APLICAÇÃO DE ALGORÍTMOS DO *AUTOMATED MACHINE LEARNING* EM UM CENÁRIO RELEVANTE

Application of Automated Machine Learning algorithms in a relevant scenario

Pedro Henrique Bunn Schmitt [1] Gregory Chagas da Costa Gomes [2]

[1] Acadêmico do Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina – IFSC.
[2] Docente do Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina – IFSC.

Resumo

O conceito de inteligência artificial consiste no desenvolvimento de sistema computadorizados que realizam tarefas análogas a inteligência humana. Oficializado em 1959, o termo toma relevância no mundo da tecnologia e dele surgem inúmeras ramificações, entre elas a do Machine Learning e posteriormente o Automated Machine Learning. Este trabalho teve como objetivo experienciar a utilização do AutoML (Automated Machine Learning) em uma aplicação de predição da temperatura de rotores de um conjunto de dados de motores elétricos. Apresentou-se os conceitos de inteligência artificial e suas ramificações, os de modelos de aprendizado do Machine Learning, quais sejam, Modelo de Aprendizado Supervisionado, Não Supervisionado e Reforçado. Foram vistos artigos que traziam comparações entre métodos tradicionais como Deep Learning e Machine Learning, em comparação ao presente método de estudo nas áreas de saúde e meio ambiente. Após a apresentação do tema e uma pequena contextualização sobre o campo de atuação da área, foi demonstrado qual o caminho percorrido até o encontro do modelo para predição da temperatura do rotor, evidenciando de modo gradual os programas, bibliotecas e códigos utilizados. Por fim, foi feita apresentação e comparação dos resultados do algoritmo encontrado em relação aos resultados disponíveis na plataforma Kaggle. O objetivo foi apresentar como foi realizado a criação de modelos de Automated Machine Learning e a aplicação do mesmo para um conjunto de dados foi plenamente atingido através da produção e publicação dos Notebooks e este foi alcançado. Nesta pesquisa foi encontrado um algoritmo de Machine Learning para o conjunto de dados de motores elétricos capaz de fazer predições da temperatura dos rotores apresentando assim resultados

promissores. No entanto, entende-se necessárias outras pesquisas buscando adquirir conhecimento e aprofundamento no assunto.

Palavras-Chave: Machine Learning. Automated Machine Learning. Predição de Temperatura.

Abstract

The concept of artificial intelligence consists in the development of computerized systems that perform tasks analogous to human intelligence. Officialized in 1959, the term takes on relevance in the world of technology and from it arise numerous ramifications, including Machine Learning and later Automated Machine Learning. This work aimed to experience the use of AutoML (Au-tomated Machine Learning) in an application to predict the rotor temperature of a dataset of electric motors. The concepts of artificial intelligence and its ramifications, the Machine Le-arning learning models were presented, namely, Supervised, Unsupervised and Reinforced Learning Model. Articles were seen that brought comparisons between traditional methods such as Deep Learning and Machine Learning, compared to the present method of study in the areas of health and the environment. After the presentation of the theme and a little contextualization about the field of action of the area, it was demonstrated the path taken to find the model for predicting the temperature of the rotor, gradually showing the programs, libraries and codes used. Finally, the results of the algorithm found were presented and compared with the results available on the Kaggle platform. The objective was to present how the creation of Automated Machine Learning models was carried out and its application to a dataset was fully achieved through the production and publication of Notebooks and this was achieved. However, it is understood that further discussions and more research time would be needed that go beyond this work to acquire greater knowledge on the subject and reliability in the results found.

Keywords: Machine Learning. Automated Machine Learning. Temperature Prediction.

1 INTRODUÇÃO

Com o conhecimento adquirido nas últimas décadas, nosso planeta produziu uma grande quantidade de dados e informações em que o computador, seus softwares e hardwares foram imprescindíveis para o aprofundamento deste conhecimento. Desta forma se apresentou necessária a criação de ferramentas computacionais mais sofisticadas e autônomas que conseguissem gerir e utilizar dados. As máquinas começaram a aprender com suas experiências passadas, para gerar soluções de problemas, análise de comportamentos de consumidores, predição de valores dos mais diversos campos a partir de grupos de dados entre tantos outros casos de aplicação. Com esta proposta surge o campo do Machine Learning, também conhecido como aprendizado de máquina (FACELI, 2011; PETROVA, 2022). O Automated Machine Learning - AutoML, atua neste campo facilitando a parametrização dos modelos de Deep Learning a fim de não ser mais necessário a expertise de um usuário para “configuração do mesmo” (HE; ZHAO; CHU, 2021).

O tema abordado neste trabalho possui uma gama de aplicações em diversas áreas, sendo apresentado como um grande ajudante para predições e estudos de caso de aplicações, como pode ser visto no trabalho feito sobre análise de água no rio Korattur, em Chennai na Índia. Neste estudo os autores compararam os 12 resultados entre dois algoritmos, um criado em Automated Machine Learning, que teve por objetivo automatizar a criação de algoritmos de Deep Learning fazendo com que não fosse necessária a assistência humana na escolha de parâmetros para o modelo e o outro modelo em Machine Learning. A finalidade deste trabalho citado apresentou o quanto de praticidade e segurança ocorre com a utilização de um modelo de AutoML para uma aplicação real e que a acurácia dos resultados foi maior quando utilizado as ferramentas de autoML em relação aos métodos tradicionais de Machine Learning (VENKATA et al., 2021).

O Curso de Engenharia Mecatrônica do Instituto Federal de Santa Catarina tem em seu currículo diferentes disciplinas com conteúdos envolvendo engenharia da informática, engenharia da eletricidade, engenharia de controle e

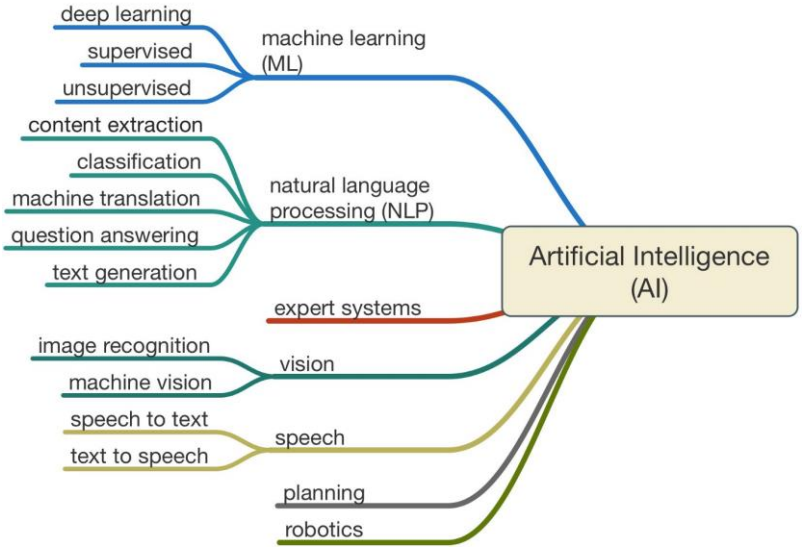
engenharia mecânica assim estabelecendo “os quatro pilares dos sistemas de automação atuais: sistemas mecânicos, eletrônica, controle e informática”. (SANTA CATARINA, 2012).

Durante o curso, disciplinas trouxeram conteúdos de sistemas de informática com foco na programação, os quais chamaram especial atenção do pesquisador, levando ao aceite da proposta realizada pelo orientador para a execução de um trabalho utilizando AutoML, sendo escolhido um grupo de dados de motores elétricos, onde seria predito qual a temperatura dos rotores. Desta forma este trabalho junta disciplinas vistas no curso, o desafio de explorar o campo da ciência de dados e o exercício da engenharia mecatrônica.

1.1 Inteligência Artificial e suas ramificações

A Inteligência Artificial se mostra um assunto amplamente discutido no campo acadêmico e de dados, fato visto ao ser realizada uma pesquisa pelo termo “*Artificial Intelligence"* em um banco de dados de publicações acadêmicas, no caso, o “Science Direct”, encontramos documentos e publicações demostrando que o campo de estudo de inteligência artificial não é tão novo quanto se pode pensar.

Figura 1 – Áreas de atuação da Inteligência Artificial



Fonte: Kumar (2018).

É possível observar pela Figura 1 que existem sete ramificações de

categorias que partem do bloco da Inteligência Artificial, porém as de maior destaque, informadas pelo autor são a *NLP, Vision, Robotics e Machine Learning (*KUMAR*,* 2018*).*

A *NLP* tem a função de trazer ao computador a habilidade de entender a linguagem falada e escrita pelos humanos, a partir da combinação da linguagem computacional com estatística, *Machine Learning* e *Deep Learning* (IBM, 2020a). O *Vision* traz a visão computacional, a qual é apresentada como uma das ramificações do campo da IA que permite, a partir de imagens, vídeos e outras entradas visuais, tomar ações e fazer recomendações baseadas nessas informações. Existe uma definição interessante colocada pela IBM onde apresenta que a IA faz com que o computador pense e a visão computacional faz com que ele possa ver, entender e observar (IBM, 2022). A categoria *Robotics* é apresentada como o campo da robótica, em que o foco é dado na criação de robôs que executam tarefas difíceis, perigosas ou repetitivas para os humanos como trabalho em hospitais, limpeza, agricultura entre outros.

Após ser feita uma breve revisão das ramificações que surgem a partir da inteligência artificial, é feito um breve resumo da área do *Machine Learning* e posteriormente da área do *Auto Machine learning* por ser o objeto de estudo deste trabalho.

1.1.1 Machine Learning

Machine Learning-ML pode ser definido como o estudo da forma na qual os agentes computacionais podem realizar ações. Entende-se por agentes computacionais aqueles que por meio do ambiente através de seus sensores aperfeiçoam suas percepções e seus conhecimentos para tomarem decisões baseados em experiências e dados (RUSSEL; NORVIG, 2003).

A área do *Machine Learning* é uma das que mais cresce no campo das ciências da computação pois, tanto a quantidade de dados quanto a forma como eles são processados e transformados em conhecimento, aumenta de forma exponencial ao longo dos anos. Outro ponto importante é ressaltar a quantidade de tipos de dados que os algoritmos da área começaram a conseguir tratar ao longo do avanço da tecnologia, como: páginas *web*, arquivos de áudio, vídeos, gráficos entre tantos outros (ALPAYDIN, 2014). Com estas ponderações feitas, qual a definição de *Machine Learning*?

De acordo com *Samuel (1959) apud Géron* (2017, p. 20) "*Machine Learning* é a arte de dar ao computador a habilidade de aprender a programar sem ser necessário explicitamente programar em linhas de código" ou por outra explicação mais técnica dada por Alpaydin *(*2014)*,* ML utiliza a estatística para a criação de modelos matemáticos, os quais por meio de um algoritmo, aprendem com dados de treino ou experiências passadas e podem fazer predições, sendo estes modelos preditivos, ou que possuem o papel de "ganhar" conhecimento e descrever os dados passados para ele, sendo estes modelos descritivos, ou ainda pode atuar como uma junção destes dois tipos (ALPAYDIN, 2014). Então, pelo que pode ser visto, o método descrito é utilizado com a finalidade de fazer a máquina "aprender" por conta própria através de um algoritmo. Porém, existe algum benefício nisso?

Para entender a utilização do ML, podemos usar como exemplo do controle de spam de um e-mail. Uma mensagem de correio eletrônico necessita passar por inúmeros filtros de palavras e endereços eletrônicos até que seja direcionada para a caixa de spam ou para a lixeira. Com isto em mente, a ideia de criar regras manuais de programação para direcionar cada nova mensagem se torna inviável e pouco preciso, visto as inúmeras especificações de palavras chaves que precisam ser criadas diariamente. Com o uso do ML isto não é necessário. A partir da criação de um modelo preciso e bem organizado novas palavras seriam adicionadas automaticamente e o controle de spam possuiria mais acurácia e ainda uma baixa necessidade de manutenção.

1.2 Automated Machine Learning

O *Automated Machine Learning,* também chamado de AutoML apresenta como objetivo automatizar partes do processo da criação do modelo do *Machine Learning* trazendo facilidade no momento da execução de tarefas repetitivas, como pré processamento das *features,* seleção de modelos, melhora dos *hyperparameters,* mais rapidamente para que o foco não seja mais nestes pontos, mas na análise dos resultados e interpretação dos mesmos (VENKATA *et al*., 2021).

Existem níveis de automação do AutoML separados em categorias de encontro de estimadores e preditores, encontro de algoritmos e encontro de

algoritmos de *meta* aprendizado.

A primeira categoria se refere a tarefa de mapear a partir das entradas das variáveis os resultados das saídas, como exemplos podem ser dados a escolha do valor dos parâmetros de um algoritmo de regressão linear para uma tarefa específica ou um código de classificação de baixo nível, no qual se apresenta como necessário especificar cada uma das condições de entrada de um conjunto de valores de variáveis e, a partir deste conjunto de entrada gerar regras de saídas do conjunto, os resultados. Neste caso a ferramenta de AutoML atuaria nas escolhas dos parâmetros para o primeiro caso e no segundo caso, para o entendimento das condições de entrada e as ações tomadas a partir destas condições.

Para a segunda categoria apresentada, o papel da ferramenta do AutoML é dada pelo encontro dos algoritmos para o dado problema, escolha dos hiper parâmetros, análise e pré-processamento dos dados. Como terceira categoria é colocado o encontro e utilização de algoritmos de meta aprendizado, os quais possuem como objetivo melhorar os resultados de modelos existentes comparando os resultados encontrados por eles e sugerindo mudanças nos hiper-parâmetros ou recomendando o uso de outros algoritmos de aprendizado. (ESCALANTE, 2020)

As bibliotecas ou métodos destacados pelo autor para a segunda categoria que possui como objetivo o encontro de algoritmos de forma automatizada são *PSMS, Heterogeneous surrogate Evolution, Ensemble PSMS, TPOT e GPS: GAPSO-FMS.* Para a terceira categoria utilizando algoritmos de meta aprendizado são apresentadas pelo as bibliotecas: *Auto-WEKA, Multiobjective surrogatebased FMS, AutoSkLearn e Neural Architecture Search* (ESCALANTE, 2020).

É colocado pelo autor alguns dos desafios que o campo do AutoML apresenta como a não transparência de como os modelos funcionam, a parte de pré processamento dos dados que é importante, porém ainda pouco explorada pela área, dificuldade para lidar com conjunto de dados muito extensos e reprodutibilidade dos resultados encontrados utilizando do método do AutoML. Porém mesmo com estes apontamentos Escalante (2020) traz que com o progresso na primeira década de utilização das ferramentas do AutoML foi muito interessante e que muito pode se esperar dos próximos anos desta área de

estudo. (ESCALANTE, 2020).

Com as ideias de inteligência artificial, *Machine Learning* e *Auto Machine Learning* surge o objetivo deste trabalho de aplicar e documentar o processo de utilização de algoritmos de *Automated Machine Learning*-AutoML em um cenário relevante.

2 METODOLOGIA

A partir desta escolha surgiu a concepção de um trabalho mais didático com objetivo de trazer resultados sobre um conjunto de dados de motores elétrico e, a partir dele a predição da temperatura dos rotores utilizando das bibliotecas *Auto-Sklearn, TPOT e H2O* do campo do AutoML, adentrando no mundo do *Machine Learning,* introduzindo os conceitos, os programas, estudos da utilização do AutoML em outras áreas do conhecimento, a apresentação das ferramentas e a comparação entre os resultados encontrados com os disponíveis na plataforma Kaggle.

Iniciou-se o trabalho da busca de conhecimentos por meio das mais diversas ferramentas, sites e artigos, a fim de entender um pouco mais sobre o mundo do *Data Science.* Os primeiros passos se deram para entender e se familiarizar com a linguagem *Python.*

Segundo Sebesta (2010) a profundidade da capacidade das pessoas de pensar é influenciada pelo poder da expressividade da linguagem que elas utilizam. A partir desta colocação, é possível entender porque foi esta a linguagem de programação escolhida para a pesquisa. Ela apresenta uma escrita de fácil entendimento, com diversas bibliotecas para visualização e análise de dados além de uma proporcionar uma rápida interação com os dados através dos terminais, uma tarefa necessária quando o assunto é *Machine Learning* (MÜLLER; GUIDO, 2017). Também foi necessário aprender sobre a biblioteca *pandas,* amplamente utilizada no tratamento e organização de dados e no campo do *Machine Learning,* entendendo melhor sobre seus algoritmos, classificações e métodos. Optou-se por fazer os cursos disponíveis na plataforma do *Kaggle,* site que foi de extrema importância na aquisição de conhecimento sobre o campo de estudo.

Feitos os cursos da plataforma, também foi necessário buscar como se

dava o funcionamento das bibliotecas de *Automated Machine Learning. A*s bibliotecas que foram escolhidas para este estudo foram a *Auto-Sklearn,* TPOT e H2O, escolhidas por possuírem uma boa documentação, serem bem comentadas e apresentarem avaliações positivas pelos usuários do *GitHub* (BALAJI; ALLEN, 2018). Para experimentação destas bibliotecas utilizou-se o ambiente *Jupyter Notebook amplamente* empregado no campo do *Machine Learning,* sendo uma ótima ferramenta para análise exploratória de dados (MÜLLER; GUIDO, 2017).

Para utilização da biblioteca *Auto-Sklearn,* foi necessário um sistema operacional baseado em Linux, demanda suprida de duas formas, a primeira sendo através de uma máquina virtual executada pelo software *Oracle VM VirtualBox* e posteriormente sendo feita a migração para *Google Colab,* um produto desenvolvido pelo *Google Research* que permite a escrita e execução de códigos em Python principalmente utilizado para o campo do *Machine Learning,* pela necessidade de mais poder de processamento. *(*GOOGLE, *2022).*

Com as plataformas escolhidas e apresentadas, o próximo passo foi encontrar um *dataset* relevante no contexto da Engenharia Mecatrônica sendo escolhido um grupo de dados sobre motores elétricos, na sequência preparar os dados para o treinamento do modelo a partir da biblioteca Pandas. Escolheu-se então o grupo de dados hospedado na plataforma *Kaggle* sobre a temperatura de um motor elétrico disponibilizado pela universidade de Paderborn (BÖCKER, 2021), sendo que o Quadro 1 demonstra as variáveis dos grupos de dados.

Quadro 1 - Variáveis do grupo de dados

Variáveis	Funções	Unidade
u_q	Tensão do componente	V
Coolant	Temperatura do refrigerador	ºC
Stator_winding	Temperatura do bobina do estator	ºC
u_d	Tensão do componente	V
Stator_tooth	Temperatura dos dentes do estator	ºC
Motor Speed	Velocidade do motor	RPM
i_d	Corrente do componente	I
i_q	Corrente do componente	I
pm	Temperatura do rotor	ºC
Stator Yoke	Temperatura da coroa do estator	ºC

Fonte: Elaboração própria (2022).

É interessante ressaltar que as variáveis i_d, i_q, u_d e u_q são

resultantes da estratégia de controle do motor que tem referência com a velocidade e o torque. Böcker (2021) não traz mais informações sobre estas variáveis, porém para o presente estudo foi necessário apenas conhecer seus valores, sem trazer interferências ao estudo.

Para treinamento dos modelos, foi escolhida a variável *"pm",* temperatura do rotor, como a variável "*target"* e retirada a variável "*profile_id",* a qual se refere a duração das sessões de testes com o motor pois no estudo não foi levado em consideração os tempos de cada um dos testes*.* A variável "*pm"* foi escolhida como valor alvo a se predizer pois economicamente para uma indústria automotiva, conhecer este valor traz a possibilidade de fabricar motores elétricos com menos material e utilizar da capacidade máxima do equipamento por saber quais são os comportamentos do mesmo (BÖCKER, 2021).

Além disso, fez-se o embaralhamento dos dados da tabela para que todos os valores das classes fossem bem distribuídos quando separados em treino e teste (MÜLLER; GUIDO, 2017). Neste estudo, os dados foram separados em 75 % em treino e o restante de 25 % em teste.

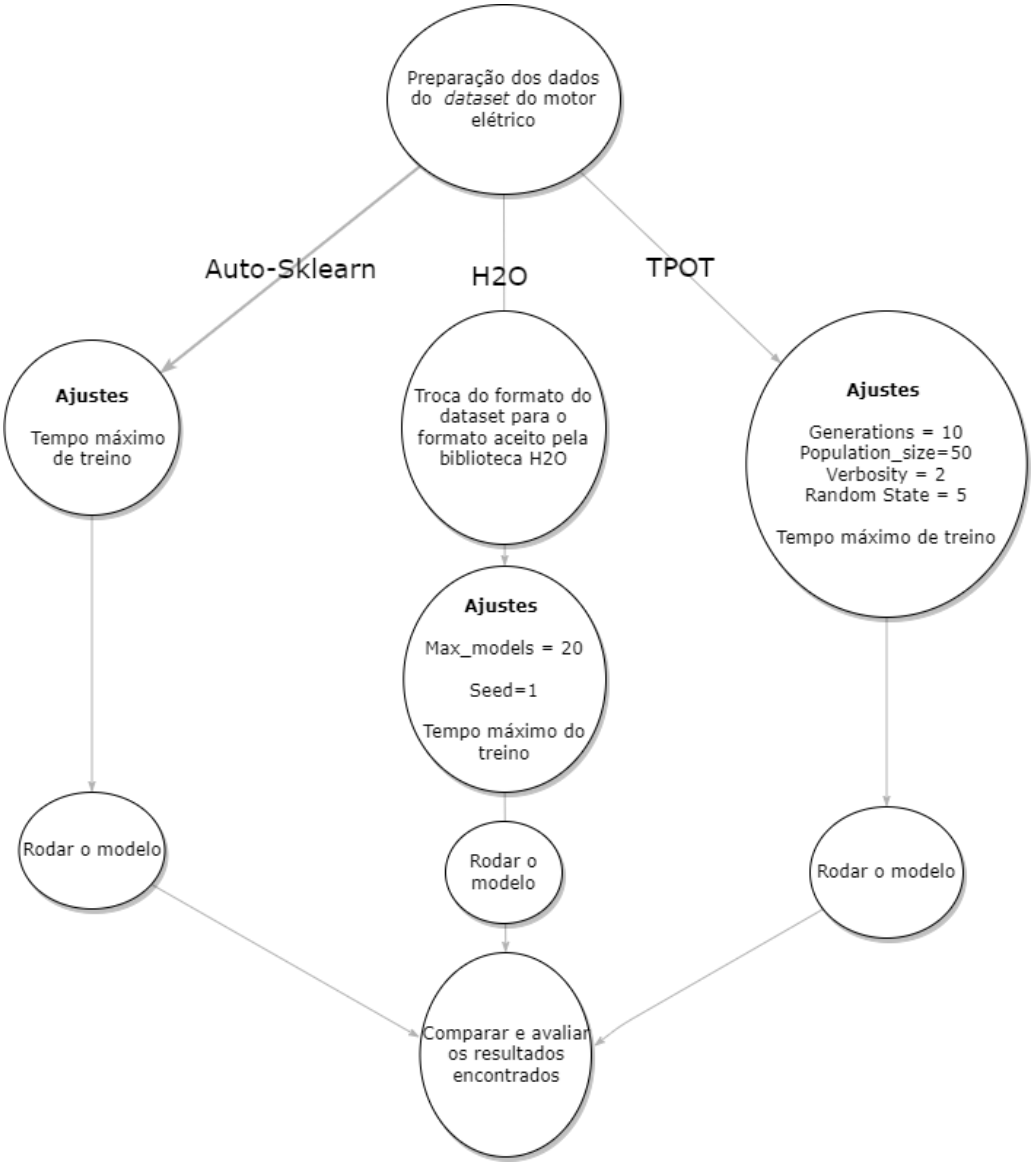
Quando utilizados os modelos *Auto-Sklearn* e TPOT bastou serem configurados os parâmetros do método de regressão. Para o primeiro modelo foi configurado o tempo de treino limite para cada algoritmo e o tempo limite de treino igual ao padrão. Para o segundo modelo foram estabelecidos a quantidade de gerações e populações, responsáveis pela quantidade de interações do *dataset*, respectivamente afetando no tempo de treinamento, a variável v*erbosity* responsável pela quantidade de informações que o modelo se comunica, campo *random state* que possui o objetivo da repetibilidade dos resultados do treinamento do modelo e o tempo máximo de treinamento. Todos estes parâmetros foram encontrados de forma empírica partindo dos valores utilizados nos exemplos disponibilizados pelos criadores das bibliotecas (OLSON, 2022; AUTO SKLEARN, 2022).

A biblioteca H2O possui a particularidade de não compreender o conjunto de dados no formato de *DataFrames*, um formato de tabela que possui linhas e colunas rotuladas o qual foi utilizado pelos outros dois métodos. Assim sendo, foi necessária a troca do formato do *dataset* para o formato H2O . Para seu treinamento, foram configuradas o máximo de modelos que poderiam ser treinados, o tempo máximo de treino por modelo, o tempo máximo de

treinamento e o *seed,* que é responsável pela reprodutibilidade dos resultados.

Na Figura 2 se apresenta o caminho traçado até chegar aos resultados do treinamento pelas bibliotecas.

Figura 2 - Fluxograma das etapas desenvolvidas para predição dos algoritmos



Fonte: Elaboração própria (2022).

3 RESULTADOS

Após serem apontadas as etapas para o tratamento dos dados do *dataset* sobre motores elétricos e a criação de modelos de AutoML, foram gerados os resultados os quais são apresentados como algoritmos e hiperparâmetros. As

métricas foram a do erro quadrático médio (*Mean Squared error)*, coeficiente de determinação R^2 e o *k-fold cross-validation.* Inicia-se esta etapa com os resultados gerados a partir da biblioteca *Auto-Sklearn.*

Na utilização da biblioteca *Auto-Sklearn* o único parâmetro configurado foi o do tempo limite de treinamento de aproximadamente 3 horas este parâmetro foi necessário pois o ambiente de execução *Colab* na versão plano gratuito contém várias restrições, entre elas o tempo limite para a utilização da plataforma de modo contínuo de no máximo 12 horas (DAVID, 2021).

Pode-se levantar o questionamento do porque não utilizar o tempo máximo de execução que a plataforma oferece. Em testes realizados com tempos maiores a plataforma interrompeu o serviço de treinamento ou a biblioteca não apresentou o comportamento esperado sendo assim foi encontrado pelo autor o tempo de treino de aproximadamente 3 horas como sendo o ideal. Dada a execução do programa foi encontrado, a partir da função *leaderboard,* que o melhor algoritmo encontrados pela biblioteca *Auto-Sklearn* foi o . *K_nearest_neighbors* com os parâmetros *n_neighbors* = 2, p=1 e *weights* = *distance.*

A partir deste algoritmo e dos parâmetros encontrados pelo modelo foi executada a predição da temperatura do rotor, "*pm",* utilizando o grupo de dados dos motores elétricos. Com isto foram obtidos os resultados demonstrados na Figura 3.

Figura 3 - Predição e resultado dos valores de treino e teste

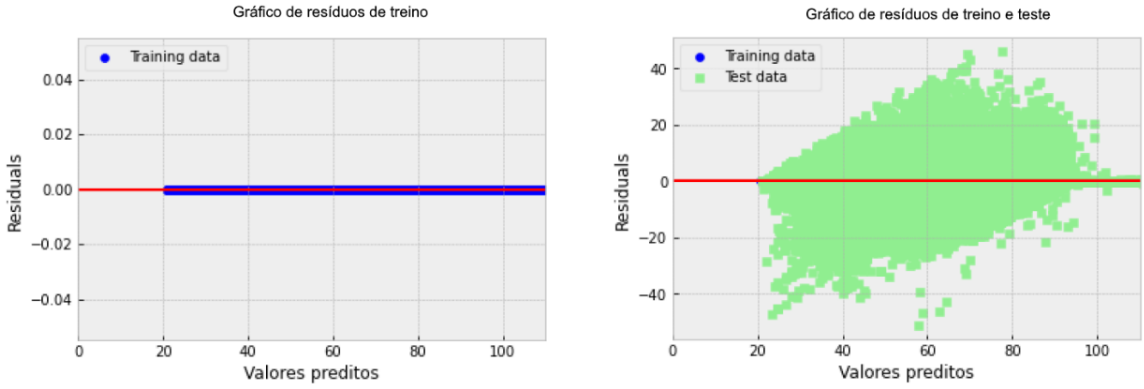
```
In [11]:

print("Scores R2 de treino", sklearn.metrics.r2_score(y_train,Pred_train_y))
print("Scores R2 de teste", sklearn.metrics.r2_score(y_test,Pred_test_y))

Scores R2 de treino 1.0
Scores R2 de teste 0.9716096096469911
```

Fonte: Elaboração própria (2022).

Como pode ser visto nas linhas do *notebook* o valor médio de R^2 também conhecido como coeficiente de determinação, um coeficiente que é considerado base para avaliar modelos de regressão (CHICCO; WARRENS; JURMAN, 2021), apresentou como resultado de treino o valor 1 e para o valor de teste

0,9716. Com os coeficientes encontrados, também foi executado os gráficos residuais de treino e de treino e teste, os quais podem ser vistos na Figura 4.

Figura 4- Gráficos residuais Auto-Sklearn

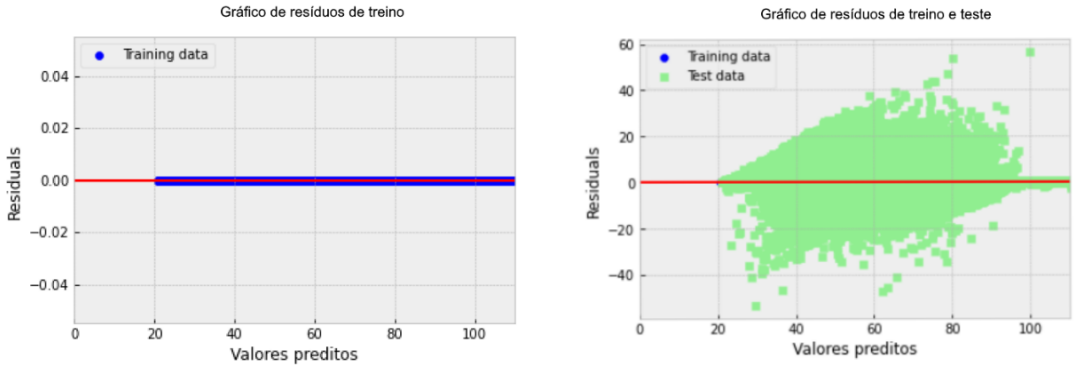


Fonte: Elaboração própria (2022).

Com os gráficos residuais traçados, utilizou-se do método *Cross-validation* utilizando a função *KFold,* a qual foi parametrizada para dividir os dados em 10 grupos e com o embaralhamento dos valores ativo

O resultado médio do coeficiente de determinação foi de 0,9576 e o valor da raiz quadrática média dos erros foi igual a 3.8937. Com os resultados da biblioteca *Auto-Sklearn* retratados, serão apresentados os dados encontrados pela biblioteca TPOT. Os parâmetros que foram estabelecidos para o uso da ferramenta foram *Generation* com o valor 10, *Population_size* com o valor 50, *Verbosity* com valor 2, *Random_State* com valor 5 e o tempo máximo de treino de 3 horas. O tempo de treino para encontro do melhor valor, utilizando do *dataset* de motores elétricos, foi de aproximadamente 3 horas e o algoritmo encontrado foi o *KNeighborsRegressor* com seus respectivos parâmetros. Com estes dados em mãos, foi feito a predição dos valores do rotor para os conjuntos de treino e teste e foram obtidos os valores do coeficiente de determinação para os dados de treino e teste como sendo iguais a 1 e 0.9574. Após isto foi executado o gráfico de resíduos apresentado pela Figura 5.

Figura 5 - Gráficos residuais TPOT



Fonte: Elaboração própria (2022).

Pode-se notar que os gráficos de resíduos foram similares se comparados com os da Figura 4 em que se utilizou a biblioteca do *Auto-Sklearn.* Então foi feita a validação cruzada utilizando a função *Kfold* onde foram feitas a separação dos dados em 10 grupos assim como na validação com a biblioteca *Auto-Skleran.* O resultado médio gerado pelo coeficiente de determinação foi de 0,9404 e o resultado da raiz quadrática média dos erros foi igual a 4,6369. Então foi executado o treinamento com a biblioteca H2O. Os parâmetros que foram utilizados são a limitação do máximo de modelos que poderiam ser treinados igual a 20 e o *seed,* que tem função de especificar números que são gerados de aleatória, igual a 1. O resultado do treino pode ser visto na Figura 6.

Figura 6 - Resultado encontrado pela biblioteca H2O

```
aml = H2OAutoML(max_models=20, seed=1)
aml.train(x=X, y=y, training_frame=h2o_frame)

AutoML progress: |
20:36:19.134: AutoML: XGBoost is not available; skipping it.

████████████████████████████████████████████████████
21:23:49.451: XRT_1_AutoML_1_20220624_203619 [DRF XRT (Extremely Randomized Trees)] failed: java.lang.OutOfMemoryError: GC over
head limit exceeded

▉
22:56:40.224: GBM_grid_1_AutoML_1_20220624_203619 [GBM Grid Search] failed: java.lang.OutOfMemoryError: GC overhead limit excee
ded

| 100%celled) 100%
```

Fonte: Elaboração própria (2022).

Foram feitos testes com trocas de variáveis e especificados tempos de treino para o uso da biblioteca H2O, porém nenhuma dessas configurações foi capaz de gerar a predição de algoritmos para os testes alcançando o final de

treino sem resultados.

4. DISCUSSÔES

Um dos primeiros questionamentos do autor foi se realmente as parametrizações feitas pelos algoritmos de AutoML melhoravam os resultados no treinamento com o *dataset* de motores elétricos. Para o teste, o autor criou um modelo com o algoritmo *KNeighbors* encontrado pela biblioteca *Auto-Sklearn* sem parâmetros e outro modelo com os parâmetros fornecidos pelo modelo e fez a comparação dos coeficientes de determinação e raiz quadrada do erro quadrático médio (RMSE). Nas Figuras 7 e 8 são apresentadas respectivamente as linhas de códigos e os resultados dos algoritmos após validação cruzada.

Figura 7 – Resultado da acurácia do algoritmo com parâmetro

```
knn_model= neighbors.KNeighborsRegressor(n_neighbors=2, p=1, weights='distance')
kfold  = KFold(n_splits=10, shuffle=True) # shuffle=True, Shuffle (embaralhar) the data.
result = cross_val_score(knn_model, X, y, cv = kfold)
rmse_cv=np.sqrt(-1*cross_val_score(knn_model,X,y ,cv=kfold,scoring='neg_mean_squared_error').mean())

print("Média do R^2 para a validação cruzada K-Fold: {0}".format(result.mean()))
print("Raiz quadrática do erro quadrático médio", rmse_cv)
```

```
Média do R^2 para a validação cruzada K-Fold: 0.9579376214412669
Raiz quadrática do erro quadrático médio 3.905702677535099
```

Fonte: Elaboração própria (2022).

Figura 8 - Resultados da acurácia do algoritmo sem parâmetros

```
knn_model= neighbors.KNeighborsRegressor()
kfold  = KFold(n_splits=10, shuffle=True) # shuffle=True, Shuffle (embaralhar) the data.
result = cross_val_score(knn_model, X, y, cv = kfold)
rmse_cv=np.sqrt(-1*cross_val_score(knn_model,X,y ,cv=kfold,scoring='neg_mean_squared_error').mean())

print("Média do R^2 para a validação cruzada K-Fold: {0}".format(result.mean()))
print("Raiz quadrática do erro quadrático médio", rmse_cv)
```

```
Média do R^2 para a validação cruzada K-Fold: 0.9471572268425781
Raiz quadrática do erro quadrático médio 4.374370851302784
```

Fonte: Elaboração própria (2022).

Na Figura 7 é visto que os resultados encontrados para o erro quadrático médio e do coeficiente de determinação médio foram melhores se comparados com os resultados da acurácia do algoritmo sem parâmetros da Figura 8, assim é visto que os parâmetros encontrados pela ferramenta de AutoML melhoraram

o desempenho da regressão.

Com os resultados foi feita a análise de cada uma das bibliotecas e de seus achados. Tanto a biblioteca *Auto-Sklearn* quanto a biblioteca TPOT alcançaram, após seus respectivos treinos de mesmos intervalos, o mesmo algoritmo, o *KNeighborsRegressor,* porém apresentaram o parâmetro *n_neighbors* com valor diferente, sendo 73 para o TPOT e 2 para o *Auto-Sklearn.*

Esta variação de parâmetro resultou em diferentes valores para a qualificação dos modelos e o que apresentou melhores resultados foi o encontrado a partir da biblioteca *Auto-Sklearn.* O valor do coeficiente de determinação encontrado pelo algoritmo sinalizado por esta biblioteca foi de 0.9716 enquanto o algoritmo encontrado pela biblioteca TPOT teve seu resultado igual a 0,9574. Após isto, para maior confiança nos valores encontrados pelos modelos, foi executada a validação cruzada e o resultado do coeficiente de determinação diminuiu, porém seguiu o mesmo padrão, o modelo do *Auto-Sklearn* apresentando maior acurácia em seu algoritmo em relação ao algoritmo encontrado pelo modelo TPOT, estes valores comentados sobre a precisão dos modelos e seus valores de parâmetros podem ser consultados no Quadro 2. Com isto, o autor avaliou que os melhores resultados para este estudo foram os da biblioteca *Auto-Sklearn* porém levanta o ponto de restrição para sua utilização, a necessidade de um sistema operacional baseado em Linux.

Em relação ao não funcionamento da biblioteca H2O para geração de algoritmos, os quais fariam o treinamento e predição do *dataset*, seria necessário mais tempo para testes e mais conhecimento sobre a biblioteca, sobre Python e sobre *datascience* para que fosse possível entender o que ocorreu e o porquê da performance da mesma não ter sido a esperada. Levanta-se aqui a necessidade e possibilidade de um estudo para entendimento do acontecido.

Uma outra análise interessante foi verificar se os resultado do coeficiente de determinação encontrado pelo modelo *Auto-Sklearn,* após a validação cruzada, se mostravam compatíveis com os resultados dos algoritmos encontrados pelos usuários da plataforma *Kaggle.* Para isto, foi feita uma busca na plataforma por usuários que estavam utilizando o *dataset* do motor elétrico e predizendo o valor da temperatura do rotor, *‘pm’.* Com isto foram encontrados estudos em que apresentaram erros quadráticos médios (SAURAV, 2020; ASARKAR, 2020) menores dos que os encontrados pelo autor. Os resultados

são apresentados pelo Quadro 2.

Quadro 2 - Resultados encontrados pelos algoritmos

Fonte de encontro do algoritmo	Algorítmo utilizado	Parâmetros	Raiz do erro quadrático médio (RMSE) após o *cross validation*	Coeficiente de determinação (R^2) após o *cross validation*
Autor - Auto-Sklearn	KNeighborsRegressor	n_neighbors=2 p=1 weights='distance'	3.8937	0,9574
Autor - TPOT	KNeighborsRegressor	n_neighbors=73 p=1 weights='distance'	4.6369	0,9716
Asarkar (2020)	*Ridge bag*	Base_estimator = ridge n_estimators = 2 Random_state = 0 n_jobs = -1	0,374	-
Saurav (2020)	Regressão linear	-	0,7722	-

Fonte: Elaboração própria (2022).

Como é apresentado pelo Quadro 2, os resultados das predições deste estudo não apresentaram os melhores valores se comparados com os outros estudos, porém a velocidade com que os modelos foram obtidos e a não necessidade de vasto conhecimento na área para encontro do algoritmo e seus parâmetros se apresentou como a vantagem do AutoML. É importante salientar que nos estudos de Saurav (2020) não foi utilizado da validação cruzada ou *Cross-validation,* podendo ser o fator para a diferença de precisão dos valores na medida do erro quadrático médio entre os resultados encontrados por ele e os deste estudo, sendo necessárias outras pesquisas para comprovar esta teoria.

4 CONCLUSÃO

Este trabalho trouxe um breve histórico do surgimento da inteligência artificial e a partir de suas ramificações apresentou a área do *Machine Learning* e posteriormente o *Automated Machine Learning*, suas ferramentas, bibliotecas e divisões, com estas ferramentas fez-se a predição da temperatura do rotor de motores elétricos.

O AutoML foi explicado e foram apresentadas quais as etapas

necessárias para o tratamento dos dados do *dataset*, bem como os programas utilizados e os testes realizados. Para entender a acurácia dos resultados gerados pelos modelos, esses foram comparados com outros trabalhos presentes na plataforma *Kaggle.* A partir disto foi feito o paralelo entre o comportamento dos resultados encontrados na pesquisa com os relatados nos artigos utilizados na revisão de literatura sendo observado o mesmo comportamento.

Para o pesquisador foi entendida como desafiadora a proposta inicial considerando a falta de conhecimento sobre a linguagem *Python* e sobre a ciência de dados, porém com a site *Kaggle,* as documentações das bibliotecas e o acervo digital trazido por meio da internet, foi possível atingir os objetivos e apresentar os caminhos tomados para encontro do modelo de predição utilizando as ferramentas de AutoML.

O planejamento elaborado para execução do trabalho foi realizado na integra e a partir dele, os objetivos traçados foram alcançados com sucesso, porém seriam necessárias mais pesquisas sobre as bibliotecas de AutoML, ferramentas de *datascience* e estatística aplicada na área da ciência de dados para aprofundamento da análise dos resultados. Uma outra observação importante se dá na não apresentação de resultados por parte da biblioteca H2O, na qual se sugere outros estudos buscando o entendimento do caso.

Em relação aos objetivos traçados para a pesquisa, conseguiu-se utilizar uma base de dados pública com relevância na área da Engenharia Mecatrônica. Foram aplicadas ferramentas do AutoML por meio de ambientes de programação largamente utilizados por cientistas de dados para geração de modelos de predição, no caso do presente estudo, predição da temperatura do rotor a partir do *dataset* de motores elétricos. Também foi realizada a avaliação dos resultados da acurácia dos modelos gerados a partir do *dataset* de motores elétricos sendo todo este processo documentado desta forma atingindo-se os objetivos da pesquisa. Com o resultado final foi organizado um repositório no *GitHub* contendo os códigos e *dataset* utilizados,para mais fácil reutilização e aperfeiçoamento do mesmo. LINK que será utilizado no Github: https://github.com/Pedrobum/AutoML_/tree/main/AutoML-main

O autor acredita que o resultado desta monografia foi satisfatório e que a proposta de um trabalho didático que apresenta os termos, um pouco do mundo

do *datascience* e a utilização das ferramentas do AutoML na prática foi alcançado.

REFERÊNCIAS

ALPAYDIN, E. Introduction to Machine Learning. 3rd. ed. London: The MIT Press, 2014

BALAJI, A; ALLEN, A. Choosing the best AutoML Framework. 2018. Disponível em: https://medium.com/georgian-impact-blog/automatic-machine-learning-aml-landscape-survey-f75c3ae3bbf2. Acesso em: 10 jul. 2022.

BÖCKER. Electric Motor Temperature. 2021. Disponível em: https://www.kaggle.com/datasets/wkirgsn/electric-motor-temperature. Acesso em: 28 jun. 2022.

CHICCO, D.; WARRENS, M. J.; JURMAN, G. The coefficient of determination R-squaredis more informative than SMAPE, MAE,MAPE, MSE and RMSE in regressionanalysis evaluation. PeerJ Comput. Sci, [s.l.], v. 7, p. e623, 2021. Disponível em: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8279135/. Acesso em: 28 jun. 2022.

DAVID. Alternative to Colab Pro: Comparing Google's Jupyter Notebooks to Gradient Notebooks. 2021. Disponível em: https://blog.paperspace.com/alternative-to-google-colab-pro/. Acesso em: 28 jun. 2022.

ESCALANTE, H. Automated Machine Learning – a brief review at the end of the early years. 2022. Disponível em: https://arxiv.org/pdf/2008.08516.pdf Acesso em: 22 jul. 2022.

FACELI, K. et al. Inteligência artificial: uma abordagem de aprendizado de máquina. Rio de Janeiro: LTC, 2011.

GOOGLE. Colaboratory Frequently Asked Questions. 2022. Disponível em: https://research.google.com/colaboratory/faq.html. Acesso em: 28 jun. 2022.

HE, X.; ZHAO, K.; CHU, X. AutoML: A survey of the state-of-the-art. Knowledge-Based Systems, [s.l.], v. 212, jan. 2021. Disponível em: https://www.sciencedirect.com/science/article/pii/S0950705120307516. Acesso em: 23 fev. 2022.

IBM CLOUD EDUCATION. Natural Language Processing (NLP). 2020a. Disponível em: https://www.ibm.com/cloud/learn/natural-language-processing. Acesso em: 21 mai. 2022.

IBM. What is computer vision? 2022. Disponível em: https://www.ibm.com/topics/computer-vision. Acesso em: 21 mai. 2022.

KUMAR, C. Artificial Intelligence: Definition, Types, Examples, Technologies. 2018. Disponível em: https://chethankumargn.medium.com/artificial-intelligence-definition-types-examples-technologies-962ea75c7b9b. Acesso em: 11 mai. 2022.

MÜLLER, A. C.; GUIDO, S. Introduction to Machine Learning with Python. A guide for data scientists. Massachusetts (USA): O'Reilly, 2017.

OLSON, R. S. Overview. 2022. Disponível em: http://epistasislab.github.io/tpot/examples/#titanic-survival-analysis. Acesso em: 02 jul. 2022.

PETROVA, S. This is How Companies are Applying Machine Learning to Remain Competitive. 2022. Disponível em: https://adevait.com/machine-learning/companies-using-machine-learning. Acesso em: 10 jul. 2022.

RUSSEL, S.; NORVIG, P. Artificial Intelligence A Modern Approach Third Edition. 3rd. ed. Nova Jersey (USA): Prentice Hall, 2010 .

SANTA CATARINA. IFSC. Projeto Pedagógico do Curso Engenharia Mecatrônica. 2012. Disponível em: http://cs.ifsc.edu.br/portal/files/florianopolis_PPC_engenharia_mecatronica.pdf. Acesso em: 10 jul. 2022.

SAURAV. Exploratory Analysis of Electric Motor temperature. 2020. Disponível em: https://www.kaggle.com/code/sauravmom/exploratory-analysis-of-electric-motor-temperature. Acesso em: 2 jul. 2022.

SEBESTA, R. W. conceitos de linguagem de programação. 9. ed. Porto Alegre: Artmed, 2010.

VENKATA, V. P. D. et al. Automating water quality analysis using ML and auto ML techniques. Environmental research, [s.l.], v. 202, p. 111720, 2021. Disponível em: https://pubmed.ncbi.nlm.nih.gov/34297938/ Acesso em: 28 fev. 2022