

Uso de regressores para a geração de conhecimento na avaliação do ensaio de tração ambiente

Keila de C. Freitas^{1,2}, MSc. Kairo de B. Guimarães¹,
Luxmar A. Xavier¹, Thiago A. de O. Silva³

¹USIMINAS

Av. Pedro Linhares Gomes, 5.431 - Bairro Usiminas – Ipatinga- MG

²Curso de Especialização em Ciência de Dados
Universidade Federal de Ouro Preto

³Instituto de Ciências Exatas e Aplicadas
Universidade Federal de Ouro Preto – João Monlevade, MG – Brasil

keila.freitas@usiminas.com, kairo.guimaraes@usiminas.com

luxmar.xavier@usiminas.com, thiago@ufop.edu.br

Abstract. *The ambient tensile test is responsible for evaluating mechanical properties in the steelmaking process, which returns results such as: resistance limit (LR), yield strength (LE) and elongation (ALO). The objective is to build models for each property, using process variables and chemical composition data, to predict the test responses, helping in the analyzes carried out by specialists. Nine different regression algorithms with satisfactory determination coefficients for the three indicators were adjusted. The resulting models contribute to the decision making of the test by obtaining gains in sample evaluation time and also by proposing a method that identifies false positives.*

Resumo. *O ensaio de tração ambiente é responsável por avaliar propriedades mecânicas no processo siderúrgico, que retorna resultados como: limite de resistência (LR), limite de escoamento (LE) e alongamento (ALO). O objetivo é construir modelos para cada propriedade, através das variáveis de processo e dados de composição química, para prever as respostas do ensaio auxiliando nas análises realizadas por especialistas. Foram ajustados nove diferentes algoritmos de regressão com coeficiente de determinação satisfatórios para os três indicadores. Os modelos resultantes contribuem para tomada de decisão do ensaio por obter ganhos no tempo de avaliação de amostras e, também, pela proposta de um método que identifica falsos positivos.*

1. Introdução

O ensaio de tração é um ensaio mecânico utilizado na metalurgia que permite conhecer as propriedades mecânicas de um material. A partir do conhecimento das propriedades mecânicas em diferentes condições de trabalho e a partir de análises de riscos e de falhas, é possível tomar decisões relativas à aplicação final do material.

A realização do ensaio consiste na aplicação de uma força uniaxial em um corpo de prova padronizado de forma a deformá-lo na direção desse esforço, até o rompimento

do corpo de prova. Desta forma, é possível obter a medida de propriedades importantes como, por exemplo, limite de resistência (LR), limite de escoamento (LE) e alongamento (ALO). Durante o processo, o material sofre dois tipos de deformação, a deformação elástica, que se inicia com a aplicação da tensão, e a deformação plástica, irreversível, que ocorre quando o material é submetido a uma carga que excede seu limite elástico. O limite de escoamento retorna a tensão máxima que o material suporta sem causar deformação plástica, ou seja, a deformação irreversível. O limite de resistência traz a informação de qual a tensão máxima suportada pelo material até que ele se rompa. O alongamento é a medida da deformação sofrida pelo material durante o ensaio até sua ruptura. Na Figura 1, F_0 é a área onde se espera que o rompimento ocorra, l_0 é a área útil utilizada para cálculo do alongamento, L é a largura da peça, para o ensaio de tração cilíndrico d_0 corresponde ao diâmetro da peça e para o ensaio de tração retangular, a corresponde a área útil da peça. [Kliauga and Ferrante 2009].



Figura 1. Corpos de prova de tração cilíndrica e retangular. Fonte: elaborado pelo autor.

Em [Bahrami et al. 2005], os autores utilizam dados de temperatura de deformação no processo e dados da fase de martensita do aço, onde a composição química exerce um efeito mais evidente. A identificação das fases do aço necessita de um ensaio de microestrutura que não foi realizado no objeto de estudo deste trabalho. Dentro do mesmo contexto, mas com objetivo diferente, em [Lin et al. 2010], os autores utilizam os resultados do ensaio de tração e dados do processo para estudar as taxas de densidade de fluxo no processo de laminação utilizando o modelo Johnson-Cook (ver [Johnson and Cook 1985]).

A utilização de regressores para a estimação das propriedades mecânicas de aços auxiliará na análise detalhada atuando como indicador complementar que permitirá a melhoria da acurácia do ensaio. A depender dos valores estimados pelo ensaio e das especificações do produto, a amostra pode ser classificada como adequada ou não. Os indicadores podem auxiliar na identificação de amostras classificadas erroneamente como adequadas, o falso positivo, amostras rejeitadas de forma equivocada, falso negativo, além de reforçar o verdadeiro positivo e o verdadeiro negativo, que correspondem respectivamente quando a amostra é classificada de forma correta como adequada e não adequada. Espera-se que os indicadores desenvolvidos diminuam o tempo de análise do engenheiro responsável pela análise detalhada.

No contexto estudado, a empresa utiliza relatórios dos resultados do ensaio de tração para auxiliar a tomada de decisão no que tange a aprovação de resultados situados nos limites da especificação do produto. É possível detectar, a partir da análise dos resultados, interferências indesejadas no processo produtivo. Atualmente o setor realiza, para todas

as linhas de atuação da empresa, cerca de 200 mil ensaios de tração em um ano. Desse montante, 5% dos registros necessitam de uma avaliação realizada por especialistas. Em cerca de 40% dos casos que são enviados para pendência é necessário uma análise minuciosa, no qual se gasta um tempo significativo. A análise detalhada visa a confirmar, ou não, o resultado do ensaio de tração. Além de demandar alocação de várias horas para esta atividade, há uma forte dependência experiência / conhecimento tácito dos especialistas nas análises realizadas e a necessidade de padronização dos critérios de decisão utilizados. Atualmente a faixa de aceitação dos clientes é cerca de 30%, então a garantia de um erro em 5% é adequado para a faixa imposta pelos clientes.

Segundo [de Souza 1989], a composição química de um aço deve ser compatível com a sua aplicação para garantir o desempenho de suas funções sem a ocorrência de rupturas, deformação excessiva oriundos de esforços mecânicos, oxidação ou corrosão em ambientes e/ou desgastes em ambiente abrasivos. Os autores estudam o impacto do carbono e da adição de outros elementos químicos na liga com o objetivo de alterar as propriedades mecânicas do material.

O presente trabalho tem como foco a aplicação de modelos de aprendizado de máquina para estimativa de resultados do ensaio de tração a partir das condições operacionais e da composição química dos materiais. Anteriormente, [Bahrami et al. 2005] e [Lin et al. 2010] utilizaram modelos de estimação dentro do mesmo processo. Porém, na pesquisa do primeiro autor, é utilizado dados das fazes martensitas do aço, o que não atenderia a aplicação atual, visto que não existem esses dados para análise. O segundo trabalho mencionado, além de utilizar um modelo físico para a análise, o modelo *Johnson-Cook*, o trabalho faz o inverso do que o trabalho proposto pretende.

Dentro do contexto apresentado, o objetivo do trabalho é construir indicadores de atenção para cada um dos resultados: LE, LR e ALO do ensaio de tração ambiente através de dados de processo, assim como é feito nas análises manuais. Como contribuição são apresentados modelos que, a partir dos dados de processos do produto e dos dados de composição química do material, é possível prever o valor dos indicadores de propriedades mecânicas e comparar os valores estimados com os resultados do ensaio de tração.

O trabalho a seguir está distribuído nas seguintes seções: Procedimentos Metodológicos, onde são apresentados as características do problema estudado e os procedimentos metodológicos iniciais; Métodos de Solução, para detalhar os modelos implementados; Resultados onde são apresentados os resultados obtidos da aplicação dos modelos para a solução do problema; e, por último, Conclusões, para elencar as descobertas realizadas através do presente estudo.

2. Procedimentos Metodológicos

A presente pesquisa foi conduzida a partir da metodologia de Descoberta de Conhecimento em Bases de Dados, em inglês *Knowledge Discovery in Databases - KDD*. De acordo com [Fayyad et al. 1996], o KDD tem a finalidade de avaliar e identificar padrões que façam sentido, que garanta o retorno de conhecimento útil e válidos em grandes bases de dados do mundo real.

O processo de KDD é composto pelas seguintes etapas: (i) seleção da base de dados; (ii) limpeza e pré-processamento; (iii) aplicação de transformações necessárias;

(iv) identificação, seleção e aplicação de métodos de mineração de dados; (v) avaliação dos resultados e (vi) consolidação do conhecimento gerado.

2.1. Coleta dos dados

Todos os dados usados no projeto foram extraídos de uma base armazenada em um banco de dados. A análise apresentada refere-se aos materiais provenientes do processo de laminação de tiras a quente a partir de um recorte temporal de 1 ano, no período de 05/2021 até 04/2022.

2.2. Tratamento dos dados

Na base estudada, a unidade de medida dos resultados do ensaio de tração precisou passar por conversão. Havia resultados nas unidades de medidas Mega Pascal (MPa), Newton por milímetro quadrado (N/mm²) e 1000 libras por polegadas ao quadrado (KSI), sendo assim, foi necessário converter para que todos os valores fossem para a unidade MPa.

Para evitar distorções na escala de valores na análise, como exemplo, o valor de espessura e o valor da largura, os dados foram padronizados obtendo a dispersão apresentada na Figura 2, essa padronização foi realizada através do algoritmo *preprocessing.StandardScaler()*, do pacote *Scikit-Learn*. No eixo x da imagem de padronização, foi feita a identificação das variáveis de acordo com a ordem apresentada na Tabela 1. As colunas 3 - temperatura de bobinamento e 21 - código especial de operação, apresentaram dados faltantes, e os valores nulos foram substituídos pela média da coluna.

Foram identificados *outliers* na base, mas optou-se por manter, pois ao longo das análises foi percebido que eles possuíam informações relevantes com potencial para geração de conhecimento aos modelos aplicados.

2.3. Separação da base de dados e validação dos resultados

Para a construção e validação dos modelos, a base foi dividida em dois *splits*, sendo 80% separado para o treinamento, totalizando 18.025 registros e 20% para os testes, 4.507 registros. Esse padrão foi seguido para a aplicação de todos os modelos que serão apresentados na próxima seção.

Para realizar a comparação dos resultados, foi realizado a *Cross validation* utilizando o pacote *Scikit-Learn*. No caso das propriedades LE e LR foram utilizados um total de 10 *splits* e na propriedade do ALO foram utilizados um total de 5 *splits*, porque para a propriedade ALO foi utilizada Redes Neurais Artificiais, o que será mencionado nos itens a seguir, e a execução desse algoritmo com 10 *splits* estava resultando em elevado consumo computacional. O parâmetro *shuffle* foi setado como verdadeiro para que os dados fossem embaralhados antes de serem divididos em grupos. Os demais parâmetros foram utilizados com valores padrão.

2.4. Variáveis

Como proposta de solução foram usados dados de processos, dados do ensaio de tração propriamente dito e dados de composição química do material. Foram coletadas as 28 variáveis descritas na Tabela 1 para a análise das 3 propriedades trabalhadas, LE, LR e ALO. A segunda coluna apresenta a descrição dessas variáveis e, na terceira, Modelos,

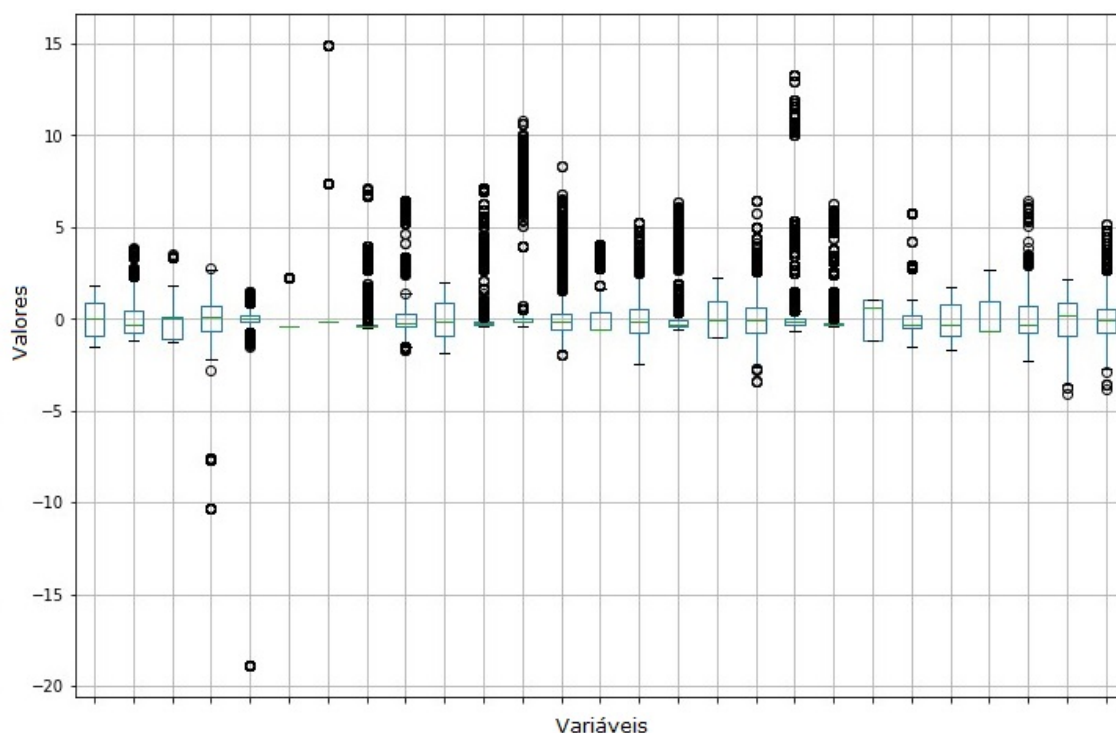


Figura 2. Desvio padrão e *outliers* após padronização da base. Fonte: elaborado pelo autor.

foi informado quais variáveis foram utilizadas nos modelos de cada uma das propriedades mecânicas.

As linhas 25, 26 e 27, marcadas com um *, referem-se às variáveis resposta. A variável de resultado do valor obtido de LR foi incluída como variável de entrada do ALO, bem como a variável com o resultado do cálculo do valor de LE dividido por LR. Essas variáveis foram inseridas na avaliação do alongamento após análise dos resultados obtidos visto que são informações que influenciam diretamente no valor de alongamento.

As variáveis 1 e 2 são referentes às dimensões do produto, a variável 24 é um agrupamento decorrente da variável espessura, as variáveis 3, 4 e 5 são referentes ao processo de laminação, as variáveis 6 e 7 indicam pendências de amostragem, as variáveis das linhas 8 até 20 são valores obtidos dos elementos químicos do material. As variáveis 21, 22 e 23 são de especificações/orientações para a operação da linha de laminações a quente e, por último, a variável 28 é um cálculo feito entre as variáveis resposta. Além das 28 variáveis que foram apresentadas na Tabela 1, utilizadas conforme as necessidades de cada modelo, apresentado na terceira coluna, todos os modelos foram implementados utilizando esta base, com 22532 registros.

3. Métodos de solução

Neste trabalho foram utilizados diferentes algoritmos na análise. Para realizar a estimativa, foi utilizada Regressão Linear, Árvores de Decisão e Redes Neurais Artificiais. Neste ponto, foi utilizado também, *ensemble learners* como o *Random Forest*, *Bagging*, *AdaBoost*, *Voting* e *Stacking*, como também não é escopo do presente artigo detalhar tais algoritmos, para esse propósito, é sugerido os textos [Alpaydin 2020], [Raschka 2015]

Tabela 1. Variáveis utilizadas

Ordem	Descrição	Modelos		
		LE	LR	ALO
1	Espessura	✓	✓	✓
2	Largura	✓	✓	✓
3	Temperatura de bobinamento	✓	✓	✓
4	Temperatura do acabamento	✓	✓	✓
5	Quantidade de vezes que passou pelo encruamento	✓	✓	✓
6	Se apresenta pendência por composição química	✓	✓	
7	Se apresenta pendência por alongamento			✓
8	Valor do cromo (Cr)	✓	✓	✓
9	Valor do carbono (C)	✓	✓	✓
10	Valor do manganês (Mn)	✓	✓	✓
11	Valor do molibdênio (Mo)	✓	✓	✓
12	Valor do boro (B)	✓	✓	✓
13	Valor do fósforo(P)	✓	✓	✓
14	Valor do titânio (Ti)	✓	✓	✓
15	Valor do nitrogênio (N)	✓	✓	✓
16	Valor do cobre (Cu)	✓	✓	✓
17	Valor do nióbio (Nb)	✓	✓	✓
18	Valor do alumínio (Al)	✓	✓	✓
19	Valor do vanádio (V)	✓	✓	✓
20	Valor do silício (Si)	✓	✓	✓
21	Código especial de operação	✓	✓	✓
22	Valor visado de carbono	✓	✓	✓
23	Valor visado de manganês	✓	✓	✓
24	Grupo de espessura	✓	✓	✓
25	Valor obtido de LE *	✓		
26	Valor obtido de LR *		✓	✓
27	Valor obtido de ALO *			✓
28	Valor obtido de LE/LR			✓

* Variáveis respostas.

e [Zhang and Ma 2012]. Para a implementação da maioria dos modelos, foi utilizado o pacote *Scikit-Learn* ([Pedregosa et al. 2011]). Em duas alternativas de estimativa da propriedade ALO foi utilizado o pacote *Keras* ([Chollet et al. 2015]).

3.1. Agrupamento de materiais

Há uma quantidade elevada de tipos de materiais provenientes do processo de laminação de tiras a quente, com diferentes aplicações finais, conseqüentemente isso pode resultar em variações nos valores das propriedades mecânicas. Por esse motivo, foi proposto inicialmente agrupar as observações com o intuito de melhorar o desempenho dos estimadores. As observações foram agrupadas de duas formas: (i) com base na espessura do material e (ii) por meio de clusterização. Adicionalmente, foram apresentados os resultados obtidos para o ajuste dos modelos sem agrupamentos.

Os produtos laminados a quente, presentes na base extraída para o estudo, possuem faixa de espessura entre 1,50 mm e 20,00 mm. Como primeira alternativa de agrupamento e com o auxílio de um especialista, foi realizada uma divisão em 3 grupos, o primeiro grupo de 1,50 a 4,99 mm, o segundo de 5,00 a 9,99 mm e o terceiro sendo o acima de 10,00 mm.

Como alternativa de agrupamento, as observações também foram clusterizadas utilizando o algoritmo do *Kmeans* por meio do pacote *SciKit-Learn*. Nessa etapa as va-

riáveis respostas foram excluídas da análise. A Figura 3 apresenta a variação média por *cluster* para a quantidade de grupos variando de 1 a 19. Com o auxílio do método de *Elbow*, definiu-se a utilização de 5 grupos.

3.2. Estimadores

Para cada propriedade mecânica estudada buscou-se ajustar diferentes algoritmos. Neste trabalho não foi utilizada nenhuma metodologia para calibração de parâmetros, a parametrização escolhida foi identificada de forma empírica a partir de alguns testes preliminares. Todos os parâmetros não mencionados nesta seção foram utilizados com valores padrões.

Para as três propriedades, utilizando o pacote SciKitLearn, foram implementados os seguintes modelos:

- **Regressão Linear** de acordo com as configurações predefinidas da classe *sklearn.linear_model.LinearRegression*.
- **Árvore de regressão** com profundidade máxima de 5 camadas por meio da classe *sklearn.tree.DecisionTreeRegressor*.

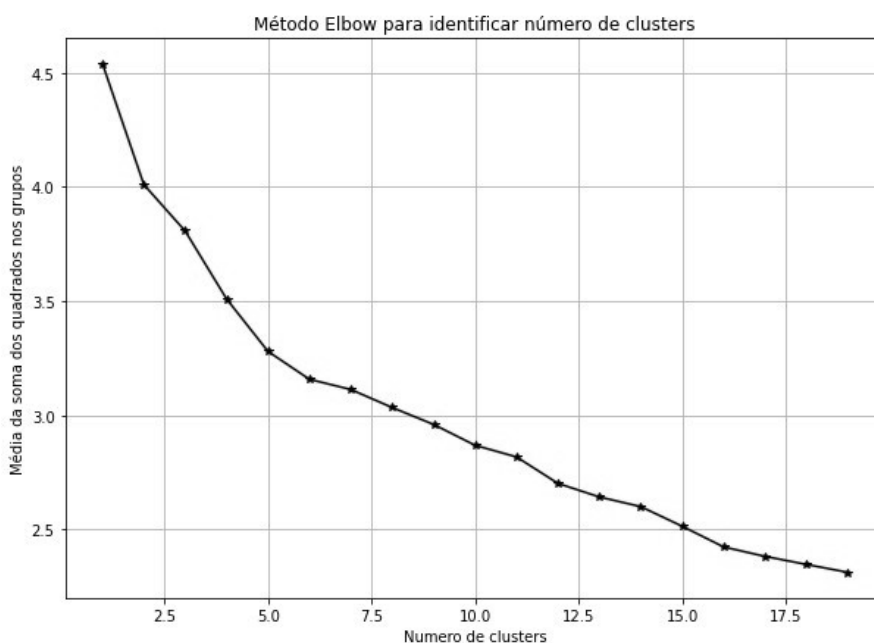


Figura 3. Gráfico de *Elbow*. Fonte: elaborado pelo autor.

Para o ALO, visto que os coeficientes de determinação foram os menores encontrados na regressão e na árvore de decisão, foram implementados também modelos de Redes Neurais Artificiais:

- Utilizando a classe *sklearn.neural_network.MLPRegressor* com a função de ativação *Relu*, uma camada, 100 neurônios e otimizador Adam.
- Utilizando a classe *tf.keras.Sequential* com duas camadas (32x1), com função de ativação linear e elu e otimizador Nadam.

3.2.1. Ensemble Learners

Para todas as propriedades mecânicas, foram testadas combinações de modelos utilizando *Ensemble Learners*. Foram aplicados 5 métodos para as estimativas de LE, LR e ALO: (i) *Random Forest*, (ii) *Ada Boost Regressor*, (iii) *Bagging Regressor*, (iv) *Voting Regressor* e (v) *Stacking Regressor*. Novamente, todos os métodos foram implementados utilizando classes do pacote SciKit-Learn.

Para a utilização do *Random Forest*, foram necessários ajustes na classe `sklearn.ensemble.RandomForestRegressor` configurada para executar o aprendizado considerando 250 estimadores. No ajuste do *Ada Boost Regressor* foi utilizada a classe `sklearn.ensemble.AdaBoostRegressor` com árvores de regressão com o estimador base e 300 estimadores. Para o método *Bagging Regressor*, a classe `sklearn.ensemble.BaggingRegressor` foi configurada com o estimador base sendo árvores de regressão e com o parâmetro *bootstrap features* verdadeiro, para que a amostragem dos componentes do modelo fosse realizada com reposição.

O *Voting Regressor* foi implementado por meio da classe: `sklearn.ensemble.VotingRegressor` utilizando os seguintes modelos regressão linear, árvore de regressão e o *Random Forest* para o LE e LR. No caso do ALO, além desses modelos, foi acrescentada a rede neural implementada pela classe `sklearn.neural_network.MLPRegressor`.

Por fim, o modelo *Stacking Regressor* foi implementado usando a classe `sklearn.ensemble.StackingRegressor` com os mesmos estimadores utilizados na implementação do *Voting Regressor* com a ressalva para este modelo é necessário configurar um estimador final, neste caso foi utilizado o *Random Forest*. O parâmetro *cross-validation* (cv) foi definido com o valor 10, isso significa que serão utilizadas 10 divisões durante a validação cruzada e parâmetro *passthrough* foi definido como verdadeiro para que os dados originais também fossem utilizados para o treinamento do estimador final.

4. Resultados

Considerando o uso prático do modelo, pode-se dizer que os resultados obtidos foram adequadas para a variância explicada de 95% para a estimativa das propriedades LE, LR e ALO aferidas pelo ensaio de tração. O desempenho dos estimadores para a característica ALO foi pior do que o desempenho obtido para as outras duas propriedades, no entanto, ainda assim são capazes de gerar ganhos para o processo. A divisão da base em *clusters* de observações gerou perda de desempenho em relação aos modelos ajustados na base sem agrupamento. Já a divisão por espessuras, embora tenha sido ligeiramente melhor, não apresentou diferenças estatísticas significativas para a aplicação sem agrupamentos. No entanto, a possibilidade de usar os melhores modelos para cada grupo de espessura, potencializa a viabilidade desta abordagem a partir dos resultados obtidos a seguir.

4.1. Limite de Escoamento

A Tabela 2 apresenta os resultados da aplicação de 7 modelos com uma divisão da base em 3 formatos, por *cluster*, por grupo de espessura e um terceiro grupo em que não foi aplicada qualquer tipo de divisão. Nesta tabela a sigla CD significa o coeficiente de determinação (R_2 -square), responsável por medir o quão bem um modelo de regressão se

ajusta aos dados observados, sendo assim, quanto maior o valor, mais ajustados estarão os resultados.

Tabela 2. Resultado dos modelos para LE - Coeficiente de determinação (em %)

Modelos	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Sem divisão	Esp. 1	Esp. 2	Esp. 3
	CD	CD	CD	CD	CD	CD	CD	CD	CD
Regressão	87,0	86,0	83,0	80,0	76,0	94,0	95,0	96,0	90,0
Árvore	87,8	87,4	80,8	82,1	86,4	94,0	95,9	97,1	94,2
Randon Forest	94,9	92,8	88,2	84,4	89,0	97,6	97,9	98,2	96,4
AdaBoost [◇]	44,3	84,3	80,0	76,0	74,3	92,9	94,1	91,7	84,3
Bagging [◇]	86,0	84,8	81,3	80,3	76,5	93,3	94,2	95,3	89,8
AdaBoost [♣]	92,5	90,5	85,1	81,9	87,7	95,9	96,8	97,4	95,5
Bagging [♣]	90,7	89,0	82,7	84,4	86,9	95,1	96,4	97,3	94,4
Voting[◇]	92,9	91,3	87,7	83,3	85,8	96,5	97,0	97,9	95,3
Stacking[♣]	94,6	92,5	88,7	83,1	89,1	97,6	97,8	98,4	95,7

◇ Executado para a regressão. ♣ Executado para a árvore de decisão. ◇ Executado para regressão, árvore de decisão e *random forest*.
♣ Executado para regressão e árvore, sendo o estimador final *random forest*.

A validação cruzada foi aplicada e confirmou os bons resultados encontrados ao executar os modelos. Os resultados da validação cruzada podem ser visualizados na Tabela 3. Aqui o STD ou *standart* é o desvio padrão, utilizado para explicar a variação e a dispersão dos dados.

Tabela 3. Cross validation para os modelos para LE - Coeficiente de determinação (em %)

Modelos	Cluster 0		Cluster 1		Cluster 2		Cluster 3		Cluster 4		Sem Cluster		Esp. 1		Esp. 2		Esp. 3	
	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD
Regressão	87,1	1,6	85,4	0,8	83,0	1,8	77,8	5,8	74,0	9,0	74,5	4,6	94,6	0,4	95,3	0,5	88,6	5,1
Árvore de Decisão	88,2	1,6	97,2	1,0	80,6	1,2	80,6	4,0	84,9	1,8	85,4	2,9	95,1	0,3	96,7	0,4	92,3	2,7
Randon Forest	94,2	1,5	92,4	0,6	87,7	1,6	83,5	3,2	89,6	2,0	89,7	1,6	97,7	0,1	98,0	0,3	95,2	1,5
Voting	92,4	1,7	90,8	0,3	86,4	1,4	84,0	3,1	87,0	1,8	86,0	1,4	96,9	0,2	97,5	0,4	94,2	1,5
Stacking	93,6	1,3	91,8	0,7	87,6	1,6	84,2	2,2	87,1	1,9	87,0	3,9	97,6	0,1	97,9	0,3	95,1	1,8

Nos resultados da Tabela 2, foi verificado que os melhores modelos ficam com o *random forest* e o *stacking*, revezando entre o primeiro e segundo lugar. Outro modelo que teve um resultado bastante satisfatório foi o *voting*, que apresentou resultados próximos do melhor modelo. Observa-se que os métodos *stacking* e *voting* internamente trabalham com o modelo do *random forest*, que obteve o melhor resultado em muitos casos, os resultados foram muito aproximados para todas as divisões propostas na Tabela 2, por *clusterização*, sem divisões e a divisão por espessura. Porém, quando avaliada a Tabela 3, observa-se que os melhores resultados referem-se a divisão por espessura, no entanto, sem apresentar diferença estatística significativa.

4.2. Limite de Resistência

O LR, de todos os itens estudados, obteve o melhor resultado nos modelos aplicados, com um percentual de 98,7% de coeficiente de determinação. Assim como o LE, este parâmetro foi avaliado em 7 modelos diferentes, apresentados na Tabela 4.

Na análise de cada um dos modelos aplicados foi observado que na maioria dos casos, o *stacking* foi o método que apresentou os melhores resultados, sendo que a aplicação na base sem qualquer tipo de divisão obteve o melhor resultado. Outro modelo que se destacou bastante na aplicação à esta base foi o *random forest*, que obteve resultados muito próximos do modelo anterior. O modelo *voting* também obteve resultados bastante

Tabela 4. Resultado dos modelos para LR - Coeficiente de determinação (em %)

Modelos	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Sem divisão	Esp. 1	Esp. 2	Esp. 3
	CD	CD	CD	CD	CD	CD	CD	CD	CD
Regressão	93,0	93,0	96,0	85,0	80,0	97,0	96,0	97,0	94,0
Árvore	84,3	91,7	94,4	84,4	89,0	94,4	92,5	96,1	93,8
Random Forest	96,0	96,2	96,8	86,1	91,9	98,7	98,5	98,7	97,9
AdaBoost [◇]	90,6	92,8	95,8	73,7	77,3	93,1	95,3	86,8	93,5
Bagging [◇]	91,2	92,4	95,4	84,3	80,0	96,0	94,9	96,2	93,8
AdaBoost[♣]	94,0	94,8	95,6	85,0	90,0	97,2	97,2	98,2	97,5
Bagging [♣]	89,9	93,9	94,7	86,6	89,4	96,7	95,4	97,6	96,2
Voting [♡]	94,3	95,3	96,4	88,0	91,0	98,0	97,4	98,2	96,8
Stacking[♣]	96,0	96,3	96,8	86,2	92,1	98,7	98,5	98,9	98,1

◇ Executado para a regressão. ♣ Executado para a árvore de decisão. ♡ Executado para regressão, árvore de decisão e *random forest*.
 ♣ Executado para regressão e árvore, sendo o estimador final *random forest*.

satisfatórios. Observa-se que esses modelos que tiveram bons resultados para o LE. Como mencionado anteriormente, os métodos *voting* e *stacking* se baseiam no modelo *random forest*.

Tabela 5. Cross validation para os modelos para LR - Coeficiente de determinação (em %)

Modelos	Cluster 0		Cluster 1		Cluster 2		Cluster 3		Cluster 4		Sem Cluster		Esp. 1		Esp. 2		Esp. 3	
	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD
Regressão	91,0	3,2	93,5	0,3	96,5	0,3	82,0	6,3	83,4	2,6	83,1	2,6	96,3	0,3	96,9	0,4	95,3	1,1
Árvore de Decisão	81,7	2,7	91,9	0,5	94,8	0,8	78,3	6,7	92,0	1,3	92,2	1,9	92,7	0,6	96,3	0,6	93,3	2,1
Random Forest	94,9	1,9	96,5	0,1	97,2	0,5	83,0	5,8	94,4	1,6	94,2	1,7	98,4	0,1	98,6	0,5	97,7	0,8
Voting	93,0	2,0	95,4	0,4	96,9	0,3	84,1	8,7	95,2	1,2	92,9	1,7	97,5	0,1	98,2	0,4	96,9	0,8
Stacking	94,7	1,9	96,5	0,2	97,3	0,4	82,7	7,5	93,8	1,7	93,3	1,8	98,5	0,1	98,7	0,4	98,0	0,4

4.3. Alongamento

Assim como as outras duas propriedades, LE e LR, o ALO foi avaliado com regressão, árvore de decisão, *Random Forest*, *Ada Boost*, *Bagging*, *Voting* e *Stacking*, também foram aplicados modelos de redes neurais artificiais utilizando o pacote *Keras* e o pacote *Scikit-Learn*, conforme resultados apresentados na Tabela 6.

Tabela 6. Resultados dos modelos para ALO - Coeficiente de determinação (em %)

Modelos	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Sem divisão	Esp. 1	Esp. 2	Esp. 3
	CD	CD	CD	CD	CD	CD	CD	CD	CD
Regressão	42,0	69,0	44,0	63,0	50,0	64,0	67,0	64,0	81,0
Árvore	51,2	65,3	65,5	64,1	46,3	80,3	79,4	70,4	80,9
Random Forest	55,3	77,6	75,3	82,6	54,1	87,2	88,9	84,0	88,8
RN Keras (Linear)	40,8	69,3	43,6	62,4	47,4	64,1	67,0	63,7	77,9
RN Keras (Elu)	13,8	75,2	65,7	80,3	51,0	82,6	85,2	78,8	85,6
MLP (Relu)	55,8	76,4	69,3	80,0	50,9	85,3	86,9	80,8	86,8
AdaBoost [◇]	42,4	64,9	15,8	53,9	41,3	61,4	65,5	54,8	51,9
Bagging [◇]	40,3	67,6	42,8	60,1	53,6	62,6	65,4	60,4	76,5
AdaBoost [♣]	52,5	69,7	61,1	78,3	50,6	81,5	82,1	77,9	87,3
Bagging [♣]	54,2	70,5	66,3	68,5	53,7	80,9	82,5	72,6	85,7
Voting [♡]	57,4	76,0	72,4	79,1	55,4	84,8	85,8	80,7	87,9
Stacking[♣]	58,4	77,0	73,1	81,4	51,6	86,8	88,4	82,8	88,8

◇ Executado para a regressão. ♣ Executado para a árvore de decisão. ♡ Executado para regressão, árvore de decisão, MLP e *random forest*.
 ♣ Executado para regressão, árvore e MLP, sendo o estimador final *random forest*.

Na tabela de resultados do ALO observa-se que os melhores resultados estão com o *random forest*, sendo que a base sem divisão e com divisão por grupo de espessura

tiveram os melhores resultados, cerca de 87,2%. O segundo melhor modelo foi o *stacking*, que já foi apresentado, também trabalha com o *random forest* na sua construção. O terceiro melhor resultado foi com a rede neural multicamadas MLP, sigla de *Multilayer Perceptron*.

Tabela 7. Cross validation dos modelos para ALO - Coeficiente de determinação (em %)

Modelos	Cluster 0		Cluster 1		Cluster 2		Cluster 3		Cluster 4		Sem Cluster		Esp. 1		Esp. 2		Esp. 3	
	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD	CD	STD
Regressão	44,5	7,1	68,2	1,3	42,7	1,8	65,2	1,3	52,8	9,0	53,2	4,3	65,6	1,1	64,0	2,0	74,8	2,5
Árvore de decisão	57,0	3,1	65,8	1,6	62,7	2,4	68,5	4,3	50,1	2,4	48,8	9,2	78,5	0,7	70,0	2,2	75,5	5,5
Random Forest	61,6	7,8	77,7	0,8	74,0	2,3	84,0	0,6	57,4	6,0	57,1	5,7	87,8	0,4	84,8	1,1	86,0	2,5
Voting	63,7	3,9	75,8	1,4	70,3	2,6	80,5	1,1	61,0	1,6	60,1	3,2	84,8	0,5	81,4	1,0	82,1	1,8
Stacking	63,5	5,9	76,5	1,0	72,2	1,5	83,3	0,3	57,8	5,3	57,4	3,8	87,1	0,4	84,4	1,9	84,0	1,2
MPL Relu	61,6	2,8	75,6	0,3	67,6	2,4	81,0	1,1	59,1	5,0	57,3	7,3	85,8	0,4	82,1	1,0	80,4	1,8

Na validação cruzada do parâmetro ALO, observa-se que os modelos com algum tipo de divisão ficaram próximos dos resultados dos modelos executados durante o treinamento e os testes iniciais. Sendo assim, o modelo *random forest* e *stacking* tiveram os melhores resultados, quando dividido por grupo de espessura, no entanto, devido ao desvio padrão, observa-se que as abordagens são estatisticamente equivalentes.

5. Conclusões

Neste trabalho foi desenvolvido modelos de aprendizado de máquinas capazes de estimar propriedades mecânicas de aços a partir de dados do processo de fabricação e da composição química dos mesmos. Os modelos poderão auxiliar a realização de análise dos resultados do ensaio de tração com o fornecimento de indicadores para as propriedades de tração - LE, LR e ALO.

Os modelos foram desenvolvidos para os produtos provenientes do processo de laminação de tiras a quente a partir de 22532 registros e os resultados obtidos estão acima de 95% de acurácia esperada para as propriedades LE e LR e acima de 85% de acurácia esperada para a propriedade ALO. Devido ao volume de ensaios realizados e da necessidade de realizar uma análise minuciosa em 5% dos resultados apontados como não conformes, estima-se que o uso dos modelos desenvolvidos no presente trabalho poderá reduzir cerca de 500 horas anuais de análise dos especialistas.

Além disso, o uso do modelo visa auxiliar na identificação de materiais que estão dentro do especificado e que, por alguma falha no ensaio, o valor não seja o verdadeiro, isto é, os falsos positivos. Nesse sentido, os modelos desenvolvidos auxiliarão no controle de qualidade dos produtos, diminuindo a geração de reclamações oficiais dos clientes.

O trabalho foi avaliado e construído para os produtos tiras a quente, porém existe grandes possibilidades de aplicação aos outros produtos da empresa, chapas grossas, tiras a frio e produtos galvanizados. Espera-se com este resultado proporcionar para a empresa a geração de bons modelos para a avaliação de todos os corpos de prova do ensaio de tração. Outra aplicação com potencial de ganho para empresa, é a avaliação de outros ensaios de teste mecânico, tais como ensaio de dureza e charpy.

Por fim, é importante que a aplicação dos modelos desenvolvidos seja acompanhada por especialistas e aprimorados a ponto de, com o tempo, ser possível utilizá-los

no processo de tomada de decisão para definir as variáveis controladas do processo de produção, com o objetivo de obter produtos de maior qualidade e menor custo.

Referências

- Alpaydin, E. (2020). *Introduction to machine learning*. MIT press.
- Bahrami, A., Anijdan, S. M., and Ekrami, A. (2005). Prediction of mechanical properties of dp steels using neural network model. *Journal of alloys and compounds*, 392(1-2):177–182.
- Chollet, F. et al. (2015). Keras. <https://keras.io>.
- de Souza, S. A. (1989). *Composição química dos aços*. Editora Blucher.
- Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., et al. (1996). Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of KDD-96*, pages 82–88.
- Johnson, G. R. and Cook, W. H. (1985). Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures. *Engineering fracture mechanics*, 21(1):31–48.
- Kliauga, A. M. and Ferrante, M. (2009). *Metalurgia básica para ourives e designers: do metal à joia*. Editora Blucher.
- Lin, Y., Chen, X.-M., and Liu, G. (2010). A modified johnson–cook model for tensile behaviors of typical high-strength alloy steel. *Materials Science and Engineering: A*, 527(26):6980–6986.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Raschka, S. (2015). *Python machine learning*. Packt publishing ltd.
- Zhang, C. and Ma, Y. (2012). *Ensemble machine learning: methods and applications*. Springer.