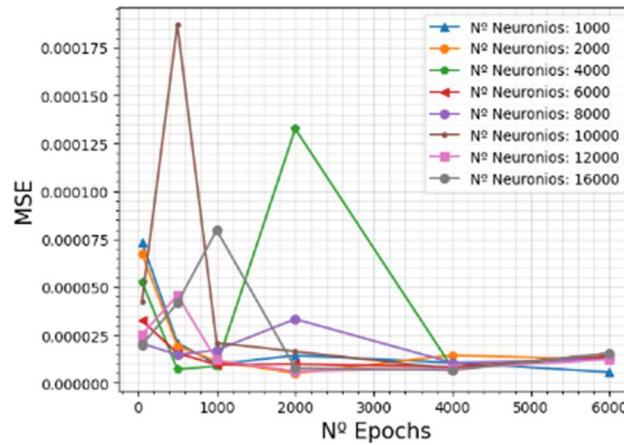


Figura 4.5- Efeito do Número de Épocas no erro

Fonte: Própria

Tabela 4.11- Tabela em ordem decrescente de erro, 10 primeiras configurações.

Dados	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE	
0	252	2	2000	Adam	32	2000	1min16s	0.999781	0.000005
1	252	2	1000	Adam	32	6000	3min35s	0.999759	0.000006
2	252	2	12000	Adam	32	2000	6min54s	0.999735	0.000006
3	252	2	16000	Adam	32	4000	23min6s	0.999718	0.000007
4	252	2	4000	Adam	32	500	0min25s	0.999694	0.000007
5	252	2	10000	Adam	32	4000	9min46s	0.999688	0.000007
6	252	2	12000	Adam	32	4000	13min48s	0.999685	0.000007
7	252	2	16000	Adam	32	2000	11min35s	0.999668	0.000008
8	252	2	4000	Adam	32	4000	3min20s	0.999665	0.000008
9	252	2	6000	Adam	32	4000	4min48s	0.999642	0.000008

Fonte: Própria

vi) Análise da influência do tipo de função transferência

Tabela 4.12- Parâmetros para testes de diferentes funções de transferência.

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1
Camadas intermediárias	Fixo	2
Neurônios	Variável	1000 ou 2000 ou 4000 ou 6000
Batch_size	Fixo	32
Nº épocas	Fixo	50
Função transferência	Variável	ReLU e/ou Sigmoid e/ou Tanh
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

A Tabela 4.12 mostra as configurações das redes neurais treinadas para estimar qual função transferência se adequa melhor aos dados. Existe 9 possibilidades de configuração de rede neural baseado em composições diferentes de funções transferência.

Como a rede possui 2 camadas intermediárias as opções são:

Tabela 4.13- Diferentes configurações de funções transferência por camada.

PRIMEIRA CAMADA	SEGUNDA CAMADA
ReLU	Relu
ReLU	Tanh
ReLU	Sigmoid
Tanh	Relu
Tanh	Relu
Tanh	Sigmoid
Sigmoid	Relu
Sigmoid	Tanh
Sigmoid	Sigmoid

Fonte: Própria

A Tabela 4.13 mostra as 10 simulações de menor MSE, em ordem crescente. A partir desta tabela é possível verificar que os menores erros estão associados as redes neurais com pelo menos uma das funções sendo ReLu. Pode-se concluir que a rede neural com duas funções ReLu é uma boa escolha.

B) Modelo Otimizado

De acordo com M. Venturini [5] modelo de rede neural com múltiplos *outputs* geram resultados melhores do que modelos com um único output, logo, o modelo foi construído com os ambos os *outputs* Nusselt e Fator de atrito, de acordo com a Figura 4.6.

O capítulo 4.1 foi cumulativamente orientando a forma de construir uma rede neural para uma PCHE, sendo os dados oriundos de uma correlação. A Tabela 4.15 mostra as configurações da rede neural a ser treinada e será responsável de prever os resultados do Nusselt e Fator de atrito do trocador de calor.

Tabela 4.14- MSE para diferentes composições de função transferência e com diferentes números de neurônios.

Dados	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE	
0	252	relu/relu	2	1000	Adam	32	500	0min20s	0.999837	0.000004
1	252	relu/tanh	2	1000	Adam	32	500	0min19s	0.999800	0.000004
2	252	relu/tanh	2	2000	Adam	32	500	0min20s	0.999770	0.000005
3	252	relu/relu	2	4000	Adam	32	500	0min26s	0.999709	0.000006
4	252	relu/relu	2	2000	Adam	32	500	0min20s	0.999705	0.000007
5	252	relu/sigmoid	2	1000	Adam	32	500	0min19s	0.999695	0.000007
6	252	relu/tanh	2	4000	Adam	32	500	0min26s	0.999325	0.000015
7	252	relu/relu	2	6000	Adam	32	500	0min37s	0.999002	0.000022
8	252	tanh/relu	2	2000	Adam	32	500	0min20s	0.998669	0.000029
9	252	tanh/relu	2	6000	Adam	32	500	0min37s	0.998423	0.000035

Fonte: Própria

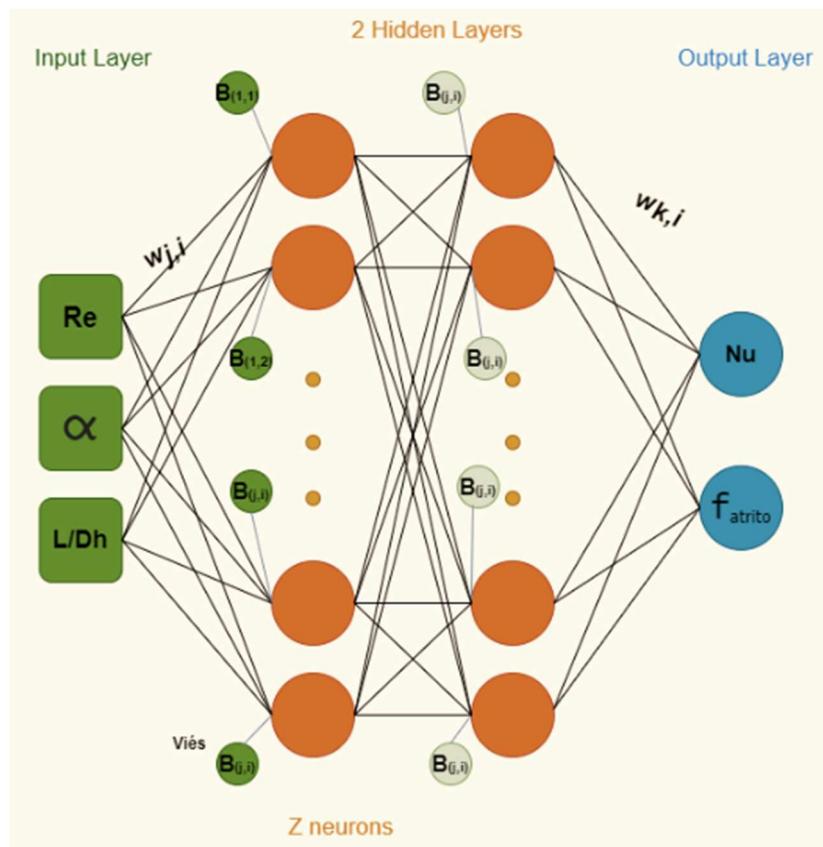


Figura 4.6- Ilustração da rede neural, destaque para valores de input e output.

Fonte: Própria

Tabela 4.15- Propriedades da rede neural

Input	$Re, \alpha, \frac{L}{D_h}$
Output	Nu, Fat
Dados	Conjunto 1
Nº Camadas intermediárias	2
Nº de Neurônios	4000
Nº Épocas	500
Função transferência	[Relu,Relu]
Função otimização	Adam
Função erro	MSE
Nº dados para treinamento	176
Nº dados para teste	76
Quantidade de coeficientes	16.028.002

Fonte: Própria

i) Imprevisibilidade devido a divisão dos dados

Esse tópico visa ao estudo do impacto na rede neural da divisão dos dados para o conjunto treinamento (70%) e para o conjunto teste (30%). Essa divisão não é aleatória, mas imprevisível, pois tenta distribuir os dados para 2 grupos de forma homogênea, permitindo que ambos possuem informações representativas de todas as variáveis, no entanto essa divisão gera redes neurais distintas entre si, porque elas serão treinadas com dados diferentes. Resumindo, o conjunto de dados de treinamento de uma rede neural será diferente da outra, isso ocorre no momento da separação dos dados de teste e dados de treinamento.

Para estudar esse fenômeno foram realizadas 100 simulações, as configurações da ANN estão descritas na Tabela 4.16. Como a configuração da rede neural possui uma característica de um tempo de treino pequeno, essas 100 simulações levaram cerca de 40 minutos.

A Figura 4.7 possui um gráfico de Boxplot, no qual podemos ver a distribuição dos dados, através da mediana, dos quartis e da dispersão dos “outliers”. A mancha azul é uma espécie de histograma no qual é possível visualizar a região de concentração de dados. É

possível concluir que a seleção dos dados de treino e teste influenciará minimamente na performance na rede, contudo é importante saber se a rede que foi treinada está numa região fora da curva ou se ela de fato na região de performance que era esperada. Para resolver isso foi calculado o intervalo de confiança de 95%, baseado nessa amostra de 100 simulações. Esse intervalo de confiança serve tanto para medir a performance da ANN como serve de referência de performance desejável.

O boxplot é uma análise, o qual é possível uma visualização da posição, dispersão, simetria, caudas e valores discrepantes (*outliers*) do conjunto de dados.

Posição – Em relação à posição dos dados, observa-se a linha central do retângulo (a mediana ou segundo quartil).

Dispersão – A dispersão dos dados pode ser representada pelo intervalo interquartílico que é a diferença entre o terceiro quartil e o primeiro quartil (tamanho da caixa), ou ainda pela amplitude que é calculada da seguinte maneira: valor máximo – valor mínimo. Embora a amplitude seja de fácil entendimento, o intervalo interquartílico é uma estatística mais robusta para medir variabilidade uma vez que não sofre influência de outliers.

Simetria – Um conjunto de dados que tem uma distribuição simétrica, terá a linha da mediana no centro do retângulo. Quando a linha da mediana está próxima ao primeiro quartil, os dados são assimétricos positivos e quando a posição da linha da mediana é próxima ao terceiro quartil, os dados são assimétricos negativos. Vale ressaltar que a mediana é a medida de tendência central mais indicada quando os dados possuem distribuição assimétrica, uma vez que a média aritmética é influenciada pelos valores extremos.

Caudas – As linhas que vão do retângulo até aos outliers podem fornecer o comprimento das caudas da distribuição.

Outliers – Já os outliers indicam possíveis valores discrepantes. No boxplot, as observações são consideradas outliers quando estão abaixo ou acima do limite de detecção de outliers.

O limite de detecção de outliers é construído utilizando o intervalo interquartílico, dado pela distância entre o primeiro e o terceiro quartil. Sendo assim, os limites inferior e superior de detecção de outlier são dados por:

Limite Inferior = Primeiro Quartil – 1,5 × (Terceiro Quartil – Primeiro Quartil)

Limite Superior = Terceiro Quartil + 1,5 × (Terceiro Quartil – Primeiro Quartil)

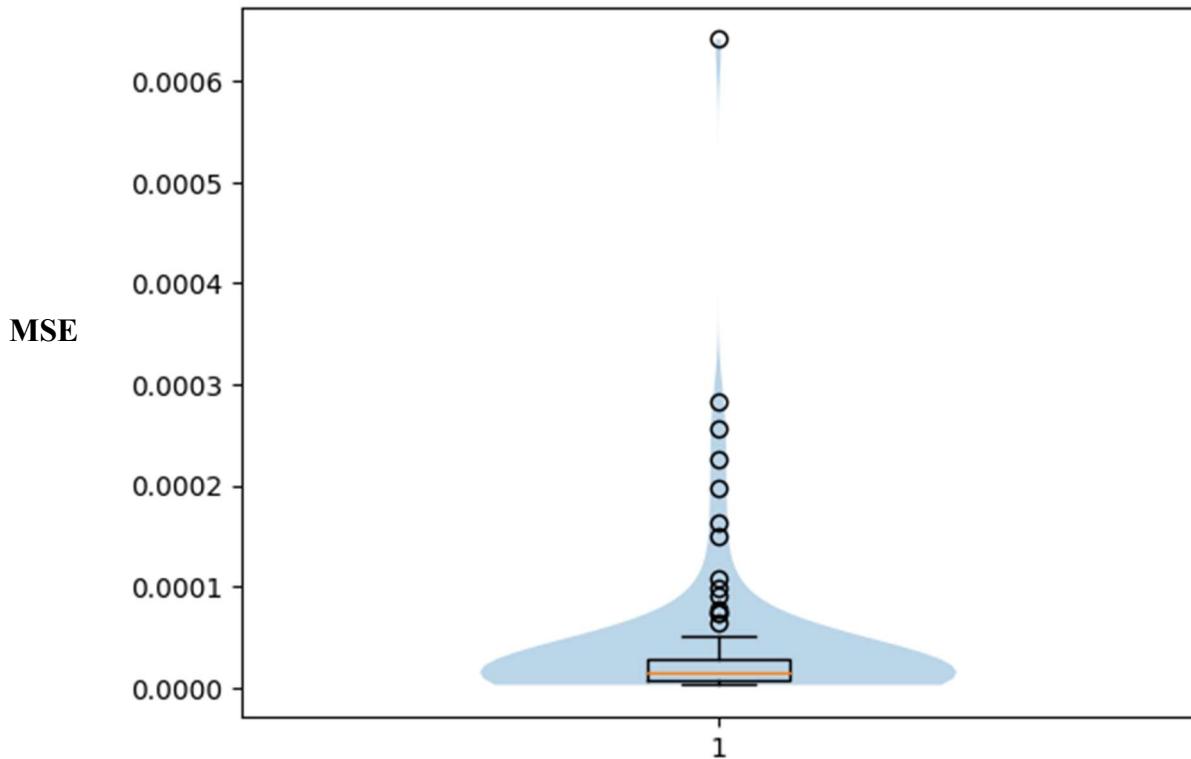


Figura 4.7- BoxPlot associado a um histograma.

Fonte: Própria

As Equação 4.5 e 5.6 servem para o cálculo da margem de erro e intervalo de confiança baseado nas informações da amostra. A Tabela 4.16 agrupa todas as informações.

$$\text{Margem de erro} = z_{95\%} \cdot \frac{s}{\sqrt{n}} \quad (4.5)$$

$$IC_{95\%} = \bar{x} \pm \text{Margem de erro} \quad (4.6)$$

Tabela 4.16- Parâmetros para cálculo do Intervalo de confiança de 95%.

PARÂMETRO	VALOR
n	100
\bar{x}	$3,79 \times 10^{-5}$
s	7.99×10^{-5}
$Z_{95\%}$	1,96
<i>Margem de erro</i>	$1,56 \times 10^{-5}$
$IC_{95\%}$	$2,2 \times 10^{-5} \leq x_i - \bar{x} \leq 5,36 \times 10^{-5}$

Fonte: Própria

Portanto, $IC_{95\%}$ permitiu definir quando uma ANN, baseada no MSE, está fora da região de performance que era esperada, por conseguinte, é esperado que em 95% das vezes que dividamos os dados o MSE respectivo esteja dentro deste intervalo e caso não esteja devemos considerar como um “outlier” e descartar a rede neural. Por outro lado, o $IC_{95\%}$ revela também que a ordem de grandeza do MSE daquele tipo de configuração de rede neural está de fato na ordem de 10^{-5} .

ii) Teste massivo da rede neural

Uma característica deste trabalho é que os dados podem ser gerados por uma correlação, logo é possível criar uma quantidade enorme de dados para avaliar o quanto bem a rede neural interpola-os. Como teste, foram gerados 29.791 dados, lembrando que a rede neural foi treinada com apenas 176 dados (70% de 252). É possível verificar o resultado na Figura 4.8.

As informações extraídas da figura são: Os valores de Nusselt foram bem ajustados, tendo pouca dispersão de dados; o Fator e atrito apresentou uma concentração de dispersão no intervalo de 0,6 e 1,2 superior ao valor de referência, no entanto não ultrapassou em regra mais que 15% do valor de referência. O fato das previsões do Fator e atrito serem superiores ao valor almejado talvez indiquem que o modelo necessite de uma quantidade maior de dados para treinar a rede.

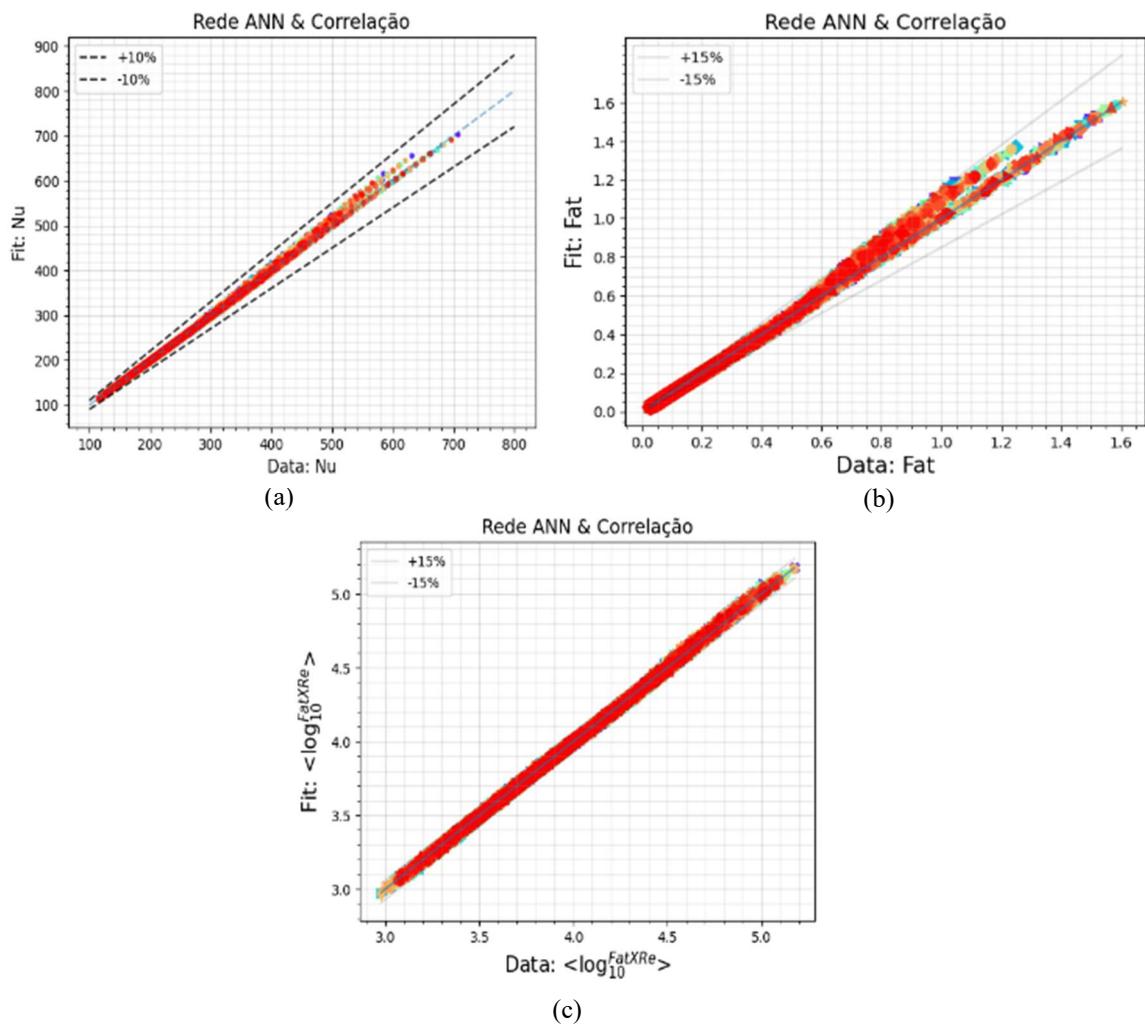


Figura 4.8- Teste massivo da rede neural.

Fonte: Própria

iii) Análise do Nusselt e Fator de atrito

A Figura 4.9 representa a redução do erro durante o treinamento da rede neural, o MSE final alcançado foi de $1,3234 \times 10^{-5}$. A rede neural possui um total de 16.028.002 de pesos(w) e vieses(b), todos foram atualizados durante o processo de aprendizagem.

A Figura 4.10 mostra a performance da modelo para um trocado com características geométricas definidas. Os parâmetros geométricos de $\alpha = 17^\circ$ e $\frac{L}{D_h} = 12$ foram escolhidos por se tratarem parâmetros geométricos normalmente empregado em PCHEs. O erro global relativo máximo obtido no momento da validação da rede neural com o conjunto teste foi

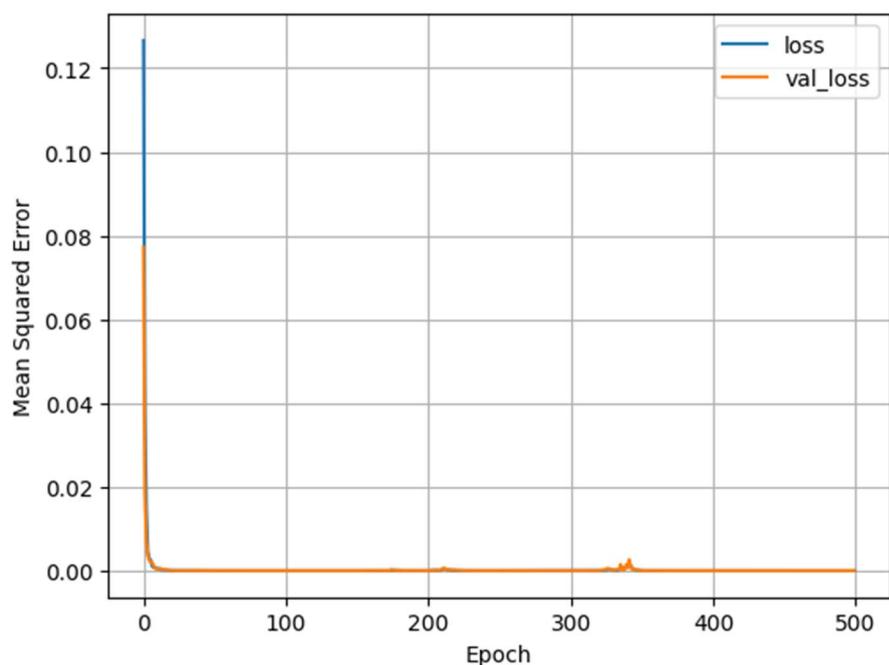


Figura 4.9- Redução do MSE por época. Curva azul treinamento, curva laranja teste.

Fonte: Própria

atribuído para os *outputs*, de forma que a previsão de Nusselt sempre terá um erro associado de 0,73% e o Fat um erro 2,05%.

A Figura 4.11 mostra a performance do ajusto quando o $\frac{L}{D_h} = 12$ permanece constante e o α varia. A Figura 4.12 mostra justamente o inverso, o α permanece constante, enquanto o $\frac{L}{D_h}$ varia. Os resultados se mostraram satisfatórios a medida do que se esperava.

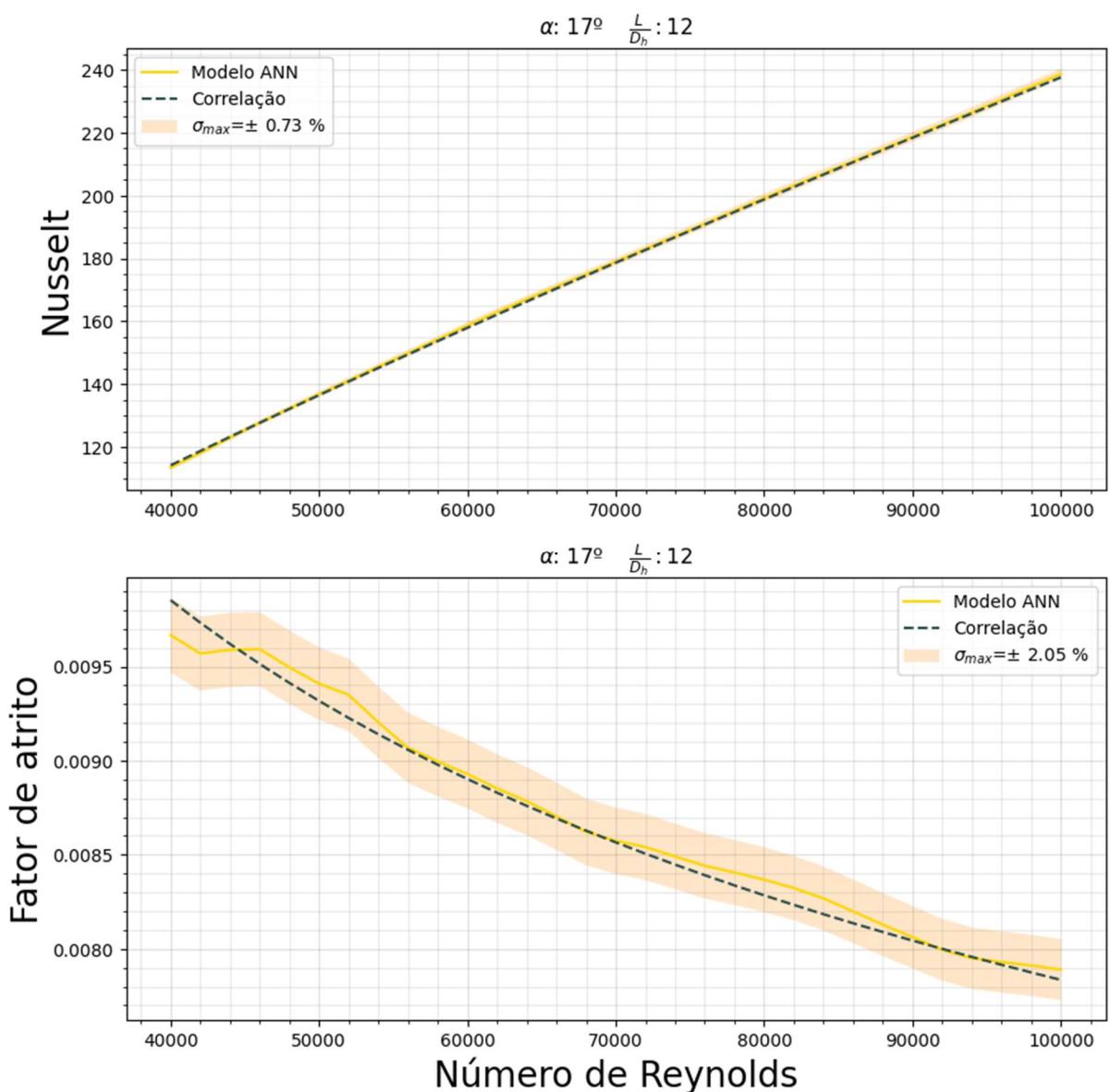


Figura 4.10- Ajuste curva ANN X Correlação

Fonte: Própria

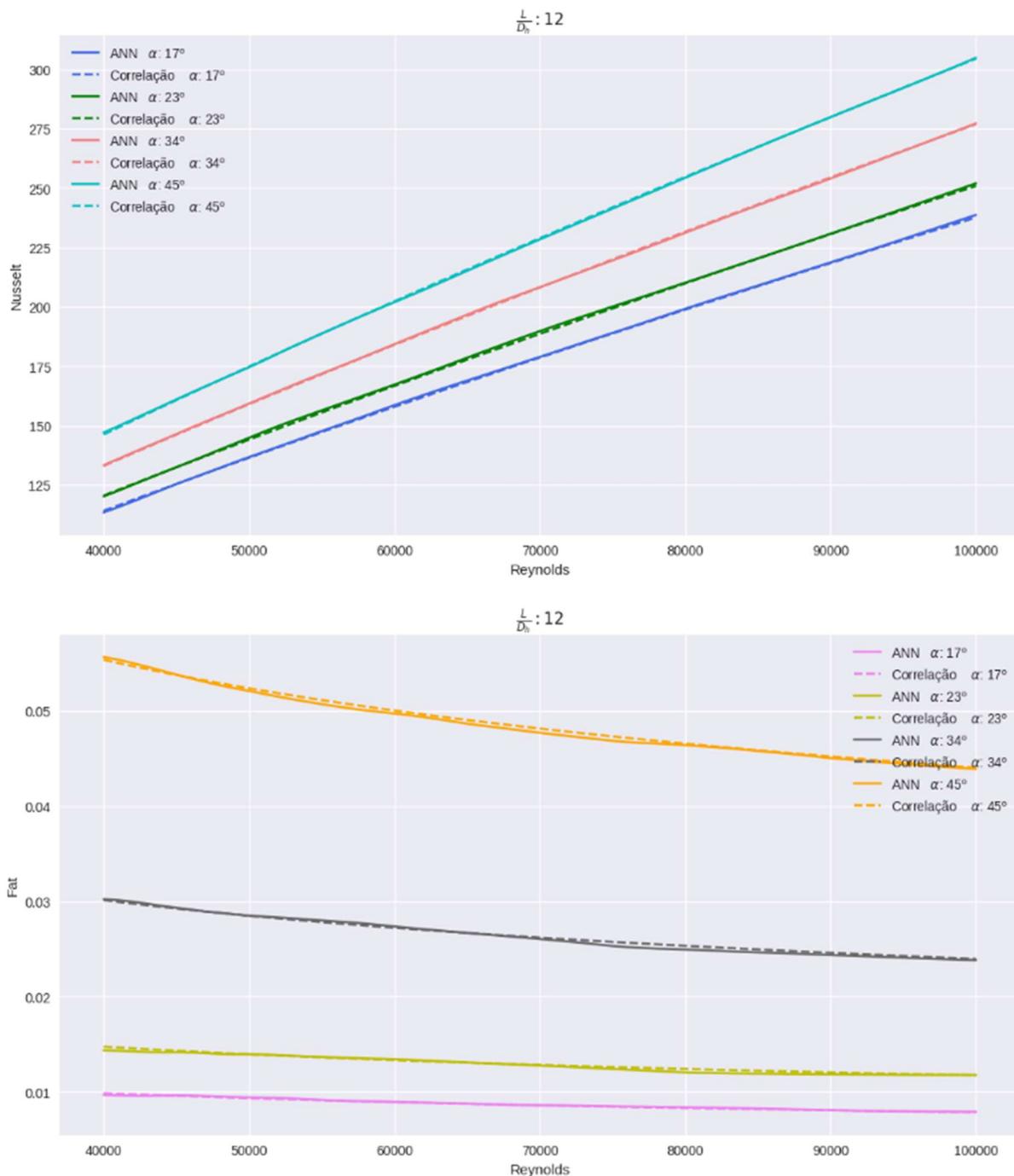


Figura 4.11- Ajuste do modelo a variação do ângulo α

Fonte: Própria

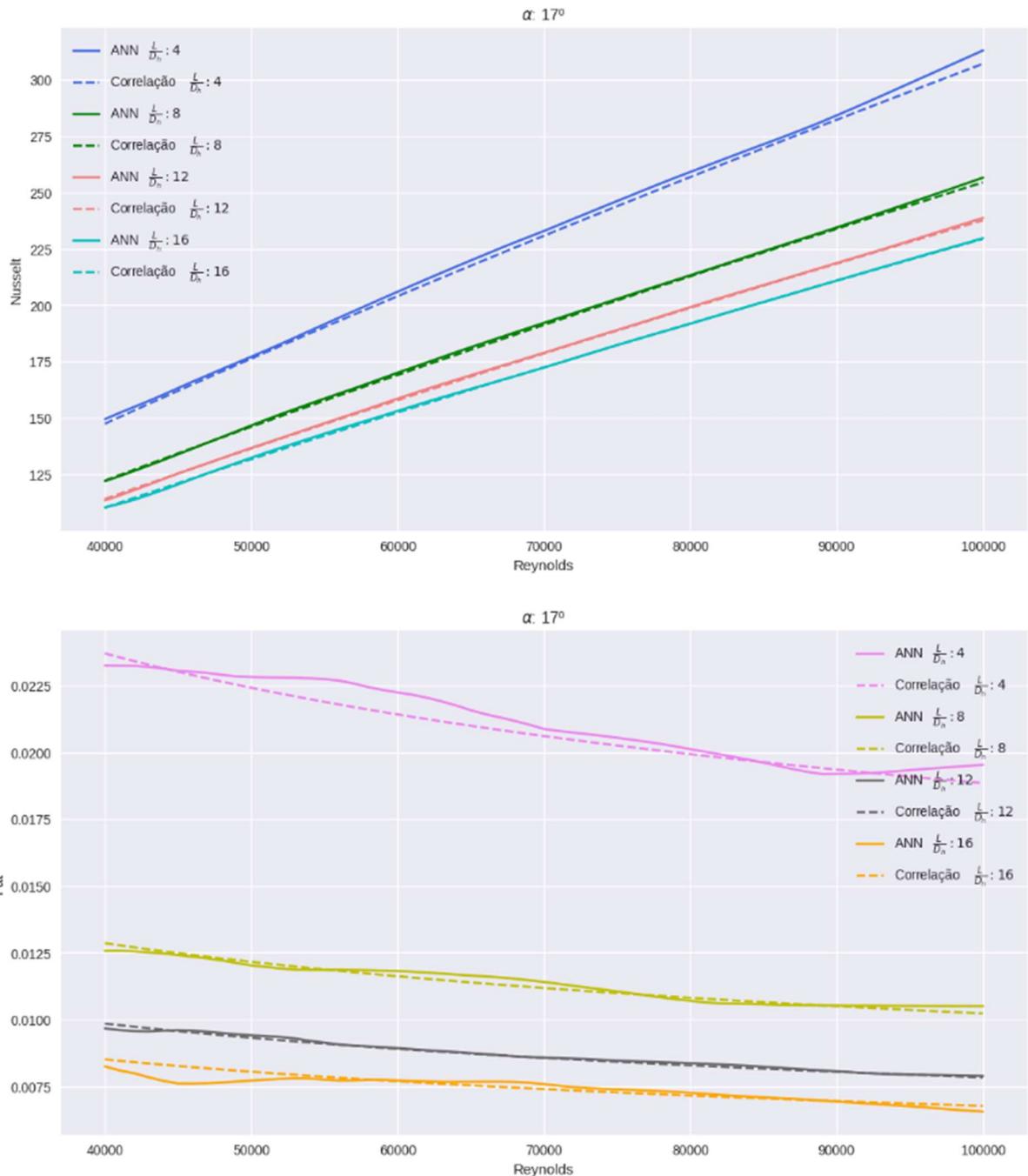


Figura 4.12- Ajuste do modelo ao se variar o ângulo $\frac{L}{D_h}$.

Fonte: Própria

4.2 Análise Correlação com ruído branco

Com a finalidade de testar a rede neural diante da incerteza das medidas do input, foram feitas simulações inserindo essa característica. As estimativas foram baseadas no trabalho de [10]. Para calcular o valor do erro $\frac{L}{D_h}$ foi necessário utilizar a fórmula de propagação de erros, equação 4.7.

$$\sigma_{\frac{L}{D_h}} = \sqrt{\left(\frac{\sigma_L}{D_h}\right)^2 + \left(\frac{L \times \sigma_{D_h}}{D_h^2}\right)^2} \quad (4.7)$$

Foram simulados dois casos, os quais se diferenciam na magnitude do desvio padrão do $\frac{L}{D_h}$. Além disso, para cada caso realizou-se testes com duas e três camadas intermediárias, a fim de avaliar a performance de cada estrutura. Embora a configuração de três camadas intermediárias possua mais neurônios, esse não é o motivo desses testes, pois a literatura já disse que esse aumento de neurônios não representa ganho significativo de predição; portanto, o que se buscou mostrar com esses testes foi a capacidade de acrescentar uma nova camada com uma função transferência diferente. Vale lembrar que cada acréscimo de camada intermediária a quantidade de simulações é multiplicada por dez. Desse modo, enquanto duas camadas intermediárias possuem cem combinações diferentes de função transferência, a configuração com três possui mil combinações.

4.2.1 Caso 1

Tabela 4.17- Propagação de erro

PARÂMETRO	VALOR	DESVIO PADRÃO
L	8,76 mm	$\pm 0,1$ mm
D_h	0,721 mm	$\pm 0,01$ mm
$\frac{L}{D_h}$	12,15mm	$\pm 0,22$

Fonte: Própria

Após o cálculo da propagação do desvio padrão de L e D_h para o $\frac{L}{D_h}$, é possível montar a tabela de desvio padrão dos dados do input.

Tabela 4.18- Tabela de desvio padrão do input (ruído branco)

PARÂMETRO	FAIXA	DESVIO PADRÃO
Re	[40.000; 100.000]	± 5000
α	[15º; 45º]	$\pm 2^\circ$
$\frac{L}{D_h}$	[2,8; 19,3]	$\pm 0,22$

Fonte: Própria

A) Duas camadas intermediárias

Tabela 4.19- Características das simulações

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1 $Re \pm 1\%$ $\alpha \pm 2^\circ$
Ruído	Fixo	$\frac{L}{D_h} \pm 0,14$
Split_size	Fixo	30%
Regularização	Variável	MinMax ou PowerTransformation ou RobustScaler
Camadas intermediárias	Fixo	2
Neurônios	Variável	500 ou 1000 ou 2000 ou 4000
Batch_size	Fixo	32
Nº épocas	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

Tabela 4.20- Melhores estruturas de rede para 2 camadas

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
252	30.0	MinMax	swish/gelu	2	1000	Adam	32	200	0min8s	0.966463	0.001074
252	30.0	MinMax	swish/swish	2	2000	Adam	32	200	0min8s	0.966126	0.001085
252	30.0	MinMax	relu/tanh	2	1000	Adam	32	200	0min7s	0.965699	0.001099
252	30.0	MinMax	swish/gelu	2	4000	Adam	32	200	0min11s	0.965684	0.001099
252	30.0	MinMax	elu/gelu	2	1000	Adam	32	200	0min8s	0.965610	0.001102
252	30.0	MinMax	relu/softsign	2	2000	Adam	32	200	0min8s	0.965105	0.001118
252	30.0	MinMax	softsign/swish	2	1000	Adam	32	200	0min8s	0.965031	0.001120
252	30.0	MinMax	selu/relu	2	2000	Adam	32	200	0min8s	0.964940	0.001123
252	30.0	MinMax	elu/swish	2	500	Adam	32	200	0min7s	0.963610	0.001166
252	30.0	MinMax	relu/swish	2	1000	Adam	32	200	0min8s	0.963576	0.001167
252	30.0	MinMax	softsign/gelu	2	1000	Adam	32	200	0min8s	0.963488	0.001170
252	30.0	MinMax	relu/sigmoid	2	1000	Adam	32	200	0min7s	0.963318	0.001175
252	30.0	MinMax	gelu/gelu	2	1000	Adam	32	200	0min8s	0.963215	0.001178
252	30.0	MinMax	softsign/swish	2	4000	Adam	32	200	0min10s	0.962990	0.001186
252	30.0	MinMax	elu/relu	2	500	Adam	32	200	0min8s	0.962360	0.001206
252	30.0	MinMax	relu/sigmoid	2	500	Adam	32	200	0min7s	0.962214	0.001210
252	30.0	MinMax	gelu/relu	2	500	Adam	32	200	0min8s	0.961950	0.001219
252	30.0	MinMax	relu/relu	2	500	Adam	32	200	0min8s	0.961839	0.001222
252	30.0	MinMax	relu/swish	2	4000	Adam	32	200	0min10s	0.961775	0.001224
252	30.0	MinMax	tanh/swish	2	4000	Adam	32	200	0min11s	0.961771	0.001225

Fonte: Própria

Assim, foram realizadas 100 simulações com a melhor configuração da rede neural - a primeira linha da Tabela 4.20. O resultado pode ser observado na Tabela 4.21.

Tabela 4.21- Impacto no ajusto devido ao ruído branco (2 camadas intermediárias)

MEDIDAS	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,0011	480,58	$1,71 \times 10^{-4}$
s_{MSE}	$3,65 \times 10^{-4}$	146,39	$8,84 \times 10^{-5}$
$\bar{x}_{\text{Erro Máx}}$	-	21,49%	23,54%
$s_{\text{Erro Máx}}$	-	10,32%	18,98%

Fonte: Própria

B) Três camadas intermediárias

Tabela 4.22- Características das simulações com 3 camadas intermediárias

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1 $Re \pm 5000$ $\alpha \pm 2^\circ$
Ruído	Fixo	$\frac{L}{D_h} \pm 0,14$
Split_size	Fixo	30%
Regularização	Fixo	MinMax
Camadas intermediárias	Fixo	3
Neurônios	Fixo	1000
Batch_size	Fixo	32
Nº épocas	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

Foram realizadas 1000 simulações, pois existia em cada camada intermediária 10 possibilidades de função transferência. As melhores combinações de função transferência estão listadas na Tabela 4.23.

Tabela 4.23- Melhores estruturas de rede para 3 camadas.

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
252	30.0	MinMax	softsign/relu/relu	3	1000	Adam	32	200	0min8s	0.966619	0.000815
252	30.0	MinMax	relu/softsign/sigmoid	3	1000	Adam	32	200	0min8s	0.961640	0.000936
252	30.0	MinMax	relu/elu/relu	3	1000	Adam	32	200	0min8s	0.961095	0.000950
252	30.0	MinMax	softsign/softmax/relu	3	1000	Adam	32	200	0min8s	0.958159	0.001021
252	30.0	MinMax	swish/softsign/relu	3	1000	Adam	32	200	0min8s	0.958029	0.001024
252	30.0	MinMax	relu/elu/elu	3	1000	Adam	32	200	0min8s	0.957630	0.001034
252	30.0	MinMax	tanh/softmax/gelu	3	1000	Adam	32	200	0min8s	0.957520	0.001037
252	30.0	MinMax	selu/tanh/relu	3	1000	Adam	32	200	0min7s	0.957433	0.001039
252	30.0	MinMax	elu/softmax/swish	3	1000	Adam	32	200	0min8s	0.957416	0.001039
252	30.0	MinMax	swish/selu/relu	3	1000	Adam	32	200	0min8s	0.957096	0.001047
252	30.0	MinMax	relu/tanh/softsign	3	1000	Adam	32	200	0min8s	0.956926	0.001051
252	30.0	MinMax	relu/tanh/relu	3	1000	Adam	32	200	0min8s	0.956170	0.001070
252	30.0	MinMax	relu/selu/relu	3	1000	Adam	32	200	0min8s	0.956154	0.001070
252	30.0	MinMax	relu/tanh/sigmoid	3	1000	Adam	32	200	0min8s	0.956056	0.001073
252	30.0	MinMax	softsign/softmax/gelu	3	1000	Adam	32	200	0min8s	0.955853	0.001077
252	30.0	MinMax	relu/softsign/tanh	3	1000	Adam	32	200	0min8s	0.955785	0.001079
252	30.0	MinMax	relu/tanh/selu	3	1000	Adam	32	200	0min8s	0.955615	0.001083
252	30.0	MinMax	softsign/relu/softsign	3	1000	Adam	32	200	0min8s	0.955442	0.001087
252	30.0	MinMax	gelu/softmax/elu	3	1000	Adam	32	200	0min8s	0.955432	0.001088
252	30.0	MinMax	gelu/softmax/tanh	3	1000	Adam	32	200	0min8s	0.955360	0.001089

Fonte: Própria

Assim, foram realizadas 100 simulações com a melhor configuração da rede neural - a primeira linha da Tabela 4.23. O resultado pode ser observado na Tabela 4.24.

Tabela 4.24- Impacto no ajusto devido ao ruído branco (3 camadas intermediárias)

MEDIDAS	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,001462	509,74	$2,36 \times 10^{-4}$
s_{MSE}	$5,35 \times 10^{-4}$	201,36	$1,10 \times 10^{-4}$
$\bar{x}_{Erro Máx}$	-	17,95%	29,46%
$s_{Erro Máx}$	-	4,86%	15,93%

Fonte: Própria

4.2.2 Caso 2

As simulações para o caso 2 são semelhantes ao caso 1, só que os valores dos ruídos do input foram alterados de acordo com a Tabela 4.25.

Tabela 4.25- Ruído input para o caso 2

PARÂMETRO	FAIXA	DESVIO PADRÃO
Re	[40.000; 100.000]	± 5000
α	[15º; 45º]	$\pm 2º$
$\frac{L}{D_h}$	[2,8; 19,3]	$\pm 0,5$

Fonte: Própria

A) Duas camadas intermediárias

Tabela 4.26- Características das simulações

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1
Ruído	Fixo	$Re \pm 1\%$ $\alpha \pm 2^\circ$ $\frac{L}{D_h} \pm 0,5$
Split_size	Fixo	30%
Regularização	Variável	MinMax ou PowerTransformation ou RobustScaler
Camadas intermediárias	Fixo	2
Neurônios	Variável	500 ou 1000 ou 2000 ou 4000
Batch_size	Fixo	32
Nº épocas	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

Tabela 4.27- Melhores estruturas de rede para 2 camadas

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
252	30.0	MinMax	swish/gelu	2	4000	Adam	32	200	0min10s	0.924786	0.001485
252	30.0	MinMax	swish/swish	2	4000	Adam	32	200	0min10s	0.922108	0.001538
252	30.0	MinMax	tanh/gelu	2	2000	Adam	32	200	0min8s	0.921170	0.001556
252	30.0	MinMax	gelu/gelu	2	1000	Adam	32	200	0min8s	0.919705	0.001585
252	30.0	MinMax	swish/softsign	2	500	Adam	32	200	0min7s	0.919662	0.001586
252	30.0	MinMax	swish/selu	2	1000	Adam	32	200	0min7s	0.919214	0.001595
252	30.0	MinMax	selu/elu	2	1000	Adam	32	200	0min7s	0.917007	0.001638
252	30.0	MinMax	tanh/gelu	2	1000	Adam	32	200	0min7s	0.916304	0.001652
252	30.0	MinMax	swish/swish	2	500	Adam	32	200	0min8s	0.914569	0.001686
252	30.0	MinMax	swish/swish	2	1000	Adam	32	200	0min7s	0.914392	0.001690
252	30.0	MinMax	gelu/gelu	2	500	Adam	32	200	0min8s	0.913938	0.001699
252	30.0	MinMax	elu/swish	2	500	Adam	32	200	0min8s	0.913730	0.001703
252	30.0	MinMax	selu/tanh	2	2000	Adam	32	200	0min8s	0.913529	0.001707
252	30.0	MinMax	softsign/selu	2	500	Adam	32	200	0min7s	0.913529	0.001707
252	30.0	MinMax	softsign/swish	2	500	Adam	32	200	0min7s	0.913382	0.001710
252	30.0	MinMax	softsign/swish	2	1000	Adam	32	200	0min7s	0.913172	0.001714
252	30.0	MinMax	gelu/swish	2	4000	Adam	32	200	0min10s	0.912848	0.001720
252	30.0	MinMax	selu/swish	2	500	Adam	32	200	0min7s	0.912576	0.001726
252	30.0	MinMax	softsign/gelu	2	500	Adam	32	200	0min8s	0.912141	0.001734
252	30.0	MinMax	tanh/softmax	2	500	Adam	32	200	0min7s	0.912076	0.001736

Fonte: Própria

Assim, foram realizadas 100 simulações com a melhor configuração da rede neural - a primeira linha da Tabela 4.27. O resultado pode ser observado na Tabela 4.28.

Tabela 4.28- Impacto no ajusto devido ao ruído branco (2 camadas intermediárias)

MEDIDAS	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,002110	758,06	$3,53 \times 10^{-4}$
s_{MSE}	$6,27 \times 10^{-4}$	212,94	$1,57 \times 10^{-4}$
$\bar{x}_{\text{Erro Máx}}$	-	22,90%	39,46%
$s_{\text{Erro Máx}}$	-	4,84%	16,78%

Fonte: Própria

B) Três camadas intermediárias

Tabela 4.29- Características das simulações com 3 camadas intermediárias

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Conjunto 1
Ruído	Fixo	$Re \pm 5000$ $\alpha \pm 2^\circ$ $\frac{L}{D_h} \pm 0,5$
Split_size	Fixo	30%
Regularização	Fixo	MinMax
Camadas intermediárias	Fixo	3
Neurônios	Fixo	1000
Batch_size	Fixo	32
Nº épocas	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

Tabela 4.30- Melhores estruturas de rede para 3 camadas.

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
252	30.0	MinMax	selu/sigmoid/softsign	3	1000	Adam	32	200	0min7s	0.930649	0.001799
252	30.0	MinMax	gelu/gelu/gelu	3	1000	Adam	32	200	0min8s	0.930211	0.001810
252	30.0	MinMax	swish/gelu/sigmoid	3	1000	Adam	32	200	0min8s	0.926793	0.001899
252	30.0	MinMax	relu/softmax/gelu	3	1000	Adam	32	200	0min8s	0.926061	0.001918
252	30.0	MinMax	swish/swish/gelu	3	1000	Adam	32	200	0min8s	0.925918	0.001922
252	30.0	MinMax	relu/sigmoid/sigmoid	3	1000	Adam	32	200	0min7s	0.925225	0.001940
252	30.0	MinMax	tanh/swish/swish	3	1000	Adam	32	200	0min8s	0.925193	0.001941
252	30.0	MinMax	softsign/swish/sigmoid	3	1000	Adam	32	200	0min8s	0.925076	0.001944
252	30.0	MinMax	relu/sigmoid/tanh	3	1000	Adam	32	200	0min8s	0.924715	0.001953
252	30.0	MinMax	swish/swish/sigmoid	3	1000	Adam	32	200	0min7s	0.924424	0.001961
252	30.0	MinMax	relu/softplus/softplus	3	1000	Adam	32	200	0min7s	0.924376	0.001962
252	30.0	MinMax	selu/swish/elu	3	1000	Adam	32	200	0min8s	0.924295	0.001964
252	30.0	MinMax	swish/softsign/gelu	3	1000	Adam	32	200	0min8s	0.924259	0.001965
252	30.0	MinMax	swish/swish/tanh	3	1000	Adam	32	200	0min8s	0.924148	0.001968
252	30.0	MinMax	relu/relu/tanh	3	1000	Adam	32	200	0min7s	0.923709	0.001979
252	30.0	MinMax	swish/selu/gelu	3	1000	Adam	32	200	0min8s	0.923646	0.001981
252	30.0	MinMax	swish/gelu/softplus	3	1000	Adam	32	200	0min8s	0.923595	0.001982
252	30.0	MinMax	relu/softplus/swish	3	1000	Adam	32	200	0min7s	0.923434	0.001986
252	30.0	MinMax	elu/elu/sigmoid	3	1000	Adam	32	200	0min8s	0.923375	0.001988
252	30.0	MinMax	softsign/swish/gelu	3	1000	Adam	32	200	0min8s	0.923326	0.001989

Fonte: Própria

A simulações desses 1000 casos levou cerca de 4 horas.

Assim, foram realizadas 100 simulações com a configuração da rede neural conforme a primeira linha da Tabela 4.30. O resultado pode ser observado na Tabela 4.31.

Tabela 4.31- Impacto no ajuste devido ao ruído branco (3 camadas intermediárias)

MEDIDAS	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,0044	1283,82	$8,68 \times 10^{-4}$
s_{MSE}	0,0017	518,11	$4,76 \times 10^{-4}$
$\bar{x}_{Erro\ Máx}$	-	28,21%	47,87%
$s_{Erro\ Máx}$	-	5,16%	17,62%

Fonte: Própria

Segundo as análises é possível inferir que: a inclusão da terceira camada intermediária não resultou em melhorias predição; a rede neural consegue lidar com erros na ordem do caso 1; a ANN não apresentou bons resultados de predição no caso 2, especialmente para o fator de atrito.

4.2.3 Análise da influência do ruído de entrada na predição de Nusselt e Fator de Atrito

Esta seção visa à análise da interferência do ruído nos valores de entrada (*input*) na predição do Nusselt e Fator de Atrito. Inicialmente, é avaliado a influência dos desvios do Reynolds no treinamento, em segundo do α e terceiro do $\frac{L}{D_h}$.

Tabela 4.32 – Configuração da rede neural nesta seção de testes

Dados	Split_sample	Função_transf	Camad_Int	Neurônios	Optimizer	Batchs	Epoch
252	30%	Swish/Gelu	2	1000	Adam	32	200

Fonte: Própria

A) Ruído exclusivamente no Re

i) Relação entre o ruído na entrada e o erro na predição

Para analisar a influência do ruído do número de Reynolds no treinamento da rede neural, foi gerado 252 dados e para cada dado foi adicionado um ruído no Re. Os ruídos seguiram um comportamento de uma distribuição normal com média zero e desvio padrão de 5000, vale lembrar que Reynolds opera na faixa de 40.000 e 100.000. O resultado pode ser observado na Tabela 4.33, no qual o MSE global aumentou em uma ordem de grandeza se comparada com o caso sem ruído.

Tabela 4.33 – MSE médio da rede neural

ERRO	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,0007	455	$1,46 \times 10^{-5}$

Fonte: Própria

Visualiza-se na Figura 4.32 a relação entre o erro relativo no Reynolds causa pelo ruído e o impacto dele na predição. Nesse contexto, observa-se que não existe uma linearidade entre erro relativo do ruído e o erro relativo na predição, portanto valores que foram medidos com incerteza e usados para treinar na rede não impactarão na predição desses valores. Conclui-se que a rede neural identifica os pontos que estão com ruído acentuado e reduz sua importância no treinamento, no entanto a rede neural no um todo fica menos precisa.

Além disso, se chega na mesma conclusão quando se analisa o erro relativo do ruído no α e $\frac{L}{D_h}$, Figura 4.13 e 4.14.

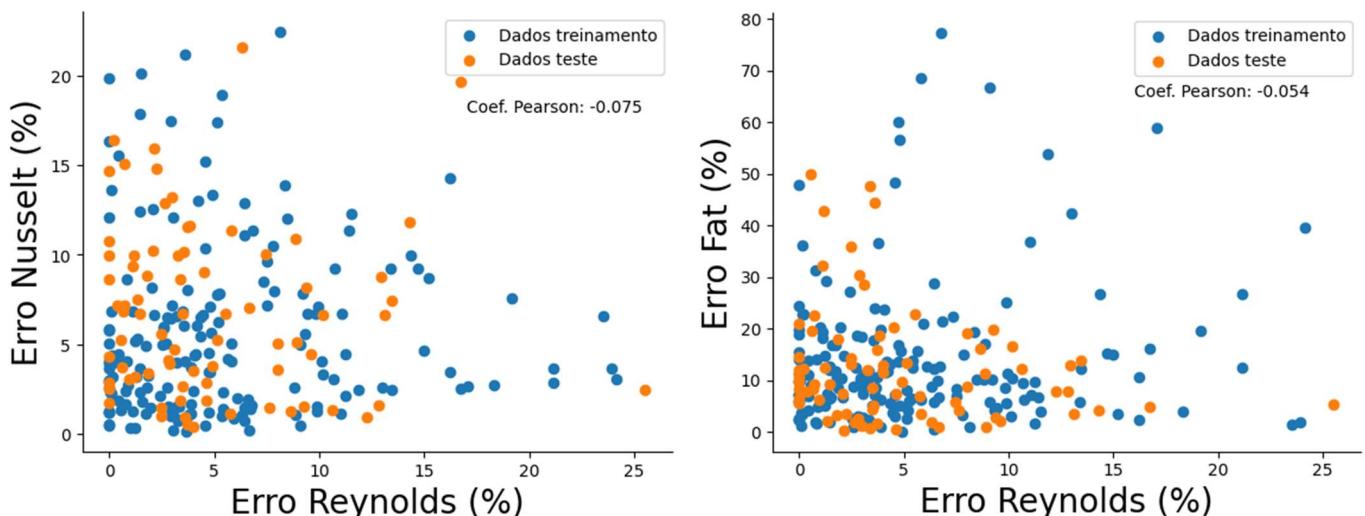


Figura 4.13 – Comparação do erro relativo na entrada durante o treinamento e o erro relativo da saída.

Fonte: Própria

- ii) Coeficiente de Pearson entre o aumento do ruído e o aumento do erro na saída da rede neural.

Nesse tópico foram realizadas 10 simulações e o ruído em cada uma delas foi aumentado, inicialmente de 500, depois um ruído de 1000 até se chegar à um ruído de 5000 Re. Para cada ruído foi calculado o MSE Global da rede neural; o MSE do Nusselt; e o MSE do Fator de atrito. Após determinar os erros, foi calculado, por meio do coeficiente de Pearson, a relação entre o aumento do ruído e o aumento dos MSE, isso é mostrado na Tabela 4.34.

Tabela 4.34 – Pearson entre ruído e MSE

COEFICIENTE DE PEARSON

10 valores de ruído com distância uniformes no intervalo de 500 e 5000 Re

Ruído Re ↔ MSE _{Global}	Ruído Re ↔ MSE _{Nu}	Ruído Re ↔ MSE _{Fat}
0,64	0,87	0,52

Fonte: Própria

B) Ruído exclusivamente no α

- i) Relação entre o ruído na entrada e o erro na predição.

Para analisar a influência do ruído do ângulo no treinamento da rede neural, foi gerado 252 dados e para cada dado foi adicionado um ruído no α . Os ruídos seguiram um comportamento de uma distribuição normal com média zero e desvio padrão de 2° , vale lembrar que α opera na faixa de 15° e 45° . O resultado pode ser observado na Tabela 4.35, no qual o MSE global aumentou em uma ordem de grandeza se comparada com o caso sem ruído.

Tabela 4.35 – MSE médio da rede neural

ERRO	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,0008	130,05	$9,34 \times 10^{-5}$

Fonte: Própria

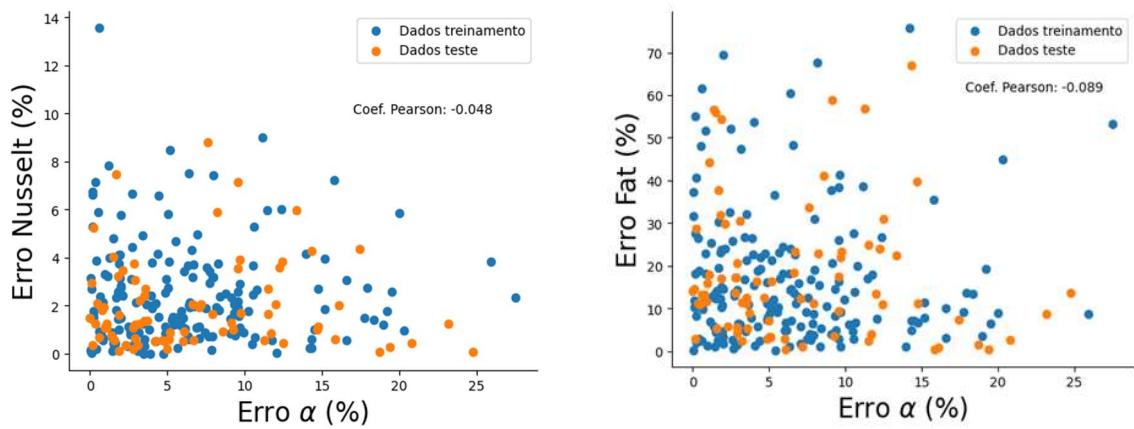


Figura 4.14 - Comparação do erro relativo na entrada durante o treinamento e o erro relativo da saída.

ii) Coeficiente de Pearson entre o aumento do ruído e o aumento do erro na saída da rede neural.

Nesse tópico foram realizadas 11 simulações e o ruído em cada uma delas foi aumentado, conforme a nota da Tabela 4.36. Para cada ruído foi calculado o MSE Global da rede neural; o MSE do Nusselt; e o MSE do Fator de atrito. Após determinar os erros, foi calculado, por meio do coeficiente de Pearson, a relação entre o aumento do ruído e o aumento dos MSE.

Tabela 4.36 – Pearson entre ruído e MSE

COEFICIENTE DE PEARSON

Para 11 valores de ruídos de α crescente.

Ruído $\alpha \leftrightarrow \text{MSE}_{\text{Global}}$

Ruído $\alpha \leftrightarrow \text{MSE}_{\text{Nu}}$

Ruído $\alpha \leftrightarrow \text{MSE}_{\text{Fat}}$

0,91

0,92

0,72

Nota: A sequência de ruído seguiu a seguinte ordem: [0,25; 0,5; 0,75; 1; 1,5; 1,8; 2; 2,2; 2,5; 3; 3,5]

Fonte: Própria

C) Ruído exclusivamente no $\frac{L}{D_h}$

i) Relação entre o ruído na entrada e o erro na predição

Para analisar a influência do ruído do $\frac{L}{D_h}$ no treinamento da rede neural, foi gerado 252 dados e para cada dado foi adicionado um ruído. Os ruídos seguiram um comportamento de uma distribuição normal com média zero e desvio padrão de 0,22, vale lembrar que $\frac{L}{D_h}$ opera na faixa de 2,8 e 19,3. O resultado pode ser observado na Tabela 4.37, no qual o MSE global aumentou em uma ordem de grandeza se comparada com o caso sem ruído, e o impacto do ruído nessa razão impactou menos no MSE do Nusselt do que o ruído no Reynolds e no ângulo.

Tabela 4.37 – MSE médio da rede neural

ERRO	GLOBAL	NUSSELT	FAT
\bar{x}_{MSE}	0,0002	61,1	$3,29 \times 10^{-5}$

Fonte: Própria

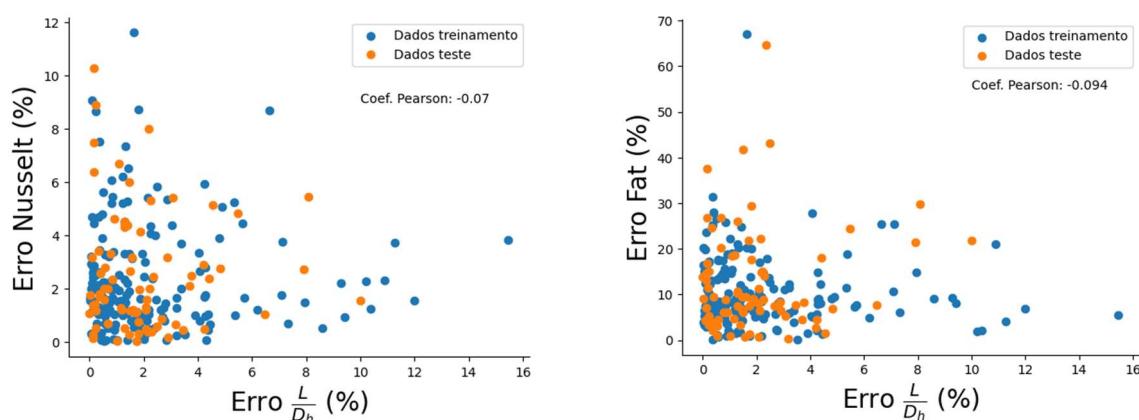


Figura 4.15 - Comparação do erro relativo na entrada durante o treinamento e o erro relativo da

Fonte: Própria

ii) Coeficiente de Pearson entre o aumento do ruído e o aumento do erro na saída da rede neural

Nesse tópico foram realizadas 12 simulações com o ruído crescente em cada uma delas. Para cada ruído foi calculado o MSE Global da rede neural; o MSE do Nusselt; e o MSE do

Fator de atrito. Após determinar os erros, foi calculado, por meio do coeficiente de Pearson, a relação entre o aumento do ruído e o aumento dos MSE – Tabela 4.38.

Tabela 4.38 – Pearson entre ruído na entrada e MSE

COEFICIENTE DE PEARSON

Para 12 valores de ruídos de $\frac{L}{D_h}$ crescente.

Ruído $\frac{L}{D_h} \leftrightarrow \text{MSE}_{\text{Global}}$	Ruído $\frac{L}{D_h} \leftrightarrow \text{MSE}_{\text{Nu}}$	Ruído $\frac{L}{D_h} \leftrightarrow \text{MSE}_{\text{Fat}}$
0,78	0,87	0,65

Nota: A sequência de ruído seguiu a seguinte ordem: [0,02; 0,4; 0,06; 0,08; 0,1; 0,125; 0,14; 0,16; 0,18; 0,2; 0,25; 0,3]

Fonte: Própria

É possível, a partir da Figura 4.16 comparar o impacto que cada ruído causa na predição da ANN. Desse modo, a primeira coluna explora o MSE global, a segunda explora o MSE específico do Nusselt e a terceira o MSE do fator de atrito.

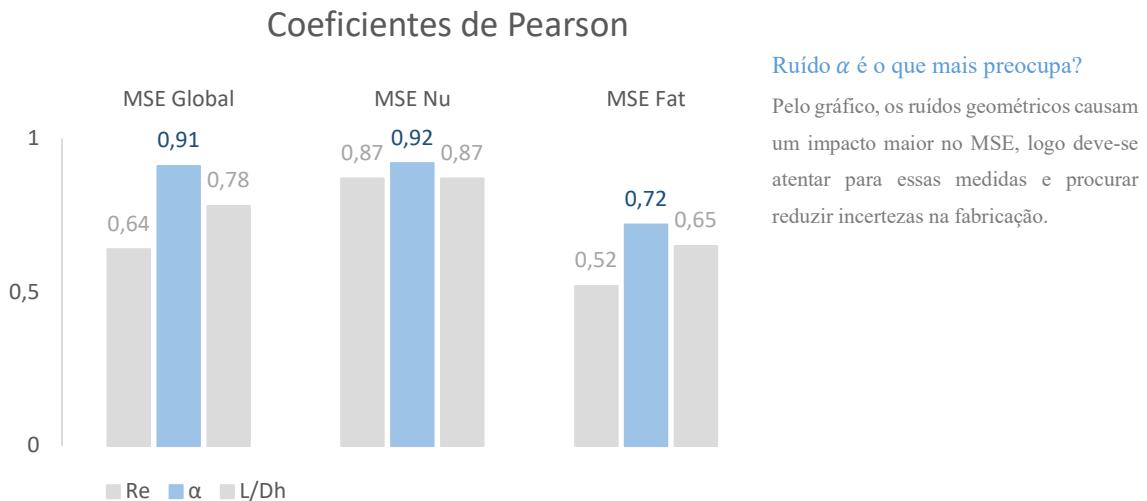


Figura 4.16 - Compilado dos coeficientes de Pearson calculados entre os ruídos de entrada e os erros.

Fonte: Própria

4.3 Análise simulação CFD 3D

4.3.1 Análise do tamanho dos dados de treinamento

A rede neural é influenciada pela quantidade de dados utilizadas para treiná-la. Assim, essa seção analisou essa influência do número de dados no treinamento da rede. Em primeiro lugar, os dados foram divididos de 11 formas diferentes: 5% para teste, 10%, 20%, 30%, até a divisão de 95% dos dados para teste. Para cada tipo de divisão, por exemplo, para a divisão de 30% foram testados 600 tipos de redes neurais diferentes. Os parâmetros dessas simulações estão na Tabelas 4.39.

Tabela 4.39- Parâmetros das simulações

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	Simulação 3D
Split_size	Variável	5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 95%.
Regularização	Fixo	MinMax
Camadas intermediárias	Fixo	2
Neurônios	Variável	50, 100, 500, 1000, 2000, 4000
Batch_size	Fixo	32
Nº épocas	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

A Tabela 4.40 mostram os parâmetros otimizados para cada tipo de tamanho do conjunto de teste e treinamento. A primeira linha da tabela informa que a rede neural otimizada para o conjunto com 5% é a que possui 500 neurônios e utiliza a função transferência Relu na primeira camada intermediária e Elu na segunda camada intermediária.

Tabela 4.40- Melhor estrutura para cada split size

SPLIT_SIZE	NEURÔNIOS	FUNÇÃO TRANSFERÊNCIA	MSE
5%	500	relu/elu	0,000103
10%	2000	relu/relu	0,000104
20%	1000	relu/gelu	0,000310
30%	500	swish/relu	0,000167
40%	2000	relu/tanh	0,000511
50%	500	swish/relu	0,001052
60%	500	elu/relu	0,002250
70%	2000	relu/selu	0,002560
80%	2000	relu/relu	0,002535
90%	100	elu/relu	0,003374
95%	2000	gelu/relu	0,008904

Fonte: Própria

As Tabelas 4.41, 4.42 e 4.43 mostram os resultados de uma análise estatística para os modelos de rede neural da tabela acima. Para cada linha da Tabela 4.40 foram feitas 100 simulações idênticas, só que em cada uma dessas 100 simulações os dados selecionados se alteravam. Isso visou a redução da imprevisibilidade da separação dos dados, pois, ainda não há um método que otimize a divisão de dados em dados de treinamento e dados de teste. Logo, a partir do estudo das tabelas abaixo, é possível concluir que à medida que o número de dados para treinamento diminui a rede neural fica mais imprecisa. Essa perspectiva qualitativa foi devida a análise quantitativa. Uma conclusão prática é que não seria recomendável o treinamento com menos de 119 dados.

Tabela 4.41- Percentual dos dados usados no treinamento da rede

PARÂMETROS	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	95%
\bar{x}_{MSE}^{Global}	0,000917	0,000819	0,000842	0,000770	0,001155	0,001259	0,001449	0,002540	0,003808	0,010811	0,020476
S_{MSE}^{Global}	0,001045	0,000834	0,000620	0,000338	0,000572	0,000707	0,000632	0,001258	0,002167	0,005291	0,010977
Margem de erro _{95%}	$2,04 \times 10^{-4}$	$1,63 \times 10^{-4}$	$1,22 \times 10^{-4}$	$6,638 \times 10^{-5}$	$1,13 \times 10^{-4}$	$1,121 \times 10^{-4}$	$1,239 \times 10^{-4}$	$2,467 \times 10^{-4}$	$4,248 \times 10^{-4}$	0,001037	0,002151

Fonte: Própria

Tabela 4.42- Percentual dos dados usados no treinamento da rede

PARÂMETROS	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	95%
$\bar{x}_{MSE}^{Nusselt}$	85,27	62,29	64,41	87,23	128,96	145,34	154,22	288,67	284,58	1132,47	3107,30
$S_{MSE}^{Nusselt}$	108,53	46,19	42,79	41,59	72,12	93,56	83,98	127,64	152,89	893,77	2864,73
$\bar{x}_{Erro Máx}^{Nusselt}$	4,59%	4,97%	5,56%	7,22 %	9,42%	10,23%	11,20%	18,10%	15,09%	26,74%	44,37%
$S_{Erro Máx}^{Nusselt}$	2,24%	1,79%	2,10%	2,27%	4,77%	4,04%	4,13%	12,60%	3,56%	13,76%	26,22%

Fonte: Própria

Tabela 4.43- Percentual dos dados usados no treinamento da rede

PARÂMETROS	5%	10%	20%	30%	40%	50%	60%	70%	80%	90%	95%
\bar{x}_{MSE}^{Fat}	$6,77 \times 10^{-5}$	$6,40 \times 10^{-5}$	$6,57 \times 10^{-5}$	$5,29 \times 10^{-5}$	$7,99 \times 10^{-5}$	$8,58 \times 10^{-5}$	$1,02 \times 10^{-4}$	$1,74 \times 10^{-4}$	$2,98 \times 10^{-4}$	$7,66 \times 10^{-4}$	0,001209
S_{MSE}^{Fat}	$9,04 \times 10^{-5}$	$7,95 \times 10^{-5}$	$5,64 \times 10^{-5}$	$2,82 \times 10^{-5}$	$5,03 \times 10^{-5}$	$5,53 \times 10^{-5}$	$5,12 \times 10^{-5}$	$1,10 \times 10^{-4}$	$2,00 \times 10^{-4}$	$3,99 \times 10^{-4}$	0,000683
$\bar{x}_{Erro Máx}^{Fat}$	14,61%	15,64%	18,47%	17,72%	20,84%	21,00%	24,87%	37,08%	44,95%	74,19%	124,74%
$S_{Erro Máx}^{Fat}$	9,90%	8,10%	9,86%	7,29%	8,41%	8,29%	11,38%	20,25%	19,77%	81,71%	249,72%

Fonte: Própria

4.3.2 Estrutura da ANN

Para encontrar a melhor rede neural para o conjunto de dados oriundo da simulação em CFD, foram feitos testes com 1400 redes neurais e escolhida aquela que apresenta o menor valor para o MSE. Os parâmetros das simulações variaram conforme a Tabela 4.44.

Tabela 4.44- Simulações realizadas para os dados CFD

PARÂMETRO	CONDIÇÃO	VALOR
Dados	Fixo	119
Split_size	Fixo	10%
Regularização	Variável	MinMax ou PowerTransformation
Camadas intermediárias	Fixo	2
Neurônios	Variável	25 ou 50 ou 100 ou 500 ou 1000 ou 2000 ou 4000
Batch_size	Fixo	32
Nº ÉPOCAS	Fixo	200
Função transferência	Variável	Relu, sigmoid, tanh, swish, elu, gelu, selu, softplus, softsign e softmax
Função otimização	Fixo	Adam
Função erro	Fixo	MSE

Fonte: Própria

O total de casos simulados foi de 1400, 10 tipos de função na primeira camada, 10 tipo de função de transferência na segunda camada, 2 tipos de regularização e 7 tipos de quantidade de neurônio. As tabelas abaixo apresentam os 10 melhores casos para o tipo de regularização MinMax e as 10 melhores configurações de ANN para a regularização *PowerTransformation*.

Tabela 4.45- 10 menores MSE para a Transformada de potência

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
119	10.0	PowerTrans	relu/gelu	2	1000	Adam	32	200	0min7s	0.998235	0.001850
119	10.0	PowerTrans	selu/relu	2	2000	Adam	32	200	0min7s	0.997864	0.002239
119	10.0	PowerTrans	relu/swish	2	2000	Adam	32	200	0min7s	0.997736	0.002374
119	10.0	PowerTrans	relu/softsign	2	1000	Adam	32	200	0min6s	0.997720	0.002391
119	10.0	PowerTrans	swish/relu	2	4000	Adam	32	200	0min8s	0.997656	0.002458
119	10.0	PowerTrans	relu/selu	2	1000	Adam	32	200	0min6s	0.997610	0.002506
119	10.0	PowerTrans	tanh/relu	2	4000	Adam	32	200	0min8s	0.997549	0.002569
119	10.0	PowerTrans	softsign/selu	2	2000	Adam	32	200	0min7s	0.997544	0.002575
119	10.0	PowerTrans	elu/relu	2	4000	Adam	32	200	0min8s	0.997531	0.002589
119	10.0	PowerTrans	gelu/relu	2	500	Adam	32	200	0min7s	0.997515	0.002605

Fonte: Própria

Deve se analisar a terceira e última coluna da Tabela 4.45 e comparar com a Tabela 4.46. Por conseguinte, é de fácil observação que o tipo de regularização MinMax apresentou melhores resultados que PowerTrans.

Tabela 4.46- 10 menores MSE para a MinMax

Dados	Split_sample(%)	Regularization	Funcao_Transf	Cama_Int	Neuronios	Optimizer	Batchs	Epochs	Tempo_treino	R_2	MSE
119	10.0	MinMax	relu/relu	2	500	Adam	32	200	0min6s	0.993791	0.000176
119	10.0	MinMax	relu/selu	2	2000	Adam	32	200	0min7s	0.992854	0.000202
119	10.0	MinMax	selu/relu	2	500	Adam	32	200	0min6s	0.992616	0.000209
119	10.0	MinMax	elu/relu	2	2000	Adam	32	200	0min7s	0.992606	0.000209
119	10.0	MinMax	relu/elu	2	500	Adam	32	200	0min6s	0.992382	0.000216
119	10.0	MinMax	softsign/relu	2	2000	Adam	32	200	0min7s	0.992315	0.000218
119	10.0	MinMax	softsign/relu	2	500	Adam	32	200	0min6s	0.992290	0.000218
119	10.0	MinMax	swish/relu	2	1000	Adam	32	200	0min7s	0.992216	0.000220
119	10.0	MinMax	relu/tanh	2	500	Adam	32	200	0min6s	0.991892	0.000230
119	10.0	MinMax	relu/elu	2	4000	Adam	32	200	0min8s	0.991475	0.000241

Fonte: Própria

4.3.3 Imprevisibilidade devido a divisão dos dados

A divisão dos dados de treinamento e de teste é uma etapa importante para o treinamento da rede neural, essa seção visa a mostrar as consequências de se alterar os dados que fazem parte do treinamento e dos que fazem parte do teste. A Tabela 4.47 constem a média e desvio padrão de cem testes realizados alterando os dados que pertenciam ao grupo de teste e de treinamento, lembrando que a porcentagem de separação se manteve. Para a rede neural treinada seja válida seu MSE precisa estar no intervalo $IC_{95\%}$.

Tabela 4.47- Parâmetros para cálculo do Intervalo de confiança de 95%

PARÂMETRO	VALOR
n	100
\bar{x}_{MSE}	$8,35 \times 10^{-4}$
s_{MSE}	$7,77 \times 10^{-4}$
$z_{95\%}$	1,96
Margem de erro	$1,52 \times 10^{-4}$
$IC_{95\%}$	$6,83 \times 10^{-4} \leq x_i - \bar{x} \leq 9,87 \times 10^{-4}$

Fonte: Própria

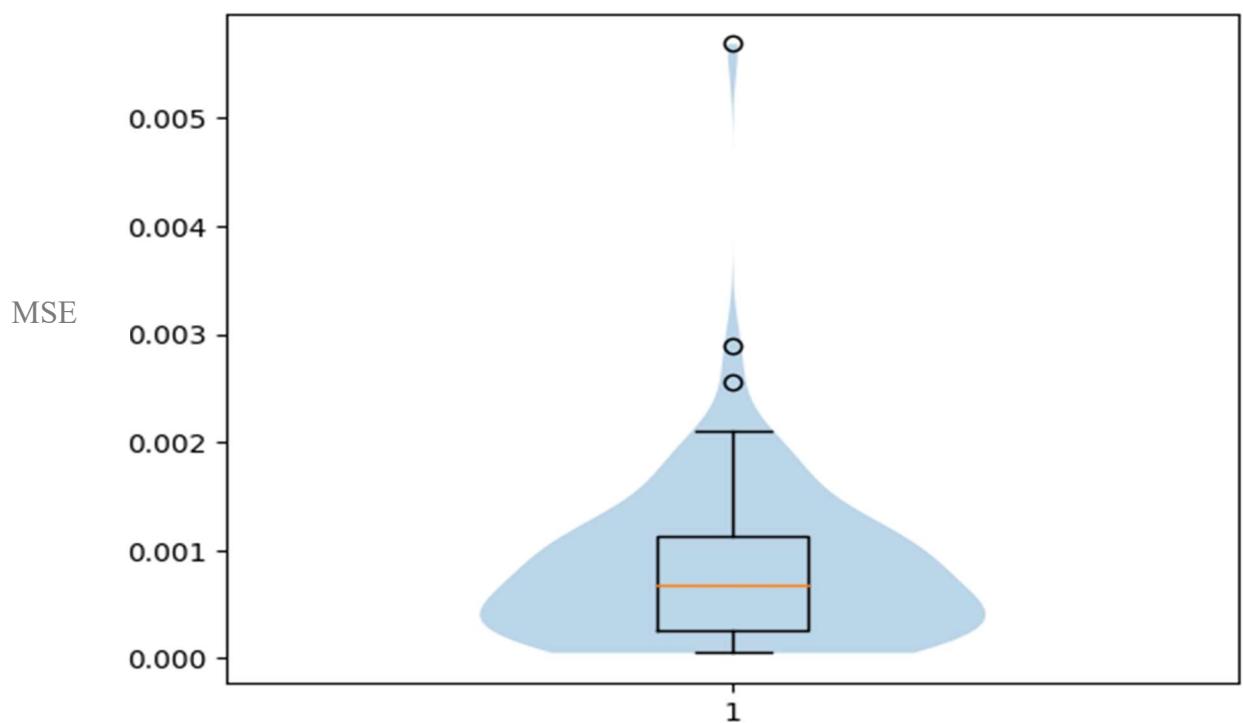


Figura 4.17- BoxPlot

Fonte: Própria

4.3.4 Performance da ANN

Nesta seção será avaliada a performance de previsão da rede neural artificial, os dados para treinamento serão os 119 dados da simulação CFD 3D, sendo 30% destinado para teste e 70% para treinamento da rede, as demais configurações da rede segue de acordo com a Tabela 4.48

Tabela 4.48 – Configuração da ANN avaliada

Dados	Split_sample	Função_transf	Camad_Int	Neurônios	Optimizer	Batchs	Epoch	R_2	MSE
119	30%	Relu/Relu	2	500	Adam	32	200	0,98	0,0007

Fonte: Própria

A Tabela 4.49 e 4.50 compara os resultados da rede neural com a correlação, quando ambas são usadas para prever os resultados do CFD. Os pontos usados para comparação na tabela são os próprios pontos usados para treinar a rede neural e a correlação. É possível

Tabela 4.49- Comparaçao dos resultados para Nusselt entre ANN e Correlaçao

Re	alpha	LDh	Nu 3D	Nu ANN	Nu Corr	Erro Rel. ANN (%)	Erro Rel. Corr (%)
40000.0	30.0	3.155976	239.727903	239.760773	235.957266	0.013711	1.572882
90000.0	15.0	4.715950	257.982852	257.857361	274.927061	0.048643	6.567959
100000.0	15.0	2.829570	350.034547	349.754364	352.970439	0.080044	0.838744
100000.0	15.0	14.147851	238.674375	238.981339	251.081014	0.128612	5.198145
100000.0	15.0	9.000000	248.425037	248.824173	266.598166	0.160667	7.315337
50000.0	15.0	9.000000	141.910529	142.169418	156.528441	0.182431	10.300794
80000.0	15.0	2.829570	293.533918	294.116699	295.457482	0.198540	0.655313
70000.0	45.0	3.865265	398.121342	397.245270	429.349205	0.220052	7.843805
80000.0	30.0	15.779878	241.658259	241.054962	234.489386	0.249649	2.966534
90000.0	22.5	9.000000	257.320937	258.027222	269.032025	0.274476	4.551160
70000.0	17.5	12.143189	190.640005	191.195038	198.978905	0.291142	4.374161
80000.0	45.0	19.326324	263.837307	264.828339	247.415408	0.375622	6.224252

Fonte: Própria

observar nas duas últimas colunas o erro relativo de cada método, e conclui-se que a rede neural obteve resultados melhores que a correlação, tanto para o Nusselt - Tabela 4.49-, quanto para o Fator de atrito, -Tabela 4.50.

Tabela 4.50- Comparaçao dos resultados para Fat entre ANN e Correlação

Re	alpha	LDh	Fat 3D	Fat ANN	Fat Corr	Erro Rel. ANN (%)	Erro Rel. Corr (%)
80000.0	45.0	6.442108	0.087504	0.087653	0.099768	0.169997	14.015161
90000.0	15.0	4.715950	0.010758	0.010845	0.013248	0.810827	23.141538
80000.0	15.0	2.829570	0.028285	0.027926	0.022232	1.269061	21.400750
80000.0	30.0	5.259959	0.042161	0.042705	0.049295	1.291222	16.922107
60000.0	45.0	9.000000	0.075853	0.074733	0.070799	1.476804	6.662906
40000.0	30.0	9.000000	0.030208	0.031022	0.031720	2.694308	5.003491
50000.0	45.0	9.000000	0.076084	0.073555	0.074101	3.324450	2.607018
90000.0	45.0	9.000000	0.079946	0.077202	0.063974	3.432183	19.978491
40000.0	22.5	4.930576	0.025950	0.024912	0.033514	4.001141	29.151062
70000.0	30.0	5.259959	0.043249	0.041469	0.050969	4.114531	17.850015
80000.0	30.0	15.779878	0.019823	0.020737	0.015185	4.611140	23.398239
100000.0	45.0	3.865265	0.201585	0.211301	0.180722	4.819748	10.349276

Fonte: Própria

As Figuras 4.18 e 4.19 representam graficamente as comparações feitas com: CFD 3D, rede neural e correlação. Os pontos vermelhos representam o valor CFD e é a referência; a curva amarela, rede neural; área alaranjada, erro máximo de previsão da rede neural; curva azul, correlação.

Para o Nusselt ambos os métodos se ajustarem bem, no entanto, para o Fator de atrito a rede neural mostrou um ajuste melhor.

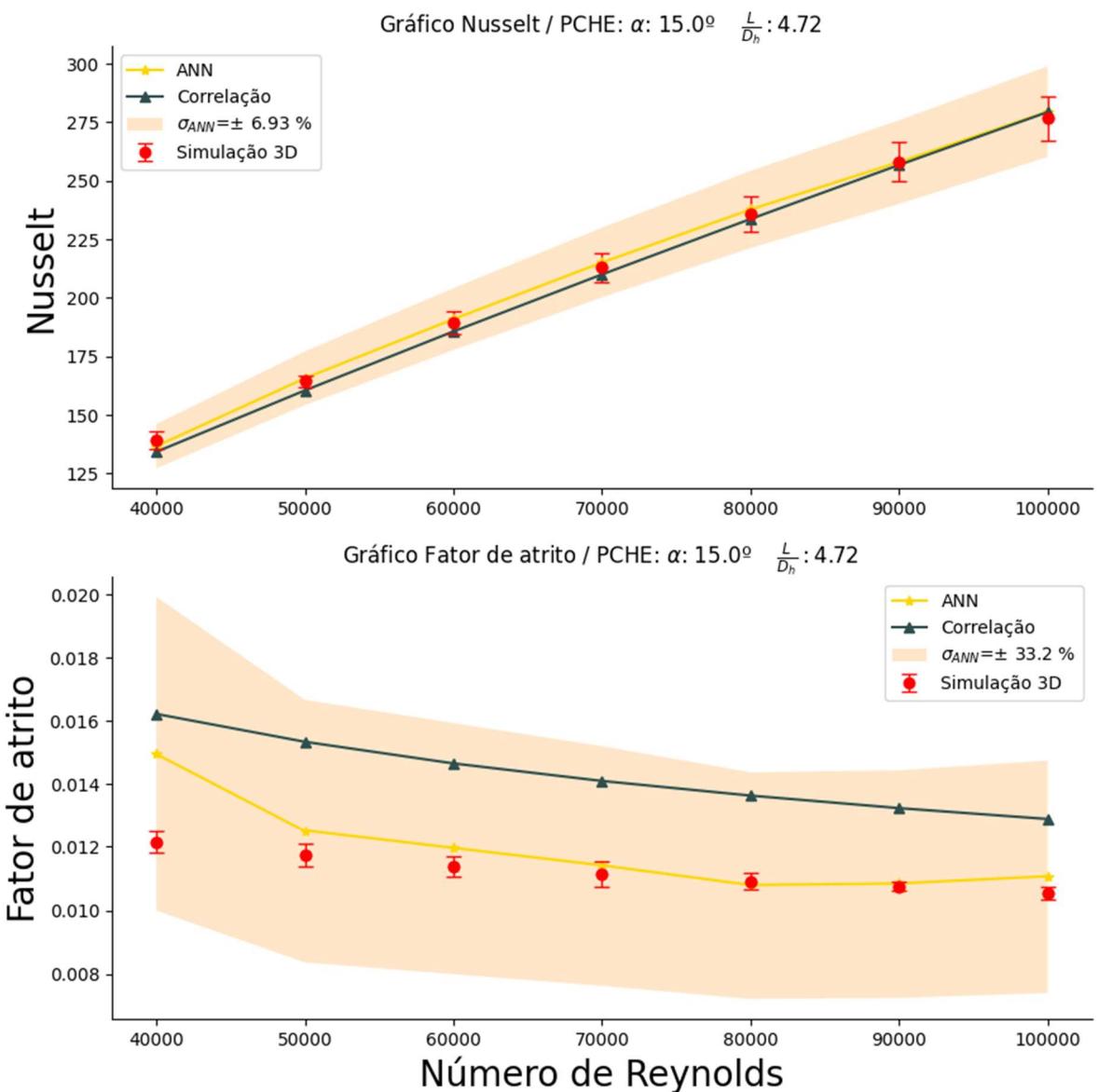


Figura 4.18- Comparaçāo do ajuste da ANN e Correlaçāo para α e

$$\frac{L}{D_h} \text{ pequenos}$$

Fonte: Própria

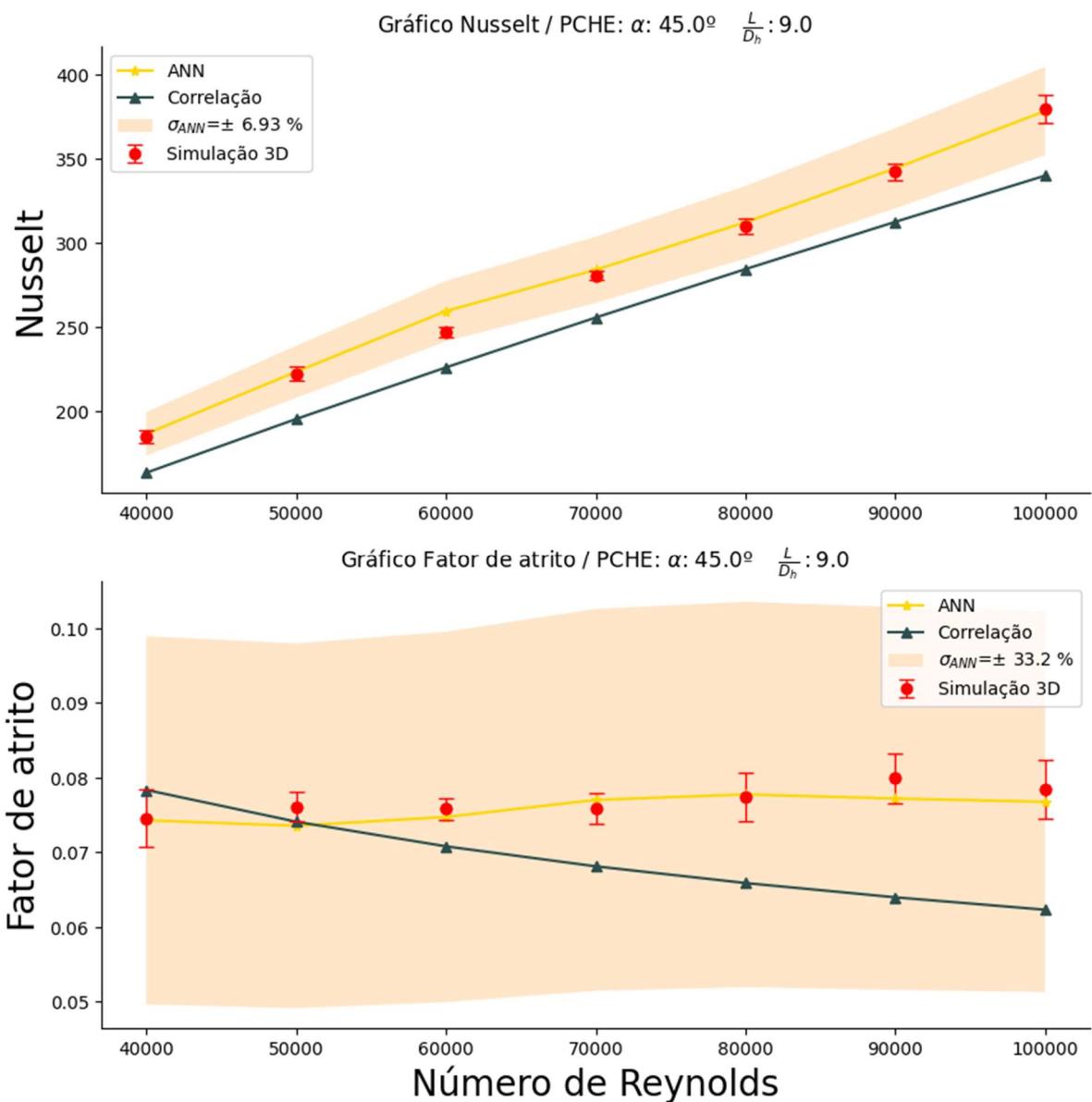


Figura 4.19- Comparaçāo do ajuste da ANN e Correlaçāo para α grande.

Fonte: Própria

4.3.5 Gráfico 3D da rede neural

A) Nusselt

A Figura 4.20 possui um propósito de uma análise qualitativa do Nusselt. Vale ressaltar que são todas a mesma figura, só que observadas de ângulos diferentes. As bolinhas pretas são os valores de referência (CFD 3D), a superfície foi construída a partir da rede neural. O valor de Reynolds é fixo.

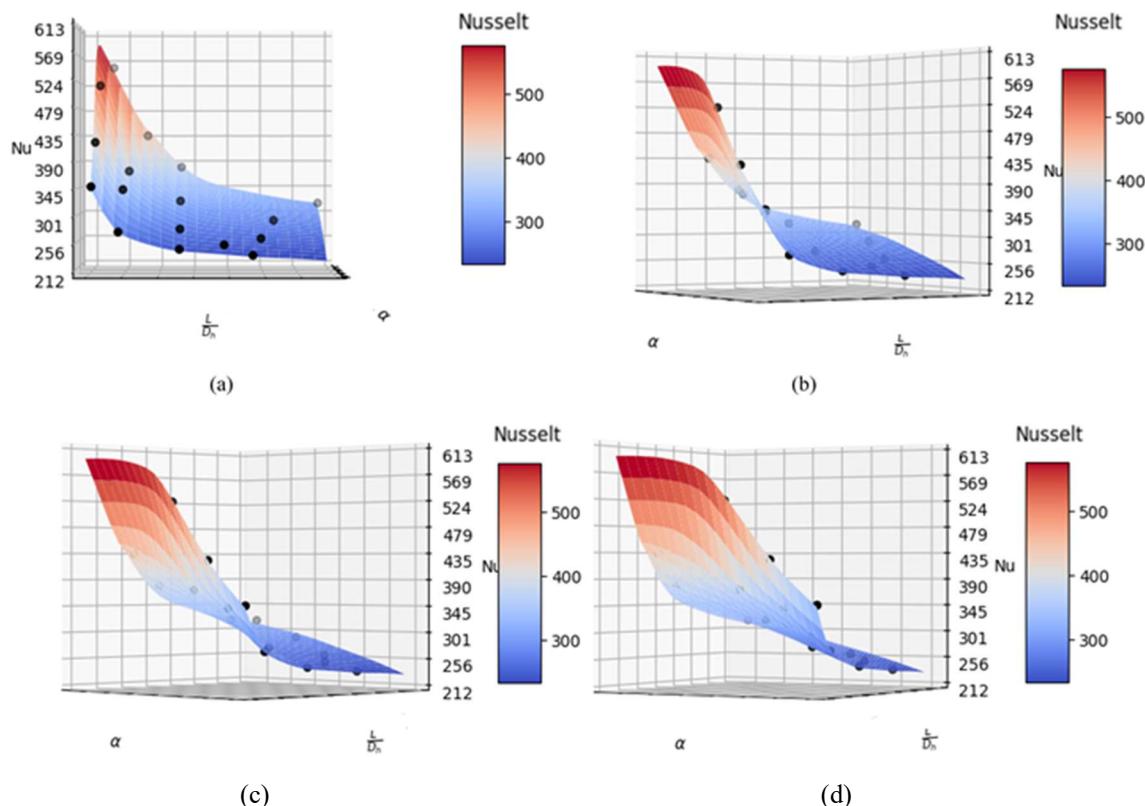


Figura 4.20- Gráfico 3D de Nusselt para Reynolds de 1000. As figuras (a), (b), (c) e (d) representam uma rotação em torno do eixo Z.

Fonte: Própria

B) Superfícies para diferentes Reynolds

Na Figura 4.21 é possível observar o comportamento do Nusselt com o aumento do número de Reynolds, à medida que o valor Reynolds cresce ocorre o deslocamento, de maneira geral, superfície do Nusselt.

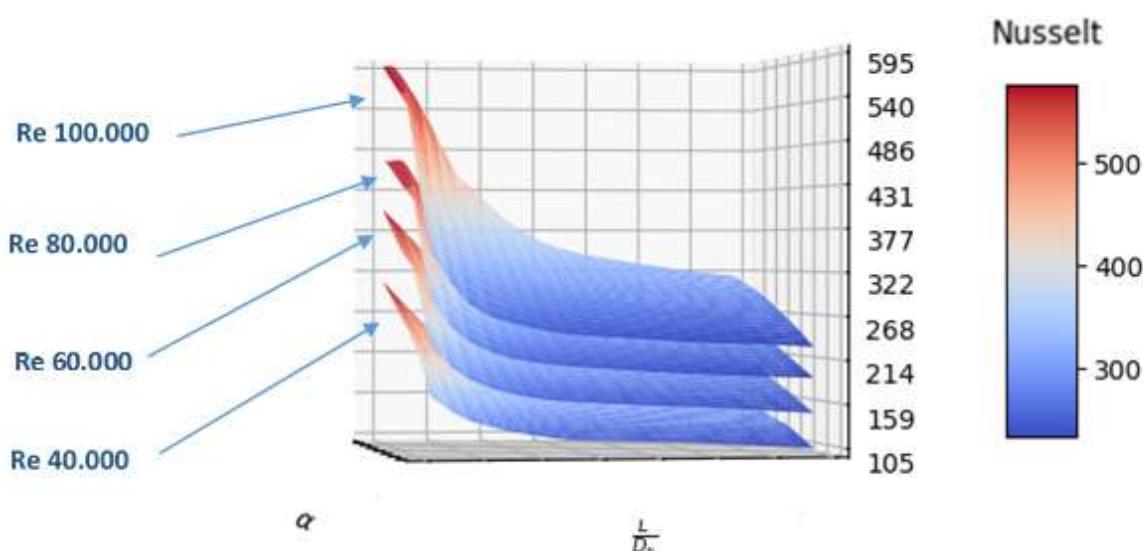


Figura 4.21- Análise de gráfico 3D para quatro Reynolds diferentes.

Fonte: Própria

C) Fator de atrito

A Figura 4.22 possuem um propósito de uma análise qualitativa do Fator de atrito. Vale ressaltar que são todas a mesma figura, só que observadas de ângulos diferentes. As bolinhas pretas são os valores de referência (CFD 3D), a superfície foi construida a partir da rede neural. O valor de Reynolds é fixo.

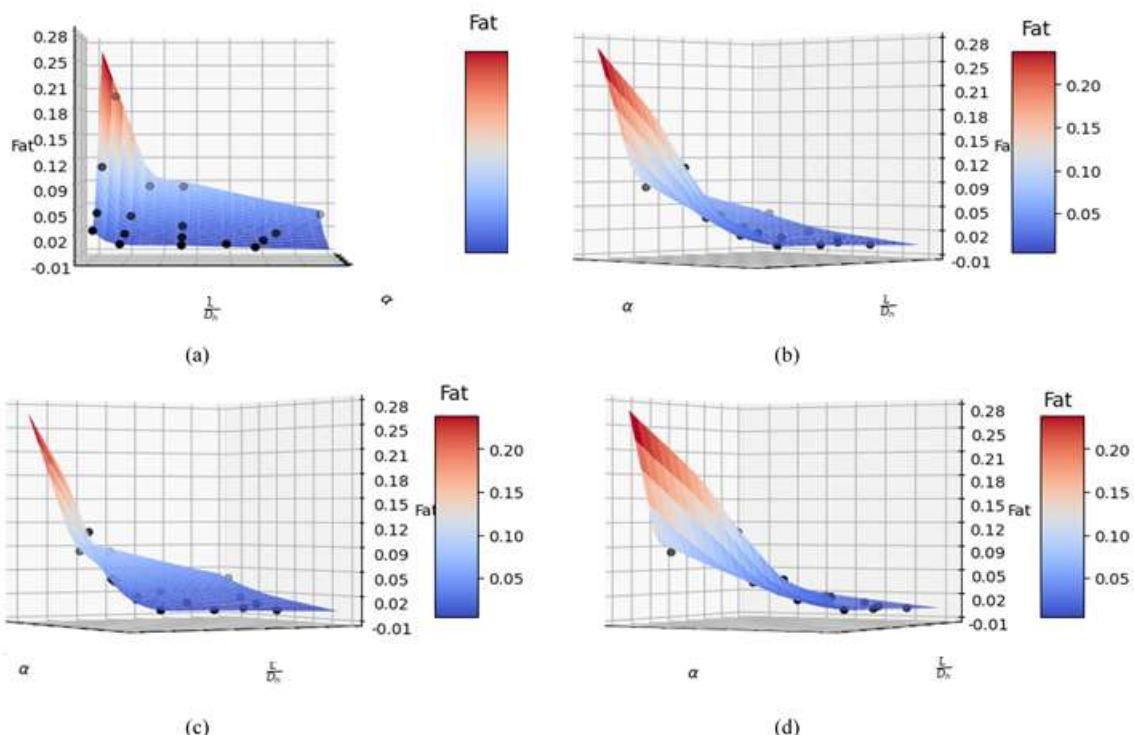


Figura 4.22- Gráfico 3D do Fator de atrito para Reynolds de 1000. As figuras (a), (b), (c) e (d) representam uma rotação em torno do eixo Z.

Fonte: Própria

4.4 Análise do treinamento da rede neural com dados da simulação CFD 3D incluindo ruído na saída

Nessa seção, a rede neural além de ser treinada pelos dados oriundos da simulação CFD 3D, também será acrescentada no treinamento os desvios padrões desses dados. Como se sabe dados experimentais possuem incertezas nas suas medições, portanto é importante analisar o comportamento da rede neural treinada à luz dessa característica.

Pode-se observar uma amostra desses dados utilizados para treinamento da rede no Anexo A.3, os desvios padrões do Nusselt e Fator de atrito estão nas colunas Std Nu e Std Fat respectivamente. A estrutura da rede neural segue as configurações da Tabela 4.51.

Tabela 4.51 – Estrutura da rede neural

Dados	Split_sample	Função_transf	Camad_Int	Neurônios	Optimizer	Batchs	Epoch	R_2	MSE
119	30%	Relu/Relu	2	500	Adam	32	200	0,94	0,0567

A rede apresenta uma redução do MSE a cada época conforme a Figura 4.23, nota-se um deslocamento entre o erro medido nos dados de treinamento e teste, sendo o do teste maior. Além disso, à medida que o número de iterações ocorre o MSE se estabiliza, parando de diminuir.

As Tabelas 4.52 e 4.53 separam, cada, 12 exemplos os quais compararam a performance entre a ANN e o método de correlação tradicional. Nesse sentido, as duas últimas colunas das tabelas representam o erro relativo entre a predição e o valor esperado; a Tabela 4.52 mostra a análise acerca do Nusselt e a Tabela 4.53, do Fator de atrito.

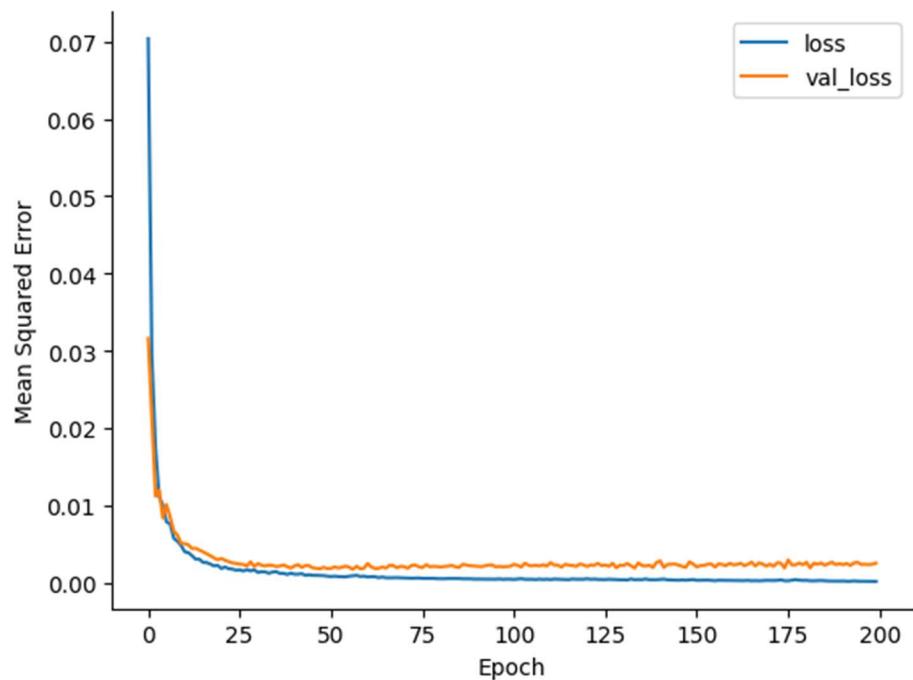


Figura 4.23 – Gráfico do MSE por época.

Fonte: Própria

Tabela 4.52 – Comparação entre os erros relativos da ANN e correlação para o Nusselt.

Re	alpha	LDh	Nu 3D	Nu ANN	Nu Corr	Erro Rel. ANN (%)	Erro Rel. Corr (%)
70000.0	22.5	2.958	328.73	328.839996	300.44	0.03	8.61
70000.0	22.5	14.792	199.24	199.089996	181.25	0.07	9.03
100000.0	45.0	19.326	313.24	312.910004	265.36	0.11	15.29
90000.0	15.0	14.148	222.61	222.309998	210.52	0.13	5.43
40000.0	15.0	9.000	117.00	116.639999	116.73	0.30	0.22
50000.0	30.0	9.000	175.97	176.880005	166.01	0.52	5.66
80000.0	45.0	19.326	264.29	262.410004	221.97	0.71	16.01
100000.0	22.5	4.931	345.72	348.220001	318.34	0.72	7.92
60000.0	17.5	12.143	167.80	166.330002	158.34	0.87	5.64
90000.0	15.0	2.830	299.96	297.309998	305.16	0.88	1.73
60000.0	45.0	19.326	210.56	208.270004	176.34	1.09	16.25
90000.0	22.5	14.792	243.02	245.789993	221.62	1.14	8.81

Fonte: Própria

Tabela 4.53 - Comparação entre os erros relativos da ANN e correlação para o Fator de atrito.

Re	alpha	LDh	Fat 3D	Fat ANN	Fat Corr	Erro Rel. ANN (%)	Erro Rel. Corr (%)
90000.0	22.5	14.791728	0.012549	0.012561	0.009925	0.09	20.91
50000.0	30.0	5.259959	0.044444	0.044572	0.055442	0.29	24.74
80000.0	45.0	6.442108	0.086969	0.088355	0.099768	1.59	14.72
90000.0	30.0	3.155976	0.107785	0.109642	0.089459	1.72	17.00
90000.0	15.0	14.147851	0.007335	0.007475	0.006607	1.91	9.92
100000.0	22.5	4.930576	0.020925	0.020516	0.026653	1.95	27.37
40000.0	15.0	2.829570	0.029507	0.030150	0.026438	2.18	10.40
70000.0	45.0	6.442108	0.084647	0.087198	0.103155	3.01	21.87
60000.0	30.0	9.000000	0.028792	0.029680	0.028662	3.08	0.45
80000.0	45.0	19.326324	0.040374	0.041635	0.027048	3.12	33.01
60000.0	45.0	19.326324	0.040356	0.042027	0.029065	4.14	27.98
100000.0	17.5	12.143189	0.009880	0.010388	0.008036	5.14	18.66

Fonte: Própria

As Figuras 4.24 e 4.25 mostram visualmente como é o ajuste da rede neural comparado com o método de correlação. Nesses 2 exemplos selecionados é importante dizer que alguns pontos em vermelhos foram utilizados para treinar a rede neural. Ademais, o ruído impactou na performance da rede neural o erro relativo máximo para o Nusselt dobrou se comparado com treinamento sem ruído e o erro relativo máximo do Fator de atrito também apresento um acréscimo. A Figura 4.24 foi construída para o caso no qual as configurações geométricas são críticas no intervalo inferior, isto é, elas estão entre os menores valores geométricos possíveis. A Figura 4.25 apresenta o ajuste num caso no qual o ângulo apresenta o valor máximo.

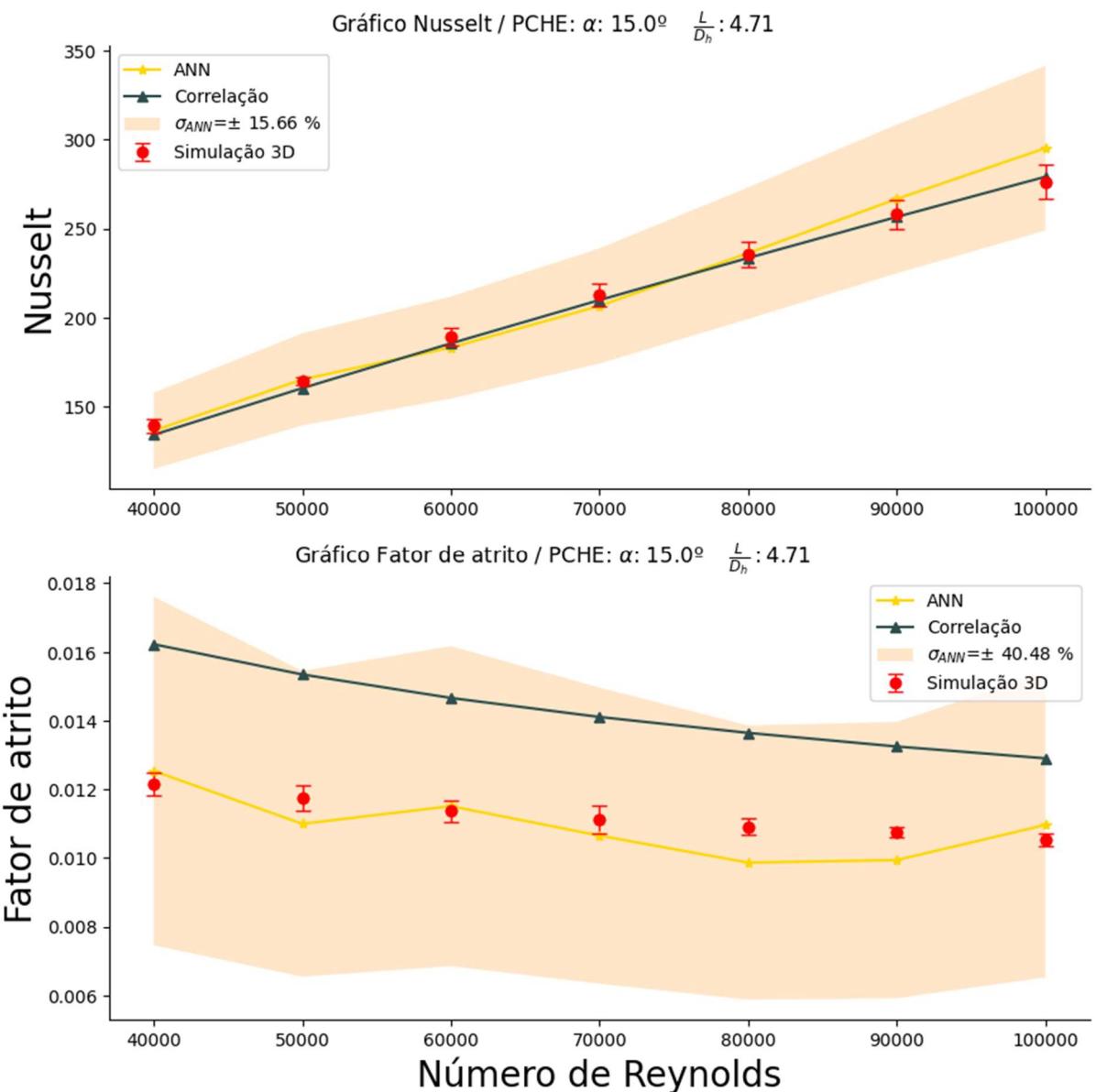


Figura 4.24 – Ajuste da rede neural para condição geométrica específica.

Fonte: Própria

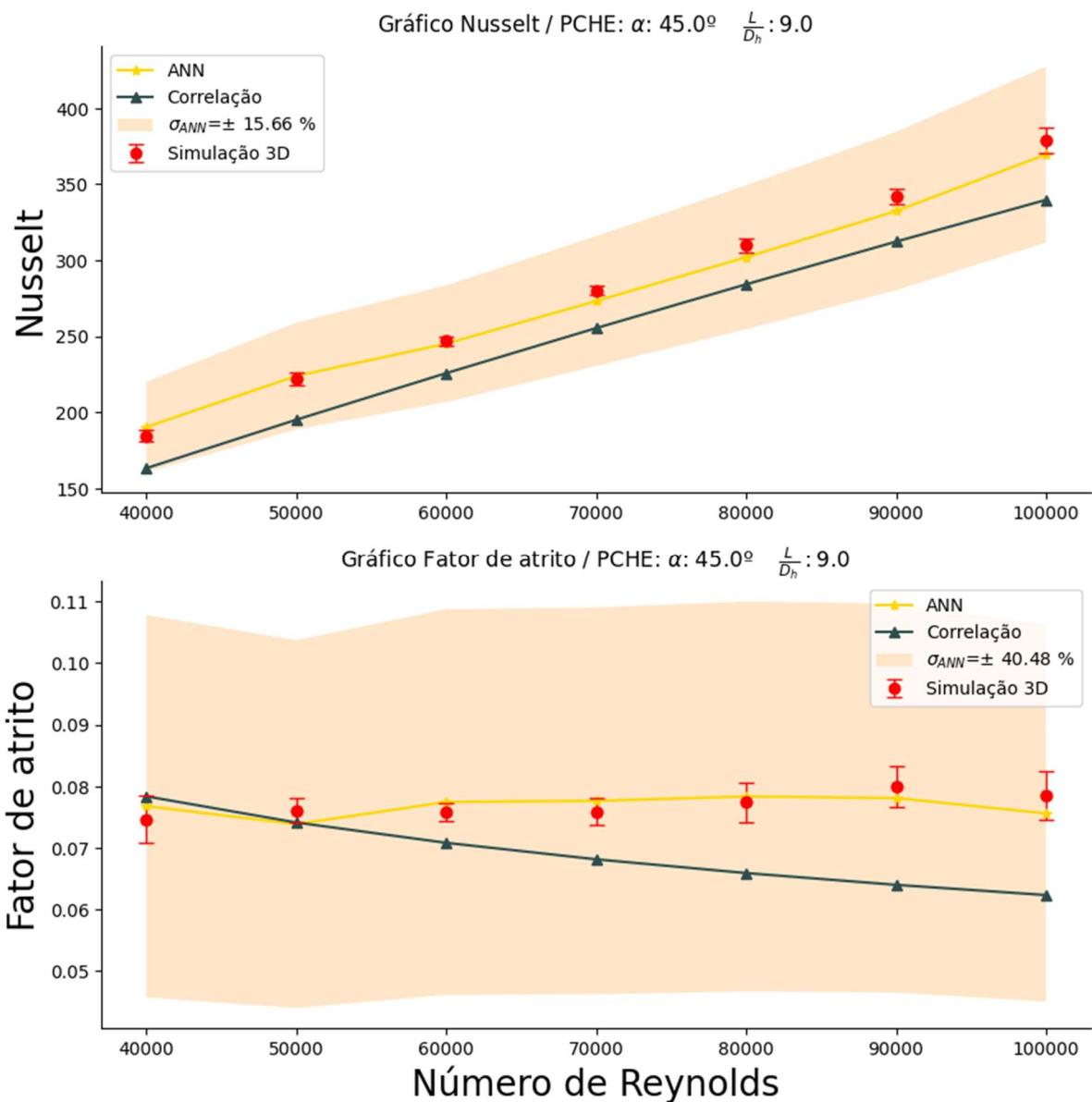


Figura 4.25 – Ajuste da rede neural para condição geométrica específica.

Fonte: Própria

5 Considerações finais

A análise das condições operacionais do trocador de calor compacto com a adoção de modelos de redes neurais permite prever a performance do equipamento, mesmo que as configurações internas sejam mantidas em segredo pelo fabricante, de modo que isso se torna de grande valia para as empresas que desejam operar os trocadore de calor que condições distintas do projeto inicial, poupando, por conseguinte, a compra de outro equipamento.

A abordagem por redes neurais exibe capacidades notáveis para generalizar a complexa dependência entre as variáveis, nesse trabalho ela apresentou um ajuste bem próximo da curva de correlação, sendo uma alternativa viável para de previsão de valores de Nusselt e Fator de atrito para trocadores de calor tipo PCHE, alcançando assim o objetivo inicial proposto.

O ponto decisivo é que a técnica fica condicionada a obtenção de dados, quanto mais informação for adicionada aos dados, como, por exemplo, características geométricas ou outras medidas operacionais, mais representativo será o resultado. Vale ressaltar que a coleta de dados pode ser feita por meio de sensores- não ficando restrito a coleta de dados via simulações- e como os sensores geram uma quantidade grande de dados acaba, consequentemente, tornando a redes neurais ainda mais viável.

Ainda, vale a ressaltar que a rede neural poderia ser apresentada por meio de uma Interface Gráfica do Usuário (GUI), disponibilizando a utilização da rede neural para aqueles que não estão familiarizado com a linguagem de programação Python.

Um futuro trabalho poderá ser feito utilizando as redes neurais recorrentes (RNN) para descrever o comportamento transiente de um trocador de calor, já que esse tipo de redes neurais consegue processar uma grande variedade de comportamentos dinâmicos.

6. Referências Bibliográficas

- [1] T. Ma, M.-J. Li, J.-L. Xu and F. Cao, "Thermodynamic analysis and performance prediction on dynamic response characteristic of PCHE in 1000 MW S-CO₂ coal fired power plant," 2019.
- [2] A. C. P. d. Oliveira, S. Jian and C. P. Naveira-Cotta, "ANÁLISE NUMÉRICA DE TROCADOR DE CALOR DO TIPO CIRCUITO IMPRESSO (PCHE) APLICADO A REATOR DE FUSÃO NUCLEAR," Rio de Janeiro, 2020.
- [3] A. MESHRAM, A. JAISWAL and S. KHIVSARA, "Modeling and analysis of a printed circuit heat exchanger for supercritical CO₂ power cycle applications," Applied Thermal, 2016.
- [4] A. T. d. Silva and C. P. N. Cotta, "DESENVOLVIMENTO DE INTERFACE GRÁFICA PARA SELEÇÃO E ESTIMATIVA DE COEFICIENTES DE CORRELAÇÕES DE NUSSELT PARA TROCADORES DE CALOR," Rio de Janeiro, 2019.
- [5] M. Venturini, "SIMULATION OF COMPRESSOR TRANSIENT BEHAVIOR THROUGH RECURRENT NEURAL NETWORK MODELS," ASME Turbo Expo 2005, Ferrara, 2005.
- [6] H. Asgari, "Modelling, Simulation and Control of Gas Turbines using Artificial Neural Networks," Christchurch, New Zealand, 2014.
- [7] S.-J. Yoon, J. O'Brien, M. Chen, P. Sabharwall and X. Sun, "Development and validation of Nusselt number and friction factor correlations for laminar flow in semi-circular zigzag channel of printed circuit heat exchanger," ELSIVIER, Idaho, 2017.
- [8] E. Reynoso-Jardón, A. Tlatelpa-Becerro, R. Rico-Martínez and C.-R. M. a. G. Urquiza, "Artificial Neural Networks (ANN) to Predict Overall Heat Transfer," Research India Publications, 2019.
- [9] F. Diez, L. Navas-Gracia, A. Martínez-Rodríguez, A. Correa-Guimaraes and L. Chico-Santamarta, "Modelling of a flat-plate solar collector using artificial neural networks for," Elsevier, Valladolid, Spain, 2019.
- [10] C. P. N. Cotta, S. Jian, H. M. da Fonseca, J. R. B. de Souza and C. P. Oliveira, "RELATÓRIO TÉCNICO FINAL DO PROJETO LABMEMS-PETROBRAS," Rio de Janeiro, 2017-2020.
- [11] T. Ma, P. Zhang, H. Shi, Y. Chen and Q. Wang, "Prediction of flow maldistribution in printed circuit heat exchanger," 2020.
- [12] M. Saeed and M.-H. Kim, "Thermal-hydraulic analysis of sinusoidal fin-based printed circuit heat exchangers for supercritical CO₂ Brayton cycle," Elsevier, Daegu, 2019.

- [13] Y. Yang, H. Li, M. Yao, Y. Zhang, C. Zhang, L. Zhang and S. Wu, "Optimizing the size of a printed circuit heat exchanger by multi-objective genetic algorithm," 2019.
- [14] OECD, "Artificial Intelligence in Society," OECD Publishing, Paris, 2019.
- [15] I. C. Collares and C. P. N. Cotta, "ANÁLISE COMPUTACIONAL DE PARÂMETROS DE OPERAÇÃO DE ANÁLISE COMPUTACIONAL DE PARÂMETROS DE OPERAÇÃO DE," Rio de Janeiro, 2019.
- [16] C. Minghui, S. Xiaodong, N. Richard, S. Isaac, U. Vivek and S. Piyush, "Pressure drop and heat transfer characteristics," Applied Thermal Engineering, 2016.
- [17] C. Minghui, S. Xiaodong, N. Richard, S. Isaac, U. Vivek and S. Piyush, "Experimental and numerical study of a printed," Annals of Nuclear Energy, 2016.
- [18] KIM, S. G., L. Y. and Y. AHN, "CFD aided approach to design printed circuit heat," Annals of Nuclear Energy, 2016.
- [19] Y. F. Schelble and C. N. Cotta, "ANÁLISE TEORICA DE TROCADORES DE CALOR TIPO CIRCUITO IMPRESSO VIA ABORDAGEM DE MEIOS POROSOS," Rio de Janeiro, 2016.
- [20] T. MA, L. LI and X. Y. XU, "Study on local thermal–hydraulic performance and optimization," Energy Conversion and Management, 2015.
- [21] L.V. Kamble et al., "Heat Transfer Studies using Artificial Neural Network - a Review," 2014.
- [22] Tlatelpa Becerro Angel et al., "Heat transfer analysis for air-water heat exchangers using artificial neural networks (ANN)," 2014.
- [23] I. KIM and X. SUN, "CFD study and PCHE design for secondary heat exchangers with FLiNaKHelium for SmAHTR," Nuclear Engineering and Design, 2014.
- [24] KIM, I. H., NO, H. C. and J. I. LEE, "Thermal-hydraulic physical models for a Printed Circuit Heat Exchanger covering He, He–CO₂ mixture, and water fluids using experimental data and CFD," Elsevier, Columbus, 2013.
- [25] Mohammad Ali Ahmadi et al., "Evolving artificial neural network and imperialist competitive algorithm for prediction oil flow rate of the reservoir," 2012.
- [26] N. S. Berbish, M. Moawed, M. Ammar and R. I. Afifi, "Heat transfer and friction factor of turbulent," Heat and Mass Transfer, 2010.
- [27] KIM, I. H., NO, H. C. and J. I. LEE, "Thermal hydraulic performance analysis of the printed," Nuclear Engineering, 2009.
- [28] T. ISHIZUKA, Y. KATO, Y. MUTO, K. NIKITIN, N. L. TRI and H. HASHIMOTO, "ThermalHydraulic Characteristics of a Printed Circuit Heat Exchanger in a Supercritical CO₂ Loop," the 11th International Topical Meeting on Nuclear Reactor Thermal-Hydraulic, 2005.

Apêndice A

Anexos

A.1 Conjuntos de dados

Conjunto 1 (252 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 50.000, 60.000, 70.000, 80.000, 90.000, 100.000
α	15° 21° 27° 33° 39° 45°
$\frac{L}{D_h}$	2,8 6,1 9,4 12,7 16 19,3

Conjunto 2 (343 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 50.000, 60.000, 70.000, 80.000, 90.000, 100.000
α	15° 20° 25° 30° 35° 40° 45°
$\frac{L}{D_h}$	2,8 5,6 8,3 11, 13,8 16,6 19,3

Conjunto 3 (847 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 50.000, 60.000, 70.000, 80.000, 90.000, 100.000
α	15° 18° 21° 24° 27° 30° 33° 36° 39° 42° 45°
$\frac{L}{D_h}$	2,8 4,4 6,1 7,7 9,4 11, 12,7 14,3 16, 17,6 19,3

Conjunto 4 (2197 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 45.000, 50.000, 55.000, 60.000, 65.000, 70.000, 75.000, 80.000, 85.000, 90.000, 95.000, 100.000
α	15° 17,5° 20° 22,5° 25° 27,5° 30° 32,5° 35° 37,5° 40° 42,5° 45°
$\frac{L}{D_h}$	2,8 4,2 5,6 6,9 8,3 9,7 11, 12,4 13,8 15,2 16,6 17,9 19,3

Conjunto 5 (5733 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 45.000, 50.000, 55.000, 60.000, 65.000, 70.000, 75.000, 80.000, 85.000, 90.000, 95.000, 100.000
α	15° 16,5° 18° 19,5° 21° 22,5° 24° 25,5° 27° 28,5° 30° 31,5° 33° 34,5° 36° 37,5° 39° 40,5° 42° 43,5° 45 °
$\frac{L}{D_h}$	2,8 3,6 4,4 5,3 6,1 6,9 7,7 8,6 9,4 10,2 11, 11,9 12,7 13,5 14,3 15,2 16, 16,8 17,6 18,5 19,3

Conjunto 6 (16337 combinações)

PARÂMETRO	VALORES AVALIADOS
Re	40.000, 43.750, 47.500, 51.250, 55.000, 58.750, 62.500, 66.250, 70.000, 73.750, 77.500, 81.250, 85.000, 88.750, 92.500, 96.250, 100.000
α	15° 16° 17° 18° 19° 20° 21° 22° 23° 24° 25° 26° 27° 28° 29° 30° 31° 32° 33° 34° 35° 36° 37° 38° 39° 40° 41° 42° 43° 44° 45°
$\frac{L}{D_h}$	2,8 3,4 3,9 4,4 5, 5,6 6,1 6,6 7,2 7,8 8,3 8,9 9,4 10, 10,5 11, 11,6 12,2 12,7 13,2 13,8 14,4 14,9 15,4 16, 16,6 17,1 17,7 18,2 18,8 19,3

A.2 Amostra de 40 dados do Conjunto 1

<i>Re</i>	α (°)	$\frac{L}{D_h}$	Nu	Fat
80.000	15	9,40	217,03	0,0335
50.000	39	16,00	171,51	0,1126
40.000	15	19,30	113,03	0,0291
100.000	27	12,70	282,61	0,0589
100.000	45	19,30	297,81	0,1040
40.000	39	19,30	136,94	0,0978
50.000	27	19,30	149,23	0,0493
50.000	39	2,80	351,86	0,9305
90.000	15	9,40	238,48	0,0325
80.000	45	19,30	249,12	0,1099
40.000	15	9,40	124,65	0,0398
90.000	39	6,10	382,16	0,2996
70.000	21	16,00	191,33	0,0370
100.000	33	2,80	547,03	0,5192
100.000	27	2,80	482,08	0,3198
90.000	33	12,70	276,23	0,0884
70.000	21	6,10	237,17	0,0821
70.000	15	9,40	195,05	0,0346
100.000	21	2,80	417,94	0,1779
80.000	27	6,10	291,55	0,1332
60.000	27	19,30	172,67	0,0471
90.000	45	2,80	623,91	1,1452
90.000	27	16,00	247,30	0,0494
90.000	39	9,40	322,93	0,1774
60.000	39	9,40	233,47	0,1963
80.000	45	16,00	262,33	0,1355
80.000	27	19,30	217,35	0,0439
60.000	45	9,40	249,14	0,2729
50.000	27	16,00	154,53	0,0572
100.000	45	12,70	336,85	0,1674
60.000	27	2,80	320,36	0,3634
60.000	15	19,30	156,34	0,0263
50.000	21	12,70	152,16	0,0471
70.000	27	6,10	262,01	0,1378
70.000	21	2,80	314,19	0,1945
40.000	27	9,40	146,44	0,0993
60.000	33	19,30	181,00	0,0649
40.000	15	12,70	118,92	0,0341
70.000	45	16,00	235,75	0,1401
80.000	45	2,80	567,81	1,1794

A.3 Rol exemplificativo dos dados da simulação CFD 3D com seus respectivos desvios padrão

<i>Re</i>	α ($^{\circ}$)	$\frac{L}{D_h}$	Nu	Std Nu	Fat	Std Fat
40.000	15	2,82957	171,4894	10,21866	0,028471	0,001186
40.000	15	4,71595	139,1123	3,583475	0,012145	0,000338
40.000	15	9	117,385	2,338968	0,009492	0,000339
40.000	15	14,14785	114,2563	1,805343	0,008245	0,000223
40.000	17,5	12,14319	119,7099	1,042672	0,010071	0,000113
40.000	22,5	2,958346	212,7974	7,996251	0,061967	0,014134
40.000	22,5	4,930576	169,7229	3,680187	0,02595	0,00313
40.000	22,5	9	131,4188	3,098685	0,018038	0,000755
40.000	22,5	14,79173	130,9323	3,34683	0,013472	0,000542
40.000	30	3,155976	239,7279	11,87185	0,091726	0,011246
40.000	30	5,259959	186,6531	8,74247	0,04991	0,003859
40.000	30	9	154,7217	2,637292	0,030208	0,000643
40.000	30	15,77988	139,3405	1,065796	0,021621	0,00027
40.000	45	3,865265	271,2087	17,98739	0,227346	0,0103
40.000	45	6,442108	217,4383	9,260136	0,091301	0,001919
40.000	45	9	184,5891	3,742902	0,074574	0,003863
40.000	45	19,32632	153,7106	1,456163	0,041066	0,000658
50.000	15	2,82957	203,3541	11,35625	0,028921	0,001096
50.000	15	4,71595	164,3806	2,376612	0,011744	0,000365
50.000	15	9	141,9105	2,408053	0,009235	0,000212
50.000	15	14,14785	135,9783	1,975237	0,007663	0,000494
50.000	17,5	12,14319	144,695	1,318724	0,010146	7,14E-05
50.000	22,5	2,958346	252,4345	7,777234	0,055929	0,010192
50.000	22,5	4,930576	198,6567	3,869385	0,023842	0,001519
50.000	22,5	9	157,6408	1,719389	0,016669	0,000413
50.000	22,5	14,79173	151,3835	2,454237	0,012905	0,000252
50.000	30	3,155976	281,4289	7,735444	0,084778	0,015939
50.000	30	5,259959	226,4746	1,412368	0,044921	0,002728
50.000	30	9	176,6845	1,604857	0,02848	0,000361
50.000	30	15,77988	168,3149	1,312424	0,02069	0,000132

A.4 Resumo de correlações da literatura

Autor	Correlações	Fluido	Ângulo
Berbish <i>et al.</i> (2010)	$Nu = 0,0228Re^{0,8}$; $f = 0,487Re^{-0,26}$; $8.242 \leq Re \leq 5.7794$.	Ar	0°
Chen <i>et al.</i> (2016)	$Nu = (0,01352 \pm 0,0094)Re^{0,080058 \pm 0,0921}$; $1.200 \leq Re \leq 1.850$; $Nu = (3,6361 \times 10^{-4} \pm 0,7855 \times 10^{-4})Re^{1,2804 \pm 0,0273}$; $1.850 < Re \leq 2.900$.	He-He	15°
Chen <i>et al.</i> (2016)	$Nu = (0,047516 \pm 0,015662)Re^{0,633151 \pm 0,044606}$; $1.200 \leq Re \leq 1.850$; $Nu = (3,680123 \times 10^{-4} \pm 1,184389 \times 10^{-4})Re^{1,282182 \pm 0,042068}$; $1.850 < Re \leq 2.900$.	He-He	15°
Chen <i>et al.</i> (2016)b	$Nu = (0,05516 \pm 0,0016)Re^{0,69195 \pm 0,00559}$; $f = 17,639Re^{-0,8861 \pm 0,0017}$; $1.400 < Re \leq 2.200$ $Nu = (0,09221 \pm 0,01397)Re^{0,62507 \pm 0,01949}$; $f = 17,639Re^{-0,019044 \pm 0,001692}$; $2.200 < Re \leq 3.558$.	He-He	15°
Kim <i>et al.</i> (2009)	$Nu = 3,255 + 0,00729(Re - 350)$; $350 < Re < 800$; $Pr = 0,66$; $f \cdot Re = 16,51 + 0,01627Re$; $350 < Re < 1.200$.	He-He	15°
Kim <i>et al.</i> (2013)	$Nu = 4,089 + 0,00365RePr^{0,58}$; $f \cdot Re = 15,78 + 0,0487Re^{0,84}$; $0 < Re < 2.500$; $0,66 < Pr < 13,41$; $Nu = 4,089 + 0,00497Re^{0,95}Pr^{0,55}$; $f \cdot Re = 15,78 + 0,0557Re^{0,82}$; $0 < Re < 3000$; $0,66 < Pr < 13,41$.	He-H ₂ O He-CO ₂	15°

Ishizuka et al. (2005)	$h_h = 0,2104Re + 44,16 ;$ $f_h = -2 \times 10^{-6}Re + 0,0467 ;$ $2.400 \leq Re \leq 6.000 ;$	CO ₂ -CO ₂ 32,5° (h), 40° (c)
	$U = 0,1106Re + 15,943 ;$ $2.400 \leq Re \leq 6.000 ;$	
	$f_c = -2 \times 10^{-6}Re + 0,1023 ;$ $5.000 \leq Re \leq 13.000 .$	
Kays & London	$Nu = jRePr^{1/3} ;$ $j = 0,24Re^{-0,425} ;$ $f = 1,08Re^{-0,425} ;$ $400 < Re < 3.000 .$	0°
Meshram et al. (2016)	$Nu_h = 0,0493Re^{0,77}Pr^{0,55} ;$ $f_h = 0,8386Re^{-0,5985} + 0,00295 ;$ $500K < T_b < 630K ; 5.000 \leq Re \leq 26.000 ;$ $Nu_h = 0,0514Re^{0,76}Pr^{0,55} ;$ $f_h = 0,8385Re^{-0,5978} + 0,00331 ;$ $600K < T_b < 730K ; 5.000 \leq Re \leq 26.000 ;$ $Nu_h = 0,0685Re^{0,705}(D/2)^{-0,122} ;$ $f_h = 0,0648Re^{-0,254}(D/2)^{-0,0411} ;$ $500K < T_b < 630K ; 1,2mm \leq D \leq 2,6mm ; 8.500 \leq Re \leq 20.500 ;$ $Nu_c = 0,0718Re^{0,71}Pr^{0,55} ;$ $f_c = 0,8657Re^{-0,5755} + 0,00405 ;$ $400K < T_b < 500K ; 5.000 \leq Re \leq 26.000 ;$ $Nu_c = 0,0661Re^{0,743}Pr^{0,55} ;$ $f_c = 0,8796Re^{-0,5705} + 0,00353 ;$ $500K < T_b < 600K ; 5.000 \leq Re \leq 26.000 ;$ $Nu_c = 0,0117Re^{0,843}(D/2)^{-0,0405} ;$ $f_c = 0,0759Re^{-0,241}(D/2)^{-0,089} ;$ $400K < T_b < 500K ; 1,2mm \leq D \leq 2,6mm ; 8.500 \leq Re \leq 20.500 .$	CO ₂ -CO ₂ 0°

Ishizuka et al. (2005)	$h_h = 0,2104Re + 44,16 ;$ $f_h = -2 \times 10^{-6}Re + 0,0467 ;$ $2.400 \leq Re \leq 6.000 ;$	CO ₂ -CO ₂ 32,5° (h), 40° (c)
	$U = 0,1106Re + 15,943 ;$ $2.400 \leq Re \leq 6.000 ;$	
	$f_c = -2 \times 10^{-6}Re + 0,1023 ;$ $5.000 \leq Re \leq 13.000 .$	
Kays & London	$Nu = jRePr^{1/3} ;$ $j = 0,24Re^{-0,425} ;$ $f = 1,08Re^{-0,425} ;$ $400 < Re < 3.000 .$	0°
Meshram et al. (2016)	$Nu_h = 0,0493Re^{0,77}Pr^{0,55} ;$ $f_h = 0,8386Re^{-0,5985} + 0,00295 ;$ $500K < T_b < 630K ; 5.000 \leq Re \leq 26.000 ;$	CO ₂ -CO ₂ 0°
	$Nu_h = 0,0514Re^{0,76}Pr^{0,55} ;$ $f_h = 0,8385Re^{-0,5978} + 0,00331 ;$ $600K < T_b < 730K ; 5.000 \leq Re \leq 26.000 ;$	
	$Nu_h = 0,0685Re^{0,705}(D/2)^{-0,122} ;$ $f_h = 0,0648Re^{-0,254}(D/2)^{-0,0411} ;$ $500K < T_b < 630K ; 1,2mm \leq D \leq 2,6mm ; 8.500 \leq Re \leq 20.500 ;$	
	$Nu_c = 0,0718Re^{0,71}Pr^{0,55} ;$ $f_c = 0,8657Re^{-0,5755} + 0,00405 ;$ $400K < T_b < 500K ; 5.000 \leq Re \leq 26.000 ;$	
	$Nu_c = 0,0661Re^{0,743}Pr^{0,55} ;$ $f_c = 0,8796Re^{-0,5705} + 0,00353 ;$ $500K < T_b < 600K ; 5.000 \leq Re \leq 26.000 ;$	
	$Nu_c = 0,0117Re^{0,843}(D/2)^{-0,0405} ;$ $f_c = 0,0759Re^{-0,241}(D/2)^{-0,089} ;$ $400K < T_b < 500K ; 1,2mm \leq D \leq 2,6mm ; 8.500 \leq Re \leq 20.500 .$	

Meshram <i>et al.</i> (2016)	$Nu_h = 0,0174 Re^{0,893} Pr^{0,7};$	CO ₂ -CO ₂	36°		
	$f_h = 0,867 Re^{-0,522} + 0,040;$				
$470K < T_b < 630K ; \quad 5.000 \leq Re \leq 32.000 ;$					
$Nu_h = 0,0205 Re^{0,869} Pr^{0,7};$					
$f_h = 0,819 Re^{-0,671} + 0,044;$					
$580K < T_b < 730K ; \quad 5000 \leq Re \leq 32000 ;$					
$Nu_h = 87,56 \left(\frac{P}{12}\right)^{-0,178} \left(\frac{\theta}{116}\right)^{-0,9306};$					
$f_h = 0,0375 \left(\frac{P}{12}\right)^{-0,9639} \left(\frac{\theta}{116}\right)^{-2,409};$					
$470K < T_b < 630K ; Re = 14.500 ; 100^\circ \leq \theta \leq 120^\circ ; 8mm \leq P \leq 12mm.$					
$Nu_c = 0,0177 Re^{0,871} Pr^{0,7};$					
$f_c = 0,869 Re^{-0,512} + 0,041;$					
$400K < T_b < 520K ; 5.000 \leq Re \leq 32.000 ;$					
$Nu_c = 0,0213 Re^{0,876} Pr^{0,7};$					
$f_c = 0,804 Re^{-0,711} + 0,045;$					
$500K < T_b < 640K ; \quad 5.000 \leq Re \leq 32.000 ;$					
$Nu_c = 85,95 \left(\frac{P}{12}\right)^{-0,171} \left(\frac{\theta}{116}\right)^{-0,8912};$					
$f_c = 0,0395 \left(\frac{P}{12}\right)^{-0,9479} \left(\frac{\theta}{116}\right)^{-2,239};$					
$400K < T_b < 520K ; Re = 14.500 ; 100^\circ \leq \theta \leq 120^\circ ; 8mm \leq P \leq 12mm .$					
Ngo <i>et</i> <i>al.</i> (2007)	$Nu = (0,1696 \pm 0,0144) Re^{0,629 \pm 0,009} Pr^{0,317 \pm 0,014};$	CO ₂ -CO ₂	52°		
	$f = (0,1924 \pm 0,0299) Re^{-0,091 \pm 0,016};$				
$3.500 < Re < 22.000 ; 0,75 < Pr < 2,2 .$					
$f_h = (0,3390 \pm 0,0285) Re^{-0,158 \pm 0,009};$					
$3.500 < Re < 20.000 ;$					
$f_c = (0,3749 \pm 0,1293) Re^{-0,154 \pm 0,036};$					
$6.000 < Re < 22.000$					

Nikitin <i>et al.</i> (2006)	$h = 2,52Re^{0,681};$ $f = (-1,402 \times 10^{-6} \pm 0,087 \times 10^{-6})Re + (0,04495 \pm 0,00038);$ $2.800 \leq Re \leq 5.800;$	CO ₂ -CO ₂	32,5° (h), 40° (c)
Oyakawa (1989)	$h = 5,49Re^{0,625};$ $f = (-1,545 \times 10^{-6} \pm 0,099 \times 10^{-6})Re + (0,09318 \pm 0,00090);$ $6.200 \leq Re \leq 12.100.$	-	0°
Seo <i>et al.</i> (2015)	$j = 0,4Re^{-0,4}\left(\frac{2D}{P}\right)^{0,75};$ $f = 5j;$ $600 < Re < 3000;$ $j = 0,4Re^{-0,36}\left(\frac{2D}{P}\right)^{0,75};$ $f = 4,8Re^{-0,36}\left(\frac{2D}{P}\right)^{1,5};$ $10^4 < Re < 10^5.$	H ₂ O-H ₂ O	0°
Kim <i>et al.</i> (2016)	$h_c = 0,1706N_{clc}^{0,44}Re_c^{0,324}Pr^{1/3}(\mu_c/\mu_w)^{0,14}(k/D_h)_c;$ $100 < Re_c < 550;$ $h_h = 0,1729N_{chl}^{0,44}Re_h^{0,324}Pr^{1/3}(\mu_h/\mu_w)^{0,14}(k/D_h)_h;$ $Nu_h = 0,7203Re_h^{0,1775}Pr^{1/3}(\mu_h/\mu_w)^{0,14};$ $100 < Re_h < 850;$ $f_{Ncl} = 1,3383Re^{-0,5003};$ $100 < Re_h < 850.$	CO ₂ -CO ₂	32,5° (h), 40° (c)
	$Nu = (0,0292 \pm 0,0015)Re^{0,8138 \pm 0,0050};$ $f = (0,2515 \pm 0,0097)Re^{-0,2031 \pm 0,0041};$ $2.000 < Re < 58.000; 0,7 < Pr < 1,0.$		
	$Nu = (0,0188 \pm 0,0032)Re^{0,8742 \pm 0,0162};$ $f = (0,2881 \pm 0,0032)Re^{-0,1322 \pm 0,0041};$ $2.000 < Re < 55.000; 0,7 < Pr < 1,0.$		

Yoon <i>et al.</i> (2017)	$Nu = 5,05 + (0,02\alpha + 0,003)RePr^{0,6};$ $100 \leq Re \leq 550; 5^\circ \leq \alpha \leq 15^\circ;$ $Nu_h = (0,71\alpha + 0,289) \left(\frac{l_r}{D_h}\right)^{-0,087} Re^{(-0,11(\alpha-0,55)^2 - 0,004(l_r/D_h)\alpha + 0,54)} Pr^{-0,56};$ $Nu_c = (0,18\alpha + 0,457) \left(\frac{l_r}{D_h}\right)^{-0,038} Re^{(-0,23(\alpha-0,74)^2 - 0,004(l_r/D_h)\alpha + 0,56)} Pr^{-0,58};$ $550 \leq Re \leq 2.000; 15^\circ \leq \alpha \leq 45^\circ.$	He-He	-

Fonte: [10]