REVISTA CIENTÍFICA SENAI-SP

UTILIZANDO RAPIDMINER PARA MANUTENÇÃO PREDITIVA

USING RAPIDMINER FOR PREDICTIVE MAINTEINANCE

Thiago Sales Araujo[1], [i]
Daniel Barbuto Rossato[2], [ii]

Data de submissão: (18/12/2021) Data de aprovação: (26/07/2022)

RESUMO

Este trabalho apresenta um exemplo de implementação de Manutenção Preditiva em equipamentos utilizando o *software* RapidMiner. Durante a operação de um equipamento podem ser coletados alguns dados, que com o emprego de aprendizado de máquina, podem ajudar a prever falhas e evitar que elas aconteçam, reduzindo desta forma a quantidade de paradas e mantendo o bom desempenho do equipamento por mais tempo do que quando aplicados métodos de manutenção convencionais. No desenvolvimento deste artigo foram aplicados alguns algoritmos de aprendizado de máquina em um conjunto de dados sintéticos, modelados com base na operação de uma fresadora e são exibidos os resultados obtidos com as configurações adotadas.

ABSTRACT

This work presents an example of Predictive Maintenance implementation in equipment using RapidMiner software. During an equipment operation some data can be collected, that with the use of machine learning, can help to predict failures and avoid them to occur, thus reducing downtime and keep equipment with good performance level for longer time than when used conventional maintenance methods. In the development of this article some machine learning algorithms were applied in a synthetic dataset, modeled based on the operation of a milling machine and the obtained results with the adopted configurations are displayed.

1 INTRODUÇÃO

A manutenção é muito importante para todas as empresas, tendo em vista que a vida útil dos ativos, juntamente com o tempo de operação em bom nível dos equipamentos pode impactar no custo produtivo e consequentemente afetar as margens de lucro, competitividade e saúde financeira do negócio (DIAMOND; MARFATIA, 2013). É seguro dizer que todas as companhias visam ter seus equipamentos funcionando com a menor queda de

[1] Thiago Sales Araujo. Pós-graduando em Automação e Controle na Faculdade SENAI "Mariano Ferraz". E-mail: thiago.araujo8@senaisp.edu.br
[2] Daniel Barbuto Rossato. Docente, Doutorando e Mestre em Engenharia Elétrica da Faculdade SENAI "Mariano Ferraz". E-mail: daniel.rossato@sp.senai.br

performance e quantidade de paradas possíveis ao longo do tempo. A manutenção preditiva é uma metodologia que ao coletar dados de determinado processo e seus equipamentos, possibilita prever quando o equipamento irá apresentar defeito antes mesmo que a falha em sua operação ocorra. Sendo assim, é possível substituir os intervalos fixos de manutenção preventiva de acordo com a necessidade indicada pelas informações obtidas diminuindo o tempo de parada. Os registros de dados do equipamento, como temperatura, ruído e pressão podem ser processados utilizando aprendizado de máquina para realizar as predições que auxiliarão nas manutenções a serem realizadas. Esta metodologia de manutenção, ao utilizar acompanhamento periódico com juntamente com a análise de dados, permite que não seja obrigatoriamente pré-determinado um calendário fixo de inspeções (TOTVS, 2021).

O monitoramento constante da operação permite que defeitos nas máquinas sejam previstos com antecedência, antes de se tornarem obstáculos, quando ainda apresentam somente indícios, que muitas vezes passariam despercebidos, não fosse pelas ferramentas de aprendizado de máquinas com o advento da manutenção preditiva, com o monitoramento contínuo.

Conforme indicado por um levantamento da empresa de consultoria empresarial (MCKINSEY, 2017), o tempo de inatividade de máquina pode ser reduzido entre 30% e 50% e proporcionar um acréscimo de vida útil entre 20% e 40% para uma máquina através da manutenção preditiva.

Algumas das vantagens proporcionadas pela manutenção preditiva, são a prevenção de paradas forçadas (que normalmente ocorrem após falha grave de um equipamento), aumento do tempo disponível de certo equipamento, aumento da vida útil, menor número de reparos e desmontagem de equipamentos e detecção de falhas que possam causar pausa na linha de produção.

O avanço da tecnologia tem permitido que as indústrias possuam cada vez mais sensores, permitindo o monitoramento praticamente em tempo real dos processos e equipamentos. Diversas características monitoradas podem ser utilizadas para a implementação da manutenção preditiva por meio de ferramentas de aprendizado de máquina. Dentre algumas das variáveis mais utilizadas para previsões, estão a análise de vibração, ultrassom, termografia (análise de temperatura), análise de óleo, trincas e de ruídos, podendo eventualmente serem utilizadas outras características que possam ter influência nos erros mais comuns para certo equipamento.

Como resultado, a manutenção preditiva acaba por proporcionar redução de custos de fabricação, maior segurança para os profissionais envolvidos no processo e maior lucratividade para a empresa que a adota.

Neste artigo serão utilizados, como exemplo, dados estáticos de um equipamento industrial para fins de treinamento e testes de algoritmos de aprendizagem de máquina, ilustrando a eficiência do método e comparando o desempenho dos algoritmos aplicados. Porém, numa situação real, os dados idealmente possuem característica mais dinâmica, sendo coletados e visualizados até mesmo em tempo real. A partir da análise do conjunto de dados, erros podem ser previstos e a manutenção realizada antes que um eventual problema se torne maior, a ponto de ocasionar parada na fabricação e consequente prejuízo.

2 DESENVOLVIMENTO

A seguir serão apresentadas as principais metodologias de manutenção e como o processamento dos dados pode ser realizado para predição de falhas com o uso do RapidMiner, que é um *software* que possibilita a mineração de dados e aprendizado de máquinas para realização de predições.

Numa situação real com a aplicação da metodologia de manutenção preditiva, os dados são coletados em tempo real, com dados sendo fornecidos por diversos sensores a todo momento (RAPIDMINER, s.d.). Após o modelo ter sido treinado pelo algoritmo de manutenção preditiva, o gerenciamento da manutenção pode ser melhorado, com eventuais falhas sendo previstas antes de ocorrer, possibilitando reparos antecipadamente. Para efeito de demonstração de funcionamento e eficiência, serão utilizados os dados sintéticos (estáticos) de um *dataset* criado para fins de estudos.

2.1 Categorias de Manutenção

Manutenção corretiva: é uma das formas mais antigas de se fazer manutenção e existe desde antes da mecanização da indústria. Nesta modalidade, os reparos somente são realizados após os equipamentos apresentarem defeitos ou falharem, parando de funcionar. Podem também ser realizadas não necessariamente em caráter de urgência, porém visando corrigir o desempenho de uma máquina. A manutenção corretiva pode ser planejada - quando é percebida uma diminuição na performance de um equipamento - ou não planejada, quando ocorre uma falha fortuita para a qual não existe uma previsão ou preparo para reparo a ser feito antes da falha. As manutenções corretivas não planejadas costumam ser bastante custosas e demandar um longo período para serem realizadas (ENGEMAN, 2021?; IBM, 2019);

Manutenção preventiva: é realizada em intervalos programados, com a troca de partes pré-determinadas, com base na experiência. Este tipo de manutenção é bastante difundido e utilizado em várias empresas. São estipuladas medições como número de ciclos realizados, tempo de uso, distância percorrida, dentre outros. Como existe um plano de manutenção, com a programação do que será trocado e quando as trocas serão feitas, a parada não é feita de forma inesperada, podendo evitar quedas de eficácia e mantendo a confiabilidade nos equipamentos. Desta forma, a manutenção costuma ser mais barata que a corretiva e evita perdas por paradas imprevistas. Pelo fato de as trocas serem roteirizadas, por vezes culmina com a troca de peças que não precisavam ser trocadas, porém esta filosofia de manutenção reduz a degeneração da máquina e é capaz de aumentar sua vida útil (ENGEMAN, 2021?; IBM, 2019);

Manutenção preditiva: os reparos são feitos quando os dados indicam que uma falha pode estar prestes a ocorrer. O aumento do uso de sensores em conjunto com uso de *softwares* torna possível fazer ajustes quando acontecem mudanças em parâmetros de controle, como características pneumáticas, hidráulicas, elétricas, mecânicas e outras, o que pode ser visualizado muitas vezes em tempo real. Esse monitoramento contribui para evitar que a empresa tenha gastos evitáveis, seja por parada inesperada de sua linha de produção, com manutenção imprevista (como na manutenção corretiva) ou troca de peças sem necessidade (a exemplo do que pode ocorrer com a manutenção preventiva). A manutenção preditiva por vezes pode ser executada remotamente, sem a vistoria física, que demandaria técnicos, paradas e em alguns casos desmonte da máquina em questão.

Com a previsão mais apurada de quais peças substituir e quando fazer a troca, este método permite além da redução de custos e aumento da vida útil e disponibilidade para uso da máquina, a verificação de causas dos defeitos, o que pode possibilitar alterações em processos para evitar desgastes futuros do maquinário (ENGEMAN, 2021?; IBM, 2019).

2.2 Mineração de Dados

A mineração de dados é encarada como parte de uma nova revolução industrial (SANTANA, 2019) e tem ganhado cada vez mais espaço mundialmente, devido as possibilidades que proporciona, em diversos campos, desde fornecer vantagem competitiva com economia, aumento de produtividade e previsão de vendas no mundo dos negócios, como para a saúde no auxílio de diagnósticos e política na previsão de votos, quanto para muitos outros ramos de extrema importância e interesse para a sociedade. Ela consiste basicamente em extrair informações de um volume grande dados com o uso de ferramentas de *software*, permitindo estabelecer correlações entre os dados que facilmente passariam despercebidas por olhos humanos ou por ferramentas tradicionais. A utilização de variados tipos de algoritmos na mineração de dados permite a redução de erros humanos e enviesamento nas previsões.

Para usufruir das possibilidades da mineração de dados, é necessário captar dados, realizar a preparação deles e aplicar algoritmos, que podem ser do tipo supervisionado (dados históricos são usados para treinar o algoritmo, a fim de identificar padrões, classificar itens e realizar previsões de valores), não-supervisionados (o algoritmo tenta achar padrões e correlações em dados não categorizados) e semi-supervisionados (que são uma mistura dos tipos anteriores). Esses algoritmos também são categorizados de acordo com seu funcionamento, sendo comumente enquadrados como algoritmos de regressão, *clustering* (agrupamento de dados) *e* classificação (DOTY, 2020).

Diversas ferramentas e linguagens de programação podem ser empregadas na mineração de dados, estando dentre as mais comuns: R, RapidMiner, SQL, Python e Excel, conforme indicado por uma pesquisa realizada pelo site KDNuggets (KDNUGGETS, 2015).

Neste artigo exploraremos um *dataset* utilizando RapidMiner. O RapidMiner é uma ferramenta que permite a preparação de dados, aplicação de algoritmos de aprendizado de máquina de forma simples e rápida (RAPIDMINER, s.d.).

A ferramenta permite uso de diversos algoritmos, como regressão linear, k-means *clustering*, algoritmos baseados em árvores de decisão (*decision tree)*, redes neurais, para citar algumas. Adiante serão abordados com um pouco mais detalhe os algoritmos que serão usados para análise do conjunto de dados objeto de estudo deste artigo.

Os algoritmos de aprendizado de máquina se dividem basicamente em 3 grupos: classificação, regressão e agrupamento (VELASQUEZ, 2020), conforme resumido a seguir.

- Classificação: utiliza dados de entrada para gerar um classificador que indica qualidade de um valor não observado inicialmente, como indicar a partir de dados de entrada se um equipamento apresenta ou não falhas. Exemplos de algoritmos: Árvore de Decisão, *Naive Bayes*, KNN, Floresta Aleatória;
- Regressão: também prevê resposta a partir de dados de entrada, porém em vez de categorizar (classificar), estima valor numérico, como por exemplo estimar valor gasto por cliente num restaurante, baseado em sua idade e renda. Exemplos de algoritmos: Regressão Linear, ARIMA;

- Agrupamento: agrupa dados em grupos chamados de "*clusters*". Esses grupos possuem semelhanças entre si e diferenças em relação a outros grupos, como por exemplo, o agrupamento de clientes em grupos de acordo com o consumo. Exemplos de algoritmos: *k-Means Clustering, Agglomerative Clustering*.

Para verificação da performance de um algoritmo do tipo classificação (que é o tipo dos algoritmos adotados neste trabalho), são utilizados alguns métodos, dentre eles acurácia (observaçõescorretasclassificadas/númerototaldeobservaçõesclassificadas), erro (1-Acurácia) e a matriz de confusão. A matriz de confusão é uma ferramenta poderosa, pois ajuda a detalhar os dados previstos ao separar verdadeiros positivos e negativos (TP e TN, respectivamente) e falsos positivos e falsos negativos (FP e FN, respectivamente). Para a matriz de confusão são calculados ainda a precisão (TP/TP+FP) e recall (TP/TP+FN).

2.3 Dataset

Para a realização de predição de falhas utilizando o RapidMiner, foram utilizados dados sintéticos que refletem dados encontrados em manutenção preditiva na indústria.

Usualmente os dados captados por sensores, termômetros e outros nas indústrias podem possuir dados de alguns parâmetros faltantes em determinado ponto ou mesmo medições que devem ser descartadas por estarem muito discrepantes das demais (chamadas *outliers*) por diversos motivos, como uma interferência no momento da coleta da medição. Esses dados precisariam ser filtrados para não interferir nos dados realmente importantes para o treinamento do modelo para predição de falhas.

O conjunto de dados foi doado para estudos ao Repositório de Aprendizado de Máquina da Universidade da Califórnia Irvine por Stephan Matzka.

O dataset sintético para manutenção preditiva desenvolvido por Matzka, toma como base uma fresadora em operação e contém 10.000 linhas de dados e 14 características dispostas em colunas. Na tabela 1 é mostrado um resumo do conjunto de dados considerado neste estudo.

Tabela 1 – Resumo dos dados do *dataset*

Nome	Tipo	Faltante	Estatísticas		
UID (Unique Identifier) *Identificador único*	Inteiro	0	(Mínimo) 1	(Máximo) 10000	(Média) 5000,50
Product ID *ID do Produto*	Nominal	0	(Menos frequente) M24859 (1)	(Mais frequente) H29424 (1)	(Valores) H29424 (1), H29425 (1), ...[9998 mais]
Type *Tipo*	Nominal	0	(Menos frequente) H (1003)	(Mais frequente) L (6000)	(Valores) L (6000), M (2997), ...[1 mais]
Air temperature [K] *Temperatura do ar [K]*	Real	0	(Mínimo) 295,300	(Máximo) 304,500	(Média) 300,005

Nome	Tipo	Faltante	Estatísticas		
Process temperature [K] *Temperatura do processo [K]*	Real	0	(Mínimo) 305,700	(Máximo) 313,800	(Média) 310,006
Rotational speed [rpm] *Velocidade de rotação [rpm]*	Inteiro	0	(Mínimo) 1168	(Máximo) 2886	(Média) 1538,776
Torque [Nm] *Torque [Nm]*	Real	0	(Mínimo) 3,800	(Máximo) 76,600	(Média) 39,987
Tool wear [min] *Desgaste da ferramenta [min]*	Inteiro	0	(Mínimo) 0	(Máximo) 253	(Média) 107,951
Machine failure *Falha de máquina*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,034
TWF (Tool Wear Failure) *Falha por desgaste da ferramenta*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,005
HDF (Heat Dissipation Failure) *Falha de dissipação de calor*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,011
PWF (Power Failure) *Falha de potência*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,009
OSF (Overstrain Failure) *Falha por sobrecarga*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,010
RNF (Random Failures) *Falhas aleatórias*	Inteiro	0	(Mínimo) 0	(Máximo) 1	(Média) 0,002

Fonte: Elaborada pelo autor.

A tabela 2 detalha os atributos de acordo com informações de Matzka, o autor do *dataset* sintético da fresadora. O objetivo é apenas apresentar melhor os atributos, antes da ilustração dos resultados obtidos por aplicação de algoritmos de aprendizado de máquina, sem maior aprofundamento na forma que os dados foram gerados pelo autor neste trabalho:

Tabela 2 – Resumo dos atributos do *dataset*

Atributo	Descrição
UID	Identificador único do dado que varia de 1 a 10000.
Product ID	Identificador do produto e contém as letras L (Low), M (Medium) ou H (High), onde L se traduz por baixo (50% dos produtos), M por médio (30% dos produtos) e H por alto (20% dos produtos) nas quais as letras são variantes que indicam qualidade do produto, seguida por número serial específico.
Air Temperature [K]	Temperatura do ar na unidade de medida Kelvin, gerada a partir de processo aleatório e depois normalizado para um desvio padrão de 2K em torno de 300K.
Process Temperature [K]	Temperatura do processo na unidade de medida Kelvin, gerada usando processo aleatório normalizado para um desvio padrão de 1K, adicionado à temperatura do ar mais 10K.
Rotational Speed [rpm]	Velocidade de rotação em rpm, calculada a partir de uma potência de 2860 W, sobreposta com um ruído normalmente distribuído.
Torque [Nm]	Valores de torque em Nm, distribuídos em torno de 40 Nm sem nenhum valor negativo.
Tool Wear [min]	Desgaste da ferramenta em minutos. As variantes de qualidade H, M e L adicionam, respectivamente 5, 3 e 2 minutos de desgaste à ferramenta usada no processo e uma etiqueta de 'falha da máquina' que indica se a máquina falhou neste ponto de dados.
Tool Wear Failure (TWF)	Identificador de falha por desgaste da ferramenta. A ferramenta será substituída por falha em um tempo de desgaste da ferramenta selecionado de forma aleatória entre 200 e 240 minutos (120 vezes neste conjunto de dados). A ferramenta é substituída 69 vezes e falha 51 vezes (quantidades atribuídas aleatoriamente).

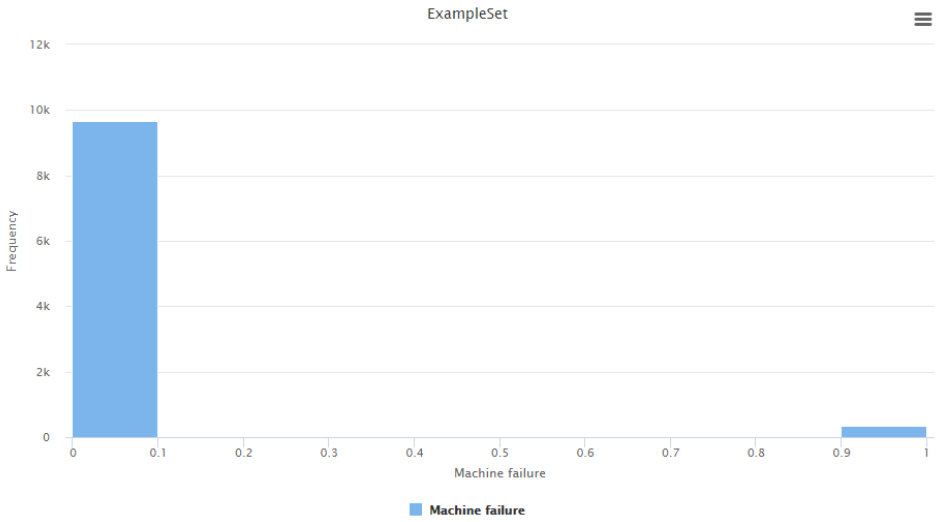
Atributo	Descrição
Heat Dissipation Failure (HDF)	Falha de dissipação de calor, que ocorre quando a dissipação de calor causa uma falha de processo. Acontece quando a diferença entre a temperatura do ar e do processo é inferior a 8,6 K e a velocidade de rotação da ferramenta for inferior a 1380 rpm. Aparece de 115 vezes no conjunto de dados.
Power Failure (PWF)	Falha de energia, que ocorre 95 vezes nos dados apresentados. O produto do torque e da velocidade de rotação (em rad/s) é igual à potência necessária para o processo. Quando esta potência estiver abaixo de 3500 W ou acima de 9000 W, o processo falha.
Overstrain Failure (OSF)	Falha por sobrecarga, que aparece em 98 pontos de dados. Acontece falha de sobrecarga quando produto do desgaste da ferramenta e torque exceder 11.000 minNm para a variante de produto L, 12.000 para a variante M e 13.000 para a variante H.
Random Failures (RNF)	Falhas aleatórias, que podem se manifestar independentemente de parâmetros do processo em 0,1% dos casos. Apesar disso dos 10000 pontos de dados deste *dataset*, as falhas aleatórias ocorrem apenas 5 vezes.

Fonte: Adaptada de site UCI

Dos dados da tabela 2, é possível verificar que as falhas de máquina podem ser de 5 tipos (TWF, HDF, PWF, OSF e RNF), que não necessariamente possuem relação entre si. Quando pelo menos um dos tipos de falha é verdadeiro, o rótulo *(label)* Falha de Máquina *(Machine Failure)* aponta 1, indicando que existe falha no processo. Entretanto para o método de aprendizado de máquina, não fica claro qual dos tipos de falha fez o processo falhar.

O histograma gerado no RapidMiner mostrado na figura 1, torna visual a diferença entre os pontos de dados com falha (rótulo *Machine Failure* = 1) e sem falha (rótulo *Machine Failure* = 0). É possível observar graficamente a frequência com que cada valor de falha aparece, sendo que distribuídos dentre os 10000 pontos do conjunto de dados, é registrado algum tipo de falha 339 vezes, enquanto pontos sem falha aparecem 9661 vezes.

Figura 1 – Histograma



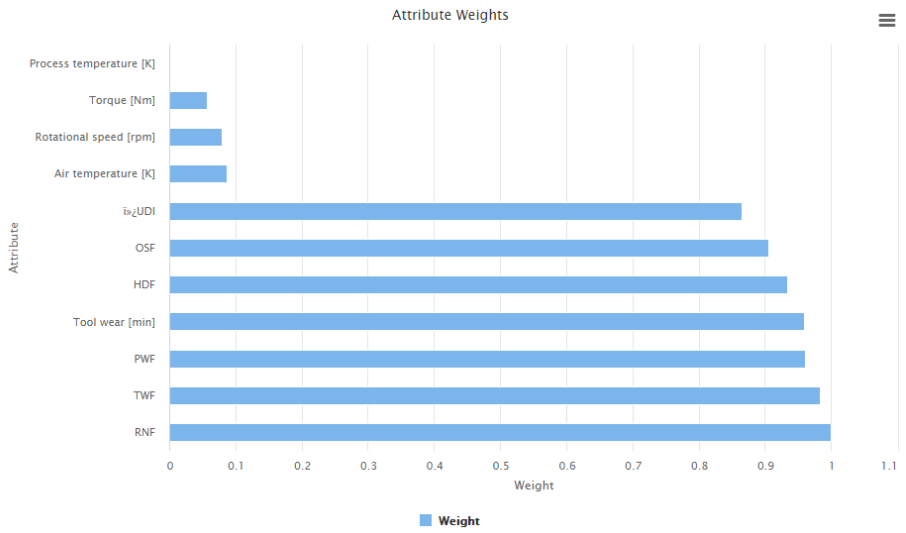
Fonte: Elaborada pelo autor.

2.4 Análise do *Dataset*

Para aplicação de modelos preditivos nos dados é necessário realizar um pré-processamento dos dados obtidos, verificando quais atributos possuem maior influência nos resultados e checar por dados duplicados, faltantes e valores discrepantes *(outliers)*, que precisam ser filtrados e removidos para evitar contaminação na predição. O conjunto de dados sintéticos apresentado, apresenta a vantagem de não possuir dados faltantes e outliers, permitindo foco na aplicação de algoritmos e a verificação de sua eficácia.

Para checar os atributos mais influentes para predição, foi utilizado o operador *Correlation Matrix* nos dados obtidos, com a obtenção dos resultados mostrados na figura 2:

Figura 2 – Pesos dos atributos



Fonte: Elaborada pelo autor.

Após análise dos atributos, foram selecionados os mais importantes (Air temperature [K], Machine failure, Process temperature [K], Rotational speed [rpm], Tool wear [min], Torque [Nm] e Type), com a remoção de UDI, Product ID (por serem apenas identificadores) e os tipos de falha (OSF, HDF, PWF, TWF e RNF), visto que eles são consequência de alterações nos parâmetros e quaisquer deles que sejam verdadeiros, entrarão na conta das falhas de máquina, fazendo a *label "*Machine Failure" que é categórica, alterar de 0 (falsa) para 1 (verdadeira).

Para aplicação no conjunto de dados, foram escolhidos os algoritmos de classificação *Decision Tree* (Árvore de Decisão)*, Random Forest* (Floresta Aleatória) *e KNN* (K-Vizinhos mais Próximos), tendo sido considerada uma proporção de 70% dos dados para treinamento e 30% para teste medição de eficácia do algoritmo. Para todos os algoritmos, ao realizar a divisão dos dados, foi considerada a configuração *shuffled subsampling*, para embaralhar os dados e evitar divisão enviesada, que poderia contaminar as predições.

2.6.1 Árvore de Decisão

Este algoritmo, de forma similar a um fluxograma apresenta nós de decisão que são divididos hierarquicamente entre "nós-raiz" (base de dados) e "nós-folha" (resultados). Os nós são ligados por meios de regras *if-then,* questionando por exemplo se um dado atributo possuir certa característica, possuirá resultado maior ou menor que X? Caso seja menor que X, irá para um lado da árvore, caso contrário, irá para o lado oposto e segue a regra nos nós seguintes. Para decidir o nó raiz, é verificado o ganho de informação entre atributos e o de maior ganho será o nó-raiz (SACRAMENTO, 2021). Para fazer as predições, o algoritmo utiliza alguns métodos, dentre eles a entropia, que leva em consideração a distribuição dos dados nas variáveis consideradas na predição, com relação a variável alvo *(label),* sendo que quanto mais desordenados estão os dados, maior é a entropia. A entropia (também chamada de medida de impureza) de variáveis de classe binária é dada pela fórmula:

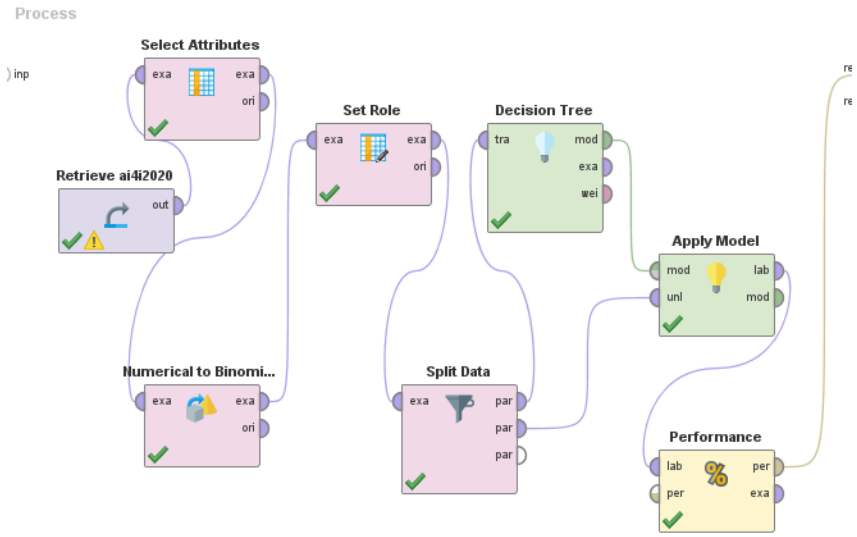
$$S = -p(a) \times \log_2 p(a) - p(b) \times \log_2 p(b)$$

Onde:
S representa a entropia
p é a representação da proporção de exemplos em relação a todo o conjunto
a e b são os valores de classe binária que uma dada variável pode assumir

Quanto menor a entropia, maior é considerado o ganho de informação de uma variável e a que tem o maior ganho é escolhida como nó raiz da árvore de decisão (MITCHELL, 1997).

Foi aplicado o algoritmo árvore de decisão no conjunto de dados, utilizando as configurações de profundidade máxima da árvore igual a 15, tamanho mínimo de folhas 4 e tamanho mínimo para separação dos nós igual a 4, como resumido no diagrama de blocos da figura 3, com a obtenção dos resultados indicados na matriz de confusão exibida na tabela 3.

Figura 3 – Diagrama de blocos do RapidMiner com aplicação do algoritmo Árvore de Decisão



Fonte: Elaborada pelo autor.

Tabela 3 – Matriz de Confusão com aplicação do algoritmo Árvore de Decisão no RapidMiner

accuracy: 97.27%

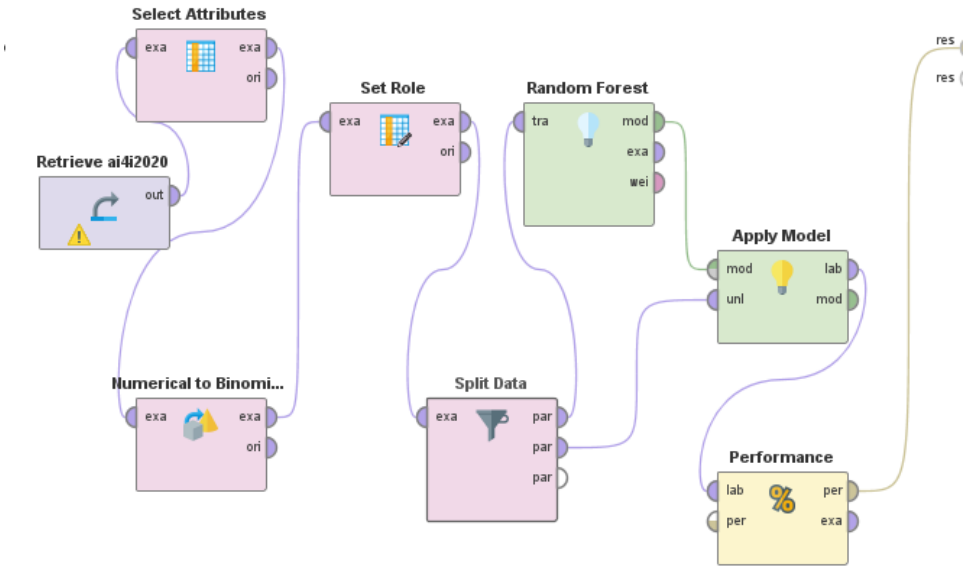
	true false	true true	class precision
pred. false	2898	79	97.35%
pred. true	3	20	86.96%
class recall	99.90%	20.20%	

Fonte: Elaborada pelo autor.

2.6.2 Floresta Aleatória

O princípio de funcionamento do algoritmo floresta aleatória é a utilização de diversas árvores de decisão, geradas a partir de amostras aleatórias do conjunto de dados. Então é selecionada a amostra mais adequada para ser nó-raiz através de votação e são gerados nós-filhos, repetidamente até chegar ao número de árvores desejado. Variáveis que mais aparecem em certo nó podem substituir valores ausentes. A média das previsões das árvores formam a predição da floresta aleatória (PESSANHA, 2019). Os parâmetros selecionados na aplicação do algoritmo foram 120 árvores aleatórias e profundidade máxima de cada árvore 8 (resumo do diagrama de blocos na figura 4), com a obtenção dos resultados exibidos na tabela 4.

Figura 4 – Diagrama de blocos do RapidMiner com aplicação do algoritmo Floresta Aleatória



Fonte: Elaborada pelo autor.

Tabela 4 – Matriz de Confusão com aplicação do algoritmo Floresta Aleatória no RapidMiner

accuracy: 97.17%

	true false	true true	class precision
pred. false	2898	82	97.25%
pred. true	3	17	85.00%
class recall	99.90%	17.17%	

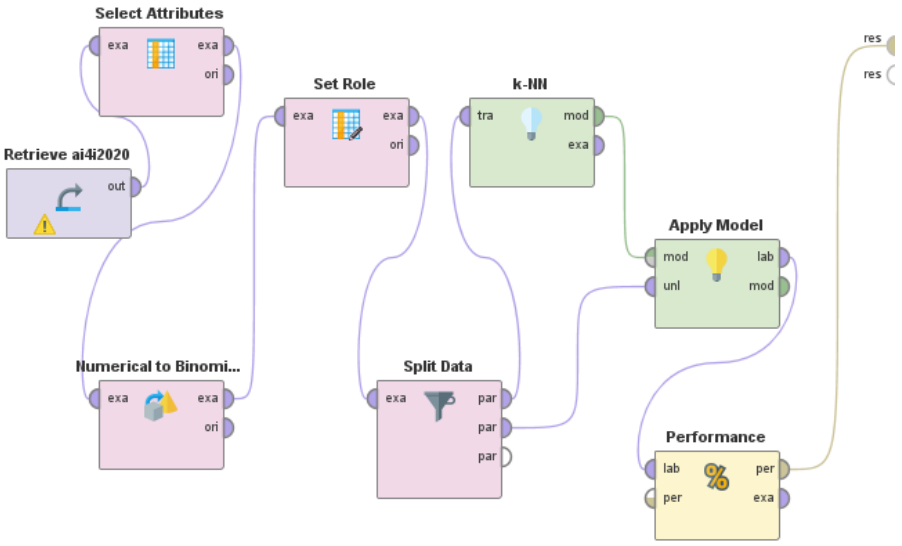
Fonte: Elaborada pelo autor.

2.6.3 K-NN

O algoritmo k-NN utiliza uma variável K (parâmetro principal) que busca vizinhos mais próximos deste ponto e decide a classe da qual ele faz parte. Para isto, é calculada a distância entre os pontos, são encontrados os pontos mais próximos e então é votada a classe do ponto que se quer realizar a predição. O número de vizinhos (k) indica a quantidade de vizinhos a serem considerados e então é realizada a medição das distâncias dos vizinhos mais próximos e é definido uma classificação mais provável com base na média das distâncias do valor desconhecido para os vizinhos mais próximos. A medição das distâncias pode ser feita por diversos métodos, como distância Euclidiana (que pode ser provada pela aplicação repetida do Teorema de Pitágoras), distância de Hamming, distância Manhattan e distância de Markowski (LUZ, 2019).

São apresentados na figura 5 o diagrama de blocos do RapidMiner e na tabela 5 a matriz de confusão da aplicação do algoritmo k-NN no conjunto de dados com os parâmetros k=7 e medição por distância Euclidiana.

Figura 5 – Diagrama de blocos do RapidMiner com aplicação do algoritmo k-NN



Fonte: Elaborada pelo autor.

Tabela 5 – Matriz de Confusão com aplicação do algoritmo k-NN no RapidMiner

accuracy: 96.83%

	true false	true true	class precision
pred. false	2893	87	97.08%
pred. true	8	12	60.00%
class recall	99.72%	12.12%	

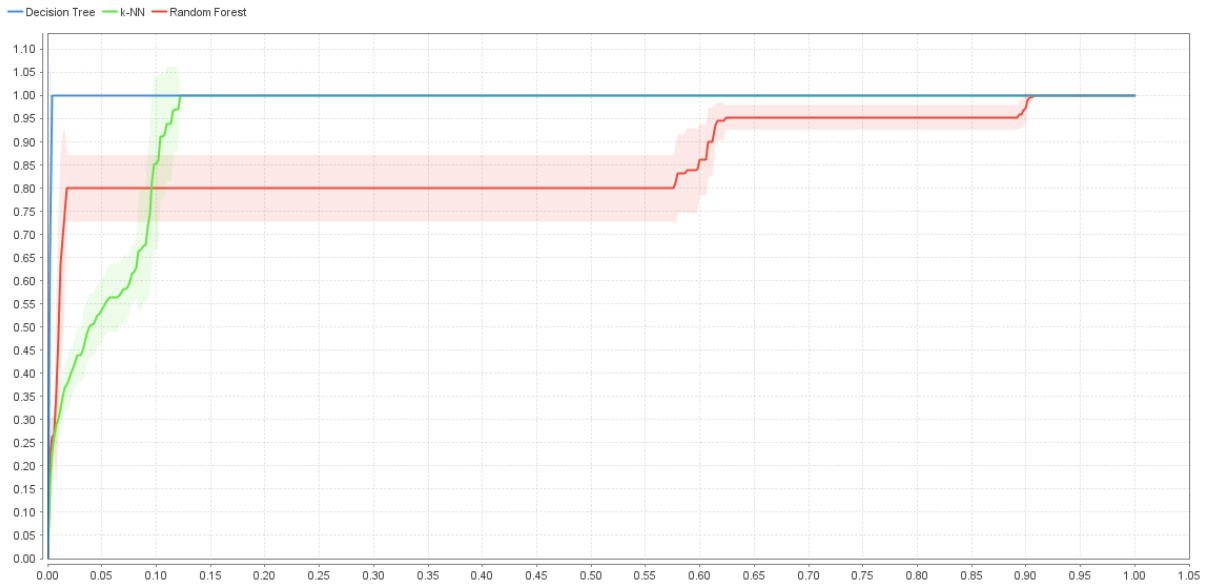
Fonte: Elaborada pelo autor.

2.6.4 Curva ROC

As curvas ROC (*Receiver Operator Characteristic Curve*, em português Curva Característica de Operação do Receptor) são utilizadas para relacionar sensibilidade (taxa de verdadeiros positivos - TP, que varia de 0 a 1), localizada no eixo Y e especificidade (taxa de verdadeiros negativos - TN), localizada no eixo X. A taxa de sensibilidade é dada por TP/(TP+FN) e a especificidade é dada por FP/(FP+TN), sendo que FN e FP representam falso negativo e falso positivo, respectivamente. As classificações dadas pela curva ROC não são perfeitas, mas ajudam a visualizar a exatidão de um modelo. A área abaixo da curva, chamada AUC (*Area Under the Curve)* gerada (variando de 0 a 1), é empregada para interpretação e quanto maior a AUC, melhor a qualidade das previsões. Um modelo hipotético com previsões 100% corretas teriam AUC = 1. Desta forma, quanto mais próximo do lado superior esquerdo da curva (mais perto de 1), melhor o desempenho do algoritmo (PARREIRA, 2018).

Na figura 6 é apresentada a curva ROC dos algoritmos aplicados gerada no RapidMiner com as configurações adotadas, 10 *folds* (dobras), separação de 70% para treinamento de dados e amostragem estratificada (mistura de dados buscando manter a proporção resultados da *label* na amostragem).

Figura 6 – Curva ROC dos algoritmos aplicados no conjunto de dados



Fonte: Elaborada pelo autor.

Das curvas ROC geradas com os parâmetros definidos, a da árvore de decisão é a que aparece mais a esquerda, no canto superior, o que mostra que possui sensibilidade e AUC altas, confirmando o indicado na sua comparação com as demais matrizes de confusão e indicando o seu melhor desempenho frente aos outros algoritmos aplicados.

3 CONSIDERAÇÕES FINAIS

Para este trabalho, foram aplicados algoritmos de classificação ao conjunto de dados sintético de manutenção preditiva e foi possível verificar que nas configurações propostas, embora todos eles tenham apresentado bons resultados na predição de falhas, vide matrizes de confusão, curva ROC e AUC, a árvore de decisão foi o método de melhor performance. Os valores de acurácia alcançados são elevados e isto se deve em grande parte aos parâmetros escolhidos, características dos algoritmos aplicados (classificação) e do conjunto de dados utilizado. Vale salientar que algumas configurações dos algoritmos empregados no artigo poderiam ser otimizadas com alguns ajustes, que por limitação de hardware não foram feitos.

Os resultados obtidos com algoritmos de classificação ilustram a possibilidade de emprego de RapidMiner para manutenção preditiva, já que o *software* é capaz de realizar previsões confiáveis, como mostrado com uso de algoritmos desse grupo em dados estáticos. Entretanto, como citado no artigo, em aplicações reais os dados não costumam ser estáticos para processos industriais. Deste modo, para trabalhos futuros, existe a possibilidade de aplicação de algoritmos de regressão, com o intuito de simular a predição de falhas de máquina utilizando o treinamento com base nos dados históricos e nos novos valores coletados de um equipamento.

REFERÊNCIAS

DIAMOND, Sthephanie; MARFATIA, Anuj. **Predictive Maintenance For Dummies.** IBM Limited Edition. Hoboken, NJ: John Wiley & Sons, Inc, 2013.

DOTY, Chris. **10 Common Machine Learning Algorithms You Need to Know**, 2020. Disponível em https://rapidminer.com/blog/10-machine-learning-algorithms/.
Acesso em: 21 ago. 2021

ENGEMAN. **Tipos de manutenção**. 2021? Disponível em https://blog.engeman.com.br/tipos-de-manutencao/.
Acesso em: 16 mai. 2021

IBM. **O que é manutenção preditiva?** 2019. Disponível em https://www.ibm.com/br-pt/services/technology-support/multivendor-it/predictive-maintenance.
Acesso em: 16 mai. 2021

KDNUGGETS. **Analytics, Data Mining, Data Science software/tools used in the past 12 months**, 2015**.** Disponível em https://www.kdnuggets.com/polls/2015/analytics-data-mining-data-science-software-used.html.
Acesso em: 21 ago. 2021

LUZ, Filipe. Algoritmo KNN para classificação**. Inferir**. Brasília, 21 fevereiro 2019. Disponível em: https://inferir.com.br/artigos/algoritimo-knn-para-classificacao/.
Acesso em: 28 ago. 2021

MATZKA, Stephan. **AI4I 2020 Predictive Maintenance Dataset Data Set**, 2020. Disponível em https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset.
Acesso em: 24 abr. 2021

MCKINSEY. **Manufacturing**: Analytics unleashes productivity and profitability, 2017. Disponível em https://www.mckinsey.com/business-functions/operations/our-insights/manufacturing-analytics-unleashes-productivity-and-profitability .
Acesso em: 02 mai. 2021

MITCHELL, Tom M. **Machine Learning.** Maidenhead, U.K: McGraw-Hill, 1997.

PARREIRA, Guilherme. **Curva ROC,** 2018. Disponível em https://gpestatistica.netlify.app/blog/curvaroc/.
Acesso em: 20 nov. 2021

PESSANHA, Cínthia. **Random Forest**: como funciona um dos algoritmos mais populares de ML, 2019. Disponível em https://medium.com/cinthiabpessanha/random-forest-como-funciona-um-dos-algoritmos-mais-populares-de-ml-cc1b8a58b3b4.
Acesso em: 27 out. 2021

RAPIDMINER. **Predictive maintenance**. Disponível em https://docs.rapidminer.com/9.7/server/use/web-services/predictive-maintenance.html. Acesso em: 16 mai. 2021

RAPIDMINER. **RapidMiner Studio**. Disponível em https://rapidminer.com/products/studio/. Acesso em: 16 mai. 2021

SACRAMENTO, Gabriel. **Árvore de decisão**: entenda esse algoritmo de Machine Learning, 2021. Disponível em https://blog.somostera.com/data-science/arvores-de-decisao. Acesso em: 06 out. 2021

SANTANA, Felipe. **As aplicações do Data Science e Big Data na Industria 4.0**, 2019. Disponível em https://minerandodados.com.br/as-aplicacoes-do-data-science-e-big-data-na-industria-4-0/ Acesso em: 21 ago. 2021

TOTVS. **Manutenção Preditiva: o que é, como funciona, vantagens e dicas**, 2021. Disponível em https://www.totvs.com/blog/gestao-industrial/manutencao-preditiva/. Acesso em: 01 mai. 2021

VELASQUEZ, Luiz Henrique. **Uma visão geral sobre machine learning – classificação**, 2020. Disponível em https://operdata.com.br/blog/uma-visao-geral-sobre-machine-learning/. Acesso em: 20 nov. 2021

AGRADECIMENTOS

Agradeço primeiramente a Deus e meus familiares, por todo apoio e incentivo ao longo de toda a minha jornada, principalmente nos momentos mais difíceis. Também agradeço aos colegas de sala e professores, em especial ao meu orientador pelo companheirismo, suporte, motivação e aprendizado proporcionados.

Sobre os autores:

———————————

[i] **Thiago Sales Araujo**

Possui graduação em Engenharia Elétrica pela Faculdade de Engenharia São Paulo (2014), pós-graduação em Administração de Empresas pela Fundação Getulio Vargas (2016) e pós-graduação em Automação e Controle pela Faculdade de Tecnologia SENAI “Mariano Ferraz” (2020). Tem experiência na área de Engenharia Elétrica, com ênfase em aplicação e suporte técnico para equipamentos e painéis elétricos.

[ii] **Daniel Barbuto Rossato**

Possui graduação em Engenharia Elétrica - Automação e Controle pela Universidade de São Paulo (2002), graduação em Licenciatura para Formadores da Educação Profissional pela Universidade do Sul de Santa Catarina (2010) e mestrado em Engenharia Elétrica - Sistemas pela Universidade de São Paulo (2009). Atualmente é professor da Faculdade de Tecnologia SENAI "Mariano Ferraz" em São Paulo no curso de Tecnologia em Automação Industrial. Tem experiência na área de Automação Industrial e Controle, atuando principalmente nos seguintes temas: controle, inteligência artificial e educação.
http://lattes.cnpq.br/2551200752400438