



Modelos de inteligência artificial aplicados na indústria 4.0

EDUARDO FERNANDO GONÇALVES OSÓRIO BIANCHI DA CÂMARA
outubro de 2023

MODELOS DE INTELIGÊNCIA ARTIFICIAL APLICADOS NA INDÚSTRIA 4.0

Eduardo Fernando Gonçalves Osório Bianchi Câmara

2023

Instituto Superior de Engenharia do Porto

Departamento de Engenharia Mecânica

isen

P.PORTO

MODELOS DE INTELIGÊNCIA ARTIFICIAL APLICADOS NA INDÚSTRIA 4.0

Eduardo Fernando Gonçalves Osório Bianchi Câmara

1171289

Dissertação apresentada ao Instituto Superior de Engenharia do Porto para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Mecânica, realizada sob a orientação do Doutor Hélio Cristiano Gomes Alves de Castro.

2023

Instituto Superior de Engenharia do Porto

Departamento de Engenharia Mecânica

isen

P.PORTO

AGRADECIMENTOS

Gostaria de agradecer ao Eng.^o Hélio Castro pela sua disponibilidade e orientação no desenvolvimento da tese, prontificando-se sempre em ajudar e indicando sempre o melhor caminho a seguir.

Um agradecimento especial á família pelo apoio e ajuda dada ao longo do caminho percorrido.

página propositadamente em branco

RESUMO

A indústria 4.0 trouxe uma modernização ao sistema produtivo através da integração em rede das entidades constituintes que, aliado à evolução das tecnologias de informação permitiu um aumento da produtividade, da qualidade do produto, otimização dos custos produtivos e personalização do produto às necessidades do cliente. Apesar da complexidade do pensamento humano, a inteligência artificial tenta replicá-lo em algoritmos, criando modelos capazes de processar bases de dados com um elevado volume de informação, gerando informação valiosa para a tomada de decisão, como é o caso do Machine Learning e o Deep Learning.

Na presente dissertação criou-se um modelo recorrendo á ferramenta open source Knime que com base num conjunto de dados fornecidos pela Bosch, parametrizou o modelo com diversas técnicas de pré-processamento e de seleção de recursos, permitindo desenvolver um modelo final para previsão de falhas internas dos produtos na linha de produção da Bosch.

No sentido de seleccionar o modelo mais eficiente, preciso e adequado ao problema em questão, recorreu-se ao Logistic Regression, Decisison Tree, Random Forest, XGBoost e por último o Naive Bayes. De forma a diminuir o tempo de aprendizagem do modelo, o poder computacional necessário e a possibilidade de ocorrer o overfitting analisou-se a possibilidade de serem aplicadas as seguintes técnicas de redução de dimensionalidade Correlation filter, Low Variance filter e o PCA.

O estudo demonstra que a eficácia dos modelos melhorou com a aplicação das técnicas acima descritas destacando-se o Logistic Regression aliado á técnica de seleção de recursos PCA, que obteve um Recall de 100% e uma precisão e accuracy, ambas com 99.43%.

PALAVRAS-CHAVE

Indústria 4.0; Inteligência Artificial; Machine Learning; Deep Learning; Knime.

página propositadamente em branco

ABSTRACT

Industry 4.0 has brought an improvement into the production system through the entities network integration, which combined with the evolution of information technology has enabled an increase in productivity, product quality, production costs optimization and product customization to customer needs.

Despite the complexity of human thought, artificial intelligence tries to replicate it in algorithms, creating models capable of database processing with a high information volume, generating valuable data for decision-making, as it is the case of Machine Learning and Deep Learning algorithms.

In this dissertation it was created a model using the open-source tool Knime that, based on a dataset provided by Bosch, applied several pre-processing tools and feature selection techniques into the model, allowing the development of an algorithm for internal product failure prediction at Bosch production line.

In order to select the most efficient, accurate and suitable model for the case study, it was analyzed different algorithms as Logistic Regression, Decision Tree, Random Forest, XGBoost and finally Naive Bayes. Regarding the decrease of the model learning time, computational power required and the possibility of overfitting, were analyzed the following dimensionality reduction techniques: Correlation filter, Low Variance filter and PCA.

The study shows that the model accuracy improved using the techniques described above, mainly the Logistic Regression combined with the resource selection technique PCA, which obtained a Recall of 100% and a precision and accuracy, both with 99.43%.

KEYWORDS

Industry 4.0; Artificial Intelligence; Machine Learning; Deep Learning; knime

página propositadamente em branco

ÍNDICE

ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABELAS	XI
LISTAS DE SIGLAS E SÍMBOLOS.....	XIII
1. INTRODUÇÃO	15
1.1. Contextualização	15
1.2. Objectivos.....	15
1.3. Metodologia	16
1.4. Estrutura do Relatório	16
2. REVISÃO BIBLIOGRÁFICA.....	17
2.1. Contextualização da Indústria 4.0	17
2.1.1. Pilares da Indústria 4.0.....	19
2.1.2. Vantagens e Desvantagens da Indústria 4.0.....	21
2.2. Inteligência Artificial.....	21
2.2.1. Machine Learning.....	23
2.2.1.1. Tipos de aprendizagem de Machine Learning.....	25
2.2.1.1.1. Aprendizagem supervisionada	25
2.2.1.1.2. Aprendizagem Não Supervisionada	26
2.2.1.1.3. Aprendizagem Semi-Supervisionada.....	27
2.2.1.1.4. Aprendizagem por Reforço	27
2.2.1.2. Modelos de Machine Learning.....	27
2.2.1.3. Vantagens e Desvantagens	31
2.2.2. Deep Learning	33
2.2.2.1. Modelos de Deep Learning	34
2.2.2.2. Vantagens e Desvantagens	36
2.2.3. Modelos Híbridos de Inteligência Artificial: Metaheurísticos.....	38
2.3. Aplicações de Inteligência Artificial.....	39
2.3.1. Aplicações de Machine Learning.....	39
2.3.2. Aplicações de Deep Learning	40
2.3.3. Aplicações dos Modelos Híbridos de Inteligência Artificial	40
3. Desenvolvimento	42
3.1. Desenvolvimento do Caso de Estudo de uma base de dados da Bosch	42
3.1.1. Descrição da base de dados Bosch	42
3.1.2. Software Knime.....	44
3.1.3. Métricas de avaliação dos modelos.....	45
3.1.4. Pré-processamento de dados	47
3.1.5. Desafios associados ao desenvolvimento dos modelos	48
3.1.6. Definição dos dados de amostra por Resampling.....	49

3.1.6.1. Hold-out.....	49
3.1.6.2. Validação cruzada K-fold	49
3.1.7. Fases relevantes para a construção de um modelo de ML.....	50
3.2. Construção e evolução dos modelos.....	51
3.2.1. Fluxograma dos Modelos de Classificação.....	51
3.2.2. Classificação supervisionada da conformidade do produto de uma linha de produção 52	
3.2.2.1. Análise dos modelos.....	55
3.2.2.2. Impacto do Feature Selection e redução da dimensionalidade dos dados no modelo 58	
3.2.2.2.1. Seleção dos recursos aplicando o filtro da Linear Correlation.....	59
3.2.2.2.2. Seleção dos recursos aplicando o filtro Low Variance	61
3.2.2.2.3. Seleção dos recursos aplicando uma redução das dimensões com a técnica PCA (Principal Componente Analysis).....	61
3.3. Discussão de resultados	63
4. Conclusão	64
4.1. Conclusões finais	64
4.2. Limitações e trabalhos futuros.....	65
REFERÊNCIAS BIBLIOGRÁFICAS	67

página propositadamente em branco

ÍNDICE DE FIGURAS

Figura 1 - As revoluções industriais [5]	17
Figura 2- Entidades da Indústria 4.0 [5]	19
Figura 4- História do desenvolvimento da IA (Adaptado de [15]).....	22
Figura 5- Diagrama das áreas integrantes da IA e da ciência da informação [3]	24
Figura 6- Processo de recolha de dados em área industrial usando o ML [26]	25
Figura 7- Fluxograma da aprendizagem supervisionada [27]	26
Figura 8- Fluxograma do processo de aprendizagem por reforço [27].....	27
Figura 9- Exemplo de uma rede neural artificial [84].....	34
Figura 10- Histograma da quantidade de produtos conformes e não conformes.....	43
Figura 11- Exemplo da base de dados da Bosch	43
Figura 12- Representação das várias funções do software Knime	44
Figura 13- Esquema explicativo do exemplo de um nó do software Knime.....	45
Figura 14- Sequenciamento de etapas constituintes do pré-processamento	47
Figura 15- Etapas do pré-processamento dos dados no KNIME	48
Figura 16- Exemplo de uma validação cruzada K-fold	50
Figura 17- Etapas da construção do modelo.....	51
Figura 18- Fluxograma dos modelos desenvolvidos no Knime	52
Figura 19- <i>Workflow</i> do desenvolvimento dos modelos	53
Figura 20- Gráfico de dispersão das primeiras 20.000 ocorrências.....	54
Figura 21- Gráfico de dispersão das primeiras 20.000 ocorrências após a normalização dos dados	54
Figura 22- Parâmetros de ajuste do nó <i>Smote</i>	55
Figura 23- Representação da árvore de decisão do modelo.....	57
Figura 24- Estatísticas do modelo <i>Logistic Regression</i>	58
Figura 25- Matriz de correlação dos recursos.....	59
Figura 26- Aplicação de filtro aos recursos com base na correlação	60
Figura 27- Redução da dimensionalidade usando o PCA (28 dimensões)	62

página propositadamente em branco

ÍNDICE DE TABELAS

Tabela 1- Características, vantagens e desvantagens da Indústria 4.0.....	21
Tabela 2- Modelos de ML.....	28
Tabela 3- Vantagens e desvantagens dos modelos de ML	31
Tabela 4- Modelos de DL.....	34
Tabela 5- Vantagens e Desvantagens dos Modelos de DL.....	36
Tabela 6- Algoritmos Metaheurísticos	38
Tabela 7- Casos de estudo com aplicações de ML.....	39
Tabela 8- Casos de estudo com aplicações de DL.....	40
Tabela 9- Casos de estudo com aplicações de modelos híbridos de IA.....	40
Tabela 10- Matriz de confusão de um modelo de ML	45
Tabela 11- Desempenho das métricas de avaliação dos modelos.....	56
Tabela 12-Desempenho das métricas de avaliação dos modelos.....	60
Tabela 13- Desempenho das métricas de avaliação dos modelos.....	61
Tabela 14- Desempenho das métricas de avaliação dos modelos.....	62

página propositadamente em branco

LISTAS DE SIGLAS E SÍMBOLOS

Lista de Siglas

ANN	Artificial Neural Network
CC	Cloud Computing
CNN	Convolucional Neural Networks
CPS	Cyber-Physical System
DL	Deep Learning
DT	Digital Twin
GA	Genetic Algorithm
GWO	Grey Wolf Optimizer
IA	Inteligência Artificial
IoT	Internet of Things
KNN	K-Nearest Neighbors
ML	Machine Learning
MLR	Multiple Linear Regression
PCA	Principal Component Analysis
RF	Random Forest
SVM	Support Vector Machine
SVR	Support Vector Regression
TIC	Tecnologia de Informação e Comunicação

página propositadamente em branco

1. INTRODUÇÃO

A presente dissertação tem como objetivo explicar a importância da criação de modelos de inteligência artificial no âmbito na indústria 4.0 e o impacto positivo criado por estas técnicas. No desenvolvimento desta dissertação pretende-se explicar os diferentes modelos e técnicas de ML e DL existentes, o modo como podem ser aplicados a bases de dados industriais de forma a criar modelos preditivos. O principal foco desta dissertação é o desenvolvimento de um modelo de previsão da qualidade do produto de uma linha de produção da Bosch, com recursos a diferentes modelos de ML e técnicas de seleção de recursos de forma a criar o modelo mais eficaz e eficiente

1.1. Contextualização

A Indústria 4.0 é um conceito de produção cada vez mais conhecido, assente em pilares como “IoT” (Internet of Things) e Smart Factory, promovendo um processo produtivo muito mais eficiente, flexível, personalizável e rastreável. O caminho da digitalização da indústria, levou a que fossem gerados um grande conjunto de dados caracterizadores do processo produtivo (Big Data), que serão tratados a partir de técnicas de Machine Learning (ML). O tratamento destes dados conduz a um maior controlo de qualidade do produto, uma avaliação eficaz dos riscos, uma otimização dos custos produtivos, definição de um plano de manutenção preditiva e prevenção das falhas [1].

Esta filosofia introduz um novo conceito de desenvolvimento de produto, no sentido da customização do produto de acordo com as necessidades do cliente, ao contrário do que acontece atualmente, em que a produção em série permite ter um baixo custo unitário de produção, porém menor versatilidade na alteração ao produto, dificultando a adaptação às necessidades do cliente que se encontram em constante mudança [2].

A inteligência artificial (IA) está a criar um impacto bastante positivo em vários setores devido à sua rápida velocidade de resposta e elevada capacidade de generalização. O ML, um subcampo da IA, fornece relações entre as variáveis de entrada e prevê as variáveis de saída, indicando tomadas de decisão apropriadas. A quantidade de dados exigida para o treino de um algoritmo de ML é elevada, constituindo uma limitação desta tecnologia [3].

Um caso particular do ML é o Deep Learning (DL) que aplica várias etapas de transformações de dados para serem eficazes na extração de informações de grandes bases de dados [4]. O DL utiliza estruturas de redes neurais artificiais para conseguir através de dados de entrada agrupar os dados em várias camadas. Os algoritmos de aprendizagem profunda realizam operações mais complexas, partindo um problema em subproblemas e é capaz de fornecer resultados mais precisos do que o ML [3].

1.2. Objectivos

O principal objetivo desta dissertação é o desenvolvimento de um modelo de previsão de falhas dos produtos da Bosch, tendo por base uma revisão bibliográfica preparatória, que inclui a aplicação de modelos de ML a casos de estudo reais. Para além deste objetivo principal, pretende-se atingir os seguintes objetivos suplementares:

- Conhecer a filosofia subjacente à indústria 4.0;

- Estudar a aplicação da IA no contexto industrial;
- Compreender os modelos associados ao IA, seja de ML, seja de DL;
- Elaborar o pré-processamento de base de dados da Bosch;
- Desenvolvimento de um modelo final de ML para a previsão de falhas dos produtos de uma linha da Bosch;
- Compreender os efeitos da utilização das técnicas de seleção de recursos nos modelos de ML.

1.3. Metodologia

Este relatório divide-se em duas componentes principais: o conhecimento da tecnologia associada e o estudo da sua aplicação a casos práticos.

O conhecimento da tecnologia está ligado à pesquisa bibliográfica realizada através da análise de vários artigos científicos contemporâneos referentes ao conjunto de palavras-chave: Indústria 4.0, Inteligência Artificial, Machine Learning e Deep Learning. Após a escolha de um caso de estudo aplicou-se diferentes modelos e técnicas para o tratamento dos mesmos, procurando obter a melhor otimização do poder computacional e da eficiência.

1.4. Estrutura do Relatório

O relatório está dividido em quatro capítulos que se apresentam de seguida.

No primeiro capítulo é realizado uma introdução ao trabalho desenvolvido, subdividido em introdução, contextualização, metodologia e estrutura do relatório.

No segundo capítulo é feito um enquadramento teórico de base ao trabalho, focando em temas fulcrais como a indústria 4.0 e a inteligência artificial.

No terceiro capítulo é apresentado todo o desenvolvimento desde o processamento da base de dados, execução de técnicas de pré-processamento, a aplicação de diferentes modelos e comparação dos resultados dos mesmos.

No quarto capítulo é exposto a discussão dos resultados bem como os trabalhos futuros a realizar.

2. REVISÃO BIBLIOGRÁFICA

Neste capítulo vai ser apresentada uma pesquisa bibliográfica com um enquadramento teórico para o desenvolvimento do caso de estudo prático, que tem como pontos fulcrais a Inteligência Artificial e a Indústria 4.0.

2.1. Contextualização da Indústria 4.0

Os métodos tradicionais de produção estão a ficar desadequados, no sentido que é necessário processos de produção mais dinâmicos e que atendam às necessidades dos clientes, pelo que está a ser implementado alta tecnologia na indústria, como sensores, softwares e conectividade sem fios (IoT), caminhando para a criação de fábricas inteligentes [5]. O conceito de “fábrica inteligente” surgiu com a quarta revolução industrial, indústria 4.0, referido pela primeira vez na Alemanha em 2011 e que envolve a automação inteligente, automatização, autoconfiguração e previsão, em que os métodos de ML e IA são amplamente aplicados [5,6].

Com esta revolução existiu a procura de uma digitalização do processo, assente na recolha e análise de dados em tempo real, com a finalidade de caminhar no sentido da customização do produto e flexibilidade da cadeia produtiva, através da utilização de máquinas inteligentes, mantendo a eficiência do ciclo produtivo. O poder da informação resultante do processamento da base de dados por técnicas de IA, transforma a cadeia de valor atual, existindo uma melhoria contínua decorrente do processamento da informação. A Figura 1 demonstra a evolução significativa que a indústria sofreu ao longo dos séculos, melhorando consecutivamente os seus processos [5]. A primeira revolução industrial começou com a exploração do vapor de água como energia motora no processo produtivo, enquanto a segunda trouxe a utilização da linha de montagem e da energia elétrica na produção. A terceira trouxe uma transição para a automação do processo produtivo, e por fim surge a indústria 4.0 com o poder da informação como auxílio à tomada de decisão [5,7,8].

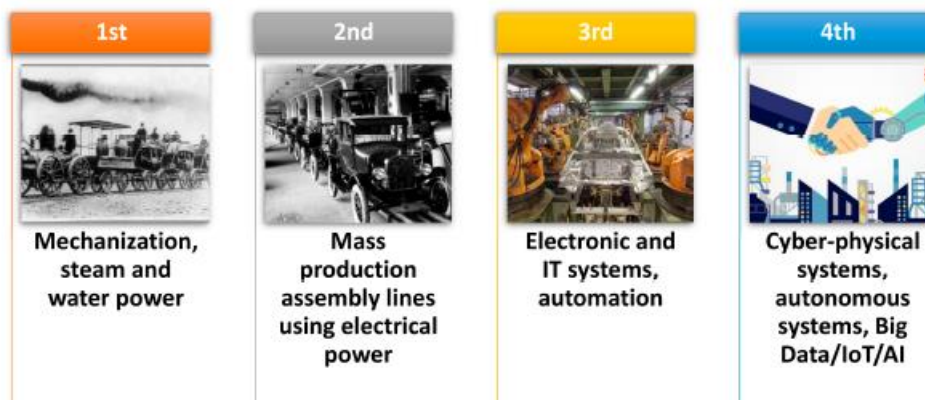


Figura 1 - As revoluções industriais [5]

Esta revolução industrial trouxe uma modernização na indústria da produção havendo uma criação de novos sistemas, protocolos de comunicação, segurança cibernética, métodos de IA, maior utilização de monitores em interfaces homem-máquina e poder computacional cada vez maior e mais compacto. Com isto o setor da produção adotou novas tecnologias digitais nos processos, desde a fase de design do produto até à gestão do ciclo de vida do produto.

A digitalização da indústria visa tornar os sistemas de produção tradicionais em sistemas capazes de produzir de forma customizada e versátil, dando resposta aos requisitos do mercado [9]. A indústria 4.0 é uma transformação digital orientada por Tecnologia de Informação e Comunicação que melhora a correlação da máquina e do homem, promovendo a melhoria da eficiência e da produtividade baseada na automação alimentada por dados em tempo real [10]. Os processos atuais são estáticos e hierárquicos, o que motiva mudanças disruptivas e custos associados a essas mudanças [9]. O verdadeiro potencial da indústria 4.0 é a conexão estabelecida entre o Cyber-Physical System e as máquinas que integram a cadeia de valor, possibilitando a tomada de decisão e resolução de problemas sem intervenção humana [11].

Na indústria 4.0 pode-se definir três níveis de implementação da tecnologia, segundo o prisma da produção: integração vertical, horizontal e circular.

- Integração vertical – agrega diversos temas de tecnologia de informação e comunicação (TIC) em níveis hierárquicos diferentes, desde os mais básicos, como sensores e atuadores, até aos níveis mais altos de gestão de produção.
- Integração horizontal – diz respeito à integração das tecnologias TIC nos mecanismos envolvidos nas etapas dos processos de produção com a troca de informação dentro da empresa e entre empresas (redes de valor).
- Integração circular – junta os dois níveis anteriores, completando a conexão do utilizador final e do ciclo de vida do produto, encerrando o ciclo de produção com a respetiva digitalização de todo o processo [9].

A inclusão de uma grande quantidade de informação digitalizada permite a obtenção de dados sobre consumo por produto, e por isso, redefinir a sua cadeia de valor de forma a ir de encontro às necessidades do cliente. A satisfação do cliente é um processo que nunca termina e exige mudança a todo o momento, isto é, a procura contínua de satisfação das exigências do cliente, uma vez que estas não são estacionárias no tempo. As empresas estendem a sua oferta ao modelo de negócio digital, fornecendo aos seus clientes soluções digitais que melhor os sirvam [12].

O surgimento da indústria 4.0 conduziu a algumas mudanças na distribuição de tarefas entre o homem e a máquina, no sentido em que as máquinas devem auxiliar o ser humano, não só no fabrico, mas também na tomada de decisão, existindo uma comunicação constante entre estes dois elementos. Esta cooperação entre o homem e a máquina deve caminhar para a aprendizagem mútua através da identificação de novos padrões de aprendizagem em diversos cenários industriais digitalizados. Um dos maiores desafios desta nova filosofia de produção é a definição da divisão do trabalho entre o homem e as máquinas, num ambiente tecnológico.

A comunicação entre máquinas garantida pela IoT criou novas soluções de autoaprendizagem, levando a uma diminuição do custo de comunicação e a um aumento da flexibilidade da produção, dos recursos de personalização, do fluxo produtivo e da qualidade dos produtos [13].

2.1.1. Pilares da Indústria 4.0

A indústria 4.0 promove a criação de ambientes inteligentes e autónomos em vários setores da indústria, baseando-se em tecnologias inovadoras de informação e comunicação, como *Cyber-Physical System (CPS)*, *IoT* e *Cloud Computing (CC)*. Os CPS resultam da conexão dos elementos físicos com os cyber elementos, apresentados esquematicamente na Figura 2. Os elementos físicos incluem componentes como sensores, painéis de operação e computadores que comunicam entre si e fornecem dados aos cyber elementos, onde ocorre o tratamento, processamento e a tomada de decisão [14]. Para além destes componentes é importante referir a IA, a robótica, o *Big Data*, a realidade virtual, *Digital Twin (DT)* e a cibersegurança [5].

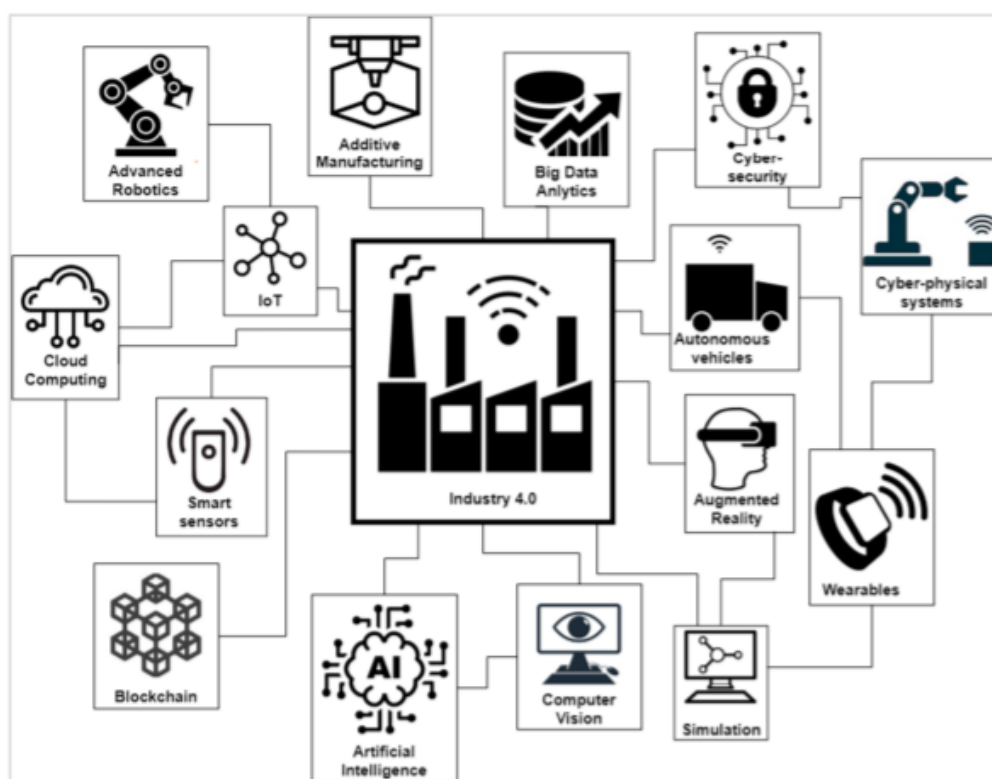


Figura 2- Entidades da Indústria 4.0 [5]

O CPS é um sistema que é controlado por algoritmos, com integração próxima entre o hardware e o software, tendo como objetivo definir a melhor decisão assente num conjunto de dados de entrada [5]. Os sistemas ciberfísicos são integrados pela computação, rede e processos físicos, em que os computadores recorrem a circuitos de feedback para a monitorização e regulação dos processos físicos [6], assente no princípio 5 C's: conexão, conversão, cibernética, cognição e configuração [7]. Atualmente a condução autónoma e a produção robótica são dois exemplos da aplicação do CPS. Na produção robótica todos os dispositivos estão conectados à rede e comunicam entre si, no sentido de ajudar o fluxo de trabalho, a autonomia, a eficiência, a funcionalidade e a segurança [5,15].

A IoT assume um papel central na indústria 4.0, e é definida como sendo a interação entre as tecnologias, integrando a comunicação entre os objetos inteligentes para monitorizar o desempenho dos dispositivos [5]. Os dispositivos estão conectados à rede, e estes constantemente digitalizam a informação recolhida pelos seus sensores, permitindo o acompanhamento em tempo real, aumentando a eficiência do fluxo de trabalho [10,15]. Os dispositivos são equipados com tecnologia de identificação para detetar, monitorizar e recolher os dados necessários [5].

O *Big Data* refere-se ao grande volume de dados que são gerados pelos sensores das máquinas, armazenados numa plataforma *cloud*, e posteriormente vão ser processados no sentido de ser possível extrair informação valiosa relativa ao sistema produtivo [5]. As soluções do *Big Data* vão lidar com um grande conjunto de dados e entram em ação para os analisar [6]. De forma resumida, o *Big Data* pode ser caracterizado pela representação dos 5 V's: volume, velocidade, variedade, veracidade e valor [5,11,16].

A IA procura simular a inteligência humana em máquinas programadas para desenvolver uma linha de pensamento semelhante à do humano, englobando vários domínios como a robótica, análise do *Big Data*, ML, *computer vision* e processamento de linguagem natural [5]. A IA permite que as máquinas funcionem de uma maneira autónoma, tomando as suas decisões de forma mais eficiente [6]. O *computer vision* captura e analisa informações visuais utilizando câmaras, com o objetivo final de representar o mundo a partir de imagens [5]. A IA funciona como o motor da digitalização e a tendência é ser cada vez mais desenvolvida e comum nos próximos anos [15].

A robótica desempenha atualmente um papel importante na vida humana, sobretudo na produção industrial, na saúde e nos transportes, caracterizando-se pela capacidade de agir a estímulos externos, deteção e raciocínio lógico [5].

A *cloud computing* está implícita em muitos componentes da indústria 4.0, tal como na *IoT*, *Big Data* e *computer vision*. A *cloud* contém recursos que incluem ferramentas e aplicativos, como armazenamento de dados, servidores, bancos de dados, rede e *software* [5,10].

A realidade virtual e aumentada permite a visualização de um ambiente de realidade virtual podendo integrar ambientes industriais em que no momento é possível redesenhar, testar e aperfeiçoar uma realidade em simulação virtual.

O DT é a representação virtual de um produto físico, contendo informações sobre o produto e capacidade para acompanhamento do ciclo de vida do mesmo. Este é composto por um produto físico e uma representação virtual do produto, caracterizado pela possibilidade de monitorização e controlo remoto em tempo real, manutenção preditiva, personalização de produtos e simulação de ambientes alternativos [5].

A cibersegurança desempenha uma tarefa de extrema responsabilidade, pois terá de evitar possíveis ataques que podem ter consequências ações das identidades constituintes do CPS. A conectividade em rede e as diversas variedades das identidades integrantes do CPS aumentam a complexidade de gestão da segurança [5] e também a sua vulnerabilidade face à exposição em rede [10].

Todas estas tecnologias apresentam pontos positivos, porém demonstram também algumas ameaças como: exposição da linha produtiva para ataques externos, necessidade de credenciais de acesso à informação, possibilidade de adulteração dos algoritmos de IA com finalidades indevidas e utilização das plataformas de simulação com propósitos errados [5].

Atualmente, existe a capacidade de lidar com grandes quantidades de dados devido ao crescimento do poder computacional e ao aparecimento do *cloud computing* [12]

2.1.2. Vantagens e Desvantagens da Indústria 4.0

Na Tabela 1 são apresentadas as características da indústria 4.0, bem como as suas principais vantagens e desvantagens.

Tabela 1- Características, vantagens e desvantagens da Indústria 4.0

Características	Referência
Permite a conexão e a comunicação entre os humanos e as fábricas inteligentes;	[5],[7]
Criação de um modelo de simulação de uma indústria;	[5]
Permite a recolha e a análise de dados em tempo real;	[5], [10]
Flexibilidade para responder às necessidades dos clientes;	[10]
Personalização em massa.	[5]
Vantagens	
Redução dos custos;	[5], [10]
Elevada eficiência;	
Ambiente competitivo e estratégico;	
Inovação da produção;	
Lucro;	
Maior qualidade do produto;	
Maior proximidade com o cliente;	
Monitorização em tempo real;	
Maior controlo do processo	
Desvantagens	
Mão de obra altamente qualificada;	[5]
Cibersegurança;	[5],[10] [7]
Elevado custo de investimento;	[5]
Funcionários apreensivos com a mudança;	[5]
Para alcançar o sucesso, é necessária uma implementação da indústria 4.0 na totalidade do sistema.	[5],[10]

2.2. Inteligência Artificial

O termo inteligência artificial foi criado pela primeira vez em 1956 por McCarthy na Universidade de Dartmouth, e tem a capacidade de replicar o pensamento humano nas máquinas, sendo uma área que evoluiu sobretudo na última década [17,18]. Esta tecnologia utiliza uma combinação da engenharia e da matemática, englobando qualquer técnica que permita às máquinas simular o comportamento humano, para atingir melhores resultados. Nos últimos tempos surgiram enormes volumes de dados em vários formatos, gerados de forma rápida, e isto pressionou o desenvolvimento de novas tecnologias, como o aumento da capacidade de processamento computacional e o desenvolvimento de novas técnicas de IA [19].

A IA tenta simular a inteligência humana e as máquinas são programadas para pensarem como seres humanos, funcionando como o motor da digitalização e a tendência é ser cada vez mais

desenvolvida e comum nos próximos anos. Na Figura 3 está representada a história do desenvolvimento da IA, bem como alguns dos acontecimentos marcantes nesta área, como a criação das primeiras redes neurais, o nascimento do DL e da análise de imagem e futuramente desenvolvimento da consciência humana [15].

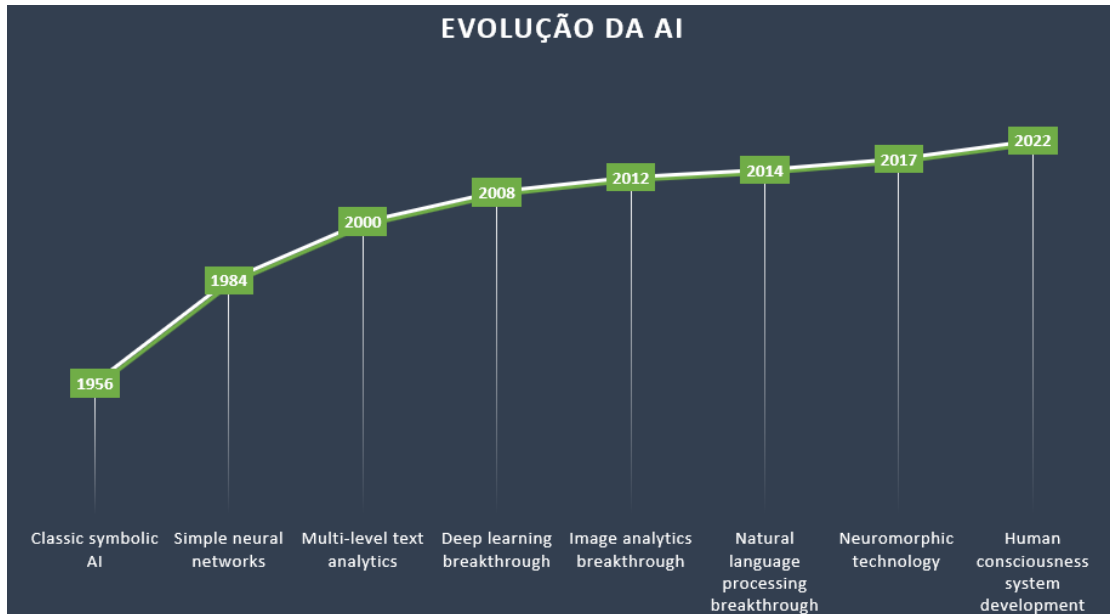


Figura 3- História do desenvolvimento da IA (Adaptado de [15])

A inteligência artificial está a revolucionar algumas indústrias, como a indústria transformadora, o retalho e as telecomunicações, onde se consegue alcançar aumentos de rentabilidade, eficiência e segurança, recorrendo ao ML, a sistemas baseados no conhecimento, visão computadorizada, robótica e otimização [18]. A aplicação da inteligência artificial não se limita às áreas anteriormente referidas, encontrando-se sob estudo a sua utilização no setor da construção, uma das indústrias com menor uso das ferramentas digitais, levando a custos por ineficiência, atrasos nos projetos, baixa qualidade no produto, decisões com reduzida informação de base e indicadores de produção e segurança baixos [20]. A IA engloba as ciências de computação, lógica, biologia e psicologia, tendo alcançado resultados em diferentes aplicações, como o reconhecimento da fala, o processamento de linguagem natural e robots inteligentes [17].

As técnicas de IA podem gerar análise descritiva, análise preditiva, análise de diagnóstico e análise prescritiva [12]. O principal objetivo do modelo de análise descritiva é resumir os dados para revelar os padrões desconhecidos que estão presentes nestes, deduzindo estatísticas acerca dos dados gerados para deteção de eventos de interesse e situações críticas dos dados monitorizados [9].

A análise preditiva usa modelos estatísticos para prever o tamanho de uma produção ou a vida útil restante das máquinas [12], que dependem dos dados de eventos de falhas passadas. A partir do histórico da máquina, um algoritmo de ML vai aprender padrões que relacionam os dados capturados do ativo monitorizado com uma variável de destino, que identificará a falha que vai ser prevista [9].

A análise de diagnóstico deteta e reporta falhas nos equipamentos e no produto [12], enquanto que a análise prescritiva propõe ações para melhorar o processo e prevê o impacto da implementação destas ações [9].

A IA apresenta aplicações em diversas cadeias produtivas que tenham um bom suporte de informação como finanças, saúde, indústria automóvel e retalho [17], em que alguns exemplos são:

- Detecção de equipamentos a partir do qual os clientes fazem login nas suas contas bancárias;
- Previsão da quantidade de dinheiro que deve ser deixada em caixa ATM, de forma a não deixar dinheiro parado que pode estar a ser investido e rentabilizado;
- A base de dados dos utentes pode promover informação valiosa para correlações de causas de doenças;
- Redução do custo de inventário numa linha produtiva com a previsão do que se vai consumir;
- No setor das compras online, apresenta destaque nos produtos que o consumidor procura e necessita com base no histórico e nas tendências dos clientes semelhantes [12].

2.2.1. Machine Learning

A IA engloba o desenvolvimento de sistemas computacionais com a capacidade de realizar tarefas que requerem inteligência humana, como perceção visual, reconhecimento de fala e tomada de decisão. Esta área abrange vários subcampos como o ML e DL, que permitem realizar tarefas de classificação, regressão, *clustering* ou redução da dimensionalidade de um grande volume de dados de entrada de alta dimensão [21, 22].

O ML é um campo da IA que estuda os algoritmos capazes de aprenderem de forma autónoma, a partir de uma base de dados [22]. Com o decorrer do tempo o ML começou a ser adaptado de forma a poder integrar o campo industrial permitindo evoluir no sentido da indústria 4.0 [2].

As técnicas de ML têm-se mostrado bastante eficazes na análise e resolução de problemas complexos, que através de algoritmos executam tarefas de previsão, diagnóstico, planeamento e reconhecimento com base em dados históricos. O processo de tomada de decisão definida em cada algoritmo vai sendo moldada pelos dados de treino, recolhidos a partir de sensores e impostos ao programa para que este se ajuste à saída pretendida. A quantidade e a qualidade dos dados recolhidos para a base de dados afetam diretamente a precisão da tomada de decisão do modelo de ML [1].

A partir do processamento da informação, é possível a extração de dados relevantes dos ativos monitorizados, adotando um tratamento inteligente aos dados por aplicação de técnicas ML levando a um maior controlo da qualidade dos produtos. A baixa qualidade dos produtos tem um elevado impacto a nível económico e ambiental, uma vez que os recursos e a mão-de-obra são desperdiçados, diminuindo a sustentabilidade do processo [8,9]. Este assunto refere-se ao conceito de Zero Defect Manufacturing (ZDM), e a sua implementação foi impulsionada pela evolução da

indústria 4.0, pois trouxe a capacidade de gerir uma grande quantidade de dados para a aplicação de técnicas de ML [9,10].

O ZDM é um conceito que se foca na qualidade do processo e dos produtos, reduzindo os defeitos por meio de técnicas corretivas, preventivas e preditivas, integrando técnicas de análise de dados para que nenhum produto defeituoso saia da linha de produção e seja entregue ao cliente. Esta abordagem foca-se na prevenção da ocorrência de defeitos através da melhoria contínua do processo e da utilização de técnicas de controlo de qualidade, enquanto que os modelos ML pretendem identificar padrões nos dados recolhidos, prevendo potenciais defeitos. Posto isto, os modelos ML são utilizados como ferramenta de apoio à implementação dos princípios ZDM, permitindo que as indústrias tomem melhores decisões, e melhorem os seus processos e, conseqüentemente, a qualidade dos produtos [23].

O desenvolvimento de modelos de previsão está a tornar-se cada vez mais útil no mundo industrial, sendo que os principais objetivos é prever o comportamento anormal de máquinas, ferramentas e processos industriais, antecipando o surgimento de eventos críticos e danos, de modo a prevenir eventuais perdas económicas [9].

A Figura 44 mostra que a IA e a estatística se combinam dando origem ao ML como nova forma de tratamento de dados, dentro do qual surgem áreas como as redes neurais (DL), algoritmos evolutivos e aprendizagem por reforço [3].

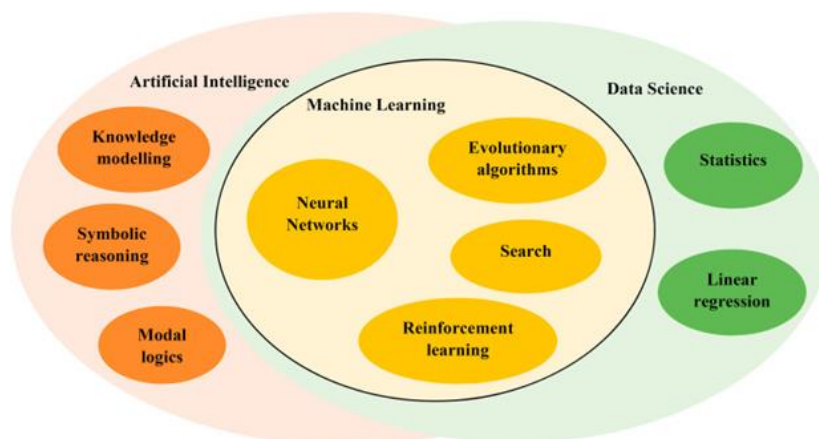


Figura 4- Diagrama das áreas integrantes da IA e da ciência da informação [3]

Com a necessidade de melhorar a monitorização do processo produtivo surgiram novos tipos de sensores com um preço acessível, o que permite que haja um grande crescimento da quantidade de dados recolhidos na indústria. A geração de grandes volumes de dados cria a necessidade de tratamento dos mesmos, surgindo novas ferramentas para estudar o potencial destes dados como o ML.

A Figura 5 representa o fluxo de trabalho típico para extração de dados, recorrendo ao ML, começando na recolha dos dados industriais com os diversos sensores, processamento, transformação, correlação e seleção da informação relevante [26].

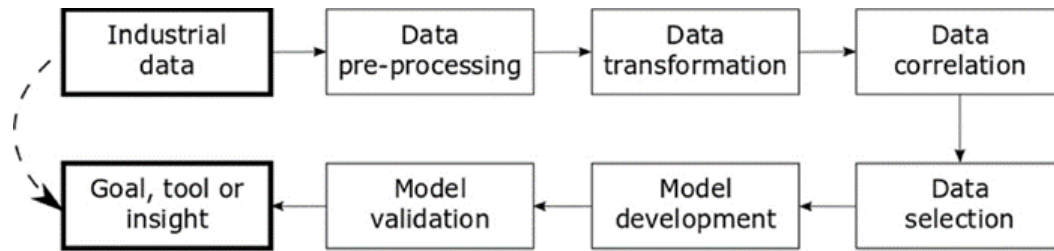


Figura 5- Processo de recolha de dados em área industrial usando o ML [26]

2.2.1.1. Tipos de aprendizagem de Machine Learning

As técnicas de ML permitem a deteção automática de padrões nos dados sem terem de ser explicitamente programados, tomando decisões de forma mais eficiente. O desenvolvimento de algoritmos de ML exigiu meios para mapear o conhecimento adquirido desde o processo de aprendizagem até à previsão, impulsionando o desenvolvimento de abordagens categorizadas [19]. O ML apresenta quatro tipos de aprendizagem que são a aprendizagem supervisionada, a aprendizagem não supervisionada, a aprendizagem semi-supervisionada e a aprendizagem por reforço [2, 14, 24, 25].

2.2.1.1.1. Aprendizagem supervisionada

Os algoritmos de aprendizagem supervisionada são treinados a partir de uma base de dados rotulados, isto é, as entradas (inputs) e as saídas (outputs) são conhecidas pelo algoritmo [1,14]. O executante do treino do algoritmo desempenha um papel importante, uma vez que vai rotular as saídas dos dados, selecionar os algoritmos e os parâmetros de controlo adequados. A aprendizagem do algoritmo advém da comparação das saídas dos dados com a previsão efetuada, identificando o diferencial. Após a execução associada a uma quantidade substancial de dados, é esperado que o modelo seja capaz de fazer previsões (outputs) de dados não rotulados [1], de forma eficaz e precisa [26].

Este tipo de aprendizagem, também denominada de aprendizagem preditiva, inclui alguns destes algoritmos: Neural Networks, Support Vector Machines, Decision Trees, Logistic Regression e Naive Bayes Classifiers. Este tipo de métodos utilizam um conjunto de dados de treino para treinar os valores de mapeamento através da minimização do erro entre o vetor de saída dos dados de treino e o vetor de saída previsto pelo algoritmo, conforme apresentado no fluxograma da Figura 6 [2, 17]. Quando é atingindo um nível aceitável de precisão, a aprendizagem é interrompida e o modelo é preparado para fazer previsões a partir de novos dados nunca vistos. O ML, quando utilizado para classificação, categoriza os dados dividindo-os em várias classes, estimando as probabilidades de classes com um conjunto de dados de treino. Quando utilizado para previsão, o ML fornece uma capacidade preditiva para a tomada de decisões [29].

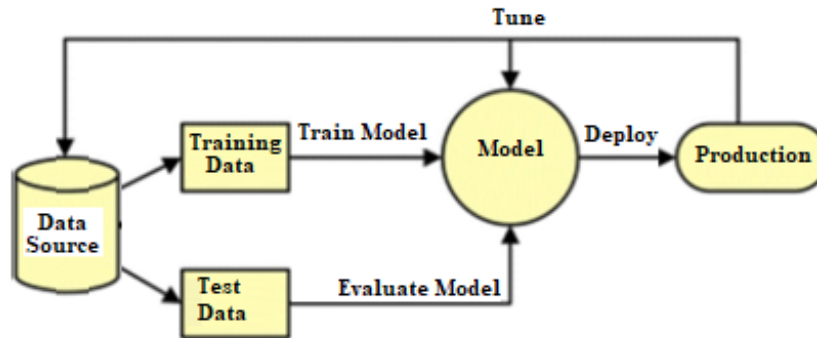


Figura 6- Fluxograma da aprendizagem supervisionada [27]

Quando as variáveis de saída são discretas, o problema é conhecido como tarefa de classificação, como por exemplo um modelo para detetar falhas de processo ou prever o nível de qualidade em novos lotes de produção. Quando as variáveis são contínuas, o problema é conhecido como tarefa de regressão, e um exemplo disto pode ser prever uma determinada propriedade física, como a espessura ou rugosidade da superfície de itens processados por máquinas de controlo numérico [2].

2.2.1.1.2. Aprendizagem Não Supervisionada

A aprendizagem não supervisionada utiliza dados não-rotulados, em que os outputs são desconhecidos [14, 27], pelo que de forma a identificar padrões das variáveis de entrada são gerados grupos de dados interligados, para obtenção de informações significativas [29]. Esta aprendizagem é usada quando as relações das variáveis de entrada são desconhecidas, não existindo diferença entre os dados de treino e os do modelo, isto é, o algoritmo terá de ser capaz de reconhecer padrões nos dados de treino, classificar corretamente os dados de teste e deduzir uma solução final correta [1]. Quando são introduzidos novos dados, os algoritmos usam os recursos apreendidos anteriormente para reconhecer a classe dos dados e são usados para agrupamento e redução de recursos [27].

Este tipo de aprendizagem, em termos gerais, pode ser dividido em três subáreas: clustering, estimativa de densidade e redução da dimensão. O clustering tem como tarefa agrupar um conjunto de objetos, de forma a que os mais semelhantes se encontrem no mesmo grupo. A estimativa da densidade apresenta um conjunto de técnicas que poderão ser usadas para descobrir propriedades úteis, como a multimodalidade e assimetria, e para fazer uma estimativa de uma função de densidade de probabilidade subjacente a um conjunto de dados. A abordagem mais comum para estimar a densidade são os histogramas redimensionados. A redução da dimensão é fulcral para a análise do Big Data, pois permite que haja uma compactação dos dados sem comprometer o conteúdo dos mesmos [2].

A análise de Principle Component Analysis (PCA) é o algoritmo mais usado para fins de monitorização [14, 2]. Este tipo de aprendizagem utiliza a Exploratory Data Analysis (EDA), que é um tipo de análise dados que usa a análise cluster para identificar vários padrões ou grupos de dados [27].

2.2.1.1.3. Aprendizagem Semi-Supervisionada

Enquanto a aprendizagem supervisionada é dispendiosa pela quantidade de dados rotulados necessários para que o modelo possa gerar resultados precisos, a aprendizagem não-supervisionada, recorrendo a dados não-rotulados, torna-se dependente da existência de padrões nas variáveis de entrada para obter precisão na resposta [4]. Para solucionar isto, os algoritmos de aprendizagem semi-supervisionada treinam um modelo com dados rotulados e não rotulados, fornecendo uma maior exatidão comparativamente com o modelo supervisionado que utiliza dados rotulados em quantidade limitada pelo seu custo [4, 24].

2.2.1.1.4. Aprendizagem por Reforço

Neste tipo de aprendizagem, os agentes executam algumas ações e recebem feedback, podendo ser negativo ou positivo, sendo o modelo atualizado de acordo com isso. Pode-se afirmar que os algoritmos de reforço aprendem através de tentativa/erro, ajustando-se para promover ações com feedback positivo. Este tipo de modelo é atualmente utilizado na condução autónoma e nos jogos online [1, 14].

Este tipo de aprendizagem não necessita de um conjunto de dados pré-existentes, mas aprende a tomar decisões de uma forma autónoma, analisando as consequências das suas ações através de um sistema de recompensa, conforme representado na Figura 7 [2,24,27]. O *Q-learning* é um dos algoritmos de aprendizagem por reforço mais populares, no qual o agente aprende os valores das ações sem a necessidade de um modelo explícito do ambiente. De uma forma geral, o objetivo final de um algoritmo de aprendizagem de reforço é produzir um agente artificial capaz de tomar boas decisões, segundo o estado do ambiente e da sua experiência [2].

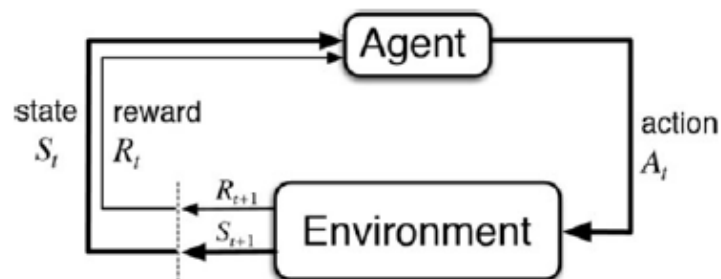


Figura 7- Fluxograma do processo de aprendizagem por reforço [27]

2.2.1.2. Modelos de Machine Learning

As técnicas de ML podem ser aplicadas em diversas áreas para resolver diferentes tipos de tarefas, tais como tarefas de regressão, classificação, *clustering*, redução de dados e deteção de anomalias [1, 25].

Os algoritmos de ML aplicados a tarefas de regressão são utilizados para otimizar os coeficientes das variáveis independentes, de forma que a previsão feita apresente um erro mínimo, originando uma variável numérica contínua [1]. Relativamente à classificação, o programa mapeia os recursos

de entrada para uma das variáveis de saída discretas, isto é, irá distribuir por categorias as variáveis de entrada [1,2]. O programa de *clustering* agrupa os dados baseando-se em padrões de semelhança entre os mesmos, dividindo-os em grupos relevantes, fornecendo ao executante do teste informação valiosa [1,28].

Nas tarefas de redução de dados, alguns pontos de dados menos relevantes são removidos por serem repetidos, não acrescentando nova informação ou por serem descontextualizados. Esta remoção a criação mais rápida de modelos, sendo utilizada como método auxiliar para outras tarefas de ML, como a tarefa de regressão e classificação. A detecção de anomalias é tratada maioritariamente por métodos de aprendizagem não supervisionadas, sendo muito semelhante à tarefa de *clustering*, agrupando as amostras. Portanto, num conjunto de dados os valores invulgares ou irregulares são identificados com a ajuda dos algoritmos de detecção de anomalias [1].

Os modelos de ML, utilizados na indústria 4.0, dividem-se conforme o tipo de aprendizagem e são apresentados na Tabela 2 com uma pequena descrição.

Tabela 2- Modelos de ML

Aprendizagem supervisionada		
Ref	Modelo	Definição
[12], [27],[32], [33],[34],[35]	<i>SVM/SVR</i>	Utilizado para tarefas de classificação e de regressão, que a partir de um conjunto de dados identifica os padrões e separa-os em duas classes possíveis. O SVM é usado como um método de seleção de recursos para escolher o que mais se adequa partindo dos dados pré-processados.
[12], [27],[36]	<i>Bayesian Networks/Naïve Bayesian Networks</i>	Baseando-se na teoria da probabilidade, descrevem as relações de probabilidade entre diversas variáveis, isto é, fornecem informações acerca das dependências probabilísticas das mesmas. Pode ser usado para diferentes tarefas de ML, como <i>clustering</i> , classificação supervisionada e detecção de anomalias.
[12], [33], [35],[37]	<i>K-Nearest Neighbors</i>	Utilizado para classificação e regressão, em que os valores de entrada correspondem ao conjunto de dados mais próximos. Na tarefa de classificação define-se a classe tendo em conta aquela que é a mais comum nos nós vizinhos, enquanto que na tarefa de regressão a saída é um parâmetro de uma propriedade determinado pela média dos nós mais próximos. É um classificador não paramétrico que usa o valor da distância para fazer previsões, sem necessidade de construção de um modelo, assentando apenas a partir de vetores de recursos e rótulos de classes.
[12], [35],[38]	<i>Artificial Neural Networks</i>	Modelo de processamento de dados que recolhe, armazena e processa informações empíricas para simular a função do cérebro humano. Deteta os padrões em

		<p>dados rotulados com ajuste dos pesos entre camadas para os dados de treino.</p> <p>Simula a computação descentralizada do sistema nervoso central por processamento paralelo, oferecendo capacidade de resposta dinâmica a entradas externas.</p>
[12], [35],[39]	<i>Multiple Linear Regression</i>	Técnica estatística que define uma função aproximada entre a resposta desejada (output) e as variáveis independentes de previsão (input).
[12], [35], [40]	<i>Decision Tree/Regression Tree</i>	<p>Representação gráfica de um modelo de classificação, onde os nós internos correspondem aos testes dos atributos (nós de decisão) e os nós das folhas corresponde aos rótulos de classe previstos.</p> <p>As árvores de regressão são usadas para prever valores contínuos enquanto que as árvores de decisão preveem valores categóricos.</p>
[12]	<i>Addictive models</i>	Mapeamento da variável alvo, como uma soma de transformações não lineares das variáveis de entrada.
[12],[41]	<i>Logistic Regression</i>	Modelo de análise de regressão linear generalizado, em que a sua implementação inclui três etapas: descoberta da função de previsão, construção da função de perda e definição dos parâmetros de regressão para minimização da função de perda. Pode ser aplicado em tarefas de regressão, classificação e multiclassificação.
[12]	<i>Bag of words</i>	Identifica as palavras mais frequentes num conjunto de texto, preparando um histograma em forma de vetores.
[12]	<i>Locally Weighted Training</i>	Algoritmo não-paramétrico com o objetivo de estimar funções não lineares usando um modelo linear simples, capaz de prever uma saída realizando a média (simples ou ponderada) dos valores dos pontos mais próximos do ponto de teste.
[12], [33],[35],[42]	<i>Random forest</i>	Método usado para classificação e regressão, que se baseia na criação de uma infinidade de árvores de decisão, de forma aleatória partindo dos dados da amostra original. De cada uma destas árvores de decisão sairá uma saída, isto é, a definição de uma classe. A classe que tiver mais votação será a tomada de decisão do algoritmo.
Aprendizagem não supervisionada		
Ref	Modelo	Definição

[12], [34],[43]	<i>K-means</i>	Algoritmo usado para tarefas de <i>clustering</i> , isto é, reúne os dados semelhantes e divide-os em categorias, podendo os dados serem numéricos ou categóricos. Este algoritmo não é adequado para problemas de dados mistos, com variáveis categóricas e numéricas em simultâneo. O método agrupa os dados, identificando o grupo com menor distância à variável em análise, criando os <i>clusters</i> . Cada vez que é adicionado um elemento ao grupo o seu valor médio é atualizado.
[35]	<i>Principle Component Analysis</i>	Procedimento estatístico que utiliza a transformação ortogonal para converter o conjunto de variáveis correlacionadas em variáveis linearmente não correlacionadas. É utilizada na monitorização do processo, redução da dimensão, deteção de anomalias e de <i>outliers</i> .
[35]	<i>Kernel Density Estimation</i>	Modelo não paramétrico que estima a função densidade de probabilidade de uma variável aleatória, com o objetivo de estimar variáveis e processos e estatísticas de controlo. É um dos métodos mais utilizados na indústria de processo.
[35]	<i>Gaussian Mixture Model</i>	Modelo probabilístico capaz de detetar a presença de subpopulações dentro de uma população geral utilizando dados não-rotulados. Este método é útil quando as variáveis do processo apresentam relações altamente não lineares.
[12]	<i>Self Organizing Map</i>	Algoritmo capaz de agrupar grandes quantidades de dados através do mapeamento com dados de treino, porém é um processo demorado.
Aprendizagem por reforço		
Ref	Modelo	Definição
[12],[44]	<i>PILCO</i>	Os algoritmos RL apresentam ineficiência na obtenção de informação uma vez que necessitam de muitos dados de treino até gerar informação valiosa. O modelo PILCO reduz este defeito, utilizando a aprendizagem de um modelo probabilístico dinâmico consegue gerar informação ao fim de poucas tentativas.
[12]	<i>SMART</i>	Utiliza valores de ação para representar a função de valor, sendo aproximados pelo modelo de simulação do domínio de produção e recorrendo a uma rede neural.
[2],[45]	<i>Q-Learning</i>	O Q-learning é um dos algoritmos de aprendizagem por reforço mais populares, no qual o agente aprende os

		valores das ações sem a necessidade de um modelo explícito do ambiente, estudando a função Q no processo de decisão finito de Markov. Esta função faz uma previsão do valor esperado do benefício que vai ser introduzido numa dada ação.
--	--	---

2.2.1.3. Vantagens e Desvantagens

Na Tabela 3 estão representadas as vantagens e desvantagens dos principais modelos de ML.

Tabela 3- Vantagens e desvantagens dos modelos de ML

Aprendizagem supervisionada			
Ref	Modelos	Vantagens	Desvantagens
[12],[32],[34]	<i>SVM/SVR</i>	Alto desempenho Alta precisão Capacidade de tratar grande volume de dados	Requisito de elevada memória
[12]	<i>Bayesian Networks/ Naïve Bayesian Networks</i>	Requisitos limitados de armazenamento Robustez com poucos dados	Tolerância limitada a atributos independentes Elevado poder computacional
[12],[37]	<i>K-Nearest Neighbors</i>	Intuitivo e simples Facilidade de implementação em problemas de multiclasse Não necessita de dados treino	Elevado poder computacional Desempenho dependente da dimensionalidade
[12],[34],[38]	<i>Artificial Neural Networks</i>	Lida bem com um grande conjunto de dados; Deteta todas as possíveis interações entre variáveis de previsão; Deteção implícita de relações não lineares complexas entre variáveis dependentes e independentes	Elevada dependência do hardware (GPU);

		<p>Elevada capacidade para prever, classificar e combinar dados</p> <p>Dados de entrada mínimos para obter resultados positivos.</p>	
[12]	<i>Multiple Linear Regression</i>	Capacidade de determinar a influência relativa de uma ou mais variáveis de previsão no valor alvo	Não lida bem com dados incompletos
[12],[46]	<i>Decision Tree/Regression Tree</i>	<p>Normalização dos dados não necessária;</p> <p>Baixa influência da falta de dados;</p> <p>Intuitivo e simples;</p>	<p>Propensão para instabilidade do algoritmo;</p> <p>Elevado tempo para treinar os dados</p>
[12]	<i>Additive models</i>	Quando o número de previsões potenciais é grande, modela relações não lineares altamente complexas	Alta propensão para sobreajuste.
[12],[41]	<i>Logistic Regression</i>	<p>É interpretável;</p> <p>Fácil de ajustar ao conjunto de dados;</p> <p>Fornece probabilidades previstas bem calibradas.</p>	Apenas resolve problemas lineares.
[12]	<i>Bag of words</i>	<p>Interpretação e implementação simples;</p> <p>Tem um grande sucesso em problemas de previsão;</p> <p>Elevada flexibilidade na personalização de dados.</p>	<p>O vocabulário requer uma morfologia cuidadosa;</p> <p>Representações dispersas difíceis de modelar.</p>
[12]	<i>Locally Weighted Training</i>	Previsão não paramétrica por funções de custo local.	Requisitos de memória aumentam exponencialmente com conjunto de dados maiores.
[12],[34],[42]	<i>Random forest</i>	<p>Elevada precisão;</p> <p>Bons resultados com dados limitados</p>	Não é possível melhorar a precisão de um conjunto de dados de alta dimensão.

Aprendizagem não supervisionada			
Ref	Modelos	Vantagens	Desvantagens
[12]	<i>K-means</i>	Implementação simples Facilidade em adaptação a novos modelos.	Agrupamento errado de valores discrepantes.
[12]	Self Organizing Map	Agrupa grandes conjuntos de dados; Mapeamento de dados de fácil leitura	Treino lento; Não constrói um modelo degenerativo para dados.
Aprendizagem por reforço			
Ref	Modelos	Vantagens	Desvantagens
[12]	<i>PILCO</i>	Não depende de conhecimento especializado.	Não tem em conta a correlação temporal
[12]	<i>SMART</i>	Permite a incorporação de conhecimento de domínio na parametrização.	A taxa de convergências é lenta em problemas discretos.

2.2.2. Deep Learning

O desenvolvimento de algoritmos de ML exigiu meios para mapear o conhecimento adquirido desde o processo de aprendizagem até à previsão, impulsionando o desenvolvimento de abordagens categorizadas, como a aprendizagem de representação [19]. O DL é um ramo do ML que se baseia num conjunto de algoritmos que tentam modelar abstrações de alto nível em dados, usando várias camadas de processamento com estruturas complexas, e este assenta em representações da aprendizagem dos dados [47].

O DL é um tipo de aprendizagem de representação que tem a flexibilidade de representar o mundo segundo uma hierarquia de conceitos, em que cada conceito é definido em relação a conceitos mais simples. Isto permite que os modelos computacionais aprendam representações com diversos níveis de abstração, compostas por várias camadas de processamento [19]. O DL é o núcleo do ML que visa desenvolver alguns algoritmos capazes de aprender com base na experiência de dados, alcançando uma posição importante na área da visão computacional, processamento de linguagem natural e no campo da física [44,45]. Um exemplo das diferentes representações que o DL pode realizar num tratamento de imagem é a representação por um vetor de valores de intensidade por pixel ou por um conjunto de arestas [47].

A escolha da representação pode facilitar a velocidade da aprendizagem, por isso têm vindo a ser feitos estudos de forma a melhorar as representações e a criar modelos para aprender estas representações a partir bases de dados não rotulados em grande escala. Algumas das representações são inspiradas nos avanços da neurociência e são baseadas na interpretação do processamento de informações e padrões de comunicação de um sistema nervoso [47].

Este método de aprendizagem pode ser definido como sendo uma classe de algoritmos de ML, que usam uma cascata de muitas camadas de unidades de processamento não linear para a extração e transformação dos recursos. Cada camada sucessiva usa a saída da camada anterior como entrada. Os algoritmos podem ser supervisionados e não supervisionados, pelo que o primeiro diz respeito à análise de padrões enquanto que o segundo diz respeito à classificação [44, 45].

As *Deep Neural Networks* (DNN) tem se tornado o método de DL mais usado no desenvolvimento de aplicações de DL. Contrariamente às redes neurais artificiais, que têm um número limitado de nós e camadas interconectadas, as DNN apresentam uma estrutura complexa com muitos nós que influencia o tempo de treino e a precisão [29]. Estes modelos são constituídos por três camadas: a camada de entrada, as camadas ocultas e as camadas de saída com funções de transferência entre elas, como apresentado na Figura 8. As redes neurais artificiais são insensíveis ao ruído, estas podem determinar a conexão subjacente entre as variáveis de entrada do modelo e os dados de saída, conduzindo a uma boa capacidade de generalização [31,46]. O funcionamento do DL tendo como base o sistema de redes neurais artificiais, utiliza algoritmos de aprendizagem e o *Big Data* para aumentar a eficiência dos processos de treino [50].

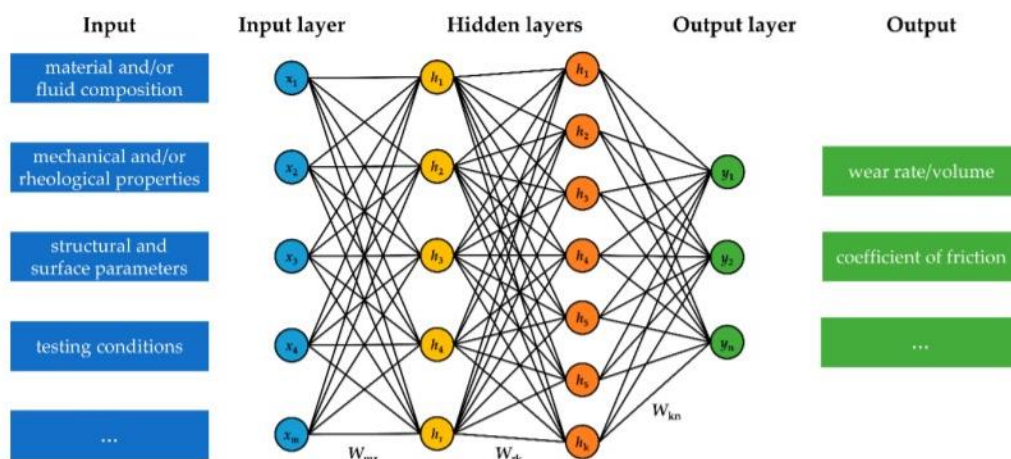


Figura 8- Exemplo de uma rede neural artificial [84]

Uma rede neural simples é constituída pela entrada (recurso de entrada), oculta e saída. Um neurónio leva em conta os recursos de entrada e o parâmetro de peso para fazer o cálculo da soma ponderada dos recursos de entrada [51].

2.2.2.1. Modelos de Deep Learning

Os modelos de DL, utilizados na indústria 4.0, dividem-se conforme o tipo de aprendizagem e são apresentados na Tabela 4 com uma pequena descrição.

Tabela 4- Modelos de DL

Aprendizagem supervisionada		
Ref	Modelos	Definição

[22],[47],[52]	<i>Deep Neural Networks</i>	Rede neural artificial composta por várias camadas ocultas entre as camadas de entrada e saída, podendo modelar relações não lineares complexas. Aplicação na deteção e análise de imagens.
[12],[22],[53]	<i>Convolutional Neural Network</i>	Este modelo é constituído por <i>convolucional layers</i> , <i>nonlinear layers</i> e <i>pooling layers</i> . A primeira camada transforma a informação de entrada e cria um conjunto de ferramentas locais, enquanto que a segunda camada aplica as funções de ativação que aplicam o mapeamento não linear de recursos. A terceira camada extrai a informação mais relevante utilizando ferramentas estatística, como o máximo e a média.
[12]	<i>Multiple Linear Perceptron</i>	É constituído por uma camada invisível entre a camada de entrada e a de saída, que utiliza uma função de ativação não-linear.
[12], [22],[54],[55]	<i>Recurrent Neural Network</i>	Rede neural utilizada para supervisionar dados sucessivos, pelo que pretende prever os dados lógicos partindo de valores anteriores. Constituído por duas fases e três camadas: fases de treino e de teste, e a camada de entrada, a oculta e a de saída. Este modelo executa sempre a mesma função para cada variável de entrada.
[12], [22], [56],[57]	<i>Restricted Boltzmann Machine</i>	Modelo constituído por camadas visíveis e ocultas cuja informação relevante é determinada com base nas soluções de ativação, funciona como o bloco de base das redes profundas atualmente desenvolvidas.
[12],[58]	<i>YOLOv2</i>	Modelo relacionado com a deteção e processamento em tempo real de imagens, permitindo a identificação de padrões estipulados.
Aprendizagem não supervisionada		
Ref	Modelos	Definição
[12], [22],[59]	<i>Auto Encoders</i>	É constituído por dois componentes, o codificador e o decodificador, sendo que estão separados por uma ou mais camadas intermédias. Este modelo consegue lidar com elevada complexidade de dados sem treino prévio, porém requer elevado poder computacional. Foi aplicado em várias áreas como classificação de imagens e reconhecimento facial.

[22],[47],[52]	<i>Deep Neural Network</i>	A definição está acima representada, com a diferença que neste caso os dados não são rotulados.
[47],[50],[60]	<i>Deep Belief Network</i>	Modelo probabilístico composto por várias camadas ocultas, em que cada camada é composta por módulos de aprendizagem simples. Tem interação com os modelos DNN e RBM.
[12],[21],[22]	<i>Convolutional Neural Network</i>	A definição está acima representada, com a diferença que neste caso os dados não são rotulados.
[12],[22],[54],[55]	<i>Recurrent Neural Network</i>	A definição está acima representada, com a diferença que neste caso os dados não são rotulados.
[12],[22],[56],[57]	<i>Restricted Boltzmann Machine</i>	A definição está acima representada, com a diferença que neste caso os dados não são rotulados.
[12]	<i>CAMP-BD</i>	Este modelo resulta da combinação do CNN com o RNN e apresenta duas vantagens: capacidade de lidar com maior volume de dados pois utiliza o DL e de resolver desafios complexos.

2.2.2.2. Vantagens e Desvantagens

Na Tabela 5 estão representadas as vantagens e desvantagens dos principais modelos de DL.

Tabela 5- Vantagens e Desvantagens dos Modelos de DL

Aprendizagem supervisionada			
Ref	Modelos	Vantagens	Desvantagens
[12],[50]	Convolutional Neural Network	Elevada precisão no reconhecimento de imagem; Elevada eficiência na extração de informação.	Necessita de muitos dados de treino; Elevado custo computacional.
[12]	Recurrent Neural Network	Destaca-se no manuseio de dados sequenciais.	Necessita de muitos dados de treino; Existem muitos parâmetros de entrada para ajustar; Elevada complexidade computacional.
[12]	Restricted Boltzmann Machine	Eficaz na conclusão de padrões e na extração de recursos.	Elevada complexidade de treino.

[12]	YOLOv2	Bom em processamento em tempo real.	Dificuldade em operar com grupos de objetos pequenos.
Aprendizagem não supervisionada			
Ref	Modelos	Vantagens	Desvantagens
[12],[50]	Auto Encoders	Intuitivo; Sem necessidade de conhecimento prévio dos dados; Bom manuseamento de grandes conjuntos de dados.	Elevado poder computacional; Hardware caro.
[12],[48],[50]	Convolutional Neural Network	Elevada precisão no reconhecimento de imagem; Elevada eficiência na extração de recursos	Necessita de muitos dados de treino; Elevado poder computacional;
[12]	Recurrent Neural Network	Destaca-se no manuseio de dados sequenciais.	Necessita de muitos dados de treino; Existem muitos parâmetros de entrada para ajustar; Elevada complexidade computacional.
[12]	Restricted Boltzmann Machine	É eficaz na conclusão de padrões e na extração de recursos.	Elevada complexidade de treino.
[12]	CAMP-BD	Alavancagem de grandes dados capturados no ambiente da indústria 4.0.	Como este se encontra numa fase inicial, não são apresentadas desvantagens.

2.2.3. Modelos Híbridos de Inteligência Artificial: Metaheurísticos

Os modelos híbridos resultam da combinação dos algoritmos de ML com algoritmos metaheurísticos, no sentido da previsão realizada ser mais precisa, minimizando o erro da mesma. Alguns exemplos de modelos metaheurísticos são o Particle Swarm Optimizacion (PSO), Genetic Algorithm (GA), Invasive Weed Optimizacion (IWO), o Cultural Algorithm (CA), Ant Colony Optimizacion (ACO), Firefly algorithm (FFA), Artificial Bee colony (ABC) [61]. Os problemas do mundo real apresentam um grande volume de dados, no entanto nem todos os recursos são necessários para obter informações úteis dessa base de dados, pelo que os modelos metaheurísticos selecionam quais os recursos relevantes, aumentando o desempenho do modelo [62].

Os algoritmos metaheurísticos são processos de otimização que alcança uma boa solução, tendo um comportamento estocástico, isto é, iniciam o processo de otimização criando soluções aleatórias. Estes métodos são aplicados com sucesso nas áreas de engenharia em ciências, nomeadamente nos campos industriais, engenharia elétrica e civil, e na análise de dados. Em aplicações biomédicas estes métodos identificam o melhor gene a partir de um gene candidato e em aplicações de análise de imagem reconhecem os melhores pixels perante um certo critério [62].

Na fase treino dos modelos de ML, é necessário o ajuste de alguns hiperparâmetros dos algoritmos, tais como a taxa de generalização e a taxa de aprendizagem, que são utilizados para controlar o processo de aprendizagem. Os algoritmos metaheurísticos são algoritmos de otimização e podem ser combinados com os algoritmos de ML para a determinação destes hiperparâmetros [63].

Tabela 6- Algoritmos Metaheurísticos

Algoritmos Metaheurísticos		
Ref	Algoritmo	Definição
[34],[61],[64],[65],[66],[67]	Particle Swarm Optimizacion (PSO)	Técnica de otimização que imita o comportamento de padrões da natureza (bandos de pássaros ou cardumes de peixes) em que as partículas são os indivíduos da população, representando potenciais soluções. Em cada iteração é determinada a posição e velocidade de cada partícula, recorrendo a fórmulas matemáticas, com base na melhor estimativa individual e do grupo, atualizando os possíveis valores da solução.
[34],[64],[65]	Genetic Algorithm (GA)	Algoritmo de otimização que se baseia na seleção natural de Darwin, iniciando com a geração aleatória de soluções em que cada uma é denominada de cromossoma. A função objetivo é determinada para cada uma destas soluções, sendo que a cada iteração se realiza operações de seleção, cruzamento e mutação com o objetivo de chegar à melhor solução para o problema.
[61],[68]	Invasive Weed Optimizacion (IWO)	Modelo de otimização baseado no crescimento e na dispersão das ervas daninhas na natureza, visto que estas se caracterizam por elevada estabilidade e adaptabilidade. De forma geral, as etapas do IWO são a geração de uma população inicial aleatória (soluções) no espaço de pesquisa e respetiva avaliação dos seus valores de função objetivo. Na iteração seguinte ocorre um ajuste com base na dispersão do modelo das ervas daninhas, até que se encontre a solução ótima para o problema.

[61]	Cultural Algorithm (CA)	O CA foi criado com base no modelo da evolução da cultura humana e a respetiva influência nos indivíduos. Os principais constituintes deste algoritmo são o espaço populacional, o espaço de crenças, a função de aceitação e a função de influência. Os algoritmos culturais conseguem usar vários domínios de conhecimento no espaço de crença, melhorando a pesquisa e a eficiência do algoritmo.
[66]	Ant Colony Otimizacion (ACO)	É uma técnica que se inspira no comportamento natural das colônias de formigas para determinação da rota ideal semelhante ao que acontece quando as formigas desenvolvem um caminho eficaz entre o seu ninho e a fonte da comida. Este algoritmo utiliza no seu processo de otimização formulações matemáticas combinando com uma estrutura organizada do problema.
[62],[65],[69]	Firefly algorithm (FFA)	Baseado no processo de acasalamento dos pirilampos em que estes imitem pequenos flashes de luz para troca de informação. Cada pirilampo (possível solução) emite luz cuja intensidade determina qual das soluções se vai movimentar em relação á outra, sendo que a distancia entre os pirilampos também influencia esta interação.
[34]	Artificial Bee colony (ABC)	Este algoritmo baseia-se nos comportamentos da organização de uma colónia de abelhas que assenta em dois princípios: auto-organização e divisão de trabalho. A auto-organização é um mecanismo de definição de tarefas resultante da interação entre os elementos constituintes do grupo enquanto que a divisão de trabalho diz respeito á seleção dos executantes das tarefas consoante as suas habilidades.

2.3. Aplicações de Inteligência Artificial

2.3.1. Aplicações de Machine Learning

Na Tabela 7 estão apresentadas várias aplicações de melhoria recorrendo ao ML em casos de estudo em diferentes setores de produção.

Tabela 7- Casos de estudo com aplicações de ML

Ref	Aplicações
[70]	Utilização de métodos de ML (SVR) para prever a quantidade de caudal na indústria petroquímica recorrendo a uma placa orifício.
[71]	Utilização de métodos de aprendizagem não supervisionada (PCA) para deteção de falhas e diagnóstico em plantas de produção de energia nuclear
[72]	Utilização de métodos de ML (Decision Tree e RF) para otimização do consumo de energia em máquinas de CNC
[73]	Utilização de métodos de ML (KNN, SVR, MLR) para orçamentação de custos de produção na fase de desenvolvimento do produto.
[74]	Utilização de métodos de ML (Decision Tree) para melhoria do processo produtivo na produção de canas-de-açúcar.

~

2.3.2. Aplicações de Deep Learning

Na Tabela 8 estão apresentadas várias aplicações de melhoria recorrendo ao DL em casos de estudo em diferentes setores de produção.

Tabela 8- Casos de estudo com aplicações de DL

Ref	Aplicações
[58]	Desenvolvimento de um assistente de controlo visual na secção de montagem para deteção de erros utilizando um algoritmo de DL. Os benefícios desta aplicação são aumento da produtividade, diminuição do tempo de aprendizagem de novos colaboradores e redução dos custos de não qualidade.
[75]	Criação de técnica para classificação dos grãos da indústria de cevada através de análise de imagem. Este método aumenta a precisão da identificação, atingindo o valor de 94 % de acerto e reduzindo o tempo de análise para uns escassos milissegundos.
[76]	Aplicação do CNN para deteção autónoma de situações de risco de queda dos tetos na indústria mineira, causado por elevadas tensões horizontais. Os resultados da aplicação desta técnica mostram uma precisão estatística de 86%.
[77]	Contagem e classificação das colónias bacterianas recorrendo a uma arquitetura CNN.
[78]	Desenvolvimento de um algoritmo baseado no CNN para diagnóstico de falhas de transformadores de potência e ataques cibernéticos na indústria de produção de eletricidade. Este método apresenta uma precisão de 94%.

2.3.3. Aplicações dos Modelos Híbridos de Inteligência Artificial

Na Tabela 9 são apresentadas aplicações de melhoria usando modelos híbridos de IA.

Tabela 9- Casos de estudo com aplicações de modelos híbridos de IA

Ref	Aplicações
[79]	Aplicação do modelo GWO para previsão dos caudais de água em situações extremas como cheias e secas. Verificou-se que o método apresenta maior precisão e convergência, quando comparado com os métodos de ML.
[80]	Investigação experimental de algumas variáveis do processo de maquinação utilizando o algoritmo GA para ajuste dos parâmetros do modelo ANN
[81]	Ajuste dos hiperparâmetros do modelo de SVM utilizando o GWO na indústria de construção, mais especificamente nos materiais de enchimento. Os resultados mostram uma otimização na correlação e eficiência do modelo.
[82]	Utilização de métodos metaheurísticos para otimização do modelo SVR na previsão de deformações e tensões limite em betão reforçado com fibras de compósito. Os resultados demonstram melhoria na correlação das previsões obtidas com os dados experimentais.

[83]	Aplicação do algoritmo GA ao modelo RF para melhoria da previsão do risco de atraso em projetos de construção. Os resultados demonstram a capacidade do modelo em lidar com informação complexa e não-linear existente no setor da construção.
------	--

3. DESENVOLVIMENTO

3.1. Desenvolvimento do Caso de Estudo de uma base de dados da Bosch

No ambiente industrial contemporâneo existe uma procura pela melhoria contínua dos padrões de qualidade em simultâneo com o aumento do volume de produção. Posto isto uma ferramenta capaz de prever produtos não conformes será extremamente útil para garantir ações corretivas na linha produtiva que minimizem estas ocorrências. Atualmente existem plataformas open source que permitem desenvolver estas ferramentas de forma gratuita, acessível e prática colmatando a impossibilidade de investimento de algumas empresas em equipas de desenvolvimento informático tornando mais difundida esta tecnologia.

A seleção da base de dados foi um processo iterativo de pesquisa sobretudo no site Kaggle, que apresenta diversas possibilidades de bases de dados para aplicação em modelos de Machine Learning. A maioria das bases de dados analisadas apresentavam um baixo número de recursos o que iria afetar negativamente a eficiência do algoritmo pois não seria capaz de treinar o modelo para posterior processamento de novos dados. Assim a base de dados fornecida pela Bosch destacava-se pelo elevado número de recursos e de amostras, embora apresentassem a falta de alguns valores que dificultaram o processamento. Esta base de dados constituiu o caso de estudo apresentado, uma vez que possuía todos os requisitos necessários para a aplicação de vários algoritmos de ML e a criação de um modelo.

3.1.1. Descrição da base de dados Bosch

A Bosch recolheu informações que deu origem a um grande *dataset* com o objetivo de prever os produtos não-conformes numa linha de produção em que cada linha da base de dados é representativa de um produto a realizar o seu percurso produtivo. Por exemplo, sendo apresentado um recurso como L0_S0_F8 significa linha 0, estação 0, e recurso 8. Esta informação advém de uma linha de montagem dividida em 4 segmentos e 52 estações, em que cada uma destas realiza um número variável de medições numa determinada peça ou produto totalizando 4265 recursos.

Algumas das características referentes a este conjunto de dados são:

- Número de recursos numéricos: 968
- Número de recursos categóricos: 2140
- Número de recursos temporais: 1157
- Número de observações: 1183747
- Percentagem de valores em falta: 78,5%
- Percentagem de valores não-conformes: 0,58%
- Percentagem de produtos conformes: 99.42%

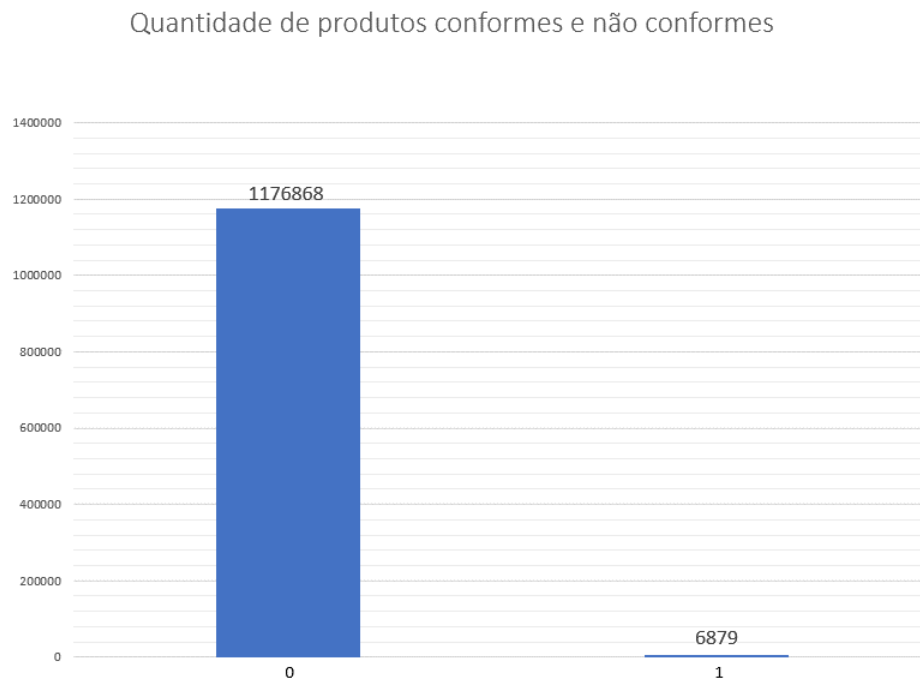


Figura 9- Histograma da quantidade de produtos conformes e não conformes

O objetivo da base de dados de teste é identificar com base num conjunto de recursos se uma dada observação se traduz numa resposta conforme, representada por 0, ou não-conforme, representada por 1, como apresentado na Figura 10. Os recursos temporais, numéricos e categóricos cooperam de forma a fornecer informação determinante à definição da resposta. Os pontos de interrogação a vermelho representam os valores que estão em falta.

D L3_S30_F3554	D L3_S30_F3559	D L3_S30_F3564	D L3_S30_F3569	I Response
-0.01	0.159	?	?	0
0.178	0.039	-0.003	0.016	0
-0.047	-0.221	0.009	0.016	0
-0.033	0.079	-0.003	0.097	0
-0.074	0.059	0	-0.038	0
0.183	-0.101	0.004	-0.038	0
-0.116	-0.041	0.004	0.124	1
0.077	-0.061	0.007	-0.038	1
-0.143	0.159	?	?	0
0.008	-0.001	?	?	0

Figura 10- Exemplo da base de dados da Bosch

Devido ao elevado número de recursos, a base de dados foi fornecida com os recursos numéricos, temporais e categóricos separados em três tabelas diferentes, permitindo o seu tratamento com menor poder computacional. O processamento dos dados vai ser efetuado no software, o KNIME, pelo que numa primeira fase é necessário adequar os dados de entrada através de técnicas de pré-processamento de dados. Como esta apresenta uma elevada percentagem de valores em falta optou-se por apenas considerar os recursos que apresentam mais de 10% de dados disponíveis melhorando assim qualidade dos dados de treino para o modelo.

3.1.2. Software Knime

O software KNIME é uma plataforma *open-source* que simplifica o processo de tratamento de dados, uma vez que o utilizador não necessita de realizar aprendizagem da linguagem de programação. Atualmente é um dos mais utilizados pela comunidade por ser simples, com uma interface intuitiva e apelativa, *open-source* e apresenta diversas soluções de ML. A forma como o utilizador pode construir o seu modelo de tratamento de dados é realizado através de um sistema de blocos pré-programados com pouca possibilidade de personalização, que permite que um utilizador com pouca experiência de programação desenvolva um modelo de trabalho pela colocação dos blocos em série. O software apresenta uma biblioteca de nós extensa o que o permite transformar os dados de entrada e assim desenvolver o fluxo de trabalho, obtendo modelos de ML eficientes.

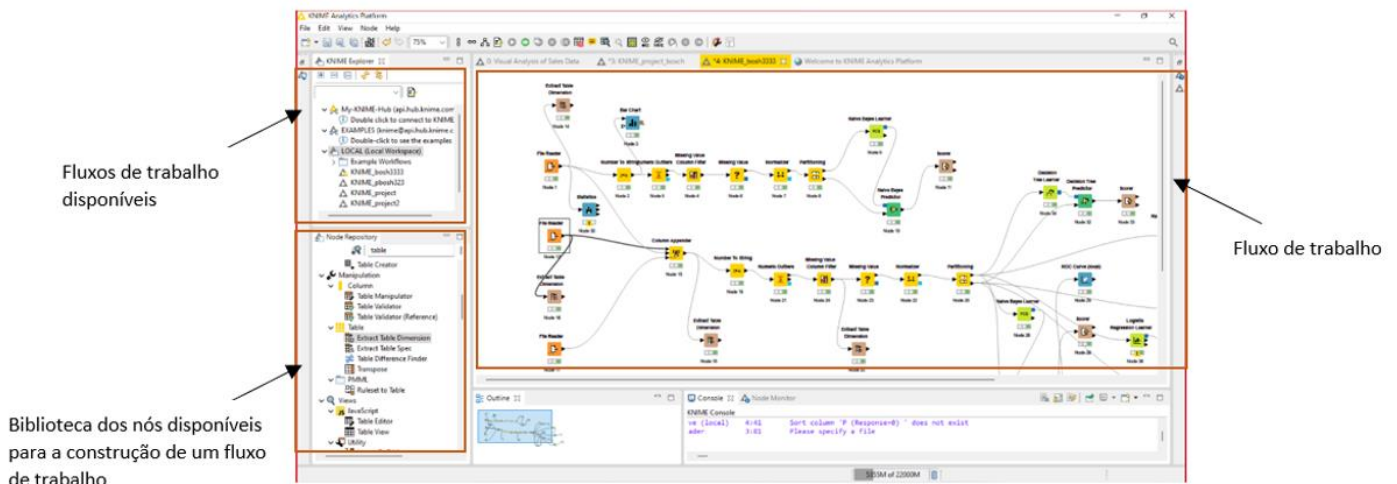


Figura 11- Representação das várias funções do software Knime

A *Knime Analytics Platform* funciona como um sistema de *Drag and Drop* em que os nós são criados ao serem arrastados do repositório de nós para o fluxo de trabalho, na parte central do ambiente de trabalho. Com esta funcionalidade a pipeline é rapidamente construída e configurada de forma rápida e intuitiva.

Os blocos representativos das funções designam-se por nós e são os elementos constituintes do modelo definido pelo utilizador. Cada um apresenta ligações de entrada e saída e uma lógica pré-programada para execução de uma determinada função. Consoante o estado do processamento apresentam uma cor diferente na barra inferior, como representado na Figura 12.

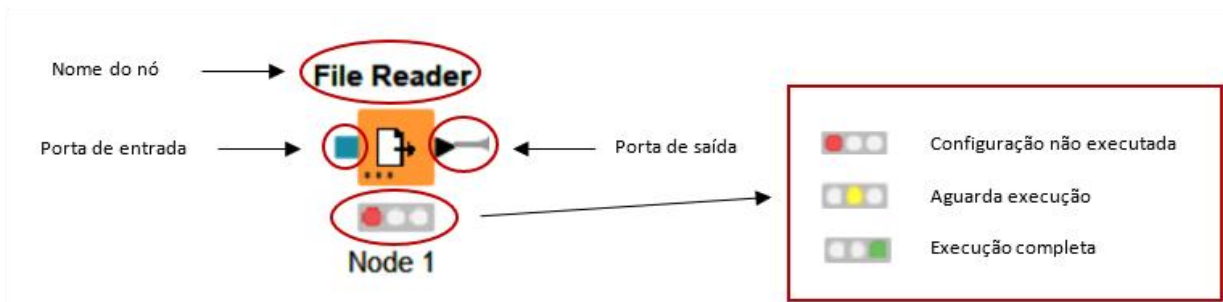


Figura 12- Esquema explicativo do exemplo de um nó do software Knime

3.1.3. Métricas de avaliação dos modelos

De forma a avaliar o desempenho dos modelos que irão ser desenvolvidos na plataforma KNIME, existem alguns parâmetros de referência que tem por base a matriz de confusão, representada na Tabela 10.

Tabela 10- Matriz de confusão de um modelo de ML

Valor real \ Valor previsto	Positivo	Negativo
Positivo	Verdadeiro Positivo (TP)	Falso Positivo (FP)
Negativo	Falso Negativo (FN)	Verdadeiro Negativo (TN)

De acordo com a Tabela 10, existem 4 cenários possíveis para a classificação dos valores previstos pelo modelo desenvolvido:

- Verdadeiro negativo (TN) – O modelo previu corretamente o valor negativo real.
- Verdadeiro positivo (TP)- O modelo previu corretamente o valor positivo real.
- Falso Negativo (FN) - O valor real correto é positivo, no entanto o valor previsto pelo modelo foi um resultado negativo, o que significa que falhou.
- Falso Positivo (FP)- O valor real correto é negativo, no entanto o valor previsto pelo modelo foi um resultado positivo o que significa que falhou.

A partir da matriz de confusão serão calculados alguns parâmetros indicadores do ajuste do modelo à base de dados e assim identificar o modelo com melhor desempenho. A métricas de avaliação que podem ser abordadas para avaliar os modelos são:

- Precisão (*Precision*)- é o indicador que define a razão entre o número de verdadeiros positivos e o número de positivos identificados.

$$Precisão = \frac{TP}{TP + FP} \quad (1)$$

- Exatidão (*Acuraccy*) – esta métrica de avaliação é considerada como sendo das mais importantes indicando a proporção de acertos do modelo. É a razão entre os valores identificados corretamente pelo modelo em relação ao número total de observações.

$$\text{Exatidão} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

- Recall - Uma sensibilidade elevada é indicadora de uma elevada percentagem de identificação de verdadeiros positivos, o que evidencia a capacidade de identificar corretamente peças conformes.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- Especificidade- Representa a razão entre o número de verdadeiros negativos e o número total de negativos identificados pelo modelo.

$$\text{Especificidade} = \frac{TN}{FP + TN} \quad (3)$$

- MCC (*Mathew correlation coeficient*) – O MCC é essencialmente um coeficiente de correlação entre as classificações binárias observadas e previstas e é geralmente considerada como uma das melhores medidas uma vez que utiliza todos os valores presentes na medida de confusão. Este método é aconselhável para bases de dados que apresentem um desequilíbrio grande entre os valores de classificação binária, que acontece nos dados relativos ao caso de estudo. O valor de saída do método é um coeficiente de correlação varia entre -1 e 1 em que o valor positivo representa uma previsão perfeita e o valor negativo indica uma total incoerência entre a previsão e a observação. O MCC pode ser obtido através da seguinte fórmula:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(FP + TP)(TP + FN)(TN + FP)(TN + FN)}} \quad (4)$$

- F1-score – É um indicador que combina a precisão e o *recall* de um modelo numa única métrica, sendo uma referência na comparação do desempenho e a qualidade dos modelos de classificação. Os valores dos scores F1 podem variar entre 0 e 1, sendo que 0 é incapaz de classificar qualquer observação na classe correta e o 1 significa que o modelo classifica perfeitamente cada observação na classe correta.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

3.1.4. Pré-processamento de dados

O pré-processamento de dados é um processo de preparação dos dados em bruto, tornando-os adequados à sua leitura pelos modelos de ML. É uma tarefa necessária para enquadrar os dados no formato necessário ao modelo, tornando a execução dos mesmo mais rápida, precisa e eficiente.

A primeira fase da pré-processamento consiste na importação dos arquivos no formato CSV para o software KNIME, especificamente as bases de dados numéricas, temporais e categóricas. Estas foram fornecidas de forma individual uma vez que as três em conjunto necessitam de maior poder computacional pois apresentam elevadas dimensões.

Na fase seguinte ignorou-se os valores que se encontravam desenquadrados dos restantes denominados *outliers*, recorrendo ao bloco “*numeric outliers*”. Optou-se por substituir os valores identificados como *outlier* por *missing values* e não eliminar a linha, uma vez que assim evita-se perda de informação com base em apenas um recurso. Em simultâneo foi efetuado a triagem dos recursos recorrendo ao bloco “*Missing Value Column Filter*”, tendo em conta que sempre que existam mais de 90% de linhas sem informação, esse recurso é eliminado. No caso de estudo em questão reduziu-se de 4268 recursos para 814 recursos, o que é prova que existiam diversos elementos não relevantes, promovendo uma diminuição do poder computacional necessário. A substituição dos *missing values* pelo valor fixo 0, leva à não criação dos valores fictícios possivelmente errados, que podem promover conclusões incorretas. Dado que esta base de dados contém uma elevada percentagem de valores desconhecidos, optou-se por não eliminar as linhas que contivessem recursos não-preenchidos, o que se iria traduzir numa elevada perda de informação.

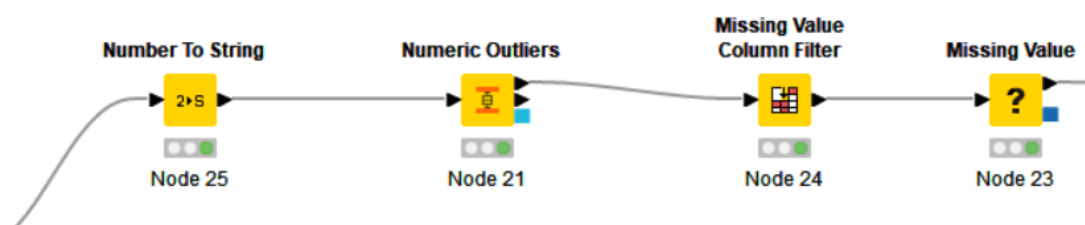


Figura 13-Sequenciamento de etapas constituintes do pré-processamento

Na terceira fase do pré-processamento realizou-se a junção das três bases de dados numa única tabela integrando a parte numérica, categórica e temporal no sentido de incluir todos os recursos que foram usados para a previsão da variável alvo (*Response*). Os modelos de *Machine Learning* utilizados necessitam que a variável a prever seja do tipo *String* pelo que para isso recorremos a um bloco denominado “*number to string*”. Por último na quarta fase do pré processamento removeu-se duas colunas duplicadas resultantes do processo de junção e irrelevantes para a análise e posteriormente exportou-se no formato CSV para utilização nos diferentes modelos. Na Figura 14 está representado o fluxo de trabalho do pré-processamento dos dados realizado na plataforma KNIME.

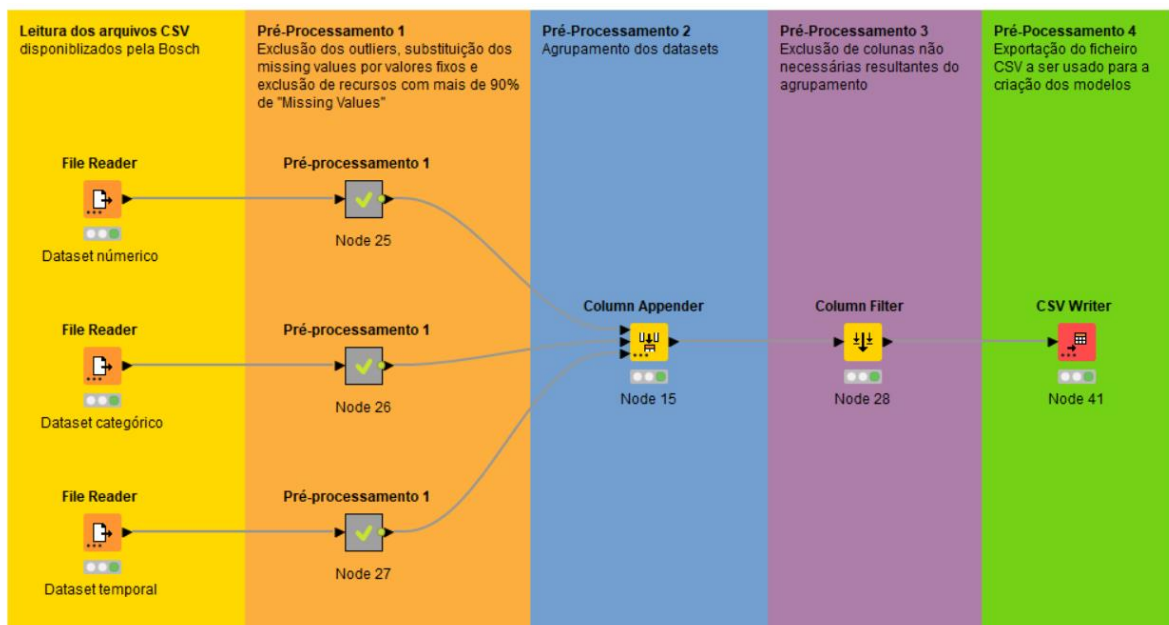


Figura 14- Etapas do pré-processamento dos dados no KNIME

3.1.5. Desafios associados ao desenvolvimento dos modelos

O principal objetivo de um modelo de ML é conseguir fazer uma boa generalização que permita realizar boas previsões a partir dos dados de entrada desconhecidos para o modelo. A qualidade do treino do modelo não se baseia só na sua precisão face a um tipo de dados de entrada, mas também na sua capacidade de ser versátil às diferentes solicitações a que pode estar sujeito. Duas dificuldades que podem surgir no treino de um modelo será o *overfitting* ou o *underfitting*.

O *overfitting* ocorre quando o modelo de ML tenta utilizar mais pontos do que o necessário e por isso tentando se ajustar a esse maior número de pontos, reduzem a eficiência e a precisão do modelo. De forma a evitar a ocorrência do *overfitting* pode-se recorrer á validação cruzada, utilização de um maior conjunto de dados de treino e remover os recursos não relevantes para o modelo. O modelo *overfitting* apresenta um bom ajuste aos dados de treino, porém quando existe a alteração significativa dos dados de entrada, este não tem a capacidade de responder em conformidade, posto isto existirá casos em que o modelo apresentará baixo erro de previsão.

O underfitting é a incapacidade de o modelo perceber a tendência dos dados não permitindo desenvolver um modelo de previsão fiável. É possível minimizar a ocorrência destes problemas através do aumento do conjunto de dados de treino e do número de recursos.

3.1.6. Definição dos dados de amostra por Resampling

O modelo de Machine Learning deve ser sempre parametrizado recorrendo a dados de treino, sendo que é de extrema importância que a amostra represente de forma fidedigna o sistema real. Para isso os dados escolhidos para a amostra podem ser sujeitos a algoritmos de resampling como o caso do Hold-out e da validação cruzada k-fold. De forma a avaliar a qualidade do ajuste do modelo às bases de dados em questão é necessário submeter o modelo a um conjunto de dados rotulados, detetando o desvio entre o valor previsto e o valor rotulado. Após este treino o modelo deverá ser sujeito a um conjunto de dados não rotulados que melhor representem as situações reais possivelmente impostas.

3.1.6.1. Hold-out

Esta técnica é usada para a criação de algoritmos de aprendizagem supervisionada, exige a exclusão de uma percentagem dos dados de teste e a sua substituição por dados de avaliação reais não-rotulados. Normalmente esta percentagem de substituição estará entre 10% e 30%. Como consequência existe uma parte dos dados de teste que são desconsiderados podendo perder-se informação relevante nessa decisão. O *resampling* por *Hold-out* está sujeito a que ocorra *overfitting* dos dados que se traduz numa habituação dos algoritmos aos dados de teste, reduzindo a sua flexibilidade a novas condições imposta por outros dados. A base de dados em questão apresenta 99.4% das respostas como 0, pelo que ao utilizar esta ferramenta correríamos o risco de o modelo se ajustar demasiado a esta resposta.

3.1.6.2. Validação cruzada K-fold

A base do conceito é a mesma que a anterior existindo uma parte da base de dados descartada na fase de treino, porém existe um conjunto de iterações que fazem alterar a percentagem de dados ignorada, originando uma validação cruzada em que todos os dados de treino acabam por fazer parte da análise. Para isto é necessário numa primeira fase definir em quantos subconjuntos k serão os dados de treino enquadrados. Em cada iteração o conjunto de valores k-1 farão a validação do conjunto k, em que tal como a Figura 15 sugere, os dados em alguma das iterações farão parte dos dados de teste. No final podemos avaliar a média aritmética das diferentes iterações, bem como calcular o seu desvio padrão que serão representativos da eficácia e ajuste dos parâmetros do modelo aos dados de teste ensaiados. Um valor de desvio padrão menor é o que se procura para obtenção de uma maior robustez do modelo.

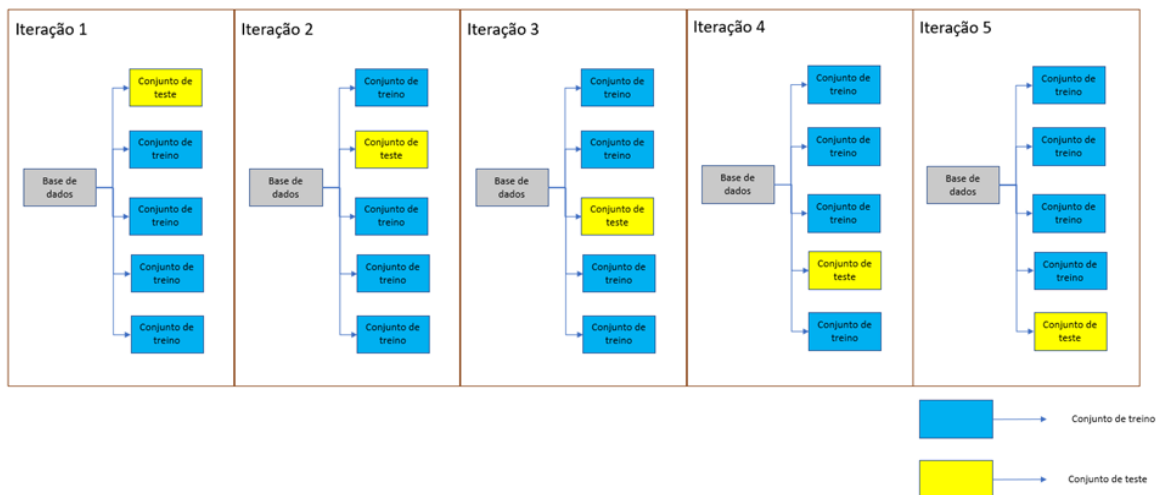


Figura 15- Exemplo de uma validação cruzada K-fold

3.1.7. Fases relevantes para a construção de um modelo de ML

A criação de um sistema de Machine Learning designado por modelo tem como ponto de partida o processamento de um conjunto de dados a partir do qual será criado uma linha de raciocínio capaz de realizar análises preditivas. Aplicado ao caso de estudo em questão começou-se por seleccionar a base de dados da linha produtiva Bosch sendo efetuado uma análise detalhada das características da base de dados. Esta etapa é uma das mais importantes na construção do modelo uma vez que a quantidade e a qualidade dos dados recolhidos afetam diretamente a eficácia do modelo.

O pré-processamento dos dados inclui tarefas como eliminação de dados ruidosos, *outliers*, valores ausentes, entre outras, responsáveis por melhorar a qualidade dos dados impostos ao modelo. O software utilizado (KNIME) possui ferramentas para realizar este tipo de pré-processamento, como foi apresentado anteriormente, e permite também realizar a seleção dos recursos mais relevantes. Dentro das várias opções simuladas e através de alguns indicadores previamente referidos no Capítulo 3.1.3 será seleccionado qual a técnica de seleção de recursos que promove melhores resultados. Na última fase executa-se o algoritmo á base de dados em questão tendo em conta os recursos seleccionados, construindo um modelo respeitando os dados de treino, como representado na Figura 16. Posteriormente o modelo desenvolvido é submetido a novos dados de forma a realizar análises preditivas.

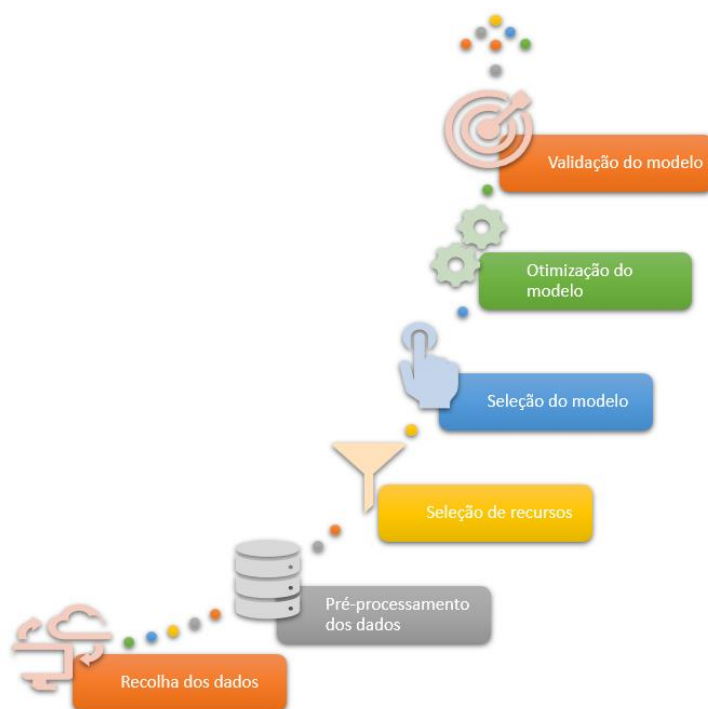


Figura 16- Etapas da construção do modelo

3.2. Construção e evolução dos modelos

De forma a prever quais os produtos que irão falhar no requisito de qualidade foram selecionados cinco modelos preditivos de *Machine Learning* com o objetivo de executar uma análise comparativa de precisão e robustez. Posteriormente aquele que apresentar melhores índices será selecionado para o modelo final. Devido às limitações do poder computacional disponível efetuou-se uma redução para cerca de 10% do número de linhas inicial. Uma vez que o número de *samples* é bastante superior ao número de *features* não foi necessário recorrer a uma eliminação de recursos para projetar os modelos preditivos. No entanto, de forma a realizar uma otimização entre o poder computacional exigido e a eficácia do modelo foi feita posteriormente uma análise do impacto da redução dos *features*.

3.2.1. Fluxograma dos Modelos de Classificação

Após o pré-processamento dos dados, estes foram submetidos à avaliação por parte de diversos modelos com o objetivo de quantificar a viabilidade e a eficiência dos modelos no reconhecimento das falhas nos requisitos de qualidade dos produtos. No que diz respeito à aprendizagem supervisionada foram escolhidos alguns modelos de classificação entre os quais o *Decision Tree*, *XGBoost*, *Naive Bayes*, *Logistic Regression* e o *Random Forest*, representado na Figura 17, constituindo uma comparação de *scores* na plataforma KNIME.

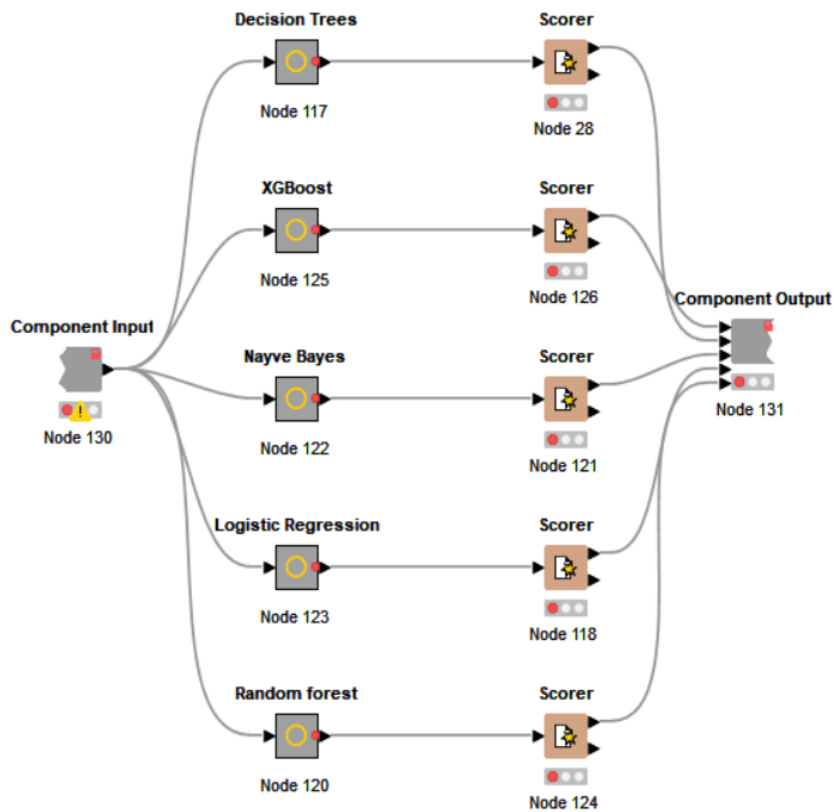


Figura 17- Fluxograma dos modelos desenvolvidos no Knime

3.2.2. Classificação supervisionada da conformidade do produto de uma linha de produção

Para a criação de modelos preditivos das falhas dos produtos das linhas de produção da Bosch, foi criado um *workflow* no KNIME automatizando o processo de avaliação dos classificadores usados nesta fase.

Foram usados vários classificadores/algoritmos para a criação de vários modelos preditivos que vão ser avaliados segundo as seguintes métricas de avaliação, como a *accuracy*, *recall*, *precisison* e o *F1-Score*. Na Figura 18 está representado o *workflow* com a representação das várias etapas para a criação dos modelos.

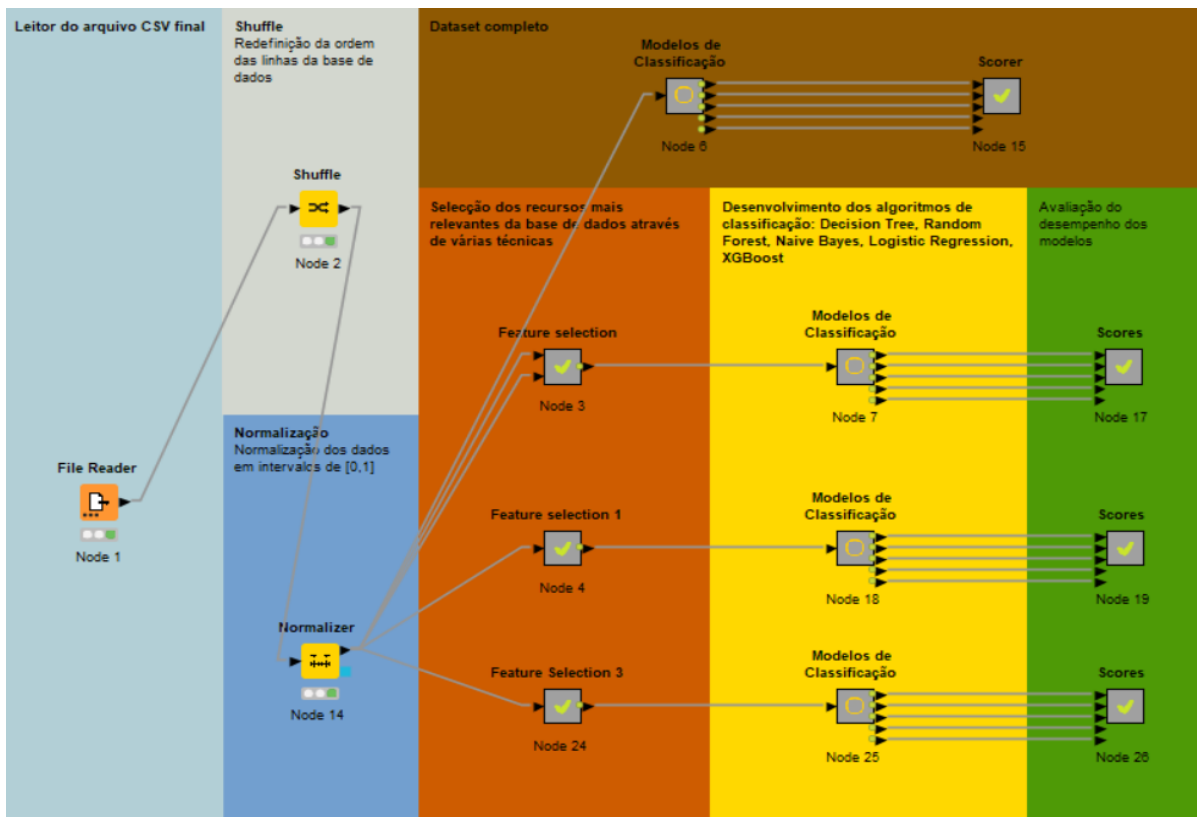


Figura 18- *Workflow* do desenvolvimento dos modelos

Para a construção dos modelos recorreu-se em primeiro lugar ao nó *shuffle* para embaralhar as linhas da base de dados, evitando o sobreajuste do modelo a uma determinada ordem de classificação dos padrões dos dados, treinando o algoritmo para quase todos os pontos de dados promovendo melhores resultados.

Posteriormente, utilizou-se o nó “*Normalizer*” de forma a colocar os valores entre [0,1], podendo também ter sido utilizado a escala de [-1,1], caso existissem valores negativos. A normalização dos dados tem extrema importância pois condensa os dados numa gama de valores e não deixa que alguns recursos dos dados assumam maior preponderância por apresentarem valores mais elevados, influenciando a eficácia do modelo. Caso se pretenda analisar os dados originais deverá reverter-se a normalização para se realizar uma análise exploratória dos dados.

A Figura 20 demonstra a dispersão de um exemplo de um recurso selecionado limitando o número de linhas às primeiras 20000 ocorrências. A Figura 20 apresenta os mesmos dados após a efetuação de uma normalização aos mesmos, o que se pode constatar comparando os dois gráficos é que a dispersão se mantém semelhante existindo apenas uma escala de grandeza entre [0,1], comprovando que não existe distorção da informação presente na base de dados.

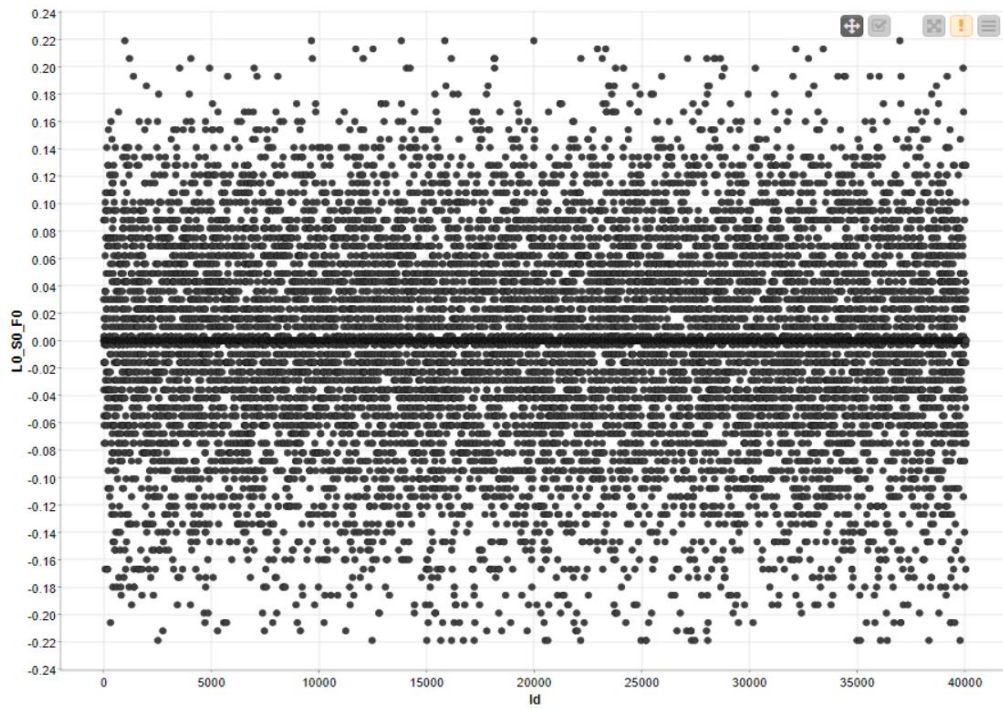


Figura 20- Gráfico de dispersão das primeiras 20.000 ocorrências

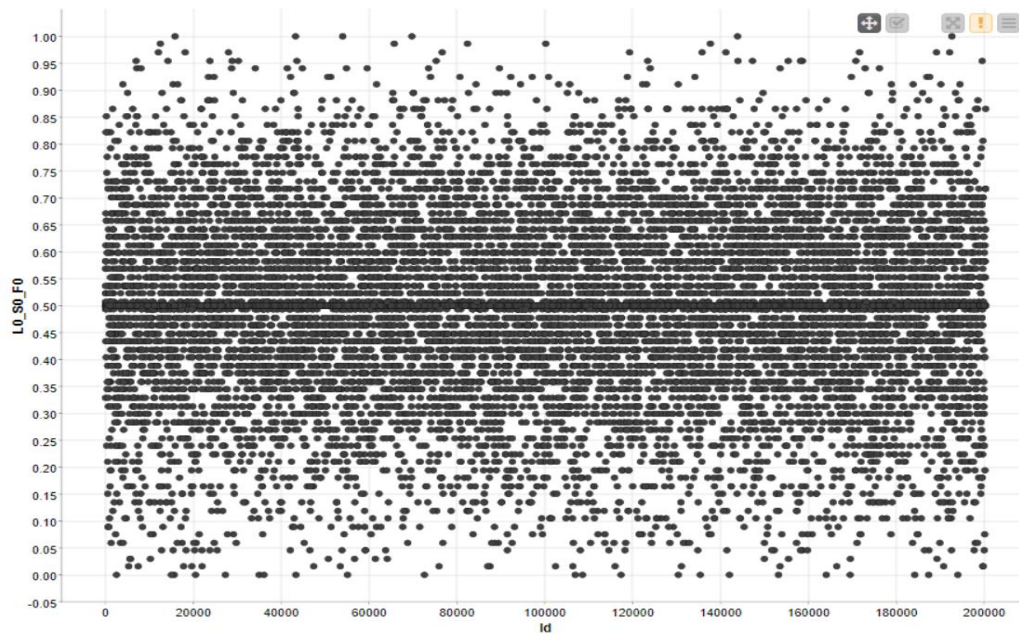


Figura 20- Gráfico de dispersão das primeiras 20.000 ocorrências após a normalização dos dados

3.2.2.1. Análise dos modelos

Recorrendo à totalidade dos recursos da base de dados da Bosch foram desenvolvidos diversos modelos de ML, determinando o valor do seu desempenho. Os algoritmos utilizados no desenvolvimento dos modelos foram o *Naive Bayes*, *Decision Trees*, *XGBoost*, *Random Forest* e o *Logistic Regression*, apresentando desempenhos relativamente diferentes, mas mantendo um denominador comum, procura que a *accuracy* seja o mais elevada possível, não desprezando as restantes métricas. A aplicação de várias técnicas de redução de dimensionalidade permite obter modelos mais robustos, rápidos e eficientes.

A base de dados utilizada apresenta um elevado desequilíbrio na distribuição das classes, isto é cerca de 99.5% dos dados são produtos conformes. Pelo que os algoritmos poderão encontrar dificuldade em prever os produtos não-conformes devido ao desequilíbrio das observações entre as classes, correspondente a uma variável binária "Response". Devido á disparidade do volume de observações, o modelo tende a categorizar a classe com maior expressão levando a uma falsa sensação de precisão.

Os modelos de previsão apresentam diferentes sensibilidades em relação ao tipo de variáveis do modelo, logo é de extrema importância a redução da assimetria dos dados ou *outliers* para a garantia de um melhor desempenho. Um exemplo são os modelos de classificação baseados em árvores que criam divisões nos dados de treino. Para minimizar este desequilíbrio entre classes deve-se recorrer a métodos que equilibram a sua distribuição por métodos de amostragem que visam aumentar a classe minoritária (*oversampling*) ou diminuir as observações da classe maioritária (*undersampling*).

Este tipo de melhoria pode ser realizado KNIME com recurso ao Smote tendo sido utilizado este nó na criação dos modelos, melhorando assim os resultados obtidos e a eficiência na previsão dos produtos não conformes, que correspondem á classe minoritária. Na Figura 21 está representado o recurso *Smote* utilizado no *oversampling* em que é possível a definição dos parâmetros deste nó de modo a torná-lo mais fiável.

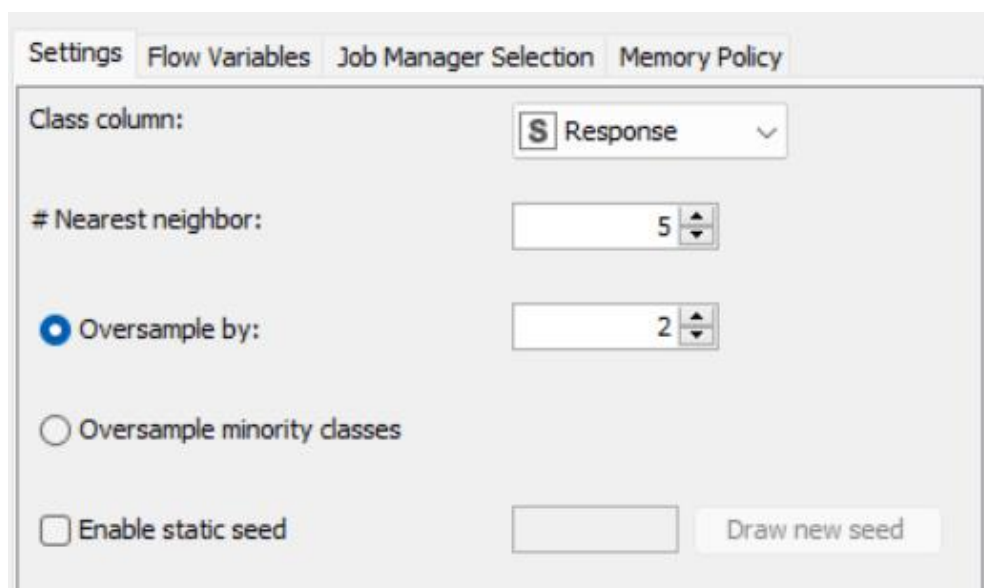


Figura 21- Parâmetros de ajuste do nó *Smote*

Diversos modelos foram ajustados de forma a melhorar o seu desempenho perante os dados de origem. Na Tabela 11 está representado um resumo das métricas de desempenho dos classificadores da base de dados.

Tabela 11- Desempenho das métricas de avaliação dos modelos

Dataset Completo (941 recursos)				
Métricas de avaliação				
Modelos	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Decision Trees</i>	99.297%	99.451%	99.844%	99.647%
<i>XGBoost</i>	99.428%	99.432%	99.999%	99.715%
<i>Naive Bayes</i>	99.280%	99.433%	99.845%	99.639%
<i>Random Forest</i>	99.280%	99.456%	99.818%	99.637%
<i>Logistic Regression</i>	99.419%	99.436%	99.999%	99.717%

De uma forma geral os modelos obtiveram bons resultados com uma *accuracy* elevada, porém foi necessário recorrer a outras métricas de avaliação que se ajustasse melhor a classes desequilibradas, tais como o *recall*, *precision* e o *F1-Score*. Para o *dataset* avaliado qualquer uma destas métricas se revelou acima dos 99%, o que demonstra a eficiência do modelo.

As matrizes de confusão dos diferentes modelos são diferentes revelando diferente número de falsos positivos, verdadeiros positivos, falsos negativos e verdadeiros negativos, influenciando os resultados transmitidos á empresa em cada caso. Se o modelo revelar um elevado número de falsos positivos serão fornecidos aos clientes produtos não-conformes quando o modelo previu que seriam conformes. Se o modelo revelar um elevado número de falsos negativos gerará prejuízo uma vez que se perderá a oportunidade de vender um produto conforme que está a ser considerado não conforme pelo modelo. A matriz de confusão é de extrema importância uma vez que constitui a origem dos dados para as métricas de avaliação.

Uma precisão elevada indica uma baixa ocorrência de falsos positivos, sendo que todos os modelos estudados apresentam esta característica. Relativamente ao parâmetro *recall* este fornece indicação da ocorrência de falsos negativos pelo que é um parâmetro a seguir quando se pretende minimizar esta ocorrência. Neste indicador destacam-se os modelos *XGBoost* e o *Logistic Regression* que obtiveram resultados de 100% evidenciando uma correta classificação dos casos positivos.

Por sua vez o F1-Score permite analisar de forma conjunta as métricas de *recall* e *precision* e é útil quando os falsos positivos e falsos negativos possuem impactos diferentes no modelo que é o que se verifica no caso de estudo. O modelo que melhor se ajusta com base na análise das métricas de avaliação é o *Logistic Regression*, no entanto todos os modelos seriam adequados para serem adotados na empresa.

Pelo método de *Decision Tree* conclui-se que os recursos mais relevantes para o desenvolvimento do modelo são o L1_S24_F1845, L3_S23_F3865, L3_S26_F3051, L3_S37_D3951 e L1_S24_F1836, como se pode observar na Figura 22.



Figura 22- Representação da árvore de decisão do modelo

Por outro lado, o modelo *Logistic Regression* define um grau de relevância para cada recurso evidenciando o seu peso na decisão do modelo através dos coeficientes de regressão logística, que são apresentados na Figura 23. Como é possível observar o primeiro coeficiente é praticamente o dobro do segundo, evidenciando a sua preponderância e o seu grau de importância na tomada de decisão do modelo.

Row ID	S - Logit	S Variable	D Coeff.
Row123	1	L1_S24_F1842	7.683
Row441	1	L0_S4_D111	3.502
Row440	1	L0_S4_D106	3.121
Row442	1	L0_S5_D115	3.077
Row443	1	L0_S5_D117	3.077
Row119	1	L1_S24_F1831	2.969
Row51	1	L0_S9_F195	2.588
Row167	1	L3_S29_F3376	2.572
Row221	1	L3_S30_F3634	2.529
Row77	1	L0_S12_F330	2.327
Row93	1	L0_S14_F366	2.139
Row216	1	L3_S30_F3589	1.922
Row226	1	L3_S30_F3669	1.744
Row225	1	L3_S30_F3664	1.652
Row69	1	L0_S11_F298	1.552
Row80	1	L0_S12_F336	1.466
Row153	1	L3_S29_F3330	1.465
Row179	1	L3_S29_F3424	1.465
Row2	1	L0_S0_F2	1.358
Row218	1	L3_S30_F3609	1.305
Row94	1	L0_S14_F370	1.293
Row241	1	L3_S30_F3784	1.288
Row248	1	L3_S30_F3819	1.278
Row54	1	L0_S10_F219	1.242
Row50	1	L0_S9_F190	1.226
Row28	1	L0_S3_F92	1.226
Row195	1	L3_S29_F3476	1.156
Row8	1	L0_S0_F14	1.153
Row758	1	L3_S35_D3900	1.15
Row759	1	L3_S35_D3905	1.15
Row760	1	L3_S35_D3910	1.15
Row761	1	L3_S35_D3915	1.15
Row111	1	L1_S24_F1565	1.108
Row122	1	L1_S24_F1838	1.095
Row99	1	L0_S15_F403	1.061
Row136	1	L2_S26_F3077	1.032
Row273	1	L1_S24_F1525=T4	0.985
Row272	1	L1_S24_F1525=T3	0.923

Figura 23- Estatísticas do modelo *Logistic Regression*

3.2.2.2. Impacto do Feature Selection e redução da dimensionalidade dos dados no modelo

A base de dados em estudo apresenta um elevado número de recursos, portanto faz sentido estudar a redução da dimensionalidade desta, uma vez que se trata de uma técnica para mapear dados de uma alta dimensão e representá-los numa baixa dimensão para visualizar os padrões que sem esta alteração não são evidentes. Trata-se de um conjunto de técnicas de agrupamento e classificação de forma a projetar os dados num espaço dimensional mais baixo.

O desafio trata-se em reduzir o número de colunas (recursos) e não o número de linhas (ocorrências) uma vez que quanto maior a quantidade de recursos, mais difícil é para o modelo se desenvolver. Outro ponto crucial a considerar é que a maioria das variáveis são frequentemente correlacionadas, o que vai introduzir muitos fatores redundantes no conjunto de treino do modelo de ML, o que torna o modelo mais complexo e aumenta a probabilidade da existência do *overfitting* implicando um mau funcionamento com dados reais.

A redução de dimensionalidade é capaz de simplificar o modelo, reduzir o tempo de treino e reduzir a probabilidade de ocorrência de *overfitting*. A simplicidade do modelo irá garantir menor grau de incorreção na previsão do mesmo. A redução da dimensionalidade dos dados, isto é, a extração dos recursos mais relevantes trará muitas vantagens, tais como:

- Eliminação de dados ruidosos e recursos redundantes.
- Melhoria da precisão e o desempenho do modelo.
- Menor poder computacional que leva a maior rapidez no treino do modelo.

Como foi referido a elevada dimensionalidade da base de dados utilizada obriga a que seja feita uma seleção ponderada dos recursos, pelo que foram aplicadas diversas técnicas de redução de dimensionalidade como a *Linear Correlation*, *Low variance*, e o *PCA (Principal Componente Analysis)*.

3.2.2.2.1. Seleção dos recursos aplicando o filtro da Linear Correlation

Por vezes os recursos apresentam padrões e tendências semelhantes representando conclusões idênticas e por isso não acrescentam informação valiosa podendo afetar negativamente o desempenho do modelo. Para essa situação deve-se recorrer à matriz de correlação de *Pearson*, de forma a identificar quais os recursos redundantes que apresentam valores de correlação idênticos. Na Figura 24 pode-se observar uma amostra dos recursos que apresentam correlação positiva a azul e correlação negativa a vermelho.

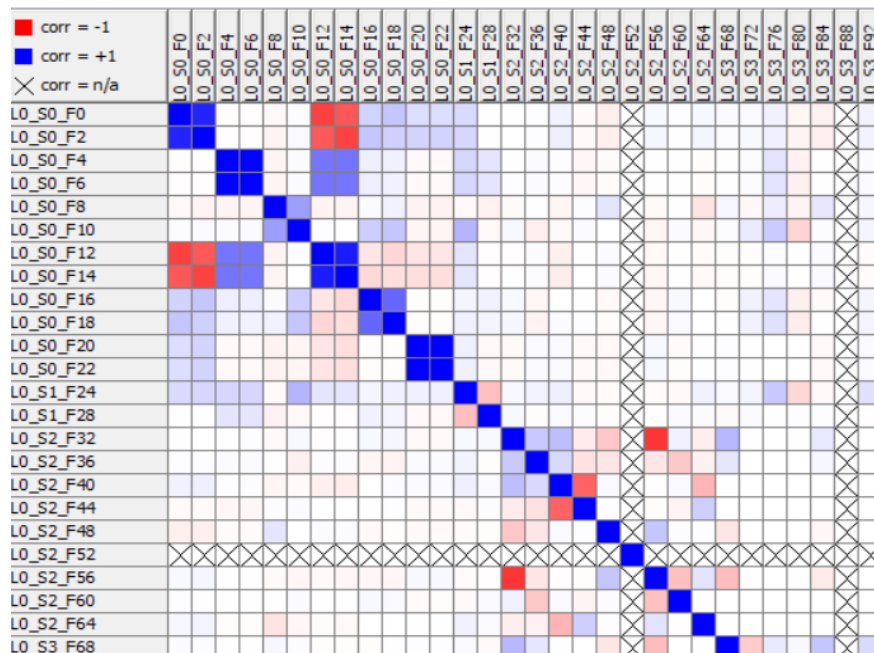


Figura 24- Matriz de correlação dos recursos

Após a identificação da correlação linear entre os recursos aplicou-se um filtro que acima de 0.9 se mantinha apenas um dos recursos, de forma a não duplicar a carga computacional do modelo, aumentando a sua eficiência. No software KNIME é possível realizar este processo, como está representado na Figura 25, porém é necessário ter em conta que a realização da normalização dos dados é fundamental para garantir o fator de escala.

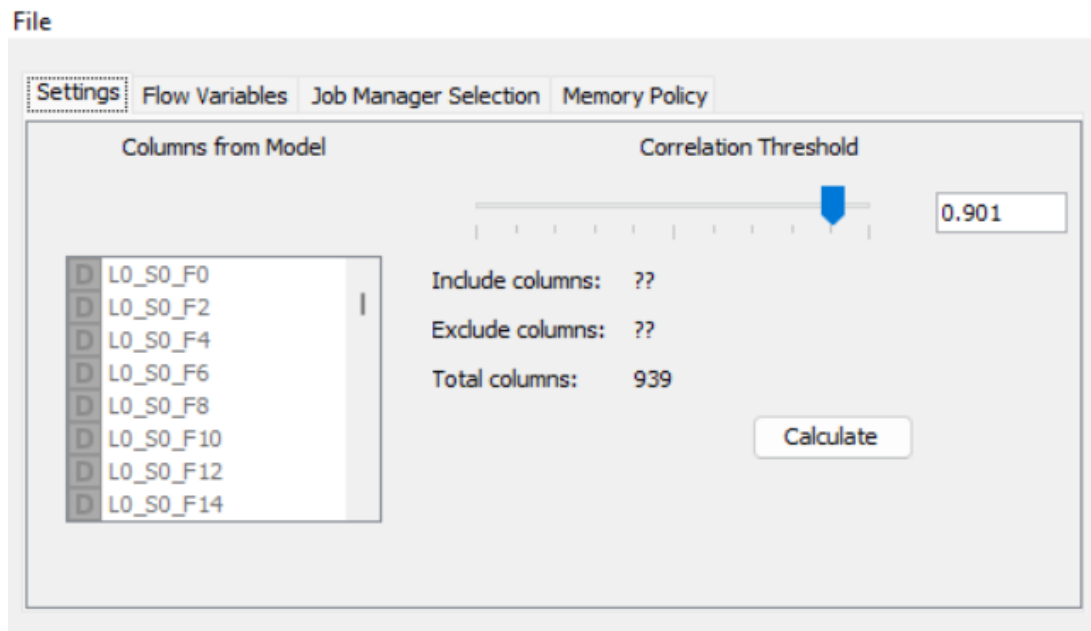


Figura 25- Aplicação de filtro aos recursos com base na correlação

Na Tabela 12 estão apresentadas as métricas de avaliação dos diferentes modelos aplicados á base de dados após a seleção de recursos utilizado o *Linear Correlation*.

Tabela 12-Desempenho das métricas de avaliação dos modelos

Técnica de seleção de recursos - <i>Linear correlation</i> (394 recursos)				
Métricas de avaliação				
Modelos	<i>Acuraccy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Decision Trees</i>	99.324%	99.446%	99.876%	99.66%
<i>XGBoost</i>	99.432%	99.431%	100%	99.715%
<i>Nayve Bayes</i>	99.432%	99.431%	100%	99.715%
<i>Random Forest</i>	99.278%	99.459%	99.817%	99.638%
<i>Logistic Regression</i>	99.430%	99.433%	100%	99.714%

As principais melhorias detetadas foi a redução do tempo de computação e o *recall* dos modelos com os valores assinalados a negrito subiram para 100% minimizando a ocorrência de falsos negativos.

3.2.2.2.2. Seleção dos recursos aplicando o filtro Low Variance

Tal como na técnica anterior a normalização é obrigatória antes da aplicação da redução da dimensionalidade. Os recursos que apresentem variâncias próximas de zero devem ser ignorados da análise, uma vez que não vão afetar a variável de saída e apenas exigirão maior poder computacional. A técnica do filtro de baixa variância define um valor limite de variância eliminando os recursos que se apresentem abaixo desse limite.

Na Tabela 13 estão apresentadas as métricas de avaliação dos diferentes modelos aplicados á base de dados após a seleção de recursos utilizando a *Low Variance*.

Tabela 13- Desempenho das métricas de avaliação dos modelos

Técnica de seleção de recursos – <i>Low Variance</i> (763 recursos)				
Métricas de avaliação				
Modelos	<i>Acuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
<i>Decision Trees</i>	99.305%	99.450%	99.850%	99.651%
<i>XGBoost</i>	99.427%	99.433%	100%	99.713%
<i>Nayve Bayes</i>	98.680%	99.440%	99.232%	99.336%
<i>Random Forest</i>	99.268%	99.460%	99.806%	99.633%
<i>Logistic Regression</i>	99.423%	99.436%	100%	99.711%

Na generalidade houve uma melhoria das métricas de avaliação destacando-se o XGBoost e o Logistic Regression que obtiveram uma melhoria significativa.

3.2.2.2.3. Seleção dos recursos aplicando uma redução das dimensões com a técnica PCA (Principal Componente Analysis)

O PCA, denominado como Principal Component Analysis, é um algoritmo de aprendizagem não supervisionado, que gera através de uma transformação estatística, um conjunto de observações não correlacionadas sendo principalmente útil para encontrar padrões fortes num conjunto de dados. O PCA procura identificar a superfície de menor dimensão para projetar os dados de maior dimensionalidade promovendo a utilização dos dados com correlação próxima de zero. Este modelo visa encontrar as direções de máxima variação nos dados com alta dimensionalidade e realizar a projeção dos mesmos num novo subespaço.

Aplicada á base de dados do caso de estudo reduziu-se a dimensão para um total de 28 dimensões conseguindo preservar no mínimo 90% da informação contida recorrendo ao nó “PCA apply”, conforme apresentado na Figura 26.

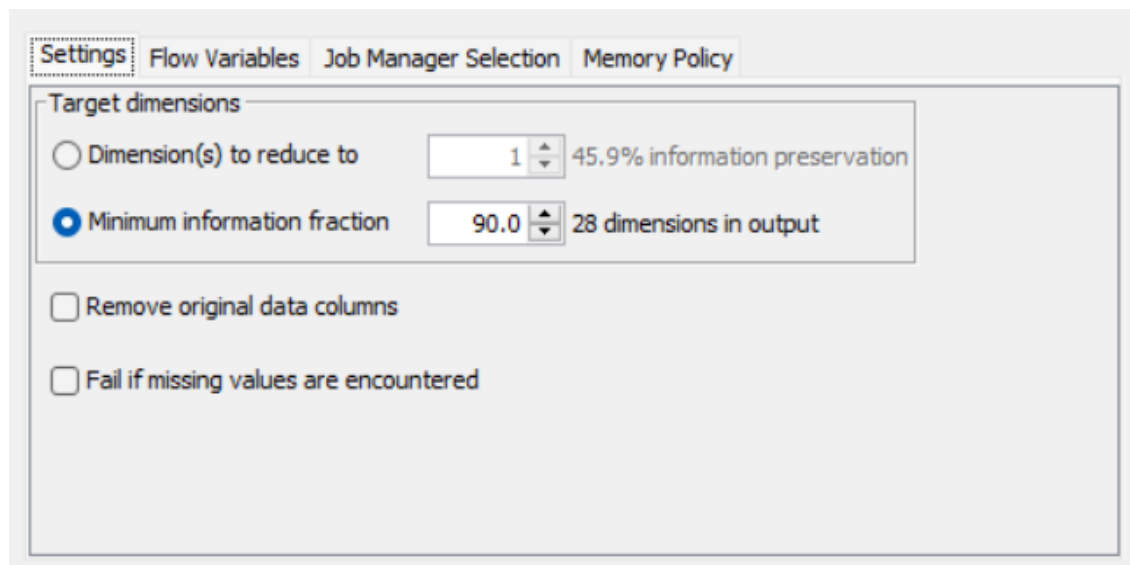


Figura 26- Redução da dimensionalidade usando o PCA (28 dimensões)

Na Tabela 14 estão apresentadas as métricas de avaliação dos diferentes modelos aplicados á base de dados, após a redução das dimensões da base de dados com recurso á técnica do PCA.

Tabela 14- Desempenho das métricas de avaliação dos modelos

PCA redução das dimensões (28 dimensões)				
Métricas de avaliação				
Modelos	Acuraccy	Precision	Recall	F1-score
Decision Trees	99.346%	99.438%	100%	99.672%
XGBoost	99.432%	99.432%	100%	99.715%
Nayve Bayes	99.410%	99.432%	100%	99.704%
Random Forest	99.273%	99.459%	99.812%	99.635%
Logistic Regression	99.432%	99.432%	100%	99.715%

3.3. Discussão de resultados

De uma forma geral todos os modelos desenvolvidos apresentaram resultados satisfatórios uma vez que apresentaram valores superiores a 99%, evidenciando uma redução do tempo de processamento após a aplicação das técnicas de seleção de recursos. No decorrer da aplicação das técnicas de seleção foram otimizados parâmetros dos algoritmos tendo em vista o aperfeiçoamento dos modelos.

Como a base de dados apresenta um desequilíbrio, as métricas de avaliação mais indicadas são o *recall*, *precision* e *F1-Score*, pelo que se se tivesse utilizado a métrica *accuracy* como ponto de referência, poderia ocorrer uma interpretação errada por parte do utilizador. O *F1-Score* permite analisar de forma conjunta as métricas de *recall* e *precision* sendo principalmente útil quando os falsos positivos e falsos negativos possuem impactos diferentes, como é o caso da base de dados utilizada.

Embora antes da aplicação das técnicas de seleção de recursos os modelos desenvolvidos já apresentassem resultados da ordem dos 99% este valor subiu para valores de 100% quando se recorreu á técnica de *Linear Correlation* associada aos algoritmos *Logistic Regression*, *XGBoost* e *Naive Bayes*. Com o objetivo de reduzir a necessidade de poder computacional através do aumento da eficiência dos modelos e da eliminação de recursos irrelevantes, aplicou-se a técnica do *Linear Correlation* evidenciando a não ocorrência de falsos negativos.

A aplicação da técnica *Low Variance* não obteve tão bons resultados como a técnica anterior, evidenciando um valor do *recall* mais baixo mesmo quando comparado com a base de dados sem qualquer técnica de processamento. Apenas o *XGBoost* e *Logistic Regression* atingiram um *recall* de 100%, enquanto que o *Naive Bayes* obteve um *recall* de 99.232%. Por último a técnica PCA utilizada para reduzir as dimensões dos recursos demonstrou elevada eficácia em todos os modelos pois obtiveram *recall* de 100% (exceto o *Random Forest*) e melhoraram o *F1-Score*, constituindo uma técnica importante para a avaliação dos modelos.

O estudo demonstrou que a eficácia dos modelos melhorou com a aplicação das técnicas anteriormente descritas, destacando-se o modelo *Logistic Regression* aliado á técnica de seleção de recursos PCA com um *recall* de 100% e uma *accuracy* de 99.43%, contudo todos os modelos estudados mostraram que seria vantajoso a sua aplicação no processamento desta base de dados.

4. CONCLUSÃO

4.1. Conclusões finais

Com base na revisão bibliográfica realizada conclui-se que a indústria 4.0 promove o aumento da produtividade, da qualidade do produto e da satisfação do cliente, permitindo uma maior personalização do produto. Os modelos de ML apresentam diferentes graus de complexidade e podem ser aplicados a bases de dados distintas, com o propósito de identificar padrões nas variáveis e dessa forma realizar previsões sobre um dado tema. Necessitam de pessoal especializado, uma vez que é necessário conhecimento prévio para utilização destes modelos na previsão eficiente de resultados.

Os modelos de DL tornaram-se mais populares nos últimos anos e apresentam uma estrutura de redes neurais dividida em camadas, ideal para tratamento de bases de dados complexas. Neste tipo de algoritmos permitem obter previsões mais exatas e precisas tendo aplicação em diversos campos como visão artificial para análise de imagens. Os modelos híbridos resultam da combinação dos modelos de ML com os algoritmos metaheurísticos que permitem aumentar a eficácia através da otimização dos parâmetros característicos dos modelos como taxa de aprendizagem e taxa de generalização. Tendo em conta o número de artigos encontrados relacionados com o tema prevê-se o crescimento da adoção destas metodologias nos setores industriais que vão permitir a evolução do sistema produtivo para sistemas autónomos.

As técnicas ML procuram identificar oportunidades para aumentar a eficiência das linhas de produção em qualquer fase do processo, uma vez que identificarão situações em que a manutenção preditiva é necessária. A solução proposta torna possível identificar diferentes padrões na base de dados, tornando possível o reconhecimento de produtos conformes e não conformes na linha de produção. A adoção deste tipo de solução é relevante, uma vez que a ocorrência de um defeito no produto é um desperdício de materiais, características e mão-de-obra.

Em geral, a criação de modelos preditivos para linhas de produção promove a automatização da monitorização, o controlo efetivo da qualidade do produto, a análise dos dados recolhidos reconhecendo padrões, para evitar futuras paragens na linha de produção e, finalmente, uma promoção da sustentabilidade na medida em que se torna possível otimizar a utilização de características essenciais.

A aplicação do modelo de ML através do Knime mostrou-se intuitiva e como uma ferramenta poderosa oferecendo diversas possibilidades de aplicação de algoritmos na base de dados em estudo, contendo a capacidade de otimizar os dados de entrada. A base de dados utilizada no caso de estudo demonstrou um desequilíbrio na classificação binária dos dados, existindo muitos produtos conformes e poucos produtos não-conformes, pelo que dificulta a aplicação do modelo de previsão. Para além disso a base de dados apresentava muitos valores em falta e após o pré processamento minimizou-se esse efeito, considerando apenas os recursos mais relevantes. Esta redução contribuiu para a menor necessidade de poder computacional aumentando a eficácia de execução do modelo, e também para evitar a ocorrência do *overfitting*. Adicionalmente, para evitar a ocorrência do *overfitting* foi também utilizada a validação cruzada no treino dos modelos tentando que seja utilizada toda a informação presente na base de dados para o treino do modelo.

A normalização dos dados permite manter a mesma escala em todas as variáveis de entrada e a aplicação de filtros com base no coeficiente de correlação evitou aparecimento da redundância de valores, sendo esta uma das técnicas de extração de recursos, *Linear Correlation*. As restantes técnicas *Low Variance* e PCA permitiram um aumento do desempenho dos modelos, nomeadamente no indicador *recall* que obteve um desempenho de 100% em quase todos os modelos.

Quanto maior for a seleção dos recursos da base de dados mais fácil e acessível fica para o modelo de previsão ser generalizado, porem a redução do número de recursos não afetou a generalização do modelo, demonstrando que a definição dos seus parâmetros também é preponderante para a definição do modelo.

Da perspetiva da implicação prática deste estudo, é possível melhorar alguns resultados importantes. Em primeiro lugar, é possível criar modelos de produção para a previsão da qualidade do produto, utilizando uma ferramenta de código aberto disponível. Em segundo lugar, são necessárias grandes capacidades de processamento computacional se a base de dados estiver a tornar-se mais complexa. Em terceiro lugar, foram utilizadas técnicas de seleção de recursos para tornar o modelo mais preciso e eficiente e o modelo que obteve a melhor precisão foi o *Logistic Regression* combinado com o PCA. E finalmente, uma empresa fabricante pode evitar perdas económicas e desperdício de recursos ao adotar este tipo de modelos preditivos. O ML demonstrou ser capaz de aumentar a velocidade de processamento de uma grande quantidade de dados.

4.2. Limitações e trabalhos futuros

Numa primeira fase, como trabalho futuro será aplicar a toda a base de dados o modelo construído e não apenas a uma parte desta. Em segunda análise pretende-se analisar a resposta do modelo desenvolvido quando submetido a outras bases de dados. Por último a evolução do modelo deve ser realizada por uma melhoria continua de ajuste a diferentes tipos de pré-processamento e outro tipo de algoritmos.

Uma das formas de divulgação do modelo criado seria a sua publicação numa plataforma online acessível a outros utilizadores para teste.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Z. Kang, C. Catal, and B. Tekinerdogan, "Machine learning applications in production lines: A systematic literature review," *Comput. Ind. Eng.*, vol. 149, 2020, doi: 10.1016/j.cie.2020.106773.
- [2] M. Bertolini, D. Mezzogori, M. Neroni, and F. Zammori, "Machine Learning for industrial applications: A comprehensive literature review," *Expert Syst. Appl.*, vol. 175, p. 114820, Aug. 2021, doi: 10.1016/j.eswa.2021.114820.
- [3] A. Sircar, K. Yadav, K. Rayavarapu, N. Bist, and H. Oza, "Application of machine learning and artificial intelligence in oil and gas industry," *Petroleum Research*. Elsevier, Jun. 04, 2021, doi: 10.1016/j.ptlrs.2021.05.009.
- [4] A. Mayr *et al.*, "Machine learning in production - Potentials, challenges and exemplary applications," in *Procedia CIRP*, Jan. 2020, vol. 86, pp. 49–54, doi: 10.1016/j.procir.2020.01.035.
- [5] V. Liagkou, C. Stylios, L. Pappa, and A. Petunin, "Challenges and opportunities in industry 4.0 for mechatronics, artificial intelligence and cybernetics," *Electron.*, vol. 10, no. 16, p. 2001, Aug. 2021, doi: 10.3390/electronics10162001.
- [6] A. Shrivastava, K. Murali Krishna, M. Lal Rinawa, M. Soni, G. Ramkumar, and S. Jaiswal, "Inclusion of IoT, ML, and Blockchain Technologies in Next Generation Industry 4.0 Environment," *Mater. Today Proc.*, Jul. 2021, doi: 10.1016/j.matpr.2021.07.273.
- [7] S. Agrawal, A. Sahu, and G. Kumar, "A conceptual framework for the implementation of Industry 4.0 in legal informatics," *Sustain. Comput. Informatics Syst.*, vol. 33, p. 100650, Jan. 2022, doi: 10.1016/j.suscom.2021.100650.
- [8] B. K. Gupta and V. Rastogi, "Integration of technology to access the manufacturing plant via remote access system - A part of Industry 4.0," *Mater. Today Proc.*, Nov. 2021, doi: 10.1016/j.matpr.2021.11.135.
- [9] A. Diez-Olivan, J. Del Ser, D. Galar, and B. Sierra, "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0," *Inf. Fusion*, vol. 50, pp. 92–111, Oct. 2019, doi: 10.1016/j.inffus.2018.10.005.
- [10] J. Moosavi, J. Bakhshi, and I. Martek, "The application of industry 4.0 technologies in pandemic management: Literature review and case study," *Healthc. Anal.*, vol. 1, p. 100008, Nov. 2021, doi: 10.1016/j.health.2021.100008.
- [11] A. Aoun, A. Ilinca, M. Ghandour, and H. Ibrahim, "A review of Industry 4.0 characteristics and challenges, with potential improvements using Blockchain technology," *Comput. Ind. Eng.*, vol. 162, p. 107746, Oct. 2021, doi: 10.1016/J.CIE.2021.107746.
- [12] T. Kotsiopoulos, P. Sarigiannidis, D. Ioannidis, and D. Tzovaras, "Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm," *Comput. Sci. Rev.*, vol. 40, p. 100341, May 2021, doi: 10.1016/j.cosrev.2020.100341.
- [13] F. Ansari, S. Erol, and W. Sihn, "Rethinking Human-Machine Learning in Industry 4.0: How Does the Paradigm Shift Treat the Role of Human Learning?," in *Procedia Manufacturing*, Jan. 2018, vol. 23, pp. 117–122, doi: 10.1016/j.promfg.2018.04.003.
- [14] A. Angelopoulos *et al.*, "Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects," *Sensors (Switzerland)*, vol. 20, no. 1, p. 109, Dec. 2020, doi: 10.3390/s20010109.

- [15] M. Sharabov and G. Tsochev, "The Use of Artificial Intelligence in Industry 4.0," *Probl. Eng. Cybern. Robot.*, vol. 73, Nov. 2020, doi: 10.7546/pecr.73.20.02.
- [16] G. Aceto, V. Persico, and A. Pescapé, "Industry 4.0 and Health: Internet of Things, Big Data, and Cloud Computing for Healthcare 4.0," *J. Ind. Inf. Integr.*, vol. 18, p. 100129, Jun. 2020, doi: 10.1016/j.jii.2020.100129.
- [17] C. Zhang and Y. Lu, "Study on artificial intelligence: The state of the art and future prospects," *J. Ind. Inf. Integr.*, vol. 23, p. 100224, Sep. 2021, doi: 10.1016/j.jii.2021.100224.
- [18] S. P. Leo Kumar, "State of The Art-Intense Review on Artificial Intelligence Systems Application in Process Planning and Manufacturing," *Eng. Appl. Artif. Intell.*, vol. 65, pp. 294–329, Oct. 2017, doi: 10.1016/j.engappai.2017.08.005.
- [19] A. F. S. Borges, F. J. B. Laurindo, M. M. Spínola, R. F. Gonçalves, and C. A. Mattos, "The strategic use of artificial intelligence in the digital era: Systematic literature review and future research directions," *International Journal of Information Management*, vol. 57. Pergamon, p. 102225, Apr. 01, 2021, doi: 10.1016/j.ijinfomgt.2020.102225.
- [20] S. O. Abioye *et al.*, "Artificial intelligence in the construction industry: A review of present status, opportunities and future challenges," *J. Build. Eng.*, vol. 44, p. 103299, 2021, doi: 10.1016/j.job.2021.103299.
- [21] A. Zabin, V. A. González, Y. Zou, and R. Amor, "Applications of machine learning to BIM: A systematic literature review," *Adv. Eng. Informatics*, vol. 51, p. 101474, Jan. 2022, doi: 10.1016/j.aei.2021.101474.
- [22] T. D. Akinosho *et al.*, "Deep learning in the construction industry: A review of present status and future innovations," *J. Build. Eng.*, vol. 32, p. 101827, Nov. 2020, doi: 10.1016/j.job.2020.101827.
- [23] F. Psarommatis, J. Sousa, J. P. Mendonça, and D. Kiritsis, "Zero-defect manufacturing the approach for higher manufacturing sustainability in the era of industry 4.0: a position paper," *Int. J. Prod. Res.*, vol. 60, no. 1, pp. 73–91, 2022, doi: 10.1080/00207543.2021.1987551.
- [24] F. Psarommatis, G. May, P. A. Dreyfus, and D. Kiritsis, "Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research," *Int. J. Prod. Res.*, vol. 58, no. 1, pp. 1–17, Jan. 2020, doi: 10.1080/00207543.2019.1605228.
- [25] F. Psarommatis, S. Prouvost, G. May, and D. Kiritsis, "Product Quality Improvement Policies in Industry 4.0: Characteristics, Enabling Factors, Barriers, and Evolution Toward Zero Defect Manufacturing," *Front. Comput. Sci.*, vol. 2, Aug. 2020, doi: 10.3389/FCOMP.2020.00026/FULL.
- [26] D. A. C. Narciso and F. G. Martins, "Application of machine learning tools for energy efficiency in industry: A review," *Energy Reports*, vol. 6. Elsevier, pp. 1181–1199, Nov. 01, 2020, doi: 10.1016/j.egy.2020.04.035.
- [27] M. Batta, "Machine Learning Algorithms - A Review," *Int. J. Sci. Res. (IJ)*, vol. 9, no. 1, pp. 381-undefined, 2020, Accessed: Jan. 07, 2022. [Online]. Available: www.ijcsit.com.
- [28] M. Farhoumandi, Q. Zhou, and M. Shahidehpour, "A review of machine learning applications in IoT-integrated modern power systems," *Electr. J.*, vol. 34, no. 1, p. 106879, Jan. 2021, doi: 10.1016/j.tej.2020.106879.
- [29] I. Lee and Y. J. Shin, "Machine learning for enterprises: Applications, algorithm selection, and challenges," *Bus. Horiz.*, vol. 63, no. 2, pp. 157–170, Mar. 2020, doi: 10.1016/j.bushor.2019.10.005.

- [30] A. Kumbhar, P. G. Dhawale, S. Kumbhar, U. Patil, and P. Magdum, "A comprehensive review: Machine learning and its application in integrated power system," *Energy Reports*, vol. 7, pp. 5467–5474, Nov. 2021, doi: 10.1016/j.egy.2021.08.133.
- [31] A. L. Tan-Torres, J. P. Brooks, B. Singh, and S. Seashols-Williams, "Machine learning clustering and classification of human microbiome source body sites," *Forensic Sci. Int.*, vol. 328, p. 111008, Nov. 2021, doi: 10.1016/j.forsciint.2021.111008.
- [32] A. Ponmalar and V. Dhanakoti, "An intrusion detection approach using ensemble Support Vector Machine based Chaos Game Optimization algorithm in big data platform," *Appl. Soft Comput.*, p. 108295, Dec. 2021, doi: 10.1016/j.asoc.2021.108295.
- [33] N. Arora and P. D. Kaur, "A Bolasso based consistent feature selection enabled random forest classification algorithm: An application to credit risk assessment," *Appl. Soft Comput. J.*, vol. 86, p. 105936, Jan. 2020, doi: 10.1016/j.asoc.2019.105936.
- [34] K. S. M. H. Ibrahim, Y. F. Huang, A. N. Ahmed, C. H. Koo, and A. El-Shafie, "A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting," *Alexandria Eng. J.*, vol. 61, no. 1, pp. 279–303, Jan. 2021, doi: 10.1016/j.aej.2021.04.100.
- [35] Z. Ge, Z. Song, S. X. Ding, and B. Huang, "Data Mining and Analytics in the Process Industry: The Role of Machine Learning," *IEEE Access*, vol. 5, pp. 20590–20616, Sep. 2017, doi: 10.1109/ACCESS.2017.2756872.
- [36] B. Mihaljević, C. Bielza, and P. Larrañaga, "Bayesian networks for interpretable machine learning and optimization," *Neurocomputing*, vol. 456, pp. 648–665, Oct. 2021, doi: 10.1016/j.neucom.2021.01.138.
- [37] Z. Zhao *et al.*, "Visual inspection system for battery screen print using joint method with multi-level block matching and K nearest neighbor algorithm," *Optik (Stuttg.)*, vol. 250, p. 168332, Jan. 2022, doi: 10.1016/j.ijleo.2021.168332.
- [38] H. Chen, S. Liu, R. M. Magomedov, and A. A. Davidyants, "Optimization of inflow performance relationship curves for an oil reservoir by genetic algorithm coupled with artificial neural-intelligence networks," *Energy Reports*, vol. 7, pp. 3116–3124, Nov. 2021, doi: 10.1016/j.egy.2021.05.028.
- [39] A. M. G. Júnior, V. V. R. Silva, L. M. R. Baccarini, and L. F. S. Mendes, "The design of multiple linear regression models using a genetic algorithm to diagnose initial short-circuit faults in 3-phase induction motors," *Appl. Soft Comput. J.*, vol. 63, pp. 50–58, Feb. 2018, doi: 10.1016/j.asoc.2017.11.015.
- [40] F. E. B. Otero, A. A. Freitas, and C. G. Johnson, "Inducing decision trees with an ant colony optimization algorithm," *Appl. Soft Comput. J.*, vol. 12, no. 11, pp. 3615–3626, Nov. 2012, doi: 10.1016/j.asoc.2012.05.028.
- [41] Y. Fan *et al.*, "Privacy preserving based logistic regression on big data," *J. Netw. Comput. Appl.*, vol. 171, p. 102769, Dec. 2020, doi: 10.1016/j.jnca.2020.102769.
- [42] L. Zhu, X. Zhou, and C. Zhang, "Rapid identification of high-quality marine shale gas reservoirs based on the oversampling method and random forest algorithm," *Artif. Intell. Geosci.*, vol. 2, pp. 76–81, Dec. 2021, doi: 10.1016/j.aiig.2021.12.001.
- [43] Y. Li, X. Chu, D. Tian, J. Feng, and W. Mu, "Customer segmentation using K-means clustering and the adaptive particle swarm optimization algorithm," *Appl. Soft Comput.*, vol. 113, p. 107924, Dec. 2021, doi: 10.1016/j.asoc.2021.107924.
- [44] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach

- to policy search,” in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 2011, pp. 465–472.
- [45] J. S. Park and J. H. Park, “Enhanced Machine Learning Algorithms: Deep Learning, Reinforcement Learning, and Q-Learning,” *J. Inf. Process. Syst.*, vol. 16, no. 5, pp. 1001–1007, Oct. 2020, doi: 10.3745/JIPS.02.0139.
- [46] T. Jiang, J. L. Gradus, and A. J. Rosellini, “Supervised Machine Learning: A Brief Primer,” *Behav. Ther.*, vol. 51, no. 5, pp. 675–687, Sep. 2020, doi: 10.1016/j.beth.2020.05.002.
- [47] B. B. Benuwa, Y. Zhan, B. Ghansah, D. K. Wornyo, and F. B. Kataka, “A review of deep machine learning,” *Int. J. Eng. Res. Africa*, vol. 24, pp. 124–136, 2016, doi: 10.4028/www.scientific.net/JERA.24.124.
- [48] C. Yang, H. Lan, F. Gao, and F. Gao, “Review of deep learning for photoacoustic imaging,” *Photoacoustics*, vol. 21, p. 100215, Mar. 2021, doi: 10.1016/j.pacs.2020.100215.
- [49] J. M. Górriz *et al.*, “Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications,” *Neurocomputing*, vol. 410, pp. 237–270, Oct. 2020, doi: 10.1016/j.neucom.2020.05.078.
- [50] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, “A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning,” *Arch. Comput. Methods Eng.*, vol. 27, no. 4, pp. 1071–1092, Sep. 2020, doi: 10.1007/s11831-019-09344-w.
- [51] Y. R. Shrestha, V. Krishna, and G. von Krogh, “Augmenting Organizational Decision-Making with Deep Learning Algorithms: Principles, Promises, and Challenges,” *SSRN Electron. J.*, Sep. 2020, doi: 10.2139/ssrn.3701592.
- [52] I. Lee and Y. J. Shin, “Machine learning for enterprises: Applications, algorithm selection, and challenges,” *Bus. Horiz.*, vol. 63, no. 2, pp. 157–170, Mar. 2020, doi: 10.1016/J.BUSHOR.2019.10.005.
- [53] F. Miao, L. Yao, and X. Zhao, “Evolving convolutional neural networks by symbiotic organisms search algorithm for image classification,” *Appl. Soft Comput.*, vol. 109, p. 107537, Sep. 2021, doi: 10.1016/j.asoc.2021.107537.
- [54] R. SaiSindhuTheja and G. K. Shyam, “An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment,” *Appl. Soft Comput.*, vol. 100, p. 106997, Mar. 2021, doi: 10.1016/j.asoc.2020.106997.
- [55] S. B. Devamane and R. L. Itagi, “Recurrent neural network based turbo decoding algorithms for different code rates,” *J. King Saud Univ. - Comput. Inf. Sci.*, Apr. 2020, doi: 10.1016/j.jksuci.2020.03.012.
- [56] O. Krause, A. Fischer, and C. Igel, “Algorithms for estimating the partition function of restricted Boltzmann machines,” *Artif. Intell.*, vol. 278, p. 103195, Jan. 2020, doi: 10.1016/j.artint.2019.103195.
- [57] C. Y. Zhang, C. L. P. Chen, D. Chen, and K. T. NG, “MapReduce based distributed learning algorithm for Restricted Boltzmann Machine,” *Neurocomputing*, vol. 198, pp. 4–11, Jul. 2016, doi: 10.1016/j.neucom.2015.09.129.
- [58] M. A. Zamora-Hernández, J. A. Castro-Vargas, J. Azorin-Lopez, and J. Garcia-Rodriguez, “Deep learning-based visual control assistant for assembly in Industry 4.0,” *Comput. Ind.*, vol. 131, p. 103485, Oct. 2021, doi: 10.1016/j.compind.2021.103485.
- [59] X. J. Wu, M. Da Xu, C. Di Li, C. Ju, Q. Zhao, and S. X. Liu, “Research on image reconstruction algorithms based on autoencoder neural network of Restricted Boltzmann Machine

- (RBM),” *Flow Meas. Instrum.*, vol. 80, p. 102009, Aug. 2021, doi: 10.1016/j.flowmeasinst.2021.102009.
- [60] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, “A review of deep learning for renewable energy forecasting,” *Energy Convers. Manag.*, vol. 198, p. 111799, Oct. 2019, doi: 10.1016/j.enconman.2019.111799.
- [61] A. Goli and H. K. Zare, “A comprehensive model of demand prediction based on hybrid artificial intelligence and metaheuristic algorithms: A case study in dairy industry,” *J. Ind. Syst. Eng.*, vol. 11, no. 4, pp. 190–203, Dec. 2018, Accessed: Jan. 06, 2022. [Online]. Available: http://www.jise.ir/article_76524.html.
- [62] P. Agrawal, H. F. Abutarboush, T. Ganesh, and A. W. Mohamed, “Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019),” *IEEE Access*, vol. 9, pp. 26766–26791, 2021, doi: 10.1109/ACCESS.2021.3056407.
- [63] S. Nematzadeh, F. Kiani, M. Torkamanian-Afshar, and N. Aydin, “Tuning Hyperparameters of Machine Learning Algorithms and Deep Neural Networks Using Metaheuristics: A Bioinformatics Study on Biomedical and Biological Cases,” *Comput. Biol. Chem.*, vol. 97, p. 107619, Apr. 2021, doi: 10.1016/j.compbiolchem.2021.107619.
- [64] A. Jaafari, E. K. Zenner, M. Panahi, and H. Shahabi, “Hybrid artificial intelligence models based on a neuro-fuzzy system and metaheuristic optimization algorithms for spatial prediction of wildfire probability,” *Agric. For. Meteorol.*, vol. 266–267, pp. 198–207, Mar. 2019, doi: 10.1016/j.agrformet.2018.12.015.
- [65] J. D. Mello-Roman and A. Hernandez, “KPLS optimization with nature-inspired metaheuristic algorithms,” *IEEE Access*, vol. 8, pp. 157482–157492, 2020, doi: 10.1109/ACCESS.2020.3019771.
- [66] A. C. Sousa, A. F. Bertachini, C. Cunha, R. Chaves, and M. L. R. Varela, “Literature review and discussion on collaborative decision making approaches in Industry 4.0,” *FME Trans.*, vol. 49, no. 4, pp. 817–826, 2021, doi: 10.5937/FME2104817S.
- [67] S. Nadizadeh Shorabeh, N. Neysani Samani, F. Minaei, H. Karimi Firozjaei, M. Homaei, and A. Darvishi Bolorani, “Developing a Decision Model Based on Decision Tree and Particle Swarm Optimization Algorithms to Identify Optimal Locations for Solar Power Plants Construction in Iran,” *SSRN Electron. J.*, Jan. 2021, doi: 10.2139/ssrn.3900348.
- [68] M. Misaghi and M. Yaghoobi, “Improved invasive weed optimization algorithm (IWO) based on chaos theory for optimal design of PID controller,” *J. Comput. Des. Eng.*, vol. 6, no. 3, pp. 284–295, Jul. 2019, doi: 10.1016/j.jcde.2019.01.001.
- [69] M. Louzazni, A. Khouya, K. Amechnoue, A. Gandelli, M. Mussetta, and A. Craciunescu, “Metaheuristic algorithm for photovoltaic parameters: Comparative study and prediction with a Firefly algorithm,” *Appl. Sci.*, vol. 8, no. 3, Feb. 2018, doi: 10.3390/app8030339.
- [70] A. R. Behesht Abad *et al.*, “Predicting oil flow rate through orifice plate with robust machine learning algorithms,” *Flow Meas. Instrum.*, vol. 81, p. 102047, Oct. 2021, doi: 10.1016/j.flowmeasinst.2021.102047.
- [71] L. M. Elshenawy, M. A. Halawa, T. A. Mahmoud, H. A. Awad, and M. I. Abdo, “Unsupervised machine learning techniques for fault detection and diagnosis in nuclear power plants,” *Prog. Nucl. Energy*, vol. 142, p. 103990, Dec. 2021, doi: 10.1016/j.pnucene.2021.103990.
- [72] M. Brillinger, M. Wuwer, M. Abdul Hadi, and F. Haas, “Energy prediction for CNC machining with machine learning,” *CIRP J. Manuf. Sci. Technol.*, vol. 35, pp. 715–723, Nov. 2021, doi: 10.1016/j.cirpj.2021.07.014.

- [73] F. Bodendorf and J. Franke, "A machine learning approach to estimate product costs in the early product design phase: A use case from the automotive industry," in *Procedia CIRP*, Jan. 2021, vol. 100, pp. 643–648, doi: 10.1016/j.procir.2021.05.137.
- [74] G. M. de Almeida, G. T. Pereira, A. S. R. de S. Bahia, K. Fernandes, and J. Marques Júnior, "Machine learning in the prediction of sugarcane production environments," *Comput. Electron. Agric.*, vol. 190, p. 106452, Nov. 2021, doi: 10.1016/j.compag.2021.106452.
- [75] S. A. A. Shah *et al.*, "Automatic and Fast Classification of Barley Grains from Images: A Deep Learning Approach," *Smart Agric. Technol.*, p. 100036, Jan. 2022, doi: 10.1016/j.atech.2022.100036.
- [76] E. Isleyen, S. Duzgun, and R. McKell Carter, "Interpretable deep learning for roof fall hazard detection in underground mines," *J. Rock Mech. Geotech. Eng.*, vol. 13, no. 6, pp. 1246–1255, Dec. 2021, doi: 10.1016/j.jrmge.2021.09.005.
- [77] T. Beznik, P. Smyth, G. de Lannoy, and J. A. Lee, "Deep learning to detect bacterial colonies for the production of vaccines," *Neurocomputing*, vol. 470, pp. 427–431, Jan. 2022, doi: 10.1016/j.neucom.2021.04.130.
- [78] M. Elsis, M. Tran, K. Mahmoud, D.-E. A. Mansour, M. Lehtonen, and M. M. F. Darwish, "Effective IoT-based Deep Learning Platform for Online Fault Diagnosis of Power Transformers Against Cyberattack and Data Uncertainties," *Measurement*, vol. 190, p. 110686, Feb. 2022, doi: 10.1016/j.measurement.2021.110686.
- [79] Y. Tikhamarine, D. Souag-Gamane, A. Najah Ahmed, O. Kisi, and A. El-Shafie, "Improving artificial intelligence models accuracy for monthly streamflow forecasting using grey Wolf optimization (GWO) algorithm," *J. Hydrol.*, vol. 582, p. 124435, Mar. 2020, doi: 10.1016/j.jhydrol.2019.124435.
- [80] C. Boga and T. Koroglu, "Proper estimation of surface roughness using hybrid intelligence based on artificial neural network and genetic algorithm," *J. Manuf. Process.*, vol. 70, pp. 560–569, Oct. 2021, doi: 10.1016/j.jmapro.2021.08.062.
- [81] B. Li, H. Yan, J. Zhang, and N. Zhou, "Compaction property prediction of mixed gangue backfill materials using hybrid intelligence models: A new approach," *Constr. Build. Mater.*, vol. 247, p. 118633, Jun. 2020, doi: 10.1016/j.conbuildmat.2020.118633.
- [82] M. E. A. Ben Seghier, B. Kechtegar, M. Nait Amar, J. A.F.O. Correia, and N. T. Trung, "Simulation of the ultimate conditions of fibre-reinforced polymer confined concrete using hybrid intelligence models," *Eng. Fail. Anal.*, vol. 128, p. 105605, Oct. 2021, doi: 10.1016/j.engfailanal.2021.105605.
- [83] Z. M. Yaseen, Z. H. Ali, S. Q. Salih, and N. Al-Ansari, "Prediction of risk delay in construction projects using a hybrid artificial intelligence model," *Sustain.*, vol. 12, no. 4, Feb. 2020, doi: 10.3390/su12041514.
- [84] A. Rosenkranz, M. Marian, F. J. Profito, N. Aragon, and R. Shah, "The use of artificial intelligence in tribology—a perspective," *Lubricants*, vol. 9, no. 1, pp. 1–11, 2021, doi: 10.3390/lubricants9010002.

