

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Erick Tomaz Oliveira

**Aplicação de redes neurais artificiais na
estimativa de propriedades mecânicas de
materiais a partir de ensaios de indentação**

Uberlândia, Brasil

2024

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Erick Tomaz Oliveira

Aplicação de redes neurais artificiais na estimativa de propriedades mecânicas de materiais a partir de ensaios de indentação

Trabalho de conclusão de curso apresentado à Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Engenharia Mecatrônica.

Orientador: Prof. Dr. Pedro Augusto Queiroz de Assis

Universidade Federal de Uberlândia – UFU

Faculdade de Engenharia Mecânica

Bacharelado em Engenharia Mecatrônica

Uberlândia, Brasil

2024

Erick Tomaz Oliveira

Aplicação de redes neurais artificiais na estimativa de propriedades mecânicas de materiais a partir de ensaios de indentação

Trabalho de conclusão de curso apresentado à Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Engenharia Mecatrônica.

Trabalho aprovado. Uberlândia, Brasil, 01 de fevereiro de 2024:

Prof. Dr. Pedro Augusto Queiroz de Assis
Orientador

Prof. Dr. Leonardo Rosa Ribeiro da Silva

Dr. Luiz Fernando Maia de Almeida

Uberlândia, Brasil
2024

Ao meu pai, que não poderá ver minhas conquistas

Agradecimentos

Ao meu orientador, que abriu diversas portas para mim e sem elas não seria capaz nem mesmo de formar. À minha irmã que vem me aturando e ajudando muito nos últimos anos. E por fim, à Isabela que esteve ao meu lado durante todo meu trajeto acadêmico.

Resumo

O ensaio de indentação instrumentada é um método não destrutivo utilizado para avaliar as propriedades mecânicas de materiais. Esse ensaio consiste em aplicar uma força controlada na superfície do material por meio de uma ponta com alta dureza e formato geométrico específico. De posse dos dados de força e de profundidade de penetração do indentador na amostra, a determinação das propriedades mecânicas do material em análise pode ser realizada de diferentes maneiras. Neste trabalho, foi implementado um método de Inteligência Artificial (IA) para a identificação dos parâmetros da equação de Ludwik (expoente de encruamento n , coeficiente de resistência ao escoamento K , e limite de escoamento σ_y) que descreve a fase plástica da curva tensão-deformação de materiais. Com esse propósito, primeiramente, simulações foram realizadas para gerar um banco de dados com 10 360 combinações diferentes de n , K e σ_y . Então, esse banco de dados foi dividido em três conjuntos distintos: Treinamento, validação e teste. O primeiro foi empregado para treinar três Redes Neurais Artificiais (RNAs) utilizadas para estimar K , n e σ_y a partir dos valores de força e profundidade da indentação simulada. Já o conjunto de treinamento foi usado para otimizar a arquitetura dessas redes. Por fim, o conjunto de teste avaliou a capacidade de generalização das RNAs desenvolvidas. Os resultados mostraram que o método de IA foi capaz de identificar o coeficiente de resistência e o limite de escoamento com boa precisão. No entanto, foi observado que o método não foi capaz de determinar com bons resultados o expoente de encruamento.

Palavras-chave: Indentação, Redes Neurais Artificiais, Elementos Finitos.

Abstract

Indentation testing is a non-destructive method used to evaluate the mechanical properties of materials. This test consists of applying a controlled force to the surface of the material through a tip with high hardness and a specific geometric shape. With the depth and force data, the determination of the mechanical properties of the material under analysis can be carried out in different ways. In this work, an Artificial Intelligence (AI) method was implemented to identify the parameters of the Ludwik equation that describes the plastic phase of the stress-strain curve of materials. For this purpose, first, simulations were performed to generate a database with 10 360 different combinations of hardening exponent n , strength coefficient K , and yield strength σ_y . Then, this database was divided into three distinct sets: Training, validation, and testing. The first was used to train three Artificial Neural Networks (ANNs) used to estimate K , n , and σ_y from the data of the simulated indentation. The validation set was used to optimize the architecture of these networks. Finally, the test set evaluated the generalization ability of the developed ANNs. The results showed that the AI method was able to identify the coefficient of resistance and yield strength parameters of the Ludwik equation with good accuracy. However, it was observed that the method was not able to determine the hardening exponent with good results.

Palavras-chave: *Instrumented Indentation Test, Artificial Neural Networks, Finite Element Method.*

Lista de figuras

| | | |
|-----------|--|----|
| Figura 1 | – Imagem do Abaqus com a simulação da indentação desenvolvida. Pode-se ver o semicírculo representando o indentador, o retângulo da amostra (com quadrado interno de malha refinada e a parte externa com malha mais grosseira). | 16 |
| Figura 2 | – Imagem da janela de configuração das saídas da simulação no Abaqus. Pode-se notar os campos <i>Forces/Reactions</i> com o parâmetro <i>Name</i> indicando ser a “base” (nome dado ao corpo de prova) e <i>Displacement/Velocity/Acceleration</i> com o parâmetro <i>Name</i> indicando ser o “indentador” marcados, correspondendo respectivamente as Forças/Reações do corpo de prova e o Deslocamento/Velocidade/Aceleração do indentador. | 17 |
| Figura 3 | – Diagrama de um Perceptron. | 19 |
| Figura 4 | – Representação da função de ativação ReLU. | 20 |
| Figura 5 | – Diagrama de uma Rede Neural Artificial. | 20 |
| Figura 6 | – Curva de treinamento quando há sobreajuste, isto é, uma piora em dados inéditos para a rede, enquanto o desempenho de treinamento melhora. Isso indica que a rede se “superespecializou” aos dados de treinamento e pode não generalizar o aprendido. | 22 |
| Figura 7 | – À direita, o gráfico de força por deslocamento com os pontos utilizados como entrada das RNAs e à esquerda, a linearização do carregamento desse gráfico destacando o coeficiente angular A , o coeficiente linear B e a área da reta $A_f \times h$ | 25 |
| Figura 8 | – Evolução do MSE para os conjuntos de dados de treinamento e validação da RNA- n | 26 |
| Figura 9 | – MSE nos conjuntos de treinamento e validação a partir da 15 ^a época, relativos à nova arquitetura para RNA- n | 26 |
| Figura 10 | – MSE nos conjuntos de dados de treinamento e validação para RNA- σ_y . Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 5 ^a época. | 27 |
| Figura 11 | – MSE nos conjuntos de dados de treinamento e validação para RNA- K . Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 5 ^a época. | 28 |
| Figura 12 | – MSE nos conjuntos de dados de treinamento e validação para RNA- K modificado. Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 8 ^a época. | 29 |

Figura 13 – Comparação entre os valores preditos e reais das propriedades mecânicas dos materiais para o conjunto de dados de teste. 30

Lista de tabelas

| | | |
|----------|---|----|
| Tabela 1 | – Faixas de valores dos parâmetros para construção do banco de dados. . | 18 |
| Tabela 2 | – Hiperparâmetros das RNAs. | 24 |
| Tabela 3 | – Resultados para 10 amostras no conjunto de dados de teste da RNA- n . | 27 |
| Tabela 4 | – Resultados para 10 amostras do conjunto de dados de teste da RNA- σ_y . | 28 |
| Tabela 5 | – Resultados para 10 amostras no conjunto de dados de teste para rede RNA- K | 29 |
| Tabela 6 | – Resultados para 10 amostras no conjunto de dados de teste para rede RNA- K modificada. | 29 |

Lista de abreviaturas e siglas

| | |
|------------|---------------------------------|
| <i>GUI</i> | <i>Graphical User Interface</i> |
| <i>GPU</i> | <i>Graphics Processing Unit</i> |
| IA | Inteligência Artificial |
| <i>MSE</i> | <i>Mean Squared Error</i> |
| ReLU | <i>Rectifier Linear Unit</i> |
| RNA | Redes Neurais Artificial |
| <i>TPU</i> | <i>Tensor Processing Unit</i> |

Lista de símbolos

| | |
|------------------|---|
| α | Taxa de aprendizado |
| b | Viés |
| ϵ | Deformação |
| J | Índice de desempenho |
| K | Coefficiente de resistência |
| n | Expoente de encruamento |
| N_b | Tamanho de lote |
| ν | Coefficiente de Poisson |
| σ | Tensão |
| σ_y | Limite de escoamento |
| σ_i | Variância dos valores do parâmetro de entrada i |
| u | Potencial de ativação |
| w_i | Pesos do perceptron/RNA |
| x_i | Entradas do perceptron/RNA |
| \overline{X}_i | Média dos valores do parâmetro de entrada i |
| y | Saída do perceptron |
| z_i | Entrada normalizada |

Sumário

| | | |
|-----|--|----|
| 1 | INTRODUÇÃO | 13 |
| 2 | SIMULAÇÃO DE ELEMENTOS FINITOS DO ENSAIO DE INDENTAÇÃO INSTRUMENTADA | 16 |
| 3 | REDES NEURAIS ARTIFICIAIS | 19 |
| 4 | RESULTADOS | 24 |
| 4.1 | Resultados na estimativa do expoente de encruamento | 25 |
| 4.2 | Resultados na estimativa do limite de escoamento | 27 |
| 4.3 | Resultados na estimativa do coeficiente de resistência | 27 |
| 4.4 | Resultados do conjunto de teste | 30 |
| 5 | CONCLUSÕES E SUGESTÕES DE TRABALHOS FUTUROS . . . | 31 |
| | REFERÊNCIAS | 33 |

1 Introdução

O conhecimento a respeito das propriedades dos materiais é fundamental para a engenharia. Desde a confecção de uma chave de fenda ao desenvolvimento de uma turbina, é necessário ter conhecimento das características de metais, ligas, compósitos, entre outros materiais, para a aplicação em cada parte do projeto. Por exemplo, no caso da chave, ter um alto limite de escoamento impede a deformação. Já no caso da turbina, resistências elevadas à fadiga e à corrosão são cruciais (ASHBY; JONES, 2012). Então, testes foram propostos na literatura para caracterização de tais propriedades. Em particular, o ensaio mais comumente empregado é de tração, pois resulta em uma curva de tensão por deformação que pode ser utilizada para determinar as características mecânicas do material. Entretanto, o ensaio de tração é destrutivo (isto é, resulta na perda da amostra testada) e requer maquinário robusto para execução, tornando inviável a realização do ensaio *in situ*. Como alternativa, é possível adotar o ensaio de indentação instrumentada para caracterização de materiais.

O ensaio de indentação instrumentada consiste na aplicação de uma força controlada na superfície do material em análise por meio de uma ponta com alta dureza e formato geométrico específico, podendo variar entre esférico, cônico, prismático, entre outros, de acordo com o tipo de ensaio (NICOLOSI, 2015). Ao longo desse processo, a profundidade de penetração da ponta no material e a força aplicada são medidas (tipicamente utilizando LVDT - *linear variable differential transformer* - e célula de carga) caracterizando o ensaio como uma indentação instrumentada. A partir dos resultados desse ensaio, é possível determinar diferentes propriedades de materiais, como a dureza. Historicamente, foi visando estimar essa propriedade que Calvert e Johnson desenvolveram as primeiras técnicas de indentação (WALLEY, 2012). Hodiernamente existem diversos estudos que utilizam da indentação para a determinação de outras propriedades como limites de escoamento e de resistência de materiais metálicos (MOHAN et al., 2021) e tenacidade à fratura (YU et al., 2023).

Comparada ao ensaio de tração, a principal vantagem da indentação é o fato de ser possível a realização não destrutiva do experimento ou utilizando um pequeno corpo de prova. Além disso, é possível realizar a indentação *in situ* com equipamentos portáteis sem a necessidade de interrupção da operação ou amostragem do material. Em contrapartida, como desvantagem da indentação, pode-se citar a dificuldade em analisar os dados resultantes do ensaio.

É importante salientar que existem diferentes métodos na literatura para realização de tal análise. Por exemplo, o leitor pode consultar Haggag (1993) e Li et al. (2016) para

detalhes sobre a caracterização de propriedades mecânicas a partir do ensaio de indentação instrumentada esférico (isto é, aquele no qual o penetrador é uma esfera) de múltiplos carregamentos. Na prática, a aplicação desses métodos nem sempre apresenta resultados satisfatórios. Isso se deve a diversos fenômenos e características do ensaio:

- Como a profundidade de indentação é reduzida (na ordem de micrometros no caso da macroindentação), propriedades superficiais advindas do processo de fabricação interferem na determinação das características do material como um todo. Alguns exemplos de processos que podem afetar o ensaio são: tratamentos térmicos superficiais (MAUVOISIN et al., 2000); encruamento causado por processos de laminação e forjamento (LIU; ZHAO; WANG, 2014); alteração da microestrutura causada por aquecimento por atrito ao utilizar máquina de corte (TAN; SHEN, 2005);
- Efeito do profundidade de indentação: quanto menor a profundidade, maior a estimativa de dureza do material (PHARR; HERBERT; GAO, 2010);
- Durante a indentação, deformações plásticas e elásticas ocorrem simultaneamente, já que o material abaixo do indentador sofre deformação plástica, além de encruamento, ao mesmo passo em que a região adjacente à deformação plástica ainda se encontra no desenvolvimento da fase elástica (TABOR, 1951). Portanto, a caracterização das propriedades das fases, sejam elas plásticas ou elásticas, apresenta uma influência mútua.

Como alternativa aos métodos convencionais de análise e visando melhor caracterização de propriedades mecânicas de materiais a partir da indentação, foi proposta na literatura a utilização de redes neurais artificiais treinadas a partir de dados gerados em simulações de elementos finitos (PARK et al., 2022; JEONG et al., 2020). A ideia fundamental consiste em simular a indentação para diferentes combinações de propriedades mecânicas contruindo um banco de dados. Então, esse banco de dados é utilizado para treinar Redes Neurais Artificiais (RNAs) que são empregadas para estimar propriedades usando os resultados do ensaio.

A partir dessa metodologia e seguindo o que foi realizado em Kim et al. (2022), o presente trabalho tem como objetivo o desenvolvimento de RNAs para estimativa de propriedades mecânicas de materiais a partir dos dados de força e profundidade provenientes de simulações do ensaio de indentação instrumentada esférico com apenas um carregamento. Para isso, utilizando elementos finitos, foram realizadas simulações de 10.360 indentações com perfis de tensão \times deformação distintos para construção de um banco de dados. Em particular tais perfis foram gerados a partir da lei de plástica de Ludwik (MUKARATI; MOSTERT; SIYASIYA, 2019). Então, esse banco de dados foi dividido

em três partes: Uma para treinamento, outra para ajuste de arquitetura e a última para avaliar a capacidade de generalização do aprendizado.

Os resultados do trabalho mostram que o método empregado teve bons resultados para duas das três redes desenvolvidas, demonstrando potencial para a área. Mais precisamente, as RNAs desenvolvidas para calcular o coeficiente de resistência ao escoamento e o limite de escoamento apresentaram erros de estimativa baixos. Por outro lado, a RNA empregada para estimar o expoente de encruamento resultou em uma média dos valores adotados durante o treinamento.

O restante deste trabalho encontra-se organizado da maneira descrita a seguir. No Capítulo 2 descrevem-se as simulações de elementos finitos desenvolvidas. No Capítulo 3 apresenta-se uma introdução a RNAs. No Capítulo 4 são apresentados os resultados do trabalho. Por fim, no Capítulo 5 é realizada a conclusão e sugestões de trabalhos futuros.

2 Simulação de elementos finitos do ensaio de indentação instrumentada

O uso de métodos de elementos finitos para simular os resultados da indentação é bastante comum (KIM et al., 2006; BHATTACHARYA; NIX, 1991; CLYNE et al., 2021) devido à alta representatividade em relação aos dados experimentais. No presente trabalho, a simulação foi realizada utilizando o software Abaqus/CAE 2017. Em particular, a amostra foi definida como um retângulo de 4x4 mm com quadrado interno de 1,5x1,5 mm próximo ao penetrador, representado como um semicírculo de 1,5 mm de diâmetro. No quadrado interno mais próximo ao indentador, adotou-se uma malha mais refinada para simulação, com elementos de tamanho 0,0025 mm. No perímetro externo dessa malha fina, definiu-se uma malha mais grosseira a partir do método de preenchimento por número, onde o tamanho da malha ao longo de uma borda é definido por um valor numérico (SMITH, 2009): quanto menor esse valor, mais grosseira a malha. Para a simulação foi definido um valor de 1. A Figura 1 mostra a simulação da indentação desenvolvida. É importante ressaltar que, para reduzir a carga computacional da simulação, o modelo axissimétrico foi adotado. Isto é, reduzindo o número de dimensões e a quantidade de nós e simplificando a geometria, otimiza-se a solução da simulação.

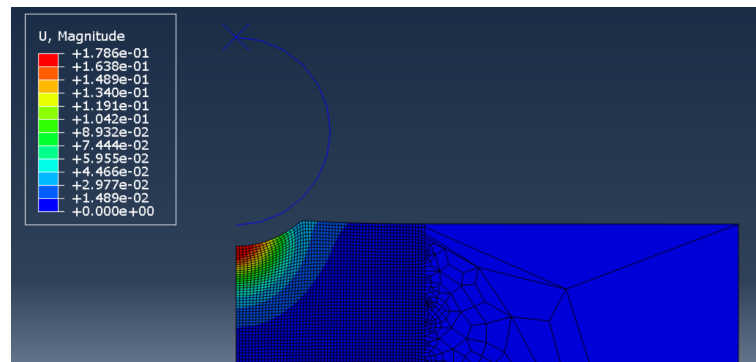


Figura 1 – Imagem do Abaqus com a simulação da indentação desenvolvida. Pode-se ver o semicírculo representando o indentador, o retângulo da amostra (com quadrado interno de malha refinada e a parte externa com malha mais grosseira).

No Abaqus, as propriedades do material são definidas a partir da criação de uma curva tensão-deformação. A parte elástica é caracterizada pelo módulo elástico de Young E e pelo coeficiente de Poisson ν . Já a parte plástica é especificada definindo um conjunto de pontos, com quantidade mínima de 2 (representando uma reta) e sem quantidade máxima. Contudo, quanto maior a quantidade de pontos, maior o custo computacional da simulação. Para esse trabalho, adotaram-se 10 pontos (tensão-deformação) para defi-

nir a fase plástica do material pois foi a quantidade mínima de pontos que resultou em uma indentação próxima ao experimental. Tais pontos forma gerados utilizando a equação empírica de tensão-deformação de Ludwik modificada (MUKARATI; MOSTERT; SIYASIYA, 2019):

$$\sigma = \sigma_y + K\epsilon^n \quad (2.1)$$

em que K é o coeficiente de resistência, n é o expoente de encruamento, σ_y é o limite de escoamento, σ e ϵ são a tensão e a deformação na fase plástica, respectivamente. Consideraram-se valores de deformação entre 2 e 8,5.

Mais ainda, na simulação definiu-se a superfície do indentador como mestre, pois assume-se que essa não sofre danos e é responsável por impor um deslocamento sobre os demais elementos da simulação. Considerou-se também um coeficiente de atrito entre as duas superfícies de 0,3. Vale ressaltar que essas configurações são adotadas tipicamente na literatura (CLYNE et al., 2021; KIM et al., 2006).

A simulação foi realizada deslocando o penetrador até 24% do seu raio. Logo, para um penetrador de raio 0,75 mm, tem-se uma profundidade de indentação de 0,18 mm. Por fim, as seguintes grandezas foram escolhidas como saídas da simulação: Forças/Reações do corpo de prova e o Deslocamento/Velocidade/Aceleração do indentador como apresentado na Figura 2.

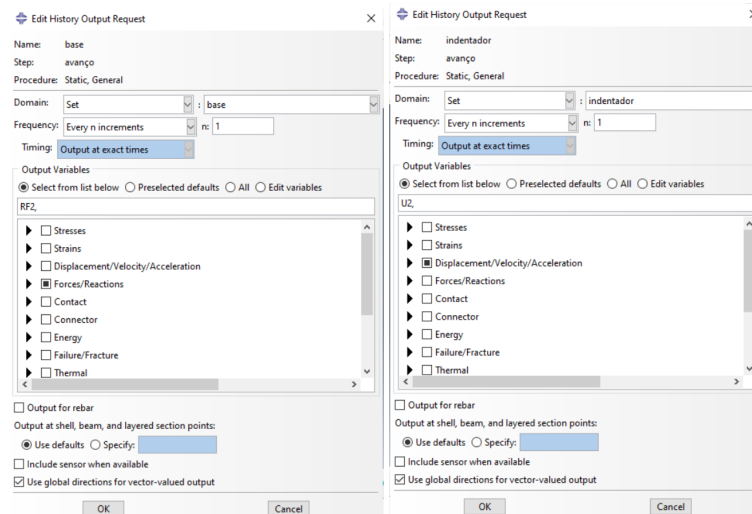


Figura 2 – Imagem da janela de configuração das saídas da simulação no Abaqus. Pode-se notar os campos *Forces/Reactions* com o parâmetro *Name* indicando ser a “base” (nome dado ao corpo de prova) e *Displacement/Velocity/Acceleration* com o parâmetro *Name* indicando ser o “indentador” marcados, correspondendo respectivamente as Forças/Reações do corpo de prova e o Deslocamento/Velocidade/Aceleração do indentador.

Essas configurações foram feitas utilizando a GUI (*Graphical User Interface*) do

Abaqus e, após isso, foram salvas em um script em Python para que a simulação possa ser replicada e acionada por meio de um script externo.

Como já mencionado, a simulação será empregada para criação do banco de dados a ser utilizado no treinamento de redes neurais artificiais. Com esse propósito, foi necessário criar um sistema de armazenamento com estrutura lógica dos arquivos, pois o Abaqus gera uma grande quantidade de dados após cada simulação, não somente os dados brutos de simulação. Com isso em mente, foi criado um script em Python com a finalidade de organizar os arquivos de simulação e registrar o material a ser simulado e os diretórios para armazenamento dos dados. Todos esses dados são salvos para que o usuário não precise inseri-los a cada execução da simulação.

À luz do que foi realizado em Kim et al. (2022), para criação do banco de dados, foram escolhidas 10.360 combinações entre coeficiente de resistência, expoente de encruamento e limite de escoamento. Com esse propósito, foram definidos intervalos para cada um desses parâmetros (vide Tabela 1). Então, empregou-se o método *Latin Hypercubic Sampling* (LHS) (MCKAY; BECKMAN; CONOVER, 1979) como amostrador dos parâmetros para a equação de Ludwik Modificada. O método LHS foi utilizado pois reduz o número de simulações necessárias para representar toda a faixa amostral. Para isso, o método separa cada faixa de valores em N partes (N representando a quantidade de amostras que para esse trabalho são 10.360). Um valor é selecionado aleatoriamente para cada uma das partes para cada variável. Por fim, todos os valores são combinados em conjuntos de forma aleatória.

Tabela 1 – Faixas de valores dos parâmetros para construção do banco de dados.

| Parâmetro | Valor Mínimo | Valor Máximo | Unidade |
|------------|--------------|--------------|---------|
| σ_y | 700 | 980 | MPa |
| K | 50 | 2500 | MPa |
| n | 0,1 | 0,3 | |

No presente trabalho foram desenvolvidas três redes distintas, correspondentes aos parâmetros da equação de Ludwik modificada apresentada. Tais redes foram denominadas RNA- n , RNA- K e RNA- σ_y . No próximo capítulo uma introdução a conceitos fundamentais de redes neurais artificiais é realizada.

3 Redes Neurais Artificiais

As Redes Neurais Artificiais são modelos computacionais inspirados no sistema nervoso biológico. Assim como as redes neurais são compostas por neurônios, as RNAs são compostas por unidades básicas denominadas perceptrons, capazes de realizar operações matemáticas simples (CHOLLET, 2018).

O perceptron é um modelo que recebe n_i entradas, denotadas por x_i , $i = 1, 2, \dots, n_i$. Essas entradas são ponderadas por pesos w_i , $i = 1, 2, \dots, n_i$, somadas e ao resultado é adicionado um viés b , gerando o potencial de ativação u . A saída do perceptron y é obtida aplicando-se uma função de ativação g ao potencial de ativação. O diagrama da Figura 3 ilustra a estrutura de um perceptron.

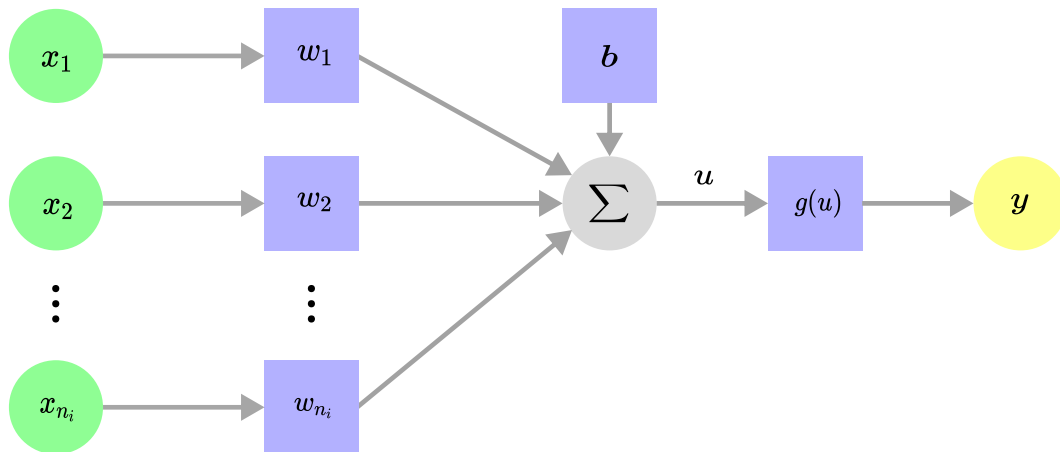


Figura 3 – Diagrama de um Perceptron.

A função de ativação é geralmente não linear, como a tangente hiperbólica. Desse modo, introduz-se não linearidades à equação, resultando em um perceptron capaz de se adequar a modelos não lineares. Como outro exemplo, na Figura 4 ilustra-se a função de ativação *Rectifier Linear Unit* (ReLU) utilizada principalmente em RNAs aplicadas em problemas de regressão (GOODFELLOW; BENGIO; COURVILLE, 2016).

Ao interconectar diversos perceptrons, forma-se um perceptron de multicamadas ou uma RNA (vide Figura 5). Nessa rede, uma camada é composta de perceptrons com o mesmo conjunto de entradas e resulta na mesma camada de saída. A partir de entradas x_i , $i = 1, 2, \dots, n_i$, pode-se calcular as saídas da primeira camada da RNA. Então, é possível calcular as saídas da segunda camada e, assim por diante, até serem obtidas as saídas da RNA (y_i , $i = 1, 2, \dots, n_o$). Esse processo é conhecido como *feed-forward* (GARDNER; DORLING, 1998).

Com valores adequados de pesos e vieses, uma RNA é capaz de representar dife-

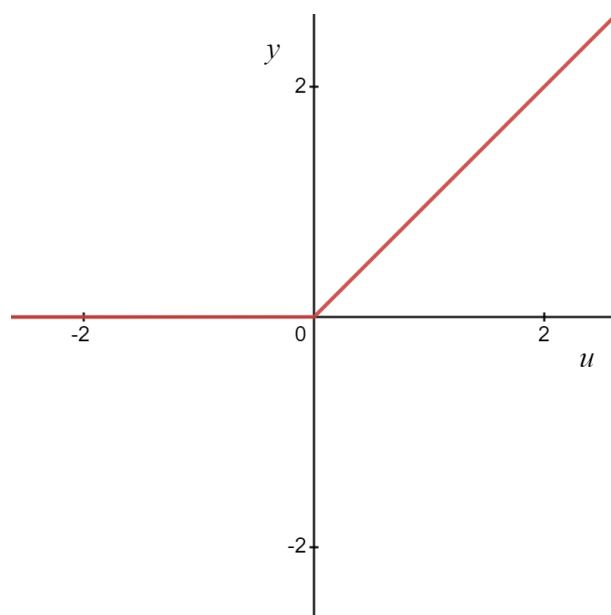


Figura 4 – Representação da função de ativação ReLU.

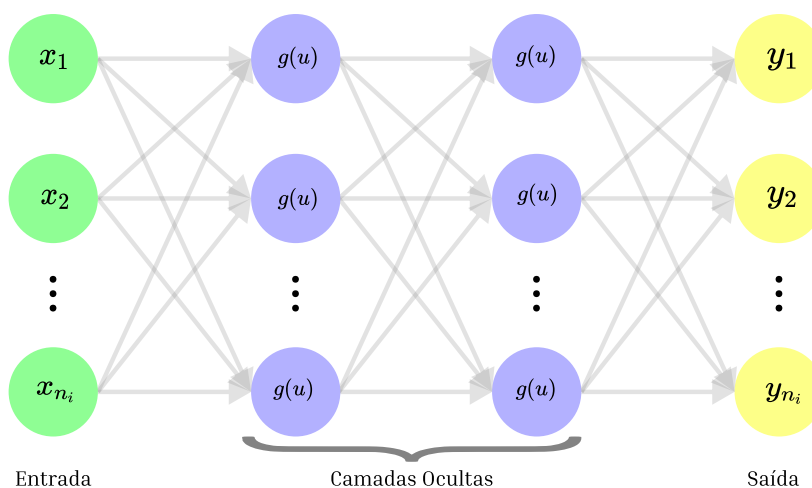


Figura 5 – Diagrama de uma Rede Neural Artificial.

rentes conjuntos de dados. O treinamento de uma rede neural consiste em encontrar tais valores adequados a partir de dados conhecidos de entradas e saídas.

Durante o treinamento, as entradas são introduzidas na RNA e as saídas são calculadas usando os valores atuais de pesos e vieses. Então são calculadas as diferenças entre essas saídas e as saídas esperadas. As diferenças obtidas são utilizadas para calcular um índice de desempenho J , por exemplo, o erro médio quadrático. O ajuste dos parâmetros é realizado de modo a minimizar o índice de desempenho, em um processo denominado *backpropagation*. Especificamente, como as funções de ativação são diferenciáveis, pode-se calcular o gradiente do custo em relação a cada peso ou viés da rede. Esse gradiente é utilizado para atualizar o valor do peso de acordo com a seguinte equação:

$$p_{nov} = p_{ant} - \alpha \frac{\partial J}{\partial p_{ant}} \quad (3.1)$$

em que p denota um peso genérico, os subíndices *nov* e *ant* indicam respectivamente os valores novo e anterior desse peso, e $\frac{\partial J}{\partial p}$ é a derivada parcial do custo em relação ao peso p . Já α é a taxa de aprendizado e determina a magnitude da atualização do peso com base em um exemplo. Se o valor de α for elevado, o treinamento pode não alcançar um mínimo; se o valor for muito baixo, o treinamento tomará mais tempo (BATTITI, 1992).

Ademais, devido a variações do erro, tanto em função da quantidade de dados necessárias para treinar a rede quanto a presença de *outliers* nos dados (BATTITI, 1992), a atualização dos pesos da rede é feita utilizando o agrupamento de um conjunto de exemplos. Esse agrupamento de dados é denominado de lote. A utilização de um tamanho de lote N_b muito pequeno faz com que o treinamento necessite de mais iterações para convergir e aumenta a probabilidade de estagnação em um mínimo local. Já um tamanho de lote excessivamente grande aumenta o custo computacional e pode resultar em um desempenho inadequado em dados não usados no treinamento (DEVARAKONDA; NAUMOV; GARLAND, 2018).

Além de α e N_b , uma outra decisão importante na criação de uma RNA é a definição da arquitetura da rede, isto é, do número de camadas, do número de perceptrons por camada (no contexto de RNAs é denominado de nós) e do tipo de função de ativação. Sabe-se que essa arquitetura tem impacto significativo no desempenho da rede (WU; YANG; CAO, 2021). Com esse propósito, pode-se realizar análises do desempenho da rede no conjunto de dados de treinamento e também em dados que não foram utilizados para treinar, mais precisamente nos conjuntos de validação e teste que serão detalhados mais adiante.

Não há metodologia específica para definição da arquitetura de uma RNA na literatura. Tal etapa é dita “mais arte do que ciência” (CHOLLET, 2018). Em compensação, as análises de desempenho da rede e da capacidade de generalização são amplamente documentadas. Um dos princípios é evitar que uma rede com nós e camadas em excesso “superespecialize” nos padrões dos dados durante o treinamento, perdendo a capacidade de generalização do aprendizado em novos dados. Esse fenômeno, representado na Figura 6, é denominado de sobreajuste (*overfit*) e tipicamente é evitado de três maneiras. A primeira é parar o treinamento antes que o sobreajuste ocorra. Entretanto, pode ser difícil definir precisamente quando parar o treinamento (SCHAFFER, 1993). O segundo método é diminuir a complexidade da rede, assim a capacidade da rede de se adaptar aos ruídos será reduzida. Contudo, isso ocorre ao custo da capacidade de generalização da rede sendo mitigada com a redução dessa complexidade da rede (SCHAFFER, 1993). O terceiro método se baseia na divisão dos dados para treinamento em três conjuntos:

- Conjunto de treinamento: adotado durante o treinamento;
- Conjunto de validação: usado para avaliar a capacidade de generalização da rede durante o treinamento e para realizar ajustes nos hiperparâmetros da rede;
- Conjunto de teste: empregado para avaliar a performance da rede após o treinamento e o ajuste paramétrico.

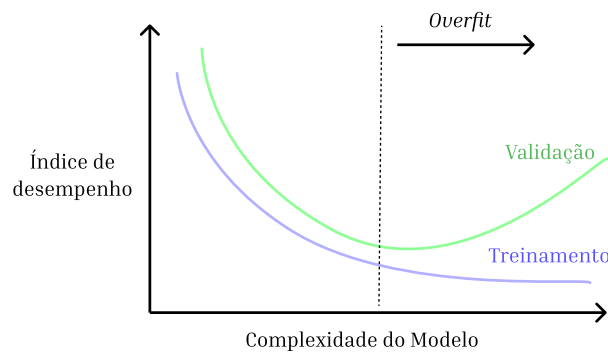


Figura 6 – Curva de treinamento quando há sobreajuste, isto é, uma piora em dados inéditos para a rede, enquanto o desempenho de treinamento melhora. Isso indica que a rede se “superespecializou” aos dados de treinamento e pode não generalizar o aprendizado.

A última solução permite que a rede mantenha capacidade máxima de generalização evitando que ocorra sobreajuste. Vale comentar que existem outras técnicas mais avançadas para evitar sobreajuste. Por exemplo, pode-se citar o *dropout* e a regularização. Para detalhes, o leitor pode consultar Zhang (2010).

Um outro fator que afeta significativamente o desempenho de RNAs é diferença de magnitude entre as entradas é uma prática amplamente utilizada na literatura normalizar as entradas da rede antes dos cálculos das saídas (CHOLLET, 2018). Esse processo de alteração dos dados de treinamento é popularmente denominado de pré-processamento. A normalização é aplicada nos dados de entrada do conjunto de treinamento de modo que todas as entradas tenham a mesma faixa. Neste trabalho, realizou-se a normalização *z-score*:

$$z_i = \frac{(x_i - \bar{X}_i)}{\sqrt{\sigma_i}}, \quad i = 1, 2, \dots, n_i \quad (3.2)$$

no qual z_i é a entrada normalizada, \bar{X}_i é a média e σ_i é a variância dessa entrada com base no conjunto de treinamento. Os valores de \bar{X}_i e σ_i são computados antes de iniciar o treinamento e esses valores são utilizados para formar uma camada fixa de normalização na arquitetura da rede. Como efeito, todas as entradas estarão entre 0 e 1 (BA; KIROS;

HINTON, 2016; ABADI et al., 2015). Vale destacar que, durante a aplicação da RNA, os mesmos parâmetros de normalização obtidos no conjunto de dados de treinamento devem ser aplicados às novas entradas.

Com a popularização da inteligência artificial, em especial de RNAs, diversas bibliotecas e plataformas de desenvolvimento foram disponibilizadas gratuitamente para uso, como KerasTM, Python e MXnetTM (KIM; WIMMER; KIM, 2022). Para esse trabalho, foi utilizada a versão em Python da *framework* desenvolvida pelo GoogleTM denominada TensorflowTM. A mesma é desenvolvida em torno da biblioteca KerasTM e é amplamente utilizada pela comunidade de desenvolvedores, tornando-a confiável e robusta. O GoogleTM oferece um serviço para programação em Python em nuvem com plano gratuito, no qual implementa essa biblioteca nativamente.

No presente estudo, o TensorflowTM será empregado para desenvolver RNAs para estimar propriedades mecânicas a partir de simulações da indentação. Na sequência, os resultados do trabalho são apresentados.

4 Resultados

Como o treinamento das redes neurais artificiais pode envolver um esforço computacional relativamente alto, todo o processo de treinamento deste trabalho foi feito na plataforma Google Colab™. Essa plataforma permite que códigos em *Python* sejam desenvolvidos e executados em nuvem utilizando de recursos dos servidores do Google™ para realizar as computações. As vantagens incluem acesso a GPUs (*graphics processing unit*) e TPUs (*tensor processing unit*) de alto rendimento mantidas e desenvolvidas pela empresa. Além disso, o código foi desenvolvido em formato Notebook, trazendo a vantagem da execução parcial do código, organização, reprodutibilidade e adaptabilidade, principalmente para análises em desenvolvimentos de pesquisa, sendo possível gerar diferentes figuras, tabelas e dados sem a necessidade de executar o código como um todo. Por fim, o ambiente virtual do site oferece suporte a diversas bibliotecas como Tensorflow™, *numpy*, biblioteca com suporte a diversas operações matemáticas, *pandas*, biblioteca para suporte na importação e análise de dados, entre outras.

No presente trabalho, foram desenvolvidas três redes distintas, uma para cada parâmetro da equação de Ludwik modificada. Tais redes foram denominadas RNA- n , RNA- K e RNA- σ_y . Todas as RNAs foram criadas inicialmente com a mesma arquitetura: quatro camadas ocultas de 64 nós, com uma camada de normalização na entrada. Como o objetivo de cada rede é estimar uma propriedade mecânica (i.e. regressão), a camada de saída foi definida como um único nó sem função de ativação. Os demais parâmetros para o treinamento e construção das redes neurais se encontram na tabela a seguir.

Tabela 2 – Hiperparâmetros das RNAs.

| | |
|--------------------------------|-----------------------------|
| Tamanho de lote | 32 |
| Índice de desempenho | Erro médio quadrático |
| Função de otimização | Adam |
| Taxa de aprendizado | 0,001 |
| Função de ativação | Relu |
| Método de Inicialização | Normal He (HE et al., 2015) |

Os dados do banco de dados foram embaralhados e separados em conjuntos de treinamento, validação e teste, com uma razão de 8:1:1. Isso é tipicamente realizado em aplicações de IA para evitar que os dados de treinamento sejam o mesmo de validação. Dessa forma, é possível identificar se houve uma superespecialização da rede para os dados de treinamento.

Assim como realizado em Kim et al. (2022), adotaram-se os seguintes parâmetros de entrada das redes neurais:

- os valores de força para as profundidades de indentação de $\{0,025; 0,1; 0,125\}$ mm e na profundidade máxima, simbolizados pelas variáveis $\{f_{25}; f_{100}; f_{125}; f_{max}\}$;
- os coeficientes linear A e angular B de uma reta ajustada nos dados de força por deslocamento experimentais de carregamento do ensaio. Também considerou-se a área abaixo dessa reta representada pela variável $A_{f \times h}$.

Esses parâmetros estão representados na Figura 7.

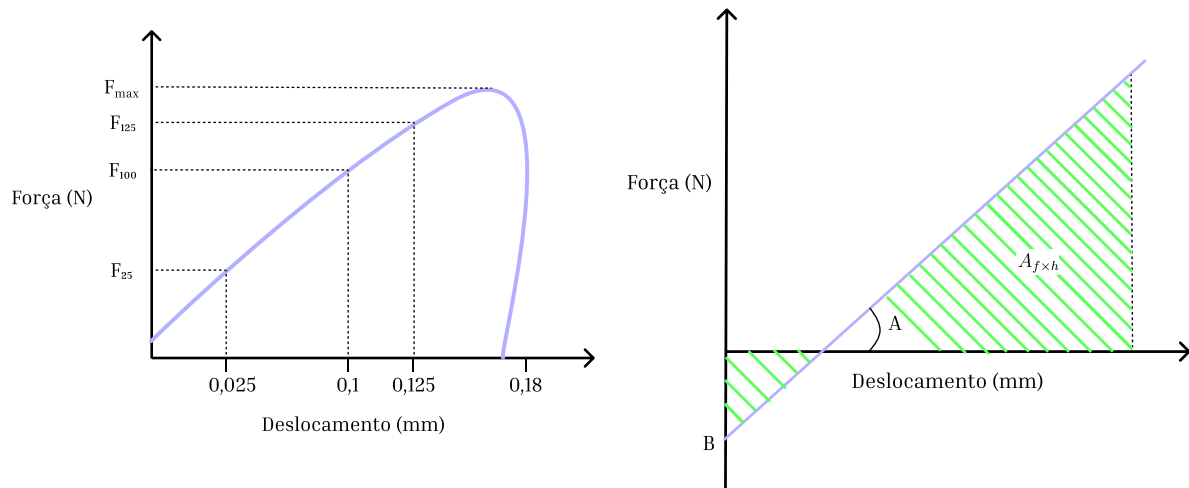


Figura 7 – À direita, o gráfico de força por deslocamento com os pontos utilizados como entrada das RNAs e à esquerda, a linearização do carregamento desse gráfico destacando o coeficiente angular A , o coeficiente linear B e a área da reta $A_{f \times h}$.

4.1 Resultados na estimativa do expoente de encruamento

O treinamento da RNA- n foi feito em 200 épocas, isto é, o *backpropagation* foi realizado 200 vezes para ajuste dos parâmetros da rede a partir dos dados do conjunto de treinamento. A evolução do erro médio quadrático (MSE - *Mean Squared Error*) ao longo do treinamento é apresentada na Figura 8. Como é possível notar, por volta da época 20, o MSE no conjunto de validação passa a aumentar ligeiramente. Isso é o comportamento de uma RNA com superespecialização (CHOLLET, 2018).

Existem diversas formas para mitigar esse problema. Aqui optou-se por reduzir a complexidade da rede adotando uma única camada oculta de 32 nós. O resultado está na Figura 9 e é possível notar que não houve superespecialização.

Após o treinamento, foram avaliadas inicialmente 10 amostras do conjunto de dados de teste. A Tabela 4 apresenta uma comparação entre as saídas calculadas pela RNA- n e os valores reais para essas 10 amostras. Pode-se verificar que essa rede resultou em uma média de erro baixa. Contudo, como será evidenciado adiante, a variação de n no

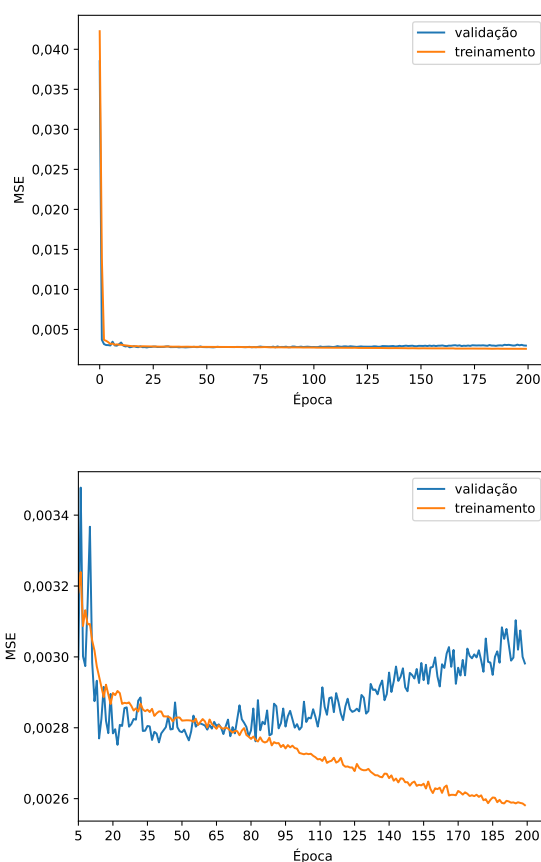


Figura 8 – Evolução do MSE para os conjuntos de dados de treinamento e validação da RNA-*n*.

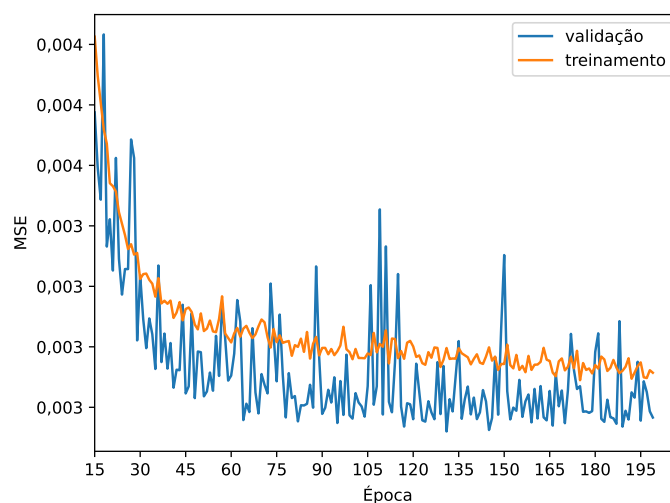


Figura 9 – MSE nos conjuntos de treinamento e validação a partir da 15^a época, relativos à nova arquitetura para RNA-*n*.

banco de dados não é significativa e a rede tem comportamento similar a de uma regressão linear.

Tabela 3 – Resultados para 10 amostras no conjunto de dados de teste da RNA- n .

| Saída real | Saída estimada | Diferença | Diferença Relativa (%) |
|----------------------|----------------|-------------|------------------------|
| 0,12 | 0,20 | -0,08 | 64,22% |
| 0,13 | 0,19 | -0,06 | 42,60% |
| 0,25 | 0,20 | 0,05 | 21,23% |
| 0,28 | 0,20 | 0,08 | 29,28% |
| 0,13 | 0,19 | -0,06 | 48,92% |
| 0,27 | 0,19 | 0,08 | 30,41% |
| 0,19 | 0,20 | -0,01 | 5,70% |
| 0,24 | 0,19 | 0,05 | 21,46% |
| 0,18 | 0,18 | 0,00 | 0,07% |
| 0,25 | 0,21 | 0,04 | 15,17% |
| Média | | 0,01 | 20,54% |
| Desvio padrão | | 0,06 | 10,70% |

4.2 Resultados na estimativa do limite de escoamento

O treinamento da rede RNA- σ_y também foi efetuado com 200 épocas e não houve superespecialização. Os resultados do treinamento estão representados na Figura 10. Uma comparação para dez amostras do conjunto de teste encontra-se na Tabela 4. Como pode ser verificado, o treinamento da RNA- σ_y resultou em desempenho satisfatório. Isto é, não houve superespecialização e a média de erro foi baixa. Logo, não foi necessário alterar a arquitetura de RNA inicial.

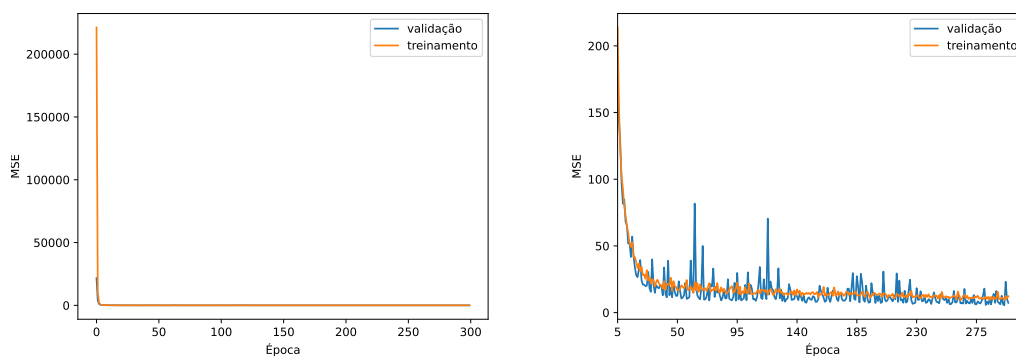


Figura 10 – MSE nos conjuntos de dados de treinamento e validação para RNA- σ_y . Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 5ª época.

4.3 Resultados na estimativa do coeficiente de resistência

O treinamento inicial da RNA- K foi feito em 200 épocas e resultou em um erro muito alto em relação ao valor máximo da faixa no qual foi construído o banco de dados. Outro treinamento foi realizado com 300 épocas, mas não houve melhora significativa. O

Tabela 4 – Resultados para 10 amostras do conjunto de dados de teste da RNA- σ_y .

| Saída real | Saída estimada | Diferença | Diferença Relativa (%) |
|------------|----------------------|-------------|------------------------|
| 823 | 829,70 | -6,70 | 0,81% |
| 822 | 819,66 | 2,34 | 0,28% |
| 950 | 948,55 | 1,45 | 0,15% |
| 899 | 899,57 | -0,57 | 0,06% |
| 895 | 894,58 | 0,42 | 0,05% |
| 921 | 922,44 | -1,44 | 0,16% |
| 893 | 891,21 | 1,79 | 0,20% |
| 810 | 807,70 | 2,30 | 0,28% |
| 783 | 781,83 | 1,17 | 0,15% |
| 743 | 743,60 | -0,60 | 0,08% |
| | Média | 0,02 | 0,25% |
| | Desvio padrão | 2,56 | 0,23% |

resultado do segundo treinamento está representado na Figura 11 e uma avaliação para o conjunto de dados de teste é apresentada na Tabela 5.

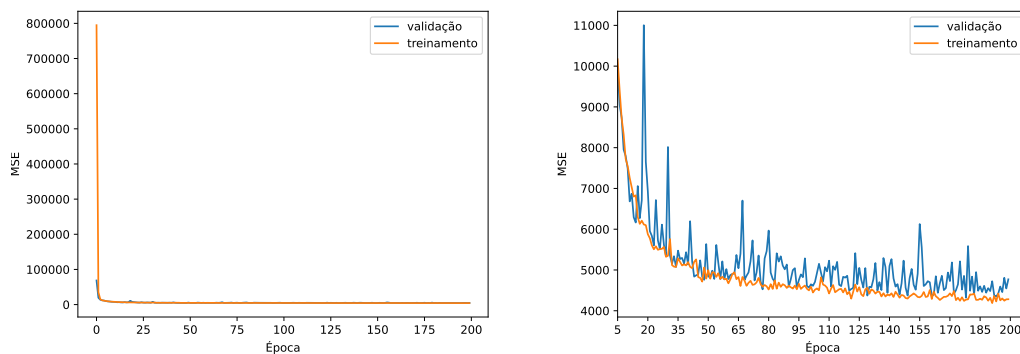


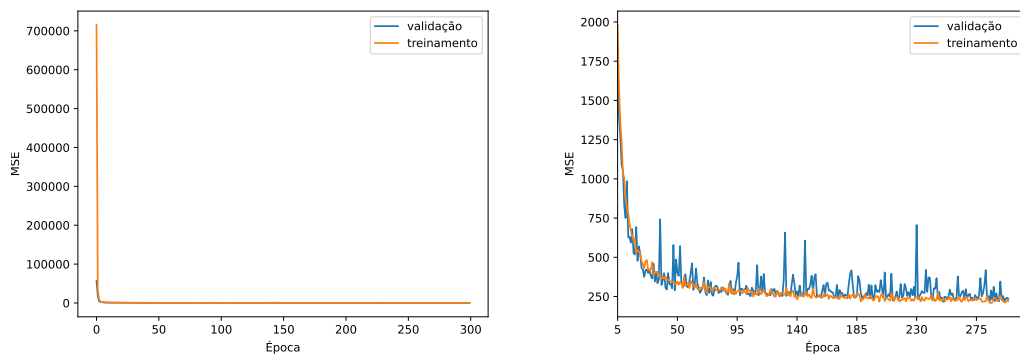
Figura 11 – MSE nos conjuntos de dados de treinamento e validação para RNA- K . Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 5ª época.

Tendo em vista as boas estimativas de σ_y e n obtidas anteriormente, incluíram-se esses parâmetros como entradas da RNA- K , visando melhoria de desempenho. O resultado do treinamento para 300 épocas é mostrado na Figura 12. Pode-se constatar que adicionar σ_y e n como parâmetros de entrada da RNA- K melhorou significativamente o desempenho da rede.

É possível notar na Figura 12 que os valores de MSE de treinamento e validação da RNA- K reduziram com a adoção dos novos parâmetros de entrada (decreceram de 4500 para 250, aproximadamente). Ademais, a Tabela 6 evidencia que incluir essas novas entradas melhora o desempenho da rede em dados inéditos. Ou seja, melhora a capacidade de generalização da rede.

Tabela 5 – Resultados para 10 amostras no conjunto de dados de teste para rede RNA- K .

| Saída real | Saída estimada | Diferença | Diferença Relativa (%) |
|------------|----------------------|--------------|------------------------|
| 1399 | 1244,45 | 154,55 | 11,05% |
| 182 | 158,04 | 23,96 | 13,17% |
| 2093 | 2148,94 | -55,94 | 2,67% |
| 1678 | 1746,94 | -68,94 | 4,11% |
| 305 | 268,02 | 36,98 | 12,13% |
| 1041 | 1099,73 | -58,73 | 5,64% |
| 793 | 807,51 | -14,51 | 1,83% |
| 270 | 250,43 | 19,57 | 7,25% |
| 2038 | 2052,63 | -14,63 | 0,72% |
| 2287 | 2377,72 | -90,72 | 3,97% |
| | Média | -6,84 | 2,35% |
| | Desvio padrão | 67,62 | 1,81% |

Figura 12 – MSE nos conjuntos de dados de treinamento e validação para RNA- K modificado. Na figura da esquerda, todos os dados de treinamento. Na figura da direita, os dados a partir da 8ª época.Tabela 6 – Resultados para 10 amostras no conjunto de dados de teste para rede RNA- K modificada.

| Saída real | Saída estimada | Diferença | Diferença Relativa (%) |
|------------|----------------------|--------------|------------------------|
| 1399 | 1390,47 | 8,53 | 0,61% |
| 182 | 180,18 | 1,82 | 1,00% |
| 2093 | 2081,09 | 11,91 | 0,57% |
| 1678 | 1671,18 | 6,82 | 0,41% |
| 305 | 303,51 | 1,49 | 0,49% |
| 1041 | 1034,12 | 6,88 | 0,66% |
| 793 | 767,98 | 25,02 | 3,16% |
| 270 | 249,34 | 20,66 | 7,65% |
| 2038 | 2047,67 | -9,67 | 0,47% |
| 2287 | 2308,57 | -21,57 | 0,94% |
| | Média | 5,19 | 0,50% |
| | Desvio padrão | 12,87 | 0,34% |

4.4 Resultados do conjunto de teste

Além das análises anteriores, foi feita uma verificação do desempenho das redes em todo o conjunto de teste. Essa verificação encontra-se na Figura 13. Mais precisamente, apresenta-se na Figura 13 gráficos nos quais os eixos das ordenadas são os valores das propriedades mecânicas estimadas pelas RNAs para todas as amostras de tal conjunto. Já os eixos das abscissas são valores reais. Logo, quanto menor o erro de estimativa, mais os dados se encontrarão distribuídos ao longo de uma reta que parte da origem com inclinação 45° . Essa reta é ilustrada pela linha vermelha na Figura 13. Pode-se verificar que, para as redes RNA- K modificada e RNA- σ_y , os dados estão próximos da reta, demonstrando que os valores preditos são semelhantes aos valores reais. Em contrapartida, a RNA- n resultou em saídas por volta de 0,20, independente do valor real, indicando um desempenho inadequado. Isso pode estar associado com a faixa dos valores de n adotada na construção do banco de dados.

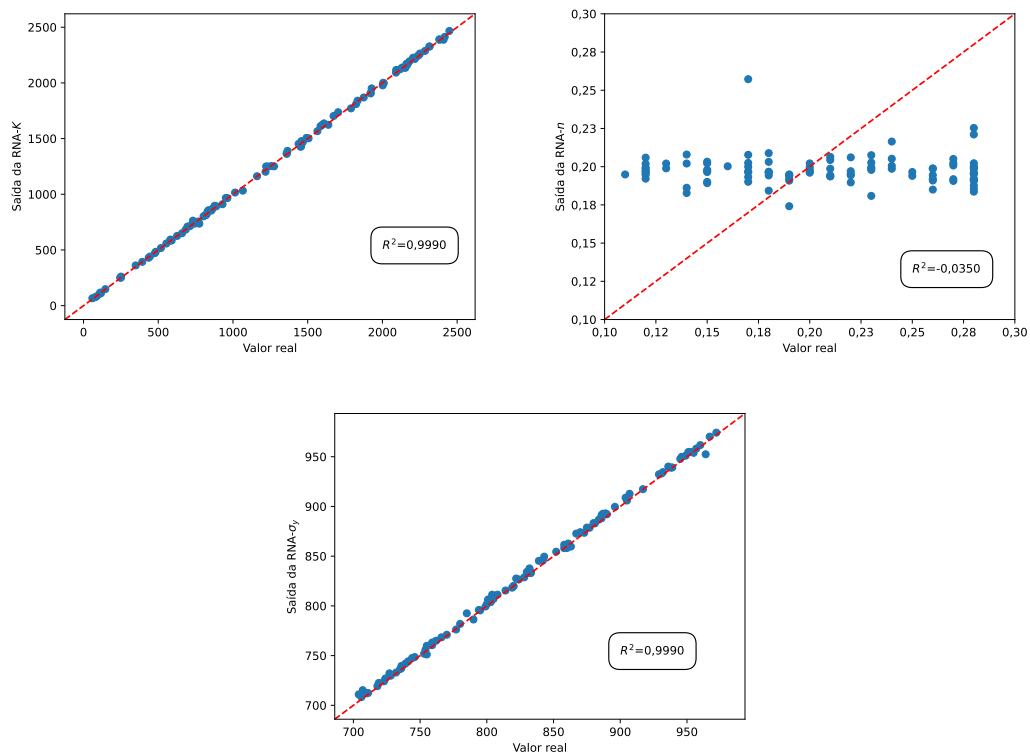


Figura 13 – Comparação entre os valores preditos e reais das propriedades mecânicas dos materiais para o conjunto de dados de teste.

5 Conclusões e sugestões de trabalhos futuros

A indentação é uma técnica comumente empregada para caracterização de materiais. Contudo, a análise dos resultados desse ensaio pode ser uma tarefa complexa. Uma possibilidade para realização de tal análise consiste em empregar técnicas de Inteligência Artificial. Neste contexto, o presente trabalho envolveu o desenvolvimento de Redes Neurais Artificiais voltadas à estimativa de propriedades mecânicas a partir dos resultados da indentação esférica de ciclo único.

Com o propósito de desenvolver tais redes, foi criada uma simulação de elementos finitos do ensaio de indentação instrumentada no qual as características dos materiais fossem parametrizados. Em particular, 10.360 combinações diferentes de expoente de encruamento n , coeficiente de resistência ao escoamento K e limite de escoamento σ_y foram adotadas para construir um banco de dados com os resultados de simulação. Posteriormente, separou-se esse banco em dados de treinamento, validação e teste, em uma razão 8:1:1, respectivamente. Essa divisão foi feita para que, após o treinamento fosse possível avaliar o desempenho da rede no conjunto de validação. Isso possibilitou a definição de uma arquitetura de rede para cada parâmetro a ser estimado. Por fim, após escolhida a arquitetura e realizado o treinamento, o conjunto de teste permitiu avaliar o desempenho da rede em dados inéditos.

Foram desenvolvidas três redes neurais para os parâmetros da equação de Ludwik que define a lei plástica dos materiais dúcteis: RNA- n , RNA- K e RNA- σ_y . As entradas das redes neurais foram f_{25} ; f_{100} ; f_{125} ; f_{max} , A , B e $A_{f \times h}$, conforme proposto na literatura. Para verificar se essas entradas poderiam causar superespecialização, foi feita a análise utilizando da matriz de correlação entre as entradas que evidenciou a pouca relação entre B e f_{25} . Contudo, como houve correlação com as outras cinco entradas, optou-se pela não remoção de B .

Inicialmente, todas as redes constituíram-se de 4 camadas de 64 nós além de uma camada de normalização na entrada da rede. Nessa configuração, a RNA- σ_y teve excelente desempenho com baixos valores de MSE e R^2 de 0,99. Entretanto, RNA- K e RNA- n não obtiverem bons desempenhos com a mesma arquitetura. Para RNA- K foi proposto tornar σ_y e n variáveis de entrada do sistema, o que melhorou seu desempenho. Já no caso da RNA- n , a arquitetura inicial levou à superespecialização e foi necessário simplificar a rede para uma única camada de 32 nós no qual resultou na resolução do problema e um MSE baixo. Entretanto, um valor de R^2 baixo com a rede predizendo valores em torno de 0,2. Esse comportamento pode ser resultado de uma faixa de dados de simulação curta de n

e uma resolução baixa, com duas casas decimais para a variável.

Para futuras pesquisas, recomenda-se a avaliação do método de IA em uma variedade maior de materiais, condições de ensaio e a utilização de dados experimentais. Além disso, sugere-se avaliar o uso de outras equações constitutivas como a equação de Swift utilizada em Kim et al. (2022). Outra sugestão é a utilização de dados de indentadores de formatos distintos no mesmo ensaio. Por fim, analisar o desempenho da rede a partir da diferença entre as curvas tensão \times deformação predita e real.

Referências

- ABADI, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from [tensorflow.org](https://www.tensorflow.org/). Disponível em: <<https://www.tensorflow.org/>>.
- ASHBY, M. F.; JONES, D. R. H. *Engineering Materials 1: An Introduction to Properties, Applications, and Design*. 4. ed. Waltham, Massachusetts, Estados Unidos da América: Elsevier, 2012.
- BA, J. L.; KIROS, J. R.; HINTON, G. E. *Layer Normalization*. 2016.
- BATTITI, R. First- and second-order methods for learning: Between steepest descent and newton's method. *Neural Computation*, v. 4, n. 2, p. 141–166, 1992. Disponível em: <<https://doi.org/10.1162/neco.1992.4.2.141>>.
- BHATTACHARYA, A.; NIX, W. Finite element analysis of cone indentation. *International Journal of Solids and Structures*, Elsevier BV, v. 27, n. 8, p. 1047–1058, 1991. ISSN 0020-7683. Disponível em: <[https://doi.org/10.1016/0020-7683\(91\)90100-t](https://doi.org/10.1016/0020-7683(91)90100-t)>.
- CHOLLET, F. *Deep Learning with Python*. 1. ed. Nova York, Estados Unidos da América: Manning Publications Co., 2018.
- CLYNE, T. W. et al. Profilometry-based inverse finite element method indentation plastometry. *Advanced Engineering Materials*, Wiley, v. 23, n. 9, ago. 2021. ISSN 1527-2648. Disponível em: <<https://doi.org/10.1002/adem.202100437>>.
- DEVARAKONDA, A.; NAUMOV, M.; GARLAND, M. *AdaBatch: Adaptive Batch Sizes for Training Deep Neural Networks*. 2018.
- GARDNER, M.; DORLING, S. Artificial neural networks (the multilayer perceptron)a review of applications in the atmospheric sciences. *Atmospheric Environment*, v. 32, n. 14, p. 2627–2636, 1998. ISSN 1352-2310. Disponível em: <[https://doi.org/10.1016/S1352-2310\(97\)00447-0](https://doi.org/10.1016/S1352-2310(97)00447-0)>.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. Cambridge, Massachusetts, Estados Unidos da América: MIT press, 2016.
- HAGGAG, F. In situ measurement of mechanical properties using novel automated ball indentation system. *ASTM Special Technical Publication*, p. 27–44, 01 1993. Disponível em: <<https://doi.org/10.1520/STP12719S>>.
- HE, K. et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV 2015)*, v. 1502, p. 1026–1034, 02 2015. Disponível em: <<https://doi.org/10.1109/ICCV.2015.123>>.
- JEONG, K. et al. Prediction of uniaxial tensile flow using finite element-based indentation and optimized artificial neural networks. *Materials and Design*, v. 196, p. 109104, nov 2020. ISSN 0264-1275. Disponível em: <<https://doi.org/10.1016/j.matdes.2020.109104>>.

KIM, S.; WIMMER, H.; KIM, J. Analysis of deep learning libraries: Keras, pytorch and mxnet. In: *2022 IEEE/ACIS 20th International Conference on Software Engineering Research, Management and Applications (SERA)*. [s.n.], 2022. p. 54–62. Disponível em: <<https://doi.org/10.1109/SERA54885.2022.9806734>>.

KIM, S. H. et al. Quantitative determination of contact depth during spherical indentation of metallic materials A FEM study. *Materials Science and Engineering: A*, v. 415, n. 1, p. 59–65, 2006. ISSN 0921-5093. Disponível em: <<https://doi.org/10.1016/j.msea.2005.08.217>>.

KIM, Y. et al. Novel deep learning approach for practical applications of indentation. *Materials Today Advances*, v. 13, p. 100207, 2022. ISSN 2590-0498. Disponível em: <<https://doi.org/10.1016/j.mtadv.2022.100207>>.

LI, Y. et al. Improvement of predicting mechanical properties from spherical indentation test. *International Journal of Mechanical Sciences*, v. 117, p. 182–196, 2016. ISSN 0020-7403. Disponível em: <<https://doi.org/10.1016/j.ijmecsci.2016.08.019>>.

LIU, Y.; ZHAO, X.; WANG, D. Determination of the plastic properties of materials treated by ultrasonic surface rolling process through instrumented indentation. *Materials Science and Engineering: A*, Elsevier BV, v. 600, p. 21–31, abr. 2014. ISSN 0921-5093. Disponível em: <<https://doi.org/10.1016/j.msea.2014.01.096>>.

MAUVOISIN, G. et al. Determination of the hardness profile on heat treated steels by hole drilling and indentation methods. *Journal of the Mechanical Behavior of Materials*, v. 11, n. 6, p. 437–456, 2000. Disponível em: <<https://doi.org/10.1515/JMBM.2000.11.6.437>>.

MCKAY, M. D.; BECKMAN, R. J.; CONOVER, W. J. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, JSTOR, v. 21, n. 2, p. 239–245, maio 1979. ISSN 0040-1706. Disponível em: <<https://doi.org/10.2307/1268522>>.

MOHAN, S. et al. Critical evaluation of spherical indentation stress-strain protocols for the estimation of the yield strengths of steels. *Experimental Mechanics*, Springer Science and Business Media LLC, v. 61, n. 4, jan 2021. ISSN 1741-2765. Disponível em: <<https://doi.org/10.1007/s11340-021-00689-7>>.

MUKARATI, T. W.; MOSTERT, R. J.; SIYASIYA, C. W. Development of a mathematical equation describing the strain hardening behaviour of metastable AISI 301 austenitic stainless steel. *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, v. 655, n. 1, p. 012008, oct 2019. ISSN 1757-899X. Disponível em: <<https://doi.org/10.1088/1757-899X/655/1/012008>>.

NICOLOSI, E. R. *Avaliação de métodos analíticos para determinação de propriedades mecânicas de aços via ensaio de macroindentação instrumentada*. Dissertação (Mestrado) — Universidade Federal de Uberlândia, 2015.

PARK, S. et al. Determination of material properties of bulk metallic glass using nanoindentation and artificial neural network. *Intermetallics*, v. 144, p. 107492, 2022. ISSN 0966-9795. Disponível em: <<https://doi.org/10.1016/j.intermet.2022.107492>>.

- PHARR, G. M.; HERBERT, E. G.; GAO, Y. The indentation size effect: A critical examination of experimental observations and mechanistic interpretations. *Annual Review of Materials Research*, Annual Reviews, v. 40, n. 1, p. 271–292, jun. 2010. ISSN 1545-4118. Disponível em: <<https://doi.org/10.1146/annurev-matsci-070909-104456>>.
- SCHAFFER, C. Overfitting avoidance as bias. *Machine Learning*, Springer Science and Business Media LLC, v. 10, n. 2, p. 153–178, fev. 1993. ISSN 1573-0565. Disponível em: <<https://doi.org/10.1007/bf00993504>>.
- SMITH, M. *ABAQUS/Standard User's Manual, Version 6.9*. Estados Unidos da América: Dassault Systèmes Simulia Corp, 2009.
- TABOR, D. *The Hardness of Metals*. Londres: Clarendon Press, 1951. (Monographs on the physics and chemistry of materials). ISBN 9780598845153.
- TAN, X.; SHEN, Y. Modeling analysis of the indentation-derived yield properties of metallic multilayered composites. *Composites Science and Technology*, v. 65, n. 11, p. 1639–1646, 2005. ISSN 0266-3538. Disponível em: <<https://doi.org/10.1016/j.compscitech.2004.12.051>>.
- WALLEY, S. M. Historical origins of indentation hardness testing. *Materials Science and Technology*, Informa UK Limited, v. 28, n. 9,10, p. 1028–1044, oct 2012. ISSN 1743-2847. Disponível em: <<https://doi.org/10.1179/1743284711Y.0000000127>>.
- WU, S.-w.; YANG, J.; CAO, G.-m. Prediction of the charpy v-notch impact energy of low carbon steel using a shallow neural network and deep learning. *International Journal of Minerals, Metallurgy and Materials*, Springer Science and Business Media LLC, v. 28, n. 8, p. 1309–1320, maio 2021. ISSN 1869-103X. Disponível em: <<https://doi.org/10.1007/s12613-020-2168-z>>.
- YU, F. et al. A new instrumented spherical indentation test methodology to determine fracture toughness of high strength steels. *Theoretical and Applied Fracture Mechanics*, v. 124, p. 103744, 2023. Disponível em: <<https://doi.org/10.1016/j.tafmec.2022.103744>>.
- ZHANG, X. Regularization. In: _____. *Encyclopedia of Machine Learning*. Boston, MA: Springer US, 2010. p. 845–849. ISBN 978-0-387-30164-8. Disponível em: <https://doi.org/10.1007/978-0-387-30164-8_712>.